

# Estimating time-varying coefficients with Gretl using the VC method

Ekkehart Schlicht

VCwrapper is a function package for Gretl (2022) that runs the executable VCG. VCG stands for “Varying coefficients estimation for Gretl”. It estimates time-varying coefficients in linear models and runs under Windows and Linux.

## 1 Introduction

The method for estimating linear models with time-varying coefficients outlined in Schlicht (2021) has found a number of implementations. The most recent and probably most useful one is offered as a function package “VCwrapper” for the statistical software Gretl (2022). It runs under Windows and Linux. The following notes describe the purpose and functioning of the software.

## 2 Purpose

The program estimates the time-varying coefficients  $a'_t = (a_{1,t}, a_{2,t}, \dots, a_{n,t})$  of the linear regression

$$y_t = a'_t x_t + u_t \quad (1)$$

with regressors  $x'_t = (x_{1,t}, x_{2,t}, \dots, x_{n,t})$  and the dependent variable  $y_t$ . The time series  $x'_t$  and  $y_t$  denote the observations at time  $t = 1, 2, \dots, T$ . The disturbance  $u_t$  is a random variable with mean zero and variance  $\sigma^2$ . The coefficients  $a_{i,t}$  are assumed to be generated by a random walk

$$a_{i,t+1} = a_{i,t} + v_{i,t}, \quad t = 1, 2, \dots, T; \quad i = 1, 2, \dots, n \quad (2)$$

with disturbances  $v_{i,t}$  with mean zero and variances  $\sigma_i^2$ . The program estimates the conditional expected values of the coefficients  $a_{1,t}, a_{2,t}, \dots, a_{n,t}$  and the variances  $\sigma^2, \sigma_1^2, \sigma_2^2, \dots, \sigma_n^2$ , given the observations  $x_{i,t}$  and  $y_t$  for  $t = 1, 2, \dots, T$  and  $i = 1, 2, \dots, n$ . The method used is the VC method described in Schlicht (2021).

The estimators for the variances are moments estimators. With given variances, the time-paths of the coefficients are determined by minimizing the weighted sum of squares

$$Q = \sum_{t=1}^T u_t^2 + \sum_{t=2}^T \sum_{i=1}^n \gamma_i (a_{i,t} - a_{i,t-1})^2 \quad (3)$$

with smoothing weights  $\gamma_i$  given by the inverse variance ratios

$$\gamma_i = \frac{\sigma^2}{\sigma_i^2}, \quad i = 1, 2, \dots, n. \quad (4)$$

The matrix of coefficients

$$A = \begin{pmatrix} a_{1,1} & a_{2,1} & \dots & a_{n,1} \\ a_{1,2} & a_{2,2} & \dots & a_{n,2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1,T} & a_{2,T} & \dots & a_{n,T} \end{pmatrix} \quad (5)$$

that minimizes the sum of squares  $Q$  gives the expectation of the coefficients  $a_{i,t}$  for given observations and given variance ratios. The VC method calculates estimates for the variances and variance ratios by a moments estimator which coincides asymptotically with a maximum-likelihood estimator in case the disturbances are Gaussian.

### 3 Installation

In order to use the software, the Gretl software must be downloaded and installed from <http://gretl.sourceforge.net/>. Run the program and enable shell commands (Tools → Preferences → General → Allow shell commands). Next, open the Gretl console (third button at the lower edge of the main window) and install the VCwrapper package by typing `pkg install https://epub.ub.uni-muenchen.de/84584/1/VCwrapper.zip`. Alternatively

the file VCwrapper.zip can be downloaded from <https://epub.ub.uni-muenchen.de/84584> and installed to Gretl by following the instructions in the manual that can be accessed through the Help menu of Gretl's main window. You will be asked whether you would like to attach VCwrapper to the Gretl menus (Figure 1). Select "Yes" and close the console.

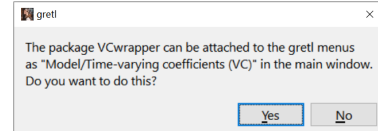


Fig. 1: Attach to menu

### 4 Usage

If the VCwrapper package is installed, the program can be invoked by opening the main window of Gretl and selecting File → Function packages... → On local machine... → VCwrapper (Figure 2). Alternatively, the *fx* button at the bottom of the main window can be clicked and VCwrapper can be selected.

If no time series data are loaded, a window appears that proposes to run a sample script. If Yes is clicked, the sample script opens (Figure 3). The script may serve to illustrate the use of VCwrapper in scripts.

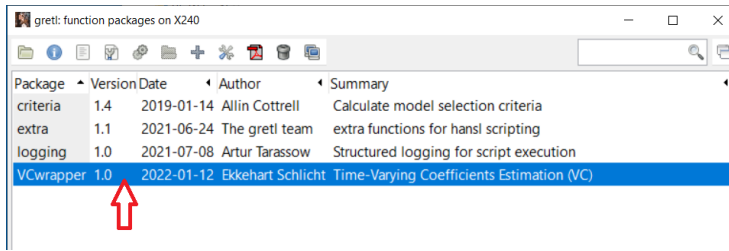


Fig. 2: Start VCwrapper

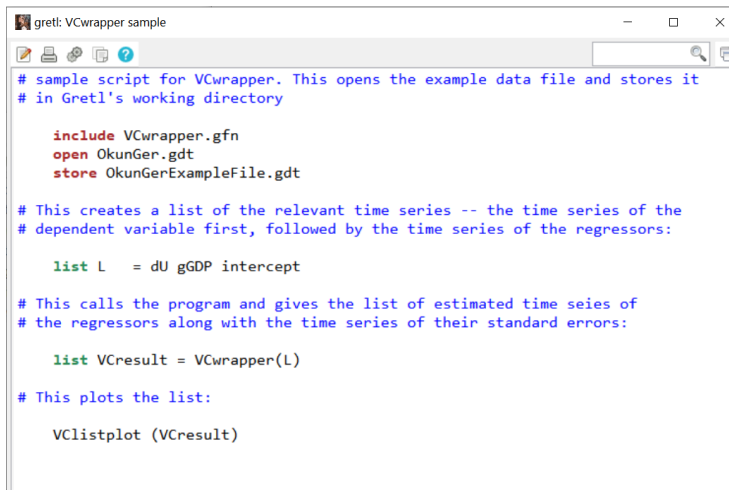


Fig. 3: Sample script

The loaded data set `OkunGer.gdt` comprises data about Okun's law for Germany. It is taken from a study by Jalles (2018) which covers many countries. In the main window, the data and the estimated time-varying coefficients for the regressors – denoted by `coeff_<name of regressor>` and their standard deviations – denoted by `stderr_<name of regressor>` – appear (Figure 4, see also Section 7 below).

The script invokes the function `VClistplot` that plots time series with confidence bands (Figure 5). It can be seen that the Okun slope is negative and decreases in absolute value over time while the intercept (the constant term of the Okun relationship at a given point in time) follows the business cycle with a downward trend.

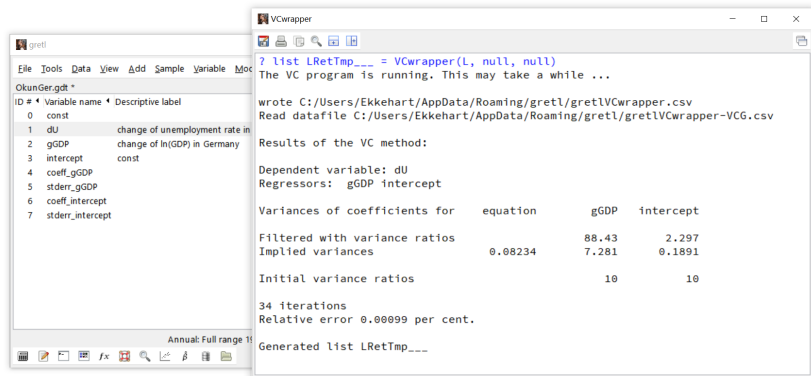
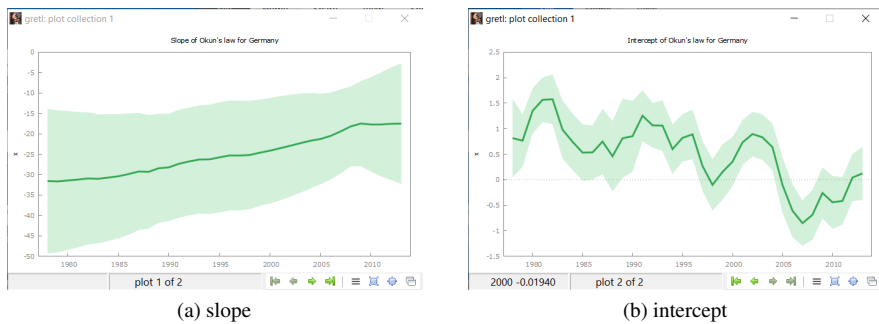


Fig. 4: VC output



(a) slope

(b) intercept

Fig. 5: Estimated time varying coefficients.

VCwrapper can also be invoked from the Gretl GUI. Load the data again by clicking File → Open data → User file..., and then go to Gretl's working directory as given at the top right of the main window. Open OkunGerExample-File.gdt. Now click again  $fx$  at the bottom of the window (Figure 2), scroll to VCwrapper (Figure 6) and click OK, or click in the menu Model → Time-varying coefficients (VC). A window will open that permits to select the time series to process (Figure 7). Select first the time series of the dependent variable (dU in the example) and then add the regressors (gGDP and intercept in the example).<sup>1</sup> Click OK twice and calculation will start. If many time-varying coefficients are to be estimated, this will take some time. Please wait until the calculations end, either because a result is obtained or because the maximum number of iterations is reached.<sup>2</sup>

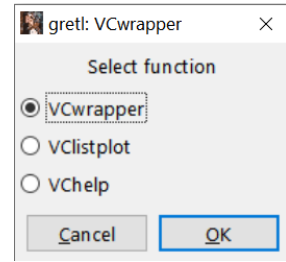
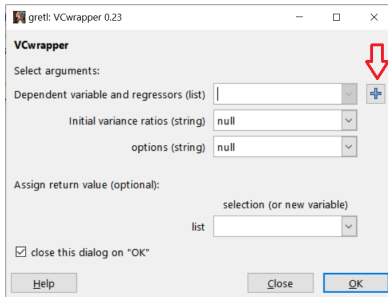
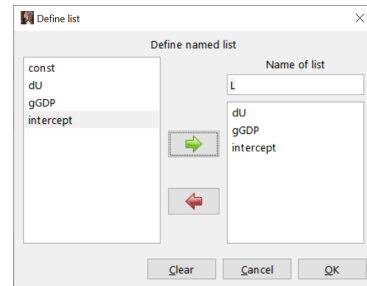


Fig. 6: VC menu



(a) Open selection window



(b) Select Series

Fig. 7: Data selection

## 5 Initial Variance Ratios

The variance ratios are the inverses of the smoothing weights, see equation (4). A high variance ratio indicates a volatile coefficient, whereas a low variance ratio stands for a more time-invariant coefficient. In the VC program, the variance ratios are determined by an iterative process that starts from the initial variance ratios. The process compares the variances of the disturbances  $u_t$  and  $v_{i,t}$  (equations (1) and (2)) with the estimates of these terms and stops if equality is achieved with a given precision.

If VCwrapper is called without supplying a vector of initial variance ratios for the regressors, the default values of 10 for each initial variance ratio are assumed (Figure 8a). You can supply other initial variance ratios by simply typing in the initial variance ratios, separated by spaces, in the initial variance ratio box (Figure 8b).

<sup>1</sup> The time-series intercept in the data set OkunGer.gdt equals Gretl's const which is (1,1,...,1). As const is a Gretl reserved term, its use creates, under Linux, sometimes problems that can be avoided by using the time series intercept = const instead.

<sup>2</sup> This can be set as an option, see Section 6 below. Default is  $10^5$ .

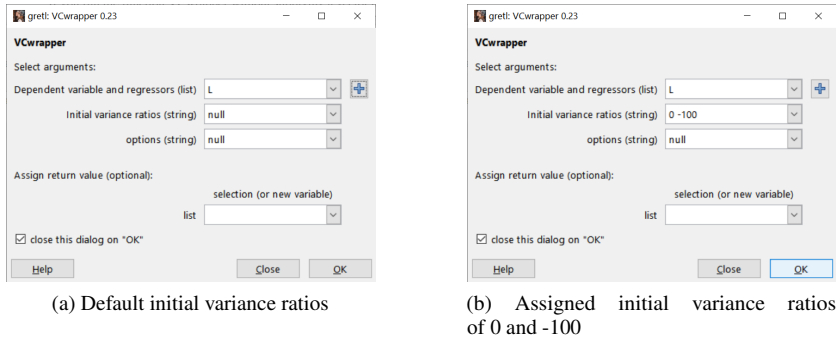


Fig. 8: Select initial variance ratios

The initial variance ratios induce the following:

- A positive initial variance ratio simply sets the starting point for the iterations. If the option `--filter` is set (see section 6 below), the program filters the time series with these variance ratios, see equations (3) and (4).
- An initial variance ratio of zero tells the program to treat the respective coefficient as time-invariant.
- A negative initial variance ratio tells the program to keep the variance ratio for the corresponding coefficient constant at the absolute value of the variance ratio. If all initial variance ratios are negative, the data are just filtered with these values.

The main purpose of selecting initial variance ratios is to determine which coefficients are to be taken as time-variant and which are to be considered as time-invariant, *i.e.* choosing for each regressor either the first or the second of the above three cases. If initial variance ratios are set negative, this allows filtering with the absolute values of the variance ratios. Extreme initial variance ratios may further be used for checking uniqueness. By selecting extreme values, the program will start from there and work along a path, covering many combinations. Non-uniqueness seems to be not an issue of practical importance, however, as it seems not to occur in practice, but it can not be excluded on theoretical grounds.

The output obtained, for example, by setting the initial variance ratio for the intercept to zero (a not very sensible choice, economically speaking) gives the results plotted in Figure 9. It can be seen that holding the intercept constant (which was originally estimated as unstable and trended) nearly eliminates the trend in the slope coefficient. The variance and the variance ratio of the other coefficient is decreased (which implies an increase in its smoothing weight  $\gamma$ ). The overall variance is shifted to the variance in the equation  $\sigma^2$ .

If the initial variance ratios of all coefficients are set to zero, all coefficients will be taken as time-invariant and the estimates will coincide with the OLS estimates.

## 6 Options

Some options are available that control the program in other ways.



Fig. 9: Estimated time varying coefficients with intercept forced to be constant.

- The option `--p n` sets the precision  $10^{-n}$ . If this difference between successive iterations is achieved, iteration stops. Default is `--p 5`.
- The option `--it n` gives the maximum number of iterations. Default is `--it 100 000`.
- The option `--f` or `--filter` filters with the inverse variance ratios as weights. It minimizes the weighted sum of squares  $Q$ , see equations (4) and (3).
- The option `--cov` prints the covariance matrix as a CSV file. Its path is given in the results page. The file is accessible for inspection and can be saved somewhere as long as the Gretl session is open. The matrix gives the expectation  $E \{ (a - \hat{a}) (a - \hat{a})' \}$  with  $a = \text{vec}(A)$  and  $A$  from equation (5). The vector  $a$  refers to the true time-varying parameters and  $\hat{a}$  denotes the VC estimates, see also Schlicht (2021, equation 2.28).
- The option `--smooth n` serves a purely descriptive purpose and does not carry any statistical significance. It decreases the estimated variance ratios by a factor of  $10^n$ . Default is  $n = 0$  (no smoothing). Smoothing increases the smoothing weights. An outcome with  $n = 2$  is depicted in Figure 10. A high smoothing value (8, for example) generates time-invariant coefficients that coincide with the corresponding OLS estimates.

4

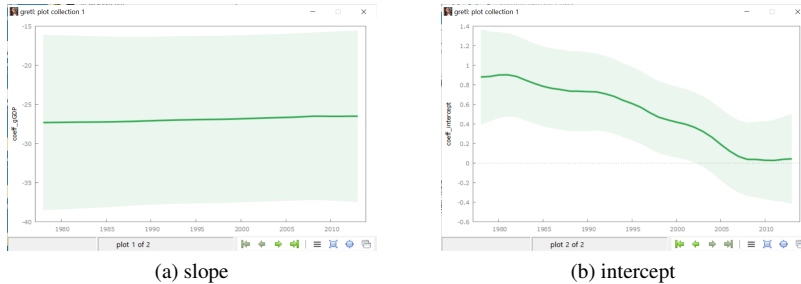


Fig. 10: Estimated time varying coefficients smoothed with factor 2. To see the effect of smoothing, compare this with Figure 5.

In scripts, estimation can be performed by calling the function `VCwrapper(L, vr, options)` which returns a list of time series of the estimated time-varying coefficients together with their standard errors. Its arguments are

argument	type	description	example
<code>L</code>	list	list of time series – dependent variable first, then regressors	see Figure 3
<code>vr</code>	string	initial variance ratios for the regressors	"2.5 -11" in case of two regressors
<code>options</code>	string	string of options	"-p 3 --cov"

Arguments that are left out are replaced by defaults, as in the example script (Figure 3).

## 7 Plotting

In addition to the plotting routines offered by Gretl, the VCwrapper package offers an additional function, `VClistplot`. This is easily configurable and can be invoked from the menu (Figure 6). The `VClistplot` window pops up (Figure 11).

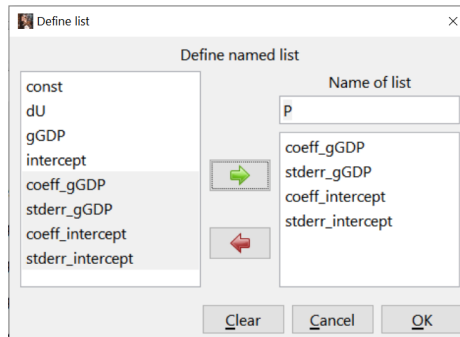


Fig. 11: The `VClistplot` window

A list of the estimated time-varying coefficients and the standard errors can be created. Confirming with OK in both windows gives the result depicted in Figure 5: The time series with a confidence band of  $\pm 1.96$  standard deviations which covers 95% of the cases if the disturbances  $u_t$  and  $v_{i,t}$  in equations (1) and (2) are normally distributed.

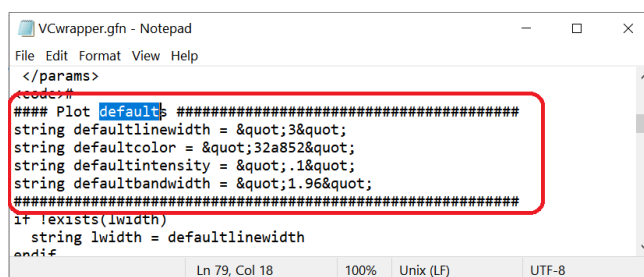
Some options are available that change the appearance:

- *linewidth* sets the thickness of the line. Default is 3.
- *linecolor* sets the color of the line as a Hex RGB number. Default is 32a852.
- *bandcolor* sets the color of the band as a Hex RGB number. Default is *linecolor*.
- *intensity* sets the intensity of the band, a value between 0 and 1. A value of 0 gives an invisible band, a value of 1 gives a solidly colored band. Default is 0.1.

For grayscale plots, set *linecolor* to 000000, for example.

The defaults can be changed by opening the file `VCwrapper.gfn` with any text editor such as Notepad. Searching for "defaults" gives the defaults (Figure 12). For grayscale graphics,





```

VCwrapper.gfn - Notepad
File Edit Format View Help
</params>
<code>#
#### Plot default #####
string defaultlinewidth = "3";
string defaultcolor = "32a852";
string defaultintensity = ".1";
string defaultbandwidth = "1.96";
#####
if !exists(lwidth)
  string lwidth = defaultlinewidth
endif
Ln 79, Col 18    100%    Unix (LF)    UTF-8

```

Fig. 12: Plot defaults

the string `"32a852"` would have to be replaced by `"000000"`, for example.

Further editing can be done by right-clicking on a plot such as Figure 5a. Select “extract plot”, right-click on the extracted plot and select “Edit” to open Gretl’s plot editor. For more features, right-click on the extracted plot and select “Save to session as icon”. In the icon view (third item at the bottom of Gretl’s main window) right-click on the icon of the time series that is to be modified and select “Edit plot commands” for editing.

In scripts, plotting can be performed by calling the function `VClistplot(P, linewidth, linecolor, bandcolor, intensity, options)`. Its arguments are

argument	type	description	example
P	list	list of of the time-varying coefficients of the regressors, each with the time-series of its standard error.	see Figure 3
linewidth	string	linewidth of estimated coefficients	“2.5”
linecolor	string	Hex RGB color number	“ff0000”
bandcolor	string	Hex RGB color number	“00ffff”
intensity	string	number between 0. and 1.	“.3”

Arguments that are left out are replaced by defaults, as in the example script (Figure 3).

**Acknowledgements** Riccardo Lucetti has suggested to make the VC method available for Gretl. Sven Schreiber of the Gretl team kindly provided the foundation of the VCwrapper.gfn function package by providing a Hansl script on which the current script is based. I thank João Tovar Jalles for permission to use the German data from his study (Jalles 2018) as an example.

## References

- Gretl (2022). *GNU Regression, Econometrics and Time-series Library*. URL: <http://gretl.sourceforge.net/>.
- Jalles, João Tovar (2018). “On the Time-Varying Relationship Between Unemployment and Output: What shapes it?” In: *Scottish Journal of Political Economy* 66, pp. 605–630. DOI: 10.1111/sjpe.12200.
- Schlicht, Ekkehart (2021). “VC - A method for estimating time-varying coefficients in linear models”. In: *Journal of the Korean Statistical Society* 50, 1164–1196. URL: <https://doi.org/10.1007/s42952-021-00110-y>.

## Appendix

### A Files in the VCwrapper Package

The package “VCwrapper.zip” contains the following files

VCwrapper.gfn	VC functions for Gretl
VCwrapper.pdf	VCwrapper documentation
OkunGer.gdt	example data file

and two folders

32bit	32 bit binaries for Windows and Linux
64bit	64 bit binaries for Windows and Linux.

Each of these contains the following four files

vcg.exe	compiled executable of VCG for Windows
vcg	compiled executable of VCG for Linux
vcg.c	source code for VCG in C
VChelp.exe	helper file for Windows that opens the documentation

### B Temporary files

The temporary files generated by VCwrapper can be found in the directories %AppData%\gretl for Windows and Home/.gretl for Linux. These files are accessible until the Gretl session is closed. The files are

VCTextoutput.txt	The text of the output screen as a text file
gretlVCwrapper.csv	the input for the VCG binary as a csv file
gretlVCwrapper-VCG.csv	the output of the VCG binary as a csv file
gretlVCwrapper-VCG-covariances.csv	the covariance matrix as a csv file if option --cov has been selected

### C License

The files in VCwrapper ©2022 by Ekkehart Schlicht and Sven Schreiber. They are published under the GNU General Public License <https://www.gnu.org/licenses/gpl-3.0.html> and are free for non-commercial use.