• Suppose $X = (x_1, \ldots, x_d)$; of interest is
  $\pi(X) = \pi(x_1) \pi(x_2 | x_1) \cdots \pi(x_d | x_1, \ldots, x_{d-1})$
• Similarly, the trial density can also be decomposed
  $q(X) = q(x_1) q(x_2 | x_1) \cdots q(x_d | x_1, \ldots, x_{d-1})$
  $w(X) = \pi(X) q(x_1 | x_2) q(x_2 | x_3) \cdots q(x_d | x_1, \ldots, x_{d-1})$
  – The weight can be computed sequentially
  – at any stage we may want to reject a partial sample because the
    “temporary” weight is already very small.
  – In practice, we create a sequence of approximations, $\pi(x_1), \pi(x_2 | x_1), \ldots$, to guide the IS at each stage.
  – See (Kong, Liu, Wong, 1994; Liu, Chen, Wong, 1998)

How to generate a “good” table?

• Let a $m \times n$ table be represented by
  $\Gamma = (X_1, \ldots, X_n)$
• Sequential importance sampling for tables:
  $q(\Gamma) = q(X_1) \times q(X_2 | X_1) \times \cdots \times q(X_n | X_1, \ldots, X_{n-1})$
  – Each $X_i$ is a m-dim vector of $c_i$ 1’s and $m-c_i$ 0’s
  – And we also have row-sum information
Details for sampling each column

- Focus on $X_1$ (since it is a recursive procedure). We can write it as $X_1 = (x_1, \ldots, x_n)^T$.
  - We prefer its $j$th position to have a “1” if $r_j$ is big.
  - Poisson-Binomial distribution: if $x_i$=1 if $p_i(1-x_i)$, then $S=Z_1+\ldots+Z_m$ has a P-B distribution.
  - Distribution of $X_j$ conditional on $c_1$ and $r_1, \ldots, r_m$ is just the above conditional distribution with $p_i$ replaced by $r_i/n$.
  - More flexible weights: $w_i = (r_i/(n-r_i))^g$.

- Constraint on column sums and row sums.

Weighted Sampling procedure

- In Chen, Dempster and Liu (1994, Biometrika)
  - Compute the normalizing constant $O(m^c_1)$
  - Sampling can be done sequentially and takes $O(m)$

For the finch data

<table>
<thead>
<tr>
<th>number of samples for each estimate</th>
<th>mean $\tilde{n}$ of $n_m$</th>
<th>time to get 100 estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>$0.7 \times 10^8$</td>
<td>10 seconds</td>
</tr>
<tr>
<td>100</td>
<td>$0.65 \times 10^8$</td>
<td>1 min 20 seconds</td>
</tr>
<tr>
<td>1,000</td>
<td>$0.72 \times 10^8$</td>
<td>12 min</td>
</tr>
<tr>
<td>10,000</td>
<td>$0.713 \times 10^8$</td>
<td>2 hours</td>
</tr>
</tbody>
</table>

Table 1: Estimations for finch data

CV$^2 = 1.3$ for $g=1$, optimal $g=1.11$, with CV$^2 = 0.7$

Test of co-inhabitation: $S^2 = \frac{1}{m(m-1)} \sum_{i\neq j} h_{ij}$

Some special cases

- If all $a_{ij} = 1$, then $S_d$ is the collection of all permutations.
- If all $a_{ij} = 0$ and $a_{ii} = 1$ for $i \neq j$, then $S_d$ is the set of all permutations with no “fixed point.”
- A bipartite graph with $2n$ vertices, $a_{ij} = 1$ if $\exists$ an edge between $x_i$ and $y_j$.
  - $S_d$ is the set of perfect matches.
- In statistical physics, a “dimer covering” problem can be stated as that of computing permanent.
Approximating Permanents

- Recall \( S_\pi = \text{perm}(A) = \sum_{\pi \in \pi} \prod_{i=1}^{n} a_{\pi(i)} \)

- For any permutation \( \pi \), define its “value” \( a(\pi) = \prod_{i=1}^{n} a_{\pi(i)} \) either one or zero.

- We want to “choose” permutations so that their “values” tend not to be zero.

- We sample \( \pi(1), \pi(2), \ldots, \pi(n) \) sequentially with

- After sampling \( \pi(1) = j \), we cross out the 1st column and the \( j \)-th row and move to the second column

- If at some step we are forced to take zero’s, we stop and give that sample a zero weight.

The State-Space Model (Hidden Markov Model)

- The State-Space Model

  \[
  \begin{align*}
  \text{Obs exp.} & \quad y_t | x_t, \phi = f_t(x_t, \phi) \\
  \text{State exp.} & \quad x_t | x_{t-1} = g_t(x_t, x_{t-1}, \theta)
  \end{align*}
  \]

- Only the \( y_t \) are observable!!!

Dynamic (Nonlinear) Models

- System: \( X_N = (x_{10}, \ldots, x_N) \)

- Of interest: estimating the \( x_t \) \textit{on-line}

A Simple Example

- Consider

  \[
  \begin{align*}
  x_t &= 0.9 x_{t-1} + \sigma_j \epsilon_t; \quad \text{with } \epsilon_t \sim N(0,1) \\
  y_t &= x_t + \eta_t; \quad \text{with } \eta_t \sim N(0, \sigma^2) \\
  J_t &\text{ are iid Bernoulli r.v. with} \\
  p(J_t = 1) &= 1 - p(J_t = 2) = \pi_t \\
  \sigma_j &\neq \sigma_t
  \end{align*}
  \]

- We only observe \( y_t \), interested in recovering \( x_t \)

The Question

- Assume that we \textit{KNOW} the parameters and \textit{KNOW} the distribution of the initial signal \( x_0 \), how should we estimate those unobserved signals?

- Kalman filter: \( x_t = a x_{t-1} + \sigma_j \epsilon_t; \quad \text{with } \epsilon_t \sim N(0,1) \\
  y_t = x_t + \eta_t; \quad \text{with } \eta_t \sim N(0, \sigma^2) \\
  J_t &\text{ are iid Bernoulli r.v. with} \\
  p(J_t = 1) &= 1 - p(J_t = 2) = \pi_t \\
  \sigma_j &\neq \sigma_t
  \]

\[
\begin{align*}
A &\text{ priori: } x_t \sim \mu, \sigma^2 \\
A &\text{ posterior: } x_t | y_t \sim \mu, \sigma^2
\end{align*}
\]

- \( \mu = \frac{a \mu_t + \sigma^2_\epsilon_j \sigma^2_t}{a \sigma^2_\epsilon_j + \sigma^2_t} \)

- \( \sigma_t' = \frac{1}{a \sigma^2_\epsilon_j + \sigma^2_t} \)

- \( \sigma_j' = \frac{1}{a \sigma^2_\epsilon_j + \sigma^2_t} \)

- Best online estimate of the \( x_t \): the Bayes estimate \( \hat{x}_t = E(x_t | y_1, \ldots, y_t) \)

- Delayed estimate \( \tilde{x}_t = E(x_t | y_1, \ldots, y_{t+1}) \)

- Question: how to compute these estimates?

  - Kalman filter for linear models
  - Numerical approximation for others.
  - Monte Carlo?
Monte Carlo Approach

- Local view: how to do

\[ x_0 \rightarrow x_1 \rightarrow x_2 \]

How to propagate from a distribution of \( x_0 \) to a distribution of \( x_1 \).

\[ g_0(x_0) \rightarrow P(x_1 | y_1) \propto \int g(x_1 | x_0)g_0(x_0)dx_0 \]

Denoted by \( g(x_1) \)

Prior weighted by the likelihood.

Back to the State-space model

- Suppose we knew \( \pi_{s-1}(x_{s-1}) = p(x_{s-1} | Y_{s-1}) \), for time \( S \)

\[ \pi_s(x_s) = p(x_s | Y_s) \propto \int \pi_{s-1}(x_{s-1})p_s(x_s | x_{s-1})dx_{s-1} \]

A Monte Carlo approximation of \( \pi_s(x_s) \)

A Graphical View of the Particle/Bootstrap Filter

Forward looking

- Examine the next few steps:

\[ [x_t | y_1, y_2, y_3] \propto \int \cdots dx_0 dx_2 dx_s \]

Approximations?

MCMC iterations?

What has been achieved?

- From a set of Monte Carlo samples at time \( t-1 \) to a set of weighted Monte Carlo samples at time \( t \):

\[ \{x_{t-1}^{(i)} \}_{i=1}^m \xrightarrow{\text{samp & wgt.}} \{x_t^{(j)}, w_t^{(j)} \}_{j=1}^m \]

- The weighted sample is proper w.r.t. the new posterior:

\[ \frac{1}{m} \sum_{i=1}^m w_t^{(i)}h(x_t^{(i)}) \xrightarrow{\text{w.m.t.}} E[h(x_t) | Y_t] \]

- We do not have to do resampling and do not have to have equally weighted samples.

\[ \{x_{t-1}^{(i)}, w_{t-1}^{(i)} \}_{i=1}^m \xrightarrow{\text{samp & wgt.}} \{x_t^{(j)}, w_t^{(j)} \}_{j=1}^m \]

Important issue: how to produce a good set of (weighted) particles

- Restriction: particles at time 0 cannot be easily regenerated or "moved."

  - If \( p(x_t | x_{t-1}^{(i)}, y_t) \) is easy to handle, then \( x_t^{(i)} \) should be drawn from it. (weighting is still needed)

  - If \( p(x_t | x_{t-1}^{(i)}, y_t) \) is not in closed form, we may do some ad hoc adjustment, e.g., kernel smoothing? MCMC iterations?

  - Other? Take an importance sampling perspective.
Global View

- If we can simulate from the "posterior"
  \[ \pi_j(x_j) = p(x_j, x_{j-1}, \ldots, x_1 | y_1, \ldots, y_j; \theta, \phi) \]
  \[ \propto g_j(x_j) \prod_{t=1}^{j} f_j(y_t, | x_t, \phi) g_j(x_t, | x_{t-1}, \theta) \]

A growing system with one \( x_j \) added a time.

With Resampling

- With weighted samples \( (w_j^{(i)}, x_j^{(i)}), j=1, \ldots, m \), we can resample \( x_j^{(i)} \) from the set of \( x_j^{(j)} \) with probability proportional to the \( w_j^{(i)} \). Then we proceed with \( (1, x_j^{(i)}) \) for processing \( x_{j+1} \).
- This is equivalent to use \( \frac{1}{m} \sum_{j=1}^{m} w_j^{(i)} p(x_{j+1} | y_j, x_j^{(i)}) \)
- for generating \( x_{j+1} \)
- Why resampling? No use if we estimate \( E(h(x_j)|\mathcal{Y}) \)
- Useful only for the future!!
- Don’t need real “resampling”, just a “reallocation.”

A Mixture Kalman Filter

- For conditional DLM,
  \[ x_t = G_{t,j} x_{t-1} + \varepsilon_{t,j}, \quad \text{if } J_{t,j} = j_1; \]
  \[ y_t = F_{t,j} x_t + \eta_{t,j}, \quad \text{if } J_{t,j} = j_2; \]

  If we knew the values of the indicators \( J_t = (J_{t,1}, J_{t,2}) \), then the model reduces to a DLM and the KF can be applied.

  - Impute the indicators using the sequential importance sampling technique.

Conditional Dynamic Linear Model

- We consider the following model:
  \[ x_t = G_{t,j} x_{t-1} + \varepsilon_{t,j}, \quad \text{if } J_{t,j} = j_1; \]
  \[ y_t = F_{t,j} x_t + \eta_{t,j}, \quad \text{if } J_{t,j} = j_2; \]

  where \( \varepsilon_{t,j} \sim N(0, \sigma^2); \eta_{t,j} \sim N(0, \sigma^2) \)

  \( J_t = (J_{t,1}, J_{t,2}) \) is called a dynamic indicator process.

  We can "reconfigure" the system!

Example: Outlier model

- Consider
  \[ x_t = 0.9x_{t-1} + \sigma_x \varepsilon_t; \quad \text{with } \varepsilon_t \sim N(0, 1) \]
  \[ y_t = x_t + \sigma_y \eta_t; \quad \text{with } \eta_t \sim N(0, \sigma^2) \]

  \( J_t \) are iid Bernoulli, with \( p(J_t = 1) = 0.9 \)

  \( \sigma_x = 0.2; \sigma_y = 1.2; \sigma = 0.8 \)

  Key:
  \[ p(J_t = j | \mathcal{E}_{t-1}; y_t) = C_j \mathcal{N}(x_t | \mu_j, V_j) \]

  A mixture of \( M=50 \) filters gave us good estimates of the state variables \( x_t \).