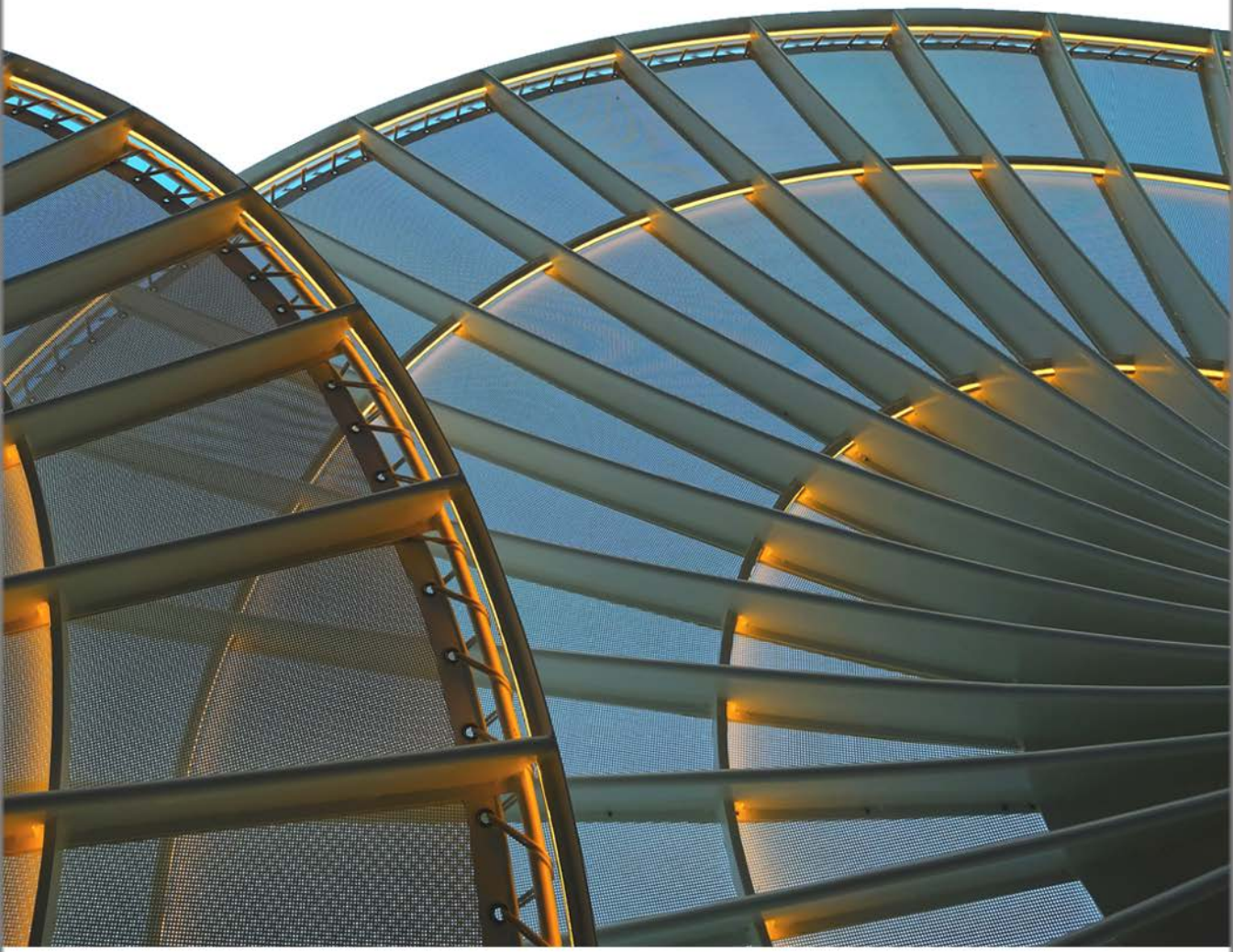
EViews® 14

Getting Started



S&P Global

EViews 14 Getting Started

S&P Global

EViews 14 Getting Started

Disclaimer

The authors and S&P Global Inc. assume no responsibility for any errors that may appear in this manual or the EViews program. The user assumes all responsibility for the selection of the program to achieve intended results, and for the installation, use, and results obtained from the program.

Trademarks

EViews® is a registered trademark of S&P Global Inc. Windows, Excel, PowerPoint, and Access are registered trademarks of Microsoft Corporation. PostScript is a trademark of Adobe Corporation. Bloomberg is a trademark of Bloomberg Finance L.P. All other product names mentioned in this manual may be trademarks or registered trademarks of their respective companies.

Third Party Licenses

This section contains third party notices or additional terms and conditions applicable to certain software technologies which may be used in one or more EViews products and/or services. Please be sure to consult the individual product files, about box, and/or install or manual documentation for specific copyright notices and author attributions. Notices on this page are current for EViews products released on or after October 1, 2017.

- diff template Library - Copyright © 2015 Tatsuhiko Kubo cubicdaiya@gmail.com. All rights reserved.
- FFmpeg Library - FFmpeg is a trademark of Fabrice Bellard, originator of the FFmpeg project.
- GZipHelper - Copyright © 1995-2002 Gao Dasheng dsgao@hotmail.com.
- Java SE Runtime Environment v1.8.0_401 licensed under Oracle Binary Code License Agreement.
- JDemetra+ v3 licensed under European Union Public License v1.2.
- jsonCPP Library - Copyright © 2007-2010 Baptiste Lepilleur and The JsonCPP Authors.
- openssl Library - Copyright © 1998-2016 The OpenSSL Project. All rights reserved.

S&P Global Inc.
Telephone: (949) 856-3368
e-mail: sales@eviews.com
web: www.eviews.com
June 24, 2024

Table of Contents

Getting Started

Congratulations on your purchase of EViews 14, the premier forecasting and analysis package for Windows-based computers. This guide will lead you step-by-step through the installation and registration procedure for EViews.

(The following discussion describes the installation and registration process for single user copies of EViews and seat licenses purchased under a Volume License Program. Setting up machines to use concurrent use licenses will require a different procedure; for details, please check with your IT support department.)

Installing EViews

Installing the Program

To begin installation, simply click on the "EViews14Installer.exe" executable program file.

- First, you will be prompted to read and accept the License Agreement, and to designate a directory into which you wish to install your copy of EViews. If you wish to change the default installation directory, click on **Browse** and navigate to the desired directory. Click on **Next** to continue.
- Next, you will be asked to enter a name and serial number. You should have been provided with a 24-character serial number as part of your purchase. Those of you who have obtained your copy of EViews as part of a Volume License agreement should obtain a serial number from your license administrator. Enter the serial number and your name as you wish it to appear in your copy of EViews, and click on **Next**.
- Select the components you wish to install and click on **Next**.
- Lastly, you will be asked about setting up a Start Menu folder containing shortcuts to the EViews example files folder and the EViews program executable. Clicking on **Next** starts the actual installation of files onto your computer.

You should note that as part of the installation procedure, EViews will prompt you to register files with the extensions ".WF1", ".WF2", ".PRG", ".EDB", ".AIPZ", and ".UIPZ".

If these extensions are already registered, possibly by an earlier version of EViews, you will be prompted to allow EViews 14 to override the existing registration. Registering the extensions is not required, but doing so will allow you to double-click on files with these extensions to launch EViews.

Once the installation procedure is completed, click on **Finish**. If you have elected to create an EViews shortcut, the EViews Start Menu folder will open. To launch EViews, double-click the EViews 14 icon. Subsequently, you may launch EViews using the shortcut on your desktop or by selecting EViews from the Start Menu shortcuts, if present, by double-clicking on EViews registered file types, or by navigating to the EViews installation directory and double-clicking on the EViews icon.

Registering EViews

What is Registration?

To use EViews 14 on a specific computer, you must first register the program using a serial number. EViews registration is the one-time process of assigning a serial number to a specific machine, sending a unique machine ID number to S&P Global Inc., and writing some information to your Windows registry or Mac application support directory. This is a simple process that can be performed in a few seconds.

Under the terms of the EViews Volume License agreement, "13C" (volume) license serial numbers may not be used to register multiple machines. Each volume licensed machine running EViews must be assigned a distinct serial number. Thus, licensing an office computer, home computer and laptop computer of a single user will require three distinct Volume License serial numbers.

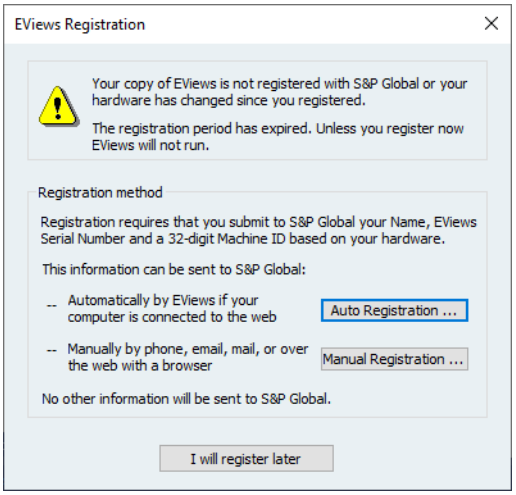
The copy of EViews may be uninstalled and reinstalled on a registered machine, updated, or moved to a different directory without re-registering the copy for that machine. In the special case where a machine's hard disk is wiped clean, but no other changes are made to the system, you may simply re-register your copy of EViews. Note that in this circumstance, reregistration on the machine will *not* count as an additional registration.

If an entire machine or a machine's hard disk is replaced, you should contact our office to unregister your previous installation prior to re-registering.

How Do I Register?

Before starting the registration process, you should first locate your EViews serial number. You most likely will need to enter this number into EViews during the registration procedure.

If the copy of EViews is not registered, EViews will display a warning dialog. The dialog will inform you that EViews is not registered for this machine and, *if applicable,* will indicate the number of additional days the unregistered copy will continue to run.

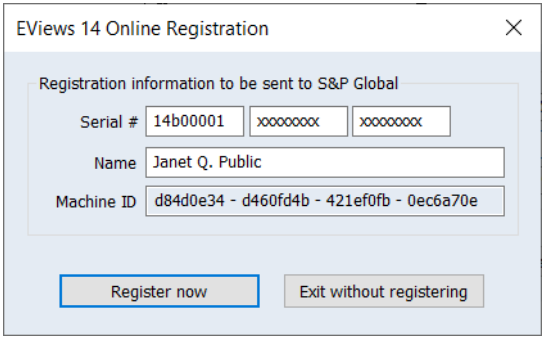


On a Windows machine, if the copy of EViews is not registered, EViews will display a warning dialog. The dialog will inform you that EViews is not registered for this machine and, *if applicable,* will indicate the number of additional days the unregistered copy will continue to run.

You may choose to register in one of two ways: you may use the EViews auto registration features (by clicking on **Auto Registration...**), or you can manually register (by clicking on **Manual Registration...**). Selecting either of the these two options will open a dialog prompting you for additional information.

Auto Registration

If your computer is connected to the Internet, auto registration makes registering EViews a snap. Simply click on the **Auto Registration...** button to display a dialog for entering your registration information.



EViews will fill out as many fields in this dialog as possible. If you wish to continue with the auto registration process, make sure that the entries in the **Serial #** and **Name** fields are filled in with the relevant information. When you click on the **Register now** button, EViews will attempt to contact one of our registration servers and, if successful, will transmit the

information contained in the dialog to the server. The server will process the information and the machine will be registered to run EViews.

You should see a message indicating that registration was completed successfully, along with the number of machines that have been registered to the serial number.

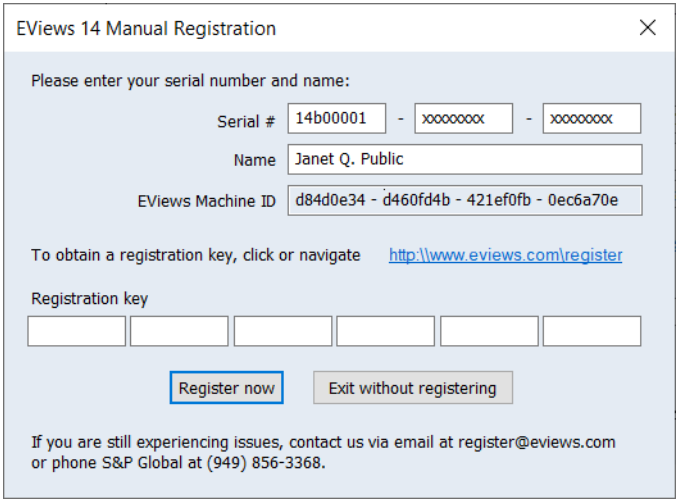
If you do not wish to continue with auto registration, click on the **Exit without registering** button and you will be returned to the main registration screen.

Note that there are some circumstances in which auto registration will fail. Obviously, auto registration will not work if the computer is not connected to the Internet. If registration fails, you should first verify that you have Internet access. Second, your computer may be behind a firewall which does not allow the required communication between your computer and our servers. Furthermore, while unlikely, it is possible that all of our registration servers are temporarily unresponsive.

If you continue to have problems with auto registration, you can choose to register manually as described in the next section, or you can contact us for assistance.

Manual Registration

If auto registration fails or if you prefer not to use the automatic registration features, you may elect to register manually. From the main registration page, click on **Manual Registration...** to display the manual registration portion of the dialog:



You must fill in the three fields in the dialog: the 24-character serial number, your name, and a 36-character registration key you must first obtain via online, by phone, or by email. EViews will help you by filling in as many fields as possible.

The easiest method of retrieving the registration key is online. Go to

http://www.eviews.com/register/

which will direct you to our registration servers. Follow the links to the registration page, and fill in the form. Enter your name, serial number, and the machine ID number as displayed in this registration dialog into the form. Click on the **Submit the form** button. You will be provided with the 36-character registration key.

Once you have obtained the key, enter it into the registration dialog. If necessary, select Help/EViews Registration... from the EViews main menu to display the registration page.

Make certain that the listed name, serial number, and registration key in the registration dialog matches exactly (this includes spaces and case) that of the registration confirmation page. Click **OK** to finish the registration process. Note that you should be able to copy-and-paste the registration key information from your browser into the dialog edit fields..

If all of the information is entered correctly, you will be informed that your registration is complete.

If you do not have access to a working web browser, you can contact our office via email or phone to obtain the key:

S&P Global Inc.
Email: register@eviews.com
Phone: 949-856-3368

Please provide a registration name, full 24-character serial number, and the machine ID number. We will then provide you with the 36-character registration key.

If you receive the key via email, you should be able to copy-and-paste the key information into the dialog edit fields.

Contact Information

Once registration is completed, EViews will display an optional contact page form. You may submit this form to send name, address, phone number, and email information to S&P Global Inc. This information is for our records only and will not be redistributed to others.

Frequently Asked Questions about Registration

While the registration procedure should be straightforward, we understand that you may still have questions. The following are answers to the most frequently asked questions:

- *How do I find my serial number and other information about my copy of EViews?*

  Your copy of EViews contains information about your registration status, as well as the product version and build date of the program. To obtain this information, simply select **Help/About EViews** from the main EViews menu.

- *I contacted you and received a key, but the key doesn't seem to work. What could be wrong?*

  The most common registration problem results from entering a name or serial number which does not match the key. You should make certain that the name and serial number both match those provided when obtaining a key. Note that while the name is not case-sensitive, it should otherwise be entered *exactly* as originally provided. If you still experience problems, please contact our office.

- *My copy of EViews does not appear to have the features for the edition that I purchased. Do I need to purchase a new copy?*

  No. Simply contact our office. Once we verify the edition of EViews that you have purchased, you should be able to re-register and upgrade your copy to enable the features.

- *I've replaced my computers and no longer have available registrations. What should I do?*

  If there are special circumstances where you need to register an additional machine, please contact our office.

- *How do I change the name in which my copy is registered?*

  Your copy of EViews contains the name in which it was first registered. If you wish to change the registration name, please contact our office.

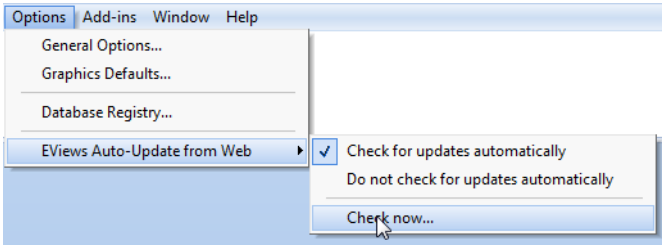
- *What if I have trouble registering?*

  We do not anticipate that you will have problems registering your copy of EViews using one of the available methods (auto-registration, manual using our web servers, or manual using email or phone). Please feel free to contact our office if you encounter difficulties.

Updating Your Copy of EViews

EViews 14 offers an automatic updating feature that can check for new updates every day, and install an updated version if available. (The automatic update feature can be enabled or disabled from the **Options/EViews Auto-Update from Web** menu item.)

Alternately, you may manually check for updates from within EViews at any time by selecting **Check now...** under the **EViews Auto-Update from Web** menu item, or by selecting **EViews Update** from the **Help** menu.



You may also visit the EViews website to check for updates to the EViews program and other components (documentation, sample data, and sample programs). Go to:

http://www.eviews.com

and navigate to the downloads area. Downloading updates *will not* require re-registration of EViews on any previously registered computer. Simply download the update, run the installer, and you will have the latest shipping copy of your software.

Where to Go For Help

Your EViews installation includes documentation in the form of an interactive Help System and PDF versions of the manuals. In addition, online documentation and user-provided support are available.

Online Help

You may also access the EViews documentation online at

https://help.eviews.com

The Help System

All of the EViews documentation may be viewed from within EViews using the help system. To access the EViews help system, go to the main menu and select **Help/EViews Help Topics...** or click on **Help/Quick Help Reference** and select a topic to jump directly to relevant subsections.

The EViews Manuals (PDF Files)

Your EViews installation includes copies of the EViews manuals in Adobe Portable Document Format (.PDF) file format. You may access the PDF files from within EViews by clicking on **Help** in the main EViews menu and selecting the file of interest. Alternately, you may navigate to the "Docs" subdirectory of your EViews installation directory to access the files directly.

Tutorials

To get you started, we have provided a set of PowerPoint tutorials illustrating the basics of EViews. These tutorials are a great way to see EViews in action.

http://www.eviews.com/Learning/index.html

The EViews Forum

User-provided online support is available via the EViews Forum.

To supplement the information provided in the manuals and the help system, we encourage you to visit the EViews Online Forum, where you can find answers to common questions about installing, using, and getting the most out of EViews. The EViews Forum is an ideal place to ask questions of and share information with other EViews users.

The forum address is:

https://forums.eviews.com

New Features in EViews 14

EViews 14 features a number of exciting changes and improvements. The following is an overview of the most important new features in Version 14.

Note that in some cases, entries will appear in more than one section as they might otherwise be overlooked.

Data Handling

Data Sources and File Formats

- StatCan SDMX database connectivity ("Statistics Canada (StatCan SDMX) Database Connectivity" on page 10).
- Updated User-interface for NOAA databases ("NOAA (National Oceanic And Atmospheric Administration)" on page 13).
- Improved User-interface for EuroStat, ECB, OECD and other SDMX databases ("Statistical Data and Metadata eXchange (SDMX) Databases" on page 24).

Seasonal Adjustment

- JDemetra+ Seasonal Adjustment ("JDemetra+ Seasonal Adjustment" on page 32).

Filtering and Smoothing

- Boosted Hodrick-Prescott Filter ("Boosted Hodrick-Prescott Filter" on page 41).

Expanded Functions

- Sample index and boolean vector functions ("Sample Index Functions" on page 44).
- Matrix utility functions ("Matrix Utility Functions" on page 45).
- String and date function vector support ("String and Date Vector Functions" on page 46).

Econometrics and Statistics

Estimation and Analysis

- Mixed Data Sampling (MIDAS) GARCH ("MIDAS GARCH" on page 47).
- Quantile ARDL Estimation ("Quantile ARDL Estimation" on page 48).
- Enhanced Elastic NET ("Enhanced Elastic Net and Lasso," on page 50).
- Improved Lasso Variable Selection ("Lasso Variable Selection" on page 55).

- ARDL Heteroskedasticity and Auto-correlation Consistent (HAC) standard errors ("Expanded ARDL HAC Standard Errors" on page 59).

Forecasting

- Facebook Prophet Forecasting ("Facebook Prophet Forecasting" on page 62).

Testing and Diagnostics

- Series-based Outlier Detection ("Outlier Detection" on page 478).
- Equation-based Outlier Detection ("Equation-based Outlier Detection" on page 68).
- Tests for Series Breaks and Change Point Detection ("Break Testing and Change Point Detection" on page 70).
- Tests for Series Trends ("Trend Testing" on page 72).
- Tests for Explosive Financial Bubbles ("Explosive Bubble Testing" on page 73).
- Major additions, extensions, and improvements in impulse response analysis of VARs, VECs, Variance Decomposition, and BTVCVARs ("Impulse Response Analysis" on page 75).

Matrix Statistical Tools

- New statistical tools for matrices and vectors ("New Matrix Object Views and Procs" on page 86).

Models

- Solve models with future endogenous variables ("Models Containing Future Endogenous Values" on page 91).

Command Language

- List of new and updated object commands and data members ("Updated Object List" on page 96).
- List of new and updated functions ("Updated Function List" on page 101).

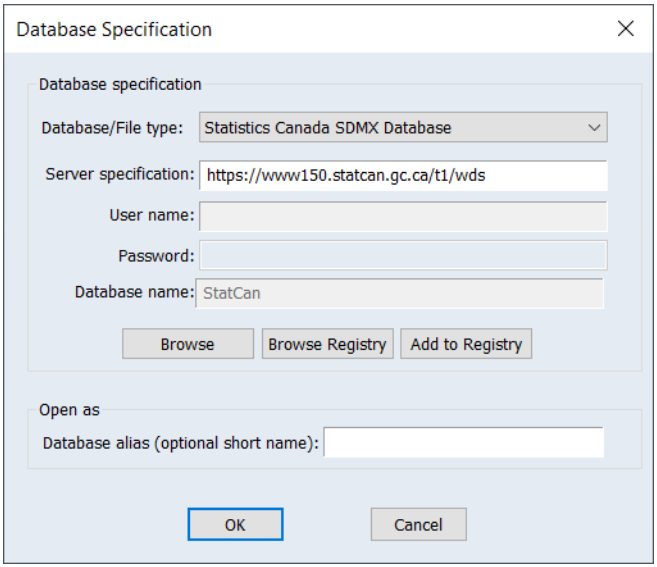
Data Handling

Statistics Canada (StatCan SDMX) Database Connectivity

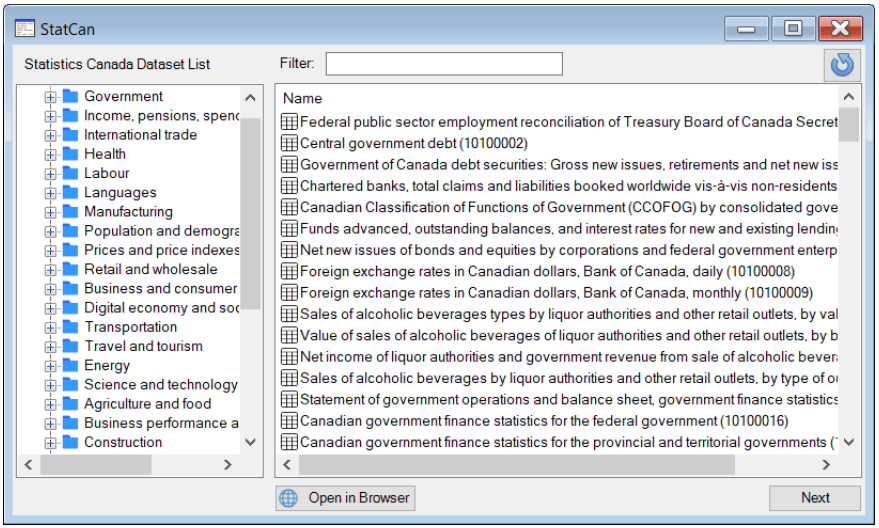
EViews 14 offers easy-to-use connectivity to online databases from Statistics Canada, the Canadian national statistical office through EViews' enhanced SDMX interface ("Statistical Data and Metadata eXchange (SDMX) Databases" on page 24).

StatCan SDMX is not available in EViews Standard Edition.

To begin, open the database windows by selecting **File/Open Database** from the primary EViews menu, select **Statistics Canada SDMX Database** from the **Database/File type** drop down menu,

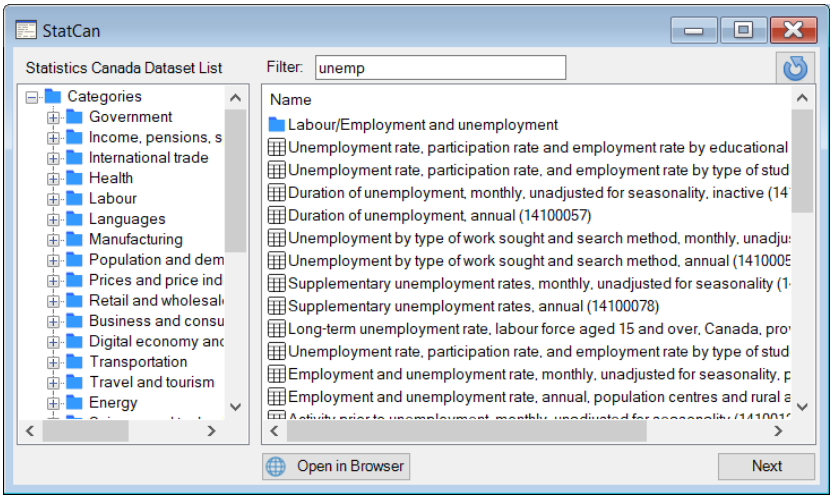


Click on **OK** to proceed and open the data browser



The left-hand side of the data browser offers tree interface through which may navigate until you locate the desired dataset.

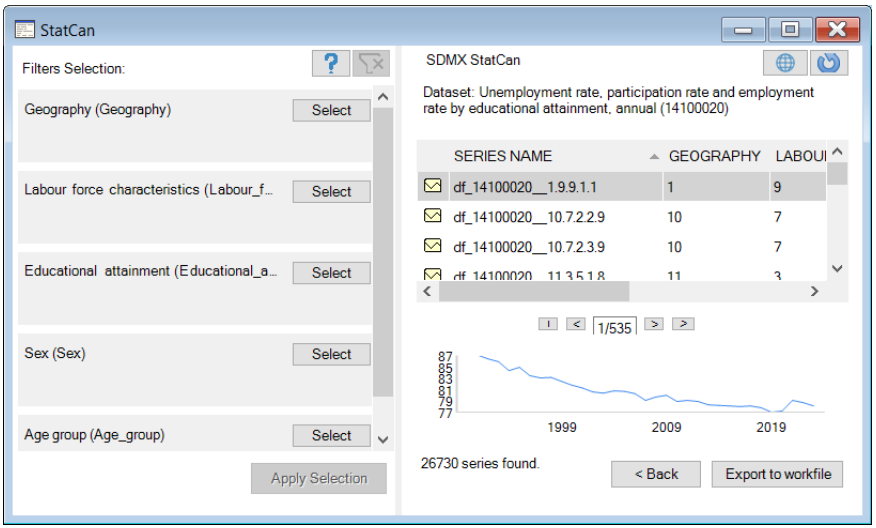
Alternately, you may use the **Filter** edit box to input a keyword, enabling a search across all categories and datasets.



Once you locate the desired dataset, you can either double-click on the dataset in the list on the right section or select it and then click **Next** to display the filter page.

Note that the StatCan SDMX databases offers a data browser on their official website that allows you to search for a dataset. Click on the **Open in Browser** button, which will open a web browser window showing the official database website.

To enhance the performance of the data explorer interface, categorization information is cached when the interface is first loaded. While a remote possibility, the database information may be updated during your EViews session. To refresh the cache, you may click on circular arrow **Refresh** button in the upper-right hand portion of the dialog.



You may use the filter page to refine your search. See "Selecting Series" on page 29 for additional detail. Once you have the data series desired, you may click on **Export to workfile** to download the data into an EViews workfile.

- See "Foreign Format Databases" on page 362 in *User's Guide I*.
- See also `dbopen` (p. 428) in the *Command and Programming Reference*.

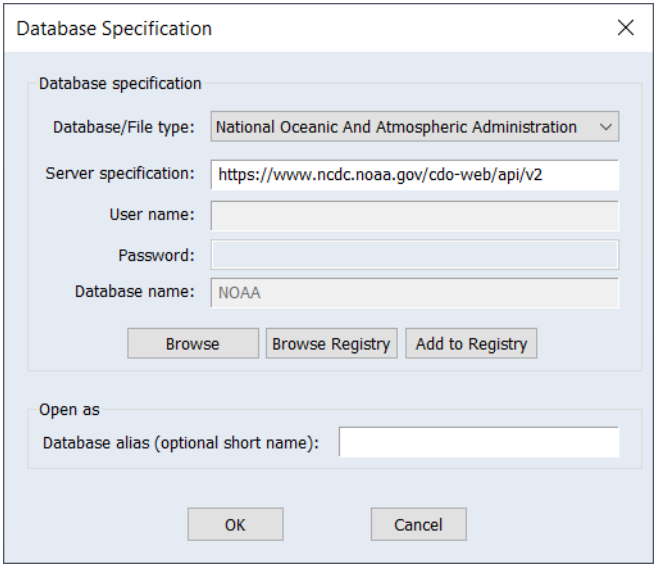
NOAA (National Oceanic And Atmospheric Administration)

The National Oceanic and Atmospheric Administration (NOAA) offers an extensive range of publicly available weather and climate datasets. For further details on NOAA data offerings, refer to:

https://www.ncdc.noaa.gov/cdo-web/webservices/v2#datasets

EViews 14 offers an all new interface that enables users to interact with EViews while simultaneously searching for data through the EDX dialog. This new NOAA database dialog includes a user-friendly table where all selected entries for import are recorded, facilitating the direct importation into a workfile. This process significantly enhances user workflow and efficiency, providing a seamless experience for data retrieval and integration into EViews.

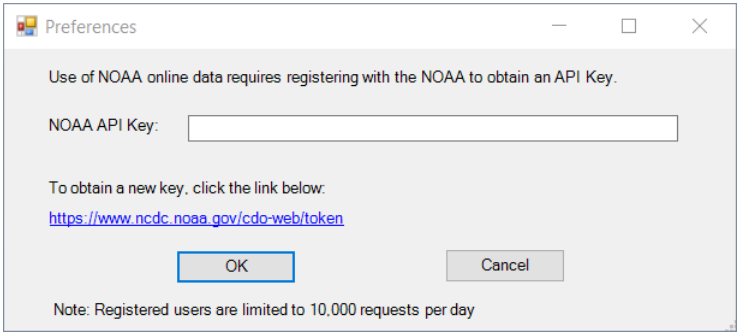
To access the NOAA database, choose **File/Open Database** from the main EViews menu, and then select **National Oceanic and Atmospheric Administration** from the **Database/File type** drop down menu.



When opening the NOAA database for the first time, you'll be asked to enter an API Key obtained from the National Oceanic and Atmospheric Administration.

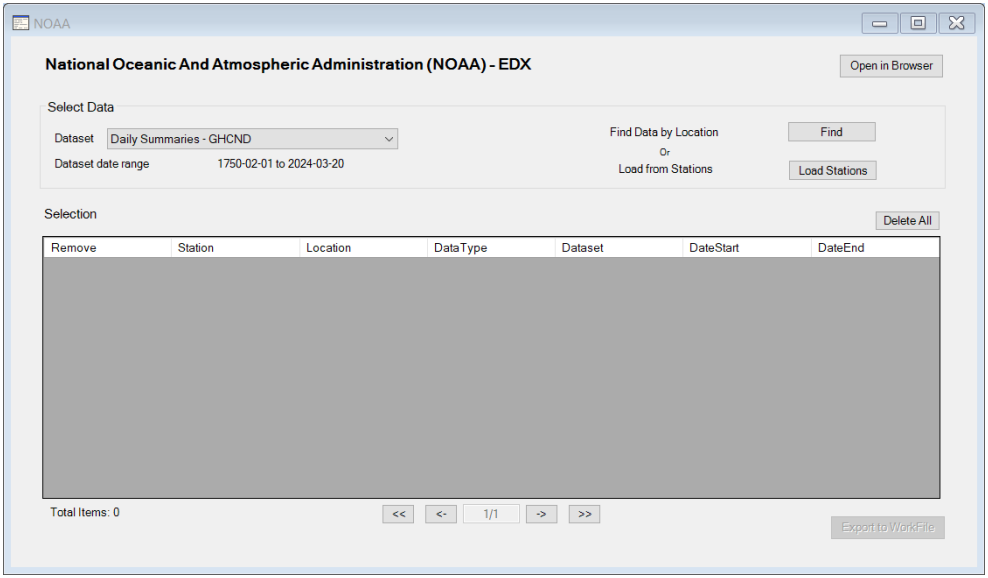
To request an API Key visit: https://www.ncdc.noaa.gov/cdo-web/token.

Enter your API key and click **OK**. The key will be saved as a user specific setting in your EViews ".ini" file. If you need to change the key later, select **View/Preferences** from the EViews database menu to modify your settings.



When you click on **OK**, EViews will open a standard database window.

Click on **Browse** to open the custom NOAA window:



This custom interface will allow us to drill down through dataset, location, data type, and station selection to specify the data of interest.

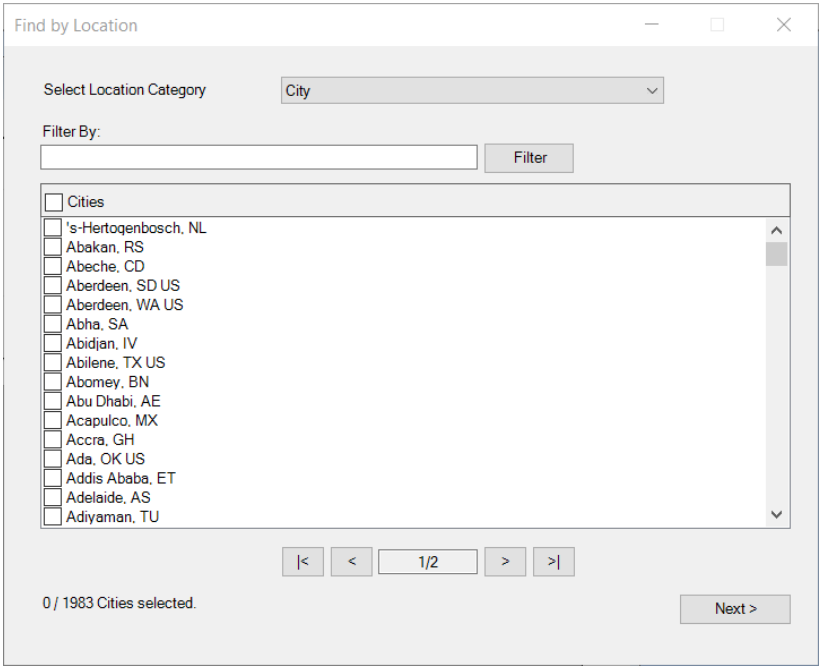
When you first display this dialog, the **Selection** area will be empty.

To select the data, first specify a dataset using the **Dataset drop** down menu.

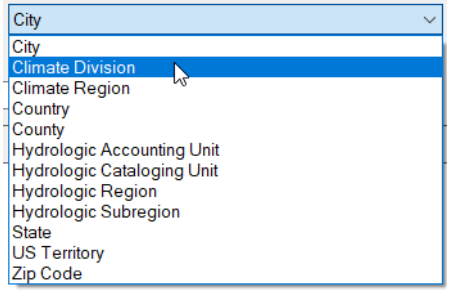
You may then choose to **Find Data by Location** by clicking on the **Find** button to select one or more locations from a list, or you elect to **Load from Stations** by clicking on the **Load Stations** button to specify stations using Station IDs.

Find Data By Location

If you choose click on the **Find** button to **Find Data by Location**, a dialog will be displayed to walk you through the selection process:



The drop down menu at the top should be used to specify a **Location Category** to browse:

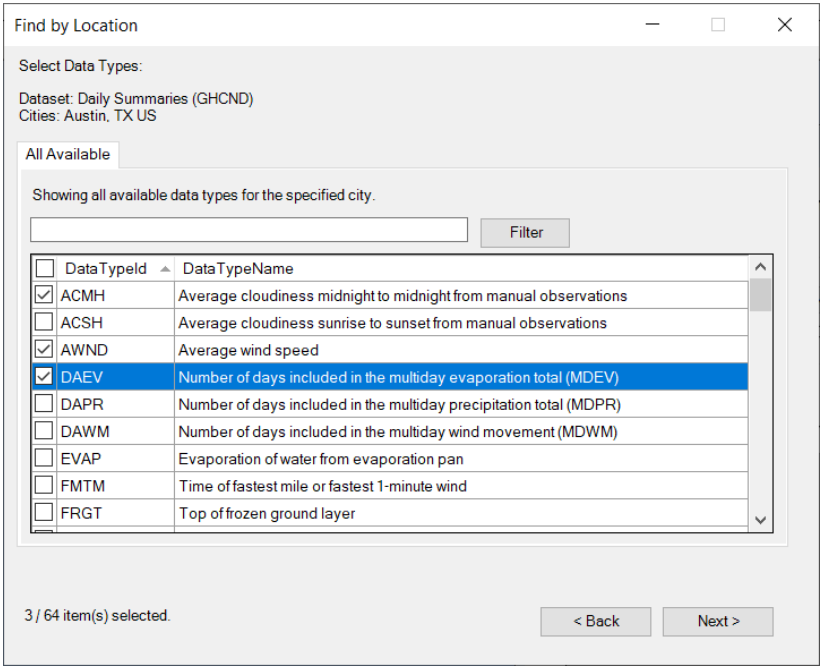


You may choose from a variety of location categories including, for example, **City**, **Climate Region**, **County**, **Hydrologic Region**, and **Zip Code**.

You can filter the displayed list of locations using a text string by entering the text in the **Filter by** edit field, and clicking on the **Filter** button.

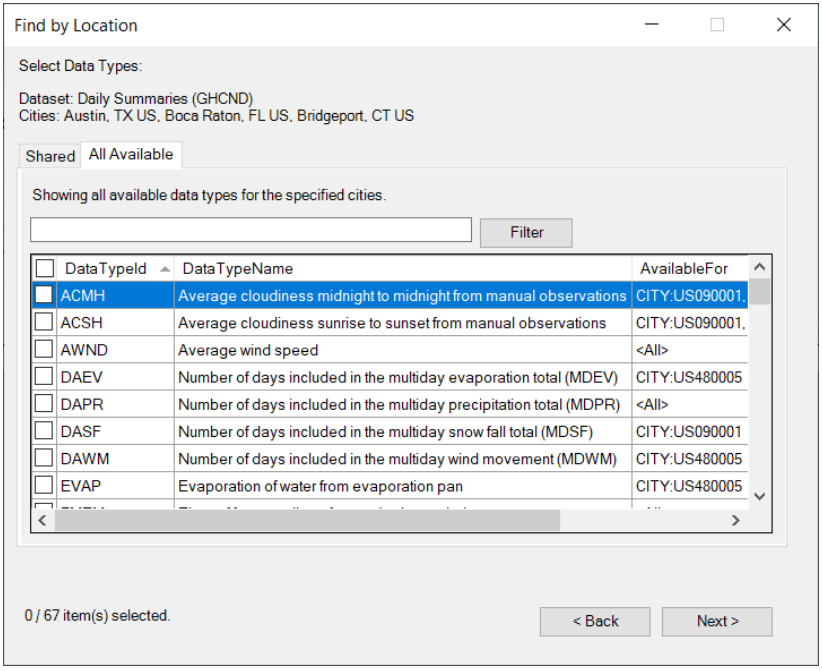
Select the desired locations from the list and click **Next** to continue.

If you select a single location, you will see a dialog prompting you to select specific data types for the selected location:



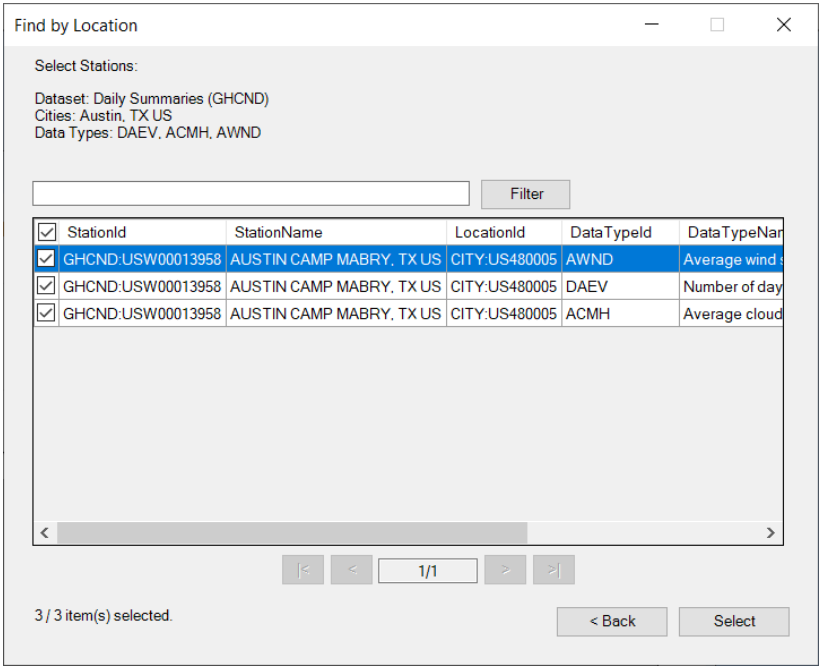
You may use the filter to pare the list if desired. Click to select and deselect the entries.

If you selected multiple locations in the location step, a EViews will display a dialog with two tabs: a **Shared** tab showing data types common to all of the locations, and an **All Available** tab listing all available data types across all of the locations.

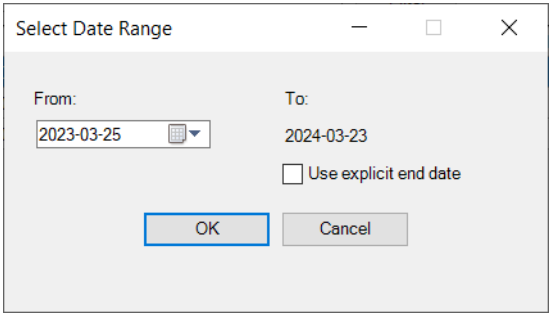


Use the **Filter** edit field and button to refine your search in the lists on both tabs, select the data types of interest, and click **Next** to view the available NOAA stations.

In the next dialog, you will find the available stations that match the selected data types and locations.

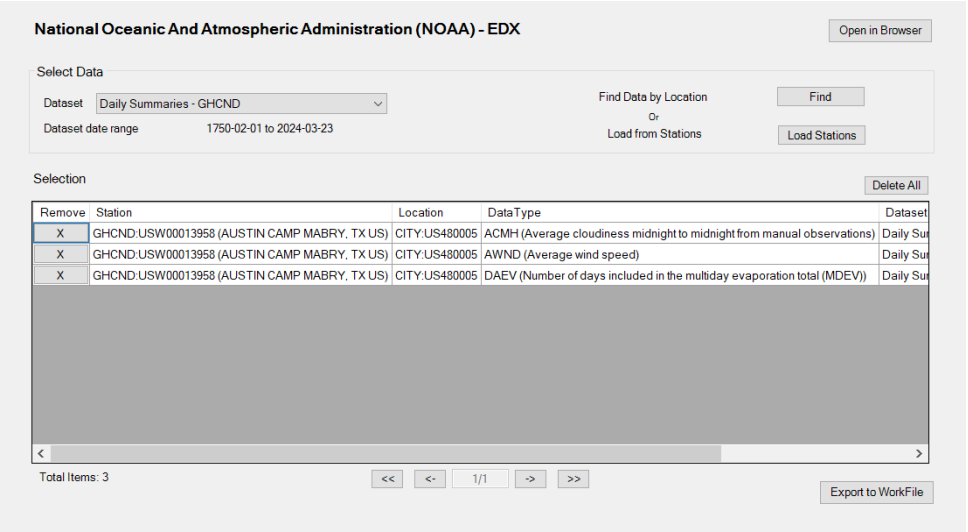


Choose the desired stations from which you want get data and the click **Select**.



You will be asked to select a date range in order to limit the data results. NOAA restricts requests for annual and monthly data to a ten-year range; all other frequencies are restricted to a one-year range.

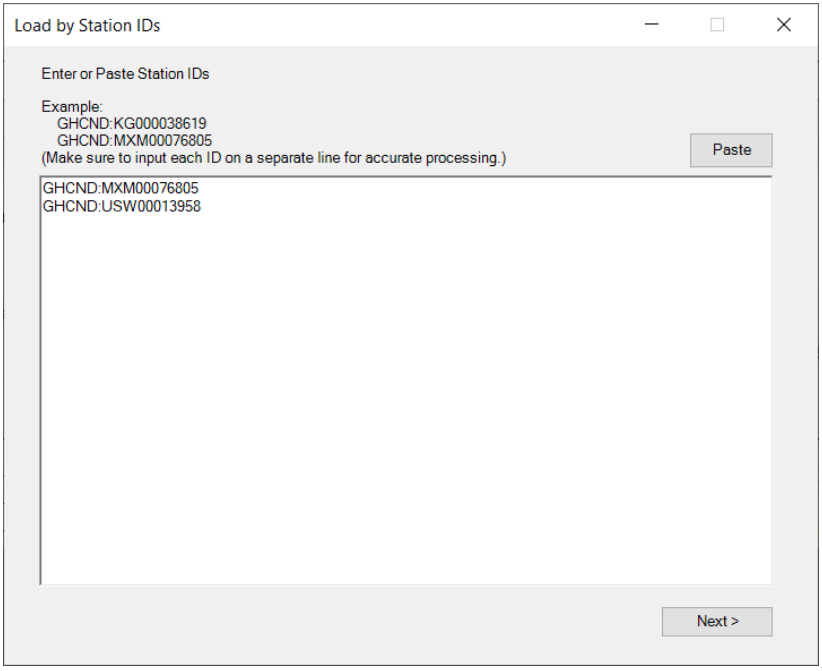
Click on **OK** to continue and display the main NOAA dialog:



You will see in the **Selection** table the data you have just selected from your search. You can directly export these data to a workfile or find additional data before proceeding with the export.

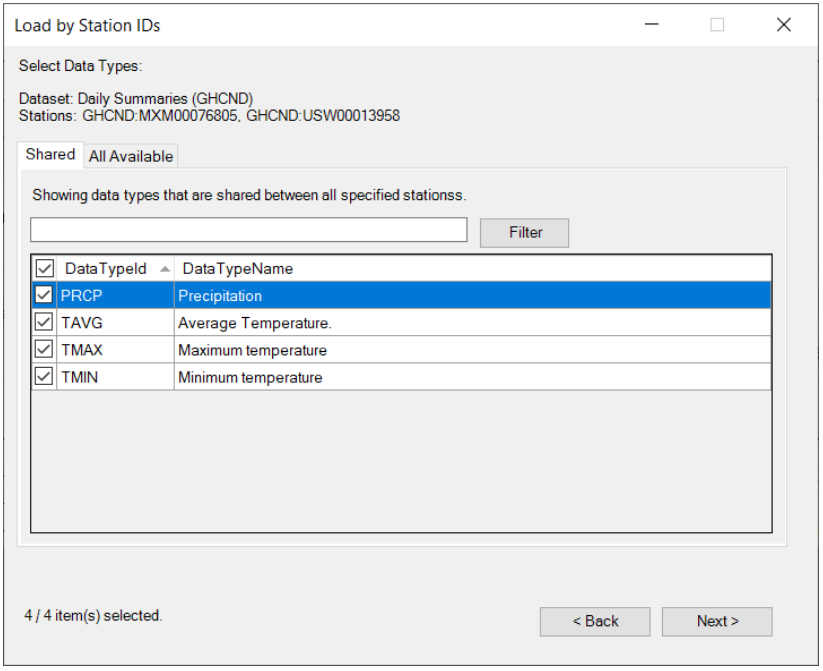
Locations From Stations

If you select **Load Stations**, instead of **Find by Location**, the displayed dialog will contain a textbox where you can manually type or paste from the clipboard the station IDs from which you want to retrieve data:

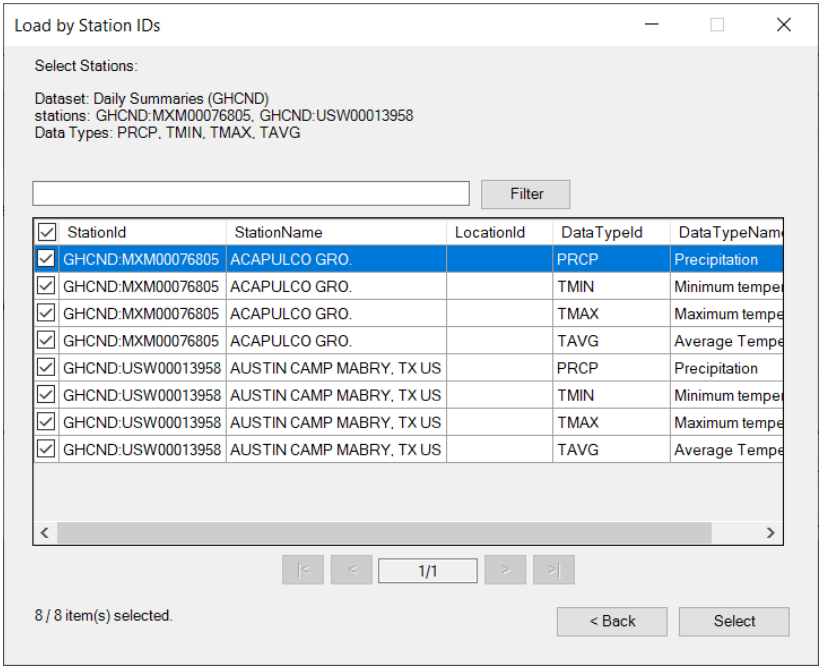


Click on **Next** to continue.

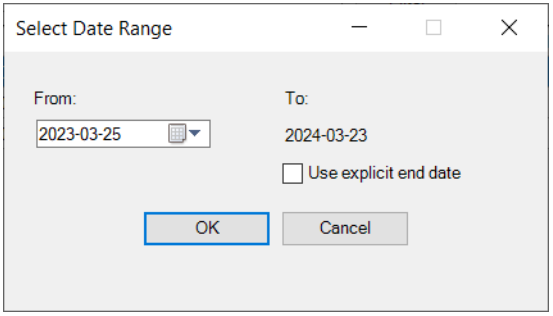
The next dialog will prompt you to select the data types. Once again, the dialog will feature one tab if there is a single location, or two tabs, a **Shared** tab for data types common to all of the locations, and an **All Available** tab for available data types across all of the locations.



Select the desired data types and click **Next** to display the next dialog where you will to select station and data type combinations:

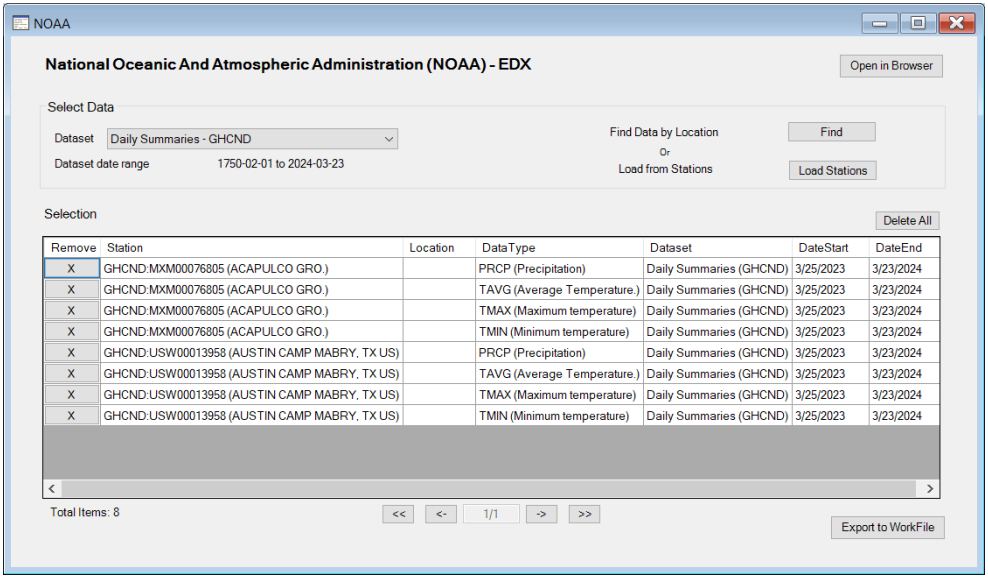


Click on **Select** to continue to the final dialog where you will choose the date range:

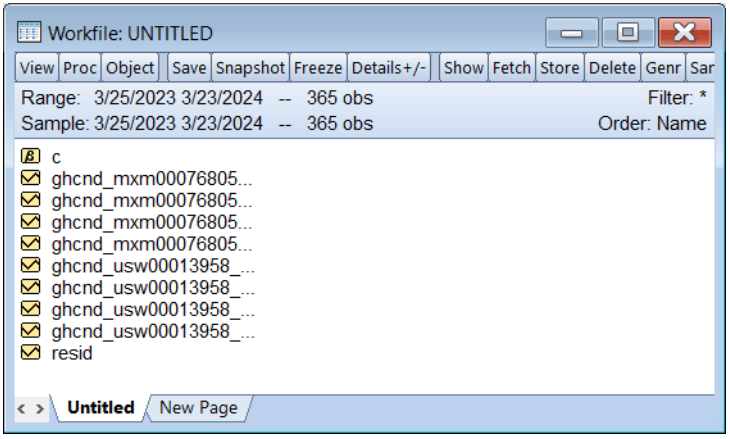


As before, note that NOAA restricts requests for annual and monthly data to a ten-year range; all other frequencies are restricted to a one-year range.

Click on **OK** to continue and display the main NOAA dialog:



Click the **Export to WorkFile** button to export the chosen data to either an existing or a new workfile.



- See "Foreign Format Databases" on page 362 in *User's Guide I*.
- See also `dbopen` (p. 428) in the *Command and Programming Reference*.

Statistical Data and Metadata eXchange (SDMX) Databases

EViews 14 offers a new, graphical user interface (GUI) designed for navigation and retrieval of data from SDMX databases.

The GUI browser for SDMX has undergone improvements to enhance user interaction. The introduction of category-based search functionality allows users to select datasets by their topics of interest, enabling more efficient searches across categories.

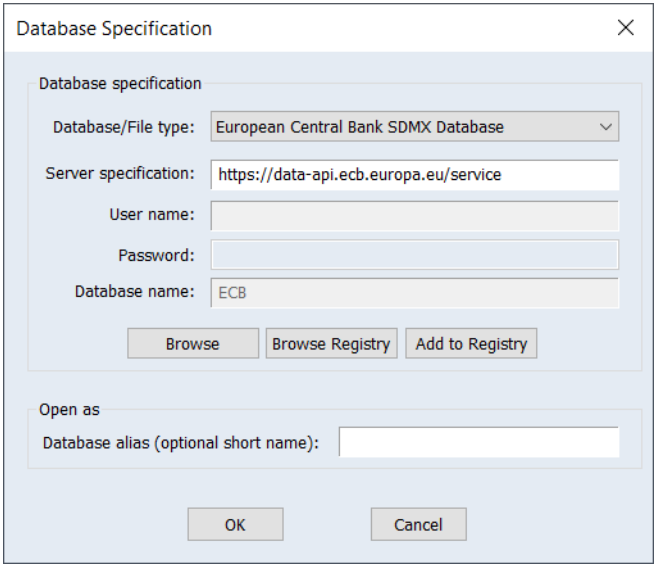
Additionally, several enhancements have been made to the series visualization window, including an improved filter search that enables users to select filters before initiating a search, significantly enhancing the speed of the process compared to previous versions. Moreover, efficiency improvements in data loading have been implemented, resulting in more responsive data loading.

SDMX Databases contain a large range of publicly available data. EViews offers direct access to the following online SDMX databases:

- Eurostat (http://ec.europa.eu/eurostat/data/database)
- ECB (European Central Bank) (http://ec.europa.eu/eurostat/data/database)
- UN (United Nations) (http://data.un.org/WS/)
- IMF (International Monetary Fund)(http://sdmxcentral.imf.org)
- OECD (Organisation for Economic Cooperation and Development) (https://data-explorer.oecd.org/)
- ABS (Australian Bureau of Statistics) (http://abs.gov.au)
- Deutsche Bundesbank (http://api.statistiken.bundesbank.de)
- Insee (L'Institut national de la statistique et des études économiques) (http://www.insee.fr/en/information/286805)
- StatCan (Statistics Canada) using SDMX Web Services (https://www150.statcan.gc.ca/n1/en/type/data)

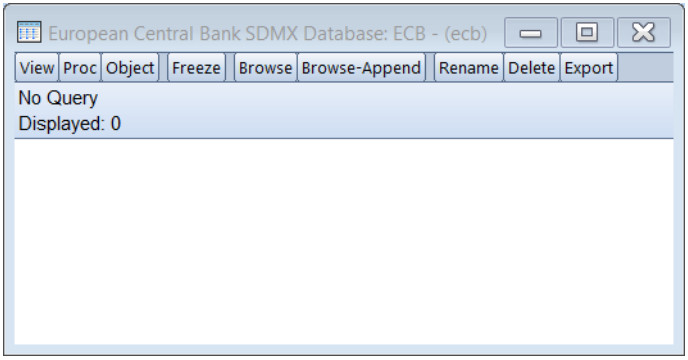
Please note that an internet connection is required to obtain SDMX online data. For more information on the datasets, please see the links above.

To begin, open the database windows by selecting **File/Open Database** from the primary EViews menu, select the desired SDMX database from the **Database/File type** drop down menu,



and click on **OK** to continue.

EViews displays the standard database dialog, indicating that you have an active connection to the data.



Click on **Browse** or **Browse-Append** to open the database.

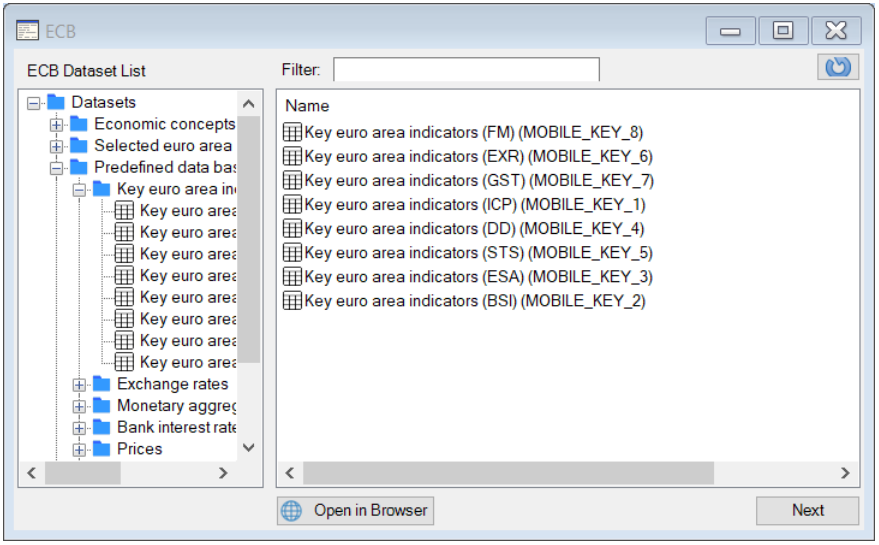
Selecting a Database

EViews offers one of two different interfaces for selecting a SDMX database: a tree structure navigator and filter for databases that offer dataset categorization information, and a simple scroll and filter interface for databases without categorization information.

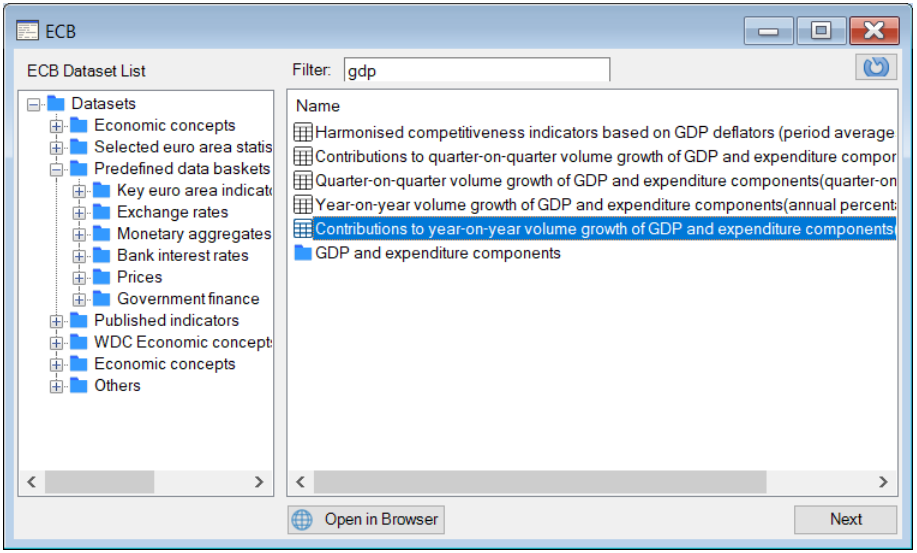
Categorized Datasets

If the SDMX database offers categorized datasets, the dialog interface offers features that facilitate dataset exploration through a nested tree structure.

The tree interface makes it easy to select or search for a specific category and then navigate through the nested tree until you locate the desired dataset. Simply use the tree structure on the left-hand pane to drill down to the desired subcategory, clicking on the node indicators as needed to inspect the contents. You may also double click on folders displayed in the right-hand pane to show its contents.



Alternatively, you can use the **Filter** edit box to input a keyword, enabling a search across all categories and datasets.



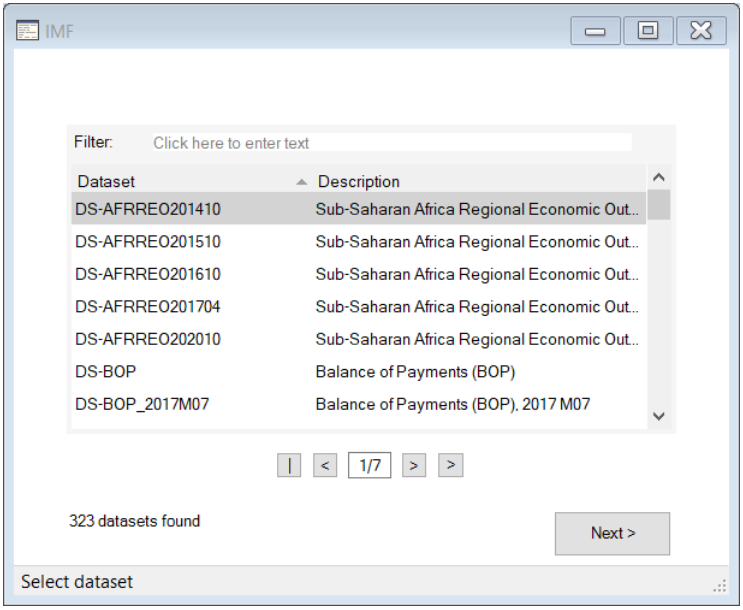
Once you locate the desired dataset, you can either double-click on the dataset in the list on the right section or select it and then click **Next**.

Some SDMX databases offer a data browser on their official website that allows you to perform more advanced search. Click on the **Open in Browser** button, which will open a web browser window showing the official database website.

To enhance the performance of the data explorer interface, categorization information is cached when the SDMX interface is first loaded. While a remote possibility, the database information may be updated during your EViews session. To refresh the cache, you may click on circular arrow **Refresh** button in the upper-right hand portion of the dialog.

Non-Categorized Datasets

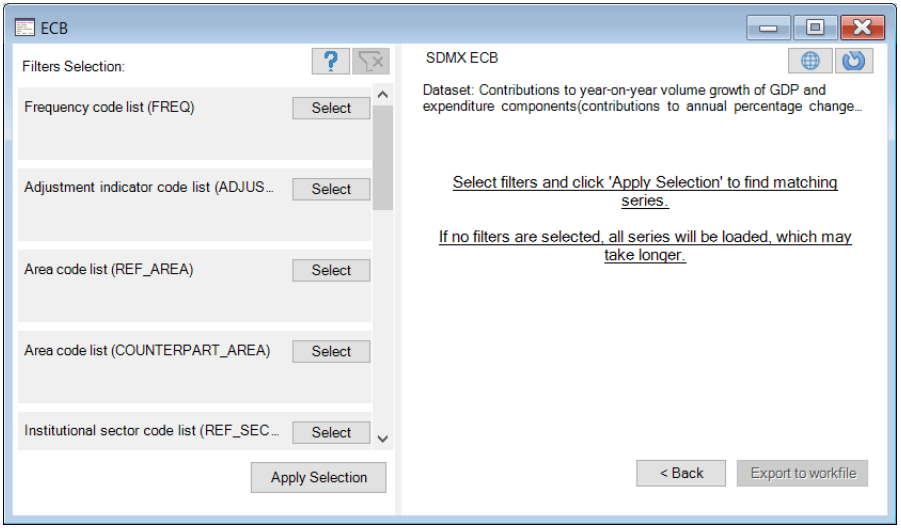
Alternatively, for SDMX databases that do not offer categorization information, the dialog interface simply offers a list of the available databases:



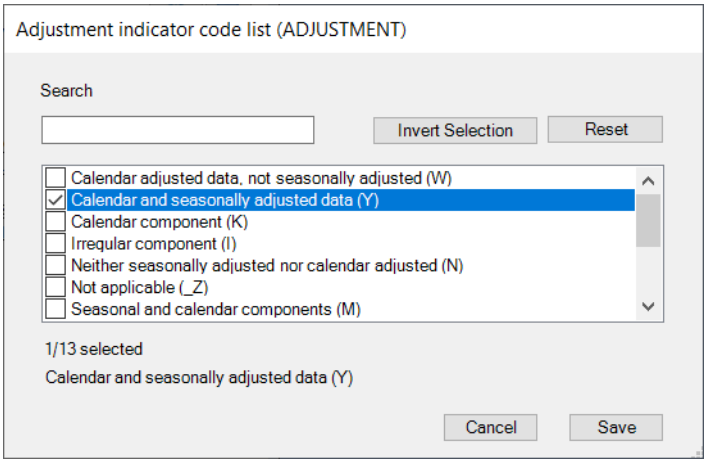
You can type a keyword in the **Filter** edit field to find a dataset. Select the dataset and click **Next** to continue.

Selecting Series

In this dialog, you will choose the series to import:



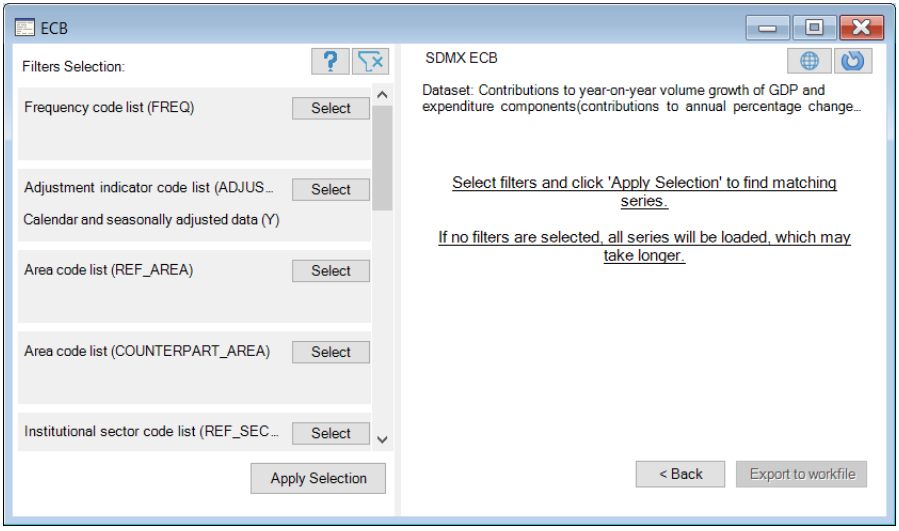
We begin by forming a list of potentially relevant series. By default, all of the series in the dataset will be in the list. If available, you may use filters on the left-hand side of the dialog to reduce the number of series to be read. To specify a filter, click on the **Select** button on the left-hand side of the dialog corresponding to a specific category. You will be presented with a dialog showing the possible filter elements:



Click on the filter element entry to toggle the selection. You may use the **Search** button to locate a specific filter entry.

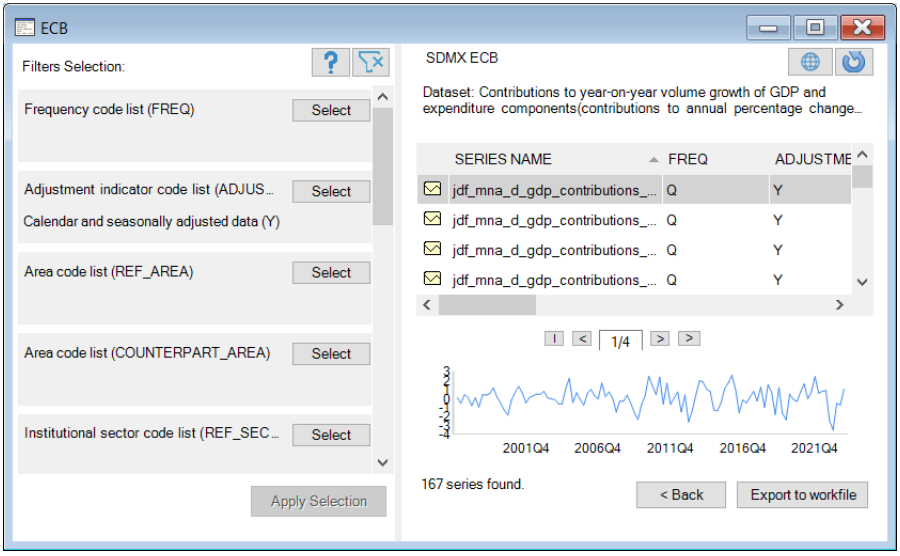
Here we use the **Adjustment indicator code list** to require data to be calendar and seasonally adjusted data:

Once the filter definition for the category is complete, click **Save** to save the filter settings and return the main selection dialog.



The dialog lists any relevant filter elements in text form under the name of the filter. If the information is not fully legible, you may hover your cursor over the text to see the full specification.

Once you finish defining any selection filters, click on the **Apply Selection** button to load a list of the relevant series in the dataset:



Here, there are 167 series found in the dataset that match the current filter. Click on any of the entries on the right to display information about the individual series and to display a thumbnail graph of the series.

You may go back and specify additional filters and click on the **Apply** button to update the list of relevant series.

Finally, you may select one or more series in the list, then click on the **Export to workfile** button to load the selected series into a new or existing workfile.

Note that the **Refresh Data** button in the upper-right corner may be used to ensure the database information is up-to-date. For some databases, there will also be an **Open in Browser** button which will take you to the selected dataset on the database website if available. This feature is applicable only to those databases that offer an online data browser.

- See "Foreign Format Databases" on page 362 in *User's Guide I*.
- See also `dbopen` (p. 428) in the *Command and Programming Reference*.

JDemetra+ Seasonal Adjustment

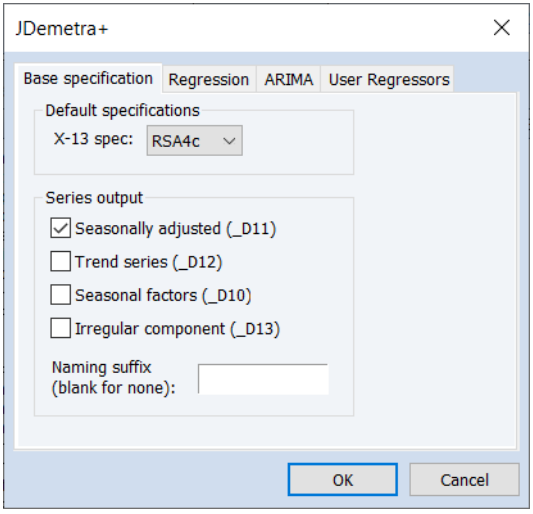
JDemetra + is an open-source seasonal adjustment and time-series analysis package created by the European Commission. EViews incorporates a subset of the functionality provided by JDemetra + to perform X-13 style seasonal adjustment on monthly and quarterly data.

While the results provided JDemetra + will often be numerically identical to the results provided by Census X-13 (for identical settings), JDemetra + offers robust handling of missing or extreme values, and computes the seasonal adjustment with reduced processing time.

When accessed from within EViews, JDemetra + will perform seasonal adjustment on the series over the current workfile sample. In contrast to other seasonal adjustment methods, JDemetra + offers automatic handling of missing values:

- For internal missing values (those not at the start or end of the sample), JDemetra + uses interpolation to fill in the missing values.
- For missing values at the beginning or end of the sample, EViews will adjust the start and end of the adjustment sample. While JDemetra + permits extrapolation of missing values at the beginning and end of the sample, EViews simply the trims the sample to remove NAs before sending the data to JDemetra + .

To perform JDemetra + seasonal adjustment in EViews, click on **Proc/Seasonal Adjustment/JDemetra + ...** from the series window menu in a monthly or quarterly workfile. This will open the JDemetra + tabbed dialog:



The first tab, **Base specification** allows you to choose one of the JDemetra+ preset specifications, as well as selecting which series to output into the workfile, and optionally specifying a suffix to be used in naming the output series.

- The **X-13 spec:** drop down menu specifies the preset specification. JDemetra+ offers a number of specifications with settings for the pre-treatment and decomposition steps of the seasonal adjustment:

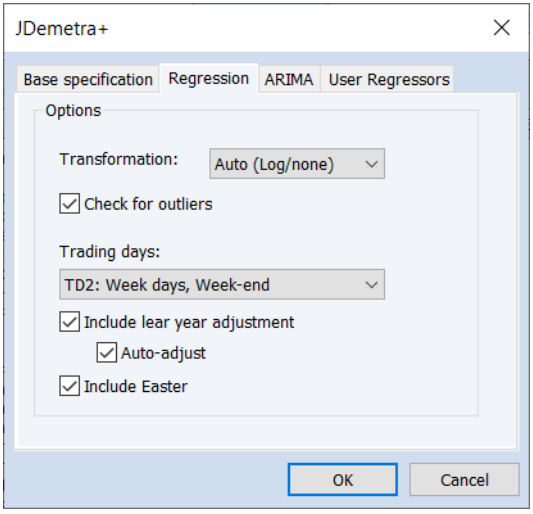
Specification	Log/Level Detection	Outlier Detection	Calendar Effects	ARIMA Model
X-11	None	None	None	None
RSA0	None	None	None	Airline (0,1,1)(0,1,1)
RSA1	Automatic	AO/LS/TC	None	Airline (0,1,1)(0,1,1)

RSA2c	Automatic	AO/LS/TC	TD2 (week-days vs week-ends), Easter	Airline (0,1,1)(0,1,1)
RSA3	Automatic	AO/LS/TC	None	Automatic selection
RSA4c	Automatic	AO/LS/TC	TD2 (week-days vs week-ends), Easter	Automatic selection
RSA5	Automatic	AO/LS/TC	TD7 (7 day of week variables), Easter	Automatic selection

For in-depth details of these specifications, we recommend browsing the JDemetra+ website (https://github.com/jdemetra).

- The **Series output** section specifies which of the output series from JDemetra+ will be exported to the workfile. By default, each output series will be created a name equal to that of the underlying series plus the type of series being created (*e.g.*, if JDemetra+ is run on the series GDP, then the seasonally adjusted D11 series will be created with a name of GDP_D11). You can use the **Naming suffix** edit field to enter an additional suffix that will be appended to the series name before the output type (*e.g.*, if you enter "_JD" as the **Naming suffix** then the D11 series for GDP will be created with a name of GDP_JD_D11).

The **Regression** tab of the dialog allows you to override some of the options set by the default specifications relating to the pre-adjustment regression step of X-13 style seasonal adjustment.



Selecting the X-13 specification on the **Base specification** tab will change the settings on this tab, but you can fine tune the default settings by using the options on this tab.

- The **Transformation** drop down menu specifies whether a log transformation should be applied to the underlying series before running the regression. **Auto (Log/none)** instructs JDemetra+ to automatically detect whether a log transform should be applied or not.
- The **Check for outliers** check box specifies whether JDemetra+ should automatically detect outliers. EViews' implementation of JDemetra+ supports only the Additive Outlier (AO), Level Shift (LS), and Temporary Change (TC) types of outliers, and when the option is checked, JDemetra+ will detect for all three types simultaneously.
- **Trading days** sets the type of calendar effects used in the pre-adjustment regression. These effects consist of created variables with the count of the number of days in each period. The options are:

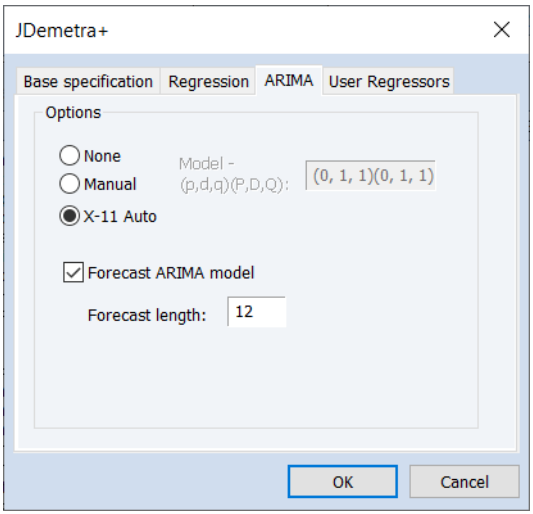
None	Do not include calendar effects
TD7	Create 7 calendar variables. The first counts the number of Mondays in the month/quarter. The second counts the number of Tuesdays, and so on.
TD4	Create 4 calendar variables. The first counts the number of days from Monday to Thursday, the second counts the number of Fridays, the third counts Saturdays, and the fourth counts Sundays.
TD3c	Create 3 calendar variables. The first counts the number of days from Monday to Thursday, the second counts the number of Fridays and Saturdays, and the third counts Sundays.
TD3	Create 3 calendar variables. The first counts the number of days from Monday to Friday, the second counts the number of Saturdays, and the third counts Sundays.
TD2	Create 2 calendar variables. The first counts the number of week days (Mon-Fri) and the second counts the number of weekend days (Sat-Sun).
TD2c	Create 2 calendar variables. The first counts the number of days from Monday to Saturday, and the second counts the number of Sundays.
User variables	Include user-provided calendar variables only.

The created variables are compacted into a single calendar variable through a transformation. For example, TD2 creates the calendar variable as NumWeekDays-(5/2)*NumWeekends.

If **User variables** is selected as the trading day type, you must specify series to use as calendar variables on the **User Regressors** tab.

- The **Include Leap Year adjustment** and **Include Easter** checkboxes control whether additional adjustments are made to the calendar effects for the impact of leap years and Easter. The **Auto-adjust** checkbox specifies whether JDemetra will automatically determine whether to drop the leap-year adjustment.

The **ARIMA** tab provides options for the estimation of the ARIMA model in the pre-adjustment step:



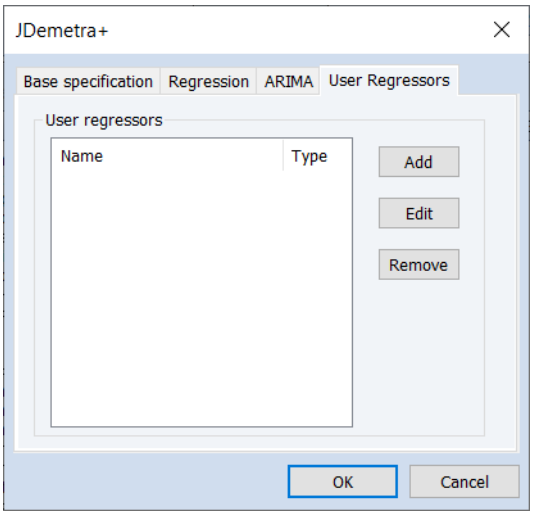
- The **ARIMA Method** section selects the method used to specify the ARIMA model to be estimated. **None** instructs JDemetra+ to not estimate an ARIMA model at all.

  If **Manual** is selected, you can enter a specific ARIMA order in $(p, d, q)(P, D, Q)$ notation, where p is the AR order, d is the differencing level, q is the MA order, P is the seasonal AR order, D is the seasonal differencing level, and Q is the seasonal MA order.

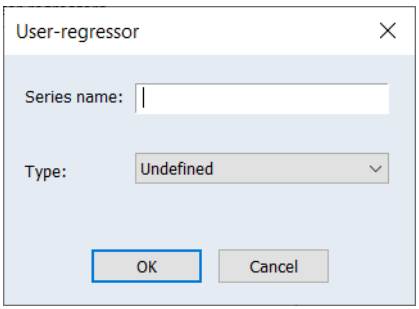
  If **X-11 Auto** is selected, JDemetra+ will use its default X-11 selection routine to select the most appropriate ARIMA order.

- The **Forecast ARIMA model** checkbox instructs JDemetra+ to forecast the ARIMA model beyond the end of the series. The number of periods forecasted can be changed using the **Forecast length** edit field. Forecasting the ARIMA model allows JDemetra+ to provide forecasts of the final seasonally adjusted series and seasonal factors. Note forecasts will only be imported into EViews if the workfile range covers the period of the forecast.

The **User Regressors** tab of the dialog supports providing user-provided exogenous series to the pre-adjustment ARIMA/regression models.



Clicking on the **Add** button brings up a dialog asking you to type in the name of the workfile variable that you would like to include as a regressor:



You may enter any valid EViews series name or expression, such as "X" or "log(X)", or "@pch(X)" in the **Series name** edit field. You may also type in a space delimited list to add multiple series at once.

The **Type** drop down menu should be used to specify a regressor type for the variable you are including (*i.e.*, **Series**, **Trend**, **Seasonal**, etc.). Changing the type of the regressors changes the exact impact that series has on the final seasonal adjustment calculation. The JDemetra+ documentation has details on the exact calculations.

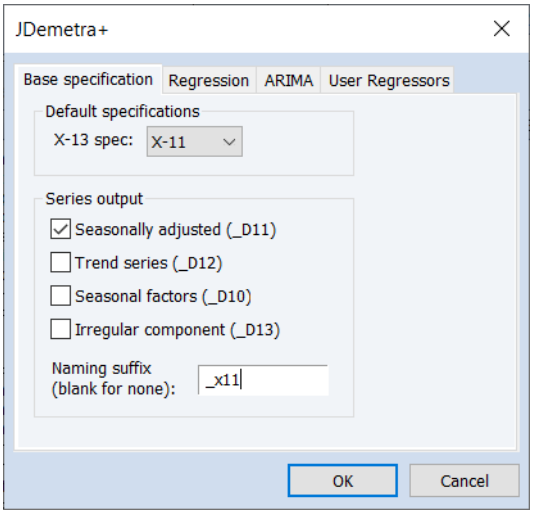
If you selected **User Variables** as the **Trading Days** type, you must add at least one user-regressor with a type of **Calendar/TradingDay**.

Example

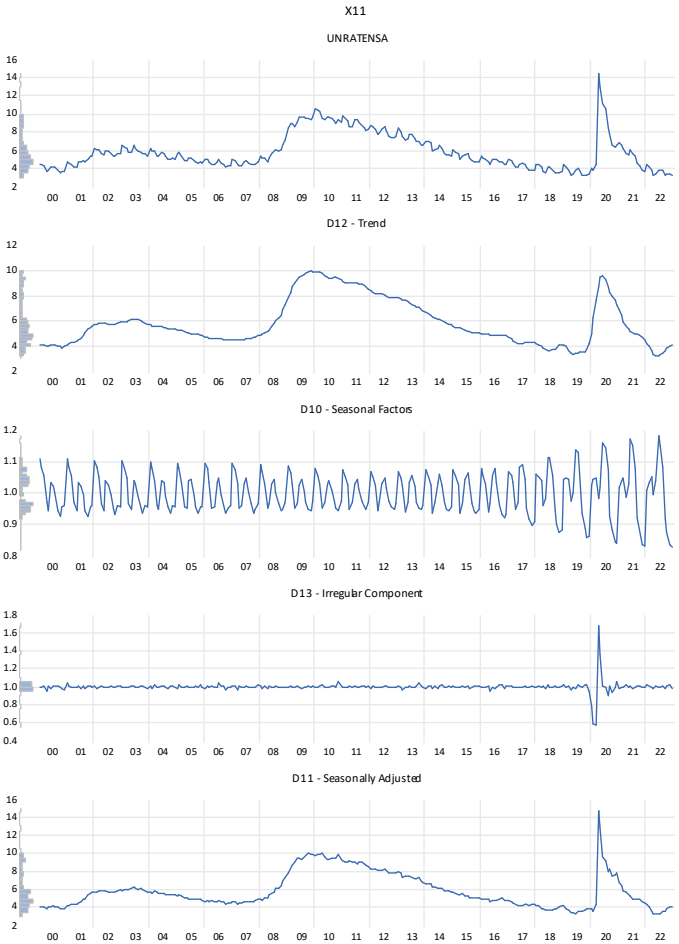
The workfile, "X13_Macro.wf1" includes a series called UNRATENSA which contains monthly non-seasonally adjusted US unemployment data from January 2005 to June 2012. We will use this series to demonstrate the user of JDemetra+ seasonal adjustment in EViews. We will later use the same series to demonstrate X-13 seasonal adjustment.

We will perform simple X-11 based seasonal adjustment on this series without any pre-adjustment regression options being set. To do so open the UNRATENSA series and click on **Proc/Seasonal Adjustment/JDemetra+ ...** to open the JDemetra+ dialog.

Simple X-11 adjustment is one of the pre-set JDemetra+ defaults, so we change the **X-13 spec:** dropdown to **X-11**. We'll elect to store the seasonally adjusted values in the workfile, and enter "_x11" in the **Naming suffix** edit field:



Clicking **OK** to display the JDemetra+ graph output:



The output is split into five graphs. The top graph displays the original unadjusted series. The second displays the decomposed trend line, the third the seasonal/cyclical factors, the fourth the irregular/residual component, and finally graph displays the seasonally adjusted values. It should be noted that this simple X-11 adjustment via JDemetra+ will give numerically identical results to using EViews' implementation of Census X-13 using the default settings.

- See "Seasonal Adjustment" on page 507 in *User's Guide I*.
- See also `Series::jdemetra` (p. 801) in the *Object Reference*.

Boosted Hodrick-Prescott Filter

The Hodrick-Prescott Filter is a widely employed smoothing method for obtaining a smooth estimate of the long-term trend component of a series. The method was first proposed in a working paper (circulated in the early 1980's and published in 1997) by Hodrick and Prescott to analyze postwar U.S. business cycles.

EViews 14 enhances the existing routines with support for the iterated (boosted) HP filter proposed by Phillips and Shi (2020).

Technically, the Hodrick-Prescott (HP) filter is a two-sided linear filter that computes the smoothed series s of y by minimizing the variance of y around s, subject to a penalty that constrains the second difference of s. The HP filter chooses the values of s to minimize:

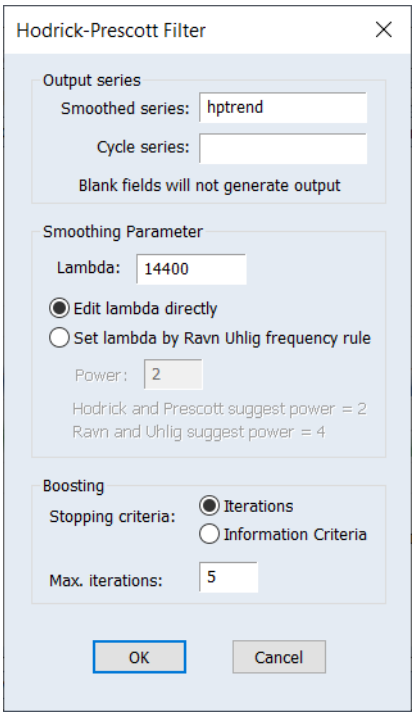
$$\sum_{t=1}^{T}(y_t - s_t)^2 + \lambda \sum_{t=2}^{T-1}((s_{t+1} - s_t) - (s_t - s_{t-1}))^2. \quad (0.1)$$

The s series is often referred to as the trend series. The cyclical component c of the original series can be computed as $c_t = y_t - s_t$.

The penalty parameter λ controls the smoothness of s. The larger the λ, the smoother the σ. As $\lambda \to \infty$, s approaches a linear trend.

Phillips and Shi (2020) have proposed iterating the HP filter to produce a "smarter smoothing device." This *boosted* HP filter takes the cyclical series, c and runs the filter on it one more time to produce a new smoothed and cycle series. The filtering process is repeated, producing a further smoothed series at each iteration. The advantage of this iterative procedure is that the final smoothed series is less reliant on the choice of λ. Phillips and Shi recommend repeating the process either after a set number of iterations, or through the use of information criteria to decide the optimal number of iterations.

To smooth the series using the Hodrick-Prescott filter, choose **Proc/Hodrick-Prescott Filter…**:



First, provide a name for the **Smoothed series**. EViews will suggest a name, but you can always enter a name of your choosing. If you wish to save a **Cycle series**, specify a name in the edit field.

Next, specify an integer value for the smoothing parameter, λ. You may specify λ directly by clicking on the **Edit lambda directly** radio button and entering a value in the **Lambda** edit field, or you may specify a value using the frequency power rule of Ravn and Uhlig (2002) (the number of periods per year divided by 4, raised to a power, and multiplied by 1600 by clicking on the **Set lambda by Ravn Uhlig frequency rule** and entering a **Power** value in the edit field.

By default, EViews will fill the defaults using the Ravn and Uhlig method with a power rule of 2, yielding the original Hodrick and Prescott values for λ:

$$\lambda = \begin{cases} 100 & \text{for annual data} \\ 1{,}600 & \text{for quarterly data} \\ 14{,}400 & \text{for monthly data} \end{cases} \tag{0.2}$$

Ravn and Uhlig recommend using a power value of 4. EViews will round any non-integer values that you enter.

The **Boosting** section of the dialog offers settings for iterative boosting of the HP filter. You may choose between stopping based on the maximum number of iterations or using an Information criteria.

If you click on **Iterations**, EViews will stop based on the entry in the **Max. Iterations** edit field. By default, there will be no boosting as only one iteration of the filter will be performed.

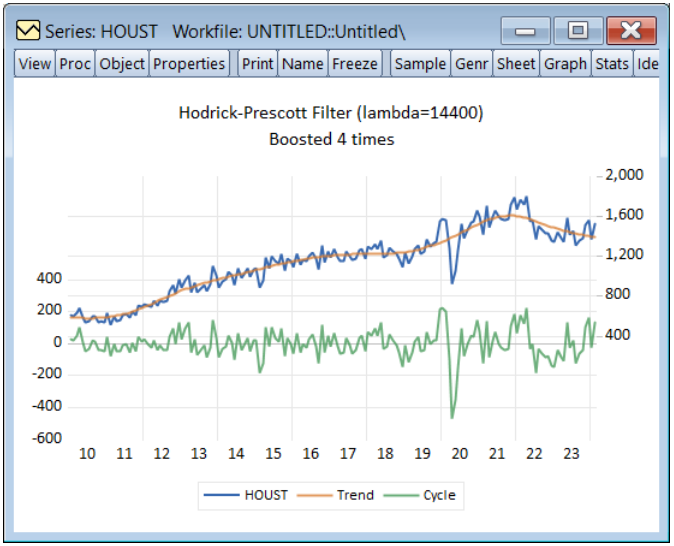
Selecting the **Information criteria** radio button instructs EViews to select the optimal number of iterations using information criteria. The **Max. Iterations** edit field should be used to specify the number of iterations to be considered.

When you click on **OK**, EViews displays a graph of the filtered series together with the original series. Note that only data in the current workfile sample are filtered. Observations for the smoothed and cyclical series outside the current sample will be filled with NAs.

For example, we may download housing starts data from the Federal Reserve of St. Louis database:

```
dbopen(type=fred, server=api.stlouisfed.org/fred)
wfcreate m 1959M01 2024M02
fetch(d=fred) houst
smpl 2010 2024m02
```

and then perform HP filtering with 5 iterations on the HOUST series using values from 2010m01 through 2024m02:



The newly created HPTREND series contains the smoothed values of HOUST.

- See also `Series::hpf` (p. 799) in the *Object Reference*.

Utility Functions

Sample Index Functions

A key EViews feature is the ability to work with subsamples of observations. There are two common ways of defining a subsample of observations in a workfile. One method is to specify a set of (0, 1) (boolean) identifiers that indicate whether each observation in the workfile is included in the subsample. A second method is to specify a list of index values for the observations in the subsample.

In some cases, working with the boolean indicators is more convenient. In others, especially when the subsample is sparse and direct access to the observation is useful, working with the index values may be preferred.

EViews 14 provides new workfile page functions that allow you to extract information from the workfile and workfile sample, and to convert between the two methods.

Current Sample Index (@pagesmplidx)

The `@pagesmplidx` function returns a vector object containing the index values for the observations in the current sample.

Suppose we have an annual workfile from 2001 to 2024, and have set the workfile sample to a subset of observations. The commands

```
wfcreate(wf=awf) a 2001 2024
smpl 2002 2003 2010 2011
```

create a workfile containing annual observations from 2001 to 2024, then sets the sample to contain observations from 2002–2003 and 2010–2011. There are 24 observations in this workfile which may be identified by index values 1 to 24. Then

```
vector ids = @pagesmplidx
```

will return IDS containing {2, 3, 10, 11}.

See `@pagesmplidx` (p. 1026) in the *Command and Programming Reference*.

Observations Index (@pageidx)

The `@pageidx(`*arg*`)` function returns a vector object containing the index values for the observations in the workfile specified in *arg*.

The argument may be a vector of date numbers, an svector of date strings, or a string object or string literal containing a space delimited list of dates.

Observation specifications that are outside of the workfile range will be ignored.

For example,

```
wfcreate(wf=qwf) q 2001 2024
vector id1 = @pageidx("2010q2 2010q3 2021q1 2023q4")
```

creates a workfile containing quarterly data from 2001 to 2024 and an ID1 vector containing the values {38, 39, 81, 92}.

See `@pageidx` (p. 1023) in the *Command and Programming Reference*.

Boolean Indicators (@pageinidx)

The `@pageinidx(`*arg*`)` function, where *arg* is a vector containing observation index values, returns a vector object, sized to match the workfile page length, that includes (0, 1) indicators for each observation, with 1's assigned to index elements in *arg*, and 0's elsewhere.

If the *arg* contains index values that are outside of the workfile range, the function will return an error.

If we have the commands

```
vector id2 = @fill(10, 27, 30, 40)
vector incl = @pageinidx(id2)
```

then INC1 is a vector with 0's everywhere except for the elements 10, 27, 30, and 40. Note that these commands are a concise equivalent to the commands

```
vector inc2 = @zeros(@wfrange)
inc2(10) = 1
inc2(17) = 1
inc2(30) = 1
inc2(40) = 1
```

Using `@pageidx` to identify observation indices and then feeding the result to `@pageinidx` offers a quick way of specifying a subsample of observations using dates:

```
wfcreate(wf=qwf) q 2001 2024
vector inc3 = @pageinidx(@pageidx("2010q2 2010q3 2021q1 2023q4")
stom(inc3, include_series)
smpl @all if include_series = 1
```

See `@pageinidx` (p. 1024) in the *Command and Programming Reference*.

Matrix Utility Functions

EViews 14 offers a few new matrix utility functions that make common operations easier to perform.

The `@lower` (p. 949) and `@upper` (p. 1167) (in the *Command and Programming Reference*) functions make it easy to create lower and upper triangular matrices from matrix and sym

objects. The functions allow you to specify an offset from the main diagonal so you can create strict lower and upper matrices, or matrices which zero out all but small lower or upper triangular regions of the source matrix:

EViews 14 also adds the `@implodeu` (p. 918) function to create a symmetric matrix from the upper triangle of a square matrix. This function complements the existing `@implode` (p. 917) which creates the sym from the lower triangle of a matrix.

The new `@incr` (p. 918) function allows for adjusting all of the columns or rows of a matrix by incrementing with a vector. You may, for example, use this function to quickly add or subtract a multiple of a vector from each column of a matrix, or add or subtract a multiple of a rowvector from each row of a matrix, eliminating the need for expensive Kronecker product operations.

String and Date Vector Functions

Earlier versions of EViews offered an extensive library of string and date functions which allowed for manipulation of string values and date numbers. You could, for example, easily compare string values, insert strings into other strings, or extract a specific word from a string containing a list of words. Or you could manipulate date numbers to find a date two weeks in the future, or determine the number of days between two dates.

Unfortunately, the EViews library for these functions was generally limited to functions of scalar values. If, for example, you wanted to perform a set of on a set of string dates held in an svector, you needed to extract each element of the svector individually, perform operation on the string and any corresponding date number, and then place the result into a new vector or svector.

EViews 14 features a completely revamped string and date function library which features vector support. Functions which previously only took single string or date number values now operate seamlessly with arguments in vector form.

Consider, for example the `@left` (p. 940) (in the *Command and Programming Reference*) function which returns the n left-most characters in a string S. The first argument of the function corresponds to an alphanumeric S, and the second argument is a numeric n.

For this function, EViews 14 will return different types of results for different combinations of valid S and n argument types. The command

```
= @left("When in the course", 4)
```

returns the string “When” which is displayed in the EViews status line.

If A is a vector object containing the values 1, 2, 3, 4,

```
svector s1 = @left("When in the course", A)
```

will produce an svector S1 containing the strings "W", "Wh", "Whe", and "When". In this case, EViews evaluates the function repeatedly using the literal value "When in the course" for S paired with each of the four values of A for n.

Alternately, if V is a 4-element svector,

```
svector s2 = @left(V, 4)
```

will return a 4-element svector S2 containing the left-most 4 characters of the corresponding elements of V.

Lastly,

```
svector s3 = @left(V, A)
```

will pair the elements of the svector V and vector A as defined above, returning an svector containing the first character of the first element of V, the first two characters of the second element of V, the first three characters of the third element of V, and the first four characters of the fourth element of V.

To take another example, suppose we wish to specify a string date, find the corresponding underlying date number, manipulate that number to add various numbers of weeks, and then find a string representation of the new date number.

We may now perform a vectorized version of this operation which finds a string representation for 11 weekly days beginning at an initial scalar date:

```
wfcreate d7 2000 202
vector weeks = @seq(0, 1, 11)
string startdate = "May 1, 2000"
svector enddates = @datestr(@dateadd(@dateval(startdate), weeks,
    "W"))
```

This operation uses the `@dateval` (p. 823) function to convert the initial string into a date number, the `@dateadd` (p. 813) function to find the date numbers associated with the initial date number and the following 10 weeks, and the `@datestr` (p. 822) function to convert the result back into a string.

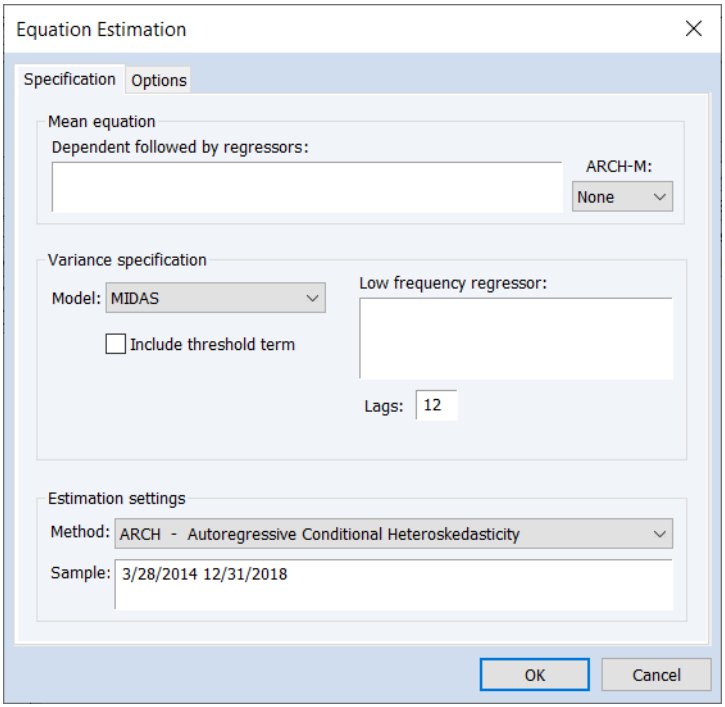
Econometrics and Statistics

MIDAS GARCH

EViews 14 estimates the multiplicative component MIDAS GARCH(1,1) model of Conrad and Kleen (2020).

Mixed Data Sampling (MIDAS) regression is an estimation technique which allows for data sampled at different frequencies to be used in the same regression. For MIDAS GARCH models, the approach is to incorporate information from a large number of lags of a lower fre-

quency series into the variance specification of an ARCH regression; incorporating, for example, lags of quarterly data in a monthly data GARCH model.



While most of the dialog should be familiar, the MIDAS GARCH variance specification uses lags of a low frequency regressor. You should enter the name of a single permanent component regressor in the **Low frequency regressor** edit field.

The syntax for specifying this variable is *pagename\seriesname* where *pagename* is the name of the page containing the series, and *seriesname* is the name of the series.

Note that EViews only allows the specification of a single low frequency regressor.

You should use the **Lag** edit field to specify the number of low frequency regressor lags to include in the permanent component.

- See "MIDAS GARCH" on page 1314 in *User's Guide II*.
- See `Equation::arch` (p. 64) in the *Object Reference*.

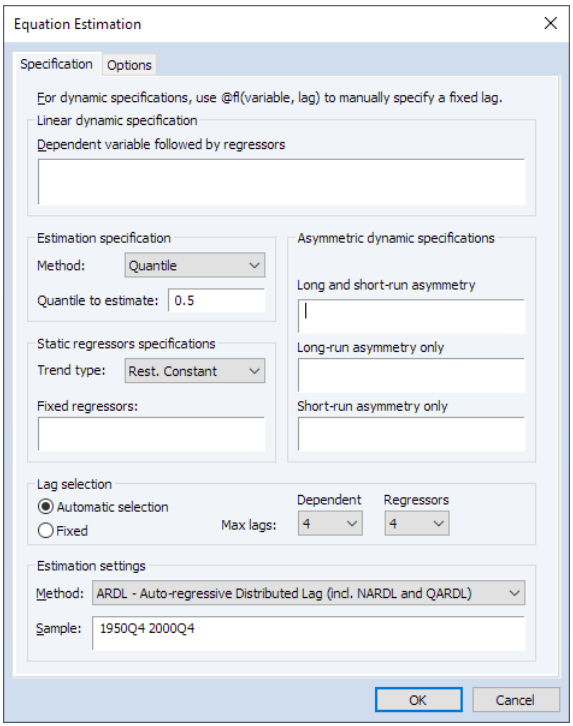
Quantile ARDL Estimation

The Quantile Autoregressive Distributed Lag (QARDL) model, introduced by Cho, Kim, and Shin (2015), is an extension of traditional ARDL models to capture the dynamics of conditional quantiles (percentiles) of the dependent variable. While conventional models provide

insights into the mean responses of the dependent variable to changes in predictors, QARDL models allow you to model the effects of changes in predictors on the quantiles of the dependent variable.

While QARDL models may be estimated using generic quantile regression tools by specifying models using with the appropriate levels, lags and lag differences of variables, EViews 14 offers an easy-to-use native interface for estimating QARDL and QNARDL models.

To estimate a QARDL or QNARDL model, select **ARDL - Autoregressive Distributed Lag Models** in the estimation **Method** drop down at the bottom of the dialog, and then select **Quantile** in the **Estimation specification** section in the middle of the left column:



Preliminary documentation for estimating Quantile ARDL is available. For primary documentation:

- See Chapter 29. "ARDL and Quantile ARDL," beginning on page 1347 in *User's Guide II*,
- See `Equation::ardl` (p. 71) for updated command documentation for Quantile ARDL estimation in the *Object Reference*.

There are new commands associated with Quantile ARDL views (all in the *Object Reference*):

- `Equation::qrcrprocess` (p. 216) displays a spool object producing a quantile process of the cointegrating relation.
- `Equation::qrecprocess` (p. 217) displays a spool object producing a quantile process for each of the conditional error correction and error correction coefficients.

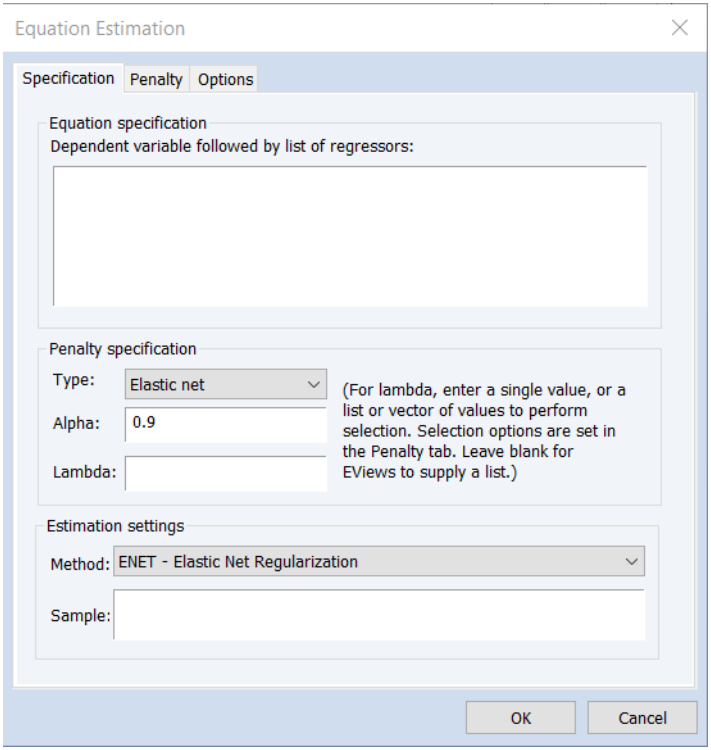
Enhanced Elastic Net and Lasso

The elastic net (Enet) estimator performs penalized least squares regression with a penalty that depends on a parameterized function of the absolute and squared values of the coefficients. Notably, the class of Enet models includes the special cases of ridge and Lasso (Least Absolute Shrinkage and Selection Operator) regression.

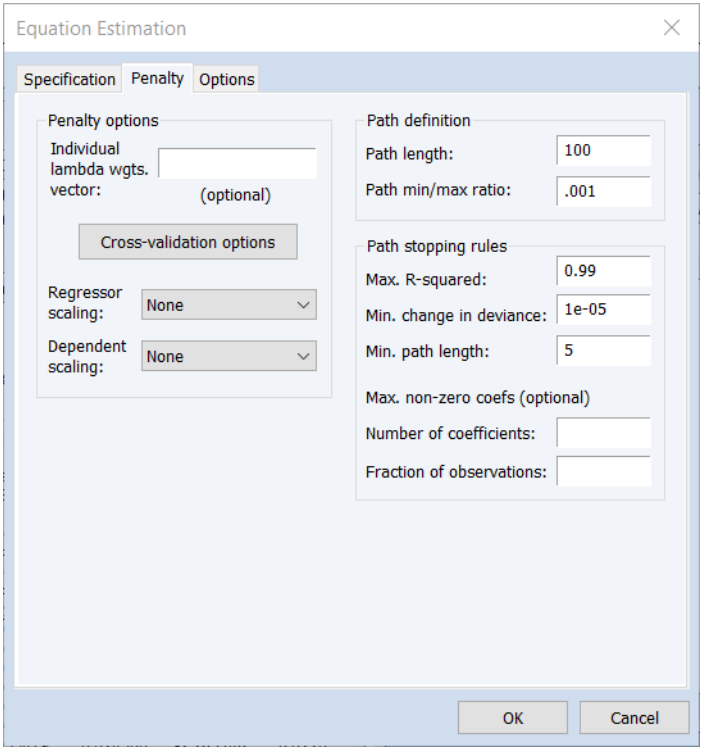
While prior versions of EViews did offer tools for elastic net estimation, EViews 14 completely updates the existing features to offer more efficient algorithms for estimation, added control over coefficient penalties including individual coefficient weights and coefficient bounds, more efficient cross-validation tools for selecting models along the lambda path, and enhanced views and tools for examining the behavior of coefficients, estimation objectives, and model fit statistics along the path.

(Note that Enet equations estimated in EViews 14 are not backward compatible with earlier versions and these equations will not be read by previous versions. Conversely, Enet equations estimated in EViews 13 and prior will need to be re-estimated in EViews 14).

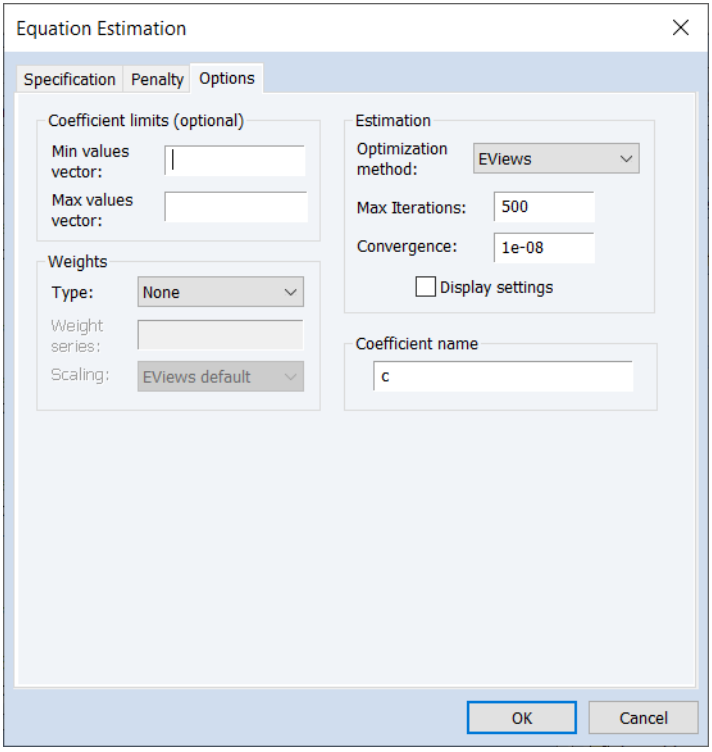
To estimate an elastic net model, select **ENET - Elastic Net Regularization** in the **Method** drop-down menu:



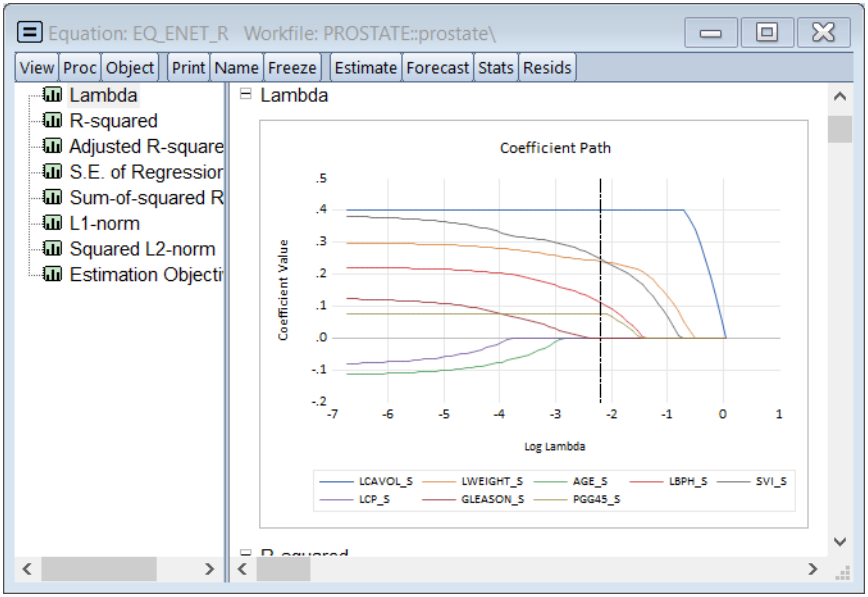
The **Penalty** page offers new settings for individual penalty weights, path definition, cross-validation, and variable scaling:



The **Options** page offers, among other things, new options for placing limits on coefficients, and optimization methods:



Once you have estimated a path model, EViews 14 offers improved views tools for examining the behavior of coefficients, estimation objectives, and model fit statistics along the path. Clicking on **View/Coefficient Path/Graphs displays** a spool object plotting coefficients against various values:



For additional discussion:

- See Chapter 37. "Elastic Net and Lasso," on page 1603 in *User's Guide II*.
- See `Equation::enet` (p. 126) for updated estimation command documentation in the *Object Reference*.

There are a number of new or improved commands associated with elastic net views (all in the *Object Reference*):

- `Equation::coefpath` (p. 95) – display graphs of the paths of the coefficients plotted against lambda, fit measures, and estimation values.
- `Equation::cvgraph` (p. 115) – display a graph of the cross-validation objective against the lambda path.
- `Equation::lambdacoefs` (p. 175) – display the spreadsheet of the matrix of coefficient values along the lambda path.
- `Equation::lambdaest` (p. 176) – display the table showing various values associated with estimation along the lambda path.
- `Equation::lambdafit` (p. 177) – display the table showing various fit statistics associated with estimates along the lambda path.
- `Equation::lambdapath` (p. 178) – display graphs of lambda against various fit and estimation measures.

- `Equation::modselgraph` (p. 205) – display a graph of the selection criteria for the top 20 models.
- `Equation::modseltable` (p. 206) – display a table of the selection criteria and measures.

New equation data members have been introduced to provide access to the results of estimation and cross-validation:

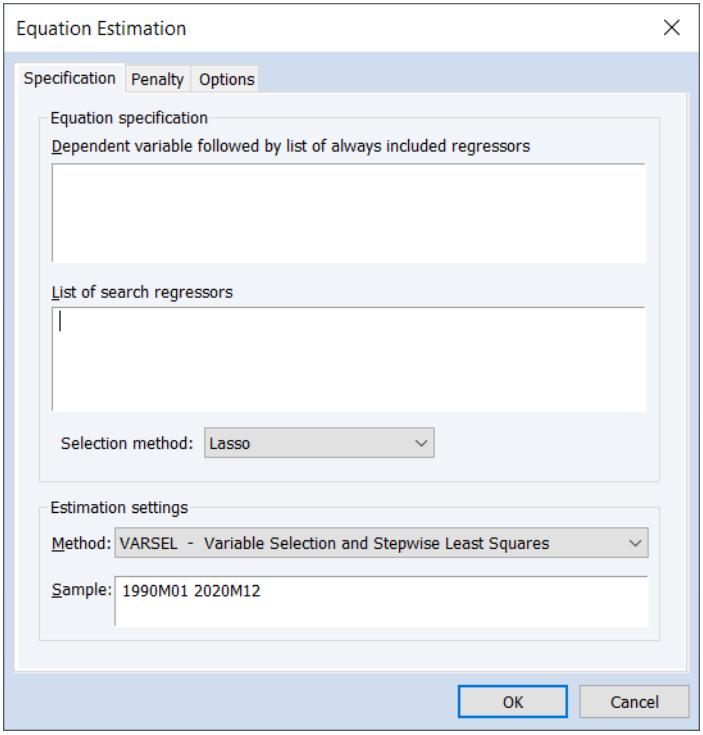
- See "Equation Data Members," on page 57 of the *Object Reference*.

Lasso Variable Selection

The Lasso variable selection (VARSEL) features in EViews have been updated to use the enhanced elastic net estimation features.

(Note that VARSEL equations estimated using Lasso in EViews 14 are not backward compatible with earlier versions and these equations will not be read by previous versions. Conversely, VARSEL equations estimated using Lasso from EViews 13 and prior will need to be re-estimated in EViews 14).

To perform a VARSEL using Lasso in EViews select **Object/New Object/Equation,** or press **Estimate** from the toolbar of an existing equation. From the **Equation Specification** dialog choose **Method: VARSEL-Variable selection and Stepwise Least Squares,** and choose **Lasso** in the **Selection** method drop down menu. EViews will display the following dialog:

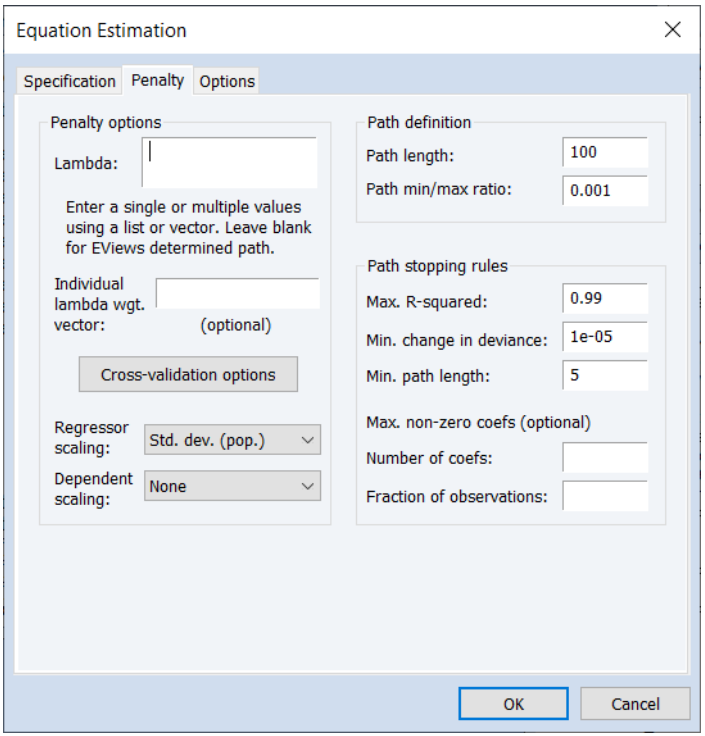


Lasso Options

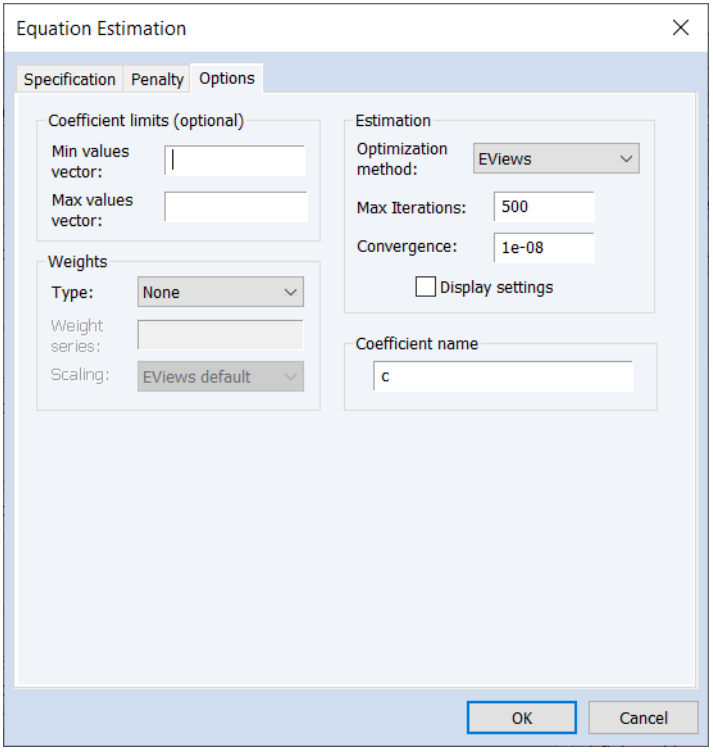
For Lasso selection, the there are two options dialog pages that allow you control the specification of the objective and the estimation method.

You may use the dialog pages to control the determination of the penalty parameter, to specify data transformation options, individual observation weights, and to control the iterative estimation procedure.

The first options page is the **Penalty** page:



The second **Options** page allows you to set coefficient limits, add observation weights, control the estimation method and convergence, and to change the coefficient name:



The Varsel documentation has been updated to describe the new interface:

- See Chapter 22. "Regression Variable Selection," on page 1071, in particular, "Lasso Options" on page 1081 and "Lasso Example" on page 1090, all in *User's Guide II*
- See `Equation::varsel` (p. 260) in the *Object Reference*.

There are a number of new or improved commands associated with the Lasso variable selection views (all in the *Object Reference*):

- `Equation::coefpath` (p. 95) – display graphs of the paths of the coefficients plotted against lambda, fit measures, and estimation values.
- `Equation::cvgraph` (p. 115) – display a graph of the cross-validation objective against the lambda path.
- `Equation::lambdacoefs` (p. 175) – display the spreadsheet of the matrix of coefficient values along the lambda path.
- `Equation::lambdaest` (p. 176) – display the table showing various values associated with estimation along the lambda path.

- `Equation::lambdafit` (p. 177) – display the table showing various fit statistics associated with estimates along the lambda path.
- `Equation::lambdapath` (p. 178) – display graphs of lambda against various fit and estimation measures.

New equation data members have been introduced to provide access to the Lasso and model selection results:

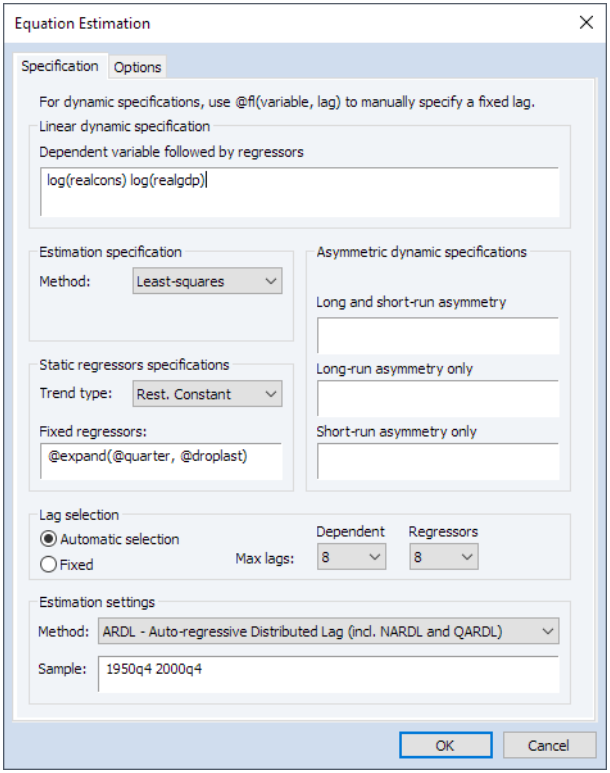
- See "Equation Data Members," on page 57 of the *Object Reference*.

Expanded ARDL HAC Standard Errors

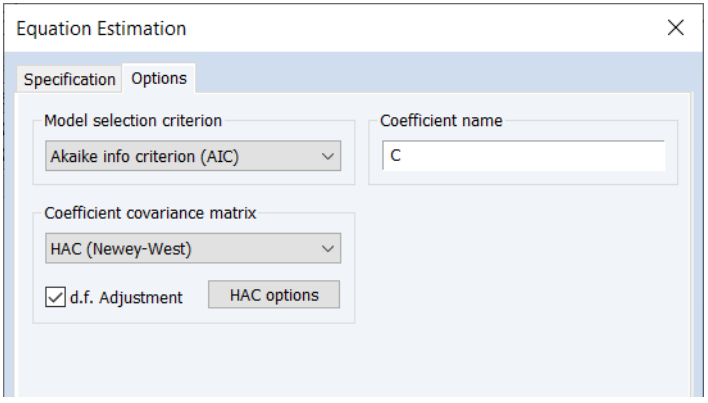
EViews 14 expands the calculation of heteroskedasticity and autocorrelation consistent (HAC) estimates of coefficient covariances in ARDL equations. Previously, HAC coefficient covariances were only computed for the base ARDL estimates of the Intertemporal Dynamics (ITD) representation of the specification. HAC estimates were not available for the Conditional Error Correction (CEC) or Error Correction (EC) forms of the model.

EViews now computes HAC estimates for all three representations.To estimate an ARDL model with HAC correction select **Object/New Object/Equation,** or press **Estimate** from the toolbar of an existing equation.

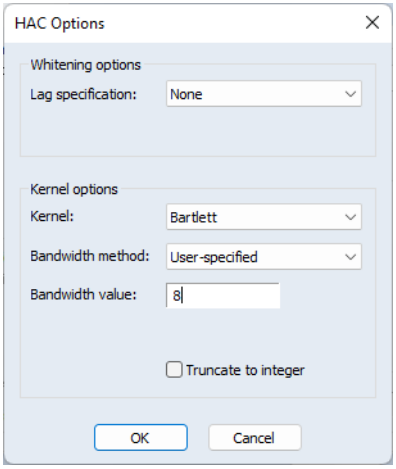
From the **Equation Specification** dialog choose the **ARDL - Auto-regressive Distribution Lag Models** in the **Method** dropdown to display the estimation dialog:



Fill on the **Specification** page as desired, and then click on the **Options** tab to display the **Coefficient Covariance Matrix** settings:

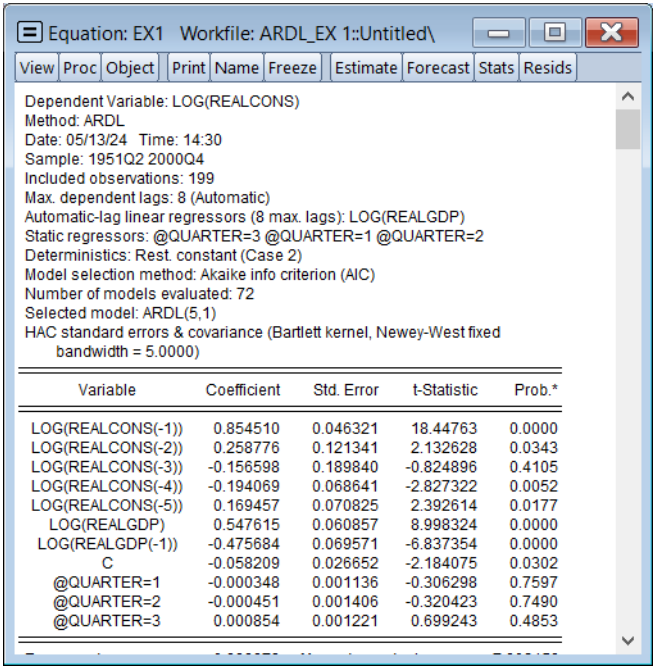


Select **HAC (Newey-West)** from the dropdown menu. You may click on the **HAC options** button to display a popup dialog that will allow you to change from the default settings:

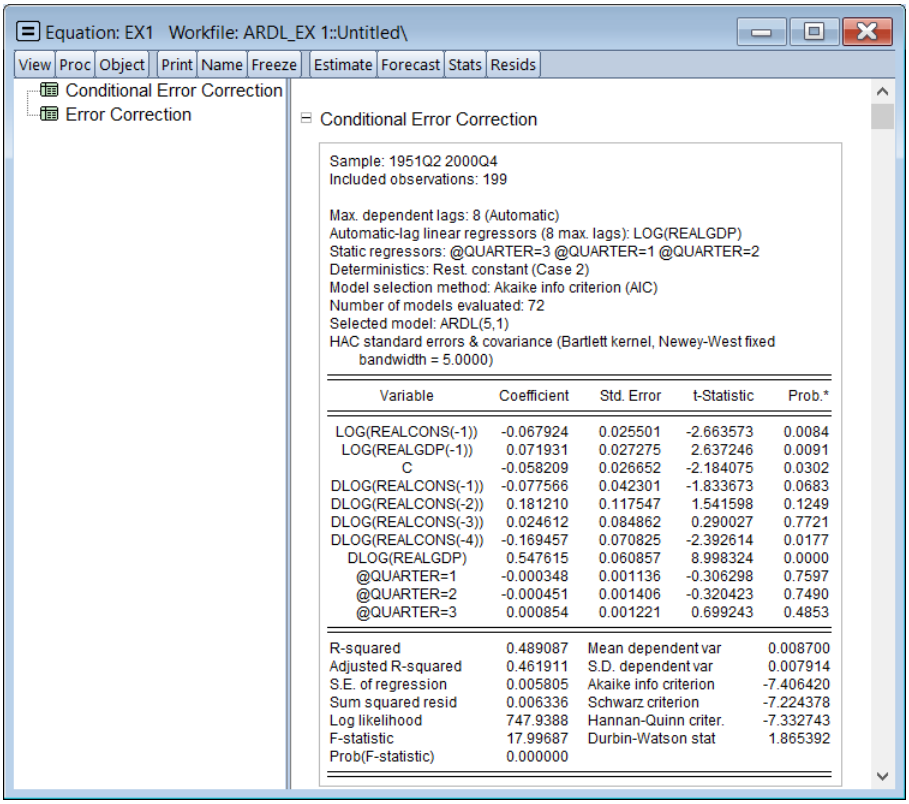


If you change the **HAC options**, click on **OK** to accept the changes, then click on **OK** on the estimation dialog to estimate your ARDL with using the specification and options.

The IC results are displayed by default:



To see the CEC and the EC results, select **View/ARDL Diagnostics/Error-correction Results** to display the estimates for the alternative representations:



Note that the labeling of output in this view may have been misleading on prior versions of EViews as the comments related to estimation of coefficient covariances only applied to the ITD specification.

- See Chapter 29. "ARDL and Quantile ARDL," on page 1347 in *User's Guide II* for documentation on ARDL estimation in EViews.
- See `Equation::ardl` (p. 71) for updated estimation command documentation in the *Object Reference*.

Facebook Prophet Forecasting

EViews 14 offers access to the powerful Facebook Prophet forecasting engine through a new, easy-to-use user interface.

Prophet is an open-source forecasting tool launched by Facebook in 2017, designed to handle time series data and predict future trends. It is specifically tailored for forecasting scenarios characterized by complex seasonality, missing data points, and outliers. Although originally designed for working with daily data, Prophet can forecast time series of any frequency. Prophet is available as both a Python and an R package.

By default, a custom Prophet Python package is installed in your EViews installation directory. If Prophet was not installed with EViews, you will need to reinstall EViews.

Background

The Prophet forecasting tool offers a flexible, easy-to-configure method of forecasting time series data. A particular emphasis of the Prophet design is its support for users with an understanding of the data-generating process but limited knowledge of time series methods, and its ability to produce a large number of forecasts with limited human intervention (Taylor and Letham 2018).

The original Prophet is a time series model which decomposes a series into additive trend, seasonal and holiday components:

$$y_t = T(t) + S(t) + H(t) + \epsilon_t \tag{0.3}$$

where $T(t)$ is the trend component, $S(t)$ represents periodic seasonal (weekly, monthly, yearly, *etc.*) components, $H(t)$ represents potentially irregular holiday effects, and ϵ_t is a normally distributed idiosyncratic error.

A multiplicative seasonal version of the Prophet model may be written as:

$$y_t = T(t) \cdot (S(t) + H(t)) \cdot \epsilon_t \tag{0.4}$$

The current EViews implementation supports the trend $T(t)$ and seasonal $S(t)$ components of the Prophet specification.

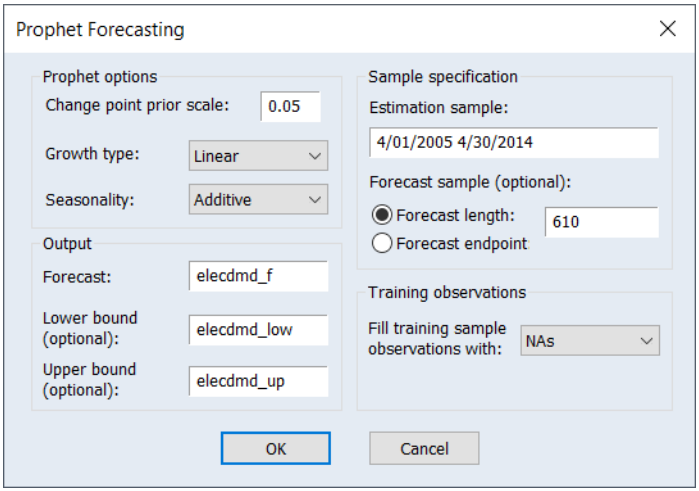
The specification of these components is described in some detail in Taylor and Letham (2018). Roughly speaking:

- The trend component $T(t)$ is modeled using a piecewise logistic or piecewise linear growth model with explicit change-points that may be specified or automatically determined. EViews supports automatic determination of change-points.
- The smooth seasonal component $S(t)$ is approximated using a linear combination of a set of Fourier series.

Estimation of the Prophet model employs Bayesian techniques with normal priors on the trend and seasonal Fourier series, and a Laplace prior on the change-points.

Using Prophet in EViews

To use Prophet in EViews to forecast a series, open the series and click on **Proc/Automatic Forecasting/Prophet…** which will bring up the **Prophet Forecasting** dialog.



The **Prophet options** section of the dialog allows you to select some of the options offered by Prophet.

- The **Change point prior scale** option specifies the strength of the sparse prior used to automatically detect change points in the underlying series trend. Larger values of the scale make the trend more flexible (*i.e.*, tend toward a larger number of change points). Lower values make the trend less flexible.
- **Growth type** specifies whether the underlying series should grow at a **Linear** or **Logistic** rate.
- The **Seasonality** combo box selects whether to model seasonality in the series in an **Additive** or a **Multiplicative** fashion.

The **Sample specification** section of the dialog includes the **Estimation Sample** box and the **Forecast length** box.

- The **Estimation Sample** determines the observations used in training the Prophet model. By default, EViews will fill the dialog with the current workfile sample.
- **Forecast length** specifies h, the number of observations that will be forecast after training. The forecast sample will start immediately after the last observation of the estimation sample and will continue for h observations. Note the workfile must be sized such that h observations exist in the workfile after the estimation sample.

- As an alternative to **Forecast length**, you may specify a **Forecast endpoint**. The observation specified as the forecast endpoint must follow the last observation in the estimation sample and fall within the range of the workfile.

The **Output** section boxes are used to name the final forecast series in the workfile.

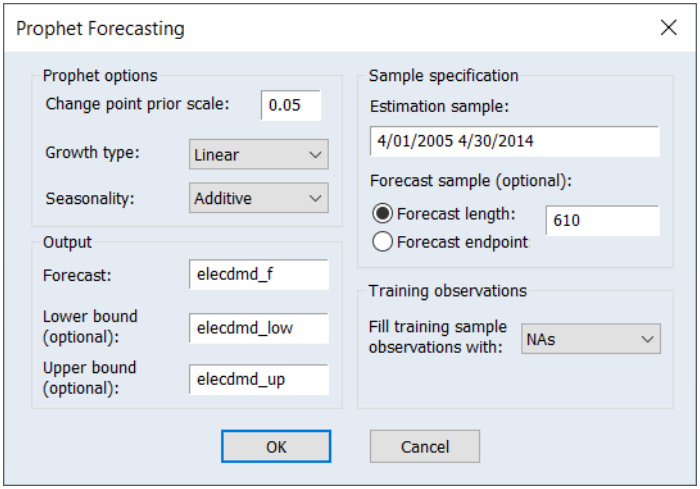
- You must provide a **Forecast** name for the output series. By default, edit field is filled with the name of the underlying series followed by an "_F".
- The optional **Lower bound** and **Upper bound** edit fields allow you to save the lower and upper forecast bounds computed by Prophet. By default, the edit fields are prefilled with the name of the underlying series followed by "_LOW" and "UP", respectively.

You may use the **Training observations** selection to specify how the forecast series' observations within the training sample will be filled in. By default, EViews overwrites the in-sample forecast values with NAs. You may instead instruct EViews to use the Prophet computed in-sample forecast for the training sample values, or to use the underlying series actual values. Any observations that are neither in the training sample nor the forecast sample will be filled with NAs.

Example

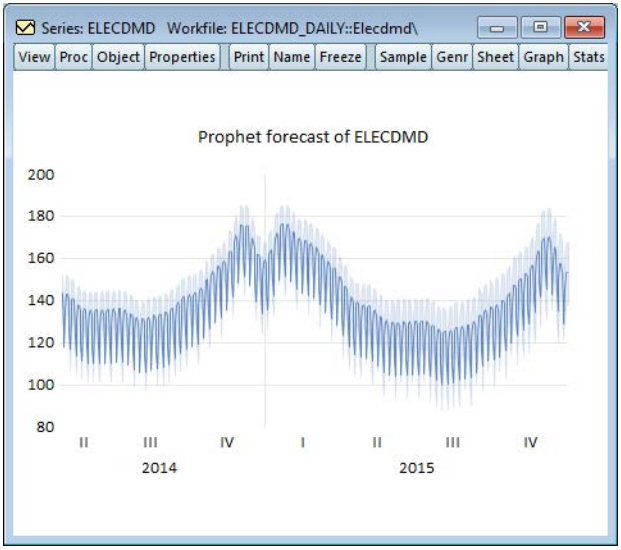
As an example of using Prophet in EViews, we forecast daily electricity demand in England and Wales, using the workfile "elecdmd_daily.wf1". This workfile contains monthly electricity demand data from April 2005 until April 2014 (in the series ELECDMD). The workfile's range extends through the end of 2015, even though data for ELECDMD is only available until April 2014.

To use Prophet to forecast the missing values, we open the ELECDMD series and click on **Proc/Automatic Forecasting/Prophet…** to bring up the Prophet dialog:



EViews automatically fills in the estimation sample to the current workfile sample (which was set to the actual data of ELECDMD), and calculates the forecast length based on the number of remaining observations.

We leave all other options at their default value and click OK to produce the forecast:



- See `Series::prophet` (p. 825) in the *Object Reference* for command documentation.

The Facebook Prophet interface is not available in EViews Standard Edition.

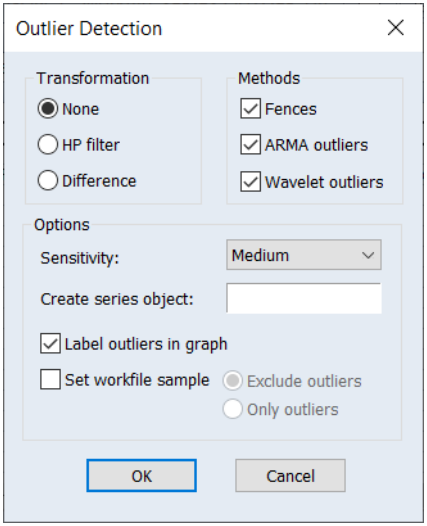
Series-based Outlier Detection

Outliers are observations in a series that significantly differ from the majority of the other observations. Whether due to measurement error, data entry mistakes, or simply natural variability in the data, outlier observations have values that stand out as unusual, rare, or abnormal when compared to the rest of the data.

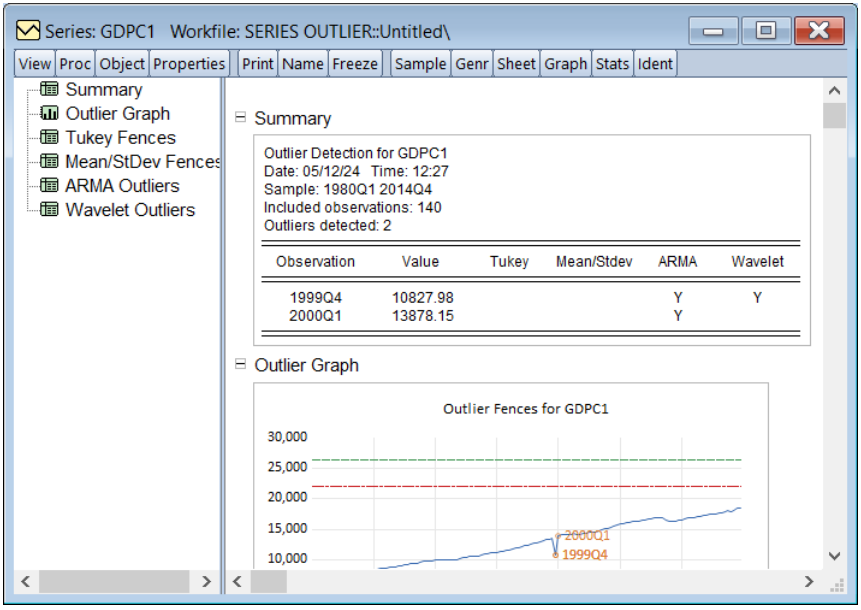
The presence of outliers can have a substantial impact on statistical analysis, as outliers can skew statistical results and lead to inaccurate conclusions if not appropriately identified and handled.

EViews 14 offers new easy-to-use tools for identifying outliers in a series or in the residuals of an estimated equation. You may use Tukey fences, mean/standard deviation fences, Wavelet outliers, and ARMA based outlier detection to identify outlier observations.

To perform outlier identification on observations in a single in EViews, open the series objctand click on **View/Outlier Detection…**. EViews will display the **Outlier Detection** dialog:



Filling out the dialog with the appropriate settings and clicking on **OK** produces a spool containing outlier identification tables and graphs:



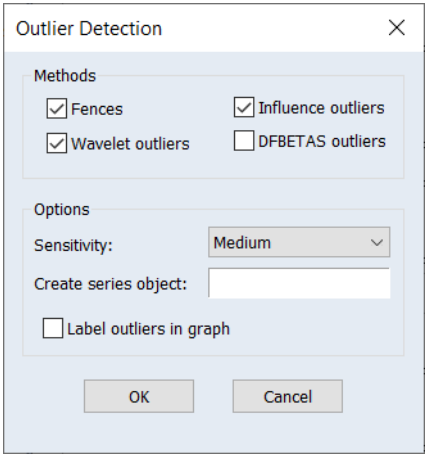
- For discussion of the identification methods and examples, see the preliminary documentation in "Outlier Detection," on page 478 in *User's Guide I*.
- See `Series::outliers` (p. 818) in the *Object Reference* for command documentation.

Equation-based Outlier Detection

The new outlier detection view of an equation performs various methods for detecting outliers associated with the equation estimates. This is a useful diagnostic tool to assess the validity of the model underlying the regression.

One set of equation outlier detection methods employs methods used to identify outliers in series ("Outlier Detection," on page 478 of *User's Guide I*). A second set of equation outlier methods examines several of the influence statistics which measure the sensitivity of regression estimates to individual observations ("Influence Statistics" on page 1265 of *User's Guide II*) to determine whether an observation is an outlier. Observations which exhibit influence that exceeds a specified bound c are identified as outliers.

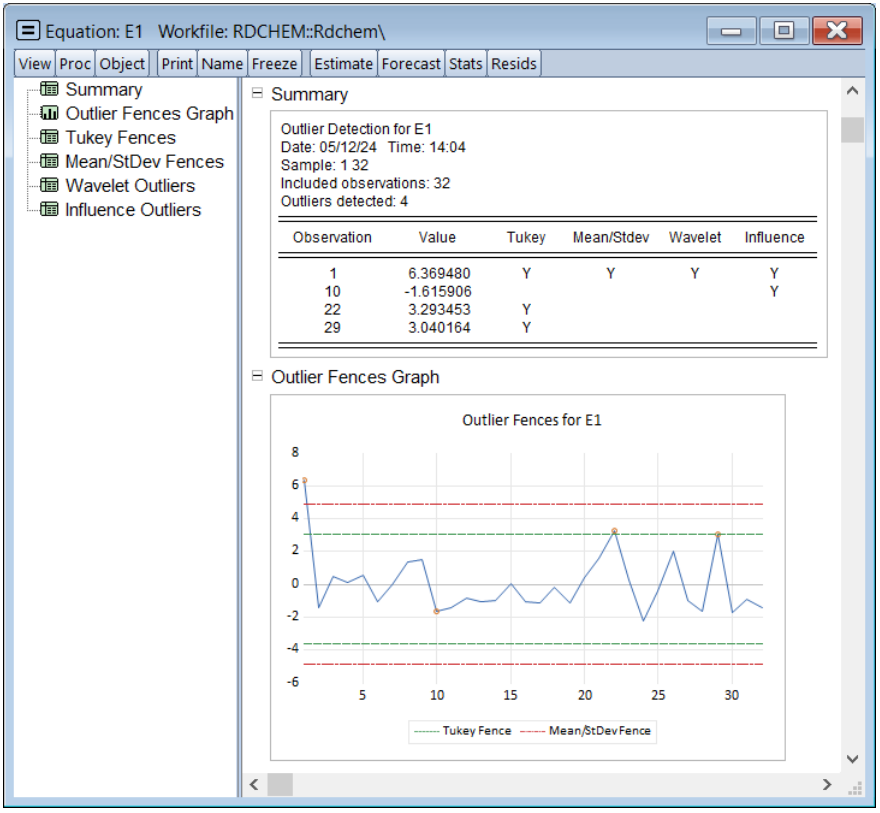
To perform outlier detection using the residuals from an equation in EViews, open the equation, and click on **View/Outlier Diagnostics...**. EViews will display the **Outlier Detection** dialog:



The **Methods** section offers checkboxes which select which of the outlier detection methods to use. The available options will depend on the method used to estimate the original equation.

- The **Influence outliers** and **DFBETAS outliers** are only available for equations estimated by linear least squares.
- **ARMA outliers** are only available for linear equations with ARMA terms.

When you click on OK, EViews will perform outlier identification using the specified methods and will display a spool containing both table and graphical results:

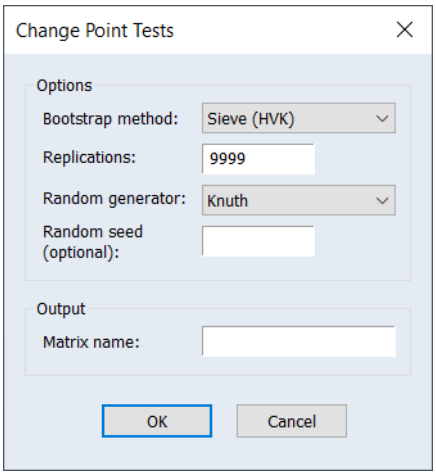


- See the documentation in "Outlier Detection," on page 1204 in *User's Guide II*.
- See `Equation::fitoutliers` (p. 137) in the *Object Reference* for command documentation.

Break Testing and Change Point Detection

EViews 14 now computes Quandt-Andrews Regression, Pettitt Ranks, and a Buishand Range and U tests for a single change in the location parameter (mean) of a series, with optional bootstrapping of the test p-values.

Simply click on **View/ Time-series Diagnostics/Change Point Tests...** from the main series menu to display a dialog containing test and output settings:



You may use the **Bootstrap method** dropdown menu to specify whether to compute bootstrap p-values, and if so, to select the bootstrap method. The remainder of the section provides options control the bootstrap computation. The **Output** section contains the **Matrix name** edit field for you to specify the name of a matrix object to store the test statistics and p-values.

The results of the change point tests are presented in a table:

Change Point Tests for HOUSTNSA
Date: 04/17/24 Time: 14:06
Sample: 2010M01 2024M02
Included observations: 170
Bootstrap test probabilities use 9999 replications (seed=1507069846)
Replications use Hall and Van Keilegom (HVK) estimates of an AR(1) process
Null Hypothesis: Series does not contain a mean change point

Test	Statistic	Change Location	Analytic Prob.	Bootstrap Prob.
Quandt-Andrews Max.	279.1501	1964M04	0.0000	0.0000
Quandt-Andrews Exp.	134.8264	1964M04	0.0000	0.0000
Quandt-Andrews Avg.	169.5444	1964M04	0.0000	0.0000
Buishand U	12.81732	1964M04	---	0.0000
Buishand Range	4.960767	1973M03	---	0.0000
Pettitt Ranks	6453.000	1965M04	0.0000	0.0000
AR Coefficients:				
Rho(1)	0.667479			

- For detailed discussion, see "Change Point Tests" on page 1761 in *User's Guide II*.

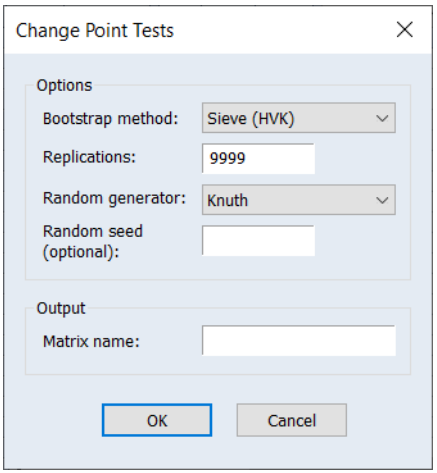
- See `Series::changepoints` (p. 773) for command documentation in the *Object Reference*.

Trend Testing

Many economic time series follow a time-trend. Identification of the presence of trends is important, both for intrinsic interest and as a preliminary step for further econometric analysis.

EViews 14 includes parametric and non-parametric tests for the presence of a trend in a series. You may compute the linear trend t-test, the quadratic trend F-test, the Mann-Kendall test, Cox-Stuart test, and the Wang, Akritas, and Van Keilegon (WAVK) tests of a trend against various alternatives, with optional bootstrapping of the test p-values

To perform both trend tests on a series in EViews, click on **View/ Time-series Diagnostics/ Trend Tests...** from the main series menu:



You may use the **Bootstrap method** dropdown menu to specify whether to compute bootstrap p-values, and if so, to select the bootstrap method. The remainder of the section provides options control the bootstrap computation. The **Output** section contains the **Matrix name** edit field for you to specify the name of a matrix object to store the test statistics and p-values.

The results of the change point tests are presented in a table:

Trend Tests for HOUSTNSA
Date: 04/17/24 Time: 14:23
Sample: 2010M01 2024M02
Included observations: 170
Bootstrap test probabilities use 9999 replications (seed=1155119733)
Replications use Hall and Van Keilegom (HVK) estimates of an AR(1) process
Null Hypothesis: Series does not contain a trend

Test	Alternative	Statistic	Analytic Prob.	Bootstrap Prob.
Trend t-stat	Linear	22.11571	0.0000	0.0000
Squared-trend F-stat	Quadratic	298.3256	0.0000	0.0000
Mann-Kendall	Monotonic	0.676627	0.0000	0.0000
Seasonal Mann-Kendall	Seasonal	14.21337	0.0000	---
Cox-Stuart	Monotonic	7.616288	0.0000	0.0000
WAVK	Any Form	444.2855	0.0000	0.0000

AR Coefficients:
Rho(1) 0.667479

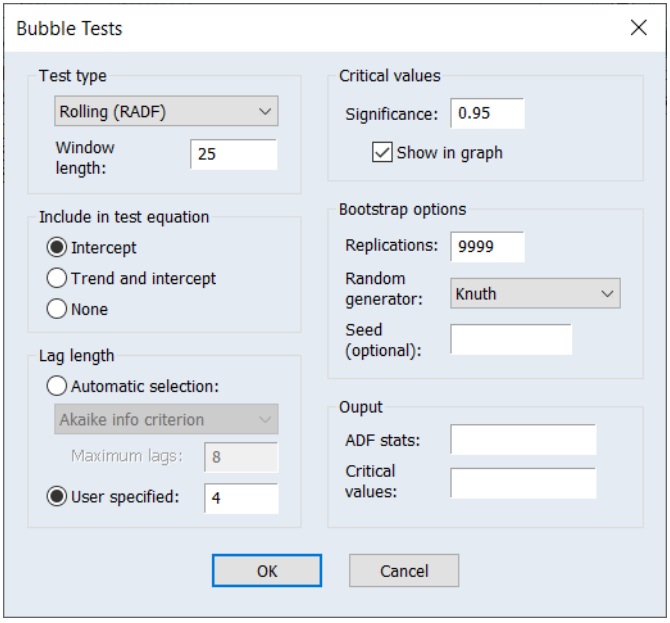
- For discussion, see "Trend Tests" on page 1767 in *User's Guide II*.
- See `Series::trendtests` (p. 862) in the *Object Reference* for command documentation.

Explosive Bubble Testing

Identification of bubbles in financial asset prices is an important topic in financial econometrics that has received considerable attention over the past decade (see Gürkaynak (2008) and Homm and Breitung (2012) for literature surveys).

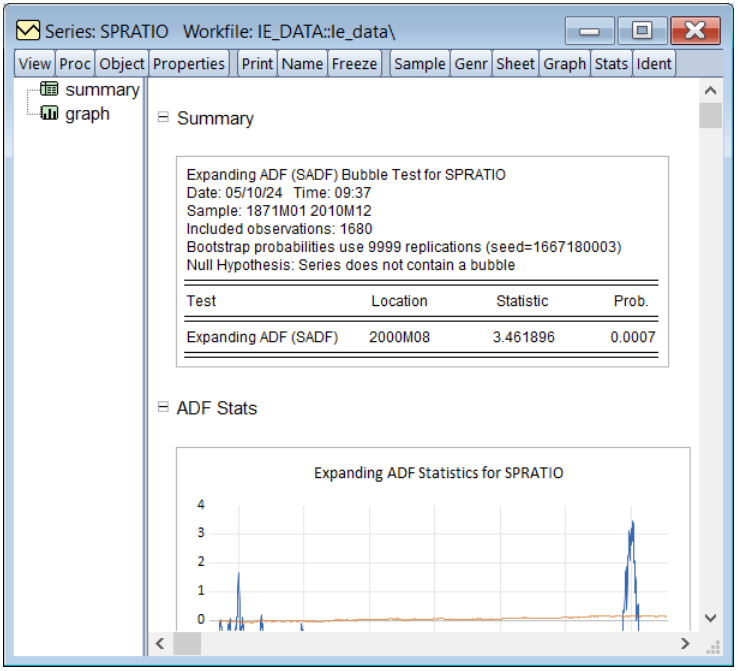
EViews 14 offers Phillips *et al.*(2011, PWY) and Phillips *et al.* (2015, PSY) tests for detection of bubbles.

To perform a bubble test in EViews, open the series and click on **View/Time Series Diagnostics/Bubble Tests…**



The dialog offers choices for different test types, test equations, ADF lag lengths, and options for computing the bootstrap test probabilities and saving results to a matrix in the workfile.

The output is in the form of a spool object containing a table showing the test results and a graph of the individual ADF statistics used in forming the test statistic.



- For detail, see "Explosive Bubble Tests" on page 1773 in *User's Guide II*.
- See `Series::bubbletest` (p. 769) in the *Object Reference* for command documentation.

Impulse Response Analysis

EViews 14 offers extensive additions and improvements to impulse response analysis of VARs and VECs, including impulse response via local projection and extensions of bootstrapping and monte carlo confidence intervals to additional models, and computation of Bayesian Time-varying Coefficient VAR fixed horizon impulse responses.

The new features include:

- "Impulse Response via Local Projection" on page 76.
- "SVAR Impulse Response and Variance Decomposition CIs" on page 78.
- "Variance Decomposition CIs" on page 80.
- "BTVCVAR Fixed Horizon Impulse-Responses" on page 82.

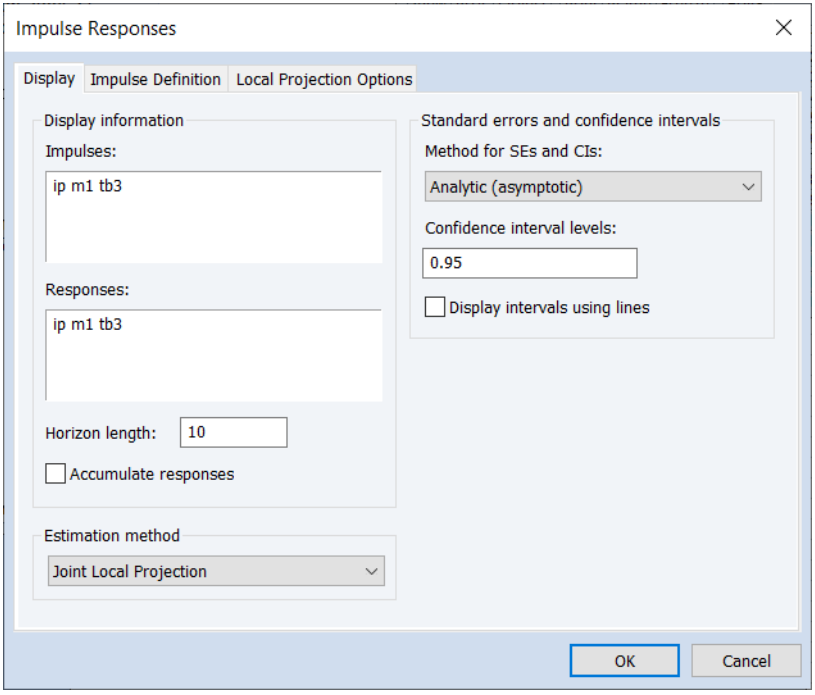
Impulse Response via Local Projection

The traditional VAR approach to impulse response estimation has both practical and theoretical shortcomings. In particular, the Wold decomposition can be difficult to derive, and may not exist, more likely in cases where the VAR is cointegrated. Further, impulse response functions derived from this method are justified only when the estimated VAR model coincides with the true DGP.

An alternative approach proposes the estimation of impulse responses via local projections (Jordà, 2005). The local projection (LP) technique is agnostic about the true DGP, and remains valid even when the Wold decomposition is undefined.

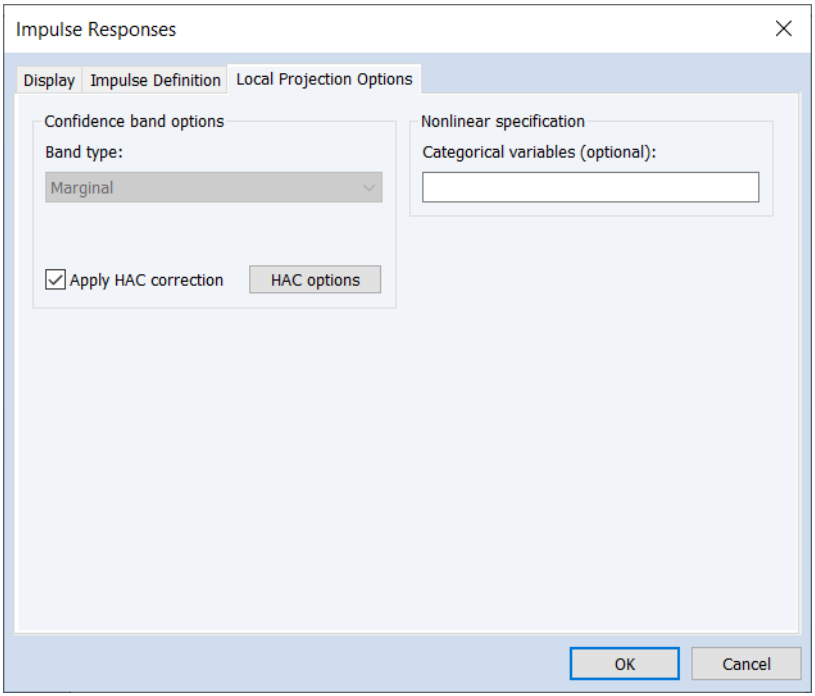
EViews 14 supports estimation of impulse responses via LP using both standard sequential and joint estimation. For sequential estimation, you may handle the serial correlation effect on covariance estimates using non-parametric HAC corrections.

Do perform impulse response analysis using LP, press the Impulse button on the toolbar of an estimated VAR or click on **View/Impulse response...** to display the dialog. Then select either **Sequential Local Projection** or **Joint Local Projection** in the impulse response **Estimation method** dropdown menu:



Most of the settings in this dialog are familiar, controlling the impulses and responses to display, the horizon length, and the method of computing CIs. If you choose the monte carlo or bootstrap simulation methods you will be presented with additional options. EViews supports the bootstrap methods outlined by Jordà (2009) and Monteil Olea and Plagborg-Moller (2020)

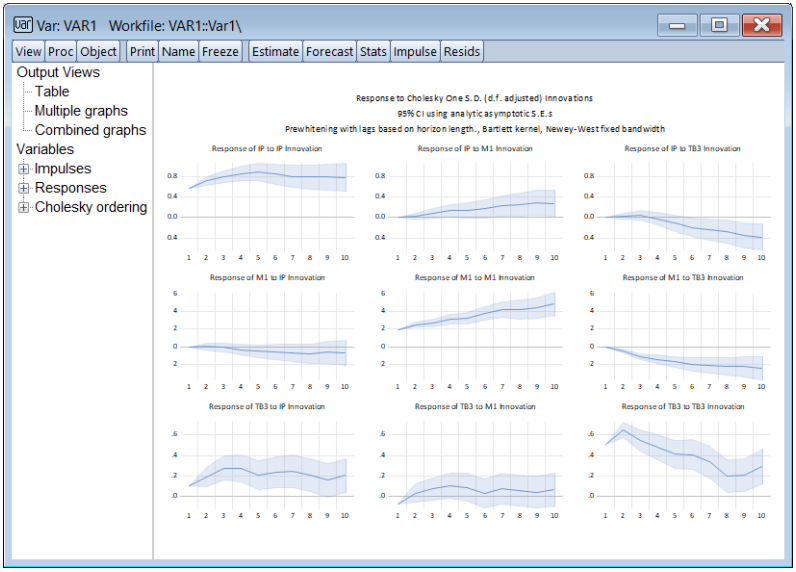
EViews will add a third tab to the dialog for setting **Local Projection Options**. Click on the tab to make the dialog page active.



The settings will differ depending on whether you are estimating using sequential or joint LP. Here we see the settings for sequential projection, which offer us the option of accounting for serial correlation using HAC adjustment.

Note that non-linear/asymmetric LP may be specified by entering one or more categorical variables in the edit field.

Click on **OK** to compute the impulse responses and display the spool output:

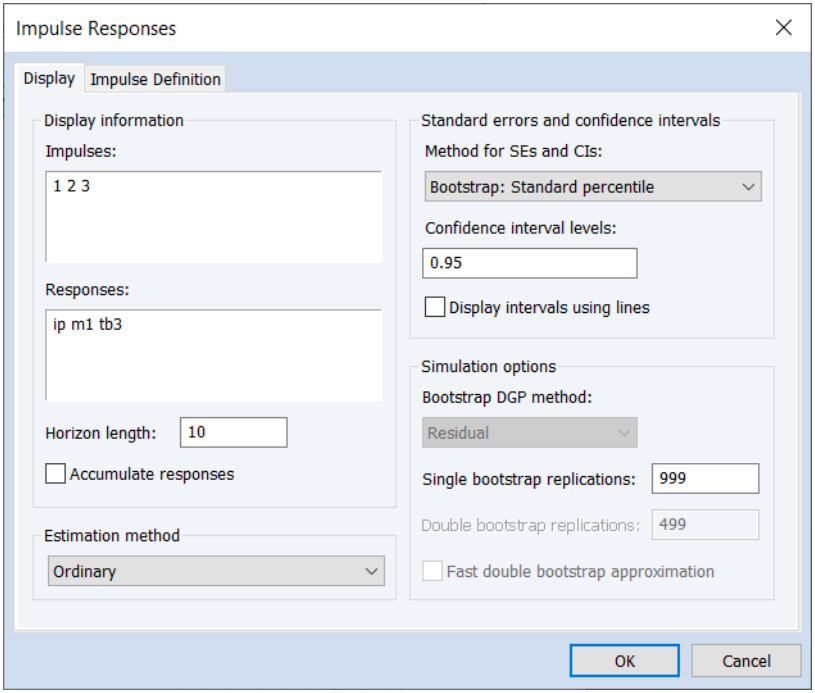


- For further detail, see "Local Projection" on page 2031 in *User's Guide II*.
- See `Var::impulse` (p. 1175) in *User's Guide II* for command documentation.

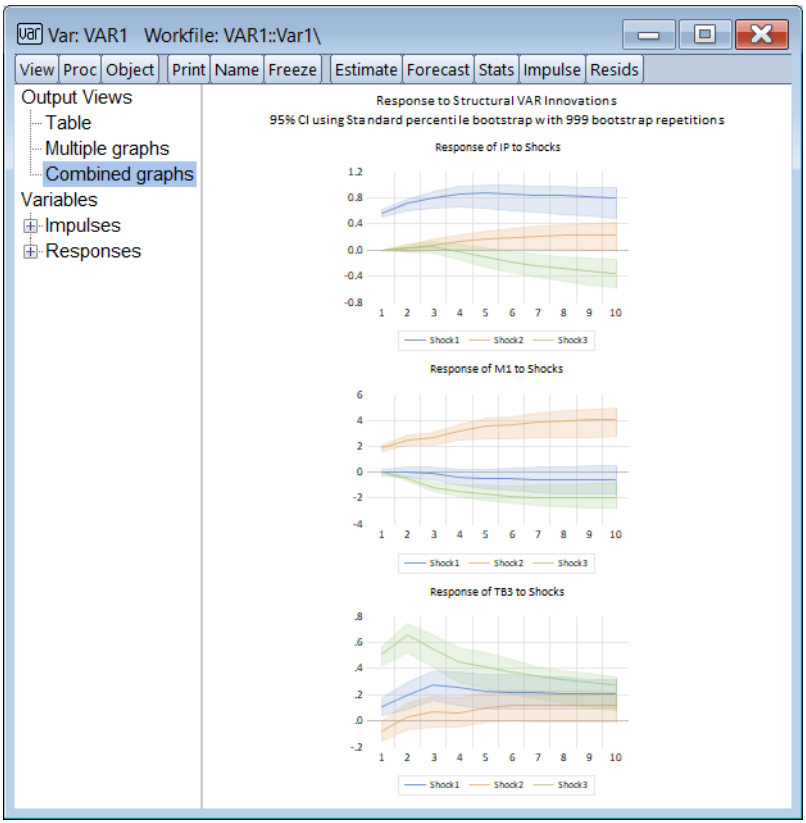
SVAR Impulse Response and Variance Decomposition CIs

Previous versions of EViews offered limited options for computing impulse response and variance decomposition CIs for structural VARS (SVARS). For impulse responses, CIs could only be computed using analytic or Monte Carlo methods; for variance decomposition, CIs and standard errors were unavailable.

EViews 14 can now perform Monte Carlo and bootstrap simulation for both SVAR impulse response and variance decomposition. If we press the **Impulse** button on the toolbar of an estimated SVAR or select **View/Impulse response...** from the menu, EViews will display the standard impulse response dialog:



Importantly, the **Method for SEs and CIs** offers both Monte Carlo and a full set of bootstrap methods and options, even is you are working with a structural factorization:



Similar features are available for variance decomposition given a structural factorization.

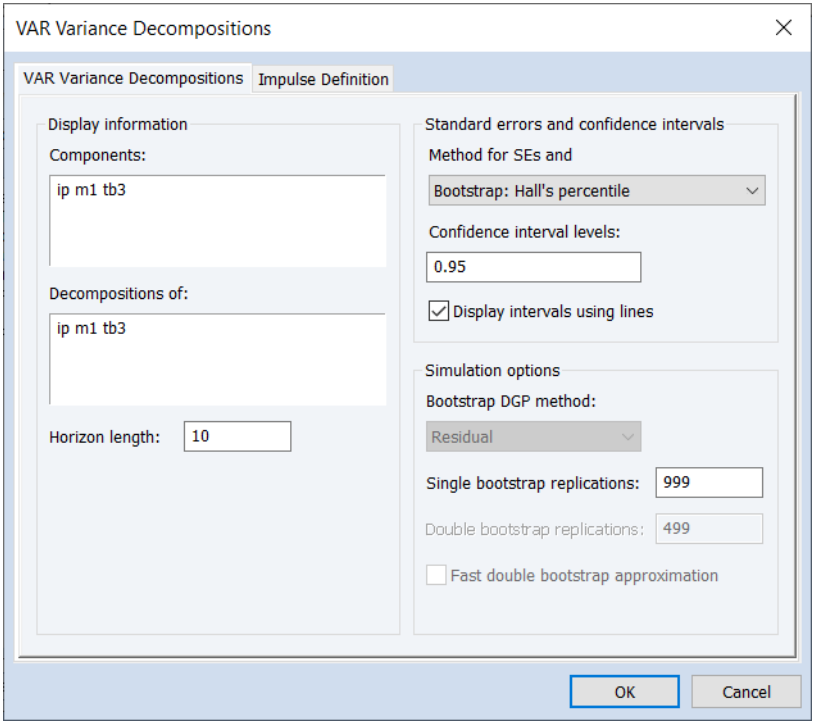
Preliminary documentation is available:

- See "Impulse Response Confidence Intervals" on page 2033 in *User's Guide II* for further discussion of confidence interval methods.
- See `Var::impulse` (p. 1175) in *User's Guide II* for command documentation.

Variance Decomposition CIs

In prior versions of EViews, confidence intervals (CI) and standard errors (SEs) for variance decomposition were only available for reduced form VAR models and only via monte carlo simulation.

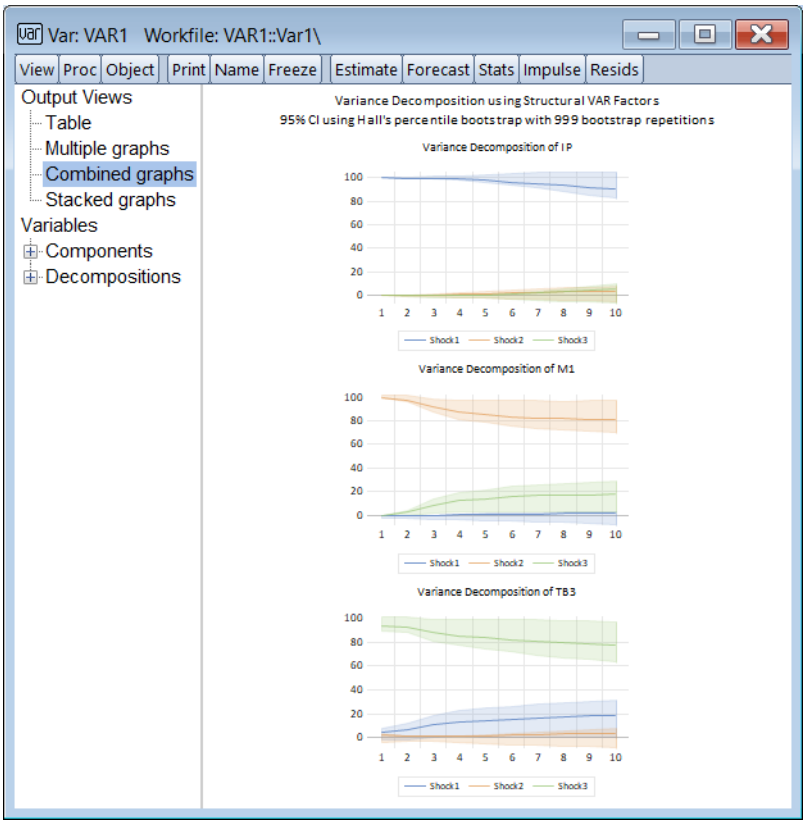
EViews 14 now offers its full suite of simulation-based CI methods for variance decomposition of both reduced form and structural models:



To perform the variance decomposition, select **View/Variance decomposition...** from the main menu of an estimated VAR.

You will be presented with a choice of **Method for SEs and CIs**. You may choose between none, Monte Carlo, and four different bootstrap methods, two of which feature a double bootstrap with optional fast double bootstrap algorithm. Depending on your choice, you will be presented with a variety of different options including the number of bootstrap replications.

Notably, all of these options are available, even if you elect to use structural impulses computed from the SVAR estimates..



Preliminary documentation is available:

- See "Impulse Response Confidence Intervals" on page 2033 in *User's Guide II* for further discussion of confidence interval methods.
- See `Var::vdecomp` (p. 1215) in *User's Guide II* for command documentation.

BTVCVAR Fixed Horizon Impulse-Responses

EViews 14 offers powerful new tools for computing impulse-responses when working with the Bayesian time-varying VAR coefficients model.

The time-varying coefficient VAR (TVCVAR) relaxes the constant parameter restriction of conventional VARs, allowing for ongoing changes in model parameters over time. Unfortunately, allowing for time-varying coefficients exacerbates the problems associated with estimating VARs with large number of parameters.

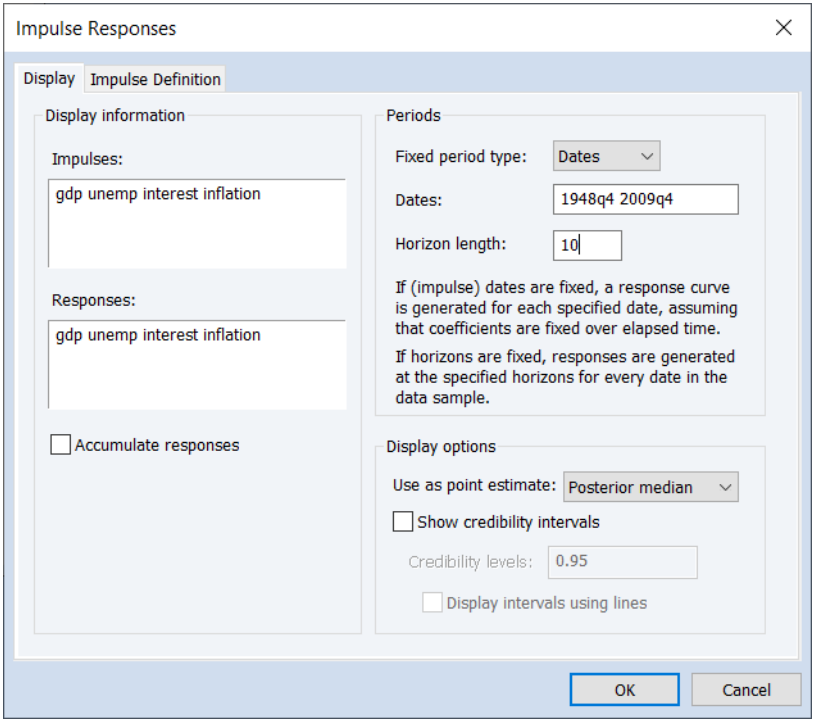
Once estimated, the time-varying nature of coefficients introduce some complications into the analysis of the coefficients of the VAR notably in the analysis of impulse-responses.

Importantly, the TVCVAR impulse response is a function of both the timing (date) of the shock and the elapsed time since the shock. Timing matters for a TVCVAR because the VAR coefficients are different at different dates, so that the system responds differently to a given stimulus.

Prior versions of EViews produced tables and graphs of cross-sectional BTVCVAR IRFs at fixed dates. When performing impulse-response analysis, you were prompted to provide one or more dates at which to compute the IRFs, and a separate IRF was computed for each date.

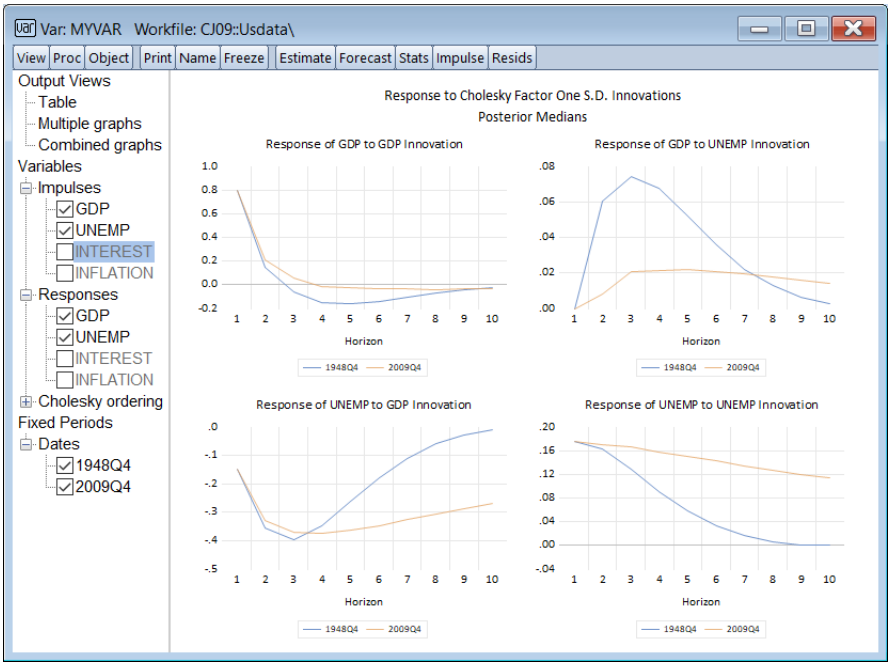
EViews 14 offers new tools for producing BTVCVAR IRFs at fixed horizons, allowing you to examine how the system response at a given elapsed time has changed with changes in the impulse date. You will be prompted to provide one or more horizons, and EViews will compute the ensemble of IRFs and will extract and display the required results.

Click on the **Impulse** button on the toolbar of an estimated BTVCVAR or select **View/ Impulse response...** from the menu, EViews will display the impulse response dialog:



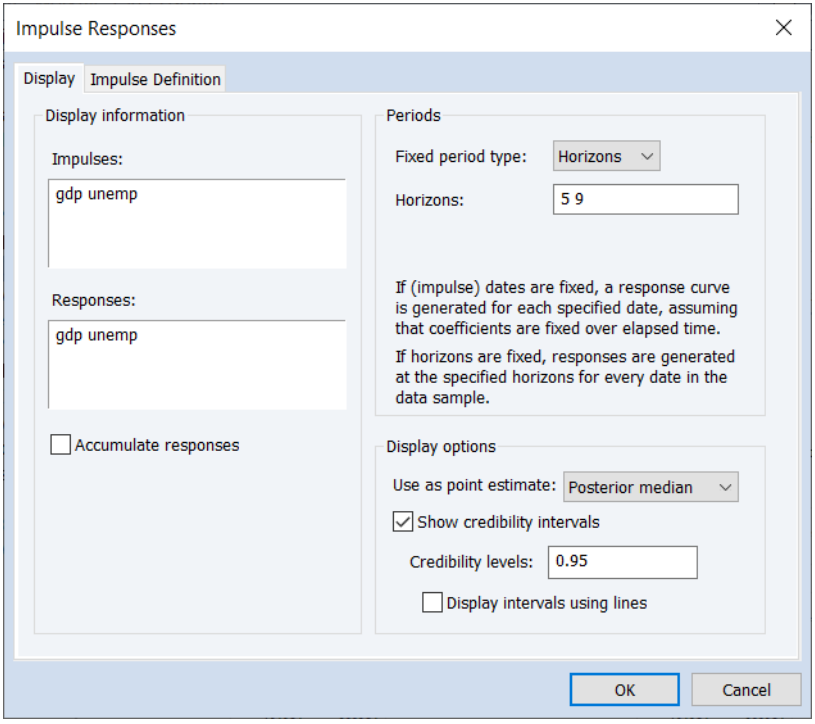
You may use the **Fixed period type** to choose whether to display IRFs at different dates (**Dates**), or do display IRFs collected at different horizons (**Horizons**).

In the dialog depicted above we specify **Dates** and ask for 10 period IRFs computed using the coefficients at two different dates (1948q4 and 2009q4):

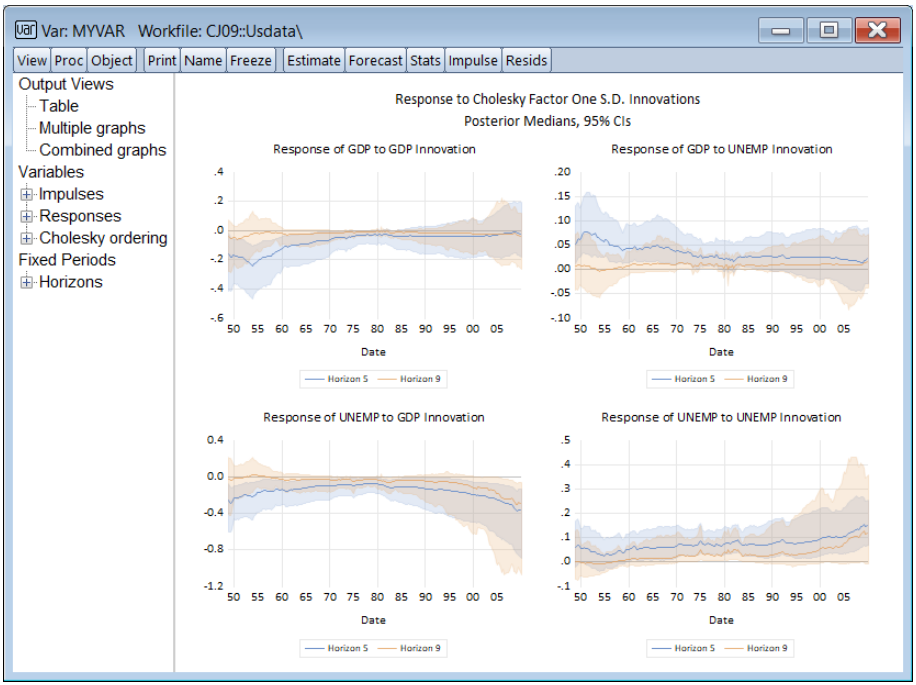


Depicted are standard 10 period impulse responses for GDP and UNEMP for the two periods of interest.

Alternately, we may specify **Horizons** and ask for the results of impulse responses computed at each period in the estimation sample for that horizon.



Here, we are asking EViews to compute the entire time-path of impulse responses at horizons of 5 and 9 periods:



Here we see the time-paths of the 5 and 9 horizon impulse responses for GDP and UNEMP.

For preliminary documentation:

- See "Impulse Response" on page 2106 in *User's Guide II* for further discussion of confidence interval computation and display in BTVCVAR models.
- See `Var::impulse` (p. 1175) in *User's Guide II* for command documentation.

Matrix Statistical Tools

New Matrix Object Views and Procs

Prior versions of EViews offered somewhat limited statistical support for data held in vectors and matrices.

While basic statistical and mathematical functions like means and variances, mathematical function evaluation, and matrix algebra were all supported, a number of the analytic and utility views and procedures that were available for series and groups were not available for matrices and vectors. You could not, for example, easily test data in a vector for normality, or examine a one-way tabulation of the values in a vector.

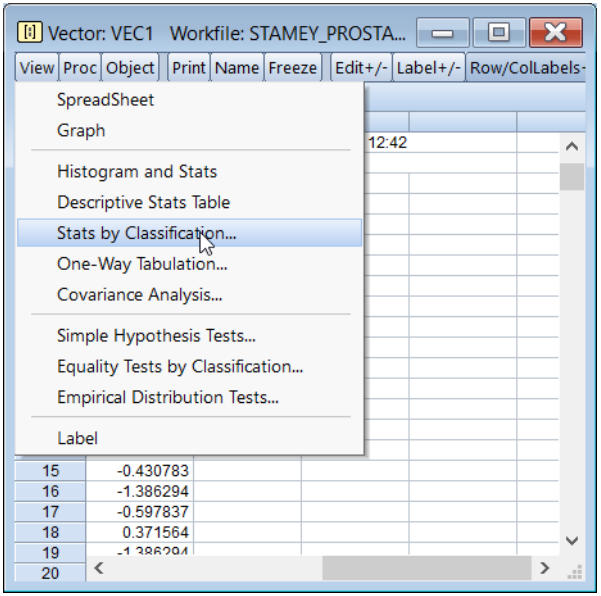
In a number of settings, we found that having the ability to say, perform interactive analysis of data in a matrix or vector would be quite useful. While the matrix data could be exported to series or a new workfile, this was obviously inconvenient.

EViews 14 expands significantly the number of tools available for working with the data in vectors and matrices.

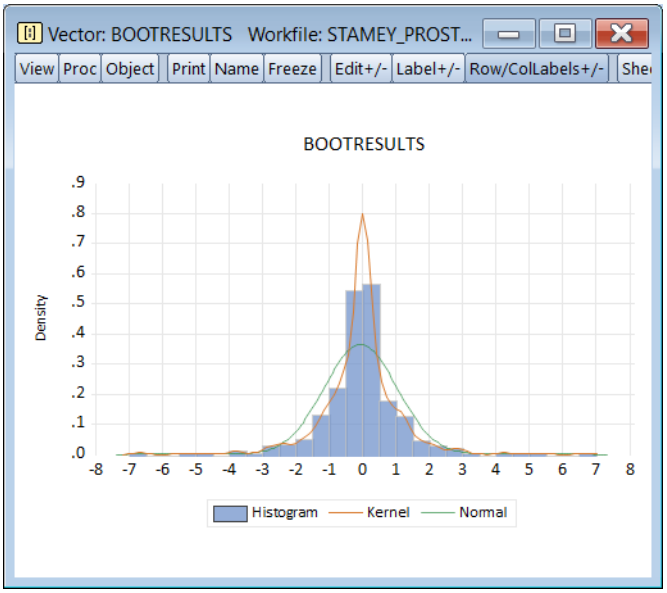
New Vector Views and Procs

For vectors you may now employ:

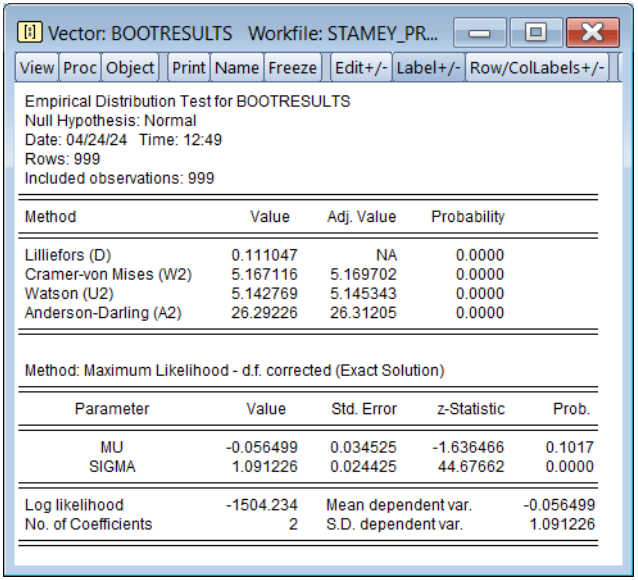
- histogram and stats view
- statistics by classification view
- one-way frequency tables
- parametric and non-parametric measures of variability
- simple hypothesis tests
- equality tests by classification
- empirical distribution tests
- resampling
- creating classification vectors
- making distribution function data (for example, saving kernel density data)



For example, if you have results from a bootstrap simulation stored in a vector, it is now easy to produce both a custom distribution graph,



and perform empirical distribution function tests



For updated command documentation see:

Vector / Rowvector

classify.....recode vector into classes defined by a grid, specified limits, or quantiles (p. 1224).

distdata.....save a matrix containing distribution plot data computed from the series (p. 1232).

edftest.....empirical distribution function tests (p. 1233).

freq.....one-way tabulation (p. 1239).

hist.....descriptive statistics and histogram (p. 1241).

resample.....resample from the observations in the vector (p. 1251).

statby.....statistics by classification (p. 1258).

stats.....descriptive statistics of the data in the vector (p. 1260).

testby.....equality test by classification (p. 1261).

teststat.....simple hypothesis tests (p. 1262).

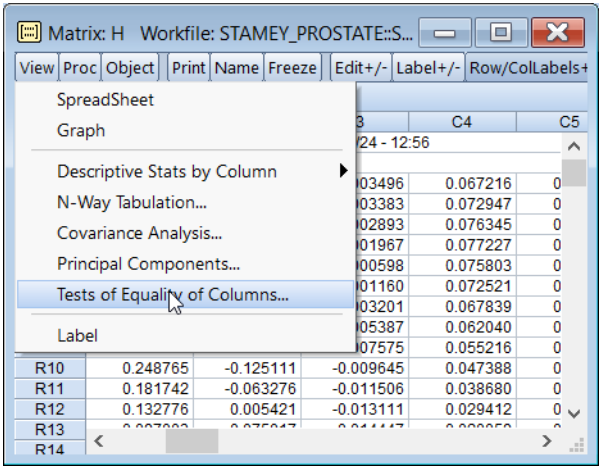
Svector

freq.....one-way tabulation (p. 977).

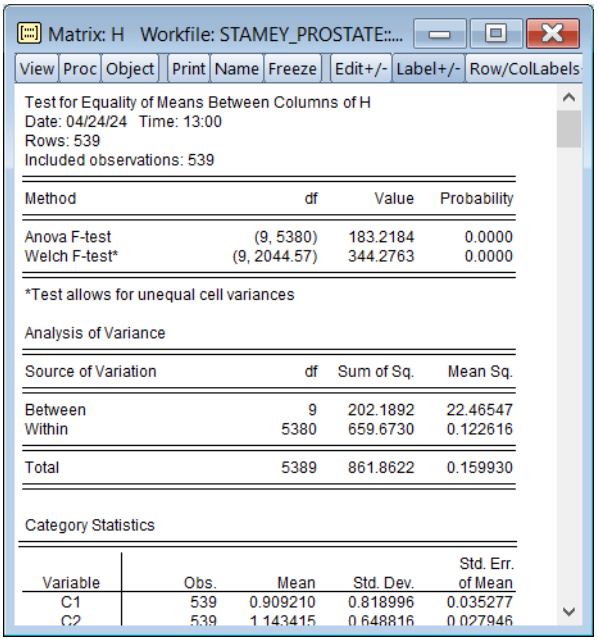
New Matrix Views and Procs

Similarly, for matrices, you may now perform:

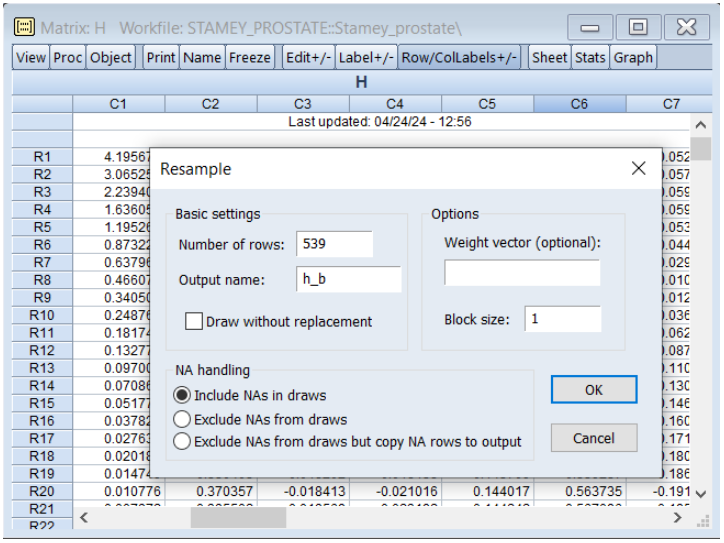
- descriptive statistics
- n-way cross-tabulation
- tests of equality of column statistics
- resampling from the rows of the matrix
- making distribution function data



For example, we can test for equality of the means between the columns of a matrix,



and resample from the rows of the matrix



For updated command documentation see:

distdata.....save a matrix containing distribution plot data computed from the matrix (p. 568).
freq.....n-way contingency table (p. 574).
resample.....resample from rows of the matrix (p. 593).
stats.....descriptive statistics for each column of the matrix (p. 600).
testbtw.....tests of equality for mean, median, or variance between series in group (p. 602).

Models

Models Containing Future Endogenous Values

EViews 14 allows you to solve models which contain future values of endogenous variables using standard Gauss-Seidel, and E-Newton or E-QNewton methods (Brayton, 2011).This new capability is central finding solutions to models involving rational expectations.

Consider a model where the equations have the form:

$$F(y(-\text{maxlag}), \ldots, y(-1), y, y(1), \ldots, y(\text{maxlead}), x) = 0 \tag{0.5}$$

where F is the complete set of equations of the model, y is a vector of all the endogenous variables, x is a vector of all the exogenous variables, and the parentheses follow the usual EViews syntax to indicate leads and lags.

Since solving the model for any particular period requires both past and future values of the endogenous variables, it is not possible to solve the model recursively in one pass. Instead, the equations from all the periods across which the model is to be solved must be treated as a simultaneous system, and solving the model will require terminal as well as initial conditions.

For example, in the case with a single lead and a single lag and a sample that runs from s to t, we must effectively solve the entire stacked system:

$$\begin{aligned} F(y_{s-1}, y_s, y_{s+1}, x) &= 0 \\ F(y_s, y_{s+1}, y_{s+2}, x) &= 0 \\ F(y_{s+1}, y_{s+2}, y_{s+3}, x) &= 0 \\ &\cdots \\ F(y_{t-2}, y_{t-1}, y_t, x) &= 0 \\ F(y_{t-1}, y_t, y_{t+1}, x) &= 0 \end{aligned} \tag{0.6}$$

where the unknowns are y_s, y_{s+1}, ..., y_t the initial conditions are given by y_{s-1} and the terminal conditions are used to determine y_{t+1}. Note that if the leads or lags extend more than one period, we will require multiple periods of initial or terminal conditions.

EViews provides three methods for solving this class of models: Gauss-Seidel, E-Newton, and E-QNewton. All three are iterative procedures that attempt to reduce the change in the endogenous variables, Δy_s, Δy_{s+1}, ..., Δy_t, to zero as the model's equations are solved repeatedly.

Gauss-Seidel

The first algorithm, Gauss-Seidel, loops through every observation in the forecast sample and at each observation solves the model while treating the past and future values as fixed. The loop is repeated until changes in the values of the endogenous variables between successive iterations become less than a specified tolerance. In essence, discrepancies between the future value of each endogenous variable and the recalculated value of that variable are diffused forwards and backwards through the observations until the discrepancies vanish, assuming the algorithm converges.

Although the Gauss-Seidel method is not guaranteed to converge, failure to converge is often a sign of model instability which results when the influence of the past or the future on the present does not die out as the length of time considered is increased. Such instability is often undesirable for other reasons and may indicate a poorly specified model.

This method is often referred to as the Fair-Taylor method, although the Fair-Taylor algorithm includes a particular handling of terminal conditions (the extended path method) that is slightly different from the options provided by EViews. When solving the model, EViews allows the user to specify fixed end conditions by providing values for the endogenous variables beyond the end of the forecast sample, or to determine the terminal conditions endogenously by adding extra equations for the terminal periods which impose either a constant level, a linear trend, or a constant growth rate on the endogenous variables for values beyond the end of the forecast period.

E-Newton and E-QNewton

The second and third methods, E-Newton and E-QNewton (Brayton, 2011), apply the well-known Newton and Broyden methods (respectively) to the problem of finding endogenous variable values such that $\Delta y = 0$. Both algorithms repeatedly construct a linear approximation to the stacked system, use that approximation to adjust the endogenous variables, and update the approximation. These methods involve calculation of the Jacobian of Δy, or an approximation thereof, and are thus more computationally taxing than Gauss-Seidel.

Weighted against the additional computational cost, these two approaches have the advantage of robustness and applicability to a broader range of models. Small models and those with few future values are frequently solved more efficiently by E-Newton, while large models or those with many future values are solved more efficiently by E-QNewton.

The E-Newton and E-QNewton methods have been implemented as EViews programs and distributed as part of the Federal Reserve's large-scale economic model, FRB/US, in the form

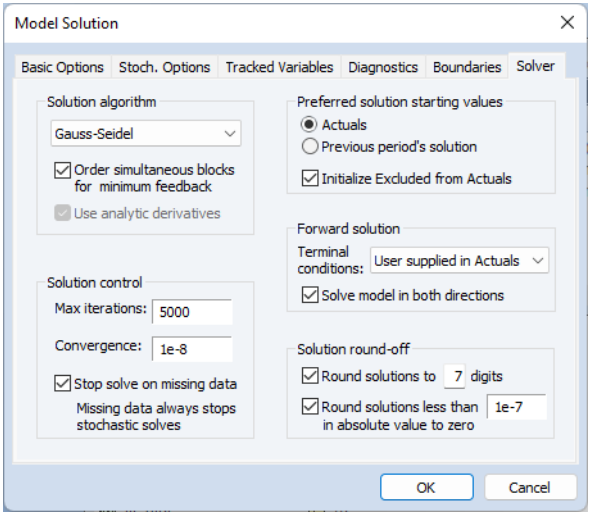
of the MCE_SOLVE_LIBRARY. There are a few notable differences between the MCE_-SOLVE_LIBRARY and the EViews implementations.

- The MCE_SOLVE_LIBRARY implementation uses a simplifying assumption that future value expressions in the model are strictly linear (option *jinit* set to "linear"). This option enables Jacobian matrix construction to be performed in a particularly efficient way. However, EViews allows for more general future dependence and currently does not support this feature so that it may take substantially longer to solve this class of linear models.
- The MCE_SOLVE_LIBRARY implementation allows for changing of endogenous variable values at observations outside the solve sample. EViews follows its standard approach and only solves for endogenous variable values at observations within the solve sample.

Forward Solution Options

You may click on the **Solve** button on the model toolbar, or select **Proc/Solve Model...** from the main model object menu to display the solution options.

Click on the **Solve** tab to display the corresponding dialog page. The **Solver** dialog page sets options relating to the non-linear equation solver which is applied to the model:



The **Solution algorithm** box lets you select the algorithm that will be used to solve simultaneous blocks within each period and futures values across all periods (if present). The following choices are available:

- **Gauss-Seidel**: the Gauss-Seidel algorithm is an iterative algorithm, where at each iteration we solve each equation in the model for the value of its associated endogenous variable, treating all other endogenous variables as fixed.

  The Gauss-Seidel algorithm requires little working memory and has fairly low computational costs, but requires the equation system to have certain stability properties for it to converge. Although it is easy to construct models that do not satisfy these properties, in practice, the algorithm generally performs well on most econometric models. If you are having difficulties with the algorithm, you may try reordering the equations, or rewriting the equations to change the assignment of endogenous variables, since these changes can affect the stability of the Gauss-Seidel iterations. (See "Gauss-Seidel," on page 2584.)

- **Newton**: Newton's method is an iterative method, where at each iteration we take a linear approximation to the model, then solve the linear system to find a root of the model.

  The Newton algorithm can handle a wider class of problems than Gauss-Seidel, but requires considerably more working memory and has a much greater computational cost when applied to large models. Newton's method is invariant to equation reordering or rewriting. (See "Newton's Method," on page 2585.)

  When this method is selected, Newton's method is used to solve simultaneous blocks but Gauss-Seidel is used to solve future values. (See "Models Containing Future Endogenous Values" on page 91.)

- **Broyden**: Broyden's method is a modification of Newton's method (often referred to as a quasi-Newton or secant method) where an approximation to the Jacobian is used when linearizing the model rather than the true Jacobian which is used in Newton's method. This approximation is updated at each iteration by comparing the equation residuals obtained at the new trial values of the endogenous variables with the equation residuals predicted by the linear model based on the current Jacobian approximation.

  Because each iteration in Broyden's method is based on less information than in Newton's method, Broyden's method typically requires more iterations to converge to a solution. Since each iteration will generally be cheaper to calculate, however, the total time required for solving a model by Broyden's method will often be less than that required to solve the model by Newton's method.

Note that Broyden's method retains many of the desirable properties of Newton's method, such as being invariant to equation reordering or rewriting. (See "Broyden's Method," on page 2586.)

When this method is selected, Broyden's method is used to solve simultaneous blocks but Gauss-Seidel is used to solve future values.

- **E-Newton**: Simultaneous blocks are solved using Broyden's method and future values are solved using Newton's method.
- **E-QNewton**: Both simultaneous blocks and future values are solved using Broyden's method.

The **Forward solution** section allows you to adjust options that affect how the model is solved when one or more equations in the model contain future (forward) values of the endogenous variables.

- The **Terminal conditions** section lets you specify how the values of the endogenous variables are determined for leads that extend past the end of the forecast period:

  If **User supplied in Actuals** is selected, the values contained in the Actuals series after the end of the forecast sample will be used as fixed terminal values. If no values are available, the solver will be unable to proceed.

  If **Constant level** is selected, the terminal values are determined endogenously by adding the condition to the model that the values of the endogenous variables are constant over the post-forecast period at the same level as the final forecasted values ($y_t = y_{t-1}$ for $t = T, T+1, \ldots, T+k-1$), where T is the first observation past the end of the forecast sample, and k is the maximum lead in the model). This option may be a good choice if the model converges to a stationary state.

  If **Constant difference** is selected, the terminal values are determined endogenously by adding the condition that the values of the endogenous variables follow a linear trend over the post forecast period, with a slope given by the difference between the last two forecasted values:

  $$y_t - y_{t-1} = y_{t-1} - y_{t-2} \tag{0.7}$$

  for $t = T, T+1, \ldots, T+k-1$). This option may be a good choice if the model is in log form and tends to converge to a steady state.

  If **Constant growth rate** is selected, the terminal values are determined endogenously by adding the condition to the model that the endogenous variables grow exponentially over the post-forecast period, with the growth rate given by the growth between the final two forecasted values:

  $$(y_t - y_{t-1})/y_{t-1} = (y_{t-1} - y_{t-2})/y_{t-2} \tag{0.8}$$

  for $t = T, T+1, \ldots, T+k-1$).

This latter option may be a good choice if the model tends to produce forecasts for the endogenous variables which converge to constant growth paths.

- The **Solve in both directions** option affects how the solver loops over periods when calculating forward solutions. When the box is not checked, the solver always proceeds from the beginning to the end of the forecast period during the Gauss-Seidel iterations. When the box is checked, the solver alternates between moving forwards and moving backwards through the forecast period.

  The two approaches will generally converge at slightly different rates depending on the level of forward or backward persistence in the model. You should choose whichever setting results in a lower iteration count for your particular model.

Updated command documentation may be found at:

solveopt ..... set solve options for model (p. 643). *(updated)*

in the *Object Reference*.

Command Language

Updated Object List

(Unless otherwise specified, all of the object views and procedures are in *Object Reference*.)

Alpha

Alpha Views

freq ..... n-way contingency table (p. 12). (*new*)

Equation

Equation Methods

arch ..... autoregressive conditional heteroskedasticity (ARCH and GARCH) (p. 64). (*updated*)

ardl ..... least squares with autoregressive distributed lags (p. 71). (*updated*)

enet ..... elastic net regression (including Lasso and ridge regression) (p. 126). (*updated*)

varsel ..... equation estimation using least squares with variable selection (uni-directional, stepwise, swapwise, combinatorial, Auto-GETS, Lasso) (p. 260). (*updated*)

Equation Views

coefpath ..... display graphs of the paths of the coefficients plotted against lambda, fit measures, and estimation values in elastic net, ridge, Lasso, and variable selection using Lasso models (p. 95). (*new*)

cvgraph.....display a graph of the cross-validation objective against the lambda path for elastic net, ridge, Lasso, and variable selection using Lasso models (p. 115). (*new*)

fitoutliers .....detect outliers in the residuals or regressors of the equation (p. 137). (*new*)

icgraph .....display a graph of the selection criterion for the top 20 models as determined by model selection during estimation (p. 170). (*updated*)

ictable.....display a table of the log-likelihood and selection criteria for the top 20 models as determined by model selection during estimation (p. 171).(*updated*)

lambdacoefs.....display the spreadsheet of the matrix of coefficient values along the lambda path in elastic net, ridge, Lasso, and variable selection using Lasso models (p. 175). (*new*)

lambdaest .....display the table showing various values associated with estimation along the lambda path in elastic net, ridge, Lasso, and variable selection using Lasso models (p. 176). (*new*)

lambdafit .....display the table showing various fit statistics associated with estimates along the lambda path in elastic net, ridge, Lasso, and variable selection using Lasso models (p. 177). (*new*)

lambdapath.....display graphs of lambda against various fit and estimation measures in elastic net, ridge, Lasso, and variable selection using Lasso models (p. 178). (*new*)

modselgraph .....display a graph of the selection criteria for the top 20 models for elastic net, ridge, Lasso, and variable selection using Lasso models (p. 205). (*new*)

modseltable.....display a table of the selection criteria and measures associated with the estimation and model selection of elastic net, ridge, Lasso, and variable selection using Lasso models (p. 206). (*new*)

qrcrprocess .....displays a spool object producing a quantile process of the cointegrating relation (p. 216). (*new*)

qrecprocess .....displays a spool object producing a quantile process for each of the conditional error correction and error correction coefficients (p. 217). (*new*)

Equation Data Members

@cvconverge .....Elastic net path cross-validation convergence test values (lambda values in rows; lambda in first column, training-test sample results in remaining columns).

@cvisvalid ..... Elastic net path cross-validation valid results indicators (lambda values in rows; lambda in first column, training-test sample results in remaining columns).

@cviters ..... Elastic net path cross-validation iterations (lambda values in rows; lambda in first column, training-test sample results in columns).

@cvobjective ..... Elastic net path cross-validation objective values (lambda values in rows; lambda in first column, training-test sample results in remaining columns).

@modselresults .... Elastic net path model selection summary (lambda values in rows; lambda in first column, followed by model selection objective, number of non-zero coefficients, and the fit statistics (sum-of-squared residuals, mean-square error, R-squared, and adjusted R-squared) associated with the estimated model.

@lambdacoefs ..... Elastic net lambda path coefficients matrix (lambda values in rows; variables in columns). Full set of variables including those with zero coefficients along the path.

@lambdaest ..... Elastic net lambda path estimation measures matrix (lambda values in rows; columns contain the lambda values, number of non-zero coefficients, estimation objective, sums-of-squares portion of the objective, L_1 portion of the objective, L_2^2 portion of the objective).

@lambdafit ..... Elastic net lambda path fit measures matrix (lambda values in rows; columns contain the lambda values, number of non-zero coefficients, R-squared, adjusted R-squared, and sums-of-squared residuals).

@lambdapath ..... Elastic net lambda path vector.

Group

Group Views

freq ..... n-way contingency table (p. 475). (*updated*)

stats ..... descriptive statistics for each series in the group (p. 513). (*updated*)

testbtw ..... tests of equality for mean, median, or variance between series in group (p. 514). (*new*)

Group Procs

distdata ..... save a matrix containing distribution plot data computed from the group (p. 472). (*new*)

resample ..... resample from series in the group (p. 493). (*new*)

Matrix

Matrix Views

freq ..... n-way contingency table (p. 574). (*new*)

Series

Series View

bubbletest ..... perform tests on the existence of a bubble in the series (p. 769). (*new*)

changepoints ..... perform tests on a change in the location parameter (mean) of the series (p. 773). (*new*)

edftest ..... empirical distribution function tests (p. 786). (*updated*)

freq ..... one-way tabulation (p. 795). (*updated*)

hist ..... descriptive statistics and histogram (p. 798). (*updated*)

outliers ..... detect outlying observations (p. 818). (*new*)

statby ..... statistics by classification (p. 853). (*updated*)

stats ..... descriptive statistics of the data in the series (p. 855).(*updated*)

testby ..... equality test by classification (p. 857). (*updated*)

teststat ..... simple hypothesis tests (p. 858). (*updated*)

trendtests ..... perform tests on the existence of a trend in the series (p. 862). (*new*)

Series Procs

classify ..... recode vector into classes defined by a grid, specified limits, or quantiles (p. 774).(*new*)

distdata ..... save a matrix containing distribution plot data computed from the series (p. 780).(*new*)

hpf ..... Hodrick-Prescott filter (p. 799). (*updated*)

jdemetra ..... executes the JDemetra+ seasonal adjustment routine on the series (p. 801).(*new*)

prophet ..... performs Facebook's Prophet forecasting on the underlying series (p. 825).(*new*)

resample ..... resample from the observations in the series (p. 826).(*new*)

Svector

Svector Views

freq ..... one-way tabulation (p. 977). (*new*)

Svector Values

@irow(*arg*) ..... Returns the indices for the rows defined by *arg* where *arg* is a string or svector of strings. The strings correspond to row labels so that *arg* = "2" specifies the first row labeled "2".

Sym

Sym Data Members

@icol(*arg*) .....Returns the indices for the columns defined by *arg* where *arg* is a string or svector of strings. The strings correspond to column labels so that *arg* = "2" specifies the first column labeled "2".

@irow(*arg*) .....Returns the indices for the rows defined by *arg* where *arg* is a string or svector of strings. The strings correspond to row labels so that *arg* = "2" specifies the first row labeled "2".

Var

Var Views

impulse.....impulse response functions (p. 1175). (*updated*)

vdecomp .....variance decomposition (p. 1215). (*updated*)

Vector

Vector Views

edftest.....empirical distribution function tests (p. 1233).(*new*)

freq.....one-way tabulation (p. 1239).(*new*)

hist .....descriptive statistics and histogram (p. 1241).(*new*)

statby.....statistics by classification (p. 1258).(*new*)

stats.....descriptive statistics of the elements of the vector (p. 1260).(*new*)

testby.....equality test by classification (p. 1261).(*new*)

teststat .....simple hypothesis tests (p. 1262).(*new*)

Vector Procs

classify.....recode vector into classes defined by a grid, specified limits, or quantiles (p. 1224).(*new*)

distdata.....save a matrix containing distribution plot data computed from the vector (p. 1232).(*new*)

resample .....resample from the rows of the vector (p. 1251).(*new*)

Vector Data Members

@irow(*arg*) .....Returns the indices for the rows defined by *arg* where *arg* is a string or svector of strings. The strings correspond to row labels so that *arg* = "2" specifies the first row labeled "2".

Updated Function List

(Unless otherwise specified, all of the object views and procedures are in *Command and Programming Reference*.)

Mathematical and Statistical Functions

@gammainc ..... Incomplete gamma function (p. 890). (*updated*)
@imax ..... Index of maximum value (p. 913). (*updated*)
@imaxes ..... Indices of maximum value (multiple) (p. 915). (*new*)
@imin ..... Index of minimum value (p. 916). (*updated*)
@imins ..... Indices of minimum value (multiple) (p. 916). (*new*)
@logit ..... Logistic transform (p. 948). (*updated*)
@maxes ..... Maximum values (multiple) (p. 964). (*new*)
@mins ..... Minimum values (multiple) (p. 974). (*new*)
@mmedian ..... Trailing moving median (ignore NAs) (p. 978). (*new*)
@movmedian ..... Trailing moving median (propagate NAs) (p. 990). (*new*)
@valcount ..... Number of matching values (p. 1174). (*new*)

String Functions

- Existing string functions have been enhanced to support vector arguments.

@winsert ..... Insert string into string after word (p. 1195). (*new*)

Date Functions

- Existing string functions have been enhanced to support vector arguments.

Matrix Functions

@cvalcount ..... Count of matching values in each column (p. 858).
@ebtw ..... Element by element test for whether values are between two other values (p. 858). (*new*)
@implodeu ..... Creates sym from upper triangle of square matrix (p. 918). (*new*)
@incr ..... Increment the columns or rows of a matrix using a vector or rowvector (p. 918). (*new*)
@lower ..... Lowercase representation of a string, or lower triangular matrix of a matrix (p. 949). (*updated*)
@upper ..... Uppercase representation of a string; or upper triangular matrix of a matrix (p. 1167). (*updated*)

EViews Compatibility Notes

The following discussion describes EViews 14 compatibility issues for users of earlier versions.

Workfile Compatibility

With few exceptions, EViews 14 workfiles are backward compatible with EViews 13. Note that the following are new or have been modified in Version 13:

- Estimation objects estimated with methods that employ new or significantly updated features (quantile ARDL, elastic net, variable selection with Lasso, MIDAS GARCH).

If you have saved workfiles containing any of the above objects and open them in earlier versions, EViews will delete the incompatible object and notify you that one or more objects were not read. If you then save the workfile, you will lose the objects. We recommend that you make a copy of any workfiles that contain these objects if you would like to use these workfiles in earlier versions of EViews.

Note also that some estimation objects written in earlier versions of EViews will only be partially read into EViews 14:

- Equations estimated using elastic net or variable selection with Lasso will require re-estimation before use.