# Lecture 22. Inference on PGMs

COMP90051 Statistical Machine Learning

Lecturer: Feng Liu

THE UNIVERSITY OF
MELBOURNE

# This lecture

- Probabilistic inference: computing (conditional) marginals from joint distributions
    * Needed to learn (posterior update) in Bayesian ML
    * Exact inference: Elimination algorithm
    * Approximate inference: Sampling

- Statistical inference: Parameter estimation
    * Fully observed case: Factors decompose under MLE
    * Latent variables: Motivates the EM algorithm

# **Probabilistic inference on PGMs**

*Computing marginal and conditional distributions from the joint of a PGM using Bayes rule and marginalisation.*
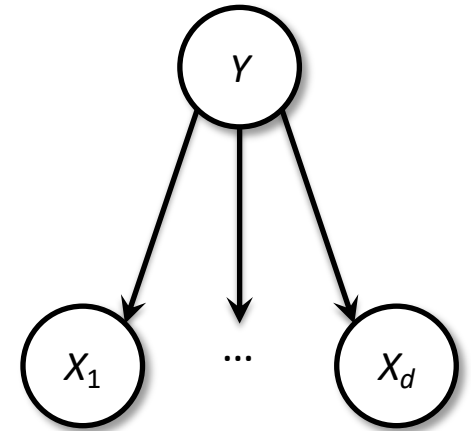
*This deck: how to do it efficiently.*

# Two familiar examples

- **Naïve Bayes** (frequentist/Bayesian)
  - Chooses most likely class given data
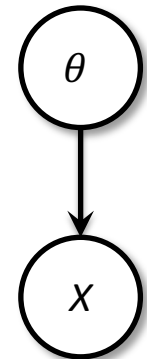  - $\Pr(Y|X_1, \dots, X_d) = \frac{\Pr(Y, X_1, \dots, X_d)}{\Pr(X_1, \dots, X_d)} = \frac{\Pr(Y, X_1, \dots, X_d)}{\sum_y \Pr(Y=y, X_1, \dots, X_d)}$

- Data $X|\theta \sim N(\theta, 1)$ with prior $\theta \sim N(0,1)$ (Bayesian)
  - Given observation $X = x$ update posterior
  - $\Pr(\theta|X) = \frac{\Pr(\theta, X)}{\Pr(X)} = \frac{\Pr(\theta, X)}{\sum_\theta \Pr(\theta, X)}$

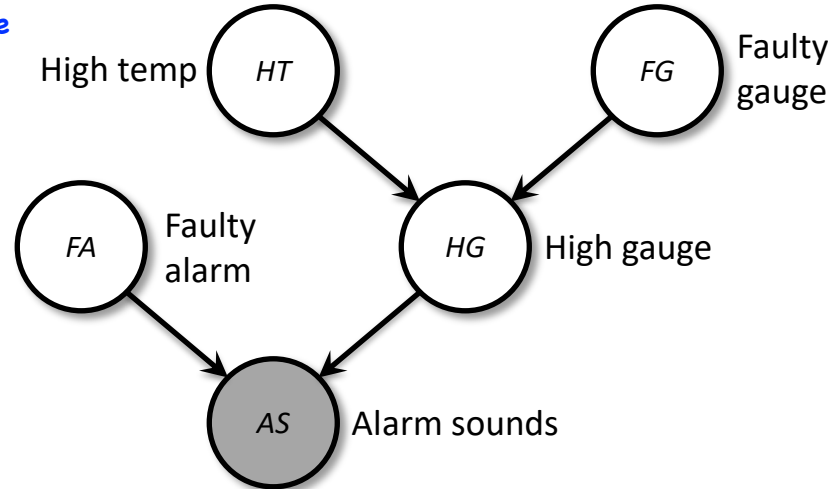- Joint + Bayes rule + marginalisation → anything

4

# Nuclear power plant

In essence, the equation is saying: "To get the joint probability of a high temperature and the alarm sounding at a specific level t, sum up the probabilities over all possible scenarios of the faulty gauge, high gauge reading, and faulty alarm."

- **Alarm sounds**; meltdown?!

- $\Pr(HT|AS = t) = \dfrac{\Pr(HT, AS{=}t)}{\Pr(AS{=}t)}$

$$= \frac{\sum_{FG, HG, FA} \Pr(AS{=}t, FA, HG, FG, HT)}{\sum_{FG, HG, FA, HT'} \Pr(AS{=}t, FA, HR, FG, HT')}$$

High temp — HT

FG — Faulty gauge

FA — Faulty alarm

HG — High gauge

AS — Alarm sounds

- Numerator (denominator similar)

expanding out sums, joint  *summing once over $2^5$ table*

$$= \sum_{FG} \sum_{HG} \sum_{FA} \Pr(HT) \Pr(HG|HT, FG) \Pr(FG) \Pr(AS = t|FA, HG) \Pr(FA)$$

distributing the sums as far down as possible  *summing over several smaller tables*
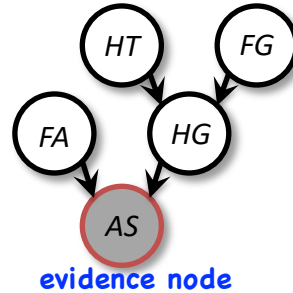
$$= \Pr(HT) \sum_{FG} \Pr(FG) \sum_{HG} \Pr(HG|HT, FG) \sum_{FA} \Pr(FA) \Pr(AS = t|FA, HG)$$

$$f(X = a) = \sum_x f(X = x) \, \delta(x = a)$$

# Nuclear power plant (cont.)

$= \Pr(HT) \sum_{FG} \Pr(FG) \sum_{HG} \Pr(HG|HT, FG) \sum_{FA} \Pr(FA) \Pr(AS = t|FA, HG)$

  eliminate $AS$: since $AS$ observed, really a no-op

**message**

$= \Pr(HT) \sum_{FG} \Pr(FG) \sum_{HG} \Pr(HG|HT, FG) \sum_{FA} \Pr(FA) \, m_{AS}(FA, HG)$

  eliminate $FA$: multiplying 1x2 by 2x2

**Process of Elimination**
1. eliminate evidence node
2. make FA and HG have a relation: message left by eliminating AS

$= \Pr(HT) \sum_{FG} \Pr(FG) \sum_{HG} \Pr(HG|HT, FG) \, m_{FA}(HG)$
3. eliminate FA, as it is not connected to other nodes
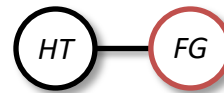4. eliminate HG
  eliminate $HG$: multiplying 2x2x2 by 2x1     5. eliminate FG

Multiplication
of tables, followed
by summing, is actually
matrix multiplication

$= \Pr(HT) \sum_{FG} \Pr(FG) \, m_{HG}(HT, FG)$

  eliminate $FG$: multiplying 1x2 by 2x2

$$m_{FA}(HG) = \begin{array}{c} FA \\ \begin{array}{|c|c|} \hline f & t \\ \hline 0.6 & 0.4 \\ \hline \end{array} \end{array} \times \begin{array}{cc} & HG \\ FA & \begin{array}{|c|c|} \hline f & t \\ \hline 1.0 & 0 \\ \hline 0.8 & 0.2 \\ \hline \end{array} \end{array}$$

$= \Pr(HT) \, m_{FG}(HT)$

# Elimination algorithm

**Eliminate** (Graph $G$, Evidence nodes $E$, Query nodes $Q$)

1. Choose node ordering $I$ such that $Q$ appears last

2. Initialise empty list active

3. For each node $X_i$ in $G$

   a) Append $\Pr(X_i | parents(X_i))$ to active

4. For each node $X_i$ in $E$

   a) Append $\delta(X_i, x_i)$ to active

5. For each $i$ in $I$

   a) potentials = Remove tables referencing $X_i$ from active

   b) $N_i$ = nodes other than $X_i$ referenced by tables

   c) Table $\phi_i(X_i, X_{N_i})$ = product of tables

   d) Table $m_i(X_{N_i}) = \sum_{X_i} \phi_i(X_i, X_{N_i})$ **largest clique**

   e) Append $m_i(X_{N_i})$ to active

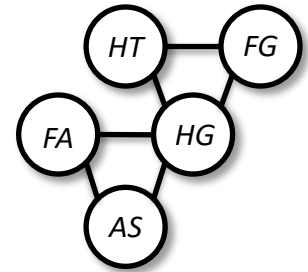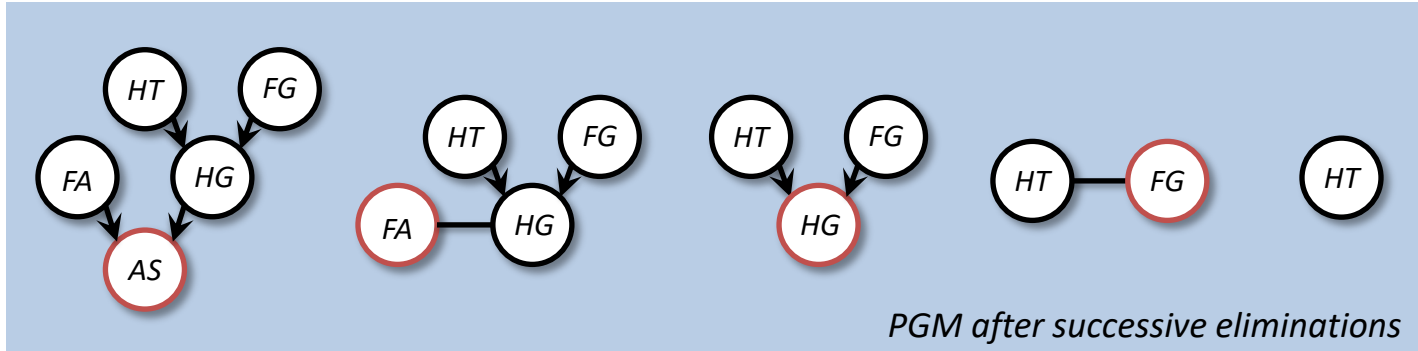6. Return $\Pr(X_Q | X_E = x_E) = \phi_Q(X_Q) / \sum_{X_Q} \phi_Q(X_Q)$

initialise

evidence

marginalise

normalise

7

# Runtime of elimination algorithm



*PGM after successive eliminations*

*"reconstructed" graph
From process called
**moralisation***

- Each step of elimination
  - ∗ Removes a node
  - ∗ Connects node's remaining neighbours
    → forms a clique in the "reconstructed" graph
    *(cliques are exactly r.v.'s involved in each sum)*

- Time complexity exponential in largest clique
  **For instance, if the largest clique has size k, the time complexity could be O(2^k)**

- Different elimination orderings produce different cliques
  - ∗ Treewidth: minimum over orderings of the largest clique
  - ∗ Best possible time complexity is exponential in the treewidth e.g. $O(2^{tw})$

# Mini Summary

(Exact) probabilistic inference on PGMs

- What? Marginalise out variables, Condition

- Why? Example: Bayesian posterior updates!

- How? The elimination algorithm

  naive way: consider whole table? ✗
  
  ✅ : use elimination algorithm

- How long? Time exponential in treewidth

  2^5 -> 2^3 in our example

Next time: Approximate PGM probabilistic inference

# Probabilistic inference by simulation

- Exact probabilistic inference can be expensive/impossible

  - Integration may not have analytical solution!

- Can we approximate numerically?

- Idea: sampling methods

  - Approximate **distribution** by **histogram of a sample**

  - We can't trivially sample: (1) only know desired distribution up to a (normalising) constant (2) naïve sampling approaches are inefficient in high dimensions.
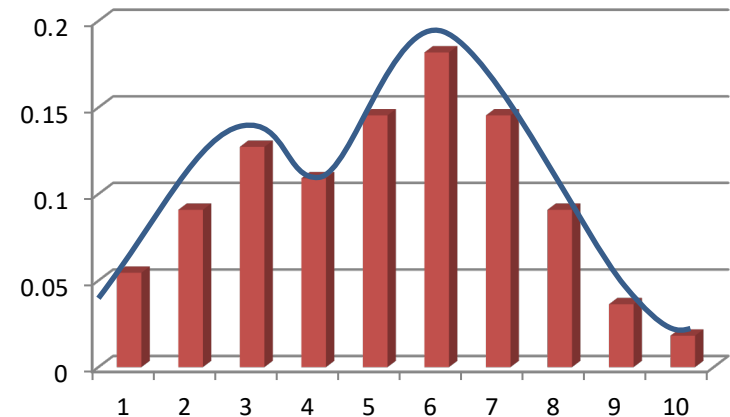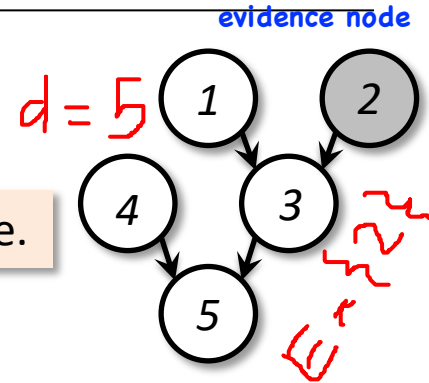
we want to obtain P(HT)

can we just simulate it? we can use the simulated node

to get the distribution / or just use it. because in the

end, we want to simulated it. cuz in the end we will

perform the sampling

we want to directly sample from the query node

# Gibbs sampling

*evidence node*

$d = 5$

Divide and conquer: Sampling single variable at a time.



- Given: D-PGM on $d$ random variables
  Given: evidence values $\mathbf{x}_E$ over variables $E \subset \{1, \dots, d\}$
  Goal: many approximately independent samples from joint conditioned on $\mathbf{x}_E$

1. Initialise with a starting $\mathbf{X}^{(0)} = \left( X_1^{(0)}, \dots, X_d^{(0)} \right)$ with $\mathbf{X}_E^{(0)} = \mathbf{x}_E$

   *randomly pick up point*
   *evidence node, give true sample value*

2. Repeat many times
   *1,3,4,5: non-evidence node (randome pickup)*

   a) Pick non-evidence node $X_j$ uniformly at random

   b) Sample single node $X_j' \sim p\left( X_j | X_1^{(i-1)}, \dots, X_{j-1}^{(i-1)}, X_{j+1}^{(i-1)}, \dots, X_d^{(i-1)} \right)$
   *sample from conditional distribution: because it is direct PGM*

   c) Save entire joint sample $\mathbf{X}^{(i)} = \left( X_1^{(i-1)}, \dots, X_{j-1}^{(i-1)}, X_j', X_{j+1}^{(i-1)}, \dots, X_d^{(i-1)} \right)$
   *subsititue to the new sample: we will use the sampled node to update our data*

- Exercise: Why always $\mathbf{X}_E^{(i)} = \mathbf{x}_E$?
  *E represnets the evidence nodes, their values are observed and fixed*

- Need not update nodes in random order, e.g. parents first order
  But do need to be able to sample from conditionals (e.g. conjugacy)
  *conditional distribution is easier to obtain than joint dis*

11

# Markov blanket

- Intuition: all the nodes that you directly depend on. *Not just your parents/children!*

- Consider node $X_i$ in D-PGM on nodes $N = \{1, \ldots, d\}$

- Markov blanket $\text{MB}(i)$ of $X_i$:
  - ∗ Nodes $B \subseteq N \backslash \{i\}$ such that…
  - ∗ $X_i$ independent of $\mathbf{X}_{\bar{B} \backslash \{i\}}$ given $\mathbf{X}_B$
  - ∗ $p(X_i \mid X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_d) = p(X_i \mid \text{MB}(X_i))$

- In D-PGM Markov blanket is:
  - ∗ Parents of $i$, children of $i$, parents of children of $i$
  - ∗ $p(X_i \mid \text{MB}(X_i)) \propto p(X_i \mid X_{\pi_i}) \prod_{k:i \in \pi_k} p(X_k \mid X_{\pi_k})$
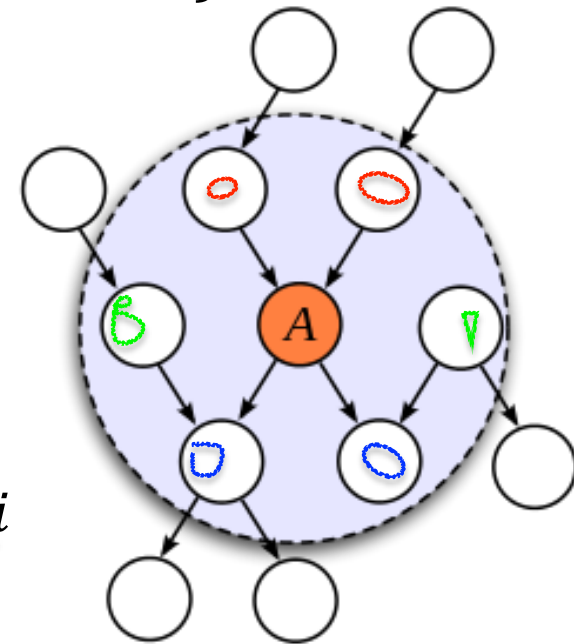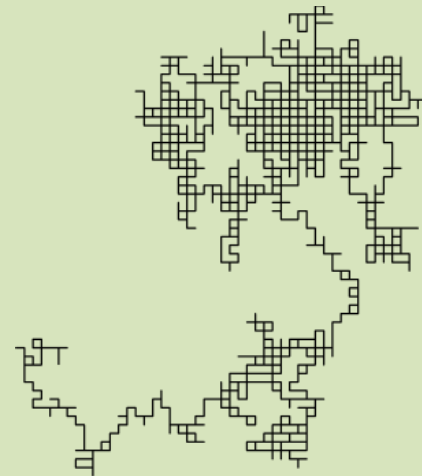


public domain

*if we find markov blanket: we find the conditional prob of Xi and we can use gibbs sampling*

# Markov Chain Monte Carlo (MCMC)

**key advantage: we don't reject any sample in gibbs sampling**

- Gibbs sampling produces a chain of samples $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \dots$ *approximating* draws from $p(\mathbf{X}_{\bar{E}}|\mathbf{X}_E = \mathbf{x}_E)$

- How good an approximation? Independent draws possible?

- Samples form a Markov chain: Each $\mathbf{X}^{(i)}$ depends only $\mathbf{X}^{(i-1)}$
    * States are all possible values taken by joint samples
    * Initial distribution $\mathbf{p}_0$ of state $\mathbf{X}^{(0)}$ given by initialisation process
    * Transition probability matrix $\mathbf{T}$ given by PGM conditional probabilities
    * Combines to: distribution $\mathbf{p}_i = (\mathbf{T})^i \mathbf{p}_0$ of state $\mathbf{X}^{(i)}$.

- Burn in: Run Gibbs long enough and $\mathbf{X}^{(i)} \sim p(\mathbf{X}_{\bar{E}}|\mathbf{X}_E = \mathbf{x}_E)$
    * "Limiting distribution" $\lim_{i \to \infty} \mathbf{p}_i$ is $p(\mathbf{X}_{\bar{E}}|\mathbf{X}_E = \mathbf{x}_E)$ under condition that no entry of $\mathbf{T}$ is zero ("ergodicity" – may not always hold)
    * Solution: throw away first few thousand samples

- Thinning: Want saved full samples to be independent
    * Neighbouring $\mathbf{X}^{(i)}, \mathbf{X}^{(i+1)}$ are highly correlated. Intuition why?
    * Solution: only keep every 100 or so samples

public domain

13

# Initialising Gibbs: Forward Sampling

- Set all evidence nodes to observed values

- Remaining nodes, parent-first order
  - Node has no parents? Sample from its D-PGM marginal
  - Sample node given previously sampled parents

- However Markov chain theory tells us MCMC converges irrespective of initial sample's distribution
  - The limiting distribution – the "equilibrium distribution" – is a property of the transition matrix (the PGM's joint) not the initial distribution

MCMC can start with any sample will always converge to the sample we desired

# Now what??
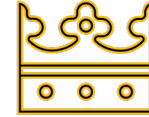
- With our $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(T)}$ in hand after running Gibbs for a while with burn-in and thinning…

- These form "i.i.d." sample of $p(\mathbf{X}_{\bar{E}}|\mathbf{X}_E = \mathbf{x}_E)$

- We can do heaps!
  a) Can approximate the distribution via a histogram of these samples (make bins, form counts).
  b) Marginalising out variables == Dropping components from samples
  c) Expectations: Estimating by sample mean of samples

- Posterior $p(\mathbf{w}|\mathbf{X}_{tr}, \mathbf{y}_{tr})$ combine (a) and (b)
  Mean posterior point estimate, combine with (c)

# Mini Summary

Approximate probabilistic inference on PGMs

- Why? Summation/integration may be costly

- Why? Integration may be impossible analytically

- Briefly: Gibbs sampling

Next time: Statistical inference on PGMs

# Statistical inference on PGMs

*Learning from data – fitting probability tables to observations (eg as a frequentist; a **Bayesian would just use probabilistic inference** to update prior to posterior)*

# Have PGM, Some observations, No tables...
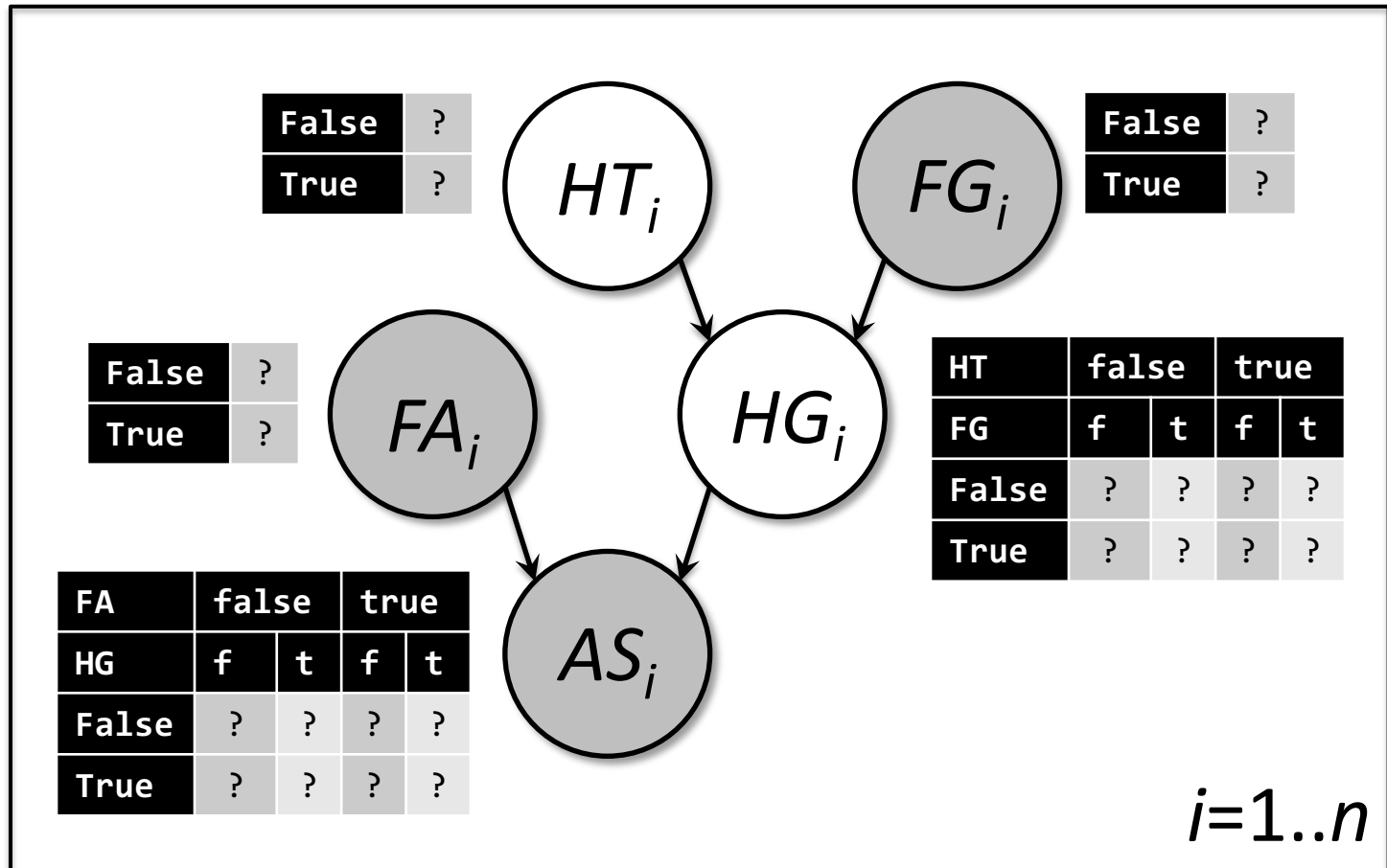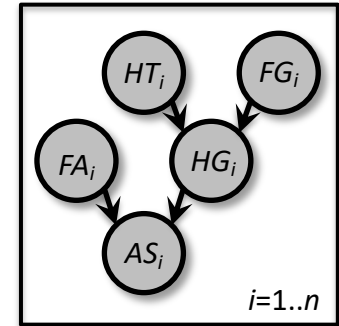
# Fully-observed case is "easy"

- ## Max-Likelihood Estimator (MLE) says
  - If we observe *all* r.v.'s $X$ in a PGM independently $n$ times $x_i$
  - Then maximise the *full* joint

  $$\arg\max_{\theta \in \Theta} \prod_{i=1}^{n} \prod_j p\left(X^j = x_i^j \mid X^{parents(j)} = x_i^{\,parents(j)}\right)$$

  max the prob that given the param, what is the prob we observe the data point we have

- ## Decomposes easily, leads to counts-based estimates
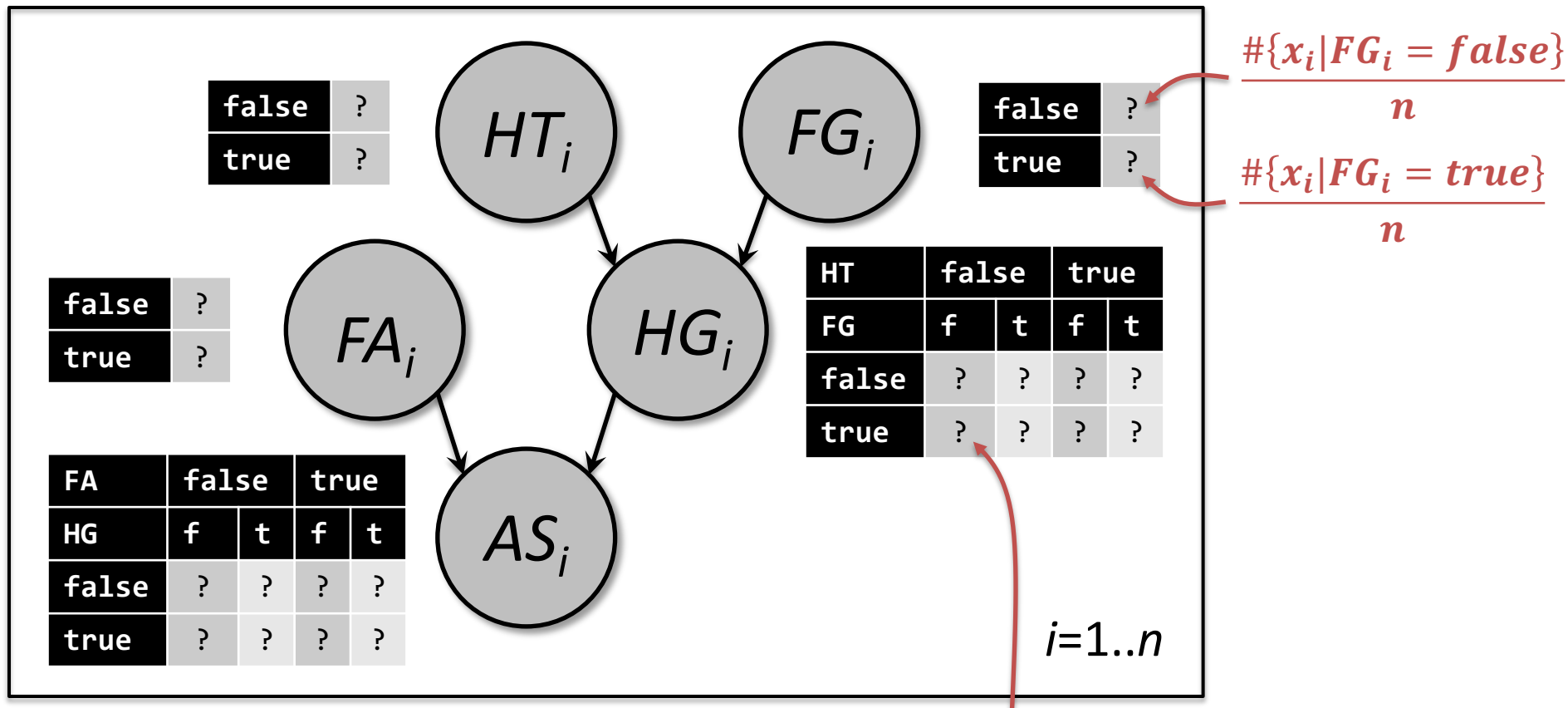  - Maximise log-likelihood instead; becomes sum of logs

  $$\arg\max_{\theta \in \Theta} \sum_{i=1}^{n} \sum_j \log p\left(X^j = x_i^j \mid X^{parents(j)} = x_i^{\,parents(j)}\right)$$

  - Big maximisation of all parameters together, decouples into small independent problems

- ## Example is training a naïve Bayes classifier

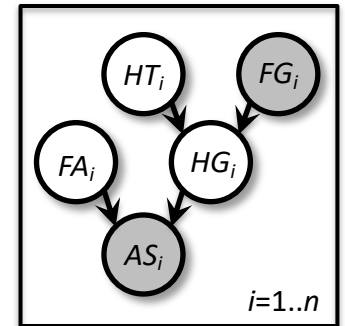decompose table: things will be easier!

19

# Example: Fully-observed case

$$\frac{\#\{x_i | FG_i = false\}}{n}$$

$$\frac{\#\{x_i | FG_i = true\}}{n}$$

| false | ? |
|---|---|
| true | ? |

| HT | false | | true | |
|---|---|---|---|---|
| FG | f | t | f | t |
| false | ? | ? | ? | ? |
| true | ? | ? | ? | ? |

| false | ? |
|---|---|
| true | ? |

| FA | false | | true | |
|---|---|---|---|---|
| HG | f | t | f | t |
| false | ? | ? | ? | ? |
| true | ? | ? | ? | ? |

*i*=1..*n*

$$\frac{\#\{x_i | HG_i = true, HT_i = false, FG_i = false\}}{\#\{x_i | HT_i = false, FG_i = false\}}$$

# Presence of unobserved variables trickier

- But most PGMs you'll encounter will have latent, or unobserved, variables



- What happens to the MLE?

  * Maximise likelihood of observed data only

  * Marginalise full joint to get to desired "partial" joint

  * $\arg\max_{\theta \in \Theta} \prod_{i=1}^{n} \sum_{\text{latent } j} \prod_{j} p\left(X^j = x_i{}^j \mid X^{parents(j)} = x_i{}^{parents(j)}\right)$

    we don't have observation for random variable, hence we need to marginalise them and then use MLE

  * This won't decouple – oh-no's*!!*

→ Use EM algorithm!

# Summary

- Probabilistic inference on PGMs
  - ∗ What is it and why do we care?
  - ∗ Elimination algorithm; complexity via cliques
  - ∗ Monte Carlo approaches as alternate to exact integration

- Statistical inference on PGMs
  - ∗ What is it and why do we care?
  - ∗ Straight MLE for fully-observed data
  - ∗ EM algorithm for mixed latent/observed data

Next time: deeper dive into HMMs and more