Coupling Jump Diffusion with Multifractal Volatility models

Yang Li

Business Dept, SIT

Advisor: Prof Khashanah

Introduction to Realized Variance

Traditional latent variable models: ARCH-GARCH, Stochastic volatility (SV) based on squared returns

- difficult estimation
- high frequency data not utilized
- standardized returns not Gaussian
- Imprecise forecasts
- multivariate extensions are difficult

Realized Variance Measures

New approach uses estimates of latent volatility based on high frequency data

- Volatility is observable
- Traditional time series models are applicable
- High dimensional multivariate modeling is feasible

Construction of Realized Variance Measures

- $p_{i,t} =$ log-price of asset i at time t
- $\mathbf{p}_t = (p_{1,t}, \ldots, p_{n,t})' = n \times 1$ vector of log prices
- $\Delta =$ fraction of a trading session associated with the implied sampling frequency,
- $m = 1/\Delta =$ number of sampled observations per trading session
- $T =$ number of days in the sample $\Rightarrow mT$ total observations

Example (Equity and Forex market):

- Prices are sampled every 1 minutes and trading takes place 6.5 hours per day $m = 390$ 5-minutes intervals per trading day $\Delta = 1/390 = 0.00256$
- Prices are sampled every 5 minutes and trading takes place 6.5 hours per day $m = 78$ 5-minutes intervals per trading day $\Delta = 1/78 = 0.0128$
- (Forex and Futures market): Prices are sampled every 30 minutes and trading takes place 24 hours per day $m = 48$ 30-minute intervals per trading day $\Delta = 1/48 \approx 0.0208$

Calculation

- Realized variance (RV) for asset i on day t

$$RV_{i,t}^{(m)} = \sum_{j=1}^{m} r_{i,t-1+j\Delta}^2, t = 1, \ldots, T$$

- Realized volatility (RVOL) for asset i on day t :

$$RVOL_{i,t}^{(m)} = \sqrt{RV_{i,t}^{(m)}}$$

- RV measures over h days for asset i:

$$RV_{i,t}^{(m)}(h) = \sum_{j=1}^{h} RV_{i,t+j}^{(m)}$$

Properties of RV

- Multifrequency volatility persistence
- Parsimonious
- Thick tails
- Convenient parameter estimation and forecasting
- Out-of-sample volatility forecasts and in-sample measures of fit significantly improve on standard models.

Multifractality of RV

- Distributions of differences in the log of realized volatility are close to Gaussian.
- Suitable to model σ_t as a lognormal random variable.
- Moreover, the scaling property of variance of RV differences suggests the model:

$$\log \sigma_{t+\Delta} - \log \sigma_t = \nu \left(W^H_{t+\Delta} - W^H_t \right)$$

where W^H is fractional Brownian motion.

Fractional Brownian motion (fBm)

Fractional Brownian motion (fBm) $\left\{W_t^H; t \in \mathbb{R}\right\}$ is the unique Gaussian process with mean zero and autocovariance function

$$\mathbb{E}\left[W_t^H W_s^H\right] = \frac{1}{2}\left\{|t|^{2H} + |s|^{2H} - |t-s|^{2H}\right\}$$

where $H \in (0,1)$ is called the Hurst index or parameter. In particular, when $H = 1/2, \mathrm{fBm}$ is just Brownian motion.

- If $H > 1/2$, increments are positively correlated.
- If $H < 1/2$, increments are negatively correlated.

Multifractality Feature Equation

When $h(q)$ varies with q, the series is multifractal. The multifractal nature is also characterized by the scaling exponent $\tau(q)$, the relation between $h(q)$ and $\tau(q)$ is

$$\tau(q) = qh(q) - 1$$

also we know multifractal spectrum function $f(\alpha)$, defined as follows

$$\alpha = h(q) + qh'(q)$$

$$f(\alpha) = q[\alpha - h(q)] + 1$$

Using MSM to Model RV

MSM Definition

$$r_t = \sigma\left(M_t\right)\varepsilon_t \quad \sigma\left(M_t\right) = \bar{\sigma}\left(M_{1,t}\ldots M_{\bar{k},t}\right)^{/2}$$

where $\epsilon_t \sim N(0,1)$ $\quad \gamma_k = 1-\left(1-\gamma_{\bar{k}}\right)^{b^{k-\bar{k}}} \approx \gamma_1 b^{k-1}$ where

$$M_{k,t} = \begin{cases} m \sim M(\theta) & \text{with probability } \gamma_k \\ M_{k,t-1} & \text{with probability } 1-\gamma_k \end{cases}$$

for the distribution of $M(\theta)$, any distribution with positive support will do the job as long as $E(m)=1$

Algorithm Design

- (i) Conduct multifractal detrending moving average algorithm(MF-DMA) to detect the existence of multifractality in the 1-min log returns of SPY.
- (ii) Perform MLE on the SPY 1-min log returns r_t and estimate the parameters of the MSM model.
- (iii) Estimate the volatility σ_t according to the MLE result in the training set and forecast $\hat{\sigma}_t$ in the testing set.

Implementation details

Basin-hopping iterates by performing random perturbation of coordinates, performing local optimization, and accepting or rejecting new coordinates .It is particularly useful algorithm for global in high-dimensional optimization.
2-step basin-hopping method combines global stepping algorithm with local minimization at each step.

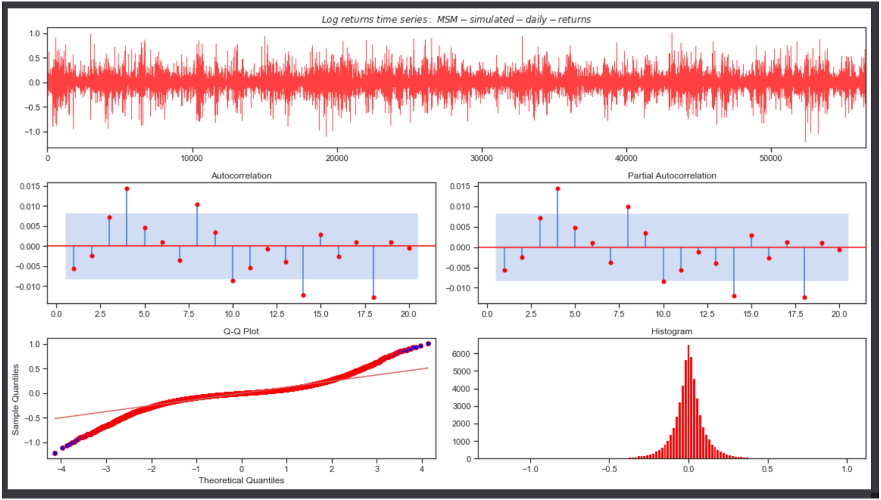
- step 1: local minimizations.
- step 2: global minimum search uses basin-hopping (scipy.optimize.basinhopping)

Optimizing to Get the Fitted Parameters

After doing so to the real data (SPY minutes data from 2016-2017 the following parameters are obtained Parameters from globalmin for real data:

- kbar = 4
- b = 14.83679
- m_0 = 1.53267
- γ_1 = 0.28359
- sigma = 0.00641
- Likelihood = -238538.76692

Using Above Parameter to Simulate Daily Returns



Research Tasks In the Future

Model design to incoporate jumps into the MSM model

```
for i in range(1,kbar):
    g_s[i] = 1-(1-g_s[0])**(b**(i))
for j in range(kbar):
    M_s[j,:] = np.random.binomial(1,g_s[j],T)
dat = np.zeros(T)
tmp = (M_s[:,0]==1)*m1+(M_s[:,0]==0)*m0
dat[0] = np.prod(tmp)
for k in range(1,T):
    for j in range(kbar):
        if M_s[j,k]==1:
            tmp[j] = np.random.choice([m0,m1],1,p = [0.5,0.5])
    dat[k] = np.prod(tmp)
dat = np.sqrt(dat)*sig* np.random.normal(size = T)   # VOL TIME SCALING
dat = dat.reshape(-1,1)
return(dat)
```

Algorithm to incoporate jumps for better fitting Method 1

in our simulation we use:
np.random.choice([m0,m1],1,p = [0.5,0.5])
For the marginal distribution $M(\theta)$, any distribution with positive support will do the job as long as E(m) = 1 for example, changing the distribution of M to Poisson distribution:

$$P[(N(t+\tau) - N(t)) = k] = \frac{e^{-\lambda\tau}(\lambda\tau)^k}{k!} \quad k = 0, 1, \ldots$$

then by applying scaling $\frac{1}{\lambda}$ to k, we have the positive support with E(m) = 1, then modify the above code to do simulations

Method 2: Modifying the transition frequencies using the Hawks Process

Definitions of Hawkes process

$$\mathbb{P}(N(t+h)-N(t)=m \mid \mathcal{H}(t))=\begin{cases}\lambda^*(t)h+\mathrm{o}(h), & m=1\\ \mathrm{o}(h), & m>1\\ 1-\lambda^*(t)h+\mathrm{o}(h), & m=0\end{cases}$$

In contrast to $\gamma_k = 1-(1-\gamma_1)^{b^{k-1}} \approx \gamma_1 b^{k-1}$

$$M_{k,t}=\begin{cases} m \sim M(\theta) & \text{with probability } \gamma_k\\ M_{k,t-1} & \text{with probability } 1-\gamma_k\end{cases}$$

we now use

$$\gamma_k^(t) = \gamma + \int_{-\infty}^{t} \alpha \mathrm{e}^{-\beta(t-s)} \mathrm{d}N(s) = \gamma + \sum_{t_i<t} \alpha \mathrm{e}^{-\beta(t-t_i)}.$$