MATH60633A – TP1

David Ardia

Important information

- This assignment is worth 10 % of the final grade.
- This assignment must be done in groups of 2-3 people; fill out the peer contract sheet.
- Send me an e-mail at david.ardia@hec.ca as soon as you have created your team.
- This assignment cannot be done using ChatGPT.
- I will only answer questions asked on the forum.
- Your R file must be uploaded to ZoneCours by the specified date. Put all your files in a single .zip folder and name it `name1_name2.zip`. Any delay will result in a penalty of 5% per hour.
- Clearly document all of your R code.
- You are not allowed to use a package for the GARCH model (e.g. **rugarch**).
- You can watch the course *GARCH Models in R* to get an introduction to GARCH models in finance.
- The total number of points is 30.
- The work presented below counts for 27 points, or 90% of the grade. The points per question are indicated in the square brackets.
- The remaining 3 points, or 10% of the grade, can be obtained through one of the following 4 options:

1. Complete the assignment in Rmarkdown.
2. In addition to the model described below, implement the asymmetric version GJR-GARCH.
3. Formally test the results of your backtest of the resulting VaRs with a test available in the package **GAS**.
4. Using your estimation results, a) standardize the returns (innovations) of the two indices using the conditional volatilities of the models, b) compute the correlation between the innovations, c) generate 1,000 scenarios for the innovations using a bivariate Gaussian using the estimated correlation, d) generate 1,000 returns for the two series using the innovations, e) generate 1,000 scenarios for an equally-weighted portfolio in the two indices for the time horizon of $T+1$, f) compute the VaR 95% of the portfolio.

Description

The objective is to estimate and backtest the value-at-risk (VaR) of portfolio of two indices using the GARCH(1,1)-Normal model. The code provided in the files is not working. You need to fix and complete it. You need to create a proper project structure with main script for running the code. This should be platform independant. Everything should be generated automatically.

Details

The GARCH model is used to capture the heteroscedasticity of financial data: the dynamics of volatility. It is widely used in risk management. You can use it to forecast the volatility or forecast the value-at-risk, the conditional quantile of the predictive distribution of asset returns.

The GARCH(1,1) model with Normal error for returns y_t is given by:

$$y_t \sim \sigma_t \, \epsilon_t \quad \epsilon_t \sim iid \, \mathcal{N}(0,1) \, ,$$

where $\sigma_t^2 = \omega + \alpha y_{t-1}^2 + \beta \sigma_{t-1}^2$. The parameters must satisfy $\omega \geq 0$ and $\alpha, \beta > 0$. In addition, the model must be covariance-stationary, that is $\alpha + \beta < 1$.

The VaR at a given time point t^* is given by $\Phi^{-1}(1-\gamma)\sigma_{t^*}$, where γ is the risk level (usually 0.95 or 0.99) and Φ^{-1} is the inverse of the cumulative normal distribution (quantile function).

Steps

Project structure

- Create a project in RStudio and a proper folders structure. [1 point]
- Code properly (style, naming, etc). [3 points]

Data

- Load the file `indices.rda` into R. Consider only index values since January 2005. [1 point]
- Compute the log-returns for both series. [1 point]

Functions

- Fix the R function `f_forecast_var.R` which computes the next-step-ahead VaR forecast using the GARCH model with normal errors. The function's inputs are a $(T \times 1)$ vector of past log-returns `y` and the Value-at-Risk level `level`. The estimation of the model is done by maximum likelihood. The function outputs the next-step-ahead VaR at the desired risk level, `VaR`, the $(T+1 \times 1)$ vector of conditional variances `sig2` and the set of MLE `theta`. [10 points]

Static estimation of the VaR

- Use the first $T = 1000$ log-returns to estimate the VaR of each index at the 95% risk level. Which one is the most risky at the $T+1$ horizon? [3 points]

Backtesting

- Using a rolling window of $T = 1000$ days, compute and store the next-step-ahead VaR at the 95% risk level for the next 1000 days. [5 points]
- Display the series of realized returns and the VaR estimates for both series. Save in a *png* file. [2 points]
- Save the results of your backtest in a *rda* file. [1 point]