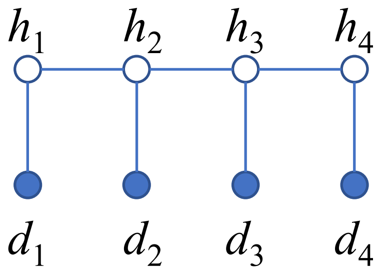
EM $P(\mathbf{d},\mathbf{h}\,|\,\boldsymbol{\lambda}) = \dfrac{e^{\boldsymbol{\lambda}\cdot\boldsymbol{\varphi}(\mathbf{d},\mathbf{h})}}{\sum_{\mathbf{d},\mathbf{h}} e^{\boldsymbol{\lambda}\cdot\boldsymbol{\varphi}(\mathbf{d},\mathbf{h})}}$

Example: Graph with no closed loops



$$\boldsymbol{\varphi}(\mathbf{d},\mathbf{h}) = \begin{Bmatrix} \varphi(d_i,h_i) : i=1,\ldots,N, \\ \psi(h_i,h_{i+1}) : i=1,\ldots,N-1 \end{Bmatrix} \quad \boldsymbol{\lambda} = \begin{Bmatrix} \lambda_i : i=1,\ldots,N, \\ \mu_i : i=1,\ldots,N-1 \end{Bmatrix}$$

$$\boldsymbol{\lambda}\cdot\boldsymbol{\varphi}(\mathbf{d},\mathbf{h}) = \sum_{i=1}^{N} \lambda_i \cdot \varphi(d_i,h_i) + \sum_{i=1}^{N-1} \mu_i \cdot \psi(h_i,h_{i+1})$$

$$Z[\boldsymbol{\lambda}] = \sum_{\mathbf{d},\mathbf{h}} e^{\boldsymbol{\lambda}\cdot\boldsymbol{\varphi}(\mathbf{d},\mathbf{h})}$$

Note In this special case: $Z[\lambda]$can be computed by Dynamic Programming because there are no closed loops.

Problem 1. Compute $P(\mathbf{d} \mid \boldsymbol{\lambda}) = \sum_{\mathbf{h}} P(\mathbf{d}, \mathbf{h} \mid \boldsymbol{\lambda})$

This can be computed by Dynamic Programming (sum rule)

If closed loops, approximated by Belief Propagation (sum-product)

Problem 2. Compute $\hat{\mathbf{h}} = \arg\max P(\mathbf{d}, \mathbf{h} \mid \boldsymbol{\lambda})$

Again, this can be computed by DP (max)

If closed loops, approximated by BP (max-product)

Problem 3. Learning λ

Data $D = \left\{ \mathbf{d}^m : m = 1, \ldots, M \right\}$ ⇒

$$\hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} \prod_{m=1}^{M} P(\mathbf{d}^m \mid \boldsymbol{\lambda})$$

$$= \arg\max_{\boldsymbol{\lambda}} \prod_{m=1}^{M} \sum_{\mathbf{h}^m} P(\mathbf{d}^m, \mathbf{h}^m \mid \boldsymbol{\lambda})$$

JOHNS HOPKINS UNIVERSITY

EM with Free Energy

Introduce distribution $Q_m(\mathbf{h}^m)$

$$F\left[\boldsymbol{\lambda}:\{Q_m(\mathbf{h}^m)\}\right]=\sum_{m=1}^{M}\left\{-\log P(\mathbf{d}^m\mid\boldsymbol{\lambda})+\sum_{\mathbf{h}^m}Q_m(\mathbf{h}^m)\log\frac{Q_m(\mathbf{h}^m)}{P(\mathbf{h}^m\mid\mathbf{d}^m,\boldsymbol{\lambda})}\right\}$$

Re-express this as:

$$F\left[\boldsymbol{\lambda}:\{Q_m(\mathbf{h}^m)\}\right]=\sum_{m=1}^{M}\left\{\sum_{\mathbf{h}^m}Q_m(\mathbf{h}^m)\log Q_m(\mathbf{h}^m)-\sum_{\mathbf{h}^m}Q_m(\mathbf{h}^m)\log P(\mathbf{h}^m,\mathbf{d}^m\mid\boldsymbol{\lambda})\right\}$$

The EM algorithm minimizes $F[\boldsymbol{\lambda}:\{Q_m(\cdot)\}]$ w.r.t. $\boldsymbol{\lambda}$ and the $\{Q_m(\cdot)\}$ alternatively

$$\boldsymbol{\lambda}^{t+1}=\arg\min_{\boldsymbol{\lambda}}F\left[\boldsymbol{\lambda}:\{Q_m^t(\cdot)\}\right]$$

$$Q_m^{t+1}(\cdot)=\arg\min_{Q_m}F\left[\boldsymbol{\lambda}^{t+1}:\{Q_m(\cdot)\}\right]$$

(B) $\boldsymbol{\lambda}^{t+1} = \arg\min_{\boldsymbol{\lambda}} \left\{ -\sum_{\mathbf{h}^m} Q_m^t(\mathbf{h}^m) \log P(\mathbf{h}^m, \mathbf{d}^m \mid \boldsymbol{\lambda}) \right\}$

(A) $Q_m^{t+1}(\mathbf{h}^m) = P(\mathbf{h}^m \mid \mathbf{d}^m, \boldsymbol{\lambda}^t)$

(A) How to compute these update rules if $P(\mathbf{h}^m, \mathbf{d}^m \mid \boldsymbol{\lambda}^t) = \dfrac{e^{\boldsymbol{\lambda}\cdot\boldsymbol{\varphi}(\mathbf{d},\mathbf{h})}}{Z[\boldsymbol{\lambda}]}$?

$$P(\mathbf{h}^m \mid \mathbf{d}^m, \boldsymbol{\lambda}^t) = \frac{P(\mathbf{h}^m, \mathbf{d}^m \mid \boldsymbol{\lambda}^t)}{P(\mathbf{d}^m \mid \boldsymbol{\lambda}^t)}$$

where $P(\mathbf{d}^m \mid \boldsymbol{\lambda}^t) = \dfrac{1}{Z[\boldsymbol{\lambda}]} \sum_{\mathbf{h}^m} e^{\boldsymbol{\lambda}\cdot\boldsymbol{\varphi}(\mathbf{d}^m,\mathbf{h}^m)}$

Hence, $P(\mathbf{h}^m \mid \mathbf{d}^m, \boldsymbol{\lambda}^t) = \dfrac{e^{\boldsymbol{\lambda}\cdot\boldsymbol{\varphi}(\mathbf{d}^m,\mathbf{h}^m)}}{\boxed{\sum_{\mathbf{h}^m} e^{\boldsymbol{\lambda}\cdot\boldsymbol{\varphi}(\mathbf{d}^m,\mathbf{h}^m)}}}$

This term can be directly computed by DP(sum) if the graph has no closed loop

Hence, $Q_m^{t+1}(\mathbf{h}^m) = P(\mathbf{h}^m \mid \mathbf{d}^m, \boldsymbol{\lambda}^t)$ can be computed (no closed loops)

(B) How to compute $\boldsymbol{\lambda}^{t+1} = \arg\min_{\boldsymbol{\lambda}} \left\{ -\sum_{\mathbf{h}^m} Q_m^t(\mathbf{h}^m) \log P(\mathbf{h}^m, \mathbf{d}^m \mid \boldsymbol{\lambda}) \right\}$

Substitute $P(\mathbf{h}, \mathbf{d} \mid \boldsymbol{\lambda}) = \dfrac{e^{\boldsymbol{\lambda}\cdot\boldsymbol{\varphi}(\mathbf{h},\mathbf{d})}}{Z[\boldsymbol{\lambda}]}$

Want to minimize: $G(\boldsymbol{\lambda}) = -\sum_{m=1}^{M} Q_m^t(\mathbf{h}^m)\cdot\boldsymbol{\lambda}\cdot\boldsymbol{\varphi}(\mathbf{h}^m, \mathbf{d}^m) - \sum_{m=1}^{M} \log Z[\boldsymbol{\lambda}]$

It can be shown that $G(\boldsymbol{\lambda})$ is a convex function of $\boldsymbol{\lambda}$ (because $\log Z[\boldsymbol{\lambda}]$ is convex)

The global minimum $\hat{\boldsymbol{\lambda}}$ occurs where

$$\frac{\partial}{\partial \boldsymbol{\lambda}} G(\hat{\boldsymbol{\lambda}}) = 0$$

when $\dfrac{1}{M}\sum_{m=1}^{M} Q_m^t(\mathbf{h}^m)\boldsymbol{\varphi}(\mathbf{h}^m, \mathbf{d}^m) = \sum_{\mathbf{h},\mathbf{d}} \boldsymbol{\varphi}(\mathbf{h}, \mathbf{d}) P(\mathbf{h}, \mathbf{d} \mid \boldsymbol{\lambda})$

i.e. when the expected statistics w.r.t. data $\mathbf{d}^m$ and $Q_m(\cdot)$
= the expected statistics of the model

Note Deriving the last equation as follows:

$$\frac{\partial}{\partial \boldsymbol{\lambda}} \log Z[\boldsymbol{\lambda}] = \frac{\partial}{\partial \boldsymbol{\lambda}} \log \sum_{\mathbf{h},\mathbf{d}} e^{\boldsymbol{\lambda}\cdot\boldsymbol{\varphi}(\mathbf{h},\mathbf{d})} = \frac{\sum_{\mathbf{h},\mathbf{d}} \varphi(\mathbf{h},\mathbf{d}) \cdot e^{\boldsymbol{\lambda}\cdot\boldsymbol{\varphi}(\mathbf{h},\mathbf{d})}}{\sum_{\mathbf{h},\mathbf{d}} e^{\boldsymbol{\lambda}\cdot\boldsymbol{\varphi}(\mathbf{h},\mathbf{d})}} = \sum_{\mathbf{h},\mathbf{d}} P(\mathbf{h},\mathbf{d} \mid \boldsymbol{\lambda}) \cdot \boldsymbol{\varphi}(\mathbf{h},\mathbf{d})$$

(recall learning notes for
learning exponential distributions)

Hence, the update rule for $\boldsymbol{\lambda}^{t+1}$ requires finding the value of $\boldsymbol{\lambda}$ so that the expected statistics (w.r.t. the data $\mathbf{d}^m$ & $Q_m(\cdot)$) are equal to the statistics of the model

Note This is a generalization of the result for Hidden Markov Models.