Probabilistic Graphical Models

Chao Liu

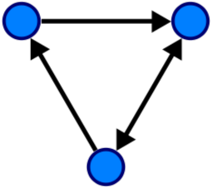
Ding Xiang Yuan

February 24, 2018

Outline

What is graph

- A graph data structure consists of a finite (and possibly mutable) set of vertices or nodes or points, together with a set of unordered pairs of these vertices for an undirected graph or a set of ordered pairs for a directed graph. These pairs are known as edges, arcs, or lines for an undirected graph and as arrows, directed edges, directed arcs, or directed lines for a directed graph.



Graph is everywhere in the field of computer science

- all the trees(avl tree, binary search tree, red-black tree...)
- dijkstra's algorithm, maximum flow algorithm...
- deep learning computing framework: Mxnet, Tensorflow

What is probability graph

- a graph comprises nodes (also called vertices) connected by links (also known as edges or arcs).
- each node represents a random variable (or group of random variables), and the links express probabilistic relation- ships between these variables
- the graph then captures the way in which the joint distribution over all of the random variables can be decomposed into a product of factors each depending only on a subset of the variables.

Why we need to study probability graph

- It provide a simple way to visualize the structure of a probabilistic model and can be used to design and motivate new models
- insights into the properties of the model, including conditional independence properties, can be obtained by inspection of the graph
- complex computations, can be expressed in terms of graphical manipulations, in which underlying mathematical expressions are carried along implicitly.

Bayesian network(directed graph)

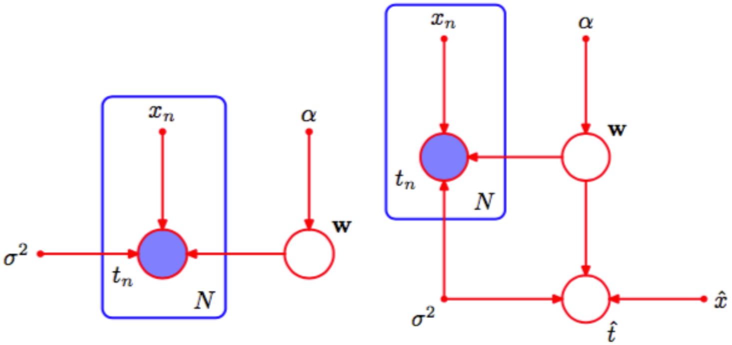
- the links of the graph have a particular directionality indicated arrows
- useful to expressing the causal relationships between random variables
- examples: ploynoimal regression

$$p(\mathbf{t}, w|x, \alpha, \sigma^2) = p(w|\alpha) \prod_{n=1}^{N} p(t_n|w, x_n, \sigma^2) \tag{1}$$

with prediction value:

$$p(\hat{t}, \mathbf{t}, w|\hat{x}, \mathbf{x}, \alpha, \sigma^2) = p(w|\alpha)p(\hat{t}|\hat{x}, w, \sigma^2)[\prod_{n=1}^{N} p(t_n|w, x_n, \sigma^2)] \tag{2}$$

Figure: figure for equation 1 and 2



Conditional independence

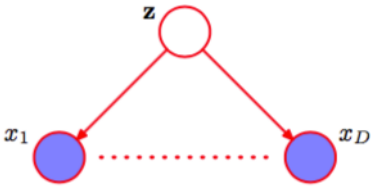
defination: $p(a|b,c) = p(a|c)$, $a \perp\!\!\!\perp b|c$

- it is very important for simplifying computation
- using D-sparation in directed graph, undirected graph however, it's more easy to see than directed graph
- example: naive bayes

$$p(D|\mu) = \prod_{n=1}^{N} p(x_n|\mu) \tag{3}$$

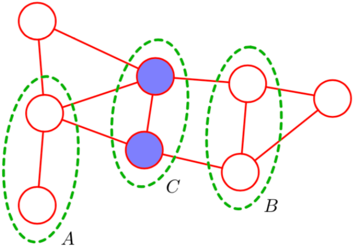
where μ is the prior

Figure: probalisic graph expression of naive bayes



Markov random fields(Undirected graph)

- conditional independence properties: $A \perp\!\!\!\perp B | C$, whether all paths that connect nodes in set A to nodes in the set B.



Factorization properties

- clique: which is defined as a subset of the nodes in a graph such that there exists link between all pairs of nodes in the subset. The set of nodes in a clique is fully connected.
- maximal clique: is a clique such that it is not possible to include any other nodes from the graph
- the joint distribution is written as a product of *potential functions* $\psi(C)$ over the maximal cliques of the graph

$$p(x) = \frac{1}{Z}\prod_{C}\psi(C)(x_C)$$

where x_C denote the node in the clique, and we do not restrict the choice of it. Z is the partition function, give by

$$Z = \sum_{X}\prod_{C}\psi(C)(X_C)$$

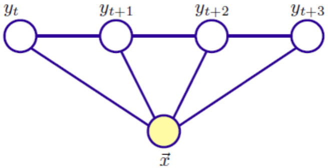
Example: CRF

- CRF(Conditional Random Field) is just a version of an MRF where all the clique poentials are conditioned on input features
- CRF can be any kind of structure, but usually we use a linear chain structure.

$$p(y|x) = \frac{1}{Z(x)} \prod_{j=1} \psi_j(y, x)$$

where we define the poential function to be:

$$\psi_j(x, y) = \exp(\sum_{i=1}^{m} \lambda_i f_i(y_{j-1}, y_j, x, j))$$



Inference

- exact inference and approximation method.
- exact inference: sum-product, max-sum algorithm
- approximation: variational methods and sampling methds etc.

Reference I

Bishop 2006
Pattern Recognition and Machine Learning.
Springer, 2009.