

Lecture 10: State Estimation of Hidden Markov Process

Lecturer: Jiantao Jiao, Tsachy Weissman Scribe: Dominic Delgado, Lewis Guignard, Greg Kehoe

In this lecture, we complete and expand the recursive algorithm for state estimation of a hidden markov process through a memoryless noisy channel.

1 Recap

Let there be a Markov Process $\{X_n\}_{n \geq 1}$ sent through a memoryless channel, and measurements $\{Y_n\}_{n \geq 1}$. Then the stochastic system is characterized by:

$$P_x, \{P_{X_t|X_{t-1}}\}_{t \geq 2}, \{P_{Y_t|X_t}\}_{t \geq 1}$$

Let

$$\begin{aligned}\alpha_t(X_t) &= p(x_t|y^t) \\ \beta_t(X_t) &= p(x_t|y^{t-1}) \\ \gamma_t(X_t) &= p(x_t|y^n)\end{aligned}$$

In lecture 9, we found that:

$$\begin{aligned}\alpha_t(X_t) &= \frac{\beta_t(X_t)p(y_t|x_t)}{\sum_{\tilde{x}_t} \beta_t(X_t)p(y_t|\tilde{x}_t)} \\ \beta_{t+1}(X_t) &= \sum_{x_t} \alpha_t(x_t)p(x_{t+1}|x_t) \\ \gamma_t(X_t) &= \sum_{X_{t+1}} \frac{\gamma_{t+1}\alpha_{t+1}p(x_{t+1}|x_t)}{\beta_{t+1}(X_{t+1})}\end{aligned}$$

2 Finding $P(x^n|y^n)$

Claim:

$$X^{t-1} - (X_t, Y^n) - X_{t+1}^n$$

ie given all observations Y^n , X^n is still a Markov Process.

Proof: We prove using the "Markov Lemma", described in Lecture 9.

$$\begin{aligned}P(x^n, y^n) &= \prod_{i=1}^n p(x_i|x_{i-1})p(y_i|x_i) \\ &= \prod_{i=1}^t p(x_i|x_{i-1})p(y_i|x_i) \prod_{i=t+1}^n p(x_i|x_{i-1})p(y_i|x_i)\end{aligned}$$

These two terms can be seen as two general funtions:

$$\begin{aligned} &= \phi_1(x^t, y^t) \phi_2(x_t^n, y_{t+1}^n) \\ &= \phi_1(x^{t+1}, x_t, y^n) \phi_2(x_t, x_{t+1}^n, y^n) \end{aligned}$$

.. Where we've trivially expanded the vectors Y to go to n as this includes any subset of Y^n .
Looking at the posterior:

$$\begin{aligned} p(x^n|y^n) &= p(x_n|y^n)p(x_{n-1}|x_n, y^n)p(x_{n-2}|x_{n-1}, y^n), \dots, p(x_1|x_2, y^n) \\ &= p(x_n|y^n) \prod_{t=1}^{n-1} p(x_t|x_{t+1}, y^t) \end{aligned}$$

We let $y^n \rightarrow y^t$ because we already have clean state information x_{t+1}

$$\begin{aligned} p(x_t|x_{t+1}, y^t) &= \frac{p(x_t, x_{t+1}|y^t)}{p(x_{t+1}|y^t)} = \frac{p(x_t|y^t)p(x_{t+1}|x_t, y^t)}{p(x_{t+1}|y^t)} \\ &= \frac{\alpha_t(x_t)p(x_{t+1}|x_t)}{\beta_{t+1}(x_{t+1})} \\ \Rightarrow p(x^n|y^n) &= \gamma_n(x_n) \prod_{t=1}^{n-1} \frac{\alpha_t(x_t)p(x_{t+1}|x_t)}{\beta_{t+1}(x_{t+1})} \\ \log p(x^n|y^n) &= \sum_{t=1}^n g_t(x_t, x_{t+1}) \end{aligned}$$

Where we define g as:

$$\begin{aligned} g_t(x_t, x_{t+1}) &:= \log \frac{\alpha_t(x_t)p(x_{t+1}|x_t)}{\beta_{t+1}(x_{t+1})}, \quad t \in \{1, \dots, n-1\} \\ g_t(x_t, x_{t+1}) &:= \log \gamma_n(x_n), \quad t = n \end{aligned}$$

- if g was a function of x_t only, we could simply use a greedy algorithm and maximize each term.
- since g_t is a function of both x_t, x_{t+1} , we can be *almost* greedy

What we want:

$$\max_{x^n} \sum_{t=1}^n g_t(x_t, x_{t+1})$$

Definition 1. for $1 \leq k \leq n$, Let

$$M_k(x_k) := \max_{x_{k+1}^n} \sum_{t=k}^n g_t(x_t, x_{t+1})$$

Then,

$$\begin{aligned}
M_k(x_k) &= \max_{x_{k+1}} \max_{x_{k+2}^n} g_k(x_k, x_{k+1}) + \sum_{t=k+1}^n g_t(x_t, x_{t+1}) \\
&= \max_{x_{k+1}} (g_k(x_k, x_{k+1}) + \max_{x_{k+2}^n} \sum_{t=k+1}^n g_t(x_t, x_{t+1})) \\
&= \max_{x_{k+1}} (g_k(x_k, x_{k+1}) + M_{k+1}(x_{k+1}))
\end{aligned}$$

.. and we see the recursive relationship

3 Bellman Equations, aka Viterbi Algorithm

aka dynamic programming

Initialization: $M_n(x_n) = \log \gamma_n(x_n)$

Recursion: For $1 \leq k \leq n$, compute M_k via M_{k+1}

Finally: $M = \max_{x_1} M_1(x_1)$

3.1 Maximizing Sequence:

$$\begin{aligned}
&(x_1^{ML}, x_2^{ML}, \dots, x_n^{ML}) \\
\text{where: } x_1^{ML} &= \arg \max_{x_1} M(x_1) \\
\text{and: } x_{k+1}^{ML} &= \arg \max_{x_{k+1}} (g_k(x_k, x_{k+1}) + M_{k+1}(x_{k+1}))
\end{aligned}$$

3.2 Reflection on forward - backards recursion

Consider:

$$\begin{aligned}
X &\sim p_x \rightarrow \boxed{p_{Y|X}} \rightarrow Y \\
p_{X|Y}(x|y) &= \frac{p_X(x)p_{Y|X}(y|x)}{\sum_{\tilde{x}} p_{Y|X}(y|\tilde{x})} \\
&= F(p_X, p_{Y|X}, y) \bullet \text{a vector}
\end{aligned}$$

Also,

$$F(p_X, p_{Y|X}) \bullet \text{a matrix} p_Y(y) = \sum_{\tilde{x}} p_x(\tilde{x}) p_{Y|X}(y|\tilde{x}) = G(p_X, p_{Y|X})$$

3.3 Recursions redefined

Forward Recursions redefined

$$\alpha_t = F(\beta_t, p_{Y_t|X_t}, y_t) \tag{1}$$

$$\beta_{t+1} = G(\alpha_t, p_{x_{t+1}|x_t}) \tag{2}$$

Backward Recursions redefined

$$\gamma_t = G(\gamma_{t+1}, F(\alpha_t, P_{x_{t+1}|x_t})) \tag{3}$$