Learning in Bayesian Networks

Yuqing Tang

Doctoral Program in Computer Science
The Graduate Center
City University of New York
ytang@cs.gc.cuny.edu

Outline

Introduction

- Learning probabilities – parameter estimations
  - Learning from complete data
  - Learning from incomplete data
- Learning network structure
  - Constraint-based learning
  - Score-based learning

Outline

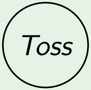
Outline

Bayesian network parameters

Question: Given the structure of a Bayesian network, what should we do to make it work?
Answer: With its structure fixed, the inference of a Bayesian network is determined by the probabilities in its conditional probability tables.

A One-node Bayesian Network

Example



- We have tossed a thumbtack, the outcome can be pin up or pin down

$$Toss \in \{down, up\}$$

- One probability table:

Toss	probability
up	?
down	?

- There is only one parameter:

$$\theta = P(Toss = up)$$

as $P(Toss = down) = 1 - P(Toss = up) = 1 - \theta$

A probabilistic model of the data collection procedure I

- Repetition: We have tossed the thumtack 100 times, among them 80 times are up and 20 times are down. We have collected the data:

$$\mathcal{D} = \{ Toss_1 = up, Toss_2 = up, \ldots, Toss_{80} = up, \\ Toss_{81} = down, \ldots, Toss_{100} = down)\}$$

- Independent experiments: We further assume the outcomes of the tosses are independent of each other. For a specific sequence of outcomes, we will have

$$\begin{aligned} & P(Toss_1 = up, Toss_2 = up, \ldots, Toss_{80} = up, \\ & \quad Toss_{81} = down, \ldots, Toss_{100} = down) \\ = \; & P(Toss_1 = up) \cdot P(Toss_2 = up) \cdots P(Toss_{80} = up) \\ & \cdot P(Toss_{81} = down) \cdots P(Toss_{100} = down) \\ = \; & \theta^{80}(1-\theta)^{20} \end{aligned}$$

A probabilistic model of the data collection procedure II

- Binomial experiment model: In the physical nature, the order of outcomes usually doesn't matter. One experiment represents a set of experiments with the same number of different outcomes: The probability of 80 times up out of 100 tosses given the parameter θ of the experiment model is then

$$\begin{aligned}
\mathcal{D} &= 80 \text{ up out of } 100 \\
M_\theta &= \text{Binomial model on } \theta \\
P(\mathcal{D}|M_\theta) & \\
&= \binom{100}{80} \cdot \theta^{80}(1-\theta)^{20} \\
&= \mu\theta^{80}(1-\theta)^{20}
\end{aligned}$$

where $\mu = \binom{100}{80}$

A probabilistic model of the data collection procedure III

- Parameter estimation: Choose the θ that can maximize the probability of the outcomes given the experiment model M_θ

$$\hat{\theta} = argmax_\theta P(\mathcal{D}|M_\theta)$$

- Calculations: The estimation can be done by solving the equation of the derivative of $P(D|M_\theta)$ being 0:

$$\begin{aligned}
&\frac{d}{d\theta}P(\mathcal{D}|M_\theta) \\
&= 80\mu\theta^{79}(1-\theta)^{20} + (-1)\cdot 20 \cdot \mu\theta^{80}(1-\theta)^{19} \\
&= \mu\theta^{79}(1-\theta)^{19}(80(1-\theta) - 20\theta) \\
&= \mu\theta^{79}(1-\theta)^{19}(80 - 100\theta) \\
&= 0
\end{aligned}$$

A probabilistic model of the data collection procedure IV

- Then we have the maximum likelihood estimation for θ as

$$\hat{\theta} = \frac{80}{100} = 0.8$$

Maximum likelihood estimation

- For multivariate Bayesian network, each case collected in the data collection process will be a vector $\mathbf{d} \in \mathcal{D}$
- We have the data collection model with parameters θ denoted by M_θ
- Assume the cases collected into $\mathcal{D}$ are independent, then the likelihood of M given data $\mathcal{D}$ is

$$L(M_\theta|\mathcal{D}) = P(\mathcal{D}|M_\theta) = \Pi_{\mathbf{d}\in\mathcal{D}} P(\mathbf{d}|M_\theta)$$

- Usually it is easier to do optimization with the additive form: We take log on both sides of the likelihood and obtain the log-likelihood

$$LL(M_\theta|\mathcal{D}) = log_2 P(\mathcal{D}|M_\theta) = \Sigma_{\mathbf{d}\in\mathcal{D}} log_2 P(\mathbf{d}|M_\theta)$$

- Maximum likelihood estimation is then looking for a $\hat{\theta}$ such that

$$\hat{\theta} = argmax_\theta L(M_\theta|\mathcal{D}) = argmax_\theta LL(M_\theta|\mathcal{D})$$

Maximum likelihood estimation by counting I

- In general, you can get the maximum likelihood estimation as the fraction of positive counts over the total number of counts
- The maximum likelihood estimation of the parameters in a Bayesian network can be done by finding the maximum likelihood estimates for each conditional probability distribution
- For each conditional probability distribution $P(X = x|Parent(X) = \mathbf{y})$, you simply count the number of cases in the data $\mathcal{D}$ for

Definition

$$P(X = x|Parent(X) = \mathbf{y}) = \frac{N(X = x, Parent(X) = \mathbf{y})}{N(Parent(X) = \mathbf{y})}$$

where $N(X_1 = x_1, X_2 = x_2, \ldots, X_k = x_k)$ is the number of cases in the dataset $\mathcal{D}$ in which the variables X_i takes value x_i $(i = 1, \ldots, k)$.

Maximum likelihood estimation by counting II

Example

Given the dataset

A	B	C
1	1	1
1	1	1
2	1	1
3	1	1

, to estimate $P(A = 1|B = 1, C = 1)$, you simply calculate

$$\begin{aligned} P(A = 1|B = 1, C = 1) &= \frac{N(A = 1, B = 1, C = 1)}{N(B = 1, C = 1)} \\ &= \frac{2}{4} \\ &= 0.5 \end{aligned}$$

Outline

Problems with maximum likelihood estimation

- When many cases are not encountered, namely the data are sparse, the maximum likelihood estimation becomes infeasible.

Example

A dataset of the number of five-letter words $T_1 T_2 T_3 T_4 T_5$ transmitted through a channel.

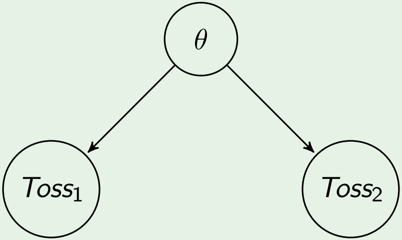
		Last three letters							
		aaa	aab	aba	abb	baa	bba	bab	bbb
First two letters	aa	2	2	2	2	5	7	5	7
	ab	3	4	4	4	1	2	0	2
	ba	0	1	0	0	3	5	3	5
	bb	5	6	6	6	2	2	2	2

As the number of all possible cases is $26^5 = 11,881,376$, it will be costly to conduct experiments to go through all these cases and experience each case with a fair amount of times to get a meaningful probability estimation. The data for many cases will be a small number or even be 0s!

The one node Bayesian network again I

Example

Model of the data collection procedure: The parameter θ causes the outcome of each toss, and given the parameter θ the outcome of the tosses are independent.



The one node Bayesian network again II

- Assume a uniform prior distribution on the parameter: $f(\theta) = 1$

$$
\begin{aligned}
f_p(\theta | Toss_1 = up) &= \frac{P(Toss_1 = up|\theta) f(\theta)}{P(Toss_1 = up)} \\
&= \frac{\theta f(\theta)}{P(Toss_1 = up)} \\
&= \frac{\theta}{P(Toss_1 = up)}
\end{aligned}
$$

- $P(Toss_1 = up)$ is the normalization factor

The one node Bayesian network again III

Definition

For a continues random variable θ with probability density function $f(\theta)$

- $P([\theta, \theta + d\theta]) = f(\theta)d\theta$
- The summation over the probabilities of all possible values of θ is then

$$\Sigma_{\theta \in [0,1]} P([\theta, \theta + d\theta]) = \int_0^1 f(\theta)d\theta$$

By the Kolmogorov axioms, we have

$$\begin{aligned}
&\Sigma_{\theta \in [0,1]} P([\theta, \theta + d\theta] | Toss_1 = up) \\
= &\int_0^1 \frac{\theta}{P(Toss_1 = up)} d\theta \\
= &\frac{1}{P(Toss_1 = up)} \int_0^1 \theta d\theta \\
= &1
\end{aligned}$$

The one node Bayesian network again IV

Therefore, we have

$$P(Toss_1 = up) \quad = \quad \int_0^1 \theta d\theta = \frac{1}{2}$$

so

$$f_p(\theta | Toss_1 = up) \quad = \quad 2\theta$$

- The best estimation for θ given the only one toss outcome is "up" is then

$$\hat{\theta} = E(\theta | Toss_1 = up) = \int_0^1 \theta f_p(\theta | Toss_1 = up) d\theta = \int_0^1 \theta(2\theta) d\theta = \frac{2}{3}$$

The one node Bayesian network again V

- Next, one more data $Toss_2 = down$

$$
\begin{aligned}
& f_p(\theta | Toss_2 = down, Toss_1 = up) \\
= \quad & \frac{P(Toss_2 = down, Toss_1 = up|\theta) f(\theta)}{P(Toss_2 = down, Toss_1 = up)} \\
= \quad & \frac{(1-\theta)\theta f(\theta)}{P(Toss_2 = down, Toss_1 = up)} \\
= \quad & \frac{(1-\theta)\theta}{P(Toss_2 = down, Toss_1 = up)}
\end{aligned}
$$

- The normalization factor can be computed by

$$
P(Toss_2 = down, Toss_1 = up) \quad = \quad \int_0^1 (1-\theta)\theta d\theta = \frac{1}{6}
$$

The one node Bayesian network again VI

- Rewrite

$$f_p(\theta | Toss_2 = down, Toss_1 = up) \quad = \quad 6\theta(1-\theta)$$

- The best estimation for θ given the only one toss outcome is “up” is then

$$\begin{aligned}
\hat{\theta} &= E(\theta | Toss_2 = down, Toss_1 = up) \\
&= \int_0^1 \theta f_p(\theta | Toss_2 = down, Toss_1 = up) d\theta \\
&= \int_0^1 \theta 6\theta(1-\theta) d\theta = \frac{1}{2}
\end{aligned}$$

Bayesian estimation (maiximum a posteriori parameters) I

- Start with a prior distribution $f(\theta)$ on the parameter θ: Put in any ideas you have about the parameters; if no idea at all, set $\theta = \mathbf{1}$
- Use the experiences $\mathcal{D}$ to update the distribution:

$$f_P(\theta|\mathcal{D}) \quad = \quad \frac{P(\mathcal{D}|\theta)f(\theta)}{P(\mathcal{D})}$$

With independency assumption on the experiences:

$$f_P(\theta|\mathcal{D}) = \frac{P(\mathcal{D}|\theta)f(\theta)}{P(\mathcal{D})} = \frac{\Pi_{\mathbf{d}\in\mathcal{D}}P(\mathbf{d}|\theta)f(\theta)}{P(\mathcal{D})}$$

- When we take the entries of the conditional probability tables in the Bayesian network as parameters θ, the conditional probability $P(\mathbf{d}|\theta)$ can be computed as the joint probability for $\mathbf{d}$ in the Bayesian network in terms of θ.

Bayesian estimation (maiximum a posteriori parameters) II

- The best estimation of θ given the data $\mathcal{D}$ is the mean of θ of the distribution $f_P(\theta|\mathcal{D})$

$$
\begin{aligned}
\hat{\theta} &= E(\theta|\mathcal{D}) \\
&= \int_0^1 \theta f_P(\theta|\mathcal{D}) d\theta \\
&= \int_0^1 \theta \frac{\Pi_{\mathbf{d}\in\mathcal{D}} P(\mathbf{d}|\theta) f(\theta)}{P(\mathcal{D})} d\theta \\
&= \mu \int_0^1 \theta \Pi_{\mathbf{d}\in\mathcal{D}} P(\mathbf{d}|\theta) f(\theta) d\theta
\end{aligned}
$$

where $\mu = \frac{1}{P(\mathcal{D})}$ is the normalizing factor which desn't depend on θ.

Bayesian estimation (maiximum a posteriori parameters) III

Definition

Given a distribution $P(X|\mathcal{D})$ of X given $\mathcal{D}$, the mean of X is

$$E(X|\mathcal{D}) = \Sigma_{x \in Domain(X)} x \cdot P(X = x|\mathcal{D})$$

For a continuous random variable $\theta \in [0, 1]$,

$$E(\theta|\mathcal{D}) = \Sigma_{\theta \in [0,1]} \theta P([\theta, \theta + d\theta]|\mathcal{D}) = \int_0^1 \theta f(\theta|\mathcal{D}) d\theta$$

- Again, we can compute $\hat{\theta}$ by counting. For a $\theta_i \in \theta$ where $\theta_i = P(X = x|Parent(X) = \mathbf{y})$, start with even prior $f(\theta_i) = 1$

$$\hat{\theta}_i = \frac{N(X = x, Parent(X) = \mathbf{y}) + 1}{\Sigma_{v \in Domain(X)} (N(X = v, Parent(X) = \mathbf{y}) + 1)}$$

An Example of Multivariate Bayesian Estimation I

Example

A dataset of the number of five-letter words $T_1 T_2 T_3 T_4 T_5$ transmitted through a channel, looking at the first two letters to estimate $P(T_2 | T_1)$:

		T_1 a	T_1 b
T_2	a	32	17
	b	20	31

- $N(X = x, Parent(X) = \mathbf{y}) + 1$:

		T_1 a	T_1 b
T_2	a	33	18
	b	21	32

An Example of Multivariate Bayesian Estimation II

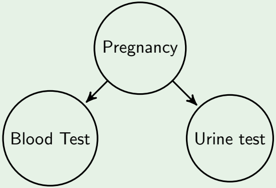
- Divided by $\Sigma_{v \in Domain(X)}(N(X = v, Parent(X) = \mathbf{y}) + 1)$ – divide each entry by the summation of its column which corresponds to all the possible values T_2 can take given each assignment to its parent T_1:

		T_1	
		a	b
T_2	a	$\frac{33}{54}$	$\frac{18}{50}$
	b	$\frac{21}{54}$	$\frac{32}{50}$

Outline

An incomplete data example

Example



Cases	Pr	Bt	Ut
$\mathbf{d}_1$	?	pos	pos
$\mathbf{d}_2$	yes	neg	pos
$\mathbf{d}_3$	yes	pos	?
$\mathbf{d}_4$	yes	pos	neg
$\mathbf{d}_5$	?	neg	?

We need to estimate the parameters

- $\theta(Pr) = Pr(Pr)$
- $\theta(Bt|Pr) = Pr(Bt|Pr)$
- $\theta(Ut|Pr) = Pr(Ut|Pr)$

Incomplete data

- Maximum likelihood estimation and Bayesian estimation only work for complete data, i.e. a data set in which each case specifies a value for each of the variables.
- Consider the incomplete data set as having been produced from a complete data set by a process that hides some of the data
  - If the probability that a particular value is missing depends only on the observed values, then the data is said to be missing at random (MAR).
  - If this probability is also independent of the observed values, then the data is said to be missing completely at random (MCAR).
  - If the data is neither MAR nor MCAR, then the process that generated the missing data is said to be nonignorable.
- Computing $\hat{\theta} = argmax_\theta P(\mathcal{D}|M_\theta)$ is not feasible in practice as the dependency among components of each case $\mathbf{d} \in \mathcal{D}$ causes $P(\mathcal{D}|M_\theta)$ to be very complicated.
- We can approximate the parameter estimation by the Expectation-Maximization (EM) algorithm.

An incomplete data example (cont.) I

- Given the data

Cases	Pr	Bt	Ut
$\mathbf{d}_1$	?	pos	pos
$\mathbf{d}_2$	yes	neg	pos
$\mathbf{d}_3$	yes	pos	?
$\mathbf{d}_4$	yes	pos	neg
$\mathbf{d}_5$	?	neg	?

  what are $N(Pr = yes, Bt = pos, Ut = pos)$ and $N(Pr = no, Bt = pos, Ut = pos)$?

An incomplete data example (cont.) II

- If we know $P(Pr = yes|Bt = pos, Ut = pos)$, we can estimate $N(Pr = yes, Bt = pos, Ut = pos)$ by its means over the data $\mathcal{D}$

$$
\begin{aligned}
& E(N(Pr = yes, Bt = pos, Ut = pos)|\mathcal{D}) \\
= \quad & \Sigma_{\mathbf{d} \in \mathcal{D}} 1 \cdot P(Pr = yes, Bt = pos, Ut = pos|\mathbf{d}) \\
= \quad & 1 \cdot P(Pr = yes, Bt = pos, Ut = pos|\mathbf{d}_1) \\
& +1 \cdot P(Pr = yes, Bt = pos, Ut = pos|\mathbf{d}_2) \\
& +1 \cdot P(Pr = yes, Bt = pos, Ut = pos|\mathbf{d}_3) \\
& +1 \cdot P(Pr = yes, Bt = pos, Ut = pos|\mathbf{d}_4) \\
& +1 \cdot P(Pr = yes, Bt = pos, Ut = pos|\mathbf{d}_5) \\
= \quad & P(Pr = yes, Bt = pos, Ut = pos|\mathbf{d}_1) \\
= \quad & P(Bt = pos, Ut = pos|\mathbf{d}_1) \cdot P(Pr = yes|Bt = pos, Ut = pos, \mathbf{d}_1) \\
= \quad & P(Pr = yes|Bt = pos, Ut = pos)
\end{aligned}
$$

An incomplete data example (cont.) III

because $P(Bt = pos, Ut = pos|\mathbf{d}_1) = 1$ is the evidence known in $\mathbf{d}_1$, and $P(Pr = yes|Bt = pos, Ut = pos)$ is the nature of the world which is independent of the data $\mathbf{d}_1$ collected

- Similarly, if we know $P(Pr = no|Bt = pos, Ut = pos)$, then

$$\begin{aligned} & E(N(Pr = no, Bt = pos, Ut = pos)|\mathcal{D}) \\ = \;& P(Pr = no|Bt = pos, Ut = pos) \end{aligned}$$

The EM algorithm

1. Choose an $\epsilon > 0$ to regulate the stopping criterion.
2. Let θ^0 – the probabilities in the tables – to be some initial estimates (chosen arbitrarily).
3. Set $t \leftarrow 0$.
4. Repeat:
   - E-step: For each node X calculate the expected counts:

$$E_{\theta^t}[N(X = x, Parents(X) = \mathbf{y})|\mathcal{D}] = \\ \Sigma_{\mathbf{d}\in\mathcal{D}} P(X = x, Parents(X) = \mathbf{y}|\mathbf{d}, \theta^t)$$

   - M-step: use the expected counts as if they were actual counts to calculate a new maximum likelihood estimate for θ:

$$\hat{\theta} = \frac{E_{\theta^t}[N(X = x, Parents(X) = \mathbf{y})|\mathcal{D}]}{\Sigma_{v\in Domain(X)} E_{\theta^t}[N(X = v, Parents(X) = \mathbf{y})|\mathcal{D}]}$$

   - Set $\theta^{t+1} = \hat{\theta}$ and $t \leftarrow t + 1$

   Until $|log_2 P(\mathcal{D}|\theta^t) - log_2 P(\mathcal{D}|\theta^{t-1}| \leq \epsilon$

Outline

The BN structure learning problem

- Suppose that there is an unknown Bayesian network BN over the universe $\mathcal{U}$ that produces the sample cases $\mathcal{D}$
- You are asked to reconstruct the BN given $\mathcal{D}$
- More insight
  - The BN gives you a distribution $P_{BN}(\mathcal{U})$
  - The data $\mathcal{D}$ gives you another distribution $P^{\#}_{\mathcal{D}}(\mathcal{U})$
  - $P^{\#}_{\mathcal{D}}(\mathcal{U})$ is close to $P_{BN}(\mathcal{U})$
  - Assume that all links in BN are essential:
    If $Parents(A)$ are the parents of A. B is one of $Parents(A)$, then there are two values b_1 and b_2 of B and a value combination $\mathbf{c}$ of $Parents(A) \setminus \{B\}$ such that $P(Aa|b_1, \mathbf{c}) \neq P(A|b_2, \mathbf{c})$.
- In practice, the task is to construct a Bayesian network M for which $P_M(\mathcal{U})$ is close to $P^{\#}_{\mathcal{D}}(\mathcal{U})$

Outline

Constasint-based learning

1. From the data $\mathcal{D}$, test a hypothesis $I(A, B, \chi)$ (A is independent of B given χ)
2. Using the tested hypotheses $\{I(A, B, \chi\}$, construct a skeleton — an undirected version of the target BN
3. Using the tested hypotheses $\{I(A, B, \chi\}$, introduce directions to the edges in skeleton by applying the following four rules in order
    - Rule 1: Introduction of V-structures
    - Rule 2: Avoid new V-structures
    - Rule 3: Avoid cycles
    - Rule 4: Choose randomly if the above 3 rules can not be applied
4. During the construction, choose simplest structure if multiple structures are valid (Ockham's Razor)
5. Learning the probabilities from $\mathcal{D}$ for the obtained Bayesian network M

Test conditional independence on data sets

Conditional mutual information

$$CMI(A,B|\chi) = \Sigma_{\chi} P^{\#}(\chi) \Sigma_{A,B} P^{\#}(A,B|\chi) log_2 \frac{P^{\#}(A,B|\chi)}{P^{\#}(A|\chi)P^{\#}(B|\chi)}$$

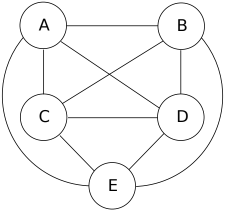
- It holds that $I_{\mathcal{D}}(A,B,\chi) \Leftrightarrow CMI(A,B|\chi) = 0$.
- $\chi^2 - test$ on the hypothesis $CMI(A,B|\chi) = 0$, and the user decides the acceptance region.

From independence tests to skeleton

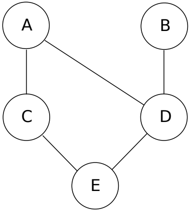
Start with the complete graph (all nodes are connected);
$i = 0$;
while *there is a node with at least* $i+1$ *neighbors* **do**
  for *all nodes* A *with* $|neighbors(A)| >= i+1$ **do**
    for *each* $B \in neighbors(A)$ **do**
      for *each* $\chi \subseteq neighbors(A) \setminus \{B\}$ **do**
        if $I(A, B, \chi)$ **then**
          remove the link $A - B$ and store $I(A, B, \chi)$;
        end
      end
    end
  end
  $i = i + 1$;
end

Algorithm 1: The PC algorithm

Data to skeleton (example) I



$\Rightarrow$



- Start with the complete graph
- $i = 0$: Test on the data: $I(A, B)$, $I(A, C)$, $I(A, D)$, $I(A, E)$, $I(B, C)$, $I(B, D)$, $I(B, E)$, $I(C, D)$, $I(C, E)$ and $I(D, E)$
  Get "yes" for $I(A, B)$ and $I(B, C)$, so $A - B$ and $B - C$ are removed

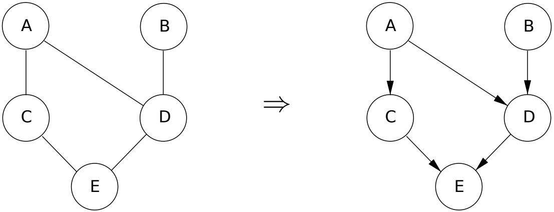
Data to skeleton (example) II

- $i = 1$: Test on the data: $I(A, C, E)$, $I(B, C, D)$, $I(B, C, E)$, $I(B, D, C)$, $I(B, D, E)$, $I(B, E, C)$, $I(B, E, D)$, $i(C, B, A)$, $I(C, D, B)$, $I(C, D, A)$
  "yes" on $I(C, D, A)$, remove $C - D$
  Continue $I(C, E, A)$, $I(C, E, B)$, $I(D, B, E)$, $I(D, E, B)$, $I(E, A, B)$, $I(E, A, D)$, $I(E, B, A)$, $I(E, C, B)$, $I(E, C, D)$, $I(E, D, A)$ and $I(E, D, C)$
- $i = 2$:...
- $i = \ldots$
- Until we reach the result on the right, and we get $I(A, B)$, $I(B, C)$, $I(C, D, A)$, $I(A, E, \{C, D\})$, and $I(B, E, \{C, D\})$ stored.

From skeleton to Bayesian Network

- Rule 1: Introduction of V-structure
  If you have three nodes, A, B, C such chat $A - C$ and $B - C$ in the skeleton, but not $A - B$, then introduce the V-structure $A \rightarrow C \leftarrow B$ if there exists an χ (possibly empty) such that $I(A, B, \chi)$ and $C \notin \chi$.
- Rule 2: Avoid new V-structure
  When Rule 1 has been exhausted, and you have $A \rightarrow C - B$ (an no link between A and B), then direct $C \rightarrow B$.
- Rule 3: Avoid cycles
  If $A \rightarrow B$ introduces a directed cycle in the graph, then do $A \leftarrow B$.
- Rule 4: Choose randomly
  If none of the rule 1-3 can be applied anywhere in the graph, choose an undirected link and give it an arbitrary direction.

Skeleton to Bayesian Network (example) I



- In the skeleton step, we get $I(A, B)$, $I(B, C)$, $I(C, D, A)$, $I(A, E, \{C, D\})$, and $I(B, E, \{C, D\})$ stored.
- Look at $A - D - B$, during the skeleton construction the PC-algorithm stored $I(A, B)$, namely $I(A, B, \emptyset)$ and $D \notin \emptyset$, apply Rule 1, we direct $A \to D \leftarrow B$.
- Look at $C - E - D$, we have $I(C, D, A)$ and $E \notin \{A\}$, apply Rule 1, we direct $C \to E \leftarrow D$. No places, Rule 1 can be applied.

Skeleton to Bayesian Network (example) II

- Rule 2 not applicable
- Look at $A - C$, apply Rule 3, to avoid cycle we have to direct $A \rightarrow C$
- We reach the result on the right

Outline

Score-based learning

1. Choose an initial structure (empty structure, a randomly chosen structure, or a prior structure constructed by the user)
2. Repeat
   1. Calculate $\Delta(A)$ for each legal arc operation $A \in \{\text{arc addition}, \text{arc deletion}, \text{arc reversal}\}$ where $\Delta(A) = score(Apply(S, A)|\mathcal{D}) - score(S|\mathcal{D})$
      - Let $\Delta^* = max_A \Delta(A)$ and $A^* = argmax_A \Delta(A)$
   2. If $\Delta^* > 0$, then set $S = Apply(S, A^*)$
3. Until $\Delta^* \leq 0$

- The key step is to look for a good score function $score(S|\mathcal{D})$ of a structure S given the data $\mathcal{D}$

Bayesian information criteria (BIC)

One popular candidate for $score(S\mathcal{D})$ is the BIC:

$$BIC(S|\mathcal{D}) = log_2 P(\mathcal{D}|\hat{\theta}_S, S) - \frac{size(S)}{2} log_2(N)$$

where $\hat{\theta}_S$ is the maximum likelihood parameters for the candidate BN structure S, $\mathcal{D}$ is the data, N is the size of $\mathcal{D}$, and $size(S)$ is an measurement on the complexity of the structure.
With independent assumption on the cases in $\mathcal{D}$, we have

$$BIC(S|\mathcal{D}) = \Sigma_{i=1}^{N} log_2 P(\mathbf{d}_i|\hat{\theta}_S, S) - \frac{size(S)}{2} log_2(N)$$

Outline

Summary

- Learning probabilities – parameter estimations
  - Learning from complete data
  - Learning from incomplete data
- Learning network structure
  - Constraint-based learning
  - Score-based learning

Acknowledgments

Lecture 6 is composed the instructor's own understanding of the subject, and materials from [Jensen and Nielsen, 2007, Chapter 6, Chapter 7] and [Korb and Nicholson, 2003, Chapter 6, Chapter 7, Chapter 8] with the instructor's own interpretations. The instructor takes full responsibility of any mistakes in the slides.

References I

Finn V. Jensen and Thomas D. Nielsen.
Bayesian Networks and Decision Graphs.
Springer Publishing Company, Incorporated, 2007.

K. Korb and A. E. Nicholson.
Bayesian Artificial Intelligence.
Chapman & Hall /CRC, 2003.