Project Report: Sequential Monte Carlo with GARCH(1,1) Application

Xiwen J

November 2019

Introduction

This project focus on the Bayesian Inference via Sequential Monte Carlo (SMC, or Particle Filter in engineering) on the parameters of GARCH(1,1) model. Once achieve the posterior samplers of parameters, we can calculate the one step ahead prediction of the volatility, σ^2, as well as its confidence interval, with which we hope to create trading strategies that take advantage of this edge.

GARCH(1,1)

The general GARCH(p,q) model is defined as below:

$$y_t = \mu + \sigma_t * \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0,1)$$
$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i * (y_{t-1} - \mu)^2 + \sum_{j=1}^{p} \beta_j * \sigma_{t-j}^2 \tag{1}$$

where y_t is the observed log return of a financial assets, a typical asset would be SP500 index, μ is the mean return of the asset. In this project, we focus on the simplest case where $p = q = 1$ in (1).

Bayesian Inference for Time Series

Under Bayesian framework, the posterior is the probability of the parameter given observed values, which is defined as[**1**]:

$$p(\boldsymbol{\theta_t}|\boldsymbol{y_{1:t}}) = \frac{p(y_t|\boldsymbol{\theta_t}) * p(\boldsymbol{\theta_t}|\boldsymbol{y_{1:t-1}})}{\int_{\boldsymbol{\theta}} p(y_t|\boldsymbol{\theta_t}) * p(\boldsymbol{\theta_t}|\boldsymbol{y_{1:t-1}}) d\boldsymbol{\theta_t}} \tag{2}$$

where $p(y_t|\boldsymbol{\theta_t})$ is the likelihood density and $p(\boldsymbol{\theta_t}|\boldsymbol{y_{1:t-1}})$ is the rolling updating of parameters, which can be ultimately traced back to the initial prior $\boldsymbol{\theta_0}$. Once we have the posterior distribution, we can calculate the one step ahead posterior predictive distribution of a new value $\hat{y}_{t+1}$ by using[**1**]:

$$p(\hat{y}_{t+1}|\boldsymbol{y_{1:t}}) = \int_{\boldsymbol{\theta}} p(\hat{y}_{t+1}|\boldsymbol{\theta_t}, \boldsymbol{y_{1:t}}) * p(\boldsymbol{\theta_t}|\boldsymbol{y_{1:t}}) d\boldsymbol{\theta_t} \tag{3}$$

Given the difficulty of finding the analytical solution of these integrals, we use Monte Carlo methods to approximate.

Why SMC

Importance Sampling

Since usually we don't know what the posterior $p(\boldsymbol{\theta_t}|\boldsymbol{y_{1:t}})$ is, we cannot directly sample from it, so we turn to importance sampling with known distribution q to find the expectation, $\mathbb{E}[f(\boldsymbol{\theta_t})]$, in our GARCH(1,1) case, one can consider f as formula (1). It can be shown [**1**] that:

$$\mathbb{E}[f(\boldsymbol{\theta_t})] = \sum_{i=1}^{N} \tilde{W}_t(\boldsymbol{\theta_t^{(i)}}) * f(\boldsymbol{\theta_t^{(i)}}) \tag{4}$$

$$\tilde{W}_t(\boldsymbol{\theta_t^{(i)}}) = \frac{w_t(\boldsymbol{\theta_t^{(i)}})}{\sum_{i=1}^{N} w_t(\boldsymbol{\theta_t^{(i)}})} \quad \textit{and} \quad w_t(\boldsymbol{\theta_t^{(i)}}) \propto \frac{p(\boldsymbol{\theta_t^{(i)}}|\boldsymbol{y_{1:t}})}{q(\boldsymbol{\theta_t^{(i)}}|\boldsymbol{y_{1:t}})}$$

where $\tilde{W}_t(\boldsymbol{\theta_t^{(i)}})$ is the normalized importance weight and $w_t(\boldsymbol{\theta_t^{(i)}})$ is the non-normalized weight. This method is very inefficient since we need to calculate the wights all over again for every extra time step.

Sequential Importance Sampling

Sequential Importance Sampling is trying to estimate the posterior for all the previous time step, it can be written **recursively** as[**1**]:

$$p(\boldsymbol{\theta_{0:t}^{(i)}}|\boldsymbol{y_{1:t}}) \propto p(y_t|\theta_t^{(i)}) * p(\theta_t^{(i)}|\theta_{t-1}^{(i)}) * p(\boldsymbol{\theta_{0:t-1}^{(i)}}|\boldsymbol{y_{1:t-1}}) \tag{5}$$

$$w_t^{(i)} \propto w_{t-1}^{(i)} * \frac{p(y_t|\theta_t^{(i)}) * p(\theta_t^{(i)}|\theta_{t-1}^{(i)})}{q(\theta_t^{(i)}|\boldsymbol{\theta_{0:t-1}^{(i)}}, \boldsymbol{y_{1:t}})} = w_{t-1}^{(i)} * \frac{p(y_t|\theta_t^{(i)}) * p(\theta_t^{(i)}|\theta_{t-1}^{(i)})}{q(\theta_t^{(i)}|\theta_{t-1}^{(i)}, y_t)} = w_{t-1}^{(i)} * p(y_t|\theta_t^{(i)}) \tag{6}$$

We have equation (6) because in our case, $\boldsymbol{\theta_t}$ is IID. In some other general cases, we can often achieve this as well based on the important assumptions that the state variable θ_t is Markov chain and independent from observations.

Another problem come along when time step increase, the weight decays, which means weights for most samples will be zero, the expectation $\mathbb{E}[f(\boldsymbol{\theta_t})]$ will be biased, plus most of computation will be wasted on those zero weighted samples.

Sequential Importance Re-sampling

Re-sampling is one of the several methods invented to overcome this problem. When the effective sample size [**2**], $ESS_t = 1/\sum_{i=1}^{N}(W_t^i)^2$, is below a certain level, samples with larger normalized weights will be replicated more, smaller weight samples get replicated less, such that after re-sampling, the total number of samples will back to N and weight of each sample will be 1/N.

However, the problem after re-sampling is that many samples are the same due to replication, to diversify our final samples, a transition step is used.

MCMC Kernel Transition

A multivariate normal random walk is applied R_t times[**3**] to the samples from re-sampling, then the replications will transit into unique values to maintain a certain level of diversity.

SMC Algorithm[3]

Algorithm 3.2 SMC Sampler

Sample $\boldsymbol{\theta}_0^i \sim \pi(\cdot)$ and set $W_0^i = 1/N$ for $i = 1, \dots, N$
for $t = 1, \dots, T$ **do**
  Reweight: $w_t^i = W_{t-1}^i \frac{\eta_t(\boldsymbol{\theta}_{t-1}^i|\boldsymbol{y})}{\eta_{t-1}(\boldsymbol{\theta}_{t-1}^i|\boldsymbol{y})}$
  Set $\boldsymbol{\theta}_t^i = \boldsymbol{\theta}_{t-1}^i$ for $i = 1, \dots, N$
  Normalization of the weights: $W_t^i = w_t^i / \sum_{k=1}^N w_t^k$ for $i = 1, \dots, N$
  Computation of ESS: $\text{ESS} = 1/\sum_{i=1}^N (W_t^i)^2$
  if $\text{ESS} < \alpha N$, with $\alpha \in (0, 1]$ **then**
    Resample the particles and set $W_t^i = 1/N$, producing a new weighted particle set $\{W_t^i, \boldsymbol{\theta}_t^i\}_{i=1}^N$
    Move each particle with an MCMC kernel distribution of invariant distribution π_t for R_t iterations
  end if
end for

Model Results

I use existing package Pymc3 to do the Bayesian inference, it has the same SMC sampling method as mentioned above. Results are as below:

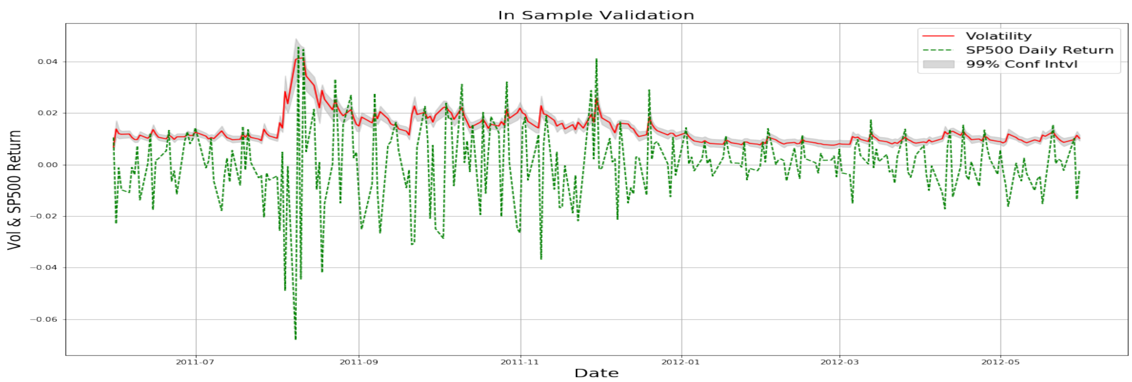

Figure 1: Historical Volatility & SP500 Daily Return

We can see that the volatility calculated from posterior distribution well captured the fluctuation and clustering in SP500 index, it also gives us the 99% confident interval of the historical volatility.

Rolling Forward Prediction

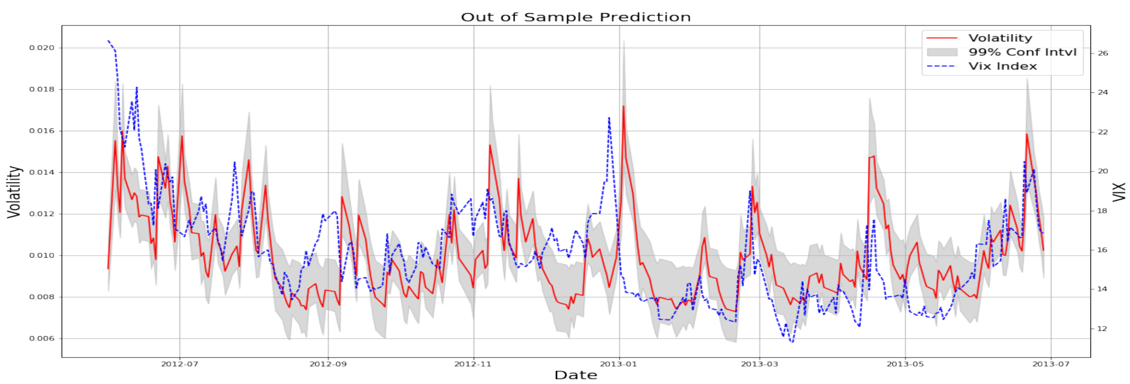

Figure 2: Rolling Forward Prediction of Volatility & VIX Index

The prediction is based on parameters from posterior distribution, once the market is close, we have one observation of SP500 return and we can calculate the next day's volatility, here for simplicity, I sample posterior only once using in sample observations, since the computation time is too long to sample iterativly rolling forward. The prediction basically captured the trend of the VIX evolution.

Simple Strategy and Result

Based on previous work, we can have one step ahead advantage of VIX index, a simple trading strategy can be constructed and the trading signals is shown as below:

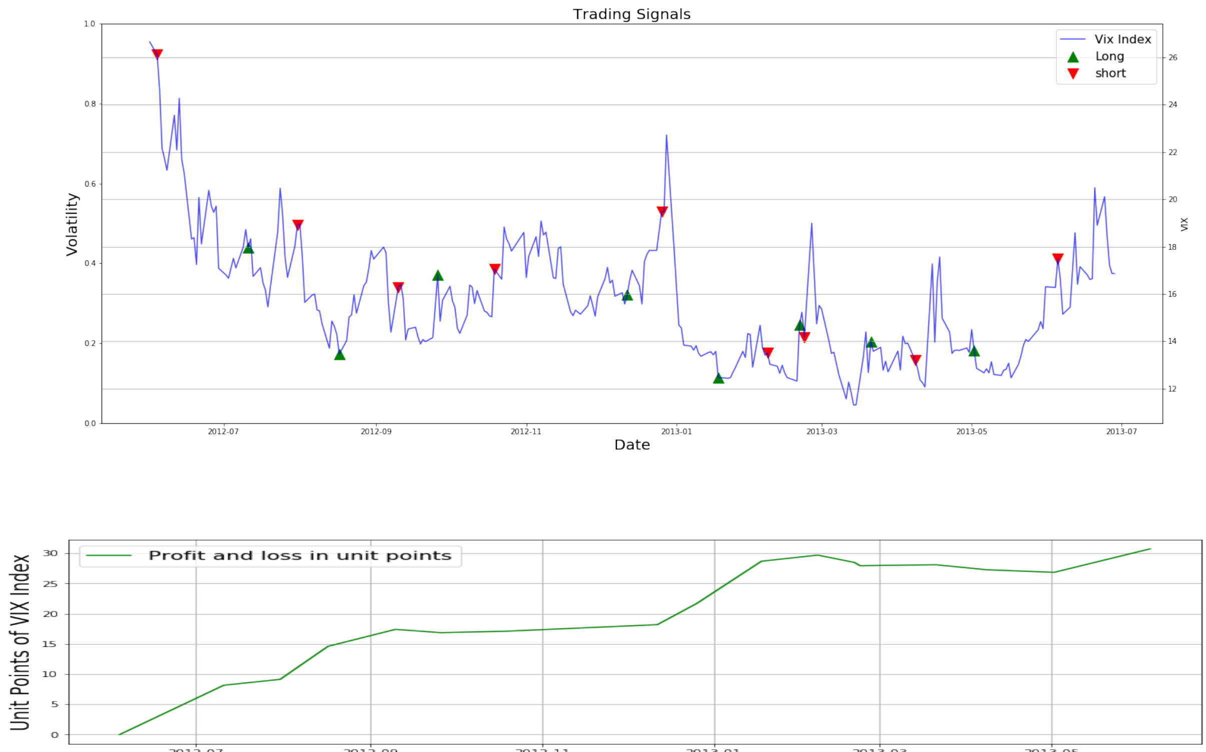

Figure 3: Trading Signals & Profit and Loss

This is just a simple case, not valid for more general cases, a lot more experiments need to be done to prove the validity of this strategy. Also this strategy is not quite good for intraday VIX data, probably we need some other strategies for intraday cases.

References

[1] Particle Filter Tutorial: from derivation to application
`https://blog.csdn.net/heyijia0327/article/details/40899819`

[2] Liu, J. S. (2008). *Monte Carlo Strategies in Scientific Computing.* Springer Science & Business Media.

[3] Dan L, Adam C and Christopher D (2019). *Efficient Bayesian estimation for GARCH-type models via Sequential Monte Carlo.* arXiv:1906.03828v1