Detailed Balance

When transitioning $\mu \to \nu$ to ensure we're sampling from the correct $P(\vec{x})$

$$P(\mu)T(\mu \to \nu) = P(\nu)T(\nu \to \mu)$$

or $$\frac{T(\mu \to \nu)}{T(\nu \to \mu)} = \frac{P(\nu)}{P(\mu)}$$

e.g.) for Boltzmann dist:

$$\frac{T(\mu \to \nu)}{T(\nu \to \mu)} = \frac{Z}{Z}\,\frac{e^{-\beta E_\nu}}{e^{-\beta E_\mu}} = e^{-\beta(E_\nu - E_\mu)}$$

Difficult part: designing the <u>update</u> itself

Design trick: split the transition probability

$$T(\mu \to \nu) = g(\mu \to \nu)A(\mu \to \nu)$$

selection probability ↑ ↑ acceptance ratio

e.g.) Ising model single-spin flip

propose an update $x_i \to -x_i$ (if ± 1)

then $g(\mu \to \nu) = \frac{1}{N}$ for any μ, ν

$$\frac{T(\mu \to \nu)}{T(\nu \to \mu)} = \frac{1/N}{1/N} \frac{A(\mu \to \nu)}{A(\nu \to \mu)} = \frac{P(\nu)}{P(\mu)}$$

ie. $$\frac{A(\mu \to \nu)}{A(\nu \to \mu)} = e^{-\beta(E_\nu - E_\mu)}$$

the "Metropolis" algorith chooses

$$A(\nu \to \mu) = 1 \quad \text{if } E_\mu < E_\nu$$

then $$A(\mu \to \nu) = e^{-\beta(E_\nu - E_\mu)}$$

when the energy is raised.

<u>Generalized version</u>

$$A(\mu \to \nu) = \min\left\{1, \frac{P(\nu)}{P(\mu)} \cdot \frac{g(\nu \to \mu)}{g(\mu \to \nu)}\right\}$$

Let's derive another algorithm used in RBMs called "Gibbs" sampling.

Some definitions:

$P(A|B)$ "prob. of A given B"

$P(A) = \sum_B p(A,B)$ "sum rule" marginal prob.

$P(A,B)$ "joint distribution"

$P(A,B) = P(B|A)\,p(A)$ "product rule"

let's use these to suggest an update that replaces x_i $(\vec{x}^\mu = (\underbrace{x_1, x_2 \cdots}_{B} \underbrace{x_i}_{A} \underbrace{\cdots x_N}_{B}))$ with a value drawn from $P(x_i | \vec{x}_{-i})$

x_i is omitted

ie.
$$g(\mu \to \nu) = P(x_i^\nu | \vec{x}_{-i}^\mu)$$
$$= P(x_i^\nu | \vec{x}_{-i}^\nu)$$

let's use
$$P(\nu) = P(\vec{x}^\nu)$$
$$= P(x_i^\nu | \vec{x}_{-i}^\nu)\,P(\vec{x}_{-i}^\nu)$$

$$\frac{A(\mu\to\nu)}{A(\nu\to\mu)} = \frac{P(\nu)}{P(\mu)}\frac{g(\mu\to\nu)}{g(\mu\to\nu)}$$

$$= \frac{P(\nu)}{P(\mu)}\frac{P(x_i^{\mu}|\vec{x}_{-i}^{\mu})}{P(x_i^{\nu}|\vec{x}_{-i}^{\nu})}$$

$$= \frac{P(x_i^{\nu}|\vec{x}_{-i}^{\nu})\,P(\vec{x}_{-i}^{\nu})\,P(x_i^{\mu}|\vec{x}_{-i}^{\mu})}{P(x_i^{\mu}|\vec{x}_{-i}^{\mu})\,P(\vec{x}_{-i}^{\mu})\,P(x_i^{\nu}|\vec{x}_{-i}^{\nu})}$$

$= 1$ since $\vec{x}_{-i}^{\mu} = \vec{x}_{-i}^{\nu}$

Therefore in Gibbs sampling the acceptance ratio is always 1.

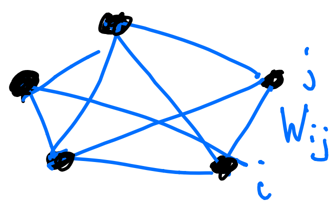
Recall: our goal is to take $\mathcal{D} = \{\vec{x}\}$ and to find (an approximate) $p_{\lambda}(\vec{x}) \simeq P(\vec{x})$

This indicates a parametrization

→ goal tune λ using $\mathcal{D}$.

The Hopfield Network (1982)

Prob. graphical model / Hopfield neural network is defined with



N nodes for n variables

$$E(\vec{x}) = -\sum_{ij} W_{ij} x_i x_j - \sum_i b_i x_i$$

using

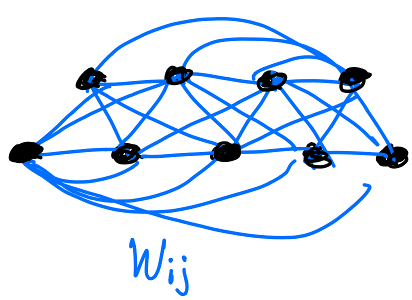
$$P_\lambda(\vec{x}) = \frac{1}{Z} e^{-E(\vec{x})}$$

"learn" $\lambda = (W, b)$ so that $P_\lambda(\vec{x}) \approx P(\vec{x})$

Boltzmann Machine (Ackley, Hinton, Sejnowski 1985)

Similar to Hopfield with an additional latent space

or "hidden units"



$\vec{h}$ hidden

$\vec{v}$ visible

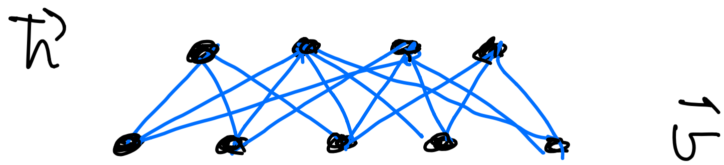
Here we have a graphical prob. dist.

$P_\lambda(\vec{v}, \vec{h})$ ie. a joint distribution

"learning" is adjusting parameters λ so that

$$P_\lambda(\vec{v}) = \sum_{\vec{h}} P_\lambda(\vec{v}, \vec{h}) \simeq P(\vec{v})$$

<u>Restricted Boltzmann Machine</u> (Hinton, Smolensky '86)



no intra-layer couplings

$$E_\lambda = -\sum_{ij} W_{ij} v_i h_j - \sum_i b_i v_i - \sum_j c_j h_j$$

parameters are $\lambda = (W, \vec{b}, \vec{c})$

and $$P_\lambda(\vec{v}, \vec{h}) = \frac{1}{Z_\lambda} e^{-E_\lambda(\vec{v}, \vec{h})}$$