Estimation in Exponential Families

Thanks to Yasemin Altun, Thomas Hofmann, Stephane Canu

Alexander J. Smola

Statistical Machine Learning Program
Canberra, ACT 0200 Australia
Alex.Smola@nicta.com.au

ICONIP 2006, Hong Kong, October 3

The Exponential Family

Definition

A family of probability distributions which satisfy

$$p(x;\theta) = \exp(\langle \phi(x), \theta \rangle - g(\theta))$$

Details

- $\phi(x)$ is called the sufficient statistic of x.
- $\mathcal{X}$ is the domain out of which x is drawn ($x \in \mathcal{X}$).
- $g(\theta)$ is the log-partition function and it ensures that the distribution integrates out to 1.

$$g(\theta) = \log \int_{\mathcal{X}} \exp(\langle \phi(x), \theta \rangle) dx$$

- Sometimes we need to specify a measure $\nu(x)$ on $\mathcal{X}$.

Example: Binomial Distribution

Tossing coins

With probability p we have heads and with probability $1 - p$ we see tails. So we have

$$p(x) = p^x (1-p)^{1-x} \text{ where } x \in \{0, 1\} =: \mathcal{X}$$

Massaging the math

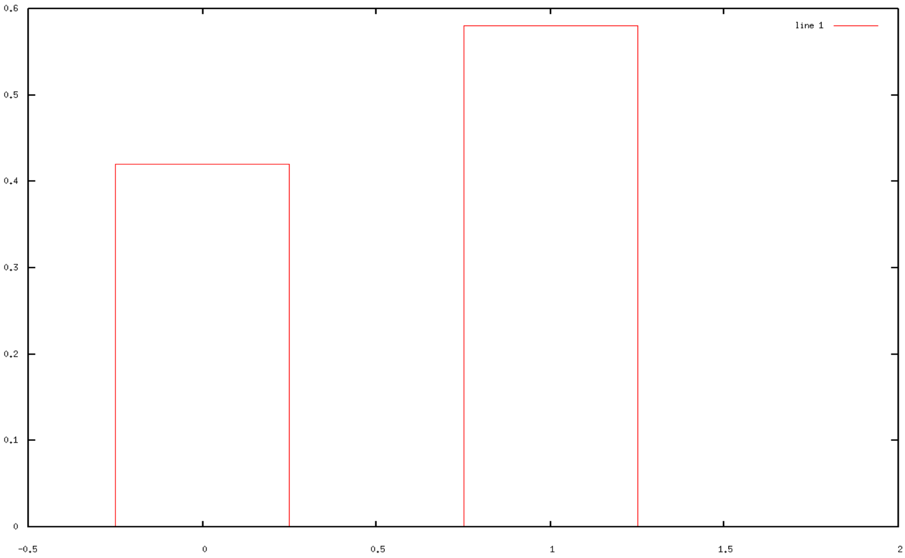
$$\begin{aligned} p(x) &= \exp \log p(x) \\ &= \exp \left(x \log p + (1-x) \log(1-p) \right) \\ &= \exp \Big(\langle \underbrace{(x, 1-x)}_{\phi(x)}, \underbrace{(\log p, \log(1-p))}_{\theta} \rangle \Big) \end{aligned}$$

Normalization

Once we relax the restriction on $\theta \in \mathbb{R}^2$ we need

$$g(\theta) = \log \left(e^{\theta_1} + e^{\theta_2} \right)$$

Example: Binomial Distribution



Example: Normal Distribution

Engineer's favorite

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\tfrac{1}{2\sigma^2}(x-\mu)^2\right) \text{ where } x \in \mathbb{R} =: \mathcal{X}$$

Massaging the math

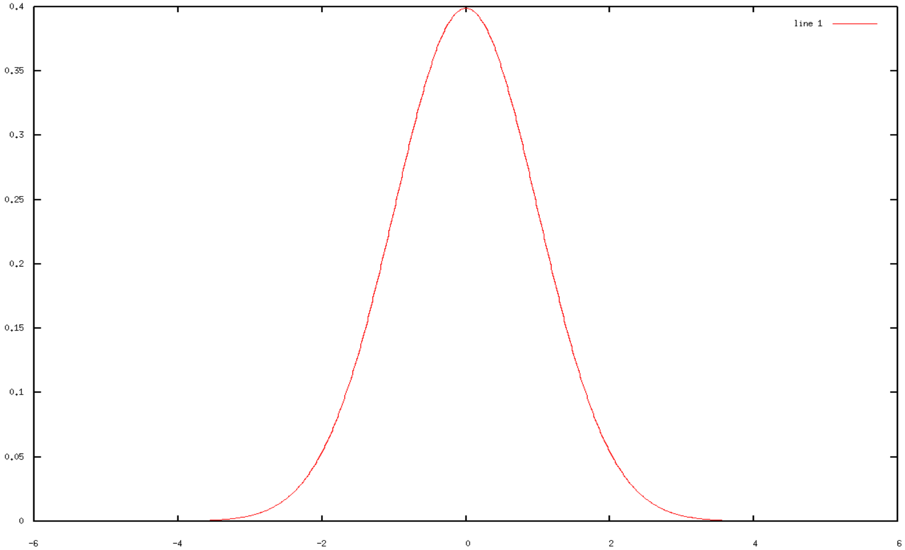
$$p(x) = \exp\left(-\frac{1}{2\sigma^2}x^2 + \frac{\mu}{\sigma^2}x - \frac{\mu^2}{2\sigma^2} - \frac{1}{2}\log(2\pi\sigma^2)\right)$$

$$= \exp\Bigg(\langle \underbrace{(x, x^2)}_{\phi(x)}, \theta\rangle - \underbrace{\left(\frac{\mu^2}{2\sigma^2} + \frac{1}{2}\log(2\pi\sigma^2)\right)}_{g(\theta)}\Bigg)$$

We need to solve (μ, σ^2) for θ. Tedious algebra yields $\theta_2 := -\frac{1}{2}\sigma^{-2}$ and $\theta_1 := \mu\sigma^{-2}$. We have

$$g(\theta) = -\frac{1}{4}\theta_1^2\theta_2^{-1} + \frac{1}{2}\log 2\pi - \frac{1}{2}\log -2\theta_2$$

Example: Normal Distribution



Example: Multinomial Distribution

Many discrete events

Assume that we have disjoint events $[1..n] =: \mathfrak{X}$ which all may occur with a certain probability p_x.

Guessing the answer

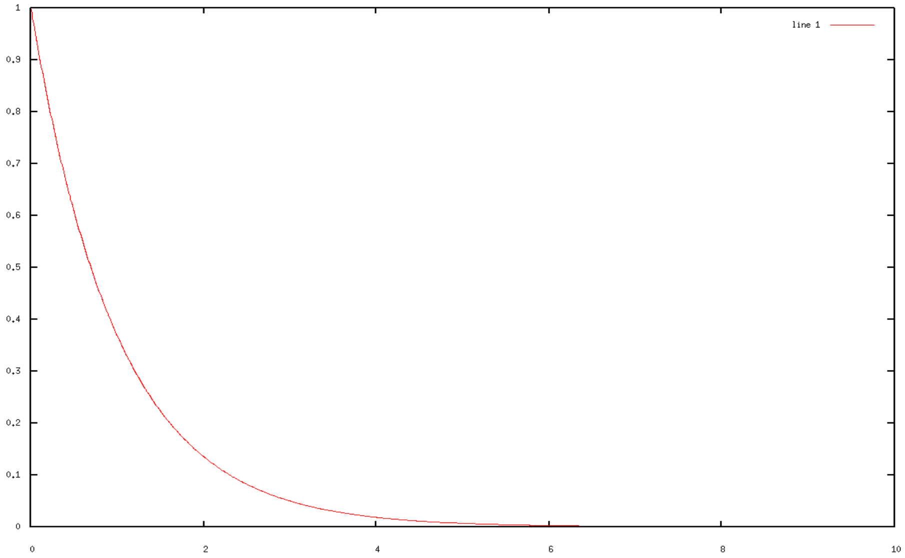
Use the map $\phi : x \to e_x$, that is, e_x is an element of the canonical basis $(0, \ldots, 0, 1, 0, \ldots)$. This gives

$$p(x) = \exp(\langle e_x, \theta \rangle - g(\theta))$$

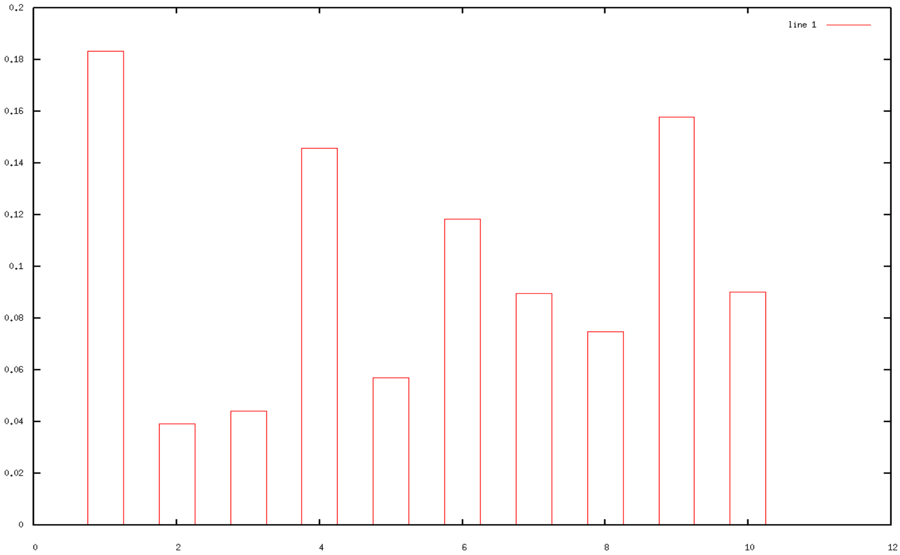
where the normalization is

$$g(\theta) = \log \sum_{i=1}^{n} \exp(\theta_i)$$

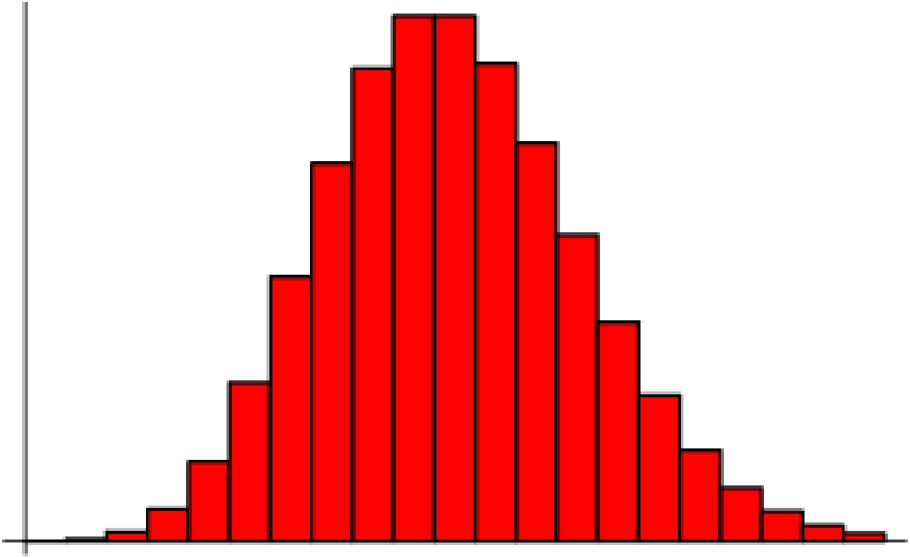
Example: Laplace Distribution



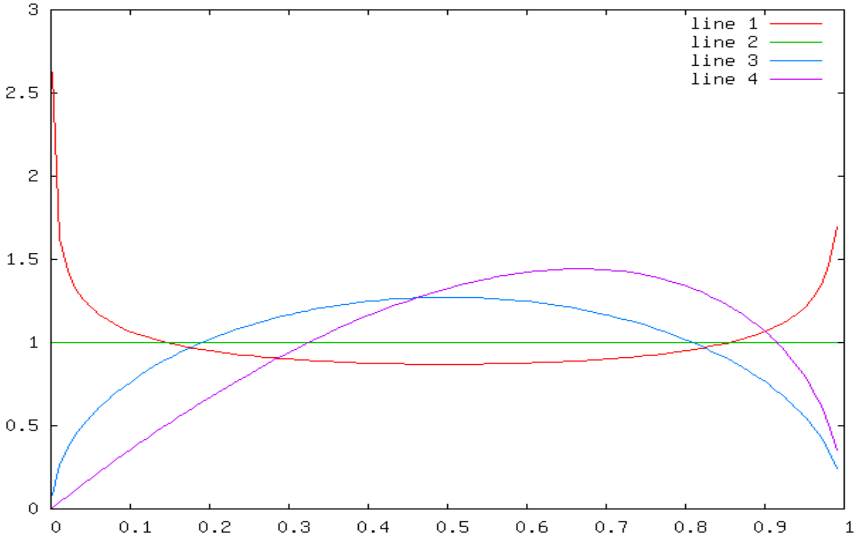
Example: Multinomial Distribution



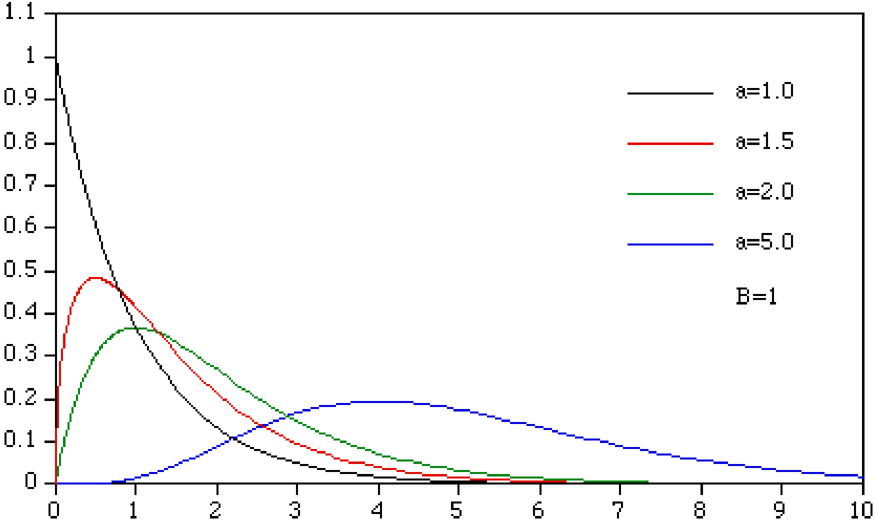
Example: Poisson Distribution



Example: Beta Distribution



Example: Gamma Distribution



Zoology of Exponential Families

Name	$\phi(x)$	Domain	Measure
Binomial	$(x, 1-x)$	$\{0, 1\}$	discrete
Multinomial	e_x	$\{1, \dots, n\}$	discrete
Poisson	x	$\mathbb{N}_0$	discrete
Laplace	x	$[0, \infty)$	Lebesgue
Normal	(x, x^2)	$\mathbb{R}$	Lebesgue
Beta	$(\log x, \log(1-x))$	$[0, 1]$	Lebesgue
Gamma	$(\log x, x)$	$[0, \infty)$	Lebesgue
Wishart	$(\log \lvert X \rvert, X)$	$X \succeq 0$	Lebesgue
Dirichlet	$\log x$	$x \in \mathbb{R}_+^n, \lVert x \rVert_1 = 1$	Lebesgue

Mini Summary

Exponential Familiy Distribution

$$p(x;\theta) = \exp(\langle \phi(x), \theta \rangle - g(\theta))$$

Examples

Binomial, Multinomial, Gaussian, Laplace, Wishart, Dirichlet, Gamma, Beta, ...

Lots of popular distributions are drawn from the exponential family. Unified treatment.

Normalization $g(\theta)$

$$g(\theta) = \log \int \exp\left(\langle \phi(x), \theta \rangle\right) dx$$

Log-partition function

g generates cumulants

$$\partial_\theta g(\theta) = \mathop{\mathbf{E}}_{x \sim p}[\phi(x)] \text{ and } \partial_\theta^2 g(\theta) = \mathop{\mathbf{Cov}}_{x \sim p}[\phi(x)]$$

... and so on for higher order cumulants ...

Consequence

$g(\theta)$ is convex

Proof

$$g(\theta) = \log \int \exp(\langle \phi(x), \theta \rangle) dx$$

$$\partial_\theta g(\theta) = \frac{\int \phi(x) \exp(\langle \phi(x), \theta \rangle) dx}{\int \exp(\langle \phi(x), \theta \rangle) dx}$$

Maximum Likelihood Estimation

Likelihood of a set

Given $X := \{x_1, \ldots, x_m\}$, drawn iid, we get

$$p(X;\theta) = \prod_{i=1}^{m} p(x_i;\theta) = \exp\left(\sum_{i=1}^{m} \langle \phi(x_i), \theta \rangle - mg(\theta)\right)$$
$$= \exp\left(m(\langle \mu, \theta \rangle - g(\theta))\right)$$

Here we set $\mu := \frac{1}{m} \sum_{i=1}^{m} \phi(x_i)$.

Maximum Likelihood

$$\underset{\theta}{\text{minimize}} - \log p(X;\theta) \iff \underset{\theta}{\text{minimize}}\, m\left(g(\theta) - \langle \mu, \theta \rangle\right)$$

First order conditions yield $\mathbf{E}[\phi(x)] = \mu$.

Benefit

Solving the maximum likelihood problem is **easy**.

Application: Discrete Events

Simple Data

Discrete random variables (e.g. tossing a dice).

Outcome	1	2	3	4	5	6
Counts	3	6	2	1	4	4
Probabilities	0.15	0.30	0.10	0.05	0.20	0.20

Maximum Likelihood Solution

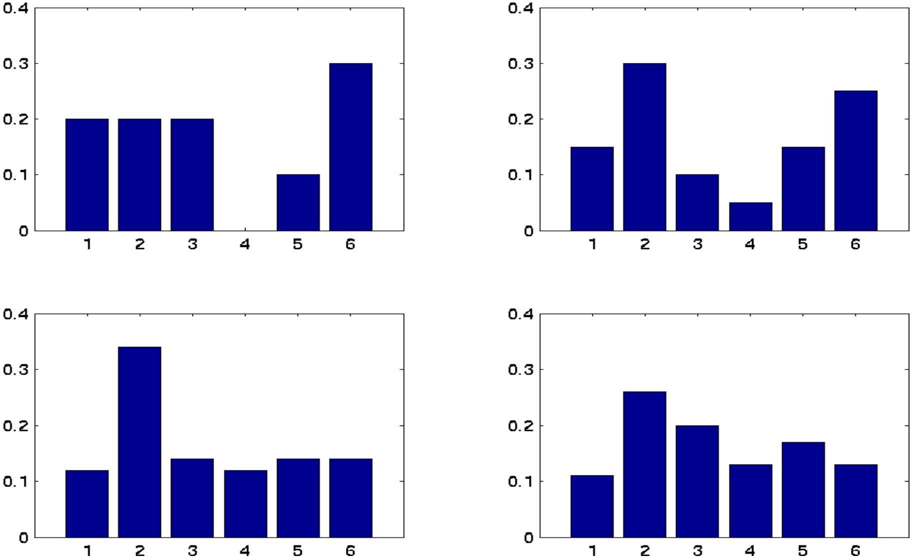
Count the number of outcomes and use the relative frequency of occurrence as estimates for the probability:

$$p_{\text{emp}}(x) = \frac{\#x}{m}$$

Problems

- Bad idea if we have few data.
- Bad idea if we have continuous random variables.

Tossing a dice



Mini Summary

Step 1: Observe Data

$x_1, \ldots, x_m$ drawn from distribution $p(x|\theta)$

Step 2: Compute Likelihood

$$p(X|\theta) = \prod_{i=1}^{m} \exp(\langle \phi(x_i), \theta \rangle - g(\theta))$$

Step 3: Maximize it

Take the negative log and minimize, which leads to

$$\partial_\theta g(\theta) = \frac{1}{m} \sum_{i=1}^{m} \phi(x_i)$$

This can be solved analytically or (whenever this is impossible or we are lazy) by Newton's method.

Caveat: Estimates can be bad if not enough data.

Priors

Problems with Maximum Likelihood

With not enough data, parameter estimates will be bad.

Prior to the rescue

Often we know where the solution should be. So we encode the latter by means of a prior $p(\theta)$.

Bayes Rule

$$p(\theta|X) \propto p(X|\theta)p(\theta)$$

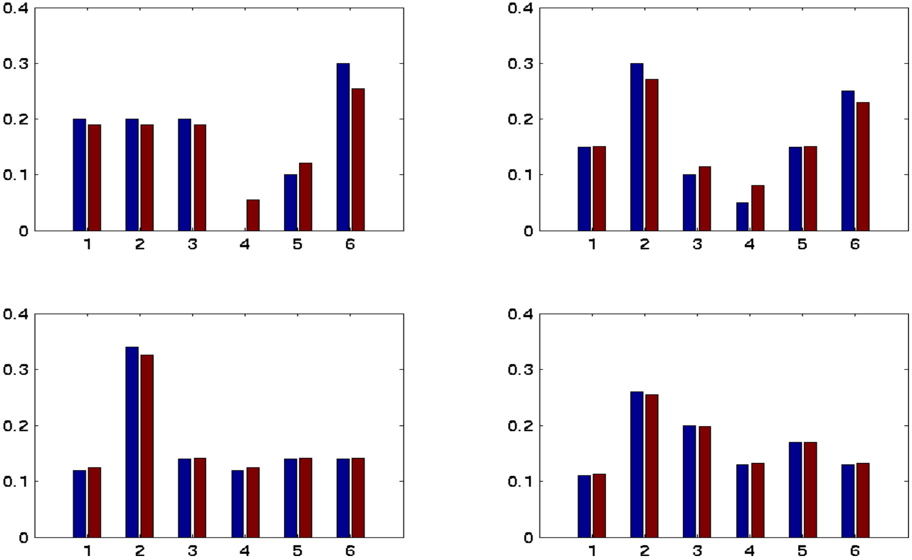
Normal Prior

$$p(\theta) \propto \exp\left(-\frac{1}{2\sigma^2}\|\theta\|^2\right).$$

Applying it (maximum a posteriori estimator)

$$-\log p(\theta|X) = m\left(g(\theta) - \langle\mu, \theta\rangle\right) + \tfrac{1}{2\sigma^2}\|\theta\|^2 + \text{ const.}$$

Tossing a dice with priors



Optimization Problems

Maximum Likelihood

$$\underset{\theta}{\text{minimize}} \sum_{i=1}^{m} g(\theta) - \langle \phi(x_i), \theta \rangle \implies \partial_\theta g(\theta) = \frac{1}{m} \sum_{i=1}^{m} \phi(x_i)$$

Normal Prior

$$\underset{\theta}{\text{minimize}} \sum_{i=1}^{m} g(\theta) - \langle \phi(x_i), \theta \rangle + \frac{1}{2\sigma^2} \|\theta\|^2$$

Mini Summary

Maximum Likelihood Estimation

- Convex optimization problem
- Match empirical observations and expectations
- Overfitting

Maximum a Posterioi Estimation

- Integration vs. Optimization
- Gaussian Prior
- Convex optimization problem

Graphical Model

Conditional Independence

- x, x' are conditionally independent given c, if

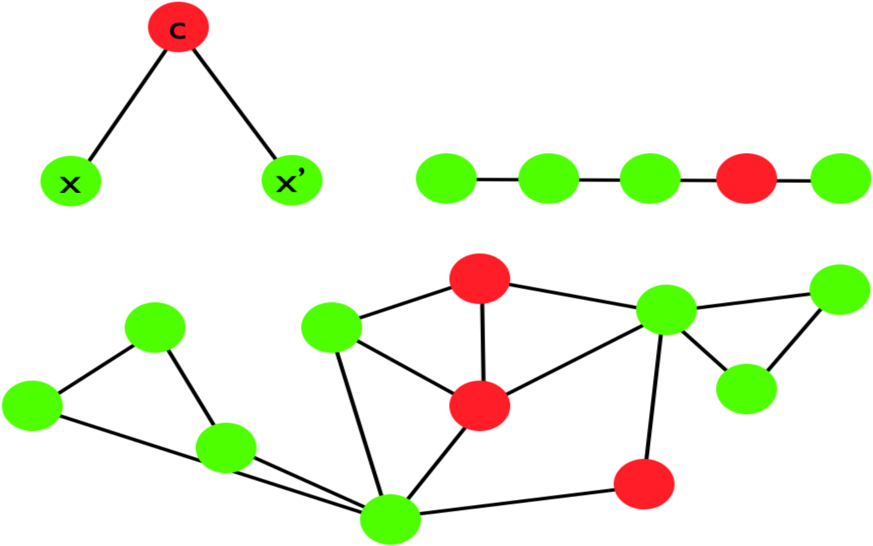
$$p(x, x'|c) = p(x|c)p(x'|c)$$

- Distributions can be simplified greatly by conditional independence assumptions.

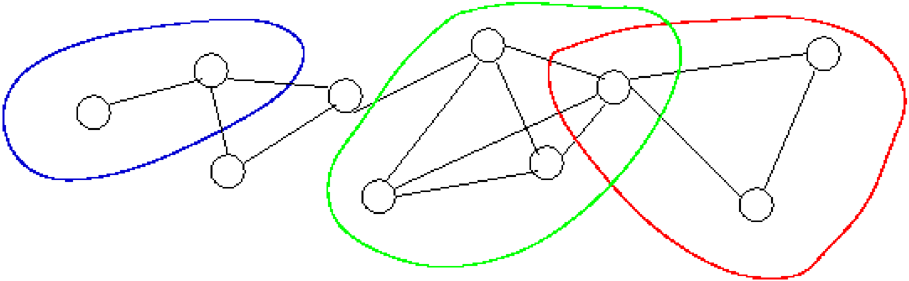
Markov Network

- Given a graph $G(V, E)$ with vertices V and edges E associate a random variable $x \in \mathbb{R}^{|V|}$ with G.
- Subsets of random variables $x_S, x_{S'}$ are conditionally independent given x_C if removing the vertices C from $G(V, E)$ decomposes the graph into disjoint subsets containing S, S'.

Conditional Independence



Cliques



Definition

- Subset of the graph which is fully connected
- Maximal Cliques (they define the graph)

Advantage

- Easy to specify dependencies between variables
- Use graph algorithms for inference

Hammersley Clifford Theorem

Problem

Specify $p(x)$ with conditional independence properties.

Theorem

$$p(x) = \frac{1}{Z} \exp\left(\sum_{c \in \mathfrak{C}} \psi_c(x_c)\right)$$

whenever $p(x)$ is nonzero on the entire domain.

Application

Apply decomposition for exponential families where $p(x) = \exp(\langle \phi(x), \theta \rangle - g(\theta))$.

Corollary

The sufficient statistics $\phi(x)$ decompose according to

$$\phi(x) = (\ldots, \phi_c(x_c), \ldots) \Longrightarrow \langle \phi(x), \phi(x') \rangle = \sum_c \langle \phi_c(x_c), \phi_c(x'_c) \rangle$$

Example: Normal Distributions

Sufficient Statistics

Recall that for normal distributions $\phi(x) = (x, xx^\top)$.

Clifford Hammersley Application

- $\phi(x)$ must decompose into subsets involving only variables from each maximal clique.
- The linear term x is OK by default.
- The only nonzero terms coupling $x_i x_j$ are those corresponding to an edge in the graph $G(V, E)$.

Inverse Covariance Matrix

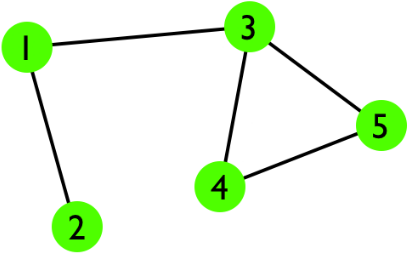
- The natural parameter aligned with $xx^\top$ is the inverse covariance matrix.
- Its sparsity mirrors $G(V, E)$.
- Hence a sparse inverse kernel matrix corresponds to graphical model!

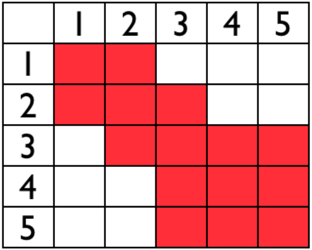
Example: Normal Distributions

Density

$$p(x|\theta) = \exp\left(\sum_{i=1}^{n} x_i\theta_{1i} + \sum_{i,j=1}^{n} x_i x_j \theta_{2ij} - g(\theta)\right)$$

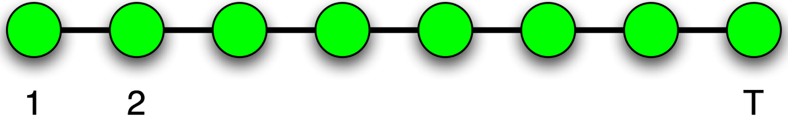
Here $\theta_2 = \Sigma^{-1}$, is the inverse covariance matrix. We have that $(\Sigma^{-1})_{[}ij] \neq 0$ only if (i, j) share an edge.





Computing $g(\theta)$

Markov Chain



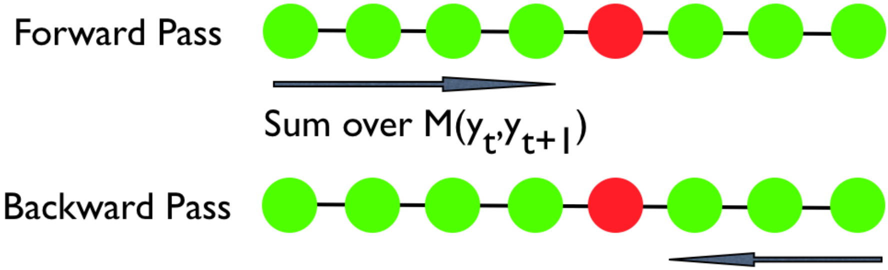
Dynamic Programming

$$g(\theta) = \log \sum_{x_1,\ldots,x_T} \prod_{t=1}^{T} \underbrace{\exp\left(\langle \phi(x_t, x_{t+1}), \theta \rangle\right)}_{M_t(x_t, x_{t+1})}$$

$$= \log \sum_{x_1} \sum_{x_2} M_1(x_1, x_2) \sum_{x_3} M_2(x_2, x_3) \ldots \sum_{x_T} M_T(x_{T-1}, x_T)$$

We can compute $g(\theta), p(x_t|\theta)$ and $p(x_t, x_{t+1}|\theta)$ via dynamic programming.

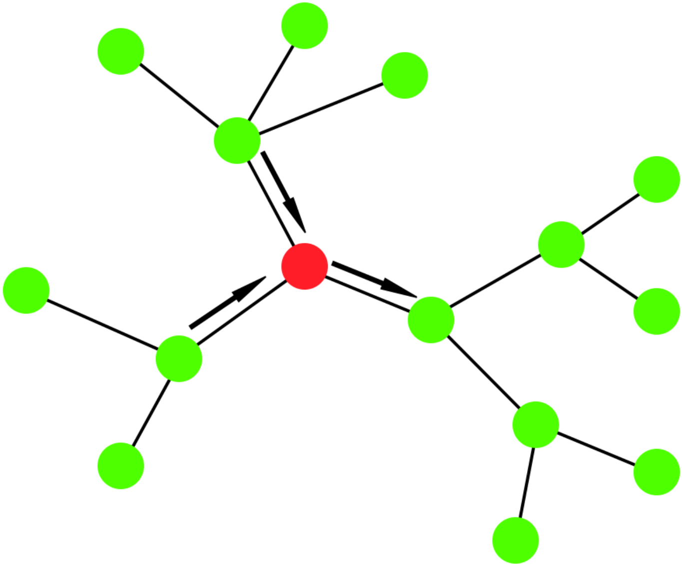
Forward Backward Algorithm



Key Idea

- Store sum over all $x_1, \ldots, x_{t-1}$ (forward pass) and over all $x_{t+1}, \ldots, x_T$ as intermediate values
- We get those values for all positions t in one sweep.
- Extend this to message passing (when we have trees).

Message Passing



Message Passing

Idea

Extend the forward-backward idea to trees.

Algorithm

- Given clique potentials $M(x_i, x_j)$
- Initialize messages $\mu_{ij}(x_j) = 1$
- Update outgoing messages by

$$\mu_{ij}(x_j) = \sum_{x_i \in \mathcal{Y}_i} \prod_{k \neq j} \mu_{ki}(x_i) M_{ij}(x_i, x_j)$$

Here (i, k) is an edge in the graph.

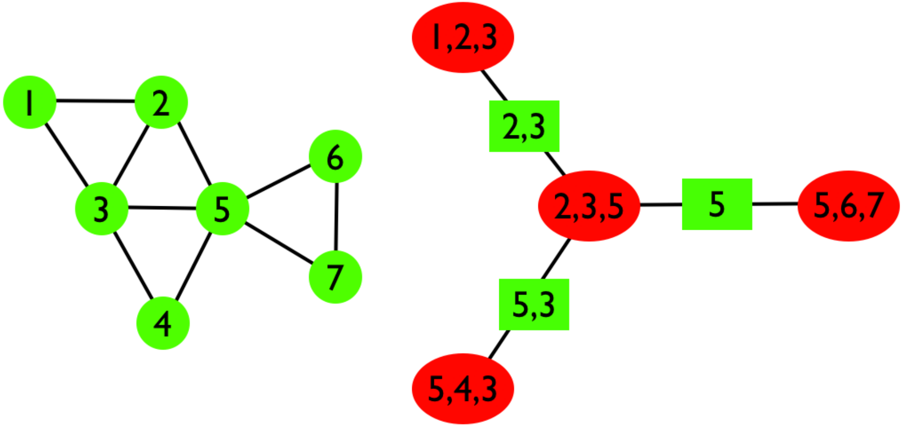
Theorem

The message passing algorithm converges after n iterations (n is diameter of graph).

Hack

Use this for graphs with **loops** and hope . . .

Junction Trees



Stock standard algorithms available to transform graph into junction tree. Now we can use message passing . . .

Junction Tree Algorithm

Idea

Messages involve variables in the separator sets.

Algorithm

- Given clique potentials $M_c(x_c)$ and separator sets s.
- Initialize messages $\mu_{c,s}(x_s) = 1$
- Update outgoing messages by

$$\mu_{c,s}(x_s) = \sum_{x_c \setminus x_s} \prod_{s' \neq s} \mu_{c',s'}(x_{s'}) M_c(x_c)$$

Here s' is a separator set connecting c with c'.

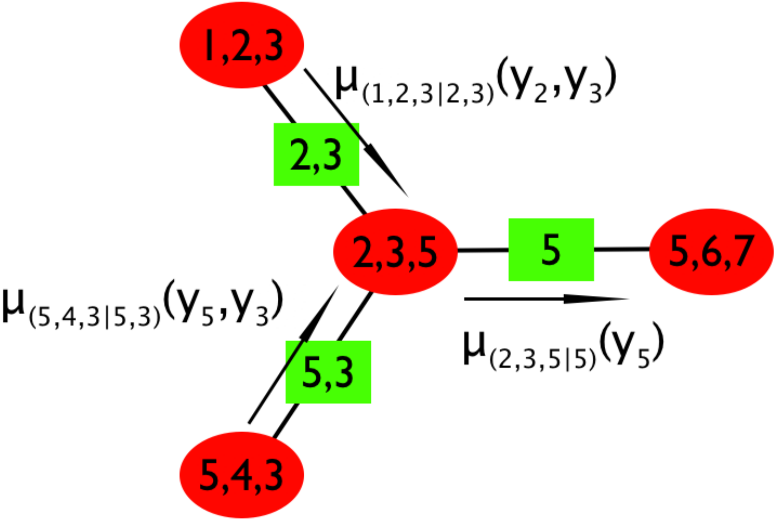
Theorem

The message passing algorithm converges after n iterations (n is diameter of the hypergraph).

Hack

Use this for graphs with **loops** and hope . . .

Example



Mini Summary

Hammersley Clifford Theorem

- Conditional Independence
- Decomposition of joint density
- Simplification of the model

Message Passing

- For Markov chains the problems decomposes
- Can solve exponential sum in linear time
- Generalization to trees
- Junction trees
- Loopy belief propagation