MS-GARCH Analysis

**CUNY SPS MSDS**

**Professor Samuel Gralnick**

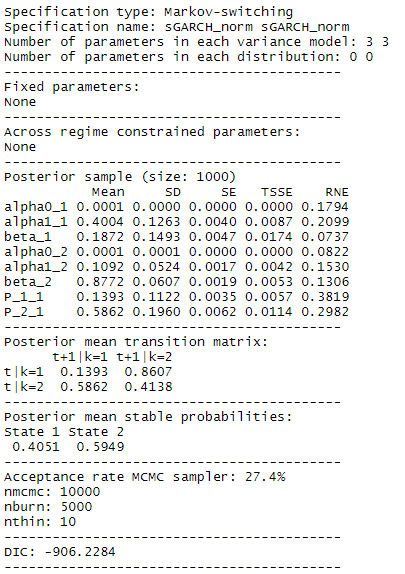
**Team Sugar Cane: Euclid Zhang, Jie Zou, Zhenni Xie**

**Fall 2022**

Previously, we demonstrated how to construct a GARCH model to analyze the volatility of an asset with heteroscedastic returns. In this analysis, we will examine if the Markov-Switching GARCH model performs better than the plain vanilla GARCH model in modeling the changing volatility of an investment asset.

We use the *MSGARCH* package in R to construct our model. The package provides two methods of constructing an MSGARCH model, the Maximum Likelihood method and the MCMC/ Bayesian method. We will focus on the MCMC/ Bayesian method in this analysis.

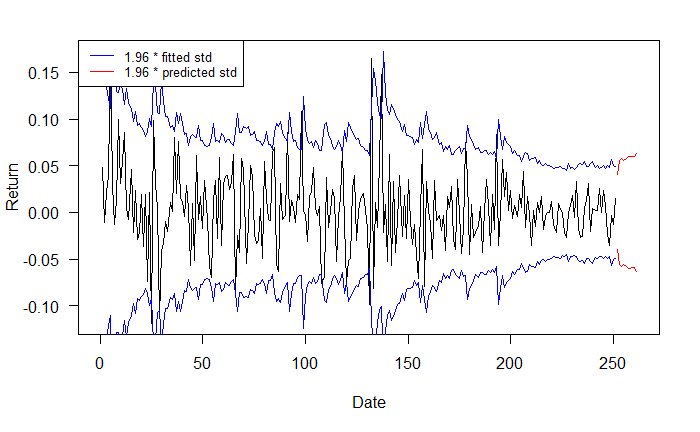
We will demonstrate the MSGARCH process using one of our assets, 'HSON', as an example. The following is the summary of the MSGARCH model with the number of states K=2.



The following are the predicted volatility for the next 10 days.

0.02050390 0.02850171 0.02955341 0.02848627 0.02915753 0.03016917 0.03047897 0.03027598 0.03036563 0.03210727

The following plots the 95% interval of the returns based on the fitted and predicted volatilities



The MSGARCH package also has built-in functions to calculate the VaR and Expected Shortfall of the predictions. These are useful tools for our investment risk management.

VaR:

0.01 0.05

252 -0.06580569 -0.04181346

253 -0.07865094 -0.04584804

254 -0.07711577 -0.05182348

255 -0.07041068 -0.04645848

256 -0.07301959 -0.04814404

257 -0.07126298 -0.04517874

258 -0.07930827 -0.04996169

259 -0.07532402 -0.04945312

260 -0.07523162 -0.04659365

261 -0.07541612 -0.05068879

Expected Shortfall:

0.01 0.05

252 -0.07915345 -0.05609930

253 -0.09617402 -0.06375279

254 -0.08336463 -0.06763657

255 -0.09291861 -0.06351952

256 -0.09127506 -0.06553790

257 -0.09613131 -0.06533303

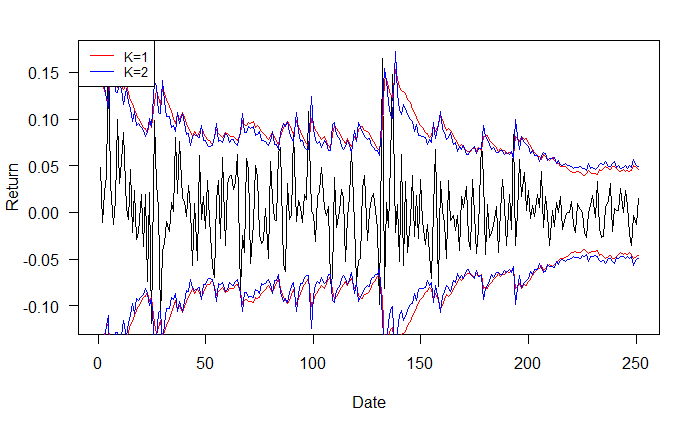
258 -0.10262766 -0.06999160

259 -0.08709552 -0.06526247

260 -0.09443796 -0.06714893

261 -0.09616612 -0.06845797

We can compare the estimated volatilities of the MSGARCH (K=2) and the simple GARCH, which is a MSGARCH model with a number of states K=1. The following plot shows the 95% interval of the returns based on the fitted volatilities of the two models.



The volatilities from the two models are very closed, but the MSGARCH model has higher volatilities in the extreme cases. The MSGARCH model has better flexibility in fitting the data because it can switch between states. So does it mean MSGARCH is better than GARCH? It may be more reliable to use the MSGARCH model to analyze the volatility of an asset. However, it may be impractical to do so for a large set of assets because it requires more time and compute power due to its increased complexity.

In our example above, we used an MSGARCH model to estimate the volatility of the returns by simply assuming that the mean return is 0. The process would be much more complicated if we add a mean model for the returns. Also, the *MSGARCH* package only produces models with 2 alphas and 1 beta for each state. Let’s disregard the limitation of the package for now. Theoretically, a MSGARCH model should have the following tuning parameters:

* The number of states K
* The number of AR terms in the mean model
* The number of MA terms in the mean model
* The number of Alpha terms in each state
* The number of Beta terms in each state

We would need to perform cross-validation or best testing in tuning the parameters to find the optimal model. This is time consuming, and investment is time sensitive. Moreover, we see from the above example that the estimated volatilities from the MSGARCH model and the GARCH model are very close. Using a MSGARCH model may not provide notable benefits compared to using a GARCH model.

“All models are wrong, but some are useful.” No single model is perfect. Whether GARCH or MSGARCH should be used is based on the circumstance. For small portfolios, we may use MSGARCH models since they are more flexible in fitting the data. For large portfolios, we suggest using GARCH for the majority of the assets and use MSGARCH for the riskiest ones.