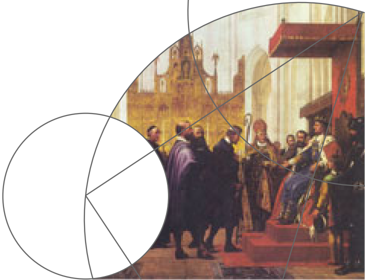
Graphical models

Elements of Machine Learning

Jens Petersen

with content from Ramya Narasimha & Radu Horaud
https://lear.inrialpes.fr/~jegou/bishopreadinggroup/chap8.pdf and
Pattern Recognition and Machine Learning,
Christopher M. Bishop



About this lecture

Probabilistic graphical models

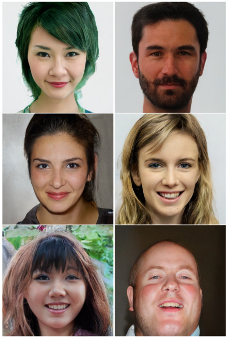
- Directed and undirected graphical models
- Conditional independence in graphical models

Why graphical models?

- Visualize the structure of a probabilistic model
- Insights into properties of the model (such as conditional independence)
- Complex computations, required for inference, can be expressed in terms of simpler graphical manipulations.

Graphical model examples

- Mixture models, factor analysis, hidden Markov models, Markov random fields, Kalman filters, Ising models, Deep Belief Networks, ...
- Neural networks can be interpreted as graphical models, and hybrid models of NN and other types of graphical models can be done
  - Allowing neural networks incorporating an element of domain knowledge
  - Generative neural network models, such as variational autoencoders (covered later).



Synthetic faces generated using a variational autoencoder [1]

[1] Generating Diverse High-Fidelity Images with VQ-VAE-2, Razavi et al., 2019

Definitions

- Graph
  - Nodes (vertices) + links (arcs, edges)
  - Node: a random variable
  - Link: a probabilistic relationship
- **Directed graphical models** or **Bayesian Networks**
  - useful to express causal relationships between variables
- **Undirected graphical models** or **Markov random fields**
  - useful to express soft constraints between variables
- **Factor graphs**
  - convenient for solving inference problems

Directed graphical model - three variables

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

Directed graphical model - three variables

$$
\begin{aligned}
p(a, b, c) = & \, p(c|a, b)p(a, b) \\
& \text{(repeated application of product rule)}
\end{aligned}
$$

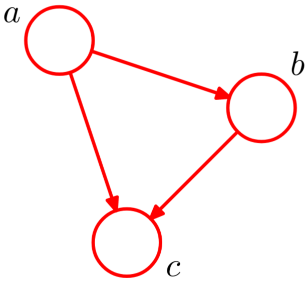
Directed graphical model - three variables

$$
\begin{aligned}
p(a, b, c) =& p(c|a, b)p(a, b) \\
=& p(c|a, b)p(b|a)p(a) \\
& \text{(repeated application of product rule)}
\end{aligned}
$$

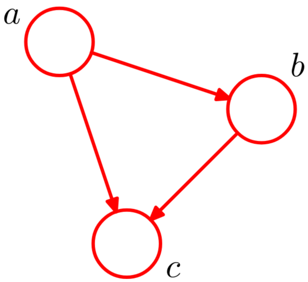
Note the lack of symmetry on the right hand side (a different order could have been chosen)

Directed graphical model - three variables

$$
\begin{aligned}
p(a, b, c) =& p(c|a, b)p(a, b) \\
=& p(c|a, b)p(b|a)p(a) \\
& \text{(repeated application of product rule)}
\end{aligned}
$$



Directed graphical model - three variables



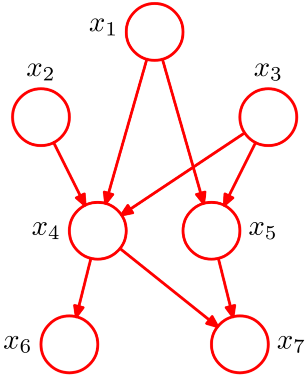
- a is **parent** node of b
- b is **child** node of a

Directed graphical models - K variables

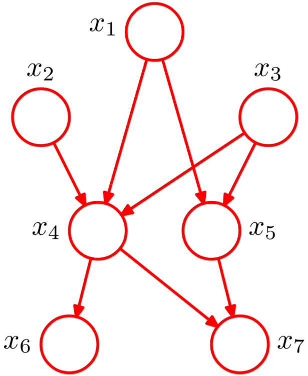
$$p(x_1, \ldots, x_K) = p(x_K | x_1, \ldots, x_{K-1}) \ldots p(x_2 | x_1) p(x_1)$$

- The associated graph is **fully connected** (there is a link between every pair of nodes)
- The absence of links conveys important information

Directed graphical models - a non fully connected example



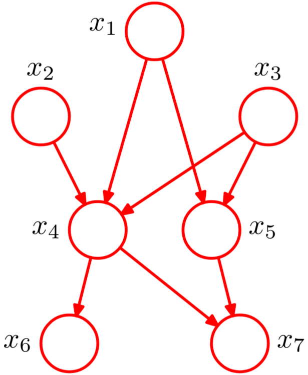
Directed graphical models - a non fully connected example



Lets write the joint distribution

$$p(x_1, \dots x_7) =$$

Directed graphical models - a non fully connected example



Lets write the joint distribution

$$
\begin{aligned}
p(x_1, \ldots x_7) = & p(x_1)p(x_2)p(x_3) \\
& p(x_4|x_1, x_2, x_3)p(x_5|x_1, x_3)p(x_6|x_4)p(x_7|x_4, x_5)
\end{aligned}
$$

Directed graphical models - *K* variables and arbitrary connectivity

The joint distribution is given by the product, over all nodes, of the distribution of each node conditioned on its parents.

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

where $\mathbf{x} = \{x_1, \ldots, x_K\}$, x_k is each node and pa_k the corresponding parents.

Directed graphical models - K variables and arbitrary connectivity

The joint distribution is given by the product, over all nodes, of the distribution of each node conditioned on its parents.

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

where $\mathbf{x} = \{x_1, \ldots, x_K\}$, x_k is each node and pa_k the corresponding parents.

- **factorization** properties of the joint distribution
- Only considering graphs with no **directed cycles** (cannot move from parent to child repeatedly and visit the same node twice)
- same as saying the graph is a **directed acyclic graph (DAG)**

Polynomial regression example

Target variable $\mathbf{t}$, given by $\mathbf{t} = y(\mathbf{x}, \mathbf{w}) + \epsilon$.

- random variables: polynomial coefficients $\mathbf{w}$ and the observed data $\mathbf{t} = \{t_1, \dots t_N\}$
- deterministic (input) variables $\mathbf{x} = (x_1, \dots, x_N)$, noise with variance σ^2, and α representing the precision of the Gaussian prior over $\mathbf{w}$, $p(\mathbf{w})$.

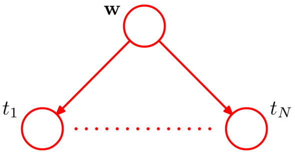
Focusing on the random variables, we can write

$$p(\mathbf{t}, \mathbf{w}) = p(\mathbf{w}) \prod_{n=1}^{N} p(t_n|\mathbf{w}),$$

Polynomial regression example

$$p(\mathbf{t}, \mathbf{w}) = p(\mathbf{w}) \prod_{n=1}^{N} p(t_n | \mathbf{w}),$$

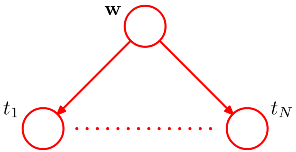
We can express this in graphical form as



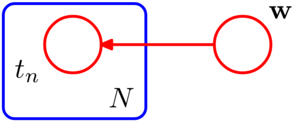
Polynomial regression example

$$p(\mathbf{t}, \mathbf{w}) = p(\mathbf{w}) \prod_{n=1}^{N} p(t_n|\mathbf{w}),$$

We can express this in graphical form as



or as



The box is called a **plate**.

Polynomial regression example

We also have input data $\mathbf{x} = (x_1, \ldots, x_N)$, noise with variance σ^2, and the hyperparameter α representing the precision of the Gaussian prior over $\mathbf{w}$, $p(\mathbf{w})$.

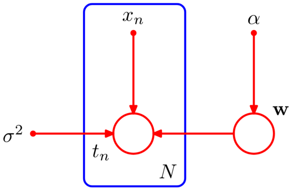
$$p(\mathbf{t}, \mathbf{w}|\mathbf{x}, \alpha, \sigma^2) = p(\mathbf{w}|\alpha) \prod_{n=1}^{N} p(t_n|\mathbf{w}, x_n, \sigma^2),$$

Polynomial regression example

We also have input data $\mathbf{x} = (x_1, \ldots, x_N)$, noise with variance σ^2, and the hyperparameter α representing the precision of the Gaussian prior over $\mathbf{w}$, $p(\mathbf{w})$.

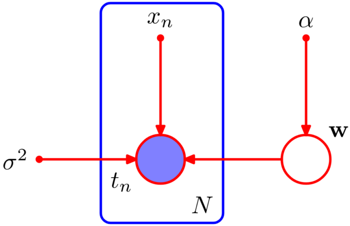
$$p(\mathbf{t}, \mathbf{w}|\mathbf{x}, \alpha, \sigma^2) = p(\mathbf{w}|\alpha) \prod_{n=1}^{N} p(t_n|\mathbf{w}, x_n, \sigma^2),$$

with the graphical representation



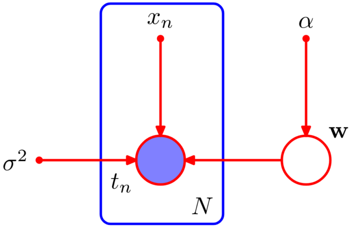
Deterministic parameters shown by small nodes

Polynomial regression example

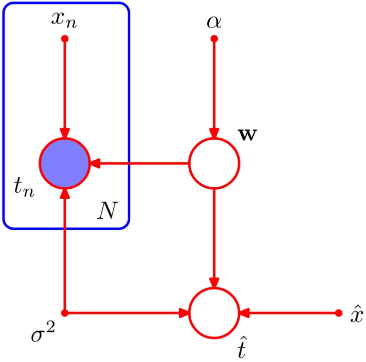


Shaded nodes are **observed values** and non shaded are not observed, called **latent** or **hidden** variables.

Polynomial regression example



Shaded nodes are **observed values** and non shaded are not observed, called **latent** or **hidden** variables.



Making predictions $\hat{t}$ for new input values $\hat{x}$.

Generative vs descriminative models

- What is a generative model and can you give an example of one?

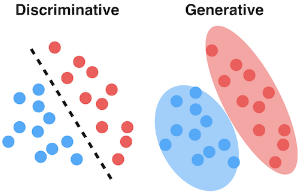
Generative vs descriminative models

- What is a generative model and can you give an example of one?
- What is a descriminative model and can you give an example one?

Generative vs descriminative models

Suppose you want to learn how to classify apples and oranges from images, you could learn to

- **generate** images of apples and oranges and use that to say something about the probability of each given image being apple or orange
- **discriminate** images of apples and oranges and use that to say something about the probability of each given image being apple or orange



1

[1]Image source: https://dataisutopia.com/blog/discremenet-generative-models

Generative vs descriminative models

What are some advantages of generative models?

Generative vs descriminative models

What are some advantages of generative models?

What are some advantages of discriminative models?

Generative models

Going back to the factorization in a DAG

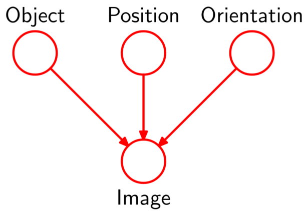
$$p(\mathbf{x}) = \prod_{k=1}^{K} p(x_k | pa_k)$$

Goal: draw a sample $\hat{x}_1, \ldots, \hat{x}_K$ from the joint distribution.

This can be accomplished using **ancestral sampling**

1. order nodes such that each node has a higher number than any of its parents
2. starting from lowest numbered node x_n and upwards
   - draw a sample from the conditional distribution $p(x_n | pa_n)$, with the parents set to their sampled values (we will get back to how to do this in Graphical Models II).

Generative models



Object recognition task, with a model describing how an image, the observed data, is formed from an object and its position and orientation.

- The graphical model above captures the causal process by which the observed data was generated.
- Such models are therefore often called **generative models**.
- We can use sampling in such models to generate new data.
- Contrast this with the polynomial regression model, which is not generative, as $\mathbf{x}$ cannot be sampled without a prior distribution.

Discrete variables

A single discrete variable $\mathbf{x}$ with K possible states (1-of-K representation)

What is 1-of-K representation?

Discrete variables

A single discrete variable **x** with K possible states (1-of-K representation)

$$p(\mathbf{x}|\mu) = \prod_{k=1}^{K} \mu_k^{x_k}$$

with $\mu = (\mu_1, \ldots, \mu_K)$ and $\sum_k \mu_k = 1$, so $K - 1$ parameters need to be specified.

With two variables

$$p(\mathbf{x_1}, \mathbf{x_2}|\mu) = \prod_{k=1}^{K} \prod_{l=1}^{K} \mu_{kl}^{x_{1k} x_{2l}}$$

with $\sum_k \sum_l \mu_{kl} = 1$, so $K^2 - 1$ parameters need to be specified.

Discrete variables

What is the problem with having many parameters in a model?

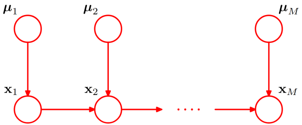
Discrete variables

- Total number of parameters for joint distribution over M variables is $K^M - 1$ (exponential in M)
- Total number of parameters for joint distribution over M independent variables is $M(K - 1)$
- Total number of parameters in the chain below is $K - 1 + (M - 1)K(K - 1)$, which grows linearly with M

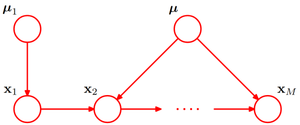
A chain with M discrete nodes, each having K states.

Discrete variables

We can also limit parameters by introducing a Dirichlet prior

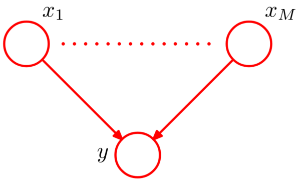


and **sharing** or **tying** parameters



Discrete variables

Suppose we need a conditional distribution $p(y|x_1, \ldots, x_M)$, even if the parameters for $x_1, \ldots, x_M$ are limited, we may need exponentially many for defining $p(y|x_1, \ldots, x_M)$.



But we can introduce parameterizations of the conditional distributions instead of complete tables.

$$p(y = 1|x_1, \ldots x_M) = \sigma(w_0 + \sum_{i=1}^{M} w_i x_i) = \sigma(\mathbf{w}^T \mathbf{x})$$

You should realize that this is more restrictive!

Conditional independence

Consider three variables a, b, and c

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

Then a is **conditionally independent** of b given c.

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

Conditional independence

Consider three variables a, b, and c

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

Then a is **conditionally independent** of b given c.

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

Shorthand $a \perp\!\!\!\perp b|c$.

Conditional independence

- Important because it simplifies the structure of a model
- and simplifies inference and learning.
- Can be tested by repeated application of sum and product rules of probability (time consuming).

Advantage of graphical models

- Conditional independence can be read directly from the graph without having to perform any analytical manipulations
- using **d-separation** (d = directed)

Conditional independence

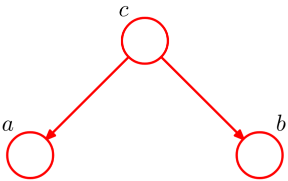
- Important because it simplifies the structure of a model
- and simplifies inference and learning.
- Can be tested by repeated application of sum and product rules of probability (time consuming).

Advantage of graphical models

- Conditional independence can be read directly from the graph without having to perform any analytical manipulations
- using **d-separation** (d = directed)

Lets look at some examples...

Conditional independence - example I



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

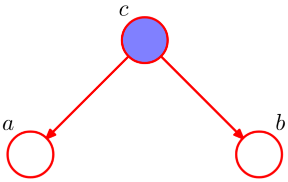
$$p(a, b) = \sum_c p(a|c)p(b|c)p(c) \text{ (marginalization with respect to } c\text{)}$$

$$\neq p(a)p(b) \rightarrow a \not\perp\!\!\!\perp b|\varnothing$$

Conditional independence - example I

On the other hand if we condition on c...

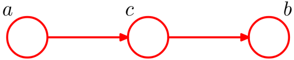
$$
\begin{aligned}
p(a,b|c) &= \frac{p(a,b,c)}{p(c)} \text{(Product rule)} \\
&= \frac{p(a|c)p(b|c)p(c)}{p(c)} \\
&= p(a|c)p(b|c) \rightarrow a \perp\!\!\!\perp b|c
\end{aligned}
$$



Graphical interpretation: consider the path from node a to b the node c is said to be **tail-to-tail** with respect to this path because the node is connected to the tails of the two arrows.

- the path causes the nodes to be dependent.
- the conditioning on c, blocks the path, and causes the nodes to be conditionally independent.

Conditional independence - example II

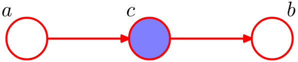


$$
\begin{aligned}
p(a, b, c) &= p(a)p(c|a)p(b|c) \\
p(a, b) &= p(a)\sum_c p(c|a)p(b|c) \text{ (marginalization with respect to } c) \\
&= p(a)\sum_c p(c|a)p(b|c, a) \text{ (exploit } b \perp\!\!\!\perp a|c) \\
&= p(a)\sum_c \frac{p(c, a)p(b, c, a)}{p(a)p(c, a)} \text{(def. cond. inde.)} \\
&= p(a)\sum_c p(b, c|a) \text{(divide out and def. cond. inde.)} \\
&= p(a)p(b|a) \rightarrow a \not\perp\!\!\!\perp b|\varnothing
\end{aligned}
$$

Conditional independence - example II

On the other hand if we condition on c...

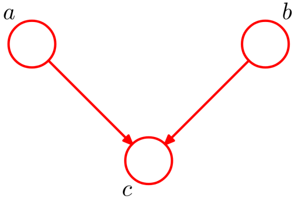
$$
\begin{aligned}
p(a,b|c) &= \frac{p(a,b,c)}{p(c)} \text{(Product rule)} \\
&= \frac{p(a)p(c|a)p(b|c)}{p(c)} \text{(From graph)} \\
&= \frac{p(a,c)}{p(c)} p(b|c) \text{(Product rule)} \\
&= p(a|c)p(b|c) \rightarrow a \perp\!\!\!\perp b|c
\end{aligned}
$$



Graphical interpretation: consider the path from node a to b the node c is said to be **head-to-tail** with respect to this path.

- the path causes the nodes to be dependent.
- the conditioning on c, blocks the path, and causes the conditional independence.

Conditional independence - example III

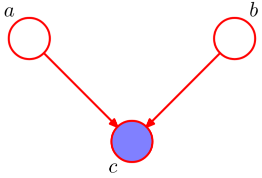


$$p(a, b, c) = p(a)p(b)p(c|a, b)$$
$$p(a, b) = p(a)p(b) \text{ (marginalizing both sides over } c)$$
$$\rightarrow a \perp\!\!\!\perp b|\varnothing$$

Conditional independence - example II

On the other hand if we condition on c...

$$p(a, b|c) = \frac{p(a, b, c)}{p(c)} \text{ (Product rule)}$$
$$= \frac{p(a)p(b)p(c|a, b)}{p(c)}$$



$\rightarrow a \not\perp\!\!\!\perp b|c$

Graphical interpretation: consider the path from node a to b the node c is said to be **head-to-head** with respect to this path.

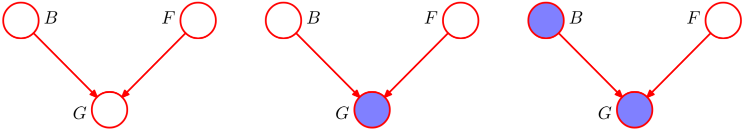
- when node c is unobserved, it blocks the path and the variables a and b are independent.
- However, conditioning on c unblocks the path and renders a and b dependent.

Conditional independence - some terminology

We say node y is a **descendant** of node x if there is a path from x to y in which each step follows the directions of the arrows.

- A head-to-head path will become unblocked if either the node, or any of its descendants, is observed.

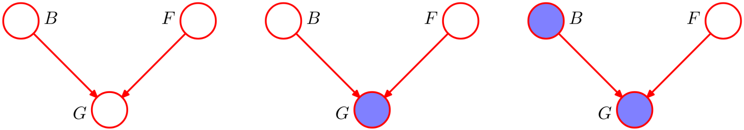
Conditional independence - fuel gauge example



- B: Battery state either 0 or 1
- F: Fuel state either 0 or 1
- G: Gauge reading either 0 or 1

Observing the reading of the gauge G makes the fuel state F and battery state B dependent. Can you give some intuition why this makes sense?

Conditional independence - fuel gauge example



- B: Battery state either 0 or 1
- F: Fuel state either 0 or 1
- G: Gauge reading either 0 or 1

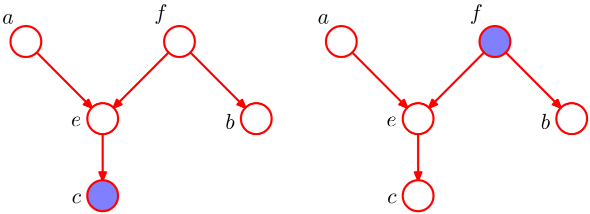
Observing the reading of the gauge G makes the fuel state F and battery state B dependent. Can you give some intuition why this makes sense?

This is because the particular reading may either be due to the battery state or the fuel state. Additionally knowing the battery state thus changes the probability of observing a given fuel state (and conversely). More details and computations can be found in PRML.

D-separation

- D = Directed
- A, B, C: non-intersecting sets of nodes
- To ascertain $A \perp\!\!\!\perp B | C$
  - Consider all paths that are blocked from any node A to any node B
  - Path is said to be a **blocked** path if it includes a node such that
    - 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
    - the arrows meeet head-to-head at the node, and neither the node, nor any of its descendants, is in the set C
  - If all paths are block then A is **d-separated** from B by C, and $A \perp\!\!\!\perp B | C$.

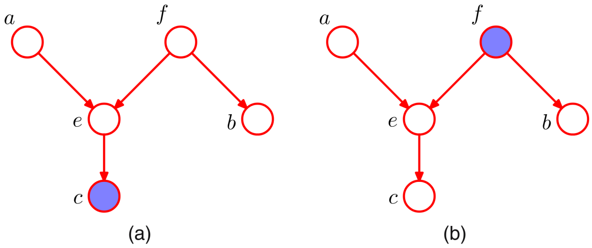
D-separation



(a) (b)

Does $a \perp\!\!\!\perp b | c$ follow from the left graph?

D-separation

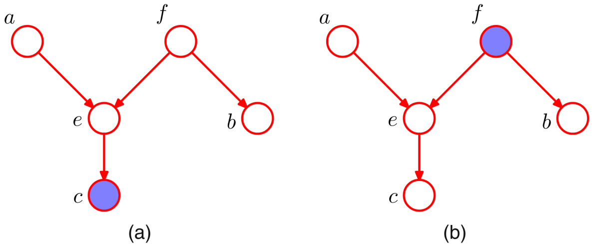


Does $a \perp\!\!\!\perp b | c$ follow from the left graph?

- path is not blocked by node f because it is tail-to-tail and not observed.
- path is not blocked by node e because despite being a head-to-head node it has a descendant c that is observed/in the conditioning set.

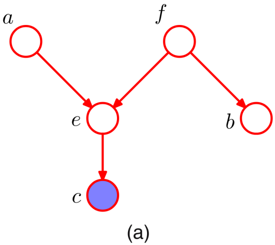
So no!

D-separation



Does $a \perp\!\!\!\perp b | f$ follow from the right graph?

D-separation



Does $a \perp\!\!\!\perp b | f$ follow from the right graph?

- path is blocked by node f because this is a tail-to-tail node that is observed
- path is also blocked by node e because it is a head-to-head node that is not observed nor are its descendants (c)

So yes!

D-separation - additional examples in PRML

- Polynomial regression
  - used to argue training set can be discarded when making predictions
- Interpretating graphical models as a filter allowing all possible distributions through, that
  - respect the conditional independencies implied by the graph
  - satisfies the directed factorization properties
  - d-separation theorem says above two sets are identical ($\mathcal{DF}$ - **directed factorization**).
- Naive Bayes
  - graphical repsentation of Naive Bayes model and conclusions from this

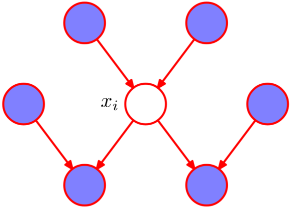
Markov Blanket

Consider a joint distribution $p(\mathbf{x}_1, \ldots, \mathbf{x}_D)$

$$p(\mathbf{x}_i|\mathbf{x}_{j\neq i}) = \frac{\prod_k p(\mathbf{x}_k|pa_k)}{\int \prod_k p(\mathbf{x}_k|pa_k)dx_i}$$

- Factors $p(x_k|pa_k)$ that do not have any functional dependence on $\mathbf{x}_i$ can be taken outside the integral and thus cancel out.
- Only factors remaining are
  - Parents and children of x_i
  - Also co-parents: corresponding to parents of node $\mathbf{x}_k$ (not $\mathbf{x}_i$)

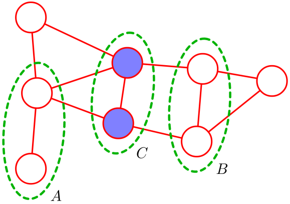
  = **The Markov Blanket** of node $\mathbf{x}_i$.

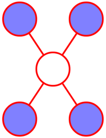


Markov random fields

- Also called **undirected graphical models**
- Nodes correspond to variables or group of variables
- Links within the graph are undirected (no arrows)
- Conditional independence is determined by simple graph separation

Markov random fields - conditional independence





- Consider all possible paths that nodes in set A to nodes in set B.
- If all such paths pass through one or more nodes in set C, then all such paths are blocked $\rightarrow A \perp\!\!\!\perp B | C$
- Testing for conditional independence in undirected graphs is therefore simpler than in directed graphs
- Right: the Markov blanket: consists of the set of neighboring nodes

Markov random fields - factorization properties

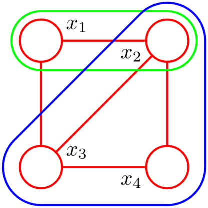
- Consider two nodes x_i and x_j that are not connected by a link then these are conditionally independent given all other nodes
- As there is no direct path between the nodes, and all other paths pass through nodes that are observed and thus blocked

$$p(x_i, x_j|\mathbf{x} \setminus \{i, j\}) = p(x_i|\mathbf{x} \setminus \{i, j\})p(x_j|\mathbf{x} \setminus \{i, j\})$$

This leads to the concept of cliques...

Markov random fields - cliques

- **Clique** - a subset of nodes in a graph such that there exists a link between all pairs of nodes in the subset
- **Maximal clique** - a clique such that it is not possible to include any other nodes from the graph without it ceasing to be a clique
- Joint distribution can thus be factored in terms of maximal cliques



A clique (green) and a maximal clique (blue)

Markov random fields - joint distribution

The joint distribution can be written as a product of potential functions $\phi_{\mathcal{C}}(\mathbf{x}_{\mathcal{C}})$ over the maximal cliques of the graph

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{\mathcal{C}} \psi_{\mathcal{C}}(\mathbf{x}_{\mathcal{C}})$$

Where Z is the partition function

$$Z = \sum_{x} \prod_{\mathcal{C}} \psi_{\mathcal{C}}(\mathbf{x}_{\mathcal{C}})$$

- With M node and K states, the normalization term involves summing over K^M states
- So (worst case) is exponential in the size of the model
- The partition function is needed for parameter learning
- For evaluating local marginal probabilities the unnormalized joint distribution can be used

Hammersley and Clifford theorem

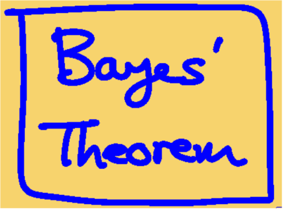
The filter analogy

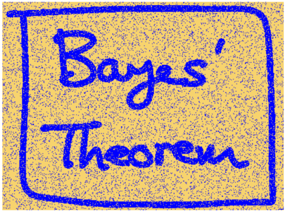
- $\mathcal{UI}$: the set of distributions that are consistent with the set of conditional independence statements read from the graph using graph separation
- $\mathcal{UF}$: the set of distributions that can be expressed as a factorization described with respect to the maximal cliques
- The Hammersley-Clifford theorem states that the sets $\mathcal{UI}$ and $\mathcal{UF}$ are identical if $\psi_{\mathcal{C}}(\mathbf{x}_{\mathcal{C}})$ is strictly positive
- In such case it can be convenient to express them as exponentials (why?)

$$\psi_{\mathcal{C}}(\mathbf{x}_{\mathcal{C}}) = \exp\left(-E(\mathbf{x}_{\mathcal{C}})\right)$$

- Where $E(\mathbf{x}_{\mathcal{C}})$ is called an energy function, and the exponential representation is called the Boltzmann distribution

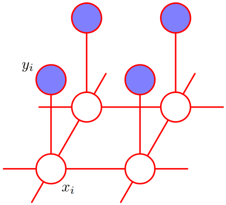
Image denoising example





- Noisy image $y_i \in \{-1, 1\}$ where i runs over all the pixels
- Unknown noise free image $x_i \in \{-1, 1\}$
- Goal: given noisy image recover noise free image

The ising model



Two types of cliques

- $-\eta x_i y_i$ giving a lower energy when x_i and y_i have the same sign and a higher energy when they have the opposite sign
- $-\beta x_i x_j$ the energy is lower when the neighboring pixels have the same sign than when they have the opposite sign
- hx_i acts as a bias (note this could seen as part of either of the above clique potentials.

The complete energy function and joint distribution

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

The joint distribution

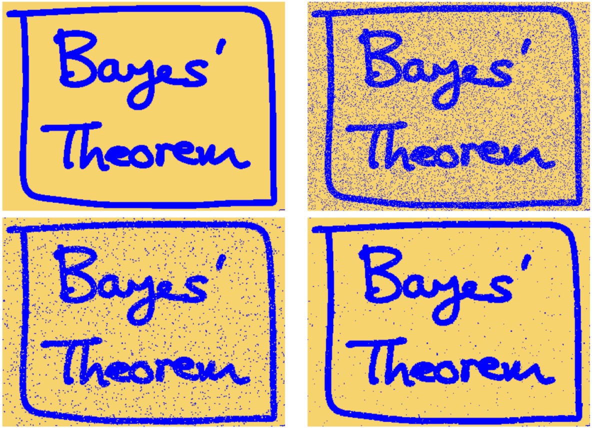
$$p(\mathbf{x}, \mathbf{y}) = \frac{1}{Z} \exp\big(-E(\mathbf{x}, \mathbf{t})\big)$$

Fixing $\mathbf{y}$ as observed values implicitly defines $p(\mathbf{x}|\mathbf{y})$

To obtain the image $\mathbf{x}$ with techniques such as **Iterated Conditional Modes** (ICM)

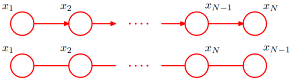
- Initialize the variables $x_i = y_i$ for all i
- For x_j evaluate the total energy for the two possible states $x_j = 1$ and $x_j = -1$ with other node variables fixed
- Set x_j to whichever state has the lower energy
- Repeat the update for another site, and so on, until some suitable stopping criterion is satisfied

The joint distribution



Denoising using MRF, from left to right and top to bottom - original, corrupted, restored using ICM, and restored using graph cut.

Relation to directed graphs - example i



Distribution for

$$\text{directed: } p(\mathbf{x}) = p(x_1)p(x_2|x_1)p(x_3|x_2)\dots p(x_N|x_{N-1})$$

$$\text{undirected: } p(\mathbf{x}) = \frac{1}{Z}\psi_{1,2}(x_1, x_2)\psi_{2,3}(x_2, x_3)\dots\psi_{x_{N-1},x_N}(x_{N-1}, x_N),$$

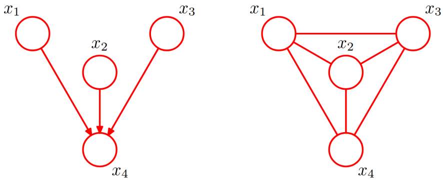
where

$$\begin{aligned}
\psi_{1,2}(x_1, x_2) &= p(x_1)p(x_2|x_1) \\
\psi_{2,3}(x_2, x_3) &= p(x_3|x_2) \\
&\vdots \\
\psi_{N-1,N}(x_{N-1}, x_N) &= p(x_N|x_{N-1})
\end{aligned}$$

Note how marginal of first node is absorbed into the first potential function and $Z = 1$.

Relation to directed graphs - example ii

With more than one parent per node, we cannot simply replace directed edges with undirected...



$$p(\mathbf{x}) = p(x_1)p(x_2)p(x_3)p(x_4|x_1, x_2, x_3)$$

x_1, x_2, x_3, and x_4 must belong to a single clique...

- Add extra links between all pairs of parents
- Anachronistically, this process of 'marrying the parents' has become known moralization
- The resulting undirected graph, is called the moral graph

Moralization

Procedure

- Add additional undirected links between all pairs of parents for each node in the graph
- Drop the arrows on the original links to give the moral graph
- Initialize all of the clique potentials of the moral graph to 1
- Take each conditional distribution factor in the original directed graph and multiply it into one of the clique potentials

Consequences of the procedure

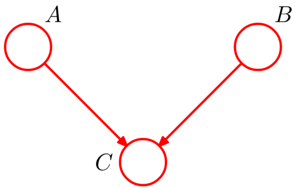
- There will always exist at least one maximal clique that contains all of the variables in the factor as a result of the moralization step
- Going from a directed to an undirected representation discards some conditional independence properties from the graph

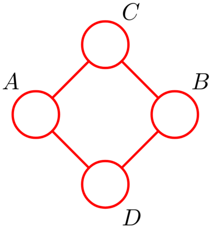
D-map and I-maps

Directed and undirected graphs express different conditional independence properties

- **D-map** of a distribution: every conditional independence statement satisfied by the distribution is reflected in the graph
- A graph with no links will be trivial D-map
- **I-map** of a distribution: every conditional independence statement implied by a graph is satisfied by a specific distribution
- Fully connected graph will give I-map for any distribution
- Perfect map: is both D-map and I-map

D-map and I-maps - examples





- Left
  - perfect map for $A \perp\!\!\!\perp B | \emptyset$ and $A \not\!\perp\!\!\!\perp B | C$
  - has no corresponding undirected graph that is a perfect map
- Right
  - perfect map for $A \not\!\perp\!\!\!\perp B | \emptyset$, $C \perp\!\!\!\perp D | A \cup B$ and $A \perp\!\!\!\perp B | C \cup D$
  - has no corresponding directed graph that is a perfect map

Questions?