PRINCIPLES OF MACHINE LEANING

Slides partially based on :

-David Barber's slides for the BRML book

-Tinne De Laet & Luc De Raedt's slides for the UAI course at KU Leuven

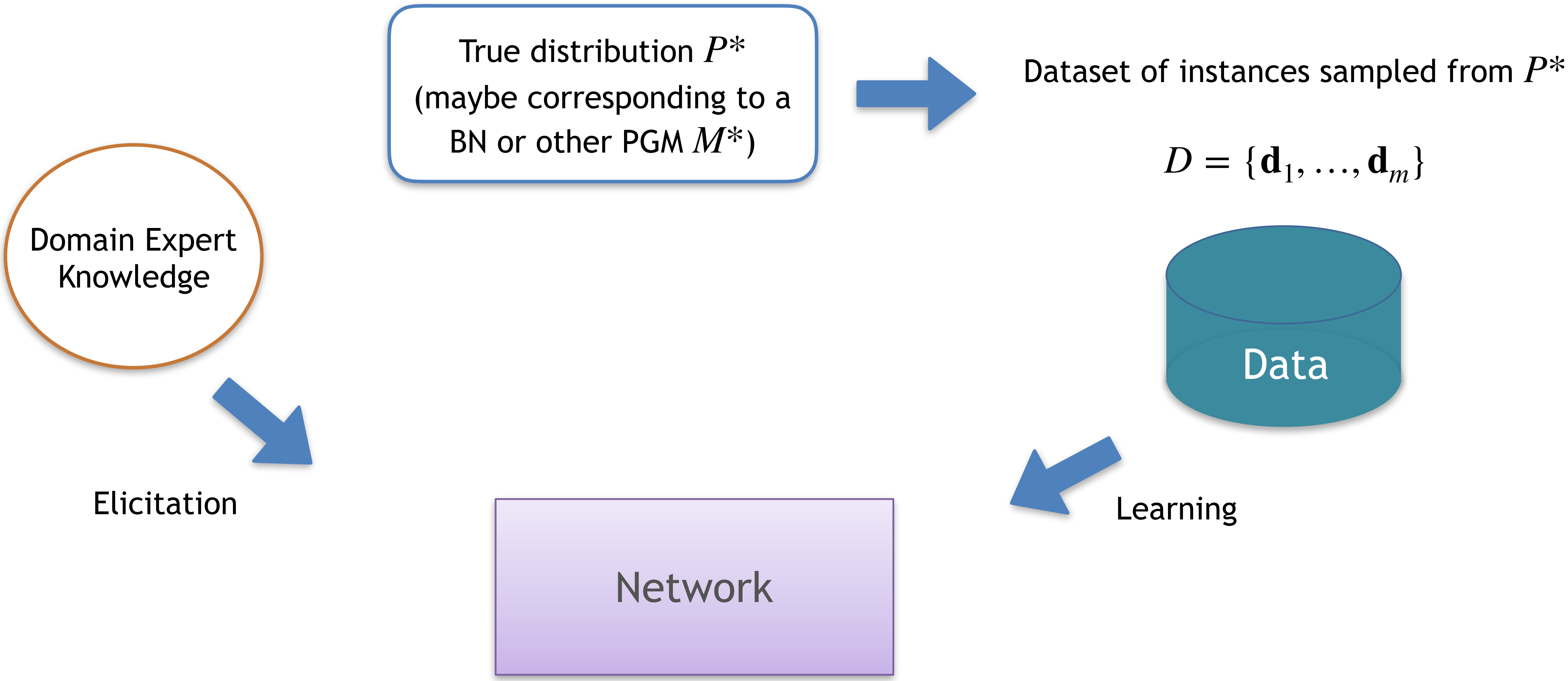
Looking forward

- How to learn Bayesian networks from data?
  - Given the graph, learn the **parameters**;
  - Learn both the **graph structure** & the **parameters**;
  - Learning as **Inference**
- Probabilistic models **involving time**
- Probabilistic models involving **objects** and r**elations**

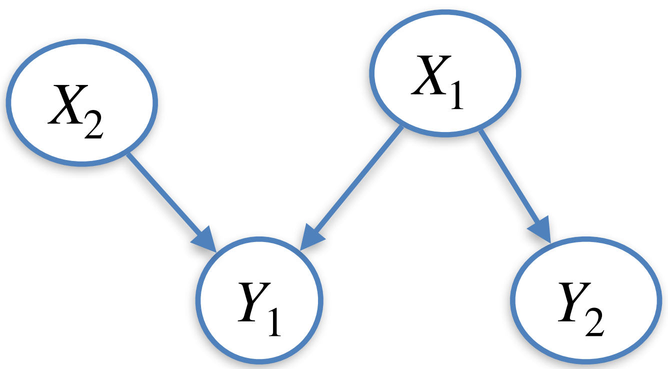
Learning a probabilistic Model from Data

- Why learn a probabilistic model **from data**?
- What are the **possible scenarios** in which this learning problem might arise?

Learning


True distribution P^* (maybe corresponding to a BN or other PGM M^*)
Dataset of instances sampled from P^*
$D = \{\mathbf{d}_1, \ldots, \mathbf{d}_m\}$
Domain Expert Knowledge
Data
Elicitation
Learning
Network

Known Structure and Complete Data



X_1	X_2	Y_1	Y_2
0	1	0	0
1	0	1	0
1	1	1	0
1	1	0	0
1	0	1	1
0	1	1	1
0	0	0	1

Learning

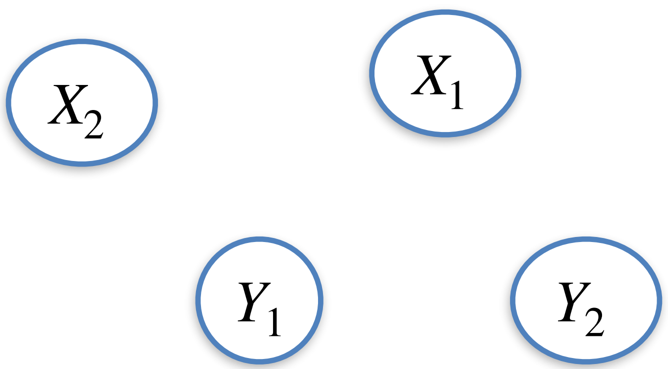
$P(Y_2 \mid X_1)$

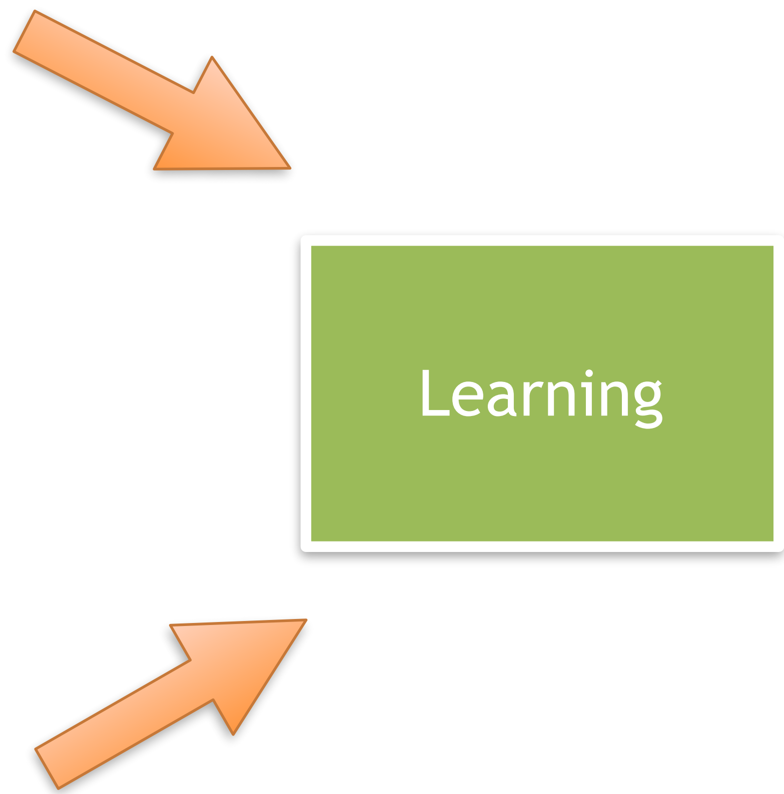
X_1	$Y_2 = 0$	$Y_2 = 1$
0	0	1
1	0.5	0.5

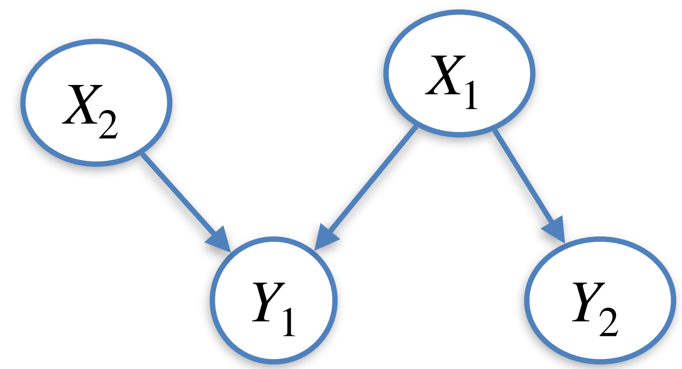
$P(Y_1 \mid X_1, X_2)$

X_1	X_2	$Y_1 = 0$	$Y_1 = 1$
0	0	0	1
0	1	0.5	0.5
1	0	0.7	0.3
1	1	0.25	0.75

Unknown Structure and Complete Data







$P(Y_2 \mid X_1)$

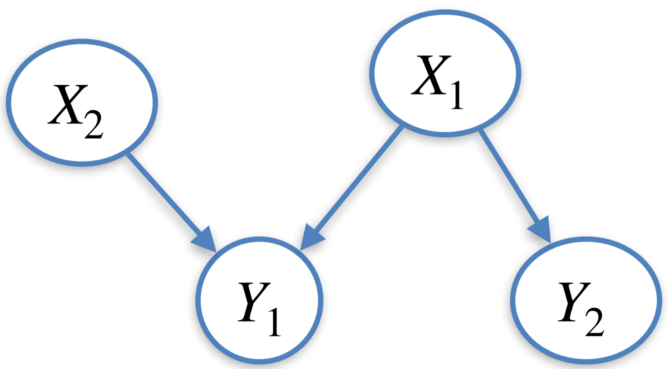
X_1	$Y_2 = 0$	$Y_2 = 1$
0	0	1
1	0.5	0.5

X_1	X_2	Y_1	Y_2
0	1	0	0
1	0	1	0
1	1	1	0
1	1	0	0
1	0	1	1
0	1	1	1
0	0	0	1

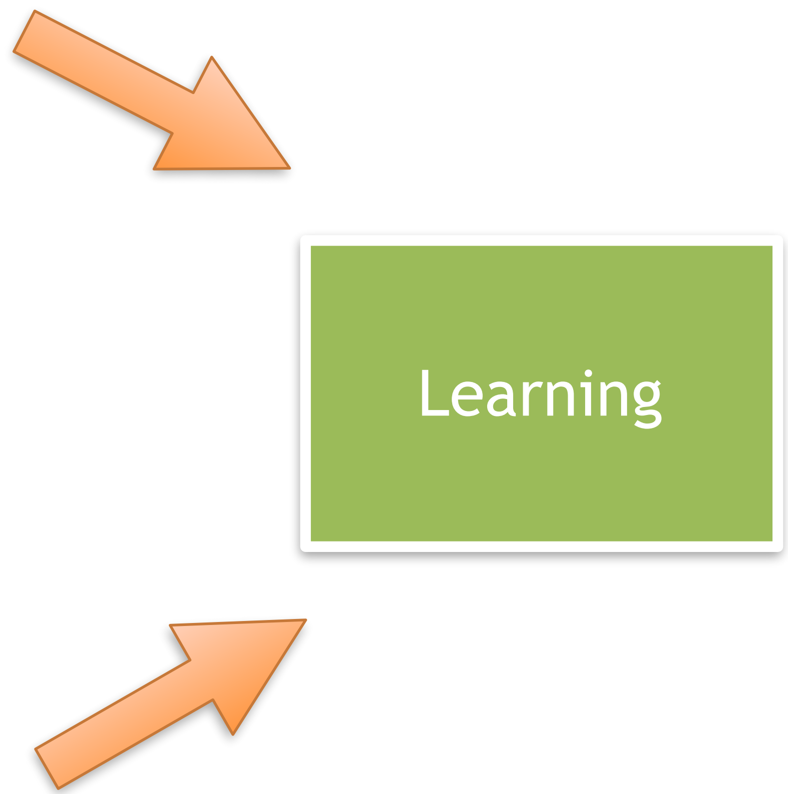
$P(Y_1 \mid X_1, X_2)$

X_1	X_2	$Y_1 = 0$	$Y_1 = 1$
0	0	0	1
0	1	0.5	0.5
1	0	0.7	0.3
1	1	0.25	0.75

Known Structure and Incomplete Data



X_1	X_2	Y_1	Y_2
0	1	0	0
1	0	1	?
1	1	1	0
1	?	0	0
1	0	1	1
0	1	?	1
?	0	0	1



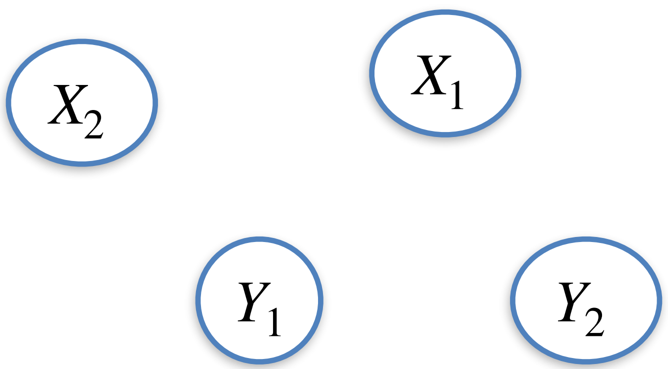
$P(Y_2 \mid X_1)$

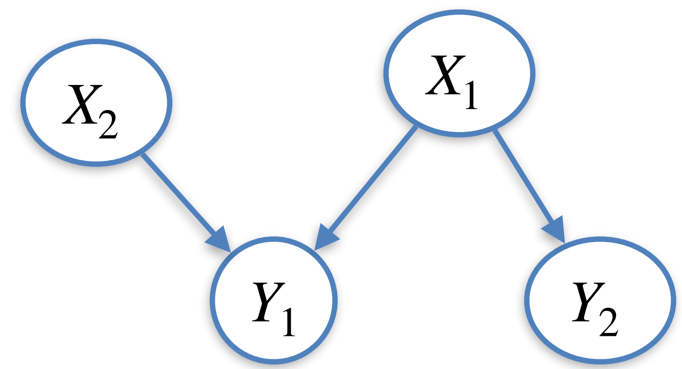
X_1	$Y_2 = 0$	$Y_2 = 1$
0	0	1
1	0.5	0.5

$P(Y_1 \mid X_1, X_2)$

X_1	X_2	$Y_1 = 0$	$Y_1 = 1$
0	0	0	1
0	1	0.5	0.5
1	0	0.7	0.3
1	1	0.25	0.75

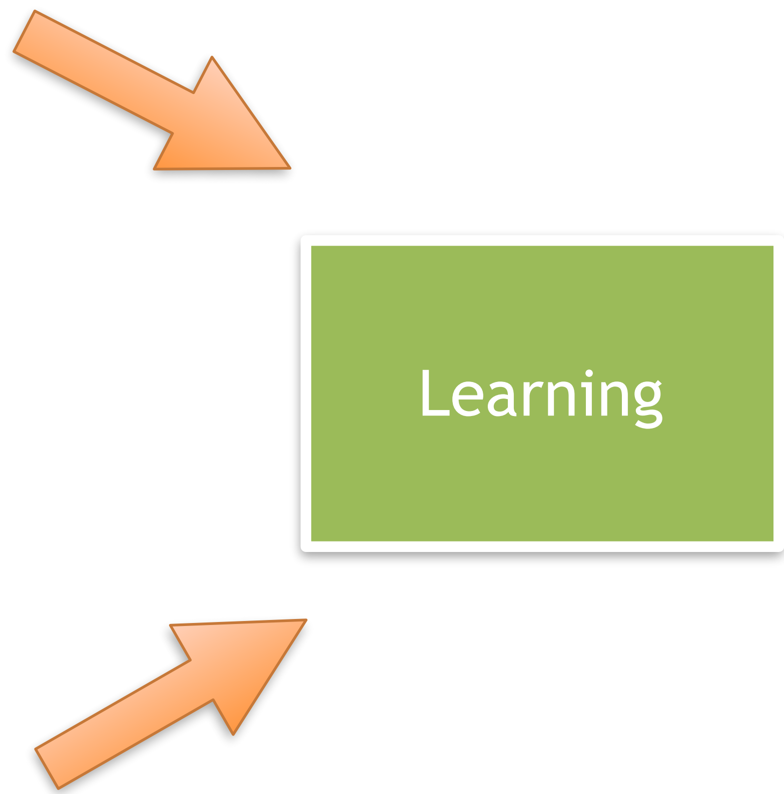
Unknown Structure and Incomplete Data





$P(Y_2 \mid X_1)$

X_1	$Y_2 = 0$	$Y_2 = 1$
0	0	1
1	0.5	0.5

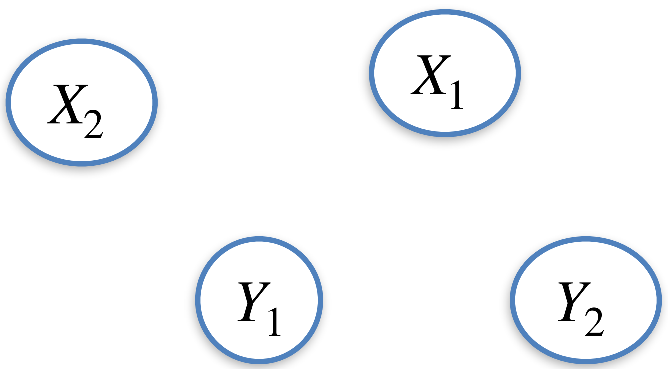


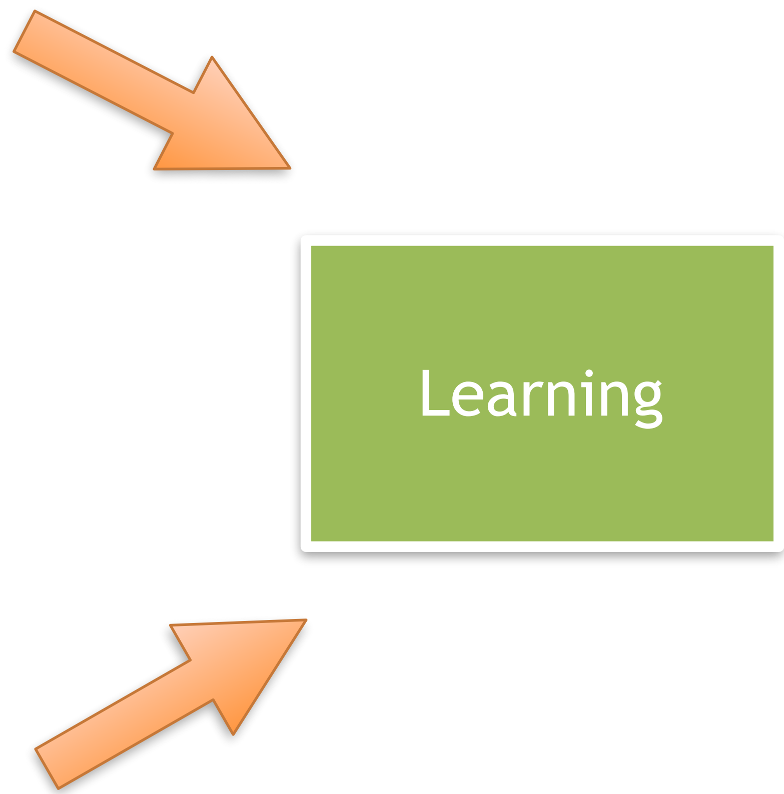
X_1	X_2	Y_1	Y_2
0	1	0	0
1	0	1	?
1	1	1	0
1	?	0	0
1	0	1	1
0	1	?	1
?	0	0	1

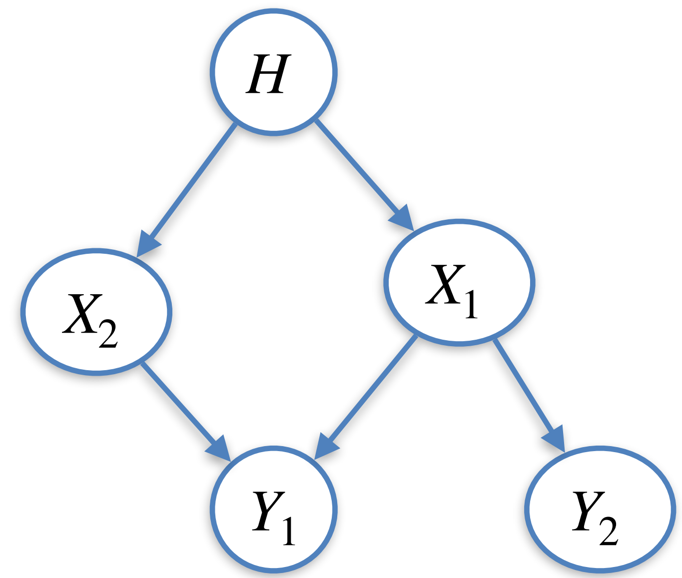
$P(Y_1 \mid X_1, X_2)$

X_1	X_2	$Y_1 = 0$	$Y_1 = 1$
0	0	0	1
0	1	0.5	0.5
1	0	0.7	0.3
1	1	0.25	0.75

Latent Variables, Incomplete Data







X_1	X_2	Y_1	Y_2
0	1	0	0
1	0	1	?
1	1	1	0
1	?	0	0
1	0	1	1
0	1	?	1
?	0	0	1

$P(Y_2 \mid X_1)$

X_1	$Y_2 = 0$	$Y_2 = 1$
0	0	1
1	0.5	0.5

$P(Y_1 \mid X_1, X_2)$

X_1	X_2	$Y_1 = 0$	$Y_1 = 1$
0	0	0	1
0	1	0.5	0.5
1	0	0.7	0.3
1	1	0.25	0.75

H is not observed, but is relevant in the final model

Learning Tasks (1)

- Goal: Answer general probabilistic queries about new instances
- Simple metric: Training set (Data) likelihood

$$P(D \mid M) = \prod_{i}^{N} p(d_i \mid M)$$ **(assuming i.i.d)**

- But we really care about new data
  - Generalisation performance: Evaluate on test set likelihood

Learning Tasks (2)

- **Goal:** Inferring the structure of the model (knowledge discovery)
  - Discovering dependencies
  - e.g. we might be able to infer the directionality of the edges in BNs,
  - existence and location of latent variables
- Often train using likelihood
- Evaluate by comparing to prior knowledge

Learning Tasks (3)

- Goal: Specific prediction task on **new instances**
  - ✳ Predict target variables Y from observed variables X

    e.g. classification
- One cares about specialised objective (e.g. accuracy)
- Convenient to **select model** optimising:
  - Likelihood $\prod_i p(d_i \mid M)$
  - Conditional likelihood $\prod_i P(\mathbf{y_i} \mid \mathbf{x_i} \mid M)$
- Important to evaluate model on “true” objective over test data

Why learning PGM

- Predictions of **structured objects**: sequences, graphs, trees
- **Exploit correlations** between several predicted variables
- Can incorporate **prior knowledge** into models
- Learning single model for **multiple tasks**
- Framework for **knowledge discovery**

Learning Bayesian Networks Parameters

Parameter Estimation with fully observable Data

- Parameter values with higher likelihood are **more likely to generate the observed data**
- We can use the likelihood function as our measure of quality for different parameter values and select the parameter value that maximises the likelihood;
- Maximum likelihood estimator (MLE)

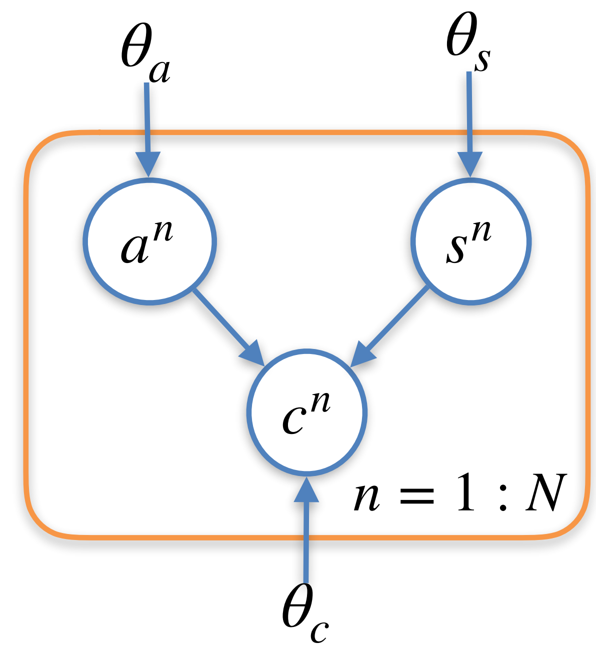
$$\theta^{ML} = \arg\max_{\theta} P(D \mid \theta)$$

Parameter estimation with Complete Data

BN Parameter Estimation with fully observable Data (Example)

- Relationship between exposure to asbestos (a), being a smoker (s) an the incidence of lung cancer (c)
  $dom(a) = \{0,1\}, \quad dom\{0,1\}, \quad dom = \{0,1\}$

$$p(a, s, c) = p(c \mid a, s)p(a)p(s)$$



- Given a list of patient records, where **each row represent a patient's data**

a	s	c
1	1	1
1	0	0
0	1	1
0	1	0
1	1	1
0	0	0
1	0	1

i.i.d

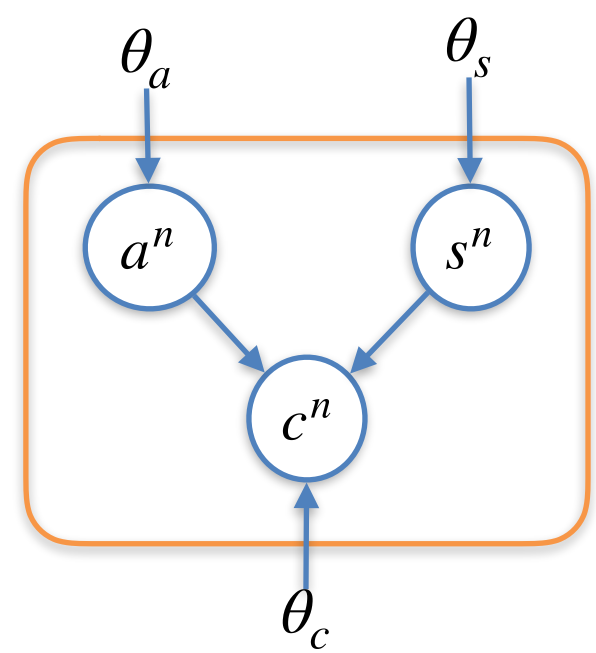
- **Prediction quality:** the likelihood of the data:

$$p(a^n, s^n, c^n \mid \theta) = p(c^n \mid a^n, s^n, \theta_c)p(a^n \mid \theta_a)p(s^n \mid \theta_s)$$

$$p(D \mid \theta) = \prod_{n=1}^{N} p(a^n, s^n, c^n)$$

- Find θ maximising likelihood: MLE

BN Parameter Estimation with fully observable Data (Example)



- To learn the table entries $P(c \mid a, s)$ we **count** the number data instances where c **is in state 1,** for each of the **4 parental states** of a and s:

$$p(c=1 \mid a=0, s=0) = \theta_c^{(0,0)} = 0,$$
$$p(c=1 \mid a=0, s=1) = \theta_c^{(0,1)} = 0.5,$$
$$p(c=1 \mid a=1, s=0) = \theta_c^{(1,0)} = 0.5$$
$$p(c=1 \mid a=1, s=1) = \theta_c^{(1,1)} = 1$$

a	s	c
1	1	1
1	0	0
0	1	1
0	1	0
1	1	1
0	0	0
1	0	1

- Similarly, based on counting
$p(a=1) = \theta_a = 4/7$, and
$p(s=1) = \theta_s = 4/7$

MLE for Bayesian Networks

- Likelihood for BN with variables $X_1, \ldots, X_M$, given data D with samples $\mathbf{d_1}, \ldots, \mathbf{d_N}$.

$$L(\theta \mid D) = \prod_{i=1}^{N} p(\mathbf{d}_i) \mid \theta)$$

$$L(\theta \mid D) = \prod_{n=1}^{N} \prod_{m=1}^{M} p(x_m^n \mid parents(x_m^n), \theta)$$

$$= \prod_{m=1}^{M} \prod_{n=1}^{N} p(x_m^n \mid parents(x_m^n), \theta)$$

$$= \prod_{m=1}^{M} L_m(D \mid \theta)$$

- If $\theta_{X_m | parents(X_m)}$ are disjoints then MLE can be computed by maximising each local likelihood separately

Summary

- Maximum Likelihood in general corresponds to the intuitive use of '**counting**' to set tables
- Convenient to assume global **parameter independence** since then the posterior factorises over the tables (**assuming i.i.d.**)
- Convenient also to assume **local parameter independence** of each conditional since then the posterior table factorises over its parental states.
- Table entries θ can be **learned** by considering only **local information**,
- The **maximum-likelihood parameter learning** problem for a Bayesian network **decomposes into separate learning problems**, one for each parameter

LEARNING WITH INCOMPLETE DATA

Overview

Hidden Variables and Missing Data

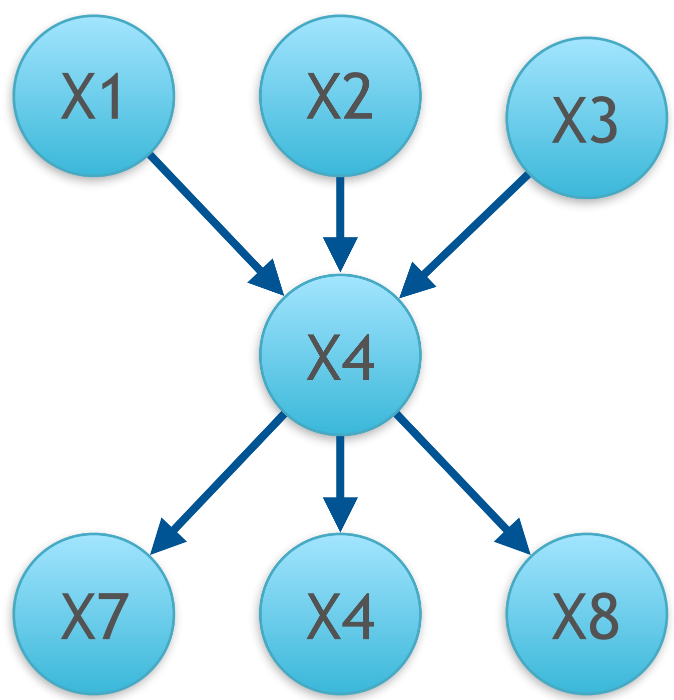
Missing Data – Partially Observed Data
In practice **data entries** are often **missing** resulting in incomplete information to specify a likelihood.

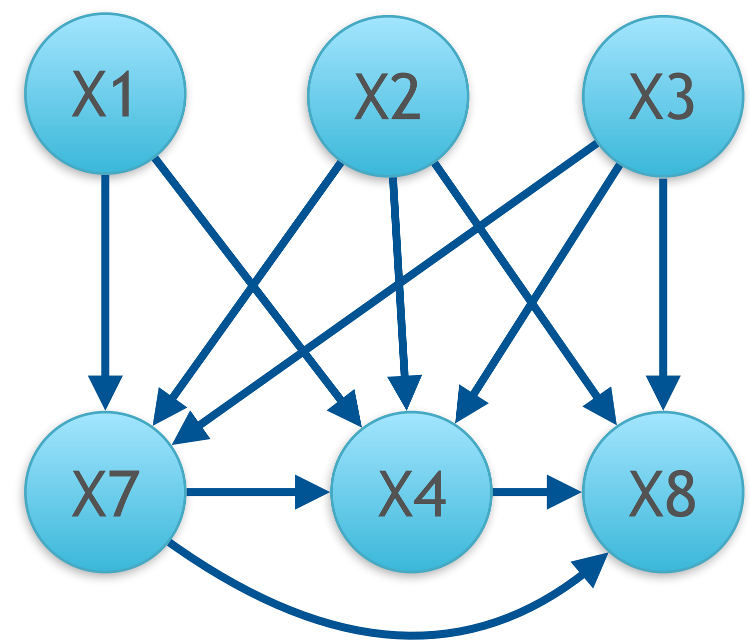
Observational Variables

- **visible:** we actually know the state
- **missing:** we would nominally know their state, but are missing for a particular datapoint.

Latent Variables: Variables that are essential for the model description but are **never observed**.

Latent Variables in BNs





- Hidden or latent variables which are not observable in the data are available for learning.

- They can **dramatically reduce the number of parameters** required to specify a BN

  - Reduce the **amount of data** needed to lean the parameters

Modelling Missing Data Mechanism

- Set of random variables defining our model

$$\mathbf{X} = \{X_1, \ldots, X_n\}$$

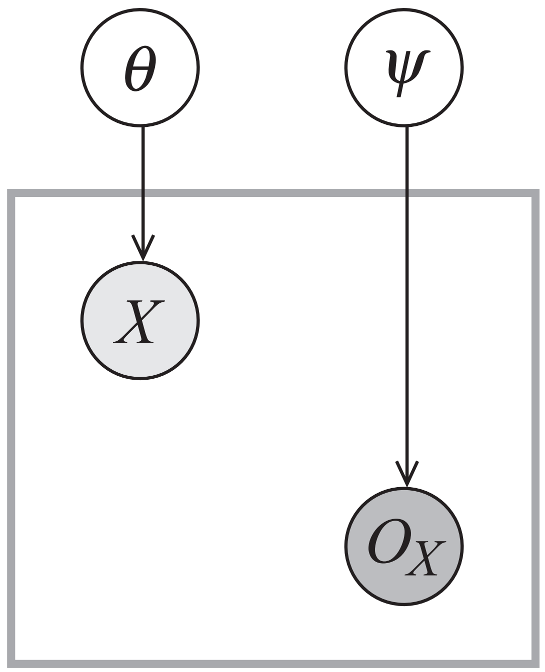
- Set of observability variables (which are always observed)

$$\mathbf{O} = \{O_1, \ldots, O_n\} \text{ such that } O_i = \begin{cases} 1 & \textbf{if } X_i \textbf{ is observed} \\ 0 & \textbf{otherwise} \end{cases}$$

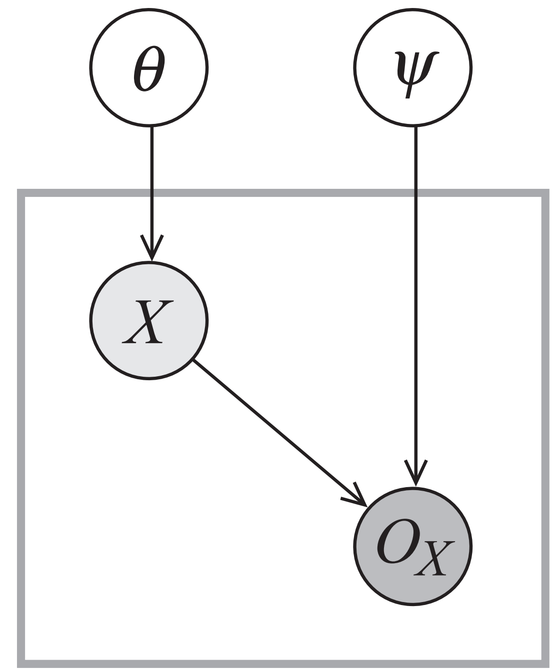
- New set of random variables that are always observed

$$dom(Y_i) = dom(X_i) \cup \{\,?\,\} \qquad Y_i = \begin{cases} X_i & \textbf{if } O_i = 1 \\ ? & \textbf{if } O_i = 0 \end{cases}$$

Modelling Missing Data Mechanism



Random missing values



Deliberate missing values

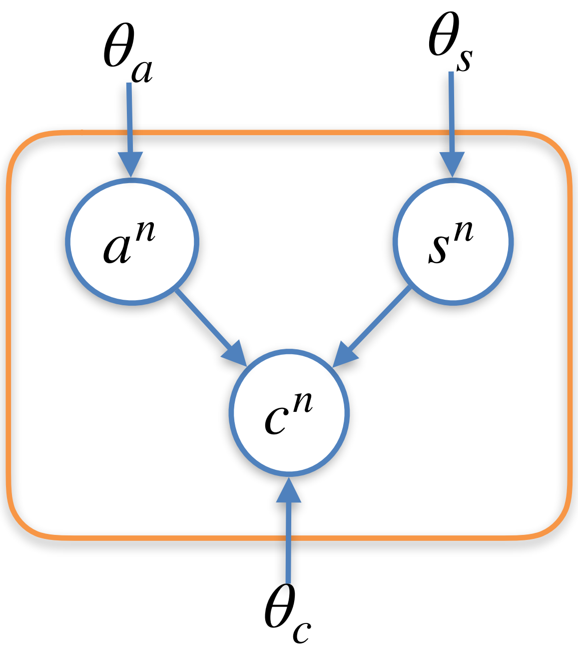
- If the mechanisms by which the data is missing depends only on the visible states we ignore the missing data and focus only on the marginal likelihood to asses parameters

Missing at Random (MAR)

Fully Observed v.s. Missing Data

- The likelihood for complete data

$$p(v^n \mid \theta) = p(a^n, s^n, c^n \mid \theta)$$
$$= p(c^n \mid a^n, s^n, \theta_c)p(a^n \mid \theta_a)p(s^n \mid \theta_s)$$



- Decomposes by variables

- Decomposes within CPDs

a	s	c
1	1	1
1	0	0
0	1	1
0	1	0
1	1	1
0	0	0
1	0	1

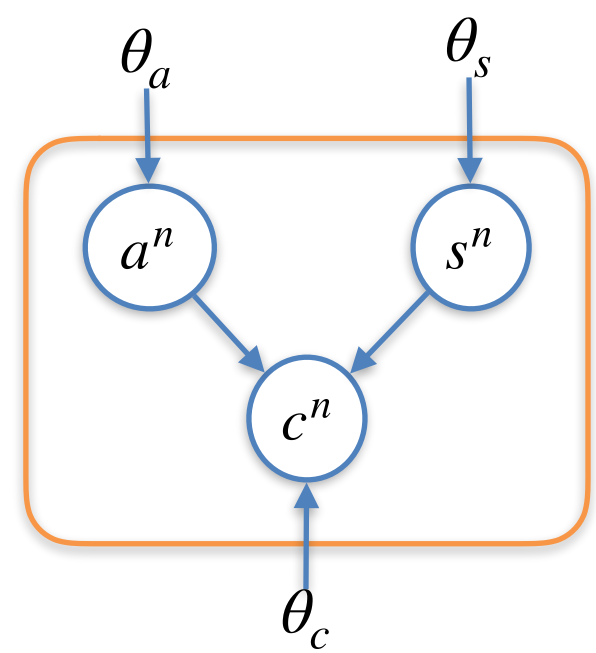
Fully Observed v.s. Missing Data

- Likelihood for incomplete data

$$L(D \mid \theta) = p(D \mid \theta) = \prod_{i=1}^{7} p(\mathbf{d_i} \mid \theta)$$

$$= p(s = 1, c = 1) \times p(a = 1, s = 0, c = 0) \times \ldots \times p(a = 1, s = 0)$$

$$= \left(\sum_{x \in dom(a)} p(x, s = 1, c = 1)) \right) \times p(a = 1, s = 0, c = 0) \times \ldots \times p(a = 1, s = 0)$$

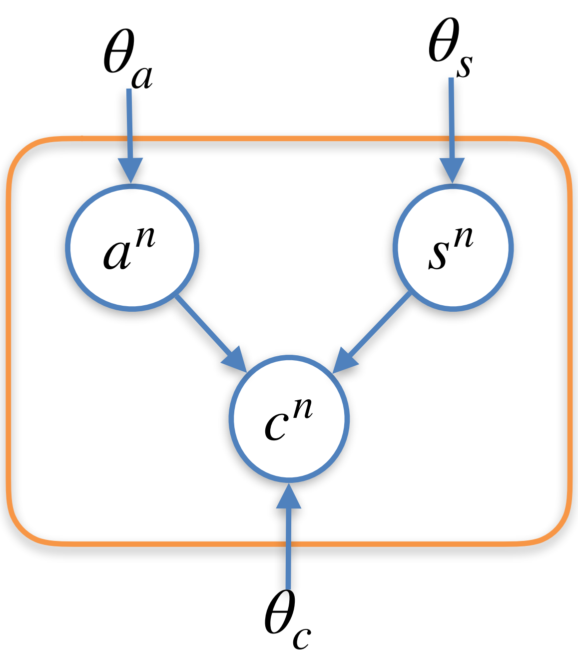


	a	s	c
$\mathbf{d_1}$	?	1	1
$\mathbf{d_2}$	1	0	0
$\mathbf{d_3}$	0	?	1
$\mathbf{d_4}$	0	?	0
$\mathbf{d_5}$	1	1	1
$\mathbf{d_6}$	?	0	0
$\mathbf{d_7}$	1	0	?

- Likelihood does not decompose by variables
- Likelihood does not decompose within CPDs
- Computing likelihood requires inference!

(sum-product computation)

Latent Variables



- For some patients, only partial information might be available

e.g., if a is not observed $v^n = \{s^n, c^n\}$

- Using the "visible" information available,

$$p(v^n | \theta) = \sum_a p(a, s^n, c^n | \theta) = \sum_a p(c^n | a, s^n, \theta_c) p(a | \theta_a) p(s^n | \theta_s)$$

- Cannot be factorised in terms of the parameters $\theta_a, \theta_c, \theta_s$

- Parameters of different tables are coupled, making the **optimisation** problem **harder**.

Maximum Likelihood with Missing Data

Likelihood for **complete data**

$$L(\theta \mid D) = p(D \mid \theta) = \prod_{i=1}^{N} p(\mathbf{d}_i) \mid \theta)$$

where N *is* the dataset size and $\mathbf{d}_i$ represents the assignments in the i-th entry data instance.

Marginal Likelihood (for **partially observed data**)

$$L(\theta \mid D) = p(D \mid \theta) = \Pi_{i=1}^{N} p(\mathbf{d}_i) \mid \theta) = \prod_{i-1}^{N} \sum_{\mathbf{h}_i} p(\mathbf{d}_i, \mathbf{h}_i \mid \theta)$$

with $\mathbf{h}_i$ the **hidden variables** in example i.

Global and local independence does not hold anymore in this case.

Identifiability

- Likelihood can have multiple global maxima
  - We can rename the values of the hidden variable
  - If H has two values, likelihood has two global maxima
- With many hidden variables, there can be an exponential number of global maxima
- Multiple local and global maxima can also occur with missing data (not only hidden variables)

Multiple Maxima

- In the case of incomplete data we are effectively summing up, the probability of all possible completions of the unobserved variables
- The overall likelihood functions is a summation of likelihood functions that correspond to the different ways to complete the data
- Results in a function with multiple maxima

Parameter Estimation with Missing Data

Expectation Maximisation

Find **maximum likelihood** solutions for models having **missing data**

$$\theta_{ML} = \mathbf{argmax}_\theta \log p(D \,|\, \theta) = \mathrm{argmax}_\theta \log \left\{ \sum_h p(\mathbf{d}, \mathbf{h} \,|\, \theta) \right\}$$

- Note: The sum over missing data appears inside the logarithm

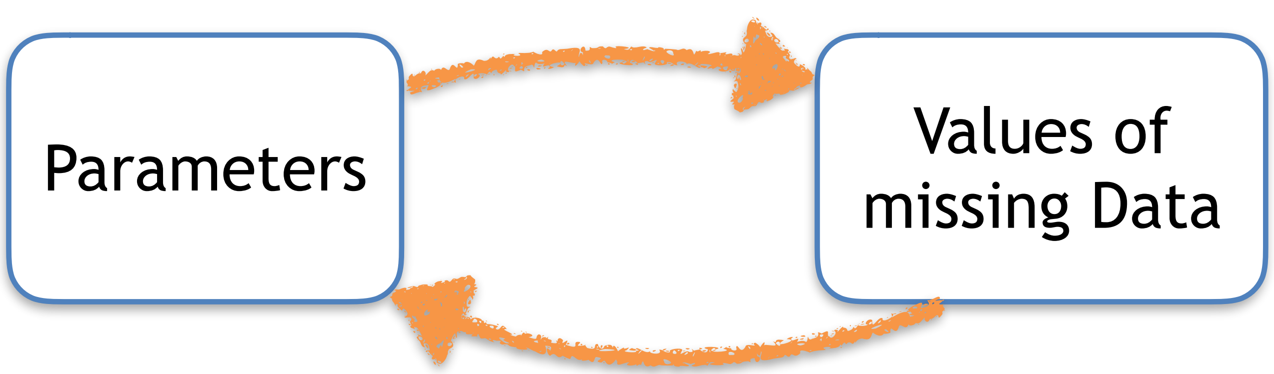
  log does NOT directly act on joint distribution $p(\mathbf{d}, \mathbf{h} \mid \theta)$

- Numerically complex that in the case when all variables are visible.

- The **Expectation-Maximisation (EM)** algorithm is an alternative optimisation algorithm than can help to produce simple an elegant updated for θ that converge to a local optimum.

Expectation Maximisation (EM)

- Special-purpose algorithm for optimising likelihood functions
- Parameter estimation is easy given complete data
- Computing probability of missing data amounts to inference given the parameters



EM Overview

- Choose a **starting point for parameters**
- **Iterate:**
  - **E-step** (Expectation): “Complete” the data using current parameters
  - **M-step** (Maximisation): Estimate parameters relative to data completion

* Guaranteed to improve $L(\theta \mid D)$ at each iteration