Realizing the future with R, C, and Java: A multi-language environment and GUI for high-frequency based volatility modeling

R/Finance, May 17/18, 2013

Chris Blakely

The Components

heavyModel is a computing environment and GUI for the analysis, estimation, and forecasting of volatility using robust indicators of daily volatility combined with HEAVY models. The environment GUI meshes the strengths of R, C, and Java to create an easy-to-use and fast interface.

- Realized Measures computed in R (*highfrequency* package, K. Boudt et al.)
- General High-frEquency BAsed VolatilitY (HEAVY) model (Shephard and Sheppard, 2009)
- Bayesian and Quasi-likelihood estimation of Heavy models (library written in gnu c using gsl)
- Java wrapper and GUI using Java Swing by Oracle for purely graphical and 'non-coding' interface for modeling

The Setup

- Denote the daily (log) returns as $r_1, r_2, \ldots, r_T$, where T total days in sample.
- Supplement information to the daily returns by a so-called *realized measure* of intraday volatility based on higher frequency data, such as second, minute or hourly data. We denote them as $RM_1, RM_2, \ldots, RM_T$ for the total number of days in the sample. Daily realized measures are an estimation of average of variance autocorrelations during a single day.
- Easiest is the realized variance computed as $RM_t = \sum_j (X_{t+t_{j,t}} - X_{t+t_{j-1,t}})^2$ where $t_{j,t}$ are the normalized times of trades on day t. Other methods for providing realized measures includes using Kernel based methods for smoother/cleaner results.

The HEAVY Model

With the realized measures computed for T days, the HEAVY model is given by:

$$\begin{aligned} Var(r_t|\mathcal{F}_{t-1}^{HF}) &= h_t = \omega_1 + \alpha RM_{t-1} + \beta h_{t-1} + \lambda r_t^2 \\ E(RM_t|\mathcal{F}_{t-1}^{HF}) &= \mu_t = \omega_2 + \alpha_R RM_{t-1} + \beta_R \mu_{t-1} \end{aligned} \tag{1}$$

where stability constraints are

- $\alpha, \omega_1 \geq 0, \beta \in [0,1]$ and $\omega_2, \alpha_R \geq 0$ with $\lambda + \beta \in [0,1]$ and $\beta_R + \alpha_R \in [0,1]$.
- $\mathcal{F}_{t-1}^{HF}$ denotes the high-frequency information from the previous day $t-1$.
- The first equation models the close-to-close conditional variance and is akin to a GARCH type model, whereas the second equation models the conditional expectation of the open-to-close variation.

Forecasting Volatility

Being important for asset allocation or risk assessment, is carried out through the use of an s-step ahead iteration

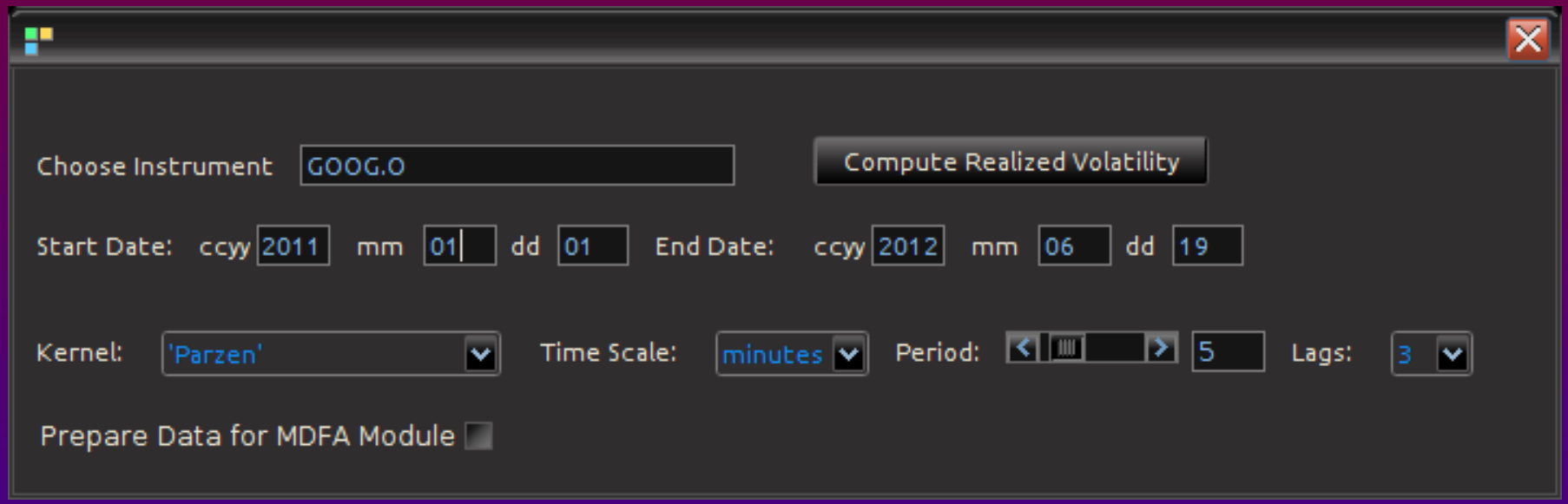
$$\begin{pmatrix} h_{t+s|t-1} \\ \mu_{t+s|t-1} \end{pmatrix} = (I + B + \cdots + B^s)w + B^{s+1} \begin{pmatrix} h_{t-1} \\ \mu_{t-1} \end{pmatrix},$$

where

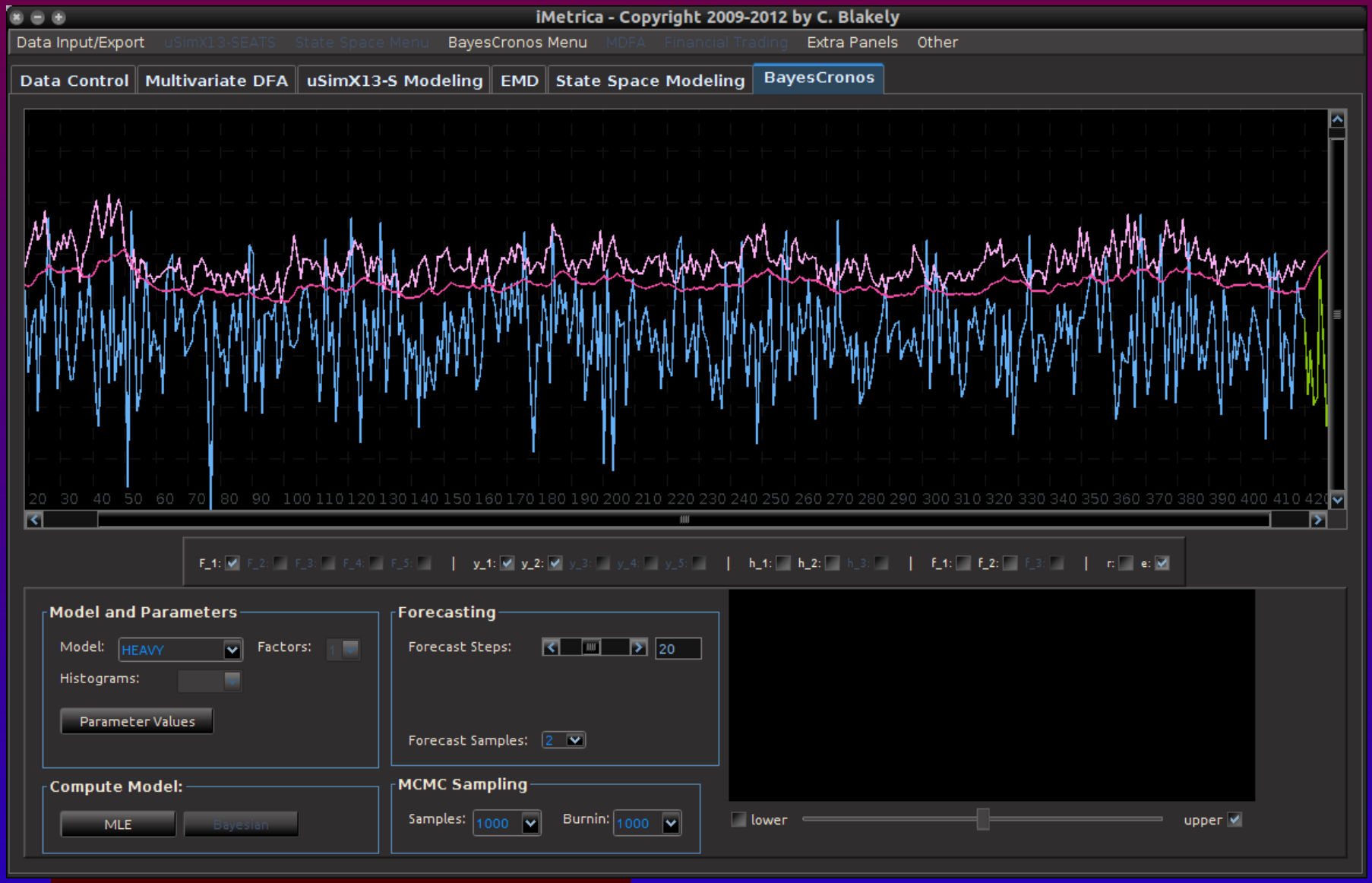
$$B = \begin{pmatrix} \beta & \lambda \\ 0 & \beta_R + \alpha_R \end{pmatrix}$$

- Forecasting distributions $F(r_{t+s}|\mathcal{F}_{t-1}^{HF})$ or $F(r_{t+1} + r_{t+2} + \cdots + r_{t+s}|\mathcal{F}_{t-1}^{HF})$ done using a model-based *bootstrapping* method.

The Java Interface



The Java Interface



The Deliverables

- The (open source) *heavyModel* package with the Java and C source can be downloaded at `sourceforge.net/projects/highfrequency/`
- Visit my 'Hybrid signal extraction' blog at `imetricablog.com` for more examples and details on this package and much more
- Personal website `c-blakely.com`