Bayesian Learning of Directed and Undirected Graphical Models

Zoubin Ghahramani†

Gatsby Computational Neuroscience Unit
University College London
http://www.gatsby.ucl.ac.uk/

† Collaborators: Matthew J Beal and Hyun-Chul Kim,

Work partly done while at the:
Center for Automated Learning and Discovery
Carnegie Mellon University

Jan 2003, AI and Statistics Workshop, Key West, FL
Directed Graphical Models

(aka Bayesian networks, belief networks, probabilistic directed acyclic graphs)

A directed acyclic graph (DAG) where each node is a random variable and the arcs represent statistical dependencies between the variables.

The DAG represents a factorization of the joint probability.

\[ p(A, B, C, D, E) = p(A) \cdot p(B) \cdot p(C|A, B) \cdot p(D|B, C) \cdot p(E|C, D) \]

Each variable is conditionally independent of its non-descendents given its parents. (More generally \( X \perp Y | Z \), if \( Z \) d-separates \( X \) from \( Y \))
Learning Model Structure

- **Conditional Independence Structure**
  What is the structure of the graph (i.e. what \( \perp \perp \) relations hold)?

- **Feature Selection**
  Is some input relevant to predicting some output?

- **Cardinality of Discrete Latent Variables**
  How many clusters in the data?
  How many states in a hidden Markov model?

- **Dimensionality of Real Valued Latent Vectors**
  What choice of dimensionality in a PCA/FA model of the data?
  How many state variables in a linear-Gaussian state-space model?
Using Bayesian Occam’s Razor to Learn Model Structure

Select the model class, $m_i$, with the highest probability given the data, $y$:

$$p(m_i|y) = \frac{p(y|m_i)p(m_i)}{p(y)}$$

$$p(y|m_i) = \int p(y|\theta_i, m_i) p(\theta_i|m_i) \, d\theta_i$$

**Interpretation of the Marginal Likelihood:** The probability that randomly selected parameters from the prior for the model class would generate $y$.

Model classes that are **too simple** are unlikely to generate the data set.

Model classes that are **too complex** can generate many possible data sets, so again, they are unlikely to generate that particular data set at random.
Computing Marginal Likelihoods can be Computationally Intractable

\[ p(y|m_i) = \int p(y|\theta_i, m_i) p(\theta_i|m_i) \, d\theta_i \]

- This can be a very high dimensional integral.

- The presence of latent variables results in additional dimensions that need to be marginalized out.

\[ p(y|m_i) = \int \int p(y,x|\theta_i, m_i) \, p(\theta_i|m_i) \, dx \, d\theta_i \]

- The likelihood term can be complicated.
Practical Bayesian approaches

- **Laplace approximations:**
  - Appeals to Central Limit Theorem making a Gaussian approximation about the posterior mode of the parameters.

- **Large sample approximations** (e.g. BIC).
  \[
  \log p(y|m_i) \approx \log p(y|\hat{\theta}, m_i) - \frac{d}{2} \log N
  \]

- **Markov chain Monte Carlo methods** (MCMC):
  - converge to the desired distribution in the limit, but:
  - many samples are required to ensure accuracy.
  - sometimes hard to assess convergence and reliably compute marginal likelihood.

- **Variational approximations**...
Lower Bounding the Marginal Likelihood

Variational Bayesian Learning

Let the latent variables be \( x \), data \( y \) and the parameters \( \theta \).
We can **lower bound** the marginal likelihood (Jensen’s inequality):

\[
\ln p(y|m) = \ln \int p(y, x, \theta|m) \, dx \, d\theta \\
= \ln \int q(x, \theta) \frac{p(y, x, \theta|m)}{q(x, \theta)} \, dx \, d\theta \\
\geq \int q(x, \theta) \ln \frac{p(y, x, \theta|m)}{q(x, \theta)} \, dx \, d\theta.
\]

Use a simpler, factorised approximation to \( q(x, \theta) \approx q_x(x)q_\theta(\theta) \):

\[
\ln p(y|m) \geq \int q_x(x)q_\theta(\theta) \ln \frac{p(y, x, \theta|m)}{q_x(x)q_\theta(\theta)} \, dx \, d\theta \\
= \mathcal{F}_m(q_x(x), q_\theta(\theta), y).
\]

Variational Bayesian Learning . . .

Maximizing this lower bound, $F_m$, leads to EM-like iterative updates:

$$q_{x}^{(t+1)}(x) \propto \exp \left[ \int \ln p(x,y|\theta, m) q_{\theta}^{(t)}(\theta) d\theta \right]$$  \hspace{1cm} \text{E-like step}$$

$$q_{\theta}^{(t+1)}(\theta) \propto p(\theta|m) \exp \left[ \int \ln p(x,y|\theta, m) q_{x}^{(t+1)}(x) dx \right]$$  \hspace{1cm} \text{M-like step}$$

Maximizing $F_m$ is equivalent to minimizing KL-divergence between the approximate posterior, $q_{\theta}(\theta) q_{x}(x)$ and the true posterior, $p(\theta, x|y, m)$:

$$\ln p(y|m) - F_m(q_{x}(x), q_{\theta}(\theta), y) = \int q_{x}(x) q_{\theta}(\theta) \ln \frac{q_{x}(x) q_{\theta}(\theta)}{p(\theta, x|y, m)} dx d\theta = \text{KL}(q||p)$$

In the limit as $n \to \infty$, for identifiable models, the variational lower bound approaches Schwartz’s (1978) BIC criterion.

(Attias, 2000; Ghahramani and Beal, 2000)
Conjugate Exponential Models

• If the joint probability of the hidden and observed data is in the exponential family

• ...and the prior over parameters is conjugate,

• ... then the variational Bayesian procedure becomes a simple modification of EM.

• This actually includes many interesting models.
The Variational Bayesian EM algorithm

**EM for MAP estimation**

**Goal:** maximize \( p(\theta | y, m) \) w.r.t. \( \theta \)

**E Step:** compute

\[
q^{(t+1)}_x(x) = p(x | y, \theta^{(t)})
\]

**M Step:**

\[
\theta^{(t+1)} = \arg\max_{\theta} \int q^{(t+1)}_x(x) \ln p(x, y, \theta) \, dx
\]

**Properties:**

- Reduces to the EM algorithm if \( q^{(t+1)}_\theta(\theta) = \delta(\theta - \theta^*) \).
- \( F_m \) increases monotonically, and incorporates the model complexity penalty.
- Analytical parameter distributions (but not constrained to be Gaussian).
- VB-E step has same complexity as corresponding E step.
- We can use the junction tree, belief propagation, Kalman filter, etc, algorithms in the VB-E step of VB-EM, but using expected natural parameters, \( \bar{\phi} \).

**Variational Bayesian EM**

**Goal:** lower bound \( p(y | m) \)

**VB-E Step:** compute

\[
q^{(t+1)}_x(x) = p(x | y, \bar{\phi}^{(t)})
\]

**VB-M Step:**

\[
q^{(t+1)}_\theta(\theta) \propto \exp \left[ \int q^{(t+1)}_x(x) \ln p(x, y, \theta) \, dx \right]
\]
Variational Bayes: Examples of Structure Learning

- finite **mixtures of Gaussians** (Attias, 1999)
- finite **mixtures of factor analysers** (for simultaneous clustering and dimensionality reduction) (Ghahramani & Beal, 1999)
- **principal components analysis (PCA)** (Bishop, 1999)
- **independent components analysis (ICA)** (Attias, 1999; Miskin & MacKay, 2000; Valpola 2000)
- linear-Gaussian **state-space models** (Ghahramani & Beal, 2000)
- **mixtures of experts** (piecewise linear regression models) (Ueda & Ghahramani, 2000)
- **hidden Markov models** (MacKay, 1995; Beal, in prep)
- **VIBES** (Bishop, Wynn, Spiegelhalter, 2002) . . .
Model Selection Task

Which of the following graphical models is the data generating process? Discrete directed acyclic graphical models: data \( y = (A, B, C, D, E)^n \)

If the data are just \( y = (C, D, E)^n \), and \((A, B)\) are hidden variables...?
A thorough Case Study

- Bipartite structure: only hidden variables can be parents of observed variables.
- Two binary hidden variables, and four five-valued discrete observed variables.

\[ \text{i}=1...n \]

- Conjugate prior is Dirichlet, Conjugate-Exponential model, so VB-EM algorithm is a straightforward modification of EM.

- **Experiment:** There are 136 distinct structures (out of 256) with 2 latent variables as potential parents of 4 conditionally independent observed vars.

- **Score** each structure for twenty varying size data sets:
  \[ n \in \{10, 20, 40, 80, 110, 160, 230, 320, 400, 430, 480, 560, 640, 800, 960, 1120, 1280, 2560, 5120, 10240\} \]

  using 3 methods: BIC, VB, and the gold standard AIS

- 2720 graph scores computed, times for each: BIC (1.5s), VB (4s), AIS (400s).
Annealed Importance Sampling (AIS)

AIS is a state-of-the-art method for estimating marginal likelihoods, by breaking a difficult integral into a series of easier ones.

Combines ideas from importance sampling, Markov chain Monte Carlo, & annealing.

Define

\[ Z_k = \int \, d\theta \, p(\theta \mid m) p(y \mid \theta, m)^{\tau(k)} = \int \, d\theta \, f_k(\theta) \]

with \( \tau(0) = 0 \implies Z_0 = \int \, d\theta \, p(\theta \mid m) = 1 \)

and \( \tau(K) = 1 \implies Z_K = p(y \mid m) \)

\[ \frac{Z_K}{Z_0} = \frac{Z_1}{Z_0} \frac{Z_2}{Z_1} \cdots \frac{Z_K}{Z_{K-1}} \]

Importance sample from \( f_{k-1}(\theta) \) as follows: \( \theta^{(r)} \sim f_{k-1}(\theta) \),

\[ \frac{Z_k}{Z_{k-1}} = \int \, d\theta \, \frac{f_k(\theta)}{f_{k-1}(\theta)} \frac{f_{k-1}(\theta)}{Z_{k-1}} \approx \frac{1}{R} \sum_{r=1}^{R} \frac{f_k(\theta^{(r)})}{f_{k-1}(\theta^{(r)})} = \frac{1}{R} \sum_{r=1}^{R} p(y \mid \theta^{(r)}, m)^{\tau(k) - \tau(k-1)} \]

- How tight are the variational bounds? Now we have a Gold Standard.
Varying the annealing schedule with random initialisation. $n = 480$, $R = 2^6 \ldots 2^{18}$
Ranking the true structure

VB score finds correct structure earlier, and more reliably
The Cheeseman-Stutz (CS) Approximation

The Cheeseman-Stutz approximation is based on:

\[
p(y|m) = p(z|m) \frac{p(y|m)}{p(z|m)} = p(z|m) \frac{\int d\theta \ p(\theta|m)p(y|\theta, m)}{\int d\theta \ p(\theta'|m)p(z|\theta', m)}
\]

which is true for any completion of the data: \( z = \{\hat{s}, y\} \).

We use the BIC approximation for both top and bottom integrals:

\[
\ln p(y|m) \approx \ln p(\hat{s}, y|m) + \ln p(\hat{\theta}|m) + \ln p(y|\hat{\theta}) - \frac{d}{2} \ln n
\]

\[
- \ln p(\hat{\theta}'|m) - \ln p(\hat{s}, y|\hat{\theta}) + \frac{d'}{2} \ln n
\]

\[
= \ln p(\hat{s}, y|m) + \ln p(y|\hat{\theta}) - \ln p(\hat{s}, y|\hat{\theta}),
\]

This can be corrected for \( d \neq d' \).

**Cheeseman-Stutz:** Run MAP to get \( \hat{\theta} \), complete data with expectations under \( \hat{\theta} \), compute CS approximation as above.
Comparison to Cheeseman-Stutz and BIC

- Averaging over about 100 samples.
- CS is much better than BIC, under some measures as good as VB.

Note: BIC and CS require estimates of the effective number of parameters. This can be difficult to compute. We estimate the effective number of parameters using a variant of the procedure described in Geiger, Heckerman and Meek (1996).
Part I: Summary

- Bayesian learning avoids overfitting and can be used to do model selection / averaging in directed graphs

- A practical approach: variational approximations

- Variational Bayesian EM for CE models and propagation algorithms

- These methods have advantages over MCMC in that they can provide fast approximate Bayesian inference. Especially important in machine learning applications with large data sets.

- Results
  - VB is uniformly better than BIC, at little computational cost
  - AIS is sometimes better than VB, but is sensitive to tuning parameters of MCMC, and 100 times slower.
  - CS is a huge improvement on BIC!
Part I: Future Work

• Other comparisons (e.g. Laplace).

• Larger networks and real-world data sets

• Variational Bayesian Cheeseman-Stutz method (coming soon!).

• Incorporate into a local search algorithm over structures (exhaustive enumeration is only of academic interest!).

• Extend to Gaussian and other non-discrete graphical models
Part II: Undirected Graphical Models

In an Undirected Graphical Model (or Markov Network), the joint probability over all variables can be written in a factored form:

\[ P(x) = \frac{1}{Z} \prod_j g_j(x_{C_j}) \]

where \( x = [x_1, \ldots, x_K] \), and

\[ C_j \subseteq \{1, \ldots, K\} \]

are subsets of the set of all variables, and \( x_S \equiv [x_k : k \in S] \).

This type of probabilistic model can be represented graphically.

**Graph Definition:** Let each variable be a node. Connect nodes \( i \) and \( k \) if there exists a set \( C_j \) such that both \( i \in C_j \) and \( k \in C_j \). These sets form the cliques of the graph (fully connected subgraphs).
Undirected Graphical Models: An Example

$$P(A, B, C, D, E) = \frac{1}{Z} g(A, C)g(B, C, D)g(C, D, E)$$

**Semantics:** Every node is conditionally independent from its non-neighbors given its neighbors.

**Conditional Independence:** $$X \perp \!\!\!\!\!\!\!\!\!\!\perp Y|V \iff p(X|Y, V) = p(X|V) \text{ for } p(Y, V) > 0$$
also $$X \perp \!\!\!\!\!\!\!\!\!\!\perp Y|V \iff p(X, Y|V) = p(X|V)p(Y|V).$$
Examples of Undirected Graphical Models

- Markov Random Fields

- Exponential Language Models

  \[ p(s) = \frac{1}{Z} p_0(s) \exp \left\{ \sum_i \lambda_i f_i(s) \right\} \]

- Products of Experts

  \[ p(x) = \frac{1}{Z} \prod_j p_j(x|\theta_j) \]

- Boltzmann Machines
Boltzmann Machines

Undirected graph over a vector of binary variables $s_i \in \{0, 1\}$. Variables can be hidden or visible (observed).

$$P(s|W) = \frac{1}{Z} \exp \left\{ \sum_{j<i} W_{ij} s_i s_j \right\}$$

where $Z$ is the partition function (normalizer)

Maximum Likelihood Learning Algorithm: a gradient version of EM

- **E step** involves computing averages w.r.t. $P(s_H|s_V, W)$ ("clamped phase"). This could be done via a propagation algorithm or (more usually) an approximate method such as Gibbs sampling.

- **M step** requires gradients w.r.t. $Z$, which can be computed by averages w.r.t. $P(s|W)$ ("unclamped phase").

Hebbian and anti-Hebbian rule:

$$\Delta W_{ij} = \eta [\langle s_i s_j \rangle_c - \langle s_i s_j \rangle_u]$$
Why Bayesian Learning?

• Useful prior knowledge can be included (e.g. sparsity)

• Avoids overfitting

• Error bars

• Model selection
A Simple Idea

Define the following joint distribution of weights $W$ and matrix of binary variables $S$, organized into $N$ rows (data vectors) and $M$ columns (features, variables). Some variables on some data points may be hidden and some may be observed.

$$p(S, W) = \frac{1}{Z} \exp \left\{ - \frac{1}{2\sigma^2} \sum_{i,j=1}^{M} W_{ij}^2 + \sum_{n=1}^{N} \sum_{j<i}^{M} W_{ij}s_{ni}s_{nj} \right\}$$

Where $Z = \int dW \sum_{S} \exp\{\ldots\}$ is a nasty partition function.

Gibbs sampling in this model is very easy!

- Gibbs sample $s_{ni}$ given all other $s$ and $W$: Bernouilli, easy as usual.
- Gibbs sample $W$ given $s$: diagonal multivariate Gaussian, easy as well.

What is wrong with this approach?
...a Strange Prior on $W$

$$p(S, W) = \frac{1}{Z} \exp \left\{ -\frac{1}{2\sigma^2} \sum_{i,j=1}^{M} W_{ij}^2 + \sum_{n=1}^{N} \sum_{j<i}^{M} W_{ij}s_{ni}s_{nj} \right\}$$

This defines a Boltzmann machine for the data given $W$, but defines a somewhat strange and hard to compute “prior” on the weights. What is the prior on $W$?

$$p(W) = \sum_S p(S, W) \propto N(0, \sigma^2 I) \sum_S \exp \left\{ \sum_{n,j<i}^{M} W_{ij}s_{ni}s_{nj} \right\}$$

where the second factor is data-size dependent, so it’s not a valid hierarchical Bayesian model of the kind $W \rightarrow S$. The second factor can be written as:

$$\sum_S \exp \left\{ \sum_{n,j<i}^{M} W_{ij}s_{ni}s_{nj} \right\} = \left( \sum_s \exp \left\{ \sum_{j<i}^{M} W_{ij}s_is_j \right\} \right)^N = Z(W)^N$$

This will not work!
Estimating $Z(W)$

To define a valid prior we need to compute $Z(W) \equiv \sum_s \exp \left\{ \sum_{j<i} W_{ij}s_is_j \right\}$.

$$p(W, S) = p(W)P(S|W) = P(W) \left[ \frac{1}{Z(W)^N} \exp \left\{ \sum_{n=1}^N \sum_{j<i} W_{ij}s_{ni}s_{nj} \right\} \right]$$

- **Mean-field**
  \[
  \ln Z(W) \geq \sum_s q(s) \sum_{j<i} W_{ij}s_is_j + H(q)
  \]
  where $q(s) = \prod_i m_i^{s_i} (1 - m_i)^{(1-s_i)}$ and $H$ is entropy.

- **Tree-based variational approximation.** Use a spanning tree for $q(s)$.

- **Bethe free energy.** Approximate $Z(W)$ by running loopy propagation and computing Bethe free energy.

- **Annealed Importance Sampling:**
  \[
  \frac{Z(W)}{Z(0)} = \frac{Z(\alpha_1 W)}{Z(0)} \frac{Z(\alpha_2 W)}{Z(\alpha_1 W)} \cdots \frac{Z(\alpha_t W)}{Z(\alpha_{t-1} W)}
  \]
  where $0 \leq \alpha_1 \ldots \leq \alpha_t = 1$.

- **Contrastive Sampling:** Brief Gibbs sampling starting at data to compute $Z(W)/Z(W')$.
The tree based approximation found an MST and then used Wiegerinck’s (UAI, 2000) variational approximation.
Metropolis with Importance Sampling Inner Loop

- Sample $S$ given $W$.

- Sample $W$ given $S$:
  - Propose $W'$ given $W$ from some symmetric proposal distribution
  - Compute acceptance probability
    \[ a(W'|W) = \min(1, \frac{p(W') p(S|W')}{p(W) p(S|W)}) \]
    where the second term is:
    \[ \frac{p(S|W')}{p(S|W)} = \exp \left\{ \sum_{n,j<i} (W'_{ij} - W_{ij}) s_{ni} s_{nj} \right\} \left( \frac{Z(W)}{Z(W')} \right)^N \]
    and the last term is estimated as below.

\[ \frac{Z(W)}{Z(W')} = \frac{\sum_s \exp \left\{ \sum_{j<i} W_{ij} s_i s_j \right\}}{\sum_s \exp \left\{ \sum_{j<i} W'_{ij} s_i s_j \right\}} = \left\langle \exp \left\{ \sum_{j<i} (W_{ij} - W'_{ij}) s_i s_j \right\} \right\rangle_{p(s|W')} \]
Contrastive Sampling to Estimate $Z(W)$

We wish to compute:

$$
\frac{Z(W)}{Z(W')} = \frac{\sum_s \exp \left\{ \sum_{j<i} W_{ij}s_is_j \right\}}{\sum_s \exp \left\{ \sum_{j<i} W'_{ij}s_is_j \right\}}
$$

- Start from data set
- Create a "corrupted" data set, for example, by Gibbs sampling briefly starting from data using $W$ (or $W'$, or many $W$s)
- Define $q(s)$ to be the empirical distribution of this corrupted data
- Then use:

$$
\frac{Z(W)}{Z(W')} \approx \frac{\sum_s q(s) \exp \left\{ \sum_{j<i} W_{ij}s_is_j \right\}}{\sum_s q(s) \exp \left\{ \sum_{j<i} W'_{ij}s_is_j \right\}}
$$

* A pseudo-Bayesian posterior can be computed by fixing $q(s)$. This concentrates "posterior" mass on weights that are good at explaining the real data and bad at explaining the corrupted data.

* We get a nice bias/variance trade-off and a huge computational savings!
Part II: Summary and Future Directions

• Tree-based approximations seem to work well, but exponentiating amplifies any errors.

• We are comparing these methods (and hopefully others) on small BMs where the exact solution can be computed.

• There is much future work in this area!

  http://www.gatsby.ucl.ac.uk/~zoubin
  http://www.variational-bayes.org/
Appendix
Conjugate-Exponential models

Let’s focus on conjugate-exponential (CE) models, which satisfy (1) and (2):

**Condition (1).** The joint probability over variables is in the exponential family:

\[ p(x, y | \theta) = f(x, y) g(\theta) \exp \{ \phi(\theta)^\top u(x, y) \} \]

where \( \phi(\theta) \) is the vector of natural parameters, \( u \) are sufficient statistics

**Condition (2).** The prior over parameters is conjugate to this joint probability:

\[ p(\theta | \eta, \nu) = h(\eta, \nu) g(\theta)^\eta \exp \{ \phi(\theta)^\top \nu \} \]

where \( \eta \) and \( \nu \) are hyperparameters of the prior.

Conjugate priors are computationally convenient and the hyperparameters have an intuitive interpretation:

- \( \eta \): number of pseudo-observations
- \( \nu \): values of pseudo-observations
A Useful Result

Theorem Given an iid data set \( y = (y_1, \ldots, y_n) \), if the model is CE then:

(a) \( q_{\theta}(\theta) \) is also conjugate, i.e.

\[
q_{\theta}(\theta) = h(\tilde{\eta}, \tilde{\nu})g(\theta)\exp \left\{ \phi(\theta)^{\top}\tilde{\nu} \right\}
\]

(b) \( q_{x}(x) = \prod_{i=1}^{n} q_{x_i}(x_i) \) is of the same form as in the E step of regular EM, but using pseudo parameters computed by averaging over \( q_{\theta}(\theta) \)

\[
q_{x_i}(x_i) \propto f(x_i, y_i) \exp \left\{ \overline{\phi}^{\top} u(x_i, y_i) \right\} = p(x_i|y_i, \overline{\phi})
\]

KEY points:

(a) the approximate parameter posterior is of the same form as the prior, so it is easily summarized in terms of two sets of parameters, \( \tilde{\eta} \) and \( \tilde{\nu} \);

(b) the approximate latent variable posterior, averaging over all parameters, is of the same form as the latent variable posterior for a single setting of the parameters, so again, it is easily computed using the usual methods.
The Bethe and Kikuchi Free Energies

- Let $z = (x, \theta)$. Form graph which includes parameters and latent variables.
- Express minus log marginal likelihood as a 'free energy' from stat. physics.

$$
- \ln p(y|m) = \min_{q(z)} \left( - \int q(z) \ln p(y, z|m) \, dz - \left( - \int q(z) \ln q(z) \, dz \right) \right)
$$

- Group $z$ into overlapping subsets and their intersections, $\{z_r\}$.
- **Bethe** if subsets include at most pairs of variables. **Kikuchi** if higher order.
- Approximate the entropy by ignoring terms of higher order:

$$
- \ln p(y|m) \approx F_{\text{Kikuchi}}(q) = \sum_r \int q_r(z_r) \mathcal{E}_r(z_r) \, dz_r + \sum_r c_r \int q_r(z_r) \ln q_r(z_r) \, dz_r
$$

- Bethe has same fixed points as BP on loopy graphs. Works well on ECC.
- Several procedures for minimizing the Bethe free energy as a functional of the approximate posterior distribution: CCCP (Yuille \textit{et al.}), Damped BP (Weiss,Murphy), Belief Optimisation (Welling & Teh)...

work in progress . . .
Clique Potentials and Markov Networks

**Definition:** a *clique* is a fully connected subgraph (usually maximal). $C_i$ will denote the set of variables in the $i^{th}$ clique.

1. Identify cliques of graph $G$

2. For each clique $C_i$ assign a non-negative function $g_i(C_i)$ which measures “compatibility”.

3. $p(X_1, \ldots, X_n) = \frac{1}{Z} \prod_i g_i(C_i)$ where $Z = \sum_{X_1 \cdots X_n} \prod_i g_i(C_i)$ is normalizer

The graph $G$ embodies the conditional independencies in $p$ (i.e. $G$ is a Markov Field relative to $p$): If $V$ lies in all paths between $X$ and $Y$ in $G$, then $X \perp\!\!\!\!\!\!\perp Y | V$. 
Hammersley–Clifford Theorem (1971)

**Theorem:** A probability function $p$ formed by a normalized product of positive functions on cliques of $G$ is a Markov Field relative to $G$.

**Definition:** The graph $G$ is a *Markov Field relative to* $p$ if it does not imply any conditional independence relationships that are not true in $p$.
(We are usually interested in the minimal such graph.)

**Proof:** We need to show that the neighbors of $X$, $\text{ne}(X)$ are a Markov Blanket for $X$:

$$p(X, Y, \ldots) = \frac{1}{Z} \prod_i g_i(C_i) = \frac{1}{Z} \prod_{i: X \in C_i} g_i(C_i) \prod_{j: X \notin C_j} g_j(C_j)$$

$$= \frac{1}{Z} f_1(X, \text{ne}(X)) f_2(\text{ne}(X), Y) = \frac{1}{Z'} p(X|\text{ne}(X)) p(Y|\text{ne}(X))$$

This shows that: $p(X, Y|\text{ne}(X)) = p(X|\text{ne}(X)) p(Y|\text{ne}(X)) \Leftrightarrow X \perp \! \! \! \perp Y|\text{ne}(X)$. 
Problems with Markov Networks

Many useful independencies are unrepresented — two variables are connected merely because some other variable depends on them:

```
Rain          Sprinkler
      |            |
      |            |
Ground wet
```

```
Rain          Sprinkler
      |            |
Ground wet
```

Marginal independence vs. conditional independence.

“Explaining Away”
Expressive Power of Bayesian and Markov Networks

No Bayesian network can represent these and only these independencies.

No Markov network can represent these and only these independencies.

No matter how we direct the arrows there will always be two non-adjacent parents sharing a common child $\implies$ dependence in Bayesian network but independence in Markov network.