

# Parameter Estimation

## Lecture #7

Acknowledgement: Some slides of this lecture are due to Nir Friedman.

## Likelihood function for a die: Multinomial sampling

Let  $X$  be a random variable with 6 values  $x_1, \dots, x_6$  denoting the six outcomes of a die. Suppose we observe a sequence of independent outcomes:

$$\text{Data} = (x_6, x_1, x_1, x_3, x_2, x_2, x_3, x_4, x_5, x_2, x_6)$$

What is the probability of this data ?

If we knew the long-run frequencies  $\theta_i$  for falling on side  $x_i$ , then,

$$P(\text{Data} | \Theta) = \theta_1^2 \cdot \theta_2^3 \cdot \theta_3^2 \cdot \theta_4 \cdot \theta_5 \cdot \left(1 - \sum_{i=1}^5 \theta_i\right)^2$$

Where  $\Theta = \{\theta_1, \theta_2, \theta_3, \theta_4, \theta_5\}$  are called the **parameters** of the likelihood function. We wish to estimate these parameters from the data we have seen.

2

## Sufficient Statistics

- ◆ To compute the probability of data in the die example we only require to record the number of times  $N_i$  falling on side  $i$  (namely,  $N_1, N_2, \dots, N_6$ ).
- ◆ We do not need to recall the entire sequence of outcomes

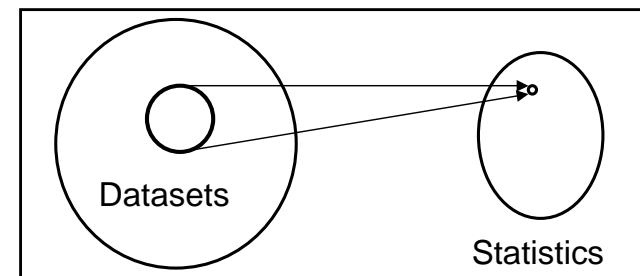
$$P(\text{Data} | \Theta) = \theta_1^{N_1} \cdot \theta_2^{N_2} \cdot \theta_3^{N_3} \cdot \theta_4^{N_4} \cdot \theta_5^{N_5} \cdot \left(1 - \sum_{i=1}^5 \theta_i\right)^{N_6}$$

- ◆  $\{N_i | i=1..6\}$  is called the **sufficient statistics** for the multinomial sampling.

3

## Sufficient Statistics

- ◆ A **sufficient statistics** is a function of the data that summarizes the relevant information for the likelihood
- ◆ Formally,  $s(\text{Data})$  is a sufficient statistics if for any two datasets  $D$  and  $D'$ 
  - $s(\text{Data}) = s(\text{Data}') \Rightarrow P(\text{Data} | \theta) = P(\text{Data}' | \theta)$



4

## Maximum Likelihood Estimate

Maximum likelihood estimate is an assignment to the parameters that maximizes the probability of data (i.e., the likelihood function).

Usually one maximizes the log-likelihood function which is easier to do and gives an identical answer:

$$\begin{aligned} \log P(\text{Data} | \Theta) &= \log \left[ \theta_1^{N_1} \cdot \theta_2^{N_2} \cdot \theta_3^{N_3} \cdot \theta_4^{N_4} \cdot \theta_5^{N_5} \cdot \left(1 - \sum_{i=1}^5 \theta_i\right)^{N_6} \right] \\ &= \sum_{i=1}^5 N_i \log \theta_i + N_6 \log \left(1 - \sum_{i=1}^5 \theta_i\right) \end{aligned}$$

A sufficient condition for maximum is:

$$\frac{\partial \log P(\text{Data} | \Theta)}{\partial \theta_i} = \frac{N_i}{\theta_i} - \frac{N_6}{1 - \sum_{i=1}^5 \theta_i} = 0$$

5

## Finding the Maximum

We have just found that:

$$\frac{N_i}{\theta_i} = \frac{N_6}{1 - \sum_{i=1}^5 \theta_i}$$

Divide the  $i^{\text{th}}$  and  $j^{\text{th}}$  equations:  $\theta_j = \frac{N_j}{N_i} \theta_i$

Sum from  $j=1$  to 6:  $1 = \frac{\sum_{j=1}^6 N_j}{N_i} \theta_i$

Hence the MLE is given by:

$$\theta_i = \frac{N_i}{N} \quad i = 1, \dots, 6$$

6

## Adding Pseudo Counts

The MLE given by

$$\theta_i = \frac{N_i}{N} \quad i = 1, \dots, 6,$$

can be misleading for small data sets because it could happen that a small data set is not typical. For example, it might be that we know that the dice is manufactured to be loaded but the small dataset we examined does not show this property.

The MAP estimate is given by

$$\theta_i = \frac{N_i + N'_i}{N + N'} \quad i = 1, \dots, 6$$

The six pseudo counts  $N'_i$  sum to  $N'$ . They express one's assessment regarding the frequencies for each side **prior** to seeing the data. Large  $N'$  indicates high confidence. Smaller than 1 values are possible.

The MAP estimate can be justified as maximizing one's **posterior** (namely, after seeing the data) best estimate of the frequencies for each side. The theory formally justifying this formula is called **Bayesian Statistics** (not covered in this course due to time constraints).

7

## Example: The ABO locus

A **locus** is a particular place on the chromosome. Each locus' state (called **genotype**) consists of two **alleles** - one parental and one maternal. Some **loci** (plural of locus) determine distinguished features. The ABO locus, for example, determines blood type.

The ABO locus has six possible genotypes {a/a, a/o, b/o, b/b, a/b, o/o}. The first two genotypes determine blood type A, the next two determine blood type B, then blood type AB, and finally blood type O.

We wish to estimate the proportion in a population of the 6 genotypes.

Suppose we randomly sampled  $N$  individuals and found that  $N_{a/a}$  have genotype a/a,  $N_{a/b}$  have genotype a/b, etc. Then, the MLE is given by:

$$\theta_{a/a} = \frac{N_{a/a}}{N}, \theta_{a/o} = \frac{N_{a/o}}{N}, \theta_{b/b} = \frac{N_{b/b}}{N}, \theta_{b/o} = \frac{N_{b/o}}{N}, \theta_{a/b} = \frac{N_{a/b}}{N}, \theta_{o/o} = \frac{N_{o/o}}{N}$$

8

## The ABO locus (Cont.)

However, testing individuals for their genotype is a very expensive test. Can we estimate the proportions of genotype using the common cheap blood test with outcome being one of the four blood types (A, B, AB, O) ?

The problem is that among individuals measured to have blood type A, we don't know how many have genotype a/a and how many have genotype a/o. So what can we do ?

We use the **Hardy-Weinberg equilibrium** rule that tells us that in equilibrium the frequencies of the three alleles  $\theta_a, \theta_b, \theta_o$  in the population determine the frequencies of the genotypes as follows:  $\theta_{a/b} = 2\theta_a \theta_b$ ,  $\theta_{a/o} = 2\theta_a \theta_o$ ,  $\theta_{b/o} = 2\theta_b \theta_o$ ,  $\theta_{a/a} = [\theta_a]^2$ ,  $\theta_{b/b} = [\theta_b]^2$ ,  $\theta_{o/o} = [\theta_o]^2$ . **So now we have three parameters that we need to estimate.**

9

## The Likelihood Function

Let  $X$  be a random variable with 6 values  $x_{a/a}, x_{a/o}, x_{b/b}, x_{b/o}, x_{a/b}, x_{o/o}$  denoting the six genotypes. The parameters are  $\Theta = \{\theta_a, \theta_b, \theta_o\}$ .

The probability  $P(X = x_{a/b} | \Theta) = 2\theta_a \theta_b$ .

The probability  $P(X = x_{o/o} | \Theta) = \theta_o \theta_o$ .

And so on for the other four genotypes.

What is the probability of Data={B,A,B,B,O,A,B,A,O,B, AB} ?

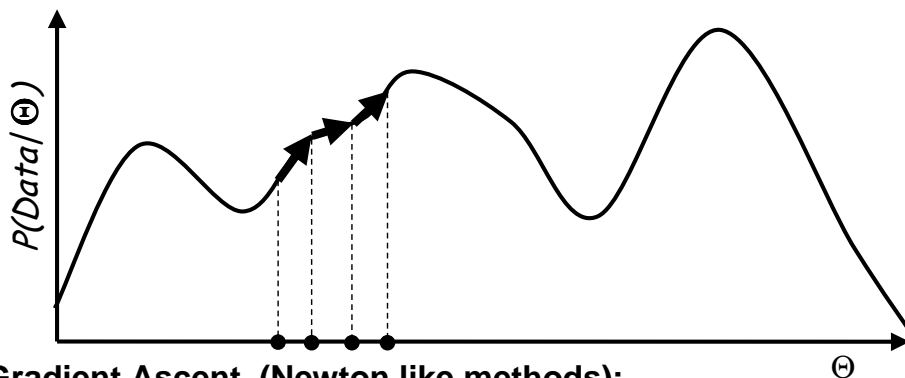
$$P(\text{Data} | \Theta) = (\theta_a^2 + 2\theta_a \theta_o)^3 (\theta_b^2 + 2\theta_b \theta_o)^5 (2\theta_a \theta_b)^1 (\theta_o \theta_o)^2$$

Obtaining the maximum of this function yields the MLE. This can be done by multidimensional Newton's algorithm.

10

## Computing MLE

- ◆ Finding MLE parameters: **nonlinear optimization** problem



### Gradient Ascent (Newton like methods):

Follow gradient of likelihood w.r.t. to parameters (As taught in your favorite Numerical Analysis course). Improve, by adding line search methods to determine step size and get faster convergence. Start at several random locations.

11

## Gene Counting

Had we known the counts  $n_{a/a}$  and  $n_{a/o}$  (blood type A individuals), we could have estimated  $\theta_a$  from  $n$  individuals as follows (and similarly estimate  $\theta_b$  and  $\theta_o$ ):

$$\theta_a \leftarrow \frac{2n_{a/a} + n_{a/o} + n_{a/b}}{2n}$$

Can we compute what  $n_{a/a}$  and  $n_{a/o}$  are expected to be ?

Using the current estimates of  $\theta_a$  and  $\theta_o$  we can as follows:

$$n_{a/a} \leftarrow n_a \frac{\theta_a^2}{\theta_a^2 + 2\theta_a \theta_o} \quad n_{a/o} \leftarrow n_a \frac{2\theta_a \theta_o}{\theta_a^2 + 2\theta_a \theta_o}$$

We repeat these two steps until the parameters converge.

12

# Gene Counting (example of EM)

Input: Counts of each blood type  $n_A, n_B, n_O, n_{AB}$  of  $n$  people.

Desired Output: ML estimate of allele frequencies  $\theta_a, \theta_b, \theta_o$ .

Initialization: Set  $\theta_a, \theta_b$ , and  $\theta_o$  to arbitrary values (say, 1/3).

Repeat

**E-step (Expectation):**

$$\begin{aligned} n_{a/a} &\leftarrow n_A \frac{\theta_a^2}{\theta_a^2 + 2\theta_a\theta_o} & n_{a/o} &\leftarrow n_A \frac{2\theta_a\theta_o}{\theta_a^2 + 2\theta_a\theta_o} \\ n_{b/b} &\leftarrow n_B \frac{\theta_b^2}{\theta_b^2 + 2\theta_b\theta_o} & n_{b/o} &\leftarrow n_B \frac{2\theta_b\theta_o}{\theta_b^2 + 2\theta_b\theta_o} \end{aligned}$$

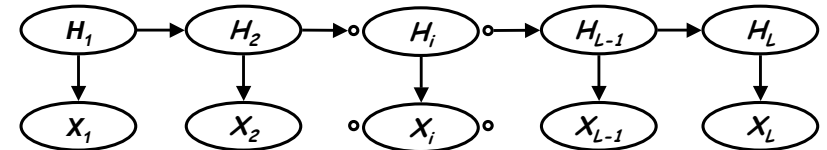
**M-step (Maximization):**

$$\theta_a \leftarrow \frac{2n_{a/a} + n_{a/o} + n_{AB}}{2n} \quad \theta_b \leftarrow \frac{2n_{b/b} + n_{b/o} + n_{AB}}{2n} \quad \theta_o \leftarrow \frac{2n_O + n_{a/o} + n_{b/o}}{2n}$$

Until  $\theta_a, \theta_b$ , and  $\theta_o$  converge

# EM for HMMs

Recall that a non-homogenous HMM is a Bayesian network of the following form with different parameters on each edge :



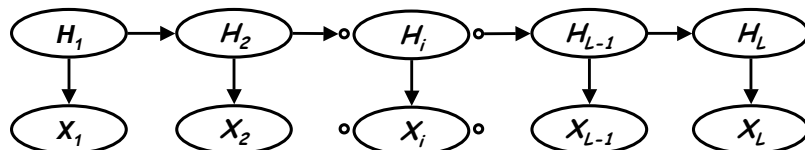
**E step:** Compute the **probability**  $p(h_i, h_{i+1} | x^j)$  for the  $j$ -th sequence  $x^j = (x_1, \dots, x_L)$  in the training data set for each of  $H_i$ 's values  $h_i$  and  $H_{i+1}$ 's values  $h_{i+1}$ .

**M step:** Estimate the **transition probability**  $p(h_{i+1} | h_i)$  via the sum

$\sum p(h_i, h_{i+1} | x^j)$  over all sequences  $x^j$  in the data set, namely,

$$p(h_{i+1} | h_i) \leftarrow \frac{\sum_j p(h_i, h_{i+1} | x^j)}{\sum_{h_{i+1}} \sum_j p(h_i, h_{i+1} | x^j)}$$

# EM for HMMs (Cont.)



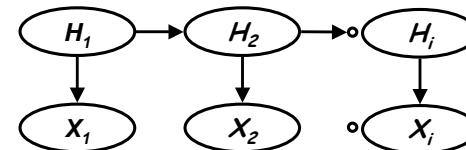
**E step:** Compute the **probability**  $p(h_i, h_{i+1} | x^j)$  for the  $j$ -th sequence  $x^j = (x_1, \dots, x_L)$  in the training data set for each of  $H_i$ 's values  $h_i$  and  $H_{i+1}$ 's values  $h_{i+1}$ .

**M step (cont.):** Estimate the **emission probability**  $p(x_i | h_i)$  via the sum

$$p(x_i | h_i) \leftarrow K \cdot \sum_{\{j | (x^j)_i = x_i\}} p(x^j, h_i) = K \cdot \sum_{\{j | (x^j)_i = x_i\}} f(h_i) b(h_i)$$

where  $K$  is a normalizing constant obtained by summing on all  $x_i$  values.

# Recall the forward algorithm



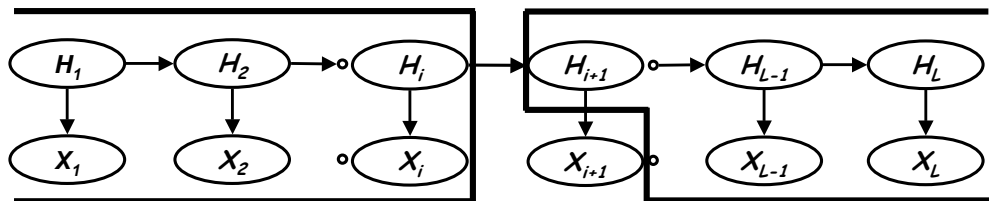
The task: Compute  $f(h_i) = P(x_1, \dots, x_i, h_i)$  for  $i=1, \dots, L$  (namely, considering evidence up to time slot  $i$ ).

Initial step:  $P(x_1, h_1) = P(h_1) P(x_1 | h_1)$

Step  $i$ :  $P(x_1, \dots, x_i, h_i) = \sum_{h_{i-1}} P(x_1, \dots, x_{i-1}, h_{i-1}) P(h_i | h_{i-1}) P(x_i | h_i)$

Recall that the backward algorithm is similar: multiplying matrices from the end to the start and summing on hidden states.

## Decomposing the E-step computation



$$P(x_1, \dots, x_L, h_i, h_{i+1}) = P(x_1, \dots, x_i, h_i) p(h_{i+1} | h_i) p(x_{i+1} | h_{i+1}) P(x_{i+2}, \dots, x_L | h_{i+1})$$

$$\underbrace{P(x_1, \dots, x_i, h_i)}_{x^j} = f(h_i) p(h_{i+1} | h_i) p(x_{i+1} | h_{i+1}) b(h_{i+1})$$

Via the forward algorithm

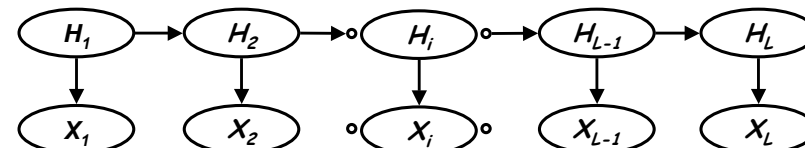
Via the backward algorithm

$$p(h_i, h_{i+1} | x^j) = \frac{f(h_i) p(h_{i+1} | h_i) p(x_{i+1} | h_{i+1}) b(h_{i+1})}{\sum_{h_i} \sum_{h_{i+1}} f(h_i) p(h_{i+1} | h_i) p(x_{i+1} | h_{i+1}) b(h_{i+1})}$$

17

## EM for homogeneous HMMs

Now the parameters on each transition probability table are the same.



**E step:** Compute, for every  $i$ , the probability  $p(H_i = a, H_{i+1} = b | x^j)$  for the  $j$ -th sequence  $x^j = (x_1, \dots, x_L)$  in the training data set for each pair of states  $a, b$ .

**M step:** Estimate the transition probability  $p(b|a)$  via the sum  $\sum_i \sum_j p(H_i = a, H_{i+1} = b | x^j)$  (The only change is the extra sum on  $i$ ).

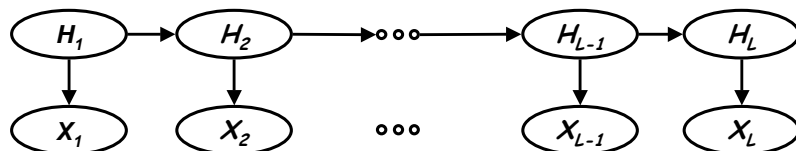
$$p(b|a) \leftarrow \frac{\sum_j p(a, b | x^j)}{\sum_b \sum_j p(a, b | x^j)}$$

Note: Similar change when learning emission probabilities  $p(x_i | h_i)$ .

18

## The CpG Island example (Summary)

The HMM we used:  $\text{Domain}(H_i) = \{I, N\} \times \{A, C, T, G\}$  (8 values)



In this representation  $P(x_i | h_i) = 0$  or  $1$  depending on whether  $x_i$  is consistent with  $h_i$ . E.g.  $x_i = G$  is consistent with  $h_i = (I, G)$  and with  $h_i = (N, G)$  but not with any other state of  $h_i$ .

Solution:

1. Learn the parameters using the EM algorithm.
2. Answer the following MAP query using Viterbi's algorithm:

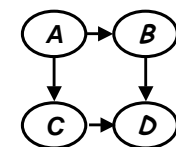
$$(h_1^*, \dots, h_L^*) = \max_{(h_1, \dots, h_L)} \arg p(h_1, \dots, h_L | x_1, \dots, x_L)$$

19

## Expectation Maximization (EM) for Bayesian networks

**Intuition (as before):**

- ◆ When we have access to all counts, then we can find the ML estimate of all parameters in all local tables directly by counting.
- ◆ However, missing values do not allow us to perform such counts.
- ◆ So instead, we compute the **expected counts** using the current parameter assignment, and then use them to compute the **maximum** likelihood estimate.



$$P(A = a | \theta)$$

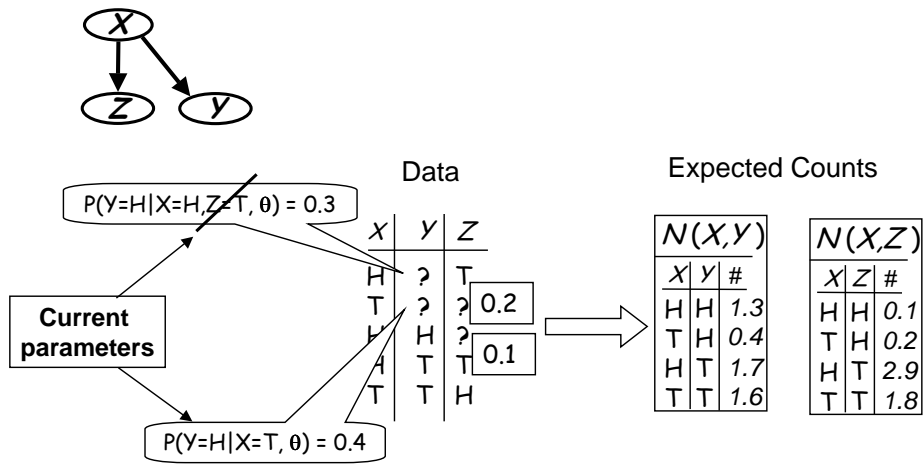
$$P(B = b | A = a, \theta)$$

$$P(C = c | A = a, \theta)$$

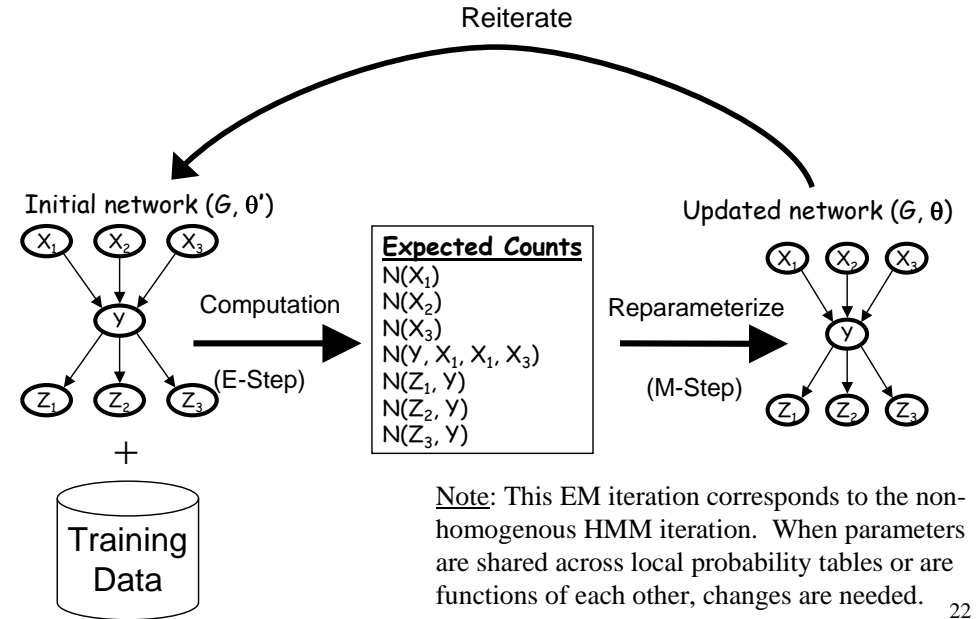
$$P(D = d | b, c, \theta)$$

20

# Expectation Maximization (EM)



# EM (cont.)



## EM in Practice

### Initial parameters:

- ◆ Random parameters setting
- ◆ “Best” guess from other source

### Stopping criteria:

- ◆ Small change in likelihood of data
- ◆ Small change in parameter values

### Avoiding bad local maxima:

- ◆ Multiple restarts
- ◆ Early “pruning” of unpromising ones

## Relative Entropy – a measure of difference among distributions

We define the relative entropy  $H(P||Q)$  for two probability distributions  $P$  and  $Q$  of a variable  $X$  (with  $x$  being a value of  $X$ ) as follows:

$$H(P||Q) = \sum x_i P(x_i) \log_2(P(x_i)/Q(x_i))$$

This is a measure of difference between  $P(x)$  and  $Q(x)$ . It is not a symmetric function. The distribution  $P(x)$  is assumed the “true” distribution used for taking **the expectation of the log of the difference** with the following properties:

$$H(P||Q) \geq 0$$

Equality holds if and only if  $P(x) = Q(x)$  for all  $x$ .

## Average Score for sequence comparisons

Recall that we have defined the scoring function via

$$\sigma(a,b) = \log \frac{P(a,b)}{Q(a)Q(b)}$$

Note that the average score is the relative entropy  $H(P || Q)$  where  $Q(a,b) = Q(a)Q(b)$ .

Relative entropy also arises when choosing amongst competing models.

25

## The setup of the EM algorithm

We start with a likelihood function parameterized by  $\theta$ .

The **observed quantity** is denoted  $\mathbf{X}=\mathbf{x}$ . It is often a vector  $x_1, \dots, x_L$  of observations (e.g.,  $N_A, N_B, N_{AB}, N_O$ , or evidence for some nodes in a Bayesian network).

The **hidden quantity** is a vector  $\mathbf{Y}=\mathbf{y}$  (e.g.  $N_{a/a}, N_{a/o}, N_{b/b}, N_{b/o}$ , states of **unobserved** variables in a Bayes network). The quantity  $y$  is defined such that if it were known, the likelihood of the completed data point  $P(\mathbf{x},\mathbf{y}|\theta)$  is easy to maximize.

The log-likelihood of an observation  $\mathbf{x}$  has the form:

$$\log P(\mathbf{x} | \theta) = \log P(\mathbf{x},\mathbf{y} | \theta) - \log P(\mathbf{y} | \mathbf{x}, \theta)$$

(Because  $P(\mathbf{x},\mathbf{y} | \theta) = P(\mathbf{x} | \theta) P(\mathbf{y} | \mathbf{x}, \theta)$ ).

26

## The goal of EM algorithm

The log-likelihood of an observation  $\mathbf{x}$  has the form:

$$\log P(\mathbf{x} | \theta) = \log P(\mathbf{x},\mathbf{y} | \theta) - \log P(\mathbf{y} | \mathbf{x}, \theta)$$

For independent points  $(x^i, y^i)$ ,  $i=1, \dots, m$ , we can similarly write:

$$\sum_i \log P(x^i | \theta) = \sum_i \log P(x^i, y^i | \theta) - \sum_i \log P(y^i | x^i, \theta)$$

We will stick to one observation in our derivation recalling that all derived equations can be modified by summing over  $\mathbf{x}$ .

The goal: Starting with a current parameter vector  $\theta'$ , EM's goal is to find a new vector  $\theta$  such that  $P(\mathbf{x} | \theta) > P(\mathbf{x} | \theta')$  with the highest possible difference.

The result: After enough iterations EM reaches a local maximum of the likelihood  $P(\mathbf{x} | \theta)$ .

27

## The Mathematics involved

Recall that the expectation of a random variable  $Y$  with a pdf  $P(y)$  is given by  $E[Y] = \sum_y y p(y)$ .

The expectation of a function  $L(Y)$  is given by  $E[L(Y)] = \sum_y L(y) p(y)$ .

A bit harder to comprehend example:

$$Q(\theta | \theta') \equiv E_{\theta'}[\log p(\mathbf{x},\mathbf{y}|\theta)] = \sum_y p(\mathbf{y} | \mathbf{x}, \theta') \log p(\mathbf{x}, \mathbf{y} | \theta)$$

The expectation operator  $E$  is linear. For two random variables  $X, Y$ , and constants  $a, b$ , the following holds

$$E[aX+bY] = a E[X] + b E[Y]$$

28

## The Mathematics involved (Cont.)

Starting with  $\log P(x|\theta) = \log P(x, y|\theta) - \log P(y|x, \theta)$ , multiplying both sides by  $P(y|x, \theta')$ , and summing over  $y$ , yields

$$\begin{aligned} \log P(x|\theta) &= \sum_y P(y|x, \theta') \log P(x, y|\theta) - \sum_y P(y|x, \theta') \log P(y|x, \theta) \\ &= E_{\theta'}[\log p(x, y|\theta)] = Q(\theta|\theta') \end{aligned}$$

We now observe that

$$\Delta = \log P(x|\theta) - \log P(x|\theta') = Q(\theta|\theta') - Q(\theta'|\theta') + \sum_y P(y|x, \theta') \log [P(y|x, \theta') / P(y|x, \theta)]$$

So choosing  $\theta^* = \operatorname{argmax}_{\theta} Q(\theta|\theta')$  maximizes the difference  $\Delta$ , and repeating this process leads to a local maximum of  $\log P(x|\theta)$ .  $\geq 0$  (relative entropy)

29

## The EM algorithm itself

**Input:** A likelihood function  $p(x, y|\theta)$  parameterized by  $\theta$ .

**Initialization:** Fix an arbitrary starting value  $\theta'$

Repeat

E-step: Compute  $Q(\theta|\theta') = E_{\theta'}[\log P(x, y|\theta)]$

M-step:  $\theta' \leftarrow \operatorname{argmax}_{\theta} Q(\theta|\theta')$

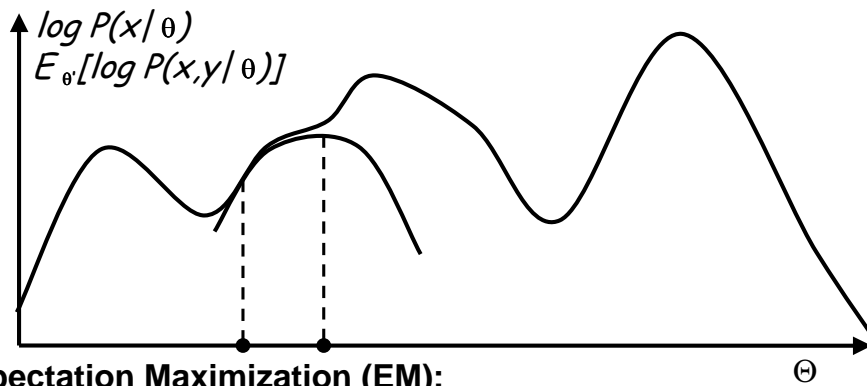
Until  $\Delta = \log P(x|\theta) - \log P(x|\theta') < \epsilon$

**Comment:** At the M-step one can actually choose any  $\theta'$  as long as  $\Delta > 0$ . This change yields the so called Generalized EM algorithm. It is important when argmax is hard to compute.

30

## MLE from Incomplete Data

◆ Finding MLE parameters: **nonlinear optimization** problem



### Expectation Maximization (EM):

Use "current point" to construct alternative function (which is "nice")  
Guaranty: maximum of new function has a higher likelihood than the current point

31

## Gene Counting Revisited (as EM)

The observations: The variables  $X=(N_A, N_B, N_{AB}, N_O)$  with a specific assignment  $x = (n_A, n_B, n_{AB}, n_O)$ .

The hidden quantity: The variables  $Y=(N_{a/a}, N_{a/o}, N_{b/b}, N_{b/o})$  with a specific assignment  $y = (n_{a/a}, n_{a/o}, n_{b/b}, n_{b/o})$ .

The parameters:  $\theta = \{\theta_a, \theta_b, \theta_o\}$ .

The likelihood of the completed data of  $n$  points:

$$\begin{aligned} P(x, y|\theta) &= P(n_{AB}, n_O, n_{a/a}, n_{a/o}, n_{b/b}, n_{b/o}|\theta) = \\ &= \left( \frac{n!}{n_{a/a}! n_{a/o}! n_{b/b}! n_{b/o}! n_{a/b}! n_{o/o}!} \right) \cdot \\ &\quad \cdot (\theta_a^2)^{n_{a/a}} (2\theta_a \theta_o)^{n_{a/o}} (\theta_b^2)^{n_{b/b}} (2\theta_b \theta_o)^{n_{b/o}} (2\theta_a \theta_b)^{n_{a/b}} (\theta_o^2)^{n_{o/o}} \end{aligned}$$

32



## The E-step of Gene Counting

The likelihood of the hidden data given the observed data of n points:

$$P(y | x, \theta') = P(n_{a/a}, n_{a/o}, n_{b/b}, n_{b/o} | n_A, n_B, n_{AB}, n_O, \theta')$$

$$= P(n_{a/a}, n_{a/o} | n_A, \theta'_a, \theta'_o) P(n_{b/b}, n_{b/o} | n_B, \theta'_b, \theta'_o)$$

$$p(n_{a/a} | n_A, \theta'_a, \theta'_o) = \binom{n_A}{n_{a/a}} \left( \frac{\theta_a^2}{\theta_a^2 + 2\theta'_a \theta'_o} \right)^{n_{a/a}} \left( \frac{2\theta'_a \theta'_o}{\theta_a^2 + 2\theta'_a \theta'_o} \right)^{n_A - n_{a/a}}$$

$$n_{a/a} \leftarrow E_{\theta'}(N_{a/a}) = n_A \left( \frac{\theta_a^2}{\theta_a^2 + 2\theta'_a \theta'_o} \right) \quad n_{a/o} \leftarrow E_{\theta'}(N_{a/o}) = n_A \left( \frac{2\theta'_a \theta'_o}{\theta_a^2 + 2\theta'_a \theta'_o} \right)$$

This is exactly the E-step we used earlier !

33

## The M-step of Gene Counting

The log-likelihood of the completed data of n points:

$$\log P(x, y | \theta) = K + n_{a/a} \log(\theta_a^2) + n_{a/o} \log(2\theta_a \theta_o) +$$

$$n_{b/b} \log(\theta_b^2) + n_{b/o} \log(2\theta_b \theta_o) + n_{a/b} \log(2\theta_a \theta_b) + n_{o/o} \log(\theta_o^2)$$

Taking expectation wrt  $Y = (N_{a/a}, N_{a/o}, N_{b/b}, N_{b/o})$  and using linearity of E yields the function  $Q(\theta | \theta')$  which we need to maximize:

$$E_{\theta'}[\log P(x, y | \theta)] = E_{\theta'}[K] + E_{\theta'}[N_{a/a}] \log(\theta_a^2) + E_{\theta'}[n_{a/o}] \log(2\theta_a \theta_o) +$$

$$E_{\theta'}[n_{b/b}] \log(\theta_b^2) + E_{\theta'}[n_{b/o}] \log(2\theta_b \theta_o) + E_{\theta'}[n_{a/b}] \log(2\theta_a \theta_b) + E_{\theta'}[n_{o/o}] \log(\theta_o^2)$$

34

## The M-step of Gene Counting (Cont.)

We need to maximize the function:

$$f(\theta_a, \theta_b, \theta_o) = n_{a/a} \log(\theta_a^2) + n_{a/o} \log(2\theta_a \theta_o) + n_{b/b} \log(\theta_b^2)$$

$$+ n_{b/o} \log(2\theta_b \theta_o) + n_{a/b} \log(2\theta_a \theta_b) + n_{o/o} \log(\theta_o^2)$$

Under the constraint  $\theta_a + \theta_b + \theta_o = 1$ .

The solution (obtained using Lagrange multipliers) is given by

$$\theta_a = \frac{2n_{a/a} + n_{a/o} + n_{AB}}{2n} \quad \theta_b = \frac{2n_{b/b} + n_{b/o} + n_{AB}}{2n} \quad \theta_o = \frac{2n_o + n_{a/o} + n_{b/o}}{2n}$$

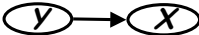
Which matches the M-step we used earlier !

35

## Outline for a different derivation of Gene Counting as an EM algorithm

Define a variable X with values  $x_A, x_B, x_{AB}, x_O$ .

Define a variable Y with values  $y_{a/a}, y_{a/o}, y_{b/b}, y_{b/o}, y_{a/b}, y_{o/o}$ .

Examine the Bayesian network: 

The local probability table for Y is  $P(y_{a/a} | \theta) = \theta_a \theta_a$ ,  $P(y_{a/o} | \theta) = 2\theta_a \theta_o$ , etc.

The local probability table for X given Y is  $P(x_A | y_{a/a}, \theta) = 1$ ,  $P(x_A | y_{b/o}, \theta) = 0$ , etc, only 0's and 1's.

Homework: write down for yourself the likelihood function for n independent points  $x_i, y_i$ , and check that the EM equations match the gene counting equations.

36