Problem Solving with AI Techniques
Hidden Markov Models

Paul Weng

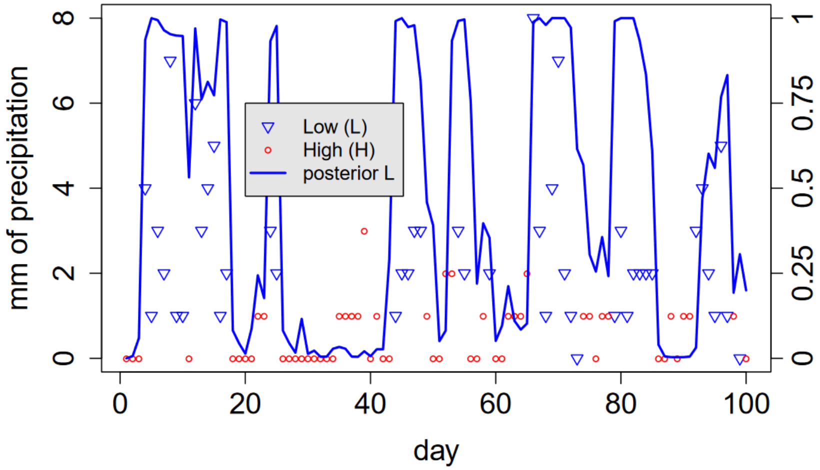
UM-SJTU Joint Institute

VE593, Fall 2018

Motivation

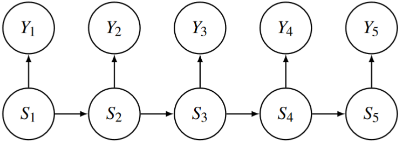
- State transition may be stochastic
- True state is not observed
- Sequential observations (time series data)
- Applications:
    - Robot localization
    - Object tracking in videos
    - Speech processing
    - Medical monitoring
- Idea: Use a Markov model with hidden state and observation nodes

Example: Precipitation



from (Nuel, 2012)

Example: Precipitation (Contd.)



$$\mathbb{P}(S_i = L \mid S_{i-1} = H) = 0.3$$
$$\mathbb{P}(S_i = H \mid S_{i-1} = L) = 0.1$$

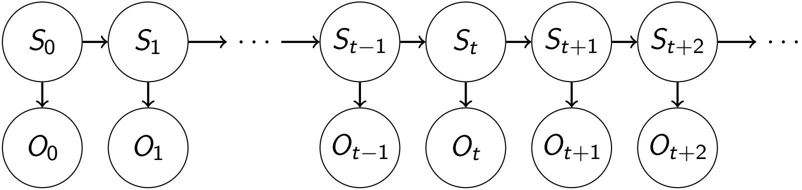
$$\mathbb{P}(Y_i = k \mid S_i = L) \sim Poisson(\lambda_L = \mathbb{E}[Y_i \mid S = L] = 3)$$
$$\mathbb{P}(Y_i = k \mid S_i = H) \sim Poisson(\lambda_H = \mathbb{E}[Y_i \mid S = H] = 0.1)$$

Table 1: Distribution of Y_i conditionally to S_i in the precipitation HMM.

k	0	1	2	3	4	5	6	7	8	9	10
$\mathbb{P}(Y_i = k \mid S_i = \mathrm{L})$	.050	.149	.224	.224	.168	.101	.050	.022	.008	.003	.001
$\mathbb{P}(Y_i = k \mid S_i = \mathrm{H})$	.607	.303	.076	.013	.002	.000	.000	.000	.000	.000	.000

Definition of Hidden Markov Model

- An HMM is defined as a tuple:
  - $\mathcal{S}$ a set of states
  - $\mathcal{O}$ a set of observations
  - $\mathbb{P}(S' \mid S)$ transition probabilities, denoted $\boldsymbol{p} = (p_{ss'})$
  - $\mathbb{P}(O \mid S)$ emission probabilities, denoted $\boldsymbol{q} = (q_{so})$
  - $\mathcal{P}(S_0)$ probability distribution of initial states, denoted $\boldsymbol{\pi} = (\pi_s)$



- This defines the joint probability:

$$\mathbb{P}(S_{0:T}, O_{0:T}) = \mathbb{P}(S_0) \prod_{t=1}^{T} \mathbb{P}(S_t \mid S_{t-1}) \prod_{t=1}^{T} \mathbb{P}(O_t \mid S_t)$$

Examples of HMM

- Speech recognition
  - Observations: acoustic signals
  - States: positions in words
- Hand gesture recognition with video camera
  - Observations: video frames
  - States: positions/orientations of hands
- GPS localization
  - Observations: GPS reading
  - States: positions on a map

What Can We Do With an HMM?

Different inference problems:

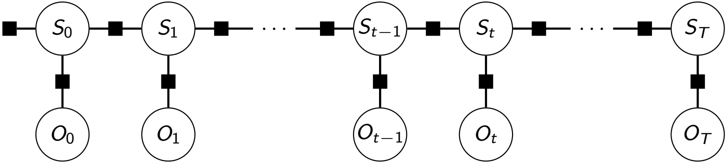
- Posterior marginal: $\mathbb{P}(S_t \mid o_{0:T})$
- Filtering: $\mathbb{P}(S_t \mid o_{0:t})$
- Prediction: $\mathbb{P}(S_{t'} \mid o_{0:t})$ where $t' > t$
- Smoothing: $\mathbb{P}(S_{t'} \mid o_{0:t})$ where $t' < t$
- Likelihood: $\mathbb{P}(o_{0:T})$
- Viterbi path: $\arg\max_{s_{0:T}} \mathbb{P}(S_{0:T} = s_{0:T} \mid o_{0:T})$

Inference in HMMs

- HMM = Bayes net with a tree structure
- Inference in HMMs is therefore efficient
- Many conditional independences:

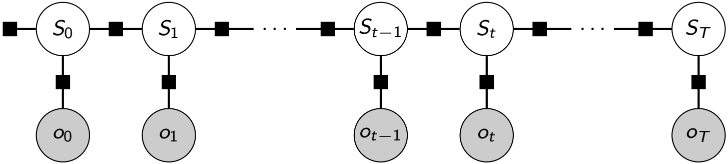
$$S_{t-k} \perp\!\!\!\perp S_{t+k'} \mid S_t$$
$$O_{t-k} \perp\!\!\!\perp O_{t+k'} \mid S_t$$

- Factor graph of an HMM



- Let's apply belief propagation to it to get the marginals $\mathbb{P}(S_t \mid o_{0:T})$

Belief Propagation in HMMs when $O_{0:T} = o_{0:T}$



Assuming $O_{0:T} = o_{0:T}$, the message passing equations yield:

Forward messages: $\mu_{S_{-1}\to S_0}(S_0) = \mathbb{P}(S_0)$

$$\mu_{S_{t-1}\to S_t}(S_t) = \sum_{S_{t-1}} \mathbb{P}(S_t \mid S_{t-1})\mu_{S_{t-2}\to S_{t-1}}(S_{t-1})\mu_{o_{t-1}\to S_{t-1}}(S_{t-1})$$

Backward messages: $\mu_{S_{T+1}\to S_T}(S_T) = 1$

$$\mu_{S_{t+1}\to S_t}(S_t) = \sum_{S_{t+1}} \mathbb{P}(S_t \mid S_{t+1})\mu_{S_{t+2}\to S_{t+1}}(S_{t+1})\mu_{o_{t+1}\to S_{t+1}}(S_{t+1})$$

Observation messages: $\mu_{o_t\to S_t}(S_t) = \mathbb{P}(o_t \mid S_t)$

Forward-Backward Algorithm

- Belief propagation is known as forward-backward algorithm with

$$\alpha_t(S_t) = \mu_{S_{t-1}\to S_t}(S_t)\mu_{o_t\to S_t}(S_t)$$
$$\beta_t(S_t) = \mu_{S_{t+1}\to S_t}(S_t)$$

- Posterior marginals:

$$\mathbb{P}(S_t \mid o_{0:T}) \propto \alpha_t(S_t)\beta_t(S_t)$$
$$\mathbb{P}(S_t, S_{t+1} \mid o_{0:T}) \propto \alpha_t(S_t)\mathbb{P}(S_{t+1} \mid S_t)\mu_{o_{t+1}\to S_{t+1}}(S_{t+1})\beta_{t+1}(S_{t+1})$$

Other Queries

- How can we solve a filtering query (e.g., $\mathbb{P}(S_t \mid o_{0:t})$)?

Other Queries

- How can we solve a filtering query (e.g., $\mathbb{P}(S_t \mid o_{0:t})$)?
- How can we solve a prediction query (e.g., $\mathbb{P}(S_{t'} \mid o_{0:t})$ with $t' > t$)?

Other Queries

- How can we solve a filtering query (e.g., $\mathbb{P}(S_t \mid o_{0:t})$)?
- How can we solve a prediction query (e.g., $\mathbb{P}(S_{t'} \mid o_{0:t})$ with $t' > t$)?
    - Belief propagation applies with $O_{t+1:t'}$ non-observed

Other Queries

- How can we solve a filtering query (e.g., $\mathbb{P}(S_t \mid o_{0:t})$)?
- How can we solve a prediction query (e.g., $\mathbb{P}(S_{t'} \mid o_{0:t})$ with $t' > t$)?
  - Belief propagation applies with $O_{t+1:t'}$ non-observed
  - Obs. messages for $\tau = t+1, \ldots t'$ become $\mu_{O_\tau \to S_\tau}(S_\tau) = \mathbf{1}$

Other Queries

- How can we solve a filtering query (e.g., $\mathbb{P}(S_t \mid o_{0:t})$)?
- How can we solve a prediction query (e.g., $\mathbb{P}(S_{t'} \mid o_{0:t})$ with $t' > t$)?
  - Belief propagation applies with $O_{t+1:t'}$ non-observed
  - Obs. messages for $\tau = t+1, \dots t'$ become $\mu_{O_\tau \to S_\tau}(S_\tau) = \mathbf{1}$
- How can we solve a smoothing query (e.g., $\mathbb{P}(S_{t'} \mid o_{0:t})$ with $t' < t$)?

Other Queries

- How can we solve a filtering query (e.g., $\mathbb{P}(S_t \mid o_{0:t})$)?
- How can we solve a prediction query (e.g., $\mathbb{P}(S_{t'} \mid o_{0:t})$ with $t' > t$)?
  - Belief propagation applies with $O_{t+1:t'}$ non-observed
  - Obs. messages for $\tau = t+1, \dots t'$ become $\mu_{O_\tau \to S_\tau}(S_\tau) = \mathbf{1}$
- How can we solve a smoothing query (e.g., $\mathbb{P}(S_{t'} \mid o_{0:t})$ with $t' < t$)?
- How can we solve a likelihood query (e.g., $\mathbb{P}(o_{0:T})$)?

Other Queries

- How can we solve a filtering query (e.g., $\mathbb{P}(S_t \mid o_{0:t})$)?
- How can we solve a prediction query (e.g., $\mathbb{P}(S_{t'} \mid o_{0:t})$ with $t' > t$)?
    - Belief propagation applies with $O_{t+1:t'}$ non-observed
    - Obs. messages for $\tau = t+1, \dots t'$ become $\mu_{O_\tau \to S_\tau}(S_\tau) = \mathbf{1}$
- How can we solve a smoothing query (e.g., $\mathbb{P}(S_{t'} \mid o_{0:t})$ with $t' < t$)?
- How can we solve a likelihood query (e.g., $\mathbb{P}(o_{0:T})$)?
    - Belief propagation applies again

Other Queries

- How can we solve a filtering query (e.g., $\mathbb{P}(S_t \mid o_{0:t})$)?
- How can we solve a prediction query (e.g., $\mathbb{P}(S_{t'} \mid o_{0:t})$ with $t' > t$)?
  - Belief propagation applies with $O_{t+1:t'}$ non-observed
  - Obs. messages for $\tau = t+1, \dots t'$ become $\mu_{O_\tau \to S_\tau}(S_\tau) = \mathbf{1}$
- How can we solve a smoothing query (e.g., $\mathbb{P}(S_{t'} \mid o_{0:t})$ with $t' < t$)?
- How can we solve a likelihood query (e.g., $\mathbb{P}(o_{0:T})$)?
  - Belief propagation applies again
  - The message received by o_0 or o_T provides the likelihood

Other Queries

- How can we solve a filtering query (e.g., $\mathbb{P}(S_t \mid o_{0:t})$)?
- How can we solve a prediction query (e.g., $\mathbb{P}(S_{t'} \mid o_{0:t})$ with $t' > t$)?
  - Belief propagation applies with $O_{t+1:t'}$ non-observed
  - Obs. messages for $\tau = t+1, \dots t'$ become $\mu_{O_\tau \to S_\tau}(S_\tau) = \mathbf{1}$
- How can we solve a smoothing query (e.g., $\mathbb{P}(S_{t'} \mid o_{0:t})$ with $t' < t$)?
- How can we solve a likelihood query (e.g., $\mathbb{P}(o_{0:T})$)?
  - Belief propagation applies again
  - The message received by o_0 or o_T provides the likelihood
- How can we compute the Viterbi path (e.g., $\arg\max_{s_{0:T}} \mathbb{P}(s_{0:T} | o_{0:T})$)?

How to Learn the Parameters of an HMM?

- Goal: Given i.i.d. training data $\mathcal{D} = \{o^1_{0:T}, \dots, o^N_{0:T}\}$, learn parameters $\boldsymbol{\theta} = \{\boldsymbol{p}, \boldsymbol{q}, \boldsymbol{\pi}\}$
- Issue: the ML and MAP approaches cannot be applied directly. Because of the hidden variables, the likelihood is not decomposable anymore:

$$
\begin{aligned}
\mathbb{P}(o_{0:T} \mid \boldsymbol{\theta}) &= \sum_{S_{0:T}} \mathbb{P}(S_{0:T}, o_{0:T}) \\
&= \sum_{S_{0:T}} \mathbb{P}(S_0) \prod_{t=1}^{T} \mathbb{P}(S_t \mid S_{t-1}) \mathbb{P}(o_t \mid S_t)
\end{aligned}
$$

- Idea: Use the Expectation-Maximization (EM) algorithm
- EM algorithm applied to HMM is called the Baum-Welch algorithm.

Principle of the general EM Algorithm

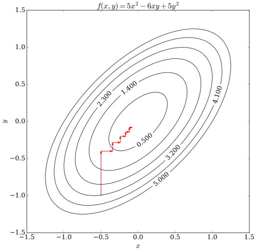
- Problem: Hard to maximize $\log \mathbb{P}(\boldsymbol{O} \mid \boldsymbol{\theta}) = \log \sum_{\boldsymbol{S}} \mathbb{P}(\boldsymbol{S}, \boldsymbol{O} \mid \boldsymbol{\theta})$ assuming there is only one observed sequence.
- Idea: Use Jensen inequality and maximize a lower bound!

$$
\begin{aligned}
\log \mathbb{P}(\boldsymbol{O} \mid \boldsymbol{\theta}) &= \log \sum_{\boldsymbol{S}} \mathbb{P}(\boldsymbol{S}, \boldsymbol{O} \mid \boldsymbol{\theta}) \\
&= \log \sum_{\boldsymbol{S}} \mathbb{Q}(\boldsymbol{S} \mid \boldsymbol{O}) \frac{\mathbb{P}(\boldsymbol{S}, \boldsymbol{O} \mid \boldsymbol{\theta})}{\mathbb{Q}(\boldsymbol{S} \mid \boldsymbol{O})} \\
&\geq \sum_{\boldsymbol{S}} \mathbb{Q}(\boldsymbol{S} \mid \boldsymbol{O}) \log \frac{\mathbb{P}(\boldsymbol{S}, \boldsymbol{O} \mid \boldsymbol{\theta})}{\mathbb{Q}(\boldsymbol{S} \mid \boldsymbol{O})} \\
&= \mathbb{E}_{\boldsymbol{S} \sim \mathbb{Q}}[\log \mathbb{P}(\boldsymbol{S}, \boldsymbol{O} \mid \boldsymbol{\theta})] + H(\mathbb{Q}) \\
&= F(\mathbb{Q}, \boldsymbol{\theta})
\end{aligned}
$$

General EM Algorithm

$$\max_{\mathbb{Q},\boldsymbol{\theta}} F(\mathbb{Q},\boldsymbol{\theta}) = \max_{\mathbb{Q},\boldsymbol{\theta}} \mathbb{E}_{\boldsymbol{S}\sim\mathbb{Q}}[\log \mathbb{P}(\boldsymbol{S},\boldsymbol{O} \mid \boldsymbol{\theta})] + H(\mathbb{Q})$$

- EM algorithm is coordinate-ascent on F
- It is therefore an iterative algorithm
- It is guaranteed to converge to a stationary point of F
- It alternates between the two following steps from initial parameter $\boldsymbol{\theta}_0$
  - Expectation step: $\mathbb{Q}_{\tau+1} = \arg\max_{\mathbb{Q}} F(\mathbb{Q},\boldsymbol{\theta}_\tau)$
  - Maximization step $\boldsymbol{\theta}_{\tau+1} = \arg\max_{\boldsymbol{\theta}} F(\mathbb{Q}_{\tau+1},\boldsymbol{\theta})$



from Wikipedia

EM Algorithm Applied to HMMs

- Expectation Step: $\mathbb{Q}_{\tau+1} = \arg\max_{\mathbb{Q}} F(\mathbb{Q}, \boldsymbol{\theta}_\tau) = \mathbb{P}(\boldsymbol{S} \mid \boldsymbol{O}, \boldsymbol{\theta}_\tau)$

$$\begin{aligned} F(\mathbb{P}(\boldsymbol{S} \mid \boldsymbol{O}, \boldsymbol{\theta}_\tau), \boldsymbol{\theta}_\tau) &= \sum_{\boldsymbol{s}} \mathbb{P}(\boldsymbol{S} \mid \boldsymbol{O}, \boldsymbol{\theta}_\tau) \log \frac{\mathbb{P}(\boldsymbol{S}, \boldsymbol{O} \mid \boldsymbol{\theta}_\tau)}{\mathbb{P}(\boldsymbol{S} \mid \boldsymbol{O}, \boldsymbol{\theta}_\tau)} \\ &= \sum_{\boldsymbol{s}} \mathbb{P}(\boldsymbol{S} \mid \boldsymbol{O}, \boldsymbol{\theta}_\tau) \log \mathbb{P}(\boldsymbol{O} \mid \boldsymbol{\theta}_\tau) \\ &= \log \mathbb{P}(\boldsymbol{O} \mid \boldsymbol{\theta}_\tau) \geq F(\mathbb{Q}, \boldsymbol{\theta}_\tau) \end{aligned}$$

- In HMMs, this is equivalent to defining (using $\mathcal{D}$):

$$\begin{aligned} \gamma_t^n(s) &= \mathbb{P}(S_t = s \mid \mathbf{0} = \boldsymbol{o}^n, \boldsymbol{\theta}_\tau) \\ \xi_t^n(s, s') &= \mathbb{P}(S_{t-1} = s, S_t = s' \mid \mathbf{0} = \boldsymbol{o}^n, \boldsymbol{\theta}_\tau) \end{aligned}$$

- They can be computed by the forward-backward algorithm!

EM Algorithm Applied to HMMs

- Maximization step

$$
\begin{aligned}
\boldsymbol{\theta}_{\tau+1} &= \arg\max_{\boldsymbol{\theta}} F(\mathbb{Q}_{\tau+1}, \boldsymbol{\theta}) \\
&= \arg\max_{\boldsymbol{\theta}} \mathbb{E}_{\boldsymbol{S}\sim\mathbb{Q}_{\tau+1}}[\log \mathbb{P}(\boldsymbol{S}, \boldsymbol{O} \mid \boldsymbol{\theta})] \\
&= \arg\max_{\boldsymbol{\theta}} \mathbb{E}_{\boldsymbol{S}\sim\mathbb{Q}_{\tau+1}}[\log \left(\mathbb{P}(S_0, \boldsymbol{\theta}) \prod_{t=1}^{T} \mathbb{P}(S_t \mid S_{t-1}, \boldsymbol{\theta}) \mathbb{P}(o_t \mid S_t, \boldsymbol{\theta})\right)] \\
&= \arg\max_{\boldsymbol{\theta}} \mathbb{E}_{\boldsymbol{S}\sim\mathbb{Q}_{\tau+1}}[\log \mathbb{P}(S_0, \boldsymbol{\theta}) + \sum_{t=1}^{T} \left(\log \mathbb{P}(S_t \mid S_{t-1}, \boldsymbol{\theta}) + \log \mathbb{P}(o_t \mid S_t, \boldsymbol{\theta})\right)]
\end{aligned}
$$

- This finally amounts to computing (using $\mathcal{D}$):

$$
\pi_s = \frac{\sum_{n=1}^{N} \gamma_0^n(s)}{N} \quad p_{ss'} = \frac{\sum_{n=1}^{N} \sum_{t=1}^{T} \xi_t^n(s, s')}{\sum_{n=1}^{N} \sum_{t=0}^{T-1} \gamma_t^n(s)} \quad q_{so} = \frac{\sum_{n=1}^{N} \sum_{t=0}^{T} \gamma_t^n(s)[o_t^n = o]}{\sum_{n=1}^{N} \sum_{t=0}^{T} \gamma_t^n(s)}
$$