

Volatility Estimation of Financial Returns Using Risk-Sensitive Particle Filters

Karel MUNDNICH, Marcos E. ORCHARD, Jorge F. SILVA, Patricio PARADA

Electrical Engineering Department, Universidad de Chile,
Av. Tupper 2007, Santiago, 8370451, Chile
morchard@ing.uchile.cl

Abstract: This article presents and analyzes the implementation of risk-sensitive particle filtering algorithm for volatility estimation of continuously compounded returns of financial assets. The proposed approach uses a stochastic state-space representation for the evolution of the dynamic system –the unobserved generalized autoregressive conditional heteroskedasticity (uGARCH) model– and an Inverse Gamma distribution as risk functional (and importance density distribution) to ensure the allocation of particles in regions of the state-space that are associated to sudden changes in the volatility of the system. A set of *ad-hoc* performance and entropy-based measures is used to compare the performance of this scheme with respect to a classic implementation of sequential Monte Carlo methods, both in terms of accuracy and precision of the resulting volatility estimates; considering for this purpose data sets generated in a blind-test format with GARCH structures and time-varying parameters.

Keywords: Bayesian estimation, stochastic volatility, particle filters, entropy.

1. Introduction

In finance, the term volatility refers to the standard deviation of the continuously compounded returns of a financial asset. Calling p the price or level of a financial asset, the returns of the financial index are defined according to

$$r_{k+1} = \log \left(\frac{p_{k+1}}{p_k} \right) \quad (1)$$

To define the concept of volatility in a proper manner, it is first necessary to introduce the concept of variance conditional to the returns:

$$\sigma_{k|\kappa}^2 = E \left[(r_k - \mu_{k|\kappa})^2 \middle| \Sigma_\kappa \right] \quad (2)$$

where $\mu_{t|\tau}$ is the conditional expectation of the returns. Considering that Σ_κ is the σ -algebra defined by the observations up to time κ , the financial returns can be modeled as a stochastic Gaussian process

$$r_k = \sigma_k \epsilon_k \quad (3)$$

where $\epsilon_k \sim N(0,1)$ and thus, $r_k \sim N(0, \sigma_k^2)$. Consequently, one can interpret volatility as a measure of the intensity of the variations undergone by a given financial asset or, equivalently, a measure of the risk associated to that asset. It is important to mention that different definitions may be associated to the concept of financial volatility; including “historic volatility”, which refers to the volatility experimented by an asset within a specific time period, and “current volatility”, which refers to the volatility of current prices.

This work is focused on the estimation of the latter, which is assumed to be time-variant.

From an information technology perspective, volatility estimation plays an important role in the development of trading algorithms for today’s stock market. Although several strategies that are currently in use include arbitrage and decision-making processes based on trend (moving average) evaluation, just few of those strategies are able to quantify (and manage) the risk that is associated to high volatility periods from the perspective of fault detection (and control) systems.

Considering Eq.(1)-(3), volatility estimation can be seen as a Bayesian filtering problem. In this regard, it is important to note that particle filters (also known as sequential Monte Carlo methods) have gained notable attention in the past few years, due to simple implementation and the excellent results they confer. Contrary to the Kalman filter, Particle Filters (PF) can be used to track the state trajectory in non-linear or non-Gaussian systems by approximating the state probability density function through a set of samples (called particles) and their correspondent weights. These characteristics make PF and their variants an interesting choice to solve the stochastic volatility estimation problem.

In literature, few examples and applications of Bayesian filtering applied in economics and finance may be found. (Harvey and Koopman, 2009) introduces a survey on the use of state-space models in economics applied to CPI and

GDP modeling. They also describe some of the most common volatility models, including the GARCH model (Bollerslev, 1986) and stochastic volatility (SV) models. He briefly describes the Kalman filter and PF, but does not show any results on stochastic volatility estimation. (Tsay, 2010) describes the use of Gibbs sampling in volatility estimation of SV models, but there is no mention about the use of PF for volatility estimation. (Tobar and Orchard, 2012) introduce the uGARCH model, stochastic variation of the GARCH(1,1) model, and use Kalman filters (classic and extended) and PF to compare the results of estimating stochastic volatility in the uGARCH and log-VE models. On the other hand, several examples can be found in literature that incorporate information-theoretic measures to analyze the output of particle filtering algorithms (Ajgl and Šimandl, 2011; Lanz, 2007; Boers *et al.*, 2010; Skoglaret *et al.*, 2009). Most of these are related to uncertainty characterization, optimality testing, and evaluation of control strategies. In (Orchard *et al.*, 2012), a detection scheme using entropy as a particle filter output processing tool is proposed.

This article proposes a risk-sensitive particle filter approach for volatility estimation. This approach also explores the use of differential entropy to implement a risk management scheme that detects high-volatility clusters based on estimates of the state probability density function. Validation of the proposed scheme is performed using ground truth data that is generated via a GARCH model. The structure of this document is as follows: In Section 2, the uGARCH model is introduced, and sub-optimal Bayesian filtering routines are referenced. In the same section, the concept of entropy is included, and insight is given towards its use in a detection scheme. In Section 3, the held experiments are described, and results of these experiments are displayed. Finally, in Section 4, conclusions are presented.

2. The uGARCH Model, Volatility Estimation, and Entropy

This section presents an introduction to the state-space stochastic uGARCH model for volatility representation (Tobar and Orchard, 2012), as well as a brief survey on risk-sensitive particle filtering (a technique used in sub-optimal Bayesian filtering). Finally, the concept of differential entropy is described in a

framework where it can be used to determine the uncertainty that is associated to PF-based estimates of the probability density function.

2.1 The uGARCH model

Economic and financial time-series are particular in the sense that the variances of the processes involved vary over time. (Mandelbrot, 1963) was one of the first to mention varying variances in return time-series, by describing clusters of high and low volatility. The uGARCH model (Tobar and Orchard, 2012) is a non-linear model that assumes non-observable volatility, and it allows the description of the evolution of processes with these characteristics over time. The uGARCH(1,1) model is

$$\sigma_k^2 = \omega + \alpha \sigma_{k-1}^2 \eta_{k-1}^2 + \beta \sigma_{k-1}^2 \quad (4)$$

$$r_k = \mu + \sigma_k \epsilon_k \quad (5)$$

where r_k is a process of returns, σ_k is the stochastic volatility, $\mu \in \mathbb{R}$, $\omega \in \mathbb{R}^+$, and $\alpha, \beta \in [0,1] \times [0,1]$ are parameters of the model that need to be estimated. Also,

$$\epsilon_k \sim N(0,1) \quad i. i. d., \forall k,$$

$$\eta_k \sim N(0, \sigma_p^2) \quad i. i. d., \forall k.$$

The following conditional probability density functions for state transition may be calculated from the model shown in Equations (4)-(5):

$$p(\sigma_k^2 | \sigma_{k-1}^2) = \frac{1}{\sqrt{2\pi\alpha\sigma_{k-1}^2(\sigma_k^2 - \omega - \beta\sigma_{k-1}^2)}} e^{\left(\frac{\omega + \beta\sigma_{k-1}^2 - \sigma_k^2}{2\alpha\sigma_{k-1}^2}\right)},$$

$$\sigma_k^2 \geq \omega + \beta\sigma_{k-1}^2 \quad (6)$$

$$p(r_k | \sigma_k^2) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{\left(\frac{-(r_k - \mu)^2}{2\sigma_k^2}\right)} \quad (7)$$

Equations (6)-(7) completely define the process that the uGARCH model states, and are the key of the implementation of Bayesian filtering routines used in this work.

2.2 Particle filters and risk-sensitive algorithms for volatility estimation

Bayesian non-linear filtering is defined as the process of using noisy observations to estimate at least the two first moments of a states vector governed by a non-linear and non-Gaussian dynamic system.

From the Bayesian perspective, a non-linear filtering procedure intends to generate an estimate of the *a posteriori* conditional probability density function of the state vector. The PF is an algorithm that aims to solve this estimation problem through an efficient selection of N particles and corresponding weights, so that the conditional probability density function of the state vector is approximated by the empirical distribution (Doucet, 1998; Maskell and Gordon, 2001; Arulampalam *et al.*, 2002; Thrun *et al.*, 2002):

$$p(x) = \sum_{i=1}^N w^{(i)} \delta(x - x^{(i)}), \quad (8)$$

$$w_k^{(i)} \propto \frac{\pi(x_k^i)}{q(x_k^i)} \propto \frac{p(z_k|x_k^i)p(x_k^i|x_{0:k-1}^i)}{q(x_k^i|x_{0:k-1}^i)}, \quad (9)$$

where $q(x_k^{(i)})$ is known as the importance sampling density function (Doucet, 1998; Maskell and Gordon, 2001; Arulampalam *et al.*, 2002). The determination of this importance density function is critical for the performance of particle filters. In the specific case of non-linear state estimation, the value of the weights $w_k^{(i)}$ of the particles is calculated after establishing that the importance density function is equivalent to the *a priori* state transition probability density functions, i.e., $q(x_k|x_{0:k-1}) = p(x_k|x_{k-1})$. Although the selection of this importance density function is appropriate to estimate the most likely probability distribution according to a specific data set, it is not intended for the tracking of high-risk and low-likelihood events (Thrun *et al.*, 2002), as events of sporadic high volatility may be considered.

In this sense, the risk-sensitive particle filter (RSPF) (Thrun *et al.*, 2002) incorporates a model of cost in the definition of the importance density function, to generate particles in regions associated to high-risk events. Mathematically, the importance density distribution is selected as follows

$$q(d_k, x_k | d_{0:k-1}^{(i)}, x_{0:k-1}^{(i)}, y_{1:k}) = \gamma_k \cdot r(d_k) \cdot p(d_k, x_k | y_{1:k}), \quad (10)$$

where d_k is a set of discrete values representing different models of volatility, x_k is a set of continuous states that describe the evolution of the financial system given the economic conditions, $r(d_k)$ is a positive function of risk

that is independent from the operation mode and γ_k is a normalization constant. This methodology has proven to be very adequate in FDI applications, improving the state tracking of states that are critical in the performance of a 6-wheel robot (Thrun *et al.*, 2002). Up to our knowledge, there are no publications of this technique in the area of volatility estimation. It is extremely important to note that this methodology uses exogenous models to evaluate and estimate the risk associated to economic conditions, a characteristic that makes the implementation more difficult in the absence of expert opinion.

The RSPF approach in this work ensures the existence of particles in the tail of the state transition probability density function, which are associated to higher risk and low likelihood. This implies the tracking of situations including model parameter change, and detection of high volatility clusters.

2.3 Entropy for PF-based detection schemes

Entropy is a concept that quantifies the amount of uncertainty of a random variable, as a function of its probability density function (Cover and Thomas, 2006). This concept is introduced formally as follows.

Let X be a discrete random variable, modeled by (Haykin, 2009): $X = \{x_k | k = 1, \dots, N\}$, where $X = x_k$ occurs with probability $p_k = \mathbb{P}(X = x_k)$, $0 \leq p_k \leq 1$, $\sum_{k=1}^N p_k = 1$. In particular, it is possible to define the gain of information after observing $X = x_k$ with probability p_k through the function (Hartley, 1927):

$$I(x_k) = \log \frac{1}{p_k} = -\log p_k,$$

where the base of the logarithm is arbitrary. Given the former facts, it is possible to define the entropy of a random variable X as the expected value of information gain for all the possible outcomes of X ; i.e.,

$$H(X) = E_X[I(x_k)] = -\sum_{k=1}^N p_k \log p_k.$$

In particle filtering-based applications, the probability density functions are defined conditionally with respect to a set of observations. For these cases (Cover and Thomas, 2006) introduce conditional entropy

as follows. If $(X, Y) \sim p(x, y)$, the conditional entropy $H(X|Y)$ is defined as

$$\begin{aligned} H(X|Y) &= \sum_{x \in X} p(x) H(Y|X = x) \\ &= - \sum_{x \in X} p(x) \sum_{y \in Y} p(y|x) \log p(y|x) \\ &= - \sum_{x \in X} \sum_{y \in Y} p(x, y) \log(p(y|x)) \quad (11) \\ &= -E[\log(p(Y|X))]. \end{aligned}$$

Expression (11) is typically used for the calculation of entropy in a particle-filtering scheme. Moreover, considering that the differential conditional entropy (12) –where the *a posteriori* state probability density function estimate (13) can be inferred from the likelihood of measurement y_k , the *a priori* state estimate $p(x_{k-1}|y_{k-1})$, and the probability of acquiring the current measurement–, then it is possible to use the properties of the logarithm to write (11) as in (14):

$$H(p(x_k|y_k)) = - \int p(x_k|y_k) \log p(x_k|y_k) dx_k, \quad (12)$$

$$p(x_k|y_k) = \frac{p(y_k|x_k)}{p(y_k|y)} p(x_k|y_{k-1}). \quad (13)$$

$$H(p(x_k|y_k)) = \log p(y_k|y_{k-1}) - \int p(x_k|y_k) [\log p(y_k|x_k) + \log p(x_k|y_{k-1})] dx_k \quad (14)$$

In addition, given that in this specific case all distributions correspond to particle-filtering estimates, both the *a priori* state estimate and the probability of measured data can be approximated by their corresponding sampled versions, as in (15)-(16):

$$p(x_k|y_{k-1}) \approx \sum_{i=1}^N w_{k-1|k-1}^{(i)} p(x_k^{(j)}|y_{k-1}^{(i)}), \quad (15)$$

$$p(x_k|y_{k-1}) \approx \sum_{i=1}^N w_{k|k}^{(i)} \delta(x_k - x_k^{(i)}), \quad (16)$$

where $w_{k-1|k-1}^{(i)}$ and $w_{k|k}^{(i)}$ are the *a priori* and posterior weight of the particle (i), respectively. Thus, replacing (15)-(16) in (14), it is possible to write:

$$H(p(x_k|y_k)) = \log p(y_k|y_{k-1}) + \sum_{j=1}^N w_{k|k}^{(j)} [\log p(y_k|x_k^{(j)}) + \log p(x_k|y_{k-1})] \quad (17)$$

Also, the term $p(y_k|y_{k-1})$ can be computed through its sampled version:

$$p(y_k|y_{k-1}) = \sum_{i=1}^N w_{k|k-1}^{(i)} p(y_k|x_k^{(i)}), \quad (18)$$

where $w_{k|k-1}^{(i)}$ are the particle weights. As a final result, the differential entropy of the particle-filtering estimate of the posterior state probability density function can be computed as in (19) (Orguner, 2009):

$$\begin{aligned} H(p(x_k|y_k)) &= \log \left(\sum_{i=1}^N w_{k|k-1}^{(i)} p(y_k|x_k^{(i)}) \right) - \\ &\sum_{j=1}^N w_{k|k}^{(j)} [\log p(y_k|x_k^{(j)}) + \log p(x_k|y_{k-1})] \quad (19) \end{aligned}$$

Equation (19) will be of use when evaluating the uncertainty associated to online estimates in dynamic processes.

Entropy-related applications for particle filtering algorithms generally aim at evaluating how many independent and identically distributed (i.i.d.) samples does the filtering algorithm require to represent regions of the state-space that accumulate the majority of the probability mass, for a given state probability density function estimate $p(x)$ (for a complete survey of use of entropy in particle filtering schemes, please refer to (Orchard *et al.*, 2012)).

In (Orchard *et al.*, 2012), the authors propose to use entropy as an uncertainty metric over the online estimate of the classical PF in a dynamic process. In this sense, this work extends this idea, calculating entropy over RSPF and PF output for uncertainty management and high-risk event detection in volatility estimation.

3. Implementation of a RSPF Framework for Volatility Estimation and Entropy-based Detection

In this section, we describe the procedure followed to generate the data used as ground truth to compare the two filtering techniques: Classic particle filter and risk-sensitive particle filter. Also, the comparison metrics are introduced, allowing the comparison between both filtering approaches in terms of accuracy and precision. As a consideration for the algorithm implementation, all schemes use 100 particles.

3.1 Volatility estimation using particle filters

In the current section, the implementation details of the particle filtering routines used in this work are presented. The notation of the

uGARCH model previously shown is used, presented by (Tobar and Orchard, 2012).

3.1.1 Classic particle filter

According to (Maskell and Gordon, 2001; Arulampalam *et al.*, 2002), it is possible to demonstrate that the optimal $q(\sigma_k^{2(i)}|\sigma_{k-1}^{2(i)})$ function in the sense of minimizing degeneration of the particle set is:

$$q(\sigma_k^{2(i)}|\sigma_{k-1}^{2(i)}) = p(\sigma_k^{2(i)}|\sigma_{k-1}^{2(i)}). \quad (20)$$

In the classical version of the PF, the particles are generated according to:

$$\sigma_k^2 \sim p(\sigma_k^2|\sigma_{k-1}^{2(i)}), \quad (21)$$

and the respective weights are updated through:

$$w_k^i = w_{k-1}^i p(r_k|\sigma_{k-1}^{2(i)}) \quad (22)$$

In the case of the classical version of the PF, the state-transition distribution and the likelihood are assumed to be known.

3.1.2 Risk-sensitive particle filter

Sampling in a risk-sensitive particle filtering approach is done through an arbitrary risk functional. Specifically, during this work the authors have considered the Inverse Gamma distribution to be an appropriate risk functional, given its relation of conjugate distribution of the Gaussian distribution. The Inverse Gamma distribution is:

$$Inv\Gamma(x; \alpha_\Gamma, \beta_\Gamma) = \frac{\beta_\Gamma^{\alpha_\Gamma}}{\Gamma(\alpha_\Gamma)} x^{-\alpha_\Gamma-1} e^{-\frac{\beta_\Gamma}{x}}. \quad (23)$$

In (23), it is possible to note that the density is parameterized by the variables α_Γ and β_Γ . During this work, the generation of particles has been done considering these two parameters in the following way:

$$\sigma_k^{2(i)} \sim Inv\Gamma(\sigma_k^2, \alpha_\Gamma, \beta_\Gamma = \sigma_{k-1}^{2(i)}), \quad (24)$$

where α_Γ is a parameter that needs to be determined and β_Γ is the value of the state in the previous time step. Thus, the weight update equation is described by:

$$w_k^i = w_{k-1}^i \frac{p(r_k|\sigma_k^i)p(\sigma_k^{2(i)}|\sigma_{k-1}^{2(i)})}{Inv\Gamma(\sigma_k^{2(i)}|\alpha_\Gamma, \sigma_{k-1}^{2(i)})}. \quad (25)$$

The numerator of (25) is calculated using Eq. (6) and (7), respectively.

3.2 Data generation

In the comparison of the particle filtering schemes, it is mandatory to establish a data set where returns and volatility are known in every instant for a given time window. The artificially created data includes a parameter change in the generating model that simulates a regime shift in the market. The authors have considered a time window of 500 steps, using the following dynamical system:

$$\sigma_{k|k-1}^2 = \omega + \alpha u_{k-1}^2 + \beta \sigma_{k-1|k-2}^2$$

$$r_k = \mu_{k|k-1} + u_k,$$

where the parameters α and β change over time according to the values present in Table 1. It is important to note that during the comparison of both algorithms, the comparison metrics are applied to the filtering output for $k = 151:500$. Values in the range $k = 1:100$ are used for parameter estimation, and data points between $k = 101:150$ are considered for the training of both PFs.

Table 1. Parameters used in the GARCH model for data generation

Time	α	β
$1 \leq k \leq 250$	0,2	0,6
$251 \leq k \leq 500$	0,1	0,85

3.3 Performance measures for particle-filter-based estimates

3.3.1 Accuracy indicator

To measure the accuracy of the estimates produced by both PF approaches, the following index is used, for every instant k of time:

$$i^{EX}(k) = \frac{|\hat{\sigma}_k^2 - \sigma_k^2|}{\sigma_k^2} \cdot 100. \quad (26)$$

This approach permits to measure the percentage difference between the estimated value and the real value (artificially generated data used as ground truth). Furthermore, it is possible to calculate the mean value over the sliding window:

$$I^{EX} = \frac{1}{T} \sum_{k=151}^T i^{EX}(k) = \sum_{k=151}^T \frac{|\hat{\sigma}_k^2 - \sigma_k^2|}{\sigma_k^2} \cdot 100$$

From the previous Equation, it is possible to observe that values close to zero indicate that the estimated value approaches the true value. Due to the inherent randomness of the filtering

process, the time window is filtered 10 times, and the average of the index I^{EX} is calculated:

$$\bar{I}^{EX} = \frac{1}{10} \sum_{n=1}^{10} I^{EX}(n). \quad (27)$$

3.3.2 Precision indicator

It is also necessary to compare the performance of the algorithms from a precision perspective. Let the instantaneous precision of a filtering structure be

$$i_X^{PR}(k) = \frac{\widehat{CI}_X(k)}{\overline{CI}_X(k)}, \quad (28)$$

where $\widehat{CI}_X(k)$ is the length of the $X\%$ confidence interval in 1-step predictions. The confidence interval has been calculated through a continuous version of the 1-step prediction probability density functions from an existent set of particles, using Epanechnikov kernels (Tobar and Orchard, 2012). The confidence interval is calculated integrating from 0 (which is the no-risk point, since null volatility implies complete certainty). Also, $\overline{CI}_X(k)$ is the average in a sliding window that considers the last 50 data points. For this reason, this index is calculated starting at $k = 151$.

Analogously to the accuracy index, it is possible to define the average of the precision index over the sliding window:

$$I_X^{PR} = \frac{1}{T} \sum_{k=51}^T i_X^{PR}(k) = \frac{1}{T} \sum_{k=51}^T \frac{\widehat{CI}_X(k)}{\overline{CI}_X(k)}.$$

Values close to 1 indicate that the estimation process reports levels of uncertainty similar to the process being studied (Tobar and Orchard,

2012). It is important to indicate that the reason for using the average over a sliding window is that the true dispersion of the data when using real values is unknown. Additionally, it is possible to calculate the average of the precision index, taken over 10 filtering realizations:

$$\bar{I}_X^{PR} = \frac{1}{10} \sum_{n=1}^{10} I_X^{PR}(n). \quad (29)$$

The indexes are used to compare the filtering output of both PF schemes.

3.4 Results in generated time series

3.4.1 Parameter estimation

For the data presented in Figure 1, a GARCH (1,1) model has been assumed and parameters have been adjusted using maximum likelihood. These estimated parameters are used as initial conditions for the filtering algorithms, and have been calculated using the first 150 steps of the simulation (see Table 2). In Figure 1 and also in the following figures, the first 100 steps have been discarded to ensure that the impact of the initial conditions has been dissipated. Hence, only steps $k = 101:500$ have been displayed (and shown as $k = 1:400$ for simplicity).

Table 2. Parameters estimates computed from artificially generated data, using steps 1:150

Parameter	Value
$\hat{\mu}$	$-2,5930 \times 10^{-4}$
$\hat{\omega}$	$9,8107 \times 10^{-6}$
$\hat{\alpha}$	0,3735
$\hat{\beta}$	0,5687

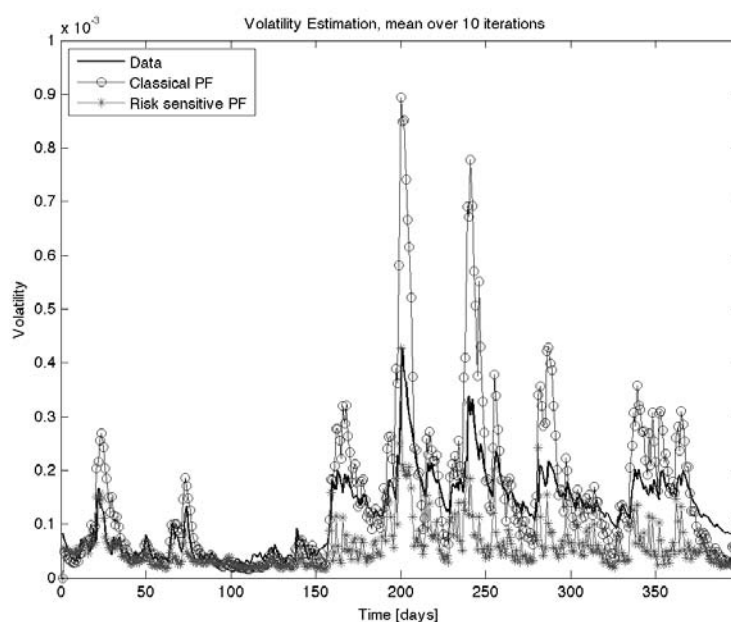


Figure 1. Average filtering output over 10 realizations with $\alpha_r = 0,7$.

Due to implementation reasons, it has been considered that the parameters α and β are part of the state vector and therefore, must be estimated.

3.4.2 Comparative results: Accuracy

Regarding accuracy indexes, results containing the accuracy comparisons of \bar{I}^{EX} between the PF and RSPF are presented in Table 3. The presented results are obtained averaging over 10 filtering processes, for the values of α_r between 0.3 and 1.6. On Table 3, it is clear that the PF works better as a state estimator than the RSPF. In particular, values around $\alpha_r = 0.7$ are the best estimator.

Table 3. Accuracy index averages over 10 filtering routines for different values of α_r

α_r	$\bar{I}^{EX} PF$	$\bar{I}^{EX} RSPF$
0,3	39,2821	66,8970
0,4	39,2821	57,7779
0,5	39,2821	57,1155
0,6	39,2821	56,5506
0,7	39,2821	56,4327
0,8	39,2821	58,9474
0,9	39,2821	58,7331
1,0	39,2821	59,2656
1,1	39,2821	68,2475
1,2	39,2821	71,6817
1,3	39,2821	66,1019
1,4	39,2821	102,3524
1,5	39,2821	147,4566
1,6	39,2821	352,9343
1,7	39,2821	605,1278

In Figure 1, the filtering results are presented for both PF and RSPF with $\alpha_r = 0.7$. Figure 1 also shows the outputs produced by both filtering schemes. The PF in general produces better results and the RSPF, but in many cases it overestimates the true value, while the RSPF underestimates the true value almost everywhere. Nevertheless, it is extremely important to note that the RSPF has a great capacity to filter states where there is a sudden and important variation in the state, producing better estimates in fast changes of volatility.

3.4.3. Comparative results: Precision

Data presented in Table 4 offer the average of the calculated precision index again for both particle filtering schemes, for every α_r . From the Table 4, it is possible to observe that the results are very balance between both particle filtering schemes. The RSPF has the best results again for values around $\alpha_r = 0.7$, even

better than the classical PF. However, it is important to note that due to the underestimation of the RSPF, the real value is not always inside the 90% confidence interval. This is an interesting point, since it reaffirms that the RSPF tracks sudden volatility changes better than the classical PF.

Although in Table 4 both filtering schemes offer values similar to 1 for the precision index, this indicator is not capable of capturing all the dynamic of the process. In point of fact, the length of the confidence intervals produces by the RSPF is shorter than the ones produced by the PF. Nevertheless, it is mandatory to consider first the accuracy of the estimators rather than precision.

Table 4. Accuracy index averages over 10 filtering routines for different values of α_r

α_r	$\bar{I}^{PR}PF$	$\bar{I}^{PR}RSPF$
0,3	1,1107	1,0729
0,4	1,1107	1,0699
0,5	1,1107	1,0743
0,6	1,1107	1,0752
0,7	1,1107	1,0513
0,8	1,1107	1,0351
0,9	1,1107	1,0459
1,0	1,1107	1,0239
1,1	1,1107	1,0567
1,2	1,1107	1,0518
1,3	1,1107	1,0380
1,4	1,1107	1,0848
1,5	1,1107	1,0740
1,6	1,1107	1,1243
1,7	1,1107	1,1465

3.5. Entropy-based detection of sudden variations in financial volatility

In this work, entropy is used as a measure of randomness of the *a posteriori* distribution generated by both PF and RSPF-based approaches. Figure 3 shows a time series for entropy of the conditional distributions, calculated according to (19). In is interesting to note that in the case of classical PF-based estimates, entropy may indeed show relevant information within a detection scheme, since high volatilities are associated with high entropies and low volatilities with low entropies. Particularly, between the 70th and 120th data points there the calculated entropy value decreases significantly, while the inverse phenomenon may be observed between the 150th and 170th data points. Additionally, as Figure 3 also shows, the information that is

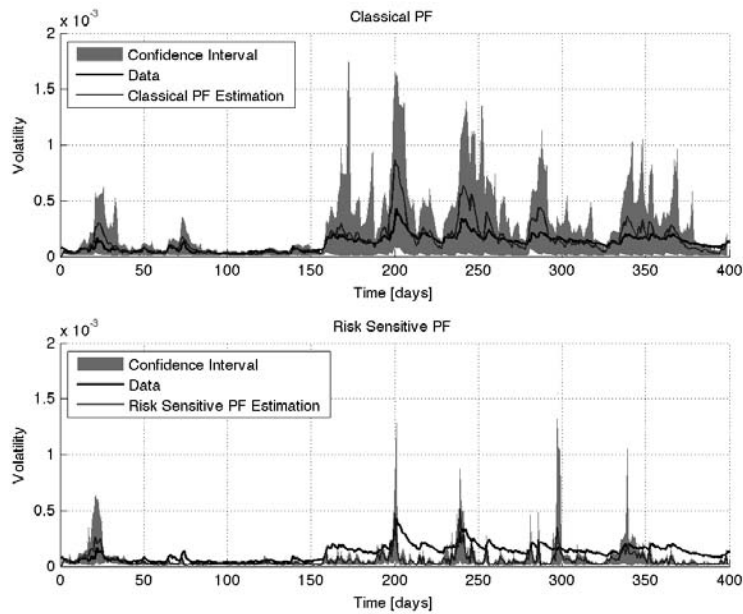


Figure 2. Filtering routine #5, with $\alpha_r = 0.7$ and 90% confidence interval.

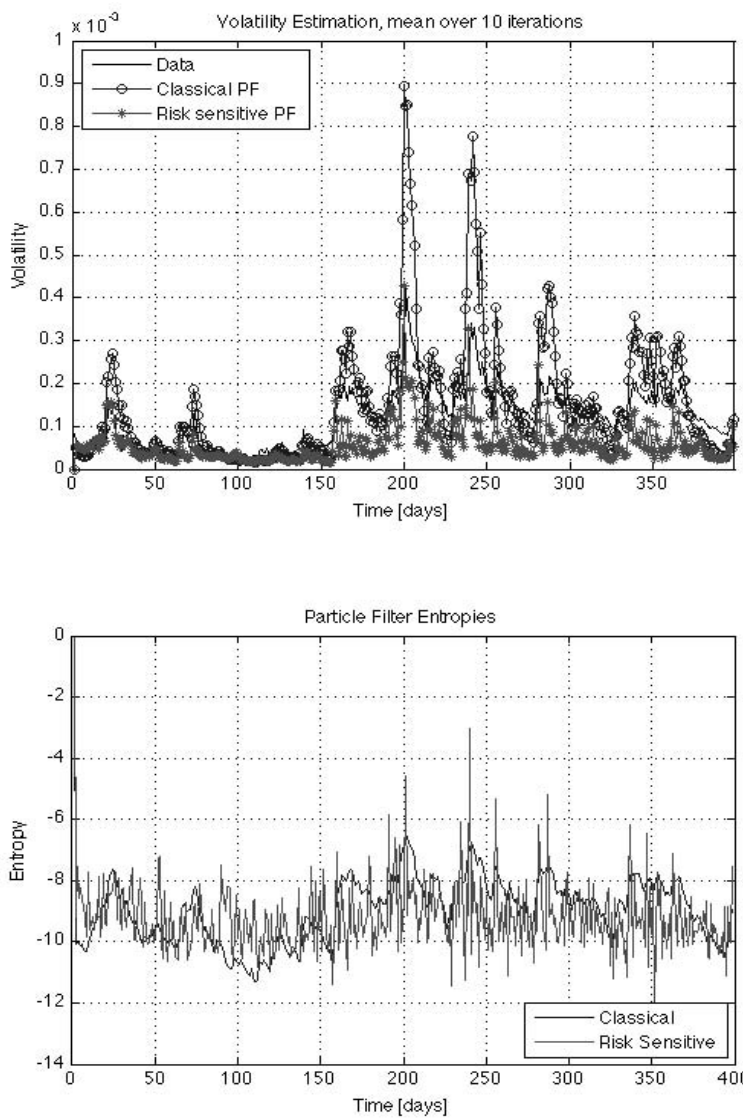


Figure 3. Entropy of the PF-based conditional probability density estimates.

provided by the entropy calculation of the RSPF-based posterior state distribution results to be highly suitable for the detection of volatility peaks (as it occurs for the 160th, 201st, 240th, 256th, and 287th data points), which typically antecedes high-volatility clusters. In general, it is possible to observe that certainty over classical PF-based estimates is well characterized by entropy calculation (see Figures 2 and 3), while entropy calculation over the RSPF scheme may conduce to an automatic peak-detection scheme.

Finally, it is important to note that entropy does not seem to show important information with respect to the regime shift introduced at the 150th data point (see Figure 3). Indeed, there is no apparent difference in entropy behavior, other than a mean change.

4. Conclusion

This article presents and analyzes the implementation of a risk-sensitive particle-filtering algorithm for volatility estimation of continuously compounded returns of financial assets. This algorithm is based on uGARCH models for state-space representation of the system, and an Inverse Gamma distribution has been used as risk functional (and importance density distribution) within the Bayesian estimation routine to ensure the allocation of particles in regions of the state-space that are associated to sudden changes in the volatility of the system. The proposed methodology is, in fact, a first step towards the inclusion of fault detection approaches in a framework for automatic detection of high-volatility clusters.

RSPF results have been compared to the volatility estimation obtained through a classical PF approach. The comparison shows that the application of classic PF-basic estimation routines has a tendency to overestimate the actual state values, while the proposed RSPF scheme tends to underestimate it. However, there are some interesting facts that require further analysis. The RSPF is capable of producing better estimations than the classical PF over rapid-changing volatility values (initial steps of high-volatility clusters), a fact that results critical for the implementation of decision-making schemes. More importantly, it is assumed that combining these results, with entropy calculation, it is possible to produce a detector for high

variations of volatility; either from low-volatility clusters to high-volatility clusters or vice-versa.

To address the afore-mentioned concept, a simple approach inspired in fault detection techniques was implemented and tested. However, from this perspective, entropy does not seem to be a correct tool for regime shift detection. Therefore, as future work in this area, it is mandatory to consider other tools to obtain an adequate detection of high volatility and produce an automatic detection scheme based on entropy. It is necessary, as well, to study the proposed techniques in other data sets, including artificially generated data and real returns from the stock market.

As a final remark, it is important to mention that future work must also include the generation of efficient code that may be used to test and the proposed algorithms in real-time, extracting stock data directly from authorized sites available at the World Wide Web.

Acknowledgements

The authors want to thank Conicyt for its financial support through the project FONDECYT #1110070.

REFERENCES

1. AJGL J., M. ŠIMANDL, **Particle Based Probability Density Fusion with Differential Shannon Entropy Criterion** Proceeding of the 14th International Conference on Information Fusion ISIF, 2011, pp. 803-810, Chicago, Illinois, USA.
2. ARULAMPALAM, M. S., S. MASKELL, N. GORDON, T. CLAPP, **A Tutorial on Particle Filters for On-line Nonlinear / Non-Gaussian Bayesian Tracking**, IEEE Transactions on Signal Processing, vol. 50, no. 2, 2002, pp. 174-188.
3. BOERS, Y., J. N. DRIESSEN, A. BAGCHI, P. K. MANDAL, **Particle Filter Based Entropy**, Proceedings of FUSION 2010, Edinburgh, UK.
4. BOLLERSLEV, T., **Generalized Autoregressive Conditional Heteroskedasticity**, Journal of Econometrics, vol. 31, no. 3, 1986, pp. 307-327.

5. COVER, T., J. THOMAS, **Elements of Information Theory**, Wiley Interscience, New York, 1991.
6. DOUCET, A., **On Sequential Monte Carlo Methods for Bayesian Filtering**, Technical Report, Engineering Department, Univ. Cambridge, 1998, UK.
7. DOUCET, A., N. DE FREITAS, N. GORDON, **An Introduction to Sequential Monte Carlo Methods**, in *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. de Freitas, and N. Gordon, Eds. NY: Springer-Verlag.
8. HARTLEY, R. V. L., **Transmission of Information**, International Congress of Telegraphy and Telephony, Lake Como, Italy, 1927.
9. HARVEY, A., S. M. KOOPMAN, **Unobserved Components Models in Economics and Finance**, *Control Systems*, IEEE, vol. 29, no. 6, 2009, pp. 71-81.
10. HAYKIN, S., **Neural Networks and Learning Machines**, Prentice Hall, vol. 10, 3rd edition, 2009.
11. LANZ, O., **An Information Theoretic Rule for Sample Size Adaptation in Particle Filtering**, *Proceedings of the 14th International Conference on Image Analysis and Processing (ICIAP)*, 2007, pp. 317-322.
12. MANDELBROT, B., **The Variation of Certain Speculative Prices**, *The Journal of Business*, vol. 36, no. 4, 1963, pp. 394-419.
13. MASKELL, S., GORDON, N., **A Tutorial on Particle Filters for On-line Nonlinear/Non-Gaussian Bayesian Tracking**, Technical Report, QinetiQ Ltd., and Cambridge University, 2001, UK.
14. ORCHARD, M., B. OLIVARES, M. CERDA, J. F. SILVA, **Anomaly Detection based on Information-Theoretic Measures and Particle Filtering Algorithms**, Annual Conference of the PHM Society 2012, Minneapolis, USA.
15. ORGUNER, U., **Entropy Calculation in Particle Filters**, *IEEE Signal Processing and Communications Applications Conference*, 2009.
16. SKOGLAR, P., U. ORGUNER, F. GUSTAFSSON, **On Information Measures based on Particle Mixture for Optimal Bearings-only Tracking**, *Proceedings of IEEE Aerospace Conference 2009*, Big Sky, Montana, USA.
17. THRUN, S., J. LANGFORD, V. VERMA, **Risk Sensitive Particle Filters**, In *Advances in Neural Information Processing Systems*, vol. 14, 2002.
18. TOBAR, F., M. ORCHARD, **Study of Financial Systems Volatility Using Suboptimal Estimation Algorithms**, *Studies in Informatics and Control*, vol. 21, Issue 1, March 2012, pp. 59-66.
19. TSAY, R., **Analysis of Financial Time Series**, John Wiley & Sons, 3rd edition, 2010.