

# C19 : Lecture 1 : Graphical Models

Frank Wood

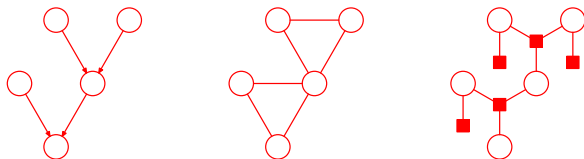
University of Oxford

January, 2017

Many figures from PRML (?)

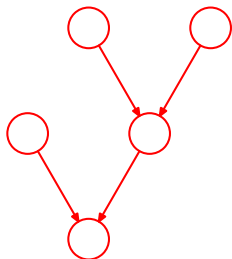
- Correspond to a subset of probability models
- Visualisation of structure of probability model
- Useful for model design and development
- Encode properties (some explicitly, others via inspection)
  - Conditional independence (inspection)
- Inference and learning can be formulated in terms of computational operations on the graph

# The Graphs of Graphical Models



Graphs consist of

- Vertices
  - Correspond to random variables (and factors)
  - Have types
  - Can represent structured object like vectors, arrays, distributions, infinite sequences, embedded graphical models, etc.
- Edges
  - These will correspond to *dependencies*
  - Can be directed (arrow) or undirected
  - Usually are associated with conditional density functions or compatibility functions



Such models are known as

- Generative graphical models
- Bayes nets

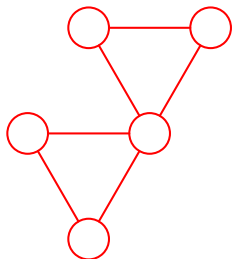
Vertices

- Variables

Edges

- Conditional probabilities

Good for generative descriptions of data. Conditional probability statements of the form “if A then B with some probability.”



Such models are known as

- Markov random fields
- Ising models

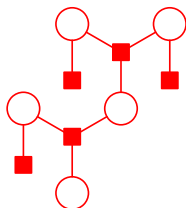
Vertices

- Variables

Edges

- “Compatibility functions”

Good for specifying constraints between variables where generative relationships aren't obvious (i.e. neighbouring pixel similarity in images)



Generalisation of the undirected and directed graphical models.

Vertices

- Circles : variables
- Squares : “Compatibility functions”

Edges

- Indicate variable is argument of compatibility function

Good for writing down and thinking about message-passing inference algorithms.

Consider an arbitrary joint distribution over three variables  $a$ ,  $b$ , and  $c$ ,  $p(a, b, c)$ . Recall that

$$p(a, b, c) = p(c|a, b)p(a, b) = p(c|a, b)p(b|a)p(a)$$

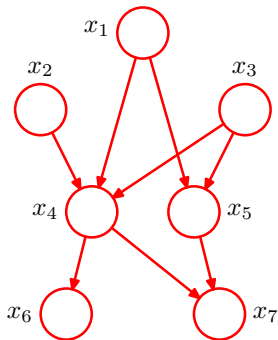
*Note* : This decomposition always holds

Also

- Corresponds to a generative scheme
  - Generate (sample)  $a$
  - Then generate (sample)  $b$  conditioned on  $a$
  - ...
- Dropping conditioning terms restricts the expressivity of the model

Graphical representation can directly be translated into joint

$$p(\mathbf{x}) = p(x_1)p(x_2)p(x_3)p(x_4|x_1, x_2, x_3)p(x_5|x_1, x_3)p(x_6|x_4)p(x_7|x_4, x_5)$$



More generally

- There is a distribution (term) for each vertex
- The variables which appear in each terms' condition are connected via inbound edges

In math

$$p(\mathbf{x}) = \prod_k p(x_k | \text{pa}_k)$$



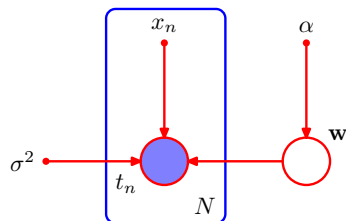
## Homework

If each term normalises (i.e. is a proper density or distribution), what value does the following expression take?

$$\int p(\mathbf{x})d\mathbf{x} = \int \int \int \int \int \int \int p(x_1)p(x_2)p(x_3) \dots p(x_7|x_4, x_5)dx_1 \dots dx_7$$

Note for discrete variables integrals may be interchanged for sums.

# Notational Shorthand for Products and Constants

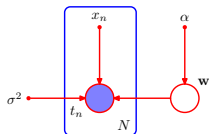


- Squares indicate multiplied replication
- Filled circles indicate “observed” variables
- Small dots (or no dots) indicate fixed quantities

In math

$$p(\mathbf{t}, \mathbf{x}, \sigma^2, \mathbf{w}, \alpha) = \prod_{n=1}^N p(t_n | \mathbf{w}, x_n, \sigma^2) p(\mathbf{w} | \alpha)$$

# Must Specify Conditional Densities



$$t_n | \mathbf{w}, x_n, \sigma^2 \sim \text{Normal}(\mathbf{w}^T x_n, \sigma^2)$$

$$\mathbf{w} | \alpha \sim \prod_{d=1}^D \text{Normal}(0, \alpha) = \text{Normal}(\mathbf{0}, \alpha \mathbf{I})$$

Homework :

- To what well known model does this correspond?
- Is this a supervised or unsupervised model?

One Goal : Posterior distribution of parameters given data

$$p(\mathbf{w}|\mathbf{t}) \propto p(\mathbf{w}) \prod_{n=1}^N p(t_n|\mathbf{w})$$

Fixed quantities often are dropped because they always appear on conditioning side of equation(s)

Quiz :

- Why is this a proportionality ( $\propto$ )? (hint : use Bayes rule)
- Can this be read from the directed graphical model? (hint : be very careful and think about conditional *independence*)
- *Important concept!* Why do we want  $p(\mathbf{w}|\mathbf{t})$ ? What are its uses?

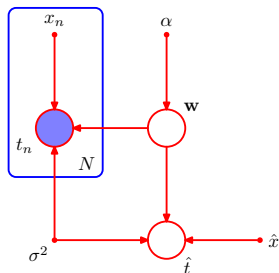
Another goal : out of sample predictions

With

$$\hat{t}|\mathbf{w}, \hat{x}, \sigma^2 \sim \text{Normal}(\mathbf{w}^T \hat{x}, \sigma^2)$$

then

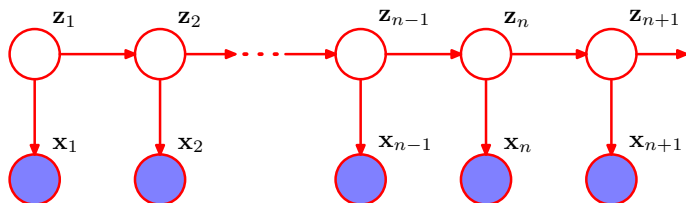
$$\begin{aligned} p(\hat{t}|\mathbf{x}, \mathbf{t}, \alpha, \sigma^2) &= \int p(\hat{t}, \mathbf{w}|\mathbf{x}, \hat{x}, \mathbf{t}, \alpha, \sigma^2) d\mathbf{w} \\ &= \int p(\hat{t}|\mathbf{w}, \hat{x}) p(\mathbf{w}|\mathbf{t}) d\mathbf{w} \end{aligned}$$



Quiz :

- How do you compute this integral?
- What does it mean to integrate out  $\mathbf{w}$ ?

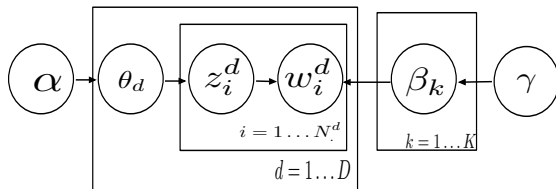
# Hidden Markov Model



$$p(z_n | z_{n-1}, \mathbf{A}) = \prod_{k=1}^K \prod_{j=1}^K A_{jk}^{z_{(n-1)j} z_{nk}}$$

$$p(z_1 | \boldsymbol{\pi}) = \prod_{k=1}^K \pi_k^{z_{1k}}$$

$$x_n | z_{nk} = 1 \sim F(\boldsymbol{\theta}_k)$$



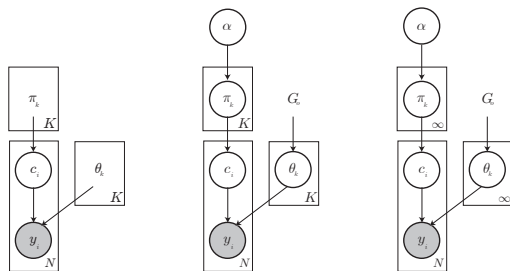
$$\theta_d \sim \text{Dir}_K(\alpha \vec{1})$$

$$\beta_k \sim \text{Dir}_M(\gamma \vec{1})$$

$$z_i^d \sim \text{Discrete}(\theta_d)$$

$$w_i^d \sim \text{Discrete}(\beta_{z_i^d})$$

# Gaussian Mixture Model

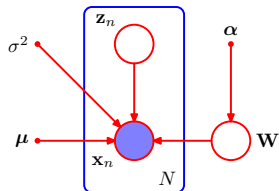


**Figure:** From left to right: graphical models for a finite Gaussian mixture model (GMM), a Bayesian GMM, and an infinite GMM

$$\begin{aligned}
 c_i | \vec{\pi} &\sim \text{Discrete}(\vec{\pi}) \\
 \vec{y}_i | c_i = k; \Theta &\sim \text{Gaussian}(\cdot | \theta_k). \\
 \vec{\pi} | \alpha &\sim \text{Dirichlet}(\cdot | \frac{\alpha}{K}, \dots, \frac{\alpha}{K}) \\
 \Theta &\sim \mathcal{G}_0
 \end{aligned}$$



# Probabilistic Principal Component Analysis



$$p(\mathbf{W}|\alpha) = \prod_{i=1}^M \left(\frac{\alpha_i}{2\pi}\right)^{D/2} \exp\left\{-\frac{1}{2}\alpha_i \mathbf{w}_i^T \mathbf{w}_i\right\}$$
$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I})$$
$$p(\mathbf{x}_n|\mathbf{z}) = \mathcal{N}(\mathbf{x}_n|\mathbf{W}\mathbf{z}_n, \sigma^2\mathbf{I})$$

- Distributions, PDFs, Sampling
  - Dirichlet
  - Discrete / Multinomial
  - Multivariate Normal
- Vector Algebra
- Procedures for determining conditional independence(s) given a graphical model

C M Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.