

Using R with EViews

WHITEPAPER AS OF 3/07/2016

EViews 7, 8, and 9 all depend on middleware software written by Thomas Baier called statconnDCOM in order to communicate with R. This middleware provides a COM Automation interface to R so that EViews can work with R data, and also send R commands to execute.

Installation

To use the EViews external interface to R, you must have the following installed on your Windows computer:

- R, version 2.8.1 or later.
- rscproxy, version 1.0-12 or later.
- statconnDCOM, version 3.1-1B2 or later.

R Installation

If you do not currently have R installed, the easiest procedure is to download it from <http://cran.stat.ucla.edu/bin/windows/base/>.

statconnDCOM Installation

statconnDCOM (for commercial use) is provided for a fee. Please go to their website for sales information: <http://www.statconn.com/products.html> (Their purchased version allows you to use both 32-bit and 64-bit R)

Additionally there is a free non-commercial version available for use, which is named statconnDCOM Home & Student. As of March 2016 the latest version is statconnDCOM3.6-0B3.

In this paper we describe how to setup statconnDCOM Home & Student version. (Purchased users of statconnDCOM can follow the same instructions but without the limitations of installing "rscproxy" into a library folder and without the restriction of 32-bit R only.)

You can find this version here: <http://rcom.univie.ac.at/download.html>.

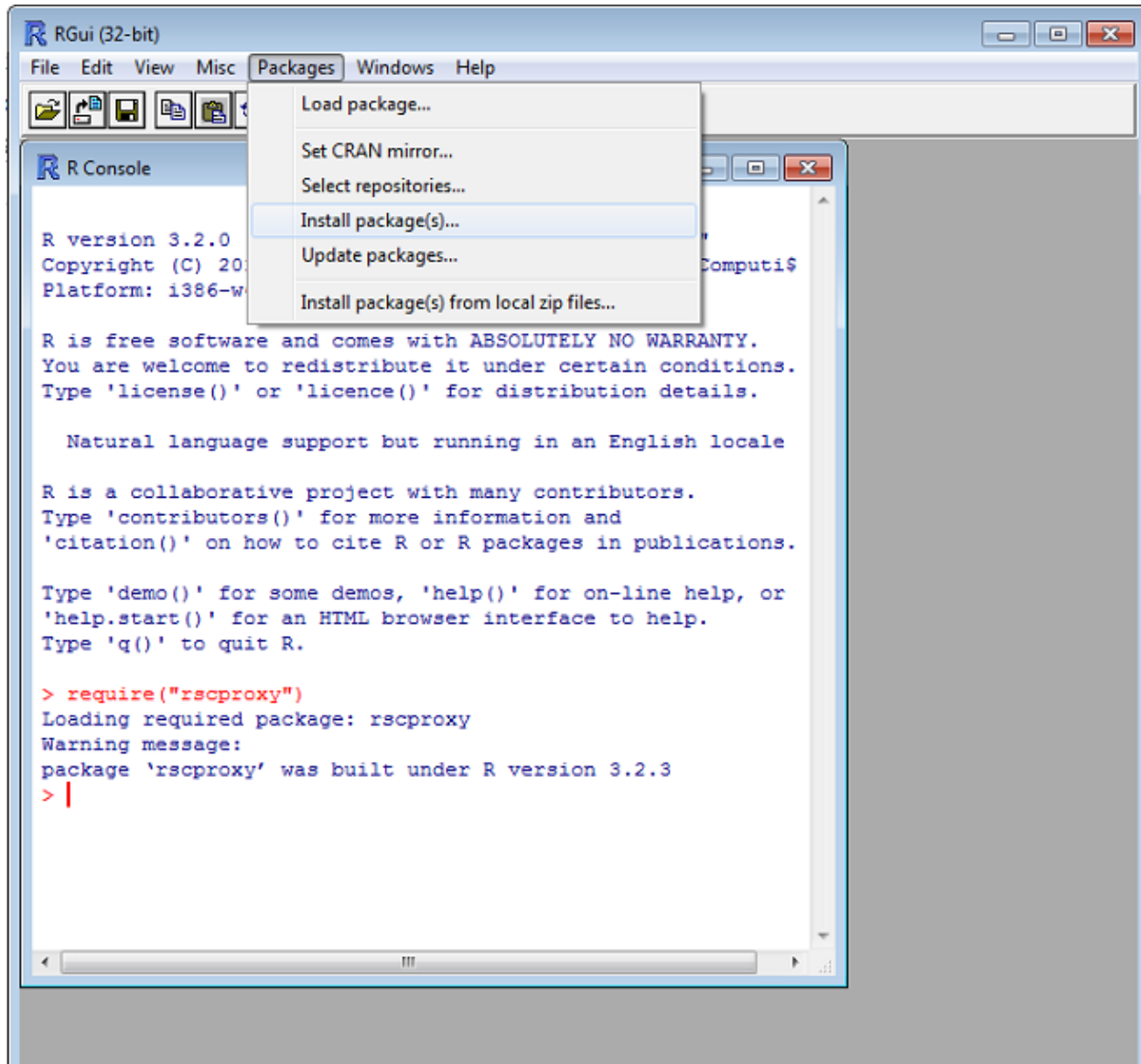
Download and install statconnDCOM from the [statconnDCOM3.6-0B3 Noncommercial](#) installer. This will install various help files and sample programs. Once installed, download and copy the "Testing.lic" file (also available from [STATCONN HOME&STUDENT LICENSE](#)) and place it into the statconnDCOM folder (usually found at: C:\Program Files (x86)\statconn\DCOM).

Notes:

- You can use 32-bit statconnDCOM with 64-bit EViews.
- Our testing of the statconnDCOM3.6-0B3_Noncommercial version found some problems with the version of R 3.1.X (April 2014- March 2015). If you have statconnDCOM3.6-0B3, we suggest you use 32-bit R 3.2.X or later instead:
(You can find older versions at: <http://cran.stat.ucla.edu/bin/windows/base/old/>)

rscproxy Installation

statconnDCOM itself requires that the "rscproxy" package be installed in R. The Home & Student version requires that "rscproxy" be installed in the R library folder, not a user-folder. To insure proper installation of this package, you must run 32-bit R with admin rights before attempting to install the "rscproxy" package. In R, go to the Packages menu->Install Packages... and pick the "rscproxy" package from the list. The package should install directly to the library folder without asking to save in the user directory. If this question is asked, please restart R with admin rights and try the install again.



To verify proper installation, type in the command:

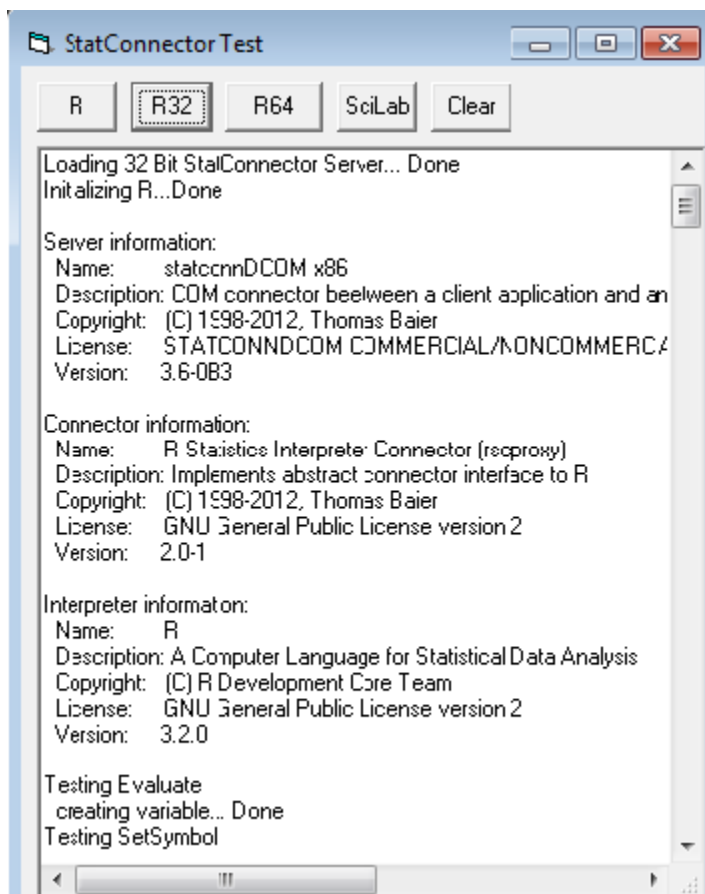
```
require("rscproxy")
```

and you should see the proper response from R:

```
Loading required package: rscproxy
```

Testing statconnDCOM with R

statconnDCOM provides a sample test program (named "Simple 01 - Basic Test") to make sure all functionality works properly between statconnDCOM and R. Look for the "Server 01 - Basic Test" program under the Windows Start button / All Programs / statconn / DCOM (or you can run the EXE directly from C:\Program Files (x86)\statconn\DCOM\samples\Simple\simple.exe). Once running, click the R32 button to begin the test for 32-bit R. You should see the statconnDCOM splash screen appear along with debug output in the large textbox below the R32 button.



If any message dialogs appear with an error, then a problem exists between statconnDCOM and R. Debugging these problems may require you to do some research into the specific problem. The following (older) PDF may provide some insight into what can cause certain errors:
<https://www.unt.edu/rss/class/splus/Usi...server.pdf>

If you wish to use statconnDCOM with other versions of R, simply run the same test program to verify that all functionality is still working. If you run into any issues, you may need to downgrade your version of R in order to fix them.

Using R with EViews

Once you've confirmed statconnDCOM is installed properly and communicating with R without issue, you can use it in EViews with the XOPEN, XPUT, XGET, and XRUN commands. Please refer to our documentation for more details on how to use the various 'X' commands.

```
xopen(type=r)
```

This command will open a connection to R from within EViews and will also open an R output log window in EViews.

Example

The following example (you can find it in EViews documentation) shows how to move data from EViews into R, how to use EViews to run an R command, and how to retrieve results from R into EViews.

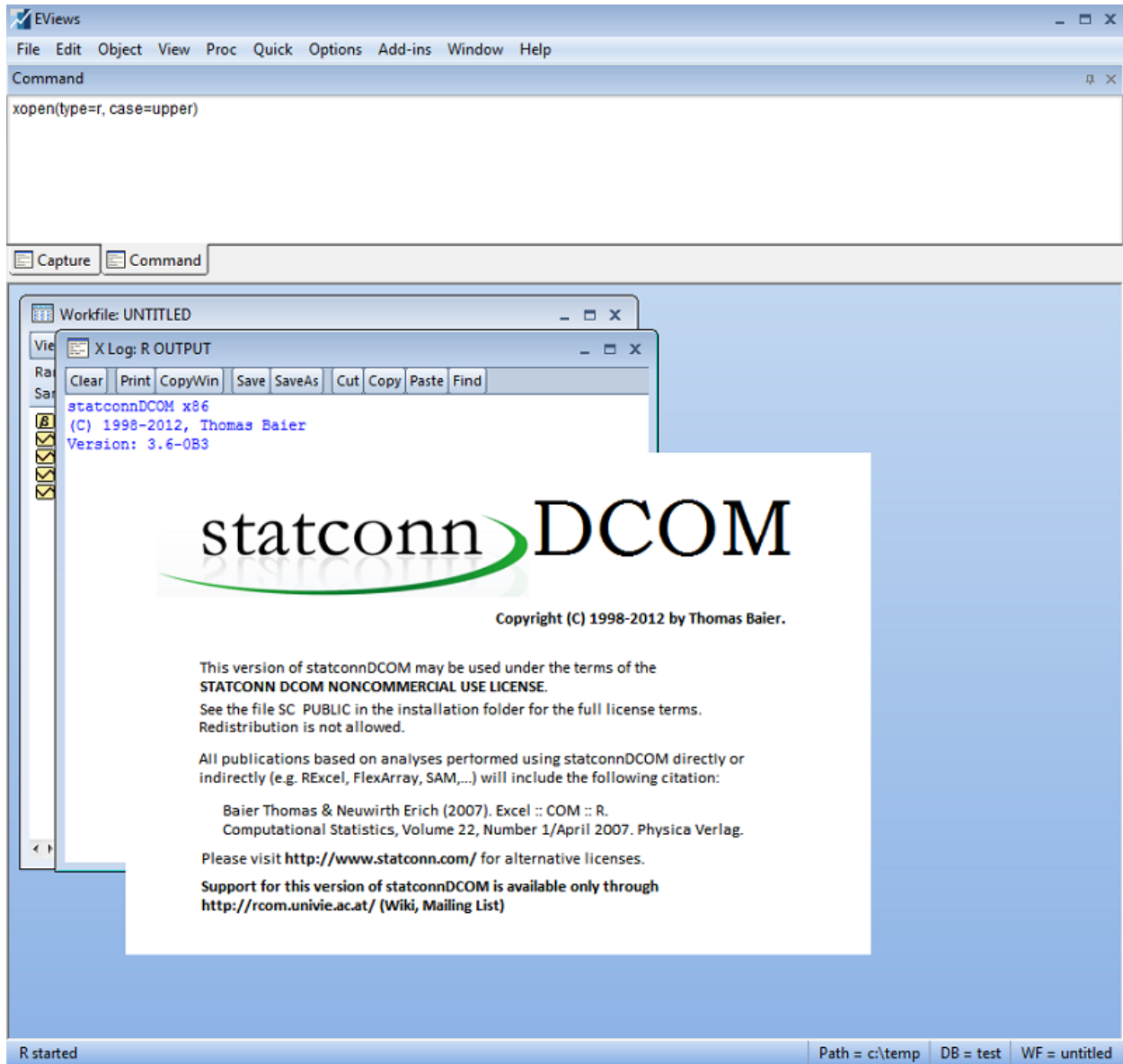
```
'create a workfile
wfccreate u 100
'create some data
series y=rnd
series x1=rnd
series x2=rnd
'open a connection to R with upper-case default output names
xopen(type=r, case=upper)
'put regressors and dependent variable into R
xput(rtype=data.frame, name=vars) y x1 x2
'run a command to perform GLM
xrun "z<-glm(Y~X1+X2, family=Gamma(link=log), data=vars)"
xrun "summary(z)"
'retrieve coefs
xget(name=beta, type=vector) z$coef
'create EViews equation
equation e1.glm(family=gamma,link=log) y c x1 x2
show e1
show beta
'close R connection
Xclose
```

The program first creates a workfile, then creates some series objects. The series Y is the dependent variable, and the series X1 and X2 and a constant are regressors. xopen is used to open a connection to R, and then xput is used to pass the series into an R data.frame container, which we name "VARS". Note that the names of the data.frame, and its contents are all uppercased in R since the connection was opened with the "case=upper" option.

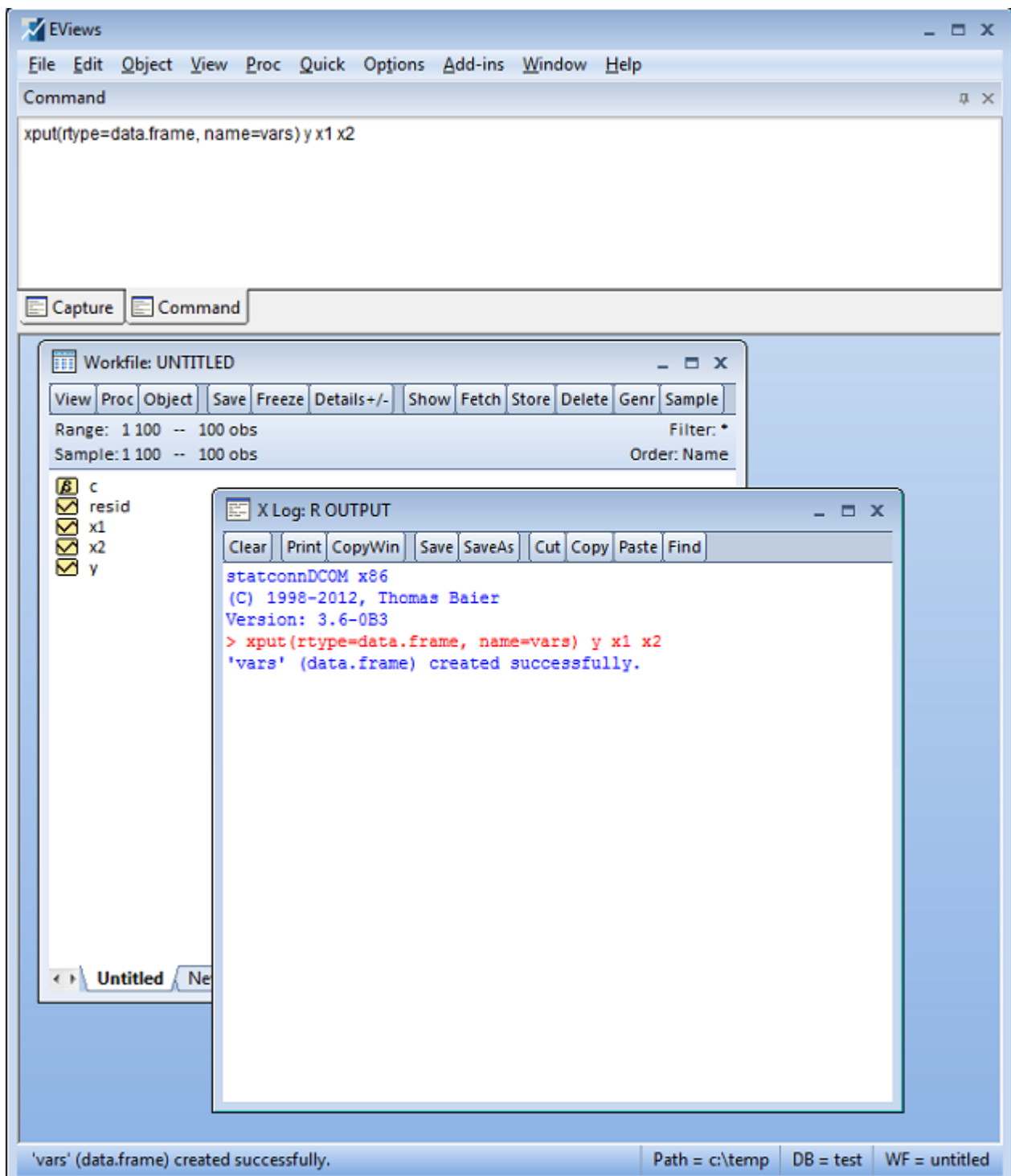
xrun is used to estimate a GLM equation in R, with the results being stored in a GLM output object, "z". xget is used to retrieve the coefficient member of z back into EViews, where it is stored as a vector called BETA.

Finally, the same GLM specification is estimated inside EViews, and the coefficient estimates are shown on the screen for comparison.

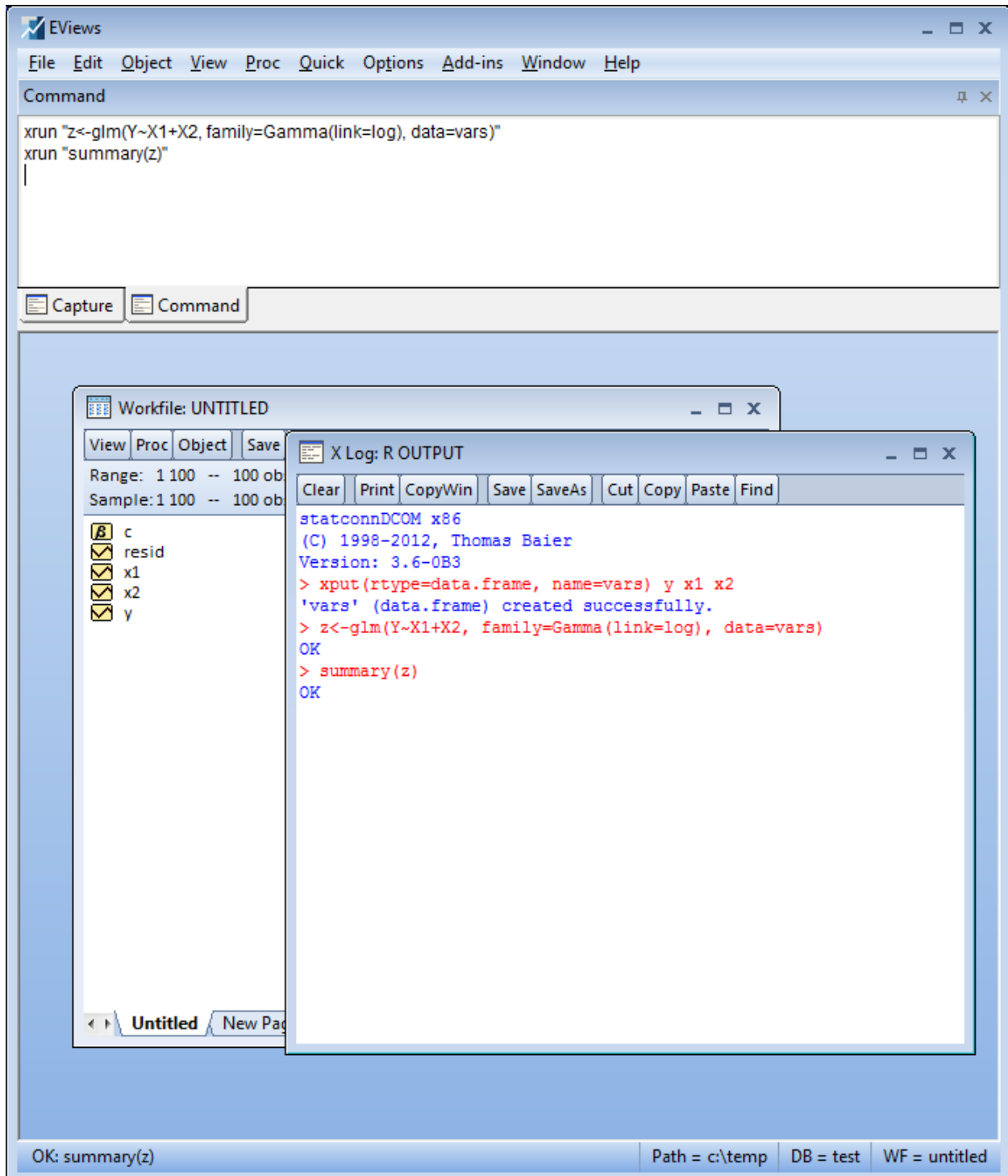
xopen is used to open a connection to R (statconnDCOM splash screen appears until connection to R is established) :



xput is used to pass an EViews object to an R data.frame container:



xrun is used to run a command in the R application:



xget is used to retrieve data from R back into EViews:

The screenshot displays the EViews software interface. At the top, the 'Command' window contains the command: `xget(name=beta, type=vector) z$coef`. Below this, the 'Workfile: UNTITLED' window shows a list of objects: `beta`, `c`, `resid`, `x1`, `x2`, and `y`. A sub-window titled 'X Log: R OUTPUT' shows the following R code and its output:

```
statconnDCOM x86
(C) 1998-2012, Thomas Baier
Version: 3.6-0B3
> xput(rtype=data.frame, name=vars) y x1 x2
'vars' (data.frame) created successfully.
> z<-glm(Y~X1+X2, family=Gamma(link=log), data=vars)
OK
> summary(z)
OK
> xget(name=beta, type=vector) z$coef
'z$coef' loaded into 'BETA' successfully.
'z$coef' loaded successfully.
```

At the bottom of the EViews window, a status bar indicates: `'z$coef' loaded successfully.` and `Path = c:\temp DB = test WF = untitled`.

Show results inside EViews:

The screenshot displays the EViews interface with the following components:

- Command Window:** Shows the execution of the command `equation e1.glm(family=gamma,link=log) y c x1 x2`, followed by `show e1` and `show beta`.
- X Log: R OUTPUT:** Displays the R console output, including the successful execution of `xput`, `glm`, `summary`, and `xget` commands.
- Equation: E1:** Shows the model specification: `Generalized Linear Model (Newton-Raphson / Marquardt steps)`. It lists the dependent variable as Y, the method, date, sample, and included observations. It also shows the family (Gamma) and link (Log).
- Table of Coefficients:**

Variable	Coefficient	Std. Error	z-Statistic	Prob.
C	-0.511026	0.144411	-3.538705	0.0004
X1	0.026016	0.208467	0.124799	0.9007
X2	-0.210329	0.201721	-1.042672	0.2971
- Model Statistics:**

Mean dependent var	0.550136	S.D. dependent var	0.307856
Sum squared resid	9.271640	Log likelihood	-37.20932
Akaike info criterion	0.804186	Schwarz criterion	0.882342
Hannan-Quinn crit...	0.835817	Deviance	56.29104
Deviance statistic	0.580320	Restr. deviance	56.64057
LR statistic	1.084624	Prob(LR statistic)	0.581402
Pearson SSR	31.25904	Pearson statistic	0.322258
Dispersion	0.322258		
- Vector: BETA:** A spreadsheet view showing the estimated coefficients for variables C1, R1, R2, and R3.

	C1
R1	-0.511029
R2	0.026017
R3	-0.210323