

Importing Data in EasyReg International

Herman J. Bierens

Pennsylvania State University

October 3, 2009

1. *About EasyReg International*

EasyReg (**Easy Regression**) International is a free econometrics software package, which can be downloaded from web page <http://econ.la.psu.edu/~hbierens/EASYREG.HTM>

EasyReg opens with the following window, containing the menus:



Picture 1: EasyReg front window

The date displayed in Picture 1 is the compilation date, i.e. the date of the installation files. However, the displayed date of April 1, 2007, is not the current date. The most recent version of EasyReg is dated October 1, 2009, which has many improvements over previous versions.

It is possible that some upgrades have not yet been included in the current version. Therefore, after installing EasyReg, it is recommended that you upgrade it, via menu item “WWW > Upgrade EasyReg to the latest version.”. At the bottom of the EasyReg web page you will find a link to a web page which lists the upgrades not yet included in the current version.

EasyReg conducts a wide variety of econometric estimation, testing and forecasting tasks on all 32 bit Windows platforms, simply by point-and-click. EasyReg is designed for use in empirical research (including my own), and for teaching econometrics. In the latter case the user can choose his or her own econometrics level.

EasyReg is called International because it accepts dots and/or commas as decimal delimiters, regardless of the local number setting of Windows. Therefore, EasyReg runs everywhere in the world without need to adjust the setting of Windows.

However, if your Windows uses a language with a non-roman alphabet such as Japanese and Chinese, and EasyReg seems to be frozen, the problem is likely due to the language setting for non-Unicode programs. If so, open Control Panel, select "Regional and Language Options", and on the "Advanced" tab, under "Language for non-Unicode programs", select English. This will fix the problem without affecting the language of Windows itself.

When you run EasyReg for the first time, most menu items are disabled, because they need data to work. Therefore, in this paper I will show now how to import data in EasyReg and how to transform them.

2. *EasyReg data files and how to import them*

EasyReg can import two types of data files, Microsoft Excel files in CSV format, and EasyReg data files in space delimited text format. I will discuss the latter type first.

2.1 *EasyReg data files in space delimited text format*

This type of file is a (Notepad or Wordpad) text file with the following format. The first line contains two numbers, the number of variables ($= k$), and the missing value code ($= m$), separated by at least one space. The next k lines contain the names of the k variables involved. The file names may contain any

character, including spaces, commas, etc., and there is no restriction on their length. However, it is recommended to keep the variable names short, say no more than ten characters. The rest of the file contains the data entries $x(j,i)$ for observation j on variable i , where $i = 1, \dots, k, j = 1, \dots, n$, separated by one or more spaces, where n is the number of observations. Data entries with value m are interpreted as missing values, except if $m = 0$, which indicates that there are no missing values. Therefore, the missing value code should be chosen such that the real data entries will never take the value m , for example by choosing $m = -99999.99$.

You may use a dot or a comma as decimal delimiter for the numerical data entries. For example, $m = -99999,99$ is also a valid missing value code. EasyReg will automatically replace a decimal comma with a decimal dot when it reads numbers. However, do not use a grouping symbol, because otherwise EasyReg cannot interpret the number involved. For example, do not use numbers with a grouping symbol like 34.123,75 (European style) or 34,123.75 (US style), but instead write them as 34123,75 or 34123.75, respectively.

Thus, the structure of this data file is:

```

k m
Name of variable 1
      :
Name of variable k
x(1,1) x(1,2) ... x(1,k)
      :      :      :      :
x(n,1) x(n,2) ... x(n,k)

```

For example, consider the following data file in Table 1 below, containing quarterly data on real consumption, GDP and investment in the US, from the first quarter of 1985 to the first quarter of 1995, together with the quarters themselves. The last two quarters contain missing values, indicated by the missing value code -99999.99 .

The inclusion of the time “variable” quarter is optional. As we will see, once you import a data file, EasyReg will ask for the type of data, either cross-section, annual time series, quarterly time series, monthly time series, or other time series, and in the time series cases the year and eventually quarter, month or other frequency (for example bi-annual) of the first observation. Anyhow, if you include a time variable, or observation number in the case of cross-section data, this variable should be a valid numerical data entry.

Table 1: *Example of a space delimited EasyReg data file*

```

4 -99999.99
Quarter
Real US Consumption
Real US GDP
Real US Investment
1985.1 2824.90 4221.80 717.80
1985.2 2849.70 4254.80 724.60
1985.3 2893.30 4309.00 719.90
1985.4 2895.30 4333.50 732.90
1986.1 2922.40 4390.50 728.20
1986.2 2947.90 4387.70 728.10
1986.3 2993.70 4412.60 723.80
1986.4 3012.50 4427.10 725.90
1987.1 3011.50 4460.00 706.80
1987.2 3046.80 4515.30 718.30
1987.3 3075.80 4559.30 733.00
1987.4 3074.70 4625.50 733.90
1988.1 3128.20 4655.30 737.70
1988.2 3147.80 4704.80 753.30
1988.3 3170.60 4734.50 758.60
1988.4 3202.90 4779.70 764.10
1989.1 3203.60 4817.60 761.70
1989.2 3212.20 4839.00 757.50
1989.3 3235.30 4839.00 753.10
1989.4 3242.00 4856.70 744.60
1990.1 3264.40 4898.30 761.80
1990.2 3271.60 4917.10 745.80
1990.3 3288.40 4906.50 740.10
1990.4 3265.90 4867.20 716.60
1991.1 3242.90 4842.00 686.40
1991.2 3259.50 4867.90 683.40
1991.3 3269.80 4879.90 685.60
1991.4 3265.30 4880.80 684.40
1992.1 3311.40 4918.50 693.50
1992.2 3325.40 4947.50 721.30
1992.3 3357.60 4990.50 728.10
1992.4 3403.40 5060.70 748.60
1993.1 3417.20 5075.30 770.70
1993.2 3439.20 5105.40 787.30
1993.3 3472.20 5139.40 808.80
1993.4 3506.20 5218.00 851.70
1994.1 3546.30 5261.10 873.40
1994.2 3557.80 5314.10 891.70
1994.3 3584.70 5367.00 910.20
1994.4 -99999.99 -99999.99 -99999.99
1995.1 -99999.99 -99999.99 -99999.99

```

Although the data entries in Table 1 are nicely lined-up, there is no need for that. As long as the data entries are separated by a space, EasyReg can read them.

The current version of EasyReg can also read this file if, instead of spaces, tabs are used as data entry separators, provided these tabs are of the type compatible with Notepad or Wordpad tabs. EasyReg will automatically convert these tabs to spaces.

2.2 Excel files in CSV format

CSV stands for Comma-Separated Values. It is one of the formats for saving an Excel file. A CSV file is a text file (and can therefore be imported in Notepad and Wordpad) where the data entries are separated by commas. The data in Table 1 in CSV format is displayed in Table 2.

Table 2: *The data in Table 1 in Excel CSV format*

```
Quarter,Real US Consumption,Real US GDP,Real US Investment
1985.1,2824.9,4221.8,717.8
1985.2,2849.7,4254.8,724.6
1985.3,2893.3,4309,719.9
1985.4,2895.3,4333.5,732.9
1986.1,2922.4,4390.5,728.2
1986.2,2947.9,4387.7,728.1
1986.3,2993.7,4412.6,723.8
1986.4,3012.5,4427.1,725.9
1987.1,3011.5,4460,706.8
1987.2,3046.8,4515.3,718.3
1987.3,3075.8,4559.3,733
1987.4,3074.7,4625.5,733.9
1988.1,3128.2,4655.3,737.7
1988.2,3147.8,4704.8,753.3
1988.3,3170.6,4734.5,758.6
1988.4,3202.9,4779.7,764.1
1989.1,3203.6,4817.6,761.7
1989.2,3212.2,4839,757.5
1989.3,3235.3,4839,753.1
1989.4,3242,4856.7,744.6
1990.1,3264.4,4898.3,761.8
1990.2,3271.6,4917.1,745.8
1990.3,3288.4,4906.5,740.1
1990.4,3265.9,4867.2,716.6
1991.1,3242.9,4842,686.4
1991.2,3259.5,4867.9,683.4
1991.3,3269.8,4879.9,685.6
1991.4,3265.3,4880.8,684.4
1992.1,3311.4,4918.5,693.5
1992.2,3325.4,4947.5,721.3
1992.3,3357.6,4990.5,728.1
1992.4,3403.4,5060.7,748.6
1993.1,3417.2,5075.3,770.7
1993.2,3439.2,5105.4,787.3
1993.3,3472.2,5139.4,808.8
1993.4,3506.2,5218,851.7
1994.1,3546.3,5261.1,873.4
1994.2,3557.8,5314.1,891.7
1994.3,3584.7,5367,910.2
1994.4,,,
1995.1,,,
```

The first record contains the variable names, separated by commas. If a variable name contains a comma itself, the name involved has to be enclosed between quotation marks (“”). Older versions of Excel also enclose variable names between quotation marks if the name contains a space. EasyReg will read these files

as well.

Missing values in a CSV file are indicated by an empty data entry. Thus, two adjacent commas (,,) indicate that the data entry corresponding to the position after the first comma is a missing value. Similarly, a comma at the end of a data record indicates that the next data entry is a missing value, and a comma as first character of a data record indicates that the first data entry is a missing value.

The CSV format in Table 2 applies if Windows uses a dot as decimal delimiter, as in the US. If Windows uses a comma as decimal delimiter, as in continental Europe, the commas are replaced by semi-colons (;), and the decimal dots by commas. The CSV file then looks like this:

Table 3: *Excel CSV format if the decimal delimiter is a comma*

```
Quarter;Real US Consumption;Real US GDP;Real US Investment
1985,1;2824,9;4221,8;717,8
1985,2;2849,7;4254,8;724,6
1985,3;2893,3;4309;719,9
1985,4;2895,3;4333,5;732,9
1986,1;2922,4;4390,5;728,2
1986,2;2947,9;4387,7;728,1
1986,3;2993,7;4412,6;723,8
1986,4;3012,5;4427,1;725,9
1987,1;3011,5;4460;706,8
1987,2;3046,8;4515,3;718,3
1987,3;3075,8;4559,3;733
1987,4;3074,7;4625,5;733,9
1988,1;3128,2;4655,3;737,7
1988,2;3147,8;4704,8;753,3
1988,3;3170,6;4734,5;758,6
1988,4;3202,9;4779,7;764,1
1989,1;3203,6;4817,6;761,7
1989,2;3212,2;4839;757,5
1989,3;3235,3;4839;753,1
1989,4;3242;4856,7;744,6
1990,1;3264,4;4898,3;761,8
1990,2;3271,6;4917,1;745,8
1990,3;3288,4;4906,5;740,1
1990,4;3265,9;4867,2;716,6
1991,1;3242,9;4842;686,4
1991,2;3259,5;4867,9;683,4
1991,3;3269,8;4879,9;685,6
1991,4;3265,3;4880,8;684,4
1992,1;3311,4;4918,5;693,5
1992,2;3325,4;4947,5;721,3
1992,3;3357,6;4990,5;728,1
1992,4;3403,4;5060,7;748,6
1993,1;3417,2;5075,3;770,7
1993,2;3439,2;5105,4;787,3
1993,3;3472,2;5139,4;808,8
1993,4;3506,2;5218;851,7
1994,1;3546,3;5261,1;873,4
1994,2;3557,8;5314,1;891,7
1994,3;3584,7;5367;910,2
1994,4;;;
1995,1;;;;
```

EasyReg automatically checks whether Windows uses a dot or a comma as decimal delimiter, and assumes that the format of the CSV file is either that in Table 2 or in Table 3, respectively.

2.3 *Importing data files in EasyReg*

To import a data file, open “File > Get data”:

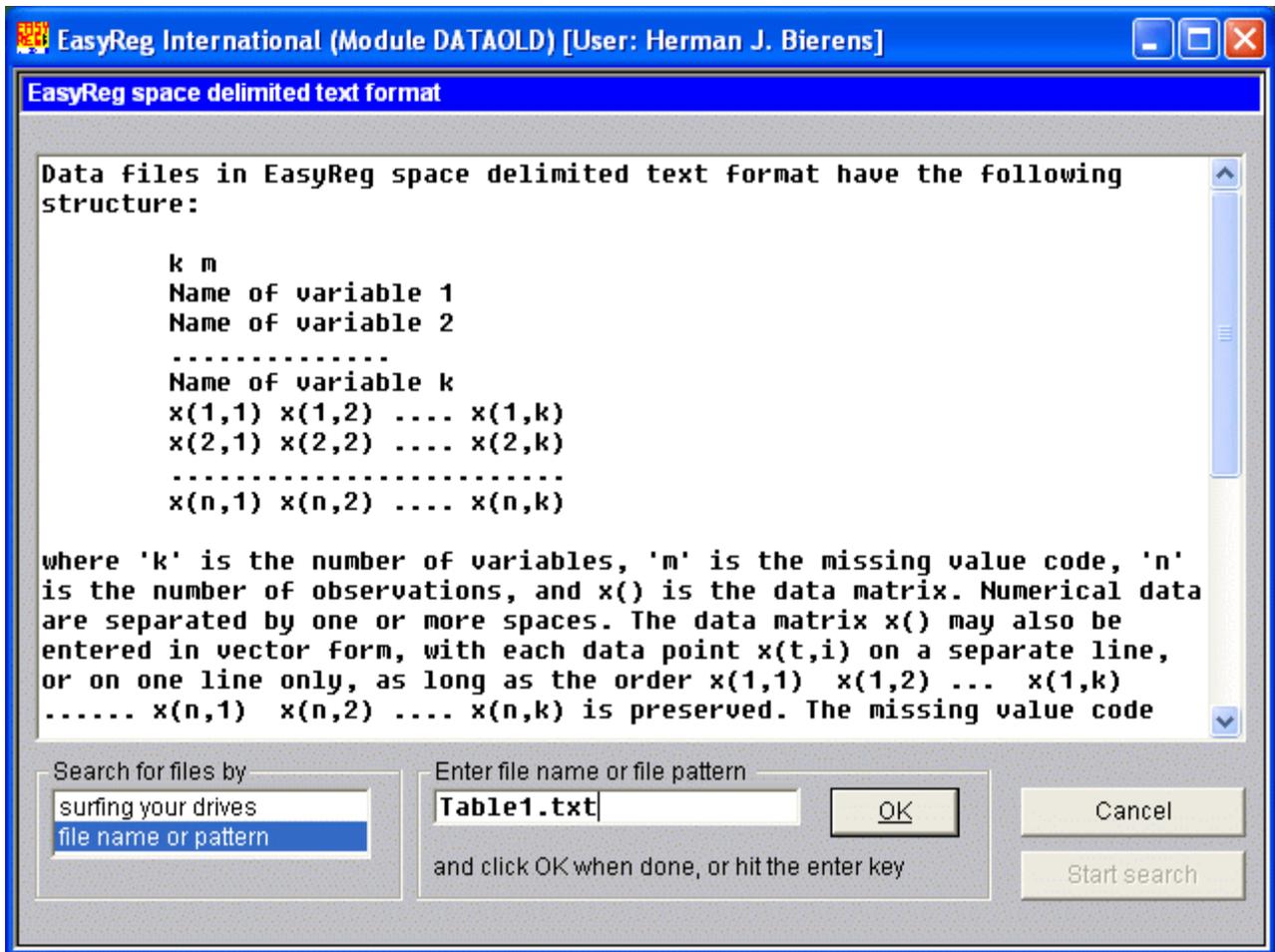


Picture 2: The File menu

2.3.1 *Importing an EasyReg data file in space delimited text format*

I will now show how to import the data in Table 1 in EasyReg. The data file name is Table1.txt, which is located in folder C:\LectureNotes\UnderGraduate\EasyReg\Data_1.

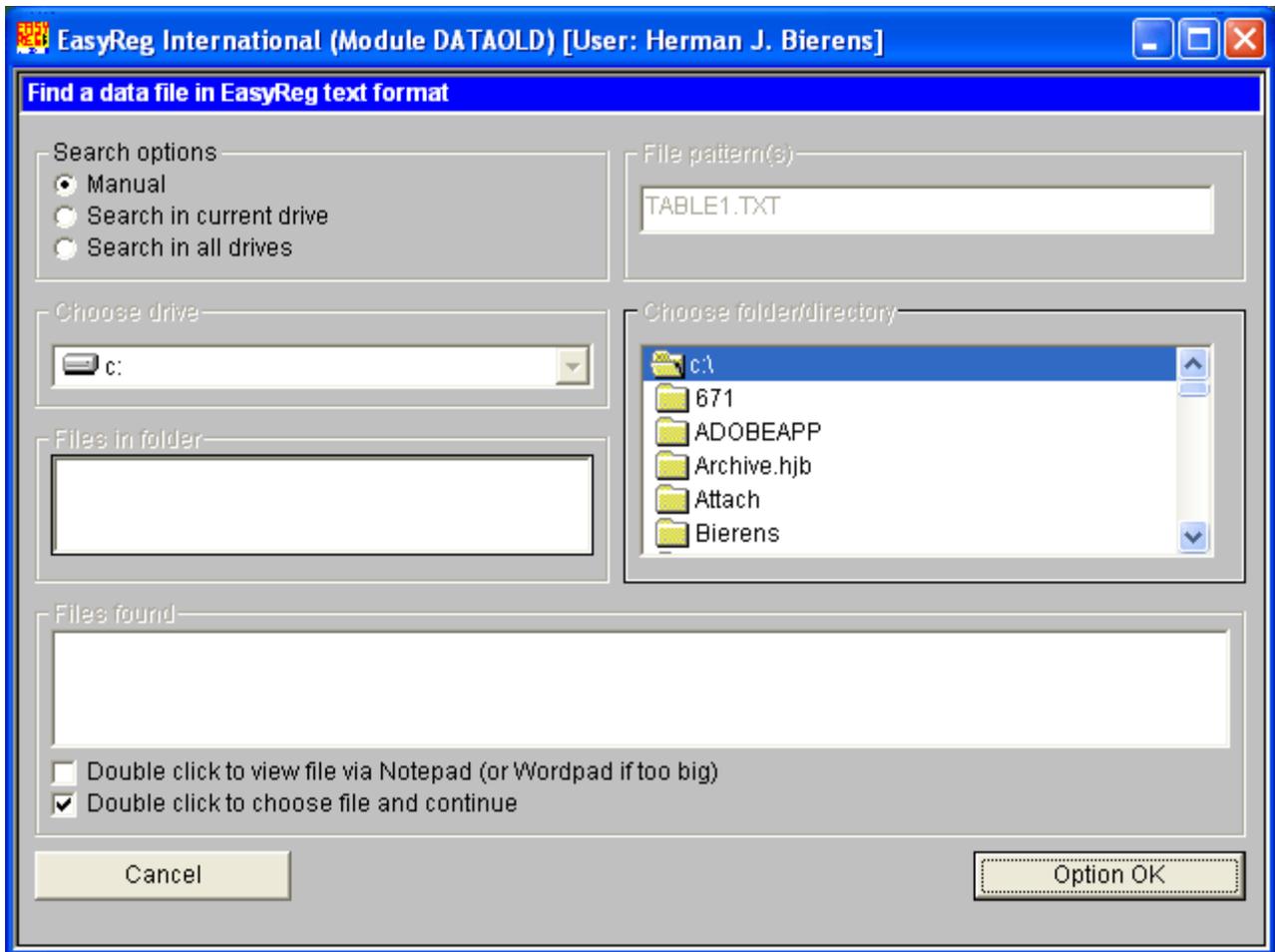
Open “File > Get data > Choose a former default EasyReg input file” (see Picture 2). Then the following window appears:



Picture 3

There are two ways to find a data file in EasyReg space delimited text format, by surfing your drives and folders, and by file name or file pattern. If you know the file name, choose the latter option, enter the file pattern or file name, click OK (or hit the enter key: Buttons with underlined text respond to the enter key as well), and then click “Start search”. Then you can scan all your drives automatically for files of this type, or navigate to the folder where the input file is located.

Because we know where to look for the file, the fastest way to find it is by choosing the option “surfing your drives”, and then click “Start search”. Then EasyReg opens the window displayed in Picture 4 below.



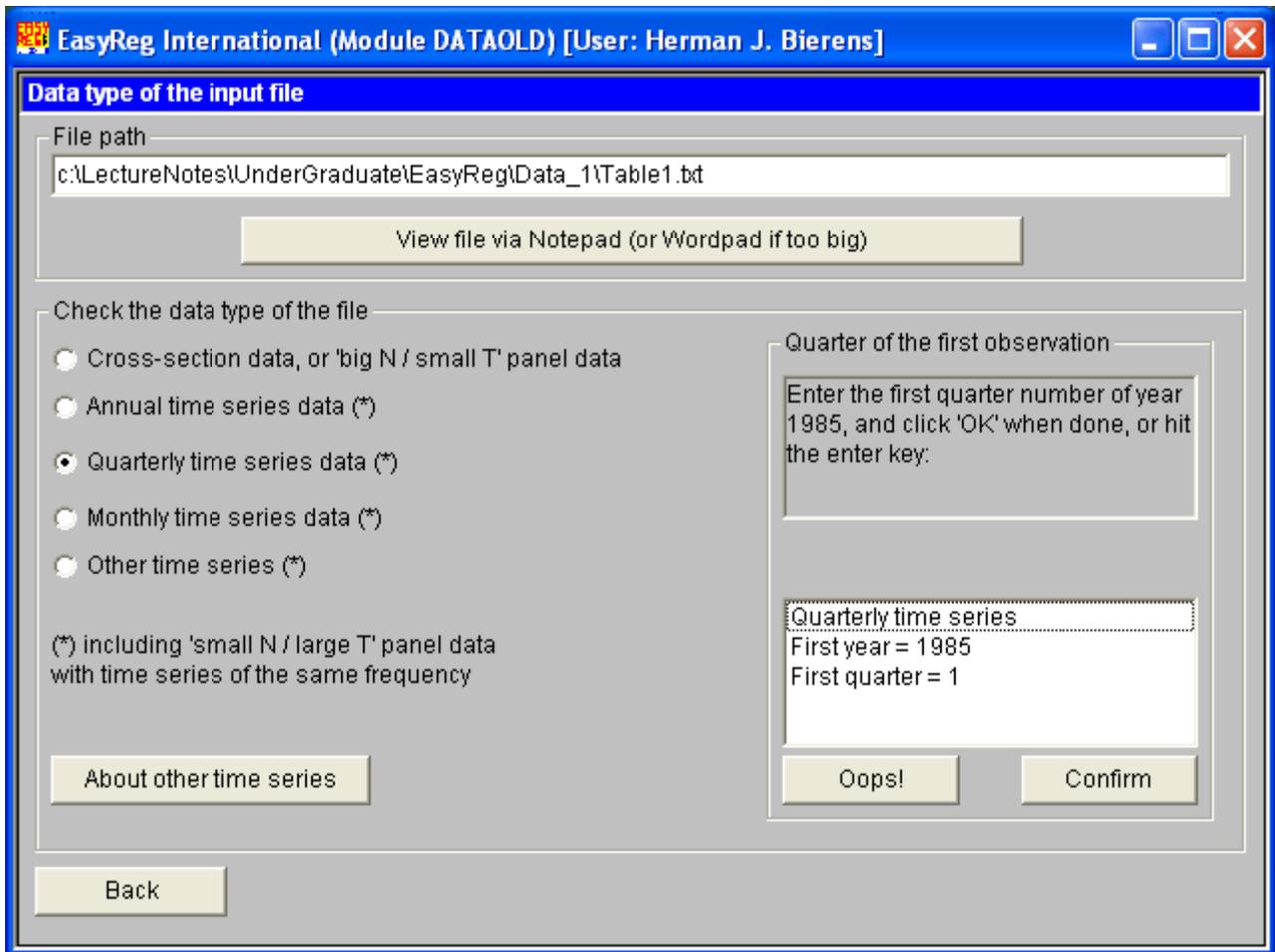
Picture 4

The default search option is “Manually”. Leave it that way, and click “Option OK”. Next, navigate to the folder where the file Table1.txt is located. This is not shown but the steps involved are self-explanatory. Then the file path

C:\LectureNotes\UnderGraduate\EasyReg\ Data_1\Table1.txt

will appear in the list box “Files found” (the last box) .

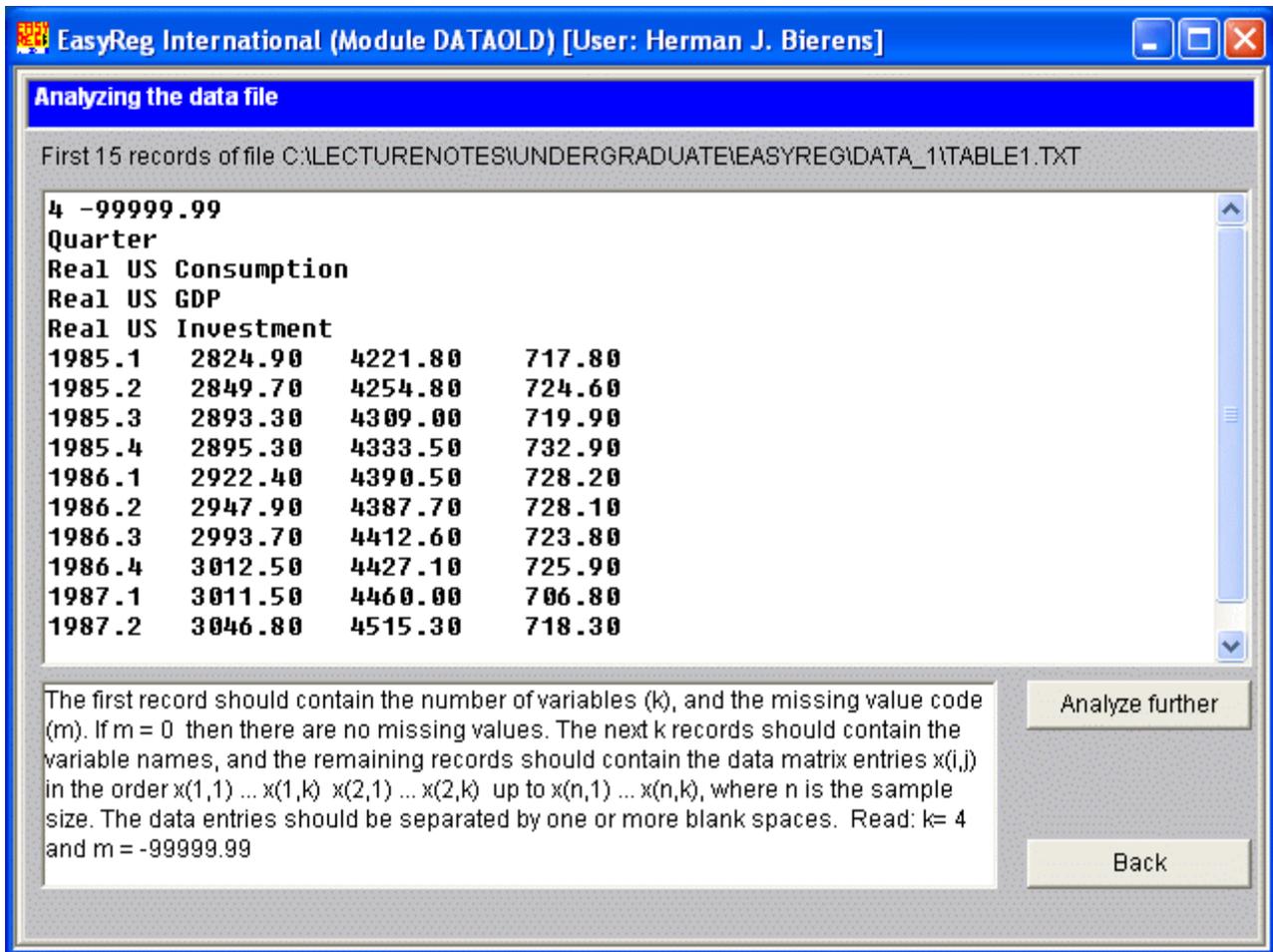
Once you double-click this file path, EasyReg will jump to the next window, which will ask you to indicate the data type, and in the time series cases the year, and for seasonal time series the quarter, month or other frequency, of the first observation. In case of the data in Table 1, the data type is quarterly time series data, starting from quarter 1 of year 1985:



Picture 5

The “Confirm” button opens a window (not shown) in which you can type any comments or information regarding this data file. This information can be retrieved later via “File > Current data” (See Picture 2). You may also skip this option. Now EasyReg will display the variable names and the next ten records of file Table1.txt. See Picture 6 below.

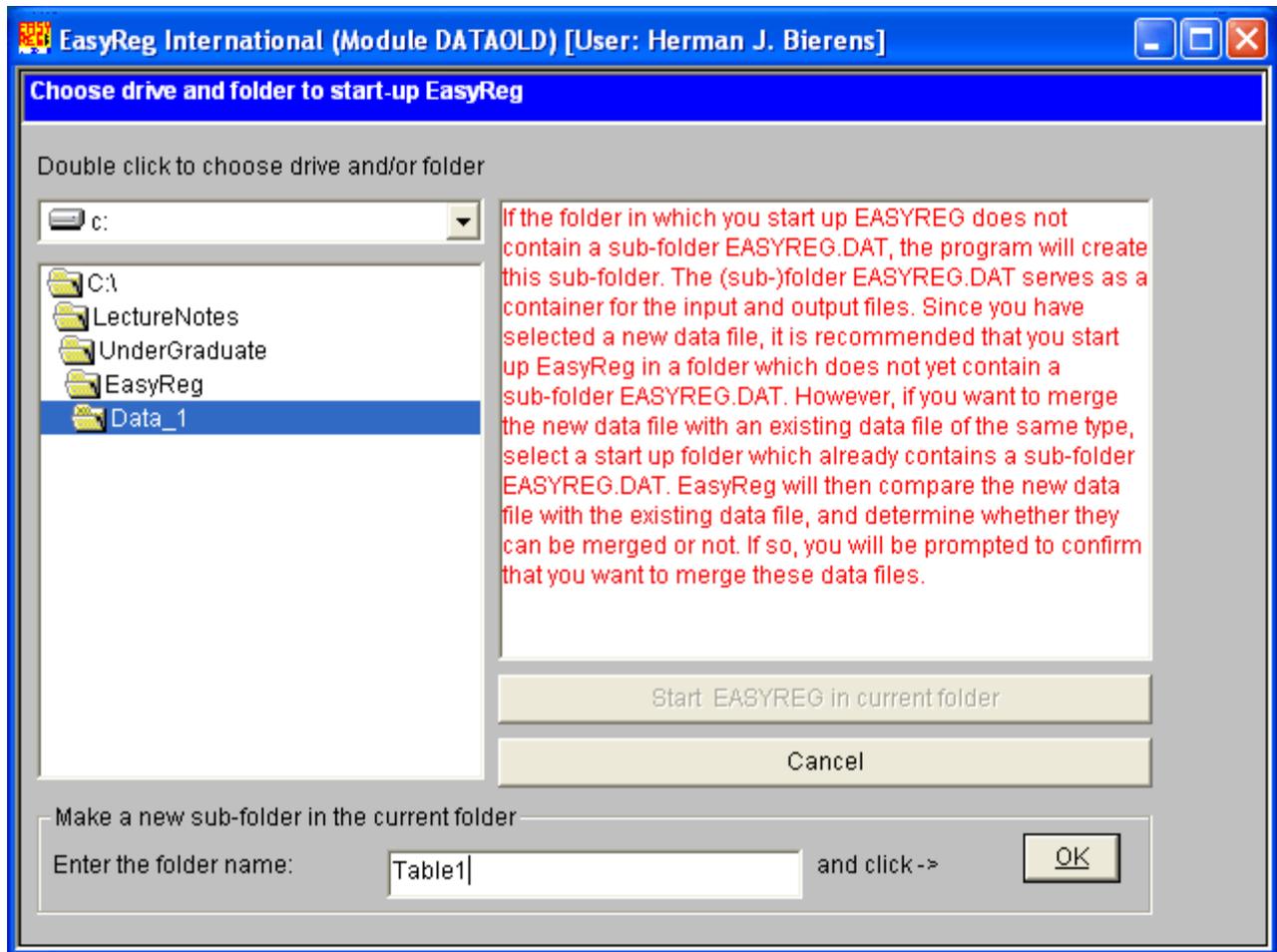
The next step is to analyze the file (via “Analyze further”), and check whether the length of the file corresponds with the number of variables. In particular, the number of numerical data entries, including missing values, divided by the number of variables, should be a natural number, n , being the sample size or the length of the time series. EasyReg also checks whether the numerical data entries are valid numbers.



Picture 6

Once the data file is analyzed and found OK, the data will be copied to a pair of random access binary files, INPUT1.RAN and INPUT2.RAN, because retrieving the data from these files is much faster than reading them from a text file. Also, the data is converted to a CSV file, INPUT.CSV, which you can open to check out whether the data file has been read correctly. The window involved is self-explanatory and therefore not shown.

In first instance the files INPUT1.RAN, INPUT2.RAN and INPUT.CSV are stored in a temporary folder. To complete the data import, you have to choose a folder where you want to store these files. EasyReg will then create a sub-folder EASYREG.DAT in that folder, which serves as a container for the input files, and later on also for the output files. The folder I will choose is a sub-folder Table1 of the folder where I had stored the original data file Table1.txt:



Picture 7

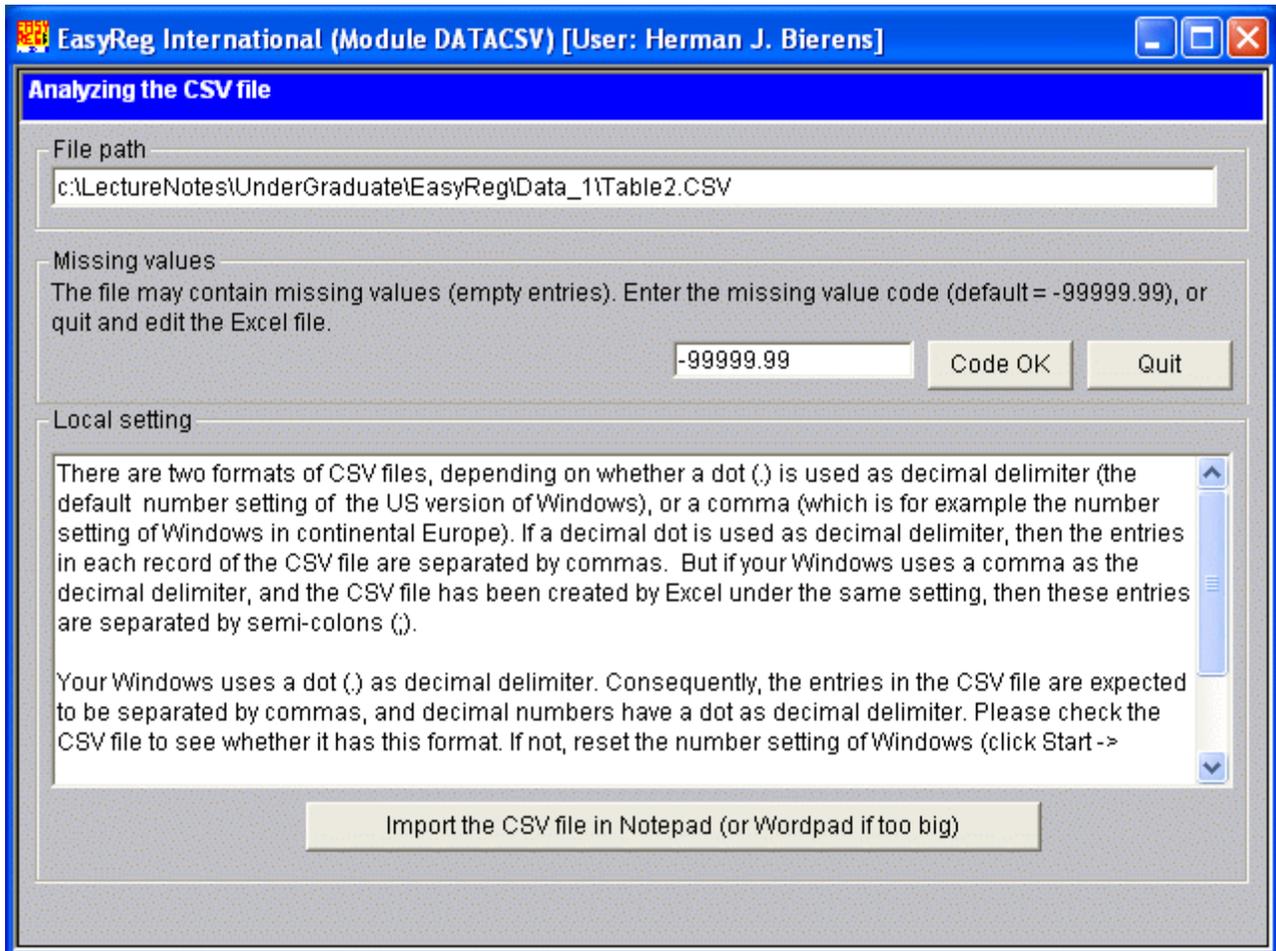
The sub-folder Table1 does not exist yet, but EasyReg can create it. Thus, type in the new sub-folder name, click OK or hit the enter key, and click the “Start EASYREG in current folder” button. Then the new input files INPUT1.RAN, INPUT2.RAN and INPUT.CSV are copied to folder

C:\LectureNotes\UnderGraduate\EasyReg\ Data_1\Table1\EASYREG.DAT,
and you are done with the data import.

2.3.2 Importing a CSV file

The file name of the data in Table 2 is Table2.CSV, which is located in the same folder as Table1.txt: C:\LectureNotes\UnderGraduate\EasyReg\Data_1. To import this file, open “File > Get data > Choose an Excel file in CSV format”. Then EasyReg opens with a window explaining what CSV files are and how they look like. The next window is the same as Picture 4. You have to navigate to the folder

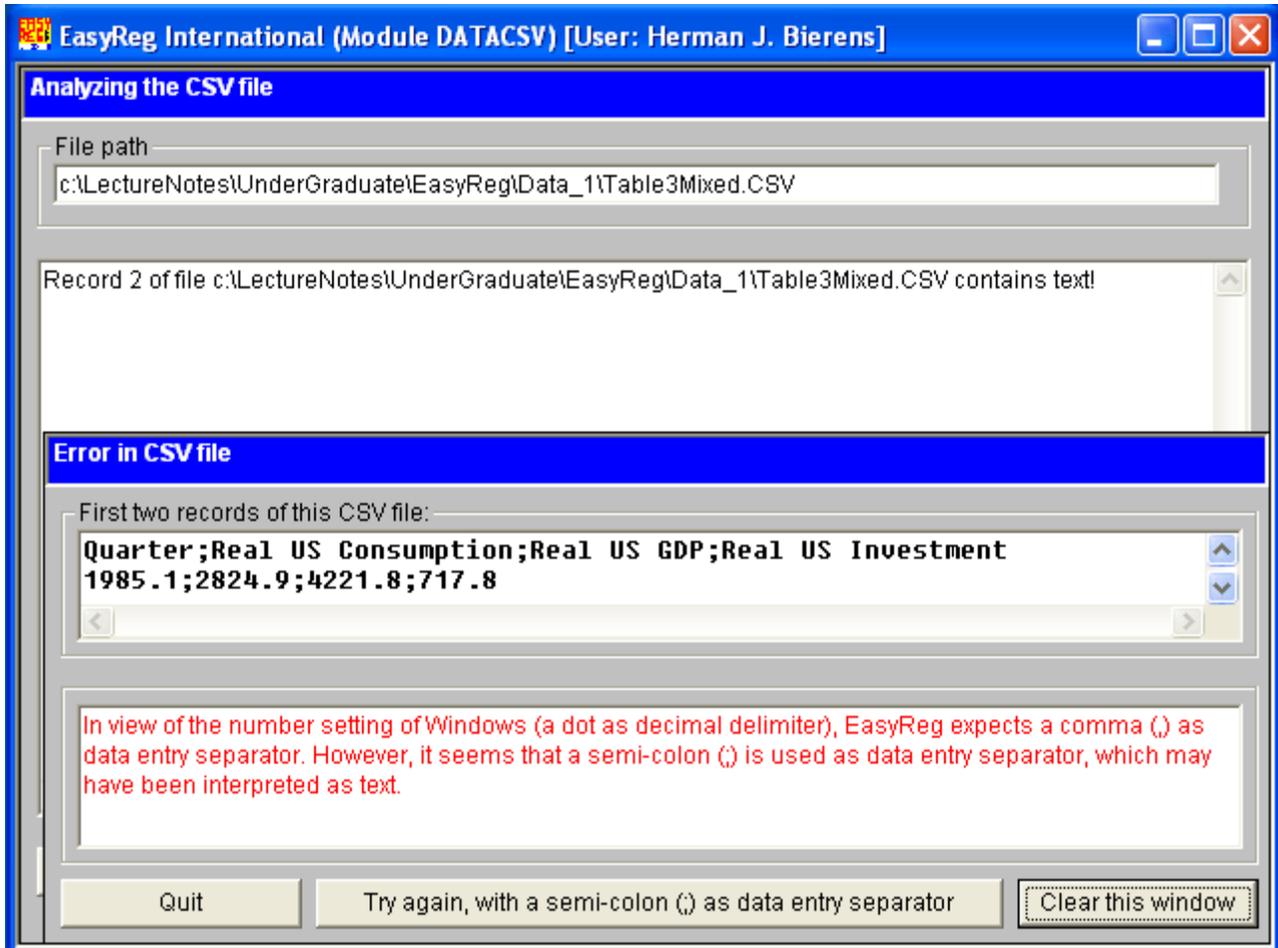
where your CSV file is located, or scan your drives for all CSV files. Once you have reached the folder the file will be displayed in the box “Files found.” The procedure is now the same as in the previous case. However, instead of Picture 6 you will get:



Picture 8

Recall that missing values in a CSV file are represented by empty data entries. Therefore, you have to specify the missing value code with which these empty entries have to be filled. I recommend to adopt the default value. Once you click Code OK, the remaining import procedures are the same as before, provided that the CSV file has the format that is expected by EasyReg. As to the latter, recall that the expected CSV format is either data entries separated by a comma (,) if Windows uses a dot (.) as decimal delimiter, as in Table 2, or data entries separated by a semi-colon (;) if Windows uses a comma as decimal delimiter, as in Table 3. However, some versions of Excel save CSV files in a mixed format: A dot as

decimal delimiter, a semi-colon as data entry separator. If the number setting of Windows is a dot as decimal delimiter, and you import a CSV file of this mixed type, you will get an error message window:



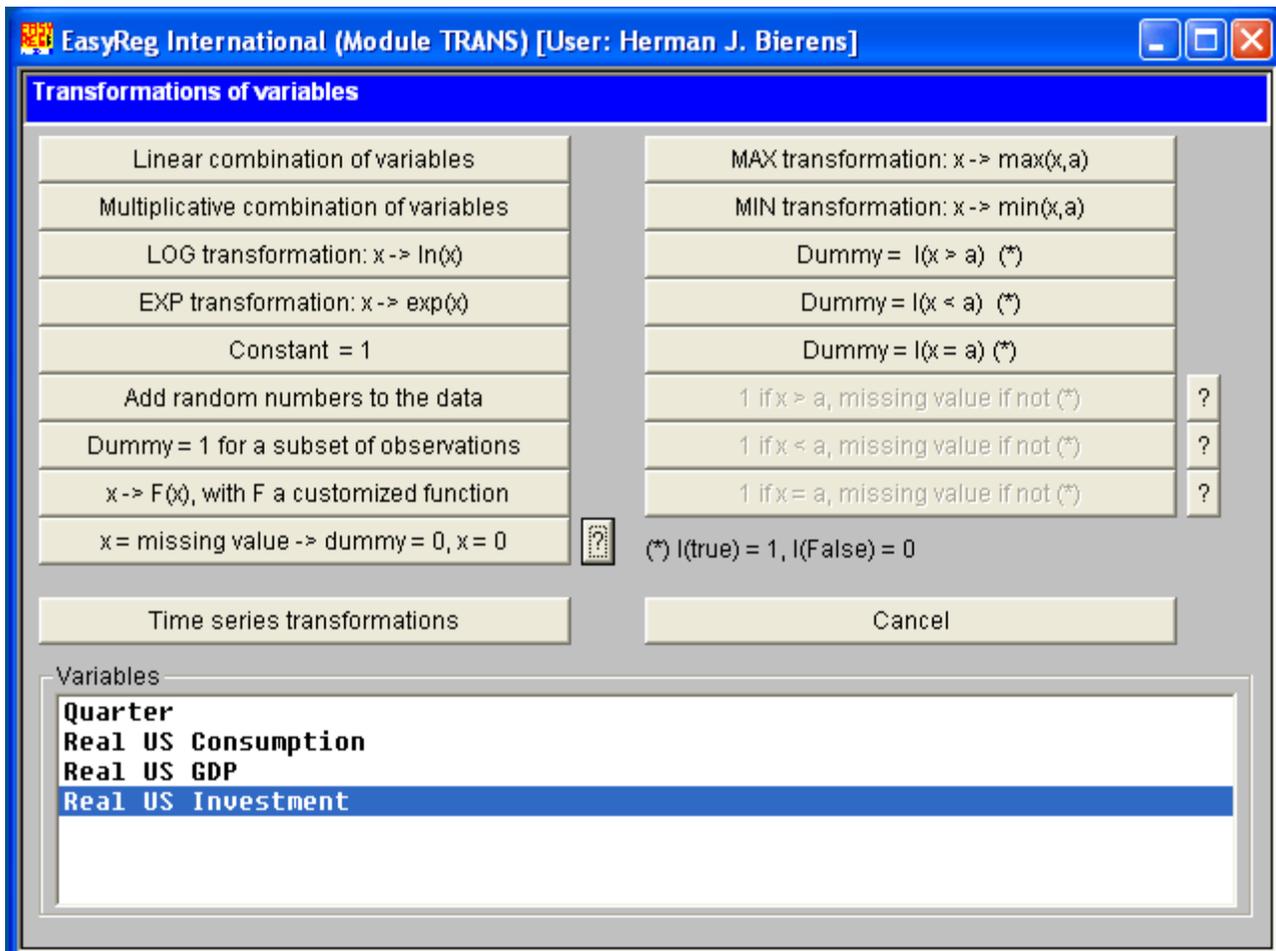
Picture 9

In this case you can read the CSV file as well, after clicking the “Try again,...” button. This will also work if the number setting of Windows is a dot as decimal delimiter and you import a European type CSV file with semi-colons as data separators and a commas as decimal delimiters. The same applies for US type CSV files if Windows uses a comma as decimal delimiter.

Finally, note that EasyReg does not need Excel itself to work with CSV files, because EasyReg reads these files as text files.

3. Data transformations

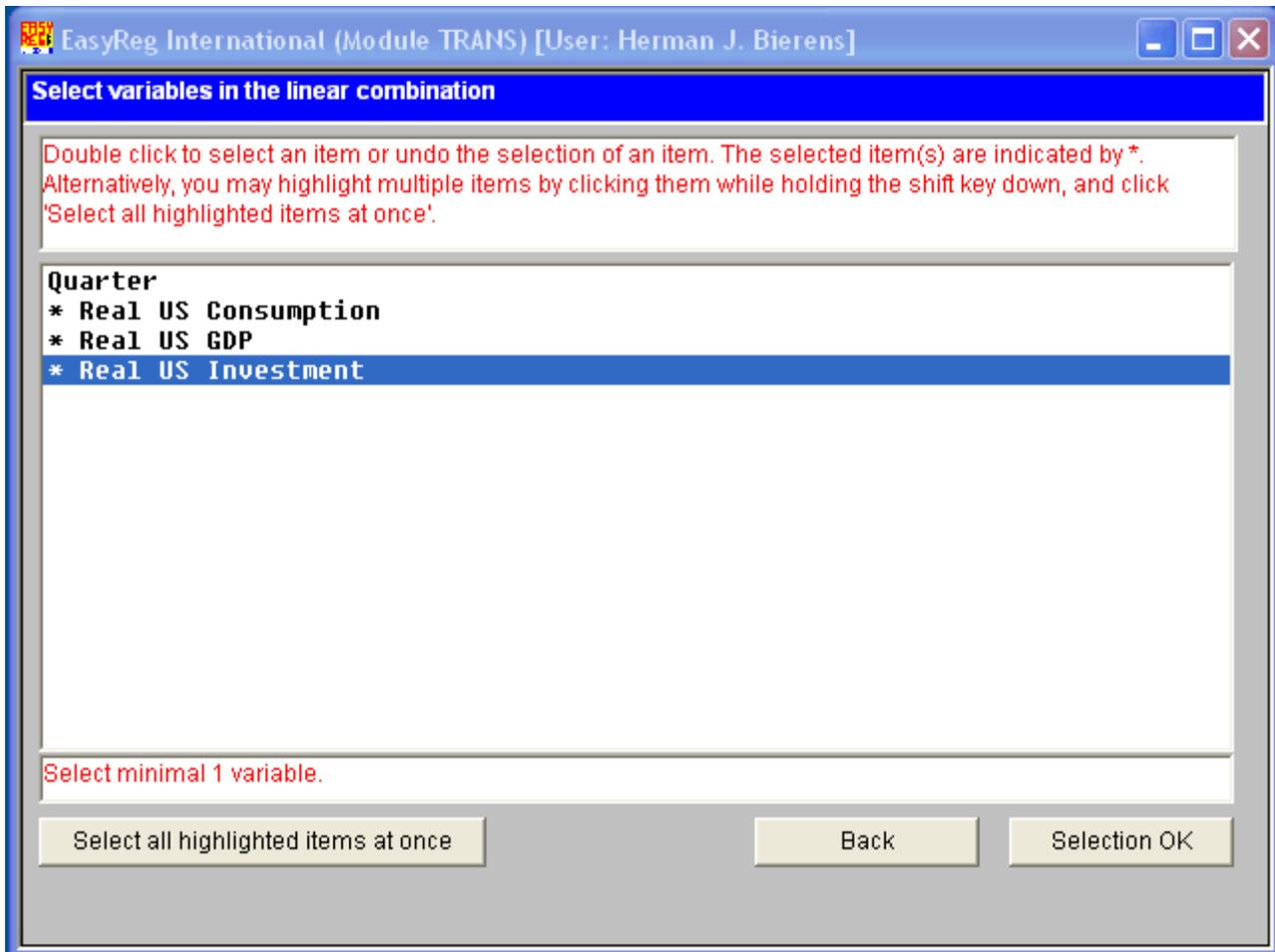
Often you have to transform your data before they can be used. If so, open “Menu > Input > Transform variables”:



Picture 10

For example, let us create a new variable, Real US GDP – Real US Consumption – Real US Investment. This is a linear combination, with coefficients 1, -1, -1. Thus, click the “Linear combination” button. Then the window displayed in Picture 11 below appears.

Double-click on Real US GDP, Real US Consumption and Real US Investment, and click “Selection OK”. EasyReg will then prompt you to enter the coefficients of each of these variables in the linear combination. Once done, you can enter a name for this linear combination. Let us call this linear combination “Other real expenditures”. Then this new variable will be added to the data.

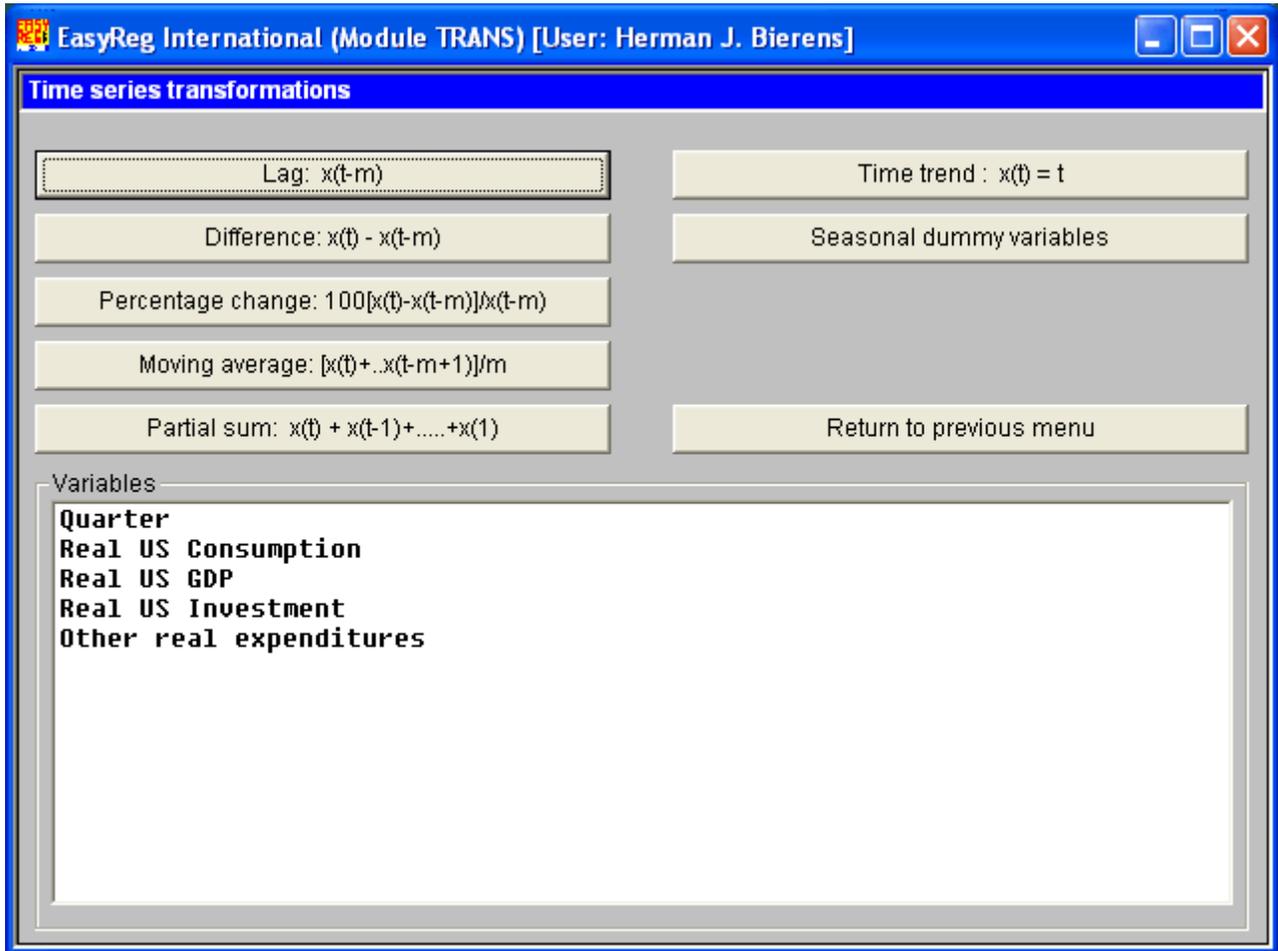


A multiplicative combination is of the form $x_1^{c_1} x_2^{c_2} \dots x_k^{c_k}$, where x_1, x_2, \dots, x_k are the variables involved and c_1, c_2, \dots, c_k are the corresponding powers. The procedure is similar to the linear combination transformation.

The LOG and EXP transformations are self-explanatory. Adding a constant 1 to the data is useful only if you want to add a constant to a variable, by taking a linear combination of 1 and a variable. You don't need it to you run a linear regression with an intercept, because the OLS module itself provides the option to include an intercept.

Creating a dummy variable for a subset of observation can be useful if you want to test whether parameters are different in a particular period or not. The other transformation options, except the option "Time series transformations", are only useful for particular advanced econometric tasks, and will therefore not be discussed here.

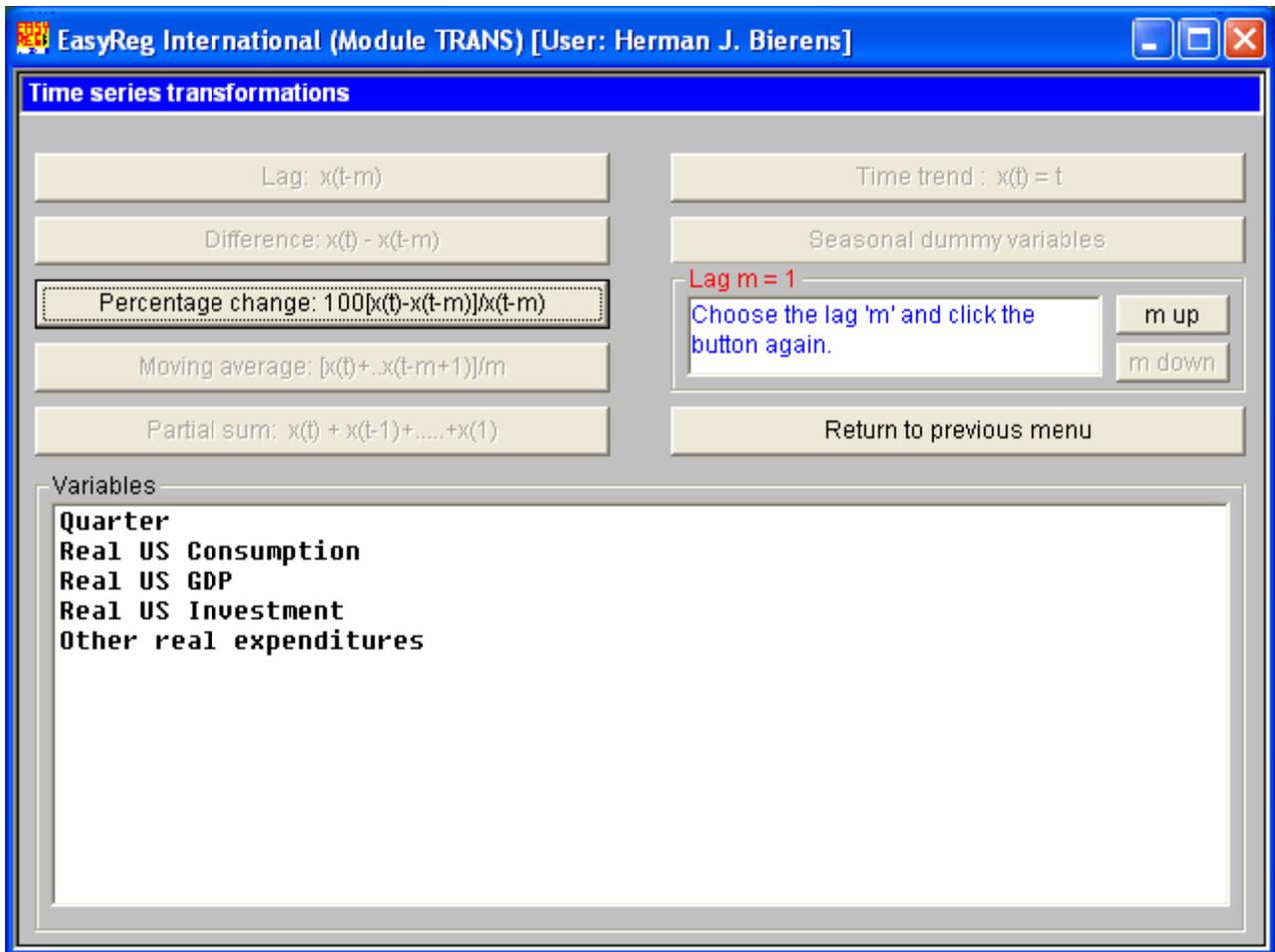
Next, click “Time series transformations”:



Picture 12

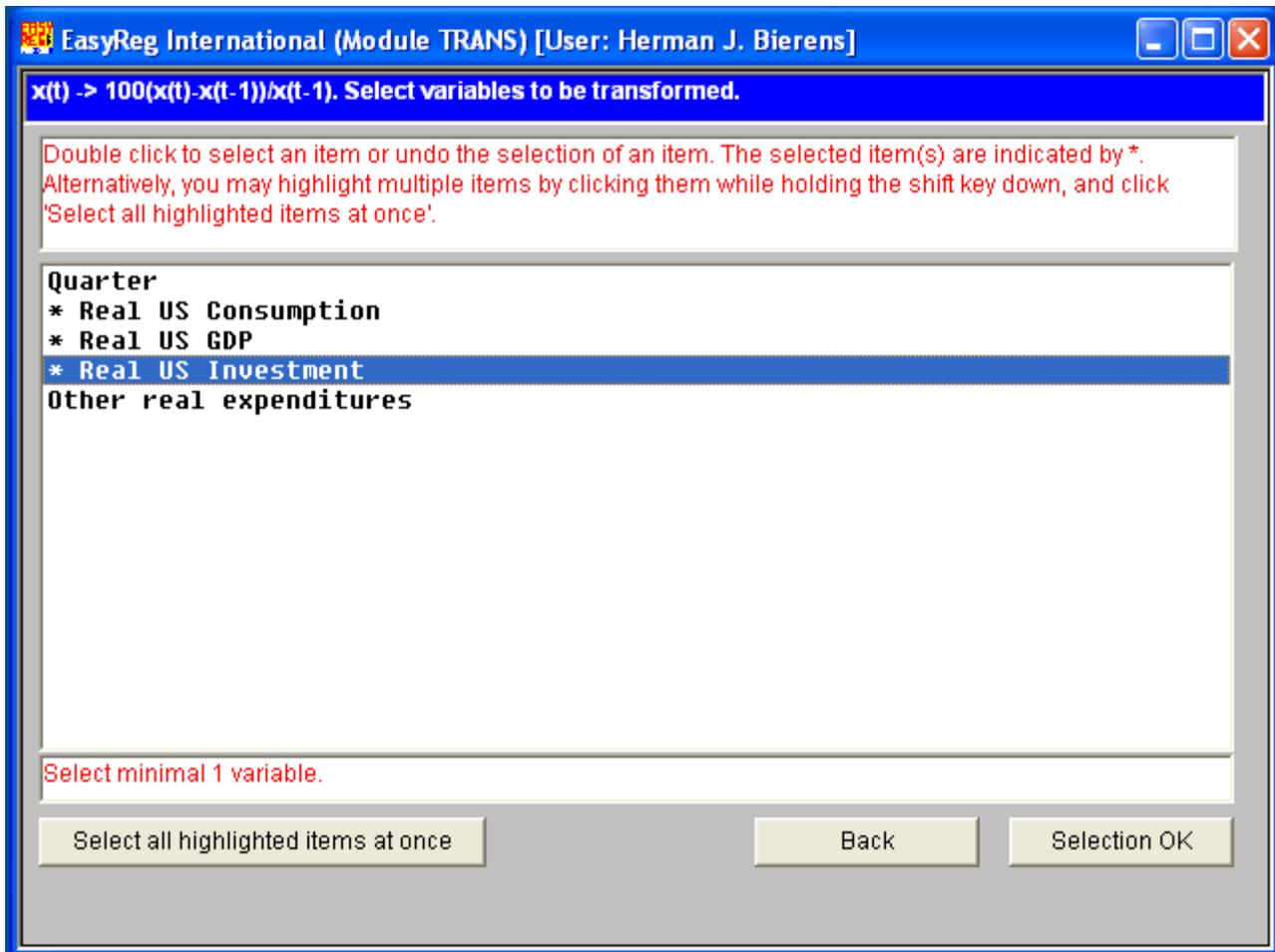
In general you don't need to add a time trend or seasonal dummy variables to the data, because the OLS module in EasyReg provides the option to include a time trend and/or seasonal dummy variables in a regression model. The same applies to the “Lag” transformation: The OLS module provides the option to include lagged dependent and/or independent variables in a regression model. The other options, except the “Difference” and “Percentage change” options, are intended for advanced econometric tasks, and will therefore not be discussed.

The most important time series transformations are the “Difference” and the “Percentage change” transformations, which are used to make the time series stationary. Once you click one of these buttons, you can specify the lag m . To demonstrate this, click “Percentage change”:



Picture 13

Here I will choose $m = 1$, and transform the variables Real US GDP, Real US Consumption and Real US Investment. Thus, click the “Percentage change” button (now with underlined text) again, select the variables Real US GDP, Real US Consumption and Real US Investment in the next window by double-clicking them, and click the “Selection OK” button. See Picture 14 below. Then, upon confirmation, three new variables will be added to the data set: %DIF1[Real US GDP], %DIF1[Real US Consumption] and %DIF1[Real US Investment], being the quarterly percentage changes of the variables involved.

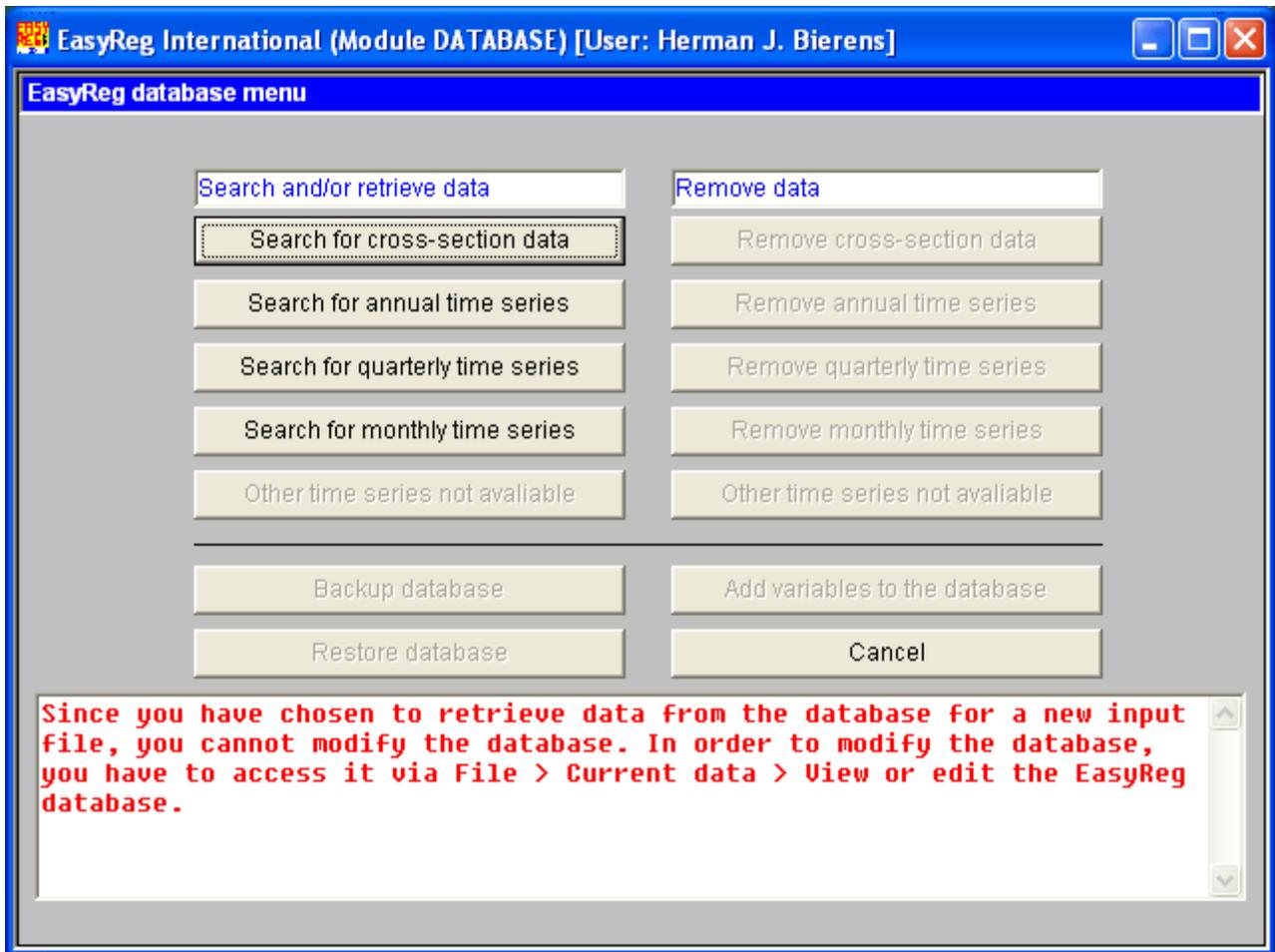


Picture 14

4. *Retrieving data from the EasyReg database*

EasyReg comes with its own database, in which you can store variables from the current input file, via “File> Current data > View or edit the EasyReg database”, or retrieve variables from for a new input file, via “File > Get data > Choose data from the EasyReg database.” In the latter case the window displayed in Picture 15 below appears.

Actually, the data in Tables 1-2 came from this database, although the time series were shortened in order to be able to display them on one page. The corresponding time series in the database are much longer, starting from quarter 1 of 1947.

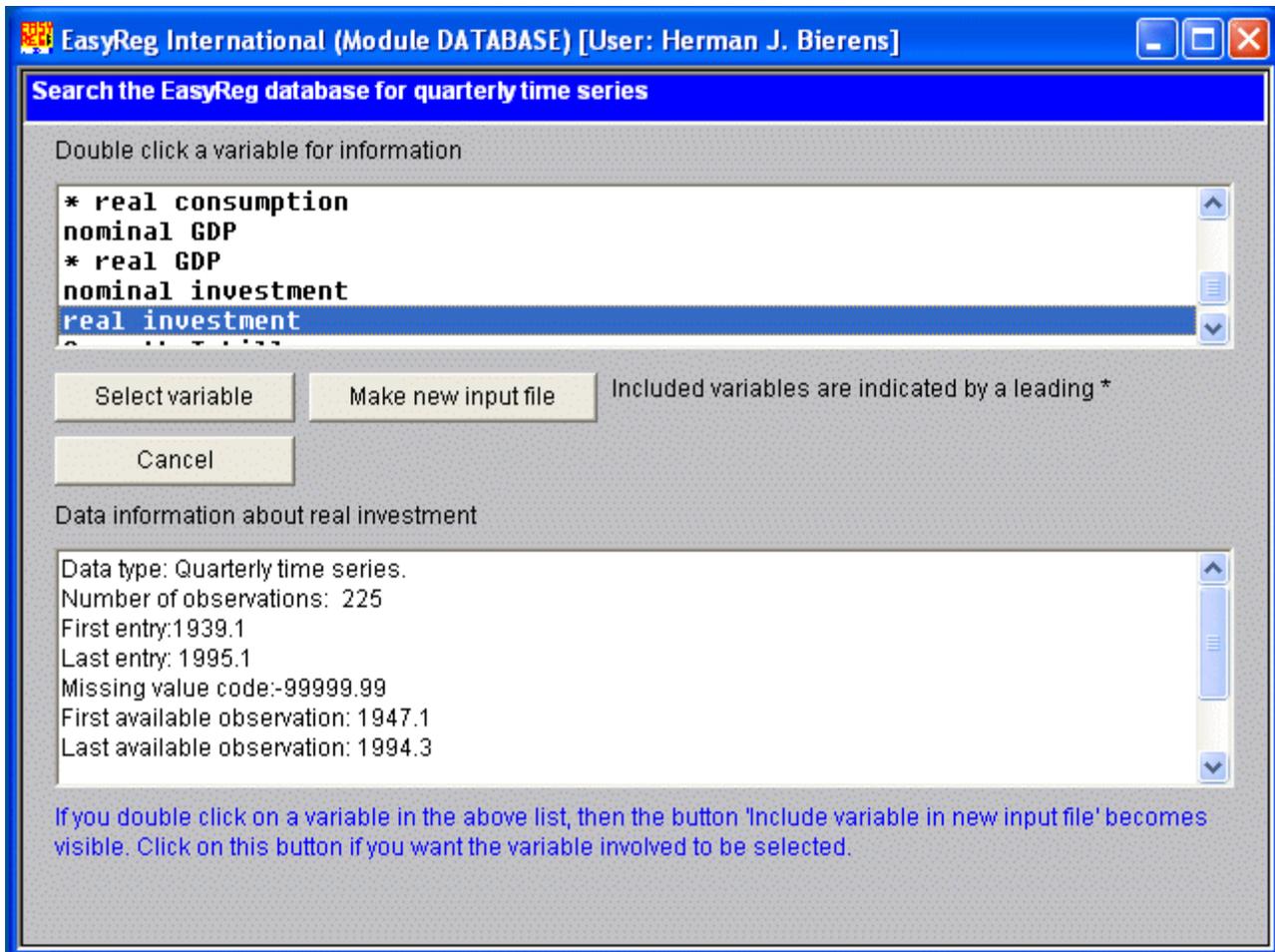


Picture 15

Let us retrieve these longer time series. Thus, click “Search for quarterly time series”. Then the window displayed in Picture 16 below appears.

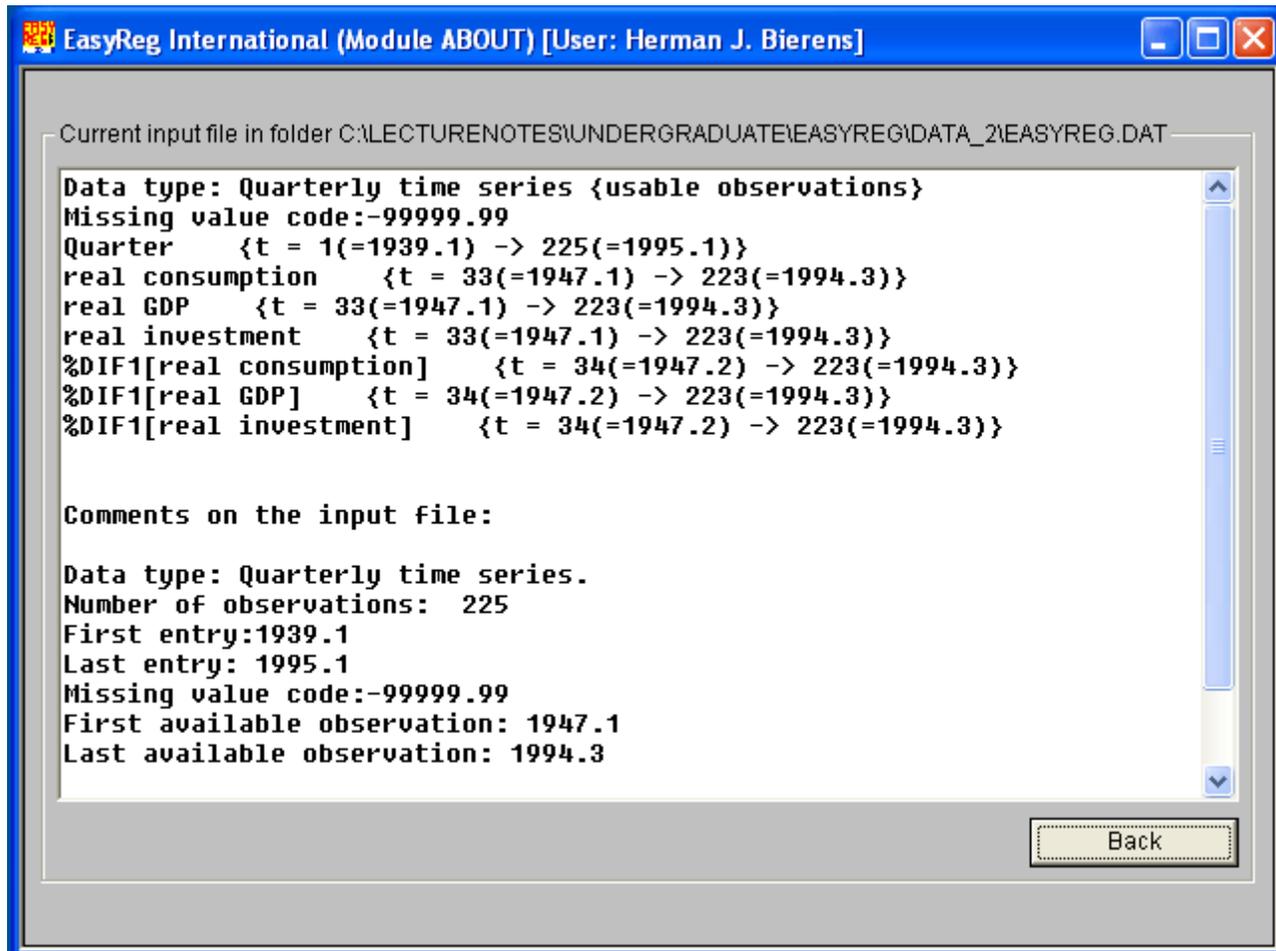
The time series involved are “real consumption”, “real GDP”, and “real investment”. Click a variable that you want to select and then click “Select variable”. Repeat this for each variable that you want to select. The selected variables are then displayed with a leading asterix (*).

EasyReg will in first instance assume that you want to store this data in the same folder as the current data set, namely C:\LectureNotes\UnderGraduate\EasyReg\ Data_1\Table1. If so, you can either overwrite the current data or merge it with the new data set, provided that in the latter case the data are of the same type. However, I will store the new data in folder C:\LectureNotes\UnderGraduate\EasyReg\Data_2. Also, I will transform the three time series by taking quarterly percentage changes, similar as before.



Picture 16

Finally, to check which is the current data set, open “File > Current input file > Show current input file”. Then the window in Picture 17 below appears. EasyReg is now ready to conduct a wide variety of econometrics tasks, for example estimation and testing a dynamic regression model. However, a discussion of these tasks and how to conduct them is beyond the scope of this paper. They are explained in detail in the various guided tours that are downloadable from the EasyReg download web page.



Picture 17: File > Current data > Show current input file