Asymptotic Genealogies of Sequential Monte Carlo Algorithms

Suzie Brown

26 April 2019

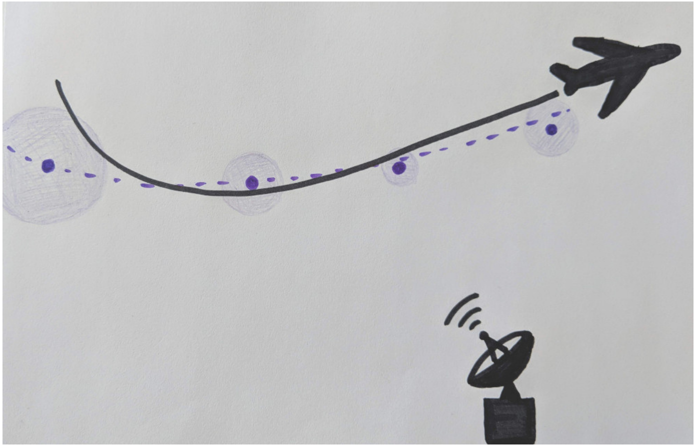
State space models

$$X_0, \ldots, X_T \in \mathcal{X}$$
$$Y_0, \ldots, Y_T \in \mathcal{Y}$$

$$X_0 \sim \mu(\cdot)$$
$$X_{t+1} \mid (X_t = x_t) \sim f(\cdot|x_t)$$
$$Y_t \mid (X_t = x_t) \sim g(\cdot|x_t)$$

$$\begin{tikzcd}
 & Y_{t-1} & Y_t & \\
\cdots \arrow[r, "f"] & X_{t-1} \arrow[u, "g"'] \arrow[r, "f"] & X_t \arrow[u, "g"'] \arrow[r, "f"] & \cdots
\end{tikzcd}$$

Target tracking



use noisy radar data to infer position/trajectory of aircraft:

- f models how aircraft moves
- g models uncertainty in radar measurements

Inference problems

Filtering: where is it now? $p(x_t | y_{0:t})$

Prediction: where will it go next? $p(x_{t+1} | y_{0:t})$

Smoothing: where has it been? $p(x_{0:t} | y_{0:t})$

Smoothing is "harder" than filtering/prediction.

Deterministic solutions

Kalman filter

Under a linear Gaussian model:

$$
\begin{aligned}
X_0 &\sim \mathcal{N}(0, \Sigma_0) \\
X_{t+1} \mid (X_t = x_t) &\sim \mathcal{N}(Ax_t, \Sigma_x) \\
Y_t \mid (X_t = x_t) &\sim \mathcal{N}(Bx_t, \Sigma_y)
\end{aligned}
$$

we can recursively compute filtering distributions. Then a backward pass of RTS smoother provides smoothing distributions.

This class of models is rather restrictive.

Extended Kalman filter

In non-linear Gaussian models, use a local linear approximation and apply Kalman filter.

This requires gradients and performs poorly in models that are very non-linear.

Unscented Kalman filter

For highly non-linear Gaussian models, replace the transition step of Kalman filter by propagating a representative set of points through f.

Deterministic solutions

- Kalman filter provides optimal filter for linear Gaussian models, but extended/unscented Kalman filter are no longer optimal.
- All of these methods require model to be Gaussian. Deterministic solutions are available for some other conjugate families, but these are still restrictive.
- Solutions are also available in the case that $\mathcal{X}$ is finite (integrals become sums), but we will generally consider $\mathcal{X}$ to be a continuous space.

Sequential Monte Carlo (SMC) provides general-purpose (stochastic) methods that do not require a tractable model. Only requires sampling from $f(\cdot|x)$, and pointwise evaluation of $g(y|x)$ up to a normalising constant for each y.

Sequential Monte Carlo

Prior: $p(x_{0:t}) = \mu(x_0) \prod_{i=1}^{t} f(x_i|x_{i-1})$

Likelihood: $p(y_{0:t}|x_{0:t}) = \prod_{i=0}^{t} g(y_i|x_i)$

Posterior: $p(x_{0:t}|y_{0:t}) \propto \mu(x_0) g(y_0|x_0) \prod_{i=1}^{t} f(x_i|x_{i-1}) g(y_i|x_i)$

- Represent posterior distribution at time t with N particles.
- Posterior factorises sequentially — avoid increase of dimension with T.

Algorithm

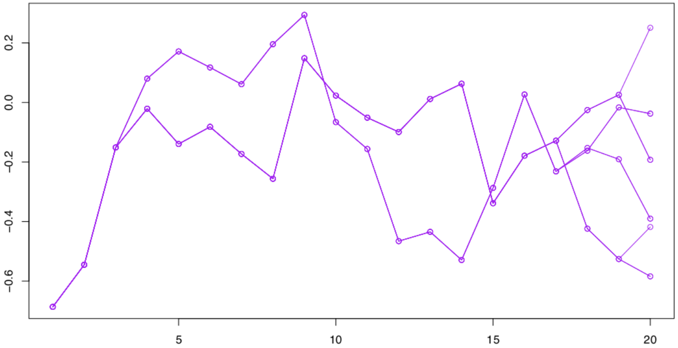
After initialisation, iterate these steps:

- **Propagate:** move the particles through the transition f
- **Calculate weights:** weight each particle according to g
- **Resample:** duplicate high-weight particles and kill off low-weight ones

Approximate posterior distribution $p(x_{0:t}|y_{0:t})$ by the empirical measure of the particles:

$$\hat{p}(x_{0:t}|y_{0:t}) = \frac{1}{N}\sum_{i=1}^{N} \delta_{X_{0:t}^{(i)}}(x_{0:t})$$

Ancestral degeneracy



For smoothing we need a sample of trajectories.

Resampling means that trajectories of time T particles coalesce backwards in time.