Advanced Machine Learning

Bert Kappen
SNN Radboud University, Nijmegen
Gatsby Unit, UCL London

February 4, 2025

Course setup

- 6 ec course
- examination based on computer exercises and presentation of research topic
- Students work in groups of three and hand in a single result
- weekly exercises tutorials
- All course materials (slides, exercises) and schedule via `http://www.snn.ru.nl/~bertk/machinelearning/adv_ml.html`

Lecture 1. Monte Carlo sampling

Based on MacKay chapter 29 and 30.

- Uniform sampling
- Importance and Rejection sampling
- Metropolis-Hasting method
- Detour on Markov processes, stationary distribution, ergodicity
- Gibbs sampling
- Hybrid Monte Carlo

Machine learning is about inference

In many machine learning problems one needs to estimate high dimensional integrals.

1. Estimate expected parameter values in the posterior of a Bayesian learning problem:

$$p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)} \qquad \mathbb{E}\,\theta = \int d\theta \theta p(\theta|D)$$

2. Estimate the 'evidence' for model selection

$$p(D|\mathcal{H}_i) = \int d\theta p(D|\theta, \mathcal{H}_i) p(\theta|\mathcal{H}_i)$$

3. Computing statistics in graphical models $p(x_1, \ldots, x_n) = \prod_{i=1}^{n} p_i(x_i|x_{\mathsf{Pa}(i)})$

$$p(x_1, x_2) = \sum_{x_3, \ldots, x_n} p(x_1, \ldots, x_n)$$

Denote $p(x) = \frac{p^*(x)}{Z}$ the probability distribution of interest with $Z = \int dx p^*(x)$. We can assume that $p^*(x)$ is easily evaluated for any x, but Z is hard to evaluate.

Problems 1,3 are of the form: Estimate

$$\Phi = \int dx p(x) \phi(x)$$

with $\phi(x)$ some function of x. Problem 2 is of the form estimate $Z = \int dx p^*(x)$.

All these problems are hard, but problem 2 (computing the partition sum) is actually harder than problems 1,3 (computing statistics) [1]

We will consider estimating Φ.

[1] For problem 2, more advanced methods, such as thermodynamic integration are needed.

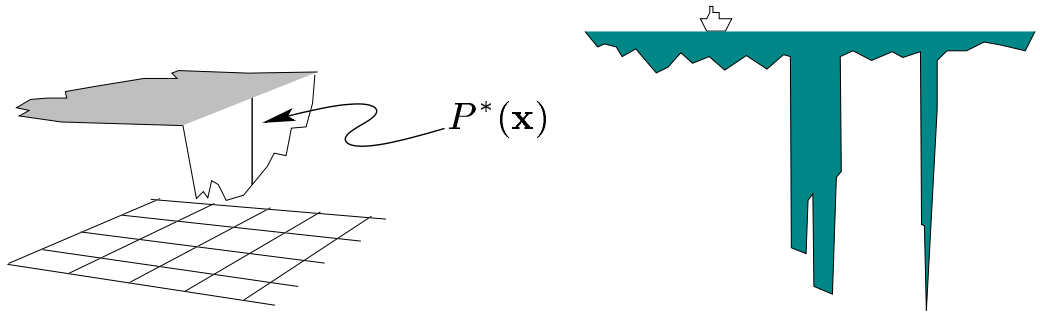
An analogy

Imagine a lake with coordinates $x = (x_1, x_2)$. You wish to estimate the average plankton concentration [2]

$$\Phi = \frac{1}{Z} \sum_x p^*(x)\phi(x) \qquad Z = \sum_x p^*(x)$$

$p^*(x)$ is the depth of the lake at x. $\phi(x)$ is the plankton concentration at x (independent of the depth). So $p^*(x)\phi(x)$ is the total amount of plankton at location x.

Z is the total volume of the lake.



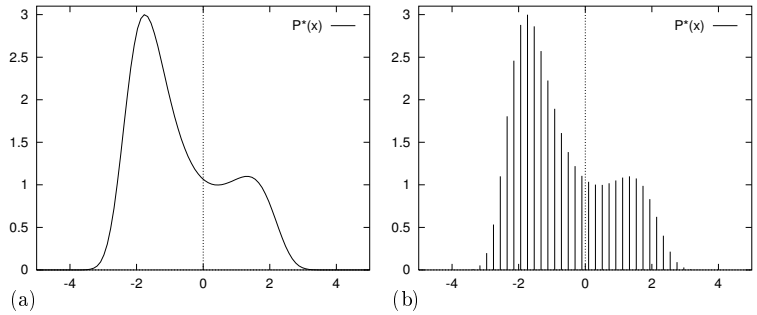
[2] We assume for the simplicity of the argument that the lake has discrete locations labeled by x.

An analogy

Go out with your boat to locations x, measure the depth with a plumbline and the local plankton concentration. If you sample all locations in the lake,

$$\hat{\Phi} = \frac{\sum_x p^*(x)\phi(x)}{\sum_x p^*(x)} = \Phi$$

you obtain the exact result.



But this is very expensive. If the lake has a locations per dimension then this requires a^2 measurements. For an n dimensional problem, the number becomes a^n exponential in n. This is called the curse of dimensionality.

An analogy

Suppose instead of visiting all locations, we visit N locations. A very good (actually the best) way to choose the locations is to choose them with a probability proportional to the depth $p(x) = p^*(x)/Z$. Of course, this is impractical, because you need to measure first the depth at all lake locations before you can start rolling your dice!

But suppose you can, then,

$$\hat{\Phi} = \frac{1}{N} \sum_{r=1}^{N} \phi(x^r) \qquad x^r \sim p(x)$$

$\hat{\Phi}$ is a random variable: each time we repeat the experiment (collecting N samples) we get a different outcome. But it is a good estimator of Φ because the average value over many such experiments is $\mathbb{E}\hat{\Phi} = \Phi$. The estimator $\hat{\Phi}$ is called unbiased.

The proof is simple. SInce the samples are drawn independent, the probability to obtain outcomes $x^1, \ldots, x^N$ is $p(x^1, \ldots, x^N) = \prod_{r=1}^{N} p(x^r)$. The expected value of $\hat{\Phi}$

is

$$\mathbb{E}\hat{\Phi} = \frac{1}{N}\sum_{r=1}^{N}\sum_{x^r} p(x^r)\phi(x^r) = \Phi$$

The variance in $\hat{\Phi}$ decreases with the number of samples N:[3]

$$\mathbb{V}\hat{\Phi} = N\mathbb{V}\frac{\phi}{N} = \frac{1}{N}\mathbb{V}\phi \qquad \mathbb{V}\phi = \sum_x p(x)(\phi(x) - \Phi)^2$$

Thus,

$$\hat{\Phi} = \Phi + O\left(\frac{1}{\sqrt{N}}\right)$$

Note, that the accuracy does not depend on the dimension of the problem. So this is great. But sampling from p is very hard!

[3] Here we make use of the useful property that if $X = \sum_i X_i$ is a sum of independent random variables, the variance in X is the sum of the variances of each X_i.

Typical set

Consider the Ising model

$$p(x) = \frac{\exp(\beta \sum_{i>j} x_i x_j w_{ij})}{Z} = \frac{p^*(x)}{Z} \qquad Z = \sum_x p^*(x)$$

with $x = (x_1, \ldots, x_n)$ and $x_i = \pm 1, i = 1, \ldots, n$.

Total number of states is 2^n, but most probability is concentrated in the so-called typical set T.

Typical set

Draw long sequence of N states from the Ising model:

$$x^1, x^2, \ldots, x^N$$

The probability of the sequence is

$$P(x^1, x^2 \ldots, x^N) = P(x^1)P(x^2) \ldots P(x^N) = \prod_x p(x)^{N(x)}$$

The number of times the value x occurs in the string is $N(x) \approx p(x)N$.

Thus,

$$P(x^1, x^2 \ldots, x^N) \approx \prod_x p(x)^{p(x)N} = \left(2^{-H}\right)^N$$

with $H = -\sum_x p_x \log p_x$ the entropy of the distribution.

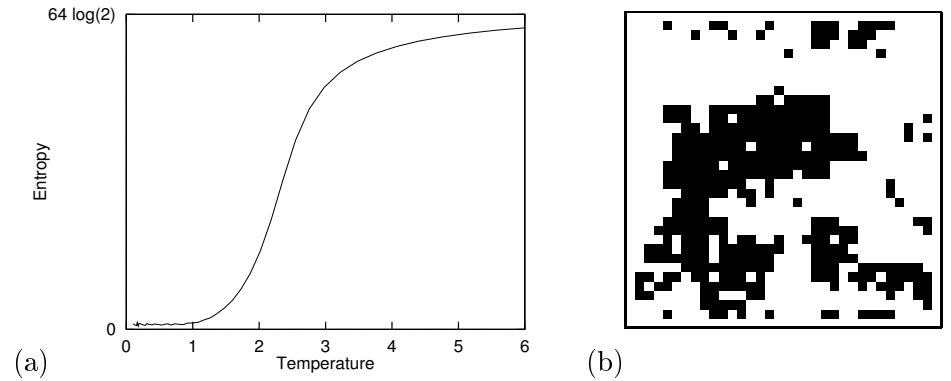
Thus, in the large N limit all typical strings have the same probability 2^{-NH} and all other (non-typical) strings have probability zero.

The formula suggests that 'on average' for a single sample, there are typical samples with probability 2^{-H} and non-typical samples with probability zero. Denote T the set of typical samples. The number of typical samples is thus $|T| \approx 2^H$.

The typical set should be compared to the total number of states 2^n and the volume fraction is 2^{H-n}. If we sample states x uniformly at random, the probability to hit an element of the typical set is 2^{H-n}. Thus, one needs on the order of $N_{\text{min}} = 2^{n-H}$ samples to hit the typical set once and therefore the number of samples

$$N \gg N_{\text{min}} = 2^{n-H}$$

For n binary spins $0 \leq H \leq n$.



Left. Entropy of a 64 spin 2d Ising model with couplings $w_{ij} = 1$. Right. Sample from a 1024 spin Ising model near the critical temperature.

- For (very) high temperature (low β) $H \approx n$ and $N_{\text{min}} \approx 1$ and uniform sampling feasible

- Around the critical temperature, $H \approx n/2$ and $N_{\text{min}} \approx 2^{n/2}$. For $n = 1000$ spins this is of order 10^{150} which is about the square of the number of particles in the universe.

In a nut shell: Uniform sampling only works for uniform distributions (in which case it is optimal!).

Sampling from a multi-variate Gaussian

How to sample from a multi-variate Gaussian distribution $\mathcal{N}(x|m, S)$?

Define C such that $CC^T = S$ and define $y = C^{-1}(x - m)$. Since this is a linear transformation, y is also Gausian distributed with

$$\begin{aligned} \mathbb{E}y &= C^{-1}(\langle x \rangle - m) = 0 \\ \mathbb{V}(y) &= \mathbb{E}yy^T = C^{-1}\mathbb{E}(x-m)(x-m)^T \left(C^{-1}\right)^T = C^{-1}S\left(C^{-1}\right)^T = 1 \end{aligned}$$

Thus sample from y can be obtained by sampling each component independently:

$$p(y) = \prod_i \mathcal{N}(y_i|0, 1)$$

Sampling from a one dimensional Gaussian can be done by sampling from a uniform distribution using the Box-Muller transformation (Exercise).

Sample from x is obtained by sampling from y and $x = Cy + m$.

Importance sampling

The task is to estimate

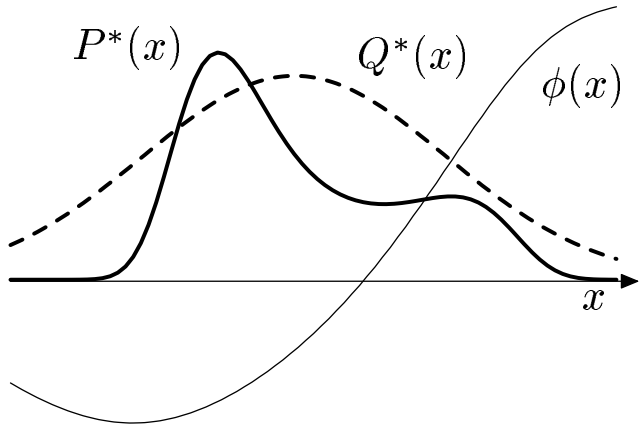
$$\Phi = \int dx p(x)\phi(x)$$

Sample from another distribution $q(x)$.

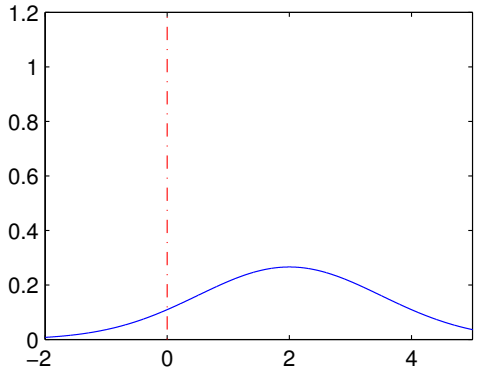
Often one can propose a sample density that is
1) better than uniform and
2) easy to sample from.

For instance, a (spherical) Gaussian:

$$Q^*(x) \propto \exp(-\sum_i x_i^2/2)$$



Importance sampling



Consider simple 1-d sampling problem. Given $p(x)$, compute

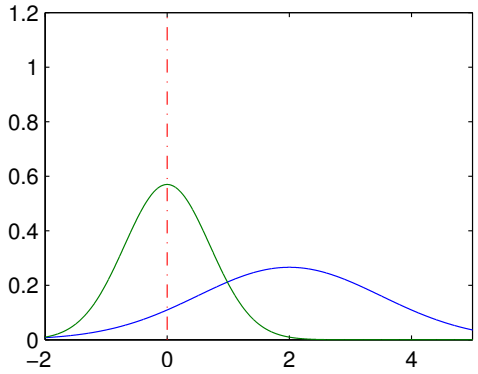
$$\Phi = \text{Prob}(x < 0) = \int_{-\infty}^{\infty} \phi(x) p(x) dx$$

with $\phi(x) = 1$ if $x \leq 0$ and zero otherwise.

Naive method: generate N samples $X_i \sim p$

$$\hat{\Phi} = \frac{1}{N} \sum_{i=1}^{N} \phi(X_i)$$

Importance sampling



Consider another distribution $q(x)$. Then

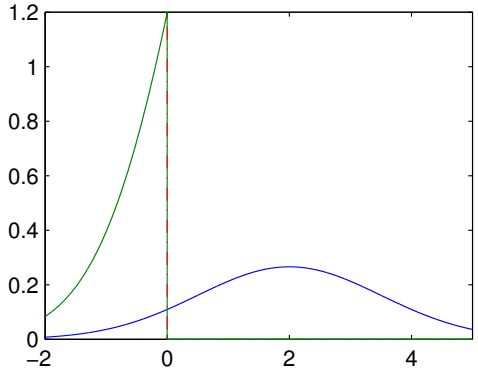
$$\Phi = \text{Prob}(x < 0) = \int_{-\infty}^{\infty} \phi(x) \frac{p(x)}{q(x)} q(x) dx$$

Importance sampling: generate N samples $X_i \sim q$

$$\hat{\Phi} = \frac{1}{N} \sum_{i=1}^{N} \phi(X_i) \frac{p(X_i)}{q(X_i)}$$

Unbiased ($\mathbb{E}\hat{\Phi} = \Phi$) for any q!

Optimal importance sampling



The distribution

$$q^*(x) = \frac{p(x)\phi(x)}{\Phi}$$

is the optimal importance sampler.

One sample $X \sim q^*$ is sufficient to estimate a:

$$\hat{\Phi} = \phi(X)\frac{p(X)}{q^*(X)} = \Phi$$

The estimator has zero variance

Normalization

So any importance sampler is unbiased $\mathbb{E}\hat{\Phi} = \Phi$, and the closer that q is to p the smaller the variance.

The importance weights $p(x)/q(x)$ assume that we can evaluate $p(x)$ (q we choose so that we can). However, most often $p(x)$ can only be easily computed, up to a constant:

$$p(x) = \frac{p^*(x)}{Z} \qquad Z = \int dx p^*(x)$$

Estimate both numerator and denonimator by sampling.

$$\Phi = \int dx p(x)\phi(x) = \frac{\int dx p^*(x)\phi(x)}{\int dx p^*(x)} = \frac{\int dx q(x)\frac{p^*(x)}{q(x)}\phi(x)}{\int dx q(x)\frac{p^*(x)}{q(x)}}$$

Sample $\{x^r\}$ from $q(x)$ and compute

$$w_r = \frac{p^*(x^r)}{q(x^r)} \qquad \hat{\Phi} = \frac{\sum_r w_r \phi(x^r)}{\sum_r w_r}$$

Importance sampling

The estimate is biased.

$$\mathbb{E}\hat{\Phi} = \mathbb{E}\left(\frac{\sum_{r=1}^{N} w_r \phi(x^r)}{\sum_{r=1}^{N} w_r}\right) \neq \frac{\mathbb{E}\sum_{r=1}^{N} w_r \phi(x^r)}{\mathbb{E}\sum_{r=1}^{N} w_r} = \frac{N\int dx p^*(x)\phi(x)}{N\int dx p^*(x)} = \Phi$$

However, for large N:

$$\sum_{r=1}^{N} w_r = N\mathbb{E}w_r + O\left(\sqrt{N}\right) \approx N\mathbb{E}w_r$$

$$\mathbb{E}\hat{\Phi} \approx \mathbb{E}\left(\frac{\sum_{r=1}^{N} w_r \phi(x^r)}{N\mathbb{E}w_r}\right) = \frac{\mathbb{E}\sum_{r=1}^{N} w_r \phi(x^r)}{N\int dx p^*(x)} = \frac{N\int dx p^*(x)\phi(x)}{N\int dx p^*(x)} = \Phi$$

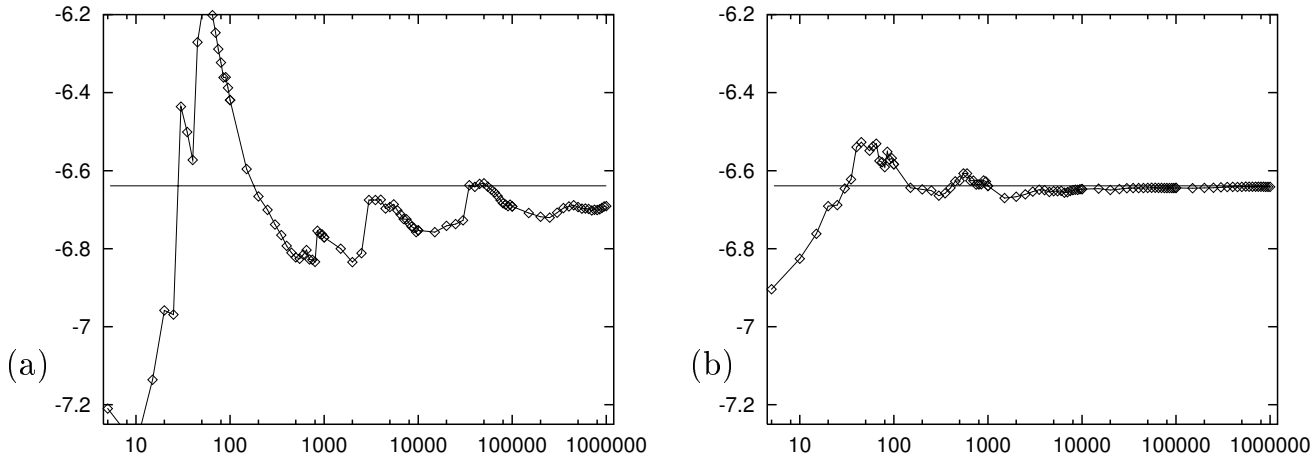
The estimator $\hat{\Phi}$ is asymptotically unbiased.

Choose sufficiently broad importance sampler

Warning: rare events give large contributions. Compare

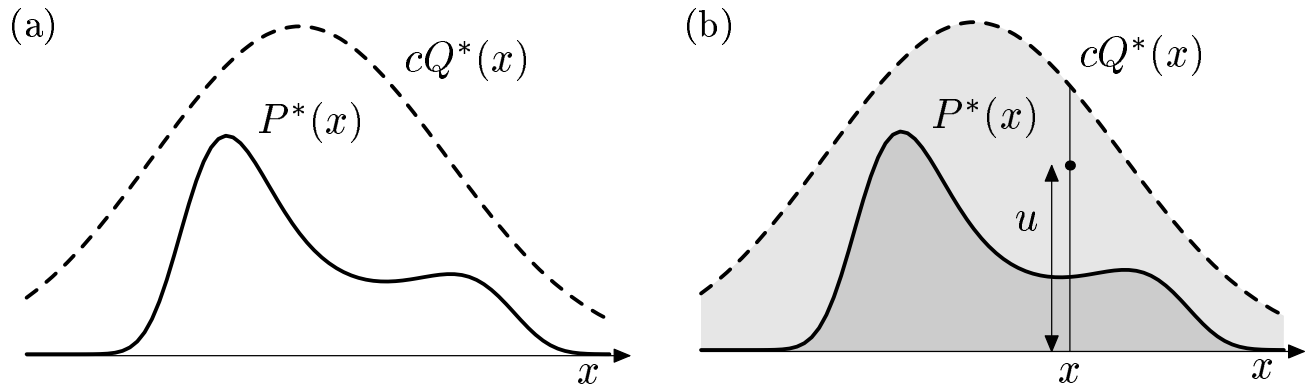
$$q(x) \propto e^{-x^2/2} \qquad q(x) \propto \frac{1}{x^2 + a^2}$$

The weights are $\propto q(x)^{-1}$: Rare events have large weights.



Toy 1-d amino acid sampling problem, showing effect of poor importance sampler. Left: Gaussian suggests convergence after 500 iterations. Right: Cauchy is broader distribution and shows more robust behaviour.

Rejection sampling



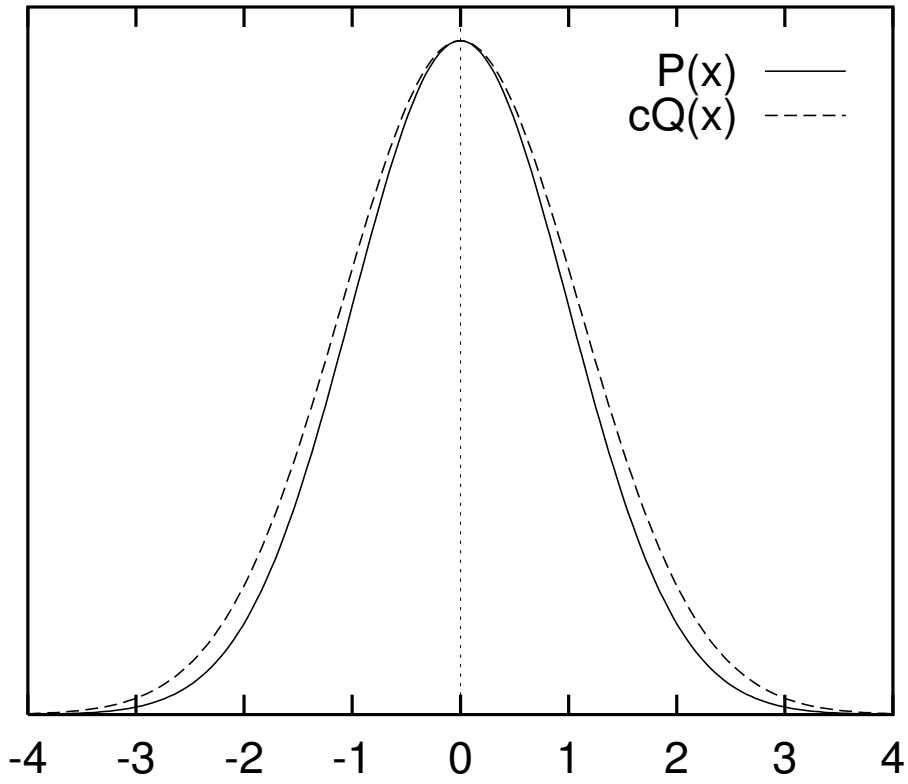
Choose, c such that for all x : $cQ^*(x) > P^*(x)$

- generate x from $Q^*(x)$
- generate u uniform from $[0, cQ^*(x)]$
- if $u > P^*(x)$ reject x, otherwise accept x

Probability of a sample x is $Q^*(x)\frac{P^*(x)}{cQ^*(x)} \propto P^*(x)$.

$$\hat{\Phi} = \sum_r \phi(x^r) \rightarrow \int dx \phi(x) p(x)$$

Rejection sampling in high dimensions



Let $p(x)$ and $q(x)$ be spherical Gaussians in n dimensions with mean 0 and $\sigma_q = 1.01\sigma_p$.

Since

$$q(x) = \left(\frac{1}{\sqrt{2\pi\sigma_q^2}}\right)^n e^{-\frac{1}{2\sigma_q^2}\sum_i x_i^2} \qquad p(x) = \left(\frac{1}{\sqrt{2\pi\sigma_p^2}}\right)^n e^{-\frac{1}{2\sigma_p^2}\sum_i x_i^2}$$

then

$$c = \frac{p(0)}{q(0)} = \left(\frac{\sigma_q}{\sigma_p}\right)^n = 1.01^n$$

With $n = 1000$ we find c=20.000. Thus volume under cq is 20.000 times the volume under p. Therefore, the acceptance rate = $\frac{\text{volume } p}{\text{volume } cq} = \frac{1}{c}$

Thus rejection sampling is inefficient in high dimensions.

Curse of dimensionality for importance sampling

A similar argument holds for importance sampling. Suppose that we wish to estimate $\Phi = \int dx p(x)\phi(x)$ and we use importance sampling with $q(x)$. The importance weights are $w(x_r) = \frac{p(x_r)}{q(x_r)}$ and

$$\hat{\Phi} = \frac{1}{N}\sum_{r=1}^{N} w(x_r)\phi(x_r) \qquad x_r \sim q(\cdot)$$

Since w_r is computed with $p(x)$ rather than $p^*(x)$, $\hat{\Phi}$ is an unbiased estimator: $\left\langle \hat{\Phi} \right\rangle = \Phi$.

Importance sampling breaks down in high dimension, because of the large variance of the sample weights $w(x) = \frac{p(x)}{q(x)}$. The way to understand this, is that when $w(x)$ is large for certain x, $q(x)$ is small. This means that $\hat{\Phi}$ depends on a large contribution $w(x)\phi(x)$, but that this contribution has low probability of occuring in the sample. This makes $\hat{\Phi}$ inaccurate. Another way to understand that the quality of the sampling deteriorates with increasing variance of $w(x)$ is to note that when $p = q$, $w(x) = 1$ and its variance is zero. The variance in w is proportional to the difference between the distributions q and p.

We compute the mean and variance of $w(x)$ for the Gaussian distributions considered for the rejection sampling example with $\sigma_p = 1$ and variable σ_q:

$$\begin{aligned}
\langle w \rangle &= \int d^n x q(x) \frac{p(x)}{q(x)} = 1 \\
\sigma_w^2 &= \int d^n x q(x) \left(\frac{p(x)}{q(x)} - 1 \right)^2 = \int d^n x \frac{p(x)^2}{q(x)} - 1 \\
&= \left(\frac{\sigma_q}{\sqrt{2\pi}} \right)^n \int d^n x e^{-\frac{1}{2}\left(2 - \frac{1}{\sigma_q^2}\right) \sum_i x_i^2} - 1 \\
&= (\sigma_q a)^n - 1
\end{aligned}$$

with $a = \frac{\sigma_q}{\sqrt{2\sigma_q^2 - 1}}$. So with $\sigma_q^2 = 1 + \epsilon$ the variance in the sampling weights is

$$\sigma_w^2 = \left(\frac{\sigma_q^2}{\sqrt{2\sigma_q^2 - 1}} \right)^n - 1 = \left(\frac{1+\epsilon}{\sqrt{1+2\epsilon}} \right)^n - 1 \approx \left(1 + \frac{1}{2}\epsilon^2 \right)^n - 1$$

For the previous example with $\epsilon = 0.02$ and $n = 1000$ we obtain $\sigma_w^2 = 0.224$ which

is not bad.

But when p and q differ more, the variance can be significant and importance sampling breaks down. For instance with $\epsilon = 0.1$ and $n = 2000$ we obtain $\sigma_w^2 \approx 21000$.

Note, that we have here treated to favorable case when the importance weights are computed from $p(x)$ rather than from $p^*(x)$ so that $\hat{\Phi}$ is unbiased. In practice, one is often forced to use $p^*(x)$ so that the importance sampling error is further increased due to (unknown) bias.

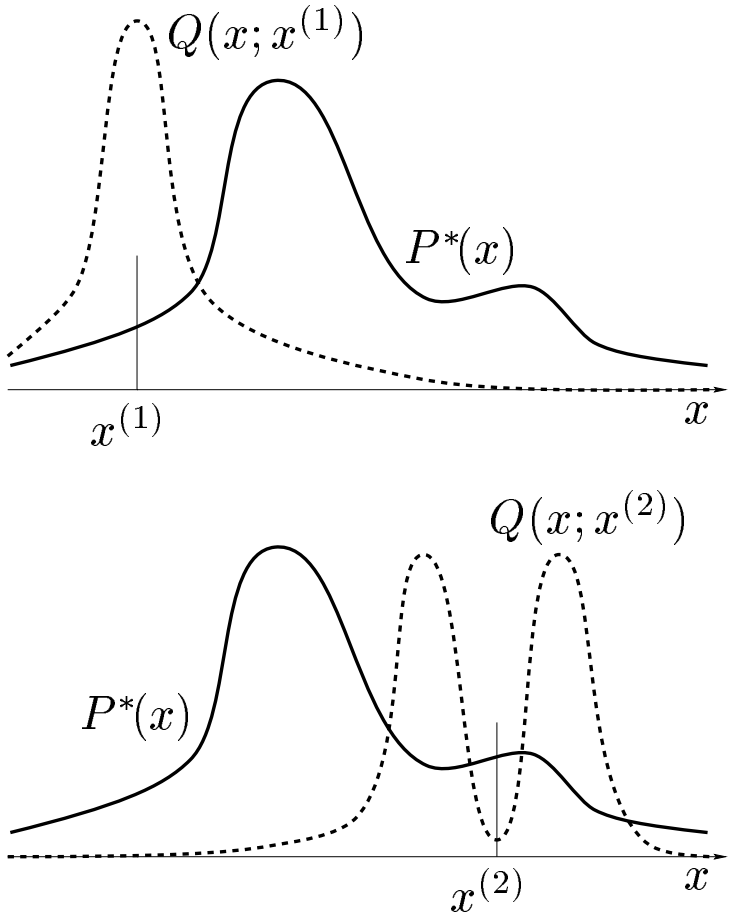
Lecture 2. Monte Carlo sampling

Based on MacKay chapter 29 and 30.

- Uniform sampling
- Importance and Rejection sampling
- Metropolis-Hasting method
- Detour on Markov processes, stationary distribution, ergodicity
- Gibbs sampling
- Hybrid Monte Carlo
- Sampling the Bayesian posterior for the perceptron

Metropolis algorithm

The Metropolis algorithm (1956) considers a sampling density which depends on the current sample value: $q(x|x^r)$.



Initialize in some random state x^1

Metropolis algorithm

At iteration r, sample x' from $q(x'|x^r)$ and compute

$$a = \frac{p^*(x')q(x^r|x')}{p^*(x^r)q(x'|x^r)}$$

If $a \geq 1$, accept x' as the new state: $x^{r+1} = x'$

Else, accept x' as the new state with probability a

If accept: $x^{r+1} = x'$, else $x^{r+1} = x^r$

Markov processes

Let $\mathcal{S}$ denote the set of all state vectors x. $x \in \mathcal{S}$ is a binary vector of length n and thus x can take on 2^n different values.

Denote $p(x)$ a probability distribution over states. $p(x)$ is a vector of length 2^n with $\sum_x p(x) = 1$.

A Markov process is a discrete time stochastic dynamical process that is defined by a transition matrix $T(x'|x)$ that specifies the transition probability from an initial state x to a final state x' in a single time step. Thus, $T(x'|x)$ is a $2^n \times 2^n$ matrix. For any x, $T(x'|x)$ is a probability vector in x':

$$\sum_{x'} T(x'|x) = 1$$

Matrices with this property are called stochastic matrices.

The Markov dynamics with initial state distribution $p_0(x)$ is

$$p_{t+1}(x') = \sum_x T(x'|x) p_t(x)$$

Markov processes

After long times, the Markov process reaches a stationary distribution.

$$p_0 \to p_1 = T\,p_0 \to p_2 = T\,p_1 \to \ldots$$

The stationary distribution p_∞ is a vector such that it is invariant under the dynamics: $T\,p_\infty = p_\infty$. Thus p_∞ is an eigenvector of T with eigenvalue 1. T always has at least one eigenvalue 1.

In the case that T has only one eigenvalue 1 the stationary distribution is unique. In this case the Markov process is called ergodic. A consequence is that the stationary distribution does not depend on the initial state.

If the eigenvalue 1 is degenerate the stationary distribution is not unique and the stationary distribution depends on the initial state.

The characteristic time it takes to reach stationarity is determined by the other eigenvalues of T.

Let us denote the eigenvalues and left and right eigenvectors of T by $\lambda_\alpha, l_\alpha, r_\alpha, \alpha = 1, \ldots, 2^n$, respectively [4]. In matrix notation we have

$$\begin{aligned} T r_\alpha &= \lambda_\alpha r_\alpha \\ l_\alpha^\dagger T &= \lambda_\alpha l_\alpha^\dagger \end{aligned}$$

Since T is a non-symmetric matrix, the left and right eigenvectors are different, non-orthogonal and complex valued. $\dagger$ denotes complex conjugation and transpose. The eigenvalues are complex valued. Under rather general conditions each set of eigenvectors spans a non-orthogonal basis of C^{2^n}. These two bases are dual in the sense that:

$$l_\alpha^\dagger r_\beta = \delta_{\alpha\beta}.$$

We see that the vector $a = (1, \ldots, 1)$ is a left eigenvector of T with eigenvalue 1:

$$(aT)(x) = \sum_{x'} a(x') T(x'|x) = \sum_{x'} T(x'|x) = 1 = a(x)$$

The corresponding right eigenvector is p_∞: $T p_\infty = p_\infty$.

[4] In general, the number of eigenvalues of T can be less than 2^n. However, for our purposes we can ignore this case

Some properties

A Markov process is called *irreducible*, or *ergodic*, on a subset of states $C \subset S$ if for any state $x \in C$ there is a finite probability to visit any other state $x' \in C$:

$$x = x^0, x^1, \ldots, x^k = x'$$

with $T(x^i|x^{i-1}) > 0, i = 1, \ldots, k$.

A subset of states $C \subset S$ is called *closed* when the Markov process can never escape from C, once entered:

$$T(x'|x) = 0 \quad \text{for all} \quad x \in C, x' \neg \in C.$$

In general, we can decompose the state space S uniquely into closed irreducible subsets C_i

$$S = T \cup C_1 \cup C_2 \ldots,$$

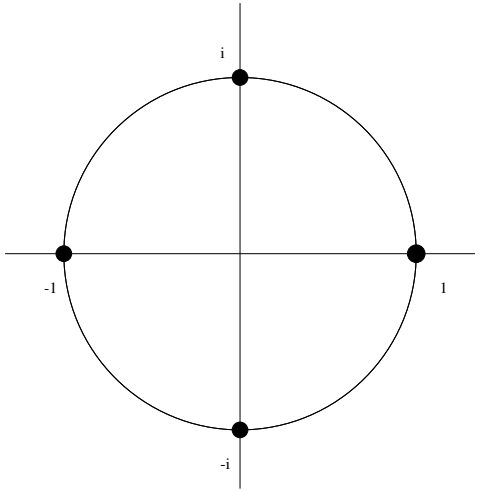
where T is a set of *transient states*.

Perron-Frobenius theorem

An irreducible Markov process T of *periodicity* d has d eigenvalues given by

$$\lambda_m = \exp(2\pi i m/d), m = 0, \ldots, d-1,$$

and all remaining eigenvalues of T are inside the unit circle in the complex plane: $|\lambda_\alpha| < 1$.



Characteristic times

We can expand T on the basis of its eigenvectors:

$$T = \sum_{\alpha=1}^{2^n} \lambda_\alpha r_\alpha l_\alpha^\dagger$$

5

In the ergodic case with periodicity 1:

$$p_t = T p_{t-1} = \ldots = T^t p_0 = \sum_\alpha \lambda_\alpha^t r_\alpha (l_\alpha^\dagger p_0) = p_\infty + \sum_{\alpha>1} \lambda_\alpha^t r_\alpha (l_\alpha^\dagger p_0)$$

[5] It is easy to check that this satisfies the eigen equations:

$$T r_\beta = \sum_\alpha \lambda_\alpha r_\alpha l_\alpha^\dagger r_\beta = \sum_\alpha \lambda_\alpha r_\alpha \delta_{\alpha\beta} = \lambda_\beta r_\beta$$

and similarly $l_\beta^\dagger T = \lambda_\beta l_\beta^\dagger$.

We can write

$$\lambda_\alpha^t = |\lambda_\alpha|^t e^{i\phi_\alpha t} = e^{-t/\tau_\alpha} e^{i\phi_\alpha t} \qquad \tau_\alpha = \frac{-1}{\log|\lambda_\alpha|}$$

The characteristic time to converge to the stationary distribution is determined by the largest $|\lambda_\alpha|$.

With higher periodicity the solution oscillates asymptotically:

$$p_t = \sum_{\alpha,|\lambda_\alpha|=1} \lambda_\alpha^t r_\alpha (l_\alpha^\dagger p_0) + \sum_{\alpha,|\lambda_\alpha|<1} \lambda_\alpha^t r_\alpha (l_\alpha^\dagger p_0) \quad \rightarrow \quad \sum_{m=0}^{d-1} e^{2\pi i m t/d} r_m (l_m^\dagger p_0)$$

For instance $p_0 = \frac{1}{2}(r_0 + r_1)$ and $d = 4$:

$$p_t(x) = \frac{1}{2}(r_0(x) + i^t r_1(x))$$

Non-ergodic behavior

A non-irreducible or non-ergodic Markov process has more than one eigenvalue 1 and therefore more than one left and right eigenvector with eigenvalue 1. Let us denote these eigenvectors by $l_1, \ldots, l_k$ and $r_1, \ldots, r_k$, respectively. Any linear combination of the right eigenvectors

$$p_\infty = \sum_{\alpha=1}^{k} \rho_\alpha r_\alpha$$

is therefore a stationary distribution, with parameters ρ_α such that $p_\infty(x) \geq 0$ for all x and proper normalization: $\sum_x p_\infty(x) = 1$. Thus, there exists a manifold of dimension $k - 1$ of stationary distributions.

The k left eigenvectors with eigenvalue 1 encode *invariants* of the dynamics

$$l_\alpha^\dagger p_{t+1} = l_\alpha^\dagger T p_t = l_\alpha^\dagger p_t \qquad \alpha = 1, \ldots, k$$

Since $l_1 \propto (1, \ldots, 1)$ the first invariant simply ensures invariance of normalisation $\sum_x p_{t+1}(x) = \sum_x p_t(x)$. Thus,

$$l_\alpha^\dagger p_0 = l_\alpha^\dagger p_\infty \qquad \alpha = 1, \ldots, k$$

The invariants determine uniquely the stationary distribution in terms of the initial conditions.

$$p_\infty = \sum_{\alpha=1}^{k} \rho_\alpha r_\alpha = \sum_{\alpha=1}^{k} (l_\alpha^\dagger p_0) r_\alpha$$

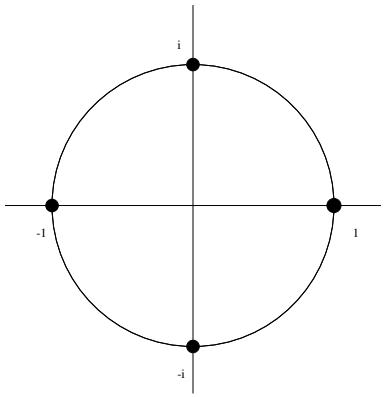
because $\rho_\alpha = l_\alpha^\dagger p_\infty = l_\alpha^\dagger p_0$.

Note, that in the ergodic case ($k = 1$) the dependence on the initial state disappears, as it should, since $l_1^\dagger p_0 = 1$ for any initial distribution p_0

Summary

The Markov process can be analysed in terms of the eigenvalues and eigenvectors of the transition matrix T.

1. There exists always an eigenvalue $\lambda = 1$.
   - If this eigenvalue is non-degenerate, the Markov process is called ergodic. The stationary distribution is unique and independent on the initial state.
   - If this eigenvalue is degenerate, the Markov process is called non-ergodic. The stationary distribution is not unique and depends on the initial state.
2. Markov processes can have multiple eigenvalues $e^{2\pi i m/d}, m = 0, \ldots, d-1$ and $|\lambda| = 1$. In this case it is called periodic with period d. In the most common case, the Markov process is non-periodic $d = 1$.
3. Eigenvalues with norm close to 1 ($|\lambda| = 1 - \epsilon$, ϵ small) imply long convergence times.



Detailed balance

Stationarity:

$$\sum_{x'} T(x|x')p(x') = p(x) = \sum_{x'} T(x'|x)p(x)$$

There always exists a stationary solution $p(x)$.

The Markov process T satisfies detailed balance if $\exists p$ such that

$$T(x|x')p(x') = T(x'|x)p(x) \text{ for all } x, x'.$$

Detailed balance is a property of the Markov dynamics T. It is not true in general.

DB holds for thermodynamic systems and Metropolis Hastings, not for chaotic systems.

If DB $p(x)$ is a stationary distribution of T. The reverse is not true.

The Metropolis algorithm again

At iteration r, sample x' from $q(x'|x^r)$ and compute

$$a = \frac{p^*(x')q(x^r|x')}{p^*(x^r)q(x'|x^r)}$$

If $a \geq 1$, accept x' as the new state: $x^{r+1} = x'$

Else, accept x' as the new state with probability a

If accept: $x^{r+1} = x'$, else $x^{r+1} = x^r$

The Metropolis algorithm is an ergodic (aperiodic) Markov process that satisfies detailed balance such that the stationary distribution is p.

Convergence of Metropolis algorithm

Metropolis algorithm is example of Markov process. Given two states x and x'. Define

$$a_{x'x} = \frac{p^*(x')q(x|x')}{p^*(x)q(x'|x)}, \; a_{xx'} = \frac{1}{a_{x'x}}$$

Suppose $a_{x'x} \geq 1$. Then

Given x**,** the probability to accept x' is

$$T(x'|x) = q(x'|x)$$

Given x'**,** the probability to accept x is

$$T(x|x') = q(x|x')a_{xx'}$$

$$\frac{T(x'|x)}{T(x|x')} = a_{x'x}\frac{q(x'|x)}{q(x|x')} = \frac{p^*(x')}{p^*(x)} = \frac{p(x')}{p(x)},$$

i.e. detailed balance. This implies that the process $T(x'|x)$ converges to $p(x)$.

Metropolis Hasting for the Ising model

Use MH to sample from the Ising model

$$p(x) = \frac{1}{Z}\exp\left(-E(x)\right) \qquad E(x) = -\frac{1}{2}\sum_{i\neq j} w_{ij}x_i x_j = -\sum_{(ij)} w_{ij}x_i x_j$$

Consider $q(x'|x)$ to implement single spin flips. Define $F_i x$ is the vector of spins obtained by flipping bit i. Then

$$q(F_i x|x) = \frac{1}{n} \qquad q(x|x) = 0$$

The MH ratio for the transition from state x to state $F_i x$ is

$$\begin{aligned} a_{F_i x,x} &= \frac{p^*(F_i x)}{p^*(x)} = \exp\left(-\Delta E\right) \\ \Delta E &= E(F_i x) - E(x) = 2x_i \sum_{j\neq i} w_{ij}x_j \end{aligned}$$

Example: 2 spins

$E = -wx_1x_2$.

There are four states $x = 1:4$ with energies:

$$E_{1:4} = \{E_{(++)}, E_{(+-)}, E_{(-+)}, E_{(--)}\} = \{-w, w, w, -w\}$$

We define $\Delta E_{x,x'} = E(x') - E(x)$:

$$\Delta E = \begin{pmatrix} 0 & 2w & 2w & 0 \\ -2w & 0 & 0 & -2w \\ -2w & 0 & 0 & -2w \\ 0 & 2w & 2w & 0 \end{pmatrix}$$

We define $q_{x,x'} = q(x'|x)$:

$$q = \frac{1}{2}\begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Example: 2 spins

Assume $w > 0$. We have for $x \neq x'$ $a_{x,x'} = \min(\exp(-\Delta E_{x,x'}), 1)$:

$$a = \begin{pmatrix} * & e^{-2w} & e^{-2w} & 1 \\ 1 & * & 1 & 1 \\ 1 & 1 & * & 1 \\ 1 & e^{-2w} & e^{-2w} & * \end{pmatrix}$$

For $x \neq x'$: $T(x'|x) = a_{x,x'} q_{x,x'}$:

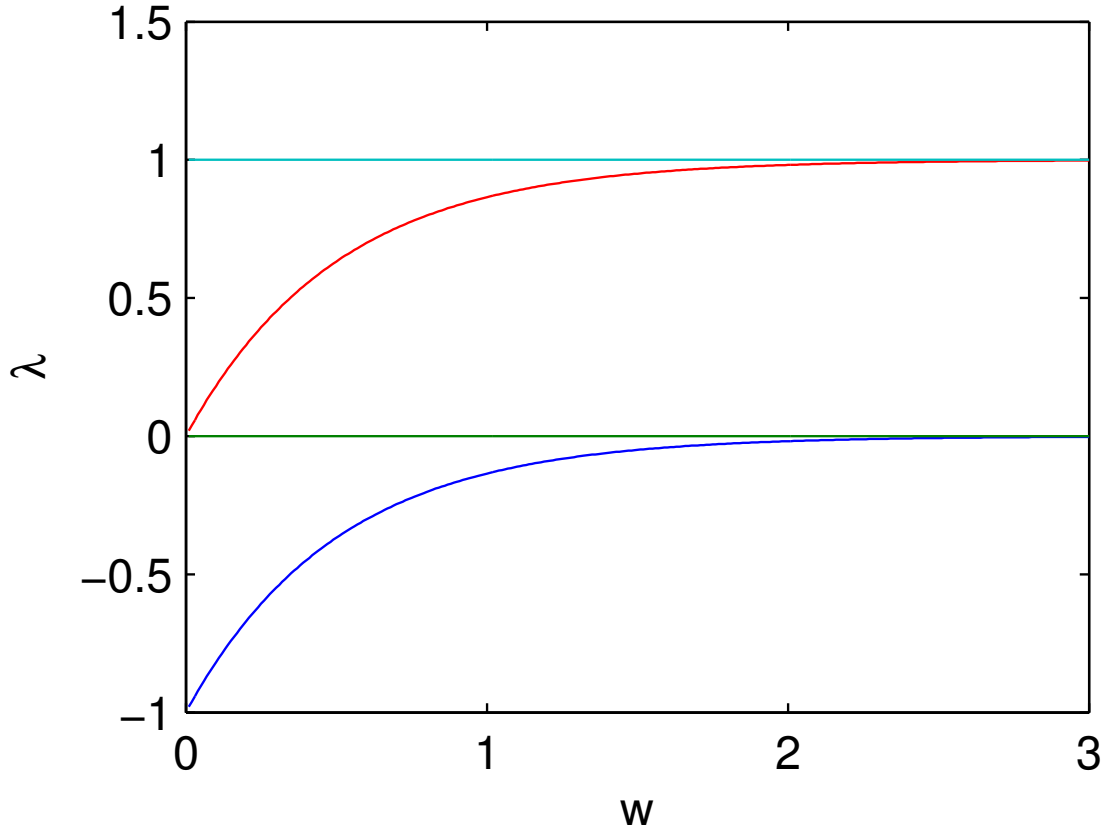
$$T = \begin{pmatrix} 1 - e^{-2w} & \frac{1}{2}e^{-2w} & \frac{1}{2}e^{-2w} & 0 \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 0 & \frac{1}{2}e^{-2w} & \frac{1}{2}e^{-2w} & 1 - e^{-2w} \end{pmatrix}$$

Diagonal terms follow from $\sum_{x'} T(x'|x) = 1$.

Interpretation of T: from high energy states 2,3 $((+,-)$ and $(-,+)$ spins transit with equal probability to one of the two low energy states 1,4 $((+,+)$ and $(-,-))$. From low energy states 1,4 their is finite probability to transit to high energy states (decreasing with w).

Example: 2 spins

Eigenvalues of T versus w. for low w there is one eigenvalue 1 and is ergodic. for high w there are two eigenvalues 1, system is non-ergodic.



Lecture 2. Monte Carlo sampling

Based on MacKay chapter 29 and 30.

- Uniform sampling
- Importance and Rejection sampling
- Metropolis-Hasting method
- Detour on Markov processes, stationary distribution, ergodicity
- Gibbs sampling
- Hybrid Monte Carlo
- Sampling the Bayesian posterior for the perceptron

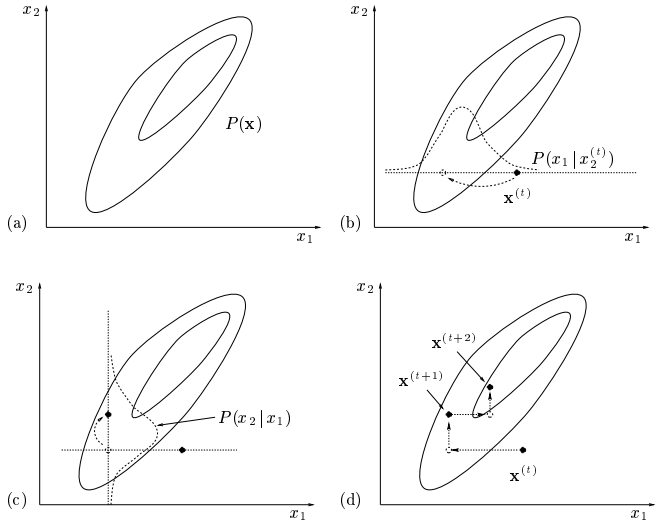
Gibbs sampling

- Consider only change of one element i of $(x_1, \ldots, x_n)$ at the time. Update $x \rightarrow x'$ according to

$$q(x'|x) = p(x'_i|x_{\setminus i})\delta_{x_{\setminus i}, x'_{\setminus i}}$$

  with $p(x'_i|x_{\setminus i})$ the conditional distribution from the target distribution $p(x)$. [6]

- Accept: $x^{r+1} = x'$



The one dimensional sampling can be done using for instance Rejection sampling.

[6] $x_{\setminus i}$ is the vector of variables without x_i. So if $x = (x_1, x_2, x_3)$ then $x_{\setminus 2} = (x_1, x_3)$.

Gibbs sampling

Each Gibbs sampling is a MH step with acceptance $a = 1$:

$$
\begin{aligned}
q(x'|x) &= p(x'_i|x_{\setminus i})\delta_{x_{\setminus i},x'_{\setminus i}} \\
a &= \frac{p(x')}{p(x)}\frac{q(x|x')}{q(x'|x)} = \frac{p(x'_i|x'_{\setminus i})p(x'_{\setminus i})}{p(x_i|x_{\setminus i})p(x_{\setminus i})}\frac{p(x_i|x'_{\setminus i})}{p(x'_i|x_{\setminus i})} = \frac{p(x'_i|x_{\setminus i})}{p(x_i|x_{\setminus i})}\frac{p(x_i|x_{\setminus i})}{p(x'_i|x_{\setminus i})} = 1
\end{aligned}
$$

where we used $x'_{\setminus i} = x_{\setminus i}$.

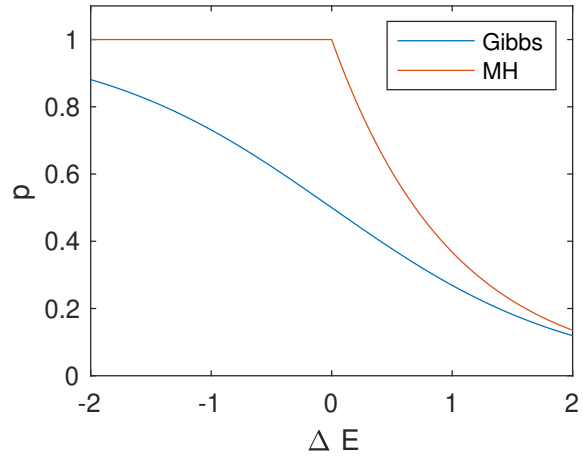
Comparison Gibbs sampling and MH sampling

$$p(x) = \frac{1}{Z}\exp\left(-E(x)\right)$$

Given state $x = (x_{\setminus i}, x_i)$, the probability to go to state $x' = (x_{\setminus i}, -x_i)$ is

Gibbs : $$p(x'|x) = p(-x_i|x_{\setminus i}) = \frac{p(-x_i, x_{\setminus i})}{p(x_{\setminus i})} = \frac{e^{-E(-x_i, x_{\setminus i})}}{e^{-E(x_i, x_{\setminus i})} + e^{-E(-x_i, x_{\setminus i})}} = \frac{1}{1 + e^{\Delta E_{x, F_i x}}}$$

MH : $$p(x'|x) = \min\left(e^{-\Delta E_{x, F_i x}}, 1\right)$$

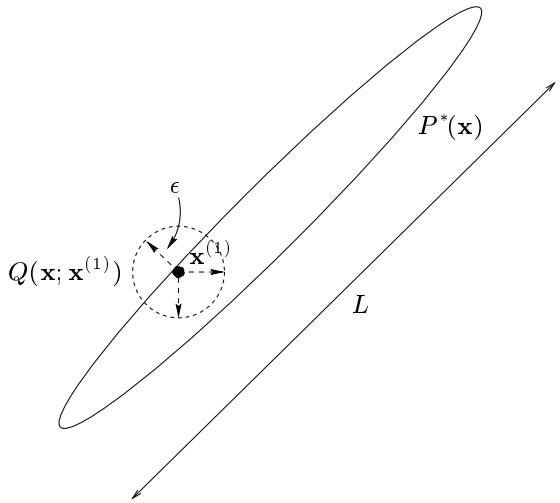


The detailed balance is the same

$$\text{Gibbs}: \quad \frac{p(x'|x)}{p(x|x')} = \frac{p(-x_i|x_{\setminus i})}{p(x_i|x_{\setminus i})} = \frac{1+e^{-\Delta E}}{1+e^{\Delta E}} = e^{-\Delta E} = \frac{p(x')}{p(x)}$$

$$\text{MH}: \quad \frac{p(x'|x)}{p(x|x')} = \frac{\min\left(e^{-\Delta E}, 1\right)}{\min\left(e^{\Delta E}, 1\right)} = e^{-\Delta E} = \frac{p(x')}{p(x)}$$

Correlations slow down sampling



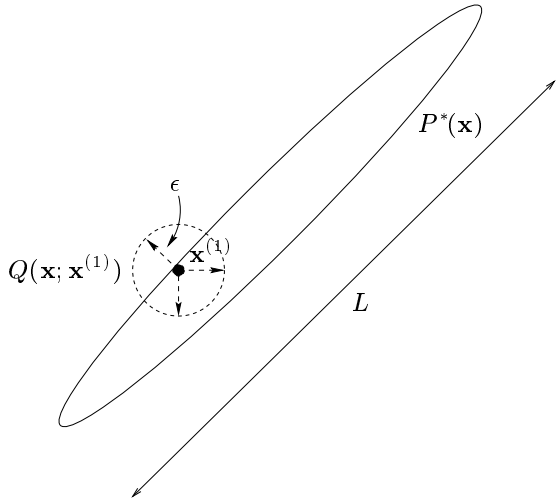
When $q(x'|x)$ is Gaussian centered on x, $\frac{q(x'|x)}{q(x|x')}$ independent of x, x':

$$a_{x'x} = \frac{p^*(x')}{p^*(x)}$$

ϵ large:
Acceptance rate $a_{x'x}$ small.

ϵ small:
Strong dependence on starting value. Many samples needed to sample.

Correlations slow down sampling



With step ϵ random, the particle moves a distance $L \approx \sqrt{T}\epsilon$ in T iterations. [7]

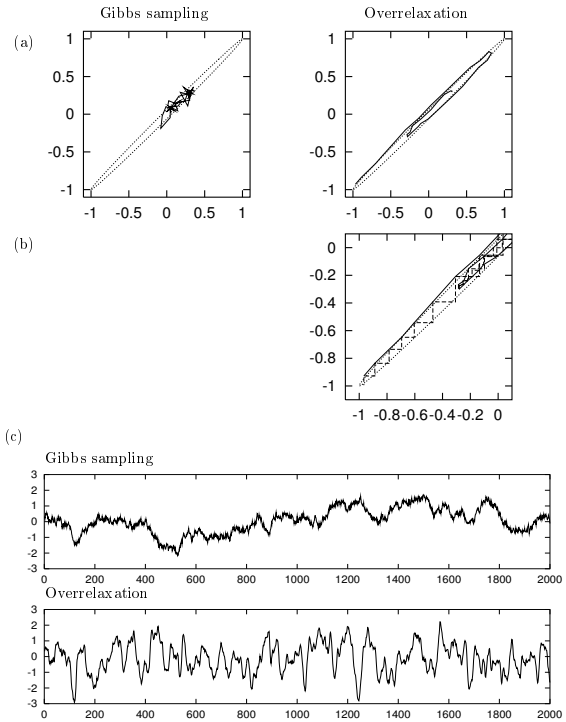
If largest length scale is L then

$$T \approx \left(\frac{L}{\epsilon}\right)^2$$

time steps are needed to sample the length L.

[7] Denote the total distance travelled $x = \sum_{i=1}^{T} x_i$ with x_i the distance in a single step with $\mathbb{E}x_i = 0$ and $\mathbb{V}x_i = \epsilon^2$. Then $\mathbb{E}x = 0$ and $\mathbb{V}x = \sum_{i=1}^{T} \mathbb{V}x_i = T\epsilon^2$. Thus the typical distance travelled is $L = \sqrt{T}\epsilon$

Correlations slow down sampling



Convergence slow when variables correlated: When two variables have marginal width L and conditional width ϵ, the number of iterations scales as $T \approx L^2/\epsilon^2$.

The Hybrid Monte Carlo Method

Let

$$P(x) = \frac{e^{-E(x)}}{Z}$$

with E and its gradient $\frac{\partial E}{\partial x_i}$ easy to compute.

Gradient information reduces random walk behaviour in Metropolis method.

Double the state space by introducing for each x_i a momentum p_i variable. Define the Hamiltonian and distribution

$$\begin{aligned} H(p,x) &= E(x) + \frac{\alpha}{2}\sum_i p_i^2 \\ P_H(p,x) &= \frac{1}{Z_H}\exp\left(-E(x) - \frac{\alpha}{2}\sum_i p_i^2\right) \end{aligned}$$

The marginal is $p(x) = \int dp P_H(p,x)$.

The Hybrid Monte Carlo Method

Generate sample (x', p') starting from (x, p) using the Hamiltonian dynamics and Metropolis-Hasting:

$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial x_i} \qquad \frac{dx_i}{dt} = \frac{\partial H}{\partial p_i}$$

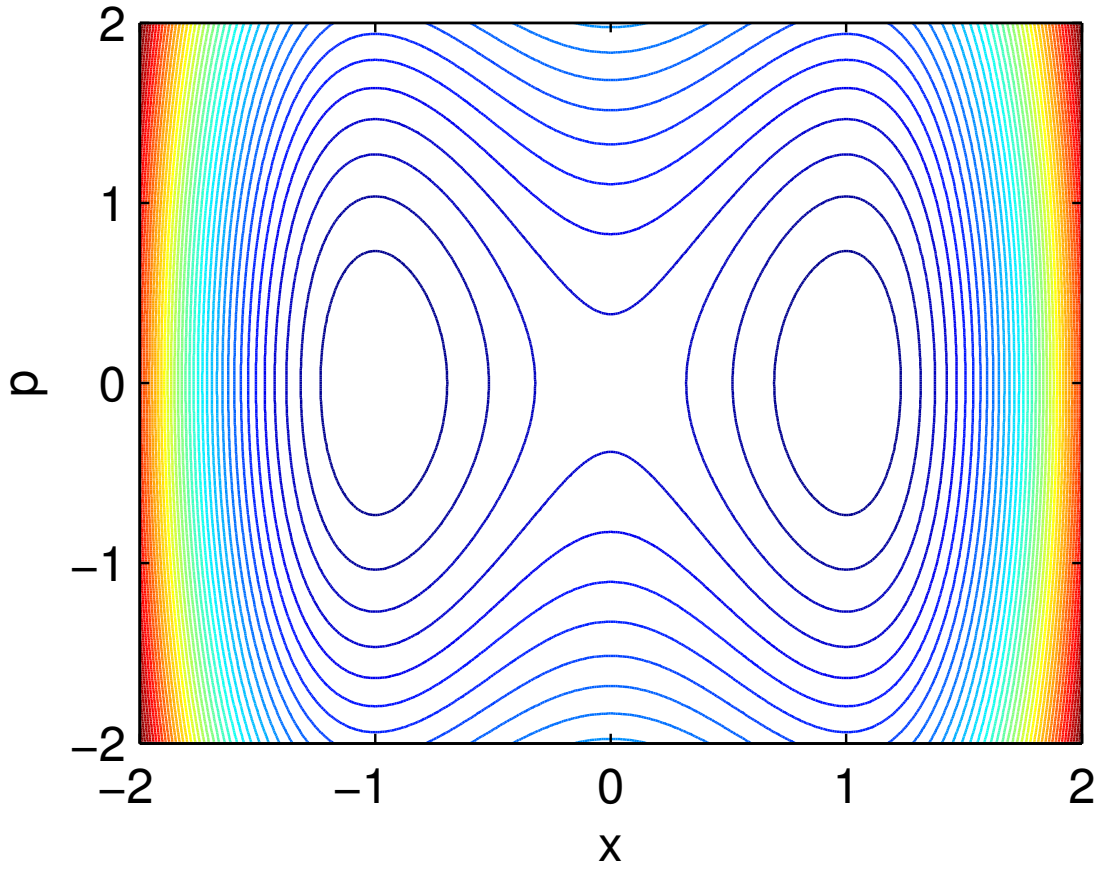
leaves H invariant.

$$\frac{dH}{dt} = \sum_i \frac{dH}{dp_i}\frac{dp_i}{dt} + \frac{dH}{dx_i}\frac{dx_i}{dt} = 0$$

Example

Consider the double well cost $E(x) = \left(x^2 - 1\right)^2$.

The Hamiltonian $H(x, p) = E(x) + \frac{1}{2}\alpha p^2$ has the form.



α small yields large p values and dynamical trajectories to large distances.

Pseudo code

Choose initial x^1.

For $t = 1 : T$:

1. choose p^t from $\mathcal{N}(0, \alpha^{-1})$, giving (x^t, p^t)

2. run Hamilton dynamics, giving (x', p')

3. Metropolis step: accept $(x^{t+1}, p^{t+1}) = (x', p')$ as new state with probability

$$\min(1, a) \qquad a = \frac{P_H(x', p')}{P_H(x, p)} = \frac{e^{-H(x', p')}}{e^{-H(x,p)}}$$

4. On rejection, $(x^{t+1}, p^{t+1}) = (x^t, p^t)$

NB: $a = 1$ by construction in theory, but deviates from 1 due to numerical integration errors.

Pseudo code

```
g = gradE ( x ) ;                    # set gradient using initial x
E = findE ( x ) ;                    # set objective function too

for l = 1:L                          # loop L times
   p = randn ( size(x) ) ;           # initial momentum is Normal(0,1)
   H = p' * p / 2 + E ;              # evaluate H(x,p)

   xnew = x ;  gnew = g ;
   for tau = 1:Tau                   # make Tau 'leapfrog' steps

      p = p - epsilon * gnew / 2 ; # make half-step in p
      xnew = xnew + epsilon * p ;  # make step in x
      gnew = gradE ( xnew ) ;      # find new gradient
      p = p - epsilon * gnew / 2 ; # make half-step in p

   endfor

   Enew = findE ( xnew ) ;           # find new value of H
   Hnew = p' * p / 2 + Enew ;
   dH = Hnew - H ;                   # Decide whether to accept

   if ( dH < 0 )                  accept = 1 ;
   elseif ( rand() < exp(-dH) ) accept = 1 ;
   else                           accept = 0 ;
   endif

   if ( accept )
      g = gnew ;    x = xnew ;     E = Enew ;
   endif
endfor
```

Example 2

$$H(x,p) = \frac{1}{2}\beta_1 x_1^2 + \frac{1}{2}\beta_2 x_2^2 + \frac{1}{2}(p_1^2 + p_2^2)$$

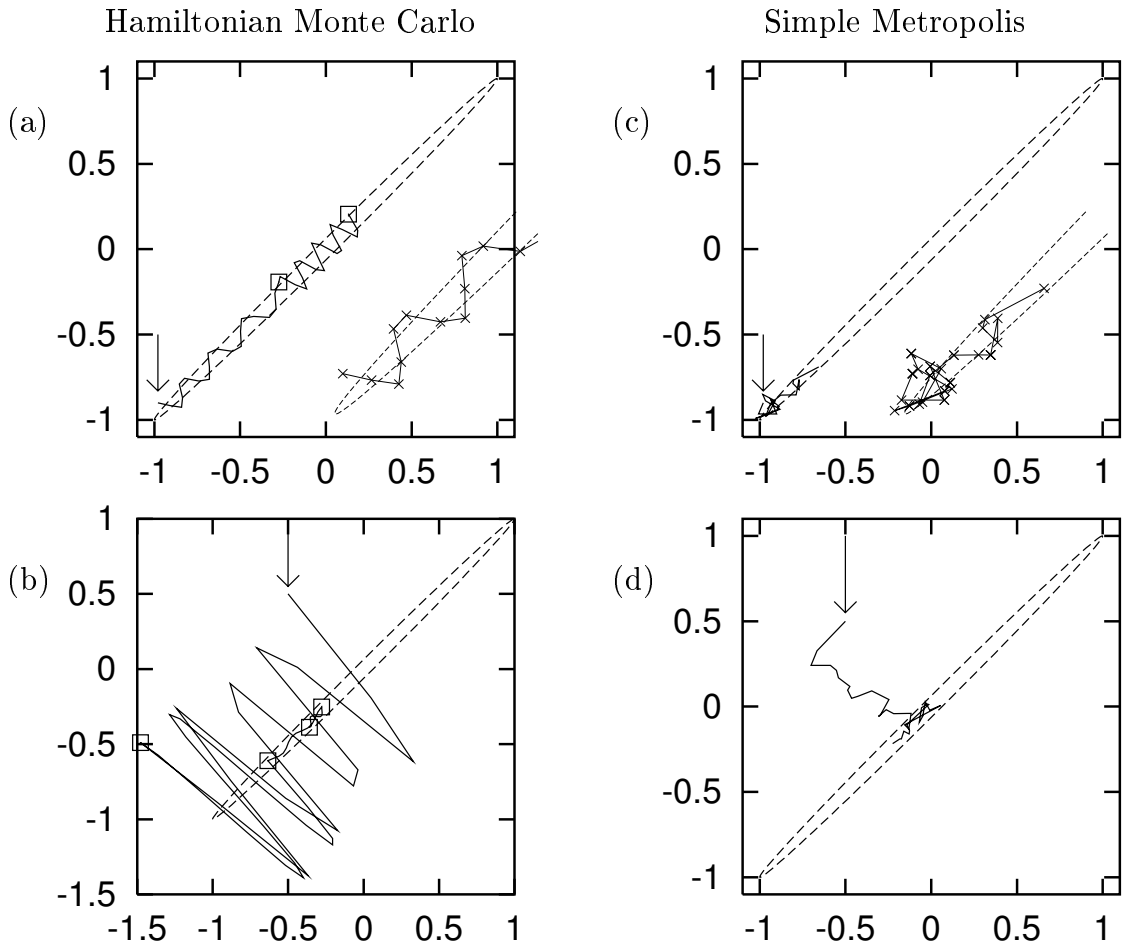
The dynamics is

$$\begin{aligned} \dot{x}_i &= \frac{\partial H(x,p)}{\partial p_i} = p_i \qquad \dot{p}_i = -\frac{\partial H(x,p)}{\partial x_i} = -\beta_i x_i \\ \ddot{x}_i &= -\beta_i x_i \end{aligned}$$

The solution is of the form $x_i(t) = \sin\left(\sqrt{\beta_i}t\right)$.

If $\beta_1 \gg \beta_2$, oscillations in x_1 much faster than in x_2.

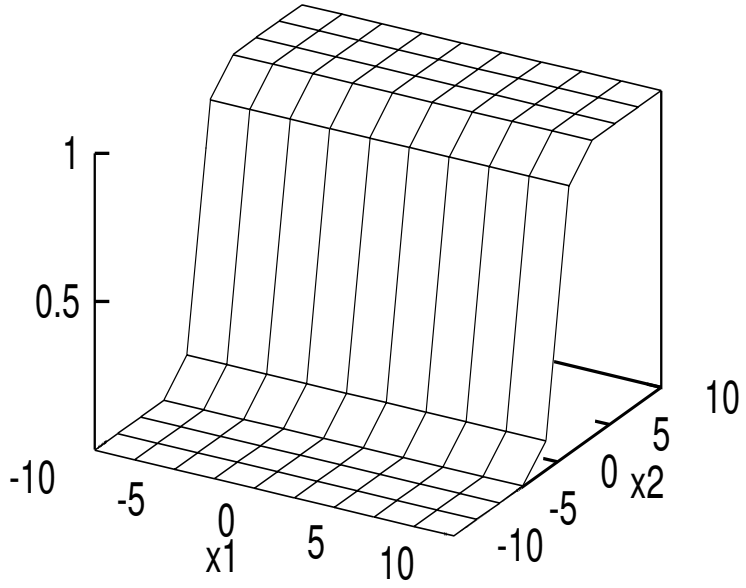
Comparison of HMC and Metropolis



Two dimensional elongated Gaussian distribution. a-b) Hybrid Monte Carlo method c-d) Metropolis method.

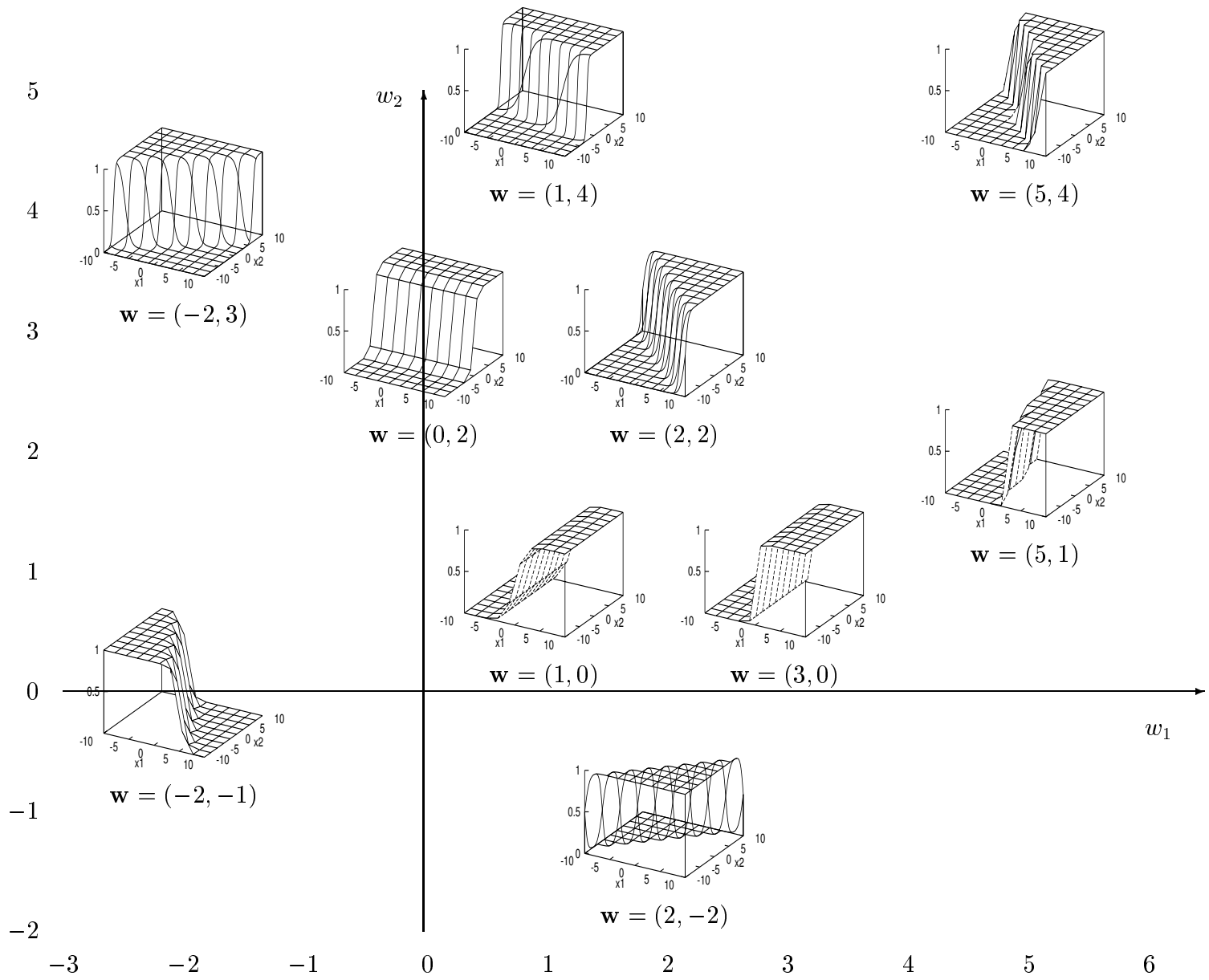
Perceptron/logistic regression

$$
\begin{aligned}
p(t=1|x,w) &= \sigma(\vec{w}\cdot\vec{x}) \\
\sigma(x) &= \frac{1}{1+\exp(-2x)} \\
\vec{w}\cdot\vec{x} &= w_0 + w_1 x_1 + w_2 x_2
\end{aligned}
$$



$\mathbf{w} = (0, 2)$

Perceptron/logistic regression



Learning as inference

Data set: $\{x^\mu, t^\mu\}, \mu = 1, \ldots, P$ with $t^\mu = \pm 1$.

Probability of data point under the model: $p(t^\mu | x^\mu, w) = \sigma(t^\mu w \cdot x^\mu)$

Likelihood:

$$p(D|w) = \prod_\mu p(t^\mu | x^\mu, w) = \exp(-G(w)) \qquad G(w) = -\sum_\mu \log(p(t^\mu | x^\mu, w))$$

Prior:

$$p(w) = \frac{\exp(-\alpha E_w(w))}{Z_w(\alpha)} \qquad E_w(w) = \sum_i w_i^2$$

makes solutions with small weights more probable.

Posterior:

$$\begin{aligned} p(w|D) &= \frac{p(D|w)p(w)}{p(D)} \propto \exp(-M(w)) \\ M(w) &= G(w) + \alpha E_w(w) \end{aligned}$$

ML versus Bayesian

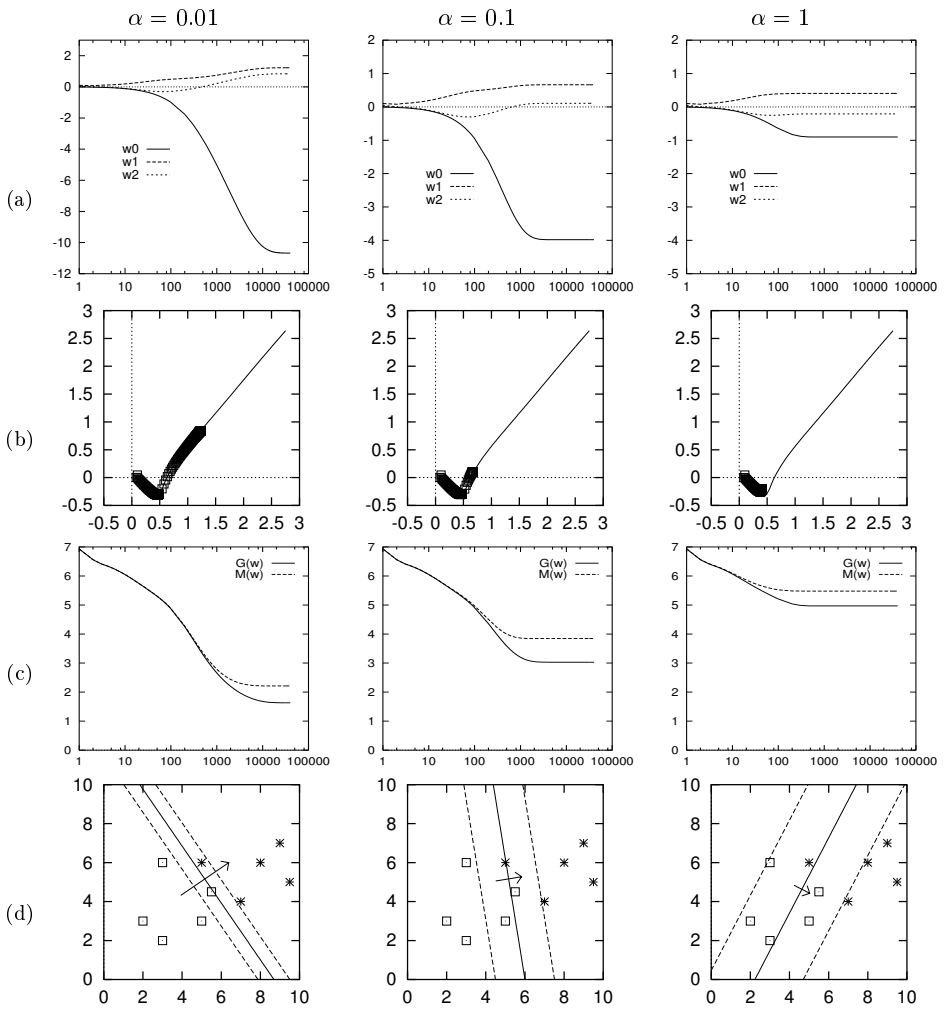
Standard in neural network learning is to compute the maximum likelihood or maximum posterior solution. For new test point a

$$\begin{aligned} D &\rightarrow w_{\text{ml}} \\ p(t|a) &= p(t|a, w_{\text{ml}}) \end{aligned}$$

Bayesian approach requires integration over multiple solutions:

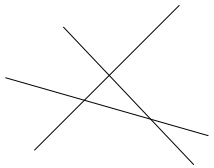
$$\begin{aligned} D &\rightarrow p(w|D) \\ p(t|a) &= \int dw p(w|D) p(t|a, w) = \langle p(t|a, w) \rangle_{p(w|D)} \end{aligned}$$

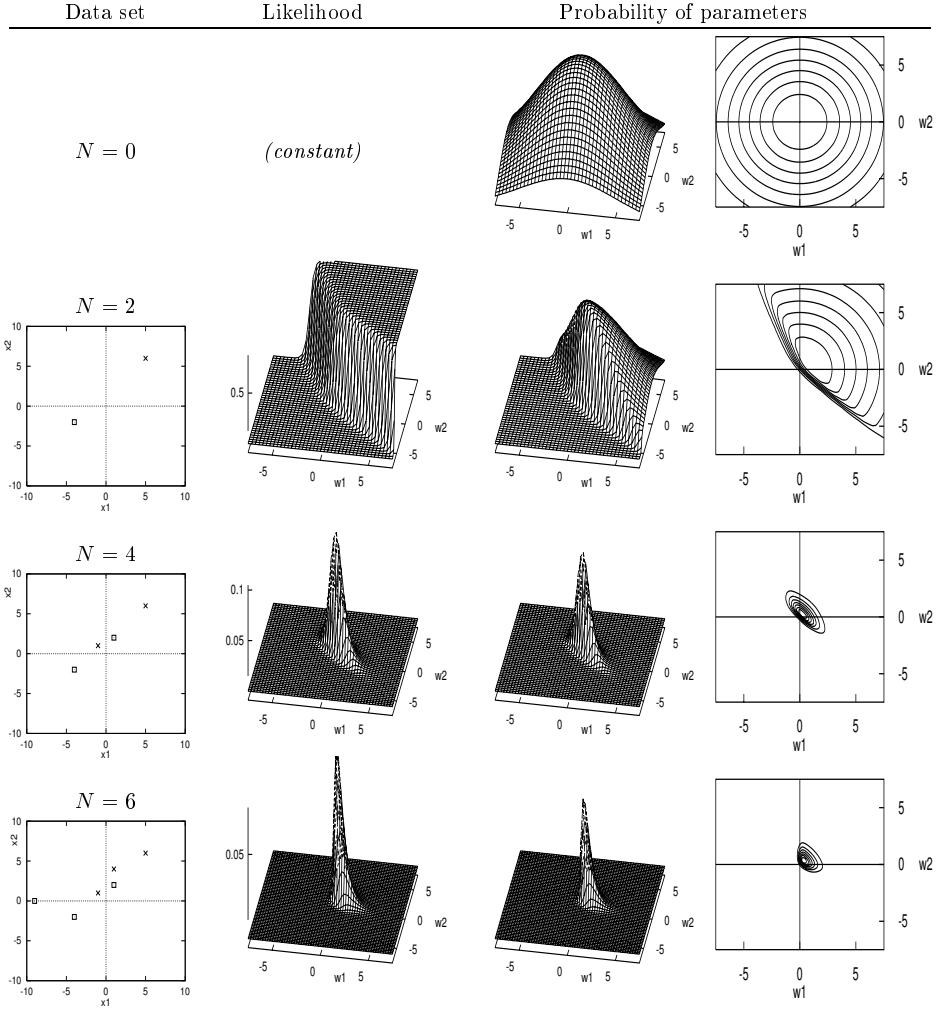
The maximum posterior solution



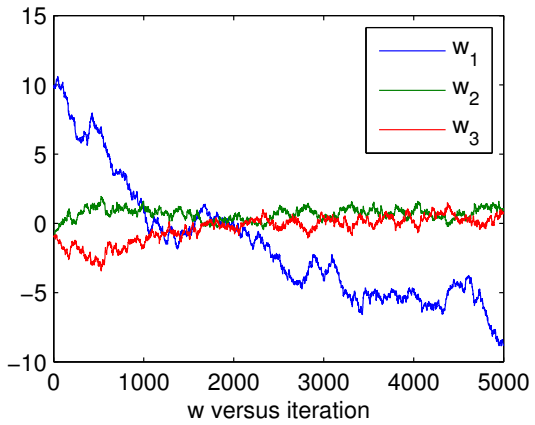
Minimizing the cost function $M(w) = G(w) + \alpha E_w(w)$ yields smoother solutions for larger α.

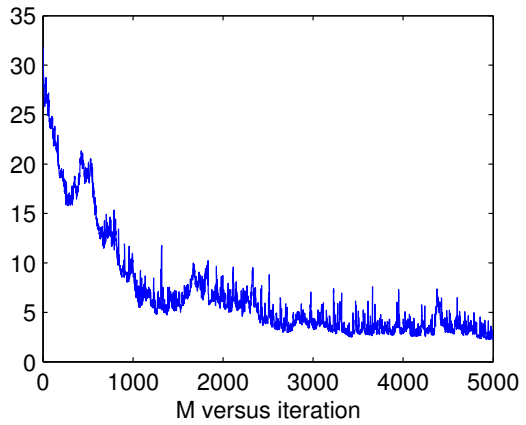
The full Bayesian solution

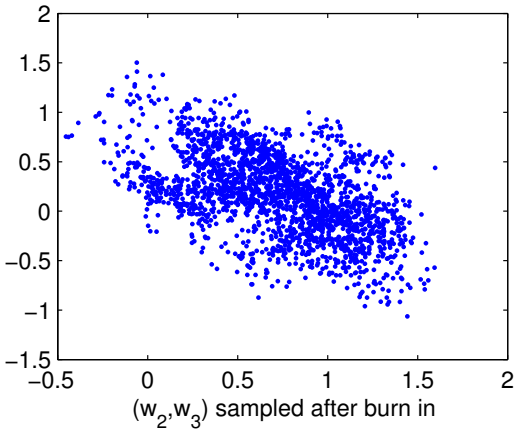


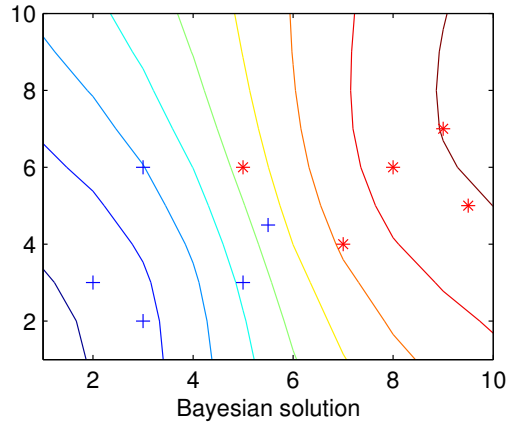


The full Bayesian solution



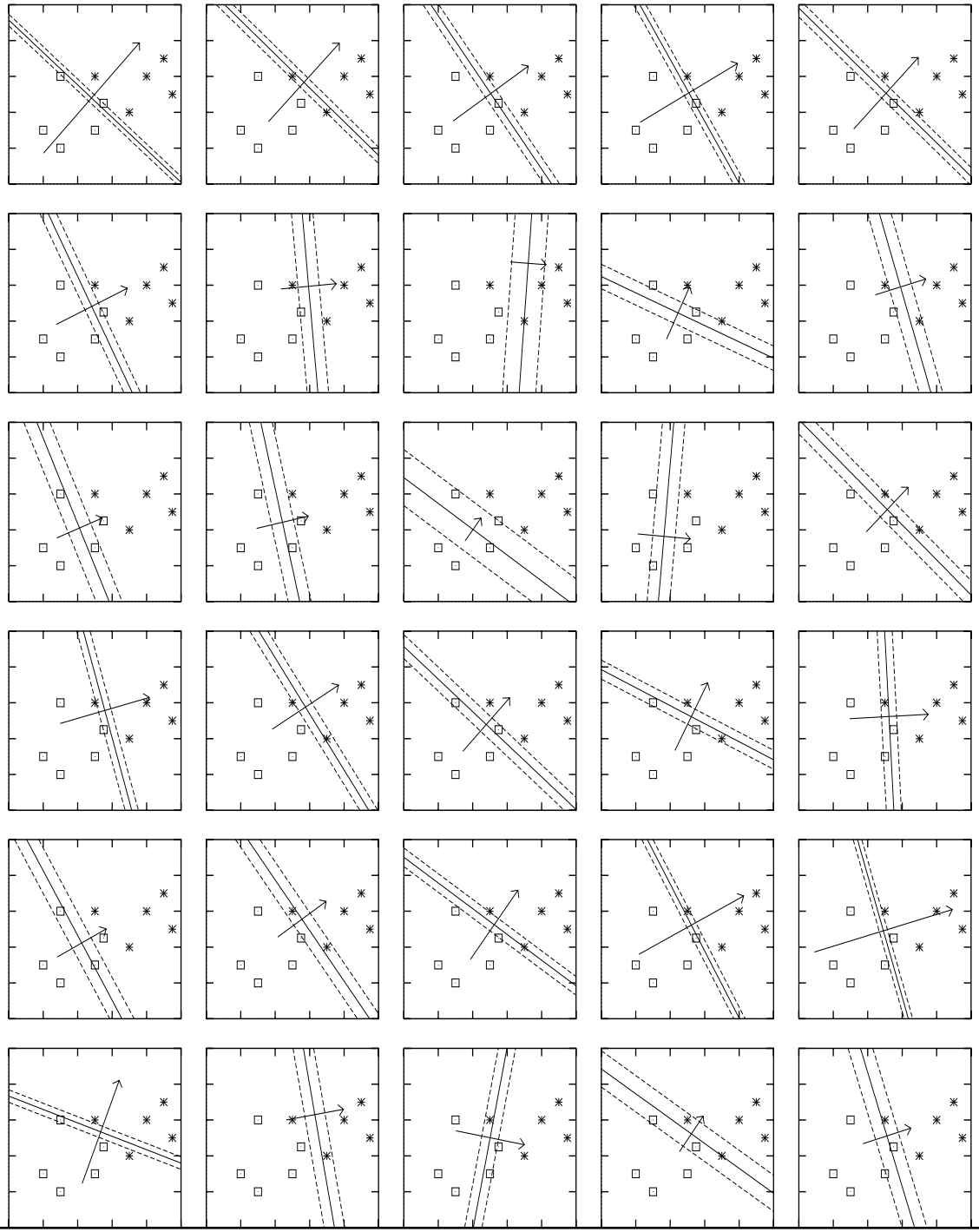






$$\alpha = 0.01, q(w'|w) = \mathcal{N}(w'|w, \sigma), \sigma = 0.1$$

$$p(t|x) = \int dw p(t|x,w)p(w|D) \approx \frac{1}{R}\sum_r p(t|x,w^r)$$



Lecture 3. The Ising model

- Illustration of Metropolis Hasting algorithm for the Ising model
- Phase transition: qualitative change when going from high temperature to low temperature. Ferro-magnetic Ising model.
- Critical slowing down
- Frustration in the anti-ferromagnetic Ising model
  - Transfer matrix method for computation of the partition sum
- Discrete optimization with iterative improvement and simulated annealing

The Ising model

Ising model is a probability distribution over vectors of binary variables. $x = (x_1, \ldots, x_N)$ with $x_i = \pm 1$.

$$p(x) = \frac{1}{Z} \exp\left(-\beta E(x)\right) \qquad E(x) = -\frac{1}{2} \sum_{i \neq j} J_{ij} x_i x_j - \sum_i h_i x_i$$

$$Z = \sum_x \exp(-\beta E(x)) = \sum_{x_1} \ldots \sum_{x_N} \exp(-\beta E(x))$$

$\beta = 1/kT$ is the inverse temperature, can be absorbed in the coupling. [8]

[8] It is sometimes convenient to write $\frac{1}{2} \sum_{i \neq j} J_{ij} x_i x_j = \sum_{(ij)} J_{ij} x_i x_j$.

Some basic relations

The mean energy of the spin system $\langle E \rangle = \sum_x E(x)p(x)$ can be computed from Z:

$$\frac{\partial \log Z}{\partial \beta} = \frac{1}{Z}\frac{\partial Z}{\partial \beta} = \frac{1}{Z}\sum_x -E(x)\exp(-\beta E(x)) = -\langle E \rangle$$

The energy fluctuations are computed similarly:

$$\begin{aligned}\frac{\partial^2 \log Z}{\partial \beta^2} &= \frac{\partial}{\partial \beta}\left(\frac{1}{Z}\frac{\partial Z}{\partial \beta}\right) = -\frac{1}{Z^2}\left(\frac{\partial Z}{\partial \beta}\right)^2 + \frac{1}{Z}\frac{\partial^2 Z}{\partial \beta^2} \\ &= -\langle E \rangle^2 + \frac{1}{Z}\sum_x E(x)^2 \exp(-\beta E(x)) = -\langle E \rangle^2 + \left\langle E^2 \right\rangle = \mathbb{V}(E)\end{aligned}$$

One calls $F = -\frac{1}{\beta}\log Z$ the free energy, or a log partition sum.

$$F = \langle E \rangle - \frac{1}{\beta}H \qquad ex.31.1$$

Metropolis Hasting for the Ising model

Use MH to sample from the Ising model

$$p(x) = \frac{1}{Z}\exp\left(-E(x)\right) \qquad E(x) = -\sum_{(ij)} w_{ij}x_i x_j$$

Consider $q(x'|x)$ to implement single spin flips. Define $F_i x$ is the vector of spins obtained by flipping bit i. Then

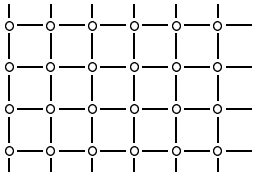
$$q(F_i x|x) = \frac{1}{n} \qquad q(x|x) = 0$$

The MH ratio for the transition from state x to state $F_i x$ is

$$\begin{aligned} a_{F_i x, x} &= \frac{p^*(F_i x)}{p^*(x)} = \exp\left(-\Delta E\right) \\ \Delta E &= E(F_i x) - E(x) = 2x_i \sum_{j \neq i} w_{ij} x_j \end{aligned}$$

Ferro-magnet

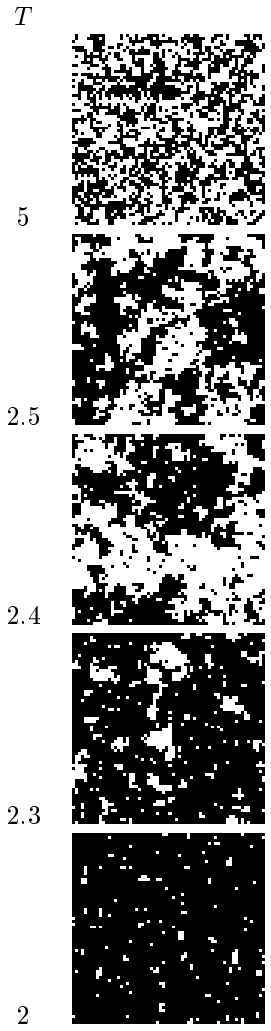
Rectangular grid with $J = 1$ and $h_i = 0$ *Ferro-magnet*, periodic boundary.



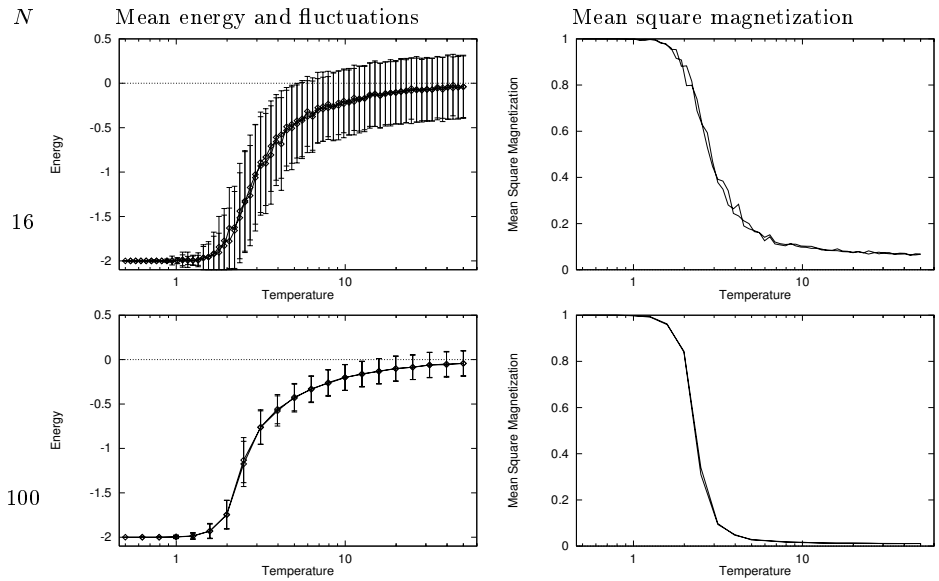
Run system at different temperatures $\beta = 0.03 \rightarrow 10 \rightarrow 0.03$ in equilibrium (check for hysteresis)
Run T iterations at each temperature, discard first 1/3 for 'burn in'.
record $\langle E \rangle, \mathbb{V}E, \left\langle m^2 \right\rangle$ as time averages:

$$\langle E \rangle = \frac{1}{T} \sum_{t=1}^{T} E(x_t) \qquad \mathbb{V}E = \frac{1}{T} \sum_{t=1}^{T} (E_t - \langle E \rangle)^2$$

$$\left\langle m^2 \right\rangle = \frac{1}{T} \sum_{t=1}^{T} m^2(x_t) \qquad m(x_t) = \frac{1}{N} \sum_{i=1}^{N} x_i$$



Ferro-magnet: mean energy and magnetizationn



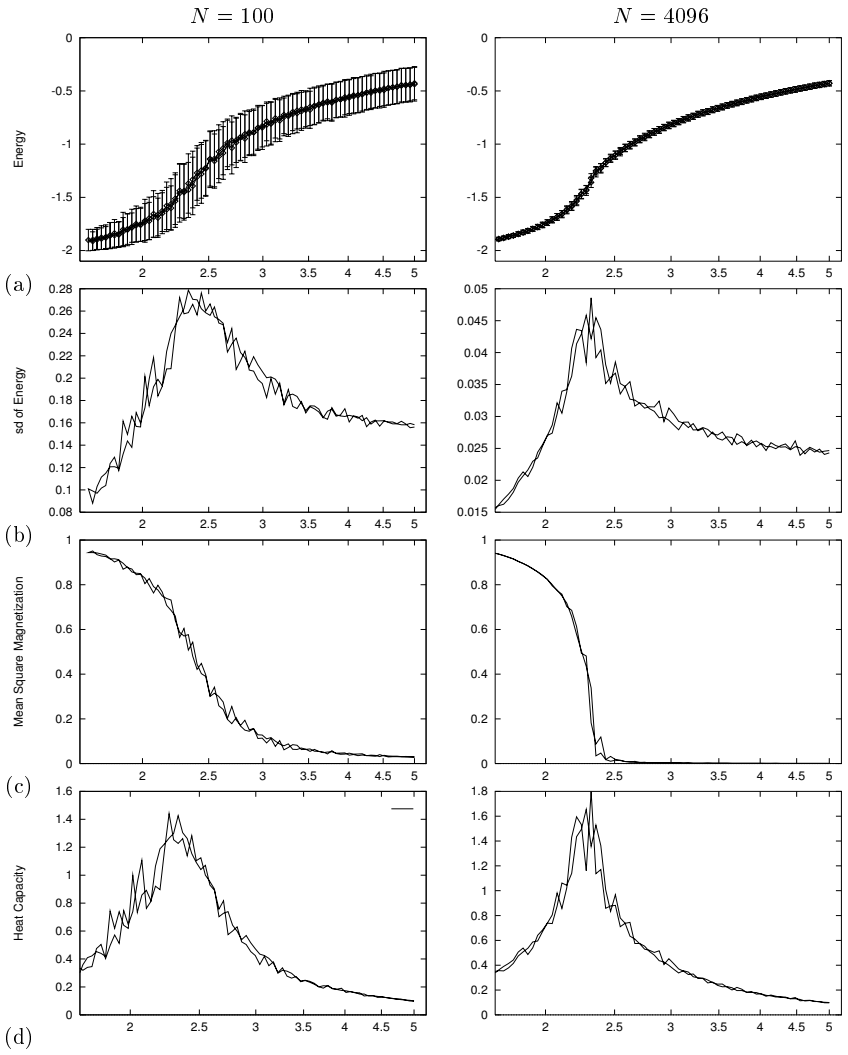
Left: Mean energy. Each spin has 4 links, thus E has $2N$ terms.

- at low temperature $\langle E \rangle / N = -2 \left\langle x_i x_j \right\rangle = -2$ (for one of the two ground states).
- At high temperature $\langle E \rangle / N = -2 \left\langle x_i x_j \right\rangle \approx 0$ and $\mathbb{V}(E/N) \propto 1/N$.

Right: Mean squared magnetisation

- is zero for high temperature
- and 'breaks' to 1 for low temperature.

Ferro-magnet: energy fluctuations



Energy fluctuations increase around the critical temperature

Phase transitions

Phase transitions are values of β for which the derivatives of

$$\log Z(\beta) = \log \sum_x e^{-\beta E(x)}$$

are discontinuous or diverge.

Phase transitions can only occur when $N \to \infty$, because for finite N $Z(\beta)$ is a smooth function.

Phase transitions

Consider the toy model: $E(x) = -\epsilon N$ when $x = (0, 0, \ldots, 0)$ and $E(x) = 0$ otherwise $(x_i = 0, 1)$. Then

$$Z = e^{N\beta\epsilon} + 2^N - 1 \qquad \lim_{\beta\to\infty} \log Z = N\beta\epsilon \qquad \lim_{\beta\to 0} \log Z = N \log 2$$

$$\frac{d \log Z}{d\beta} = N\epsilon \frac{e^{N\beta\epsilon}}{e^{N\beta\epsilon} + 2^N - 1} \qquad \frac{d^2 \log Z}{d\beta^2} = N^2\epsilon^2 \frac{(2^N - 1)e^{N\beta\epsilon}}{(e^{N\beta\epsilon} + 2^N - 1)^2}$$

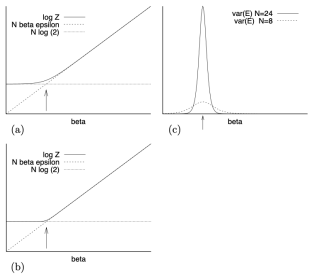


Figure B.1. (a) Partition function of toy system which shows a phase transition for large N. The arrow marks the point $\beta_c = \log 2/\epsilon$. (b) The same, for larger N. (c) The variance of the energy of the system as a function of β for two system sizes. As N increases the variance has an increasingly sharp peak at the critical point β_c. Contrast with figure B.2.

When $N \to \infty$ and $\beta = \log 2/\epsilon$, $\langle E \rangle = -\frac{N\epsilon}{2}$ and the fluctuations $\mathbb{V}(E) = \frac{N^2\epsilon^2}{4}$:

$$\langle E \rangle / N = -\frac{\epsilon}{2} \pm \frac{\epsilon}{2}$$

Phase transitions

Compare with the independent model $E(x) = \epsilon \sum_{i=1}^{N} (2x_i - 1)$. Then

$$Z = \left(e^{\beta\epsilon} + e^{-\beta\epsilon}\right)^N \qquad \lim_{\beta\to\infty} \log Z = N\beta\epsilon \qquad \lim_{\beta\to 0} \log Z = N \log 2$$

Since $\log Z = N f(\beta)$ we find for $\beta = \frac{\log 2}{\epsilon}$ that $\langle E \rangle = -\frac{3\epsilon N}{5}$ and $\mathbb{V}(E) = \frac{16}{25} N\epsilon^2$.

$$\langle E \rangle / N = -\frac{3\epsilon}{5} \pm \frac{4\epsilon}{5\sqrt{N}}$$

The energy fluctuations per spin vanish.

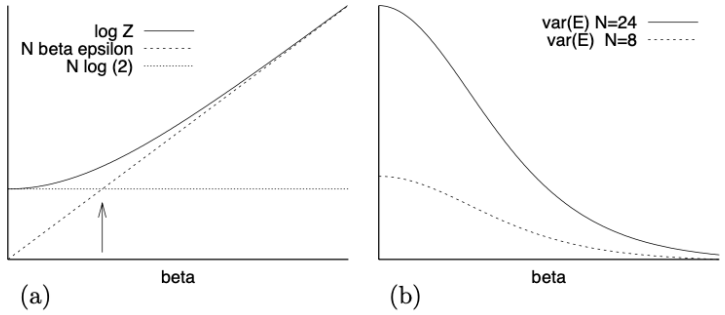


Figure B.2. The partition function (a) and energy-variance (b) of a system consisting of N independent spins. The partition function changes gradually from one asymptote to the other, regardless of how large N is; the variance of the energy does not have a peak. The fluctuations are largest at high temperature (small β) and scale linearly with system size N.

Autocorrelation

During sampling $x_1, x_2, \ldots, x_N$ one obtains samples of the quantity of interest $Q_t = \phi(x_t)$. One can estimate the autocorrelation

$$A(\tau) = \frac{\langle Q_t Q_{t+\tau} \rangle - \langle Q_t \rangle^2}{\left\langle Q_t^2 \right\rangle - \langle Q_t \rangle^2} \propto e^{-\tau/\Theta}$$

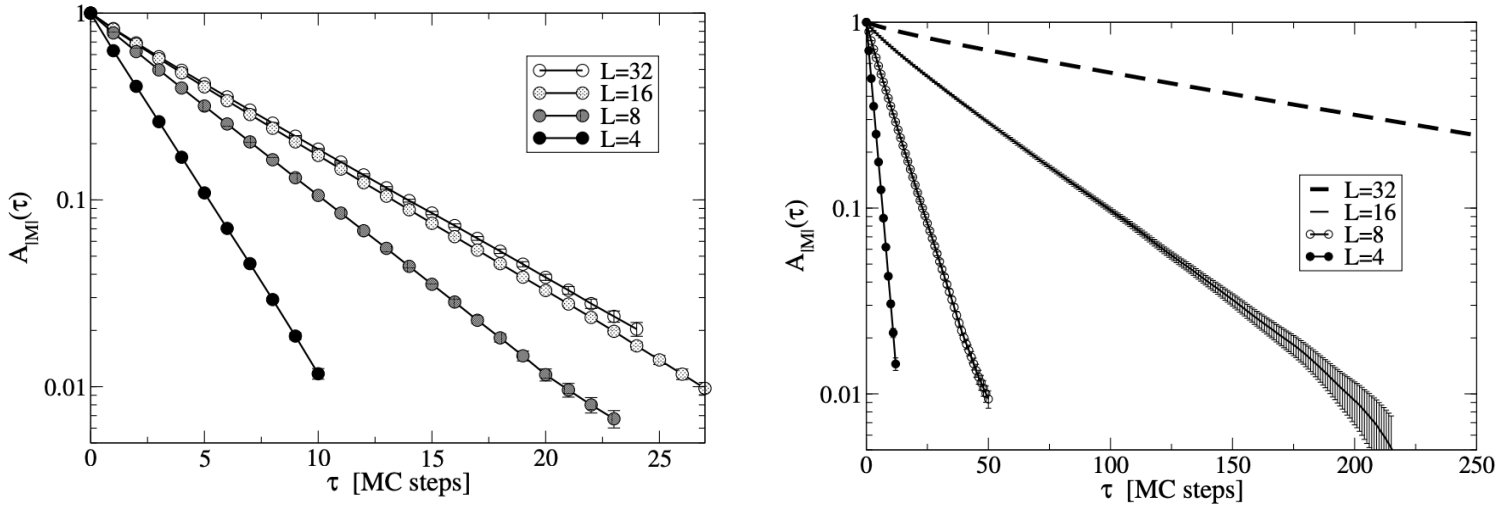
Note that $A(0) = 1$. One expects for ergodic problems that for large τ

$$\langle Q_t Q_{t+\tau} \rangle = \langle Q_t \rangle \langle Q_{t+\tau} \rangle = \langle Q_t \rangle^2 \qquad \lim_{\tau \to \infty} A(\tau) = 0$$

Θ is the characteristic time on which the samples of the Markov process are correlated. Thus, the effective number of independent samples is N/Θ.

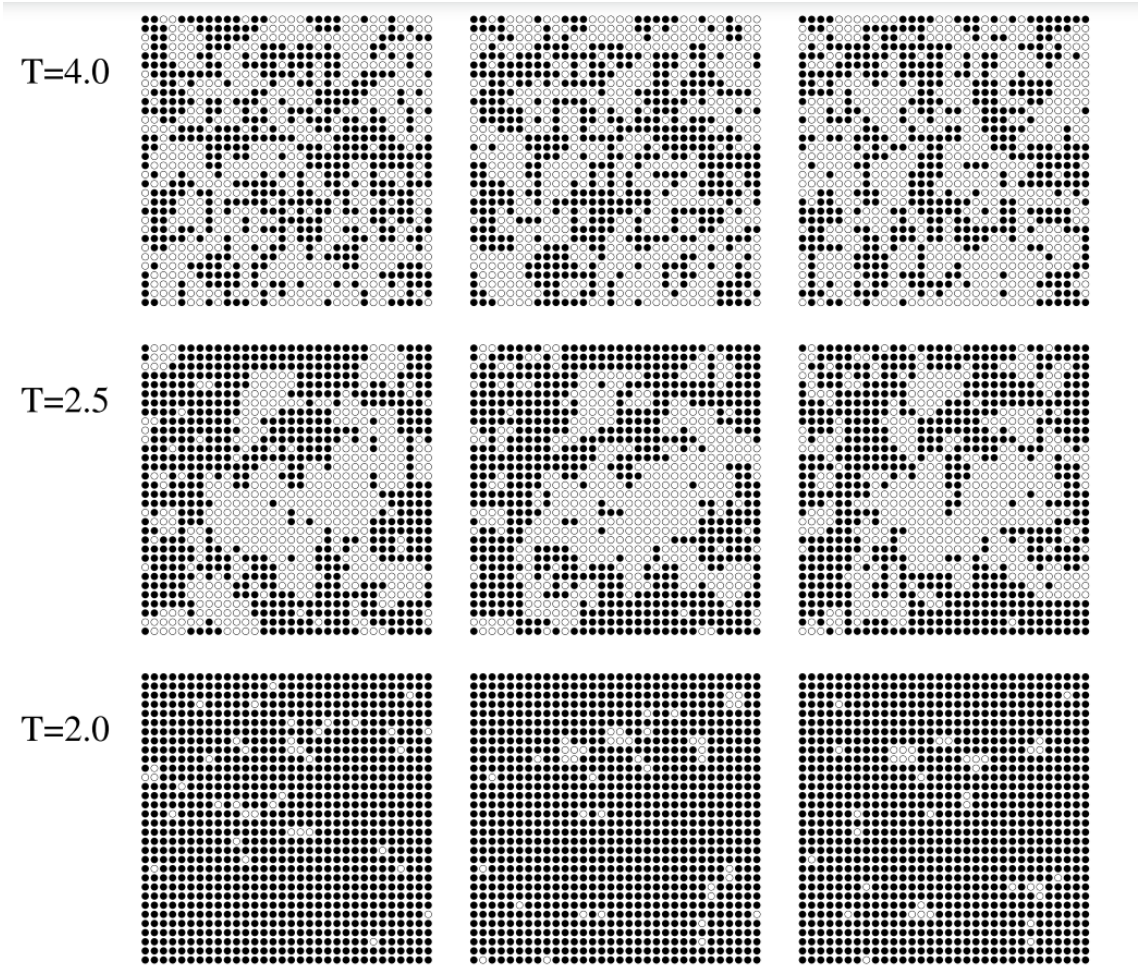
Autocorrelation

Characteristic autocorrelation time Θ is finite independent of system size at $T \neq T_c$, but diverges with system size at $T = T_c$. This is known as critical slowing down. It is related to the divergence of the fluctuations at $T = T_c$.



Autocorrelation function for $\phi(x) = \left|\frac{1}{n}\sum_{i=1}^{n} x_i\right|$ in 2D Ising models of different sizes at $T = 3$ (left) and $T = T_c$ (right) with $T_c = 2/\log(1 + 2\sqrt{2}) \approx 2.269$).

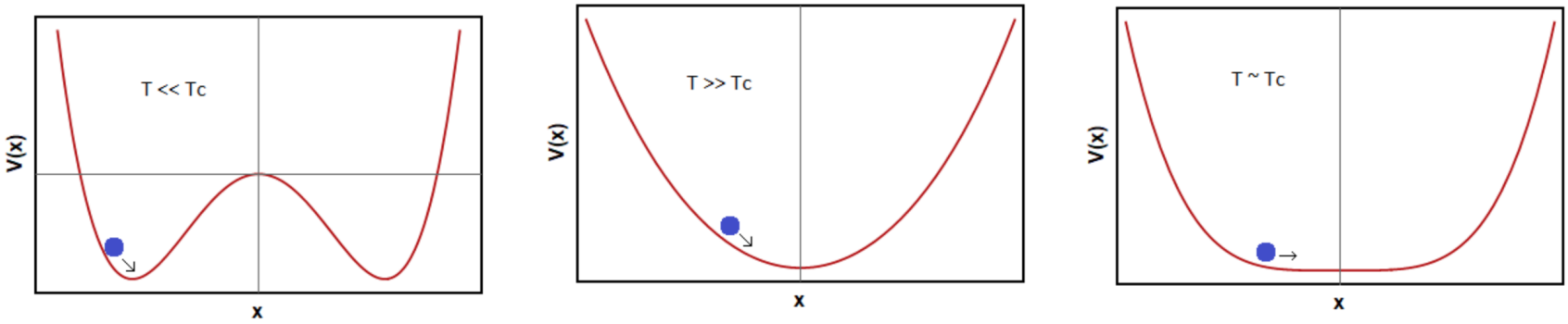
Autocorrelation



Snapshots of the Markov process on a two dimensional $L = 32$ Ising lattice at three different times and three different temperatures. The middle row shows critical slowing down: the configuration hardly changes. The top and bottom rows show significant change.

Critical slowing down

The idea can be explained within the Landau theory of phase transitions. Model $V(x) = \alpha(T - T_c)x^2 + \frac{1}{4}\beta x^4$.



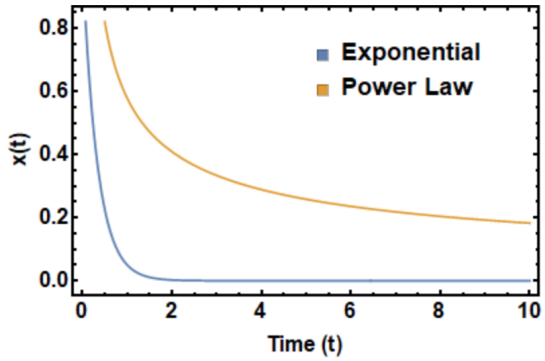
When $T \ll T_c$ or $T \gg T_c$, the potential well can be approximated by a quadratic form. In both cases, the relaxation to the optimal value for $y = x - x_{\text{opt}}$ is described as a locally quadratic potential $V(y) = \frac{1}{2}\gamma y^2$. The relaxation dynamics is

$$\dot{y} = -\frac{dV}{dy} = -\gamma y \qquad y(t) = y(0)e^{-\gamma t}$$

Thus, the relaxation is exponentially fast with characteristic time $\Theta = \gamma^{-1}$.

For $T = T_c$, the quadratic term vanishes and the potential is $V(y) = \frac{1}{4}\beta y^4$ and

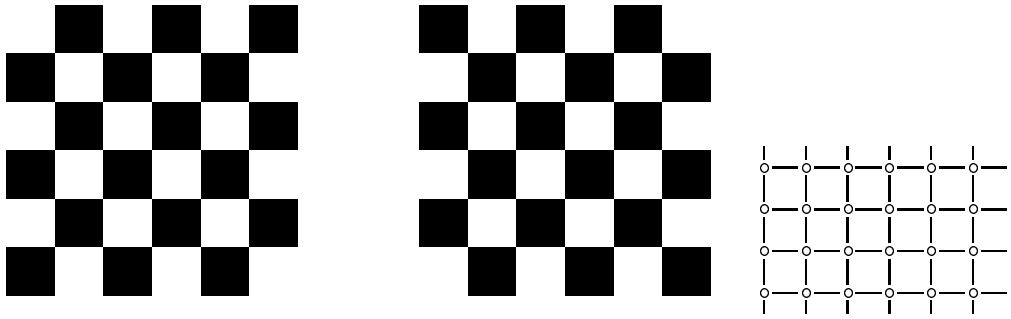
$$\dot{y} = -\beta y^3 \qquad y(t) = \frac{1}{\sqrt{\beta t}}$$



The convergence is much slowed down.

Anti-ferromagnet

Anti-ferromagnet has $J = -1$ between neighbors on the grid.



Ground states of the anti-ferromagnet are one-to-one with ground states in ferromagnet.

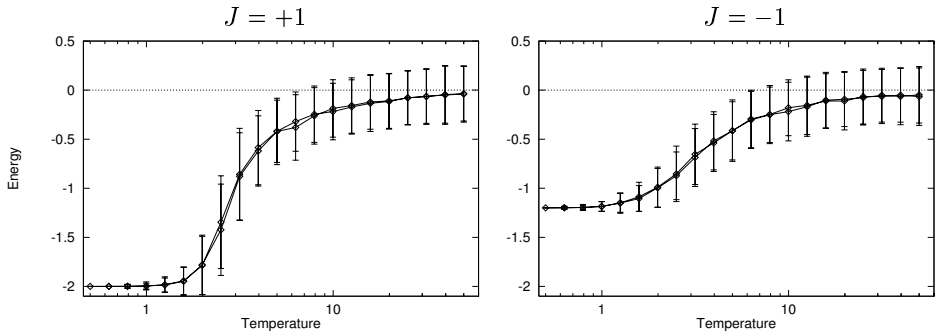
The spins in the 2D lattice form a bi-partite graph W, B.

$$E_{\text{anti}}(x_B, x_W) = -\sum_{i \in B} \sum_{j \in W} (J = -1) x_i x_j = -\sum_{i \in B} \sum_{j \in W} (J = 1) x_i (-x_j)$$
$$= E_{\text{ferro}}(x_B, -x_W)$$

The ferro and anti-ferro magnetic system have the same energies $E(x)$ by relabeling x. And thus the same physics.

Anti-ferromagnet

When the number of spins per direction is odd, this is no longer true.

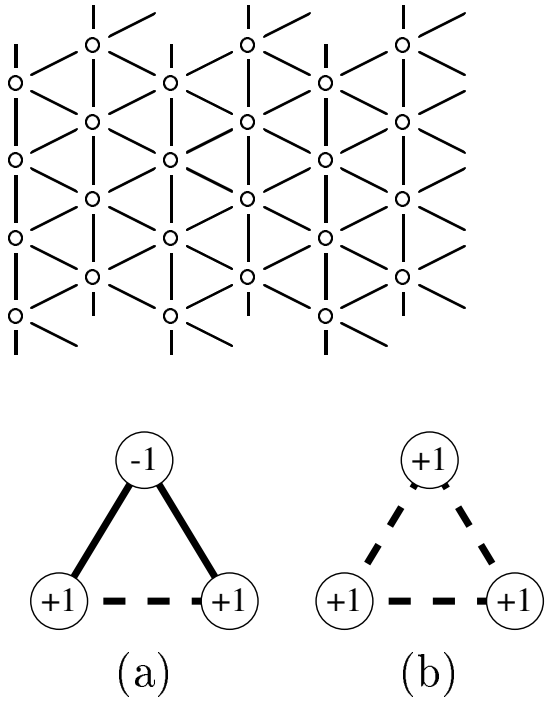


Rectangular grid of 5×5 spins with ferromagnetic (left) and anti-ferromagnetic (right) interaction.

Not all bonds $Jx_i x_j$ can be minimized. This is a boundary effect that decreases with system size.

Frustration: (anti-)ferromagnetic triangular Ising model

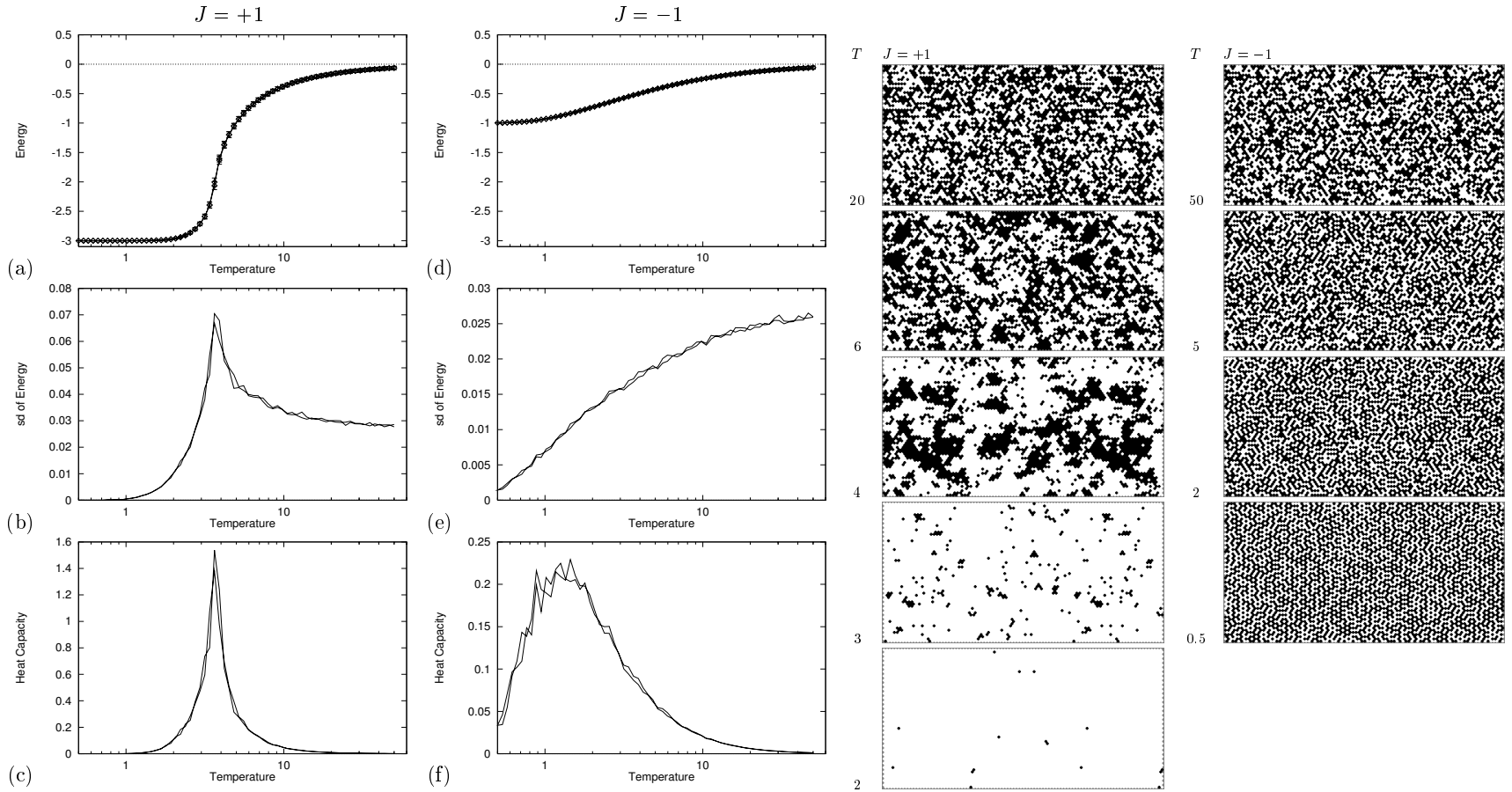
When $J = -1$, $E(x)$ cannot be minimized by minimizing all link energies $-Jx_ix_j = x_ix_j$.



Of the eight possible configurations of three spins, six have energy -1 and two have energy $+3$.

As a result: no 'clear cut' unique minimal energy state(s) but many states with similar low energy (Spin glass).

Frustration: Anti-ferromagnetic triangular Ising model



Energy (top), fluctuations (middle), heat capacity (bottom) for ferromagnet $J = 1$ (left) and anti-ferromagnet ($J = -1$) on triangular latice.

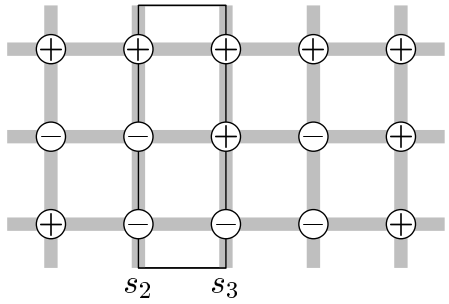
Note: different low energies due to frustration, absence of peak in $\mathbb{V}(E)$ indicates no phase transition to state with long range order.

Transfer matrix method for computation of partition function

The summation of the partition sum Z can be performed in the 2-dimensional Ising model on a lattice of size $W \times C$. The lattice is periodic in the C direction.

$$Z = \sum_x \exp(-\beta E(x)) \qquad E(x) = -\sum_{i>j} J x_i x_j$$

$$= \sum_{s_1} \sum_{s_2} \cdots \sum_{s_C} \exp\left(-\beta \sum_{c=1}^{C} E(s_c, s_{c+1})\right)$$



s_c is a binary vector of length W, $\sum_{s_c}$ contains 2^W terms. [9]

$$Z = \sum_{s_1} \sum_{s_2} \cdots \sum_{s_C} M_{s_1,s_2} M_{s_2,s_3}, \ldots, M_{s_{C-1},s_C} M_{s_C,s_1} = \sum_{s_1} \left(M^C\right)_{s_1,s_1}$$

$$= \text{Trace}(M^C) \qquad M_{s_c,s_{c+1}} = \exp(-\beta E(s_c, s_{c+1}))$$

[9] The energy is

$$E(s_c, s_{c+1}) = -\sum_{i \in c, j \in c+1} J x_i x_j - \frac{1}{4} \sum_{i,j \in c} J x_i x_j - \frac{1}{4} \sum_{i,j \in c+1} J x_i x_j$$

M is $2^W \times 2^W$ matrix.

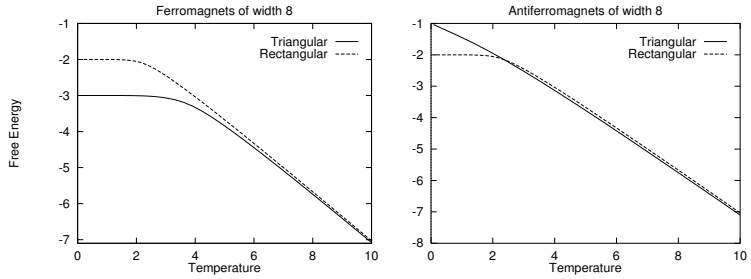
Transfer matrix method for computation of partition function

$$\lim_{C\to\infty} Z = \lim_{C\to\infty} \text{Trace}(M^C) = \lim_{C\to\infty} \sum_\alpha \mu_\alpha^C = \mu_{\max}^C$$

with $\mu_{\max}$ the largest eigenvalue of M.

Free energy per spin

$$f = -\frac{T}{WC}\log Z = -\frac{T}{W}\log \mu_{\max}$$

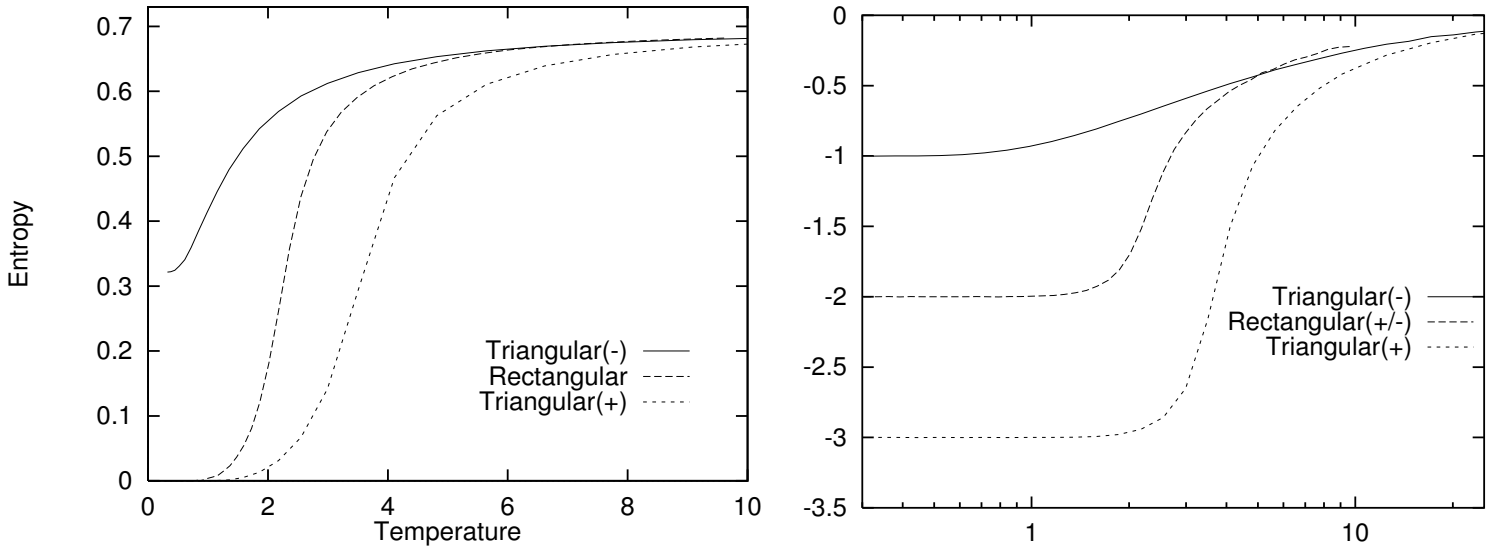


Free energy per spin $f = \langle E\rangle - TH$. At high temperature $\langle E\rangle \approx 0$ and $f \approx -TH \approx -T\log 2$. Entropy $H = -\frac{\partial F}{\partial T}$, (Ex. 31.1) at low temperature (slope), is zero for ferro and rect. anti-ferro, and non-zero for triang. anti-ferro.

Transfer matrix method for computation of partition function

Entropy and average energy is computed by finite differencing:

$$H = -\frac{dF}{dT} \approx \frac{F(T + dT) - F(T)}{dT} \qquad \langle E \rangle = F - TH$$

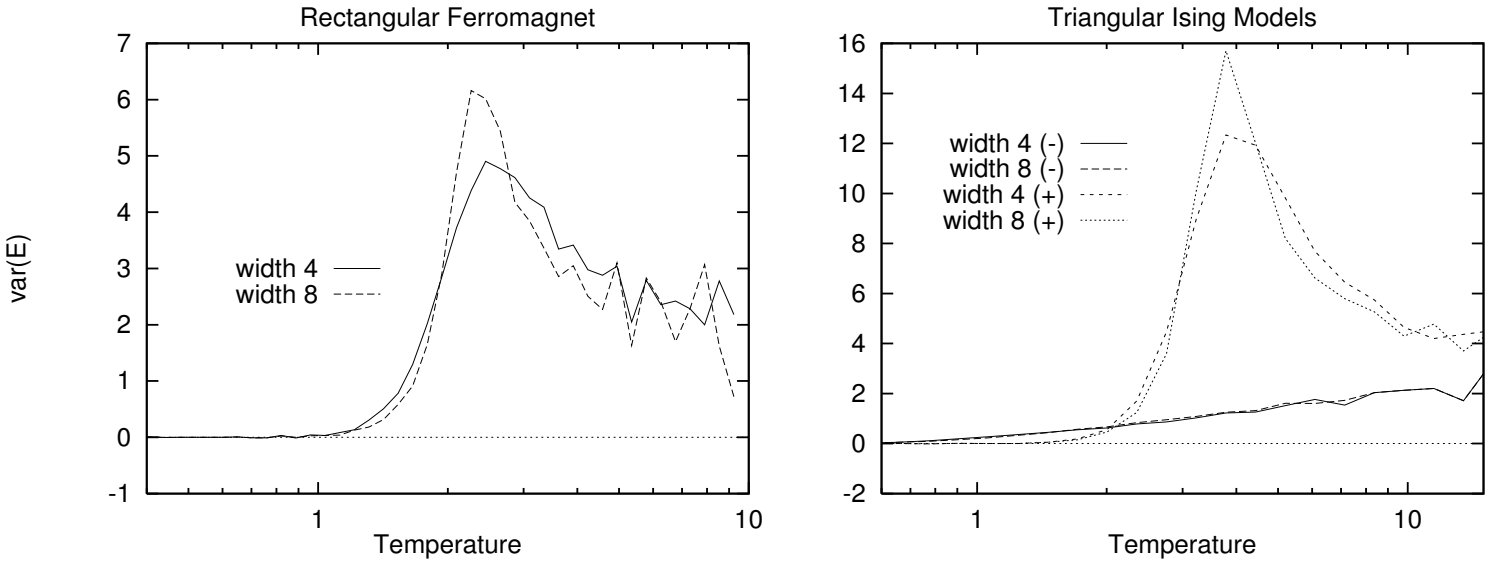


Entropy of frustrated system (-) at low temperature is non-zero. Mean energy at low temperature is higher for frustrated (-) than for non-frustrated (+) system

Direct computation of partition function

Energy fluctuation is computed by finite differencing of the mean energy

$$\mathbb{V}E = \frac{d^2 \log Z}{d\beta^2} = -\frac{d\langle E \rangle}{d\beta}$$



$\mathbb{V}E = \frac{d^2 \log Z}{d\beta^2}$ versus temperature shows signs of phase transition for rectangular and triangular ferromagnet (+), but not for the frustrated triangular anti-ferromagnet (-).

Iterative improvement

Consider an discrete optimization problem [10] to minimize a function $E(x)$:

$$\min_x E(x)$$

and denote the optimal value by x^*.

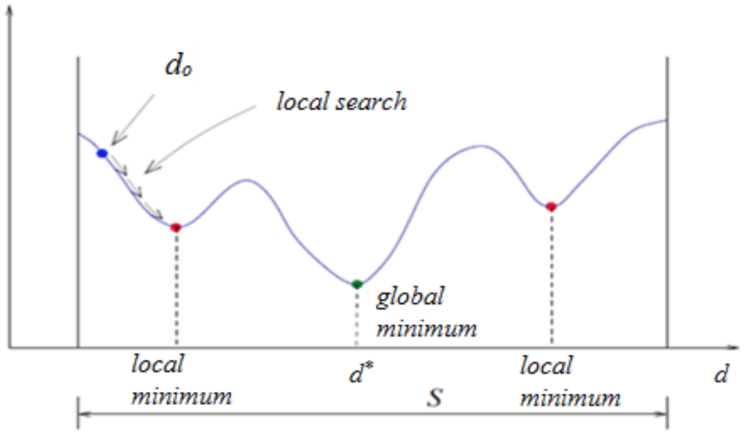
Iterative improvement. For any state, define a neighborhood $R(x)$ as the set of states x' that can be reached from x. The neighborhood can be anything, for instance single spin flips or multiple spin flips.

1: Start with a random initial state x_0
2: **for** $t = 1, 2, \ldots$ **do**
3: Sample a state $x' \in R(x_t)$
4: **if** $E(x') < E(x_t)$ **then**
5: accept $x_{t+1} = x'$
6: **else**
7: reject $x_{t+1} = x_t$
8: **end if**
9: **end for**

[10] Also referred to as a combinatoric optimization problem.

Iterative improvement

Iterative improvement is a so-called local search algorithm. These methods suffer from local minima, in the sense that the algorithm converges to a configuration $x_\infty \neq x^*$ that is locally optimal, but not globally $E(x_\infty) > E(x^*)$.



The depth $d(x)$ of state x is defined as the lowest barrier to reach x^*. Define a path $\tau = (x_0, x_1, \ldots, x_p)$ with $x_0 = x$ and $x_p = x^*$ such that x_i is in the neighborhood of $x_{i-1}, i = 1, \ldots, p$. The depth of states x is

$$d(x) = \min_\tau \tilde{d}(\tau)$$

with $\tilde{d}(\tau) = \max_{x_i} E(x_i) - E(x)$ the depth of the path τ.

Simulated annealing

Simulated annealing approximates the optimization problem by a sequence of sampling problems:

$$\min_x E(x) \quad \rightarrow p_k(x) = \frac{1}{Z} e^{-E(x)/T_k}$$

For each k one can define a MH algorithm that samples from p_k.

For decreasing T_k, the distributions $p_k(x)$ become more and more centered on x^* and

$$\lim_{T_k \to 0} p_k(x) = \delta_{x,x^*}$$

The analogy of a physical system freezing into a low temperature ordered state.

Simulated annealing

Instead of a sequence of homogeneous Markov processes, one can also define simulated annealing as a single inhomogeneous Markov process where T_t decreases in each iteration t, as follows.

Start with a random initial state x_0
for $t = 1, 2, \ldots$ **do**
    Sample a state $x' \in R(x_t)$
    Compute $a = \exp\left(-\frac{E(x')-E(x)}{T_t}\right)$ ▷ MH acceptance probability
    if $a > 1$ **then**
        accept $x_{t+1} = x'$
    else
        $x_{t+1} = x'$ w.p a and $x_{t+1} = x_t$ w.p. $1 - a$
    end if
end for

The sequence $T_t, t = 1, 2, \ldots$ is called the annealing schedule.

Simulated annealing [Aarts and Korst, 1988]

The dynamics converges $x \rightarrow x^*$ [Hajek, 1988] provided that the process is ergodic and

$$T_t = \frac{T_1}{\log_2(t+1)} \qquad \beta_t = \beta_1 \log_2(t+1) \qquad T_1 \geq D \qquad D = \max_x d(x)$$

So the initial temperature must be larger than the deepest local minimum D.

The annealing is very slow. The number of iterations t to reach a final temperature T_f is

$$t = e^{\frac{T_1}{T_f}}$$

and can be (unacceptable) large.

Simulated annealing

In practice, one uses a sequence of homogeneous Markov processes (chains) of length L and indexed by k.

Schedule
A simple choice is an exponential schedule $\beta_{k+1} = f\beta_k$. [11]

An alternative choice, proposed in [Aarts and Korst, 1988], is $\beta_{k+1} = \beta_k + \frac{\Delta\beta}{\sqrt{\mathbb{V}_k E}}$ with $\mathbb{V}_k E$ the variance of the energy in chain k.

Estimate the initial temperature by doing an initial exploration of states x and their energies $E(x)$ using the MH method with (too) high temperature. Set the initial temperature as $T_1 = \max(\Delta E)$, where $\Delta E = E(x') - E(x)$ when $x \to x'$. As a result,

$$a = \frac{p^*(x')}{p^*(x)} = e^{-\Delta E/T_1} \geq e^{-1}$$

at the start of the algorithm.

[11] $\beta_k = 1/T_k$ is the inverse temperature.

Simulated annealing

Estimate for each chain k the mean energy $\langle E \rangle_k$ and its standard deviation $\mathbb{V}_k E$. The algorithm terminates when $\mathbb{V}_k E = 0$ and the estimate of the minimal cost is $\langle E \rangle_k$.

1: Run MH at high temperature to estimate β_1
2: **while** $\mathbb{V} E_k > 0$ **do**
3: $\quad \beta_{k+1} = f\beta_k$ or $\beta_{k+1} = \beta_k + \frac{\Delta\beta}{\sqrt{\mathbb{V}_k E}}$
4: $\quad$ Run MH at temperature β_k. Compute $\langle E \rangle_k$ and $\mathbb{V}_k E$.
5: **end while**

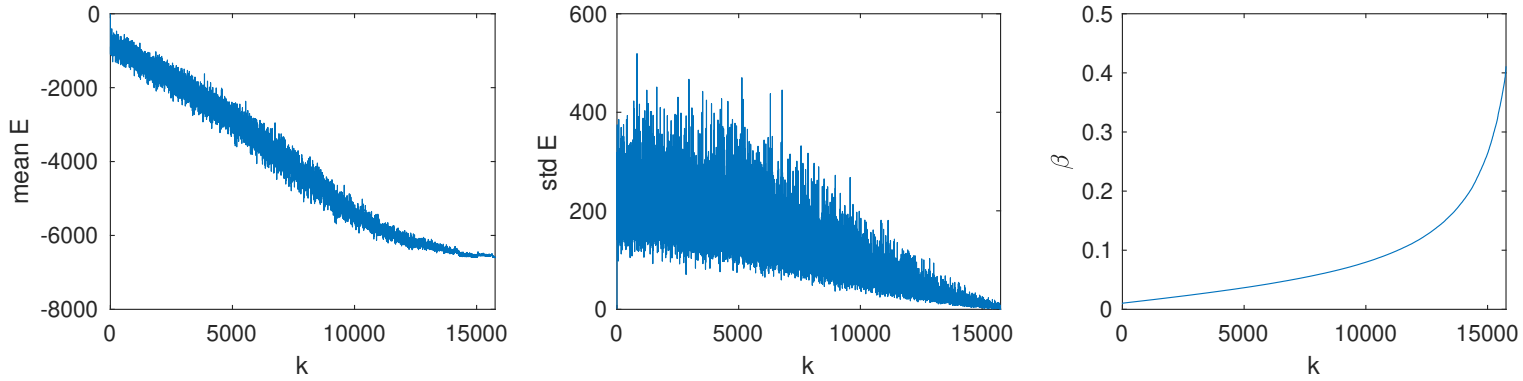
Choose $f = 1.01$ or similar.
Choose L a multiple of the neighborhood size $|R|$.
Choose $\Delta\beta$ in the range $0.001, 0.01, 0.1$

Exercise Simulated annealing

Write your own SA algorithm to miminize $E(x) = -\frac{1}{2}x'wx = -\sum_{(ij)} x_i x_j w_{ij}$.

If $w_{ij} > 0$ random, optimization is easy (ferromagnetic). There are two global minima.

If w_{ij} has both signs, optimization is hard (spin glass). There are many minima of approx equal quality.



SA on frustrated `w500` problem. AK annealing schedule with $\Delta\beta = 0.001$, chain length $L = 1000$. A solution $E = -6600$ is found in 20 sec. using 15758 chains. Left: mean E in chain k versus k. Middle: standard deviation of E in chain k versus k. Right: β_k versus k.

Lecture 4: Deterministic approximations of the Bayesian posterior

- Laplace approximation (the posterior for the perceptron)
- Variational approximation (the posterior for the Gaussian)
- Variational approximation for multi layer neural network

Laplace approximation

The Laplace approximation approximates a given a probability density $p(x)$ by a Gaussian distribution centered on the maximum $x^* = \text{argmax}_x p(x)$.

Without loss of generality we can write $p(x) = \frac{1}{Z} e^{-E(x)}$. Expand E around x^* to second order:

$$E(x) \approx E(x^*) + \sum_i (x_i - x_i^*) \left.\frac{\partial E(x)}{\partial x_i}\right|_{x=x^*} + \frac{1}{2} \sum_{ij} (x_i - x_i^*)(x_j - x_j^*) \left.\frac{\partial^2 E}{\partial x_i x_j}\right|_{x=x^*}$$

Note, that $\left.\frac{\partial E(x)}{\partial x_i}\right|_{x=x^*} = 0$ and define $H_{ij} = \left.\frac{\partial^2 E}{\partial x_i x_j}\right|_{x=x^*}$. Then [12]

$$p(x) \approx \frac{1}{Z} e^{-E(x^*)} e^{-\frac{1}{2}(x-x^*)^T H (x-x^*)} = \frac{1}{\sqrt{\det(2\pi H^{-1})}} e^{-\frac{1}{2}(x-x^*)^T H (x-x^*)}$$

The Laplace approximation is not necessarily the 'best' approximation (for instance in terms of KL divergence), but is easy to compute.

[12] $Z = \int dx e^{-E(x)} \approx e^{-E(x^*)} \int e^{-\frac{1}{2}(x-x^*)^T H (x-x^*)} = e^{-E(x^*)} \sqrt{\det(2\pi H^{-1})}$

Laplace approximation for Bayesian posterior

Consider the Bayesian learning of the perceptron discussed in MK 39 and MK 41:[13]

$$p(D|w) = \exp(-G(w)) \qquad G(w) = -\sum_{\mu=1}^{N} \log(p(t^\mu|x^\mu, w))$$

$$p(w|\alpha) \propto \frac{\exp(-\frac{1}{2}\alpha \sum_i w_i^2)}{Z_w(\alpha)}$$

$$p(w|D,\alpha) = \frac{p(D|w)p(w)}{p(D)} = \frac{1}{Z}\exp(-M(w)) \qquad M(w) = G(w) + \frac{1}{2}\alpha \sum_i w_i^2$$

We wish to compute the Bayesian prediction the probability of class $t = 0, 1$ for a new data point

$$p(t|x, D, \alpha) = \int dw p(t|x, w)p(w|D, \alpha)$$

[13] We use $t = 0, 1$ and $p(t = 1|x, w) = \sigma(w \cdot x)$ with $\sigma(x) = 1/(1 + e^{-x})$.

Laplace approximation for Bayesian posterior

We use the Laplace approximation to the posterior $p(w|D,\alpha) \approx \mathcal{N}(w|w^*, H^{-1})$ as a Gaussian centered on the maximum posterior solution $w^* = \text{argmin}_w M(w)$. w^* can be found by gradient descend.

The Hessian is

$$H_{ij} = \left.\frac{\partial^2 M}{\partial w_i w_j}\right|_{w=w^*} = \sum_{\mu=1}^{N} \sigma(h^\mu)\sigma(-h^\mu)) x_i^\mu x_j^\mu + \alpha\delta_{ij} \qquad h^\mu = \sum_{i=0}^{d} w_i^* x_i^\mu$$

H^{-1} is the covariance matrix of the posterior on w in the Gaussian approximation, which we can interpret as error bars on w. With more data ($N \to \infty$) the variance in w goes to zero ($H^{-1} \to 0$).

Thus,

$$p(t=1|x,D,\alpha) \approx \int dw p(t=1|x,w)\mathcal{N}(w|w^*, H^{-1})$$

The function $p(t=1|x,w) = \sigma(a)$ with $a = w \cdot x$. Since w is Gaussian and x is fixed,

a is also Gaussian and

$$\mathbb{E}(a) = w^* \cdot x = a^* \qquad \mathbb{V}(a) = x^T \mathbb{V}(w) x = x^T H^{-1} x = s^2$$

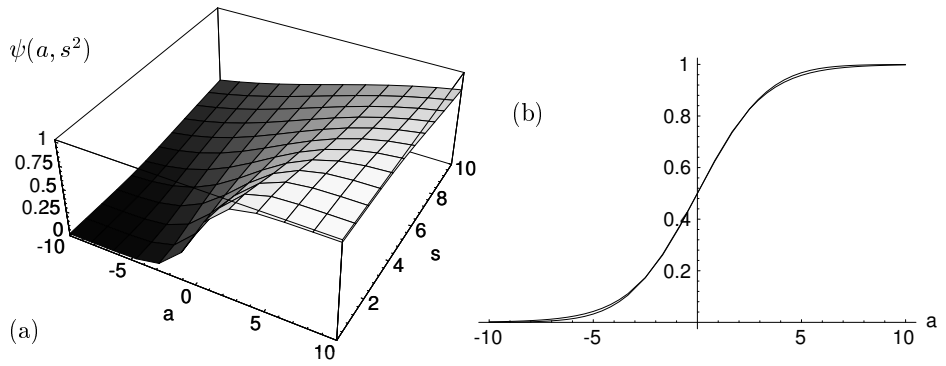
Laplace approximation for Bayesian posterior

Thus, the d-dimensional integral is reduced to a 1-dimensional integral

$$p(t = 1|x, D, \alpha) \approx \int da\sigma(a)\mathcal{N}(a|a^*, s^2)$$

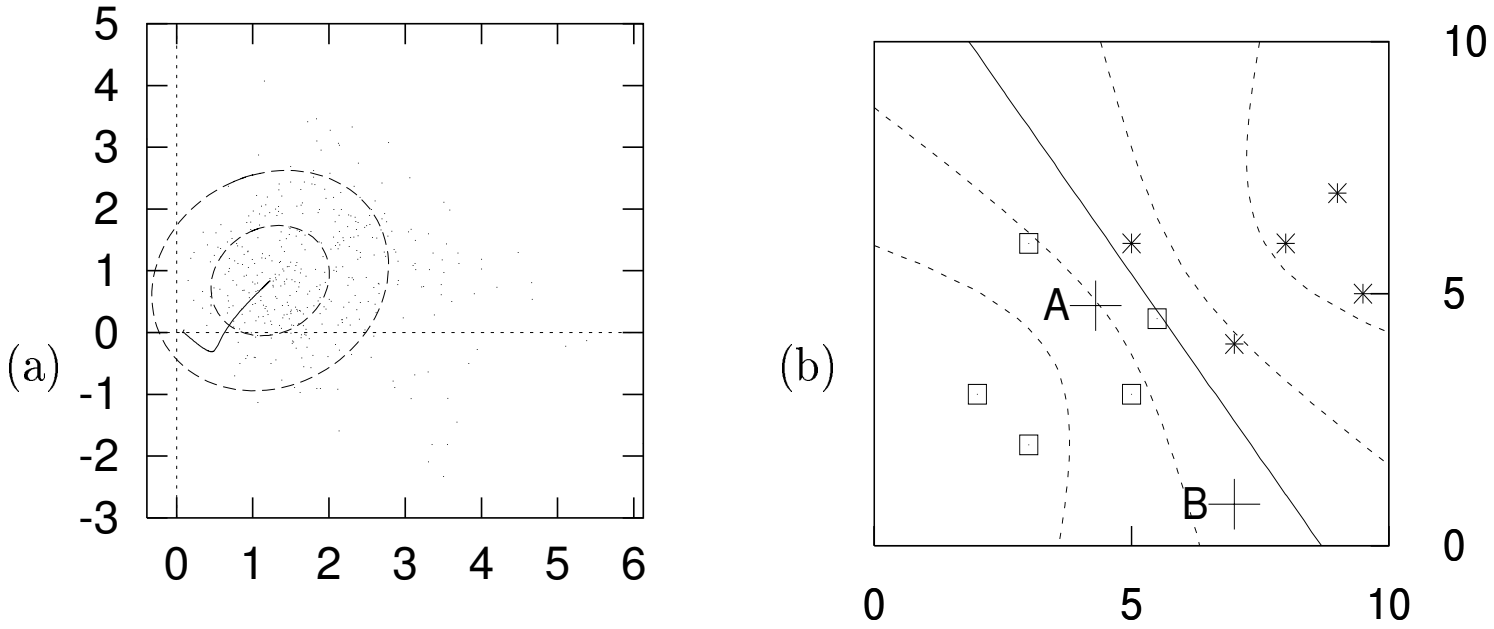
The Gaussian integral of a sigmoid function can be approximated quite well as

$$\psi(a^*, s^2) = \int da\sigma(a)\mathcal{N}(a|a^*, s^2) \approx \sigma(\kappa(s)a^*) \qquad \kappa(s) = 1/\sqrt{1 + \pi s^2/8}$$



Left: $\psi(a^*, s^2)$. Right: $\psi(a^*, s^2)$ and $\sigma(\kappa(2)a^*)$ versus a^* for $s^2 = 4$.

Laplace approximation for Bayesian posterior



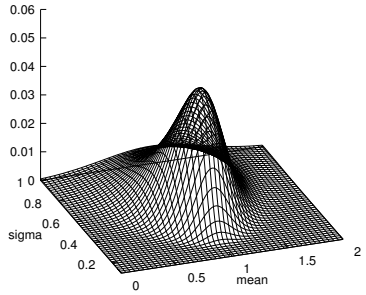
a) A projection of the Gaussian approximation onto the (w_1, w_2) plane of weight space. The one- and two-standard-deviation contours are shown. Also shown are the trajectory of the optimizer, and the Monte Carlo method samples. (b) The predictive function obtained from the Gaussian approximation and the further approximation for the sigmoid integral (41.30).

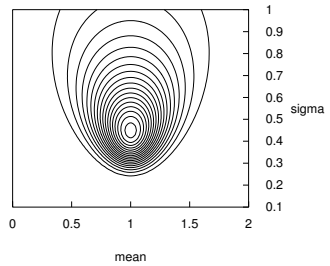
Bayesian posterior for Gaussian

Hypothesis space for a Gaussian distribution in 1 dimension:

$$p(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Give some data, we can evaluate $p(\{x_n\}_{n=1}^N|\mu,\sigma)$





Bayesian posterior for Gaussian: ML solution

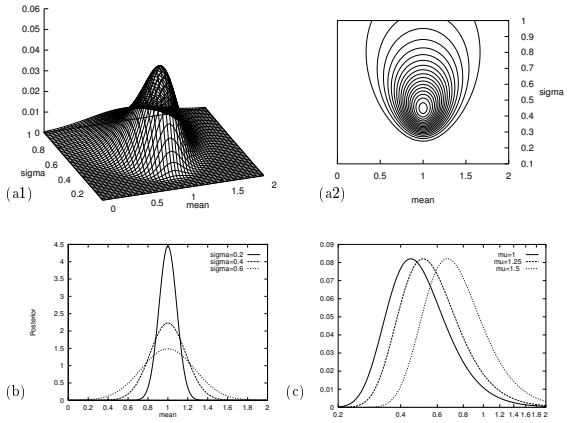
The log likelihood is

$$L = \log p(\{x_n\}_{n=1}^N | \mu, \sigma) = -N \log\left(\sqrt{2\pi}\sigma\right) - \frac{1}{2\sigma^2} \sum_n (x_n - \mu)^2$$

The maximum likelihood solution is computed from $\frac{\partial L}{\partial \mu} = \frac{\partial L}{\partial \sigma} = 0$:

$$\mu = \frac{1}{N} \sum_n x_n \qquad \sigma^2 = \frac{1}{N} \sum_n (x_n - \mu)^2$$

μ independent of σ, but σ depends on μ.



Bayesian posterior for Gaussian: Variational solution

Given the Gaussian model, flat prior on μ and the non-informative prior on the variance ($\beta = \frac{1}{\sigma^2}$):[14]

$$p(x|\mu,\beta) = \sqrt{\frac{\beta}{2\pi}}\exp\left(-\frac{1}{2}\beta(x-\mu)^2\right) \qquad p(\mu) \propto 1 \qquad p(\beta) \propto \beta^{-1}$$

The posterior for N data points $D = \{x_n\}_{n=1}^N$ is

$$p(\mu,\beta|D) \propto \beta^{N/2-1}\exp\left(-\frac{1}{2}\beta\left(N(\mu-\bar{x})^2 + S\right)\right)$$

with $\bar{x} = \frac{1}{N}\sum_n x_n$, $S = \sum_n (x_n - \bar{x})^2$.

[14] The non-informative prior is defined in the log domain. Define $x = \log\sigma$ then

$$p_x(x) = p_\sigma(\sigma)\left|\frac{d\sigma}{dx}\right|$$

The non-informative prior assumes $p_x(x)$ uniform. Then, since $\left|\frac{dx}{d\sigma}\right| = \sigma^{-1}$ we obtain $p_\sigma(\sigma) \propto \sigma^{-1}$. In terms of β, we get $p_\beta(\beta) \propto \beta^{-1}$.

Bayesian posterior for Gaussian: Variational solution

We approximate the posterior $p(\mu, \beta|D)$ by a factorized variational distribution $q(\mu, \beta) = q_\mu(\mu)q_\beta(\beta)$.

The variational approximation is computed by minimizing

$$\begin{aligned} KL(q|p) &= \int d\mu d\beta q(\mu, \beta) \log \frac{q(\mu, \beta)}{p(\mu, \beta|D)} \\ &= \int d\mu q_\mu(\mu) \log q_\mu(\mu) + \int d\beta q_\beta(\beta) \log q_\beta(\beta) \\ &\quad - \int d\mu d\sigma q_\mu(\mu) q_\beta(\beta) \log p(\mu, \beta|D) \end{aligned}$$

Bayesian posterior for Gaussian: Variational solution

We optimize with respect to $q_\mu(\mu)$ subject to normalization constraint:

$$\begin{aligned}
\mathcal{L} &= KL(q|p) + \lambda_1\left(\int d\mu q_\mu(\mu) - 1\right) + \lambda_2\left(\int d\beta q_\beta(\beta) - 1\right) \\
\frac{\partial \mathcal{L}}{\partial q_\mu(\mu)} &= \log q_\mu(\mu) + 1 - \int d\beta q_\beta(\beta) \log p(\mu, \beta|D) + \lambda_1 \\
&= \log q_\mu(\mu) + \int d\beta q_\beta(\beta) \frac{1}{2}\beta N(\mu - \bar{x})^2 + \text{const.} \\
&= \log q_\mu(\mu) + \frac{1}{2}\bar{\beta} N(\mu - \bar{x})^2 + \text{const.} = 0
\end{aligned}$$

with $\bar{\beta} = \int d\beta q_\beta(\beta)\beta$ and the constant term contains all terms independent of μ. Thus, the solution q_μ is a Gaussian distribution with mean $\bar{x}$ and variance $(N\bar{\beta})^{-1}$.

NB: q_μ depends on q_β which we still have to determine!

Bayesian posterior for Gaussian: Variational solution

We optimize with respect to $q_\beta(\beta)$ subject to normalization constraint:

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial q_\beta(\beta)} &= \log q_\beta(\beta) + 1 - \int d\mu q_\mu(\mu) \log p(\mu, \beta|D) + \lambda_2 \\
&= \log q_\beta(\beta) + 1 - \int d\mu q_\mu(\mu) \left(\log \beta^{\frac{N}{2}-1} - \frac{1}{2}\beta \left(N(\mu - \bar{x})^2 + S \right) \right) + \lambda_2 \\
&= \log q_\beta(\beta) - \log \beta^{\frac{N}{2}-1} + \frac{1}{2}\beta \left(\frac{1}{\bar{\beta}} + S \right) + \text{const.} \\
&= \log q_\beta(\beta) - \log \left(\beta^{c-1} e^{-\beta/s} \right) + \text{const.}
\end{aligned}$$

with the constant term contains all terms independent of β. In the second step we used $\int d\mu q_\mu(\mu)(\mu - \bar{x})^2 = (N\bar{\beta})^{-1}$ and we defined $c = \frac{N}{2}, \frac{1}{s} = \frac{1}{2}\left(\frac{1}{\bar{\beta}} + S\right)$.

Thus, $q_\beta(\beta) = \Gamma(\beta|c, s)$ the Gamma distribution with parameters c, s. [15]

[15] The Gamma distribution is defined for non-negative x as

$$\Gamma(x|c, s) = \frac{1}{Z}\left(\frac{x}{s}\right)^{c-1} \exp\left(-\frac{x}{s}\right) \qquad Z = \Gamma(c)s$$

Bayesian posterior for Gaussian: Variational solution

NB: q_β depends on $\bar{\beta}$ which is given in terms of q_β.

We can thus solve for $\bar{\beta}$ self-consistently.

$$\bar{\beta} = \int d\beta \Gamma(\beta|c,s)\beta = cs = \frac{N}{\frac{1}{\bar{\beta}} + S}$$

which has solution $\bar{\beta} = \frac{N-1}{S}$.

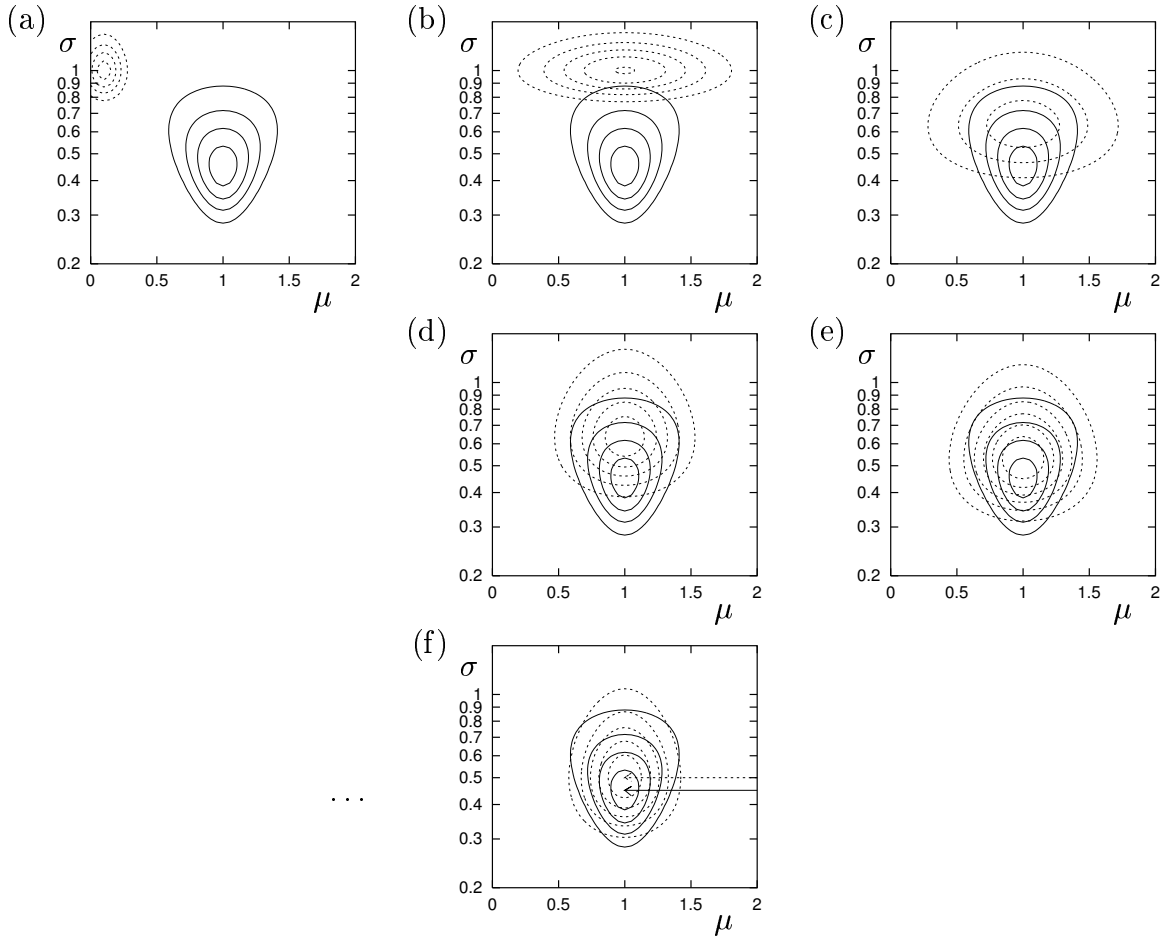
The final solution is

$$\begin{aligned} q_\mu(\mu) &= \mathcal{N}\left(\mu|\bar{x}, \frac{S}{N(N-1)}\right) \\ q_\beta(\beta) &= \Gamma(\beta|c,s) \qquad c = \frac{N}{2} \quad \frac{1}{s} = \frac{NS}{2(N-1)} \end{aligned}$$

Note, that the variational approximation has mean $\bar{\mu} = \bar{x}$ and $\bar{\sigma^2} = \frac{S}{N-1}$.

Bayesian posterior for Gaussian: Variational solution

Solid: $p(\mu, \sigma|D)$. Dashed: Iterative optimization of q_μ (b,d) and q_σ (c,e) from initial (a) converges in 15 iterations (f)



The solution can also be obtained numerically by optimizing for q_μ for fixed q_β and vise versa.

Variational multi-layered perceptron

[Barber and Bishop, 1998] develops the variational approximation for the Bayesian posterior of multi-layered perceptron using a multi-variate Gaussian distribution.

They consider a two-layer neural network with n inputs and one output

$$f(x,w) = \sum_{i=1}^{H} v_i \sigma(\sum_{j=1}^{n} u_{ij} x_j) \qquad w = \{v_i, u_{ij}\}$$

$$p(t|x,w) \propto \exp\left(-\frac{1}{2}(f(x,w)-t)^2\right)$$

The likelihood and prior

$$p(D|w) = \exp\left(-\frac{1}{2}\sum_{\mu}(f(x^\mu,w)-t^\mu)^2\right) \qquad p(w) \propto \exp\left(-\frac{1}{2}w'Aw\right)$$

with A a symmetric matrix.

Variational multi-layered perceptron

The posterior $p(w|D)$ is approximated using the variational approximation with a multivariate Gaussian distribution $q(w)$

$$\begin{aligned} KL(q|p) &= \int dw q(w) \log q(w) - \int dw q(w) \log p(w|D) \\ &= \int dw q(w) \log q(w) + \frac{1}{2} \sum_{\mu} \int dw q(w) \left(f(x^{\mu}, w) - t^{\mu}\right)^2 \\ &\quad + \frac{1}{2} \int dw q(w) w' A w \end{aligned}$$

The entropy and quadratic term are easy to compute. The data term involve terms as

$$\int dw q(w) f(x, w)^2 = \int dw q(w) \sum_{ii'} v_i v_{i'} \sigma\left(\sum_j u_{ij} x_j\right) \sigma\left(\sum_{j'} u_{i'j'} x_{j'}\right)$$

These can be reduced to 1-dimensional Gaussian integrals as we have seen in the Laplace approximation.

Entropy of multivariate Gaussian

D dimensional Gaussian with mean m_i and covariance matrix Σ_{ij}

$$p(x|m,\Sigma) = (2\pi)^{-D/2}|\Sigma|^{-1/2}\exp\left(-\frac{1}{2}(x-m)'\Sigma^{-1}(x-m)\right)$$

Entropy

$$\begin{aligned} H &= \int d^D x p(x|m,\Sigma)\log p(x|m,\Sigma) \\ &= -\log\left((2\pi)^{-D/2}|\Sigma|^{-1/2}\right) + \frac{1}{2}\sum_{i,j}^{D}\Sigma_{ij}^{-1}\int d^D x p(x|m,\Sigma)(x-m)_i(x-m)_j \\ &= -\log\left((2\pi)^{-D/2}|\Sigma|^{-1/2}\right) + \frac{D}{2} \end{aligned}$$

where we used $\int d^D x p(x|m,\Sigma)(x-m)_i(x-m)_j = \Sigma_{ij}$ and $|\Sigma|$ is the determinant of Σ.

Other terms

The quadratic term

$$\begin{aligned}\int dw q(w|m,C) w'Aw &= \int dw q(w|m,C) \left((w-m)'A(w-m) + 2m'Aw - m'Am\right) \\ &= \mathrm{Tr}(CA) + m'Am\end{aligned}$$

The data term contains terms of the form

$$\left\langle \left(c + d'w + \frac{1}{2}w'Aw\right) \sigma\left(a'w + a_0\right) \sigma\left(\left(b'w + b_0\right)\right) \right\rangle_{\mathcal{N}(w|0,I)}$$

These can be reduced to one dimensional integrals [Barber and Bishop, 1998].

The $KL(q|p)$ is optimized using gradient descend in the parameters of C, m of $q(w|C, m)$. Bayesian prediction for new data point x is approximated as

$$\langle f(x) \rangle = \int dw p(w|D) f(x, w) \approx \int dw \sum_{i=1}^{H} v_i \sigma \left(\sum_j u_{ij} x_j \right) q(w|C^*, m^*)$$

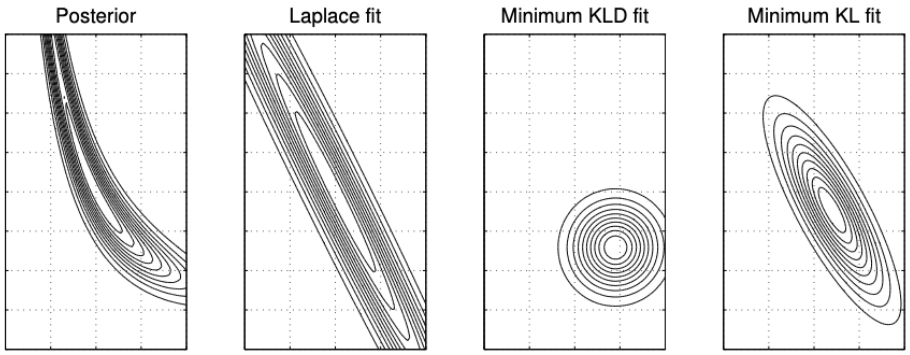


Figure 2: Comparison of various approximations to the posterior distribution for a synthetic regression problem involving two adaptive parameters (details given in the text). (a) The true posterior distribution. (b) The local Gaussian approximation obtained by Laplace's method, giving a Kullback-Leibler (KL) divergence value of 41. (c) The minimum KL fit obtained with a diagonal covariance Gaussian (KLD), giving a residual KL value of 4.6. (d) The minimum KL fit obtained using a full covariance Gaussian distribution, giving a residual KL value of 3.9.

Variational multi-layered perceptron

Application to Boston housing data. 13 inputs and one continuous output, 128 training samples, 250 test samples.

Covariance matrix of q is diagonal + rank one: $C = \text{diag}(\sigma_1^2, \ldots, \sigma_k^2) + \sum_{i=1}^{b} s_i s_i'$ with $s_i = (s_{i1}, \ldots, s_{ik})$. This choice reduces the number of free parameters in q from $k(k+1)/2$ to $(b+1)k$ with k the number of weights in the neural network.

Method	**Test Error**
Ensemble ($s = 1$)	0.22
Ensemble (diagonal)	0.28
Laplace	0.33

Deterministic approximation for spin models: Mean field approximation

- Mean field approximation
  - MF approximation for the 2d Ising model
  - relation of MF equation to stochastic dynamics
  - convergence of sequential updating
  - application of the MF approximation to the SK model
- Linear response method
- the TAP approximation

The (naive) mean field approximation for spin system

Consider $x = (x_1, \ldots, x_N)$ with $x_i = \pm 1$ and [16] [17]

$$p(x) = \frac{1}{Z} \exp\left(-E(x)\right) \qquad E(x) = -\sum_{(ij)} w_{ij} x_i x_j - \sum_i \theta_i x_i \qquad Z = \sum_x \exp\left(-E(x)\right)$$

Assume a factorized variational distribution: $q(x) = \prod_{i=1}^N q_i(x_i)$. q is found by minimizing

$$\begin{aligned} KL(q|p) &= \sum_x q(x) \log \frac{q(x)}{p(x)} = \sum_x q(x) \log q(x) + \sum_x q(x) E(x) - \log Z \\ \langle E \rangle_q &= -\sum_{(ij)} w_{ij} \left\langle x_i x_j \right\rangle_q - \sum_i h_i \langle x_i \rangle_q = -\sum_{(ij)} w_{ij} m_i m_j - \sum_i \theta_i m_i \end{aligned}$$

with $\langle x_i \rangle_q = m_i$.

[16] $\sum_{(ij)} = \frac{1}{2} \sum_{i=1}^N \sum_{j=1, j \neq i}^N$ denotes the sum over all pairs.

[17] We assume $\beta = 1$ as it can be absorbed in the couplings w_{ij}, θ_i.

The (naive) mean field approximation for spin system

We can express $q_i(x_i)$ in terms of m_i as $q_i(x_i) = \frac{1}{2}(1 + m_i x_i)$. Then

$$\begin{aligned} S &= -\sum_x q(x) \log q(x) = -\sum_x q(x) \sum_{i=1}^{n} \log q_i(x_i) = -\sum_{i=1}^{N} \sum_{x_i} q_i(x_i) \log q_i(x_i) \\ &= -\frac{1}{2} \sum_{i=1}^{N} \left((1+m_i) \log \frac{1}{2}(1+m_i) + (1-m_i) \log \frac{1}{2}(1-m_i) \right) \end{aligned}$$

18

$$\begin{aligned} KL(q|p) &= -S + \langle E \rangle_q = -\sum_{(ij)} w_{ij} m_i m_j - \sum_i \theta_i m_i \\ &\quad + \sum_i \frac{1}{2} \left((1+m_i) \log \frac{1}{2}(1+m_i) + (1-m_i) \log \frac{1}{2}(1-m_i) \right) \end{aligned}$$

[18] Where we used $\sum_x q(x) \log q_i(x_i) = \sum_{x_i} \sum_{x_{\setminus i}} q(x, x_{\setminus i}) \log q_i(x_i) = \sum_{x_i} q(x_i) \log q_i(x_i) = \sum_{x_i} q_i(x_i) \log q_i(x_i)$

We find the minimum by differentiation:

$$\frac{\partial KL(q|p)}{\partial m_i} = -\sum_{j\neq i} w_{ij} m_j - \theta_i + \frac{1}{2}\log\frac{1+m_i}{1-m_i} = 0$$

It is easy to show that $\frac{1}{2}\log\frac{1+m_i}{1-m_i} = a_i = \tanh^{-1}(m_i)$ [19] Thus,

$$m_i = \tanh\left(\sum_{j\neq i} w_{ij} m_j + \theta_i\right)$$

are the mean field equations.

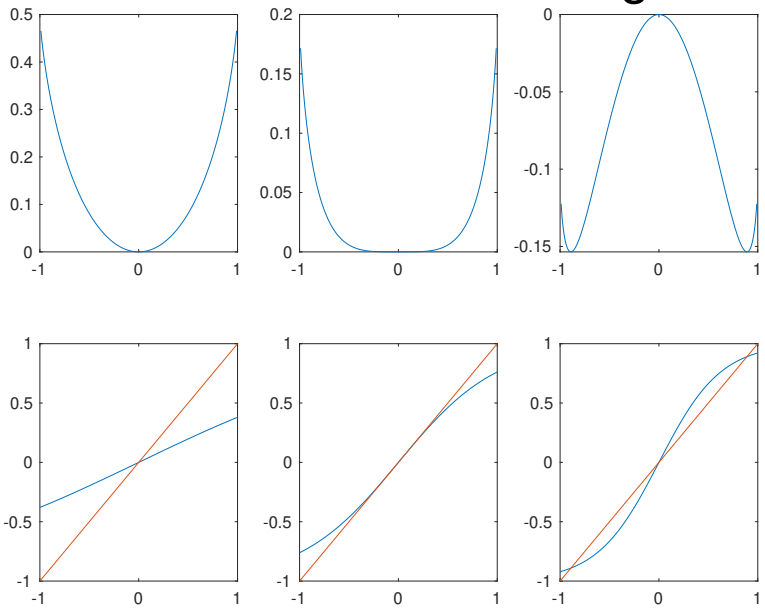
[19] Exercise: show this relation. Hint: we can define the distribution over a single bit $x = \pm 1$ as $q(x) = \frac{1}{2}(1 + mx)$ but also in the exponential form as $q(x) = \frac{e^{ax}}{e^{a}+e^{-a}}$. 1) show that $m = \tanh(a)$. Subsequently, show that $\tanh\left(\frac{1}{2}\log\frac{1+m}{1-m}\right) = m$, which proves the result.

MF approximation for the 2D Ising model

In the case of the 2d Ising model, all couplings $w_{ij} = w$ for neighbouring spins and zero otherwise and the external field $\theta_i = \theta$. Thus, $m_i = m$ and

$$m = \tanh(Cwm + \theta)$$

with C the number of neighbours in the lattice (4 for square lattice).

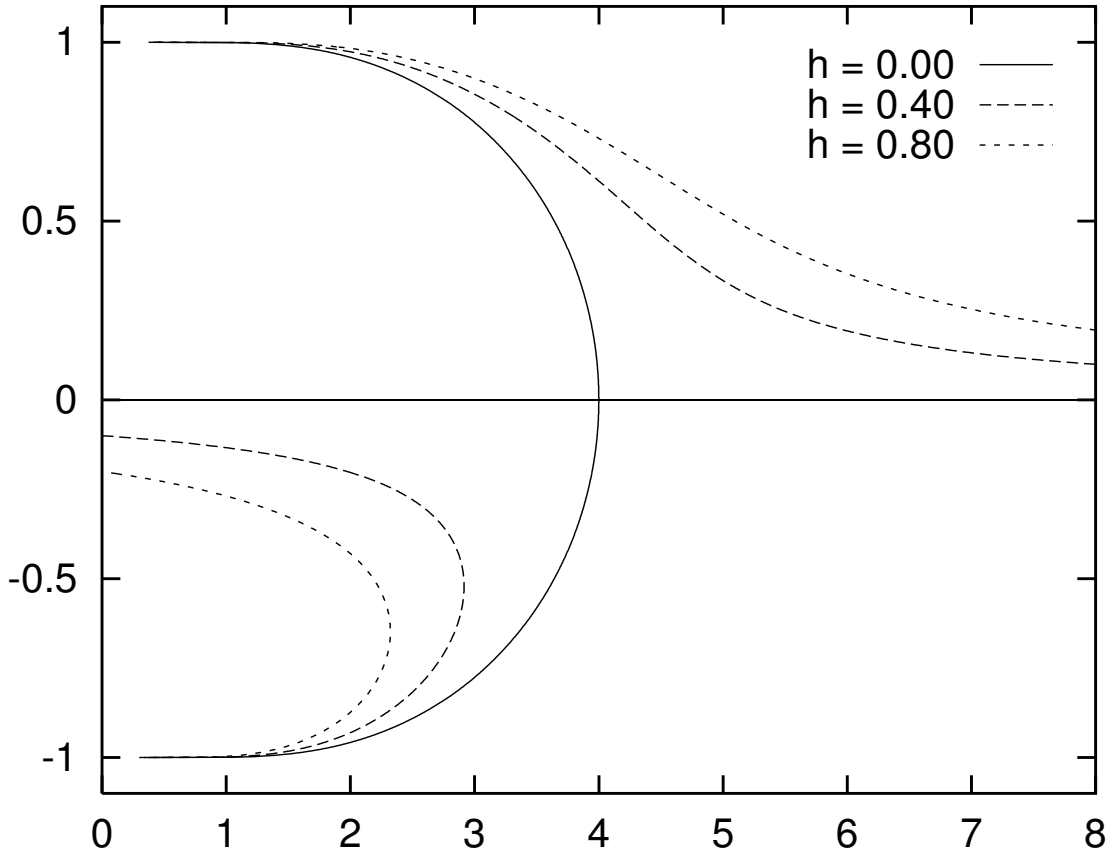


For $w < w_c$, KL is convex in m.
For $w > w_c$ KL is not convex in m (has 2 minima).

Top row: $KL(q|p)$ versus m for the 2d Ising model for $w = 0.1, 0.25, 0.4$. Bottom row: $\tanh(4wm)$ versus m for same values of w. w has a critical value $w_c = 0.25$ separating two phases.[20]

[20] In physics language, the coupling is $w = \frac{1}{T}$ with T the temperature. Then the MF estimate of $T_c = 4$. The exact value is $T_c = 2.27$.

MF approximation for the 2D Ising model



Relation of MF equation to Gibbs sampling

It is instructive to understand the relation between the mean field equations and the stochastic dynamics of the spin system, such as Gibbs sampling.

Remember that for Gibbs sampling we sample x_i from the conditional distribution $p(x_i|x_{\setminus i})$.

$$\begin{aligned} E(x) &= -\sum_{(ij)} w_{ij} x_i x_j = -x_i h_i(x_{\setminus i}) - \frac{1}{2} \sum_{k\neq i, l\neq i, k\neq l} w_{kl} x_k x_l \\ p(x_i|x_{\setminus i}) &= \frac{e^{-E(x_i, x_{\setminus i})}}{\sum_{x_i'} e^{-E(x_i', x_{\setminus i})}} = \frac{e^{x_i h_i(x_{\setminus i})}}{\sum_{x_i'} e^{x_i' h_i(x_{\setminus i})}} = \sigma\left(x_i h_i(x_{\setminus i})\right) \end{aligned}$$

with $h_i(x_{\setminus i}) = \sum_{j\neq i} w_{ij} x_j$.

We can define a Markov process by sampling in parallel [21] for all spins

$$T(x'|x) = \prod_{i=1}^{N} \sigma\left(x_i' h_i(x_{\setminus i})\right)$$

[21] The same result is obtained using sequential dynamics, where one spin at the time is updated.

Relation of MF equation to Gibbs sampling

The dynamical evolution of the distribution is $p_{t+1}(x') = \sum_x T(x'|x)p_t(x)$ and[22]

$$\langle x_i \rangle_{t+1} = \sum_{x'} x'_i p_{t+1}(x') = \sum_{x,x'} x'_i T(x'|x) p_t(x) = \left\langle \tanh\left(h_i(x_{\setminus i})\right) \right\rangle_t$$

Thus, the stationary condition $p_t \to p_\infty$ is

$$\langle x_i \rangle = \left\langle \tanh\left(h_i(x_{\setminus i})\right) \right\rangle \qquad h_i(x_{\setminus i}) = \sum_{j \neq i} w_{ij} x_j$$

This is an exact relation. It looks like the MF equations, except the average is outside the $\tanh$.

22

$$\sum_{x'} x'_i T(x'|x) = \sum_{x'_i} x'_i \sigma\left(x'_i h_i(x_{\setminus i})\right) \prod_{j \neq i} \sum_{x'_j} \sigma\left(x'_j h_j(x_{\setminus j})\right) = \sigma\left(h_i(x_{\setminus i})\right) - \sigma\left(-h_i(x_{\setminus i})\right) = \tanh(h_i(x_{\setminus i}))$$

Relation of MF equation to Gibbs sampling

In a ferromagnetic model with $w_{ij} = \frac{w^0_{ij}}{N}$ with $w^0_{ij} > 0$ the fluctuations in $h_i(x_{\setminus i})$ vanish when $N \to \infty$:

$$\langle h_i(x_{\setminus i}) \rangle = \frac{1}{N} \sum_{j \neq i} w^0_{ij} \langle x_j \rangle = \mathcal{O}(1) \qquad \mathbb{V}(h_i(x_{\setminus i})) = \frac{1}{N^2} \sum_{j \neq i} (w^0_{ij})^2 \mathbb{V}(x_j) = \mathcal{O}\left(\frac{1}{N}\right)$$

Thus, $h_i(x_{\setminus i}) \approx \langle h_i(x_{\setminus i}) \rangle$ in the large N limit and the mean field equations become exact:

$$m_i = \tanh(\langle h_i(x_{\setminus i}) \rangle) = \tanh\left(\sum_j w_{ij} m_j\right)$$

The key properties for this to hold are: $N \to \infty$, ferromagnetic couplings $w_{ij} > 0$ and each spin is coupled to $\mathcal{O}(N)$ other spins.

Convergence of sequential MF iterations

The mean field approximation is computed by minimizing [23]

$$KL(q|p) = \sum_i \sum_{x_i} q_i(x_i) \log q_i(x_i) - \sum_x \prod_i q_i(x_i) \log p(x)$$

The mean field equations are given by the coupled system of equations $\frac{\partial KL(q|p)}{\partial q_i(x_i)} = 0$.

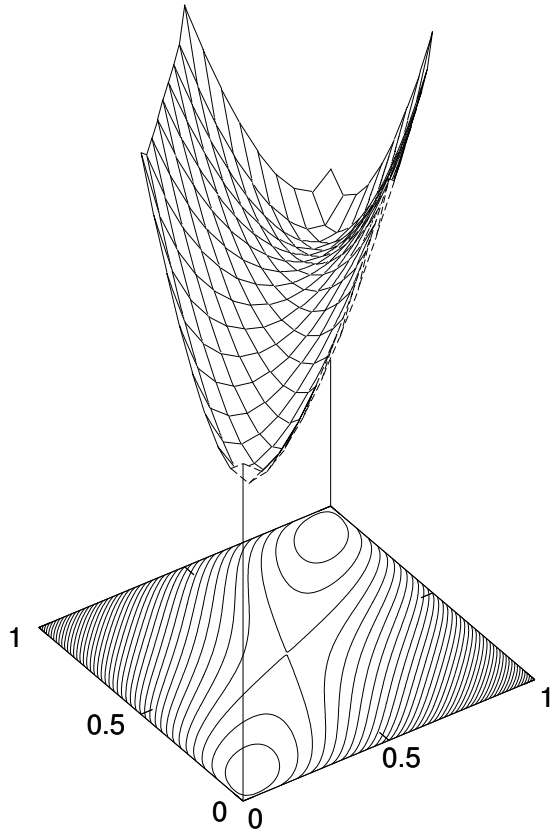
The idea of coordinate descent, aka sequential updating, is to update one of the terms of $q_i(x_i)$ at the time, while keeping the other terms $\prod_{j \neq i} q_j(x_j)$ fixed. Since the update rule results from setting the gradient $\frac{\partial KL(q|p)}{\partial q_i(x_i)} = 0$, updating $q_i(x_i)$ is equivalent to a minimization of $KL(q|p)$ with respect to $q_i(x_i)$ for fixed $q_j(x_j)$ and thus in each iteration

$$KL(q_i^{\text{new}} \prod_{j \neq i} q_j^{\text{old}} | p) \leq KL(q_i^{\text{old}} \prod_{j \neq i} q_j^{\text{old}} | p)$$

Thus, $KL(q|p)$ decreases in each coordinate descent iteration. Therefore, coordinate descent converges to a (local) minimum of $KL(q|p)$.

[23] Sums are replaced by integrals if x is continuous

(Non)-convexity of $KL(q|p)$



$KL(q|p)$ vs. $\frac{1}{2}(1+m_1), \frac{1}{2}(1+m_2)$ for $E(x) = -wx_1x_2$.

$KL(q|p)$ is convex in m_1 for fixed m_2 and visa versa, but not convex in (m_1, m_2). There may be more than one solution (local minima)

Parallel updating (fixed point iteration) is faster than sequential updating and usually preferred, but convergence is not guaranteed.

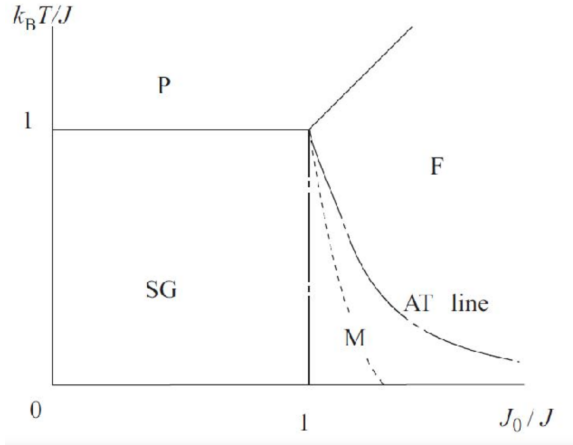
The SK model

The Sherrington-Kirkpatrick model is an Ising model of a spin glass of N fully connected spins [Sherrington and Kirkpatrick, 1975].

$$p(x) = \frac{1}{Z}\exp\left(-\beta E(x)\right) \qquad E(x) = -\sum_{(ij)} w_{ij}x_i x_j - \sum_i \theta_i x_i$$

with $w_{ij} \in \mathcal{N}(J_0/N, J^2/N)$ (so $w_{ij} = \frac{1}{N}J_0 \pm \frac{1}{\sqrt{N}}J$).

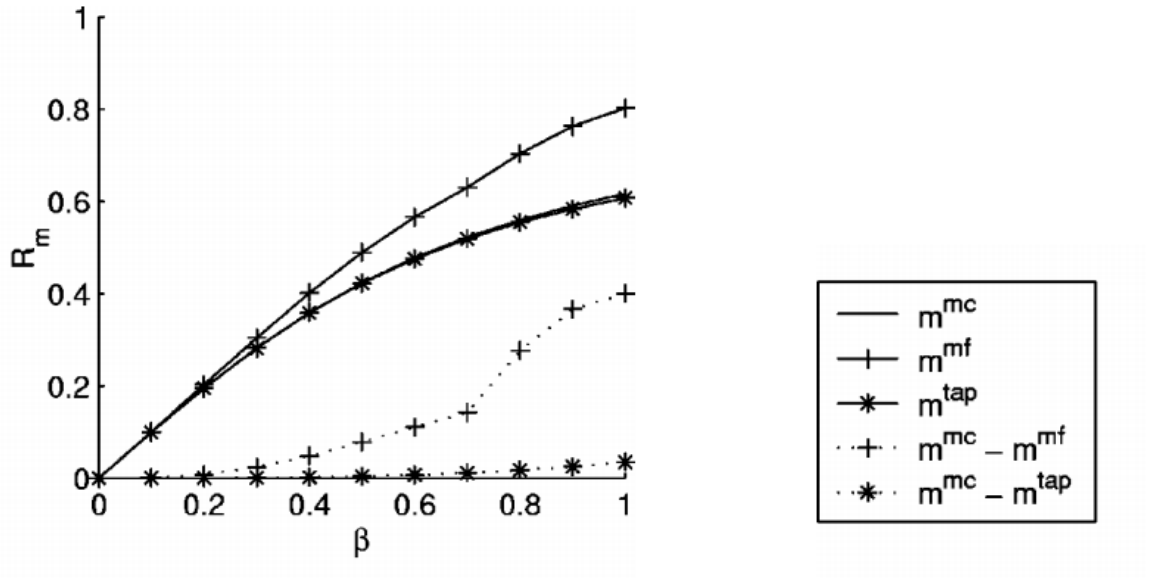
One can compute the phase diagram in the limit $N \to \infty$ and by averaging over the disorder w_{ij} and $\theta_i = 0$ using the so-called replica method.



Phase diagram of the SK model. $\beta = (k_B T)^{-1}$. The model displays a paramagnetic, ferromagnetic and spin-glass phase.

We will discuss the details of the SK model later when we treat the replica method.

Accuracy MF approximation for SK model



Accuracy of the MF approximation versus β in an SK model with $N = 100$ spins, $w_{ij} \in \mathcal{N}(0, 1/N), \theta_i \in \mathcal{N}(0, 1)$. Magnetization RMS values $R_m = \sqrt{\frac{1}{N} \sum_{i=1}^{N} m_i^2}$ (solid lines) of MF approximation and MCMC estimates. The MCMC values are proxies for the exact values. RMS error between MF and MCMC (dashed lines).

[Kappen and Spanjers, 1999]

Linear response correction

We can also compute the correlations in the mean field approximation. The crucial observation is that both the mean firing rates and the correlations can be computed as derivatives of the partition function:

$$
\begin{aligned}
Z &= \sum_x e^{\sum_{(ij)} w_{ij} x_i x_j + \sum_i \theta_i x_i} \\
\langle x_i \rangle &= \frac{\partial \log Z}{\partial \theta_i} \\
\chi_{ij} &\equiv \left\langle x_i x_j \right\rangle - \langle x_i \rangle \left\langle x_j \right\rangle = \frac{\partial^2 \log Z}{\partial \theta_i \partial \theta_j} = \frac{\partial \langle x_i \rangle}{\partial \theta_j} \approx \frac{\partial m_i}{\partial \theta_j}
\end{aligned}
$$

where in the last step we used the MF approximation for $\langle x_i \rangle$.

Linear response correction

The mean field equations give us an implicit relation between m_i and θ_j. We consider how the m_j change if we change θ_i: [24]

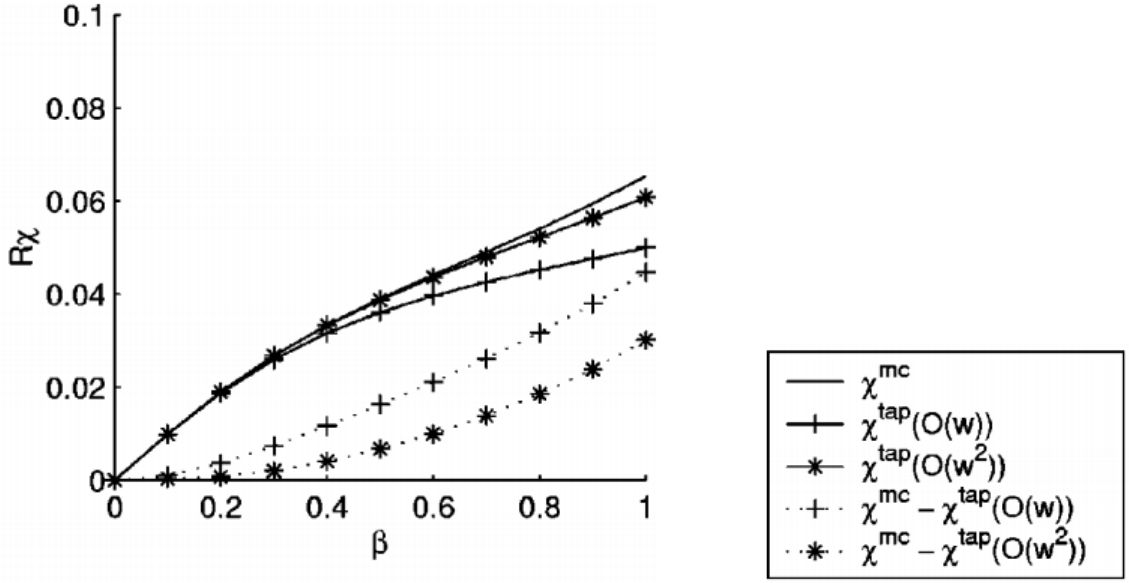
$$
\begin{aligned}
m_i + \delta m_i &= \tanh\left(\sum_j w_{ij}(m_j + \delta m_j) + \theta_i + \delta\theta_i\right) \\
&= \tanh\left(\sum_j w_{ij} m_j + \theta_i\right) + \left(\sum_j w_{ij}\delta m_j + \delta\theta_i\right)(1 - m_i^2) \\
\frac{\delta m_i}{1 - m_i^2} &= \sum_j w_{ij}\delta m_j + \delta\theta_i \qquad \sum_j \left(\frac{\delta_{ij}}{1 - m_i^2} - w_{ij}\right)\frac{\delta m_j}{\delta\theta_i} = 1
\end{aligned}
$$

The solution is

$$
\chi_{ij} = \frac{\delta m_j}{\delta\theta_i} = A^{-1}_{ij} \qquad A_{ij} = \frac{\delta_{ij}}{1 - m_i^2} - w_{ij}
$$

[24] We use that the derivative $\tanh'\left(\sum_j w_{ij}m_j + \theta_i\right) = 1/\cosh^2\left(\sum_j w_{ij}m_j + \theta_i\right) = 1 - \tanh^2\left(\sum_j w_{ij}m_j + \theta_i\right) = 1 - m_i^2$.

Accuracy of linear response approximation for SK model



Accuracy of the LR approximation versus β in a SK model with $N = 100$ spins, $w_{ij} \in \mathcal{N}(0, 1/N), \theta_i \in \mathcal{N}(0, 1)$. Correlation χ RMS values $R_\chi = \sqrt{\frac{2}{N(N-1)} \sum_{i>j} \chi_{ij}^2}$ (solid lines) of LR approximation (Tap O(w)) and MCMC estimates. The MCMC values are proxies for the exact values. RMS error between LR correlations and MCMC (dashed lines).

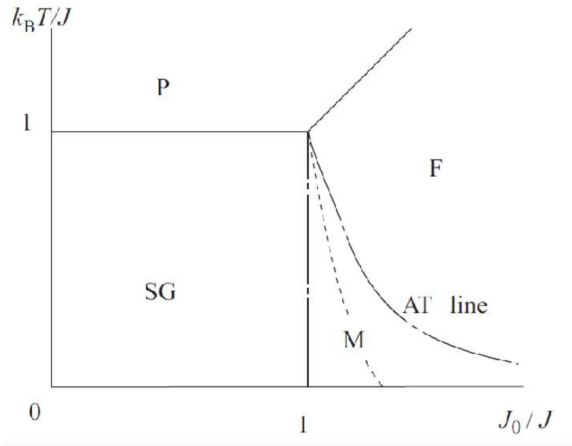
[Kappen and Spanjers, 1999]

The TAP approximation

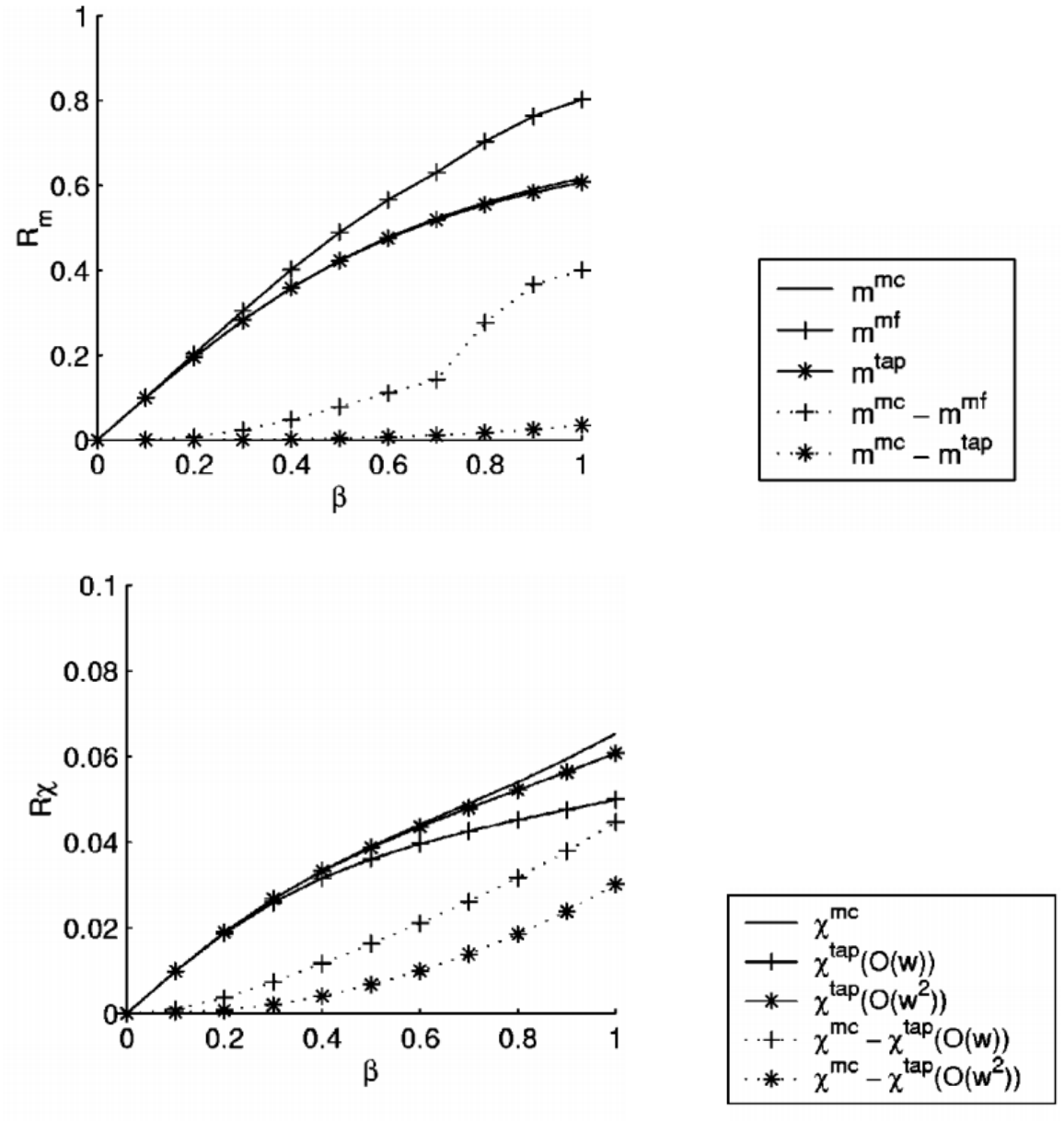
There exists a higher order version of the mean field approximation due to Thouless, Anderson and Palmer, known as the TAP approximation [Thouless et al., 1977]. For the Ising model it is given as

$$m_i = \tanh\left(\sum_j w_{ij} m_j + \theta_i - m_i \sum_j w_{ij}^2 (1 - m_j^2)\right)$$

An elegant derivation of this result is due to Plefka [Plefka, 1982]. The TAP approximation is exact for the SK model when $N \to \infty$ in the P phase and the F phase above the AT line.



The TAP approximation

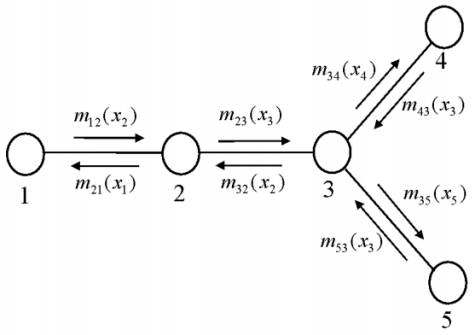


Deterministic approximation for spin models: Belief propagation

- (Loopy) belief propagation
- Derivation by the Bethe free energy

Belief propagation [Pearl, 1988]

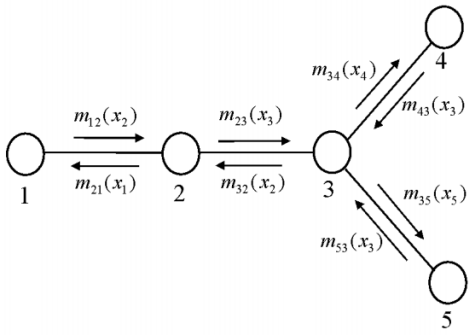
Consider a distribution $p(x_1, x_2, x_3, x_4, x_5) = \frac{1}{Z}\psi(x_1, x_2)\psi(x_2, x_3)\psi(x_3, x_4)\psi(x_3, x_5)$.



We wish to compute

$$
\begin{aligned}
p(x_3) &= \sum_{x_1,x_2,x_4,x_5} p(x_1, x_2, x_3, x_4, x_5) \\
&\propto \underbrace{\sum_{x_2} \underbrace{\sum_{x_1} \psi(x_1, x_2)}_{m_{12}(x_2)} \psi(x_2, x_3)}_{m_{23}(x_3)} \underbrace{\sum_{x_4} \psi(x_3, x_4)}_{m_{43}(x_3)} \underbrace{\sum_{x_5} \psi(x_3, x_5)}_{m_{53}(x_3)} \\
&= m_{23}(x_3) m_{43}(x_3) m_{53}(x_3)
\end{aligned}
$$

Belief propagation



For an arbitrary tree, we initialize messages as 1 at all the leaves and pass messages throughout the tree

$$m_{ij}(x_j) = \sum_{x_i} \psi(x_i, x_j) \prod_{k \in N(i) \setminus j} m_{ki}(x_i)$$

with $N(i)$ the set of neighbors of node i in the graph. The marginal probability at any node is $p(x_i) \propto \prod_{k \in N(i)} m_{ki}(x_i)$.

Loopy belief propagation

BP is exact on trees (graphs without loops). Loopy belief propagation is the same algorithm on a loopy graph.

Consider the undirected graphical model with single node and pair-wise interactions [25]

$$p(x) = \frac{1}{Z} \prod_{ij} \psi_{ij}(x_i, x_j) \prod_i \psi_i(x_i)$$

The BP equations take the form

$$m_{ij}(x_j) \propto \sum_{x_i} \psi_{ij}(x_i, x_j) \psi_i(x_i) \prod_{k \in N(i) \setminus j} m_{ki}(x_i) \tag{1}$$

Initialize all messages $m_{ij}(x_j) = 1$ and iterate until convergence (which is not guaranteed!).

[25] For instance, the Ising model has $\psi_{ij}(x_i, x_j) = e^{w_{ij} x_i x_j}$ and $\psi_i(x_i) = e^{\theta_i x_i}$.

Loopy belief propagation

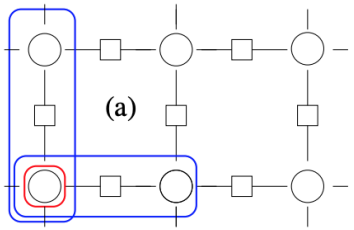
After convergence the BP estimates of the marginals (aka the beliefs) are

$$
\begin{aligned}
b_i(x_i) &\propto \psi_i(x_i) \prod_{k \in N(i)} m_{ki}(x_i) \\
b_{ij}(x_i, x_j) &\propto \psi_{ij}(x_i, x_j)\psi_i(x_i)\psi_j(x_j) \prod_{k \in N(i) \setminus j} m_{ki}(x_i) \prod_{l \in N(j) \setminus i} m_{lj}(x_j) \qquad (2)
\end{aligned}
$$

Derivation of BP: the Bethe free energy [Yedidia et al., 2001]

Consider the so-called Bethe free energy

$$
\begin{aligned}
F(\{b_{ij}, b_i\}) \quad = \quad & -\sum_{i,j} \sum_{x_i, x_j} b_{ij}(x_i, x_j) \log \psi_{ij}(x_i, x_j) - \sum_i \sum_{x_i} b_i(x_i) \log \psi_i(x_i) \\
& + \sum_{i,j} \sum_{x_i, x_j} b_{ij}(x_i, x_j) \log b_{ij}(x_i, x_j) - \sum_i (q_i - 1) \sum_{x_i} b_i(x_i) \log b_i(x_i)
\end{aligned}
$$



Pairwise Bethe

The first line is the average energy under the beliefs. The second line is the entropy of all pair beliefs b_{ij} from which we subtract single variable entropies to compensate for overcounting. q_i is the number of neighbors of node i in the graph.

Since $F(\{b_{ij}, b_i\})$ is a difference of entropies and therefore not convex.

Thm: The extremal points of the Bethe free energy are the BP equations.

Derivation of BP: the Bethe free energy [Yedidia et al., 2001]

Proof. We add Lagrange multiplier terms

$$\begin{aligned} L &= F(\{b_{ij}, b_i\}) + \sum_{ij} \sum_{x_j} \lambda_{ij}(x_j) \left(\sum_{x_i} b_{ij}(x_i, x_j) - b_j(x_j) \right) + \\ & \sum_{ij} \gamma_{ij} \left(\sum_{x_i, x_j} b_{ij}(x_i, x_j) - 1 \right) + \sum_i \gamma_i \left(\sum_{x_i} b_i(x_i) - 1 \right) \end{aligned}$$

Differentiating

$$\begin{aligned} \frac{\partial L}{\partial b_{ij}(x_i, x_j)} &= \log \frac{b_{ij}(x_i, x_j)}{\psi_{ij}(x_i, x_j)} + 1 + \lambda_{ij}(x_j) + \lambda_{ji}(x_i) + \gamma_{ij} = 0 \\ \frac{\partial L}{\partial b_i(x_i)} &= -\log \psi_i(x_i) - (q_i - 1) (\log b_i(x_i) + 1) - \sum_{j \in N(i)} \lambda_{ji}(x_i) + \gamma_i = 0 \end{aligned}$$

Thus

$$
\begin{aligned}
b_{ij}(x_i, x_j) &\propto \psi_{ij}(x_i, x_j) \exp\Big(\lambda_{ij}(x_j) + \lambda_{ji}(x_i)\Big) \\
b_i(x_i) &\propto \psi_i(x_i)^{\frac{1}{1-q_i}} \exp\left(\frac{1}{q_i - 1} \sum_{j \in N(i)} \lambda_{ji}(x_i)\right)
\end{aligned}
$$

which is identical to Eqs. 2 where we defined the messages through $\lambda_{ji}(x_i) = \log \psi_i(x_i) + \sum_{k \in N(i) \setminus j} \log m_{ki}(x_i)$. [26]

[26] $b_{ij}(x_i, x_j)$ follows by direct substitution and

$$
\begin{aligned}
b_i(x_i) &\propto \psi_i(x_i)^{\frac{1}{1-q_i}} \exp\left(\frac{q_i}{q_i - 1} \log \psi_i(x_i) + \frac{1}{q_i - 1} \sum_{j \in N(i)} \sum_{k \in N(i) \setminus j} \log m_{ki}(x_i)\right) \\
&= \psi_i(x_i) \exp\left(\frac{1}{q_i - 1} \sum_{k \in N(i)} \sum_{j \in N(i) \setminus k} \log m_{ki}(x_i)\right) = \psi_i(x_i) \exp\left(\sum_{k \in N(i)} \log m_{ki}(x_i)\right)
\end{aligned}
$$

Finally, since

$$b_i(x_i) \propto \psi_i(x_i) \prod_{k \in N(i)} m_{ki}(x_i)$$

$$b_{ij}(x_i, x_j) \propto \psi_{ij}(x_i, x_j)\psi_i(x_i)\psi_j(x_j) \prod_{k \in N(i)\setminus j} m_{ki}(x_i) \prod_{l \in N(j)\setminus i} m_{lj}(x_j)$$

The marginal constraint $b_j(x_j) = \sum_{x_i} b_{ij}(x_i, x_j)$ becomes

$$\psi_j(x_j) \prod_{l \in N(j)} m_{lj}(x_j) \propto \psi_j(x_j) \prod_{l \in N(j)\setminus i} m_{lj}(x_j) \sum_{x_i} \psi_{ij}(x_i, x_j)\psi_i(x_i) \prod_{k \in N(i)\setminus j} m_{ki}(x_i)$$

$$m_{ij}(x_j) \propto \sum_{x_i} \psi_{ij}(x_i, x_j)\psi_i(x_i) \prod_{k \in N(i)\setminus j} m_{ki}(x_i)$$

which are the BP equations Eq. 3. □

Deterministic approximation for spin models: Belief propagation (continued)

- Convergence of BP
- Max product version of BP
- Factor graph version of BP
- Applications of BP
  - Error correcting codes
  - Clustering

Convergence of BP [Mooij and Kappen, 2007]

Belief propagation is a (parallel) fixed point iteration algorithm $m_i^{t+1} = f_i(m^t)$ with i labeling the different messages. When does this converge?

Linear case: Consider the n dimensional linear system of equations $(A - 1)x = b$ with A an $n \times n$ matrix, 1 the unit matrix and b a vector and denote x^* the solution. We can compute this solution by fixed point iteration starting from a random initial x_0

$$x^{t+1} = Ax^t - b$$

with t labeling the iterations, because when $t \to \infty$ we obtain $x^\infty = Ax^\infty - b$ or $(A - 1)x^\infty = b$.

Define the spectral radius of a matrix A as $\rho(A) = \max_i |\lambda_i|$. The fixed point iteration converges from any initial x^0 to $x^\infty = x^*$ provided that $\rho(A) < 1$.

Convergence of BP [Mooij and Kappen, 2007]

Proof. Denote $y = x - x^*$. Then in terms of y the fixed point iteration becomes $y^{t+1} = Ay^t$ and should converge to $y^\infty = 0$. Suppose that A has eigenvectors v_k and eigenvalues $\lambda_k, k = 1, \ldots, n$. [27]. We can write y^t on the basis of eigenvectors of A as $y^t = \sum_{k=1}^n y_k^t v_k$ with $y_k^t = v_k' \cdot y^t$[28]. Then the fixed point iteration becomes

$$y_k^{t+1} = \lambda_k y_k^t, \qquad k = 1, \ldots, n$$

independent for each k. The solution is $y_k^t = y_k^0 (\lambda_k)^t$ and $|y_k^t| \to 0$ provided that $|\lambda_k| < 1$ for all k.

□

NB: the convergence criterion is independent of initial state y_k^0, ie. convergence is global.

[27] The eigenvalues of A are real when A is a symmetric matrix and complex valued when A is not symmetric.

[28] $'$ denotes transpose

Convergence of BP [Mooij and Kappen, 2007]

Non-linear case: For a non-linear n dimensional mapping $x_i^{t+1} = f_i(x^t)$ a linear stability analysis can be performed around the fixed point x^* that is a solution of $x_i^* = f_i(x^*)$.

Write $x_i^t = x_i^* + \delta x_i^t$. Then

$$x_i^{t+1} = f_i(x^* + \delta x^t) \approx f_i(x^*) + \sum_j \delta x_j^t A_{ij}$$

with $A_{ij} = \left.\frac{\partial f_i(x)}{\partial x_j}\right|_{x=x^*}$. The fixed point x^* is (locally) stable if $\rho(A) < 1$. This means that x^* is an attractive fixed point when the initial point x^0 is sufficiently near x^*.

But local stability does not imply that $x^t \to x^*$ for any initial x^0.

Convergence of BP [Mooij and Kappen, 2007]

Contractions A mapping $x^{t+1} = f(x^t)$ is called a contraction if for any x, y

$$d(f(x), f(y)) \leq K d(x, y)$$

with $d(x, y)$ any distance measure between x, y and $0 \leq K < 1$. If f is a contraction, the sequence $x, f(x), f^2(x), \ldots$ converges to a fixed point x^∞ independent of x.

Thm: A mapping $x^{t+1} = f(x^t)$ is a contraction if $\frac{\partial f_i(x)}{\partial x_j} = B_i(x) A_{ij}$, $A_{ij} \geq 0$ and $|B_i(x)| \leq 1$ and $\rho(A) < 1$. [29]

[29] One might intuitively think that the simpler statement holds, that f is a contraction if

$$\rho(A(x)) < 1 \qquad A_{ij}(x) = \frac{\partial f_i(x)}{\partial x_j}$$

for all x, but this is in general not true (it is true in two dimensions).

BP for the binary spin model

Consider the binary spin model $p(x) = \frac{1}{Z} e^{\sum_{(ij)} w_{ij} x_i x_j + \sum_i \theta_i x_i}$. The BP equations are

$$m_{ij}^{t+1}(x_j) \propto \sum_{x_i} e^{w_{ij} x_i x_j + \theta_i x_i} \prod_{j \in N(i) \setminus j} m_{ki}^t(x_i)$$

The messages $m_{ij}(x_j)$ are distributions over a single bit x_j and are specified by a single number. Convenient parametrizations are either μ_{ij} or a_{ij}:

$$m_{ij}(x_j) = \frac{1}{2}(1 + \mu_{ij} x_j) = \frac{e^{a_{ij} x_j}}{2\cosh(a_{ij})}$$

It is easy to show that μ_{ij} and a_{ij} are related as $\mu_{ij} = \tanh(a_{ij})$. In terms of a_{ij} the BP equation becomes[30]

$$\begin{aligned} m_{ij}^{t+1}(x_j) &= \frac{1}{Z} \sum_{x_i} \exp\left(w_{ij} x_i x_j + \theta_i x_i + \sum_{k \in N(i) \setminus j} a_{ki}^t x_i \right) = \frac{1}{Z} 2\cosh\left(w_{ij} x_j + \theta_i + \sum_{k \in N(i) \setminus j} a_{ki}^t \right) \\ \mu_{ij}^{t+1} &= \frac{m_{ij}^{t+1}(x_j = 1) - m_{ij}^{t+1}(x_j = -1)}{m_{ij}^{t+1}(x_j = 1) + m_{ij}^{t+1}(x_j = -1)} = \tanh\left(\theta_i + \sum_{k \in N(i) \setminus j} a_{ki}^t \right) \tanh(w_{ij}) \end{aligned}$$

[30] In the last step we use the identity $\cosh(a + b) = \cosh(a)\cosh(b) + \sinh(a)\sinh(b)$ with $a = w_{ij} x_j$ and $b = \theta_i + \sum_{k \in N(i) \setminus j} a_{ki}^t$.

Convergence of BP [Mooij and Kappen, 2007]

The BP equations for the binary spin model are of the form

$$\mu_{ij}^{t+1} = \tanh\left(\theta_i + \sum_{k\in N(i)\setminus j} a_{ki}^t\right)\tanh(w_{ij}) \tag{3}$$

This defines a map $a_{ij}^{t+1} = f_{ij}(a^t)$. We check whether this map is a contraction. The derivative is [31]

$$\frac{\partial a_{ij}^{t+1}}{\partial a_{i'j'}^t} = \frac{da_{ij}^{t+1}}{d\mu_{ij}^{t+1}}\frac{\partial \mu_{ij}^{t+1}}{\partial a_{i'j'}^t} = \underbrace{\frac{1-\tanh^2\left(\theta_i + \sum_{k\in N(i)\setminus j} a_{ki}^t\right)}{1-\mu_{ij}^{t+1}(a^t)}\text{sign}(w_{ij})}_{B_{ij}(a^t)}\underbrace{\tanh(|w_{ij}|)\delta_{j'i}\left(1-\delta_{i'j}\right)}_{A_{ij,i'j'}}$$

It is easy to show that $|B_{ij}(a^t)| \leq 1$[32] and A has non-negative entries. Thus, a sufficient condition for the convergence of parallel BP is that $\rho(A) < 1$.

[31] Since $\mu = \tanh(a)$ we have $\frac{d\mu}{da} = \frac{1}{\cosh^2(a)} = 1 - \tanh^2(a) = 1 - \mu^2 = \frac{da}{d\mu}^{-1}$.

[32] Substitute Eq. 3.

Note, that this condition is independent of θ_i. The bound can be further improved for $\theta_i \neq 0$.

Convergence of BP [Mooij and Kappen, 2007]

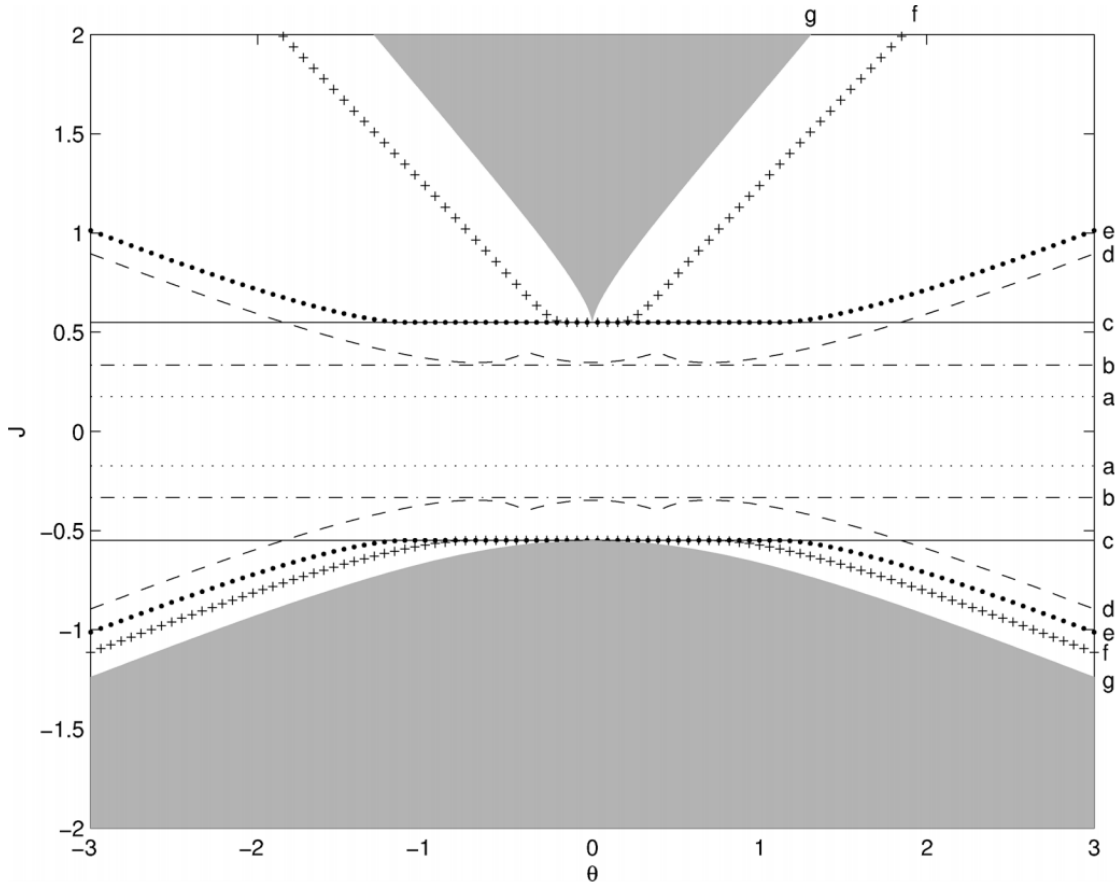


Fig. 4. Comparison of various BP convergence bounds for the fully connected $N = 4$ binary Ising model with uniform coupling J and uniform local field θ. a: Heskes' condition, b: Simon's condition, c: spectral radius condition, d: Dobrushin's condition, e: improved spectral radius condition for $m = 1$, f: improved spectral radius condition for $m = 5$, g: uniqueness of Gibbs' measure condition. See the main text (Section VI-A) for more explanation.

Ising model with uniform coupling and external field. Spectral bounds (c,e,f) are better than other bounds.

Uniqueness of Gibbs measure [Tatikonda and Jordan, 2012]

Uniqueness of the Gibbs measure on the computational tree means that BP fixed point equation

$$\mu_{ij} = \tanh\left(\theta_i + \sum_{k \in N(i) \setminus j} a_{ki}\right) \tanh(w_{ij})$$

has unique solution

In this case with $N = 4, \theta_i = 0, w_{ij} = w$ we find $\mu = \tanh(2a)\tanh(w)$. For small w the solution is $\mu = a = 0$. Linear stability analysis around this solution yields instability when the slope $2\tanh(w) = 1$ or $w = 0.5493$ (grey areas).

Convergence of BP [Mooij and Kappen, 2007]

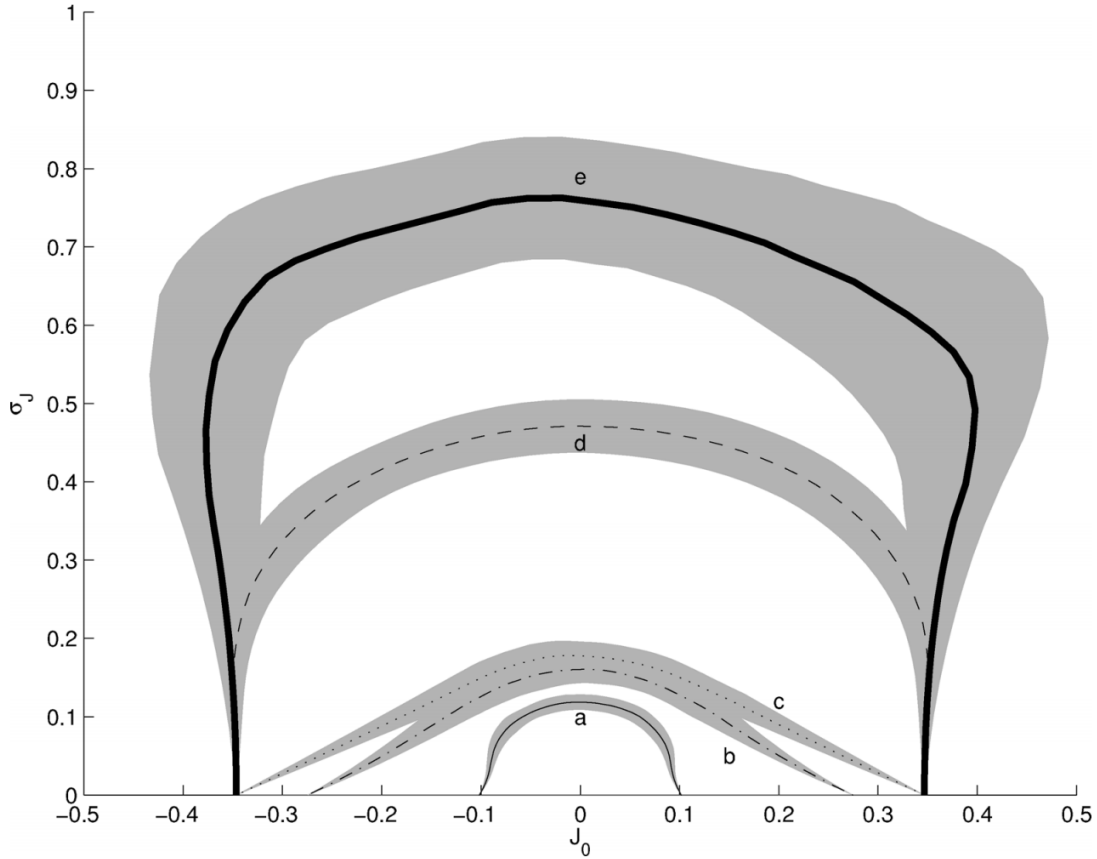
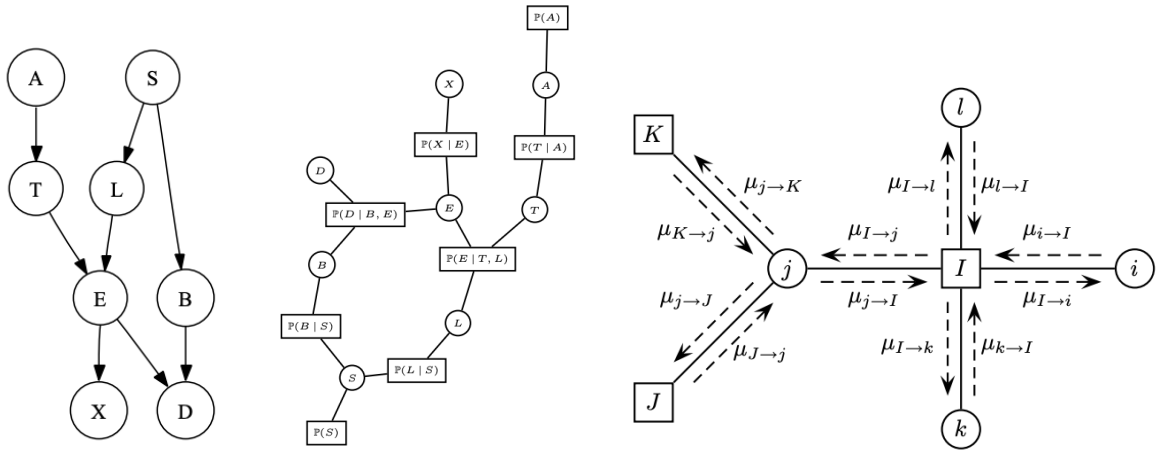


Fig. 5. Comparison of various bounds for BP convergence for toroidal Ising model of size 10×10 with normally distributed couplings $J_{ij} \sim \mathcal{N}(J_0, {\sigma_J}^2)$ and zero local fields. a: Heskes' condition, b: Dobrushin's condition, c: ℓ_1-norm condition, d: spectral radius condition, e: empirical convergence boundary. See the main text (Section VI-B) for more explanation.

SK model with couplings $w_{ij} \in \mathcal{N}(J_0, \sigma_J^2)$ and zero external field. The results are averages over 40 runs. Lines are mean bounds and grey areas are one std. Spectral bound (d) is better than other bounds.

Factor graph representation [Kschischang et al., 2001]



$$p(A,S,T,L,E,B,X,D) = p(A)p(S)p(T|A)p(L|S)p(E|T,L)p(B|S)p(X|E)p(D|E,B)$$

One can write a probabilty distribution as a factor graph. This graph has two types of nodes: Variables labeled as $i, j, \ldots$ and the factors $I, J, \ldots$, that are subsets of variables.

Factor graph version of BP has two types of messages $\mu_{i\to I}$ and $\mu_{I\to i}$:

$$\mu'_{j\to I}(x_j) \quad \propto \quad \prod_{J\in N_j\setminus I} \mu_{J\to j}(x_j) \qquad \mu'_{I\to i}(x_i) \propto \sum_{x_{I\setminus i}} \psi_I(x_I) \prod_{j\in I\setminus i} \mu_{j\to I}(x_j)$$

N_i is the set of factors that contain variable i.

Max product [Pearl, 1988]

The BP message passing rule

$$m_{ij}(x_j) \propto \sum_{x_i} \psi_{ij}(x_i, x_j)\psi_i(x_i) \prod_{j \in N(i) \setminus j} m_{ki}(x_i) \qquad b_i(x_i) \propto \psi_i(x_i) \prod_{j \in N(i)} m_{ji}(x_i)$$

is also know as the sum-product algorithm (a sum of products).

A related algorithm is the max-product algorithm

$$m_{ij}(x_j) \propto \max_{x_i} \psi_{ij}(x_i, x_j)\psi_i(x_i) \prod_{k \in N(i) \setminus j} m_{ki}(x_i) \qquad b_i(x_i) \propto \psi_i(x_i) \prod_{j \in N(i)} m_{ji}(x_i)$$

that can be used to find the maximum of $p(x) = \frac{1}{Z} \prod_{(ij)} \psi_{ij}(x_i, x_j) \prod_i \psi_i(x_i)$.

The max-product solution is $x^* = (x_1^*, \ldots, x_n^*)$ with $x_i^* = \text{argmax}_{x_i} b_i(x_i)$

The max-product algorithm is exact when the graph is a tree. For graphs with loops max-product has somewhat worse convergence than sum-product. See [Globerson and Jaakkola, 2007] for improved versions of max-product and its relation to linear programming relaxations.

Max product [Pearl, 1988]

It is sometimes convenient to write a logarithmic version of the max product algorithm with $\psi_{ij}(x_i, x_j) = e^{E_{ij}(x_i,x_j)}, \psi_i(x_i) = e^{E_i(x_i)}$ and $m_{ij}(x_j) = e^{\mu_{ij}(x_j)}$:

$$\mu_{ij}(x_j) \propto \max_{x_i} \left(E_{ij}(x_i, x_j) + E_i(x_i) + \sum_{k \in N(i) \setminus j} \mu_{ki}(x_i) \right)$$

This version is sometimes referred to as the max-sum algorithm (but it is just the same thing of course).

Applications of BP: Error correction [Frey and MacKay, 1998]

Consider sending bits 0,1 over a noisy channel. Each bit is corrupted by Gaussian noise: $y_k = x_k \pm \xi_k$ with $\mathbb{V}\xi_k = \sigma^2$.

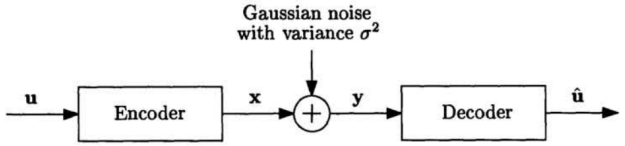


Figure 1: A communication system with a channel that adds Gaussian noise to the transmitted discrete-time sequence.

The received bit is computed by thresholding y: $z = \Theta\left(y - \frac{1}{2}\right)$.

When sending a single bit, the probability that an error occurs is

$$\begin{aligned} p(z \neq x) &= \mathcal{N}(y > \frac{1}{2} | x = 0, \sigma^2) p(x = 0) + \mathcal{N}(y < \frac{1}{2} | x = 1, \sigma^2) p(x = 1) \\ &= \mathcal{N}(y > \frac{1}{2} | x = 0, \sigma^2) \end{aligned}$$

Applications of BP: Error correction [Frey and MacKay, 1998]

We want to send a message $u_k, k = 1, \ldots, K$. We use a longer message of $x_1, \ldots, x_N$ to be able to do error correction at the receiving side. The rate of the code is K/N.

For instance, a very simple repetition code with $N = 2K$ is $x_{2k-1} = x_{2k} = u_k$ has rate $1/2$. The error is corrected by thresholding the average: $z_k = \Theta\left(\frac{1}{2}(y_{2k-1} + y_{2k}) - \frac{1}{2}\right)$. $\frac{1}{2}(y_{2k-1} + y_{2k})$ is Gaussian distributed with mean value u_k and variance $\sigma^2/2$. The probability of error is now[33]

$$\begin{aligned} p(z_k \neq u_k) &= p(z_k = 1|u_k = 0)p(u_k = 0) + p(z_k = 0|u_k = 1)p(u_k = 1) \\ &= \mathcal{N}\left(y > \frac{1}{2}|u = 0, \sigma^2/2\right) \end{aligned}$$

which is smaller.

[33] We assume the prior probability $p(u_k = 0) = p(u_k = 0) = \frac{1}{2}$.

Applications of BP: Error correction [Frey and MacKay, 1998]

More sophisticated codes can be designed, such as Hamming codes. The message $u = (u_1, \ldots, u_K)$ is encoded as $x = (u_1, \ldots, u_K, x_{K+1}, \ldots, x_N)$. See fig. 3a below.

The $x_{K+1:N}$ are required to satisfy parity checks $p(x_i|x_{1:K}) = \delta\left(x_i, \sum_{j \in Q_i} x_j\right), i = K+1, \ldots, N$ with Q_i a subset of of variables in u. [34] We can view the decoding as inference on x in the graphical model

$$p(x|y) \propto \prod_{i=1}^{K} p(x_i) \prod_{i=K+1}^{N} p(x_i|x_{1:K}) \prod_{k=1}^{N} p(y_k|x_k)$$

where $y_{1:N}$ is the observed channel output. $p(y_k|x_k)$ is the noise model that encodes how the channel corrupts the bits.

[34] We take summation modulo 2: for instance $0+1+1 = 0$. In the more powerful LDPCC codes, the u are not explicitly represented. Instead $x = Mu$ with M a $N \times K$ matrix (mod 2 addition).

Applications of BP: Error correction [Frey and MacKay, 1998]

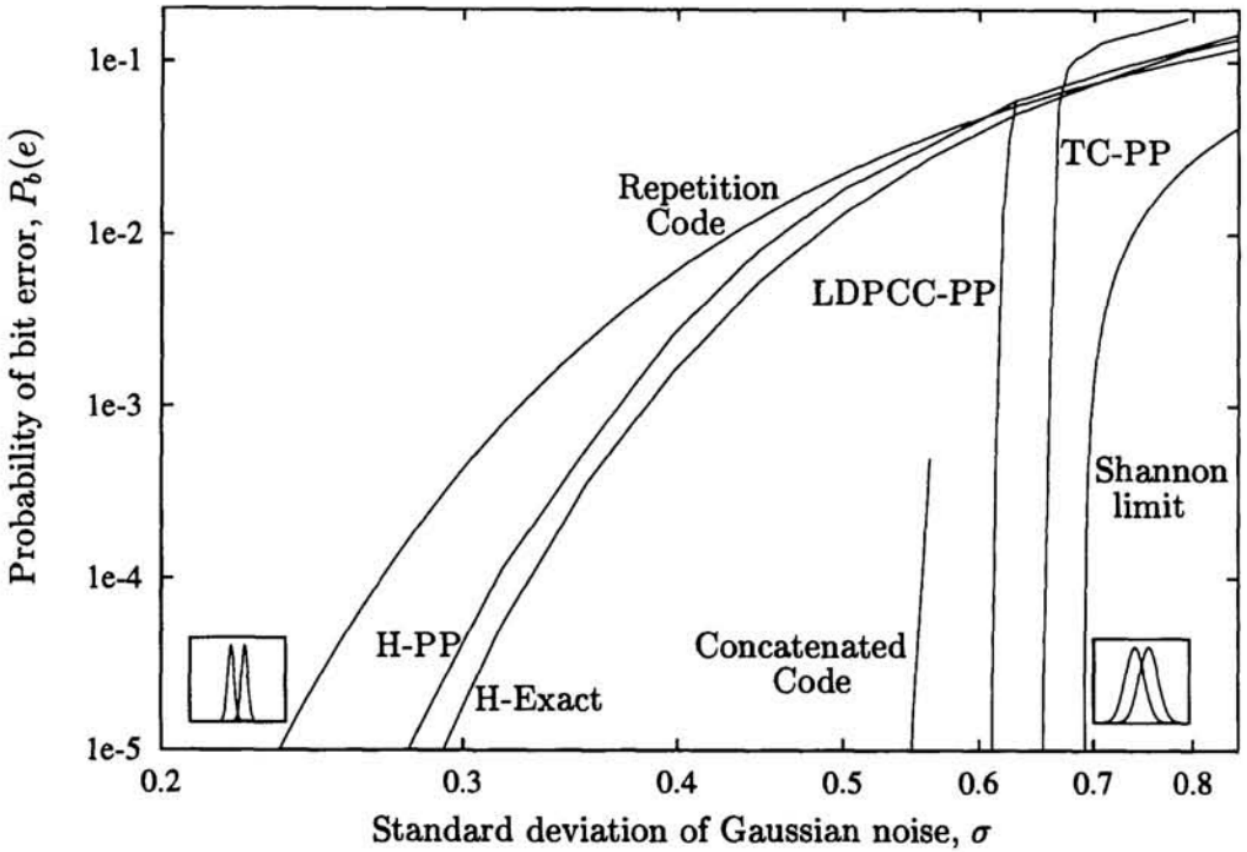


Figure 2: Probability of bit error $P_b(e)$ versus noise level σ for several codes with rates near $1/2$, using 0/1 signalling. It is impossible to obtain a $P_b(e)$ below Shannon's limit (shown on the far right for rate $1/2$). "H-PP" = Hamming code (rate $4/7$) decoded by probability propagation (5 iterations); "H-Exact" = Hamming code decoded exactly; "LDPCC-PP" = low-density parity-check coded decoded by probability propagation; "TC-PP" = turbocode decoded by probability propagation. The thumbnail pictures show the distribution of noisy received signals at the noise levels where the repetition code and the Shannon limit give $P_b(e) = 10^{-5}$.

Applications of BP: Error correction [Frey and MacKay, 1998]

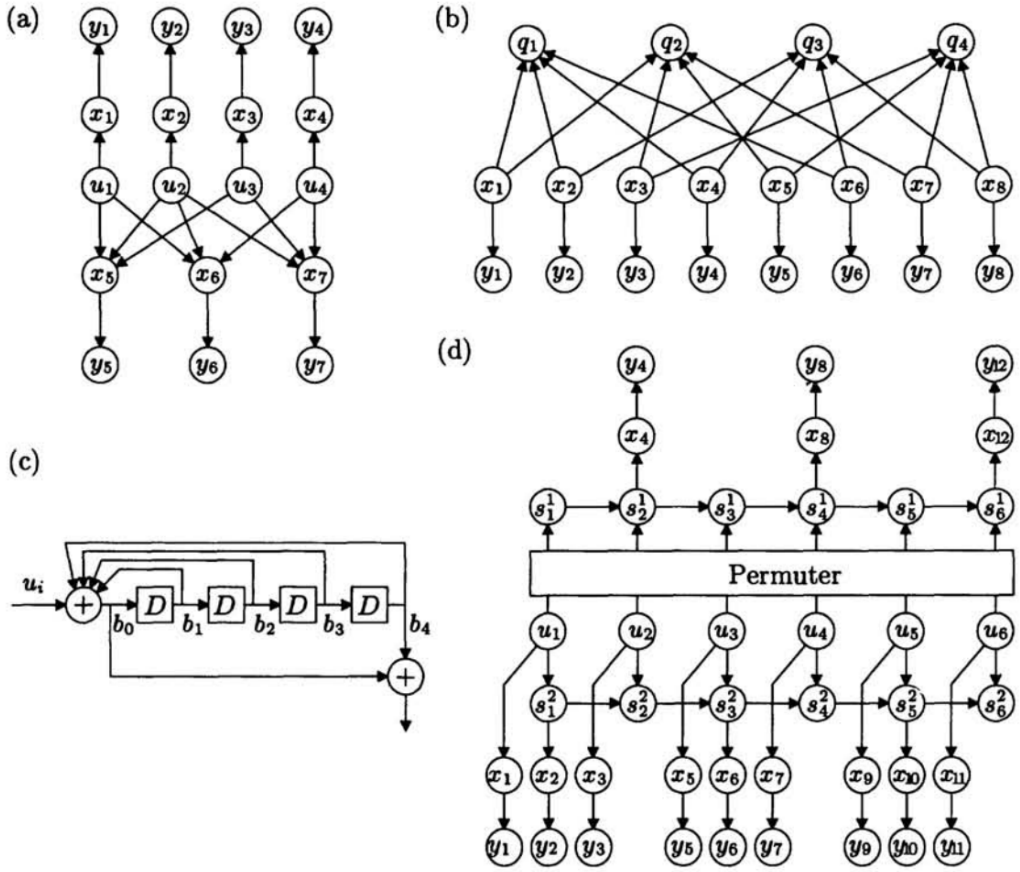


Figure 3: (a) The Bayesian network for a $K = 4$, $N = 7$ Hamming code. (b) The Bayesian network for a $K = 4$, $N = 8$ low-density parity-check code. (c) A block diagram for the turbocode linear feedback shift register. (d) The Bayesian network for a $K = 6$, $N = 12$ turbocode.

The application to error correction is one of the earliest [Gallager, 1963] and most widely used applications of BP.

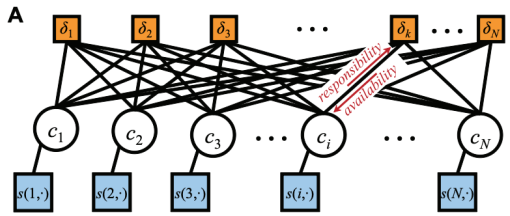
Applications of BP: Clustering [Frey and Dueck, 2007]

Consider a number of data points $x_i, i = 1, \ldots, N$ with distances $s(i, j) = |x_i - x_j|$. The objective is to cluster the points in a number of clusters.

Assign each data point i to a cluster $c_i \in \{1, \ldots, N\}$. If $c_i = i$, the data point i is a cluster center. The objective is to minimize

$$
\begin{aligned}
E(c_1, \ldots, c_N) &= \sum_{i=1}^{N} s(i, c_i) + \sum_{i<k}^{N} \delta_k(c_i, c_k) \\
\delta_k(c_i, c_k) &= \infty \qquad \text{if } c_i = k \text{ and } c_k \neq k \\
&= 0 \quad \text{otherwise}
\end{aligned}
$$

The first term aims to find a center c_i near i. The second term ensures that point i cannot choose k as its center ($c_i = k$) if k is not a center ($c_k \neq k$).



The optimization is solved using a max-sum algorithm on the factor graph.

Applications of BP: Clustering [Frey and Dueck, 2007]

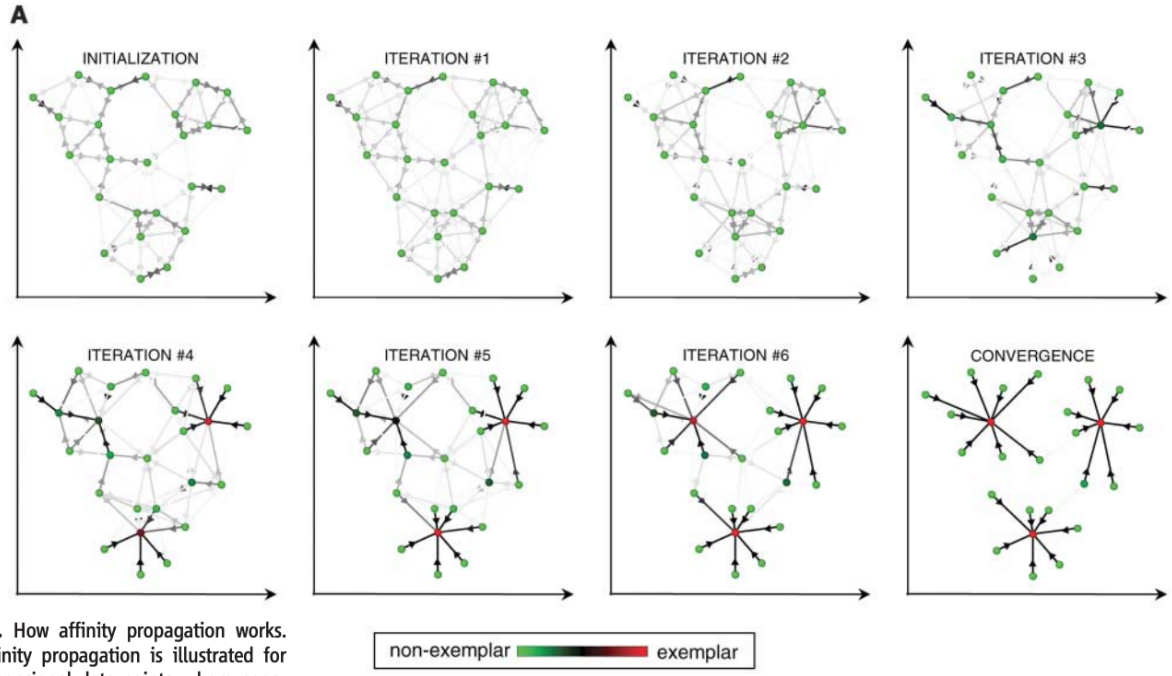


Fig. 1. How affinity propagation works. **(A)** Affinity propagation is illustrated for two-dimensional data points, where negative Euclidean distance (squared error) was used to measure similarity. Each point is colored according to the current evidence that it is a cluster center (exemplar). The darkness of the arrow directed from point *i* to point *k* corresponds to the strength of the transmitted message that point *i* belongs to exemplar point *k*. **(B)** "Responsibilities" *r(i,k)* are sent from data points to candidate exemplars and indicate how strongly each data point favors the candidate exemplar over other candidate exemplars. **(C)** "Availabilities" *a(i,k)* are sent from candidate exemplars to data points and indicate to what degree each candidate exemplar is available as a cluster center for the data point. **(D)** The effect of the value of the input preference (common for all data points) on the number of identified exemplars (number of clusters) is shown. The value that was used in (A) is also shown, which was computed from the median of the pairwise similarities.

Red points are cluster centers. Preference parameter (relative weighting of the s and δ cost terms)

controls number of clusters (D).

Applications of BP: Clustering [Frey and Dueck, 2007]

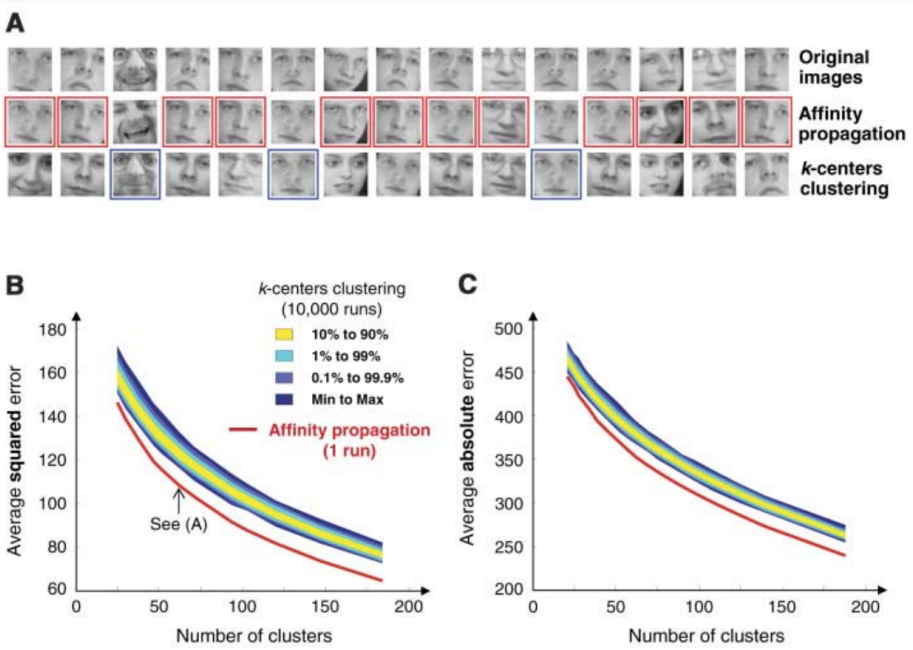


Fig. 2. Clustering faces. Exemplars minimizing the standard squared error measure of similarity were identified from 900 normalized face images (*3*). For a common preference of −600, affinity propagation found 62 clusters, and the average squared error was 108. For comparison, the best of 100 runs of *k*-centers clustering with different random initializations achieved a worse average squared error of 119. (**A**) The 15 images with highest squared error under either affinity propagation or *k*-centers clustering are shown in the top row. The middle and bottom rows show the exemplars assigned by the two methods, and the boxes show which of the two methods performed better for that image, in terms of squared error. Affinity propagation found higher-quality exemplars. (**B**) The average squared error achieved by a single run of affinity propagation and 10,000 runs of *k*-centers clustering, versus the number of clusters. The colored bands show different percentiles of squared error, and the number of exemplars corresponding to the result from (A) is indicated. (**C**) The above procedure was repeated using the sum of absolute errors as the measure of similarity, which is also a popular optimization criterion.

(B,C) Mean squared error (distance between points and their cluster centers) versus number of clusters. Affinity propagation yields better clustering than best of 10.000 k-means clusterings. (A) 15 worse clustered images and their cluster centers under the two methods.

Applications of BP: Clustering [Frey and Dueck, 2007]

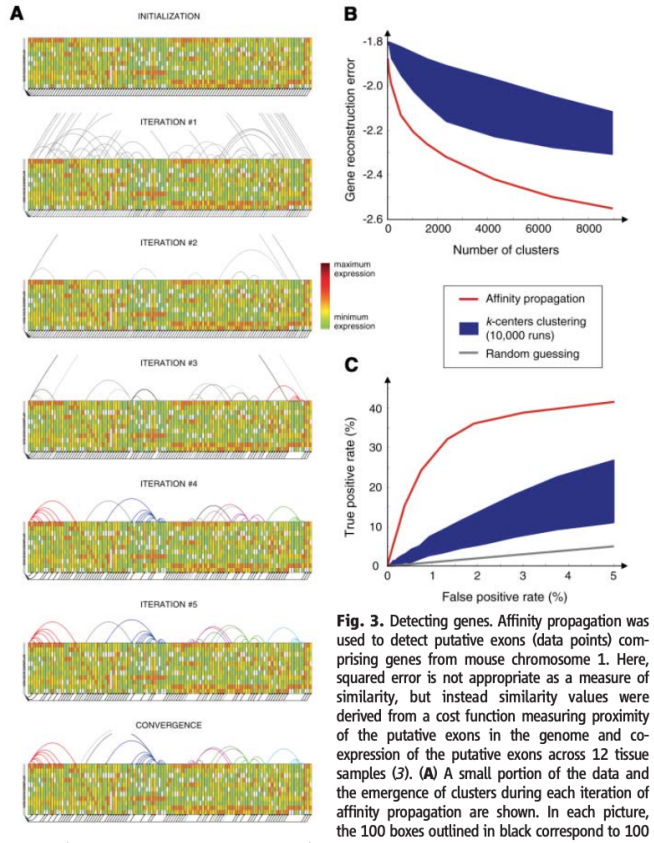


Fig. 3. Detecting genes. Affinity propagation was used to detect putative exons (data points) comprising genes from mouse chromosome 1. Here, squared error is not appropriate as a measure of similarity, but instead similarity values were derived from a cost function measuring proximity of the putative exons in the genome and co-expression of the putative exons across 12 tissue samples (*3*). **(A)** A small portion of the data and the emergence of clusters during each iteration of affinity propagation are shown. In each picture, the 100 boxes outlined in black correspond to 100 data points (from a total of 75,066 putative exons), and the 12 colored blocks in each box indicate the transcription levels of the corresponding DNA segment in 12 tissue samples. The box on the far left corresponds to an artificial data point with infinite preference that is used to account for nonexon regions (e.g., introns). Lines connecting data points indicate potential assignments, where gray lines indicate assignments that currently have weak evidence and solid lines indicate assignments that currently have strong evidence. **(B)** Performance on minimizing the reconstruction error of genes, for different numbers of detected clusters. For each number of clusters, affinity propagation took 6 min, whereas 10,000 runs of *k*-centers clustering took 208 hours on the same computer. In each case, affinity propagation achieved a significantly lower reconstruction error than *k*-centers clustering. **(C)** A plot of true-positive rate versus false-positive rate for detecting exons [using labels from RefSeq (*5*)] shows that affinity propagation also performs better at detecting biologically verified exons than *k*-centers clustering.

Clustering of 75066 exons, each represented as a vector of 12 expression levels as well as their location in the genome. A) illustration of 100 exons that are clustered as intron (black lines on the bottom) or as different exons (color lines on top). (B,C) Clustering error versus number of clusters and comparison with K means clustering.

Statistical physics approach to machine learning

- Replica method
- Cavity method
- Applications of BP and their analysis with replica and cavity method
  - Random satisfiability
  - Compressed sensing

The replica method

In many machine learning problems, the behaviour the learning is characterized by phase transitions.

An example is learning the parameters w of a perceptron with N inputs given a data set D of P samples. The learning problem is to find w that minimizes the classification error $E(D, w)$

$$E^*(D) = \min_w E(D, w)$$

We saw earlier, that for random data sets D in the limit of large N and $P = \alpha N$, the problem can be learned when $\alpha < 2$ ($E^*(D) = 0$) and cannot be learned when $\alpha > 2$ ($E^*(D) > 0$). So there is a phase transition at $\alpha = 2$.

This property hold with probability one for any data set D of a given size P and does not depend not on the details of the particular data set. We are therefore interested in

$$\overline{E^*(D)} = \sum_D p(D) E^*(D)$$

The replica method

We can write $E^*(D) = \min_w E(D,w) = -\lim_{\beta\to\infty} \frac{1}{\beta} \log \int dw e^{-\beta E(D,w)}$ [35] . Therefore

$$\overline{E^*(D)} = -\lim_{\beta\to\infty} \frac{1}{\beta} \overline{\log Z(D)} \qquad Z(D) = \int dw e^{-\beta E(D,w)}$$

We thus needs to compute $\overline{\log Z} = \sum_D p(D) \log Z(D)$.

[35] For large β we can use the Laplace approximation

$$\int dw e^{-\beta E(D,w)} = e^{-\beta E(w^*,D)} \int dw e^{-\frac{\beta}{2}(w-w^*)' H(w^*)(w-w^*)} = e^{-\beta E(w^*,D)} \left(\frac{2\pi}{\beta}\right)^{N/2} \frac{1}{\sqrt{\det H(w^*)}}$$

Taking the $\log$ and $\beta \to \infty$ gives the result.

Random satisfiability [Mezard et al., 2002]

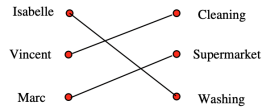
The K-satisfiability problem (K-sat) asks whether one can satisfy simultaneously a set of M constraints between N Boolean variables $y_i = 0, 1$, where each constraint is a clause built as the logical OR involving K variables. An instance of 3-sat is

$$(\neg y_1 \vee y_2 \vee \neg y_3) \wedge (y_1 \vee \neg y_4 \vee y_5) \wedge \ldots$$

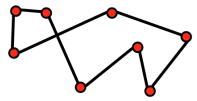
3-sat is at the core of combinatorial optimization theory. An efficient algorithm for solving 3-sat would immediately lead to other algorithms for efficiently solving thousands of different NP hard combinatorial problems [Garey and Johnson, 1979].

Optimisation problems

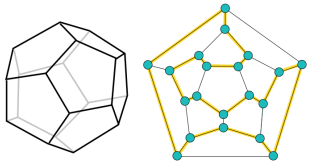
Assignment ("easy", in P)



Travelling salesman ("hard", NPC)



Hamiltonian path ("hard", NPC)



Define $x_i = 2y_i - 1$ and

$$E^*(w) = \min_x E(x, w) \qquad x^* = \text{argmin}_x E(x, w)$$

w specifies the problem instance, the K-sat problem defines the energy as

$$E = \sum_{a=1}^{M} E_a \qquad E_a = 2 \prod_{j \in V(a)} \frac{(1 + w_{a \to j} x_j)}{2}$$

with $w_{a \to j} = \pm 1$. Each clause involves K variables $x_j \pm 1$ and is satisfied ($E_a = 0$) when at least one x_j has a sign opposite to $w_{a \to j}$ (implementing the OR).

For instance $(\neg y_1 \vee y_2 \vee \neg y_3)$ is always true (1) except when $(y_1, y_2, y_3) = (1, 0, 1)$. Define $w = (1, -1, 1)$ then $E(1, 0, 1) = 1$ and zero otherwise.

Can we say something about when we can solve 3 SAT ($E^* = 0$) and when not ($E^* > 0$)?

Define $Z(w) = \sum_x e^{-\beta E(w,x)}$. Then

$$\lim_{\beta\to\infty} \log Z(w) = \lim_{\beta\to\infty} \log \sum_x e^{-\beta E(x,w)} = \lim_{\beta\to\infty} \log e^{-\beta E(w,x^*)} = -\lim_{\beta\to\infty} \beta E(w,x^*) = -E^*(w)$$

Thus, $E^*(w) = -\lim_{\beta\to\infty} \frac{1}{\beta} \log Z(w)$.

We wish to estimate $\overline{E^*(w)}$ where we average over instances of given size (number of variables), number of clauses M and variable per clause K.

The replica method

Powerful method to compute $\overline{\log Z}$ is the replica method, which is based on the following identity

$$\log x = \lim_{n \to 0} \frac{1}{n} (x^n - 1)$$

[36] Thus

$$\overline{\log Z} \quad = \quad \lim_{n \to 0} \frac{1}{n} \left(\overline{Z^n} - 1 \right)$$

We get

$$Z(w)^n = \left(\sum_x e^{-\beta E(x,w)} \right)^n = \sum_{x^1} \cdots \sum_{x^n} e^{-\beta \sum_a E(x^a,w)}$$

Z^n is the partition sum of n copies (replicas) of the original system. $\overline{Z^n}$ can be computed in the limit $N \to \infty$. $\overline{\log Z}$ is obtained by taking the number of copies $n \to 0$!

[36] Proof: $x^n - 1 = e^{n \log x} - 1 = n \log x + O(n^2)$.

Application of replica method for machine learning

The replica method has been applied to analyze many important machine learning problems:

The storage capacity of the Hopfield model [Amit et al., 1985]
Performance of error correcting codes [Kabashima and Saad, 1999]
Random satisfiability problem [Mezard et al., 2002]
Compressed sensing [Krzakala et al., 2012]

See also `https://scholar.harvard.edu/files/madvani/files/repcavmes6.pdf`

The replica method for the SK model

We will sketch[37] the derivation of the simplest replica estimate of $\overline{\log Z}$, the so-called replica symmetric solution, for the SK model

$$p(x) = \frac{1}{Z} e^{\beta \sum_{(ij)} w_{ij} x_i x_j}$$

with w_{ij} Gaussian with mean $\frac{J_0}{N}$ and variance $\frac{J^2}{N}$.

$$Z^n = \left(\sum_x e^{\beta \sum_{(ij)}^N w_{ij} x_i x_j} \right)^n = \mathrm{Tr} e^{\beta \sum_{a=1}^n \sum_{(ij)} w_{ij} x_i^a x_j^a} = \mathrm{Tr} e^{\sum_{(ij)} w_{ij} z_{ij}}$$

where a labels the different replicas and $\mathrm{Tr} = \sum_{x^1} \ldots \sum_{x^n}$ the sum over all states in all replicas and $z_{ij} = \beta \sum_a x_i^a x_j^a$.

Note, that the replicas are independent, but will get coupled when averaging over w.

[37] Details of the calculation are given in later slides and [Sherrington and Kirkpatrick, 1975].

The replica method for the SK model

1. Average over the disorder The first step is to average over w

$$\overline{Z^n} = \int dw p(w) \mathrm{Tr} \prod_{(ij)} e^{w_{ij} z_{ij}} = \mathrm{Tr} \prod_{(ij)} \int dw_{ij} p(w_{ij}) e^{w_{ij} z_{ij}}$$

because $p(w)$ is factorized: $p(w_{12}, w_{13}, \ldots) = p(w_{12}) p(w_{13}) \ldots$.

We use the Gaussian integral $\int dw \mathcal{N}(w|J_0, J^2) e^{wz} = e^{J_0 z + \frac{1}{2} J^2 z^2}$. The result is [38]

$$\overline{Z^n} = \mathrm{Tr} \exp\left(\frac{1}{2} \beta \frac{J_0}{N} \sum_a \left(\sum_i x_i^a \right)^2 + \frac{1}{2} \beta^2 \frac{J^2}{N} \sum_{(ab)} \left(\sum_i x_i^a x_i^b \right)^2 + \frac{1}{4} \beta^2 J^2 n N \right)$$

[38] See details 1.

The replica method for the SK model

2. Sum over states In order to be able to do the sum over states, we need to replace the quadratic terms $\left(\sum_i x_i^a\right)^2, \left(\sum_i x_i^a x_i^b\right)^2$ by linear terms. This can be done by using for each such term the Gaussian identity

$$e^{\lambda a^2} = \frac{\sqrt{N}}{\sqrt{2\pi}} \int dy e^{-\frac{N}{2}y^2 + \sqrt{2\lambda N} a y}$$

which introduces integration variables $y_a, y_{(ab)}$. Note, that this replaces a^2 by a. Then all spins become independent and the result is [39]

$$\begin{aligned}
\overline{Z^n} &= e^{\frac{1}{4}\beta^2 J^2 n N} \int \prod_a \sqrt{\frac{N}{2\pi}} dy_a \prod_{(ab)} \sqrt{\frac{N}{2\pi}} dy_{(ab)} \exp\left(-\frac{N}{2} \sum_a y_a^2 - \frac{N}{2} \sum_{(ab)} y_{(ab)}^2 \right) I(y)^N \\
I(y) &= \sum_x \exp\left(\sqrt{\beta J_0} \sum_a x^a y_a + \beta J \sum_{(ab)} x^a x^b y_{(ab)} \right)
\end{aligned}$$

with $x = (x^1, \ldots, x^n)$ the activity of a single spin in the different replicas.

[39] See details 2.

For instance

$$\sum_{x_1,\ldots,x_N} e^{\frac{\beta J_0}{2N}\left(\sum_i x_i\right)^2} = \sum_{x_1,\ldots,x_N} \int dy e^{-\frac{N}{2}y^2 + \sqrt{\beta J_0}\sum_i x_i y} = \int dy e^{-\frac{N}{2}y^2} \sum_{x_1,\ldots,x_N} e^{\sqrt{\beta J_0}\sum_{i=1}^N x_i y}$$

$$\sum_{x_1,\ldots,x_N} e^{\sqrt{\beta J_0}\sum_i x_i y} = \prod_{i=1}^N \sum_{x_i} e^{\sqrt{\beta J_0} x_i y} = I(y)^N \qquad I(y) = \sum_x e^{\sqrt{\beta J_0} x y}$$

$$\sum_{x_1,\ldots,x_N} e^{\frac{\beta J_0}{2N}\left(\sum_i x_i\right)^2} = \int dy e^{-\frac{N}{2}y^2} I(y)^N$$

Sum over 2^N terms replaced by one Gaussian integral!

The replica method for the SK model

3. Replica symmetric Ansatz We evaluate the integral over $y_a, y_{(ab)}$ in the limit $N \to \infty$ using the Laplace approximation

$$\overline{Z^n} = \int dy e^{-\beta n N F(y)} \approx e^{-\beta n N F(y^*))} \qquad y^* = \min_y F(y)$$

with $y = \{y_a, y_{(ab)}\}$. One makes the drastic so-called replica symmetry (RS) assumption that the optimal solution is symmetric in the sense

$$y_a \to \sqrt{J_0 \beta} m \qquad y_{(ab)} \to J\beta q$$

with only two variational parameters m, q and $\sqrt{J_0\beta}, J\beta$ convenient scalings.

The replica method for the SK model

The remaining $\sum_x$ is easily performed by introducing one more Gaussian integral. We then take the limit of $N \to \infty$ and then the limit of $n \to 0$.

The final result is [40]

$$\begin{aligned}
\overline{\log Z} &= -\beta N F(m^*, q^*) \\
F(m,q) &= \frac{1}{2} J_0 m^2 - \frac{1}{4}\beta J^2 (q-1)^2 - \frac{1}{\sqrt{2\pi}\beta} \int dz e^{-z^2/2} \log[2\cosh(\beta J_0 m + \beta J \sqrt{q} z)]
\end{aligned}$$

with m^*, q^* are obtained by minimizing $F(m,q)$:

$$\begin{aligned}
\frac{\partial F(m,q)}{\partial q} &= 0 \quad \to \quad q = 1 - \frac{1}{\sqrt{2\pi}} \int dz e^{-z^2/2} \mathrm{sech}^2(\beta J \sqrt{q} z + \beta J_0 m) \\
\frac{\partial F(m,q)}{\partial m} &= 0 \quad \to \quad m = \frac{1}{\sqrt{2\pi}} \int dz e^{-z^2/2} \tanh(\beta J \sqrt{q} z + \beta J_0 m)
\end{aligned}$$

[40] See details 3.

The replica method for the SK model [Sherrington and Kirkpatrick, 1975]

q, m can be solved for all values of J_0, J giving the phase portrait.

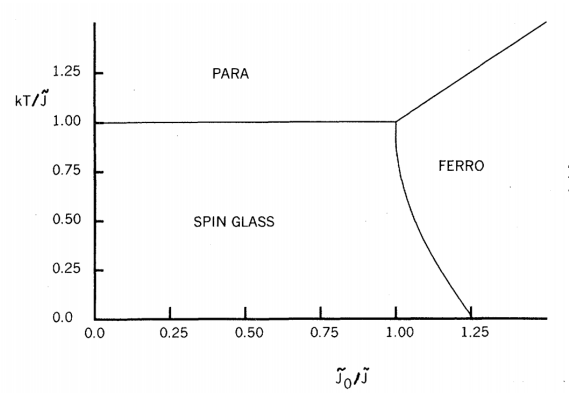
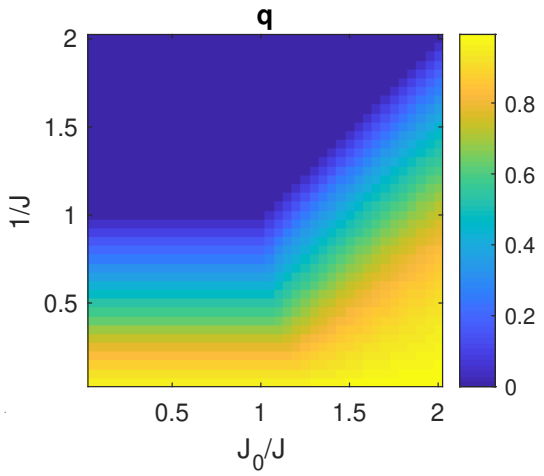
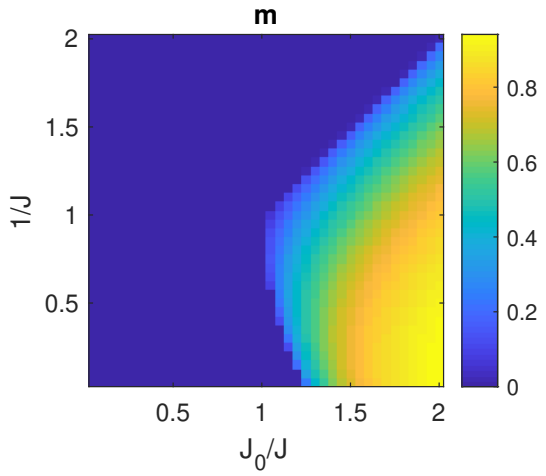


FIG. 1. Phase diagram of spin-glass ferromagnet.





One can show that m, q can be interpreted as

$$m = \overline{\langle x_i \rangle_w} \qquad q = \overline{\langle x_i \rangle_w^2}$$

where the subscript w denotes the thermal average at fixed disorder w and the overline averages over the disorder.

High temperature

The replica method for the SK model [Sherrington and Kirkpatrick, 1975]

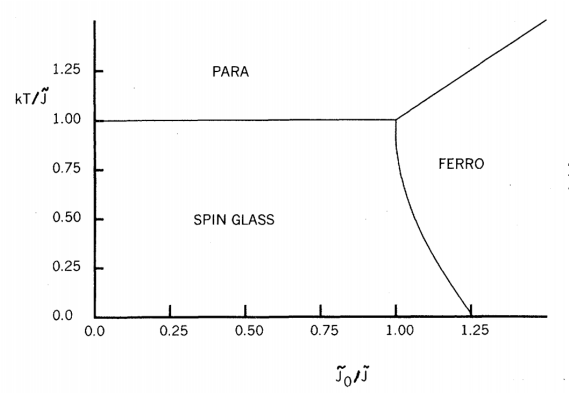
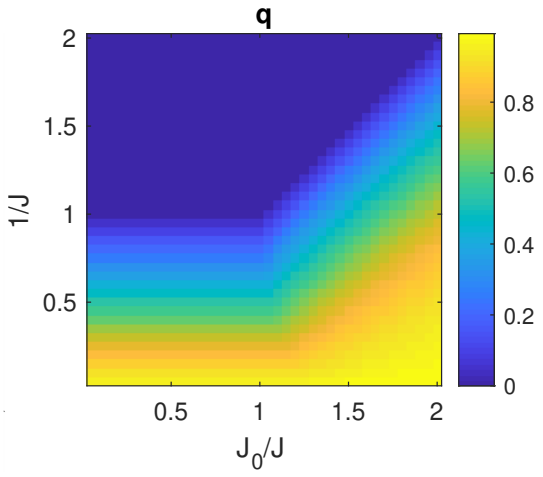
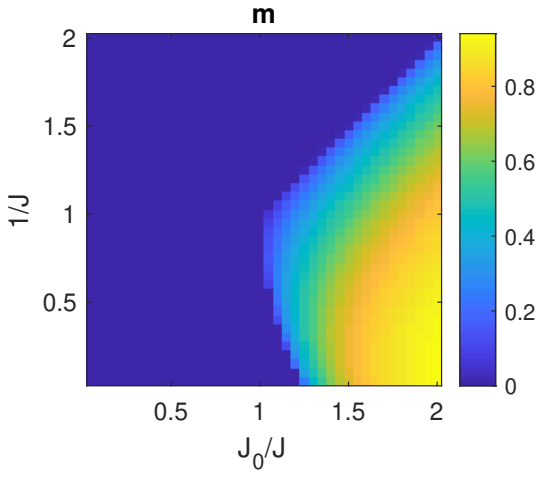


FIG. 1. Phase diagram of spin-glass ferromagnet.



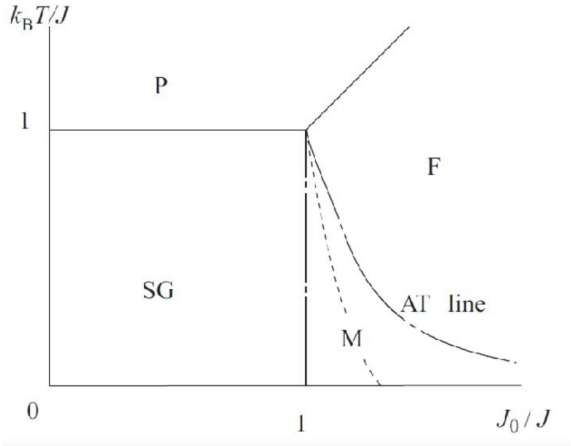


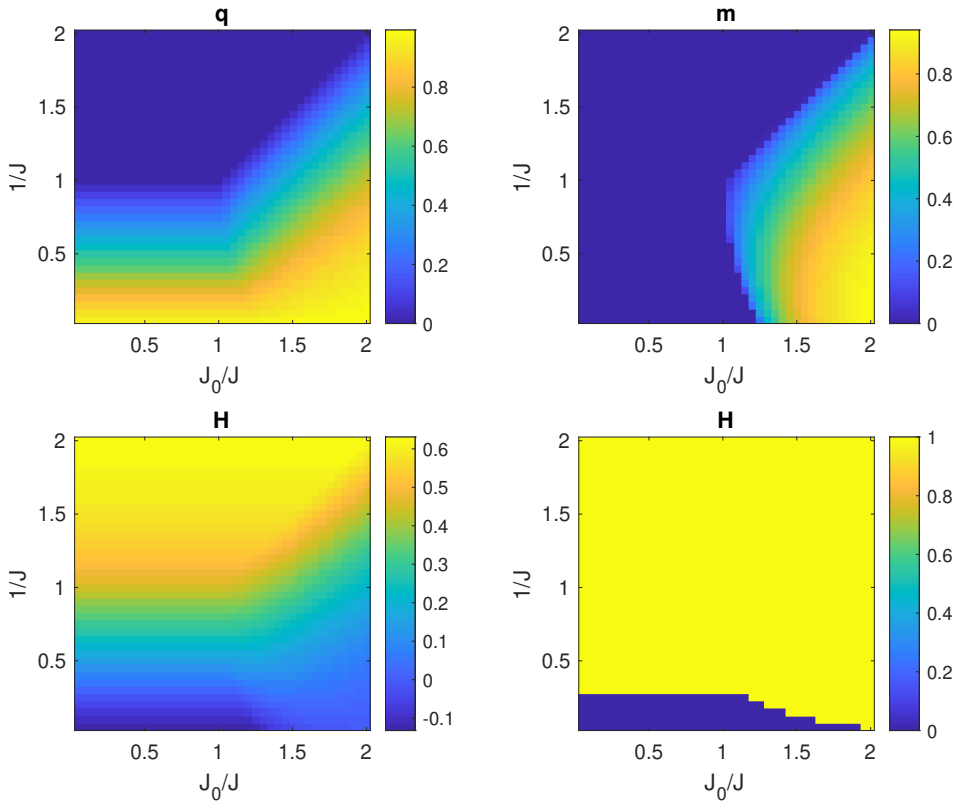
We distinguish three regions:

- $q = m = 0$ implies that $\langle x_i \rangle_w = 0$ for all w[41]: Spins have no preferred orientation for all w. This is the paramagnetic phase.
- $m, q > 0$ implies that $\langle x_i \rangle_w > 0$ for all w: Spins have preferred orientation for all w. This is the ferromagnetic phase.
- The remaining region is $m = 0$ and $q > 0$. $q > 0$ implies that $\langle x_i \rangle_w \neq 0$ for all w. $m = 0$ implies that $\langle x_i \rangle_w = \pm$ with equal probability. This is the spin glass phase. This also called frozen disorder: each spin has a random non-zero value.

[41] For all w means: for all w from the Gaussian distribution with given mean J_0/N and variance J^2/N

The validity of the replica symmetric solution





One can analyse the stability of the RS solution by computing the Hessian (a $\frac{1}{2}n(n+1)\times\frac{1}{2}n(n+1)$ matrix) in the RS solution m, q. The RS solution is unstable below the AT line and is thus no longer valid.

An indication of the failure is that the RS entropy $H = -\frac{\partial F}{\partial T}$ becomes negative at low temperature (large J), which of course is non-physical.

Details of the replica calculation step 1

We average over the independent random variables w_{ij}. We use the Gaussian integral $\int dw \mathcal{N}(w|J_0, J^2) e^{wz} = e^{J_0 z + \frac{1}{2} J^2 z^2}$. with $z_{ij} = \beta \sum_a x_i^a x_j^a$

$$\overline{Z^n} = \mathrm{Tr} \prod_{(ij)} \overline{e^{\beta w_{ij} \sum_a x_i^a x_j^a}} = \mathrm{Tr} \exp\left(\frac{J_0}{N} \sum_{(ij)} z_{ij} + \frac{J^2}{2N} \sum_{(ij)} z_{ij}^2 \right)$$

We use

$$\sum_{(ij)} z_{ij} \approx \frac{1}{2} \sum_{ij} z_{ij} = \frac{\beta}{2} \sum_{a=1}^{n} \left(\sum_{i=1}^{N} x_i^a \right)^2$$

$$\sum_{(ij)} z_{ij}^2 \approx \frac{1}{2} \sum_{ij} z_{ij}^2 = \frac{\beta^2}{2} \sum_{a,b=1}^{n} \sum_{i,j=1}^{N} x_i^a x_j^a x_i^b x_j^b = \beta^2 \sum_{(ab)} \left(\sum_{i=1}^{N} x_i^a x_i^b \right)^2 + \frac{\beta^2}{2} n N^2$$

where we ignore subleading orders of N.

Substitution gives

$$\overline{Z^n} = \mathrm{Tr} \exp\left(\frac{1}{2} \beta \frac{J_0}{N} \sum_a \left(\sum_i x_i^a \right)^2 + \frac{1}{2} \beta^2 \frac{J^2}{N} \sum_{(ab)} \left(\sum_i x_i^a x_i^b \right)^2 + \frac{1}{4} \beta^2 J^2 n N \right)$$

Details of the replica calculation step 2

In order to be able to do the sum over states, Tr, we need to replace the quadratic terms $\left(\sum_i x_i^a\right)^2, \left(\sum_i x_i^a x_i^b\right)^2$ by linear terms. This can be done by using the Gaussian identity

$$e^{\lambda a^2} = \sqrt{\frac{N}{2\pi}} \int dy e^{-\frac{1}{2}Ny^2 + \sqrt{2\lambda N}ay} \tag{4}$$

which replaces the a^2 by a at the cost of an integral.

Thus, we must introduce integration variables $y_a, y_{(ab)}$, one for each of the terms $\left(\sum_i x_i^a\right)^2, \left(\sum_i x_i^a x_i^b\right)^2$. The result is

$$\overline{Z^n} = e^{\frac{1}{4}\beta^2 J^2 nN} \int \prod_a \sqrt{\frac{N}{2\pi}} dy_a \prod_{(ab)} \sqrt{\frac{N}{2\pi}} dy_{(ab)} \exp\left(-\frac{N}{2}\sum_a y_a^2 - \frac{N}{2}\sum_{(ab)} y_{(ab)}^2\right) \tilde{I}(y)$$

$$\tilde{I}(y) = \mathrm{Tr} \exp\left(\sqrt{\beta J_0}\sum_a \sum_i x_i^a y_a + \beta J \sum_{(ab)} \sum_i x_i^a x_i^b y_{(ab)}\right)$$

Details of the replica calculation step 2

Now, the exponent of $\tilde{I}$ is a product of N terms, one for each i. Thus,

$$\tilde{I}(y) = \prod_{i=1}^{N} \left(\sum_{x_i} \exp\left(\sqrt{\beta J_0} \sum_a x_i^a y_a + \beta J \sum_{(ab)} x_i^a x_i^b y_{(ab)} \right) \right) = I(y)^N$$

$$I(y) = \sum_{x} \exp\left(\sqrt{\beta J_0} \sum_a x^a y_a + \beta J \sum_{(ab)} x^a x^b y_{(ab)} \right)$$

where we have written $\sum_{x_i} = \sum_x$ since all these sums are identical (dont depend on i) and with $x = (x^1, \ldots, x^n)$ the activity of a single spin in the different replicas.

Details of the replica calculation step 3

In terms of m, q we get by direct substitution [42]

$$\begin{aligned}
\overline{Z^n} &= e^{\frac{1}{4}\beta^2 J^2 nN} \int dm dq \exp\left(-\frac{N}{2} n\beta J_0 m^2 - \frac{N}{4} n(n-1)\beta^2 J^2 q^2\right) I(m,q)^N \\
I(m,q) &= \sum_x \exp\left(\beta J_0 m \sum_a x^a + \beta^2 J^2 q \sum_{(ab)} x^a x^b\right) \\
&= \sum_x \exp\left(\beta J_0 m \sum_a x^a + \frac{1}{2}\beta^2 J^2 q \left(\left(\sum_a x^a\right)^2 - n\right)\right)
\end{aligned}$$

In order to do $\sum_x$ we introduce another Gaussian integral using the identity Eq. 4. [43]

$$\begin{aligned}
I(m,q) &= e^{-\frac{n}{2}\beta^2 J^2 q} \frac{1}{\sqrt{2\pi}} \int dz e^{-z^2/2} \sum_x \exp\left((\beta J_0 m + \beta J\sqrt{q} z) \sum_a x^a\right) \\
&= e^{-\frac{n}{2}\beta^2 J^2 q} \frac{1}{\sqrt{2\pi}} \int dz e^{-z^2/2} \left(2\cosh\left(\beta J_0 m + \beta J\sqrt{q} z\right)\right)^n
\end{aligned}$$

[42] We use $\sum_{(ab)} x^a x^b = \frac{1}{2}\sum_{a=1}^n \sum_{b=1, b\neq a}^n x^a x^b = \frac{1}{2}\left(\sum_{a,b=1}^n x^a x^b - \sum_{a=1}^n (x^a)^2\right) = \frac{1}{2}\left((\sum_a^n x^a)^2 - n\right)$

[43] $\sum_x \exp\left((\beta J_0 m + \beta J\sqrt{q} z)\sum_a x^a\right) = \prod_{a=1}^n \sum_{x^a=\pm 1} \exp\left((\beta J_0 m + \beta J\sqrt{q} z) x^a\right) = \left(2\cosh(\beta J_0 m + \beta J\sqrt{q} z)\right)^n$.

Details of the replica calculation step 3

To lowest order in n we obtain [44]

$$I(m,q)^N = e^{-nN\left(\frac{1}{2}\beta^2 J^2 q + \frac{1}{\sqrt{2\pi}}\int dz e^{-z^2/2}\log[2\cosh(\beta J_0 m + \beta J\sqrt{q}z)]\right)}$$

Substituting $I(m,q)$ in $\overline{Z^n}$, to first order in n we obtain [45]

$$\begin{aligned}\overline{Z^n} &= \int dm dq e^{-\beta n N F(m,q)} \\ F(m,q) &= \frac{1}{2}J_0 m^2 - \frac{1}{4}\beta J^2 (q-1)^2 - \frac{1}{\sqrt{2\pi}\beta}\int dz e^{-z^2/2}\log[2\cosh(\beta J_0 m + \beta J\sqrt{q}z)]\end{aligned}$$

---

[44]

$$\frac{1}{\sqrt{2\pi}}\int dz e^{-z^2/2} f(z)^n = \frac{1}{\sqrt{2\pi}}\int dz e^{-z^2/2} e^{n\log f(z)} \approx 1 + \frac{n}{\sqrt{2\pi}}\int dz e^{-z^2/2}\log f(z) \approx \exp\left(\frac{n}{\sqrt{2\pi}}\int dz e^{-z^2/2}\log f(z)\right)$$

[45] Note, that the factor $\frac{1}{2}n(n-1)$ in $\overline{Z^n}$ comes from summing over all pairs of replicas (quadratic in n). But, in the limit $n \to 0$ the leading order in n is $-\frac{1}{2}n$!

Details of the replica calculation step 3

For large N we evaluate the integral in the Laplace approximation

$$\overline{Z^n} \approx e^{-\beta n N F(m^*,q^*)}$$

with m^*, q^* are obtained by minimizing $F(m,q)$:

$$\begin{aligned}
\frac{\partial F(m,q)}{\partial q} &= 0 \quad \rightarrow \quad q = 1 - \frac{1}{\sqrt{2\pi}} \int dz e^{-z^2/2} \text{sech}^2 \left(\beta J \sqrt{q} z + \beta J_0 m\right) \\
\frac{\partial F(m,q)}{\partial m} &= 0 \quad \rightarrow \quad m = \frac{1}{\sqrt{2\pi}} \int dz e^{-z^2/2} \tanh \left(\beta J \sqrt{q} z + \beta J_0 m\right)
\end{aligned}$$

Finally we take the limit $n \rightarrow 0$ and obtain

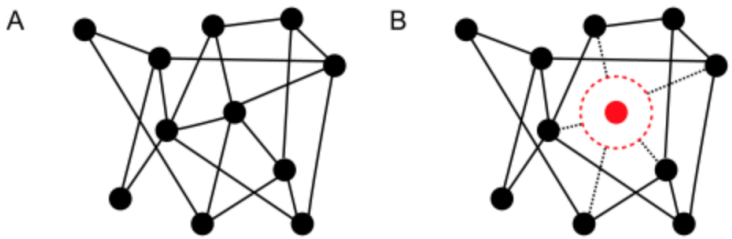
$$\overline{\log Z} = \lim_{n \rightarrow 0} \frac{1}{n} \left(\overline{Z^n} - 1\right) = \frac{1}{n} \left(e^{-\beta n N F(m^*,q^*)} - 1\right) = -\beta N F(m^*, q^*)$$

Cavity method

There is an alternative, possibly more intuitive, way to derive the MF equations for the SK model, known as the cavity method. [Onsager, 1936]

The idea is to consider a particular spin i and to separate the energy in terms involving x_i and the rest

$$E = x_i h_i + E_{\setminus i} \qquad E_{\setminus i} = \sum_{(k,l),k\neq i,l\neq i} w_{kl} x_k x_l \qquad h_i = \sum_{j\neq i} w_{ij} x_j$$



Define the cavity distribution $p_{\setminus i} = \frac{\exp(-E_{\setminus i})}{Z_{\setminus i}}$ the distribution on the model where spin i and all connections w_{ij}, $j \neq i$ have been removed.

We can then write[46]

$$\langle x_i \rangle = \frac{\langle \sinh h_i \rangle_{\setminus i}}{\langle \cosh h_i \rangle_{\setminus i}} \tag{5}$$

where $\langle \cdot \rangle_{\setminus i}$ denotes average with respect to $p_{\setminus i}$.

46

$$Z = \sum_{x \setminus i} \sum_{x_i} \exp(x_i h_i) \exp(-E_{\setminus i}) = \sum_{x \setminus i} 2 \cosh h_i \exp(-E_{\setminus i}) = 2 \langle \cosh h_i \rangle_{\setminus i} Z_{\setminus i}$$

$$\langle x_i \rangle = \frac{1}{Z} \sum_{x \setminus i} \sum_{x_i} x_i \exp(x_i h_i) \exp(-E_{\setminus i}) = \frac{1}{Z} 2 \langle \sinh h_i \rangle_{\setminus i} Z_{\setminus i} = \frac{\langle \sinh h_i \rangle_{\setminus i}}{\langle \cosh h_i \rangle_{\setminus i}}$$

Cavity method

We write h_i as a mean contribution and fluctuations

$$h_i = \langle h_i \rangle_{\setminus i} + u_i. \tag{6}$$

By construction $\langle u_i \rangle_{\setminus i} = 0$ and we further assume that u_i is symmetrically distributed under $p_{\setminus i}$: $p_{\setminus i}(-u_i) = p_{\setminus i}(u_i)$.

Substituting Eq. 6 in Eq. 5 we obtain [47]

$$\langle x_i \rangle = \tanh(\langle h_i \rangle_{\setminus i}). \tag{7}$$

This is the main result of the cavity method. It states that the expected firing rate of spin i only depends on the expected value of the local field computed *in the absence of spin i*.

[47] Since $p_{\setminus i}(-u_i) = p_{\setminus i}(u_i)$ we get $\langle e^{-u_i} \rangle_{\setminus i} = \int du_i p(u_i) e^{-u_i} = \langle e^{u_i} \rangle_{\setminus i}$ and

$$\begin{aligned}
\langle \sinh(h_i) \rangle_{\setminus i} &= \frac{1}{2} e^{\langle h_i \rangle_{\setminus i}} \langle e^{u_i} \rangle_{\setminus i} + \frac{1}{2} e^{-\langle h_i \rangle_{\setminus i}} \left\langle e^{-u_i} \right\rangle_{\setminus i} = \sinh(\langle h_i \rangle_{\setminus i}) \langle \exp u_i \rangle_{\setminus i} \\
\langle \cosh(h_i) \rangle_{\setminus i} &= \cosh(\langle h_i \rangle_{\setminus i}) \langle \exp u_i \rangle_{\setminus i}
\end{aligned}$$

Application of cavity method to SK model

As before, we consider an ensemble of random coupling matrices w_{ij} where each coupling is drawn from a Gaussian distribution with mean J_0/N and variance J^2/N. $\langle h_i \rangle_{\setminus i}$ in Eq. 7 becomes now a Gaussian random variable with

$$\overline{\langle h_i \rangle_{\setminus i}} = \sum_j \overline{w_{ij}} \left\langle x_j \right\rangle_{\setminus i} = J_0 m \qquad m = \frac{1}{N} \sum_j \left\langle x_j \right\rangle_{\setminus i}$$

$$\mathrm{Var}(\langle h_i \rangle_{\setminus i}) = \sum_{j \neq i} \left\langle x_j \right\rangle_{\setminus i}^2 \mathrm{Var}(w_{ij}) = J^2 q \qquad q = \frac{1}{N} \sum_j \left\langle x_j \right\rangle_{\setminus i}^2$$

Note that the mean and variance are independent of i to leading order in N.

In the replica symmetric approximation it is then assumed that m, q are also equal to their quenched averages $\overline{\langle x_i \rangle}$ and $\overline{\langle x_i \rangle^2}$, respectively.

Application of cavity method to SK model

The quenched average of Eq. 7 results in a Gaussian integral. The result is identical to the replica symmetric solution[48]

$$
\begin{aligned}
m &= \overline{\langle x_i \rangle} = \overline{\tanh(\langle h_i \rangle_{\backslash i})} = \int dh \mathcal{N}(h|J_0 m, J^2 q) \tanh(h) \\
q &= \overline{\langle x_i \rangle^2} = \overline{\tanh^2(\langle h_i \rangle_{\backslash i})} = \int dh \mathcal{N}(h|J_0 m, J^2 q) \tanh^2(h)
\end{aligned}
$$

[48] Define $z = \frac{h - J_0 m}{J\sqrt{q}}$. Then

$$
\begin{aligned}
m &= \frac{1}{\sqrt{2\pi} J\sqrt{q}} \int dh e^{-\frac{(h-J_0 m)^2}{2J^2 q}} \tanh(h) = \frac{1}{\sqrt{2\pi}} \int dz e^{-z^2/2} \tanh(J\sqrt{q} z + J_0 m) \\
q &= \frac{1}{\sqrt{2\pi} J\sqrt{q}} \int dh e^{-\frac{(h-J_0 m)^2}{2J^2 q}} \tanh^2(h) = \frac{1}{\sqrt{2\pi}} \int dz e^{-z^2/2} \tanh^2(J\sqrt{q} z + J_0 m) = 1 - \frac{1}{\sqrt{2\pi}} \int dz e^{-z^2/2} \mathrm{sech}^2(J\sqrt{q} z
\end{aligned}
$$

Random satisfiability [Mezard et al., 2002]

One can describe the ground state energy structure of 3-sat in terms of 'states', which are ergodic components.

- 2d Ising model at high temperature is in a single (paramagnetic) state. Correlations between spins i, j that are separated a distance d_{ij} decay as $\chi_{ij} \propto e^{-d_{ij}/\xi}$ with ξ a finite correlation length. Far away spins are uncorrelated.

- 2d Ising model at low temperature has two states (ergodic components). In each state correlations between spins decay as $\chi_{ij} \propto e^{-d_{ij}/\xi}$. But in the symmetric mixture $p(x) = \frac{1}{2}(p_+(x) + p_-(x))$ correlations do not decay with distance.

- Complex optimization problems and frustrated sytems are characterized by many states at different energies E: $\mathcal{N}(E) \propto e^{N\Sigma(E)}$ with $\Sigma(E)$ the so-called complexity. [49]

In all cases, a state is a set of configurations x that are connected by finite (ie. not $\mathcal{O}(N)$) number of spin flips.

[49] The above paramagnetic and ferromagnetic systems have complexity zero.

The RS solution assumes that the system has a single state. Broken RS corresponds to systems having multiple states.

Random satisfiability [Mezard et al., 2002]

Define $\alpha = M/N$. K-sat has many solutions when α is small (SAT phase) and no solutions when α is large (UNSAT phase). Random 3-sat can be analyzed with the replica and cavity method.

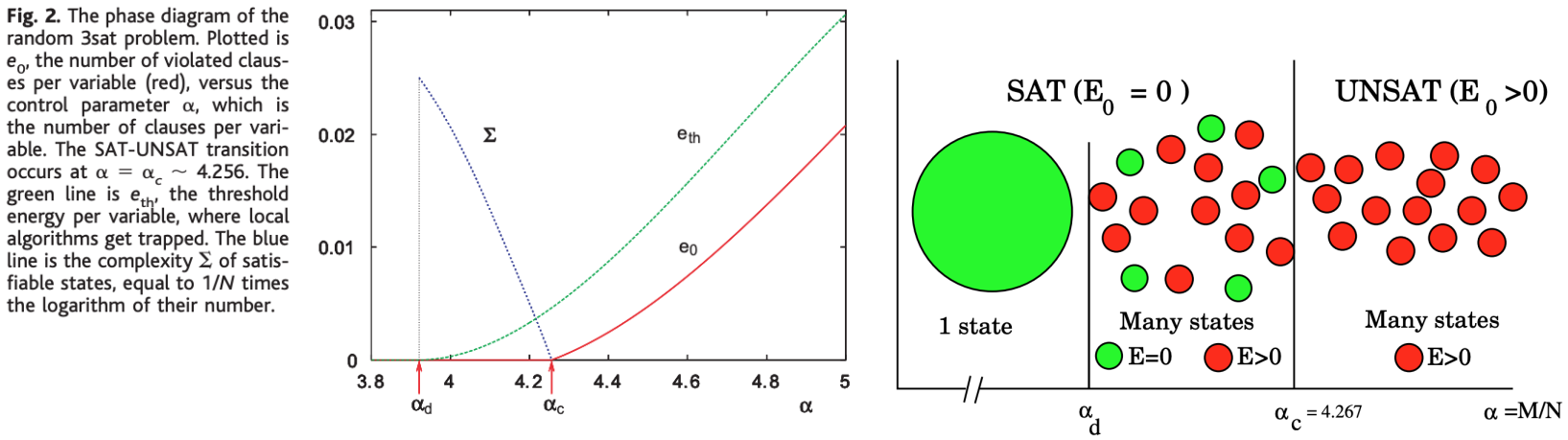


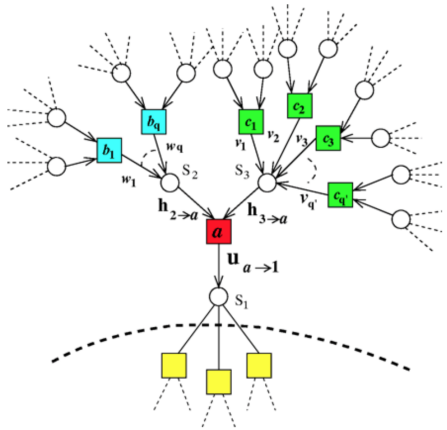
Fig. 2. The phase diagram of the random 3sat problem. Plotted is e_0, the number of violated clauses per variable (red), versus the control parameter α, which is the number of clauses per variable. The SAT-UNSAT transition occurs at $\alpha = \alpha_c \sim 4.256$. The green line is e_{th}, the threshold energy per variable, where local algorithms get trapped. The blue line is the complexity Σ of satisfiable states, equal to $1/N$ times the logarithm of their number.

(Red) Lowest energy $e_0 = E_0/N$ versus α. When $e_0 > 0$ the problem is UNSAT. (Green). Lowest energy $e_{\text{th}} = E_{\text{th}}/N$ that can be obtained with local algorithms. (Blue) Number of states $e^{N\Sigma}$ with $E = 0$.

When $\alpha_d < \alpha < \alpha_c$, the clauses cause frustration, similar to the SK spin glass: there are local minima either $E = 0$ or $E > 0$ and it is hard to find a $E = 0$ solution using local methods (such as iterative improvement or simulated annealing).

Random satisfiability [Mezard et al., 2002]

The simplest algorithm to solve the 3-sat problem is to use the max product (or equivalently max sum) on the factor graph. Because of the zero-one nature of the energy, each message $m_{i\to a}(x_i)$ or $m_{a\to i}(x_i)$ has only two values ± 1.



The survey propagation algorithm generalizes this to three values $(-1, 0, 1)$. Furthermore, a message is a distribution over these three values.

The SID (survey inspired decimation) algorithm is an iterative prodedure where in each iteration a subset of variables are clamped to ± 1 based on the survey progapation result. In the hard regime ($\alpha = 4.2$) SID confirms the solution on existing benchmarks for $N = 2000$. SID obtains solutions up to instances of size $N = 100.000$ where no other method can be appied. The complexity of SID is quadratic in N.

See [Mezard and Zecchina, 2002] for details.

Compressed sensing

Given an unknown signal which is a N-dimensional vector $x = (x_1, \ldots, x_N)$, we make M measurements $y_i = \sum_{j=1}^{N} F_{ij} x_j$. For instance, measurements of Fourier modes or wavelet coefficients. The observer knows the $M \times N$ matrix F and the measurements y. His aim is to reconstruct x.

When $M = N$ the solution is obtained by matrix inversion: $x = F^{-1} y$.

When $M < N$ the problem is underdetermined and there are many solutions.

Compressed sensing

Compressed sensing deals with the case where $M = \alpha N$ with $\alpha < 1$ and the signal s (and thus the to be reconstructed x) is sparse, in the sense that only $R = \rho_0 N$ of its components are non-zero with $\rho_0 < 1$.

In this case, exact signal reconstruction is possible whenever there are more measurements than non-zero components of x: $R < M$, using an exhaustive enumeration method. Assume $x_j \neq 0$ for $j \in A$, a subset of size R, and zero otherwise. Then we can try to solve

$$y_i = \sum_{j \in A} F_{ij} x_j$$

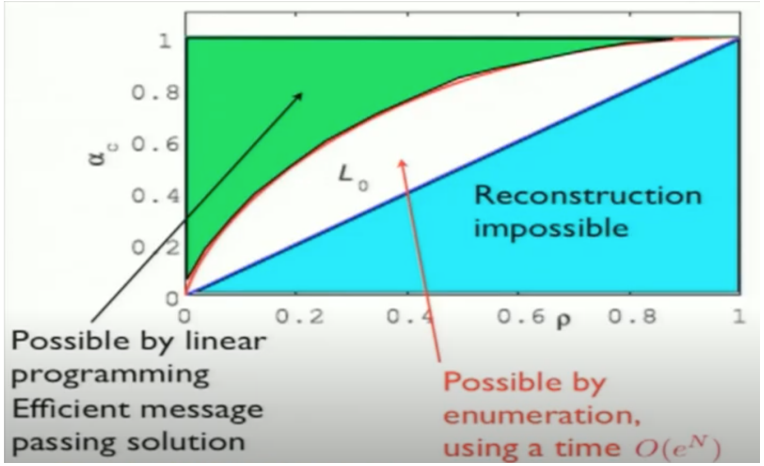
which is an overdetermined system of M equations with $R < M$ unknowns. For arbitrary subset A, the system has no solution, except when A has exactly the non-zero elements of the solution x. In this case we find the solution by inverting. This method requires to evaluate $\binom{N}{R}$ problems, which is exponential in N and thus impossible in practice.

Compressed sensing [Candes and Tao, 2005, Donoho, 2006]

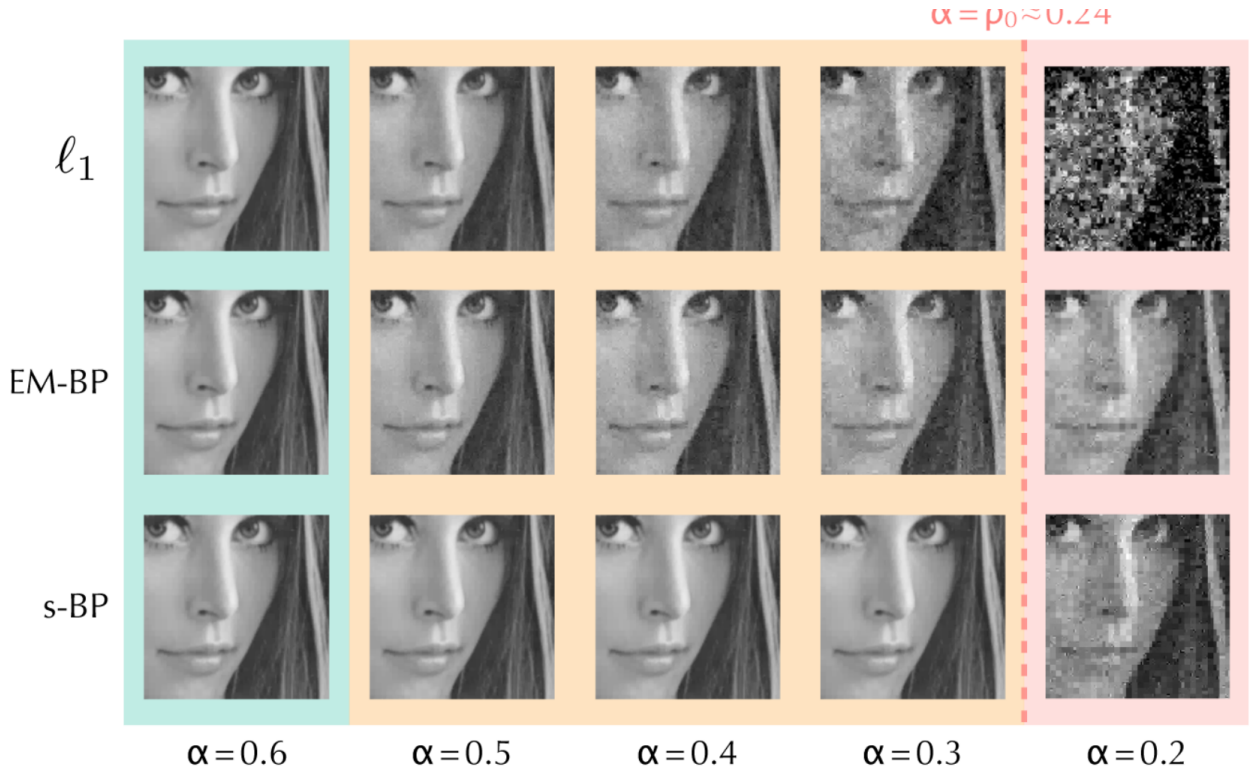
A popular approximate approach to compressed sensing is to find the solution of the following optimization problem

$$\min_{x, y=Fx} \|x\|_p \qquad \|x\|_p = \sum_{i=1}^{N} |x_i|^p$$

The true solution is found by using the $p = 0$ norm, which is the number of non-zero components in the vector x. However, this is as intractable as the previous exhaustive enumeration scheme. A practical algorithm is obtained by taking the $p = 1$ norm. This defines a convex optimization problem that can be solved efficiently.

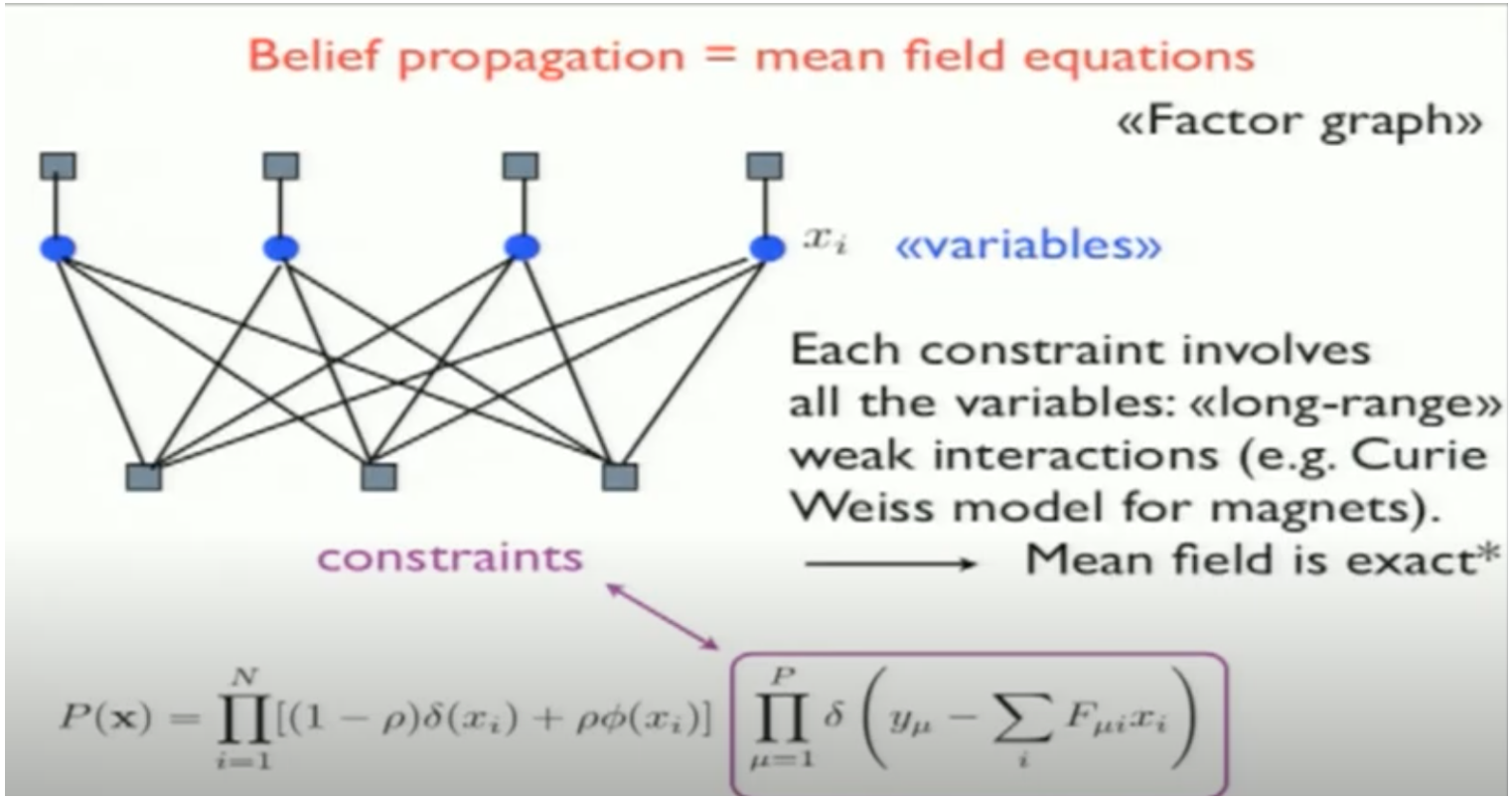


Compressed sensing



$N = (128)^2$ image o is transformed using the Haar wavelet projection $x = Wo$ which is sparse by construction with $R = \rho_0 N$ non-zero components and $\rho_0 = 0.24$. From x, the M measurements $y = Fx$ are constructed with $M = \alpha N$. The figure shows reconstruction of $o = W^{-1}x$ for various α and various methods.

Compressed sensing [Krzakala et al., 2012]



Propose BP on the probability model with Gauss-Bernouilli prior and the likelihood implements the constraints[50]. The messages are distributions over a continous variable and are approximated as Gaussian distributions.

[50] $P = M$.

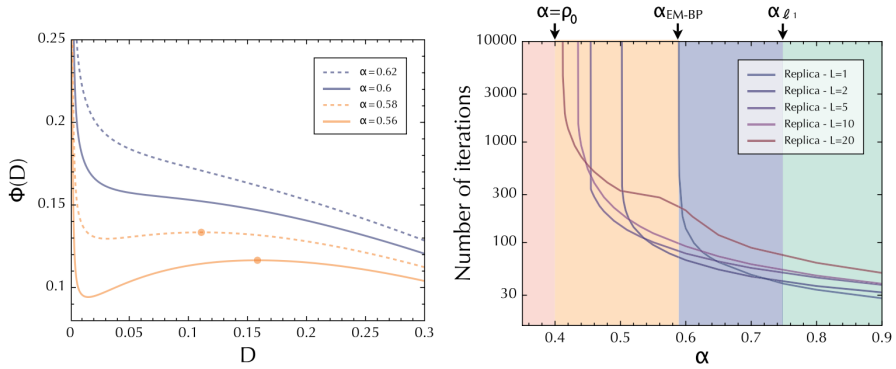
Compressed sensing [Krzakala et al., 2012]

One can analyse the average case behaviour of BP for this problem. In particular, using the replica method, one can compute the posterior probability

$$p(D) = \int dx P(x) \delta\left(D - \frac{1}{N}\sum_i (x_i - s_i)^2\right) = e^{N\Phi(D)}$$

that the reconstructed x is a distance D from the solution s. Furthermore, BP is gradient flow on $\Phi(D)$.

Compressed sensing [Krzakala et al., 2012]



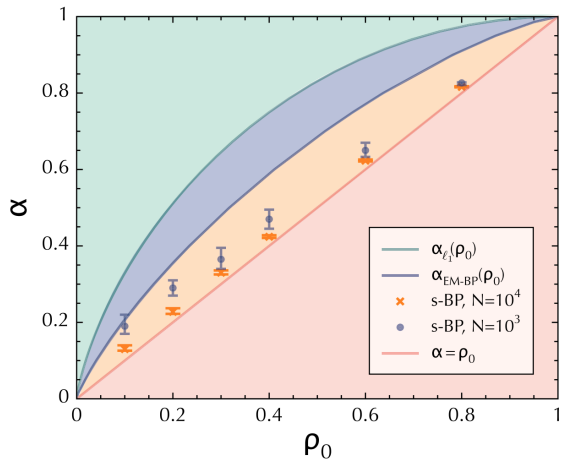


FIG. 3. When sampling from the probability $\hat{P}(\mathbf{x})$, in the limit of large N, the probability that the reconstructed signal x is at a squared distance $D = \sum_i (x_i - s_i)^2/N$ from the original signal s is written as $ce^{N\Phi(D)}$, where c is a constant and $\Phi(D)$ is the free entropy. Left: $\Phi(D)$ for a Gauss-Bernoulli signal with $\rho_0 = \rho = 0.4$ in the case of an unstructured measurement matrix $\mathbf{F}$ with independent random Gaussian-distributed elements. The evolution of the EM-BP algorithm is basically a steepest ascent in $\Phi(D)$ starting from large values of D. It goes to the correct maximum at $D = 0$ for large values of α, but it is blocked in the local maximum that appears for $\alpha < \alpha_{\text{EM-BP}}(\rho_0 = 0.4) \approx 0.59$. For $\alpha < \rho_0$, the maximum is not at $D = 0$, and exact inference is impossible. The seeding matrix $\mathbf{F}$, leading to the s-BP algorithm, succeeds in eliminating this local maximum. Right: Convergence time of the EM-BP and s-BP algorithms obtained through the replica analysis for $\rho_0 = 0.4$. The EM-BP convergence time diverges as $\alpha \to \alpha_{\text{EM-BP}}$ with the standard $L = 1$ matrices. The s-BP strategy allows one to go beyond the threshold: Using $\alpha_1 = 0.7$ and increasing the structure of the seeding matrix (here, $L = 2, 5, 10, 20$), we approach the limit $\alpha = \rho_0$. (Details of the parameters are given in Appendix E.)

Case of $\rho_0 = 0.4$. When $\alpha > \alpha_c = 0.59$ the gradient flow is towards $D = 0$ and $x = s$ is recovered. When $\alpha < \alpha_c$, BP gets stuck at $D > 0$ ($L = 1$).

Using the seeding F (like growing a cristal from an initial seed), a solution $D = 0$ is obtained for $\alpha > \alpha_c$ with $\alpha_c \to \rho_0$ for $N \to \infty$ (compare blue and red dots).

Compressed sensing [Krzakala et al., 2012]

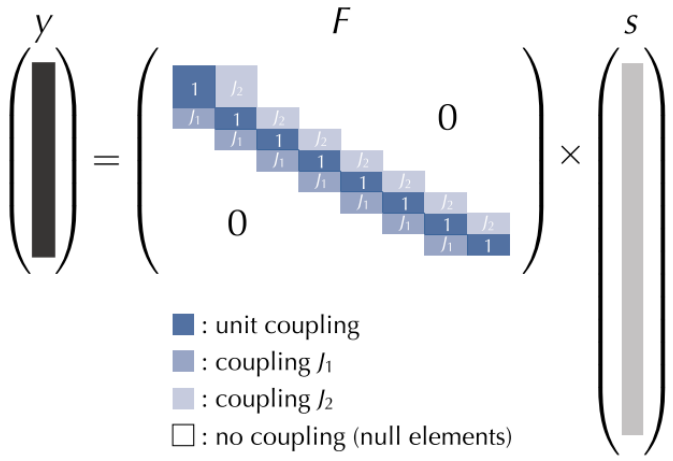


FIG. 4. Construction of the measurement matrix $\mathbf{F}$ for seeded compressed sensing. The elements of the signal vector are split into L (here, $L = 8$) equal-sized blocks; the number of measurements in each block is $M_p = \alpha_p N/L$ (here, $\alpha_1 = 1$, $\alpha_p = 0.5$ for $p = 2, \ldots, 8$). The matrix elements $F_{\mu i}$ are chosen as random Gaussian variables with variance $J_{q,p}/N$ if variable i is in the block p and measurement μ is in the block q. In the s-BP algorithm, we use $J_{p,q} = 0$, except for $J_{p,p} = 1$, $J_{p,p-1} = J_1$, and $J_{p-1,p} = J_2$. Good performance is typically obtained with relatively large J_1 and small J_2.

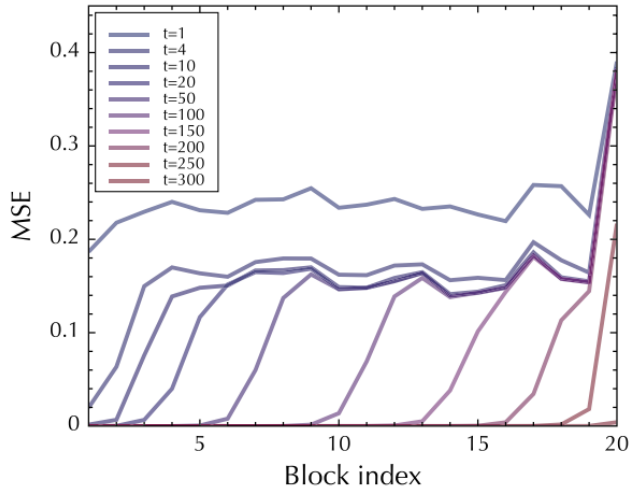
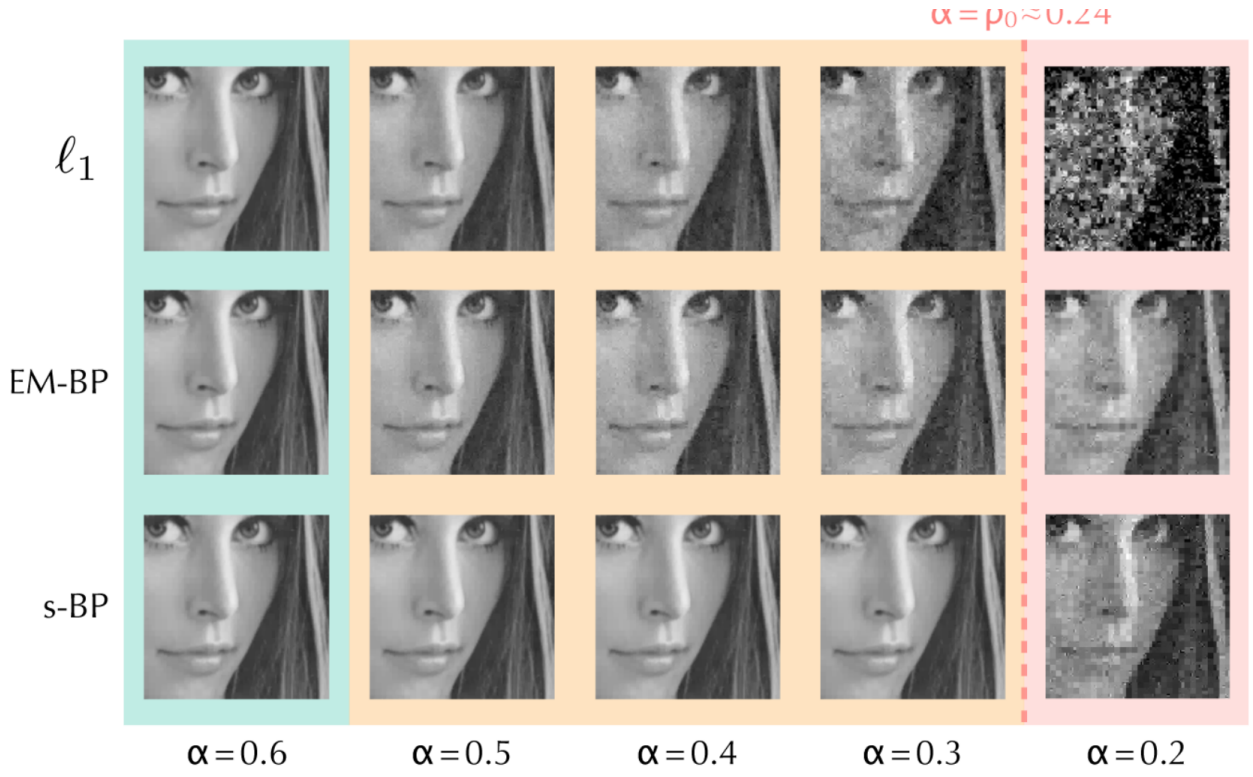


FIG. 5. Evolution of the mean-squared error at different times as a function of the block index for the s-BP algorithm. Exact reconstruction first appears in the left block whose rate $\alpha_1 > \alpha_{\text{EM-BP}}$ allows for seeded nucleation. It then propagates gradually block by block, driven by the free-entropy difference between the metastable and equilibrium states. After little more than 300 iterations, the whole signal of density $\rho_0 = 0.4$ and size $N = 50\,000$ is exactly reconstructed well inside the zone forbidden for BP (see Fig. 2). Here, we used $L = 20$, $J_1 = 20$, $J_2 = 0.2$, $\alpha_1 = 1.0$, $\alpha = 0.5$.

Compressed sensing [Krzakala et al., 2012]



$N = (128)^2$ image o is transformed using the Haar wavelet projection $x = Wo$ which is sparse by construction with $R = \rho_0 N$ non-zero components and $\rho_0 = 0.24$. From x, the M measurements $y = Fx$ are constructed with $M = \alpha N$. The figure shows reconstruction of $o = W^{-1}x$ for various α and various methods. s-BP reconstructs without errors up to the minimal $\alpha_c = \rho_0$.

See also this talk `https://www.youtube.com/watch?v=wjit44Klh3M`

Summary

The two main statistical physics methods for analysing machine learning problems are the replica method and the cavity method.

The replica symmetric (RS) assumption is valid when the system is in a single state.

- Examples are the SK model in the paramagnetic state, the 3-sat problem in the SAT phase $\alpha < \alpha_d$ or compressed sensing for $\alpha > \alpha_{EM-BP}(\rho_0)$.
- Local methods such as MF and BP are accurate.

When the system has multiple states, local methods start to fail.

- Examples are the (SK) model in the SG phase, the 3-sat problem for $\alpha_d < \alpha < \alpha_c$ and compressed sensing for $\rho_0 < \alpha < \alpha_{EM-BP}(\rho_0)$.
- In this case more complex methods, such as survey propagation (3-sat) or seeding (compressed sensing) must be used.

Boltzmann Machine

- Simplest case: no hidden units
  - Estimating the gradient with Monte Carlo sampling or mean field theory
  - Salamander retina data
- Restricted Boltzmann Machine (RBM)
  - contrastive divergence
  - Compressed sensing

Boltzmann Machines

The basic idea is to treat Boltzmann-Gibbs distribution of the Ising model as a statistical model

$$p(s|w) = \frac{1}{Z} \exp\left(\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} s_i s_j + \sum_{i=1}^{n} \theta_i s_i \right)$$

and use standard statistical tools to estimate its parameters.

Simplest case: no hidden units.

Learning: given a set of P *training patterns* $s^\mu = (s_1^\mu, \ldots, s_n^\mu)$ with $\mu = 1, \ldots, P$, find w, θ such that the $p(s|w, \theta)$ 'best' describes these data.

$$L(w, \theta) = \frac{1}{P} \sum_{\mu} \log p(s_1^\mu, \ldots, s_n^\mu | w, \theta)$$

and maximize this function wrt to w and θ.

We compute the gradients of L wrt w_{ij} and θ_i:

$$\begin{aligned}
\log p(s^\mu|q) &= \sum_{i>j} w_{ij} s_i^\mu s_j^\mu + \sum_i \theta_i s_i^\mu - \log Z \\
\frac{\partial \log p(s^\mu|w)}{\partial \theta_i} &= s_i^\mu - \frac{\log Z}{\partial \theta_i} = s_i^\mu - \langle s_i \rangle \\
\frac{\partial \log p(s^\mu|w)}{\partial w_{ij}} &= s_i^\mu s_j^\mu - \frac{\log Z}{\partial w_{ij}} = s_i^\mu s_j^\mu - \left\langle s_i s_j \right\rangle \\
\frac{\partial L}{\partial \theta_i} &= \frac{1}{P} \sum_\mu \frac{\partial \log p(s^\mu|w)}{\partial \theta_i} = \langle s_i \rangle_c - \langle s_i \rangle \\
\frac{\partial L}{\partial w_{ij}} &= \frac{1}{P} \sum_\mu \frac{\partial \log p(s^\mu|w)}{\partial w_{ij}} = \left\langle s_i s_j \right\rangle_c - \left\langle s_i s_j \right\rangle
\end{aligned}$$

with clamped statistics $\langle s_i \rangle_c = \frac{1}{P} \sum_\mu s_i^\mu, \left\langle s_i s_j \right\rangle_c = \frac{1}{P} \sum_\mu s_i^\mu s_j^\mu$ computed from the data and free statistics

$$\langle s_i \rangle = \sum_s s_i p(s), \quad \left\langle s_i s_j \right\rangle = \sum_s s_i s_j p(s)$$

computed from the Boltzmann Machine model.

Learning by gradient ascent:

$$\begin{aligned} w_{ij} &:= w_{ij} + \eta\left(\left\langle s_i s_j \right\rangle_c - \left\langle s_i s_j \right\rangle\right) \\ \theta_i &:= \theta_i + \eta(\langle s_i \rangle_c - \langle s_i \rangle) \end{aligned}$$

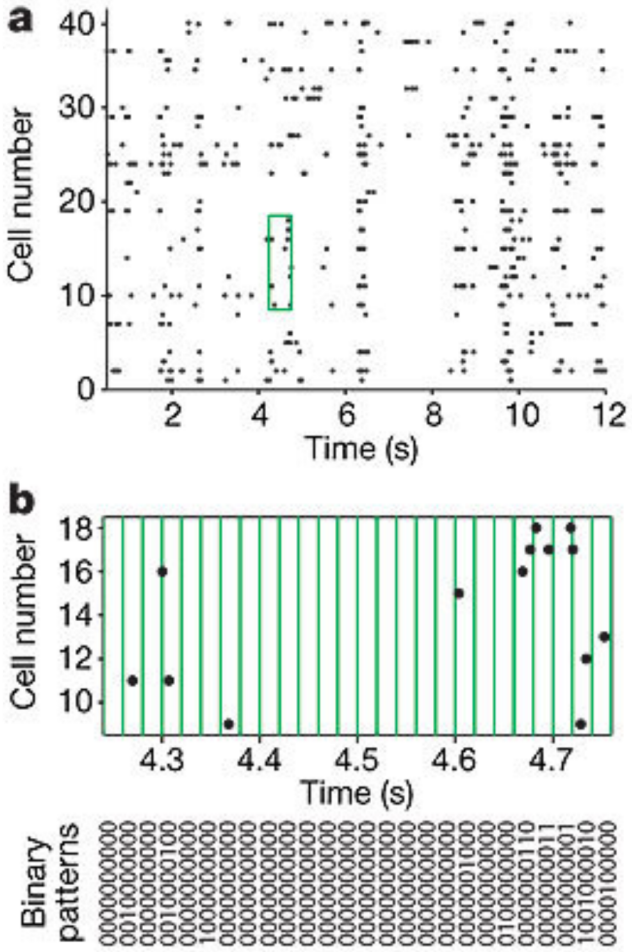
with $\eta > 0$ a small learning rate.

Learning terminates when the gradients are zero. When converged, w_{ij}, θ_i are such that the first and second order statistics of the Boltzmann distribution p and of the data are equal.

$$\langle s_i \rangle = \langle s_i \rangle_c \qquad \left\langle s_i s_j \right\rangle = \left\langle s_i s_j \right\rangle_c$$

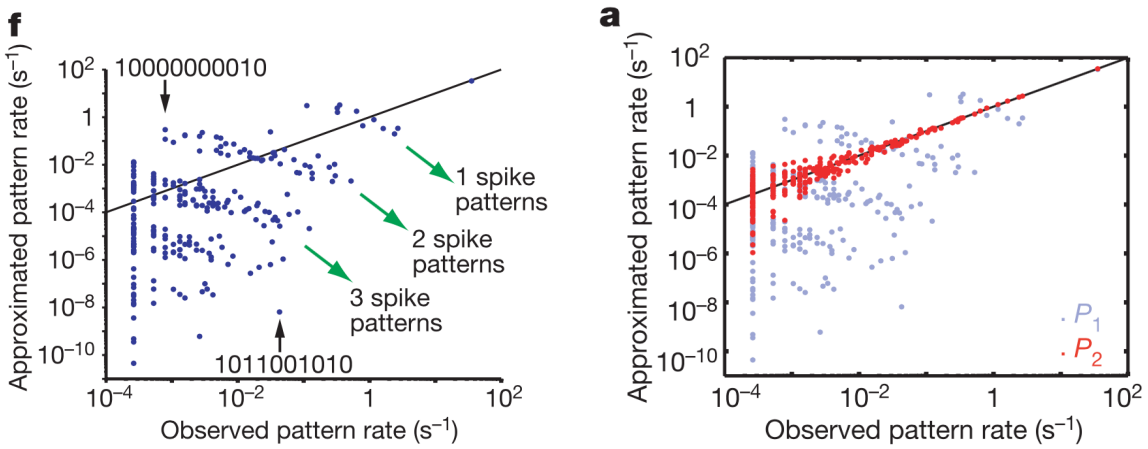
Note, that learning requires evaluation of the free statistics in each iteration, which is intractable for large models.

Neuron spike data from salamander retina



Schneidmann 2006

Neuron spike data from salamander retina



Empirical frequency of neural activity patterns versus frequency in models P1, P2.

P2 is the BM maximum likelihood solution. P1 is the maximum likelihood solution for the independent spin model ($w_{ij} = 0$).

$KL(q|p)$ with $q(s) = \frac{1}{P}\sum_\mu \delta_{s,s^\mu}$ the empirical data distribution and p the BM distribution.

$$KL(q|p_{P1}) \approx 0.1 \qquad KL(q|p_{P2}) \approx 0.001$$

Schneidmann 2006

Boltzmann Machine learning using Monte Carlo sampling

For large models the free expectations can be estimated using Monte Carlo sampling

1: Compute $\langle s_i \rangle_c$, $\left\langle s_i s_j \right\rangle_c$ from the data.
2: Start with a random initial state w_{ij}, θ_i
3: **for** $t = 1, 2, \ldots$ **do**
4: Estimate $\langle s_i \rangle$, $\left\langle s_i s_j \right\rangle$ using MH sampling
5: $\theta_i := \theta_i + \eta(\langle s_i \rangle_c - \langle s_i \rangle)$
6: $w_{ij} := w_{ij} + \eta(\left\langle s_i s_j \right\rangle_c - \left\langle s_i s_j \right\rangle)$
7: **end for**

The MH algorithm requires a burn-in in principle. However, since the sampling problem does not change much between different iterations of learning, burn-in can be ignored.

Boltzmann Machine learning using the MF and LR approximation [Kappen and Rodríguez, 1998]

For large models the free expectations can be estimated using the mean field and linear response approximations.

1: Compute $\langle s_i \rangle_c$, $\left\langle s_i s_j \right\rangle_c$ from the data.
2: Start with a random initial state w_{ij}, θ_i
3: **for** $t = 1, 2, \ldots$ **do**
4: Estimate $\langle s_i \rangle$, $\left\langle s_i s_j \right\rangle$ using MF and LR approximation
5: $\theta_i := \theta_i + \eta(\langle s_i \rangle_c - \langle s_i \rangle)$
6: $w_{ij} := w_{ij} + \eta(\left\langle s_i s_j \right\rangle_c - \left\langle s_i s_j \right\rangle)$
7: **end for**

The MF approximation for $m_i = \langle s_i \rangle$ is given by the solution of the MF equations $m_i = \tanh\left(\sum_j w_{ij} m_j + \theta_i\right)$.

The LR approximation for $\left\langle s_i s_j \right\rangle = \chi_{ij} + m_i m_j$ is given by

$$\chi_{ij} = A^{-1}_{ij} \qquad A_{ij} = \frac{\delta_{ij}}{1 - m_i^2} - w_{ij}$$

Directly solving the fixed point equations [Kappen and Rodríguez, 1998]

For a BM without hidden units, one can also directly solve the fixed point equations in the mean field and linear response approximation. Thus, no learning is needed.

From $\frac{\partial L}{\partial \theta_i} = 0$ we compute $m_i = \langle s_i \rangle = \langle s_i \rangle_c$.

Define

$$C_{ij} = \left\langle s_i s_j \right\rangle_c - \langle s_i \rangle_c \left\langle s_j \right\rangle_c \qquad \chi = \left\langle s_i s_j \right\rangle - \langle s_i \rangle \left\langle s_j \right\rangle$$

From $\frac{\partial L}{\partial w_{ij}} = 0$ we obtain

$$\left\langle s_i s_j \right\rangle = \left\langle s_i s_j \right\rangle_c \qquad \chi_{ij} = C_{ij} \qquad \chi^{-1}_{ij} = C^{-1}_{ij}$$

From the LR approximation we have $\chi^{-1}_{ij} = \frac{\delta_{ij}}{1-m_i^2} - w_{ij}$. Thus,

$$w_{ij} = \frac{\delta_{ij}}{1 - m_i^2} - \left(C^{-1}\right)_{ij}$$

From MF equations we compute θ:

$$\theta_i = \tanh^{-1}(m_i) - \sum_{j=1}^{n} w_{ij} m_j$$

Hidden units

It is interesting to consider a BM where data is only available for a subset of the neurons. The hidden units increase the set of distributions that can be modelled on the visible units.

Denote visible units by $x_i, i = 1, \ldots, n$ and hidden units by $h_a, a = 1, \ldots, m$. The BM is

$$p(x,h) = \frac{1}{Z} \exp\left(\sum_{i,j=1}^{n} w_{ij} x_i x_j + \sum_{i=1}^{n} \sum_{a=1}^{m} w_{ia} x_i h_a + \sum_{a,b=1}^{m} w_{ab} h_a h_b + \sum_{i=1}^{n} \theta_i x_i + \sum_{a=1}^{m} \theta_a h_a \right)$$

The log likelihood is

$$L = \frac{1}{P} \sum_{\mu} \log p(x^\mu) = \frac{1}{P} \sum_{\mu} \log \sum_{h} p(x^\mu, h)$$

It involves an ugly sum over h.

Restricted BM

A popular special case is the so-called restricted BM (RBM) where $w_{ab} = 0$ (and often also $w_{ij} = 0$):. Then

$$
\begin{aligned}
p(x,h) &= \frac{1}{Z}\exp\left(\sum_{i,j=1}^{n} w_{ij}x_i x_j + \sum_{i=1}^{n}\theta_i x_i\right)\prod_{a=1}^{m} e^{h_a f_a(x)} \\
p(x) &= \sum_h p(x,h) = \frac{1}{Z}\exp\left(\sum_{i,j=1}^{n} w_{ij}x_i x_j + \sum_{i=1}^{n}\theta_i x_i\right)\prod_{a=1}^{m} 2\cosh(f_a(x)) \\
p(h|x) &= \frac{p(x,h)}{p(x)} = \prod_{a=1}^{m}\frac{e^{h_a f_a(x)}}{2\cosh f_a(x)}
\end{aligned}
$$

with $f_a(x) = \sum_{i=1}^{n} w_{ia}x_i + \theta_a$.

For given x the hidden units are uncorrelated. Their expected values

$$\langle h_a\rangle_x = \tanh f_a(x) \qquad \langle h_a h_b\rangle_x = \langle h_a\rangle_x \langle h_b\rangle_x$$

Restricted BM

The gradients are easily computed:

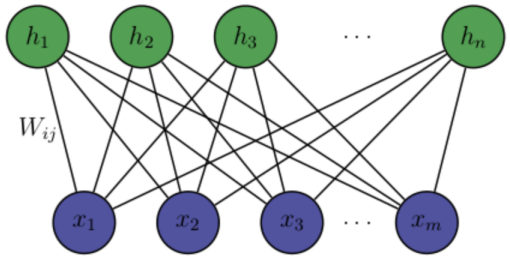
$$\begin{aligned}
\frac{\partial L}{\partial \theta_i} &= \langle x_i \rangle_c - \langle x_i \rangle \\
\frac{\partial L}{\partial w_{ij}} &= \left\langle x_i x_j \right\rangle_c - \left\langle x_i x_j \right\rangle \\
\frac{\partial L}{\partial \theta_a} &= \langle h_a \rangle_c - \langle h_a \rangle \\
\frac{\partial L}{\partial w_{ia}} &= \langle x_i h_a \rangle_c - \langle x_i h_a \rangle
\end{aligned}$$

with $\langle h_a \rangle_c = \frac{1}{P} \sum_{\mu=1}^{P} \langle h_a \rangle_{x^\mu}$ and $\langle x_i h_a \rangle_c = \frac{1}{P} \sum_{\mu=1}^{P} x_i^\mu \langle h_a \rangle_{x^\mu}$ and the free statistics are computed in the full model.

NB The complexity of the learning algorithm is similar as in the case of no hidden units: computing the clamped statistics is linear in the size of the data and the free statistics is exponential in the number of variables.

Contrastive divergence [Hinton, 2002]

When $w_{ij} = 0$ in addition to $w_{ab} = 0$, the RBM forms a bipartite graph.



Thus, given x, the h_a are independent and given h, the x_i are independent:

$$p(h|x) = \prod_{a=1}^{m} p(h_a|x) \qquad p(x|h) = \prod_{i=1}^{n} p(x_i|h)$$

The computation of the free statistics is intractable but can be computed efficiently using the idea of contrastive divergence (CD) which alternates the following steps:
1. sample $h_1, \ldots, h_m$ given $x_1, \ldots, x_n$
2. sample $x_1, \ldots, x_n$ given $h_1, \ldots, h_m$

This is iterated a small number of T times from which the free statistics are computed.

RBM for collaborative filtering [Salakhutdinov et al., 2007]

The collaborative filtering problem is: given N users that have each seen and ranked some of M movies, predict whether any of the users will like a movie that he/she has not yet seen.

The data matrix X is $N \times M$ with N, M very large. X has many missing values. For instance, for the Netflix data set there are $N = 480.189$ users and $M = 17.770$ movies and $100.480.507$ ratings. Thus the fraction of data in X is $100.480.507/(N * M) = 0.0118$.

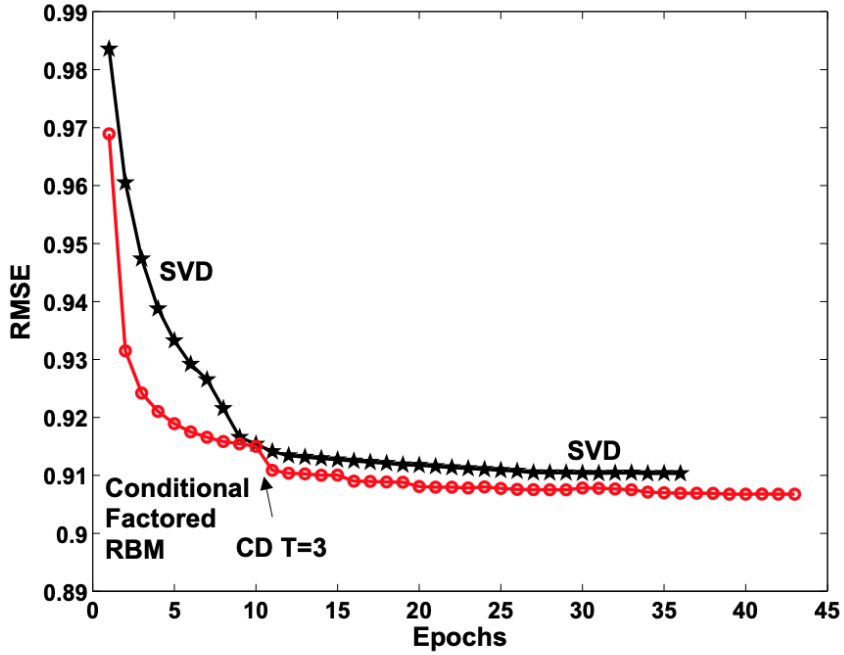
A common approach is to use singular value decomposition and write $X = UV$ with U a $N \times C$ matrix and V a $C \times M$ matrix and $C \ll N, M$.

An alternative method is to use a RBM (as discussed above with also $w_{ij} = 0$). Each row $X_{\mu,:}, \mu = 1, \ldots, N$ is a training sample.

The computation of the clamped statistics is complicated by the missing data. One simply computes these by ignoring the missing values.

The free statistics are estimated using contrastive divergence.

RBM for collaborative filtering [Salakhutdinov et al., 2007]



Comparison of RBM with SVD on Netflix data. Training data as above. Error is on test set data. This is part of the Netflix competition. Netflix posted the problem and offered a prize of $1.000.000 for an algorithm that could improve 10 % on their own RMSE=0.95. The prize was won in 2009 by a team with RMSE=0.85 using many different machine learning methods and additional heuristics.

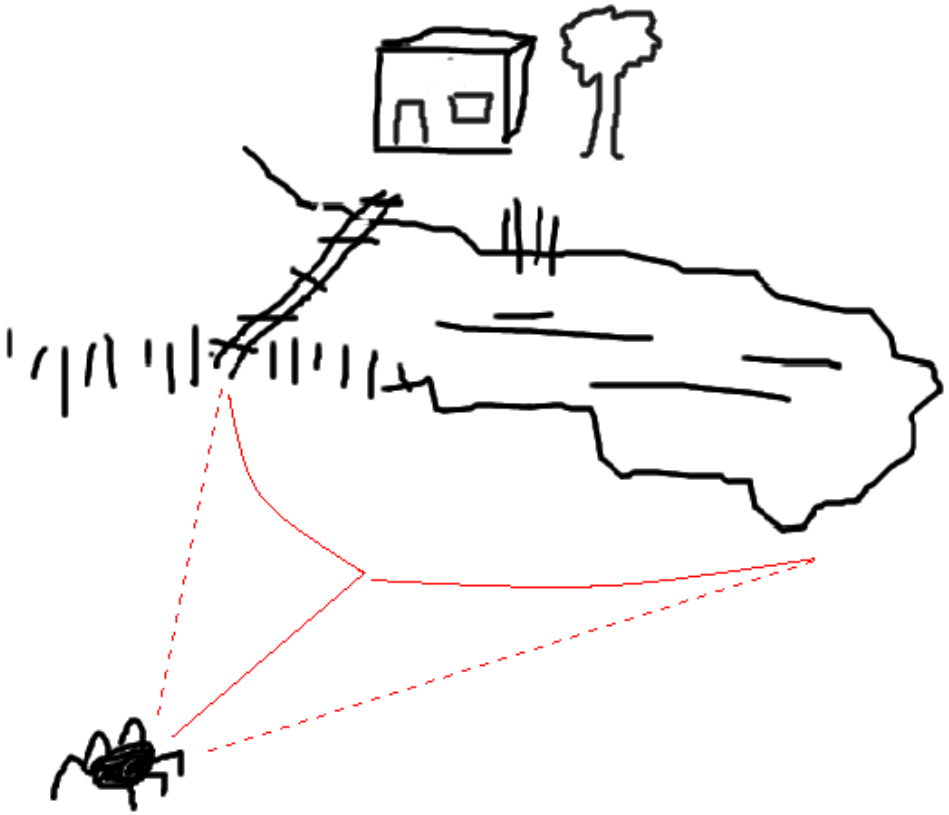
Quantum Machine learning

- Introduction
- Quantum Boltzmann machine
  - Density matrices, von Neumann entropy, cross-entropy
  - QBM learning rule
  - Salamander retina data
  - Circuit implementation

Optimal control theory

Given a current state and a future desired state, what is the best/cheapest/fastest way to get there.

Why stochastic optimal control?



Material

- H.J. Kappen. Optimal control theory and the linear Bellman Equation. In *Inference and Learning in Dynamical Models (Cambridge University Press 2010)*, edited by David Barber, Taylan Cemgil and Sylvia Chiappa
  `http://www.snn.ru.nl/~bertk/control/timeseriesbook.pdf`

- Dimitri Bertsekas, Dynamic programming and optimal control

- `http://www.snn.ru.nl/~bertk/machinelearning/`

Introduction

Optimal control theory: Optimize sum of a path cost and end cost. Result is optimal control sequence and optimal trajectory.

Input: Cost function.
Output: Optimal trajectory and controls.

Introduction

Control problems are delayed reward problems:

- Motor control: devise a sequence of motor commands to reach a goal
- Finance: devise a sequence of buy/sell commands to maximize profit
- Learning, exploration vs. exploitation

Discrete time control

Consider the control of a discrete time deterministic dynamical system:

$$x_{t+1} = x_t + f(t, x_t, u_t), \quad t = 0, 1, \ldots, T-1$$

x_t describes the *state* and u_t specifies the *control* or *action* at time t.

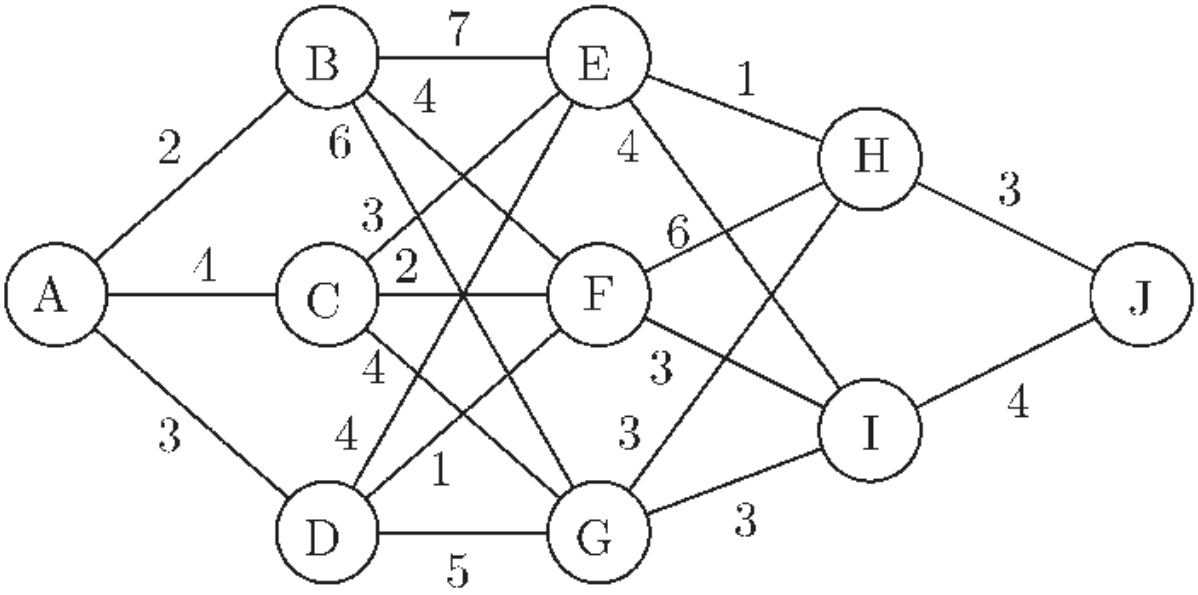
Given $x_{t=0} = x_0$ and $u_{0:T-1} = u_0, u_1, \ldots, u_T - 1$, we can compute $x_{1:T}$.

Define a cost for each sequence of controls:

$$C(x_0, u_{0:T-1}) = \phi(x_T) + \sum_{t=0}^{T-1} R(t, x_t, u_t)$$

The problem of optimal control is to find the sequence $u_{0:T-1}$ that minimizes $C(x_0, u_{0:T-1})$.

Dynamic programming



Find the minimal cost path from A to J.

$$C(J) = 0, C(H) = 3, C(I) = 4$$

$$C(F) = \min(6 + C(H), 3 + C(I))$$

Discrete time control

The optimal control problem can be solved by dynamic programming. Introduce the *optimal cost-to-go*:

$$J(t, x_t) = \min_{u_{t:T-1}} \left(\phi(x_T) + \sum_{s=t}^{T-1} R(s, x_s, u_s) \right)$$

which solves the optimal control problem from an intermediate time t until the fixed end time T, for all intermediate states x_t.

Then,

$$\begin{aligned} J(T, x) &= \phi(x) \\ J(0, x) &= \min_{u_{0:T-1}} C(x, u_{0:T-1}) \end{aligned}$$

Discrete time control

One can recursively compute $J(t, x)$ from $J(t+1, x)$ for all x in the following way:

$$\begin{aligned}
J(t, x_t) &= \min_{u_{t:T-1}} \left(\phi(x_T) + \sum_{s=t}^{T-1} R(s, x_s, u_s) \right) \\
&= \min_{u_t} \left(R(t, x_t, u_t) + \min_{u_{t+1:T-1}} \left[\phi(x_T) + \sum_{s=t+1}^{T-1} R(s, x_s, u_s) \right] \right) \\
&= \min_{u_t} \left(R(t, x_t, u_t) + J(t+1, x_{t+1}) \right) \\
&= \min_{u_t} \left(R(t, x_t, u_t) + J(t+1, x_t + f(t, x_t, u_t)) \right)
\end{aligned}$$

This is called the *Bellman Equation*.

Computes u as a function of x, t for all intermediate t and all x.

Discrete time control

The algorithm to compute the optimal control $u^*_{0:T-1}$, the optimal trajectory $x^*_{1:T}$ and the optimal cost is given by

1. Initialization: $J(T, x) = \phi(x)$

2. Backwards: For $t = T - 1, \ldots, 0$ and for all x compute

$$\begin{aligned} u^*_t(x) &= \arg\min_u \{R(t, x, u) + J(t + 1, x + f(t, x, u))\} \\ J(t, x) &= R(t, x, u^*_t) + J(t + 1, x + f(t, x, u^*_t)) \end{aligned}$$

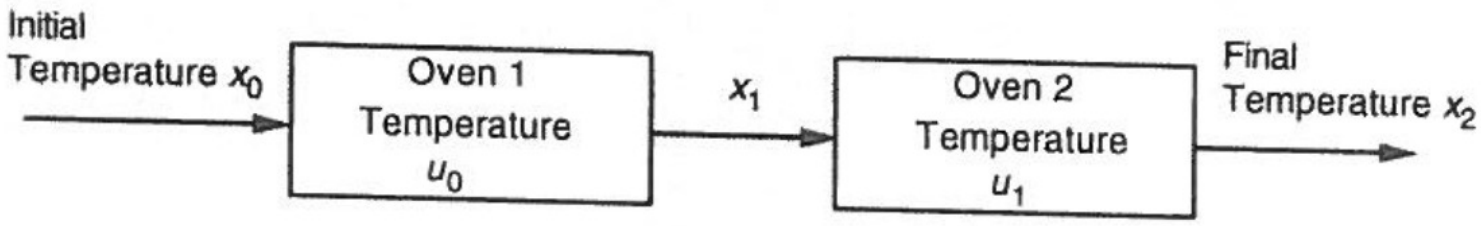
3. Forwards: For $t = 0, \ldots, T - 1$ compute

$$x^*_{t+1} = x^*_t + f(t, x^*_t, u^*_t(x^*_t))$$

NB: the backward computation requires $u^*_t(x)$ for all x.

Exercise: Two ovens

A certain material is passed through a sequence of two ovens. Aim is to reach pre-specified final product temperature x^* with minimal oven energy.



$x_{0,1,2}$ are the product temperatures initially, after pasing through oven 1 and after passing through oven 2. $u_{0,1}$ are the oven temperatures. The dynamics is

$$\begin{aligned} x_{t+1} &= (1-a)x_t + au_t \qquad t = 0, 1 \\ C &= r(x_2 - x^*)^2 + u_0^2 + u_1^2 \end{aligned}$$

- Find the optimal control solution u_0, u_1.
- Show that adding mean zero noise to the dynamics ($x_{t+1} = (1-a)x_t + au_t + w_t$ with $\langle w_t \rangle = 0$), does not change the optimal control solution.

Example: Two ovens

End cost-to-go is $J(2, x_2) = r(x_2 - x^*)^2$.

$$
\begin{aligned}
J(1, x_1) &= \min_{u_1} \left(u_1^2 + J(2, x_2) \right) = \min_{u_1} \left(u_1^2 + r((1-a)x_1 + au_1 - x^*)^2 \right) \\
u_1 &= \mu_1(x_1) = \frac{ra(x^* - (1-a)x_1)}{1 + ra^2} \\
J(1, x_1) &= \frac{r((1-a)x_1 - x^*)^2}{1 + ra^2} \\
J(0, x_0) &= \min_{u_0} \left(u_0^2 + J(1, x_1) \right) = \min_{u_0} \left(u_0^2 + \frac{r((1-a)x_1 - x^*)^2}{1 + ra^2} \right) \\
&= \min_{u_0} \left(u_0^2 + \frac{r((1-a)((1-a)x_0 + au_0) - x^*)^2}{1 + ra^2} \right) \\
u_0 &= \mu_0(x_0) = \frac{r(1-a)a(x^* - (1-a)^2 x_0)}{1 + ra^2(1 + (1-a)^2)} \\
J(0, x_0) &= \frac{r((1-a)^2 x_0 - x^*)^2}{1 + ra^2(1 + (1-a)^2)}
\end{aligned}
$$

Comments

- **Linear Quadratic Control:** Solution can be obtained in closed form because problem is linear quadratic.

- **Certainty equivalence:** Optimal control solution is unaffected by noise:

$$\begin{aligned} x_{t+1} &= (1-a)x_t + au_t + w_t \qquad t = 0, 1 \\ C &= r(x_2 - x^*)^2 + u_0^2 + u_1^2 \end{aligned}$$

with $\langle w_t \rangle = 0$.Then

$$\begin{aligned} J(1, x_1) &= \min_{u_1} \left(u_1^2 + \left\langle r((1-a)x_1 + au_1 + w_1 - x^*)^2 \right\rangle \right) \\ &= \min_{u_1} \left(u_1^2 + r((1-a)x_1 + au_1 - x^*)^2 + r \langle w_1 \rangle^2 \right) \end{aligned}$$

Continuous limit

Replace $t+1$ by $t+dt$ with $dt \to 0$.

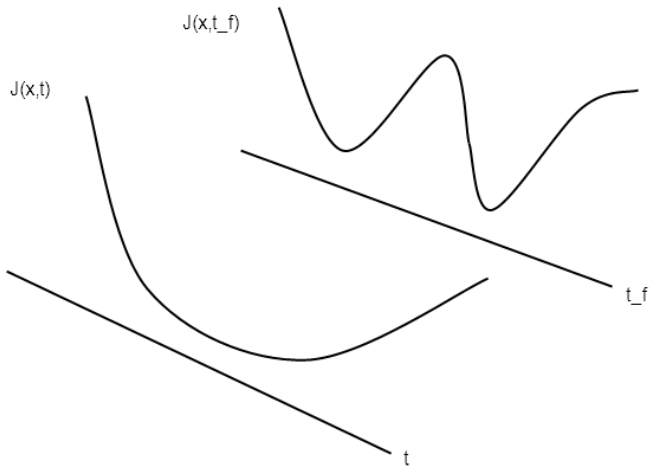
$$\begin{aligned} x_{t+dt} &= x_t + f(x_t, u_t, t)dt \\ C(x_0, u_{0\to T}) &= \phi(x_T) + \int_0' d\tau R(\tau, x(\tau), u(\tau)) \end{aligned}$$

Assume $J(x,t)$ is smooth.

$$\begin{aligned} J(t,x) &= \min_u (R(t,x,u)dt + J(t+dt, x+f(x,u,t)dt)) \\ &\approx \min_u (R(t,x,u)dt + J(t,x) + \partial_t J(t,x)dt + \partial_x J(t,x) f(x,u,t)dt) \\ -\partial_t J(t,x) &= \min_u (R(t,x,u) + f(x,u,t)\partial_x J(x,t)) \end{aligned}$$

with boundary condition $J(x,T) = \phi(x)$.

Continuous limit



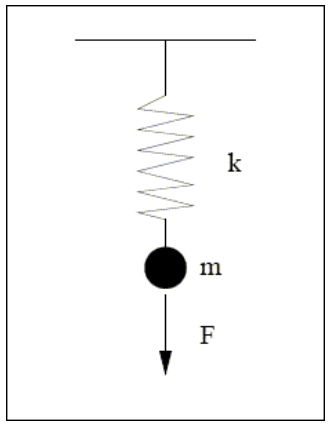
$$-\partial_t J(t,x) \quad = \quad \min_u \left(R(t,x,u) + f(x,u,t)\partial_x J(x,t)\right)$$

with boundary condition $J(x,T) = \phi(x)$.

This is called the *Hamilton-Jacobi-Bellman Equation*.

Computes the *anticipated potential* $J(t,x)$ from the future potential $\phi(x)$.

Example: Mass on a spring



The spring force $F_z = -z$ towards the rest position and control force $F_u = u$.

Newton's Law

$$F = -z + u = m\ddot{z}$$

with $m = 1$.

Control problem: Given initial position and velocity $z(0) = \dot{z}(0) = 0$ at time $t = 0$, find the control path $-1 < u(0 \to T) < 1$ such that $z(T)$ is maximal.

Example: Mass on a spring

Introduce $x_1 = z, x_2 = \dot{z}$, then

$$\begin{aligned} \dot{x_1} &= x_2 \\ \dot{x_2} &= -x_1 + u \end{aligned}$$

The end cost is $\phi(x) = -x_1$; path cost $R(x, u, t) = 0$.

The HJB takes the form:

$$\begin{aligned} -\partial_t J &= \min_u \left(x_2 \frac{\partial J}{\partial x_1} - x_1 \frac{\partial J}{\partial x_2} + \frac{\partial J}{\partial x_2} u \right) \\ &= x_2 \frac{\partial J}{\partial x_1} - x_1 \frac{\partial J}{\partial x_2} - \left| \frac{\partial J}{\partial x_2} \right|, \quad u = -\text{sign} \left(\frac{\partial J}{\partial x_2} \right) \end{aligned}$$

Example: Mass on a spring

We try $J(t,x) = \psi_1(t)x_1 + \psi_2(t)x_2 + \alpha(t)$. The HJBE reduces to the ordinary differential equations

$$\begin{aligned} \dot{\psi}_1 &= \psi_2 \\ \dot{\psi}_2 &= -\psi_1 \\ \dot{\alpha} &= -|\psi_2| \end{aligned}$$

These equations must be solved for all t, with final boundary conditions $\psi_1(T) = -1, \psi_2(T) = 0$ and $\alpha(T) = 0$.

Note, that the optimal control only requires $\partial_x J(x,t)$, which in this case is $\psi(t)$ and thus we do not need to solve α. The solution for ψ is

$$\begin{aligned} \psi_1(t) &= -\cos(t-T) \\ \psi_2(t) &= \sin(t-T) \end{aligned}$$

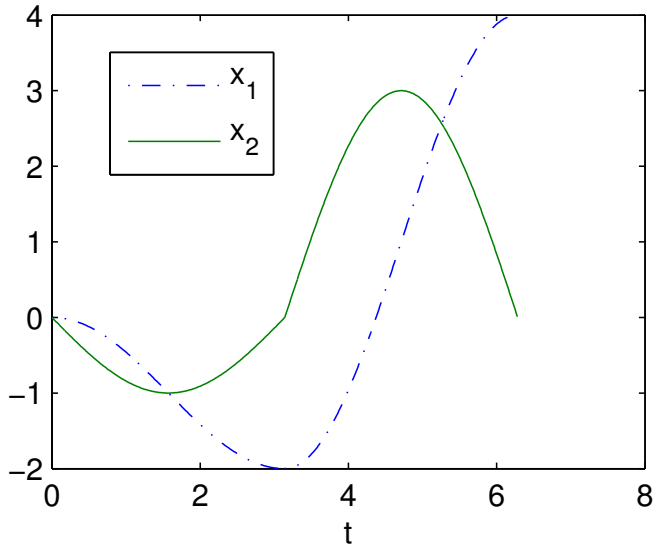
Example: Mass on a spring

The optimal control is

$$u(x,t) = -\text{sign}(\psi_2(t)) = -\text{sign}(\sin(t-T))$$

As an example consider $T = 2\pi$. Then, the optimal control is

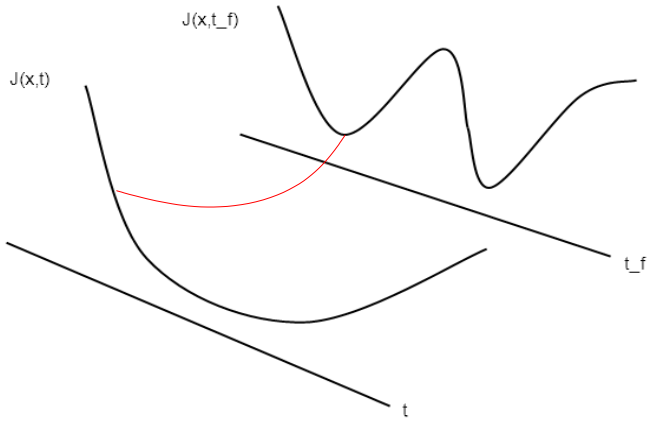
$$u = -1, \quad 0 < t < \pi$$
$$u = 1, \quad \pi < t < 2\pi$$



Pontryagin minimum principle

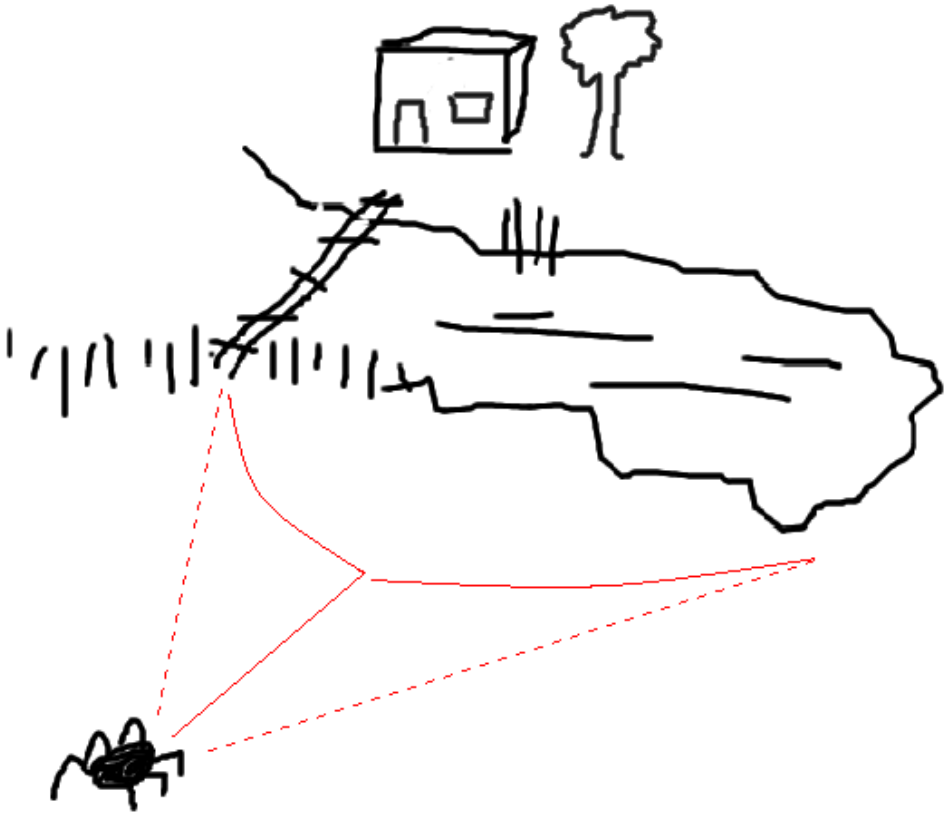
The HJB equation is a PDE with boundary condition at future time. The PDE is solved using discretization of space and time.

The solution is an optimal cost-to-go for all x and t. From this we compute the optimal trajectory and optimal control.



An alternative approach is a variational approach that directly finds the optimal trajectory and optimal control.

Stochastic control



Stochastic differential equations

Consider the random walk on the line:

$$X_{t+1} = X_t + \xi_t \qquad \xi_t = \pm 1$$

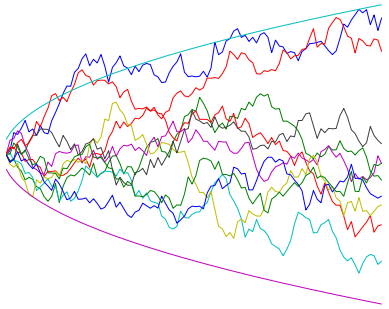
with $x_0 = 0$. We can compute

$$X_t = \sum_{i=1}^{'} \xi_i$$

Since x_t is a sum of random variables, x_t becomes Gaussian distributed with

$$\mathbb{E}x_t = \sum_{i=1}^{'} \mathbb{E}\xi_i = 0$$

$$\mathbb{V}x_t = \sum_{i,j=1}^{'} \mathbb{V}\xi_i = t$$



Note, that the fluctuations $\propto \sqrt{t}$.

Stochastic differential equations

In the continuous time limit we define

$$dX_t \quad = \quad X_{t+dt} - X_t = dW_t$$

with dW_t an infinitesimal mean zero Gaussian variable: $\mathbb{E}dW_t = 0, \mathbb{V}dW_t = \nu dt$.

Then with initial condition x_1 at t_1

$$X_t = x_1 + \int_{t_1}^{t_2} dW_s \qquad \mathbb{E}X_t = x_1 \qquad \mathbb{V}X_t = \nu t$$

is called a Wiener process or Brownian motion.

Since the increments are independent, X_t is Gaussian distributed

$$p(x_2, t_2|x_1, t_1) \quad = \quad \frac{1}{\sqrt{2\pi\nu(t_2 - t_1)}} \exp\left(-\frac{(x_2 - x_1)^2}{2\nu(t_2 - t_1)}\right)$$

Stochastic differential equations

Consider the stochastic differential equation

$$dX_t = f(X_t, t)dt + dW_t$$

W_t is a Wiener process.

In this case $\rho(x_2, t_2|x_1, t_1)$ may be very complex and is generally not known.

Define $\rho(x, t) = p(x, t|x_0, 0)$. Then (Fokker-Planck forward equation)

$$\partial_t \rho(x,t) \quad = \quad -\nabla(f(x,t)\rho(x,t)) + \frac{1}{2}\nu\nabla^2\rho(x,t), \qquad \rho(x,0) = \delta(x - x_0)$$

Define $\psi(x, t) = p(z, T|x, t)$. Then (Kolmogorov backward equation)

$$-\partial_t \psi(x,t) \quad = \quad f(x,t)\nabla\psi(x,t) + \frac{1}{2}\nu\nabla^2\psi(x,t) \qquad \psi(x,T) = \delta(z - x)$$

Example: Brownian motion

$$
\begin{aligned}
X_t &= x_0 + \int_0^t dW_s \\
\rho(x,t) &= p(x,t|x_0,0) = \frac{1}{\sqrt{2\pi\nu t}} \exp\left(-\frac{(x-x_0)^2}{2\nu t}\right) \\
\psi(x,t) &= p(z,T|x,t) = \frac{1}{\sqrt{2\pi\nu(T-t)}} \exp\left(-\frac{(x-z)^2}{2\nu(T-t)}\right)
\end{aligned}
$$

NB Verify that ρ and ψ are solutions to the forward and backward equations, respectively.

Stochastic optimal control

Consider a stochastic dynamical system

$$dX_t = f(t, X_t, u)dt + dW_t$$

W_t is a Wiener process with $\mathbb{E}dW_t^2 = \nu(t, x, u)dt$. [51]

The cost becomes an expectation:

$$C(t, x, u) \quad = \quad \mathbb{E}\left(\phi(X_T) + \int_t^T d\tau R(t, X_t, u(X_t, t))\right)$$

over all stochastic trajectories starting at x with control function $u(\cdot, t)$.

Optimize with respect to the set of functions $u(\cdot, t)$.

[51] Our notation is for one dimensional X, but the theory generalizes trivially to higher dimension.

Stochastic optimal control

We obtain the Bellman recursion

$$\begin{aligned} J(t, x_t) &= \min_{u_t} R(t, x_t, u_t)dt + \mathbb{E}J(t + dt, X_{t+dt}) \\ J(t + dt, x_t + dX_t) &= J(t, x_t) + dt\partial_t J(t, x_t) + dX_t\partial_x J(t, x_t) + \frac{1}{2}dX_t^2\partial_x^2 J(t, x_t) \\ \mathbb{E}J(t + dt, x_t + dX_t) &= J(t, x_t) + dt\partial_t J(t, x_t) + f dt\partial_x J(t, x_t) + \frac{1}{2}\nu dt\partial_x^2 J(t, x_t) \end{aligned}$$

because $\mathbb{E}dX_t = f dt$ and $\mathbb{E}dX_t^2 = \nu dt + (f dt)^2 = \nu dt + O(dt^2)$.

Thus (Stochastic Hamilton-Jacobi-Bellman equation)

$$-\partial_t J(t, x) = \min_u \left(R(t, x, u) + f(x, u, t)\partial_x J(x, t) + \frac{1}{2}\nu(t, x, u)\partial_x^2 J(x, t) \right)$$

with boundary condition $J(x, T) = \phi(x)$.

Linear Quadratic control

The dynamics is linear

$$dX_t = [A(t)X_t + B(t)u_t + b(t)]dt + \sum_{j=1}^{m}(C_j(t)X_t + D_j(t)u_t + \sigma_j(t))dW_j, \quad \left\langle dW_j dW_{j'} \right\rangle = \delta_{jj'}dt$$

The cost function is quadratic

$$\begin{aligned} \phi(x) &= \frac{1}{2}x'Gx \\ R(x,u,t) &= \frac{1}{2}x'Q(t)x + u'S(t)x + \frac{1}{2}u'R(t)u \end{aligned}$$

In this case the optimal cost-to-go is quadratic in x:

$$\begin{aligned} J(t,x) &= \frac{1}{2}x'P(t)x + \alpha'(t)x + \beta(t) \\ u_t &= -\Psi(t)x_t - \psi(t) \end{aligned}$$

Substitution in the HJB equation yields ODEs for P, α, β:

$$
\begin{aligned}
-\dot{P} &= PA + A'P + \sum_{j=1}^{m} C_j' P C_j + Q - \hat{S}'\hat{R}^{-1}\hat{S} \\
-\dot{\alpha} &= [A - B\hat{R}^{-1}\hat{S}]'\alpha + \sum_{j=1}^{m} [C_j - D_j\hat{R}^{-1}\hat{S}]' P\sigma_j + Pb \\
\dot{\beta} &= \frac{1}{2}\left|\sqrt{\hat{R}}\psi\right|^2 - \alpha' b - \frac{1}{2}\sum_{j=1}^{m} \sigma_j' P \sigma_j \\
\hat{R} &= R + \sum_{j=1}^{m} D_j' P D_j \\
\hat{S} &= B'P + S + \sum_{j=1}^{m} D_j' P C_j \\
\Psi &= \hat{R}^{-1}\hat{S} \\
\psi &= \hat{R}^{-1}(B'\alpha + \sum_{j=1}^{m} D_j' P \sigma_j)
\end{aligned}
$$

with $P(t_f) = G$ and $\alpha(t_f) = \beta(t_f) = 0$.

Example

Find the optimal control for the dynamics

$$\begin{aligned} dX_t &= udt + dW_t, \qquad \left\langle dW_t^2 \right\rangle = \nu dt \\ C &= \left\langle \frac{1}{2} G x(T)^2 + \int_0^{'} dt \frac{1}{2} u(x,t)^2 \right\rangle \end{aligned}$$

with end cost $\phi(x) = \frac{1}{2} G x^2$ and path cost $R(x,u) = \frac{1}{2} u^2$.
$(A = 0, B = 1, b = 0, C = D = 0, \sigma_j = \sqrt{\nu}, m = 1, \hat{R} = 1, \hat{S} = P, \Psi = P, \psi = \alpha)$

The Ricatti equations reduce to

$$\begin{aligned} \dot{P} &= P^2 \qquad P(T) = G \\ \dot{\alpha} &= P\alpha \qquad \alpha(T) = 0 \\ \dot{\beta} &= \frac{1}{2}\alpha^2 - \frac{1}{2}\nu P \end{aligned}$$

The solution is $\alpha(t) = 0$ and

$$P(t) = \frac{1}{c-t} \qquad \frac{1}{c-T} = G$$

and β not relevant.

$$u(x,t) = -P(t)x - \alpha(t) = -\frac{Gx}{1+G(T-t)}$$

Compare with deterministic case considered earlier, is identical due to certainty equivalence.

When $G \to \infty$ we obtain the Brownian bridge The control law and dynamics becomes

$$\begin{aligned} dx &= udt + d\xi \\ u &= \frac{-x(t_0)}{T - t_0} \end{aligned}$$

$x(T) \to 0$ w.p. 1.

Example

Find the optimal control for the dynamics

$$dX_t = udt + dW_t, \qquad \left\langle dW_t^2 \right\rangle = \nu dt$$

with end cost $\phi(x) = 0$ and path cost $R(x,u) = \frac{1}{2}(Qx^2 + Ru^2)$.

The Ricatti equations reduce to

$$\begin{aligned} -\dot{P} &= Q - R^{-1}P^2 \\ -\dot{\alpha} &= -R^{-1}P\alpha = 0 \\ \dot{\beta} &= -\frac{1}{2}\nu P \end{aligned}$$

with $P(T) = \alpha(T) = \beta(T) = 0$ and

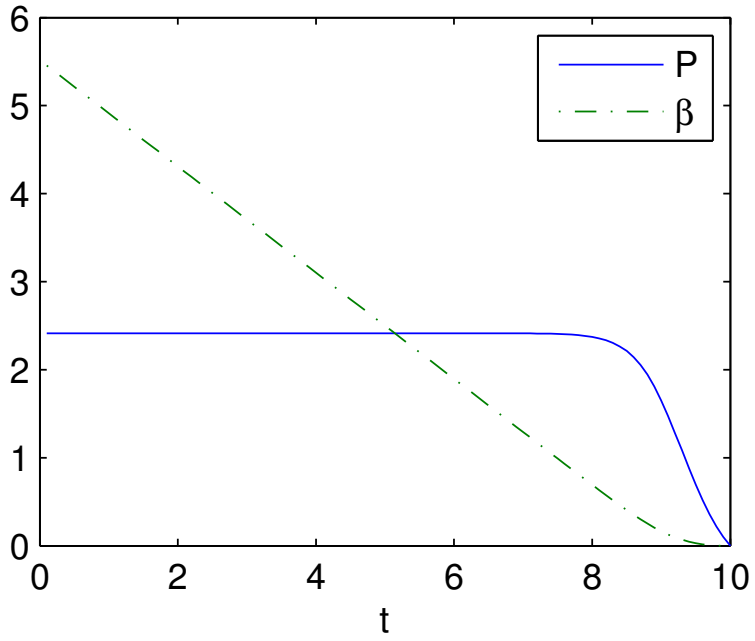
$$u(x,t) = -R^{-1}P(t)x$$

The solution is

$$
\begin{aligned}
P(t) &= \sqrt{RQ}\tanh\left(\sqrt{\frac{Q}{R}}(T-t)\right) \\
\alpha(t) &= 0 \\
\beta(t) &= \frac{1}{2}\nu R \log\cosh\left(\sqrt{\frac{Q}{R}}(T-t)\right) \\
\Psi(t) &= R^{-1}P(t) \qquad \psi(t) = 0
\end{aligned}
$$

The control is given by Eq. **??**:

$$u(x,t) = -R^{-1}P(t)x \tag{8}$$


P
β
t
0
1
2
3
4
5
6
0
2
4
6
8
10

Comments

Note, that in the last example the optimal control is independent of ν, i.e. optimal stochastic control equals optimal deterministic control.

In general:

- If $C_j = D_j = 0$ (only 'additive noise') $\dot{P}, \dot{\alpha}$ independent of noise σ, $\dot{\beta}$ depends on σ, but control independent of β. Thus control independent of σ (certainty equivalence)

- If $C_j \neq 0$ or $D_j \neq 0$, control depends on C_j, D_j, σ_j (no certainty equivalence)

Path integral control

The n-dimensional path integral control problem is defined as

$$\begin{aligned} dX_t &= f(X_t, t)dt + g(x, t)(u(X_t, t)dt + dW_t) \\ C(t, x, u) &= \mathbb{E}\left(\phi(X_T) + \int_t^T dsV(X_s, s) + \frac{1}{2}u'(X_s, s)Ru(X_s, s)\right) \end{aligned}$$

with $\mathbb{E}dW_t dW_t' = \nu dt$. g is $n \times m$ matrix, ν is $m \times m$ matrix and u, dW_t are m dimensional.

The cost is an expectation over all stochastic trajectories starting at x with control function $u(x, t)$.

The stochastic HJB equation becomes

$$-\partial_t J = \min_u \left(\frac{1}{2}u'Ru + V + (\nabla J)'(f + gu) + \frac{1}{2}\mathrm{Tr}\left(g\nu g'\nabla^2 J\right)\right)$$

which we need to solve with end boundary condition $J(x, t_f) = \phi(x)$ for all x.

Path integral control

Minimization wrt u yields: [52]

$$\begin{aligned} u &= -R^{-1} g' \nabla J \\ -\partial_t J &= -\frac{1}{2} (\nabla J)' g R^{-1} g' (\nabla J) + V + (\nabla J)' f + \frac{1}{2} \mathrm{Tr}\left(g \nu g' \nabla^2 J \right) \end{aligned}$$

Define $\psi(x,t)$ through $J(x,t) = -\lambda \log \psi(x,t)$ and impose a relation between R and ν:

$$R = \lambda \nu^{-1}$$

with λ a positive number.

[52] $u_a = -\sum_{b,i} \left(R^{-1}\right)_{ab} g_{ib}(x,t) \frac{\partial J(x,t)}{\partial x_i}$

Path integral control

Then the HJB becomes *linear* in ψ

$$-\partial_t \psi = \left(-\frac{V}{\lambda} + f'\nabla + \frac{1}{2}\mathrm{Tr}\left(g\nu g'\nabla^2\right)\right)\psi$$

with end condition $\psi(x,T) = \exp(-\phi(x)/\lambda)$ [53]

[53] We sketch the derivation for $n = m$ and $g_{ij}(x,t) = \delta_{ij}$.

$$-\frac{1}{2}(\nabla J)' R^{-1}(\nabla J) + \frac{1}{2}\mathrm{Tr}\left(\nu\nabla^2 J\right) = -\frac{1}{2}\sum_{ij}\nabla_i J R^{-1}_{ij}\nabla_j J + \frac{1}{2}\lambda\sum_{ij} R^{-1}_{ij}\nabla_{ij} J$$

$$= \frac{1}{2}\sum_{ij} R^{-1}_{ij}\left(-\nabla_i J\nabla_j J + \lambda\nabla_{ij} J\right)$$

$$= \frac{1}{2}\sum_{ij} R^{-1}_{ij}\left(-\lambda^2\frac{1}{\psi}\nabla_{ij}\psi\right)$$

since

$$-\nabla_i J\nabla_j J = -\lambda^2\frac{1}{\psi^2}\nabla_i\psi\nabla_j\psi$$

$$\nabla_{ij} J = -\lambda\nabla_i\nabla_j\log\psi = -\lambda\nabla_i\left(\frac{1}{\psi}\nabla_j\psi\right) = \lambda\frac{1}{\psi^2}\nabla_i\psi\nabla_j\psi - \lambda\frac{1}{\psi}\nabla_{ij}\psi$$

Path integral control

We identify $\psi(x,t) \propto p(z,T|x,t)$, then the linear Bellman equation

$$-\partial_t \psi = \left(-\frac{V}{\lambda} + f'\nabla + \frac{1}{2}\mathrm{Tr}\left(g\nu g'\nabla^2\right)\right)\psi$$

can be interpreted as a Kolmogorov backward equation for the process

$$\begin{aligned} dx_i &= f_i(x,t)dt + \sum_a g_{ia}(x,t)d\xi_a \\ x(t) &= \dagger \quad \text{with probability} \quad V(x,t)dt/\lambda \\ x(T) &= \dagger \quad \text{with probability} \quad \phi(x)/\lambda \end{aligned}$$

The correspondong forward equation is

$$\partial_t \rho = -\frac{V}{\lambda}\rho - \nabla(f\rho) + \frac{1}{2}\mathrm{Tr}\nabla^2 g\nu g'\rho$$

with $\rho(x,t) = p(x,t|z,0)$ and $\rho(x,0) = \delta(x-z)$.

Feynman-Kac formula

Denote $Q(\tau|x, s)$ the distribution over uncontrolled trajectories that start at x, t:

$$dx = f(x,t)dt + g(x,t)d\xi$$

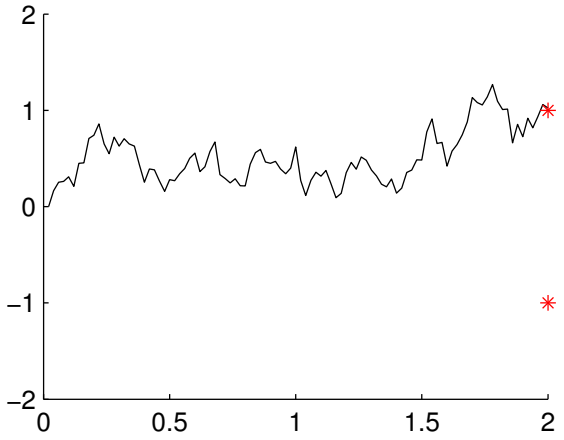
with τ a trajectory $x(t \to T)$. Then

$$\psi(x,t) = \int dQ(\tau|x,t) \exp\left(-\frac{S(\tau)}{\lambda}\right)$$

$$S(\tau) = \phi(x(T)) + \int_t' dsV(x(s), s)$$

ψ can be computed by forward sampling the uncontrolled process.

Recap of the main idea



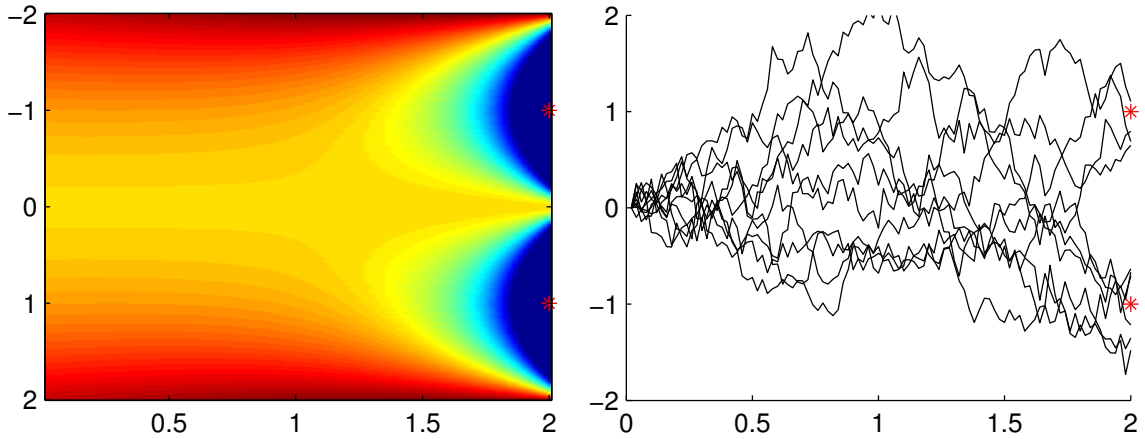
Consider a stochastic dynamical system

$$dX_t = f(X_t, u)dt + g(X_t)dW_t \qquad \mathbb{E}(dW_{t,i}dW_{t.j}) = \nu_{ij}dt$$

Given X_0 find control function $u(x,t)$ that minimizes the expected future cost

$$C = \mathbb{E}\left(\phi(X_T) + \int_0^T dtR(X_t, u(X_t,t))\right)$$

Control theory



Standard approach: define $J(x,t)$ is optimal cost-to-go from x,t.

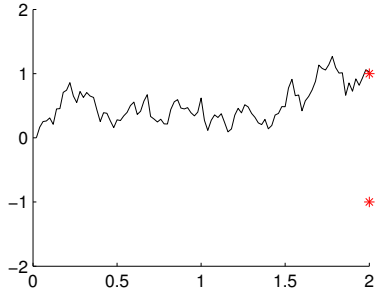
$$J(x,t) = \min u_{t:T} \mathbb{E}_u \left(\phi(X_T) + \int_t^T dt R(X_t, u(X_t,t)) \right) \qquad X_t = x$$

J satisfies a partial differential equation

$$-\partial_t J(t,x) \quad = \quad \min_u \left(R(x,u) + f(x,u)\nabla_x J(x,t) + \frac{1}{2}\mathrm{Tr}(g\nu g' \nabla_x^2 J(x,t)) \right) \quad J(x,T) = \phi(x)$$

with $u = u(x,t)$.This is HJB equation. Optimal control $u^*(x,t)$ defines distribution over trajectories $p^*(\tau)$ $(= p(\tau|x_0,0))$.

Path integral control theory



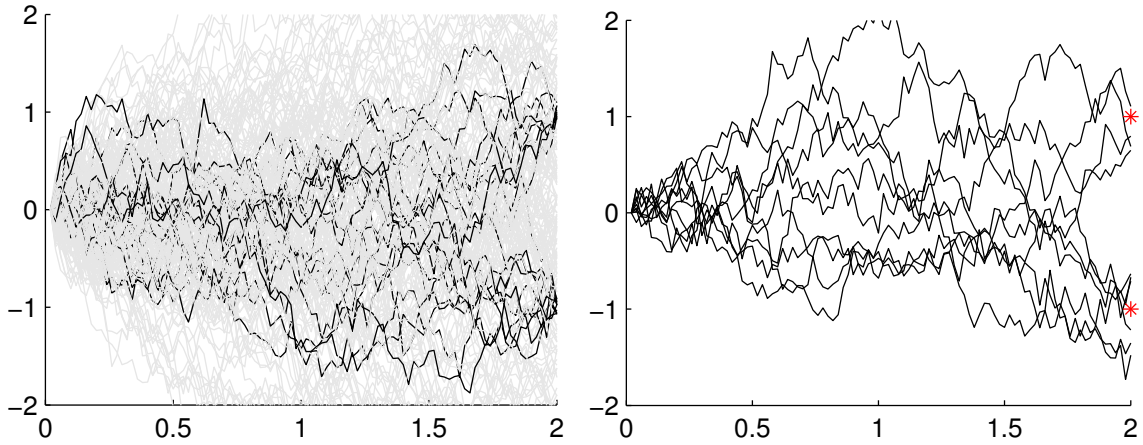
$$dX_t = \underbrace{f(X_t)dt + g(X_t)u(X_t,t)dt}_{f(X_t,u)dt} + g(X_t)dW_t \qquad X_0 = x_0$$

Goal is to find function $u(x,t)$ that minimizes

$$C = \mathbb{E}\left[\phi(X_T) + \int_0^T dt \underbrace{V(X_t,t) + \frac{1}{2}u(X_t,t)^2}_{R(X_t,u(X_t,t))}\right] = \mathbb{E}\left(S(\tau) + \int_0^T dt \frac{1}{2}u(X_t,t)^2\right)$$

$$S(\tau) = \phi(X_T) + \int_0^T V(X_t,t)$$

Path integral control theory



Equivalent formulation: Find distribution over trajectories p that minimizes [54]

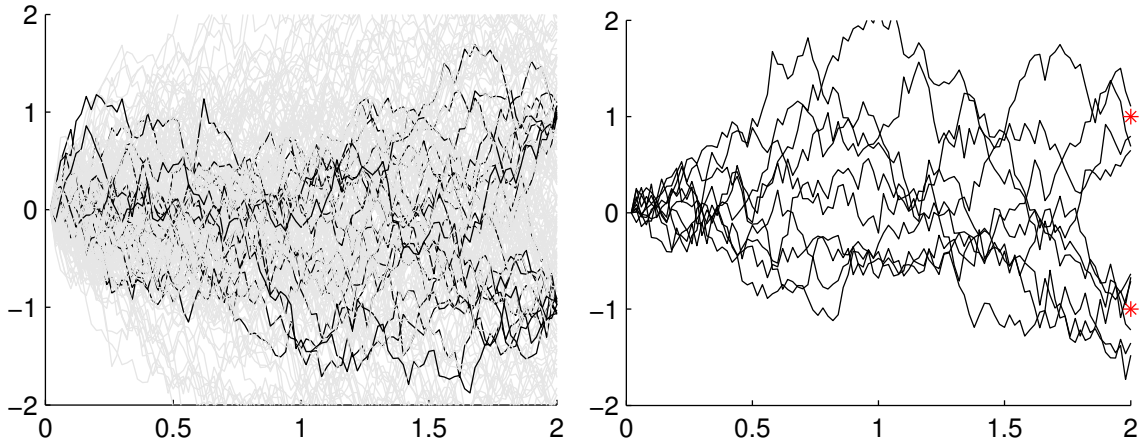
$$C(p) = \int d\tau p(\tau) \left(S(\tau) + \log \frac{p(\tau)}{q(\tau)} \right)$$

$q(\tau|x_0, 0)$ is distribution over *uncontrolled* trajectories.

The optimal solution is given by $p^*(\tau) = \frac{1}{\psi} q(\tau) e^{-S(\tau)}$

[54] $\mathbb{E}_u \int_0^T dt \frac{1}{2} u(X_t, t)^2 = \int d\tau p(\tau) \log \frac{p(\tau)}{q(\tau)}$.

Path integral control theory



Equivalent formulation: Find distribution over trajectories p that minimizes

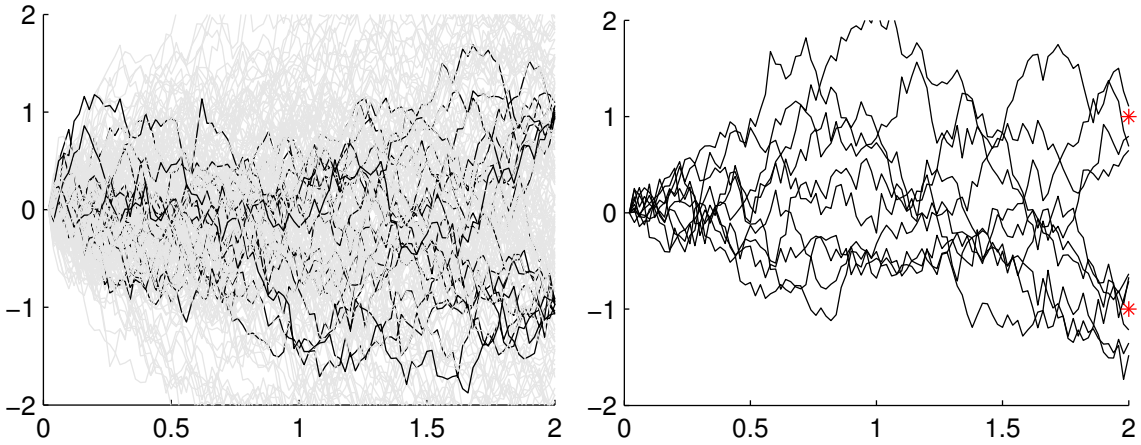
$$C(p) = \int d\tau p(\tau) \left(S(\tau) + \log \frac{p(\tau)}{q(\tau)} \right)$$

$q(\tau|x_0, 0)$ is distribution over *uncontrolled* trajectories.

The optimal solution is given by $p^*(\tau) = \frac{1}{\psi} q(\tau) e^{-S(\tau)} = p(\tau|u^*)$.

Equivalence of optimal control and discounted cost (Girsanov)

Path integral control theory



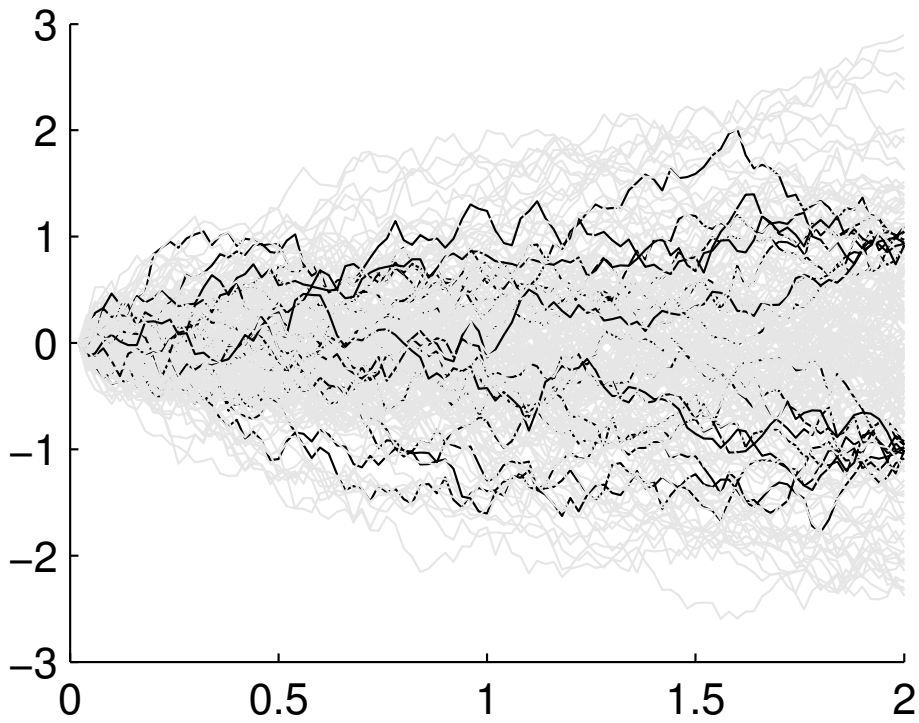
The optimal control cost is $C(p^*) = -\log \psi = J(x_0, 0)$ with

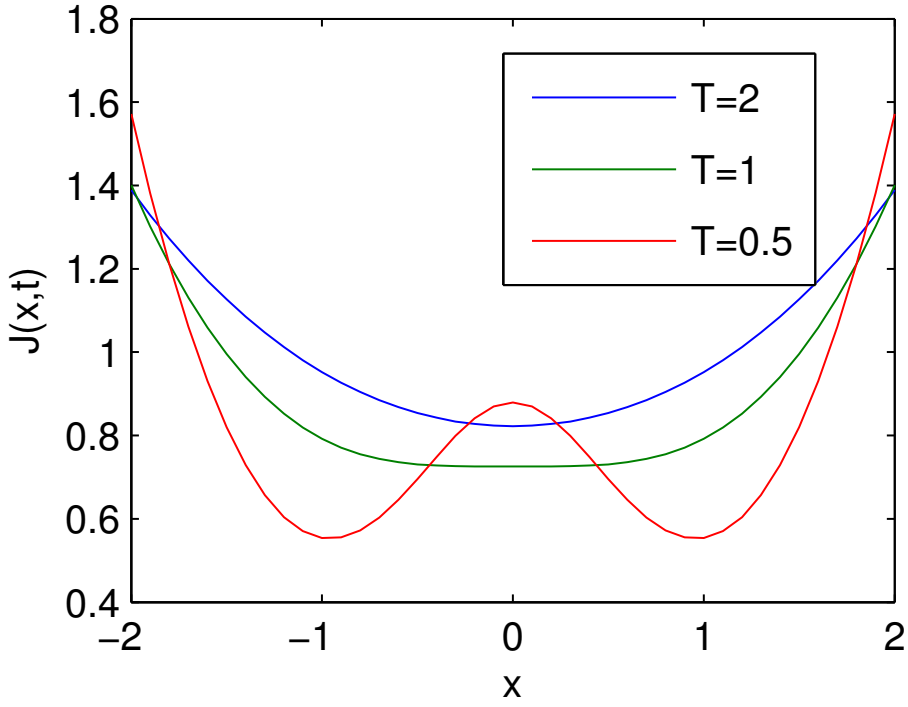
$$\psi = \int d\tau q(\tau) e^{-S(\tau)} = \mathbb{E}_q e^{-S}$$

$J(x,t)$ can be computed by forward sampling from q.

Delayed choice

Time-to-go $T = 2 - t$.



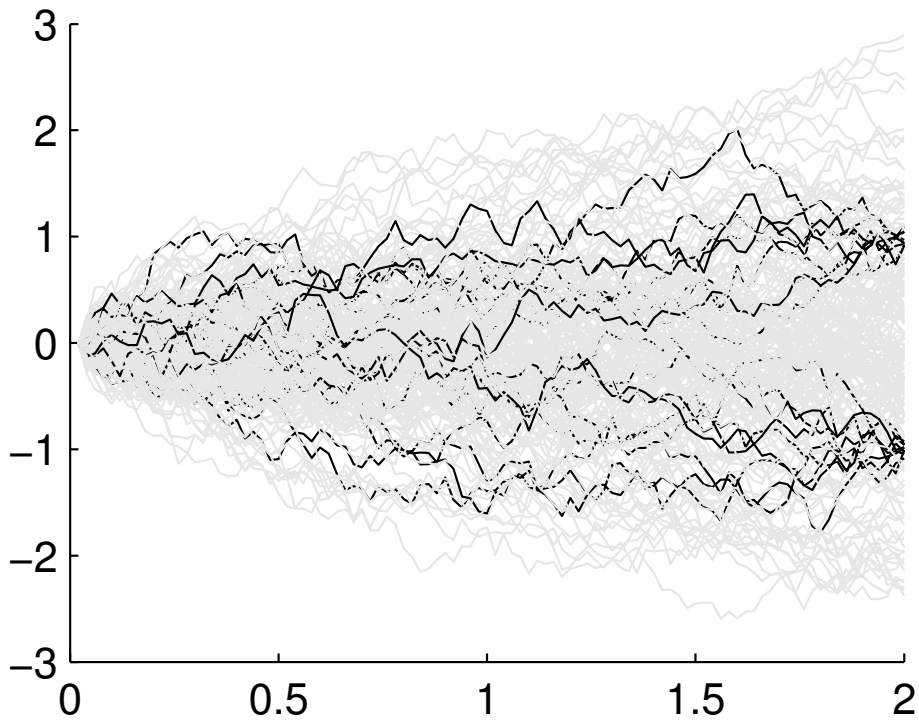


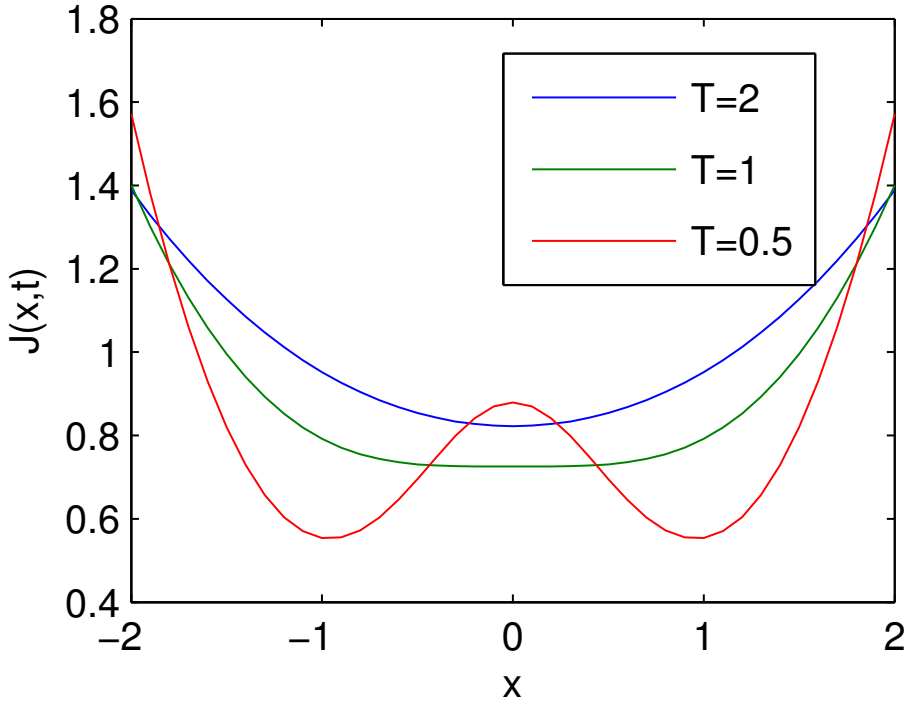
$$J(x,t) = -\nu \log \mathbb{E}_q \exp(-\phi(X_2)/\nu)$$

Decision is made at $T = \frac{1}{\nu}$

Delayed choice

Time-to-go $T = 2 - t$.





$$J(x,t) = -\nu \log \mathbb{E}_q \exp(-\phi(X_2)/\nu)$$

"When the future is uncertain, delay your decisions."


JUV 0-1 ASR 48:31
VISIT JUVENTUS.COM
25 J. HOLEBAS
C. MARCHISIO 8
1:01 / 9:41

Delayed choice (details)

$$dX_t = udt + dW_t \qquad \mathbb{E}dW_t^2 = \nu dt$$

$V = 0$, path cost is $\frac{1}{2}u^2$ and end cost $\phi(z = \pm 1) = 0, \phi(z) = \infty$ else encodes two targets at $z = \pm 1$ at $t = T$.

PI recipe:

1.

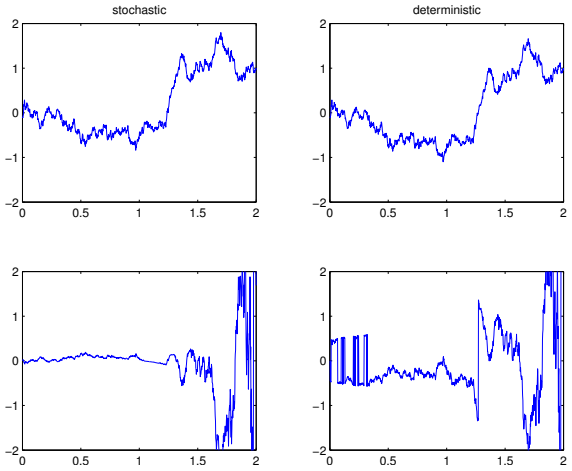
$$\begin{aligned}
\psi(x,t) &= \int dQ(\tau|x,t)\exp(-S(\tau)/\lambda) \\
S(\tau) &= \phi(x(T)) \\
\psi(x,t) &= \int dz q(z,T|x,t)\exp(-\phi(z)/\lambda) = q(1,T|x,t) + q(-1,T|x,t) \\
q(z,T|x,t) &= \mathcal{N}(z|x,\nu(T-t))
\end{aligned}$$

2. Compute

$$J(x,t) = -\lambda \log \psi(x,t) = \frac{1}{T-t}\left(\frac{1}{2}x^2 - \nu(T-t)\log 2\cosh\frac{x}{\nu(T-t)}\right)$$

3.

$$u(x,t) \quad = \quad -\nabla J(x,t) = \frac{1}{T-t}\left(\tanh\frac{x}{\nu(T-t)} - x\right)$$

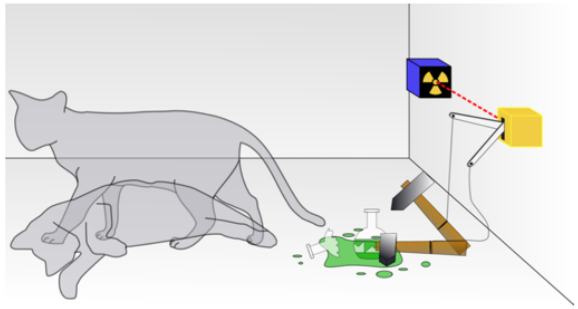


Coordination of UAVs

Chao Xu ACC 2017

The quantum picture

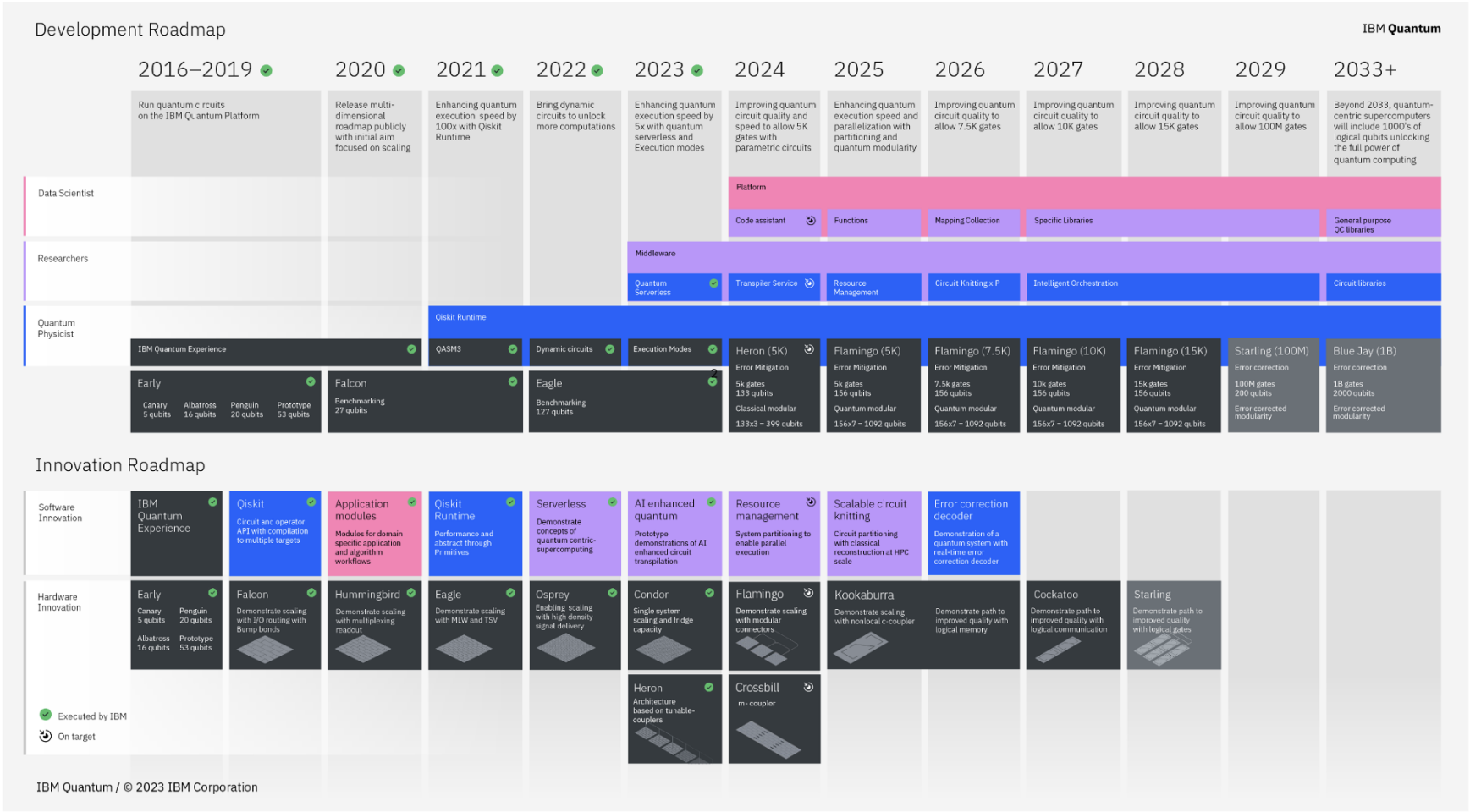
A quantum state $\psi(s)$ represents possible outcomes s simultaneously.



Define a procedure to map a probability distribution to a quantum state:

$$q(s) \qquad \leftrightarrow \qquad \psi(s)$$

Estimate expected values by performing repeated measurement on the same quantum state.



At IBM Quantum Summit 2023, the company extended the IBM Quantum Development Roadmap to 2033, and has established an IBM Quantum Innovation Roadmap through 2029. (Credit: IBM)

Quantum Computing in the NISQ era and beyond

John Preskill

Institute for Quantum Information and Matter and Walter Burke Institute for Theoretical Physics,
California Institute of Technology, Pasadena CA 91125, USA
30 July 2018

Noisy Intermediate-Scale Quantum (NISQ) technology will be available in the near future. Quantum computers with 50-100 qubits may be able to perform tasks which surpass the capabilities of today's classical digital computers, but noise in quantum gates will limit the size of quantum circuits that can be executed reliably. NISQ devices will be useful tools for exploring many-body quantum physics, and may have other useful applications, but the 100-qubit quantum computer will not change the world right away — we should regard it as a significant step toward the more powerful quantum technologies of the future. Quantum technologists should continue to strive for more accurate quantum gates and, eventually, fully fault-tolerant quantum computing.

Quantum Learning Theory

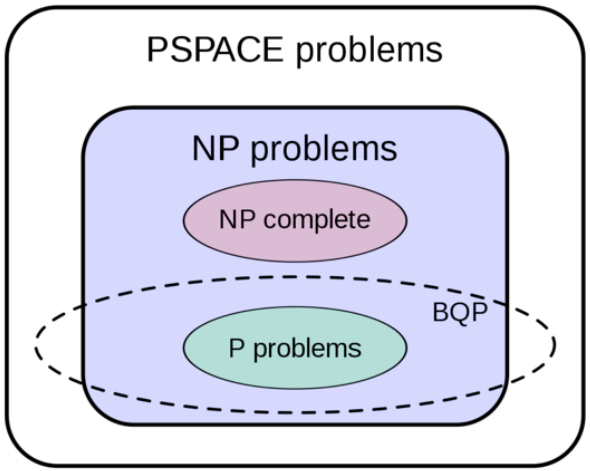


Image credits: wikipedia.org

P: solvable in poly time
NP: solution verifiable in poly time
Pspace: solvable with poly memory
BQP: Bounded Quantum Polynomial

It is conjectured that BQP solves hard problems outside of P, specifically, problems in NP. Examples are

- Integer factorization (Shor's algorithm) $\mathcal{O}\left(e^{N^{\frac{1}{3}}}\right) \rightarrow \mathcal{O}\left(N^2\right)$
- Solving sparse linear system (HHL) $\mathcal{O}\left(N\right) \rightarrow \mathcal{O}\left(\log N\right)$

Aram Harrow, Vedran Dunjko, Andrea Rochetto, Jens Eisert)

Quantum variational algorithms

Most famous example is the variational quantum eigensolver VQE to find the ground state of a Hamiltonian

$$\min_{\theta} \frac{\langle\psi|H\psi\rangle}{\langle\psi|\psi\rangle} \qquad |\psi\rangle = U(\theta)\,|0\rangle$$

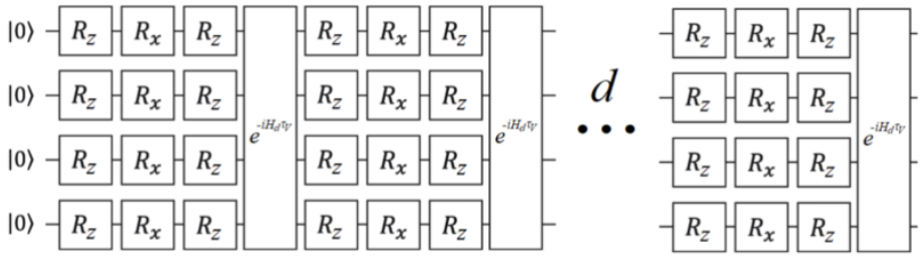
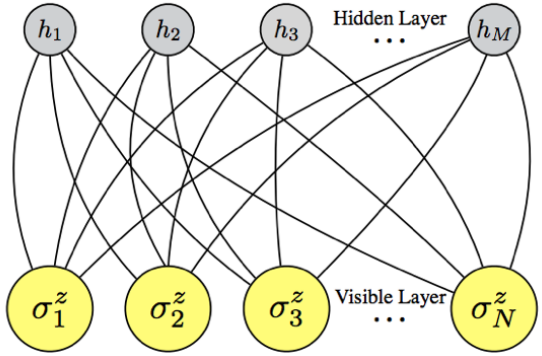


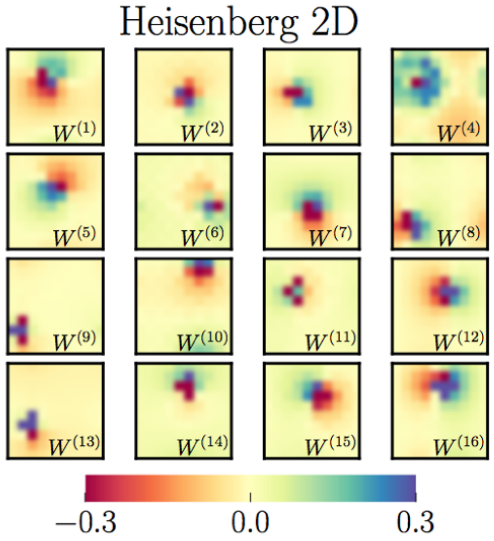
Figure 1a: Gate-based quantum circuit

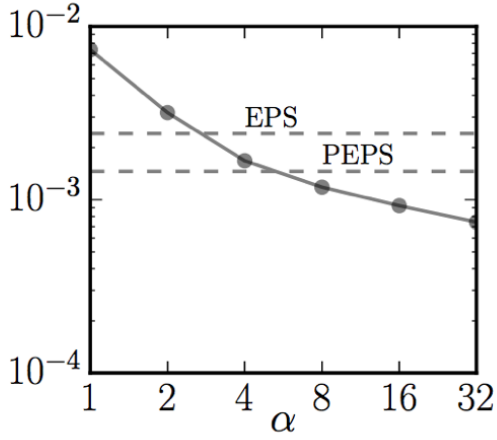
θ is found by minimizing $R(\theta) = \frac{\langle\psi|H\psi\rangle}{\langle\psi|\psi\rangle}$ with $|\psi\rangle = U(\theta)\,|0\rangle$ with respect to θ using gradient descend.

Machine learning $\rightarrow$ Quantum physics

Solving the Quantum Many-Body Problem with Artificial Neural Networks

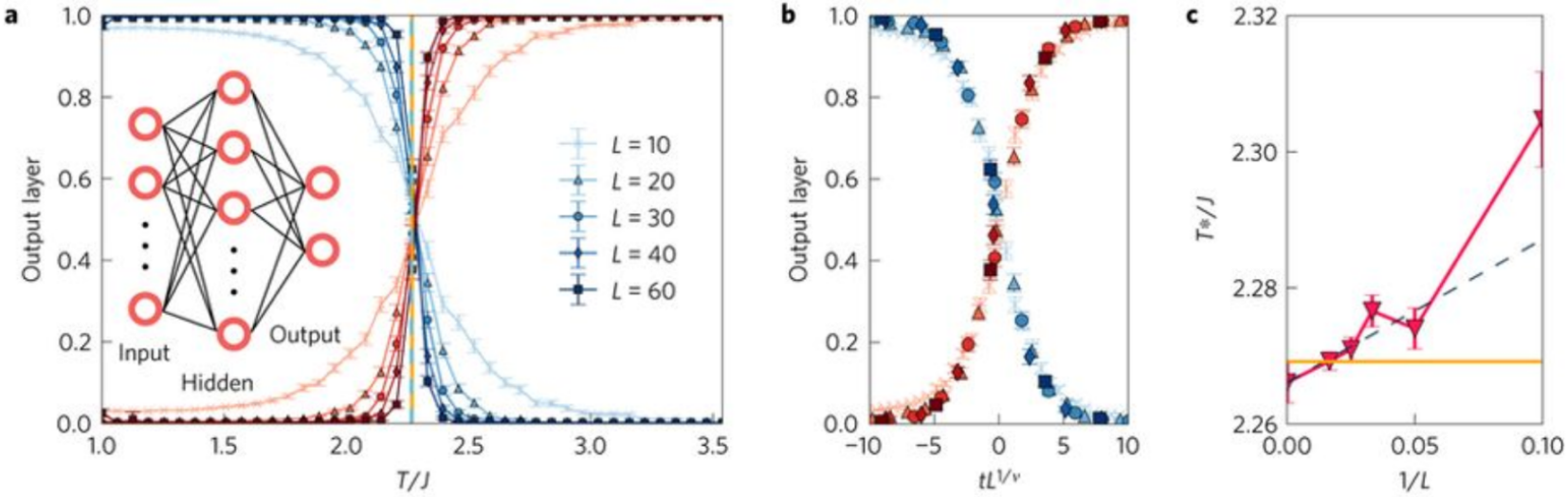






Carleo, Troyer, Science 2017

Machine learning phases of matter



Carrasquilla, Melko. Nature Physics 2017

Machine learning $\leftarrow$ Quantum physics

Classification with variational circuit [Havlíček et al., 2019]

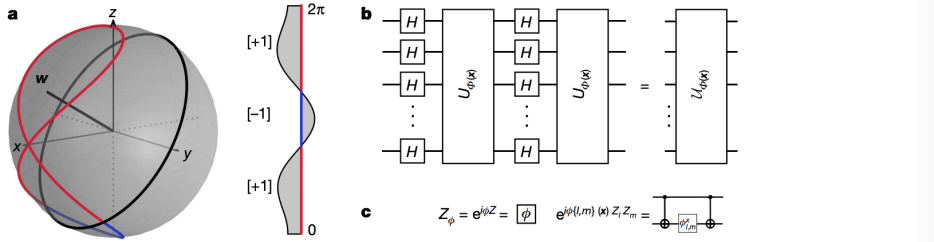


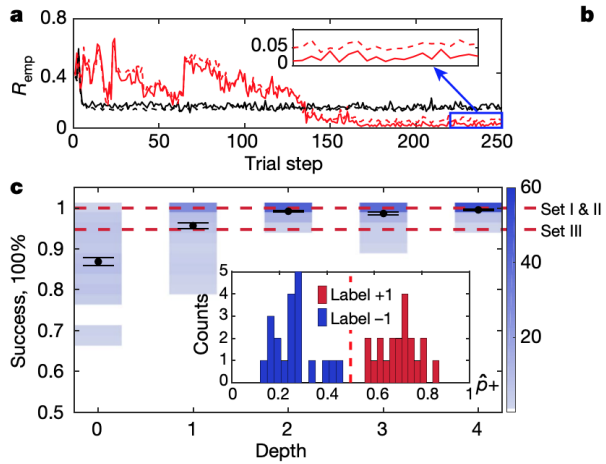
Fig. 1 | Quantum kernel functions. a, Feature map representation for a single qubit. A classical dataset in the interval $\Omega = (0, 2\pi]$ with binary labels (**a**, right) can be mapped onto the Bloch sphere (red and blue lines) by using the non-linear feature map described in **b**. For a single qubit $U_{\Phi(x)} = Z_x$ is a phase-gate of angle $x \in \Omega$. The mapped data can be separated by the hyperplane given by normal $\boldsymbol{w}$. States with a positive expectation value of $\boldsymbol{w}$ receive a $[+1]$ (red) label, while negative values are labelled $[-1]$ (blue). **b**, For the general circuit $U_{\Phi(x)}$ is formed by products of single- and two-qubit unitaries that are diagonal in the computational basis. In our experiments, both the training and testing data are artificially generated to be perfectly classifiable using the feature map. The circuit family depends non-linearly on the data through the coefficients $\phi_S(\boldsymbol{x})$ with $|S| \leq 2$. **c**, Experimental implementation of the parameterized diagonal single- and two-qubit operations using CNOTs and Z-gates.

Feature map:

$$|\Psi(x)\rangle = U_{\Psi(x)}|0\rangle \qquad U_{\Psi(x)} = e^{i\mathcal{H}(x)} \qquad \mathcal{H}(x) = \sum_{(ij)} \phi_{ij}(x)\hat{\sigma}_i^z\hat{\sigma}_j^z + \sum_i \phi_i(x)\hat{\sigma}_i^z$$

H are Hadamar gates $H(|0\rangle) = \frac{1}{2}(|0\rangle + |1\rangle) = |+\rangle$, $H(|1\rangle) = \frac{1}{2}(|0\rangle - |1\rangle) = |-\rangle$. Variational circuit of L layers $W(\theta) = U_{\text{ent}}U(\theta_L)U_{\text{ent}} \ldots U(\theta_1)U_{\text{ent}}$. Output is measurement of $\hat{F}$ on state $W(\theta)|\Psi(x)\rangle$ with probability $\left\langle \Psi(x)|W(\theta)^\dagger \hat{F} W(\theta)\Psi(x)\right\rangle$.

Classification with variational circuit [Havlíček et al., 2019]



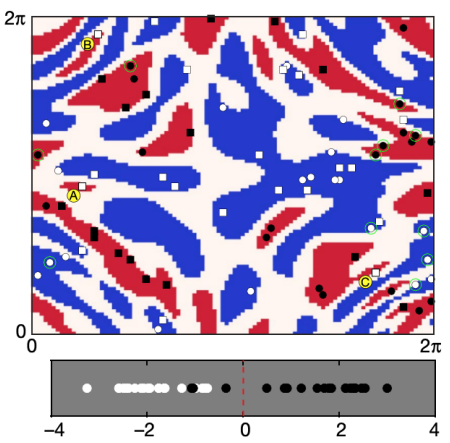


Fig. 3 | Convergence of the method and classification results. **a**, Convergence of the cost function $R_{emp}(\boldsymbol{\theta})$ after 250 iterations of Spall's SPSA algorithm. Red (or black) curves correspond to $l = 4$ (or $l = 0$). The cost function with $\hat{p}_k$ estimates obtained from zero-noise extrapolation (solid lines) is compared to the cost function with unmitigated estimates (dashed). We train three datasets per depth and perform 20 classifications per trained set. **c**, The classifications results are shown as blue histograms for all three randomly chosen unitaries (a total of 60 classifications per depth and 20 data points per classification per label), with mean values represented by black dots. The error bar is the standard error of the mean. The inset shows histograms as a function of the probability of measuring label $+1$ for one test set of 20 points per label obtained with an $l = 4$ classifier circuit, depicting classification of this set with 100% success. The dashed red lines show the results of our direct kernel estimation method for comparison, with Sets I and II yielding 100% success and Set III yielding 94.75% success. **b**, Example data used for both methods in this work. The data labels (red for $+1$ label and blue for -1 label) are generated with a gap of $\Delta = 0.3$ (white areas). The training set with 20 points per label is shown as white and black circles. For the quantum kernel estimation method we show the support vectors (green circles) and a classified test set (white and black squares). Three points are misclassified, labelled as A, B and C. For each of the test data points $\boldsymbol{s}_j$ we plot $\sum_i y_i \alpha_i^* K(\boldsymbol{x}_i, \boldsymbol{s}_j)$ at the bottom of **b**. Points A, B and C, all belonging to label $+1$, give $\sum_i y_i \alpha_i^* K(\boldsymbol{x}_i, \boldsymbol{s}_j) = -1.033, -0.367$ and -1.082, respectively.

Classification example on 2 qubits with $x = (x_1, x_2)$. Choose $\phi_i(x) = x_i, \phi_{12}(x) = (\pi - x_1)(\pi - x_2)$. Generate labels by thresholding $\left\langle \Psi(x) | V^\dagger \hat{F} V \Psi(x) \right\rangle$ (red and blue areas) with $\hat{F} = \hat{\sigma}_1^z \hat{\sigma}_2^z$. Train θ with varying number of layers $(L - 0, \ldots, 4)$ using 3 different V. Generate classification on test data thresholding at zero. 3 datapoints are misclassified (A,B,C).

Quantum Boltzmann machine [Kappen, 2020]

- Density matrices, von Neumann entropy, cross-entropy
- QBM learning rule
- Salamander retina data
- Entanglement
- Circuit implementation

Density matrix formalism

Consider discrete state space labeled by s.

The density matrix $\rho(s, s')$ is a Hermitian positive semi-definite matrix with real eigenvalues $\lambda(s)$.

$$\lambda(s) \geq 0 \qquad \mathrm{Tr}\,\rho = \sum_s \lambda(s) = 1$$

When $\rho(s, s') = p(s)\delta_{s,s'}$ the density matrix reduces to a classical probability distribution.

The expectation value of a matrix A (quantum statistics) is defined as

$$\langle A \rangle_\rho = \mathrm{Tr}\,(A\rho)$$

When A is diagonal,

$$\langle A \rangle_\rho = \langle A \rangle_p$$

with p the diagonal of ρ (classical statistics).

Quantum learning

The relative entropy between density matrices η, ρ is defined as [55]

$$S(\eta, \rho) = \mathrm{Tr}(\eta \log \eta) - \mathrm{Tr}(\eta \log \rho)$$

One can show (Klein's inequality): $S(\eta, \rho) \geq 0$ and $S(\eta, \rho) = 0$ iff $\eta = \rho$.

S generalizes the concept of cross entropy between distributions to density matrices.

Learning: Denote η the data density matrix and ρ is the model density matrix. Find ρ that minimises S.

When $\eta = \mathrm{diag}(q)$ and $\rho = \mathrm{diag}(p)$, then

$$S(\eta, \rho) = KL(q|p)$$

and minimizing S is equivalent to maximizing L_c.

[55] $\log \rho$ the matrix logarithm of ρ.

Quantum Boltzmann machine

We consider density matrix models of the form [56]

$$\rho = \frac{1}{Z} e^H \qquad Z = \mathrm{Tr}\left(e^H\right) \qquad H = \sum_r H_r w_r$$

H is the Hamiltonian of the quantum system
H_r are Hermitian matrices that describe the various quantum interactions.
$w = \{w_r, r = 1, \ldots\}$ are real parameters that parametrize ρ.

This gives the learning rule for the QBM:

$$\Delta w_r \propto -\frac{\partial S}{\partial w_r} = \langle H_r \rangle_\eta - \langle H_r \rangle_\rho$$

[56] e^H the matrix exponential of H

Quantum Boltzmann Machine

For n quantum spin variables σ_i^k we consider the Hamiltonian

$$H = \sum_{i=1}^{n} \sum_{k=x,y,z} w_i^k \sigma_i^k + \sum_{i=1, j>i}^{n} \sum_{k=x,y,z} w_{ij}^k \sigma_i^k \sigma_j^k$$

$\sigma_i^{x,y,z}$ are Pauli spin 1/2 operators. For this Hamiltonian, the learning rule becomes

$$\begin{aligned} \Delta w_i^k &\propto \left\langle \sigma_i^k \right\rangle_\eta - \left\langle \sigma_i^k \right\rangle_\rho \\ \Delta w_{ij}^k &\propto \left\langle \sigma_i^k \sigma_j^k \right\rangle_\eta - \left\langle \sigma_i^k \sigma_j^k \right\rangle_\rho \qquad k = x, y, z \end{aligned}$$

Pauli spin matrices

$\sigma_i^{x,y,z}$ are 2×2 Pauli spin matrices

$$\sigma_i^x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad \sigma_i^y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad \sigma_i^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

σ_i^z has eigenvectors that we call states and denote by their eigenvalue s_i:

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} = |s_i = 1\rangle \qquad \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |s_i = -1\rangle$$

Thus for $s_i = \pm 1$:

$$\sigma_i^z |s_i\rangle = s_i |s_i\rangle \qquad \sigma_i^x |s_i\rangle = |-s_i\rangle \qquad \sigma_i^y |s_i\rangle = i s_i |-s_i\rangle$$

Hamiltonian matrix representation

The state of an n spins system is the tensor product of the states $|s_i = \pm 1\rangle$ for each spin:

$$|s\rangle = |s_1, \ldots, s_n\rangle$$

Thus a total of 2^n states.

On this basis, H has matrix elements [57]

$$\begin{aligned}\langle s'|H|s\rangle &= \sum_{i=1}^{n}(w_i^x + iw_i^y s_i)\delta_{s',F_i s} + \sum_{i<j}^{n}\left(w_{ij}^x - w_{ij}^y s_i s_j\right)\delta_{s',F_i F_j s} \\ &+ \left(\sum_{i=1}^{n} w_i^z s_i + \sum_{i<j}^{n} w_{ij}^z s_i s_j\right)\delta_{s',s}\end{aligned}$$

[57] When $s = (s_1, \ldots, s_i, \ldots, s_n)$, $F_i s = (s_1, \ldots, -s_i, \ldots, s_n)$

Examples of quantum tomography

Given quantum statistics of an unknown quantum system

$$\left\langle \sigma_i^k \right\rangle_\eta, \left\langle \sigma_i^k \sigma_j^k \right\rangle_\eta \qquad k = x, y, z$$

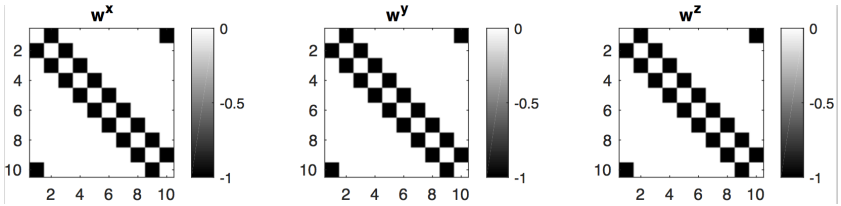
Estimate the quantum Hamiltonian that is the source of these quantum statistics.

Numerical examples:

- Given H_{true}, compute the ground state wave function ψ_{true}.
- From ψ_{true}, compute the quantum statistics $\left\langle \sigma_i^k \right\rangle_\eta, \left\langle \sigma_i^k \sigma_j^k \right\rangle_\eta$.
- Train QBM and compare learned H, ψ with $H_{\text{true}}, \psi_{\text{true}}$

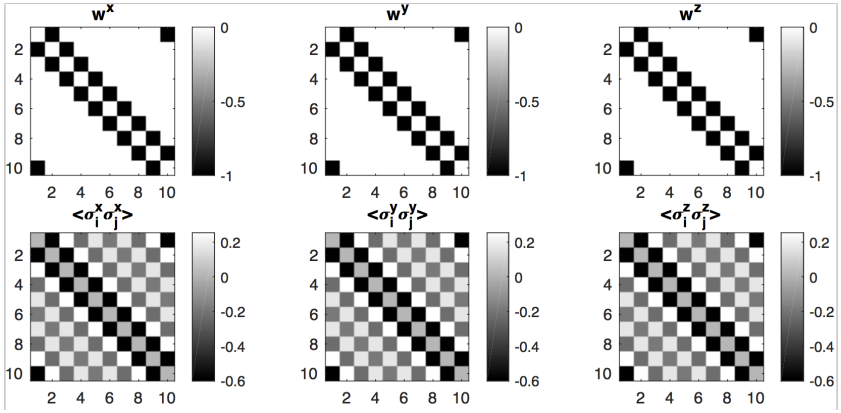
Anti-ferromagnetic Heisenberg model

The Hamiltonian has couplings $w_{ij}^{x,y,z} = -1$ for nearest neighbours and $w_{ij}^{x,y,z} = 0$ otherwise.



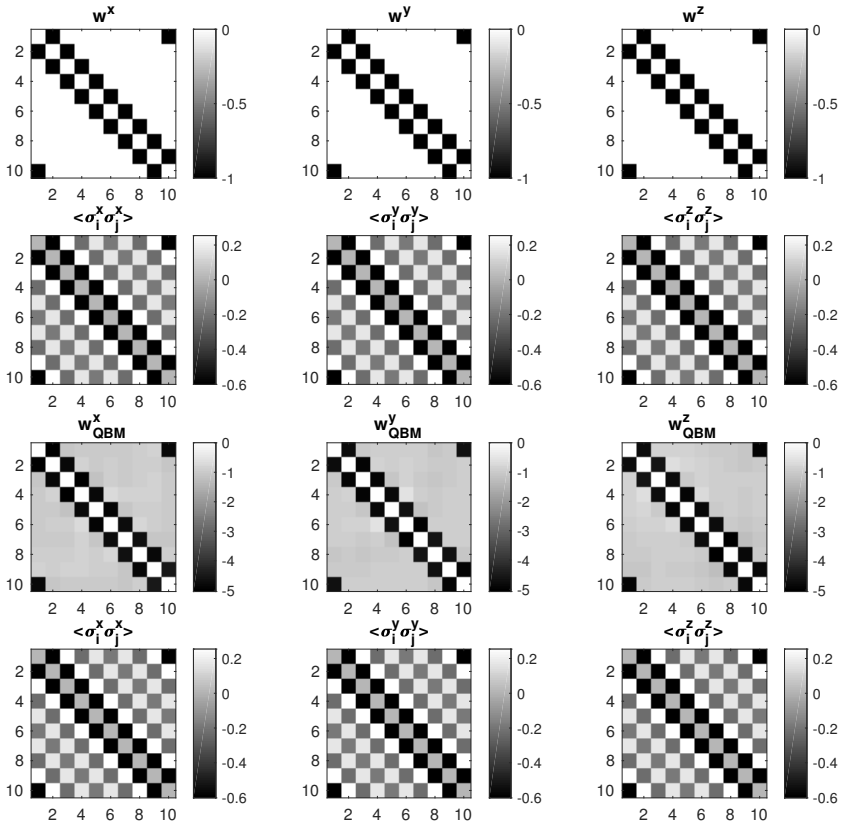
Anti-ferromagnetic Heisenberg model

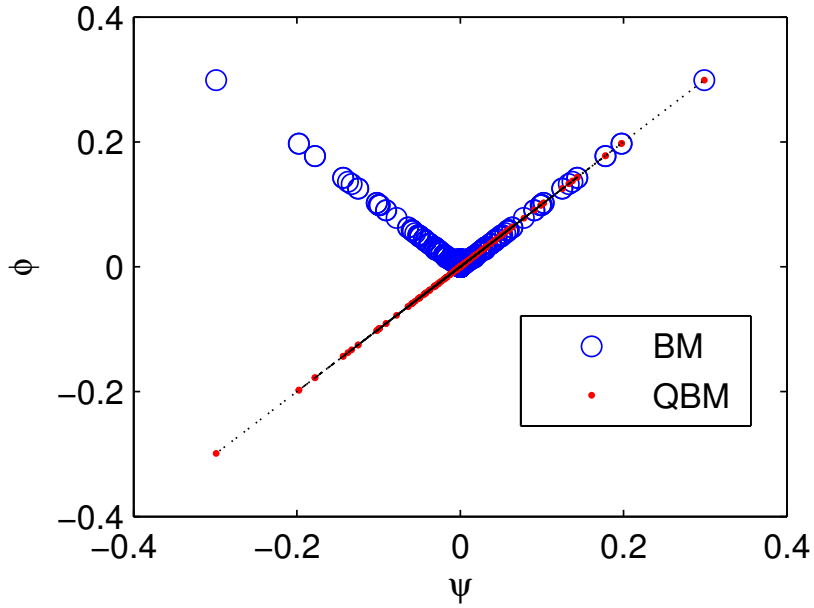
The Hamiltonian has couplings $w_{ij}^{x,y,z} = -1$ for nearest neighbours and $w_{ij}^{x,y,z} = 0$ otherwise.



Anti-ferromagnetic Heisenberg model

The Hamiltonian has couplings $w_{ij}^{x,y,z} = -1$ for nearest neighbours and $w_{ij}^{x,y,z} = 0$ otherwise.

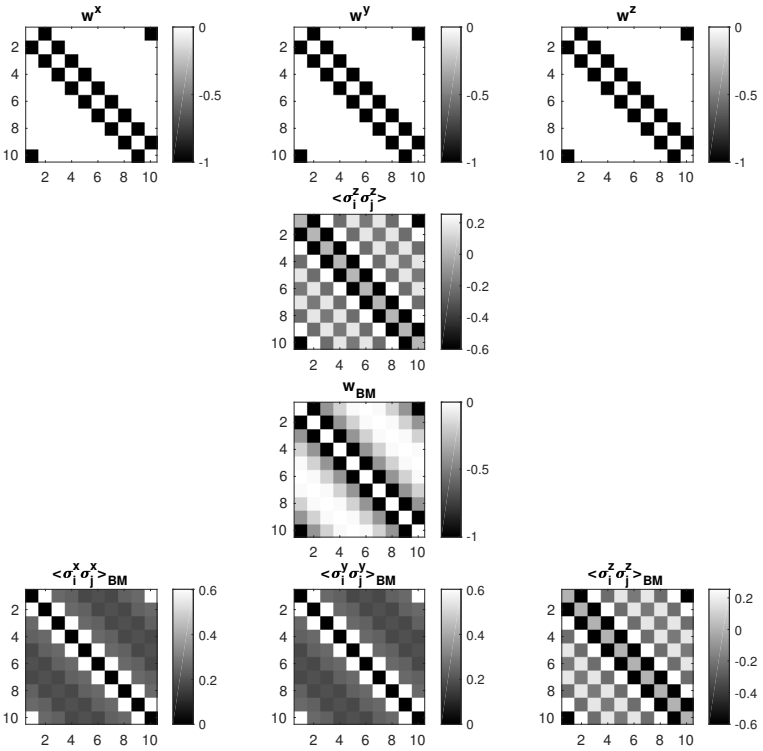


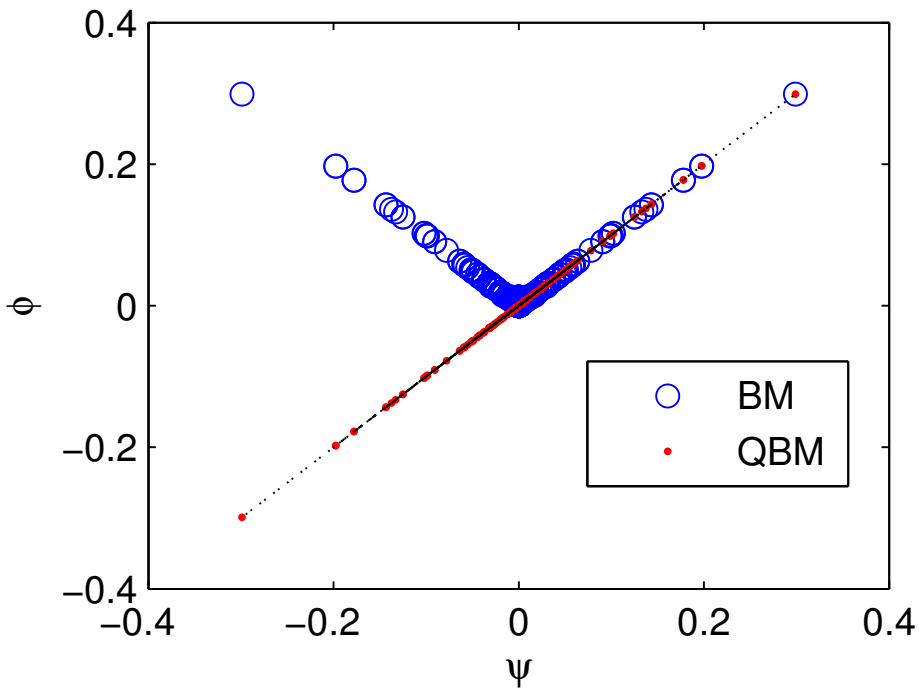


During learning, the couplings w_i^k, w_{ij}^k diverge so that $\rho \propto e^H \to \phi\phi'$ approaches a rank one solution with correct quantum statistics.

Anti-ferromagnetic Heisenberg model

The Hamiltonian has couplings $w_{ij}^{x,y,z} = -1$ for nearest neighbours and $w_{ij}^{x,y,z} = 0$ otherwise.





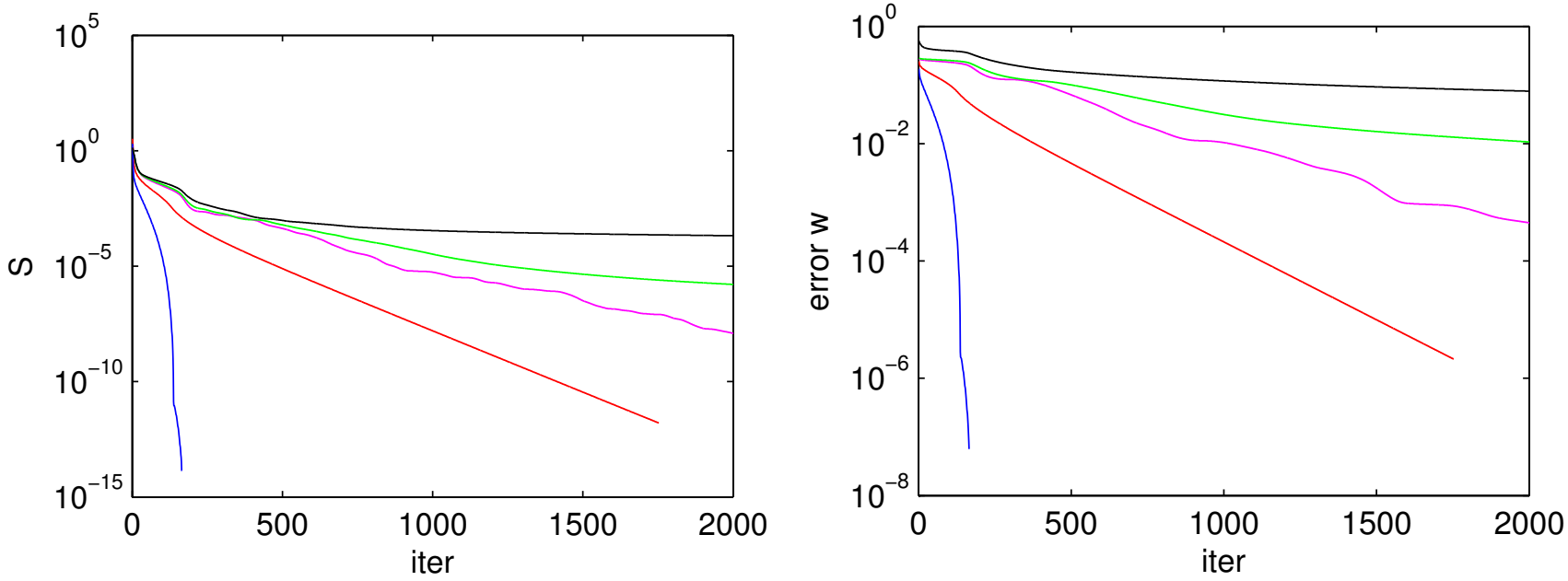
The classical BM does not yield a good solution: wrong couplings and wrong sign of $\phi = \sqrt{p_{\rm bm}}$. The BM correctly captures the classical statistics but not the quantum statistics.

Fully connected quantum spin glass

Data statistics from

$$\eta = \frac{1}{Z} e^{\beta H}$$

with H a quantum spin glass Hamiltonian.



$\beta = 0.5$ (blue), $\beta = 1$ (red), $\beta = 3$ (magenta), $\beta = 5$ (green) and $\beta = \infty$ (black).

Quantum statistics in classical data

We can also apply the QBM to learn classical data. Given data $s^\mu, \mu = 1, \ldots, P$:

$$q(s) = \frac{1}{P} \sum_{\mu=1}^{P} \delta_{s,s^\mu}$$

We construct a rank one density matrix as follows

$$\eta = |\psi\rangle \langle\psi| \qquad \langle s|\psi\rangle = \sqrt{q(s)}$$

The quantum expectations

$$\langle A \rangle_\eta = \sum_{s,s'} A(s, s') \sqrt{q(s) q(s')}$$

generalize the classical expectations and become equal for A diagonal.

Classical statistics are linear in q. The quantum statistics are quadratic in $\sqrt{q}$.

Quantum statistics in classical data

For the QBM:

$$\langle \sigma_i^x \rangle_\eta = \sum_s \sqrt{q(F_i s) q(s)} \qquad \left\langle \sigma_i^x \sigma_j^x \right\rangle_\eta = \sum_s \sqrt{q(F_i F_j s) q(s)}$$

$$\left\langle \sigma_i^y \right\rangle_\eta = 0 \qquad \left\langle \sigma_i^y \sigma_j^y \right\rangle_\eta = -\sum_s s_i s_j \sqrt{q(F_i F_j s) q(s)}$$

$$\left\langle \sigma_i^z \right\rangle_\eta = \langle s_i \rangle_q \qquad \left\langle \sigma_i^z \sigma_j^z \right\rangle_\eta = \left\langle s_i s_j \right\rangle_q$$

The quantum statistics measure similarity between patterns in the data.

Quantum learning of classical data

Minimize $S(\eta, \rho)$ with respect to ρ.

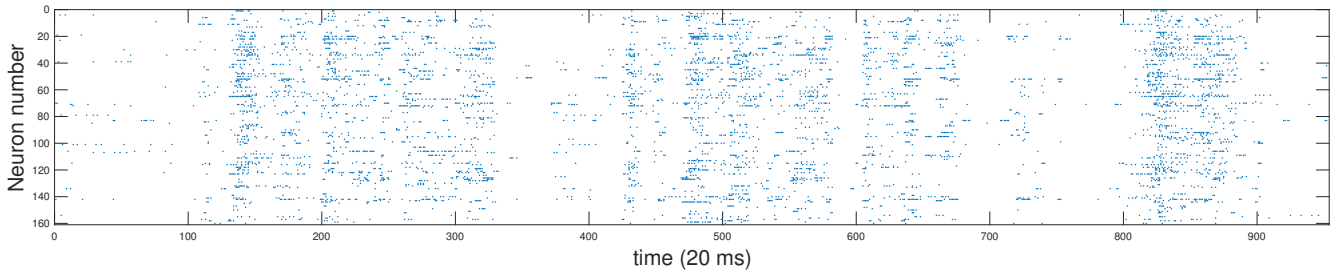
The solution is at least as good as the classical solution:

$$\min_{\rho} S(\eta, \rho) \leq \min_{p} KL(q|p)$$

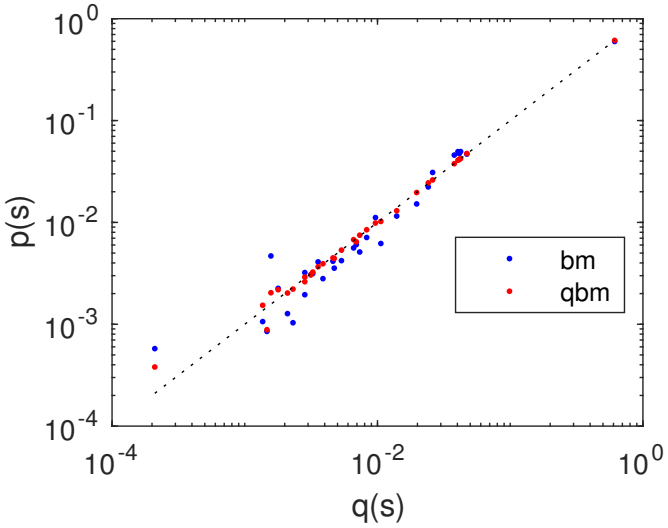
and equal when ρ is diagonal.

The optimal solution ρ represents the quantum statistics in the classical data and has no equivalent in terms of a probability distribution.

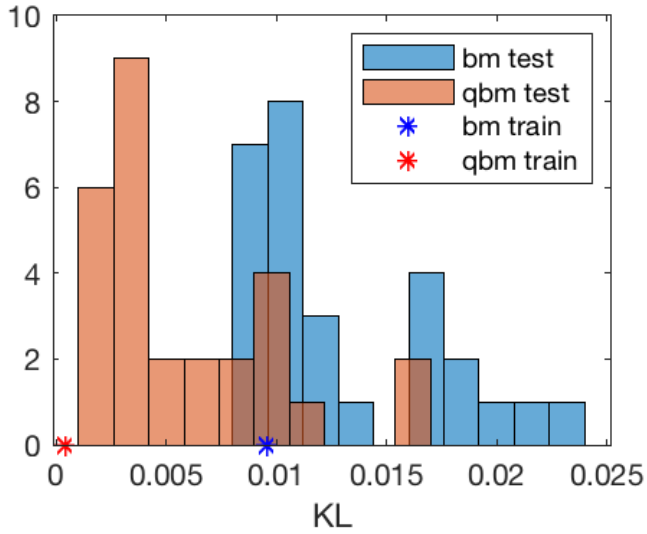
Modeling neural network activity



(a) One repeat of neural activity of 160 salamander retinal ganglion cells.



(b) p_{bm} and p_{qbm} versus q.



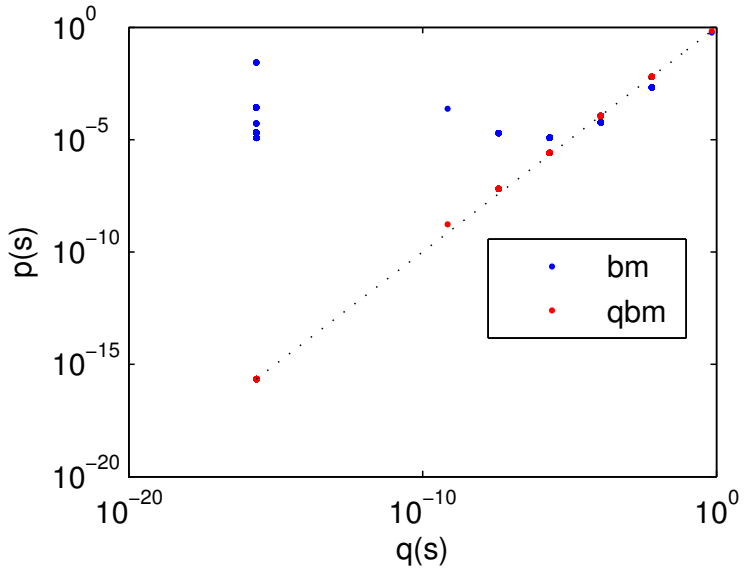
(c) Histogram of KL divergences of BM and QBM on 28 independent test sets.

QBM captures more statistics that generalize to new data.

Parity problem

Data are generated such that even parity states s have $q(s) > 0$ and odd states have $q(s) = 0$.

QBM learns this problem well: $S(\eta, \rho_{\text{qbm}}) = 4.69 \times 10^{-3}$
BM cannot learn this problem: $S(\eta, \rho_{\text{bm}}) = 2.37$



$\rho_{\text{qbm}} \approx \psi\psi^{\dagger}$. Define $p_{\text{qbm}}(s) = |\psi(s)|^2$.

$KL(q|p_{\text{qbm}}) = 1.31 \times 10^{-5}$
$KL(q|p_{\text{bm}}) = 0.451$

Classification

Given data $(x, y)^\mu, \mu = 1, \ldots, P$ with x input vectors and $y = \pm 1$ class label.

It is not easy to learn conditional density matrices. Instead, define $s = (x, y)$ and learn joint problem as before.

Data distribution

$$q(s) = q(x)q(y|x) \qquad q(y|x) = \delta_{y,f(x)} \qquad q(x) \propto e^{\theta \sum_i x_i}$$

Classifiers

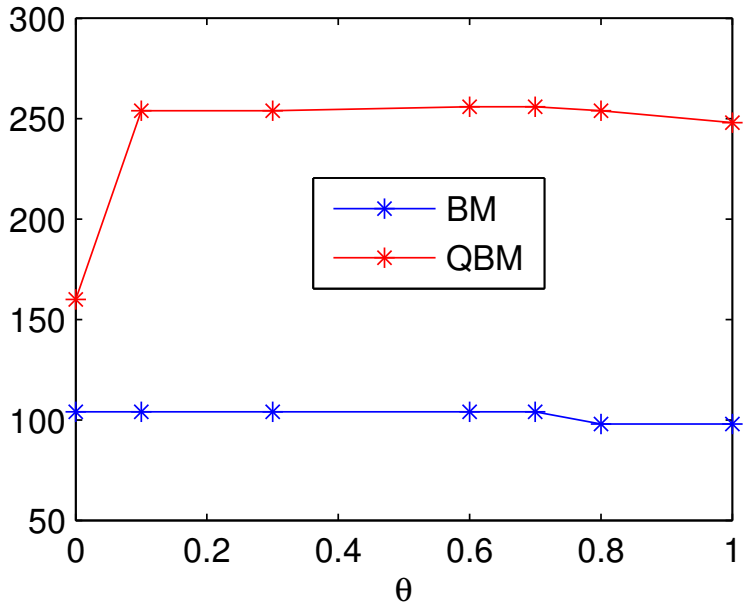
$$p_{\text{qbm}}(y|x) = \frac{|\psi(x,y)|^2}{\sum_y |\psi(x,y)|^2} \qquad p_{\text{bm}}(y|x) = \frac{p_{\text{bm}}(x,y)}{\sum_y p_{\text{bm}}(x,y)}$$

with ψ the ground state wave function of the learned Hamiltonian.

NB: p_{bm} implements linear logistic regression, p_{qbm} not.

Classification

Consider all $2^{2^n} = 256$ binary classification problems for $n = 3$ inputs.



Number of correctly classified problems for the BM and QBM as a function of θ. For the BM, 104 problems are correctly classified for $0 \leq \theta \leq 0.7$. For the QBM, 254 or more problems are correctly classified for $0.1 \leq \theta \leq 0.8$ and all 256 problems are correctly classified for $\theta \approx 0.6$–0.7.

Entanglement in classical data

The entanglement between variables in subset A and its complement B is defined as the quantum mutual information

$$I = S(\rho, \rho_A \otimes \rho_B) = h(\rho_A) + h(\rho_B) - h(\rho) \qquad \rho_A = \mathrm{Tr}_B(\rho) \qquad \rho_B = \mathrm{Tr}_A(\rho)$$

with $h(\rho) = -\mathrm{Tr}(\rho \log \rho)$.

For the data density matrix η one has $h(\eta) = 0$ and $h(\eta_A) = h(\eta_B)$ and

$$I = 2h(\eta_A)$$

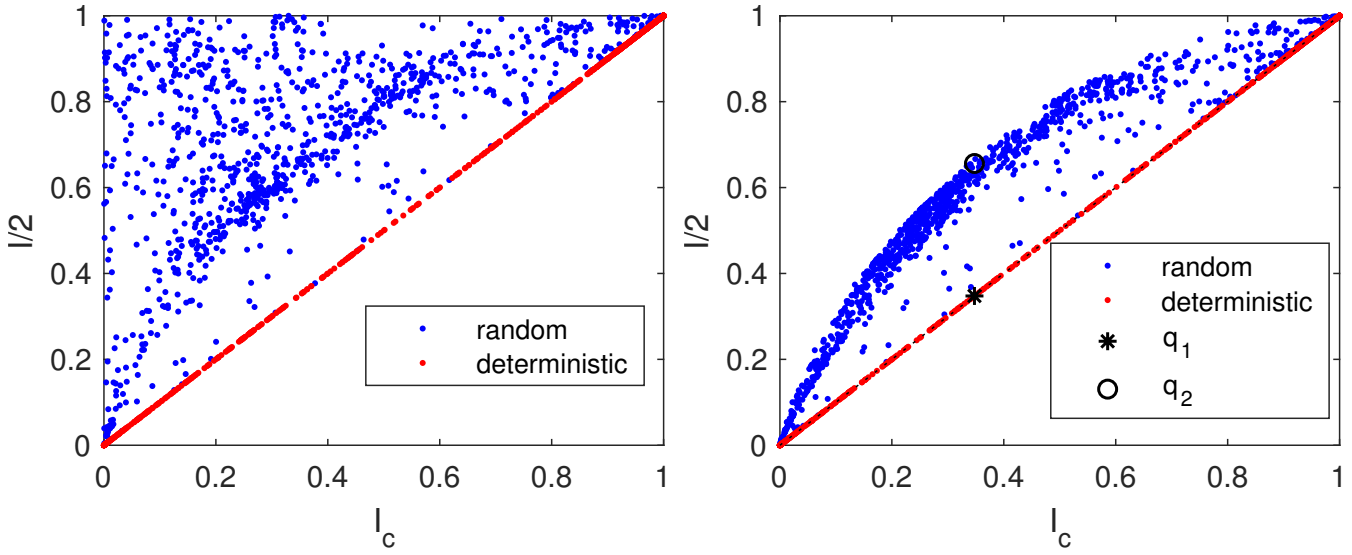
The classical mutual information is

$$I_c = KL(q, q_A \otimes q_B) \leq \min(h(q_A), h(q_B))$$

Entanglement in classical data

The transformation $\langle s|\psi\rangle = e^{i\alpha(s)}\sqrt{q(s)}$ changes the mutual information

$$\begin{array}{ccc} q(s) & \rightarrow & \eta = |\psi\rangle\langle\psi| \\ \downarrow & & \downarrow \\ I_c(q) & \leq & C = \frac{1}{2}I(\eta) \end{array}$$



Relation between $C = \frac{1}{2}I(\eta)$ and $I_c(q)$ mutual information for two sub sets (s_1, s_2) and (s_3). Blue: Random $q(s_1, s_2, s_3)$. Red: Random $q(s_1, s_2 = s_3)$. Left: random phases $\alpha(s)$; (b) phases $\alpha(s) = 0$.

Entanglement in classical data

$I_c(q) \leq C(\eta)$ in general, but we can increase the classical mutual information by measuring in another orthogonal basis.

$$|t\rangle = U\,|s\rangle \qquad \rightarrow \qquad \tilde{q}(t) = |\,\langle t|\psi\rangle\,|^2$$

By maximizing over $U = U_A \otimes U_B$ we find $I_c(\tilde{q}) = C$.

The remaining $\frac{1}{2}I(\eta)$ cannot be captured by orthogonal measurement.

The statistics of a set of non-orthogonal measurements are not described by a probability distribution using local variables.

$$q_1(s_2, s_3) = \begin{pmatrix} 0.0652 & 0 \\ 0 & 0.9348 \end{pmatrix} \qquad q_2(s_2, s_3) = \begin{pmatrix} 0.25 & 0.375 \\ 0.375 & 0 \end{pmatrix}$$

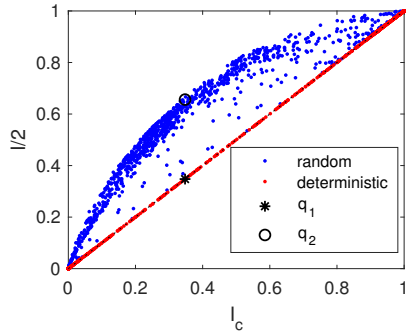
independent of s_1. By construction

$$\frac{1}{2} I(\eta_1) = I_c(q_1) = I_c(q_2) < \frac{1}{2} I(\eta_2)$$

Measurement in the orthogonal basis defined by the Schmidt decomposition yields

$$\tilde{q}_2(t_2, t_3) = \begin{pmatrix} 0.8307 & 0 \\ 0 & 0.1693 \end{pmatrix} \qquad U_A(t, s) = U_B(t, s) = \begin{pmatrix} 0.83 & 0.5577 \\ 0.5577 & -0.83 \end{pmatrix}$$

and $I_c(\tilde{q}_2) = \frac{1}{2} I(\eta_2)$.



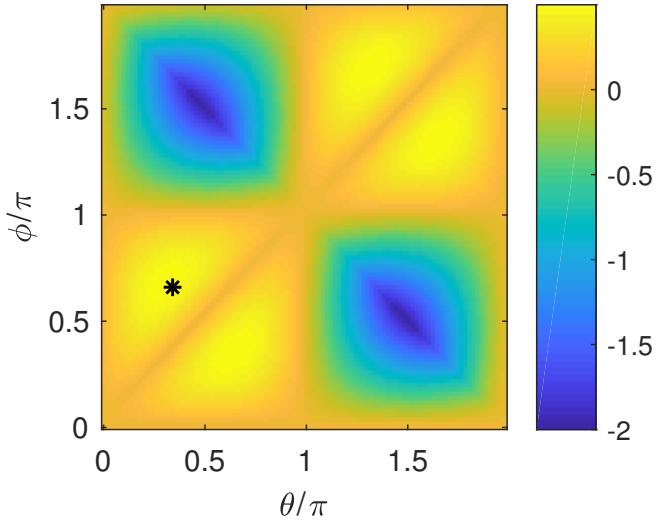
Entanglement in classical data

Consider two fully correlated binary variables $s = (s_1, s_2)$ with joint probability distribution

$$q(s_1, s_2) = \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

4 non-orthogonal measurements $A_i \otimes B_j$

$$\begin{aligned} A_1 &= \hat{\sigma}_1^z, \\ A_2 &= \hat{\sigma}_1^x \sin\theta + \hat{\sigma}_1^z \cos\theta \\ B_1 &= \hat{\sigma}_2^z \\ B_2 &= \hat{\sigma}_2^x \sin\phi + \hat{\sigma}_2^z \cos\phi \end{aligned}$$



with outputs $a_1 = b_1, a_2, b_2$. If they follow any $p(a_1, a_2, b_2)$ then

$$|\langle a_2 b_1\rangle - \langle a_1 b_2\rangle| + \langle a_2 b_2\rangle - 1 \leq 0$$

Intractability

Forward problem: $H \rightarrow \langle \ldots \rangle_\rho$ is intractable.

Inverse problem (learning): $\langle \ldots \rangle_\eta \rightarrow H$ is solved as a sequence of forward problems:

$$w \rightarrow H(w) \rightarrow \langle \ldots \rangle_\rho \qquad \Delta w \propto \langle \ldots \rangle_\eta - \langle \ldots \rangle_\rho$$

Intractability

Forward problem: $H \to \langle \ldots \rangle_\rho$ is intractable.

Inverse problem (learning): $\langle \ldots \rangle_\eta \to H$ is solved as a sequence of forward problems:

$$w \to H(w) \to \langle \ldots \rangle_\rho \qquad \Delta w \propto \langle \ldots \rangle_\eta - \langle \ldots \rangle_\rho$$

Possible accellerations

- Software: quantum belief propagation, diffusion MC
- Hardware: circuit implementation

Training Quantum Boltzmann Machines with the β-Variational Quantum Eigensolver [Huijgen et al., 2023]

Circuit implementation of gradient

$$\Delta w_r \propto \frac{\partial}{\partial w_r} L = \langle H_r \rangle_\eta - \langle H_r \rangle_\rho$$

with $\langle H_r \rangle_\rho = \mathrm{Tr}(\rho H_r)$ and $\rho = \frac{e^H}{\mathrm{Tr}(e^H)}$. Approximate ρ using β-VQE.

$$\rho = \sum_x p_\phi(x) U_\theta |x\rangle\langle x| U_\theta^\dagger.$$

$$\mathcal{L} = \mathrm{Tr}(\rho \ln \rho) + \beta \, \mathrm{Tr}(\rho H),$$

$$\mathcal{L} = \mathbb{E}_{x \sim p_\phi(x)} \left[\ln p_\phi(x) + \beta \left\langle x \middle| U_\theta^\dagger H U_\theta \middle| x \right\rangle \right]$$

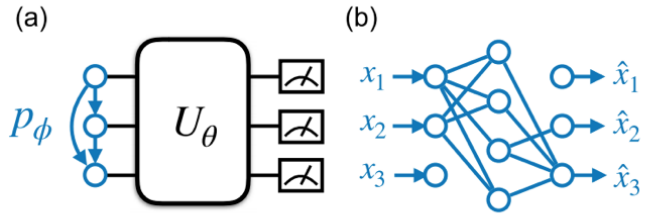


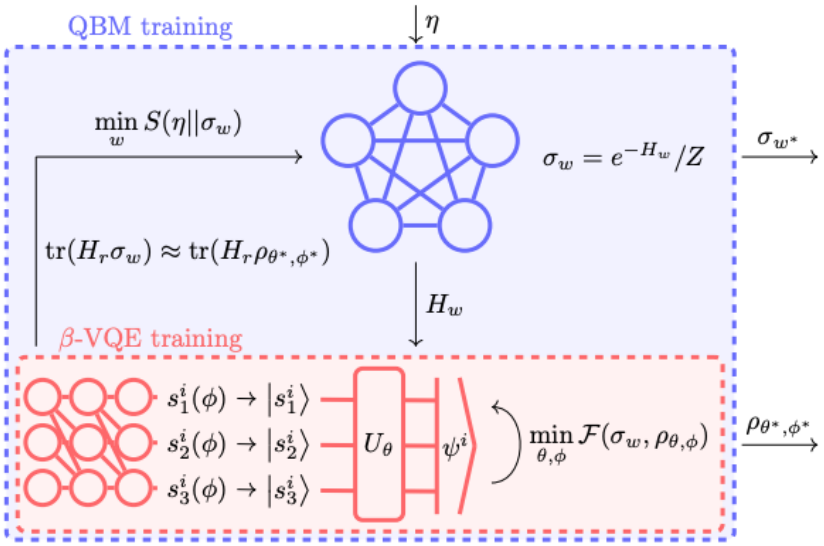
Figure 1. (a) The autoregressive network shown in blue is a classical probabilistic model that parametrizes a joint distribution in the form of equation (2). The model generates bit string as easy to prepare input product states to the quantum circuit. The neural network and the circuit produce a parametrized density matrix equation (1). (b) An implementation of the autoregressive model p_ϕ using the masked autoencoder [14]. The neural network maps bit strings to real-valued outputs which parametrizes the conditional probabilities in equation (2).

Liu et al. 2021

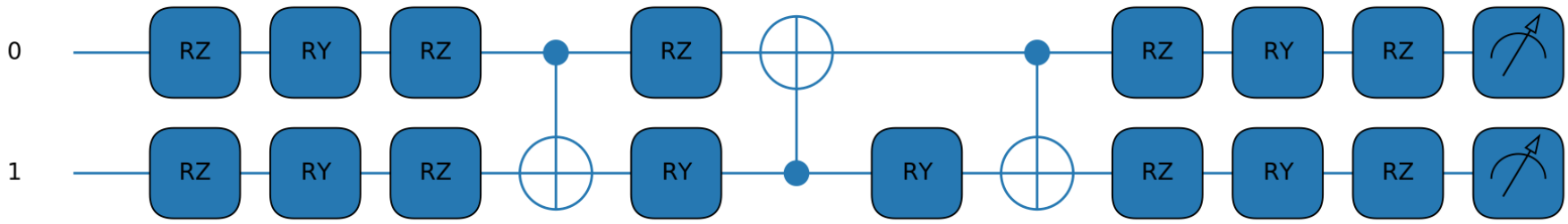
Nested loop training

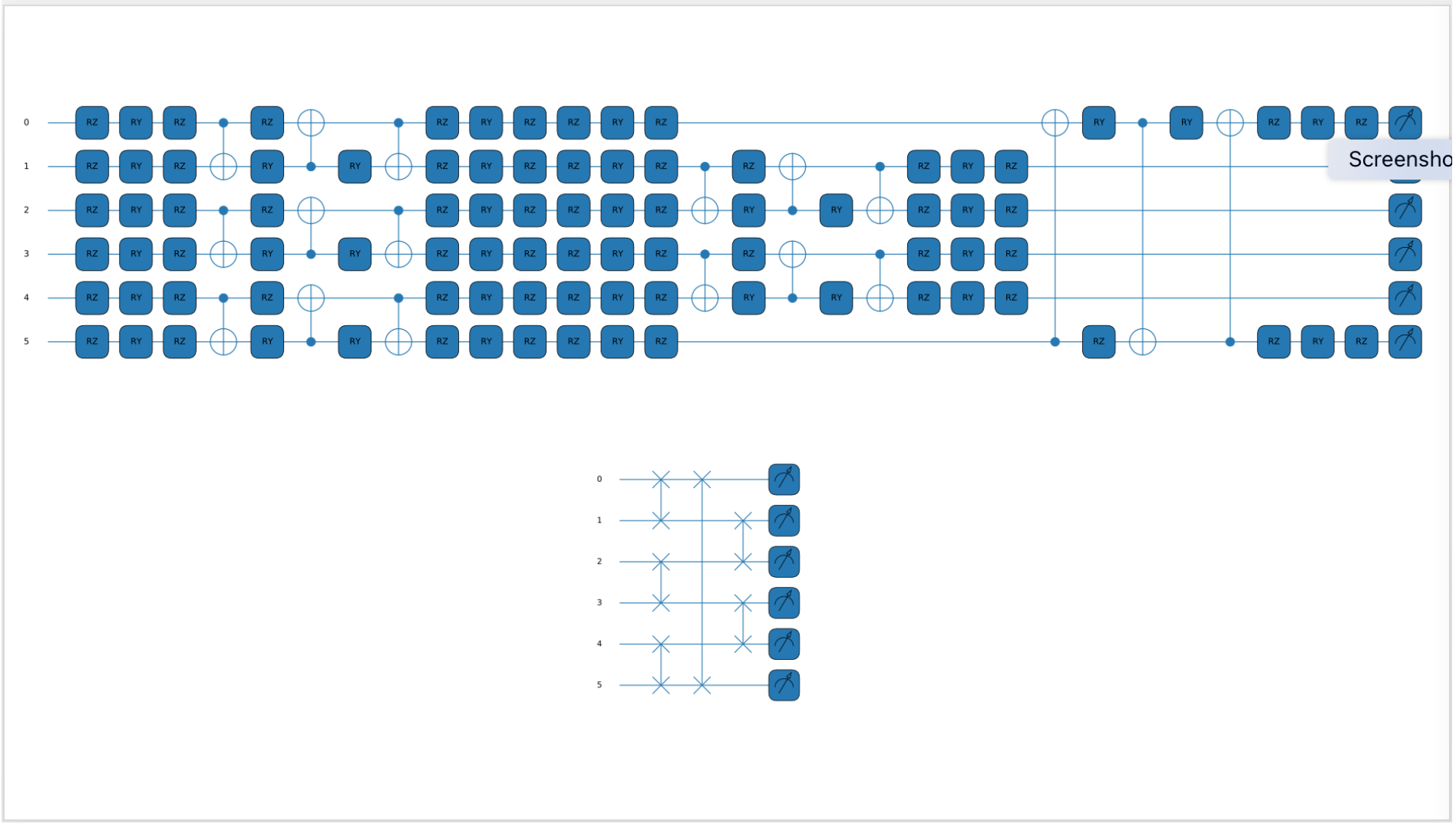
- Init w_k compute $H = \sum_k w_k H_k$.

- Loop
  - Approximate $\sigma = \frac{1}{Z} e^{\beta H}$ by ρ from β-VQE
  - Compute $\langle H_r \rangle = \mathrm{Tr}(\sigma H_r) \approx \mathrm{Tr}(\rho H_r)$
  - Update $w_k := w_k + \eta \left(\langle H_r \rangle_c - \langle H_r \rangle \right)$

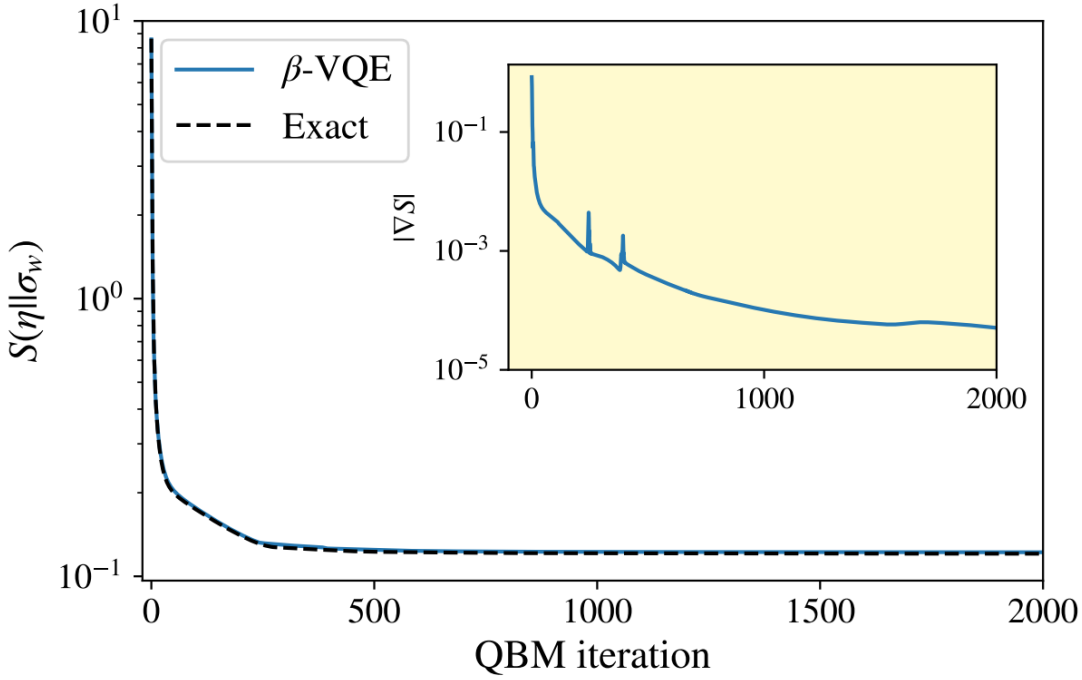


Click to add title




0
1
2
3
4
5
RZ
RY
Screensho

β-VQE approximates exact learning well



$n = 8$ salamander retina data. β-VQE has autoregressive network with 500 hidden units and quantum circuit of 3 checkerboard layers

Dependence on rank of β-VQE and problem size

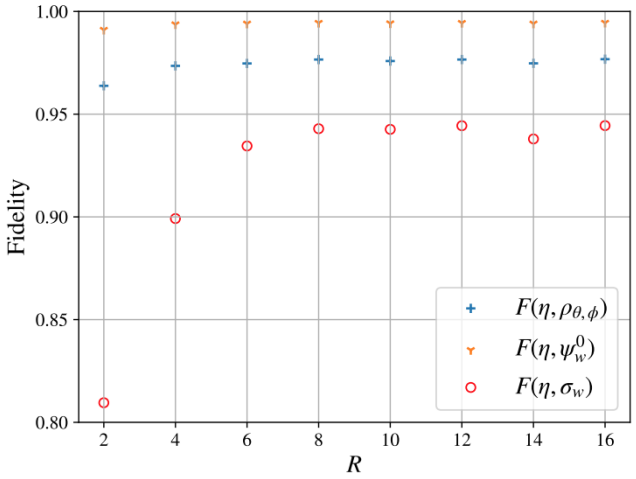


FIG. 3. Fidelity achieved with different β-VQE rank R. The β-VQE achieves fidelities > 0.96 even with small R. While the full QBM state σ_w is still mixed resulting in a lower fidelity, the ground state ψ_w^0 achieves a fidelity of > 0.99 in all cases. σ_w can be turned into a pure state by multiplying the weights by a large constant factor.

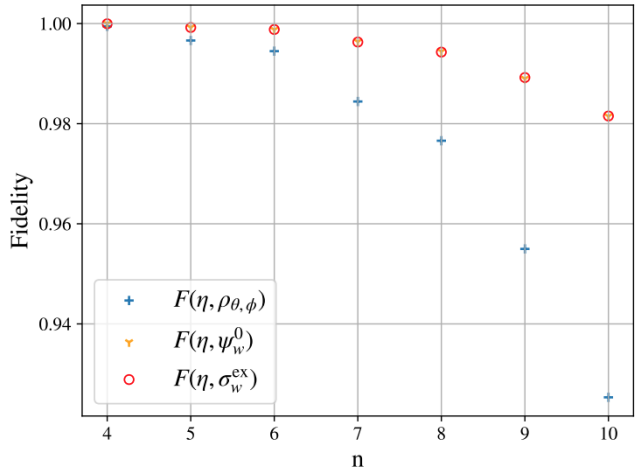


FIG. 4. Fidelity for different system sizes. For all sizes, a fidelity of $F > 0.98$ is achieved between the target state and the ground state of the QBM. The fidelity drops for larger system sizes, which is due to model mismatch between the QBM and the target.

Salamander retina data. Left: Performance versus rank R. Low rank $R = 2$ yields excellent ground state approximations. Right: Performance versus problem size n. General drop-off is due to model miss match.

Learning high rank quantum data and circuit simulation

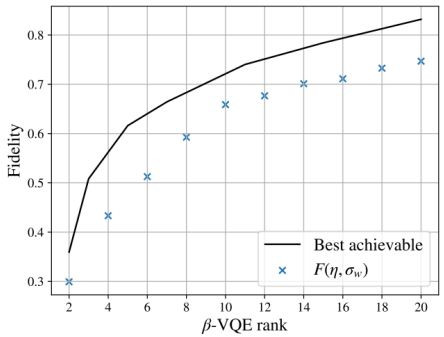


FIG. 5. Fidelity achieved between the target XXZ model η of size $n = 8$ and the QBM using the nested-loop algorithm, as a function of the rank of the β-VQE. The black line represents the fidelity between the model and an exact fixed rank approximation, and as such represents the best achievable fidelity for the corresponding fixed rank β-VQE.

C. Simulations with finite-sampling noise

All simulations in the previous sections assume access to the exact quantum state. This is possible only via classical statevector simulations which are intractable in general. To overcome this, one can execute the required circuits on a quantum computer. This introduces two new sources of error. First, the quantum computer is affected by various sources of hardware noise [31, 32]. These errors are dealt with by quantum error correction and are beyond the scope of this work. Second, the out-

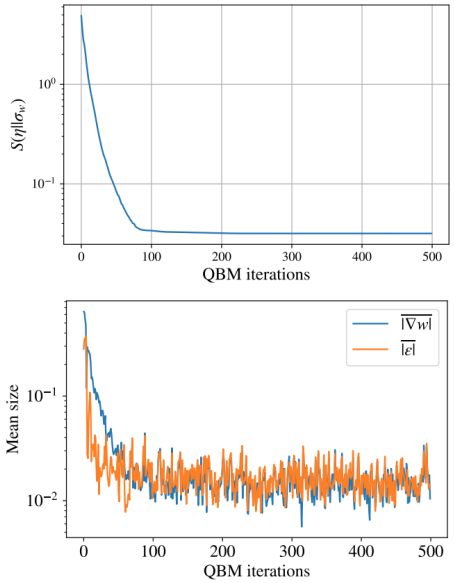


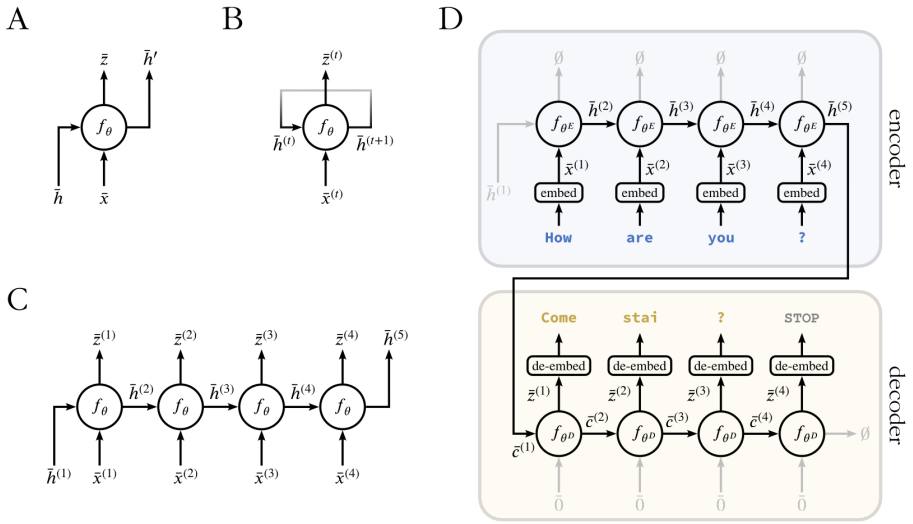
FIG. 6. Top: Relative entropy from the QBM to the salamander retina target using $R = 2$ and a depth $d = 2$ on a noiseless hardware simulation of $n = 4$ qubits. The number of shots is fixed to 5000 for both the gradient estimation of the β-VQE and the QBM. A fidelity of $F = 0.966$ is achieved. Bottom: The mean size $\overline{|\nabla S|}$ of the QBM gradient (blue) and the mean size of the error introduced by the β-VQE ϵ approximation (orange). The QBM converges until the error introduced by the β-VQE becomes the dominant factor of the gradient. The error decreases initially but finally fluctuates around $\frac{1}{\sqrt{5000}} \approx 0.14$ as expected.

One dimensional quantum XXZ model. Left: Error introduced by β VQE independent of rank approximation. Right: Quantum circuit measurement errors limits the convergence of β VQE and thus of QBM.

Large Language Models

- Recurrent architectures
  - Recurrency
  - Encoder, decoder
  - maximum likelihood training
  - draw backs
- Transformers
  - Attention, multi-headed attention
  - Transformer block
- Embeddings
- Surprising scaling
- Medical diagnosis with LLMs
- The future of AI

Sequence to sequence learning
[Hochreiter and Schmidhuber, 1997, Sutskever et al., 2014]



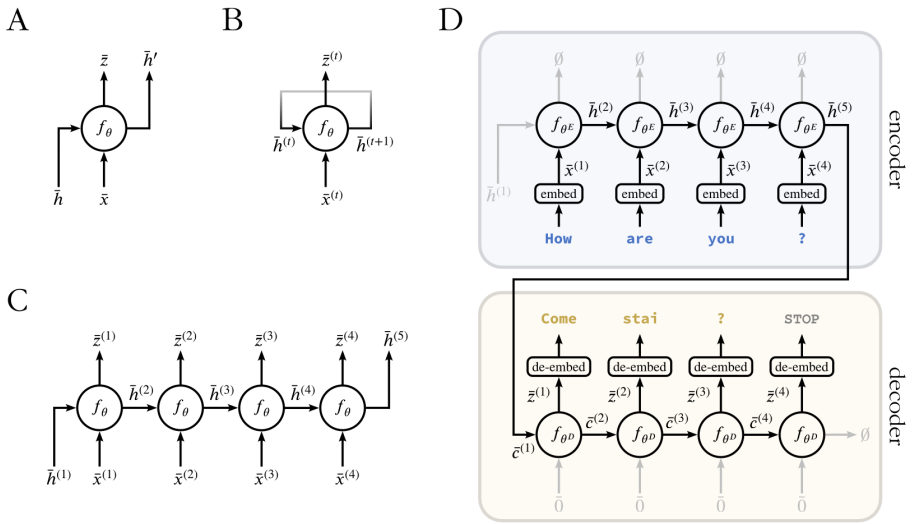
A, B: Recurrency:

$$y^t, h^{t+1} = f_\theta(x^t, h^t)$$

f_θ can be arbitrary functions, eg. deep neural networks.
x, y embedding vectors of tokenized words, h^t, h^{t+1} context vectors

This is 'word by word' translation, which fails in general.

Sequence to sequence learning
[Hochreiter and Schmidhuber, 1997, Sutskever et al., 2014]



C, D: Solution is to use two models: *encoder* and *decoder*.

Encoder maps entire input sequence $(x^1, \ldots, x^T)$ to fixed vector (h^{T+1}) and ignore the outputs y^t.
Decoder is initialized with h^{T+1}.

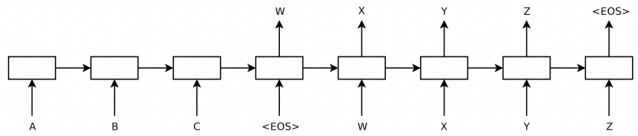


Figure 1: Our model reads an input sentence "ABC" and produces "WXYZ" as the output sentence. The model stops making predictions after outputting the end-of-sentence token. Note that the LSTM reads the input sentence in reverse, because doing so introduces many short term dependencies in the data that make the optimization problem much easier.

Learning by maximum likelihood [Sutskever et al., 2014]

The LSTM estimates the conditional probability $p(y_1, \ldots, y_{T'}|x_1, \ldots, x_T)$ where T, T' may be different. LSTM computes this through the hidden vector $c^1 = h^{T+1}$.

$$p(y^1, \ldots, y^{T'}|x^1, \ldots, x^T) = \prod_{t=1}^{T'} p(y^t|c^t, y^1, \ldots, y^{t-1})$$

$$= p(y^1|c^1, \langle EOS \rangle) p(y^2|c^2, y^1) \ldots p(\langle EOS \rangle |c^{T'}, y^{T'-1})$$

Note that each sentence ends with a special end-of-sentence symbol $\langle EOS \rangle$, which enables the model to define a distribution over sequences of all possible lengths.

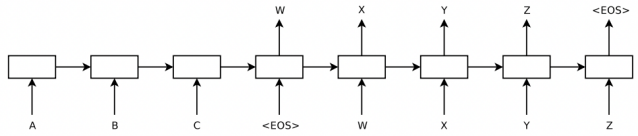


Figure 1: Our model reads an input sentence "ABC" and produces "WXYZ" as the output sentence. The model stops making predictions after outputting the end-of-sentence token. Note that the LSTM reads the input sentence in reverse, because doing so introduces many short term dependencies in the data that make the optimization problem much easier.

WMT English to French translation task [Sutskever et al., 2014]

Neural networks with 4 layers, with 1000 cells at each layer and 1000 dimensional word embeddings, with an input vocabulary of 160,000 and an output vocabulary of 80,000. We used a naive softmax over 80,000 words at each output. The resulting LSTM has 380M parameters of which 64M are pure recurrent connections (32M for the encoder LSTM and 32M for the decoder LSTM).

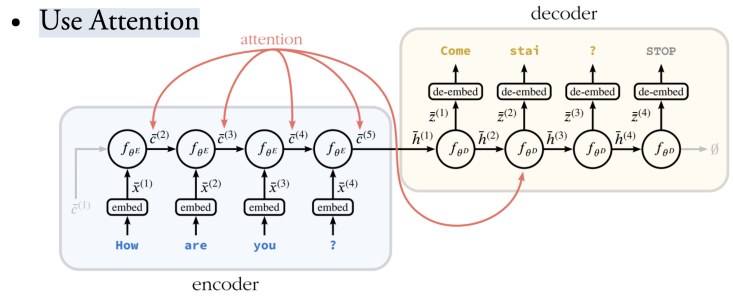
Method	test BLEU score (ntst14)
Baseline System [29]	33.30
Cho et al. [5]	34.54
State of the art [9]	**37.0**
Rescoring the baseline 1000-best with a single forward LSTM	35.61
Rescoring the baseline 1000-best with a single reversed LSTM	35.85
Rescoring the baseline 1000-best with an ensemble of 5 reversed LSTMs	**36.5**
Oracle Rescoring of the Baseline 1000-best lists	∼45

Table 2: Methods that use neural networks together with an SMT system on the WMT'14 English to French test set (ntst14).

LSTM outperforms for the first time a classical phrase based Statistical Machine Translation (SMT [29]) approach

Transformers

This basic encoder-decoder idea does not work well, because h^{T+1} must encode all input information for the decoding process.



The essential idea of the transformer is to build a function that maps $x \to x'$ based on a *context* $(c^1, \ldots, c^m)$:

$$x' = f(x, c^1, \ldots, c^m) = f\left(x + \sum_{i=1}^{m} \Omega(x, c^i) g(c^i)\right)$$

$\Omega(x, c^i)$ is the relevance, aka the *attention*, of c^i for x.

Attention

Attention basics

(vector notation)

$$\underset{d_K}{\bar{q}} = \underset{d_K \times d}{W^Q} \underset{d}{\bar{x}}$$

$$\underset{d_K}{\bar{k}^{(i)}} = \underset{d_K \times d_c}{W^K} \underset{d_c}{\bar{c}^{(i)}}$$

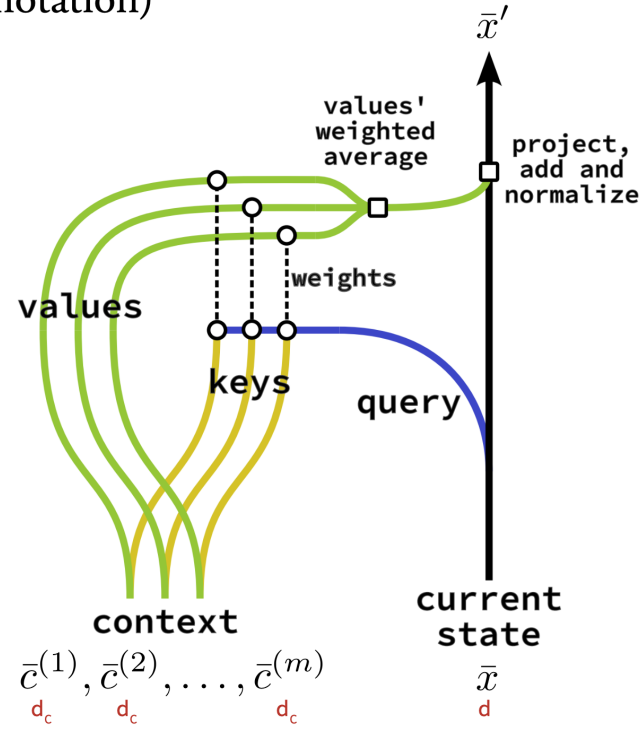
$$\underset{d_v}{\bar{v}^{(i)}} = \underset{d_v \times d_c}{W^V} \underset{d_c}{\bar{c}^{(i)}}$$

$$\omega_i = \frac{\bar{k}^{(i)} \cdot \bar{q}}{\sqrt{d_K}}$$

$$\underset{m}{\bar{\Omega}} = \text{softmax}\left(\underset{m}{\bar{\omega}}\right)$$

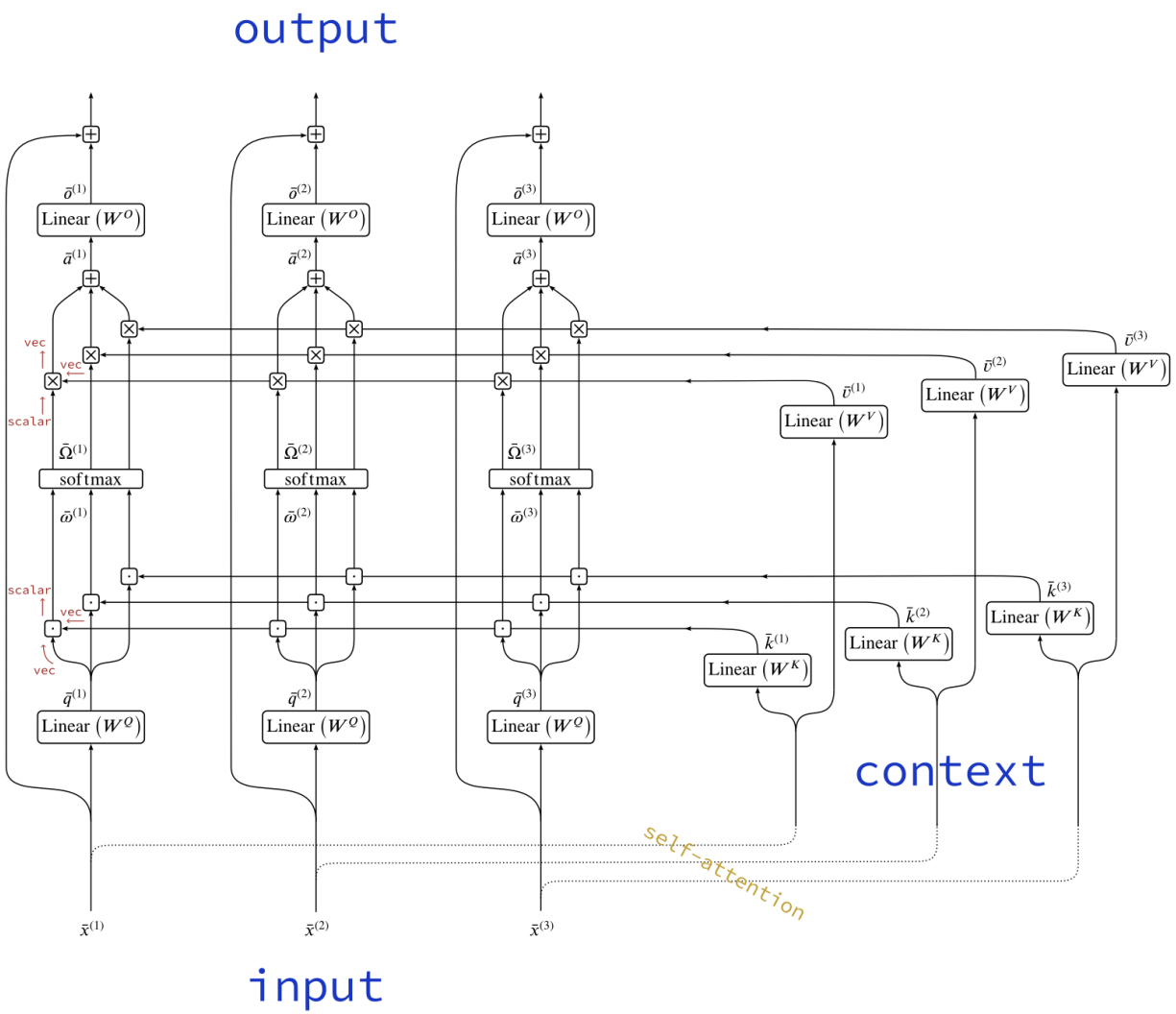
$$\underset{d_v}{\bar{a}} = \sum_i \Omega_i \underset{d_v}{\bar{v}^{(i)}}$$

$$\underset{d}{\bar{x}'} = \text{LayerNorm}\left(\underset{d}{\bar{x}} + \underset{d \times d_v}{W^O} \underset{d_v}{\bar{a}}\right)$$



$$x' = \text{Layernorm}\left(x + \sum_{i=1}^{m} \frac{e^{\omega^i} W^O W^V c^i}{\sum_j e^{\omega^j}}\right)$$

Self-attention



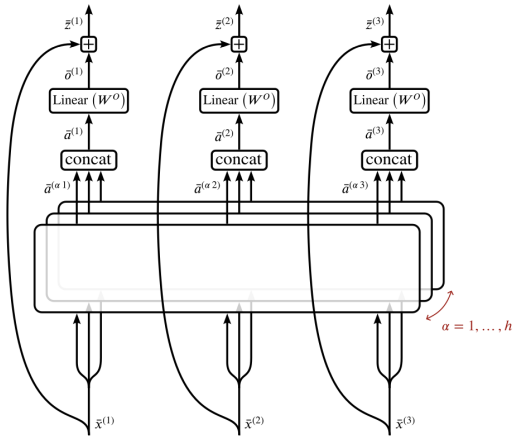
Multi-headed attention

Attention is a mapping $x^i, c^{1:m} \rightarrow a^i$ using the matrices W^Q, W^K, W^V. The output is $o^i = W^O a^i$ ($d_v \rightarrow d$ dimensions).

Multi-headed attention repeats this h times:

$$x^i, c^{1:m} \rightarrow_{W^{Q(\alpha)}, W^{K(\alpha)}, W^{V(\alpha)}} a^{\alpha,i}, \qquad \alpha = 1, \ldots, h$$

The output is $o^i = \sum_\alpha W^{O(\alpha)} a^{\alpha,i}$.



Transformer blocks

$$\bar{o}^{(i)} = \text{MultiHeadedAttention}\left(\bar{x}^{(i)}; x\right)$$ [the second x has the role of context]

$$\bar{f}^{(i)} = \text{LayerNorm}\left(\bar{x}^{(i)} + \bar{o}^{(i)}\right)$$ [first skip connection and layernorm]

$$\bar{g}^{(i)} = W^2\,\text{ReLU}\left(W^1 \bar{f}^{(i)} + \bar{b}^1\right) + \bar{b}^2$$ [2-layer MLP with parameters $W^1, \bar{b}^1$ and $W^2, \bar{b}^2$]

$$\bar{w}^{(i)} = \text{LayerNorm}\left(\bar{f}^{(i)} + \bar{g}^{(i)}\right)$$ [second skip connection and layernorm]

Attention is all you need [Vaswani et al., 2017]

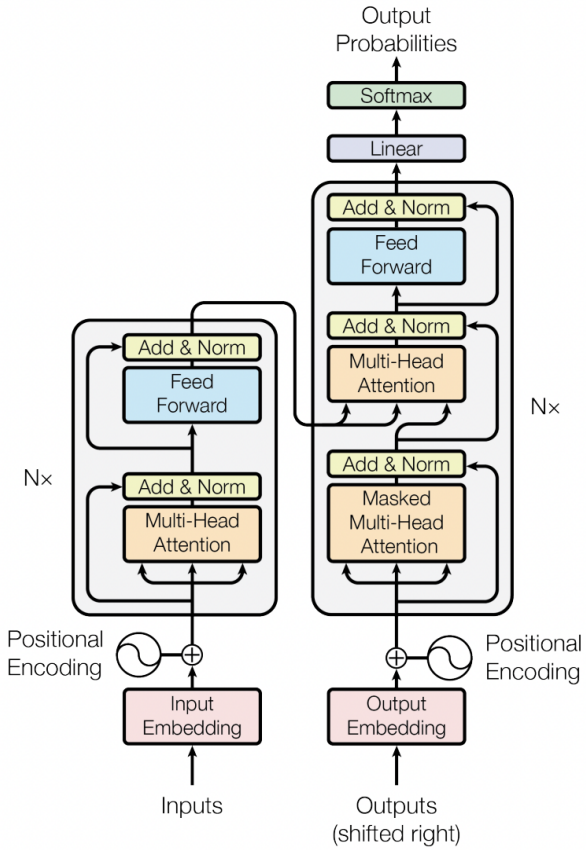
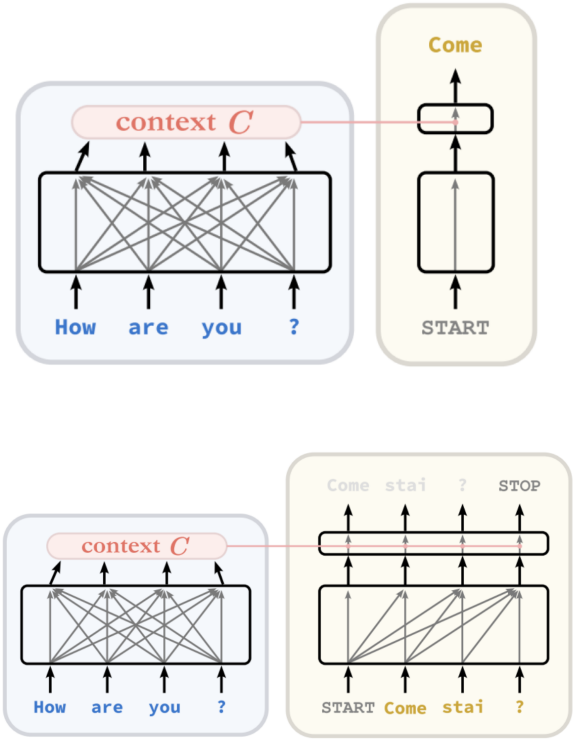


Figure 1: The Transformer - model architecture.



Training and results [Vaswani et al., 2017]

WMT 2014 English-German dataset consists of about 4.5 million sentence pairs. Sentences were encoded using byte-pair encoding, with common source-target vocabulary of about 37000 tokens.

WMT 2014 English-French dataset consists of 36M sentences using a 32000 word-piece vocabulary.

Training on one machine with 8 NVIDIA P100 GPUs. The base models was trained for 100,000 steps or 12 hours. The big models were trained for 300,000 steps (3.5 days).

Table 2: The Transformer achieves better BLEU scores than previous state-of-the-art models on the English-to-German and English-to-French newstest2014 tests at a fraction of the training cost.

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [15]	23.75			
Deep-Att + PosUnk [32]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [31]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [8]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [26]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [32]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [31]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [8]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	**$3.3 \cdot 10^{18}$**	
Transformer (big)	**28.4**	**41.0**	$2.3 \cdot 10^{19}$	

Results are better at fraction of the training time.

Tokens

The text is encoded as a sequence of *tokens* $s = 1, \ldots, T$. An individual tokens may be a character, a part of a words, a word, or a short sequence of words ('New York').

Disadvantage of character tokens is that they contain very little semantic information and that they require very long contexts.

Disadvantage of word tokens is that the number of tokens T is very (too) large; synonyms (capitals, plurals, conjugation, missspelling).

Nevertheless, is word tokenization most common.

An alternative is byte pair encoding, which is in between character and word tokenization.

- Start with character tokens (order 30).
- assign new token to most frequent consecutive pair and retokenize
- repeat up to a maximum number of tokens (10^4), keeping only the most frequent ones.

Embeddings

Encode each token $s = 1, \ldots, T$ as a one-off vector of length T: $t = (0, \ldots, 0, \underbrace{1}_{\text{position } s}, 0, \ldots, 0)$. The vector representation of the token is $x = W^E t$.

Curiously encoded words sometimes obtain meaningful numerical relations:

$$\text{'king'} - \text{'man'} + \text{woman'} = \text{'queen'}$$

The embedding matrix W^E is typically learned together with the other parameters of the Transformer.

De-embedding

The output of the Transformer is a vector y. It is mapped onto a token by computing $q = \left(W^E\right)^T y$ and either

1. compute $s = \text{argmax} q(s)$ or

2. compute $p(s) = \frac{e^{q(s)}}{\sum_{s'} e^{q(s')}}$ and sample from p.

(The latter is why ChatGPT gives each time different answers).

For translation to a different language one uses a de-embedding matrix $W^D \neq \left(W^E\right)^T$ which is also trained.

Performance scaling with data, parameters, and FLOPs [Wei et al., 2022]

Few shots prompting

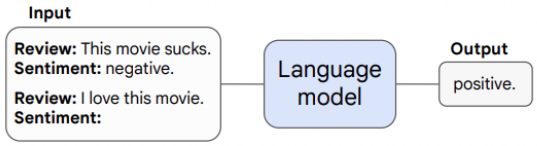


Figure 1: Example of an input and output for few-shot prompting.

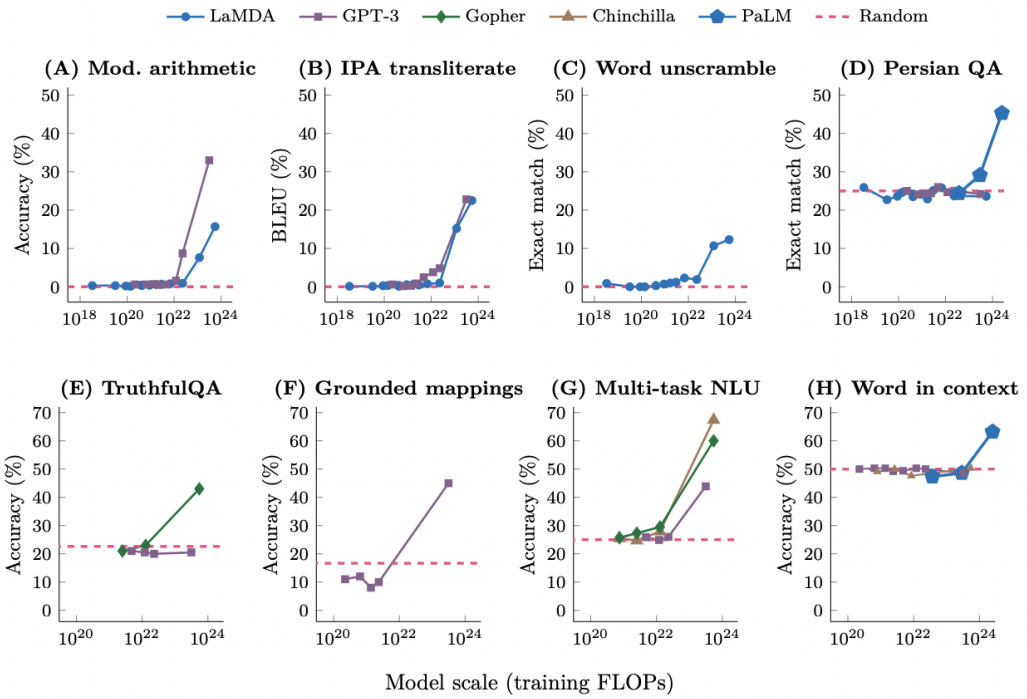


Figure 2: Eight examples of emergence in the few-shot prompting setting. Each point is a separate model. The ability to perform a task via few-shot prompting is emergent when a language model achieves random performance until a certain scale, after which performance significantly increases to well-above random. Note that models that used more training compute also typically have more parameters—hence, we show an analogous figure with number of model parameters instead of training FLOPs as the x-axis in Figure 11. A–D: BIG-Bench (2022), 2-shot. E: Lin et al. (2021) and Rae et al. (2021). F: Patel & Pavlick (2022). G: Hendrycks et al. (2021a), Rae et al. (2021), and Hoffmann et al. (2022). H: Brown et al. (2020), Hoffmann et al. (2022), and Chowdhery et al. (2022) on the WiC benchmark (Pilehvar & Camacho-Collados, 2019).

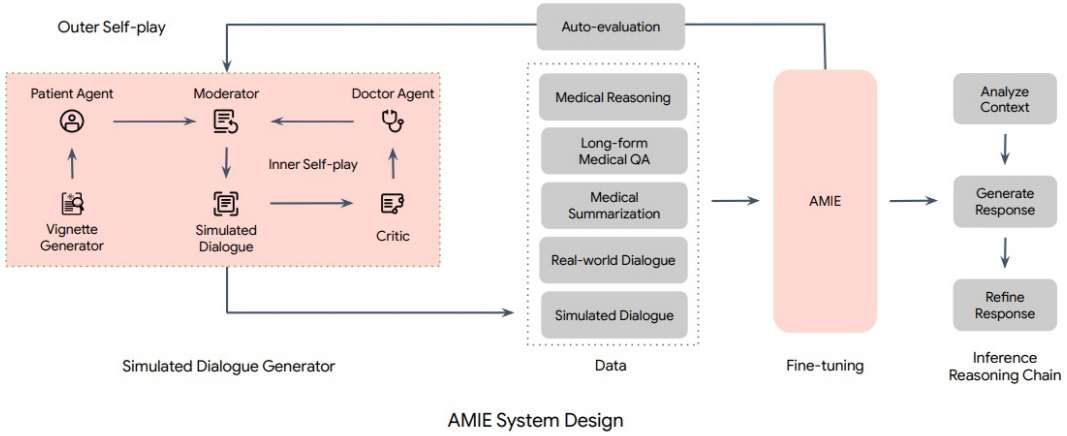
Towards Conversational Diagnostic AI

Tao Tu*,[1], Anil Palepu*,[1], Mike Schaekermann*,[1],
Khaled Saab[1], Jan Freyberg[1], Ryutaro Tanno[2], Amy Wang[1], Brenna Li[1], Mohamed Amin[1],
Nenad Tomasev[2], Shekoofeh Azizi[2], Karan Singhal[1], Yong Cheng[2], Le Hou[1], Albert Webson[2],
Kavita Kulkarni[1], S. Sara Mahdavi[2], Christopher Semturs[1],
Juraj Gottweis[1], Joelle Barral[2], Katherine Chou[1], Greg S. Corrado[1], Yossi Matias[1],
Alan Karthikesalingam†,[1] and Vivek Natarajan†,[1]

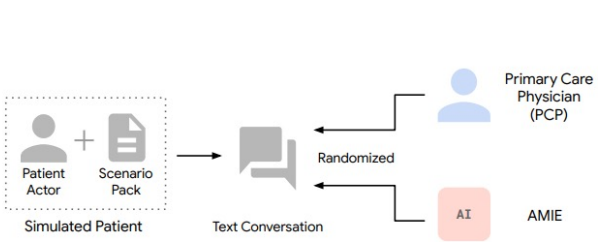
[1]Google Research, [2]Google DeepMind

At the heart of medicine lies the physician-patient dialogue, where skillful history-taking paves the way for accurate diagnosis, effective management, and enduring trust. Artificial Intelligence (AI) systems capable of diagnostic dialogue could increase accessibility, consistency, and quality of care. However, approximating clinicians' expertise is an outstanding grand challenge. Here, we introduce AMIE (Articulate Medical Intelligence Explorer), a Large Language Model (LLM) based AI system optimized for diagnostic dialogue. AMIE uses a novel self-play based simulated environment with automated feedback mechanisms for scaling learning across diverse disease conditions, specialties, and contexts. We designed a framework for evaluating clinically-meaningful axes of performance including history-taking, diagnostic accuracy, management reasoning, communication skills, and empathy. We compared AMIE's performance to that of primary care physicians (PCPs) in a randomized, double-blind crossover study of text-based consultations with validated patient actors in the style of an Objective Structured Clinical Examination (OSCE). The study included 149 case scenarios from clinical providers in Canada, the UK, and India, 20 PCPs for comparison with AMIE, and evaluations by specialist physicians and patient actors. AMIE demonstrated greater diagnostic accuracy and superior performance on 28 of 32 axes according to specialist physicians and 24 of 26 axes according to patient actors. Our research has several limitations and should be interpreted with appropriate caution. Clinicians were limited to unfamiliar synchronous text-chat which permits large-scale LLM-patient interactions but is not representative of usual clinical practice. While further research is required before AMIE could be translated to real-world settings, the results represent a milestone towards conversational diagnostic AI.

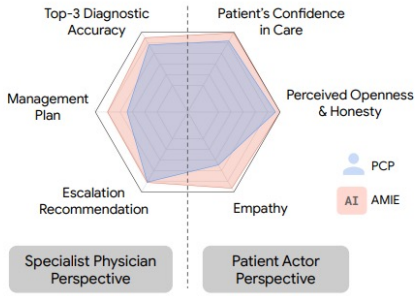
AMIE (Articulate Medical Intelligence Explorer)



AMIE System Design



Randomized Study Design for Remote
Objective Structured Clinical Examination (OSCE)



AMIE Outperforms PCPs on
Multiple Evaluation Axes for Diagnostic Dialogue

Google 2024

Training AMIE

- **Vignette Generator**: AMIE leverages web searches to craft unique patient vignettes given a specific medical condition.
- **Simulated Dialogue Generator**: Three LLM agents play the roles of patient agent, doctor agent, and moderator, engaging in a turn-by-turn dialogue simulating realistic diagnostic interactions.
- **Self-play Critic**: A fourth LLM agent acts as a critic to give feedback to the doctor agent for self-improvement. Notably, AMIE acted as all agents in this framework. We describe each component in detail below.

Patient Agent Instruction:
You are a patient chatting with a doctor over an online chat interface. The doctor has never met you before. <patient vignette> Respond to the doctor's questions honestly as they interview you, asking any questions that may come up.

Doctor Agent Instruction:
You are an empathetic clinician asking a patient about their medical history over an online chat interface. You know nothing about the patient in advance. Respond to the patient with a single-turn response to better understand their history and symptoms. Do not ask more than two questions. If the patient asks a question, be sure to answer it appropriately.

Moderator Instruction:
The following is a conversation between a doctor and a patient: <dialog> The conversation should only come to an end if the doctor has finished giving the patient a diagnosis and treatment plan and the patient has no questions left. A conversation also comes to an end if the doctor or patient says goodbye. Question: has the conversation come to an end? Yes or No.

A.2 Example of Simulated Dialogue After Self-critique

Condition: Carpal Tunnel Syndrome
A 55-year-old female presents with a 3-month history of the following symptoms.
****Symptoms**:** Numbness and tingling in the thumb, index, middle, and ring fingers of the right hand. The numbness is worse at night and wakes her up from sleep. She also has pain in the hand and wrist that radiates up the forearm. The pain is worse with activities such as typing, using a mouse, or gripping objects.
****Past Medical History**:** Hypertension, hypothyroidism.
****Past Surgical History**:** None.
****Past Social History**:** Works as a data entry clerk.
****Patient Questions**:** What is causing my symptoms? What can I do to make them better?

Round 1 Simulated Dialogue

Doctor Agent (AMIE): So, how can I help you today?
Patient Agent (AMIE): Hi, I'm a 55-year-old woman and I've been having numbness and tingling in my right hand for the past 3 months. It's worse at night and it wakes me up from sleep. I also have pain in my hand and wrist that goes up my forearm. The pain is worse when I type, use a mouse, or grip objects. I'm wondering what's causing these symptoms and what I can do to make them better.
Doctor Agent (AMIE): I understand your concern. To better advise you, could you tell me:
1) Aside from the numbness and pain, have you noticed any weakness in your hand, like difficulty holding small items or clumsiness?
2) Any neck pain, or pain radiating down your arm beyond the forearm?
3) Do you have any medical conditions we should be aware of, and what medications do you take regularly?
Patient Agent (AMIE): 1. I have noticed some weakness in my hand. It's difficult to grip objects and I've dropped things a few times. 2. I don't have any neck pain or pain radiating down my arm beyond the forearm. 3. I have hypertension and hypothyroidism. I take levothyroxine and lisinopril for these conditions.
Doctor Agent (AMIE): What kind of work do you do? Does it involve repetitive hand movements, like typing or assembly? Have you had any injuries to your right hand or wrist in the past, even if long ago?
Patient Agent (AMIE): I work as a data entry clerk, so I do a lot of typing. I haven't had any injuries to my right hand or wrist.
Doctor Agent (AMIE): Aside from the numbness, pain, and weakness, have you noticed any swelling in your hand or wrist, especially in the morning?
Patient Agent (AMIE): I haven't noticed any swelling in my hand or wrist.
Doctor Agent (AMIE): This is helpful. Based on what you've shared, two possibilities come to mind. One is **carpal tunnel syndrome**, which is common with repetitive hand use and causes the exact symptoms you describe. The other is **cervical radiculopathy**, a nerve issue in the neck, but that usually has neck pain too.
Patient Agent (AMIE): I see. What can I do to treat these conditions?
Doctor Agent (AMIE): For now, I'll focus on the more likely one, carpal tunnel. **First**, wrist splinting, especially at night, is key. **Second**, over-the-counter pain meds like ibuprofen can help. **Third**, and this is crucial, **ergonomic evaluation** of your workstation is needed to reduce strain on your hand.
Patient Agent (AMIE): Thank you for the advice. I will try those things. I'll also look into getting an ergonomic evaluation of my workstation.
Doctor Agent (AMIE): Please do. **If symptoms don't improve significantly in 4-6 weeks**, or if weakness worsens rapidly, an in-person evaluation is needed. A doctor can do nerve tests and confirm the diagnosis, and stronger options like steroid injections or surgery exist, but let's start conservative.
Patient Agent (AMIE): That sounds like a good plan. I'll follow your advice and see how things go. I'll be sure to contact a doctor if my symptoms don't improve or if they get worse. Thank you for your help.

Round 1 Critique for Doctor Agent (AMIE)

Overall, the doctor did a good job of gathering information and explaining the patient's condition in a clear, concise manner. The questions were targeted to differentiate between carpal tunnel syndrome and other potential causes, leading to a more confident diagnosis.

Here are a few specific suggestions for improvement:
1. ****Early Reassurance:**** After the initial symptom description, a brief reassurance like, "These are concerning symptoms, but we'll work together to figure this out," can build rapport early on.
2. ****Symptom Specificity:**** Instead of asking general weakness, ask, "Which fingers are weak? Is it gripping, pinching, or fine movements?" This helps pinpoint nerve involvement.
3. ****Neck Pain:**** Instead of just asking about presence, ask, "Does neck movement make hand symptoms better/worse? Any tingling down the BACK of your arm?" This helps rule out cervical issues more definitively.
4. ****Differential:**** Mentioning other possibilities, like cubital tunnel syndrome or even arthritis, shows broader thinking, even if less likely.
5. ****Treatment Nuance:**** Instead of just listing options, tailor them: "Splinting helps MOST at night, NSAIDs are for WHEN pain flares, ergonomics is KEY to PREVENTING worsening."

These refinements make the diagnosis more robust and the plan more actionable, while still maintaining the doctor's empathetic and professional tone.

Round 2 Simulated Dialogue

Doctor Agent (AMIE): So, how can I help you today?
Patient Agent (AMIE): Hi, I've been having numbness and tingling in my right hand for the past few months. It's worse at night and wakes me up from sleep. I also have pain in my hand and wrist that goes up my forearm. The pain is worse when I type, use a mouse, or grip objects. I'm wondering what's causing these symptoms and what I can do to make them better.
Doctor Agent (AMIE): These are concerning symptoms, but we'll work together to figure this out. To get a clearer picture, **which specific fingers are weak?** Is it trouble gripping, pinching, or fine movements like buttoning?
Patient Agent (AMIE): I have weakness in my thumb, index, middle, and ring fingers. I have trouble gripping objects, such as opening jars or lifting heavy objects. I also have difficulty with fine movements, such as buttoning my shirt or typing on a keyboard.
Doctor Agent (AMIE): Understood. Now, **does moving your neck or head in any direction make the hand symptoms better or worse?** Also, any tingling specifically down the BACK of your arm, not just the side?
Patient Agent (AMIE): I don't think moving my neck or head makes the symptoms any better or worse. And I don't have any tingling down the back of my arm, just the side.
Doctor Agent (AMIE): That's helpful to know. For context, **what kind of work do you do?** Does it involve repetitive hand movements, like typing or assembly?
Patient Agent (AMIE): I work as a data entry clerk, so I do a lot of typing and mousing. I've been doing this job for the past 10 years.
Doctor Agent (AMIE): Given your symptoms and job, **carpal tunnel syndrome** is likely, where a nerve in the wrist is compressed. **Cubital tunnel syndrome**, affecting the elbow, can be similar, but less common. Any elbow pain or numbness on the pinky side of the hand?
Patient Agent (AMIE): I don't have any elbow pain or numbness on the pinky side of my hand.
Doctor Agent (AMIE): That makes carpal tunnel more probable. **To be thorough, any medical conditions or medications we should be aware of?**
Patient Agent (AMIE): I have hypertension and hypothyroidism. I take medication for both of these conditions. I'm also taking a multivitamin and a fish oil supplement.
Doctor Agent (AMIE): Good to know. Based on everything, **carpal tunnel is the most likely cause**. Treatment focuses on reducing nerve pressure.
First, wrist splinting, especially at night, is key. This keeps the wrist neutral, preventing nerve pinching while you sleep, which is often when symptoms flare.
Second, over-the-counter pain meds like ibuprofen can help WHEN pain is bad, but they don't address the root cause.
MOST IMPORTANTLY, an ergonomic evaluation of your workstation is crucial. This means adjusting your chair, keyboard, mouse, etc., to reduce the strain on your hand during the day, preventing further damage.
Patient Agent (AMIE): Thank you for the information. I will try the wrist splinting and the ergonomic evaluation. I am also wondering if there are any other treatments that I can try, such as physical therapy or surgery?
Doctor Agent (AMIE): **Physical therapy is definitely an option,** they can teach exercises to strengthen your hand and wrist, improving function. **Surgery is considered if conservative measures fail** after a few months, but let's see how your body responds first.
Patient Agent (AMIE): That sounds great. I will start with the conservative measures and see how I do. I will also look into physical therapy. Thank you for your help.

A.4.1 Example AMIE Consultation

Patricia Wilson (45 yr) -

Chest Pain

Probable diagnosis: Angina or non-ST-elevation myocardial infarction

Plausible alternative diagnoses: Biliary disease, Dyspepsia

AMIE's Top 10 DDx:

1. Unstable Angina
2. Acute Myocardial Infarction
3. Stable Angina
4. Gastroesophageal Reflux Disease (GERD)
5. Atypical Chest Pain
6. Costochondritis
7. Peptic Ulcer Disease
8. Gallbladder Disease (biliary colic)
9. Panic Attack/Anxiety
10. Pulmonary Embolism

Online Text-based Consultation with AMIE

Patient Actor:
Hi doctor, I experienced central chest pain accompanied by discomfort in the upper stomach that happened two hours ago

AMIE:
I'm sorry to hear you're experiencing this discomfort, especially in two key areas. To help me get a clearer picture, can you describe the chest pain in more detail? Is it a pressure, tightness, burning, stabbing sensation, or something else entirely?

Patient Actor:
The chest pain I experienced was more of a pressure and tightness, especially in the center of my chest. It felt like there was a heavy weight on my chest, and the discomfort extended to my upper stomach. I wouldn't describe it as a burning or stabbing sensation, more like a continuous pressure that was quite noticeable. The heaviness also seemed to radiate down my left arm. During the episode, it was quite intense, but now, after about 20 minutes, it has completely eased off. I hope this provides a clearer picture of what I went through

AMIE:
That's a very clear description, thank you. It helps me narrow down possibilities, but to give you the best advice, I need some more background. Firstly, what is your age and do you have any significant medical history? This includes things like heart problems, high blood pressure, or past surgeries.

Patient Actor:
I'm 45 years old, and in terms of medical history, about five years ago, I was diagnosed with gallstones. However, I opted not to have surgery at that time, and it hasn't caused any recent issues. I had varicose vein stripping done 15 years ago, and when I was 23, I had pneumonia. As for other conditions, I don't have a history of heart problems or high blood pressure. Currently, I'm not on any prescribed medications

AMIE:
45 is relatively young for heart problems, which is reassuring. However, your symptoms, especially the pressure-like pain and radiation to your arm, ARE concerning for a few possibilities. To help me narrow it down further:

Family history: Is there any heart disease, like heart attacks or angina, in your close relatives?

Lifestyle: Do you currently smoke or drink alcohol?

Patient Actor:
In my family history, my father had high blood pressure, and I believe he had a heart attack in his mid-fifties. my mother tends to get chest infections occasionally. Regarding lifestyle, I do smoke about 10 cigarettes per day, and I have been doing so since the age of 25. However, I drink alcohol infrequently, mainly on special occasions.

AMIE:
Thank you for providing that detailed information, it's very helpful. Given your family history of heart problems, especially your father's heart attack, and your smoking habit, we can't ignore the possibility that this is heart-related. However, the fact that your pain has resolved on its own makes a typical heart attack less likely.

It's possible this is angina, which is chest pain caused by reduced blood flow to the heart. It can feel similar to a heart attack, but is usually less severe and goes away with rest. Another possibility is GERD, which is heartburn caused by stomach acid refluxing into the esophagus. This can also cause chest pain, but it's usually accompanied by a burning sensation.

To be sure what's causing your symptoms, I strongly recommend you seek immediate in-person evaluation . This will likely involve an electrocardiogram (ECG) to check your heart's electrical activity, and blood tests to look for signs of heart damage or other issues. Depending on the initial findings, further tests may be needed.

I understand this is worrying, but it's crucial to act fast even if it turns out NOT to be a heart attack. These symptoms are significant and need prompt attention to rule out serious causes and get you the right treatment.

Patient Actor:
I appreciate your thorough explanation, and I understand the urgency of the situation. Given the potential seriousness of the symptoms, I will seek immediate in-person evaluation as you've advised. The possibility of angina or GERD is certainly concerning, and I want to ensure that I receive the necessary tests and examinations to determine the cause and appropriate treatment. Thank you for your guidance, and I'll proceed to the clinic promptly to address this issue

AMIE:
I'm glad to hear you're taking this seriously, as prompt action is key here. At the clinic, be sure to relay ALL the details we discussed, including family history, for a full risk assessment. This will help them determine the best next steps, which may be immediate treatment or referral to a cardiologist for specialized care.

Patient Actor:
Thank you again for your assistance and advice

AMIE:
You're very welcome. My goal is to help you get the best care possible, so don't hesitate to reach out if further questions arise after your in-person evaluation. Take care, and I hope you get answers and relief soon.

A.4.2 Example PCP Consultation

Patricia Wilson (45 yr) -

Chest Pain

Probable diagnosis: Angina or non-ST-elevation myocardial infarction

Plausible alternative diagnoses: Biliary disease, Dyspepsia

PCP's Top 10 DDx:

1. Acute Coronary Syndrome
2. GERD
3. Biliary Colic
4. Pulmonary Embolism
5. Pancreatitis
6. N/A
7. N/A
8. N/A
9. N/A
10. N/A

Online Text-based Consultation with PCP

Patient Actor:
Hi Doctor. how are you?

Primary Care Physician:
I'm great. How can I help you?

Patient Actor:
I recently experienced an episode of chest pain and discomfort. About two hours ago, while walking home after lunch with my daughter, I developed central chest pain along with some discomfort in the upper stomach.

Primary Care Physician:
I'm sorry to hear that. Are you still in pain?

Patient Actor:
No, I'm not currently in pain. The chest pain and discomfort I experienced about two hours ago lasted for around 20 minutes, and since then, I haven't felt any pain.

Primary Care Physician:
I'm glad you are not uncomfortable anymore. Chest pain can sometimes be quite serious, though. Do you mind if I ask you so more questions? For example, have you ever had similar pain before?

Patient Actor:
No, this is the first time when i'm having this kind of pain

Primary Care Physician:
Is it associated with any other symptoms? Like nausea/vomiting? Or shortness of breath?

Patient Actor:
Yes, during the episode of chest pain, I did experience nausea. Additionally, there was a heavy feeling in my left arm. I didn't feel sweaty or breathless with the pain,

Primary Care Physician:
How old are you, may I ask? And what kind of medical conditions do you have?

Patient Actor:
I'm 45 years old. I'm not on any prescribed medications, but I do take a vitamin preparation that I buy from the chemist.

Primary Care Physician:
Do you know what type of Vitamin? Also, please confirm that you do not have any hyigh blood pressure, diabetes, high cholesterol, smoking history, or family history of heart attacks.

Patient Actor:
I take a vitamin preparation that I purchase from the chemist, but I don't have the specific details about its composition. As for my health history, I don't have high blood pressure, diabetes, or high cholesterol. However, I do have a smoking history of about 10 cigarettes per day since the age of 25. In terms of family history, my father had high blood pressure and had a heart attack in his mid-fifties. I have two older brothers, and I recently found out that one of them has high cholesterol.

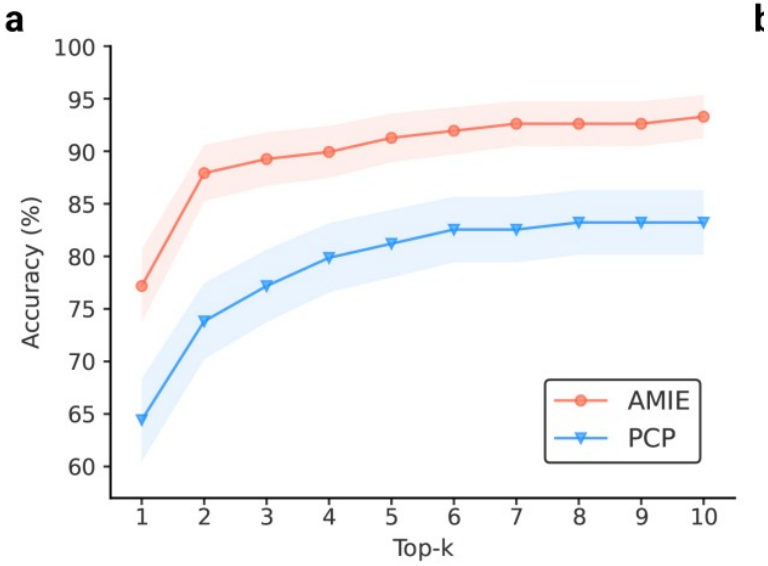
Primary Care Physician:
Can you describe the quality of the pain? Was there anything that made it better or worse? For example, sometimes pain is aggravated by exertion - walking at a fast pace/relieved by rest. Sometimes chest pain is made worse by taking a deep inspiration. do either of these apply? Anything else you can think of?

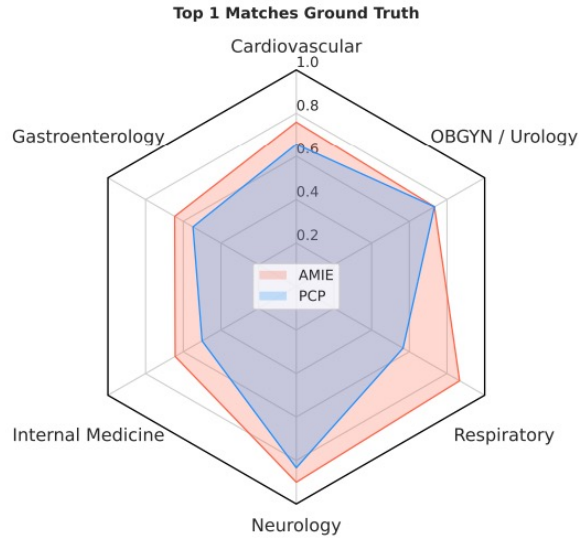
Patient Actor:
I wouldn't say the pain was aggravated by taking a deep breath, but walking seemed to worsen it. Resting or sitting down provided some relief. It wasn't a sharp or stabbing pain, more like a continuous pressure.

Primary Care Physician:
Did you have any palpitations? Did you feel as though you might faint? Any other symptoms?

Patient Actor:
No, I didn't experience palpitations during the episode of chest pain. I also didn't feel as though I might faint. The main symptoms were the central chest pain, discomfort in the upper stomach, a heavy feeling in my left arm, and nausea. While the pain was intense during the first few minutes, it became a constant and unpleasant sensation, and it completely eased off after about 20 minutes. Since then, I haven't had any lingering symptoms, and I currently feel well

Primary Care Physician:
I've heard enough to know that your problem needs to be escalated. I would like for you to go to your nearest emergency room should your pain recur.





Tested on 149 patient cases from medical exams: cardiovascular (29), respiratory (30), gastroenterology (31), neurology (30), urology, obstetric, and gynecology domains (15), and internal medicine (14).

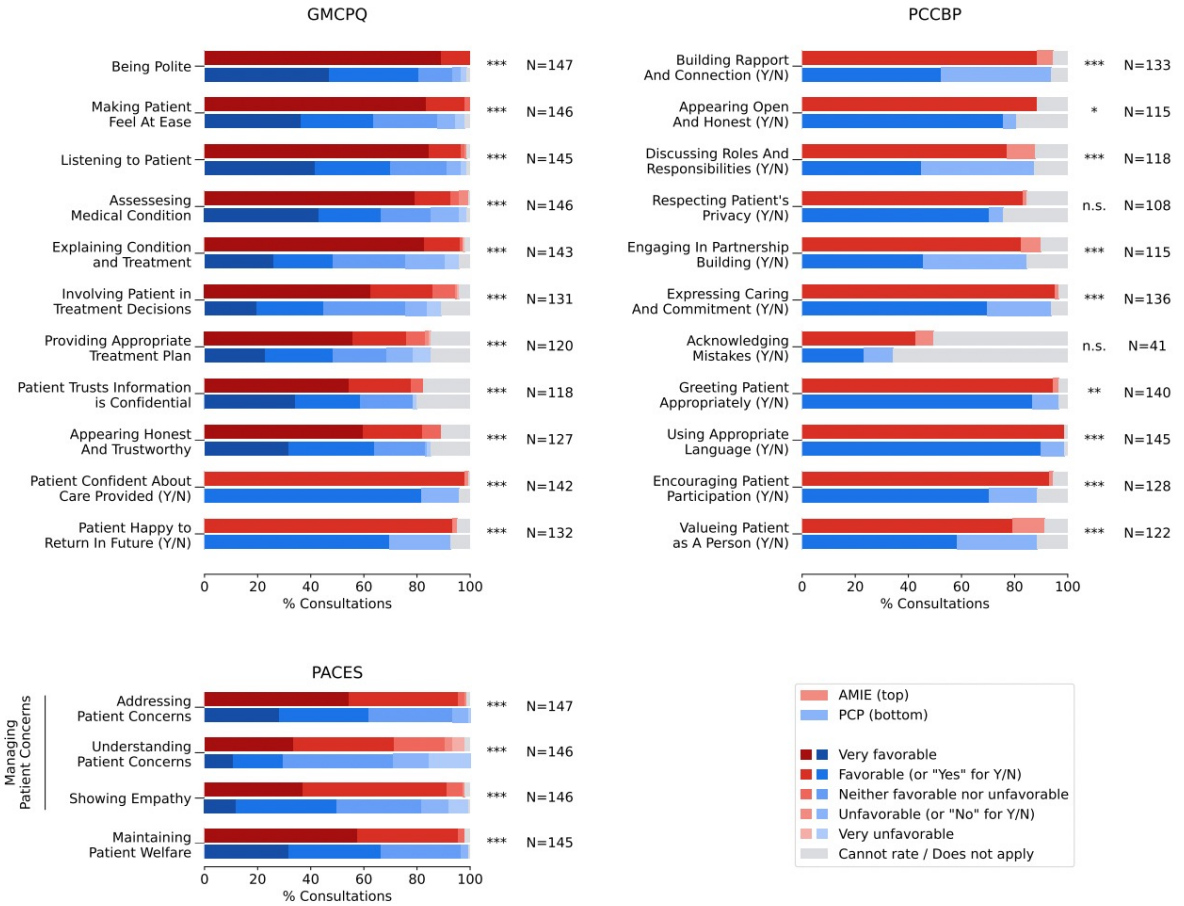


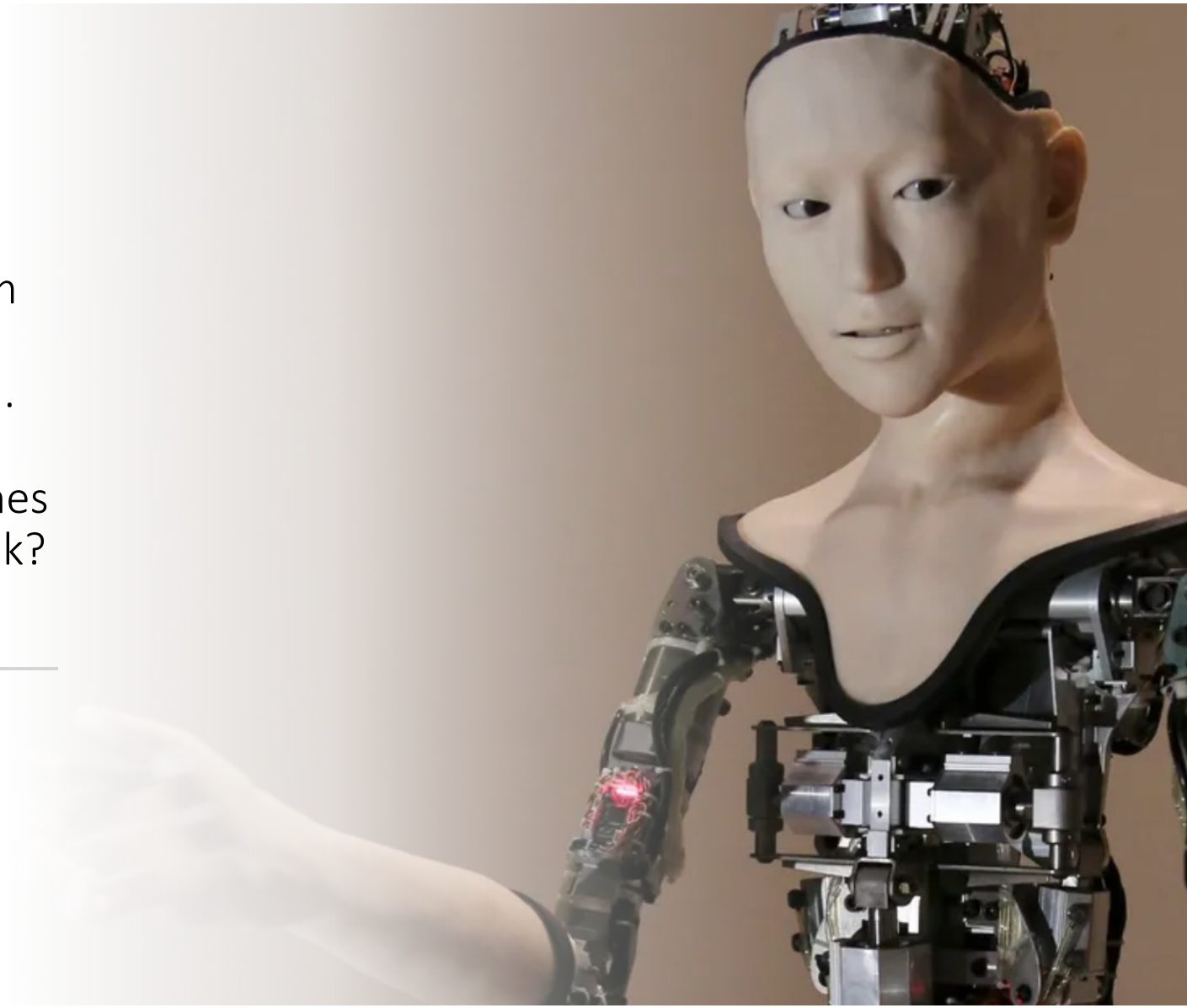
Figure 4 | Patient actor ratings. Conversation qualities as assessed by patient actors upon conclusion of the consultation. For illustration purposes, all responses from five-point rating scales were mapped to a generic five-point scale ranging from 'Very favorable' to 'Very unfavorable'. For Yes/No questions, a (positive) 'Yes' response was mapped to the same color as 'Favorable' and a (negative) 'No' response to the same color as 'Unfavorable'. Rating scales were adapted from the General Medical Council Patient Questionnaire (GMCPQ), the Practical Assessment of Clinical Examination Skills (PACES), and a narrative review about Patient-Centered Communication Best Practice (PCCBP). Details on question wording and response options are provided in Section A.1. Asterisks represent statistical significance ($* : p < 0.05$, $** : p < 0.01$, $*** : p < 0.001$, $n.s.$: not significant).

Patient actors rated AMIE superior to human doctors on 24 of 26 criteria

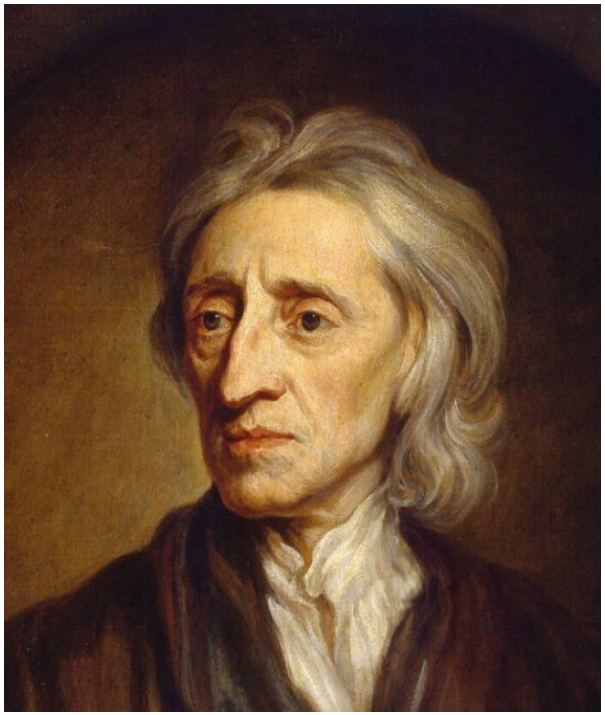
Specialist physicians rated AMIE superior to human doctors on 28 of 32 criteria (not shown)

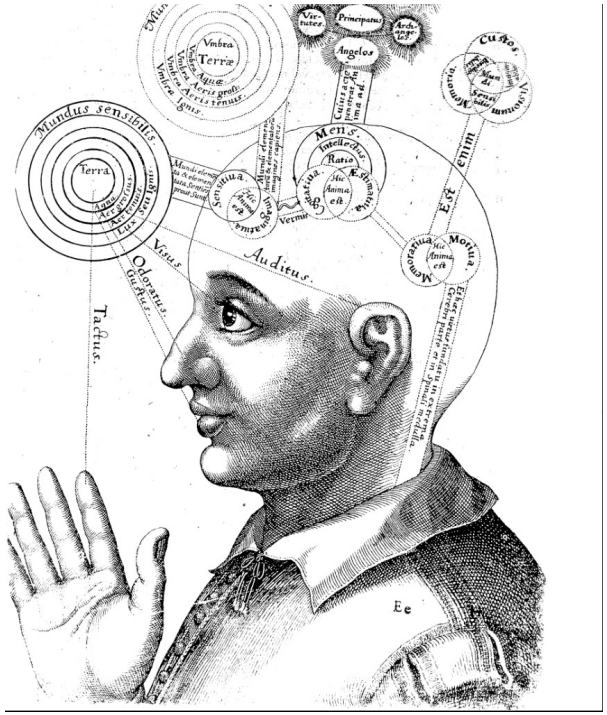
We zijn wellicht in staat om machines te maken die menselijk gedrag vertonen.

Maar hebben deze machines ook bewustzijn zoals u en ik?



Wat is bewustzijn?





- John Locke (1690): ”Consciousness is the perception of what passes in a man's own mind"


Bewustzijn is subjectief:
ik heb alleen toegang tot mijn eigen bewustzijn en niet dat van anderen.
Nagel 1974
AI is objectief:
het gaat over het realiseren van menselijk gedrag in een machine
Nagel 1974

AI is een computerprogramma, software en hardware.

Is de software bewust? Of de hardware? Of de combinatie?

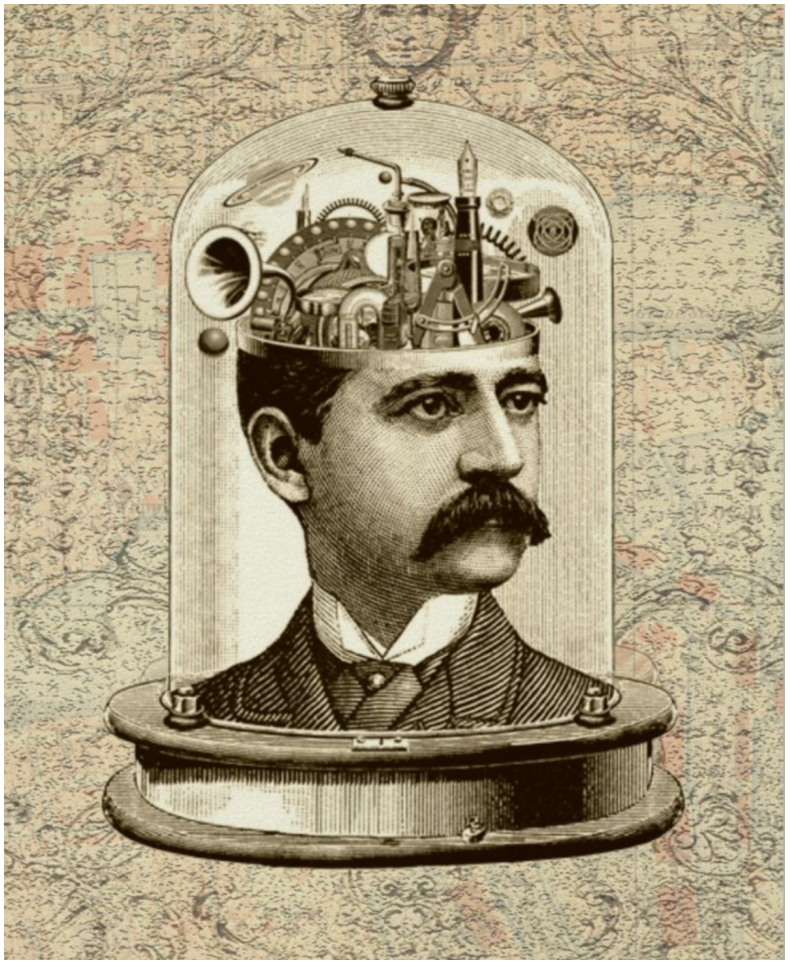
```
 *
 * @var boolean
 */
define('PSI_INTERNAL_XML', false);

if (version_compare("5.2", PHP_VERSION, ">")) {
    die("PHP 5.2 or greater is required!!!");
}
if (!extension_loaded("pcre")) {
    die("phpSysInfo requires the pcre extension to php in order to work
        properly.");
}

require_once APP_ROOT.'/includes/autoloader.inc.php';

// Load configuration
require_once APP_ROOT.'/config.php';

if (!defined('PSI_CONFIG_FILE') || !defined('PSI_DEBUG')) {
    $tpl = new Template("/templates/html/error_config.html");
    echo $tpl->fetch();
    die();
                                    javascript
                                    strtolower(
```



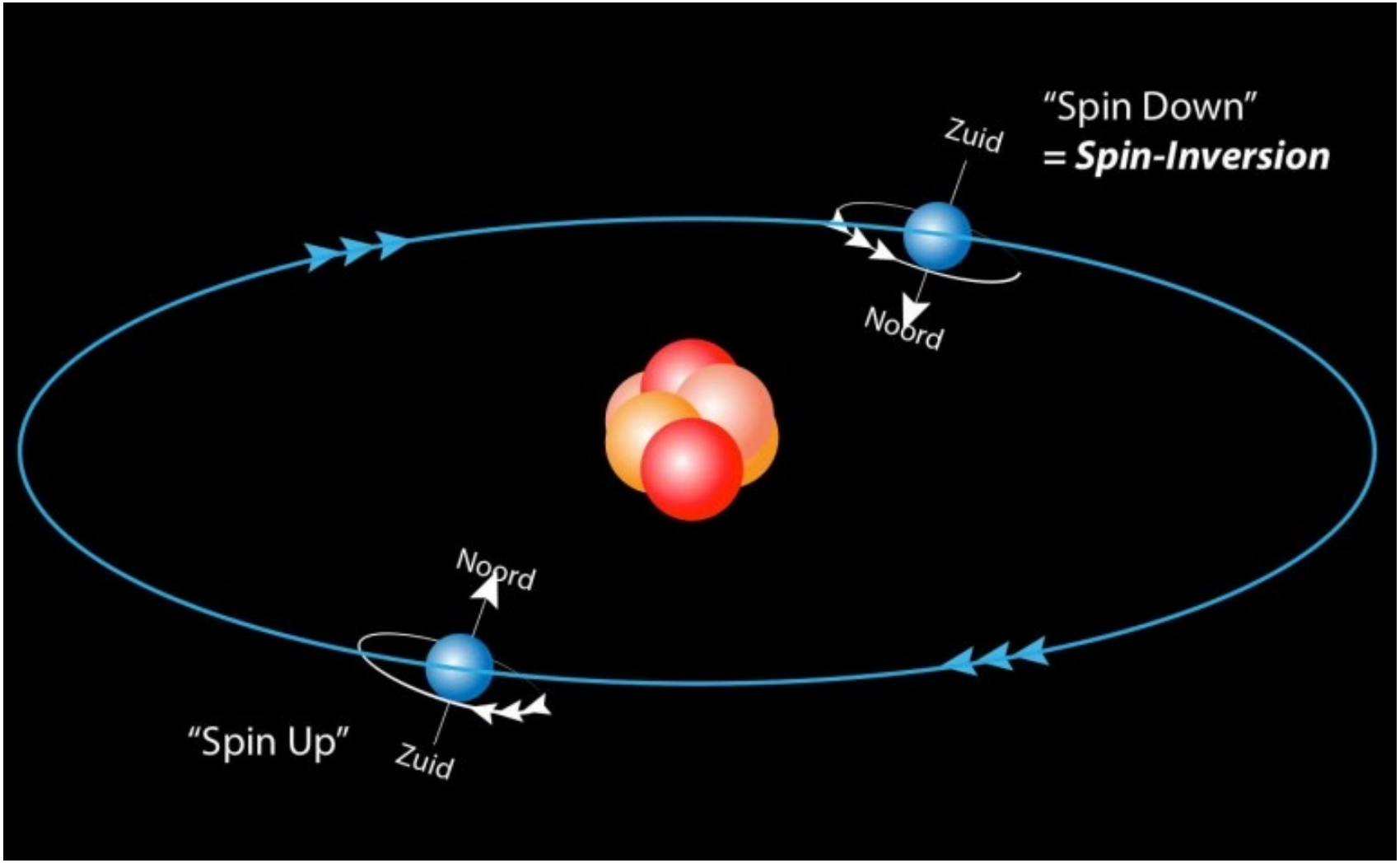
Een klassiek model van de werking van het brein is een machine.

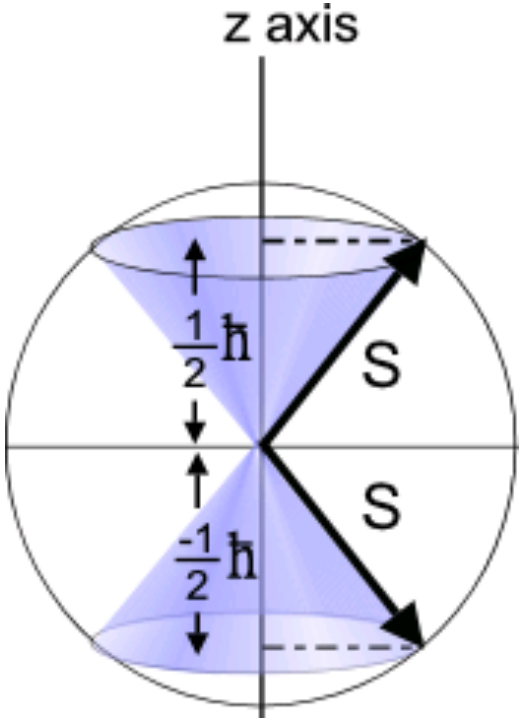
Een keuze

Optie 1: Een quantum mechanism voor bewustzijn?

Optie 2: Accepteer dat niet alles wiskundig te beschrijven is.

Quantum mechanica


"Spin Down"
= Spin-Inversion
Zuid
Noord
Noord
"Spin Up"
Zuid

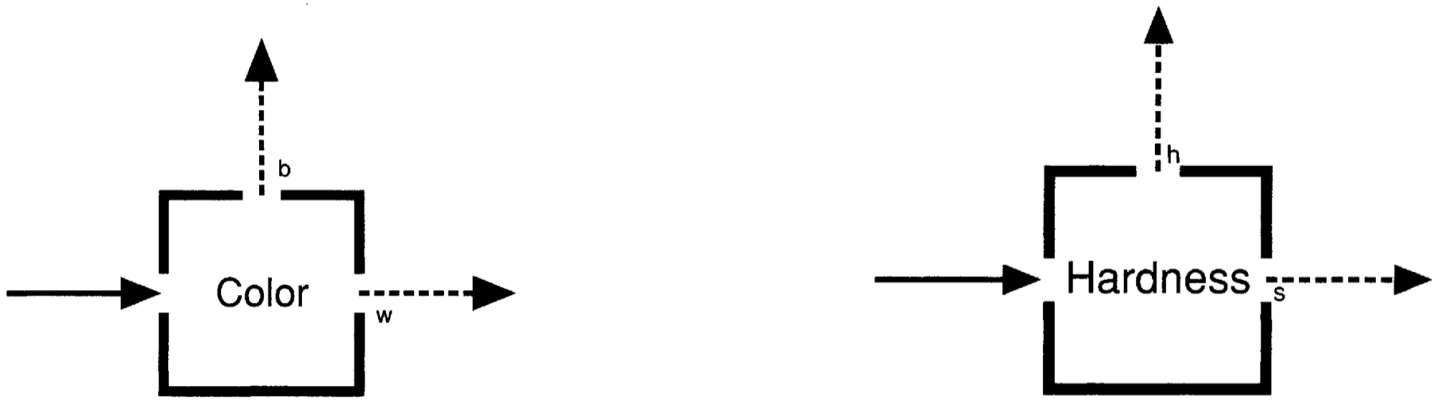


Spin is een vector met 3 componenten, maar niet alle 3 componenten kunnen gelijktijdig worden gemeten

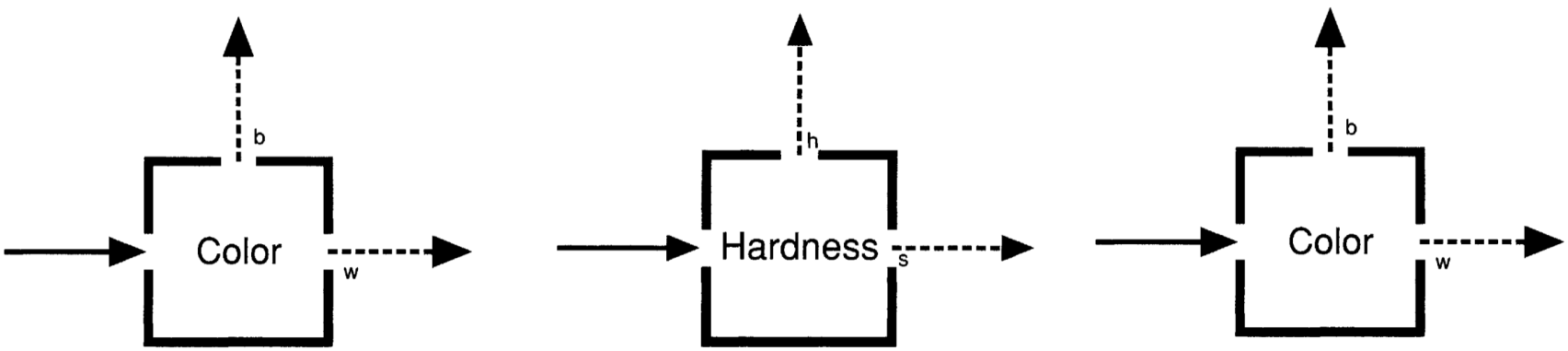
Heisenberg’s onzekerheids principe

Electronen hebben twee kleuren (b, w) en twee hardheden (h,s).

Je kunt apparaten bouwen die kleur of hardheid kunnen meten.

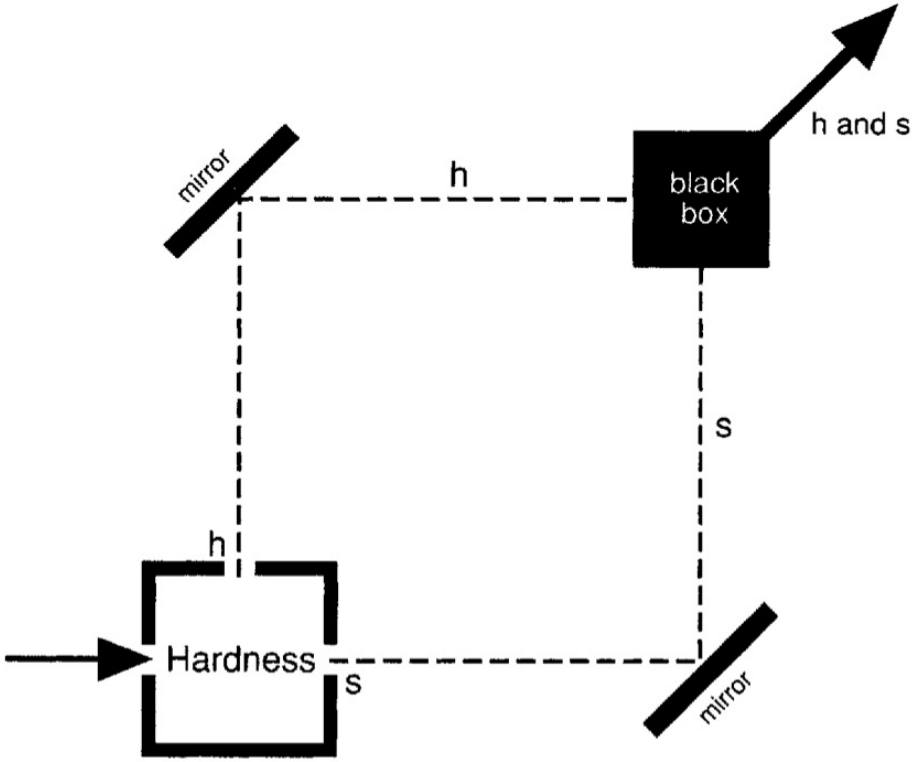


Heisenberg’s onzekerheids principe



Kleur en hardheid kunnen niet allebei worden bepaald.

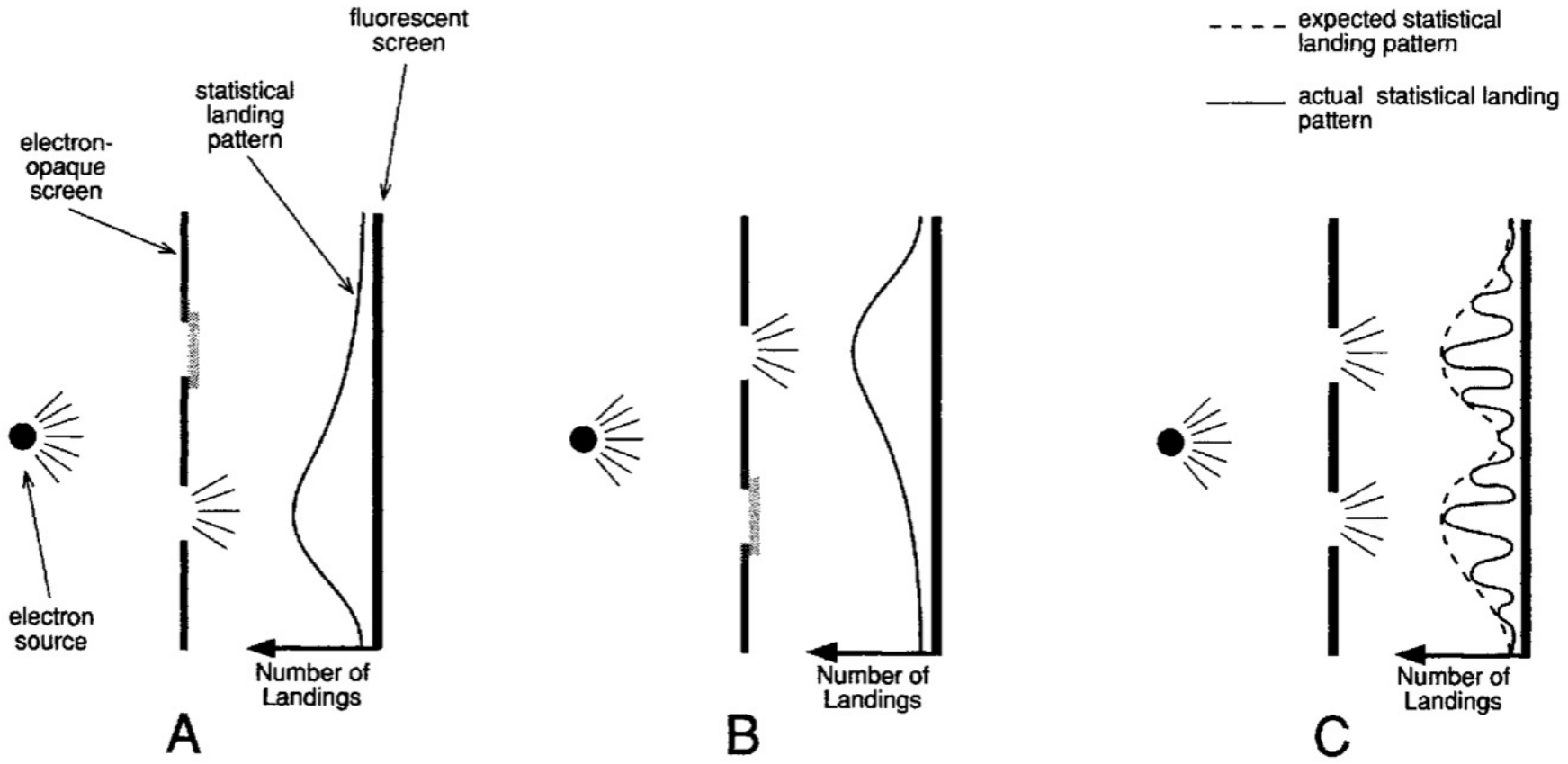
Superpositie



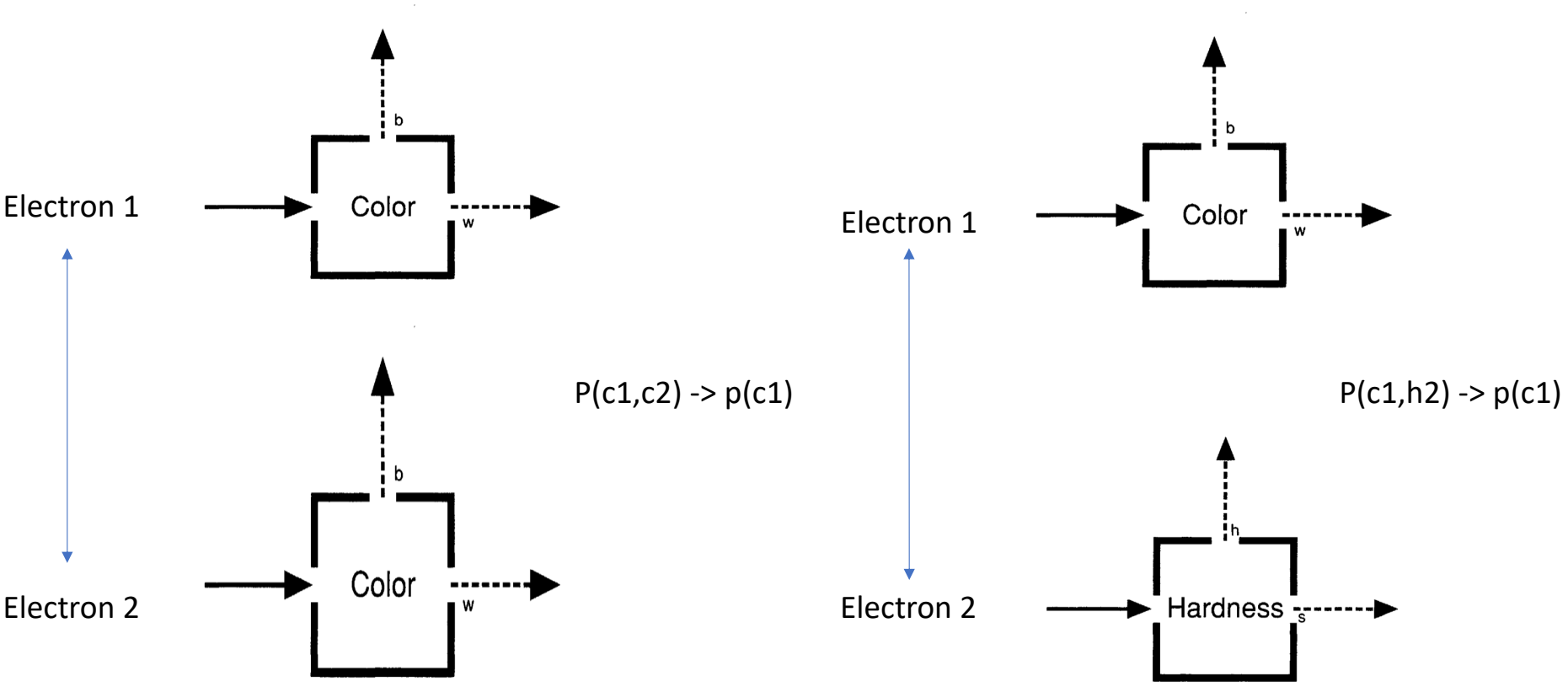
We meten de kleur van het uitgaand electron

Inkomend	Uitgaand
Hard	50 % wit, 50 % zwart
Zacht	50 % wit, 50 % zwart
Wit	100 % wit

Superpositie



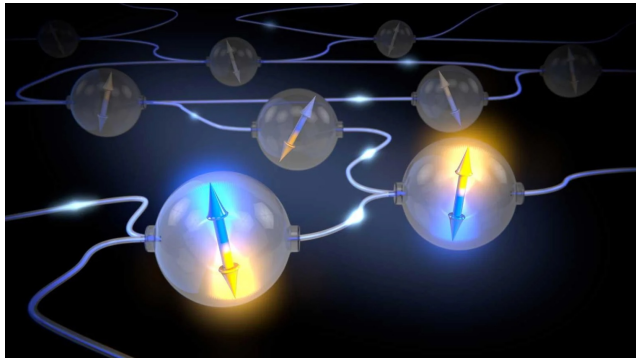
Niet-localiteit en entanglement



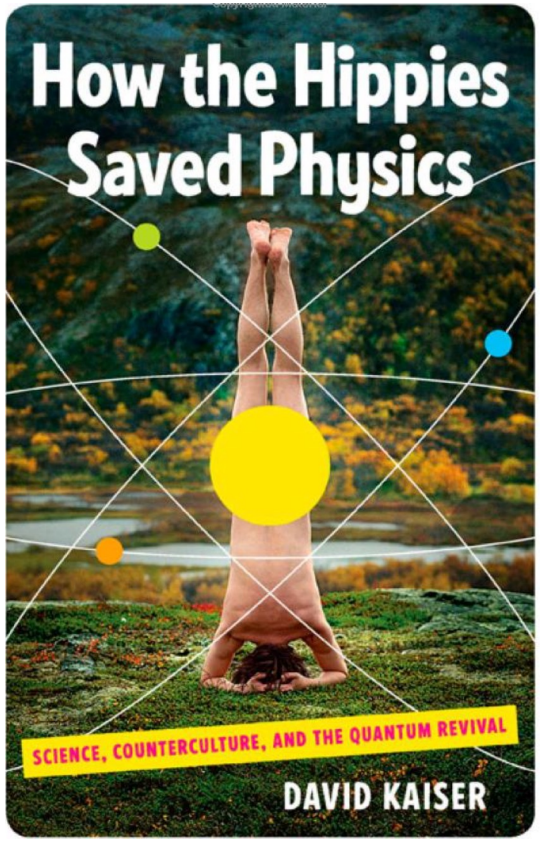
Electron is als cameleon. De kleur verandert met de omgeving

Quantum weirdness

- Onzekerheids principe: Eigenschappen van deeltjes zijn niet vast, maar veranderen door observatie
- Superpositie: deeltjes zijn op twee plekken tegelijkertijd
- Entanglement en niet-localiteit. De wereld is niet reductionistisch te beschrijven in termen van deeltjes met eigenschappen.



Entanglement



Alles is verbonden

Quantum brein

Brein is te warm voor quantum effecten

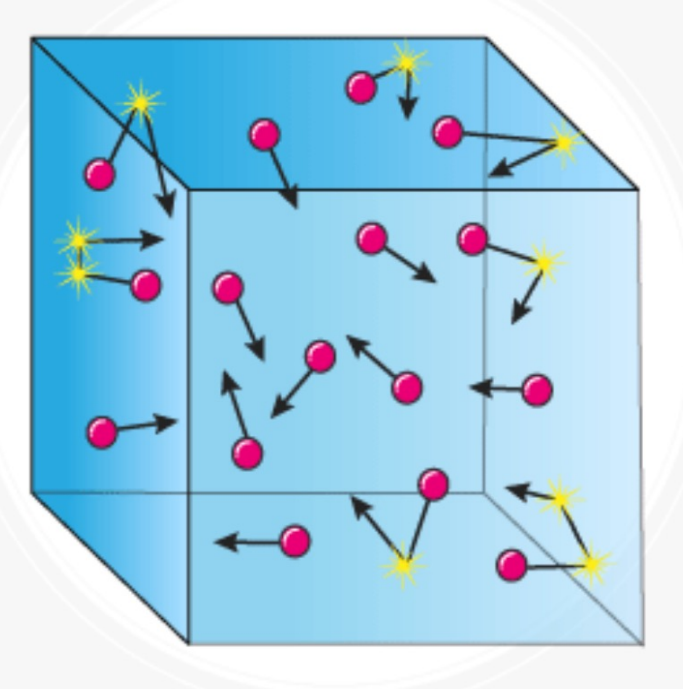
Thermische energie evenredig met temperatuur $E_{\text{th}} = k_B T$:

$$T = 300K \qquad \rightarrow \qquad E_{\text{th}} = 0.013eV$$

Quantum energie bepaalt frequentie $E_q = hf$

$$E_q > E_{\text{th}} \qquad \rightarrow \qquad f > 10^{13} Hz$$

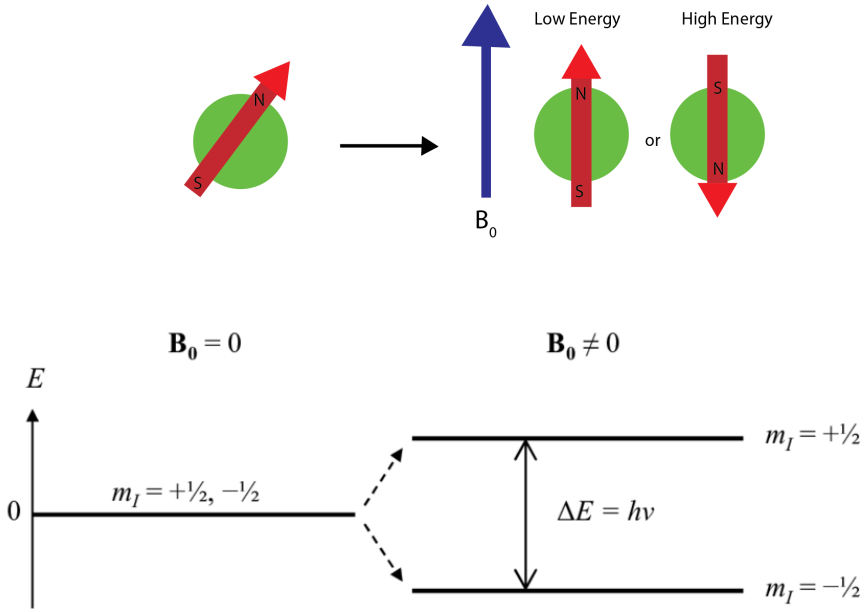
Dit is een tijdschaal van korter dan 10^{-13} sec,
veel korter dan neurale tijdschalen $10^{-3} - 10^{-6}$ sec.

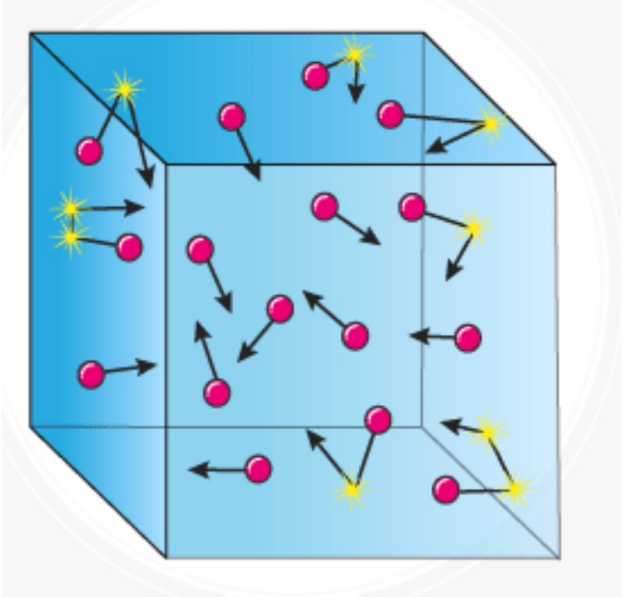


Tegmark 2000

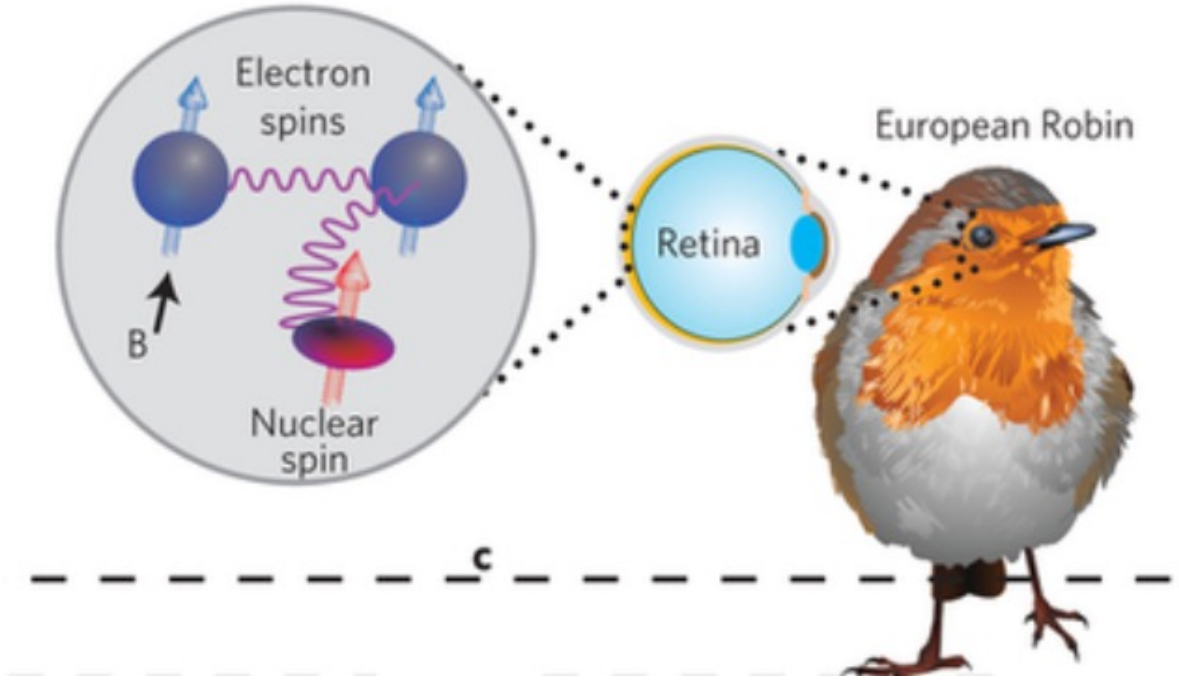
Het magnetisch kompas is gebaseerd op de interactie van het aardmagnetisch veld met de spin van electronen.

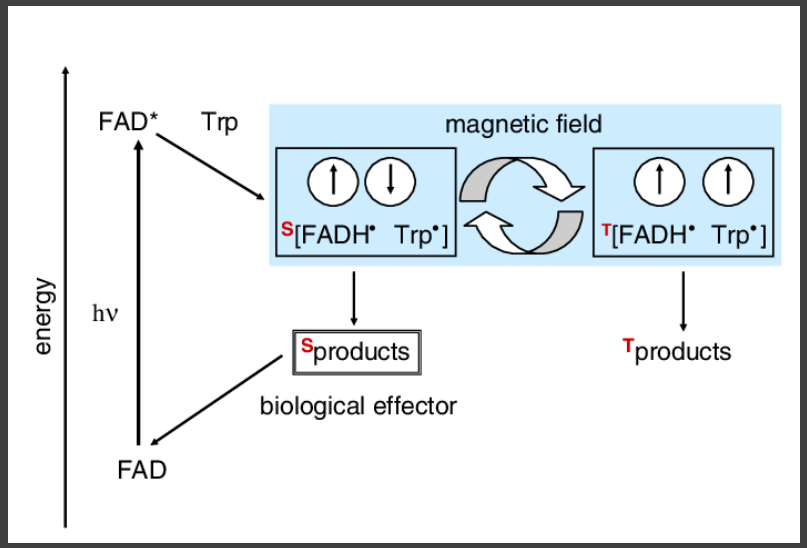
Maar: de energie van deze interactie is veel kleiner dan de thermische interacties: $E_B \approx 10^{-6} E_{\text{th}}$.

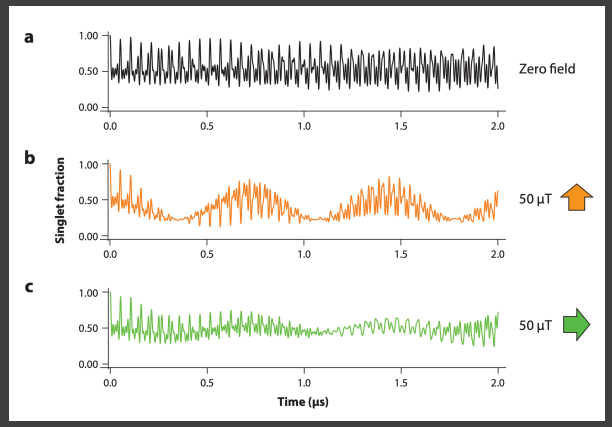




Magnetisch kompas

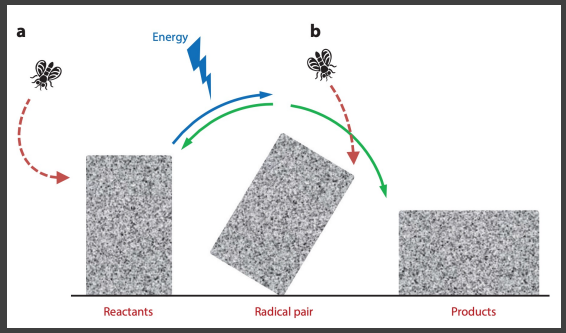




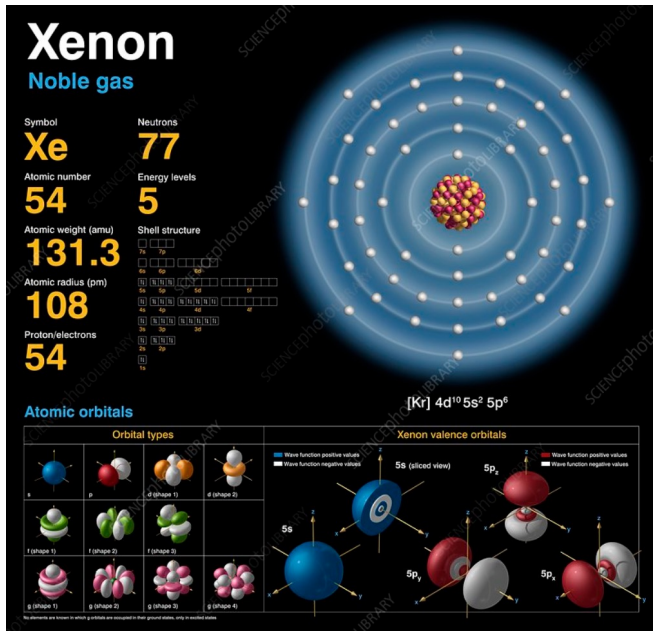


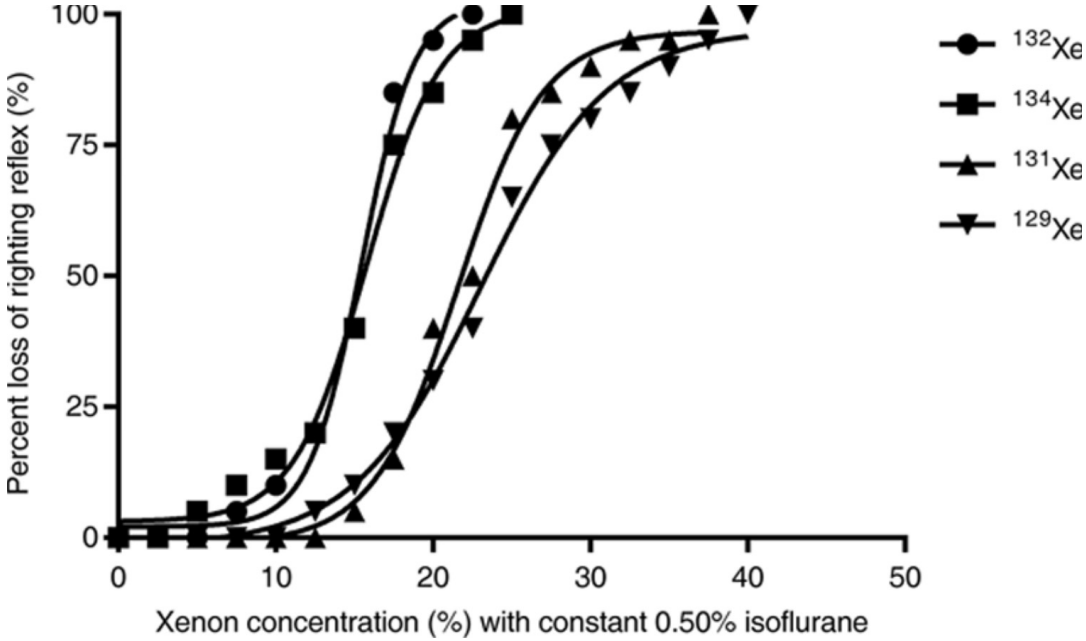
Magnetisch kompas

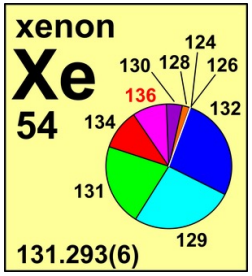
- Licht brengt het molecuul in instabiele toestand.
- De electron spin toestand verandert van singlet naar triplet door interactie met de kern spin en aardmagneetveld.
- Dit bepaalt het reactieproduct.



Kern spin beinvloedt bewustzijn in muizen







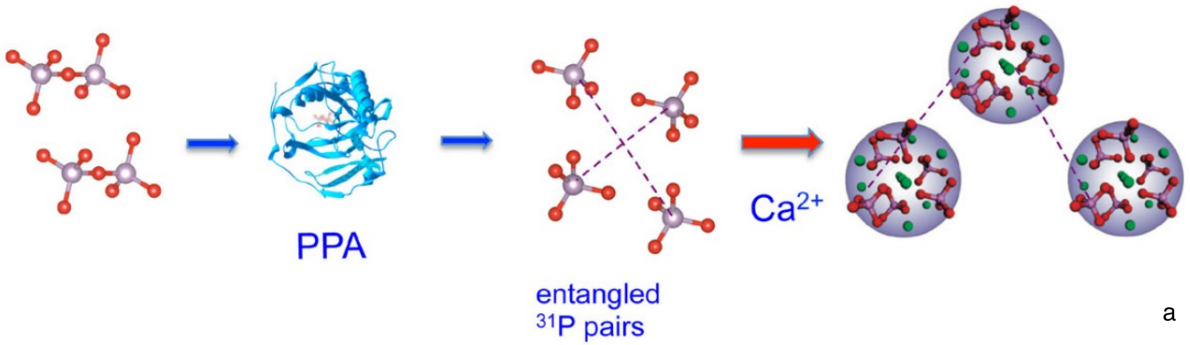
$^{129}Xe, ^{131}Xe, ^{134}Xe, ^{132}Xe$ heeft kern spin $1/2, 3/2, 0, 0.$

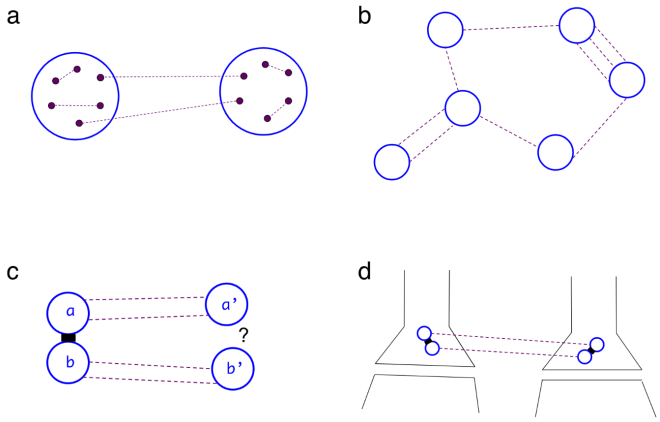
Li 2018

Entangled kern spins

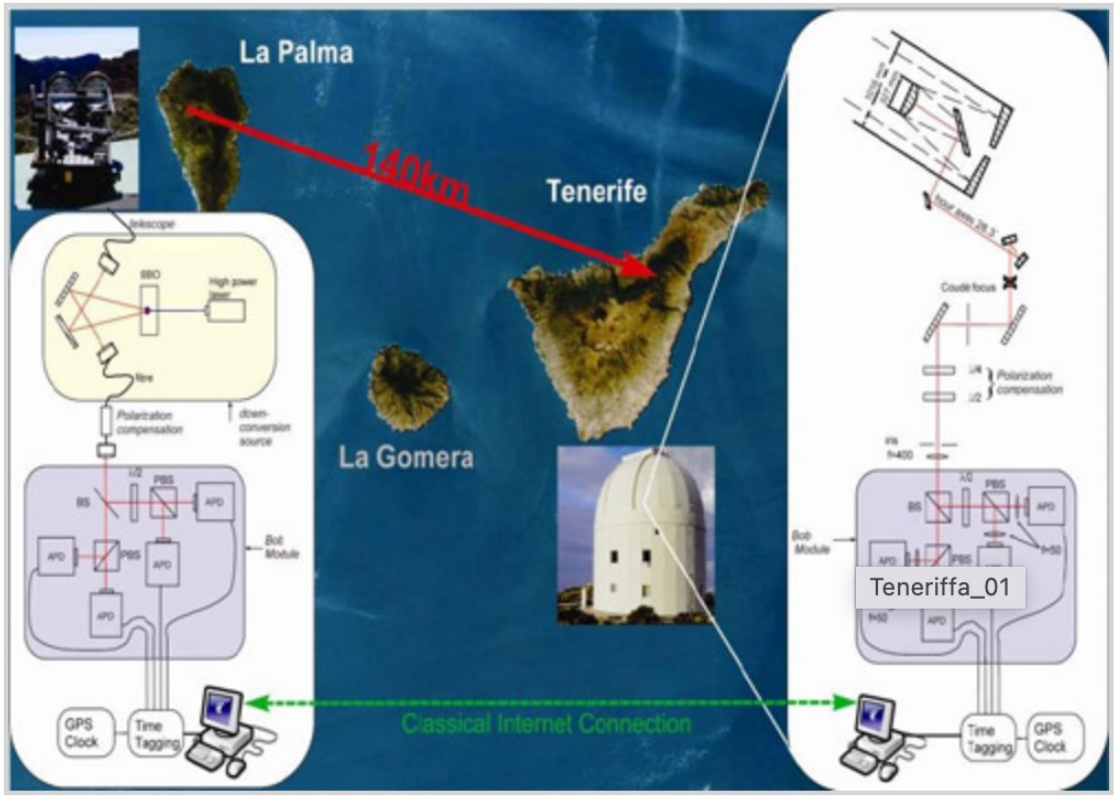
Coherentietijden van minuten

Entangled fosfor kern spins van fosfaat



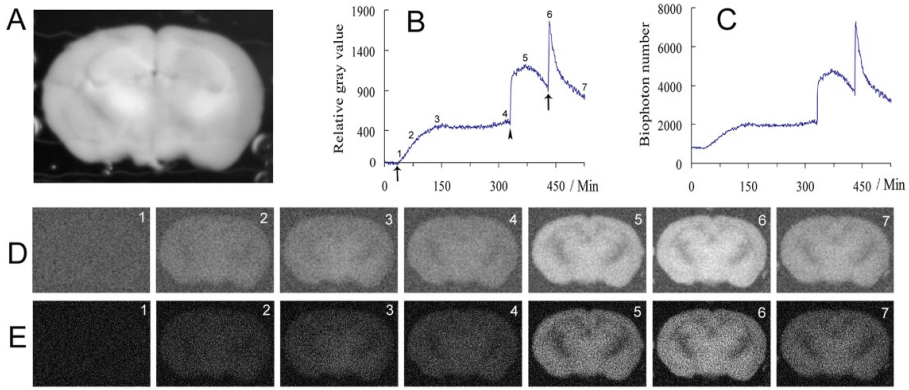


Fisher 2015

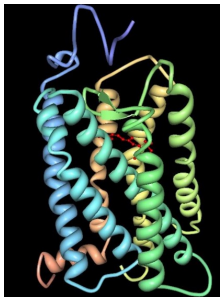


Zeilinger 2012

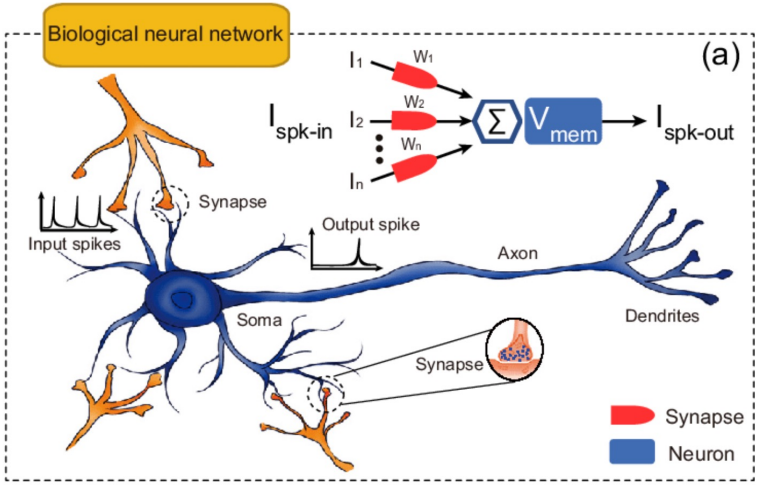
Foton network van entangled kern spins



Neuronen zenden ongeveer 1 foton per neuron per seconde en transmissie langs axonen.



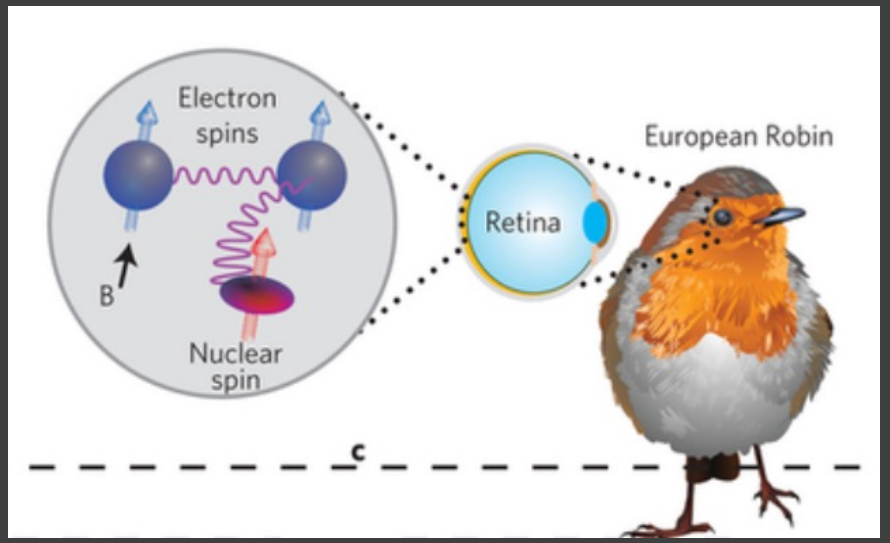
Opsins converteren fotonen in electro-chemische activiteit.



Simon 2017

Quantum brein

- Quantum effecten in het brein door interactie van electron en kern spin (‘radical pair mechanism’)
  - Magnetisch kompas van vogels
  - Effect van Xe isotope op narcose en bewustzijn
  - Effect van Li isotope op de werking voor bipolaire stoornis
  - ….
  - entanglement lijkt hierbij niet essentieel.
- Interessante voorstellen voor quantum bewustzijn met entangled kern spin netwerken, maar (zeer) omstreden



Een keuze

Optie 1: Een quantum mechanism voor bewustzijn?

Optie 2: Accepteer dat niet alles wiskundig te beschrijven is.

Wetenschap is goed en belangrijk, maar kan wellicht niet alles verklaren.

Er is de neiging dat we onze wetenschappelijke kijk op de werkelijkheid gelijkstellen met de werkelijkheid.

Dat we gaan geloven dat er niet meer bestaat dan wat onze modellen en theorien kunnen verklaren.

Zou dit voor bewustzijn gelden, wellicht?

Zen and the art of motorcycle maintenance. R. Pirsig 1974

Quality: iets wat bestaat, maar niet formeel kan worden gedefinieerd.

Hetzelfde geldt voor liefde, schoonheid, kunst en wellicht ook bewustzijn.

Dit zijn de dingen die het leven de moeite waard maken, en misschien kan de wetenschap hier nooit iets over zeggen.

Zen and the art of motorcycle maintenance. R. Pirsig 1974

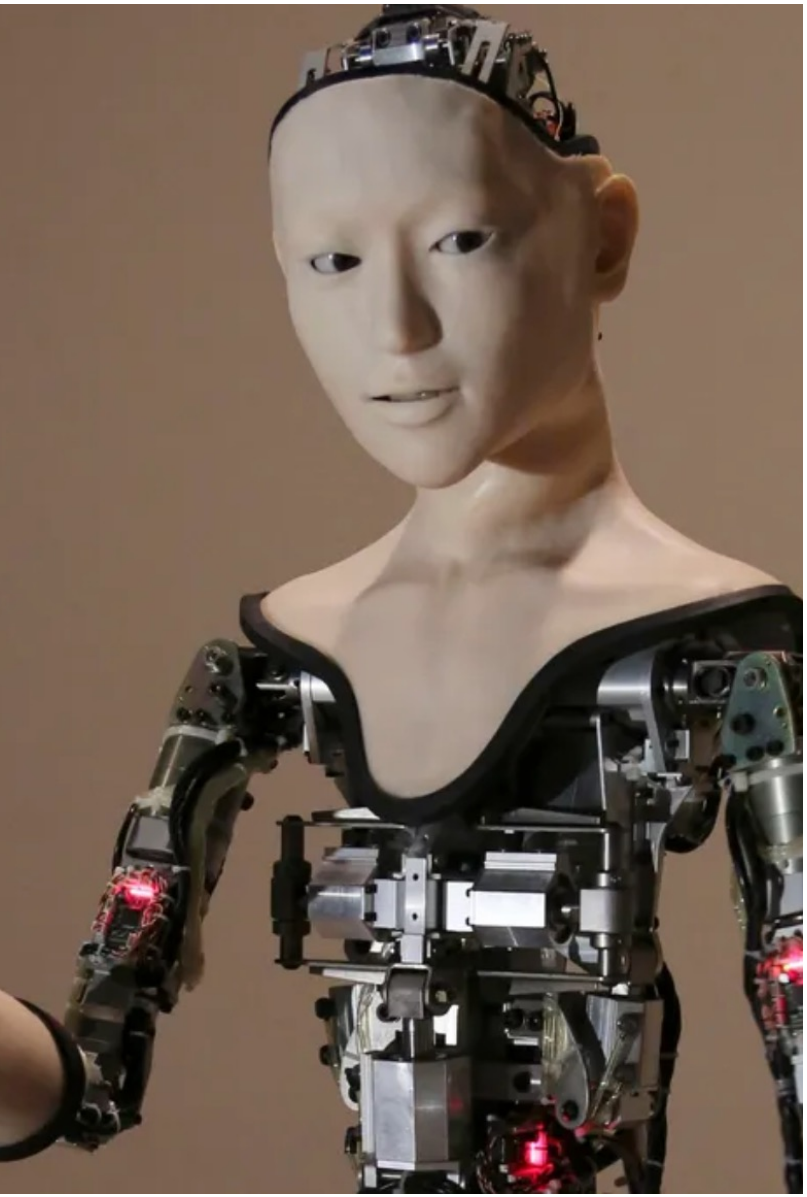
El sueno de la razon produce muonstros

F. Goya, Madrid 1799

Mijn keuze

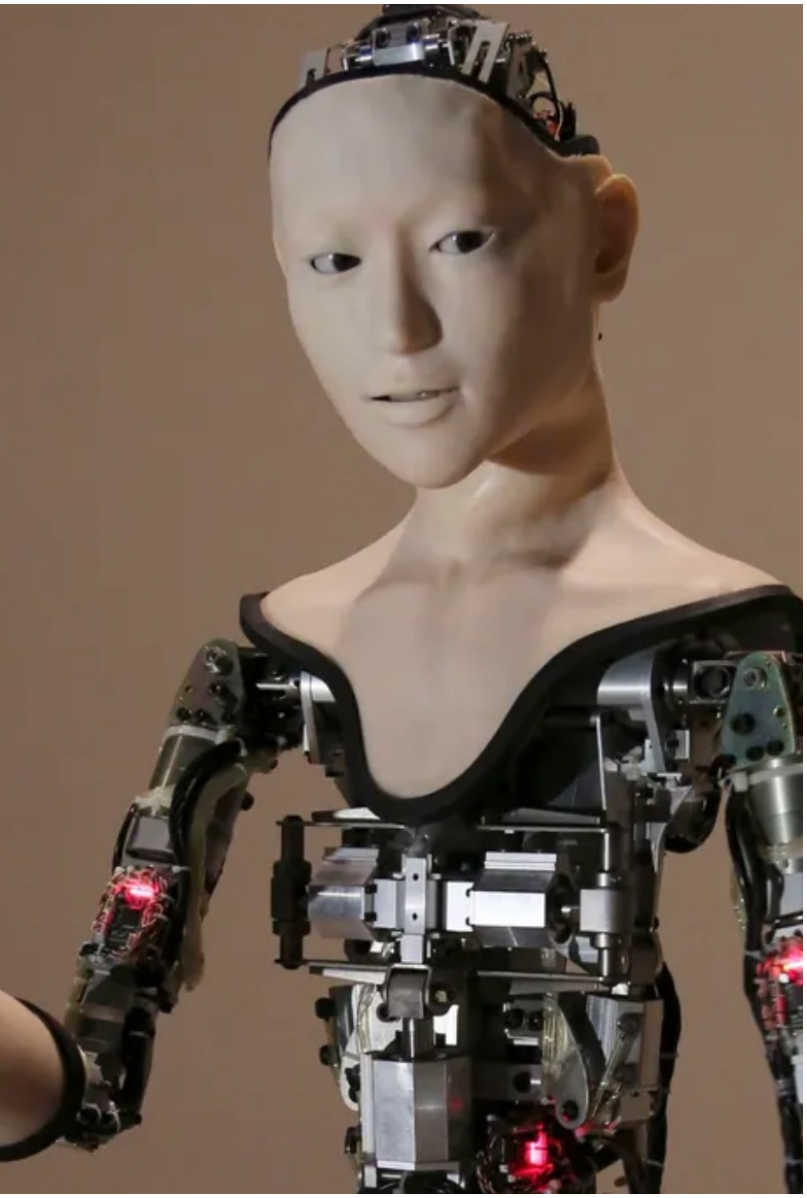
Optie 1: Een quantum mechanism voor bewustzijn?

Optie 2: Accepteer dat niet alles wiskundig te beschrijven is.



Samenvattend

- Computers worden steeds intelligenter, en mogelijk (veel) slimmer dan mensen
- Bewustzijn is niet meetbaar in termen van intelligent gedrag, maar is een subjectieve persoonlijke beleving.
- Huidige AI methodes, of nieuwe methods gebaseerd op klassieke fysica, kunnen deze subjectieve beleving niet in een machine namaken.
- Quantum entanglement is mogelijk alternatief
  - Quantum effecten spelen een rol in het brein (radical pairs, kern spin)
  - De rol van entanglement hierbij is onduidelijk
  - goede theorieen van quantum bewustzijn ontbreken
- Misschien is bewustzijn niet wetenschappelijk verklaarbaar



Discussie

- Maatschappelijke gevolgen
  - AI is een technologische ontwikkeling en komt voort uit een materialistisch denken: er bestaat alleen materie en verder niks. Hoe waarborgen we de menselijke maat?
  - Hoe behouden wij onze meerwaarde, en hoe willen wij met machines samenleven?
  - Hebben wij uberhaupt invloed op deze ontwikkeling?

References

[Aarts and Korst, 1988] Aarts, E. and Korst, J. (1988). *Simulated annealing and Boltzmann machines*. New York, NY; John Wiley and Sons Inc.

[Amit et al., 1985] Amit, D., Gutfreund, H., and Sompolinsky, H. (1985). Storing infinite numbers of patterns in a spin glass model of neural networks. *Physical Review Letters*, 55:1530–1533.

[Barber and Bishop, 1998] Barber, D. and Bishop, C. M. (1998). Ensemble learning for Multi-Layer Networks. In Jordan, M. I., Kearns, M., and Solla, S., editors, *Advances in Neural Information Processing Systems NIPS 10*, pages 395–401. MIT Press.

[Candes and Tao, 2005] Candes, E. J. and Tao, T. (2005). Decoding by linear programming. *IEEE transactions on information theory*, 51(12):4203–4215.

[Donoho, 2006] Donoho, D. L. (2006). Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306.

[Frey and Dueck, 2007] Frey, B. J. and Dueck, D. (2007). Clustering by passing messages between data points. *science*, 315(5814):972–976.

[Frey and MacKay, 1998] Frey, B. J. and MacKay, D. J. (1998). A revolution: Belief propagation in graphs with cycles. In *Advances in neural information processing systems*, pages 479–485.

[Gallager, 1963] Gallager, R. (1963). *Low-density parity check codes*. MIT Press.

[Garey and Johnson, 1979] Garey, M. and Johnson, D. (1979). *Computers and intractability: a guide to the theory of NP-completeness*. Freeman, San Francisco.

[Globerson and Jaakkola, 2007] Globerson, A. and Jaakkola, T. (2007). Fixing max-product: Convergent message passing algorithms for map lp-relaxations. *Advances in neural information processing systems*, 20:553–560.

[Hajek, 1988] Hajek, B. (1988). Cooling schedules for optimal annealing. *Mathematics of Operations Research*, 13:311–329.

[Havlíček et al., 2019] Havlíček, V., Córcoles, A. D., Temme, K., Harrow, A. W., Kandala, A., Chow, J. M., and Gambetta, J. M. (2019). Supervised learning with quantum-enhanced feature spaces. *Nature*, 567(7747):209–212.

[Hinton, 2002] Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800.

[Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

[Huijgen et al., 2023] Huijgen, O., Coopmans, L., Najafi, P., Benedetti, M., and Kappen, H. J. (2023). Training quantum boltzmann machines with the beta-variational quantum eigensolver. *arXiv preprint arXiv:2304.08631*.

[Kabashima and Saad, 1999] Kabashima, Y. and Saad, D. (1999). Statistical mechanics of error-correcting codes. *EPL (Europhysics Letters)*, 45(1):97.

[Kappen and Rodríguez, 1998] Kappen, H. and Rodríguez, F. (1998). Efficient learning in Boltzmann Machines using linear response theory. *Neural Computation*, 10:1137–1156.

[Kappen and Spanjers, 1999] Kappen, H. and Spanjers, J. (1999). Mean field theory for asymmetric neural networks. *Physical Review E*, 61:5658–5663.

[Kappen, 2020] Kappen, H. J. (2020). Learning quantum models from quantum or classical data. *Journal of Physics A: Mathematical and Theoretical*, 53(21):214001.

[Krzakala et al., 2012] Krzakala, F., Mézard, M., Sausset, F., Sun, Y., and Zdeborová, L. (2012). Statistical-physics-based reconstruction in compressed sensing. *Physical Review X*, 2(2):021005.

[Kschischang et al., 2001] Kschischang, F. R., Frey, B. J., and Loeliger, H.-A. (2001). Factor graphs and the sum-product algorithm. *IEEE Transactions on information theory*, 47(2):498–519.

[Mezard et al., 2002] Mezard, M., Parisi, G., and Zecchina, R. (2002). Analytic and algorithmic solution of random satisfiability problems. *Science*, 297.

[Mezard and Zecchina, 2002] Mezard, M. and Zecchina, R. (2002). The random k-satisfiability problem: from an analytic solution to an efficient algorithm. *Phys. Rev. E*, 66:056126.

[Mooij and Kappen, 2007] Mooij, J. and Kappen, H. (2007). Sufficient conditions for convergence of the sum-product algorithm. *IEEE Information Theory*, 53:4422–4437.

[Onsager, 1936] Onsager, L. (1936). Electric moments of molecules in liquids. *Journal of the American Chemical Society*, 58:1486–1493.

[Pearl, 1988] Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of Plausible Inference*. Morgan Kaufmann, San Francisco, California.

[Plefka, 1982] Plefka, T. (1982). Convergence condition of the TAP equation for the infinite-range Ising spin glass model. *Journal of Physics A*, 15:1971–1978.

[Salakhutdinov et al., 2007] Salakhutdinov, R., Mnih, A., and Hinton, G. (2007). Restricted boltzmann machines for collaborative filtering. In *Proceedings of the 24th international conference on Machine learning*, pages 791–798.

[Sherrington and Kirkpatrick, 1975] Sherrington, D. and Kirkpatrick, S. (1975). Solvable model of Spin-Glass. *Physical review letters*, 35:1792–1796.

[Sutskever et al., 2014] Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27.

[Tatikonda and Jordan, 2012] Tatikonda, S. and Jordan, M. I. (2012). Loopy belief propogation and gibbs measures. *arXiv preprint arXiv:1301.0605*.

[Thouless et al., 1977] Thouless, D., Anderson, P., and Palmer, R. (1977). Solution of 'Solvable Model of a Spin Glass'. *Phil. Mag.*, 35:593–601.

[Tu et al., 2024] Tu, T., Palepu, A., Schaekermann, M., Saab, K., Freyberg, J., Tanno, R., Wang, A., Li, B., Amin, M., Tomasev, N., et al. (2024). Towards conversational diagnostic AI. *arXiv preprint arXiv:2401.05654*.

[Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

[Wei et al., 2022] Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., et al. (2022). Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

[Yedidia et al., 2001] Yedidia, J., Freeman, W., and Weiss, Y. (2001). Generalized belief propagation. In Leen, T., Dietterich, T., and Tresp, V., editors, *Advances in Neural Information Processing Systems 13 (Proceedings of the 2000 Conference)*, pages 689–695. MIT Press.