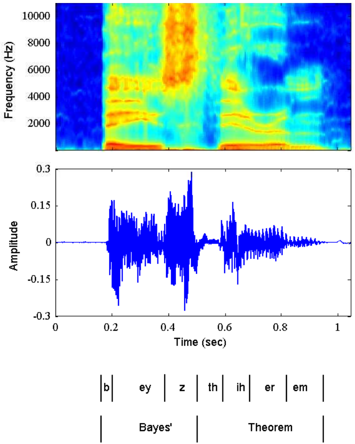
STAT 339
Hidden Markov Models II

May 6, 2020

EM for Estimating Model Parameters

Outline

HMM Application: Segmentation in Speech Recognition



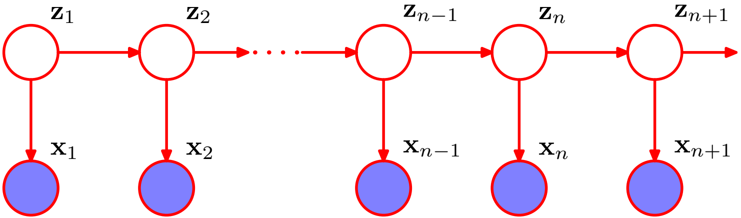
- Data, $x_n, n = 1, \ldots, N$
- Inference goal, latent state, z_n at each n
- Assume z_n is a discrete thing: a categorical “label” for the state.

A Generative Model

We can construct a generative model of the joint distribution of the $\mathbf{z}$ and the $\mathbf{x}$

$$p(\mathbf{z}, \mathbf{x}) = \prod_{n=1}^{N} p(z_n \mid z_{n-1}) p(x_n \mid z_n)$$

This corresponds to the graphical model below



Another Example: Activity Recognition From Pose[1]

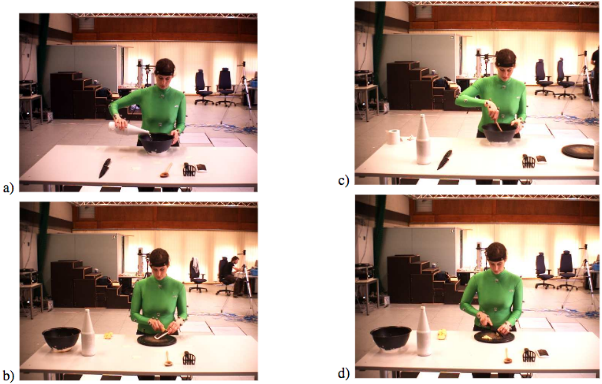


Figure: Display of five complex human motion sequences in a kitchen scenario: (top left) pour water into a bowl, (bottom-left) grate an apple, (top right) stir, (bottom right) cut fruit

[1] Gehrig et al. (2009). HMM-based Human Motion Recognition with Optical Flow Data, *9th International Conference on Humanoid Robots*

Another Example: Activity Recognition From Pose[1]

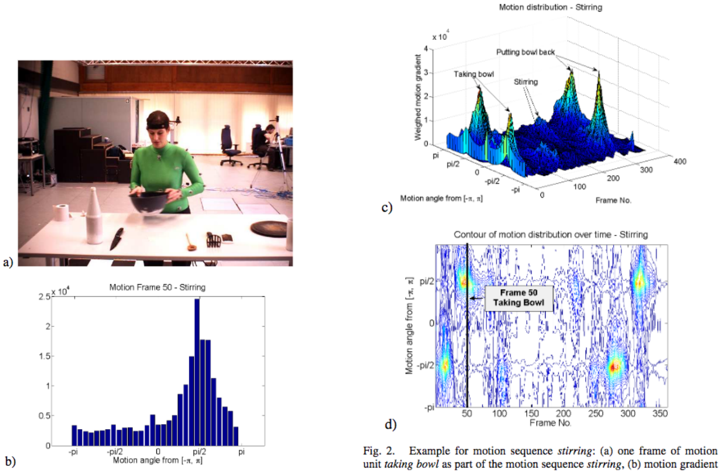


Fig. 2. Example for motion sequence *stirring*: (a) one frame of motion unit *taking bowl* as part of the motion sequence *stirring*, (b) motion gradient histogram for that particular frame, (c) distribution of optical flow gradients for the complete motion sequence, (d) motion distribution over time with two peaks indicating the beginning and the end of the motion sequence *stirring*.

Figure: Top left: one frame of raw video; Bottom left: Distribution of sensed motion angles for one frame; Right: motion angles during the entire "stirring" activity

Outline

Inference in HMMs

Given full specification of the component distributions (transition and emission probabilities), we might want to

1. **Find the marginal distribution** of a particular state $p(z_{n'})$ or observation $p(x_{n'})$ (e.g., predict the future or recover the past)
2. Find the **most likely hidden sequence** given data: $\mathrm{argmax}_{\mathbf{z}}\, p(\mathbf{z} \mid \mathbf{x})$
3. Get **samples** from $p(\mathbf{z} \mid \mathbf{x})$
4. Evaluate **marginal likelihood** $p(\mathbf{x})$ of some data (e.g., for model comparison).

Learning HMMs

If we don't know the transition and emission probabilities, we might want to

1. **Find MLE** transition matrix and emission parameters, marginalizing out the hidden sequence:

$$\hat{\mathbf{A}}, \hat{\boldsymbol{\theta}} = \underset{\mathbf{A},\boldsymbol{\theta}}{\operatorname{argmax}}\, p(\mathbf{x} \mid \mathbf{A}, \boldsymbol{\theta}) = \underset{\mathbf{A},\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{z_1} \cdots \sum_{z_N} p(\mathbf{x}, \mathbf{z} \mid \mathbf{A}, \boldsymbol{\theta})$$

$$= \underset{\mathbf{A},\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{z_1} \cdots \sum_{z_N} \prod_{n=1}^{N} p(z_n \mid z_{n-1}, \mathbf{A}) p(x_n \mid z_n, \boldsymbol{\theta})$$

   where $\mathbf{A}$ is the $K \times K$ matrix with entries

$$a_{k,k'} := p(z_n = k' \mid z_{n-1} = k)$$

   and $\boldsymbol{\theta}$ is a set of parameters of the "emission distributions" for each state.

2. Do some **model averaging** using a posterior distribution over $\mathbf{A}$ and $\boldsymbol{\theta}$; e.g., by **getting samples**

$$\mathbf{A}^{(s)}, \boldsymbol{\theta}^{(s)} \sim p(\mathbf{A}, \boldsymbol{\theta} \mid \mathbf{x})$$

Outline

Estimating transition and emission models

- An HMM is defined by its transition model, and its emission model:

$$a_{k'k} := p(z_n = k \mid z_{n-1} = k')$$
$$p_k(\mathbf{x}_n) := p(\mathbf{x}_n \mid z_n = k)$$

- Can we learn these from data?

Outline

Maximum Likelihood Estimation

- We can parameterize the model using

$$p(z_n = k \mid z_{n-1} = k', \mathbf{A}) = a_{k'k}$$
$$p(\mathbf{x}_n \mid z_n = k, \boldsymbol{\theta}) = p_k(\mathbf{x}_n \mid \boldsymbol{\theta}_k)$$

- Then we have a likelihood function for $\mathbf{A}$ and $\boldsymbol{\pi}$ given $\mathbf{z}$ and data, $\mathbf{x}$

$$\begin{aligned} p(\mathbf{z}, \mathbf{x} \mid \mathbf{A}, \boldsymbol{\theta}) &= \prod_{n=1}^{N} p(z_n \mid z_{n-1}) p(\mathbf{x}_n \mid z_n) \\ &= \prod_{n=1}^{N} a_{z_{n-1} z_n} p_{z_n}(\mathbf{x}_n \mid \boldsymbol{\theta}_k) \\ &= \left(\prod_{k'=1}^{K} \prod_{k=1}^{K} a_{k'k}^{N_{k'k}} \right) \left(\prod_{k=1}^{K} \prod_{n : z_n = k} p_k(\mathbf{x}_n \mid \boldsymbol{\theta}_k) \right) \end{aligned}$$

where $N_{k'k}$ is the number of transitions from state k' to state k in $\mathbf{z}$

Max. Likelihood Estimation

- We have a likelihood function for $\boldsymbol{\theta}$ and $\mathbf{A}$ given $\mathbf{z}$ and data, $\mathbf{x}$

$$p(\mathbf{z}, \mathbf{x} \mid \mathbf{A}, \boldsymbol{\theta}) = \left(\prod_{k'=1}^{K} \prod_{k=1}^{K} a_{k'k}^{N_{k'k}} \right) \left(\prod_{k=1}^{K} \prod_{n:z_n=k} p_k(\mathbf{x}_n \mid \boldsymbol{\theta}_k) \right)$$

  where $N_{k'k}$ is the number of transitions from state k' to state k in $\mathbf{z}$
- Factorizes into a piece with only $\mathbf{A}$, and pieces with only one $\boldsymbol{\theta}_k$ each!
- Except...
- This assumes we have $\mathbf{z}$, which... we don't.

EM Returns!

- Fortunately, if we have a current guess about $\mathbf{A}$ and $\boldsymbol{\theta}$, then we can compute

$$p(z_n = k \mid \mathbf{A}, \boldsymbol{\theta}, \mathbf{x}_{1:N}) \text{ for each } k$$

- Why is this useful? (Hint: Have we done something like this before?)
- Quantum states to the rescue!
- Simply assign each data point to *every* state, with weight

$$q_{nk} := p(z_n = k \mid \mathbf{A}, \boldsymbol{\theta}, \mathbf{x}_{1:N})$$

- **We can compute these with the forward-backward algorithm.**

Estimating $\boldsymbol{\theta}$

- Now the "quantum" likelihood becomes

$$L(\mathbf{A}, \boldsymbol{\theta}; \mathbf{Q}) = \left(\prod_{k'=1}^{K} \prod_{k=1}^{K} a_{k'k}^{N_{k'k}} \right) \left(\prod_{k=1}^{K} \prod_{n=1}^{N} p_k(\mathbf{x}_n \mid \boldsymbol{\theta}_k)^{q_{nk}} \right)$$

  where $N_{kk'}$ is the number of transitions from state k' to state k' in $\mathbf{z}$
- The $\boldsymbol{\theta}_k$ are now separated into factors that can be maximized separately...
- ... But we still don't know what to use for $N_{kk'}$...

Quantum transitions

- **Idea:** Each transition from z_{n-1} to z_n is split into quantum transitions, one for **each combination** of k and k'
- We want these weights to be

$$q_{nk'k} := p(z_{n-1} = k', z_n = k \mid \mathbf{A}, \boldsymbol{\theta}, \mathbf{x}_{1:N})$$

- How to calculate this?

Calculating quantum transition weights

We have defined

$$q_{nk'k} := p(z_{n-1} = k', z_n = k \mid \mathbf{A}, \boldsymbol{\theta}, \mathbf{x}_{1:N})$$

By definition of conditional probability, this is

$$q_{nk'k} = \frac{p(z_{n-1} = k', z_n = k, \mathbf{x}_{1:N} \mid \mathbf{A}, \boldsymbol{\theta})}{p(\mathbf{x}_{1:N} \mid \mathbf{A}, \boldsymbol{\theta})}$$

The numerator factors, using conditional independence (and omitting dependence on $\mathbf{A}$ and $\boldsymbol{\theta}$ in the notation just for clutter-reduction)

$$q_{nk'k} = \frac{p(z_{n-1} = k', \mathbf{x}_{1:n-1})p(z_n = k \mid z_{n-1} = k')p(x_n \mid z_n = k)p(\mathbf{x}_{n+1:N} \mid z_n = k)}{p(\mathbf{x}_{1:N})}$$

This is now written in terms of parameters and “message” vectors:

$$q_{nk'k} = \frac{m_{n-1,k'} a_{k'k} b^*_{nk} r_{nk}}{\mathbf{m}_N^\top \mathbf{1}}$$

Quantum transitions

- We now have a "quantum" weight for each pairwise combination of hidden states

$$q_{nk'k} = \frac{m_{n-1,k'} a_{k'k} b^*_{nk} r_{nk}}{\mathbf{m}_N^\top \mathbf{1}}$$

- Then, the "quantum" counts are all in terms of these:

$$q_{nk} = \sum_{k'=1}^{K} q_{nk'k} \qquad \tilde{N}_{k'k} = \sum_{n=1}^{N} q_{nk'k}$$

- So the complete "quantum" likelihood becomes

$$L(\mathbf{A}, \boldsymbol{\theta}; \mathbf{Q}) = \left(\prod_{k'=1}^{K} \prod_{k=1}^{K} a_{k'k}^{\tilde{N}_{k'k}} \right) \left(\prod_{k=1}^{K} \prod_{n=1}^{N} p_k(\mathbf{x}_n \mid \boldsymbol{\theta}_k)^{q_{nk}} \right)$$

which we can now maximize as in the mixture model.

EM for HMMs

The E-step involves forward-backward, and gives us "quantum" transition counts:

$$q_{nk'k} = \frac{m_{n-1,k'} a_{k'k} b^*_{nk} r_{nk}}{\mathbf{m}_N^\mathsf{T} \mathbf{1}}$$

The complete "quantum" likelihood becomes

$$L(\mathbf{A}, \boldsymbol{\theta}; \mathbf{Q}) = \left(\prod_{k'=1}^{K} \prod_{k=1}^{K} a_{k'k}^{\tilde{N}_{k'k}} \right) \left(\prod_{k=1}^{K} \prod_{n=1}^{N} p_k(\mathbf{x}_n \mid \boldsymbol{\theta}_k)^{q_{nk}} \right)$$

where

$$q_{nk} = \sum_{k'=1}^{K} q_{nk'k} \qquad \tilde{N}_{k'k} = \sum_{n=1}^{N} q_{nk'k} \qquad \tilde{N}_k = \sum_{k'=1}^{K} \tilde{N}_{k'k}$$

The M-step involves choosing $\hat{\mathbf{A}}$ and $\hat{\boldsymbol{\theta}}$ to maximize (the log of) this quantum likelihood subject to constraints:

$$\hat{a}_{k'k} = \tilde{N}_{k'k} / \tilde{N}_{k'} \qquad \hat{\boldsymbol{\theta}}_k = \underset{\boldsymbol{\theta}_k}{\operatorname{argmax}} \sum_{n=1}^{N} q_{nk} \log(p_k(\mathbf{x}_n \mid \boldsymbol{\theta}_k))$$

Outline

Summary of EM for HMMs We have developed the EM algorithm to do MLE of the HMM transition and emission parameters.

1. **E-step:** Execute forward-backward to compute the forward and backward messages, $\mathbf{m}_1, \ldots, \mathbf{m}_N$ and $\mathbf{r}_N, \ldots, \mathbf{r}_1$, and use them to compute "quantum" transition weights

$$q_{nk'k}^{(s)} := \frac{m_{n-1,k'}^{(s)} \hat{a}_{k'k}^{(s-1)} \hat{b}_{nk}^{*(s-1)} r_{nk}^{(s)}}{\mathbf{m}_N^{(s)\top} \mathbf{1}}$$

$$q_{nk}^{(s)} = \sum_{k'=1}^{K} q_{nk'k}^{(s)} \qquad \tilde{N}_{k'k}^{(s)} = \sum_{n=1}^{N} q_{nk'k}^{(s)} \qquad \tilde{N}_{k}^{(s)} = \sum_{k'=1}^{K} \tilde{N}_{k'k}^{(s)}$$

2. **M-step:** Maximize the "quantum" likelihood w.r.t $\mathbf{A}, \boldsymbol{\theta}$ to get

$$\hat{a}_{k'k}^{(s)} = \tilde{N}_{k'k}^{(s)} / \tilde{N}_{k'}^{(s)} \qquad \hat{\boldsymbol{\theta}}_k^{(s)} = \underset{\boldsymbol{\theta}_k}{\operatorname{argmax}} \sum_{n=1}^{N} q_{nk}^{(s)} \log(p_k(\mathbf{x}_n \mid \boldsymbol{\theta}_k))$$

where $\hat{b}_{nk}^{*(s)} = p_k(x_n \mid \hat{\boldsymbol{\theta}}_k^{(s)})$