Introduction to Big Data and Machine Learning
Graphical Models

Dr. Mihail

October 29, 2019

Graphical Models

Probability

- Sum rule and product rule of probability

Graphical Models

Probability

- Sum rule and product rule of probability
- Sum rule: if there are n ways to do A and m ways to do B, then the number of ways to do A **or** B is $n + m$, if A and B are independent

Graphical Models

Probability

- Sum rule and product rule of probability
- Sum rule: if there are n ways to do A and m ways to do B, then the number of ways to do A **or** B is $n + m$, if A and B are independent
- Product rule: if there are n ways to do A and m ways to do B, then the number of ways to do A **and** B is nm

Graphical Models

Probability

- Sum rule and product rule of probability
- Sum rule: if there are n ways to do A and m ways to do B, then the number of ways to do A **or** B is $n + m$, if A and B are independent
- Product rule: if there are n ways to do A and m ways to do B, then the number of ways to do A **and** B is nm
- Almost all inference and learning manipulations in ML can be expressed by repeated application of sum rule and product rule

Diagrams help

Diagrammatic representations

- We could formulate and solve probabilistic models by using only algebraic manipulations
- It is advantageous to augment analysis using diagrammatic representations of probability distributions, called *probabilistic graphical models*
- They offer several advantages:
  1. They provide a simple way to visualize the structure of a probabilistic model and can be used to design and motivate new models
  2. Insights into the properties of the model, including conditional independence properties, can be obtained by inspection of the graph
  3. Complex computations, required to perform inference and learning in sophisticated models, can be expressed in terms of graphical manipulations, in which the underlying mathematical expressions are carried along implicitly

Graphs

Definitions

- A graph comprises of a set of nodes (also called vertices) connected by links (also known as edges or arcs)
- In a probabilistic graphical model, each node represents a random variable (or group of random variables) and the links express probabilistic relationships between
- 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 variables
- There are two main types:
  1. Directed graphical models, also known as Bayesian Networks
  2. Undirected graphical models, also known as Markov Random Fields

Bayes Nets

Example

- Consider an arbitrary joint distribution $p(a, b, c)$, over three variables a, b and c

Bayes Nets

Example

- Consider an arbitrary joint distribution $p(a, b, c)$, over three variables a, b and c
- Applying the product rule, we can write:

$$p(a, b, c) = p(c|a, b)p(a, b) \tag{1}$$

Bayes Nets

Example

- Consider an arbitrary joint distribution $p(a, b, c)$, over three variables a, b and c
- Applying the product rule, we can write:

$$p(a, b, c) = p(c|a, b)p(a, b) \tag{1}$$

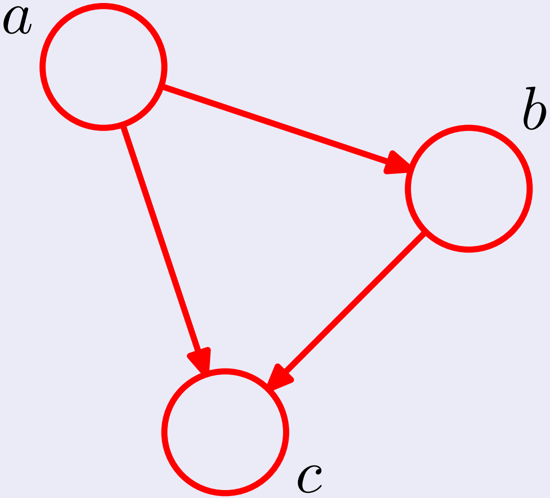
- After a second application of the product rule

$$p(a, b, c) = p(c|a, b)p(b|a)p(a) \tag{2}$$

- This decomposition holds for ANY distribution

Graphical Representation

$p(a, b, c) = p(c|a, b)p(b|a)p(a)$



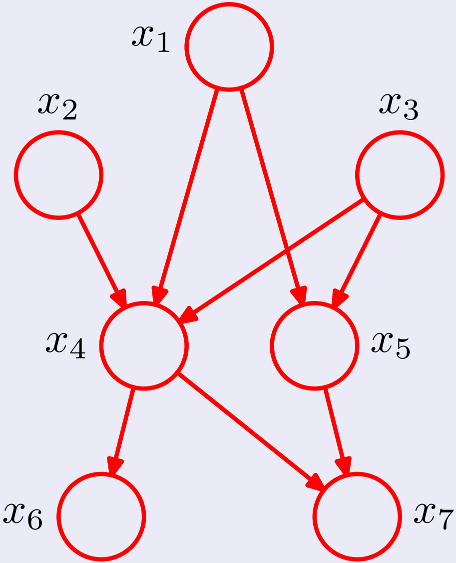
In general

For K variables

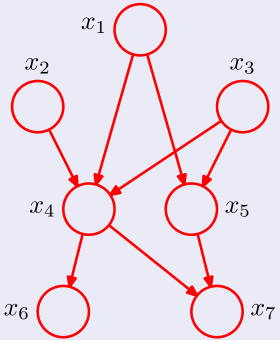
- $p(x_1, \ldots, x_K) = p(x_K | x_1, \ldots, x_{K-1}) \ldots p(x_2 | x_1) p(x_1)$
- This graph is fully connected, since there is a link between every pair of nodes
- It is the **absence** of links that conveys interesting information

Another example

Consider



Joint Distribution



- $p(x_1)p(x_2)p(x_3)p(x_4|x_1,x_2,x_3)p(x_5|x_1,x_3)p(x_6|x_4)p(x_7|x_4,x_5)$
- The joint is given by the *product* over all the nodes in the graph, of a conditional distribution In general:

$$p(x) = \prod_{k=1}^{K} p(x_k|pa_k) \tag{3}$$

Example: polynomial regression

- The random variables in this model are the vector of polynomial coefficients w and the observed data $t = (t_1, \ldots, t_N)^T$
- Input data $x = (x_1, \ldots, x_N)^T$
- Noise variance σ^2
- Precision of Gaussian over w is α

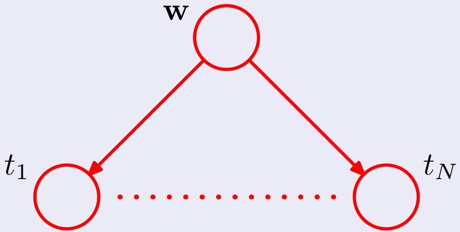
Random variables

- The joint distribution is given by the prior $p(w)$ and N conditional distributions $p(t_n|w)$ for $n = 1, \ldots, N$, so that:

$$p(t, w) = p(w) \prod_{n=1}^{K} p(t_n|w) \tag{4}$$

Graphical Model

Many arcs



Graphical Model

Many arcs

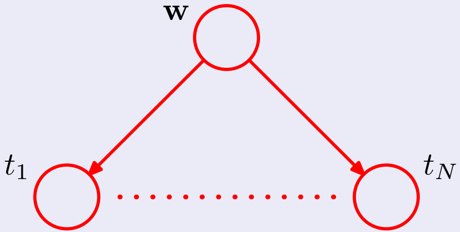
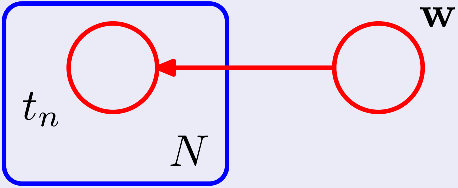
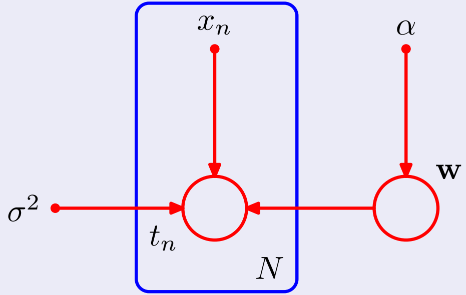


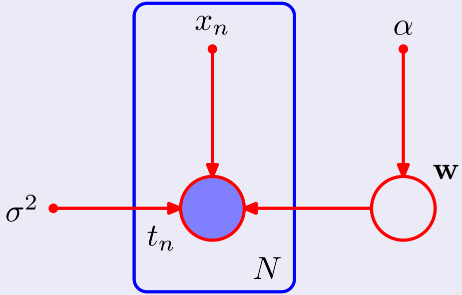
Plate notation



Showing deterministic parameters explicitly



Showing deterministic parameters explicitly



- Observed varibles are shaded

Conditional Independence

- Consider three variables: a, b and c
- Suppose that the conditional distribution of a, given b and c is such that it does not depend on the value of b:

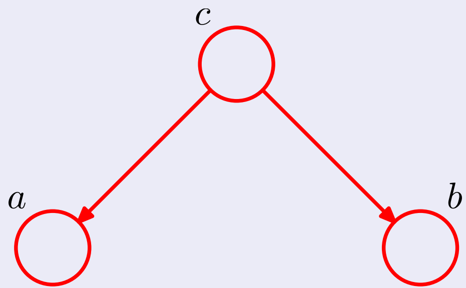
$$p(a|b,c) = p(a|c) \tag{5}$$

- We say that a is **conditionally independent** given of b given c
- This can be expressed as follows:

$$\begin{aligned} p(a,b|c) &= p(a|b,c)p(b|c) \\ &= p(a|c)p(b|c) \end{aligned} \tag{6}$$

- Conditioned on c, the joint distribution of a and b factorizes into the product of the marginal distribution of a and the marginal distribution of b
- Variables a and b are statically independent, given c

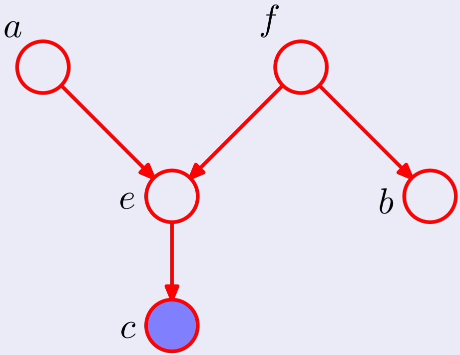
Conditional Independence



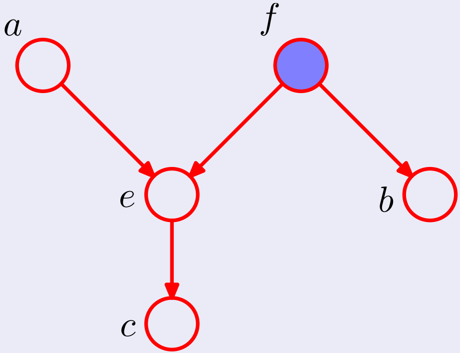
D-separation

- Consider a directed graph in which A, B, and C are arbitrary, non-intersecting set of nodes
- We want to ascertain whether a particular conditional independence statement $A \perp\!\!\!\perp B|C$ is implied by a given directed acyclic graph
- To do so, we consider all possible paths from any node in A to any node in B
- Any such path is said to be *blocked* if it includes a node such that either:
    1. The arrows on the path meet either head-to-tail or tail-to-tail at the node, and the node is in the set C or
    2. The arrows meet head-to-head at the node, and neither the node, nor any of its descendants, is in the set C

Illustration



- The path from a to b is not blocked by node f because it is tail-to-tail node for this path, and is not observed, nor is it blocked by node e because, although the latter is a head-to-head node, it has a descendant c in the conditioning set.
- Thus, $a \perp\!\!\!\perp b|c$ does **NOT** follow



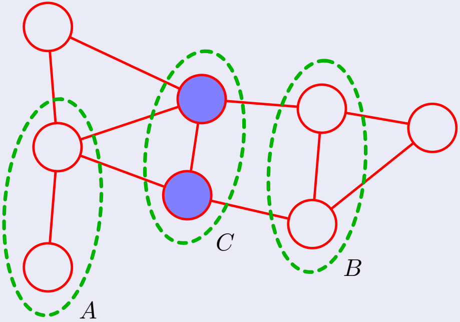
- The path from a to b is blocked by node f because this is a tail-to-tail node that is observed. It is also blocked by node e.

Markov Random Fields

Definition

- A Markov Random Field (MRF) has a set of nodes each of which corresponds to a variable or group of variables, as well as the set of links which connects a pair of nodes
- The links are not directed
- This means conditional independence is now simply determined by graph separation

MRF Conditional Independence



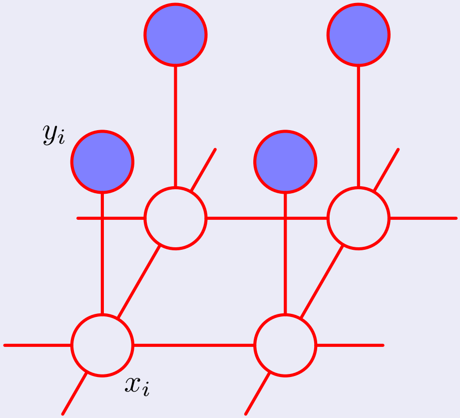
- Here, every path from every node in the set A to every node in the set B passes through at least one node in the set C

MRF application

Image denoising

- Consider an observed, noisy image described by an array of binary pixel values $y_i \in \{-1, +1\}$, where the index $i = 1, \ldots, D$ runs over all pixels
- We shall suppose that the image is obtained by taking an unknown noise-free image, described by binary pixel values $x_i \in \{-1, +1\}$ and randomly flipping the sign of pixels with some small probability
- Because the noise level is small, we know that there will be a strong correlation between x_i and y_i
- This knowledge is captured using an MRF

MRF



- An undirected graphical model representing a MRF for image de-noising

MRF cliques

Two types

- $\{x_i, y_i\}$ have an associated energy function that expresses the correlation between these variables. We pick a simple one $-\eta x_i y_i$ (η-eta) where the energy is lowest when they share the same sign
- $\{x_i, x_j\}$ pairs, neighboring pixels. Here, we can also choose a simple energy function, such as $-\beta x_i x_j$ where β is a positive constant

Model

Energy function

$$E(x, y) = h \sum_i x_i - \beta \sum_{\{i,j\}} x_i x_j - \eta \sum_i x_i y_i \tag{7}$$

Probability distribution

$$p(x, y) = \frac{1}{Z} e^{-E(x,y)} \tag{8}$$

Inference

ICM

- Iterated Conditional Modes
- Simple idea: coordinate-wise gradient ascent
- Steps:
  1. Initialize x_i by $x_i = y_i$ for all i
  2. Repeat until convergence, one node at a time, evaluate total energy for the two possible states $x_i = -1$ and $x_i = -1$, pick lowest