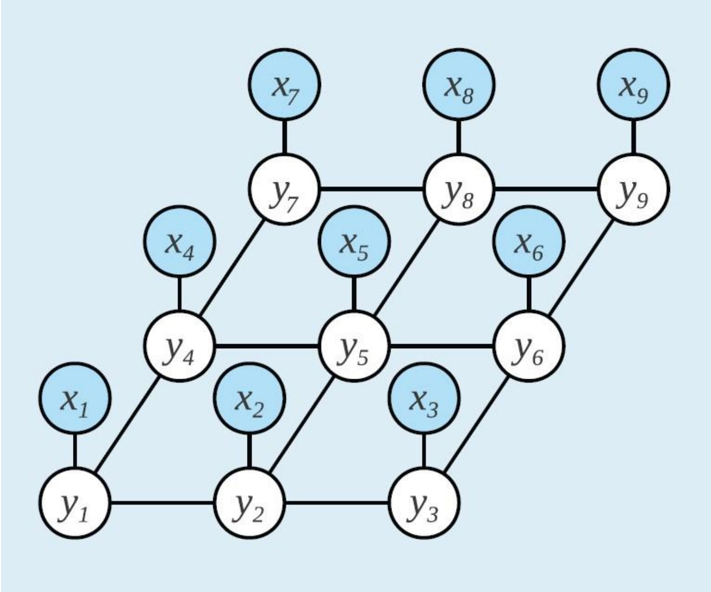
Probabilistic Graphical Models in Bioinformatics

Lecture 9: score-based structure learning; partially observed data



Hesam Montazeri
Department of Bioinformatics, IBB, University of Tehran

Ordibehesht 11, 1398
(May 1, 2019)

University of TEHRAN

Score-based structure learning

Learning general Bayesian networks

- **Theorem:** Finding the maximum-score network with at most $d \geq 2$ parents for each variable is *NP-hard*.

- We will use heuristic algorithm to search the space of graphs and return a high-scoring one.

- Local operators:
  - Edge addition
  - Edge deletion
  - Edge reversal

- Search techniques
  - Greedy hill climbing
  - Simulated annealing
  - …

Greedy hill climbing

University of TEHRAN

- The search procedure
  - Pick an initial network structure g
    - Empty network
    - A random choice
    - The best tree
  - At each iteration
    - We consider all legal networks in neighbors of g using local operators
    - Apply the change that leads to the best improvement
  - Stopping condition: when no modification improves the score
- Two issues:
  - Stucking in a local maxima
  - Reaching a *plateau* (a large set of neighboring networks that have the same score).

Score decomposition and search

University of TEHRAN

$$\text{score}(\mathcal{G} : \mathcal{D}) = \sum_i \text{FamScore}(X_i \mid \text{Pa}^{\mathcal{G}}_{X_i} : \mathcal{D}),$$

We define the delta score as:

$$\delta(\mathcal{G} : o) = \text{score}(o(\mathcal{G}) : \mathcal{D}) - \text{score}(\mathcal{G} : \mathcal{D})$$

Proposition 18.5 *Let $\mathcal{G}$ be a network structure and* score *be a decomposable score.*

- *If o is "Add $X \rightarrow Y$," and $X \rightarrow Y \notin \mathcal{G}$, then*

$$\delta(\mathcal{G} : o) = \text{FamScore}(Y, \text{Pa}^{\mathcal{G}}_{Y} \cup \{X\} : \mathcal{D}) - \text{FamScore}(Y, \text{Pa}^{\mathcal{G}}_{Y} : \mathcal{D}).$$

- *If o is "Delete $X \rightarrow Y$" and $X \rightarrow Y \in \mathcal{G}$, then*

$$\delta(\mathcal{G} : o) = \text{FamScore}(Y, \text{Pa}^{\mathcal{G}}_{Y} - \{X\} : \mathcal{D}) - \text{FamScore}(Y, \text{Pa}^{\mathcal{G}}_{Y} : \mathcal{D}).$$

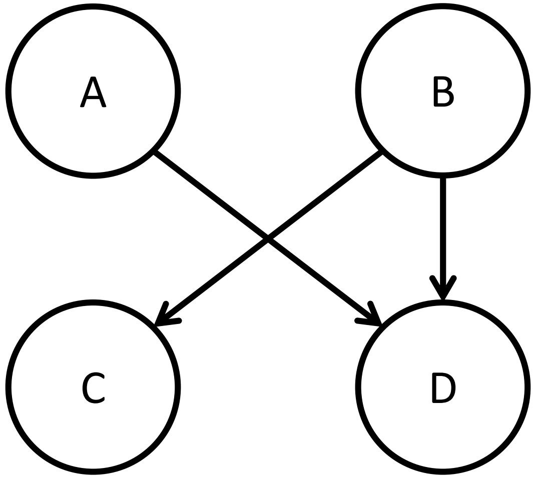
- *If o is "Reverse $X \rightarrow Y$" and $X \rightarrow Y \in \mathcal{G}$, then*

$$\begin{aligned} \delta(\mathcal{G} : o) \quad = \quad & \text{FamScore}(X, \text{Pa}^{\mathcal{G}}_{X} \cup \{Y\} : \mathcal{D}) + \text{FamScore}(Y, \text{Pa}^{\mathcal{G}}_{Y} - \{X\} : \mathcal{D}) \\ & - \text{FamScore}(X, \text{Pa}^{\mathcal{G}}_{X} : \mathcal{D}) - \text{FamScore}(Y, \text{Pa}^{\mathcal{G}}_{Y} : \mathcal{D}). \end{aligned}$$

Score decomposition and search-example

Question: compute the delta score for the following operations

1. Add $A \rightarrow B$
2. Add $C \rightarrow D$
3. Remove $A \rightarrow D$
4. Remove $B \rightarrow C$
5. Reverse $B \rightarrow C$
6. Reverse $B \rightarrow D$



Structure learning methods

- Constraint-based structure learning
- Score-based structure learning
- **Bayesian model averaging methods**

Bayesian model averaging

- Fully Bayesian inference
  - We consider the structure as a random variable

- Bayesian prediction

$$P(\xi[M+1] \mid \mathcal{D}) = \sum_{\mathcal{G}} P(\xi[M+1] \mid \mathcal{D}, \mathcal{G}) P(\mathcal{G} \mid \mathcal{D}),$$

where $$P(\mathcal{G} \mid \mathcal{D}) = \frac{P(\mathcal{G}) P(\mathcal{D} \mid \mathcal{G})}{P(\mathcal{D})}.$$

- How to approximate $P(g \mid D)$?
  - often by Markov chain Monte Carlo (MCMC) over structures

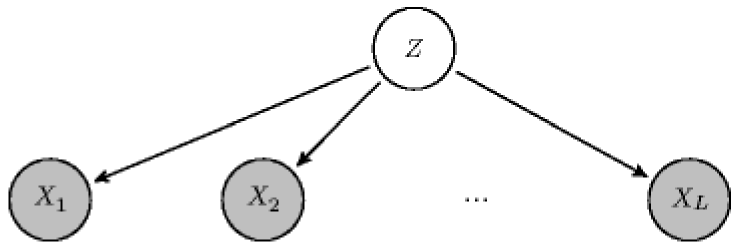
Partially observed data

	Known structure (Parameter estimation)	**Unknown structure (Structure learning)**
Fully observable	*MLE* *Bayesian methods*	*Constraint-based methods* *Score-based methods e.g. hill climbing*
Partially observable		

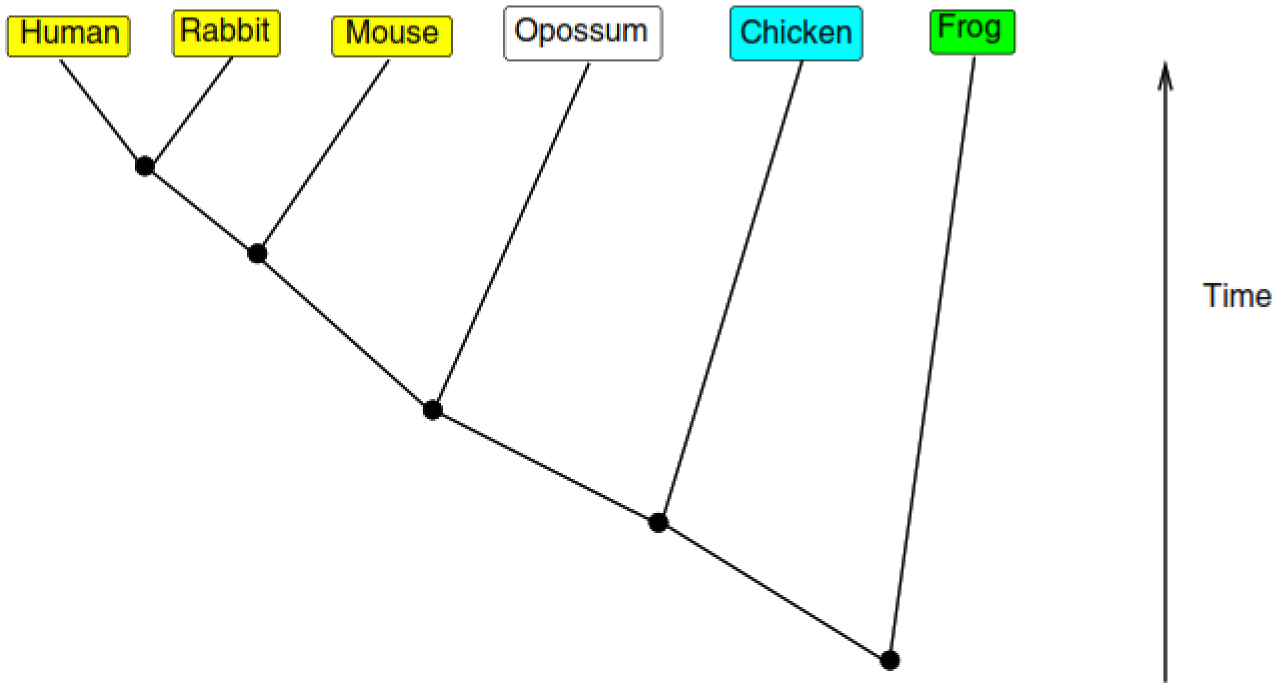
Incomplete data

- In real-world applications of learning, we rarely have fully observed data.

- Incomplete data
  - Hidden variables
  - Missing values

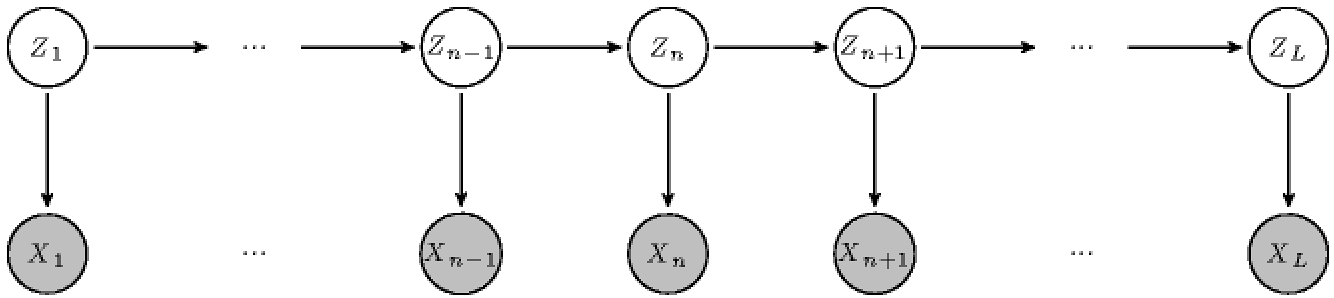


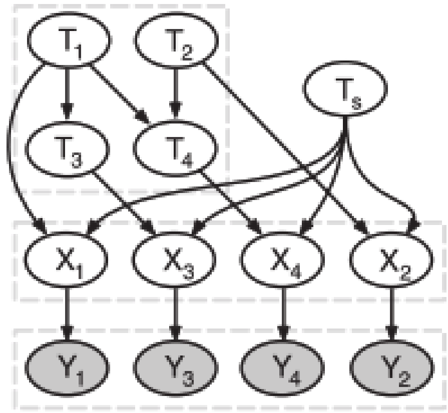
Clustering



Phylogenetic trees

```
         MMMMMMMMMMMMMXXMMMMMMMMMMMMYMMMMYMMMMM
Seq1  =  CTRPNNNTRKSIRPQIGPGQAFYATGD-IGDI-RQAHC
Seq2  =  CGRPNNHRIKGLR--IGPGRAFFAMGAIRGGEIRQAHC
```





Cancer progression

The likelihood function

- Question: write the likelihood function for $X \to Y$
  - Based on counts $M[x^0, y^0], \dots$.

$$L(\boldsymbol{\theta}_X, \boldsymbol{\theta}_{Y|x^0}, \boldsymbol{\theta}_{Y|x^1} : \mathcal{D}) =$$
$$\theta_{x^1}^{M[x^1]} \theta_{x^0}^{M[x^0]} \cdot \theta_{y^1|x^0}^{M[x^0,y^1]} \theta_{y^0|x^0}^{M[x^0,y^0]} \cdot \theta_{y^1|x^1}^{M[x^1,y^1]} \theta_{y^0|x^1}^{M[x^1,y^0]}.$$

- Data:
  - $M[x^1, y^1] = 13,\ M[x^1, y^0] = 16, M[x^0, y^1] = 10, M[x^0, y^0] = 4$

$$\theta_{x^1}^{29}(1-\theta_{x^1})^{14} \cdot \theta_{y^1|x^0}^{10}(1-\theta_{y^1|x^0})^4 \cdot \theta_{y^1|x^1}^{13}(1-\theta_{y^1|x^1})^{16}.$$

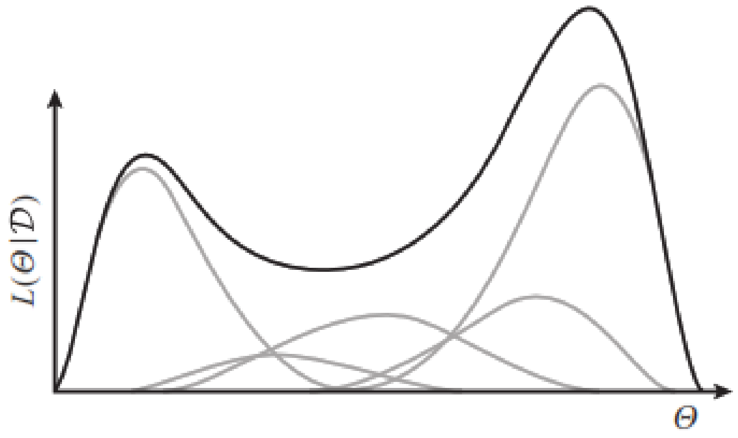
  - The function is well-behaved ; it is log-concave and has a unique global maximum.

- Now assume the first observation was $X[1] = x^0, Y[1] = y^1$. Consider we observed only $Y[1] = y^1$.
- Question: write down modified likelihood.

$$\theta_{x^1}^{29}(1-\theta_{x^1})^{13} \cdot \theta_{y^1|x^0}^{9}(1-\theta_{y^1|x^0})^4 \cdot \theta_{y^1|x^1}^{13}(1-\theta_{y^1|x^1})^{16}\left[\theta_{x^1}\theta_{y^1|x^1} + (1-\theta_{x^1})\theta_{y^1|x^0}\right].$$

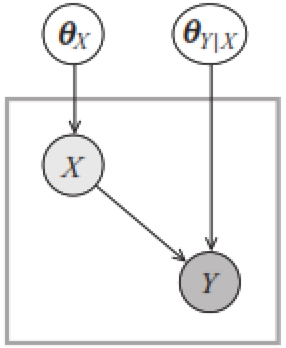
A multimodal likelihood function with incomplete data

- Complete data likelihood defines a unimodal function.
- Their sum can be a multimodal function.
    - Mixture of peaks



$$\theta_{x^1}^{29}(1-\theta_{x^1})^{13}\cdot\theta_{y^1|x^0}^{9}(1-\theta_{y^1|x^0})^{4}\cdot\theta_{y^1|x^1}^{13}(1-\theta_{y^1|x^1})^{16}\left[\theta_{x^1}\theta_{y^1|x^1}+(1-\theta_{x^1})\theta_{y^1|x^0}\right].$$

- Are parameters independence given incomplete data?
    - We lose the property of *parameter independence*
    - The likelihood function is not decomposable!



On global decomposability of likelihood function

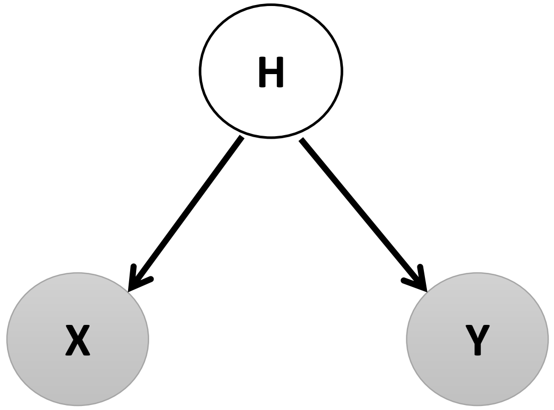
- The probability of observing x and y

$$P(x, y) = \sum_h P(h)P(x \mid h)P(y \mid h).$$

- Likelihood function

$$L(\boldsymbol{\theta} : \mathcal{D}) = \prod_{x,y} \left(\sum_h P(h)P(x \mid h)P(y \mid h) \right)^{M[x,y]}$$

Can we write the likelihood as the product of local likelihoods?



- Likelihood function in the general case:

$$L(\boldsymbol{\theta} : \mathcal{D}) = P(\mathcal{D} \mid \boldsymbol{\theta}) = \sum_{\mathcal{H}} P(\mathcal{D}, \mathcal{H} \mid \boldsymbol{\theta}).$$

exponential numbers of modes (worst case)

a unimodal function

Likelihood function for IID instances:

$$L(\boldsymbol{\theta} : \mathcal{D}) = \prod_m P(\boldsymbol{o}[m] \mid \boldsymbol{\theta}) = \prod_m \sum_{\boldsymbol{h}[m]} P(\boldsymbol{o}[m], \boldsymbol{h}[m] \mid \boldsymbol{\theta}).$$

The learning problem becomes substantially more complex!

Parameter estimation for the partially observed case

University of TEHRAN

- We cover methods for maximum likelihood estimation.

- Problem definition:
  - **Input**
    - a network structure g and the form of the CPDs
    - A data set D that consists of M partial instances
  - **Goal:** Find the values $\hat{\theta}$ that maximize the log-likelihood function: $\hat{\theta} = \arg\max_{\theta} l(\theta : D)$
  - The problem requires optimizing a highly nonlinear and multimodal function over a high-dimensional space.

- Two main classes of methods for performing this optimization
  - A generic nonconvex optimization algorithm such as *gradient ascent*
  - *Expectation maximization:* a more specialized approach for optimizing likelihood functions.

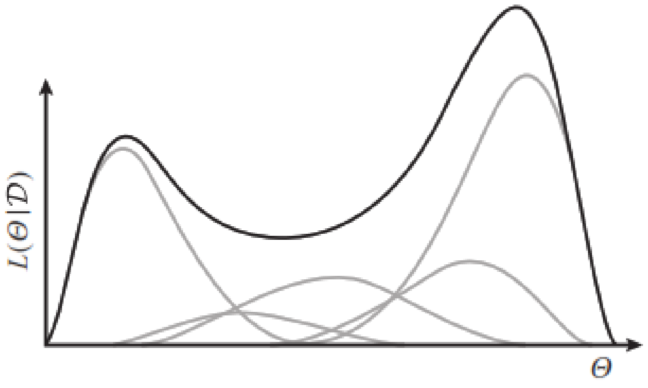
Gradient ascent

- Review A.5.1 and A.5.2 from the text book

Algorithm A.10 Simple gradient ascent algorithm

```
Procedure Gradient-Ascent (
    θ^1,     // Initial starting point
    f_obj,   // Function to be optimized
    δ        // Convergence threshold
)
    t ← 1
    do
        θ^{t+1} ← θ^t + η∇f_obj(θ^t)
        t ← t + 1
    while ||θ^t − θ^{t−1}|| > δ
    return (θ^t)
```

$$\nabla f = \left\langle \frac{\partial f}{\partial \theta_1}, \dots \frac{\partial f}{\partial \theta_n} \right\rangle.$$



- **Question:** consider $X \sim \text{Poisson}(\lambda)$ with $P(x) = \frac{\lambda^x e^{-\lambda}}{x!}$. Suppose $D = \{4, 6, 3, 7\}$, $\lambda^1 = 10$, and $\eta = 1$.
    - Write the gradient-ascent updating formula for λ.
    - Compute λ^4

Gradient ascent for Bayesian networks

- Let $D = \{o[1], \; \ldots, o[M]\}$ be a partially observed data set and X be a variable and U its parent in g. Then

$$\frac{\partial \ell(\boldsymbol{\theta} : \mathcal{D})}{\partial P(x \mid \boldsymbol{u})} = \frac{1}{P(x \mid \boldsymbol{u})} \sum_{m=1}^{M} P(x, \boldsymbol{u} \mid \boldsymbol{o}[m], \boldsymbol{\theta}).$$

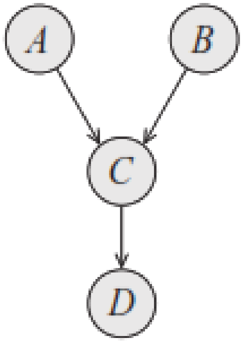
- Given θ is a parameter in a CPD, we can use *chain rule of derivative*

$$\frac{\partial \ell(\boldsymbol{\theta} : \mathcal{D})}{\partial \theta} = \sum_{x, \boldsymbol{u}} \frac{\partial \ell(\boldsymbol{\theta} : \mathcal{D})}{\partial P(x \mid \boldsymbol{u})} \frac{\partial P(x \mid \boldsymbol{u})}{\partial \theta},$$

Example

University of TEHRAN

- given $o = \langle a^1, ?, ?, d^0 \rangle$ for the following network



$$\frac{\partial \ell(\boldsymbol{\theta} : \mathcal{D})}{\partial P(x \mid \boldsymbol{u})} = \frac{1}{P(x \mid \boldsymbol{u})} \sum_{m=1}^{M} P(x, \boldsymbol{u} \mid \boldsymbol{o}[m], \boldsymbol{\theta}).$$

$$\begin{aligned}
\boldsymbol{\theta}_{a^1} &= 0.3 \\
\boldsymbol{\theta}_{b^1} &= 0.9 \\
\boldsymbol{\theta}_{c^1 \mid a^0, b^0} &= 0.83 \\
\boldsymbol{\theta}_{c^1 \mid a^0, b^1} &= 0.09 \\
\boldsymbol{\theta}_{c^1 \mid a^1, b^0} &= 0.6 \\
\boldsymbol{\theta}_{c^1 \mid a^1, b^1} &= 0.2 \\
\boldsymbol{\theta}_{d^1 \mid c^0} &= 0.1 \\
\boldsymbol{\theta}_{d^1 \mid c^1} &= 0.8.
\end{aligned}$$

Probabilities of the consistent cases with o

$$\begin{aligned}
P(\langle a^1, b^1, c^1, d^0 \rangle) &= 0.3 \cdot 0.9 \cdot 0.2 \cdot 0.2 = 0.0108 \\
P(\langle a^1, b^1, c^0, d^0 \rangle) &= 0.3 \cdot 0.9 \cdot 0.8 \cdot 0.9 = 0.1944 \\
P(\langle a^1, b^0, c^1, d^0 \rangle) &= 0.3 \cdot 0.1 \cdot 0.6 \cdot 0.2 = 0.0036 \\
P(\langle a^1, b^0, c^0, d^0 \rangle) &= 0.3 \cdot 0.1 \cdot 0.4 \cdot 0.9 = 0.0108.
\end{aligned}$$

$$\begin{aligned}
P(\langle a^1, b^1, c^1, d^0 \rangle \mid \boldsymbol{o}) &= 0.0492 \\
P(\langle a^1, b^1, c^0, d^0 \rangle \mid \boldsymbol{o}) &= 0.8852 \\
P(\langle a^1, b^0, c^1, d^0 \rangle \mid \boldsymbol{o}) &= 0.0164 \\
P(\langle a^1, b^0, c^0, d^0 \rangle \mid \boldsymbol{o}) &= 0.0492.
\end{aligned}$$

$$\begin{aligned}
\frac{\partial \log P(\boldsymbol{o})}{\partial P(d^1 \mid c^0)} &= \frac{P(d^1, c^0 \mid \boldsymbol{o})}{P(d^1 \mid c^0)} = \frac{0}{0.1} = 0 \\
\frac{\partial \log P(\boldsymbol{o})}{\partial P(d^0 \mid c^0)} &= \frac{P(d^0, c^0 \mid \boldsymbol{o})}{P(d^0 \mid c^0)} = \frac{0.8852 + 0.0492}{0.9} = 1.0382 \\
\frac{\partial \log P(\boldsymbol{o})}{\partial P(d^1 \mid c^1)} &= \frac{P(d^1, c^1 \mid \boldsymbol{o})}{P(d^1 \mid c^1)} = \frac{0}{0.8} = 0 \\
\frac{\partial \log P(\boldsymbol{o})}{\partial P(d^0 \mid c^1)} &= \frac{P(d^0, c^1 \mid \boldsymbol{o})}{P(d^0 \mid c^1)} = \frac{0.0492 + 0.0164}{0.2} = 0.328.
\end{aligned}$$

1. Inference for each instance $P(X[m], U[m] \mid o[m], \theta)$ at each iteration
2. Ensure that parameters describe a legal probability distribution
   - Reparametrization
   - Lagrange multipliers