Building Bayes Networks:

Parameter Learning

Parameter Learning

Given:
- The graph underlying a graphical model for the domain.
- A database of samples from domain of interest.

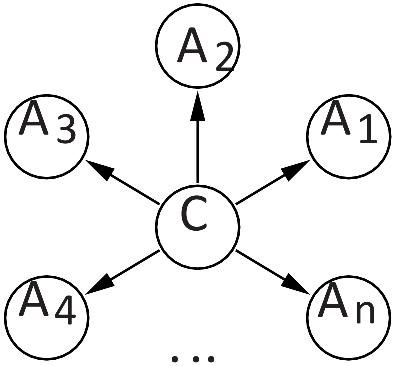
Goal:
- Find „good“ values (estimates) for the numeric parameters (e.g. probabilities) of the model.

Naive Bayes Classifiers

A naive Bayes classifier is a Bayesian network with star-like structure.

The class attribute is the only unconditional attribute.

All other attributes are conditioned on the class C only.



The structure of a naive Bayes classifier is fixed if the attributes have been selected.

The only remaining task is to estimate the parameters of the needed (conditional) probability distributions.

Probabilistic Classification

A **classifier** is an algorithm that assigns a class from a predefined set to a case or object, based on the values of descriptive attributes.
An **optimal** probabilistic classifier assigns the most probable class.

- Let $U = \{A_1, \ldots, A_m\}$ be a set of descriptive attributes with domains $\mathrm{dom}(A_k)$, $1 \leq k \leq m$.
- Let $A_1 = a_1, \ldots, A_m = a_m$ be an instantiation of the descriptive attributes.
- An optimal classifier should assign the class c_i for which

$$P(C = c_i \mid A_1 = a_1, \ldots, A_m = a_m) =$$

$$\max_{j=1}^{n_C} P(C = c_j \mid A_1 = a_1, \ldots, A_m = a_m) =$$

Bayes Rule

$$\max_{j=1}^{n_C} \frac{P(A_1 = a_1, \ldots, A_m = a_m \mid C = c_j) \cdot P(C = c_j)}{P(A_1 = a_1, \ldots, A_m = a_m)} =$$

„Naive“ Assumption

$$\max_{j=1}^{n_C} \frac{P(A_1 = a_1 \mid C = c_j) \times \ldots \times P(A_m = a_m \mid C = c_j) \cdot P(C = c_j)}{P(A_1 = a_1, \ldots, A_m = a_m)}$$

unrealistic, simplifying, but often successful!

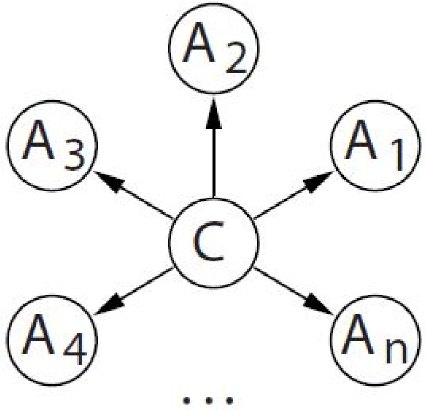
Naive Bayes Classifiers (continued)

Consequence: Manageable amount of data to store.
Store distributions $P\ (C = c_j)$ and $P(A_k = a_k \mid C = c_j)$.

Classification: Compute for all classes c_j

$$\frac{P(A_1 = a_1 \mid C = c_j)\, x. . .x\ P(A_m = a_m \mid C = c_j) \cdot P(C = c_j)}{P(A_1 = a_1, . . . , A_m = a_m)}.$$

and predict the class c_i for which this value is largest.



Decomposition formula:

$$P(C = c\ , A_1 = a_1, . . . , A_n = a_n)$$
$$= P(C = c\ ) \cdot \prod_{j=1}^{n} P(A_j = a_j | C = c\ )$$

Naive Bayes Classifiers: Parameter Estimation

Estimation of Probabilities:

Nominal/Categorical Attributes

$$\hat{P}(A_k = a_k \mid C = c_i) = \frac{\#(A_k = a_k, C = c_i) + \gamma}{\#(C = c_i) + n_{A_k}\gamma}$$

$\#(\varphi)$ is the number of example cases that satisfy the condition φ

n_{A_j} is the number of values of the attribute A_j.

γ is called **Laplace correction**

$\gamma = 0$: Maximum likelihood estimation.

Common choices: $\gamma = 1$ or $\gamma = \frac{1}{2}$.

Laplace correction help to avoid problems with attribute values that do not occur with some class in the given data.

It also introduces a bias towards a uniform distribution.

Naive Bayes Classifiers: Parameter Estimation

Estimation of Probabilities:

Metric/Numeric Attributes: Assume a normal distribution.

$$P(A_k = a_k \mid C = c_i) = \frac{1}{\sqrt{2\pi}\sigma_k(c_i)} \exp\left(-\frac{(a_k - \mu_k(c_i))^2}{2\sigma_k^2(c_i)}\right)$$

Estimate of mean value

$$\hat{\mu}_k(c_i) = \frac{1}{\#(C = c_i)} \sum_{j=1}^{\#(C=c_i)} a_k(j)$$

Estimate of variance

$$\hat{\sigma}_k^2(c_i) = \frac{1}{\xi} \sum_{j=1}^{\#(C=c_i)} (a_k(j) - \hat{\mu}_k(c_i))^2$$

$\xi = \#(C = c_i)$: Maximum likelihood estimation
$\xi = \#(C = c_i) - 1$: Unbiased estimation

Naive Bayes Classifiers: Example

A simple database and estimated (conditional) probability distributions.

No	Sex	Age	Blood pr.	Drug
1	male	20	normal	A
2	female	73	normal	B
3	female	37	high	A
4	male	33	low	B
5	female	48	high	A
6	male	29	normal	A
7	female	52	normal	B
8	male	42	low	B
9	male	61	normal	B
10	female	30	normal	A
11	female	26	low	B
12	male	54	high	A

P (Drug)	A	B
	0.5	0.5

P (Sex | Drug)	A	B
male	0.5	0.5
female	0.5	0.5

P (Age | Drug)	A	B
μ	36.3	47.8
σ^2	161.9	311.0

P (Blood Pr. | Drug)	A	B
low	0	0.5
normal	0.5	0.5
high	0.5	0

Naive Bayes Classifiers: Example

Which Drug for (male,61,normal)?

$P(\text{Drug A}) \cdot P(\text{male} \mid \text{Drug A}) \cdot P(61 \mid \text{Drug A}) \cdot P(\text{normal} \mid \text{Drug A})$

$= 0.5 \cdot 0.5 \cdot 0.004787 \cdot 0.5 = 5.984 \cdot 10^{-4}$

$P(\text{Drug B}) \cdot P(\text{male} \mid \text{Drug B}) \cdot P(61 \mid \text{Drug B}) \cdot P(\text{normal} \mid \text{Drug B})$

$= 0.5 \cdot 0.5 \cdot 0.017120 \cdot 0.5 = 2.140 \cdot 10^{-3}$

$P(\text{Drug A} \mid \text{male, 61, normal}) = 0.219$, **P(Drug B | male, 61, normal) = 0.781**

Decision: B

Which Drug for (female,30,normal)?

$P(\text{Drug A}) \cdot P(\text{female} \mid \text{Drug A}) \cdot P(30 \mid \text{Drug A}) \cdot P(\text{normal} \mid \text{Drug A})$

$= 0.5 \cdot 0.5 \cdot 0.027703 \cdot 0.5 = 3.471 \cdot 10^{-3}$

$P(\text{Drug B}) \cdot P(\text{female} \mid \text{Drug B}) \cdot P(30 \mid \text{Drug B}) \cdot P(\text{normal} \mid \text{Drug B})$

$= 0.5 \cdot 0.5 \cdot 0.013567 \cdot 0.5 = 1.696 \cdot 10^{-3}$

P (Drug A | female, 30, normal) = 0.671 , $P(\text{Drug B} \mid \text{female, 30, normal}) = 0.329$

Decision: A

Using this method a decision can be made for combinations of attribute values, that are not in the data base, i.e. new data.

Naive Bayes Classifiers: Simple Example

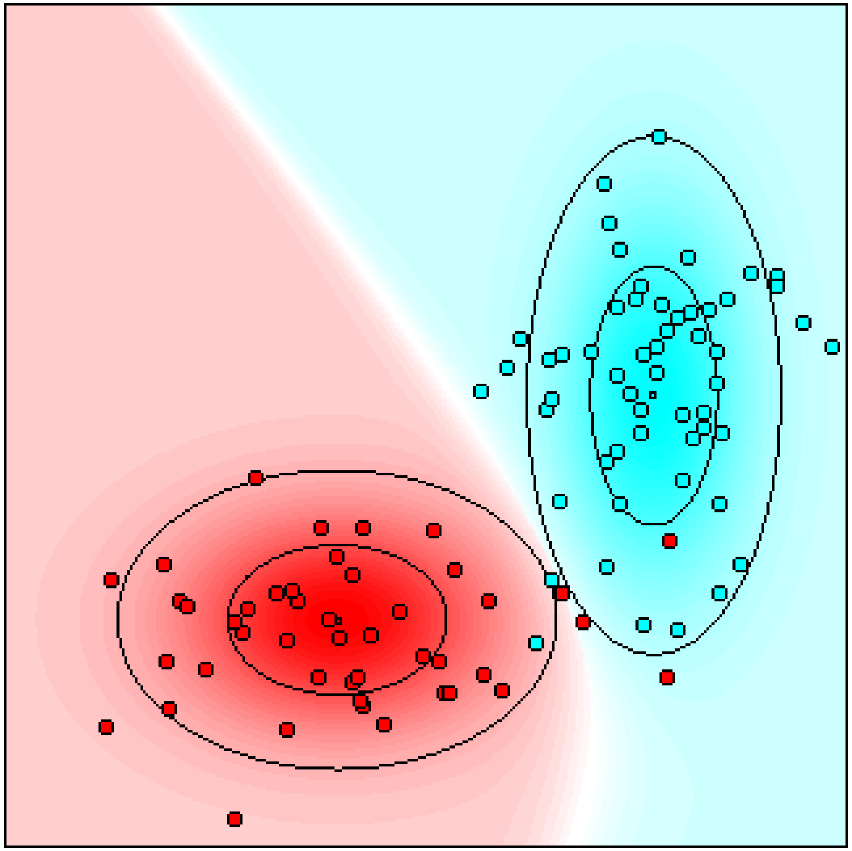
100 labelled data points: red, blue

Two classes red or blue, assumed to be normal distributed

The landscape indicates the classification of all point of the plane, the intensity of the color indicates the probability of the most probable class

The two conditional probabilities are shown
Small squares: mean values
Inner ellipses: one standard deviation
Outer ellipses: two standard deviations

Classes overlap: classification is good, cannot be not perfect



Naive Bayes Classifier

Naive Bayes Classifiers: Simple Example

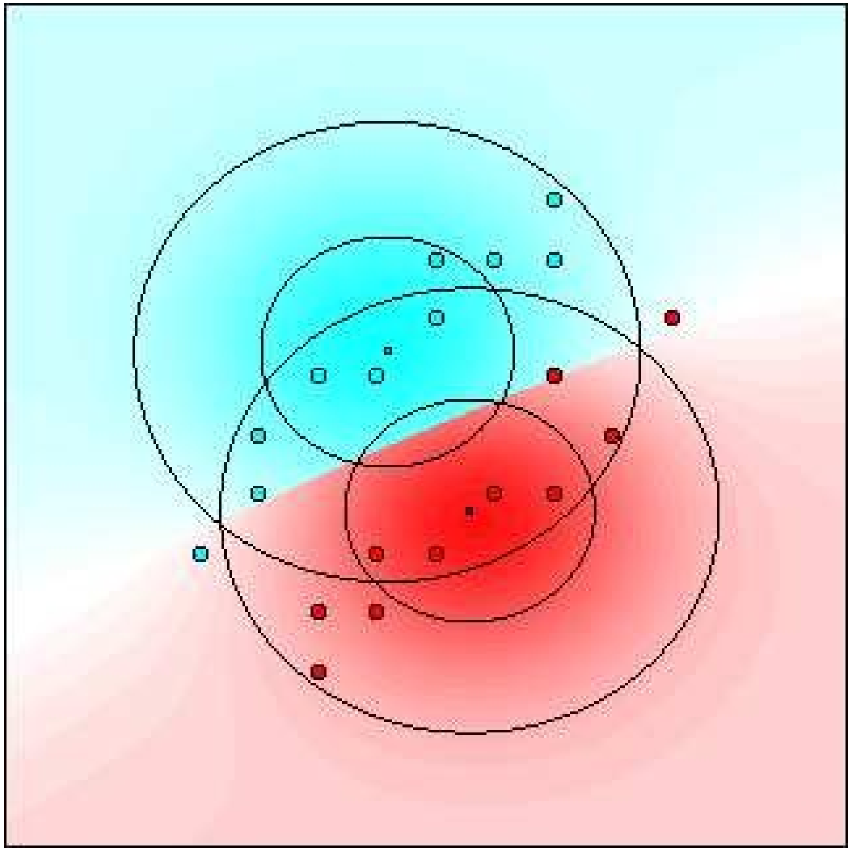
20 labelled data points: red, blue

Two classes red or blue, assumed to be normal distributed

The landscape indicates the classification of all point of the plane, the intensity of the color indicates the probability of the most probable class

The two conditional probabilities are shown
Small squares: mean values
Inner ellipses: one standard deviation
Outer ellipses: two standard deviations

Attributes are NOT conditionally independent given the class, classification still rather good.



Naive Bayes Classifier

Parameter Learning for Bayesian Networks

Probability values can estimated by using methods of inductive statistics:

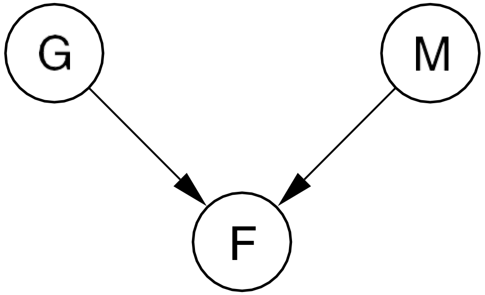
(i) Given a data set and a decomposition

(ii) Estimation of the parameters for the decomposed representation

.

	P(G)
$a_{11} = \mathrm{g}$	
$a_{12} = \overline{\mathrm{g}}$	

	P(M)
$a_{12} = \mathrm{m}$	
$a_{22} = \overline{\mathrm{m}}$	



P(F|G,M)	g, m	$\mathrm{g}, \overline{\mathrm{m}}$	$\overline{\mathrm{g}}, \mathrm{m}$	$\overline{\mathrm{g}}, \overline{\mathrm{m}}$
$a_{31} = \mathrm{f}$				
$a_{32} = \overline{\mathrm{f}}$				

V = {G, M, F}

dom(G) = {g, $\bar{\mathrm{g}}$}

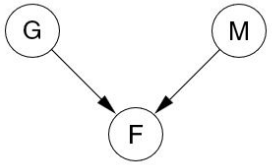
dom(M) = {m, $\bar{\mathrm{m}}$}

dom(F) = {f, $\bar{\mathrm{f}}$}

Parameter Learning using Parameter Estimating Methods

Flu G	$\overline{\mathsf{g}}$	$\overline{\mathsf{g}}$	$\overline{\mathsf{g}}$	$\overline{\mathsf{g}}$	g	g	g	g
Malaria M	$\overline{\mathsf{m}}$	$\overline{\mathsf{m}}$	m	m	$\overline{\mathsf{m}}$	$\overline{\mathsf{m}}$	m	m
Fever F	$\overline{\mathsf{f}}$	f	$\overline{\mathsf{f}}$	f	$\overline{\mathsf{f}}$	f	$\overline{\mathsf{f}}$	f
#	34	6	2	8	16	24	0	10

Database D with 100 entries for 3 attributes.



As the structure given by the graph of the previous slide suggests, the probability of $P(g, m, f)$ can be computed by:

$$P(\mathsf{g}, \mathsf{m}, \mathsf{f}) = P(\mathsf{g})P(\mathsf{m})P(\mathsf{f} \mid \mathsf{g}, \mathsf{m})$$

Estimates for these probabilities can be calculated, e.g. using the database

$$\hat{P}(\mathsf{f} \mid \mathsf{g}, \mathsf{m}) = \frac{\hat{P}(\mathsf{f}, \mathsf{g}, \mathsf{m})}{\hat{P}(\mathsf{g}, \mathsf{m})} = \frac{\frac{\#(\mathsf{g},\mathsf{m},\mathsf{f})}{|D|}}{\frac{\#(\mathsf{g},\mathsf{m})}{|D|}} = \frac{\#(\mathsf{g}, \mathsf{m}, \mathsf{f})}{\#(\mathsf{g}, \mathsf{m})} = \frac{10}{10} = 1.00$$

$$\hat{P}(\mathsf{f} \mid \overline{\mathsf{g}}, \overline{\mathsf{m}}) = \frac{\hat{P}(\mathsf{f}, \overline{\mathsf{g}}, \overline{\mathsf{m}})}{\hat{P}(\overline{\mathsf{g}}, \overline{\mathsf{m}})} = \frac{\frac{\#(\overline{\mathsf{g}},\overline{\mathsf{m}},\mathsf{f})}{|D|}}{\frac{\#(\overline{\mathsf{g}},\overline{\mathsf{m}})}{|D|}} = \frac{\#(\overline{\mathsf{g}}, \overline{\mathsf{m}}, \mathsf{f})}{\#(\overline{\mathsf{g}}, \overline{\mathsf{m}})} = \frac{6}{40} = 0.15$$

Note: Relative frequencies constitute optimal estimators in the case of multinomial distributions. The estimates are calculated for parameters according to the decomposition (i.e. less parameter estimates are needed)

Parameter Learning using the Maximum Likelihood Approach

Maximum likelihood estimation (MLE): General method of estimating the parameters of an (unknown) probability distribution by maximizing a likelihood function, so that under the assumed statistical model the observed data is most probable.

A given set of observations (x_1,…x_n) is considered as a random sample from an unknown population. The goal of maximum likelihood estimation is to make inferences about the population that is most likely to have generated the sample, specifically the joint probability distribution of the random variables.

Associated with each probability distribution is a unique vector of parameters (p_1,…,p_m) that index the probability distribution within a parametric family gives a real-valued function, the likelihood function L_n((x_1,…x_n); (p_1,…,p_m)) . The goal of maximum likelihood estimation is to find the values of the model parameters that maximize the likelihood function over the parameter space.

This optimization task is often very complex. So approximation methods based on iterative procedures are used.

Parameter Learning using the Maximum Likelihood Approach

Example: How biased is an unfair coin?

Probability of tossing H is p, T is 1-p. The goal is to determine the parameter p, which is known to be 1/3, 1/2 or 2/3. The coin is tossed 80 times, we assume an iid sequence (identical-independent-distributed) of random variables. The resulting random sample might be something like x_1 = H, x_2 = T, ..., x_{80} = T, and the count of the number of heads "H" is observed. Suppose the outcome is 49 heads and 31 tails.

Using maximum likelihood approach, the coin with the largest likelihood can be found, given the data that were observed. The distribution of the binomial distribution with sample size equal to 80 and number of H's equal to 49 but for different values of p (the "probability of success"), the likelihood function L_{80} (p,(x_1,…,x_{80})) for p equal to 1/3, 1/2, 2/3 takes the values:

$$\mathrm{P}\,[\,\mathrm{H} = 49 \mid p = \tfrac{1}{3}\,] = \binom{80}{49}(\tfrac{1}{3})^{49}(1-\tfrac{1}{3})^{31} \approx 0.000,$$

$$\mathrm{P}\,[\,\mathrm{H} = 49 \mid p = \tfrac{1}{2}\,] = \binom{80}{49}(\tfrac{1}{2})^{49}(1-\tfrac{1}{2})^{31} \approx 0.012,$$

$$\mathrm{P}\,[\,\mathrm{H} = 49 \mid p = \tfrac{2}{3}\,] = \binom{80}{49}(\tfrac{2}{3})^{49}(1-\tfrac{2}{3})^{31} \approx 0.054.$$

The likelihood is maximized when p = ⅔, and so this is the maximum likelihood estimate (MLE) for *p.*

Likelihood of a Database given a known Bayesian Network structure

General likelihood of a database D given a known Bayesian network structure B_S and the parameters B_P:

$$P(D \mid B_S, B_P) \quad = \quad \prod_{i=1}^{n} \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} \theta_{ijk}^{\alpha_{ijk}}$$

General potential table:

A_i	Q_{i1}	$\cdots$	Q_{ij}	$\cdots$	Q_{iq_i}
a_{i1}	θ_{i11}	$\cdots$	θ_{ij1}	$\cdots$	θ_{iq_i1}
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
a_{ik}	θ_{i1k}	$\cdots$	θ_{ijk}	$\cdots$	θ_{iq_ik}
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
a_{ir_i}	θ_{i1r_i}	$\cdots$	θ_{ijr_i}	$\cdots$	$\theta_{iq_ir_i}$

$$P(A_i = a_{ik} \mid \text{parents}(A_i) = Q_{ij}) = \theta_{ijk}$$

$$\sum_{k=1}^{r_i} \theta_{ijk} = 1$$

Maximum Likelihood Estimate of a Database

Let B_P be the description of the parameters, B_S be the given structure and D the data. The likelihood of the calculated probabilities $P(D \mid B_S, B_P)$ can be computed under presence of three assumptions:

1. The data generation process can be described exactly by a Bayesian network (B_S, B_P)
2. The single tuples of the dataset are independent of each other.
3. All tuples are complete, therefore no missing values hinder the probability inference

The first assumption legitimates the search of an appropriate Bayesian network. The

second assumption is required for an unbiased observation of dataset tuples.

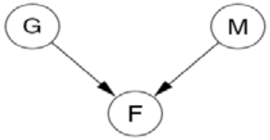
Assumption three ensures the inference of B_P using D and B_S as shown on the previous slides.

Example Maximum Likelihood Estimate of a Database

D: 100 random samples

Flu G Malaria M Fever F	$\overline{\text{g}}$ $\overline{\text{m}}$ $\overline{\text{f}}$	$\overline{\text{g}}$ $\overline{\text{m}}$ f	$\overline{\text{g}}$ m $\overline{\text{f}}$	$\overline{\text{g}}$ m f	g $\overline{\text{m}}$ $\overline{\text{f}}$	g $\overline{\text{m}}$ f	g m $\overline{\text{f}}$	g m f
#	34	6	2	8	16	24	0	10

B_S : The given DAG



B_P : The unknown parameters

P(g), P(m), P(f I g,m),…,P(f I $\overline{\text{g}}$, $\overline{\text{m}}$)

Likelihood Function with data D and parameters

$$P(D \mid B_S, B_P) \quad =$$

$$= \underbrace{\overbrace{P(\mathsf{g},\mathsf{m},\mathsf{f})}^{\text{Case 1}} \cdot \cdots \cdot \overbrace{P(\mathsf{g},\mathsf{m},\mathsf{f})}^{\text{Case 10}}}_{\text{10 times}} \quad \cdots \quad \underbrace{\overbrace{P(\overline{\mathsf{g}},\mathsf{m},\mathsf{f})}^{\text{Case 51}} \cdot \cdots \cdot \overbrace{P(\overline{\mathsf{g}},\mathsf{m},\mathsf{f})}^{\text{Case 58}}}_{\text{8 times}} \quad \cdots \quad \underbrace{\overbrace{P(\overline{\mathsf{g}},\overline{\mathsf{m}},\overline{\mathsf{f}})}^{\text{Case 67}} \cdot \cdots \cdot \overbrace{P(\overline{\mathsf{g}},\overline{\mathsf{m}},\overline{\mathsf{f}})}^{\text{Case 100}}}_{\text{34 times}}$$

$$\| \qquad\qquad\qquad \| \qquad\qquad\qquad \|$$

$$= \underbrace{\overbrace{P(\mathsf{g},\mathsf{m},\mathsf{f})^{10}}} \quad \cdots \quad \underbrace{\overbrace{P(\overline{\mathsf{g}},\mathsf{m},\mathsf{f})^{8}}} \quad \cdots \quad \underbrace{\overbrace{P(\overline{\mathsf{g}},\overline{\mathsf{m}},\overline{\mathsf{f}})^{34}}}$$

$$\| \qquad\qquad\qquad \| \qquad\qquad\qquad \|$$

$$= \overbrace{P(\mathsf{f} \mid \mathsf{g},\mathsf{m})^{10} P(\mathsf{g})^{10} P(\mathsf{m})^{10}} \quad \cdots \quad \overbrace{P(\mathsf{f} \mid \overline{\mathsf{g}},\mathsf{m})^{8} P(\overline{\mathsf{g}})^{8} P(\mathsf{m})^{8}} \quad \cdots \quad \overbrace{P(\overline{\mathsf{f}} \mid \overline{\mathsf{g}},\overline{\mathsf{m}})^{34} P(\overline{\mathsf{g}})^{34} P(\overline{\mathsf{m}})^{34}}$$

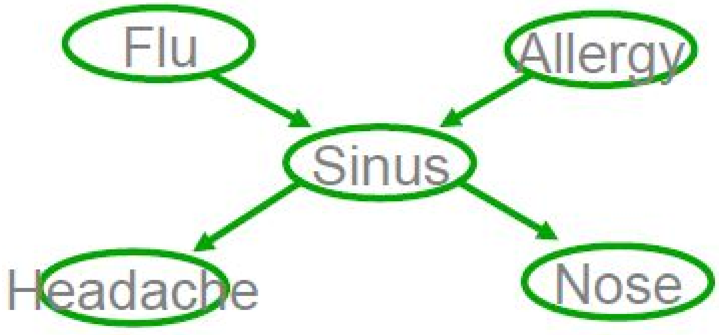
$$= P(\mathsf{f} \mid \mathsf{g},\mathsf{m})^{10} P(\overline{\mathsf{f}} \mid \mathsf{g},\mathsf{m})^{0} P(\mathsf{f} \mid \mathsf{g},\overline{\mathsf{m}})^{24} P(\overline{\mathsf{f}} \mid \mathsf{g},\overline{\mathsf{m}})^{16}$$

$$\cdot P(\mathsf{f} \mid \overline{\mathsf{g}},\mathsf{m})^{8} P(\overline{\mathsf{f}} \mid \overline{\mathsf{g}},\mathsf{m})^{2} P(\mathsf{f} \mid \overline{\mathsf{g}},\overline{\mathsf{m}})^{6} P(\overline{\mathsf{f}} \mid \overline{\mathsf{g}},\overline{\mathsf{m}})^{34}$$

$$\cdot P(\mathsf{g})^{50} P(\overline{\mathsf{g}})^{50} P(\mathsf{m})^{20} P(\overline{\mathsf{m}})^{80}$$

How to maximize the likelihood?

Likelihood Estimate: Fully Observed Data



- Random Variables F,A,S,H and N with two values 0 and 1 each.
- Database of the K observed cases

- Example: Consider learning the parameter

$$\theta_{s|ij} \equiv P(S=1|F=i, A=j)$$

- Max Likelihood Estimate is

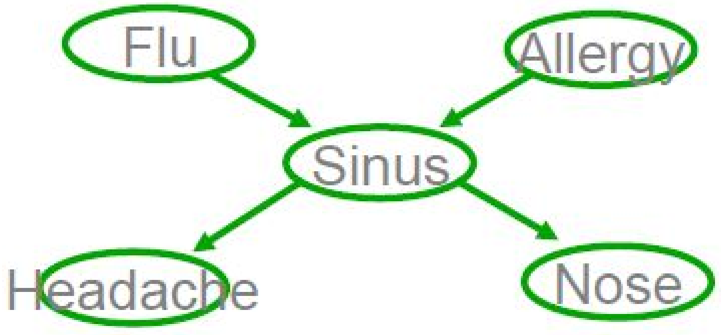
$$\theta_{s|ij} = \frac{\sum_{k=1}^{K} \delta(f_k = i, a_k = j, s_k = 1)}{\sum_{k=1}^{K} \delta(f_k = i, a_k = j)}$$

k^th training example

$\delta(x)$ = 1 if x=true,
= 0 if x=false

Kronecker-Delta

Likelihood Estimate: Fully Observed Data



- Random Variables F,A,S,H and N with two values 0 and 1 each.
- Database of the K fully observed cases

- Example: Consider learning the parameter

$$\theta_{s|ij} \equiv P(S=1|F=i, A=j)$$

- Max Likelihood Estimate is

$$\theta_{s|ij} = \frac{\sum_{k=1}^{K} \delta(f_k = i, a_k = j, s_k = 1)}{\sum_{k=1}^{K} \delta(f_k = i, a_k = j)}$$

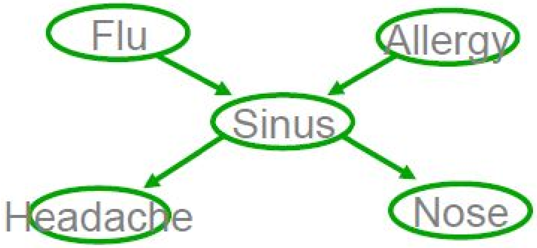
k^{th} training example

δ(x) = 1 if x=true,
= 0 if x=false

Kronecker-Delta

Why?

Likelihood Estimate: How to solve the maximization task?



- Random Variables F,A,S,H and N with two values 0 and 1
- Database of fully observed K cases

Log likelihood function

$$P(data|\theta) = \prod_{k=1}^{K} P(f_k, a_k, s_k, h_k, n_k)$$

$$P(data|\theta) = \prod_{k=1}^{K} P(f_k)P(a_k)P(s_k|f_k a_k)P(h_k|s_k)P(n_k|s_k)$$

$$\log P(data|\theta) = \sum_{k=1}^{K} \log P(f_k) + \log P(a_k) + \log P(s_k|f_k a_k) + \log P(h_k|s_k) + \log P(n_k|s_k)$$

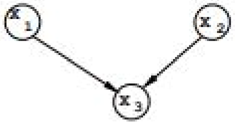
Learning a parameter via maximization

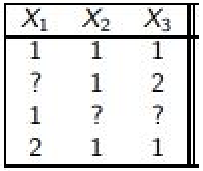
$$\frac{\partial \log P(data|\theta)}{\partial \theta_{s|ij}} = \sum_{k=1}^{K} \frac{\partial \log P(s_k|f_k a_k)}{\partial \theta_{s|ij}}$$

$$\theta_{s|ij} = \frac{\sum_{k=1}^{K} \delta(f_k = i, a_k = j, s_k = 1)}{\sum_{k=1}^{K} \delta(f_k = i, a_k = j)}$$

Useful methods for Parameter Learning in Bayesian Networks

Missing Values





X_1	X_2	X_3
1	1	1
?	1	2
1	?	?
2	1	1

To deal with missing values, we need to make the **missing at random (MAR) assumption**:
Actual value of X and the event X-is-missing are conditionally independent given other observed variables:

P(X|X-is-missing, other observed variables) = P(X|other observed variables)

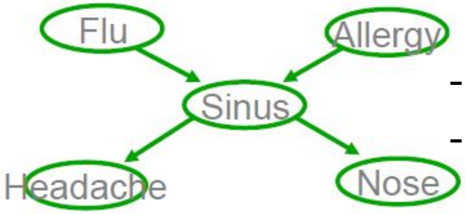
The assumption is sometimes not true. However, it can be made often true by introducing an auxiliary variable.

Adaptation

Often the model is incomplete, the modeled domain is drifting over time, or the model quite simple does not reflect the modeled domain properly**.**

Sequential updating, also known as **adaptation** or **sequential learning**, makes it possible to update and improve the conditional probability distribution for a domain as observations are made.

Likelihood Estimate: Partly Observed Data



- Random Variables F,A,S,H and N with two values 0 and 1
- Data base of cases with values in F,A,N,H , but no observations for S

X all observed variable values (over all examples) , **Z** all unobserved variable values

Maximum Likelihood cannot be calculated in the case of missing values. Instead the **expectation-maximization (EM) algorithm** is used.
The EM tries to optimize $E_{Z|X,\theta}[\log P(X,Z|\theta)]$

In the above example:

$$\log P(X,Z|\theta) = \sum_{k=1}^{K} \log P(f_k) + \log P(a_k) + \log P(s_k|f_k a_k) + \log P(h_k|s_k) + \log P(n_k|s_k)$$

$$E_{P(Z|X,\theta)} \log P(X,Z|\theta) \;=\; \sum_{k=1}^{K}\sum_{i=0}^{1} P(s_k = i|f_k, a_k, h_k, n_k)$$
$$[logP(f_k) + \log P(a_k) + \log P(s_k|f_k a_k) + \log P(h_k|s_k) + \log P(n_k|s_k)]$$

Likelihood Estimate: Partly Observed Data

EM is a general procedure for learning from partly observed data

Given observed variables X, unobserved Z

Iterate until convergence:

- E Step: Use X and current θ to calculate $P(Z|X,\theta)$
- M Step: Replace current θ by

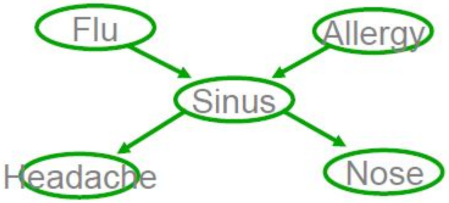
$$E_{P(Z|X,\theta)}[\log P(X,Z|\theta')]$$

Guaranteed to find local maximum.
Each iteration increases $E_{P(Z|X,\theta)}[\log P(X,Z|\theta')]$

Likelihood Estimate: Partly Observed Data

EM and estimating $\theta_{s|ij}$

observed X = {F,A,H,N}, unobserved Z={S}



E step: Calculate P(Z_k|X_k; θ) for each training example, k

$$P(S_k = 1|f_k a_k h_k n_k, \theta) = E[s_k] = \frac{P(S_k = 1, f_k a_k h_k n_k|\theta)}{P(S_k = 1, f_k a_k h_k n_k|\theta) + P(S_k = 0, f_k a_k h_k n_k|\theta)}$$

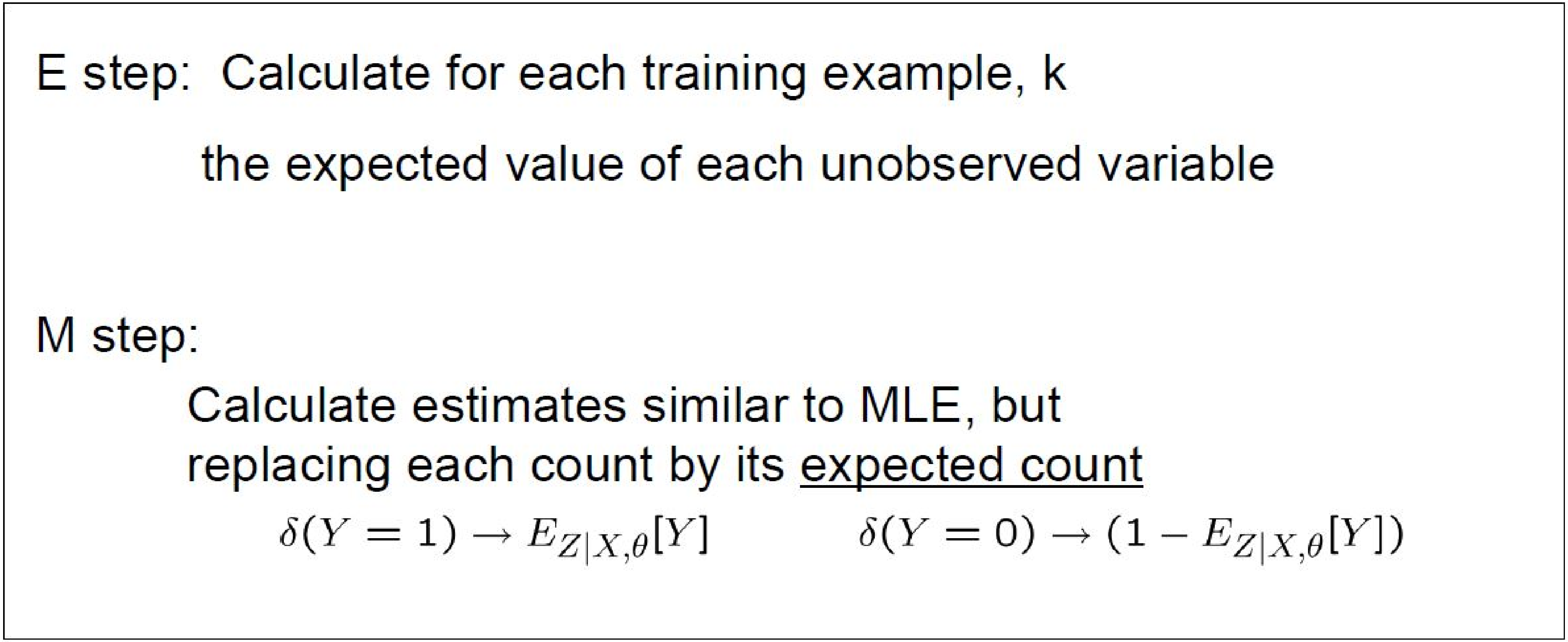
M step: update all relevant parameters. For example:

$$\theta_{s|ij} \leftarrow \frac{\sum_{k=1}^{K} \delta(f_k = i, a_k = j)\ E[s_k]}{\sum_{k=1}^{K} \delta(f_k = i, a_k = j)}$$

Recall MLE was: $\theta_{s|ij} = \dfrac{\sum_{k=1}^{K} \delta(f_k = i, a_k = j, s_k = 1)}{\sum_{k=1}^{K} \delta(f_k = i, a_k = j)}$

Likelihood Estimate: Partly Observed Data

More general situation: Given observed set X, unobserved set Z of Boolean values



E step: Calculate for each training example, k

the expected value of each unobserved variable

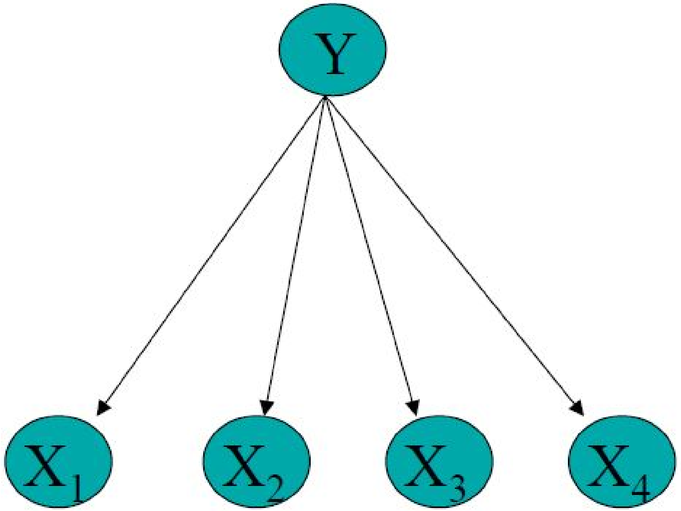
M step:

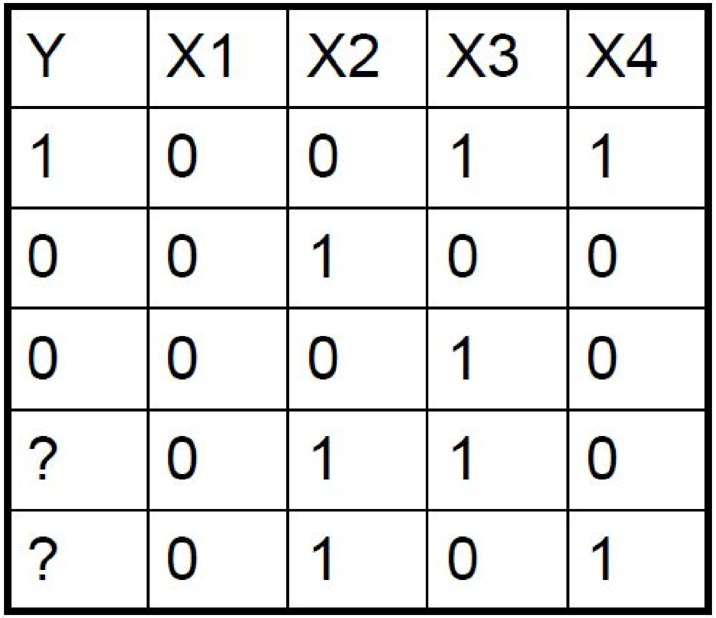
Calculate estimates similar to MLE, but replacing each count by its expected count

$$\delta(Y=1) \rightarrow E_{Z|X,\theta}[Y] \qquad \delta(Y=0) \rightarrow (1 - E_{Z|X,\theta}[Y])$$

Example: Naive Bayes Classifier with Unlabeled Data

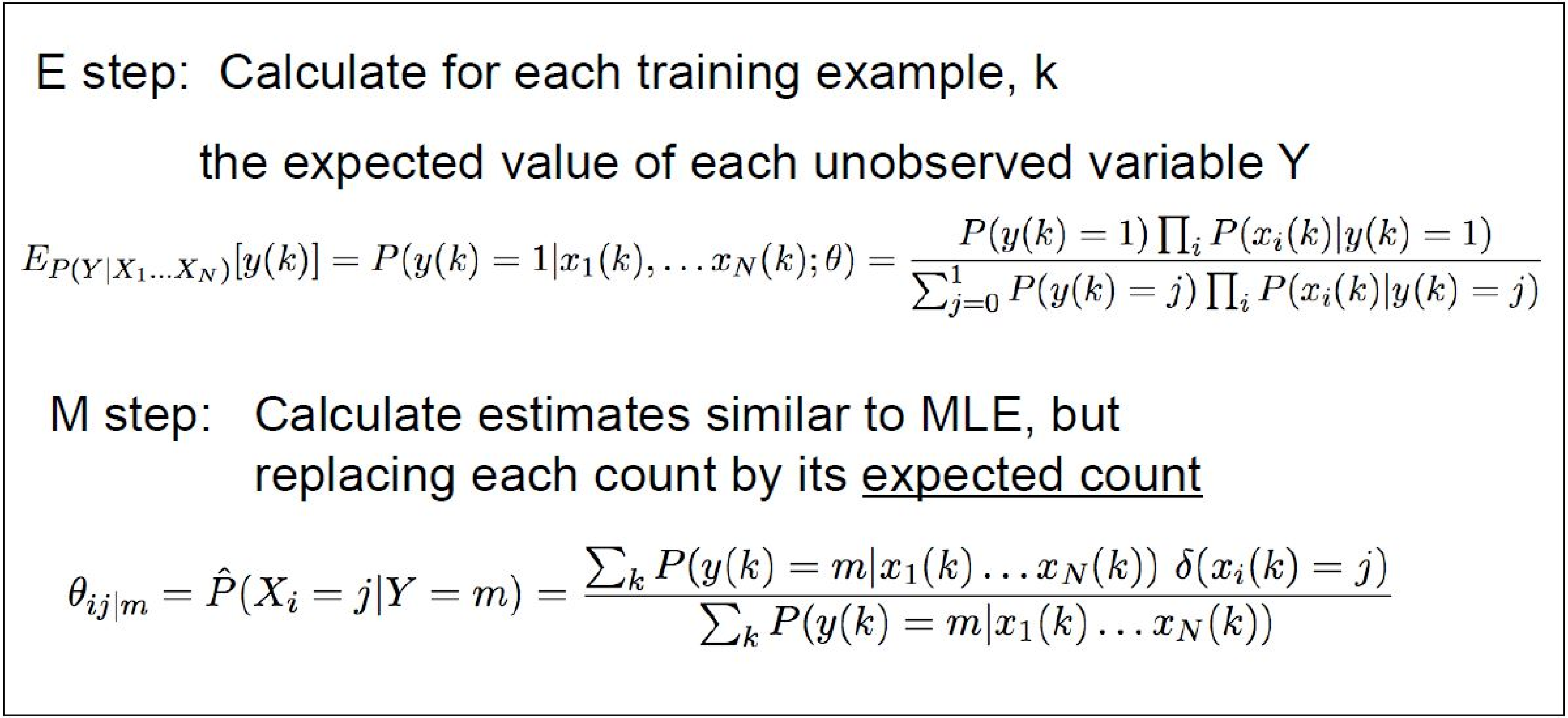
Learn P(Y|X)





Y	X1	X2	X3	X4
1	0	0	1	1
0	0	1	0	0
0	0	0	1	0
?	0	1	1	0
?	0	1	0	1

Example: Naive Bayes Classifier with Unlabeled Data



E step: Calculate for each training example, k

the expected value of each unobserved variable Y

$$E_{P(Y|X_1 \ldots X_N)}[y(k)] = P(y(k) = 1 | x_1(k), \ldots x_N(k); \theta) = \frac{P(y(k) = 1) \prod_i P(x_i(k) | y(k) = 1)}{\sum_{j=0}^{1} P(y(k) = j) \prod_i P(x_i(k) | y(k) = j)}$$

M step: Calculate estimates similar to MLE, but replacing each count by its expected count

$$\theta_{ij|m} = \hat{P}(X_i = j | Y = m) = \frac{\sum_k P(y(k) = m | x_1(k) \ldots x_N(k)) \; \delta(x_i(k) = j)}{\sum_k P(y(k) = m | x_1(k) \ldots x_N(k))}$$

MLE would be: $\hat{P}(X_i = j | Y = m) = \frac{\sum_k \delta((y(k) = m) \wedge (x_i(k) = j))}{\sum_k \delta(y(k) = m)}$