Learning Energy-Based Models of High-Dimensional Data

Geoffrey Hinton
Max Welling
Yee-Whye Teh
Simon Osindero

www.cs.toronto.edu/~hinton/EnergyBasedModelsweb.htm

Discovering causal structure as a goal for unsupervised learning

- It is better to associate responses with the hidden causes than with the raw data.
- The hidden causes are useful for understanding the data.
- It would be interesting if real neurons really did represent independent hidden causes.

A different kind of hidden structure

Instead of trying to find a set of independent hidden causes, try to find factors of a different kind.

Capture structure by finding constraints that are Frequently Approximately Satisfied.

Violations of FAS constraints reduce the probability of a data vector. If a constraint already has a big violation, violating it more does not make the data vector much worse (i.e. assume the distribution of violations is heavy-tailed.)

Two types of density model

Stochastic generative model using directed acyclic graph (e.g. Bayes Net)

$$p(d) = p(b)\,p(c\,|\,b)\,p(d\,|\,b,c)$$

Synthesis is easy

Analysis can be hard

Learning is easy after analysis

Energy-based models that associate an energy with each data vector

$$p(d) \;=\; \frac{e^{-E(d)}}{\sum_{c} e^{-E(c)}}$$

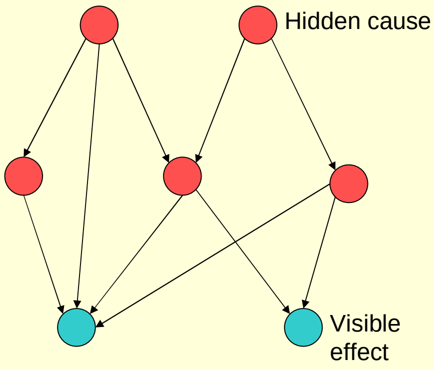
Synthesis is hard

Analysis is easy

Is learning hard?

Bayes Nets

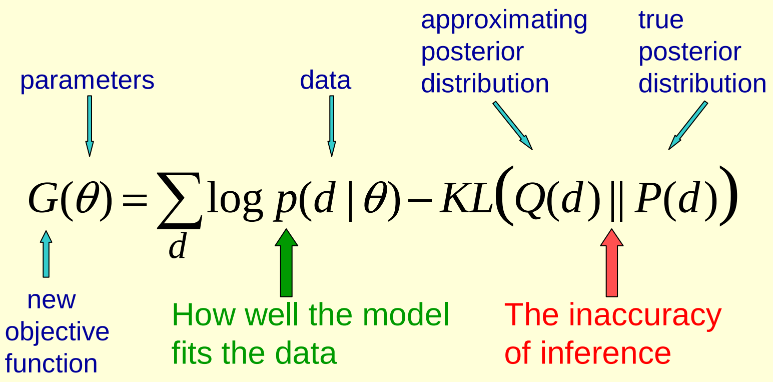
- It is easy to generate an unbiased example at the leaf nodes.
- It is typically hard to compute the posterior distribution over all possible configurations of hidden causes.
- Given samples from the posterior, it is easy to learn the local interactions



Approximate inference

- What if we use an approximation to the posterior distribution over hidden configurations?
  - e.g. assume the posterior factorizes into a product of distributions for each separate hidden cause.
- If we use the approximation for learning, there is no guarantee that learning will increase the probability that the model would generate the observed data.
- But maybe we can find a different and sensible objective function that is guaranteed to improve at each update.

A trade-off between how well the model fits the data and the tractability of inference



parameters

data

approximating posterior distribution

true posterior distribution

$$G(\theta)=\sum_{d}\log p(d\,|\,\theta)-KL\Big(Q(d)\,\|\,P(d)\Big)$$

new objective function

How well the model fits the data

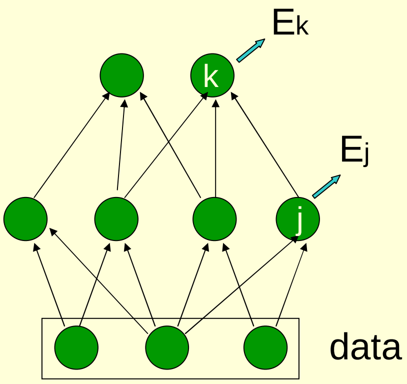
The inaccuracy of inference

This makes it feasible to fit very complicated models, but the approximations that are tractable may be very poor.

Energy-Based Models with deterministic hidden units

- Use multiple layers of deterministic hidden units with non-linear activation functions.
- Hidden activities contribute additively to the global energy, E.

$$p(d)=\frac{e^{-E(d)}}{\sum_{c} e^{-E(c)}}$$



Maximum likelihood learning is hard

- To get high log probability for d we need low energy for d and high energy for its main rivals, c

$$\log p(d) = -E(d) - \log \sum_c e^{-E(c)}$$

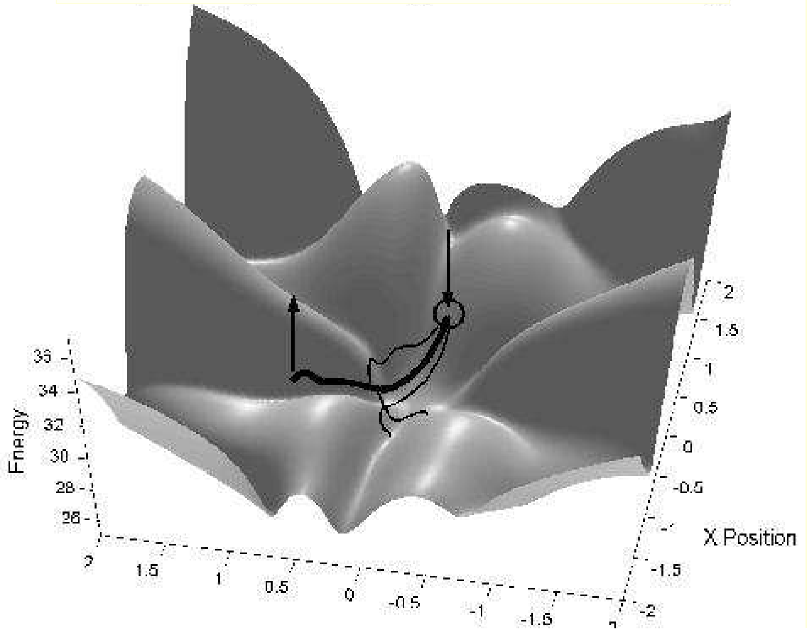
$$\frac{\partial \log p(d)}{\partial \theta} = -\frac{\partial E(d)}{\partial \theta} + \sum_c \underset{\uparrow}{p(c)} \frac{\partial E(c)}{\partial \theta}$$

To sample from the model use Markov Chain Monte Carlo

Hybrid Monte Carlo

- The obvious Markov chain makes a random perturbation to the data and accepts it with a probability that depends on the energy change.
  - Diffuses very slowly over flat regions
  - Cannot cross energy barriers easily
- In high-dimensional spaces, it is much better to use the gradient to choose good directions and to use momentum.
  - Beats diffusion. Scales well.
  - Can cross energy barriers.

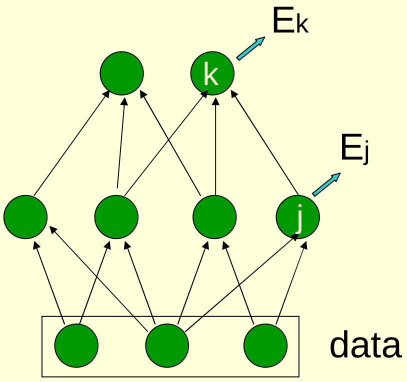
Trajectories with different initial momenta



Backpropagation can compute the gradient that Hybrid Monte Carlo needs

1. Do a forward pass computing hidden activities.
2. Do a backward pass all the way to the data to compute the derivative of the global energy w.r.t each component of the data vector.

   works with any smooth non-linearity



The online HMC learning procedure

1. Start at a datavector, d, and use backprop to compute $\underline{\partial E(d)/\partial \theta}$ for every parameter.

2. Run HMC for many steps with frequent renewal of the momentum to get equilbrium sample, c.

3. Use backprop to compute $\underline{\partial E(c)/\partial \theta}$

4. Update the parameters by :
$$\Delta\theta = \varepsilon\left(-\underline{\partial E(d)/\partial\theta} + \underline{\partial E(c)/\partial\theta}\right)$$

A surprising shortcut

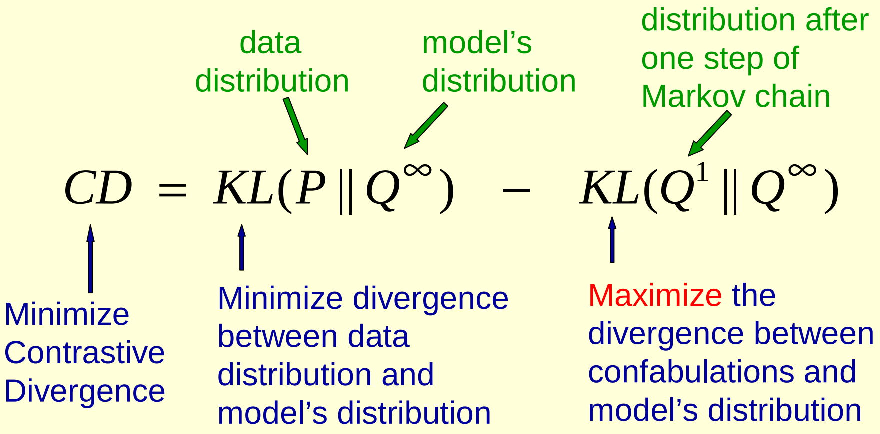
- Instead of taking the negative samples from the equilibrium distribution, use slight corruptions of the datavectors. Only add random momentum once, and only follow the dynamics for a few steps.
  - Much less variance because a datavector and its confabulation form a matched pair.
  - Seems to be very biased, but maybe it is optimizing a different objective function.
- If the model is perfect and there is an infinite amount of data, the confabulations will be equilibrium samples. So the shortcut will not cause learning to mess up a perfect model.

Intuitive motivation

- It is silly to run the Markov chain all the way to equilibrium if we can get the information required for learning in just a few steps.
  - The way in which the model systematically distorts the data distribution in the first few steps tells us a lot about how the model is wrong.
  - But the model could have strong modes far from any data. These modes will not be sampled by confabulations. Is this a problem in practice?

Contrastive divergence

Aim is to minimize the amount by which a step toward equilibrium improves the data distribution.



$$CD = KL(P \,\|\, Q^{\infty}) - KL(Q^{1} \,\|\, Q^{\infty})$$

data distribution

model's distribution

distribution after one step of Markov chain

Minimize Contrastive Divergence

Minimize divergence between data distribution and model's distribution

Maximize the divergence between confabulations and model's distribution

Contrastive divergence

$$-\frac{\partial KL(Q^0 \| Q^\infty)}{\partial \theta} = -\left\langle \frac{\partial E}{\partial \theta} \right\rangle_{Q^0} + \left\langle \frac{\partial E}{\partial \theta} \right\rangle_{Q^\infty}$$

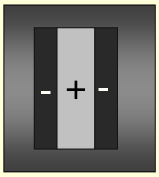
$$-\frac{\partial KL(Q^1 \| Q^\infty)}{\partial \theta} = -\left\langle \frac{\partial E}{\partial \theta} \right\rangle_{Q^1} + \left\langle \frac{\partial E}{\partial \theta} \right\rangle_{Q^\infty} - \frac{\partial Q^1}{\partial \theta} \frac{\partial KL(Q^1 \| Q^\infty)}{\partial Q^1}$$

Contrastive divergence makes the awkward terms cancel

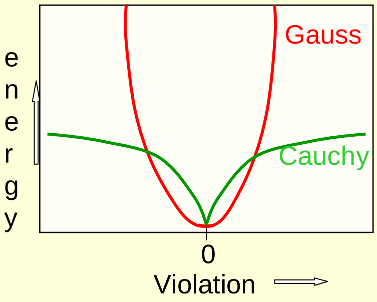
changing the parameters changes the distribution of confabulations

Frequently Approximately Satisfied constraints

- The intensities in a typical image satisfy many different linear constraints very accurately, and violate a few constraints by a lot.
- The constraint violations fit a heavy-tailed distribution.
- The negative log probabilities of constraint violations can be used as energies.



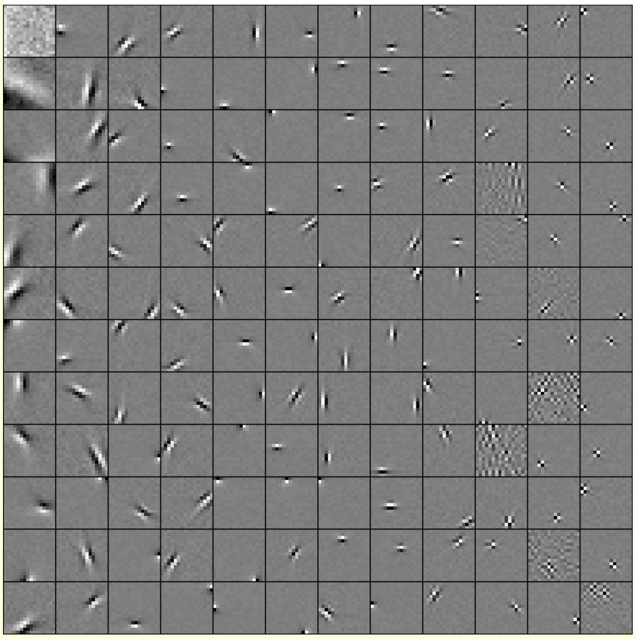
On a smooth intensity patch the sides balance the middle



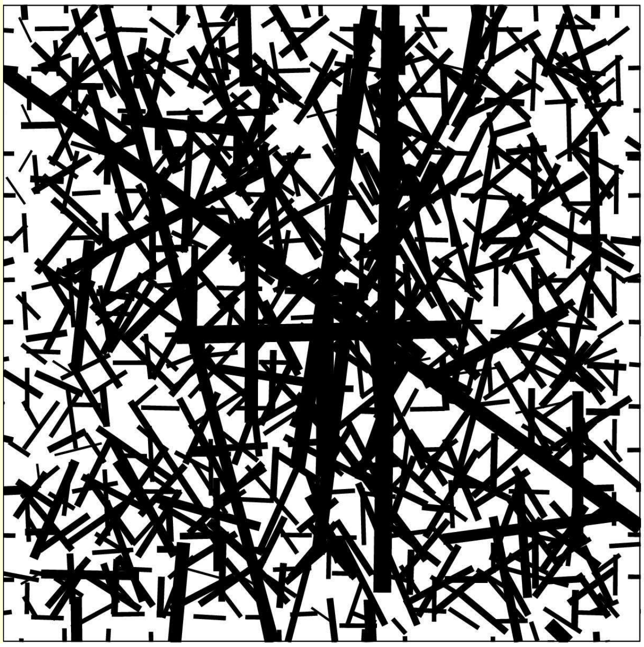
Learning constraints from natural images (Yee-Whye Teh)

- We used 16x16 image patches and a single layer of 768 hidden units (3 x overcomplete).
- Confabulations are produced from data by adding random momentum once and simulating dynamics for 30 steps.
- Weights are updated every 100 examples.
- A small amount of weight decay helps.

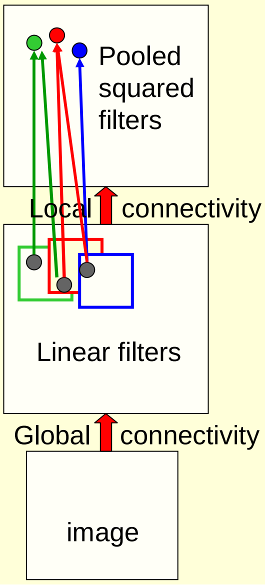
A random subset of 768 basis functions



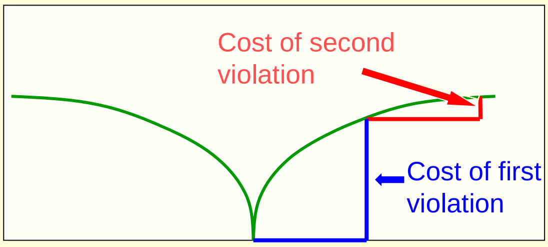
The distribution of all 768 learned basis functions

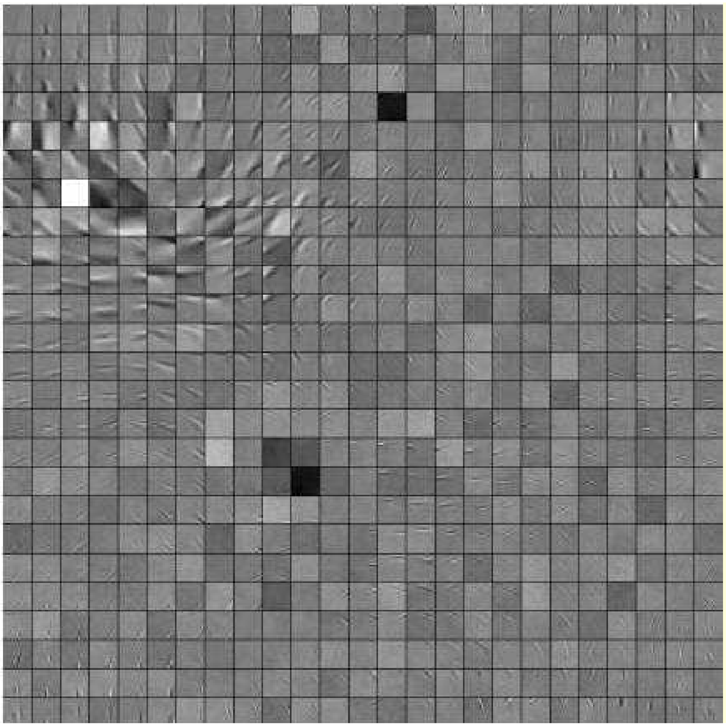


How to learn a topographic map



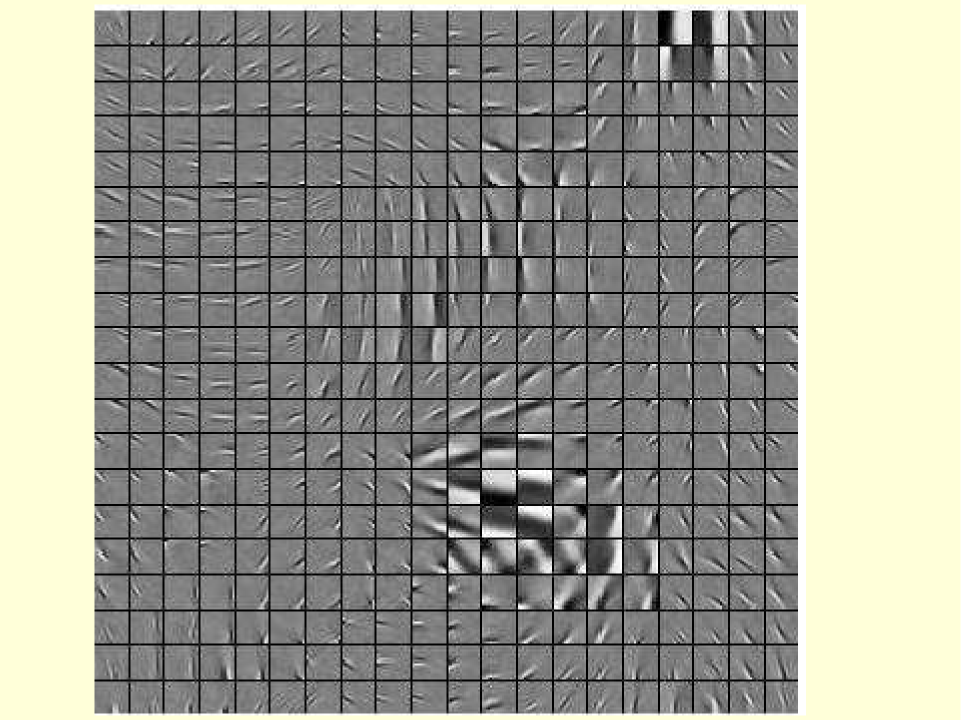
The outputs of the linear filters are squared and locally pooled. This makes it cheaper to put filters that are violated at the same time next to each other.

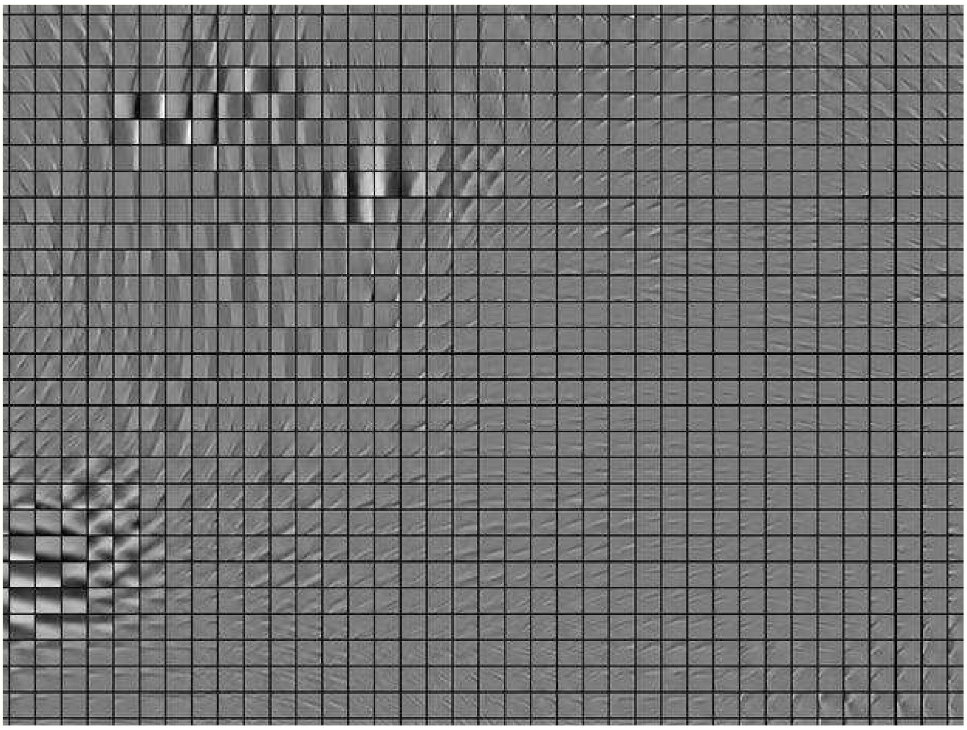


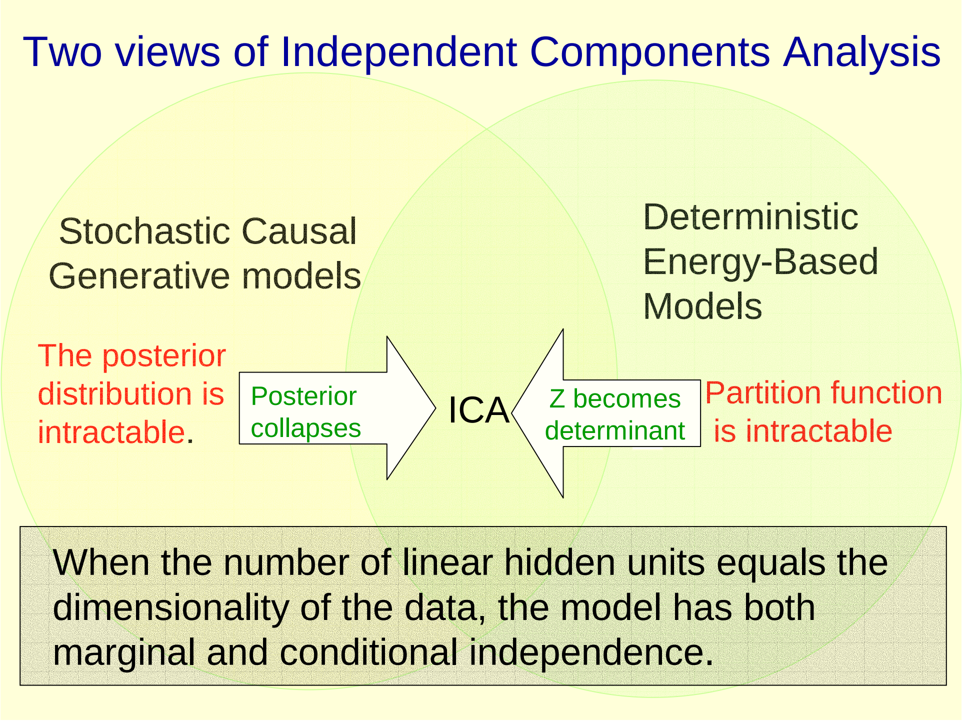


Faster mixing chains

- Hybrid Monte Carlo can only take small steps because the energy surface is curved.
- With a single layer of hidden units, it is possible to use alternating parallel Gibbs sampling.
  - Much less computation
  - Much faster mixing
  - Can be extended to use pooled second layer (Max Welling)
  - Can only be used in deep networks by learning one hidden layer at a time.







Two views of Independent Components Analysis

Stochastic Causal Generative models

Deterministic Energy-Based Models

The posterior distribution is intractable.

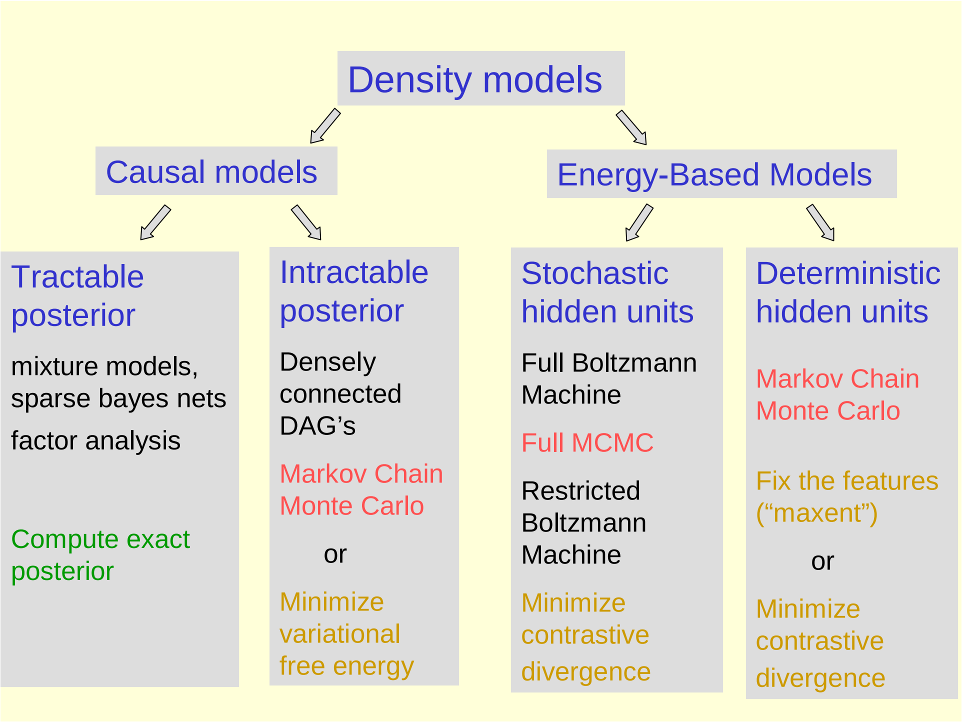
Posterior collapses

ICA

Z becomes determinant

Partition function is intractable

When the number of linear hidden units equals the dimensionality of the data, the model has both marginal and conditional independence.



Density models

Causal models

Energy-Based Models

Tractable posterior

mixture models, sparse bayes nets

factor analysis

Compute exact posterior

Intractable posterior

Densely connected DAG's

Markov Chain Monte Carlo

or

Minimize variational free energy

Stochastic hidden units

Full Boltzmann Machine

Full MCMC

Restricted Boltzmann Machine

Minimize contrastive divergence

Deterministic hidden units

Markov Chain Monte Carlo

Fix the features ("maxent")

or

Minimize contrastive divergence

Where to find out more

- www.cs.toronto.edu/~hinton has papers on:
  - Energy-Based ICA
  - Products of Experts

- This talk is at www.cs.toronto.edu/~hinton