



April 15, 2019

---

# Financial Econometrics report

---

*Authors Group 2:*

Abdullah KETELDIJK (2561520)  
a.keteldijk@hotmail.com

## **Abstract**

In this report we describe a method of automating the process of volatility model comparison. We demonstrate how a the method uses different types models and probability distributions to predict stockmarket volatility as a proof of concepts. However this method is not without its challenges that still have to be worked out. These challenges are also described in this paper.

# 1 Volatility Model Generator Method

During the process of programming we found that only a few variables changed for every new model or distribution implemented. This prompted us to write a method that enables us to add any model or distribution by only adding the formula for it. It allows to estimate models in a very efficient way and it makes code maintenance easier.

We called this method the Volatility Model Generator (VMG). To test it, we used fifteen volatility models and five probability distributions (see appendix). The method allows us to estimate and compare all possible combinations of these models and distributions. In total this resulted into 75 different models, which we compared by means of the Diebold-Mariano test (see appendix).

The method works as follows. First we define the model names and distribution names in a list. This list is then used for parameter initialization. Each distribution has its own initialization parameters and parameter bounds. These are used as input for the `fmincon` optimization algorithm (in Matlab), together with the returns. We also include the names of the model and distribution as a string so it can be used for if statements to determine what formula should be used.

After this we used the model name to determine which updating equation should be used for the  $h_t$  (conditional variance). We use the distribution name to determine which log likelihood function needs to be used. The GAS models uses an updating equation that includes a score function. We therefore made sure that this is applied separately from the standard updating equation. We made a similar exception for the log likelihood function of the Realized GARCH model. Here we made sure that for this model the second part of the log likelihood function was also included. For this part we only used the Gaussian distribution. This process is then repeated for every volatility model and distribution by means of a double for loop (one for each model, one for each distribution).

Next we use the estimated parameters to forecast in the out-of-sample period. For this we used a similar approach of double for loops to account for all the estimated models. The results are then used to construct the Diebold-Mariano table (see appendix). We applied Newey West standard errors to calculate the test statistics by means of MAE. Finally we calculate the AIC and BIC in order to compare the models their in sample performance.

We used this model as a proof of concept to show that this is a more efficient way to find the best volatility model given different probability distributions. Not only is it very easy to add or edit different models, it also clears the clutter of having too many programs in one folder or having too many lines of code in one file. We think this method can be used to speed up research volatility models by allowing very quick model comparison with just one click of the button.

## 1.1 Further Research

We think that the VMG would be a very promising next step for our research. Due to time constraints we were not able to completely analyse the results of our VMG method. This would allow us to very easily test additional models and distributions. However it should be noted that the VMG method can still be developed further. One feature that we wanted to implement is a function that selects the top "n" best performing models. This could be done by using an error measurements like MSE or MAE, but als by using the DM test or the AIC and BIC. These measurements can all be used to automatically evaluate a large number of methods.

Furthermore there are some issues regarding the model estimation that we would like to address. First we would like to improve the initialization. Right now the parameters are initialized for each distribution separately. As a consequence the initialization includes parameters for all the models regardless of the model that is estimated. The parameter estimates that are not used, barely change during estimation, however this might lead to issues with other optimization algorithms like Newton-Raphson or Max BFGS. It would lead to one or more rows and columns of zero, which makes it impossible to calculate the inverse of the Hessian matrix. However, this issue can be solved by slightly changing the initialization structure of our method.

Next we would like to change the way we calculate the AIC and BIC of the realized GARCH model. At the moment our VMG method takes the whole likelihood value when calculating the VMG, while it should only be the first part for the AIC and BIC (In order to calculate the correct AIC and BIC, we made a separate program). We also would like to use multiple distributions for the second part of the log likelihood. Finally we would like to add backtesting and VaR to the VMG method. These changes would allow us to deliver the all results of the models discussed in this paper very easily.