Lecture 2
Day 2.

Note Title 4/2/2006

Probability Probability on Graphs.

Simplest Case.

Two variables. joint probability $P(a,b)$

conditional prob $p(a|b)$ & $p(b|a)$

marginal probs $p(a)$ & $p(b)$

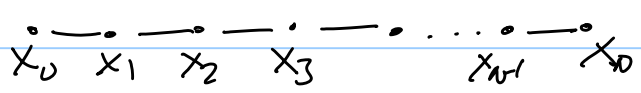
$$P(a,b) = P(a|b)P(b) = P(b|a)P(a)$$

Implies Bayes Rule

$$P(b|a) = \frac{P(a|b)P(b)}{P(a)}$$ //

Many Problems in Artificial Intelligence, Bioinformatics, Finance, Physics, Statistics can be formulated in terms of probability distributions defined over graphs.

The model we considered for the last two lectures. is only a special case



This lecture gives a rapid tour over these classes of models.

(1) Causal Models / Directed Graphical Models.
(2) Undirected Graphical Models.
(3) Hidden Markov Models.
(4) Stochastic Grammars.

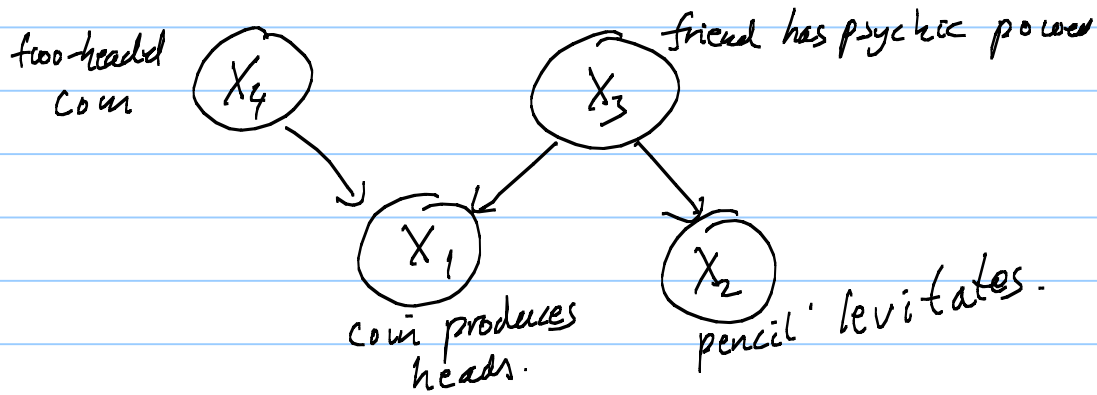
Sampling Methods can be used to compute properties of interest.

Page 2 -

Judea Pearl (UCLA) showed the ability of probabilistic models on graphs to person reasoning tasks (1988). Unlike standard logical methods (standard in AI) at the time these prob. methods were able to revise conclusion to take into account new information.

Friend claims psychic powers — test on coin tossing
— test on pencil levitating

X_1 represent truth of coin being flipped to give heads
X_2 " pencil levitating.
X_3 " friend having psychic powers
X_4 " use of a "two-headed coin.



Represent the joint distribution:

$$P(X_1, X_2, X_3, X_4) = P(X_1 | X_3, X_4) P(X_2 | X_3) p(X_3) p(X_4)$$

The graph structure (nodes and edges) represents the dependencies between variables — the Markov structure.

Direct Relation: $P(X_2 | X_1, X_3, X_4) = P(X_2 | X_3)$

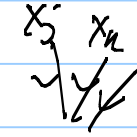
If you know X_3, then knowing X_4 (or X_1) won't give info on X_2

Indirect. But if you don't know X_3 or X_1, knowing X_4 gives info on X_2.

This is a special case of a Directed Graphical Model:

$$P(X_1, X_2 \ldots, X_M) = \prod_i P(X_i \mid Pa(X_i))$$

where $Pa(X_i)$ are the parents of X_i

E.G. 

(X_j, X_k, X_ℓ) are the parents of X_i

Markov Condition:

Conditioned on parents, each variable X_i is independent of all other variables.

Directed Graphical models make the causal structure of the data 'fairly explicit' (there is more to causality - see Pearl 2000 - e.g. correlation does not equal causation.

- Eating Icecream is correlated with drowning.
- But Icrecream doesn't cause drowning
- To verify this, need to intervene - e.g. ban selling Icecream at Beaches, see that drowning stays fixed.

Advance of Graphical Structure

(i) Knowledge - which variables influence each other

(ii) Data Reduction - full model with N variables needs $K^N - 1$ numbers specified - (but only 8 for psychic example and not $2^4 - 1 = 15$). Efficient computation

(iii) Intervention - causality (Pearl 2000)

Page 4

Computations are simplified by exploiting the graphical structure:

Eg. $$P(X_1=1) = \sum_{X_2}\sum_{X_3}\sum_{X_4} P(X_1=1, X_2, X_3, X_4)$$

$$= \sum_{X_2}\sum_{X_3}\sum_{X_4} P(X_1=1|X_3, X_4)\, P(X_2|X_3) P(X_3) P(X_4)$$

$$= \sum_{X_3}\sum_{X_4} P(X_1=1|X_3, X_4)\, P(X_3) P(X_4).$$

Sum over X_2 can be done automatically — because $\sum_{X_2} P(X_2|X_3) = 1$

This is a simple example of dynamic programming. The psychic graph has no closed loops.

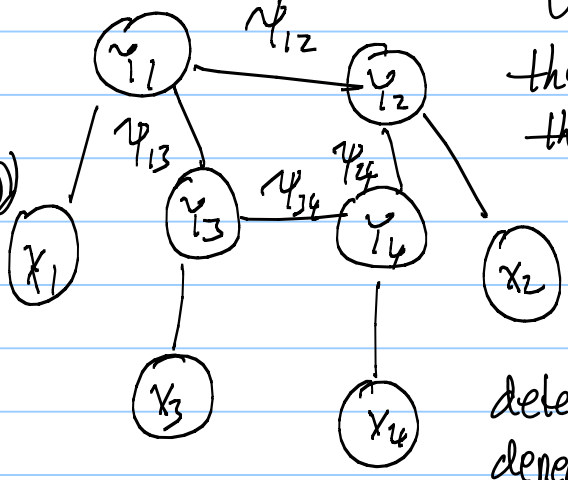
Note: Directed Graphical Models are expressed in form $P(\underline{X}) = \prod_i P(X_i | Pa(X_i))$

→ so sampling from them is straightforward.
→ sample parents, then sample children

Undirected Graphical Models: Markov Random Fields

Often two types of variables.
observed $\{x_i\}$, unobserved $\{y_i\}$

Markov condition
$P(y_i | \underline{y}_{/i})$
$= P(y_i | \{y_j : j \in N(i)\})$



Undirected edges define the neighbourhood structure of the graph.
E.G. $N(i)$ denotes the neighbours of node i.

The neighbourhood structure determines the probabilistic dependencies – i.e. Markov condition

The graph edges are associated with potentials and not with conditional probabilities (unlike Directed models).
The probability distribution is expressed in terms of potential functions. E.G.

$$P(\underline{x}|\underline{y}) = \prod_i P(x_i|y_i)\, P(\underline{y})$$

with $P(\underline{y}) = \frac{1}{Z} \prod_{ij \in \mathcal{E}} e^{\psi_{ij}(y_i, y_j)} \prod_i e^{\psi_i(y_i)}$ where $\mathcal{E}$ is the edges.

Note: If this graph has no closed loops, then it can be converted into a directed graph using DP (like last lecture). This makes sampling straightforward.
If it has closed loops then sampling is much harder and needs Markov Chain Monte Carlo (MCMC) – see later in this course.

Example: 2D-Ising Models and Potts models

Ising Model in 1D $\pi(\underline{x}) = \frac{1}{Z} e^{-E(\underline{x})}$

$E(\underline{x}) = \beta \sum_{i=1}^{N} x_{i-1} x_i \qquad x_i \in \{\pm 1\}$

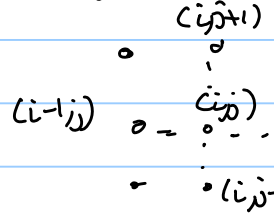
Can be converted (previous lecture) to a directed model

$\pi_0(x_0)\, \pi_1(x_1|x_0) \cdots \pi_N(x_N|x_{N-1})$

A 2-D Ising model is defined over a lattice with coordinates i,j

$\pi(\underline{x}) = \frac{1}{Z} e^{-E[\underline{x}]}$

$x_{ij} \in \{\pm 1\}$



neighbours

$E[\underline{x}] = -\sum_{(i,j)(k,l)} J_{ij,kl}\, x_{ij}\, x_{k,l}$

where $J_{ij,kl} = 0$ unless $(k,l) \in N(i,j)$

EG. $N(i,j) = \{(i-1,j), (i+1,j), (i,j+1), (i,j-1)\}$

nearest neighbours (on lattice)

In general, can't convert this into conditional distributions — need MCMC to do sampling.

More Generally, Potts Model. $x_{ij} \in \{s_1, \ldots s_k\}$ k possible values

$E[\underline{x}] = -J \sum_{ij,kl} \varphi_{ij,kl}(x_{ij}, x_{kl})$

EG.: Special case $E[\underline{x}] = -J \sum_{ij,kl} \delta_{x_{ij}, x_{kl}}$ $\qquad \varphi(x_{ij}, x_{kl}) = \delta_{x_{ij}, x_{kl}}$ Kronecker delta

Many applications of Ising/Potts — Physics, Information Theory, Vision, Bioinformatics

Hidden Markov Models (HMM)

Used for speech and language processing

Sequence of T observations $\{x_t : t = 1, \ldots T\}$

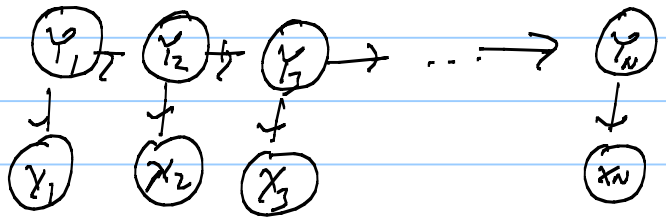
generated by hidden states $\{y_t : t = 1 \ldots T\}$

Joint Distribution:

$$P(\{y_t\}, \{x_t\}, w) = P(w) P(y_1 | w) P(x_1 | y_1, w) \prod_{t=2}^{T} P(y_t | y_{t-1}, w) P(x_t | y_t, w).$$

1-D nature of graph (no closed loops) means that dynamic programming can be used.

Word w.



Applying HMM's to recognize words requires DP algorithms to: (i) learn $P(x_t | y_t, w)$ & $P(y_t | y_{t-1}, w)$ for each w.

(ii) evaluate the probability

$$P(\{x_t\}, w) = \sum_{\{y_t\}} P(\{y_t\}, \{x_t\}, w)$$ for word.

(Also EM)

(iii) estimate $w^* = \arg\max_w \sum_{\{y_t\}} P(\{y_t\}, w | \{x_t\})$

Page 8.

Probabilistic Context Free Grammars. (PCFG)

or Stochastic CFG (SCFG)

Define non-terminal nodes

S, NP, VP, AT, NNS, VBD, PP, IN, DT, NN

where S is a sentence.

VP is a verb phrase . . .

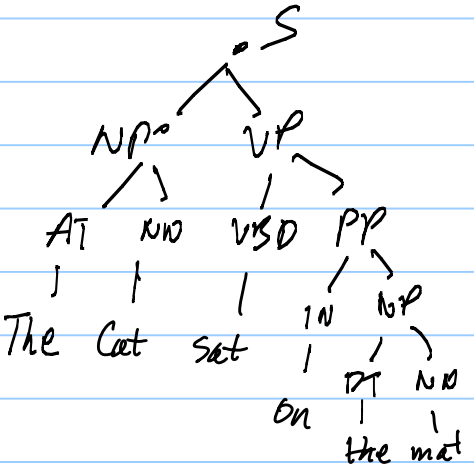
Terminal nodes are words from a dictionary

(eg. "the" "cat" "sat" "on" "the" "mat").

Define production rules which are applied to non-terminal nodes to generate child nodes.

eg. $S \rightarrow NP, VP$ or $NN \rightarrow$ "cat"

Define probability distribution for the choice of rule used



Generate a sentence by starting with node S, and sampling the production rules.

Parse an input sentence by choosing the most probable parse tree

Learn probabilities of rules.

DP useful – no closed loops / independence.

Summary.

Many types of probability models on graphs - Directed Models, Undirected (eg Ising Potts), Hidden Markov Models, Stochastic Grammars.

Sampling can be used to estimate properties of the models $\rightarrow \sum_x \Pi(x) h(x)$ or $\hat{x} = \arg\max_x \Pi(x)$

As in previous lectures, if the graphs have no closed loops then dynamic programming (DP) can help. In general, sampling is possible by expressing the distributions in terms of conditional probabilities. See previous two lectures, and the next few lectures.

Sampling is much harder if the graphs have closed loops - requires MCMC (2nd half of course)

Note: these models also motivate a harder problem - how to learn these distributions from training examples? eg. learn HMMs for speech, stochastic grammars. Requires EM algorithms, Data Augmentation Sampling - later in course if time.