Bioinformatics

Introduction to Hidden Markov Models

Hidden Markov Models and Multiple Sequence Alignment

Slides borrowed from

Scott C. Schmidler

(MIS graduated student)

Outline

- Probability Review
- Markov Chains
- Hidden Markov Chains
- Examples in HMMs for Protein Sequence
- Algorithm Review for HMMs

Motivation: Composing a Drama by Mimicking Shakespeare

- Assume we want to write a drama of Shakespeare style
- We collect a large set of Shakespeare's works
- Define a vocabulary V = $\{X_1, X_2, ..., X_N\}$
- Build a model $P(X_i|X_j)$ for $i, j = 1, ..., N$
- To compose a drama, generate words from the model $P(X_i|X_j)$

- Though this is too simplistic to be useful, this naive model can be extended and refined to mimic the writing style of Shakespears'

Markov Approximations to English

- From Shannon's original paper:
  1. *Zero-order approximation*:

     XFOML RXKXRJFFUJ ZLPWCFWKCYJ FFJEYVKCQSGHYD QPAAMKBZAACIBZLHJQD

  2. *First-order approximation*:

     OCRO HLI RGWR NWIELWIS EU LL NBNESEBYA TH EEI ALHENHTTPA OOBTTVA NAH RBL

  3. *Second-order approximation*:

     ON IE ANTSOUTINYS ARE T INCTORE ST BE S DEAMY ACHIN D ILONASIVE TUCOOWE AT TEASONARE FUSO TIZIN ANDY TOBE SEACE CITSBE

Markov Approximations (cont.)

From Shannon's paper

4. *Third-order approximation*:

   IN NO IST LAT WHEY CRATICT FROURE BIRS GROCID PONDENOME OF DEMONSTURES OF THE REPTABIN IS REGOACTIONA OF CRE

Markov random field with 1000 "features," no underlying "machine" (Della Pietra et. Al, 1997):

WAS REASER IN THERE TO WILL WAS BY HOMES THING BE RELOVERATED THER WHICH CONISTS AT RORES ANDITING WITH PROVERAL THE CHESTRAING FOR HAVE TO INTRALLY OF QUT DIVERAL THIS OFFECT INATEVER THIFER CONSTRANDED STATER VILL MENTTERING AND OF IN VERATE OF TO

Word-Based Approximations

1. *First-order approximation*:

REPRESENTING AND SPEEDILY IS AN GOOD APT OR COME CAN DIFFERENT NATURAL HERE HE THE A IN CAME THE TO OF TO EXPERT GRAY COME TO FURNISHES THE LINE MESSAGE HAD BE T

2. *Second-order approximation*:

THE HEAD AND IN FRONTAL ATTACK ON AN ENGLISH WRITER THAT THE CHARACTER OF THIS POINT IS THEREFORE ANOTHER METHOD FOR THE LETTERS THAT THE TIME OF WHO EVER TOLD THE PROBLEM FOR AN UNEXPETED

Shannon's comment:

"It would be interesting if further approximations could be constructed, but the labor involved becomes enormous at the next stage."

Motivation: Composing a Symphony of Beethoven Style

- We want to compose a symphony of Beethoven style
- We collect a large set of Beethoven's works
- Define a vocabulary $V = \{X_1, X_2, ..., X_N\}$ of musical notes
- Build a model $P(X_i|X_j)$ for $i, j = 1, ..., N$
- To compose a symphony, generate note symbols from the model $P(X_i|X_j)$

Modeling Biological Sequences

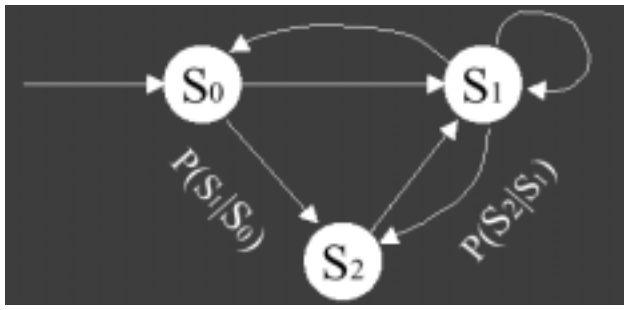
- Collect a set of sequences of interest
- Define a vocabulary V = {X_1, X_2, ..., X_N}
  - For DNA sequences: N = 4 and V = {A, T, G, C}
  - For protein sequences: N = 20 and V = {amino acids}
- Build (learn) a model $P(X_i|X_j)$ for i, j = 1, ..., N or in more general $P(X|w)$ with X = X_1, X_2, ..., X_M and model parameter vector w
- The model can be used to
  - To generate typical sequences from the class of training sequences, e.g. protein family
  - To compute the probability of an observed sequence O being generated from the model class
  - and others
- Hidden Markov models (HMMs) are a class of stochastic generative models effective for building such probabilistic models.

Probability Review

- Probability notation:
  - Probability: $P(A) \qquad P(A) \geq 0, \sum_A P(A) = 1$
  - Joint probability: $P(A, B)$
  - Conditional probability: $P(A \mid B) = P(A, B) / P(B)$
  - Marginal probability: $P(A) = \sum_B P(A, B)$
  - Independence: $P(A, B) = P(A)P(B)$
  - Bayes'rule: $P(B \mid A) = P(A \mid B)P(B) / P(A)$

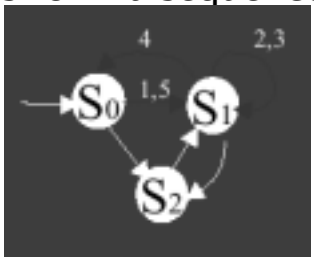
Markov Chains

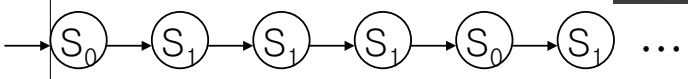
- Markov property: $P(X_0, X_1, \ldots, X_t) = P(X_0)P(X_1 \mid X_0)\ldots P(X_t \mid X_{t-1})$
- Formally:
  - State space $\mathbf{S} = \{S_1, \ldots, S_N\}$
  - Transition matrix $\mathbf{P} = P(X_t = S_i \mid X_{t-1} = S_j)\ 1 \le i, j \le N$
  - Initial distribution $\pi = P(X_0 = S_i),\ 1 \le i \le N$
- CS intuition
  - Stochastic finite automaton



Markovian Sequence

- States through which the chain passes form a sequences:
  Example: $S_0, S_1, S_1, S_1, S_0, S_1, \ldots$
- Graphically:
- By the Markov property:

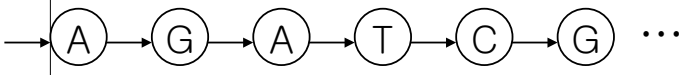




$$P(Sequence) = P(S_0, S_1, S_1, S_1, S_0, S_1, \ldots)$$
$$= \pi(S_0)P(S_1 \mid S_0)P(S_1 \mid S_1)$$

Example

- Markov chain for generating a DNA sequence:
- Sequence probability:



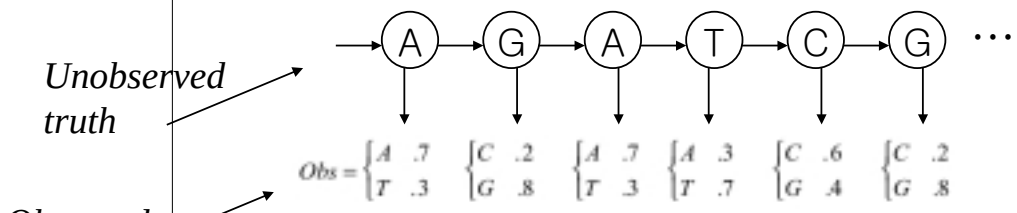
$$P(AGATCG) = \pi(A)P(G \mid A)P(A \mid G)P(T \mid A)\,\mathsf{K}$$

Dinucleotide frequency (e.g. base-stacking)

Hidden Markov Chains

- Observed sequence is a *probabilistic function* of underlying Markov chain
  - Example: HMM for a (noisy) DNA sequence (see e.g. Churchill 1989)

True state sequence unknown, but observation sequence gives us a clue

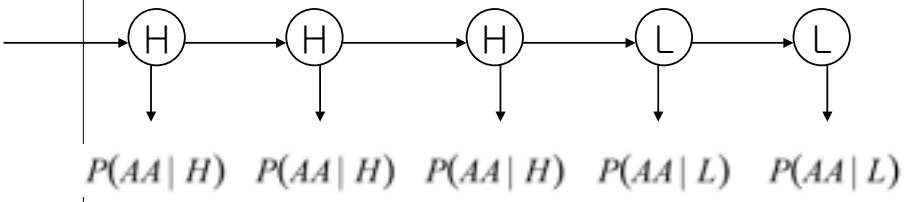


Unobserved truth

Observed noisy sequence data

Example: Hidden Markov Chain for Protein Sequence

- State space is backbone secondary structure
  - Used for prediction (Asai *et. al.*, Stultz *et. al.*)



- State space is side chain environment
  - Used for fold-recognition (Hubbard *et. al.*)

A HMM for Multiple Protein Sequences (Krogh *et. al.*)

- "Match" states are model (consensus) positions
- Position-specific deletion penalties
- Position-specific insertion frequencies
- Path through states aligns sequence to model

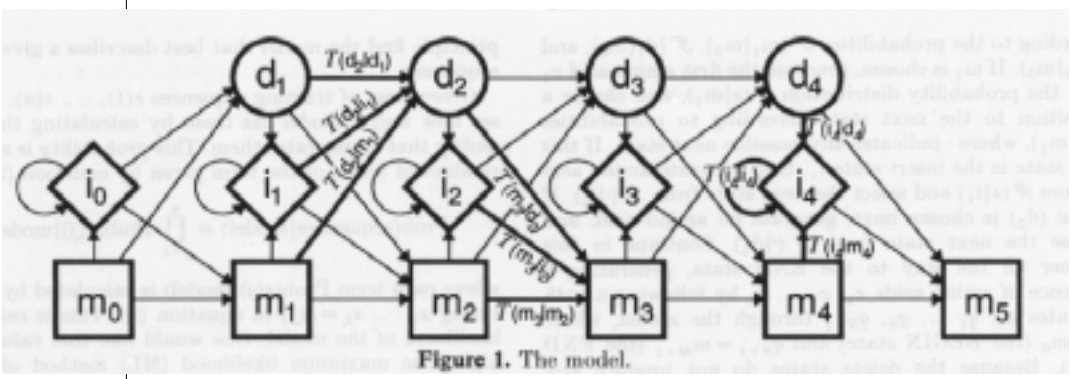


Figure 1. The model.

Figure from (Krogh *et. al.* 1994)

Example: Multiple Alignment of Globin Sequences

Figure from (Krogh *et. al.* 1994)

HMM-based Multiple Sequence Alignment

- Multiple alignment of k sequences is $O(n^k)$, so instead:
  1. Estimate a statistical model for the sequences
     - Use head start PROFILE alignment
     - Start from scratch with unaligned sequences (harder)
  2. Align each remaining sequence to the model
  3. Alignment yields assignments of equivalent sequence elements within the multiple alignment

Example: Aligning Sequence to Model

- Given an HMM model for a protein family:
  Align a new sequence to the model
  (*d* states are gaps, *i* states are insertions)

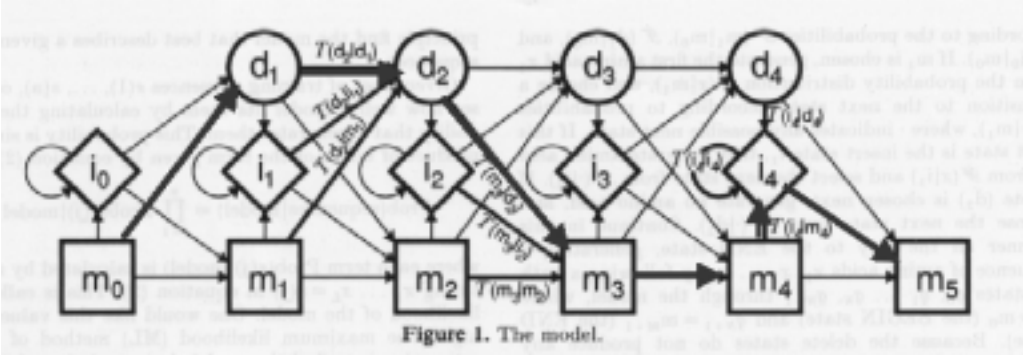


Figure 1. The model.

Computing with HMMs

- Three tasks:

1. Probability of an observed sequence
   Given $O_1, O_2, \ldots, O_r$ find $P(O_1, O_2 \mathrm{K}, O_r)$
   (nontrivial since state sequence unobserved)
2. Most likely hidden state sequence
   Given $O_1, O_2, \ldots, O_r$ compute $\arg\max_{S_1, \mathrm{K}, S_r} P(S_1, \mathrm{K}, S_r \mid O_1, \mathrm{K}, O_r)$
2. Most likely hidden state sequence
   Given observed sequence $\{O^1, \ldots, O^n\}$ find

$$\arg\max_{\theta} P(O^1, \mathrm{K}, O^n \mid \theta)$$

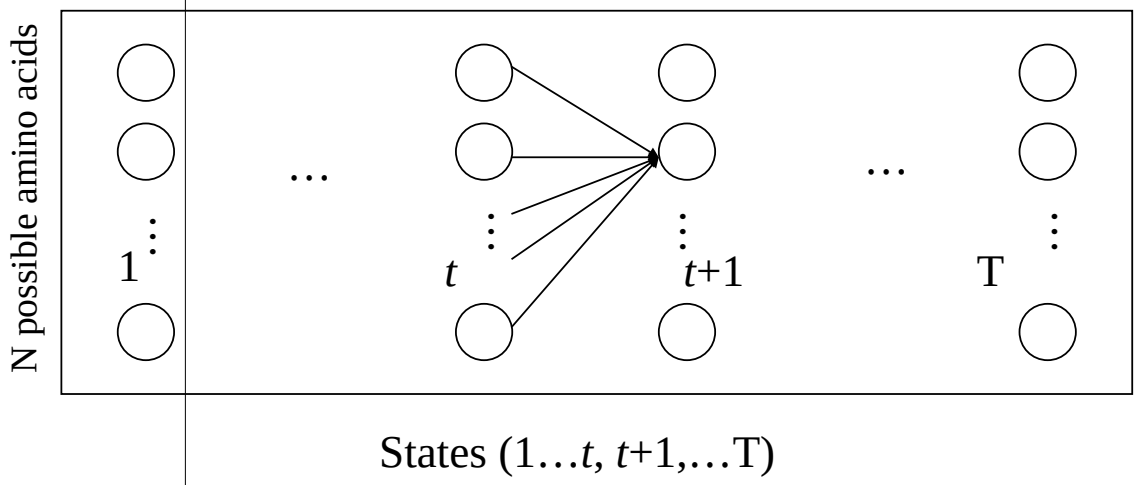
Computing Likelihood of Observed Sequence

- Compute $P(O_1, O_2, \mathsf{K}, O_r)$
  - True state sequence unknown
  - Must sum over all possible paths
  - Number paths $O(T^N)$
  - Markovian structure permits:
    - Recursive definition and hence
    - Efficient calculation by dynamic programming

$$P(O_1, \mathsf{K}, O_r) = \sum_{S_0, S_1, \mathsf{K}, S_r} P(O_1, O_2, \mathsf{K}, O_r \mid S_0, S_1, \mathsf{K}, S_r) P(S_0, S_1, \mathsf{K}, S_r)$$

- Key observation: Any path must be in exactly one state at time *t*

Key Idea for HMM Computations

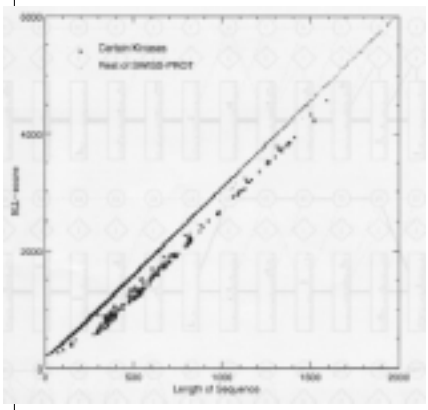


States (1…*t*, *t*+1,…T)

Example: Searching Protein Database with HMM Profile

- For each sequence in database:
- Does sequence "fit" model?
- Score by $P(O_1, O_2, \ldots, O_r)$, compute Z-score adjusted for length

Protein Kinases:



Globins:

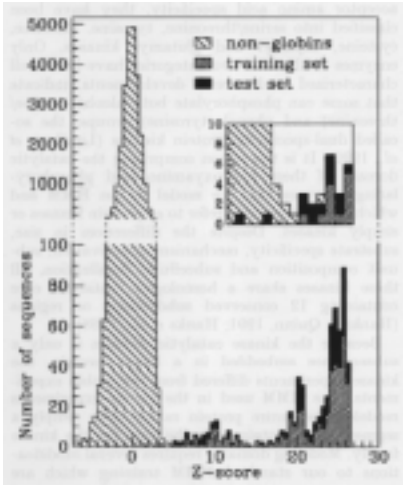


Figure from (Krogh *et. al.* 1994)

Estimate Alignment and Model Parameters - *Simultaneously*

- Key idea – missing data:
  - *What if we know the alignment*?
    - Parameters easy to estimate:
      - Calculate (expected) number of transitions
      - Calculate (expected) frequency of amino acids
  - *What if we knew the parameters*?
    - Alignment easy to find
      - Align each sequence to model using Viterbi algorithm
      - Align residues in match states

Other “details”

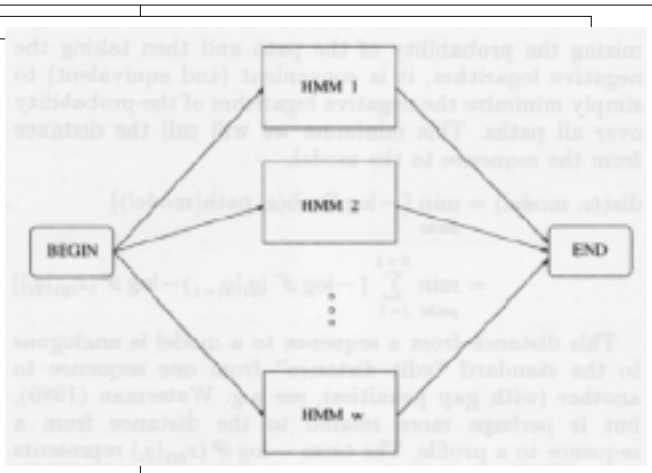
- How many states in model?
- How to initialize parameters?
- How to avoid local models?

  See (Krong *et. al*., 1994) for some suggestion

Multiple Protein Sequence Alignment

- Give a set of sequences:
  - Estimate HMM model using optimization for parameter search (Baum-Welch, EM)
  - Align each sequence to model (Viterbi)
  - Match states of model provide columns of resulting multiple alignment

Extensions



Clustering subfamilies

Modeling domains

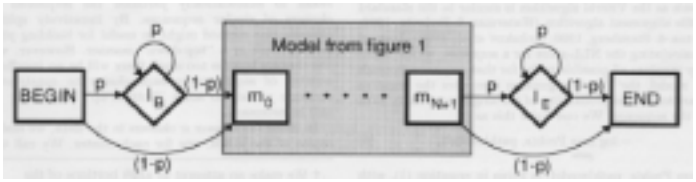


Figure from (Krogh *et. al*. 1994)

Tradeoffs

- Advantages:
  - Explicit probabilistic model for family
  - Position specific residue distributions, gap penalties, insertions frequencies
- Disadvantages:
  - Many parameters, requires more data of care
  - Traded one hard optimization problem for another

HMM Summary

- Powerful tool for modeling protein families
- Generalization of existing profile methods
- Data-intensive
- Widely applicable to problems in bioinformatics

References

- Bioinformatics Classic: Krogh *et. al*. (1994)

  Hidden Markov models in computational biology: applications to protein modeling, *J. Mol. Biol*. 235: 1501-1531

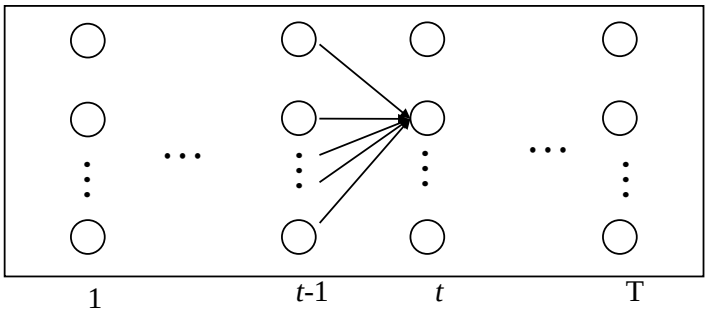
- Book: Eddy & Durbin, 1999. See web site.

- Tutorial: Rabiner, L. (1989) A tutorial on hidden Markov models and selected applications in speech recognition, *Proc IEEE*, 77(**2**), 257-286

Forward-backward Algorithm

- Forward pass:
  - Define $\alpha_t(j) = \left[\sum_{i=1}^{N} \alpha_{t-1}(i) P(S_j \mid S_i)\right] P(O_i \mid S_j)$
  - Prob. Of subsequence $O_1, O_2, \ldots, O_t$ when in S_j at t

Key obs: any path must be in 1 of N states at t



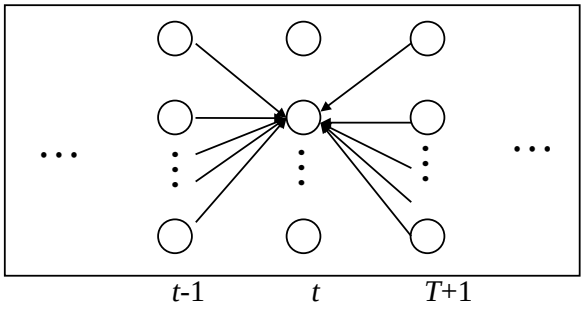
Forward-backward Algorithm

- Notice $P(O_1, O_2, \mathrm{K}, O_r) = \sum_{1}^{N} \alpha_t(j)$
- Define an analogues backward pass so that:

$$\beta_t(j) = \sum_{i=1}^{N} \beta_{t-1}(i) P(S_i \mid S_j) P(O_{t+1} \mid S_i)$$

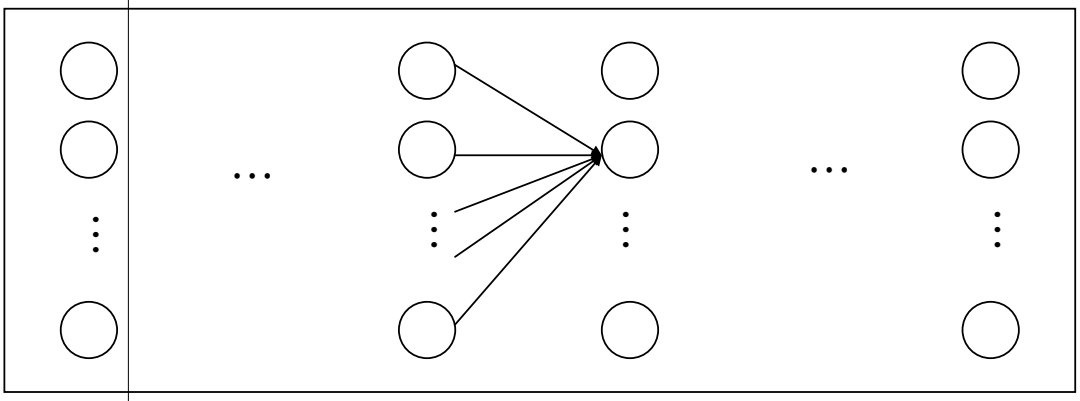
and

$$P(O_t \text{ came from } S_j) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i)\beta_t(i)}$$



Finding Most Likely Path

- Forward pass:
  - Replace summation with maximization
  - Max prob. of subseq. O_1, O_2, …, O_r When in S_j at t
  - Again: $\max P(O_1, O_2, \mathsf{K}, O_r) = \max_{1 \le j \le N} \alpha_T(j)$, then trace back



Baum-Welch Algorithm (Expectation-Maximization)

- Set parameters to expected values given observed sequences:
  - State transition probs:
  - Observation probs:

$$P(S_j \mid S_i) = \frac{\sum_{t=1}^{T-1} P(in\ S_i\ at\ t, in\ S_j\ at\ t+1 \mid O)}{\sum_{t=1}^{T-1} P(in\ S_i\ at\ t \mid O)}$$

$$P(obs \mid S_i) = \frac{\sum_{t=1}^{T-1} P(in\ S_i\ at\ t \mid O) * 1(O_t = obs)}{\sum_{t=1}^{T-1} P(in\ S_i\ at\ t \mid O)}$$

  - Recalculate expectations with new probabilities
  - Iterate to convergence
    - Guaranteed $P(O^1, \mathsf{K}, O^n \mid \theta)$ strictly increasing, converge to local mode (See Rabiner, 1989 for details)