Hidden State and Parameter Estimation Using Particle Filtering

Andrew Soane

African Institute of Financial Markets and Risk Management, University of Cape Town

September 27, 2018

AIFMRM
AFRICAN INSTITUTE OF FINANCIAL MARKETS AND RISK MANAGEMENT

Overview

- Introduction to filtering
  - Filtering problem
  - Particle filtering
- Implementation
  - Overview
  - Filtering stock prices
  - Filtering volatility
  - Filtering parameters

Introduction

Filtering Problem

- Separate unobservable/hidden states from a source of noise.
- We receive a signal corrupted by noise and wish to determine is there is indeed a signal or if it is just noise. For Example:
  - GPS signal processing.
  - Aircraft tracking using radar signals.
  - Separate model-implied noise from a series of stock prices.
- Learn model parameters by treating them as unobservable states.
- Detect or test validity of certain models on a series of observed data.

Particle Filtering

- Particle filtering is a sequential Monte Carlo method used to filter non-linear, non-Gaussian stochastic models.
- The algorithm has four primary stages:
  - Propagation
  - Measurement
  - Prediction
  - Update
- Relies on certain, known, prior knowledge of the observed and unobservable processes.
  - We aim to use posterior knowledge to estimate the unobserved processes.
- PF's end result is a sequence of distributions through time of the unobservable states.
  - Unlike deterministic filters which just give point estimates through time.

Implementation

The Model

Suppose we have a discrete stock price model as follows:

$$X_{t+1} = f(X_t, Z_t, \Theta)$$
$$Z_{t+1} = g(Z_t, \Theta)$$

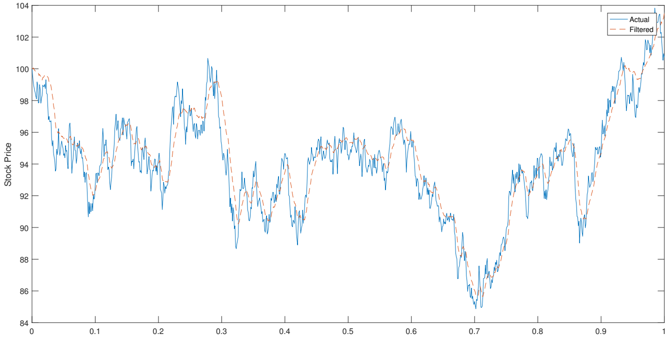
Where f and g are possibly non-linear functions and Θ is the set of parameters defining the model.
Furthermore, suppose we can calculate an option price, C_t under this model

$$C_t = \mathcal{C}(X_t, Z_t, \Theta)$$

Example

An example: consider the Black-Scholes model.
We are interested in filtering the stock price process given a series of at-the-money call options .

Figure: Filtered Estimate of Stock Price Path



Volatility filtering

Initially, our focus is on filtering the volatility Z_t from a series of stock prices, assuming we know Θ.
We then ask if its possible to filter both Z_t and Θ just knowing X_t.

Data

From here on out, we consider testing this filtering method on a simulated data set.
That is:

- Simulate paths of X and Z using an assumed form for their dynamics
- Attempt to recover Z knowing only X and the form of the model
- We hope that the estimated path of Z will be close to the actual simulated path

Prior Information

- Prior information required for Latent State Estimation:
  - Initial distribution: $p(Z_0)$.
  - Transition distribution: $p(Z_{t+1}|Z_t, \Theta)$.
  - Measurement equation: $p(X_{t+1}|X_t, Z_t, \Theta)$.
- Prior information required for parameter estimation:
  - Prior distribution of parameter set: $p(\Theta)$.
  - A way of updating our parameter estimates based on the observable information.
    - MLEs
    - Bayesian posterior estimates
    - Simulating parameters based on the previous time step's distribution

Algorithm Overview

Consider the previous example of filtering the log stock price from a series of option prices. Using N discrete time steps and M particles:

1. Initialise: $X_0 = \log(100)$ and $C_0 = \mathcal{C}(X_0)$
2. For $i = 1 : N$, $j = 1 : M$
   Propagation: $X^j_{i+1} \sim p(X_{i+1}|X^j_i)$
   Measurement: $C^j_{i+1} = \mathcal{C}(X^j_i)$
   Prediction: $w^j_{i+1} \propto \phi(C_{i+1}; C^j_{i+1}, 1)$[1]
   Update: Resample $\{X^j_{i+1}\}^M_{j=1}$ according to $\{w^j_{i+1}\}^M_{j=1}$

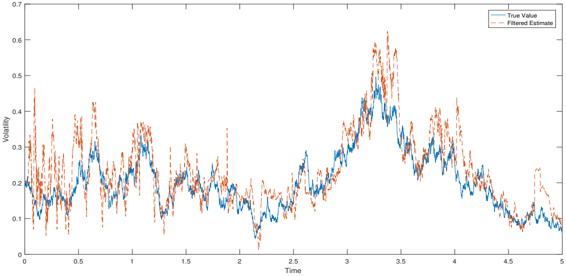
What we are left with at each time $t+1$ is an empirical estimate of the distribution of X_{t+1}.

[1] $\phi(x; \mu, \sigma^2)$ is the Gaussian distribution with mean μ, variance σ^2 evaluated at x.

Filtering Volatility and Jumps

Assuming the stock price is driven by a random volatility and a random jump process. Our aim is to predict the volatility and jump process.

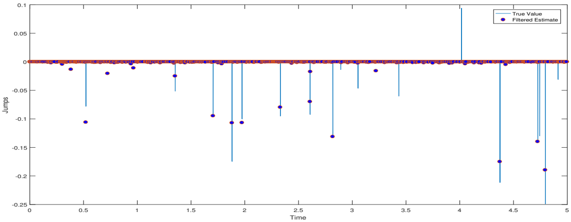
Figure: Volatility Estimates



Filtered Estimates of the Jump Process

We assume a jumps occur at exponential times with Gaussian sizes.

Figure: Jump Process Estimate



The model used was

$$X_{t+1} = \mu X_t + \sqrt{Z_t}\epsilon_1 + J_t$$

$$Z_{t+1} = \kappa + \theta Z_t + \sigma\sqrt{Z_t}\epsilon_2$$

Bayesian Statistics

Bayes' Theorem states:

$$p(\Theta|X_t) = \frac{L(\Theta; X_t)p(\Theta)}{p(X)}$$

where $p(\Theta)$ is the prior distribution of Θ,
This gives us a way of "sampling" a parameter set and then measuring their influence on X_t.

Parameter Estimation

We make use of the Likelihood Factorisation Theorem:

$$p(\Theta, s_t, Z_t|X_t) = p(\Theta|s_t)p(s_t, Z_t|X_t),$$

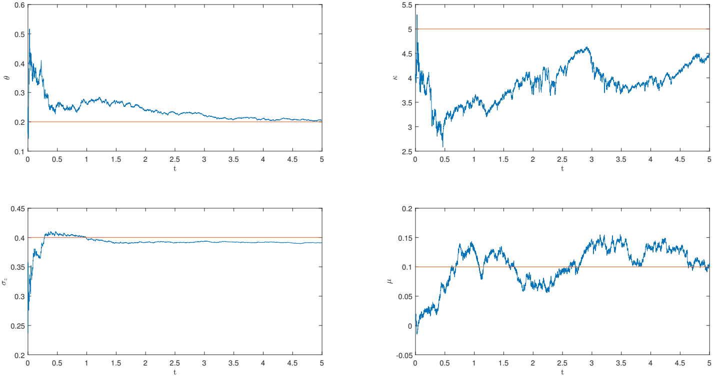
where s_t is a set of sufficient statistics for Θ
This decomposes the filtering problem into two steps:

- Hidden state estimation.
- Parameter estimation.

We've seen how hidden state estimation can be done in a stochastic volatility/jump model.
It is left to show how parameters can be sequentially learned using particle filtering.

Bayesian Results

To illustrate the results, we assume Z_t and q_t are known.
We wish to use PF to estimate the parameters $\theta, \kappa, \sigma, \mu$.



Conclusion and Further Research

- The aim of the project was to formulate a simple, easy-to-implement algorithm for filtering latent states and estimating parameters in a stochastic volatility/jump model.
- In practice, we require simultaneous estimation of latent states and parameters, and we have shown how the sub-problems differ, and how one can be solved if the other is known.
- Further research needs to be done to merge the two solutions to allow for simultaneous estimation. Possibly by:
  - Incorporating more data into the observed process, for example, by augmenting option price data with the stock price series.
  - Marginalising latent state and parameters from the Bayesian posterior distributions.
  - Using some form of smoothing on parameter estimates to disallow explosion of variance in the parameter estimations.
  - More informative prior distributions.

Agenda

Speaker	Topic	Timing
	Tea Break	15.00 – 15.30
Zaid Parak, Kerissa Varma and Julian Ramiah	Ask the CISO: Your questions asked and answered	15.30 -16.30
Judy Faure	Close	16.30 – 16.45