6.891 Machine learning and neural networks

Tommi Jaakkola
MIT AI LAB
tommi@ai.mit.edu

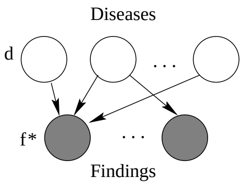
LECTURE 22 (December 5, 2000)

FINAL EXAM – Thursday December 7

Topics

- Parameter estimation in graphical models
  - maximum likelihood, maximum a posteriori (MAP)
- Model selection
  - minimum description length principle
  - Bayesian score
  - learning the graph structure
- About the final...

Parameter estimation in Bayesian networks

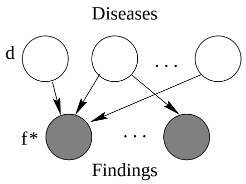


- Given the structure of a Bayesian network, we wish to adjust the parameters in the model so as to best capture the observed data
- Several possible estimation criteria
  1. Maximum likelihood (ML)
  2. Maximum a posteriori (MAP)
  3. Bayesian

     ...

Parameter estimation cont'd

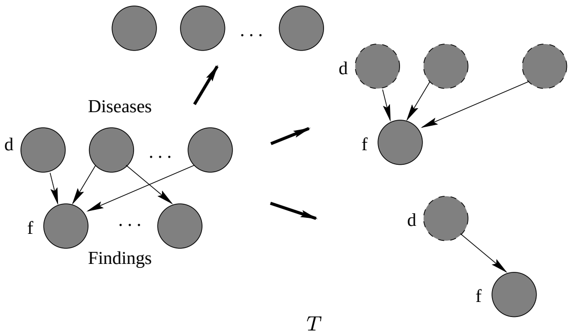
- We distinguish two cases
  1. *Complete observations* where we have a value assignment for each variable in the graph model
     - typically feasible regardless of the criterion
  2. *Incomplete observations* where some of value assignments are missing
     - typically much harder...
     - it matters how the values are missing



incomplete observations

Complete observations

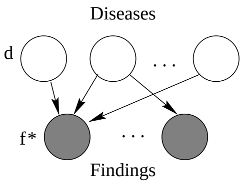
- When we have complete observations, the estimation problem reduces to n independent problems of estimating each conditional probability separately (e.g., via maximum likelihood)



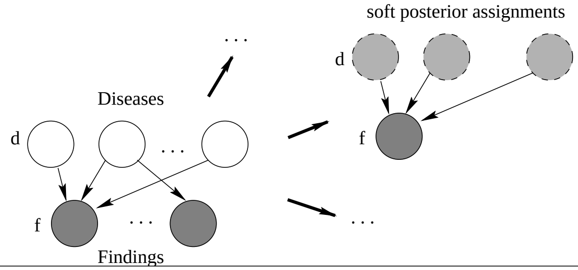
$$\text{For example:} \quad \max_{\theta_i} \sum_{t=1}^{T} \log P(f_i^t | d_{pa_i}^t, \theta_i) \tag{1}$$

where θ_i are the parameters associated with the conditional probability of finding f_i given its "parents" d_{pa_i}

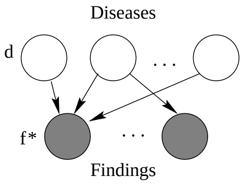
Estimation with incomplete data



- The estimation problem no longer separates when we have incomplete data
- We regain separation by *inferring* values for the hidden variables



EM and Bayesian networks



- The EM-algorithm

  E-step: compute the posterior probabilities over the unobserved (hidden) variables

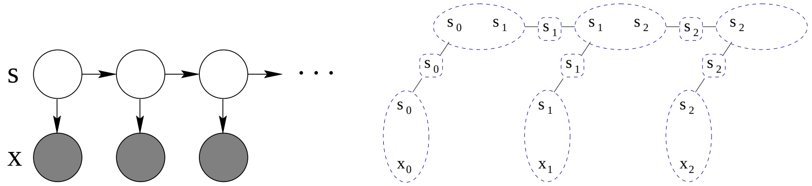
$$\text{For example:} \quad P(d_{pa_i} | f^t, \theta^o) \tag{2}$$

  M-step: estimate the conditional probabilities by maximizing the likelihood of the observed *and* inferred value assignments

$$\text{For example:} \quad \max_{\theta_i} \sum_{t=1}^{T} \left[\sum_{d_{pa_i}} P(d_{pa_i} | f^t, \theta^o) \, \log P(f_i^t | d_{pa_i}, \theta_i) \right] \tag{3}$$

E-step and the junction tree algorithm

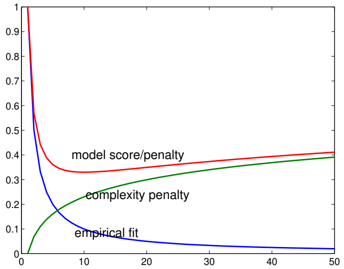
- Example: HMM



- We obtain the required posterior probabilities in the E-step directly from the junction tree algorithm

Model selection

- As in structural risk minimization in classification we have to find a criterion that somehow balances the model complexity with the empirical fit



Minimum description length

- Minimum description length (MDL) principle:

  We find a model that attains the minimum total encoding length: the length of the code needed to describe the observed data given the model and the length of the code needed to describe the model itself.

$$\text{Total DL} = \overbrace{(\text{DL of data given the model})}^{\text{empirical fit}} + \overbrace{(\text{DL of model})}^{\text{complexity}}$$

- A long description length (DL) may result from

  a) poor explanation (likelihood) of the observed data

  b) too complex model (too many choices a priori)

Digression: encoding length

- Suppose we have a sequence of n random numbers $y_1, y_2, \ldots$ each drawn with probability $P(y)$

  11211113141111121...

- We use $-\log_2 P(y)$ bits to encode each number y

  Higer probability $\Rightarrow$ smaller number of bits

- On average, it would take

$$\begin{aligned} \frac{1}{n}\sum_{t=1}^{n} -\log_2 P(y_t) &= \frac{1}{n}\sum_{y} -N(y)\log_2 P(y) && (4)\\ &\rightarrow \sum_{y} -P(y)\log_2 P(y) && (5)\\ &= H(y) && (6)\end{aligned}$$

bits to encode a single number. Here $H(y)$ is the Shannon entropy (uncertainty) of the random numbers.

Minimum description length cont'd

- Suppose we observe n i.i.d. training examples $D = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$
- Description length of the observed data given a specific setting of the model parameters $\bar{\theta}$:

$$-\log P(D|\bar{\theta}) = -\sum_{t=1}^{n} \log P(\mathbf{x}_t|\bar{\theta}) \qquad (7)$$

- Description length of the model

$$-\log P(\bar{\theta}) \qquad (8)$$

where $P(\bar{\theta})$ is a prior distribution over the parameters.

- Are these sufficient? How do we set the parameter $\bar{\theta}$?

Minimum description length cont'd

- We have to also decide the precision at which to encode the parameters $\bar{\theta}$ (why would be spare anyy bits for useless parameters?)
- We have to add a precision cost to the model description length

$$-\log P(\bar{\theta}) - \log(\delta) \tag{9}$$

where the parameters $\bar{\theta}$ are now described upto precision δ.

High precision (small δ) $\Rightarrow$ large precision cost

- The total description length is found by adding all the terms and optimizing the sum with respect to $\bar{\theta}$ and the precision δ

$$\text{Total DL} = \min_{\bar{\theta},\delta} \left\{ -\sum_{t=1}^{n} \log P(\mathbf{x}_t|\bar{\theta}) - \log P(\bar{\theta}) - \log(\delta) \right\} \tag{10}$$

(note that the possible choices of $\bar{\theta}$ depend on the precision δ)

Minimum description length cont'd

- Asymptotically (when the number of observations n is large) the description length reduces to

$$\text{Total DL} \approx -\sum_{t=1}^{n} \log P(\mathbf{x}_t|\hat{\theta}) + \frac{d}{2}\log(n) \tag{11}$$

where d is the number of parameters in the model and $\hat{\theta}$ is the maximum likelihood parameter estimate.

- This still has the right flavor...

Minimum description length: example

- We have two binary variables x_1 and x_2 and two competing models

Model 0: two parameters needed for $P(x_1)P(x_2)$
Model 1: three parameters needed for $P(x_1, x_2)$

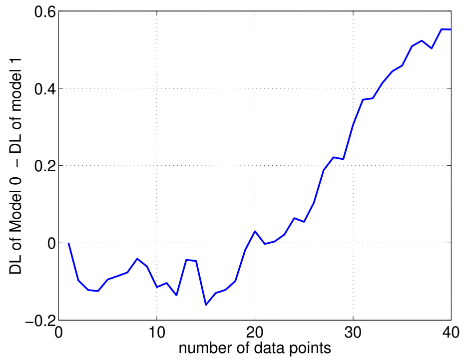
- Observed data:

x_1	x_2
1	0
0	1
1	1
...	...

(here $x_1 = 1$ and $x_2 = 1$ with probability 0.5 and otherwise x_1 and x_2 are selected uniformly at random)

Minimum description length: example

- The difference between the description lengths of the two models as a function of the number of data points:



The figure is averaged over several random generation of the training set

Bayesian score

- MDL can be viewed as an approximation to the *Bayesian score*

$$\text{Bayesian score} = \log \int P(D|\theta)P(\theta)d\theta \tag{12}$$

  where D is the observed data.

- In a too complex model, the parameter space is huge and only a tiny fraction of such space explains the data well $\Rightarrow$ low score
- In a too simple model, no setting of the parameters explains the data well $\Rightarrow$ low score