Smoother

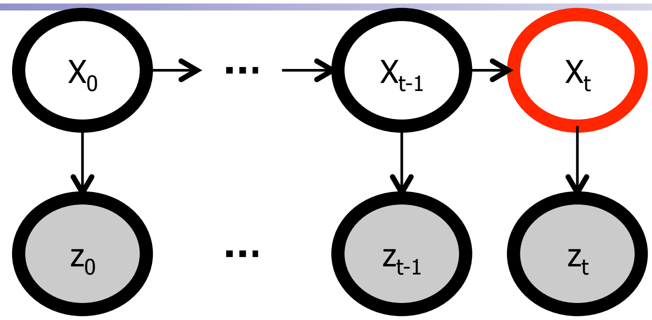
Pieter Abbeel
UC Berkeley EECS

Many slides adapted from Thrun, Burgard and Fox, Probabilistic Robotics

Overview

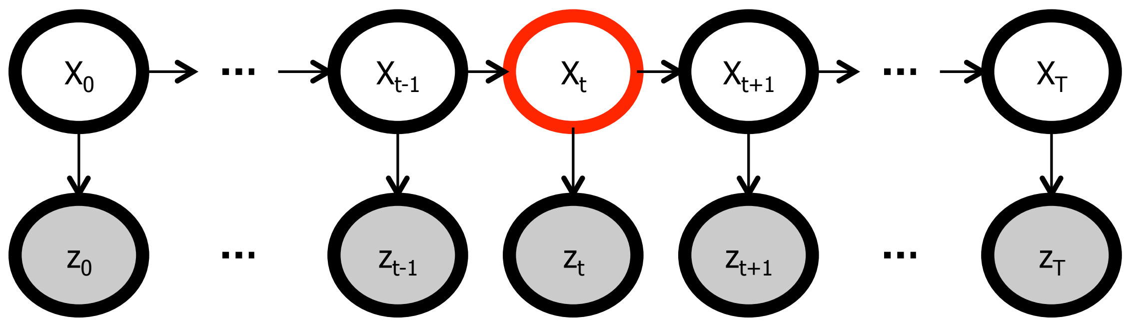
- **Filtering:**

$$P(x_t|z_0, z_1, \ldots, z_t)$$



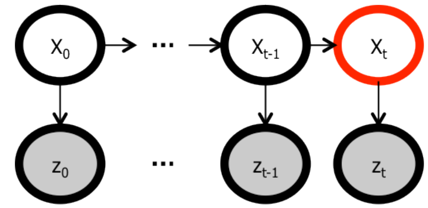
- **Smoothing:**

$$P(x_t|z_0, z_1, \ldots, z_T)$$



- Note: by now it should be clear that the "u" variables don't really change anything conceptually, and going to leave them out to have less symbols appear in our equations.

Filtering



$$
\begin{aligned}
P(x_2|z_0,z_1,z_2) &\propto P(x_2,z_0,z_1,z_2) \\
&= \sum_{x_0,x_1} P(z_2|x_2)P(x_2|x_1)P(z_1|x_1)P(x_1|x_0)P(z_0|x_0)P(x_0) \\
&= P(z_2|x_2)\sum_{x_1} P(x_2|x_1)P(z_1|x_1)\sum_{x_0} P(x_1|x_0)P(z_0|x_0)P(x_0)
\end{aligned}
$$

$$P(x_0, z_0)$$

$$P(x_1, z_0)$$

$$P(x_1, z_0, z_1)$$

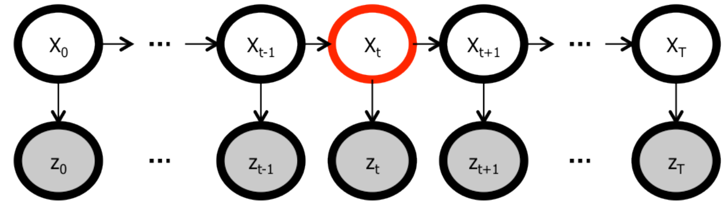
$$P(x_2, z_0, z_1)$$

$$P(x_2, z_0, z_1, z_2)$$

- Generally, recursively compute:

$$
\begin{aligned}
P(x_{t+1}, z_0, \ldots, z_t) &= \sum_{x_t} P(x_{t+1}|x_t)P(x_t, z_0, \ldots, z_t) \\
P(x_{t+1}, z_0, \ldots, z_t, z_{t+1}) &= p(z_{t+1}|x_{t+1})P(x_{t+1}, z_0, \ldots, z_t)
\end{aligned}
$$

Smoothing



$$
\begin{aligned}
& P(x_2|z_0, z_1, z_2, z_3, z_4) \\
\propto\ & P(x_2, z_0, z_1, z_2, z_3, z_4) \\
=\ & \sum_{x_0,x_1,x_3,x_4} P(z_4|x_4)P(x_4|x_3)P(z_3|x_3)P(x_3|x_2)P(z_2|x_2)P(x_2|x_1)P(z_1|x_1)P(x_1|x_0)P(z_0|x_0)P(x_0) \\
=\ & \sum_{x_3,x_4} P(z_4|x_4)P(x_4|x_3)P(z_3|x_3)P(x_3|x_2)P(z_2|x_2) \left(\sum_{x_1} P(x_2|x_1)P(z_1|x_1) \left(\sum_{x_0} P(x_1|x_0)P(z_0|x_0)P(x_0) \right) \right) \\
=\ & \left(\sum_{x_3} P(z_3|x_3)P(x_3|x_2) \left(\sum_{x_4} P(z_4|x_4)P(x_4|x_3) \right) \right) P(z_2|x_2) \left(\sum_{x_1} P(x_2|x_1)P(z_1|x_1) \left(\sum_{x_0} P(x_1|x_0)P(z_0|x_0)P(x_0) \right) \right)
\end{aligned}
$$

$b(x_3) = P(z_4|x_3)$

$P(x_1, z_0, z_1)$

$b(x_2) = P(z_3, z_4|x_2)$

$P(x_2, z_0, z_1, z_2)$

- Generally, recursively compute:
  - Forward: (same as filter)

$$
\begin{aligned}
P(x_{t+1}, z_0, \ldots, z_t) &= \sum_{x_t} P(x_{t+1}|x_t)P(x_t, z_0, \ldots, z_t) \\
P(x_{t+1}, z_0, \ldots, z_t, z_{t+1}) &= p(z_{t+1}|x_{t+1})P(x_{t+1}, z_0, \ldots, z_t)
\end{aligned}
$$

  - Backward:

$$
\begin{aligned}
P(z_{t+1}, \ldots, z_T|x_{t+1}) &= P(z_{t+1}|x_{t+1})P(z_{t+2}, \ldots, z_T|x_{t+1}) \\
P(z_{t+1}, \ldots, z_T|x_t) &= \sum_{x_{t+1}} P(x_{t+1}|x_t)P(z_{t+1}, \ldots, z_T|x_{t+1})
\end{aligned}
$$

  - Combine: $P(x_t, z_0, \ldots, z_T) = P(x_t, z_0, \ldots, z_t)P(z_{t+1}, \ldots, z_T|x_t)$

Complete Smoother Algorithm

- Forward pass (= filter):

  1. Init: $a_0(x_0) = P(z_0|x_0)P(x_0)$
  2. For $t = 0, \ldots, T-1$
     - $a_{t+1}(x_{t+1}) = P(z_{t+1}|x_{t+1}) \sum_{x_t} P(x_{t+1}|x_t) a_t(x_t)$

- Backward pass:

  1. Init: $b_T(x_T) = 1$
  2. For $t = T-1, \ldots, 0$
     - $b_t(x_t) = \sum_{x_{t+1}} P(x_{t+1}|x_t) P(z_{t+1}|x_{t+1}) b_{t+1}(x_{t+1})$

- Combine:

  1. For $t = 0, \ldots, T$
     - $P(x_t, z_0, \ldots, z_T) = P(x_t, z_0, \ldots, z_t) P(x_{t+1}, z_{t+1}, \ldots, z_T|x_t) = a_t(x_t) b_t(x_t)$

Note 1: computes for all times t in one forward+backward pass

Note 2: can find $P(x_t \mid z_0, \ldots, z_T)$ by simply renormalizing

Important Variation

- Find $P(x_t, x_{t+1}, z_0, \ldots, z_T)$

- Recall:
$$\begin{aligned} a_t(x_t) &= P(x_t, z_0, \ldots, z_T) \\ b_t(x_t) &= P(z_{t+1}, \ldots, z_T | x_t) \end{aligned}$$

- So we can readily compute

$$P(x_t, x_{t+1}, z_0, \ldots, z_T)$$
$$= P(x_t, z_0, \ldots, z_t) P(x_{t+1} | x_t, z_0, \ldots, z_t) P(z_{t+1}, \ldots, z_T | x_{t+1}, x_t, z_0, \ldots, z_t)$$ (Law of total probability)
$$= P(x_t, z_0, \ldots, z_t) P(x_{t+1} | x_t) P(z_{t+1}, \ldots, z_T | x_{t+1})$$ (Markov assumptions)
$$= a_t(x_t) P(x_{t+1} | x_t) b_{t+1}(x_{t+1})$$ (definitions a, b)

Exercise

- Find $P(x_t, x_{t+k}, z_0, \ldots, z_T)$

Kalman Smoother

- = smoother we just covered instantiated for the particular case when $P(X_{t+1} \mid X_t)$ and $P(Z_t \mid X_t)$ are linear Gaussians
- We already know how to compute the forward pass (=Kalman filtering)
- Backward pass:

$$b_t(x_t) = \int_{x_{t+1}} P(x_{t+1}|x_t)P(z_{t+1}|x_{t+1})b_{t+1}(x_{t+1})dx_{t+1}$$

- Combination:

$$P(x_t, z_0, \ldots, z_T) = a_t(x_t)b_t(x_t)$$

Kalman Smoother Backward Pass

- TODO: work out integral for b_t
- TODO: insert backward pass update equations
- TODO: insert combination → bring renormalization constant up front so it's easy to read off $P(x_t \mid z_0, \ldots, z_T)$

Matlab code data generation example

```
A = [   0.99    0.0074;   -0.0136    0.99]; C = [ 1 1 ; -1 +1];
x(:,1) = [-3;2];
Sigma_w = diag([.3 .7]); Sigma_v = [2 .05; .05 1.5];
w = randn(2,T); w = sqrtm(Sigma_w)*w;  v = randn(2,T); v = sqrtm(Sigma_v)*v;
for t=1:T-1
    x(:,t+1) = A * x(:,t) + w(:,t);
    z(:,t) = C*x(:,t) + v(:,t);
end
% now recover the state from the measurements
P_0 = diag([100 100]);  x0 =[0; 0];
% run Kalman filter and smoother here
% + plot
```

Kalman filter/smoother example