Probability Graph

2021年6月11日 14:47

P46 Probabilistic Graphical Model

PGM
- Representation
  - Directed Graph — eg. Bayesian Network
  - Undirected Graph — eg. Markov Network
  - Gaussian Graph (continuous)
    - Gaussian BN
    - Gaussian MN
- Inference
  - Precise Inference
  - Approximate Inference
    - Deterministic Approximation (确定性近似) — eg. Variational Inference
    - Stochastic Approximation — eg. Markov Chain Monte Carlo Method (MCMC)
- Learing
  - Structure Learing
  - Parameter Learning
    - Complete Data
      - Directed
      - Undirected
    - Hidden Variable → EM

High dimensional random variabes $P(x_1, x_2, \cdots, x_p)$ $\begin{cases} \text{Marginal Probility } P(x_i) \\ \text{Conditional Probility } P(x_i|x_j) \end{cases}$

Margin distribution is the probability distribution of a subset of a collection of random variables

For second dimensional random continuous variables x_1, x_2 (↓ discrete……)

$$\begin{cases} \text{Sum Rule}: P(x_1) = \int P(x_1, x_2)\, dx_2 \\ \text{Product Rule (乘法法则)}: P(x_1, x_2) = P(x_1)P(x_2|x_1) = P(x_2)P(x_1|x_2) \\ \text{Bayesian Rule}: P(x_2|x_1) = \frac{P(x_1, x_2)}{P(x_1)} = \frac{P(x_1,x_2)}{\int P(x_1,x_2)dx_2} = \frac{P(x_2)P(x_1|x_2)}{\int P(x_2)P(x_1|x_2)dx_2} \end{cases}$$

For high dimensional $x_1, x_2, \cdots, x_p$: Chain Rule : $P(x_1, x_2, \cdots, x_p) = \prod_{i=1}^{p} P(x_i) P(x_i | x_1, x_2, \cdots, x_{i-1})$

↘ Dilemma : the higher the dimension, the more complex the computing is

↓ simplify $\xrightarrow{\text{mutually independent}}$ $P(x_1, x_2, \cdots, x_p) = \prod_{i=1}^{p} P(x_i)$ $\xrightarrow{x_j \perp x_i | j<i}$ Markov Property

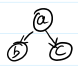
↓ Naive Bayes : $P(x|y) = \prod_{i=1}^{p} P(x_i|y)$

P47 Bayesian Network - Conditional Independent

conditional independent : $x_A \perp x_B | x_C$, x_A, x_B, x_C are disjoint sets.

factorization : $P(x_1, x_2, \cdots, x_p) = \prod_{i=1}^{p} P(x_i | x_{pa(i)})$

$\xrightarrow[\text{tail} \qquad \text{head}]{}$

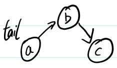
tail to tail 

(因子分解) fatorization : $p(a,b,c) = P(a)p(b|a)p(c|a)$ chain rule : $P(a,b,c) = p(a)p(b|a)p(c|b,a)$

$\therefore p(c|a) = P(c|b,a) \Rightarrow p(c|a)p(b|a) = p(c|b,a)p(b|a) \Rightarrow p(c|a)p(b|a) = p(c,b|a)$

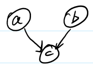
$\therefore b \perp c | a$ ⟺ If a is observed, the path is blocked (b, c are independent of each other)

If a isn't observed, the opposite is true

head to tail 

fatorization : $p(a,b,c) = p(a)p(b|a)p(c|b)$ $= P(c|b) = p(c|b,a)$

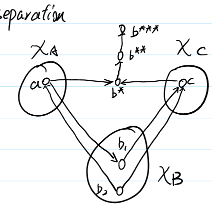
$\therefore a \perp c | b$ same as "tail to tail"

head to head 

$\cdots \Rightarrow P(b) = P(b|a) \Rightarrow a \perp b$

If c is observed, the path is connected (a, b related)

P48 Bayesian Network - D-separation



Global Markov Property:

a belongs the set X_A, c belongs to set X_C

b is on a path that connects a, c

If the path is "tail to tail" or "head to tail", b belongs to the set X_B

If the path is "head to head", b and its child notes don't belong to the set X_B

The method of using GMP to determine the independence between sets is D-separation

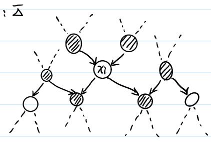
Markov Blanket :

$$x \in \mathbb{R}^p \quad P(x_i|x_{-i}) = \frac{P(x_i|x_{-i})}{P(x_{-i})} = \frac{P(x)}{\int_{x_i} p(x)dx} = \frac{\sum_{j=1}^{p} P(x_j|x_{pa(j)})}{\int_{x_i} \sum_{j=1}^{p} P(x_j|x_{pa(j)})dx}$$

$\begin{cases} \text{Not related to } x_i : \Delta \\ \text{Related to } x_i : \bar{\Delta} \end{cases}$

$x_{-i} = (x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_p)$

$\therefore P(x_i|x_{-i}) = f(\bar{\Delta})$ $\because f(\bar{\Delta}) = \begin{cases} P(x_i|x_{pa(i)}) \\ P(x_{child(i)} | x_i, x_{pa(child(i))}) \end{cases}$



P49 Bayesian Network - example

Bayesian Network
- single : Naive Bayes
- hybrid : GMM
- + time dimension
  - Markov Chain
  - Gaussian Process (infinity dimensional Gaussian distribution)
- Continuous : Gaussian Bayes Network

→ Dynamic Model
- HMM discrete
- LDS — Kalman Filter (卡曼滤波) (continuous, linear) ↓ Gaussian
- Particle Filter (粒子滤波) (Non-Gaussian) (Non-Linear)

From single to hybrid, from finite to infinite

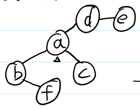
$\underbrace{\text{From single to hybrid}}_{\text{space (discrete} \to \text{continuous)}}$, $\underbrace{\text{from finite to infinite}}_{\text{time}}$

P50 Markov Random Field – Representation – Conditional Independence

① Global Markov $\quad x_A \perp x_C \mid x_B$

$a \in x_A,\ c \in x_C$ If b is on the path connected a, c, $b \in x_B$

② Local Markov



$a \perp \{-a\text{-neighbors}\} \mid \{\text{neighbors}\}$

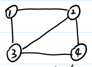
$\rightarrow a \perp \{e, f\} \mid \{b, c, d\}$

③ Pairwise Markov

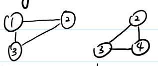
$x_i \perp x_j \mid x_{-i-j} \quad (i \neq j)$

P51 Markov Random Field – Representation – Factorization

Clique (团): A subset of vertices of an undirected graph such that every two distinct vertices in the clique are adjacent.



clique: ①–② ⋯⋯ maximal clique:

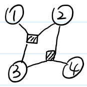


Splitting the undirected graph into the product of maximal clique protential functions

$P(X) = \frac{1}{Z} \prod_{i=1}^{k} \phi(x_{ci})$ $\quad$ k is the number of maximal cliques, x_{ci} is the set of vertices of maximal clique C_i

$\phi(x_{ci}) = \exp\{-E(x_{ci})\}$ $\quad \phi(x)$ is potential functions (势函数) $E(x)$ is energy function (能量函数)

$Z = \sum_{X} \prod_{i=1}^{k} \phi(x_{ci}) = \sum_{x_1}\sum_{x_2}\cdots\sum_{x_p} \prod_{i=1}^{k} \phi(x_{ci})$



according to Hammersley–Clifford theorem? Conditional Independence $\Longleftrightarrow$ Factorization ▨ represent a protential functional

P52 Inference – Introduction

Inference is to find the probability: $P(X) = P(x_1, x_2, \cdots x_p)$

- marginal probability: $P(x_i) = \sum_{x_1} \cdots \sum_{x_{i-1}} \sum_{x_{i+1}} \cdots \sum_{x_p} P(x)$ or $P(x_i) = \int_{x_1 \cdots x_{i-1} x_{i+1} \cdots x_p} P(x)\,dx$
- conditional probability: $P(x_A \mid x_B) \quad X = x_A \cup x_B$
- MAP (maximum a posteriori) Inference: $\hat{z} = \arg\max_{z} P(z \mid x) \propto \arg\max_{z} P(z, x)$

Inference
- Precise Inference
  - Variable Elimination (VE) 变量消除 → Extention
  - Belief Propagation (BP) 信念传播 ⇒ Sum-Product Algorithm (和积算法) ⇒ For tree structure
  - Junction Trees Algorithm 联结树 ⇒ For all graph structure ← Extention
- Approximate Inference
  - Loop Blief propagation ⇒ For cycle
  - Monte Carlo Inference, Importance Sampling, MCMC
  - Variation Inference (变分推断)

P53 Inference – Variable Elimination

ⓐ → ⓑ → ⓒ → ⓓ $\quad$ Assume that a, b, c, d are discrete variable

$P(d) = \sum_{a,b,c} P(a,b,c,d)$

$= \sum_{a,b,c} P(a)P(b|a)P(c|b)P(d|c)$

$= P(a=A_0)P(b=B_0|a=A_0)P(c=C_0|b=B_0)P(d|c=C_0) + \cdots + P(a=A_m)P(b=B_m|a=A_m)P(c=C_m|b=B_m)P(d|c=C_m)$

Variable Elimination: $P(d) = \sum_{a} P(a)P(b|a) \sum_{b,c} P(c|b)P(d|c)$

$= \sum_{b} \phi_a(b) P(c|b) \sum_{c} P(d|c) = \sum_{c} \phi_b(c) P(d|c) = \phi_c(d)$

↓ $\quad$ Function on b

Using the Multipliative distribution law

$ab + ac = a(b+c)$

Undirected graphs are the same as directed graphs

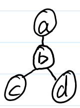
Disadvantages of VE ① repetitive computation ② order of elimination

P54 Inference – Variable Elimination to Belief propagation

ⓐ → ⓑ → ⓒ → ⓓ $\quad P(d) = \underbrace{\sum_{c} P(d|c) \sum_{b} P(c|b) \underbrace{\sum_{a} P(b|a)P(a)}_{m_{a\to b}(b)}}_{m_{b\to c}(c)}$ $\quad$ Forward Algorithm

ⓐ → ⓑ → ⓒ → ⓓ $\quad P(c) = \sum_{b} P(c|b) \sum_{a} P(a)P(b|a) \sum_{d} P(d|c)$ $\quad$ Forward-Backward Algorithm

→ → ←

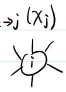


$P(a,b,c,d) = \frac{1}{Z} \psi_a(a)\cdot\psi_b(b)\cdot\psi_c(c)\cdot\psi_d(d)\cdot\psi_{ab}(a,b)\cdot\psi_{bc}(b,c)\cdot\psi_{cd}(c,d)$

$\begin{cases} m_{b\to a}(x_a) = \sum_{x_b} \psi_{ab}\, \psi_b\, m_{c\to d}(x_b)\, m_{d\to b}(x_b) \\ P(a) = \psi_a m_{b\to a}(x_a) \end{cases}$ $\quad x_i$ is a random variable of one of vertexs

⇓ generalize

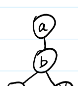
$\begin{cases} m_{j\to i}(x_i) = \frac{1}{Z} \sum_{x_j} \psi_{ij}\, \psi_j \prod_{k \in NB(j)-i} m_{k\to j}(x_j) \\ P(i) = \psi_i \prod_{k\in NB(i)} m_{k\to i}(x_i) \end{cases}$ $\quad$ #NB: Neighborhood

⟶  vertex i may have many branches

Belief propagation: Only computing $m_{j\to i}(x_i)$ instead of computing margin probabilities $P(a), P(b) \cdots$

P55 Inference – Belief Propagation

BP: $m_{j\to i} = \sum_{j} \psi_{ij} \underbrace{\psi_j \prod_{k\in NB(j)-i} m_{k\to j}(x_j)}_{\text{belief}(j)}$



$\begin{cases} \text{belief}(b) = \psi_b\, \text{children}(b) \\ P(a) = \sum_{b} \psi_{ab}\, \text{belief}(b) \end{cases}$

Take messaging as an example. Collect information of b and b's children vertexs. Finally plus the information between a, b

Belief Propagation (Using graph travelsal) (Sequential Implementation)

① Get root, assume a is root

② Collect message, pseudocode ⇒ for x in NB(root)

collect(x)

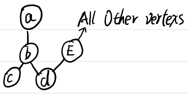
③ Distribute message for x in NB(root)
distribute(x)

Then get $m_{j\to i}$ for all $i, j \in V$

P56.57 Max-Product

Max-Product is an improvement of Belief Propagation (Sum-Product)

$$\begin{cases} \max_{x_1 \cdots x_p} P(x_1 \cdots x_p | E) = \max_{x_{last}} \psi_{last} \prod_{k \in NB(last)} m_{k\to last}(x_{last}) \\ m_{j\to i} = \max_{x_i} \psi_{ij} \psi_j \prod_{k \in NB(j)-i} m_{k\to j}(x_j) \end{cases}$$

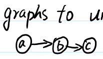
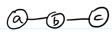


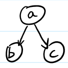
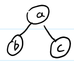
$$\hat{x}_a, \hat{x}_b, \hat{x}_c, \hat{x}_d = \underset{x_a,x_b,x_c,x_d}{\operatorname{argmax}} P(x_a, x_b, x_c, x_d | E)$$

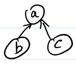
The final solution are $\hat{x}_a, \hat{x}_b, \hat{x}_c, \hat{x}_d$ and $\max_{x_a,x_b,x_c,x_d} P(x_a, x_b, x_c, x_d | E)$

P59 Moral Graph

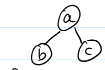
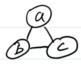
Convert directed graphs to undirected graphs for more generalization

① head to tail  $P(a,b,c) = \underbrace{P(a)P(b|a)}_{\phi(a,b)} \underbrace{P(c|b)}_{\phi(b,c)}$ (two clique) $\Rightarrow$ 

② tail to tail  $P(a,b,c) = \underbrace{P(a)P(b|a)}_{\phi(a,b)} \underbrace{P(c|a)}_{\phi(a,c)} \Rightarrow$ 

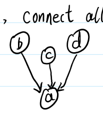
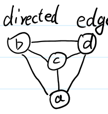
③ head to head  $P(a,b,c) = \underbrace{P(b)P(c)P(a|b,c)}_{\phi(a,b,c)}$

If the converted graph is like this, $\{a,b,c\}$ is not a clique

 $\Rightarrow$ 

So it should connect b, c

All in all, connect all parent vertexs of x_i, and convert all the directed edge to undirected edge

e.g.  $P(a,b,c,d) = P(b)P(c)P(d)P(a|b,c,d)$ $\Rightarrow$ 

P58 Factor Graph

A factor graph is a biopartite graph representing the factorization of a function
(二分图)

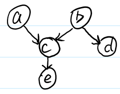
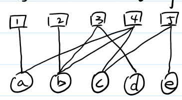
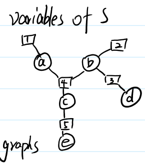
∵ A directed graph can be converted to an undirected graph (Moral Graph), but the process may introduce cycle

∴ Factor graph have two effects:

① Remove the cycles from the graphs and transform them into acyclic graphs

② Make the computing easy

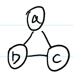
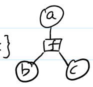
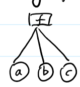
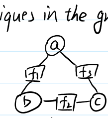
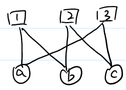
$P(x) = \prod_S f_S(x_S)$, S is a subset of verters of the graph, x_S is the subset of random variables of S

 $P(a,b,c,d,e) = \underbrace{P(a)}_{f_1(a)}\underbrace{P(b)}_{f_2(b)}\underbrace{P(d|b)}_{f_3(b,d)}\underbrace{P(c|a,b)}_{f_4(a,b,c)}\underbrace{P(e|c)}_{f_5(c,e)} \Rightarrow$  & 

$P(a,b,c,d,e) = f_1(a) f_2(b) f_3(b,d) f_4(a,b,c) f_5(c,e)$

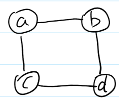
\# Directed graphs can be converted into undirected graph and then represented as factor graphs

An undirected graph can be represented as a varity of factor graphs depending on the cliques in the graph

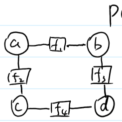
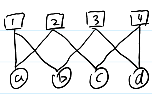
 $\Rightarrow$ clique: $\{a,b,c\}$  &  or cliques: $\{a,b\}\{b,c\}\{a,c\}$  & 

$P(a,b,c) = \frac{1}{Z} f(a,b,c)$

$P(a,b,c) = \frac{1}{Z} f_1(a,b) f_2(b,c) f_3(a,c)$

 cliques: $\{a,b\}\{a,c\}\{b,d\}\{c,d\}$

$P(a,b,c,d) = \frac{1}{Z}\phi_1(a,b)\phi_2(a,c)\phi_3(b,d)\phi_4(c,d)$

$\Rightarrow$  & 

$P(a,b,c,d) = \frac{1}{Z} f_1(a,b) f_2(a,c) f_3(b,d) f_4(c,d)$