Math primer for the Neural Networks course

Roberto Esposito
roberto.esposito@unito.it

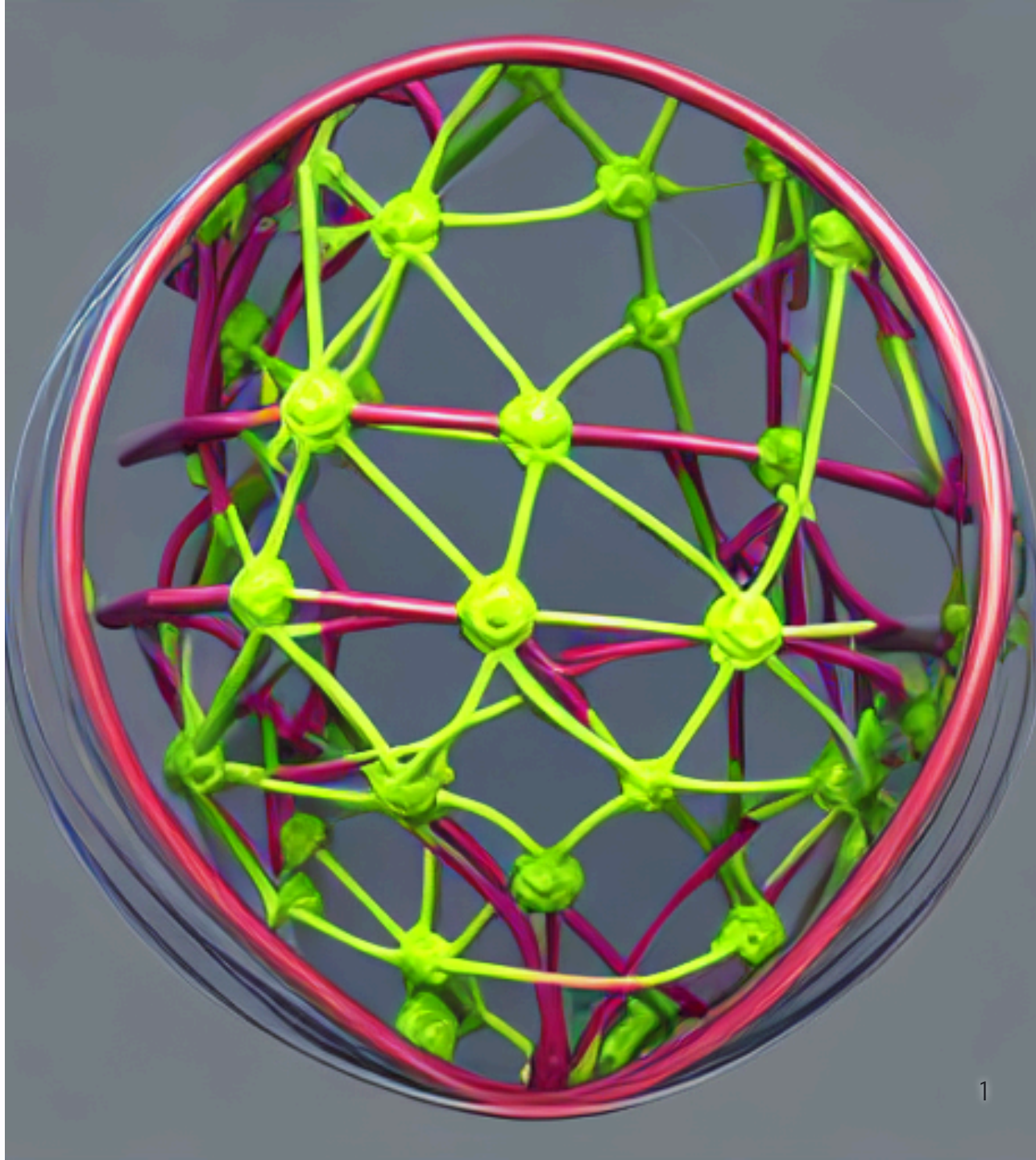


Image dreamed by stable diffusion
Prompt: "information theory, 3d model"

Information theory knowledge poll

Information Theory

Information Theory

Information theory is a branch of applied mathematics that revolves around quantifying how much information is present in a signal.

In this course, we mostly use a few key ideas from information theory to **characterize probability distributions** or **quantify similarity between probability distributions**.

Quantifying information

The basic intuition behind information theory is that the quantity of information carried by a message depends on how likely it is: learning that an **unlikely event** has occurred **is more informative** than learning that a likely event has occurred.

Example

Learning that today rained in the Sahara desert is more informative than learning that today rained in London.

We want to formalize this intuition:

- An event with probability 100% is **perfectly unsurprising and yields no information**.
- The **less probable** an event is, the more surprising it is and the **more information** it yields.
- **Independent events should have additive information**. For example, finding out that a tossed coin has come up as heads twice should convey twice as much information as finding out that a tossed coin has come up as heads once.

più il messaggio è improbabile più è alta la quantità di informazione che esso veicola

Shannon's Entropy Measure

We can quantify the uncertainty of an event using the concept of *self-information*:

$$I(x) = -\log P(x)$$

Shannon entropy captures the average amount of "information" across all possible outcomes of a random variable:

$$H(\mathrm{x}) = E_{\mathrm{x}\sim P}[I(\mathrm{x})] = -\sum_{x} P(x) \log P(x)$$

expected della self-information

Shannon's Source Coding Theorem states that $H(\mathrm{x})$ provides a lower bound for the average length of codewords in an optimal encoding of the possible values of x.

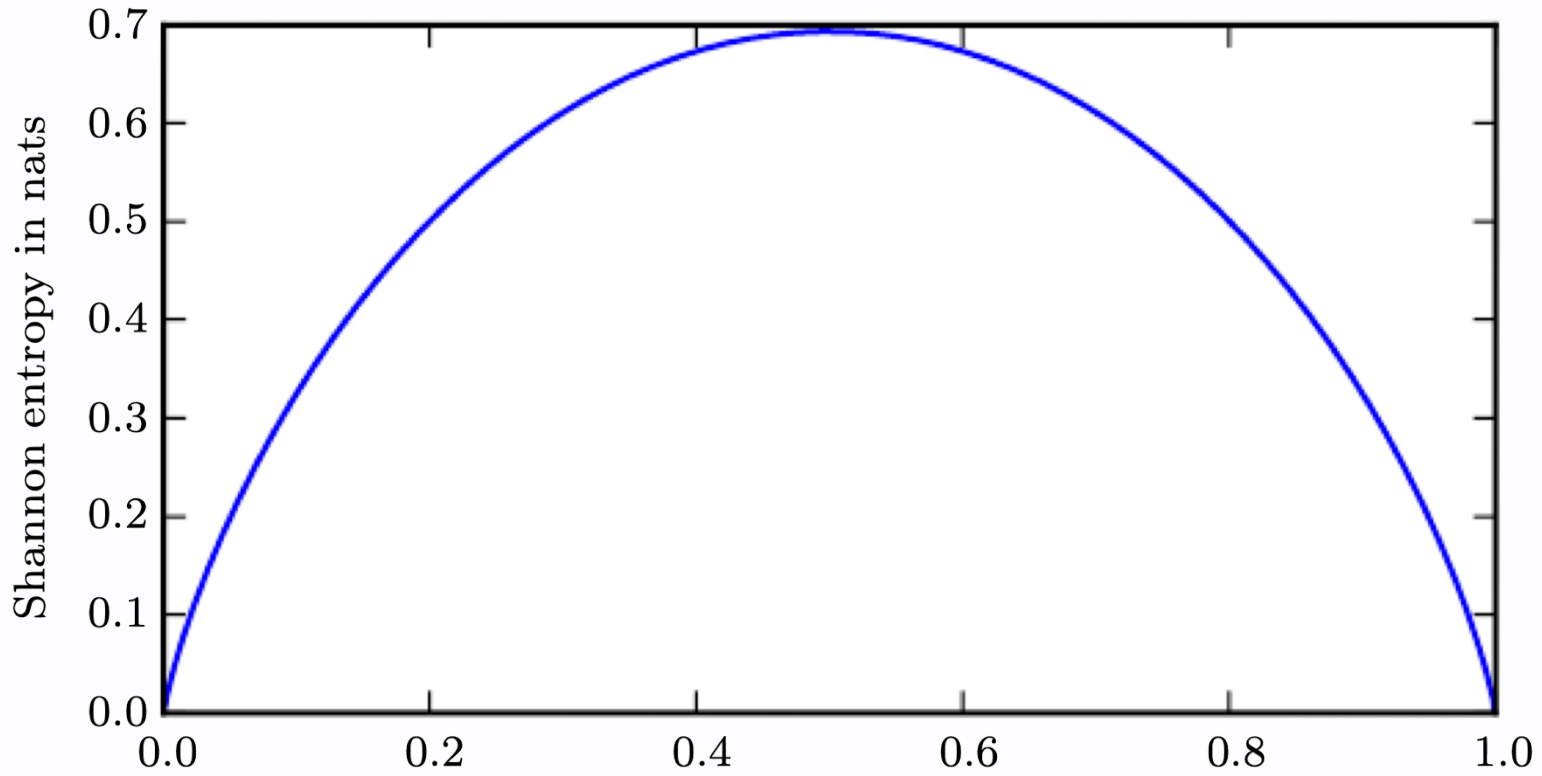


Figure: The entropy of a random variable $\mathbf{x} \sim \mathrm{Bernoulli}(\phi)$ as ϕ varies from 0 to 1.

Kullback-Leibler (KL) divergence

If we have two separate distributions $P(\mathbf{x})$ and $Q(\mathbf{x})$ over the same random variable $\mathbf{x}$, we can measure how different these distributions are using the **Kullback-Leibler divergence:**

$$D_{\mathrm{KL}}(P \parallel Q) = E_{\mathbf{x}\sim P}\left[\log \frac{P(x)}{Q(x)}\right] = E_{\mathbf{x}\sim P}[\log P(x) - \log Q(x)].$$

In the case of discrete variables, it is the **extra amount of information** needed to send a message containing symbols drawn from probability distribution P, when we use a code that was designed to minimize the length of messages drawn from probability distribution Q.

Properties of the KL divergence

- the KL divergence is alwasy non-negative,
- the KL divergence is 0 $\iff$ P and Q are the same distribution (or are equal *almost anywhere* in the case of continuous variables),
- the KL **is not** a distance: a distance should also be symmetric and satisfy the triangle inequality, but the KL divergence does not.

The fact that the KL divergence is not symmetric has important consequences when one needs to minimize the *distance* between two distributions P and Q.

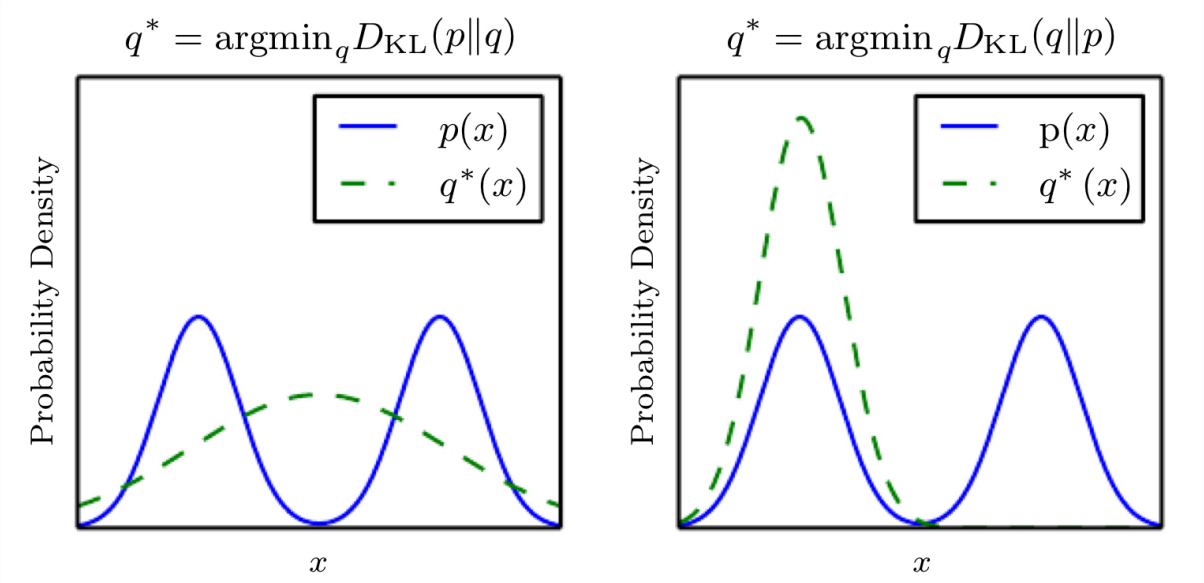
About minimizing the KL divergence

Minimizing $D_{\mathrm{KL}}(p \parallel q)$ can be very different than minimizing $D_{\mathrm{KL}}(q \parallel p)$

$$D_{\mathrm{KL}}(p \parallel q) = E_{\mathrm{x}\sim p}\left[\log \frac{p(x)}{q(x)}\right].$$

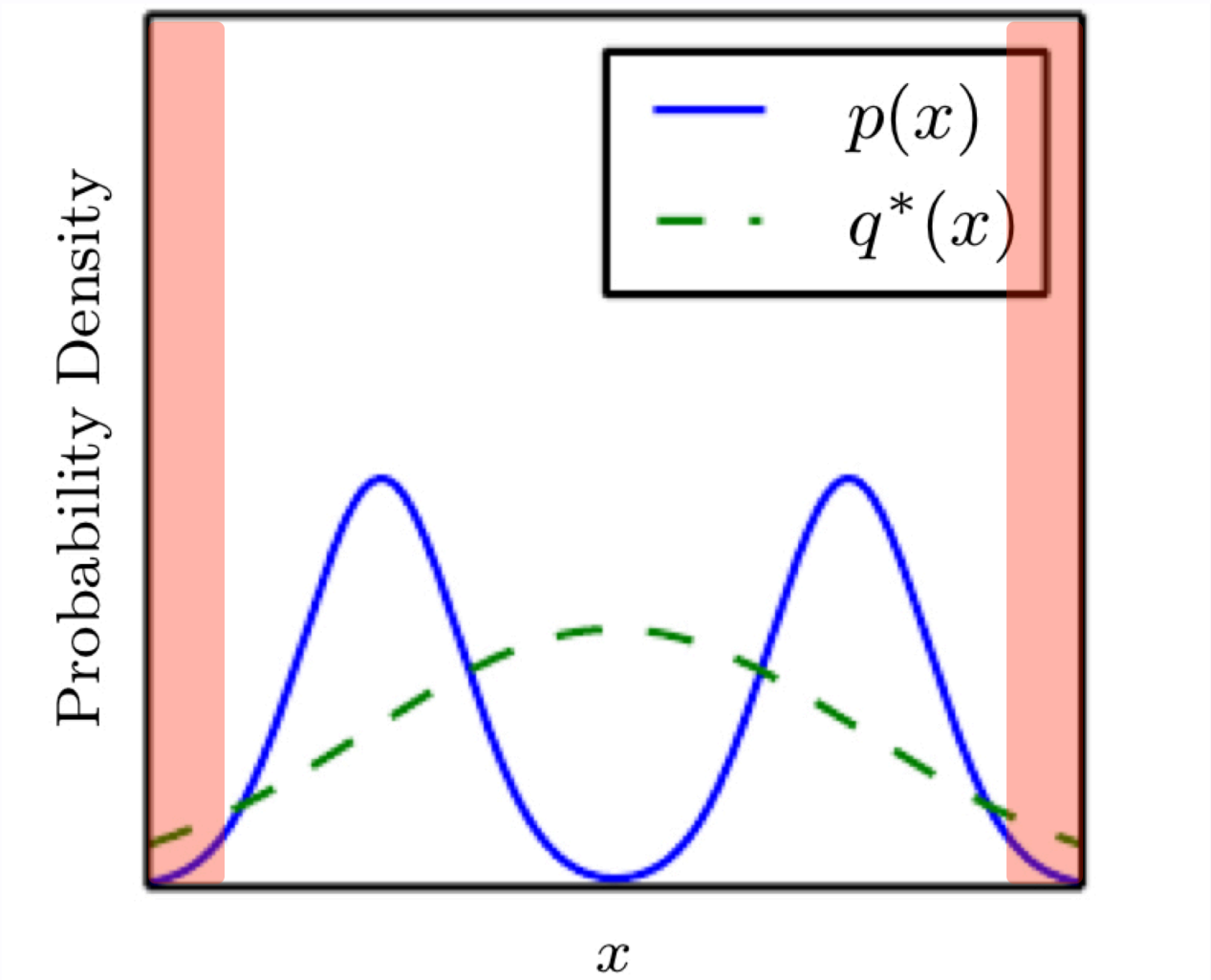
Assume:

- p is a mixture of two gaussians with two distinct modes;
- q is a single gaussian that we want to *optimize* so that it matches p as well as possible.

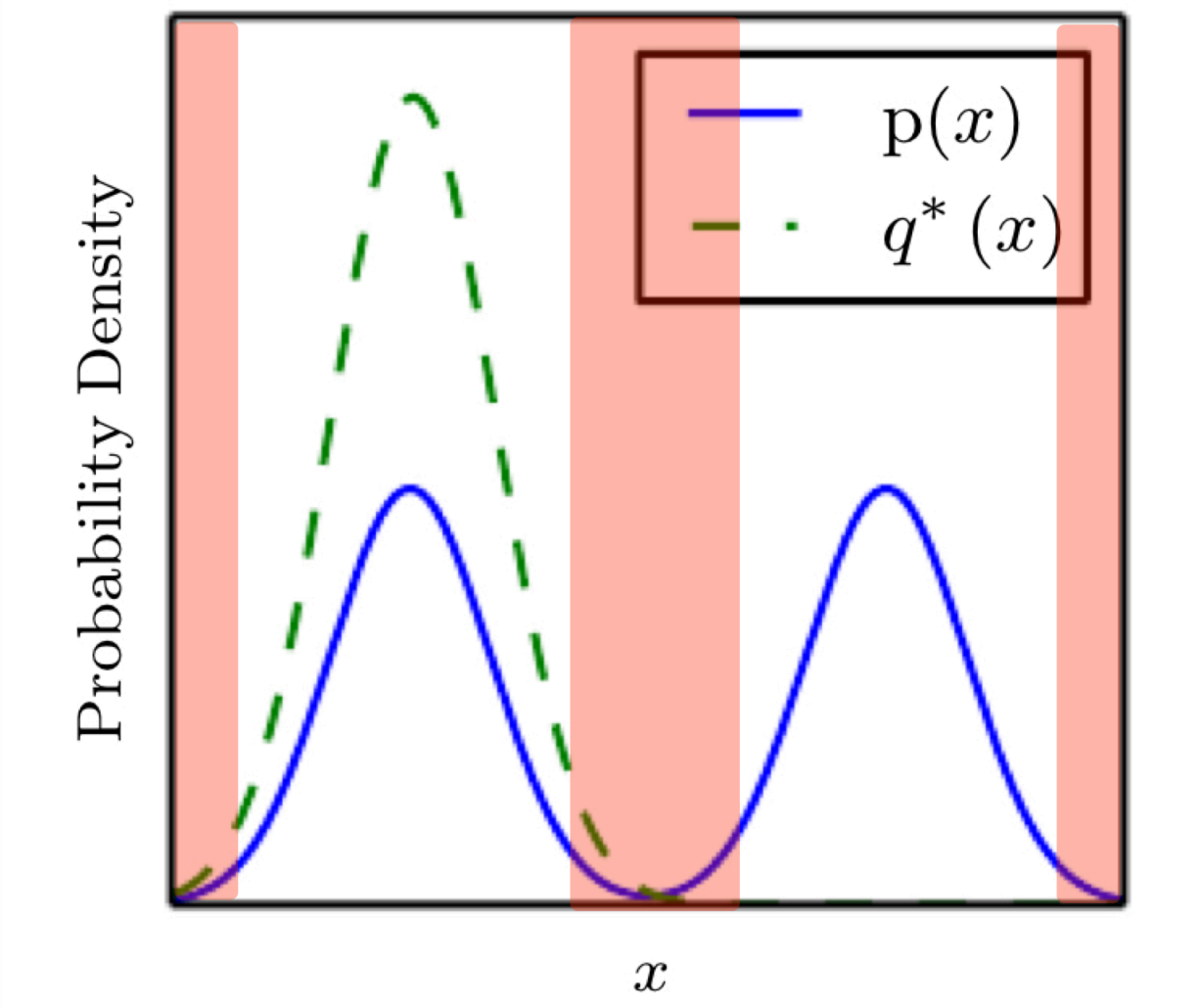


$$D_{\mathrm{KL}}(q \parallel p) = E_{\mathrm{x}\sim q}\left[\log \frac{q(x)}{p(x)}\right].$$

$$\min_q \left[D_{\mathrm{KL}}(p \parallel q) \right] = \min_q \left[\mathbb{E}_{\mathbf{x} \sim p} \left[\log \frac{p(x)}{q(x)} \right] \right]$$



$$\min_q \left[D_{\mathrm{KL}}(q \parallel p) \right] = \min_q \left[\mathbb{E}_{\mathrm{x} \sim q} \left[\log \frac{q(x)}{p(x)} \right] \right]$$



Cross Entropy

A quantity that is closely related to the KL divergence is the **cross-entropy**

$$H(P, Q) = H(P) + D_{\mathrm{KL}}(P \parallel Q) = -E_{\mathrm{x}\sim P}[\log Q(x)]$$

i.e., the cross entropy is the average number of bits needed to encode messages for code Q with a code designed for P.

Notes:

- similar to $D_{KL}(P \parallel Q) = E_{\mathrm{x}\sim P}[\log P(x) - \log Q(x)]$:
  - similar expression, but it lacks the term $\log P(x)$;
  - conceptual difference is that D_{KL} measures the expected **extra** number of bits, H measures the total number of bits.
- minimizing $H(P, Q)$ w.r.t. Q is the same as minimizing $D_{KL}(P \| Q)$ (why?)

Graphical Models

a.k.a. Structured Probabilistic Models

Often probability distributions can be split into many factors. For instance, assume that a random variable $\mathbf{a}$ influences the value of another variable $\mathbf{b}$ and that in turn $\mathbf{b}$ influences $\mathbf{c}$, but that they are otherwise independent. We can represent the whole distribution as:

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

These factorizations can greatly reduce the number of parameters needed to describe the distribution.

Example

Assume all $\mathrm{a}, \mathrm{b}, \mathrm{c}$ assume values in $\{1, 2, 3, 4, 5\}$. To completely describe the probabilities involved we would need to specify $5^3 = 125$ probability values.

If the distribution factorizes as above we only require: 5 parameters to describe $p(a)$, 25 parameters to describe $p(b|a)$ and 25 parameters to specify $p(c|b)$.

Graphical Models

Factorizations over distributions can be visually described using graphs.

Directed models

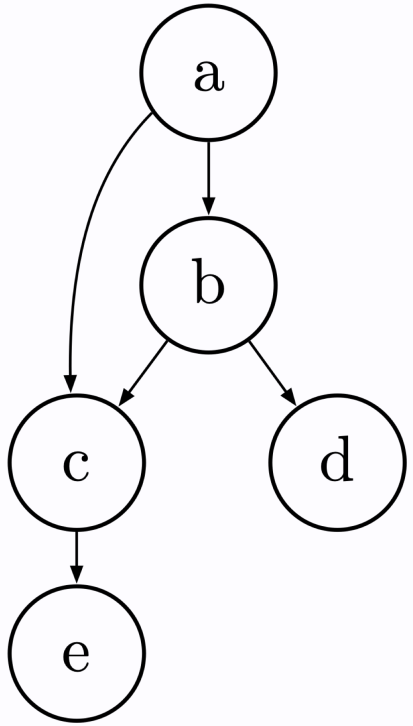
Use graphs with directed edges. They represent factorizations into conditional probabilities distributions.

- one factor for every random variable $\mathbf{x}_i$
- the factor consists of the conditional distribution of $\mathbf{x}_i$ given its parents.

$$p(\mathbf{x}) = \prod_i p(\mathbf{x}_i | Pa_{\mathcal{G}}(\mathbf{x}_i))$$

where $Pa_{\mathcal{G}}(\mathbf{x}_i)$ is the set of parents of $\mathbf{x}_i$.

Example



$$p(\mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{d}, \mathrm{e}) = p(\mathrm{a})p(\mathrm{b}|\mathrm{a})p(\mathrm{c}|\mathrm{a}, \mathrm{b})p(\mathrm{d}|\mathrm{b})p(\mathrm{e}|\mathrm{c})$$

Undirected Models

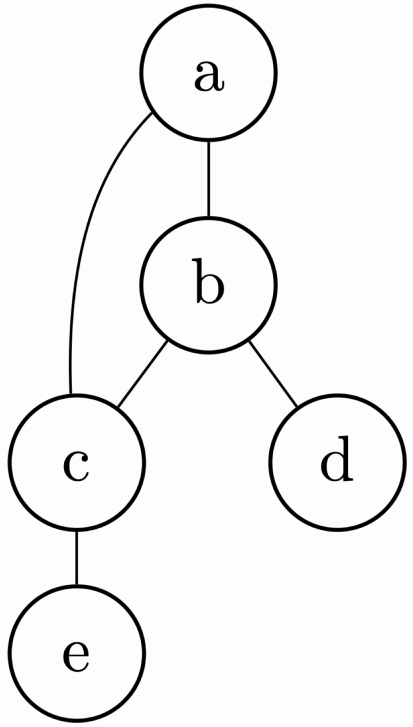
Use graphs with undirected edges. They represent factorizations using a set of functions (not necessary, nor common that they are probabilistic distributions).

- one factor $\phi^{(i)}$ per clique $\mathcal{C}^{(i)}$ in the graph;
- the factorization is the normalized product of all factors.

$$p(\mathbf{x}) = \frac{1}{Z} \prod_i \phi^{(i)} \left(\mathcal{C}^{(i)} \right)$$

with $Z = \sum_{\boldsymbol{x} \in \mathbf{x}} \prod_i \phi^{(i)}(\mathcal{C}^{(i)})$

Example



$$p(\mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{d}, \mathrm{e}) = \frac{1}{Z} \phi^{(1)}(\mathrm{a}, \mathrm{b}, \mathrm{c}) \phi^{(2)}(\mathrm{b}, \mathrm{d}) \phi^{(3)}(\mathrm{c}, \mathrm{e})$$

Information theory knowledge poll