QMLSV User's Guide

Chirico Paolo

August 2025.

1 The *QMLSV* package

The gretl package QMLSV allows the estimation of the following univariate stochastic volatility model:

$$\begin{array}{l} r_t = \mu + \sigma_t^2 \varepsilon_t \\ h_{t+1} = \omega + \beta h_t + \eta_t \end{array} \qquad \begin{bmatrix} \varepsilon_t \\ \eta_t \end{bmatrix} \sim NID\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & \gamma \\ \gamma & \sigma_\eta^2 \end{bmatrix} \right) \tag{1}$$

where: r_t is the log-return on an asset, simply "return" in the following; $h_t = \ln \sigma_t^2$, i.e. the log-square volatility, is an order-1 autoregressive latent process; ε_t and η_t are zero-mean Gaussian innovations[1], serially independent, with $E(\eta_t \varepsilon_{t-k}) = \gamma$ if $k = 0$, and zero otherwise.

The estimation is performed using three alternative QML methods[2]:

1. the standard QML method (Ruiz, 1994; Harvey et al, 1994);
2. the modified QML method proposed by Harvey and Shephard (1996) (HS-QML);
3. the Iterative QML method (Chirico, 2024) (IQML).

All three methods can be applied under two alternative assumptions about the distribution of the return innovation ε_t:

1. Gaussian distribution;
2. Student-t distribution with given degrees of freedom.

[1] Actually, the return innovation ε_t can also be assumed Student-t distributed (see below).
[2] See Chirico (2024) for the features of the methods.

and two about the type of volatility:

1. stationary volatility;
2. random walk volatility.

The package consists in three public functions:

- `qml_sv`, which performs the QML estimation with the the chosen method;
- `qml_fc` which performs a *out-of-sample* volatility forecast for a given estimated model;
- `qml_plot` which applies the gig package's graphical functions (Lucchetti and Balietti, 2018) to a given estimated model;

and four auxiliary functions which serve the operation of the main function `qml_sv`. The syntax of `qml_sv` is:

```
bundle Mod = qml_sv(returns, method, integ, constant, smooth,
degrees, vest)
```

where:

- `returns (series)` is the series of return;
- `method (int)` sets the estimation method: 1 = standard QML, 2 = HS-QML, 3 = IQML;
- `integ (bool)` fixes the type of volatility: 0 = stationary volatility 1 = integrated volatility (see below);
- `constant (bool)` set the model with the constant μ or without it: 0 = no constant, 1 = constant;
- `smooth (bool)` set the output in the bundle: if =1, the smoothed state $\tilde{h}_t$ is given, otherwise (=0) the one-step-ahead expected state $\hat{h}_{t|t-1}$ is given[3];
- `degrees (scalar)` are the degrees of freedom set for the Student-t distribution of ε_t. The maximum accepted value is 100, which is also the default value. By convention the default values implies Gaussian ε_t;
- `vest (bool)` if =1, provides the ex-post ML estimate of the Student-t degrees of ε_t distribution using the standardised residuals derived from the previous model estimation.

In case of random walk volatility the parameters ω and β are fixed to 0 and 1 respectively, and the gretl's Kalman filter option `diffuse=2` is set[4].

Bundle objects

The *qml_sv* function returns a bundle containing several estimation and filtering/forecasting results, in particular:

- `vol (series)` is the series of the filtered (or smoothed if `smooth=1`) volatilities;
- `stduhat (series)` is the series of the filtered (smoothed) return innovations;
- `coeff (matrix)` is the vector of the model parameter estimates;

[3]In both cases, step 3. of the procedure uses the smoothed state $\tilde{h}_t$.
[4]See Chapter 36 of the Gretl User's Guide (Cottrell and Lucchetti, 2025).

- SE (matrix) is the vector of the corresponding standard errors.

Forecasting

The qml_fc function integrates a bundle/model previously created by qml_sv with a matrix (fcast) containing the in-sample and out-of-sample forecasts. The function also produces an output window with the out-of-sample forecasts. The qml_fc syntax is: qml_fc(&Mod, horizon), where Mod (bundle) is the model for which forecasting is required; horizon (int) is the number of times out of sample for for forecasting.

Graphs

The qml_plot function applies the following graphical functions native of the gig package:

1. gig_plot which performs the time-series plot of mean-corrected returns (residuals) and predicted/smoothed volatilities[5];
2. gig_dplot compares the kernel-estimated density distribution of the standardised residuals (estimated return innovation) with the theoretical distribution assumed for ε_t.

The syntax function is qml_plot(&Mod, vplot, dplot), where Mod (bundle) is the model for which graphs are required; vplot (bool) = 1 generates the time series plot of residuals and volatilities; vplot (bool) = 1 generates the kernel density plot of std residuals.

2 Examples

The following examples are carried out using a dataset available in gretl library: the Bollerslev-Ghysels dataset, which reports 1974 nominal returns on Mark/Pound exchange rate in the period from January 3, 1984, to December 31, 1991 (Bollerslev and Ghysels, 1996). Since the package's functions are applicable to both the entire dataset and subsamples, the following examples use the subsample corresponding to the last 500 observations (approximately 1990 and 1991). On this subsample, we estimate model (1) with the standard QML method:

After loading the dataset and setting the subsample with the following commands:

```
open b-g.gdt
smpl 1475 1974
include QMLSV.gfn
bundle Mod1 = qml_sv(Y,1,0,1,1,100,1)
```

[5]The graph title remains "Residuals and Conditional Standard Deviation" which is consistent with conditional heteroskedasticity models (Bollerslev (1986)). In this case (stochastic volatility models), it should be changed to "Residuals and Volatilities": once the graph has been created, click on the menu icon at the bottom right and then on "edit".

where Y is the series of the nominal return on Mark/Pound exchange rate. The QML standard estimation also includes the following settings: stationary volatility; constant; smoothed volatility; Gaussian return innovation (i.e. `degrees=100`); ex-post Student-t degrees estimation. The resulting output is:

```
STOCHASTIC VOLATILITY MODEL BY QML ESTIMATION
Dependent variable: Y
Used sample: 1475 - 1974, Valid obs. T = 500

Constant estimation:

coefficient   std. error     z      p-value
-----------------------------------------------------
const      0.00520612    0.0180946    0.2877   0.7736


Volatility model:

Type of volatility: stationary volatility
Estimation method: Standard QML
Gaussian return innovations


coefficient   std. error     z      p-value
-----------------------------------------------------
s_eta       0.192473     0.0635437     3.029   0.0025  ***
omega      -0.0574553    0.0370129    -1.552   0.1206
beta        0.976695     0.0149155    65.48    0.0000  ***


Llik: -1149.0366       AIC:  2306.0732
BIC:   2322.9316       HQC:  2312.6884

Ex-post estimation of Student-t degrees:

coefficient   std. error     z      p-value
------------------------------------------------------
v            7.36963      1.20611     6.110   9.95e-010 ***
```

The ex-post estimate of v indicates that the standardized residuals have heavy tails and that it would seem more appropriate repeating the QML estimation setting a value of v (e.g. 8) close to the estimated value (7.39). Furthermore, the estimates of `omega` and `beta` seem to leave room for the hypothesis of random walk volatility. Furthermore, the presence of the leverage effect may be hypothesized. QML estimation with the QML-HS method in case of integrated volatility and Student-t_8 return innovations (besides constant and smoothed states), is performed by the command:

```
bundle Mod2i = qml_sv(Y,2,1,1,1,8,1)
```

Once a model has been estimated, and its bundle created, functions `qml_plot` and `qml_fc` can be used to make graphs and forecasts. The command `qml_fc(&Mod1,10)` produces a matrix (`Mod1.fc`) containing the in-sample and of-sample forecasts, and the following output:

```
Out of sample forecasted volatility:

horizon       forecast
+  1          0.31178
+  2          0.31129
+  3          0.31081
+  4          0.31035
+  5          0.30990
+  6          0.30945
+  7          0.30902
+  8          0.30860
+  9          0.30819
+ 10          0.30779
```

The command `qml_plot(&Mod1,1,1)` produces the plot of the time series of residuals and volatilities and the plot of the density distribution of the standardized residuals (Fig. 1).

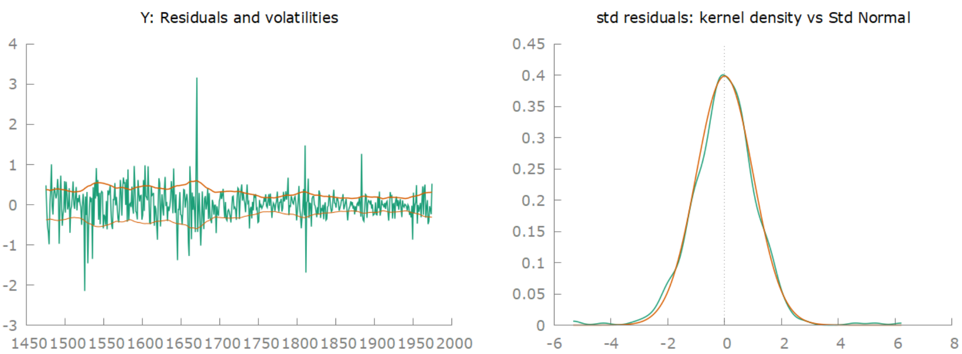

Fig. 1 volatility time series and kernel density of residuals

References

Bollerslev T (1986) Generalized autoregressive conditional heteroskedasticity. Journal of Econometrics 31:307–327

Bollerslev T, Ghysels E (1996) Periodic autoregressive conditional heteroscedasticity. Journal of Business & Economic Statistics 14(2):139–151

Chirico P (2024) Iterative qml estimation for asymmetric stochastic volatility models. Statistical Methods & Applications 33(3):885–900

Cottrell A, Lucchetti RJ (2025) Gretl user's guide. online, https://gretl.sourceforge.net/gretl-help/gretl-guide.pdf

Harvey A, Shephard N (1996) Estimation of an asymmetric stochastich volatility model for asset returns. Journal of Business and Economic Statistics 14(4):429–434

Harvey A, Ruiz E, Shepherd N (1994) Multivariate stochastic volatility methods. Review of Economic studies 61:247–264

Lucchetti RJ, Balietti S (2018) The gig package. online, http://gretl.sourceforge.net, version 2.3

Ruiz E (1994) Quasi-maximum likelihood estimation of stochastic volatility models. Journal of econometrics 63(1):289–306