Bayesian Networks Learning
– Parameter Estimation

Assume the domain \( X = \{X_1, \ldots, X_n\} \) is governed by some underlying distribution \( P^*(X) \).

\( P^* \) is induced by some Bayesian network \( B^* = (G, \Theta) \).

Given a data set \( D = \{x[1], \ldots, x[M]\} \) of \( M \) samples from \( P^* \).

Samples are \textit{i.i.d.} - \textit{independent and identically distributed}.

The task is to recover the Bayesian network model.
Learning Bayesian networks

Data + Prior Information → Learner

Known Structure, Complete Data

- Network structure is specified
  - Inducer needs to estimate parameters
- Data does not contain missing values
**Unknown Structure, Complete Data**

- Network structure is not specified
  - Inducer needs to select arcs & estimate parameters
- Data does not contain missing values

**Known Structure, Incomplete Data**

- Network structure is specified
- Data contains missing values
  - Need to consider assignments to missing values
Unknown Structure, Incomplete Data

- Network structure is not specified
- Data contains missing values, or hidden variables?
  - Need to consider assignments to missing values

Parameter Estimation

- Network structure is specified
- Data set D consists of fully observed instances of the network variables
  \[ D = \{ x[1], \ldots, x[M] \} \]
- Estimate network parameters
Parameter Estimation

- Use a set $D=\{x[1], \ldots, x[M]\}$ of training samples drawn independently from a parametric model $P(x : \theta)$ to estimate the unknown parameter vector $\theta$.

- Parameter estimation: a classic problem in statistics
  - Maximum-Likelihood (ML) estimation
  - Bayesian estimation

Maximum-Likelihood Estimation

- IID data samples $D=\{x[1], \ldots, x[M]\}$
- Likelihood function
  $$L(\theta : D) = P(D | \theta) = \prod_{k=1}^{M} P(x[k] : \theta)$$
  
  the likelihood of $\theta$ w.r.t. the set of samples

- ML estimate of $\theta$ is, by definition, the value $\hat{\theta}$ that maximizes $L(\theta; D)$
- it is usually easier to work with the log-likelihood function
  $$l(\theta : D) = \log P(D | \theta) = \sum_{k=1}^{M} \log P(x[k] : \theta)$$
Example: Discrete Case

- Single binary variable
  
  \[ P(X=1) = \theta, \quad P(X=0) = 1 - \theta \]

  \[ P(x \mid \theta) = \theta^x (1 - \theta)^{1-x} \]

  Bernoulli distribution

  \[ L(\theta : D) = \prod_{k=1}^{M} P(x[k] : \theta) = \theta^{N_1} (1 - \theta)^{N_0} \]

- Sufficient statistics:
  
  \( N_1 \): number of 1’s in D, \( N_0 \): number of 0’s in D

- Log-likelihood
  
  \[ l(\theta : D) = N_1 \ln \theta + N_0 \ln (1 - \theta) \]

- ML estimation

  \[ \hat{\theta}_{ML} = \frac{N_1}{N_1 + N_0} = \frac{N_1}{M} \]
Multi-valued discrete random variables \( \{1, \ldots, K\} \)
\( \theta_i = P(X = i) \)

\[
P(x | \Theta) = \prod_{i=1}^{K} \theta_i^{\delta_{xi}}
\]

\[
L(\Theta : D) = \prod_{j=1}^{M} P(x[j] : \Theta) = \prod_{i=1}^{K} \theta_i^{N_i}
\]

- Sufficient statistics \( N_i \): the # of times \( i \) appears in \( D \)

\[
\hat{\theta}_{iML} = \frac{N_i}{\sum_j N_j} = \frac{N_i}{M}
\]

MLE for Bayesian Networks

- Training data has the form:

\[
D = \begin{bmatrix}
\vdots & \vdots & \vdots & \vdots \\
\end{bmatrix}
\]
Likelihood Function

- By definition of network, we get

\[
    L(\Theta : D) = \prod_m \rho(E[m], B[m], A[m], C[m] : \Theta)
\]

\[
    = \prod_m \begin{pmatrix}
        \rho(E[m] : \Theta) \\
        \rho(B[m] : \Theta) \\
        \rho(A[m] | B[m], E[m] : \Theta) \\
        \rho(C[m] | A[m] : \Theta)
    \end{pmatrix}
\]

Likelihood Function

- Rewriting terms, we get

\[
    L(\Theta : D) = \prod_m \rho(E[m], B[m], A[m], C[m] : \Theta)
\]

\[
    \prod_m \rho(E[m] : \Theta)
\]

\[
    \prod_m \rho(B[m] : \Theta)
\]

\[
    = \prod_m \begin{pmatrix}
        \rho(A[m] | B[m], E[m] : \Theta) \\
        \rho(C[m] | A[m] : \Theta)
    \end{pmatrix}
\]
Likelihood Function

- Rewriting terms, we get

\[
L(\Theta : D) = \prod_{m} P(E[m], B[m], A[m], C[m] : \Theta) \\
= \prod_{m} P(E[m] : \Theta_E) \\
= \prod_{m} P(B[m] : \Theta_B) \\
= \prod_{m} P(A[m] | B[m], E[m] : \Theta_{A|BE}) \\
= \prod_{m} P(C[m] | A[m] : \Theta_{C|A})
\]

General Bayesian Networks

Generalizing for any Bayesian network:

\[
L(\Theta : D) = \prod_{m} P(x_1[m], \ldots, x_n[m] : \Theta) \\
= \prod_{m} \prod_{i} P(x_i[m] | Pa_i[m] : \Theta_i) \\
= \prod_{i} L_i(\Theta_i : D) \quad \rightarrow \text{local likelihood function}
\]

Global Decomposition of the likelihood function

⇒ Independent estimation problems
MLE

• Assuming discrete variables (CPTs) leads to further decomposition \(\rightarrow\) local decomposition of the likelihood function

\[
L_i(\Theta_i : D) = \prod_m P(x_i[m] | Pa_i[m] : \Theta_i)
= \prod_{pa_i} \prod_m P(x_i[m] | Pa_i : \Theta_{x_i|pa_i})
= \prod_{pa_i} \prod \theta_{x_i|pa_i} N(x_i, pa_i)
\]

\[
\hat{\theta}_{x_i|pa_i} = \frac{N(x_i, pa_i)}{N(p_a_i)}
\]

Bayesian Inference

• Represent uncertainty about parameters using a probability distribution over parameters
• Learning using Bayes rule

\[
P(\theta | x[1], \ldots x[M]) = \frac{P(x[1], \ldots x[M] | \theta)P(\theta)}{P(x[1], \ldots x[M])}
\]
Example: Discrete Variable

- Single binary variable
  \[ P(X=1) = \theta, \quad P(X=0) = 1 - \theta \]
  \[ L(\theta : D) = \theta^{N_1} (1 - \theta)^{N_0} \]

- What prior \( p(\theta) \) to use?

Beta distribution

\[
\text{Beta } (\theta \mid \alpha_1, \alpha_0) = \frac{\Gamma(\alpha_1 + \alpha_0)}{\Gamma(\alpha_1)\Gamma(\alpha_0)} \theta^{\alpha_1 - 1}(1 - \theta)^{\alpha_0 - 1}
\]

\[
0 \leq \theta \leq 1
\]

\[
E(\theta) = \frac{\alpha_1}{\alpha_1 + \alpha_0}
\]

- The parameters \( \alpha_1 \) and \( \alpha_2 \) are positive reals, often called hyperparameters
- Gamma Function
  \[
  \Gamma(x + 1) = x\Gamma(x)
  \]
  \[
  \Gamma(1) = 1, \quad \Gamma(x) = (x - 1)! \text{ for integer } x
  \]
Assume the prior is a Beta distribution

\[ p(\theta) = Beta(\theta | \alpha_1, \alpha_0) = c \theta^{\alpha_1-1}(1 - \theta)^{\alpha_0-1} \]

The posterior density \( p(\theta | D) \)

\[ p(\theta | D) = c^1 \cdot p(D | \theta)p(\theta) = Beta(\theta | N_1 + \alpha_1, N_0 + \alpha_0) \]

- The property that the posterior distribution follows the same parametric form as the prior distribution is called **conjugacy**
- Beta prior is a **conjugate family** for the binomial distribution

\[ P(X = 1 | D) = \int P(X = 1 | \theta)p(\theta | D)d\theta = \int \theta p(\theta | D)d\theta = \frac{N_1 + \alpha_1}{N_1 + N_0 + \alpha_1 + \alpha_0} \equiv \hat{\theta}_{BE} \]

- It can be proved that:
  - If the prior is well-behaved – i.e. does not assign 0 density to any feasible parameter value, then both MLE and Bayesian estimate converge to the same value in the limit
  - Both **almost surely** converge to the underlying distribution \( P(X) \)
  - But the ML and Bayesian approaches behave differently when the number of samples is small
Multi-valued discrete random variables \( \{1, \ldots, k\} \)

\[
\theta_i = P(X = i)
\]

\[
P(D | \theta) = \prod_{i=1}^{k} \theta_i^{N_i}
\]

Sufficient statistics \( N_i \): the # of times \( i \) appears in \( D \)

Assume the prior \( p(\theta) \) is a Dirichlet distribution \( \text{Dir}(\theta | \alpha) \)

Dirichlet distribution with hyperparameters \( \alpha_i \)'s

\[
\text{Dir}(\theta | \alpha) = \frac{\Gamma(\alpha)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \prod_{i=1}^{k} \theta_i^{\alpha_i - 1}
\]

\[
0 \leq \theta_i \leq 1, \quad \sum_{i=1}^{k} \theta_i = 1, \quad \alpha = \sum_{i=1}^{k} \alpha_i
\]

\[
E(\theta_i) = \frac{\alpha_i}{\alpha}
\]
• Multi-valued discrete random variables \{1, \ldots, k\}

• Assume the prior \( p(\theta) \) is a Dirichlet distribution \( \text{Dir}(\theta|\alpha) \) with hyperparameters \( \alpha_i \)'s

• Then the posterior density \( p(\theta|D) \) is also a Dirichlet distribution with hyperparameters \( \alpha_i^* N_1, \ldots, \alpha_k^* N_k \)

\[
p(\theta | D) = c \cdot P(D | \theta) p(\theta) = \text{Dir}(\theta | N + \alpha)
\]

• Dirichlet prior is a conjugate family for the multinomial distribution

---

• Bayesian estimates

\[
P(X = i | D) = \int P(X = i | \theta) p(\theta | D) d\theta
\]

\[
= \int \theta_i \text{Dir}(\theta | N + \alpha) d\theta = \frac{N_i + \alpha_i}{M + \alpha} \equiv \hat{\theta}_{iBE}
\]

• The hyperparameters \( \alpha_i \) can be thought of as "imaginary" counts from our prior experience

• \( \alpha \): imaginary equivalent sample size

• Let \( \rho \) be prior belief about \( \theta_i \): \( \alpha_i = \alpha \rho_i \)

• The larger the equivalent sample size, the more confident we are in our prior

• Laplace estimates: \( \alpha = k, \alpha_i = 1 \)
Summary of Bayesian estimation

- Treat the unknown parameters as random variables
- Assume a prior distribution for the unknown parameters
- Update the distribution of the parameters based on data
- Finally compute \[ p(x|D) \]

Bayesian Estimation in BNs

- **Meta-network** for \( P(\Theta,D) \)
- Priors for each parameter group are independent
- Data instances are independent given the unknown parameters
Bayesian Estimation in BNs

- *Global parameter independence* assumption
  \[
P(\Theta) = \prod_i P(\Theta_{X_i | Pa_i})
  \]

- Global Decomposition of the likelihood function
  \[
  L(\Theta : D) = P(D | \Theta) = \prod_i L_i(\Theta_{X_i | Pa_i} : D)
  \]

- Posterior parameter independence
  \[
P(\Theta | D) = \prod_i P(\Theta_{X_i | Pa_i} | D)
  \]

- We can solve the prediction problem for each CPD independently

This can be “read” from the network by d-separation

Complete data \implies \text{posteriors on parameters are independent}

Can compute posterior over parameters separately!
Bayesian Estimation in BNs - CPTs

- Local parameter independence assumption

\[ P(\Theta_{X_i|Pa_i}) = \prod_{pa_i} P(\Theta_{X_i|pa_i}) \]

Bayesian Nets & Bayesian Prediction

- Posterior parameter independence

\[ P(\Theta \mid D) = \prod_i \prod_{pa_i} P(\Theta_{X_i|pa_i} \mid D) \]

- Assume Dirichlet prior

\[ P(\Theta_{X_i|pa_i}) = Dir(\Theta_{X_i|pa_i} \mid \alpha_{x_i|pa_i}, \ldots) \]

- Then

\[ P(\Theta_{X_i|pa_i} \mid D) = Dir(\Theta_{X_i|pa_i} \mid \alpha_{x_i|pa_i} + N(x_i, pa_i), \ldots) \]
Bayesian Nets & Bayesian Prediction

- Bayesian estimation

\[
\tilde{\theta}_{x_i|pa_i} = P(X_i = x_i \mid Pa_i = pa_i, D) = \frac{\alpha_{x_i|pa_i} + N(x_i, pa_i)}{\alpha_{pa_i} + N(pa_i)}
\]

Assessing Priors for Bayesian Nets

The BDe prior

- Introduce an equivalent sample size \( \alpha \) and a prior distribution \( P'(X) \), and set

\[
\alpha_{x_i|pa_i} = \alpha P'(x_i, pa_i)
\]

- We can represent \( P' \) as a BN \( (G_0, \Theta_0) \), and set

\[
\alpha_{x_i|pa_i} = \alpha P(x_i, pa_i \mid G_0, \Theta_0)
\]

Use BN inference to compute this

- E.g., empty BN with uniform distribution

\[
\alpha_{x_i|pa_i} = \frac{\alpha}{|X_i \mid \mid Pa_i|}
\]
Learning Parameters: Summary

- Estimation relies on **sufficient statistics**
  - For multinomials: counts $N(x_i, p_a_i)$
  - Parameter estimation

$$\hat{\theta}_{x_i|p_{a_i}} = \frac{N(x_i, p_{a_i})}{N(p_{a_i})} \quad \text{MLE}$$

$$\tilde{\theta}_{x_i|p_{a_i}} = \frac{\alpha_{x_i|p_{a_i}} + N(x_i, p_{a_i})}{\alpha_{p_{a_i}} + N(p_{a_i})} \quad \text{Bayesian (Dirichlet)}$$

- Both are asymptotically equivalent and consistent
- Both can be implemented in an on-line manner by accumulating sufficient statistics