Bayes nets and autoregressive models

A: The Bayes net formalism

Alarm example

Pearl. *Probabilistic reasoning in intelligent systems*. 1988

Mr. Holmes receives a call from his neighbor Dr. Watson, who states that he hears the sound of a burglar alarm from the direction of Mr. Holmes' house. While preparing to rush home, Mr. Holmes recalls that Dr. Watson is known to be a tasteless practical joker...

As Mr. Holmes is preparing to leave his office, he recalls that his daughter is scheduled to arrive home at any minute. If greeted by an alarm sound, she probably ($P = 0.70$) would phone him for instructions.

As he is debating whether or not to rush home, Mr. Holmes remembers reading in the instruction manual of his alarm system that the device is sensitive to earthquakes and can accidentally ($P = 0.20$) be triggered by one.

Bayes net formalism

Let $X_1, \ldots, X_n$ be a collection of random variables.

- Let G be a **directed acyclic graph** (DAG) with nodes named $X_1, \ldots, X_n$.
- Let Π_i denote the parents of X_i in G.

If P is a distribution over $(X_1, \ldots, X_n)$, we say P **factors over** G if

$$P(X_1, \ldots, X_n) = P(X_1|\Pi_1)P(X_2|\Pi_2)\cdots P(X_n|\Pi_n).$$

Bayes net formalism (cont'd)

A DAG G represents the family of probability distributions that factor over G.

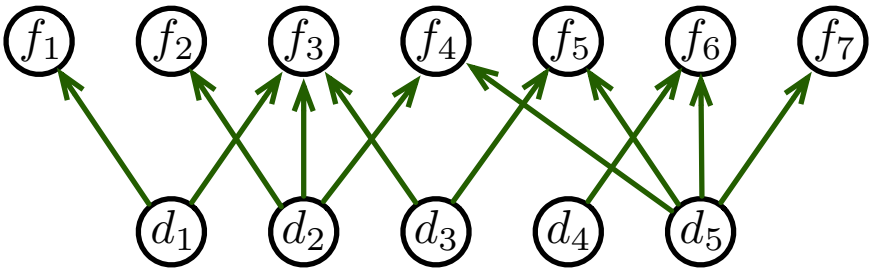
1. Is there a DAG that captures *all* probability distributions over the n variables?

2. If the X_i take on finitely many values, what additional information is needed to specify a probability distribution over G?

Example: QMT-DT network

Shwe et al (1991). *Probabilistic Diagnosis Using a Reformulation of the INTERNIST-1/QMR Knowledge Base.*

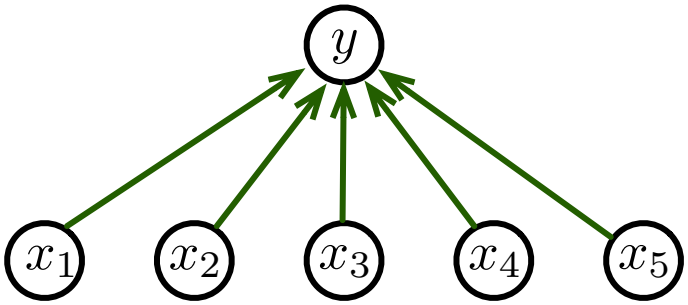
A Bayes net over more than 600 diseases and 4000 symptoms (findings).



$$\Pr(d, f) = \prod_i \Pr(d_i) \prod_j \Pr(f_j | \text{parents in } d)$$

Example (cont'd): Noisy-OR conditional probabilities

Generalization of $Y = X_1 \vee X_2 \vee X_3 \vee \cdots$



$$\Pr(Y = 0 | X_1, X_2, \ldots) = (1 - \lambda) \prod_i (1 - p_i)^{X_i}.$$

B: Inference in Bayes nets

Inference: three tasks

Suppose we have values for some of the nodes, $X_E = x_E$ ("evidence").

(1) Conditional probability query: for some $S \subset [n]$, what is the conditional distribution $\Pr(X_S | X_E = x_E)$?

- Often S consists of a single variable, e.g. $\Pr(\text{Flu} \mid \text{Fever}, \text{Headache})$
- Can be achieved by summing out the other variables:

$$\Pr(X_S = x_S, X_E = x_E) = \sum_{x_R} \Pr(X_S = x_S, X_R = x_R, X_E = x_E)$$

where $R = [n] \setminus (E \cup S)$. Why not do it this way?

Inference: three tasks

(2) Most probable explanation: Given evidence $X_E = x_E$, find

$$\text{MPE}(X_R | X_E = x_E) = \arg\max_{x_R} \Pr(X_R = x_R, X_E = x_E)$$

where $R = [n] \setminus E$.

- Speech recognition: If S_t is acoustic signal at time t and P_t is phoneme being uttered, find $\text{MPE}(\{P_t\} | \{S_t\})$.
- Another use: Fill in missing data.
- Naive time complexity like conditional probability query: sum becomes max.
- Note: Most likely value of X_R cannot be reconstructed from marginals of individual variables in X_R.

Inference: three tasks

(3) Maximum a posteriori: Given evidence $X_E = x_E$ and variables $S \subset [n] \setminus E$, find

$$\text{MAP}(X_S | X_E = x_E) = \arg\max_{x_S} \Pr(X_S = x_S | X_E = x_E).$$

- Given a few symptoms, what is the most likely assignment to the disease variables? (What would MPE do in this case?)
- Strict generalization of MPE. Both a sum and a max.

The bad news

All these inference tasks are NP-hard.

In fact, even the following problem is NP-hard:

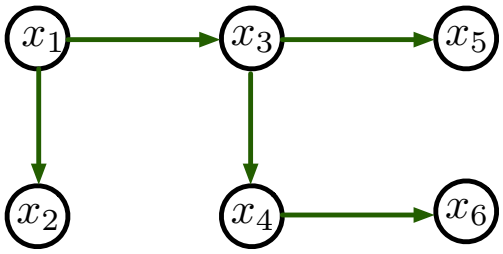
- Input: Bayes net over Boolean variables $X_1, \ldots, X_n$; variable X_i; value $v \in \{0, 1\}$
- Question: Is $\Pr(X_i = v) > 0$?

Do you see why?

Exact inference

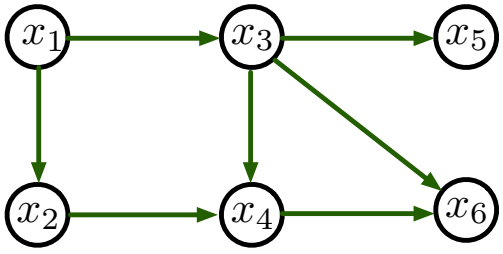
Suppose $X_1, \ldots, X_n$ take on finitely-many values and their joint distribution factors over a graph G.

(1) Exact inference is easy if G is a tree.
E.g. Using dynamic programming, which can also be done via a distributed, message-passing scheme.



Exact inference (cont'd)

(2) General G: exact inference can be done using the **junction tree algorithm** in time exponential in the **tree-width** of G.



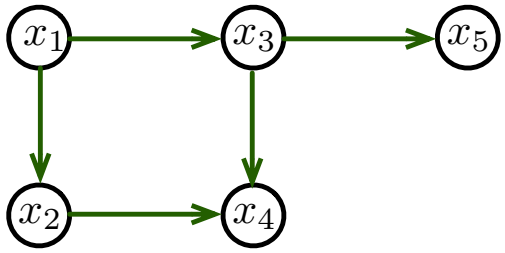
Approximate inference

- Sampling
- Belief propagation
- Variational methods

C: Learning directed graphical models

Fixed structure, fully observable

Maximum-likelihood parameters can be found efficiently.



Fixed structure, with hidden variables

Maximum-likelihood parameter estimation is typically NP-hard.

Example: mixtures of Gaussians.

Usual approach: EM algorithm.

Unknown structure

Finding the maximum-likelihood structure is NP-hard.

Well, there is one special case: *branchings* (trees in which each node has ≤ 1 parent), often called *Chow-Liu trees*.

Summary of basic properties

Directed graphical models (aka Bayes nets, belief nets):

- Can capture any probability distribution
- Intuitive factored form
- Popular for representing causal dependencies
- A pictorial language in which to specify families of probability models

Algorithmic status:

- Sampling: easy
- Computing likelihood/density: easy if fully-observable
- Inference: hard, but approximate methods are available
- Learning: many cases

Suitability for (1) density estimation (2) generation (3) representation learning?

D: Examples of neural autoregressive models

The autoregressive framework

To model d-dimensional data $X = (X_1, X_2, \ldots, X_d)$:

- Pick an ordering of the variables, $\sigma(1), \ldots, \sigma(d)$
- Explicitly learn the conditional probability functions

$$\Pr(x_{\sigma(i)} | x_{\sigma(1)}, \ldots, x_{\sigma(i-1)})$$

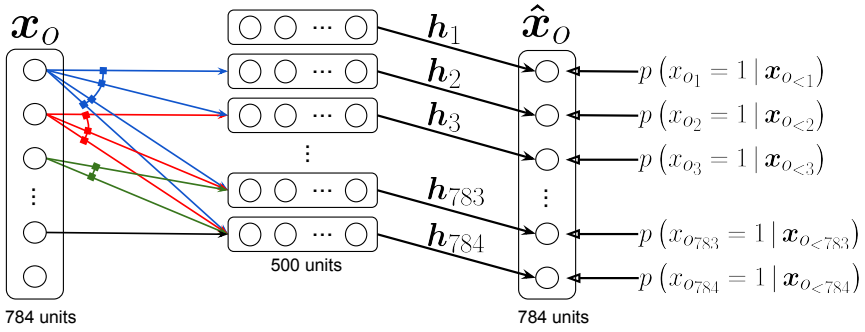
For instance, we can fit neural nets to these functions.

NADE (neural autoregressive density estimation)

Larochelle, Murray (2011)

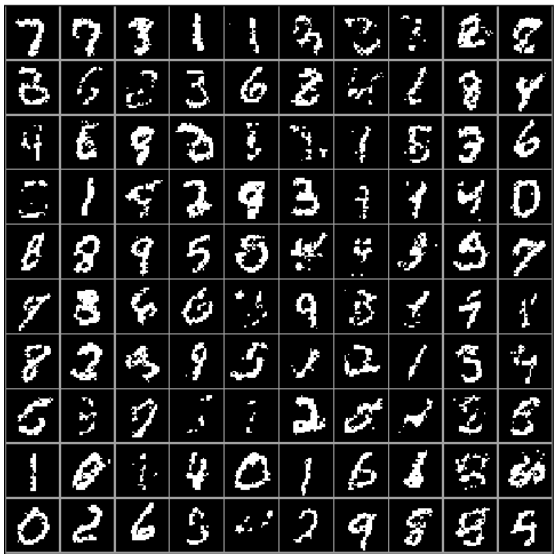
Binary variables:

- Each $\Pr(x_{\sigma(i)}|x_{\sigma(1)}, \ldots, x_{\sigma(i-1)})$ is a net with one hidden layer and sigmoid output.
- Tie weights to reduce the number of parameters.



Question: How to handle non-binary variables?

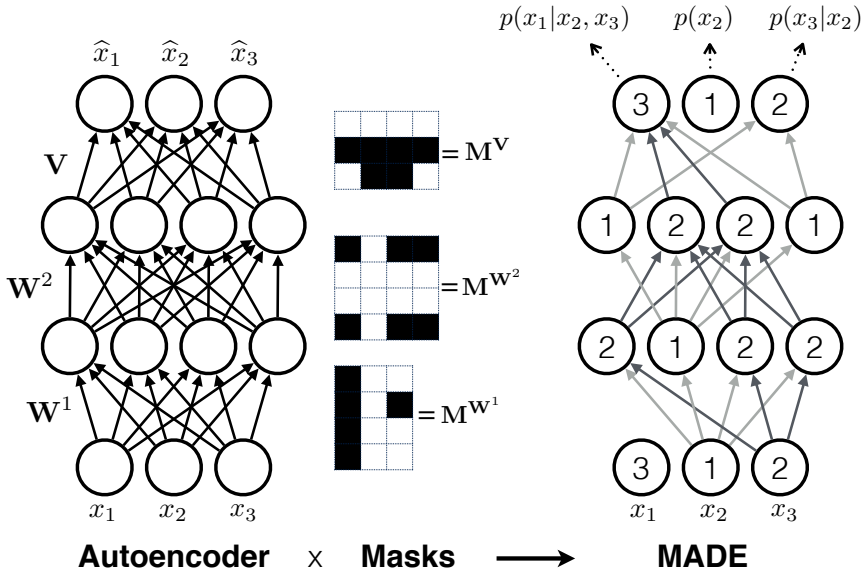
NADE on binarized MNIST



Question: How can one evaluate a generative model like this?

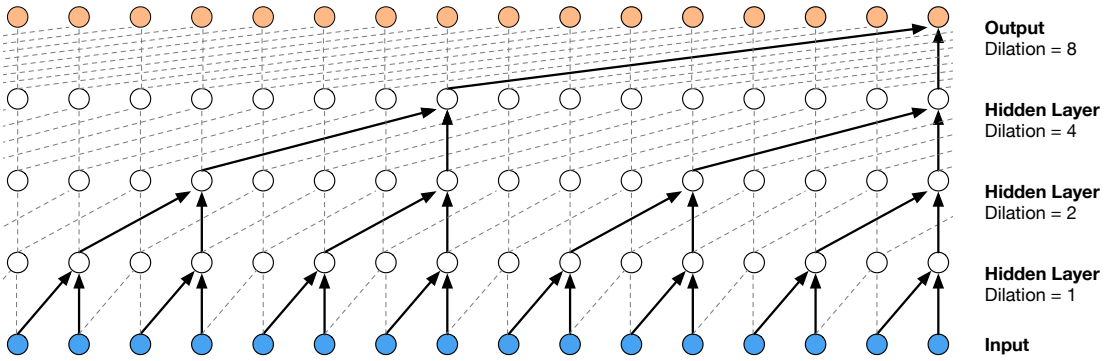
MADE (masked autoencoder for density estimation)

Germain, Gregor, Murray, Larochelle (2015)



WaveNet

Van den Oord, Dieleman, Zen, Simoyan, Vinyals, Graves, Kalchbrenner, Senior, Kavukcuoglu (2016).



- Output: Softmax over 256 values on a log-scale.
- Text-to-speech: conditional generation.