Introduction to Particle Filters for Data Assimilation

Mike Dowd
Dept of Mathematics & Statistics (and Dept of Oceanography)
Dalhousie University, Halifax, Canada
Problem Statement

• Noisy time series observations on the system state

\[ y_{1:T} = (y_1, \ldots, y_T) \]

• a discrete time stochastic dynamic model describing the time evolution of the system state

\[ x_t = \theta x_{t-1} + n_t \]

e.g. Bivariate AR(1) process

• We want to estimate the system state (and sometimes parameters)
Noisy Observations of System State

Observations
Estimation of System State by Particle Filters

Particle Filter Estimate of the State
Target: Evolution of the Probability Distribution Of the State through Time
**A General Framework for DA**

- Hierarchical Bayesian Model

\[ p(x_{1:T}, \theta | y_{1:T}) \propto p(y_{1:T} | x_{1:T}, \theta) \cdot p(x_{1:T} | \theta) \cdot p(\theta) \]

- **Special Cases in DA**
  - Parameter estimation (no model error)
  - Filtering solution for state *
  - Smoothing solution for state
  - Joint state & parameter estimation
  - Inference on dynamic model structure
**Data Assimilation: A Statistical Framework**

- **State Space Model:**
  \[ x_t = d(x_{t-1}, \theta, e_t) \]
  \[ y_t = h(x_t, \phi, v_t) \]

- The general goal is to make inferences about the state \( x_t \) and sometimes the static parameters* \( \theta \) and \( \phi **. 

- We use explicit priors on process noise & observation errors (i.e. parametric distributions for \( e_t \) and \( v_t \)). Assume \( y_t \) conditionally independent given \( x_t \), and that \( e_t \) and \( v_t \) are independent.

- For DA, we want to do space-time problems: space is incorporated in \( x_t \)

\[ \Rightarrow \text{Solutions involve sequential Monte Carlo based for nonlinear and nonGaussian state space models} \]

* Dynamic parameters are part of the state  
** drop \( \phi \) hereafter, and defer \( \theta \)
Problem Classes

We want to estimate the system state at time $t$, or $x_t$, given the observations up to time $s$, or $y_{1:s} = (y_1, y_2, \ldots, y_s)$.

A) Filtering (Nowcast)

\[ s = t \rightarrow \text{Filtering} \]

B) Forecasting

\[ s > t \rightarrow \text{Prediction} \]

C) Smoothing (Hindcast)

\[ s < t \rightarrow \text{Smoothing} \]

- data available
- no data available
- estimation time
Sequential Estimation

Single stage transition of system from time $t-1$ to time $t$

- We’ll drop parameters $\theta$ … for now
Probabilistic Solution for Filtering

A single stage transition of the system for time $t-1$ to $t$ involves:

**Prediction:**

$$p(x_t | y_{1:t-1}) = \int p(x_t | x_{t-1}) \cdot p(x_{t-1} | y_{1:t-1}) \, dx_{t-1}$$

**Measurement update:**

$$p(x_t | y_{1:t}) = \frac{p(y_t | x_t) \cdot p(x_t | y_{1:t-1})}{p(y_{1:t})}$$

→ General Filtering Solution:

$$p(x_t | y_{1:t}) \propto p(y_t | x_t) \int p(x_t | x_{t-1}) \cdot p(x_{t-1} | y_{1:t-1}) \, dx_{t-1}$$

do for $t=1, \ldots, T$, given $p(x_0)$ and $y_{1:t} = (y_1, y_2, \ldots, y_T)$
Sampling Based Estimation

Let \( \{x_{t|t}^{(i)}, w_t^{(i)}\}, i = 1, \ldots, n \) be a random sample* from \( p(x_t | y_{1:t}) \)

drawn from the target distribution (the filter density)

where: \( x_{t|t}^{(i)} \) are the support points
\( w_t^{(i)} \) are the normalized weights

The filter density can then be approximated as

\[
p(x_t | y_{1:t}) \approx \sum_{i=1}^{n} w_t^{(i)} \delta(x_{t|t} - x_{t|t}^{(i)})
\]

*referred to an ensemble, made up of a collection of particles (which are sample members)
Sampling Based Estimation

(a) unweighted ensemble, n=25

(b) weighted ensemble, n=25

(c) unweighted ensemble, n=250

(d) weighted ensemble, n=250
Sequential Importance Sampling (SIS)

At time $t-1$ we have:

- $\{x_{t-1|t-1}^{(i)}, w_{t-1}^{(i)}\}$, draws from $p(x_{t-1} \mid y_{1:t-1})$
- $y_t$, an observation

If a candidate $x_{t|t}^{(i)}$ is drawn from a proposal $q(x_{t|t})$

then $w_t^{(i)} \propto \frac{p(x_{t|t}^{(i)} \mid y_t)}{q(x_{t|t}^{(i)} \mid y_t)}$

or $w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(y_t \mid x_{t|t}^{(i)}) \cdot p(x_{t|t}^{(i)} \mid x_{t-1|t-1}^{(i)})}{q(x_{t|t}^{(i)})}$

A draw from $p(x_t \mid y_{1:t})$ is then available as $\{x_{t|t}^{(i)}, w_t^{(i)}\}$

see Arulampalam et al. 2002
**SIS Remarks**

- SIS solves the filtering problems for the nonlinear, non-Gaussian state space model via an algorithm for sequential “online” estimation of the system state relying on a proposal and weight update.

- A common simplification is to let the proposal be the the prior (the predictive density) $q(x_{t|t}) = p(x_t | x_{t-1})$ leading to:
  $$w_t^{(i)} \propto w_{t-1}^{(i)} p(y_t | x_{t-1}^{(i)})$$

- The main practical problem with SIS is ‘weight collapse” wherein after a few time steps, all the weights become concentrated on a few particles. How to fix?
**Sequential Importance Resampling (SIR)**

- The major breakthrough in sequential Monte Carlo methods (Gordon et al. 1993) was to incorporate a resampling step to obviate weight collapse.

- Specifically, for each time $t$, $\{x^{(i)}_{t|t-1}, w^{(i)}_{t}\}$ is a sample from $p(x_t | y_{t-1})$.

- We can resample (with replacement) the $x^{(i)}_{t|t}$ with a probability proportional to $w^{(i)}_{t}$ (i.e. a weighted bootstrap).

- This yields a modified sample $\{x^{(i)}_{t|t}\}$, wherein particles were both killed and replicated, and the weights are now equal, i.e. $w^{(i)}_{t} = 1 \forall i$.

**Remaining Issue:** sample impoverishment - some particles replicated often in the unweighted sample $\{x^{(i)}_{t|t}\}$. Less severe than weight collapse (but of course related).
Basic Particle Filter Algorithm

Given: (1) dynamical model, (2) observations, (3) initial conditions

for \( t = 1 \) to \( T \)

(a) Prediction: \( \{ x_{t-1|t-1}^{(i)} \} \rightarrow \{ x_{t|t-1}^{(i)} \} \) as
\[
x_{t|t-1}^{(i)} = d(x_{t-1|t-1}^{(i)}, e_t^{(i)}) \quad \text{for } i = 1, \ldots, n
\]

(b) Measurement: \( \{ x_{t|t-1}^{(i)} \} \rightarrow \{ x_{t|t}^{(i)} \} \) as
- \( w_t^{(i)} \propto p(y_t | x_{t|t-1}^{(i)}) \) for \( i = 1, \ldots, n \)
- resample with replacement \( \{ x_{t|t-1}^{(i)} \} \) using weights \( w_t^{(i)} \)

\( \rightarrow \) yields \( \{ x_{t|t}^{(i)} \} \)

end (for t)
Particle Filter Schematic

Single Stage Transition for SIR

Forecast  Assign weights/Resample

\[ \{ X_{t-1|t-1}^{(i)} \} \sim p(x_{t-1} \mid Y_{t-1}) \]

\[ \{ X_{t|t-1}^{(i)} \} \sim p(x_{t} \mid Y_{t-1}) \]

\[ \{ X_{t|t}^{(i)} \} \sim p(x_{t} \mid Y_{t}) \]

- particle
- observation
Interpretation: Particle History
for (t in 2:T)
{
    xf = t(apply(xa, 1, dynamicmodel))
    yobs = matrix(y[t,], 1, ncol(y))
    w = apply(matrix(xf[, 1], np, 1), 1, dmvnorm, x = yobs, sigma = R)
    m = sample(1:np, size = np, prob = w, replace = T);
    xa = xf[m,]
}

- **Prediction step** from t-1 to t
- **Extract measurement at time t**
- **Assign weights**
- **Weighted resampling**
Element 1. Prediction Step

\[
\begin{align*}
\{x_t^{(i)}|t-1\} & \xrightarrow{} \{x_t^{(i)}|t\} \\
\text{draw from } p(x_{t-1}|y_{1:t-1}) & \xrightarrow{} \text{draw from } p(x_t|y_{1:t-1})
\end{align*}
\]

filter density at time t-1
predictive density at time t

Role: acts as proposal distribution for particle filter

\[
x_t^{(i)} = d(x_{t-1|t-1}^{(i)}, e_t^{(i)}) \text{ for } i = 1,\ldots,n
\]

Remarks:
- computationally expensive to generate large samples for big GFD models
- hard to effectively populate a large dimension state space with particles
**Element 2. Measurement Update**

\[
\begin{align*}
\{x_{t|t-1}^{(i)}\} & \quad \xrightarrow{\text{draw from } p(x_t \mid y_{1:t-1})} \quad \{x_{t|t}^{(i)}\} \\
\text{predictive density} & \quad \text{at time } t-1 \\
y_t & \quad \text{filter density} \quad \text{at time } t
\end{align*}
\]

\[
\text{uses } p(x_{t|t} \mid y_{1:t}) \propto p(y_t \mid x_{t|t-1}) \cdot p(x_{t|t-1})
\]

(Bayes’ Formula)

**Approaches:**

- **PF/SIR:** weighted resample of \( \{x_{t|t-1}^{(i)}, w_t^{(i)}\} \rightarrow \{x_{t|t}^{(i)}\} \\
- **Metropolis Hastings MCMC**
- **Ensemble Kalman filter** (approximate)
- **plus many others** (auxiliary, unscented, ….) that modify proposal distributions
**Sequential Metropolis-Hastings**

Construct (independence) chain to yield sample from target filter distribution.

for \( k = 1 \) to \( K \)

1. Generate candidate from predictive density:
   
   draw \( x_{t|t-1}^* \) from predictive density \( p(x_t | y_{1:t-1}) \)

2. Evaluate acceptance probability
   
   \[
   \alpha = \min \left(1, \frac{p(y_t | x_{t|t-1}^*)}{p(y_t | x_{t|t}^{(k)})} \right)
   \]

3. Let \( x_{t|t}^* \) enter the posterior sample \( \{x_{t|t}^{(i)}\} \) with probability \( \alpha \), otherwise retain \( x_{t|t}^{(k)} \)

end (for \( k \))

yields \( \{x_{t|t}^{(i)}\} \), a sample from \( p(x_t | y_{1:t}) \)

- flexible, configurable and efficient (+ a parallel with SIR)
- can alleviate sample impoverishment (resample-move)
Limitations

**Theory:** convergence to target marginal distribution $p(x_t \mid y_{1:t})$ holds under weak assumptions (e.g. Kunsch 2005)

**Practical Problem:** The standard particle filtering (SIR) when used in DA suffers from sample impoverishment (recall path degeneracy)

Typically, the prediction ensemble far away from the observed state (due to small ensemble size, and high dimensional state space).

$\Rightarrow$ likelihood is poorly represented and estimation compromised.

What modifications are made to improve particle filters?
Some Fixes

Add “Jitter”: overdisperse the sample of predictive density (inflate process error)

Make a Gaussian approximation, or transformation/anamorphosis:
Can Use Kalman filter updating equation

Approximate the Likelihood function: change its functional form, or inflate or alter measurement error.

Error Subspace: confine stochasticity in parameters only. Dimension reduction.

Use Fixed lag smoother, Batch processing incorporate observations from multiple times into observation update.

Clever Proposal Distributions and look-ahead filters: move beyond using “prior” (predictive density) as proposal.
What about Parameter Estimation?

1. *State Augmentation*: append parameters to the state

\[
\tilde{x}_t = \begin{pmatrix} x_t \\ \theta_t \end{pmatrix}
\]

Can use particle filter machinery to solve (Kitagawa 1998). Also, see multiple iterative filtering (Ionides 2006)

2. *Likelihood Methods*: Use sample based likelihoods

\[
[y_{1:T} | \theta] = L(\theta | y_{1:T}) = \prod_{t=1}^{T} \int [y_t | x_t, \theta][x_t | y_{1:t-1}, \theta] dx_t \\
\approx \prod_{t=1}^{T} \frac{1}{n} \sum_{i=1}^{n} p(y_t | x_{t|t-1}^{(i)})
\]
Summary

1. State space model is statistical framework for considering dynamic models and measurements

2. Importance sampling provides theoretical basis for particle filters

3. Sequential importance resampling (SIR) yields a straightforward and workable algorithm for the filtering problem

4. For the realistic problems in data assimilation, particle filters must be modified either via approximations, or clever use of proposal distributions.
Some Key References


