Belief Propagation

- Inference problems arising in

  - statistical physics
  - computer vision
  - error correcting codes / communication
  - energy systems
  - power grid
  - autonomous transport
  - robotics
  - biomedical systems
  - infrastructure systems
  - AI

  can be solved.

Principles behind belief propagation (BP) algorithm

- an efficient way to solve inference problems based on passing local messages.

- connection between BP algorithm and the Bethe approximation of statistical physics.

- BP → converges to a fixed point that is also a stationary point of the Bethe approximation to the free energy.

Inference and Graphical models

- Forward - backward algorithm
- Viterbi algorithm
- Iterative decoding algorithms for Gallager codes
- Pearl's belief propagation algorithm for Bayesian network
- Kalman filter
- Transfer - matrix approach

Special cases of BP algorithm.

- We look into graphical models and Bayesian networks

In contrast to Markov networks, which are undirected graphs, Bayesian networks are directed graphical models.

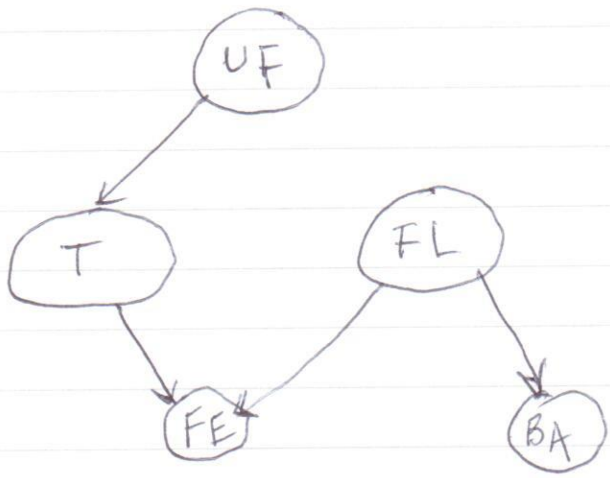
- Represents the joint distribution of a set of random variables.

Let us take a hypothetical medical diagnostics

Bayesian network as an example.

The following interdependencies can be inferred (conditional) from the example;

- Fever (FE) is independent of body ache (BA) given flu (FL) (common cause)
- Fever (FE) is independent of unhealthy food (UF) given typhoid (T) (indirect cause).



How to construct the graphical model from the statistical dependencies?

The dependencies are described as follows;

1. Unhealthy food (UF) increases the chance of typhoid (T).
2. Flu (FL) can lead to both fever (FE) and bodyache (BA).
3. The presence of either typhoid (T) or flu (F) can be manifested in form of fever (FE).

Given a BN, we can answer question such as

Which one is more probable, typhoid or flu, given fever and unhealthy food?

In addition, the networks need a set of local parameters which are conditional probabilities of each variable given its parent in the graph.

For example,

$P(FE|T, FL)$ is the conditional probability of fever given flu and typhoid.

Joint probability is represented in term of local parameters implying an important saving in the number of required parameters.

Probabilities (conditional) depend only on the parent nodes.

In our example,

$$p(\{x\}) \equiv p(x_{UF}, x_T, x_{FL}, x_{FE}, x_{BA})$$

$$= p(x_{UF}) \cdot p(x_T|x_{UF}) \cdot p(x_{FL}) \cdot p(x_{FE}|x_T, x_{FL}) \, p(x_{BA}|x_{FL})$$

Generally, in a acyclic graph of N random variables x_i

$$p(x_1, x_2, \cdots, x_N) = \prod_{i=1}^{N} p(x_i | Par(x_i))$$

where $Par(x_i)$ denotes the states of the parent node i.

If node i has no parents then we take $p(x_i|Par(x_i)) = p(x_i)$.

Our goal will be to compute certain marginal probabilities.

For example, the marginal probability for x_N,

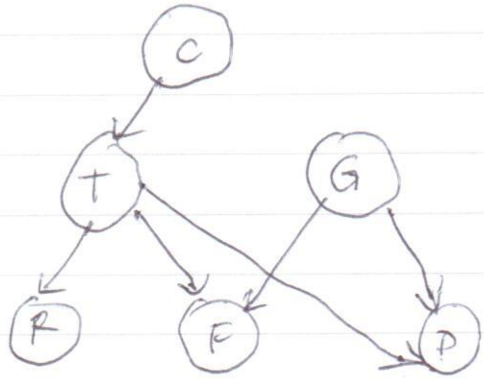
$$p(x_N) = \sum_{x_1} \sum_{x_2} \cdots \sum_{x_{N-1}} p(x_1, x_2, \ldots, x_N)$$

We refer to marginal probabilities that are computed approximately as 'beliefs' and denote the belief at node i by $b(x_i)$.

Representation;

BN represents the joint distribution of a set of n random variables as a acyclic graph and a set of conditional probability table (CPT).

Example of a simple BN



R is conditionally independent of C, G, F, D given T

i.e.

$$P(R \mid C, T, G, F, D) = P(R \mid T).$$

Parameters :

To complete specification in BN, we need to define its parameters.

- Root nodes : Vector of marginal probabilities
- Other nodes : CPT of the variable given its parents.

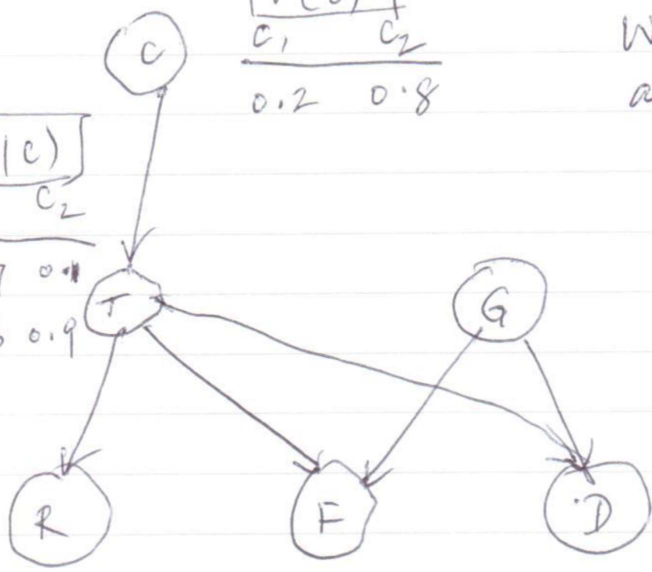
$\boxed{P(C)}$

C_1	C_2
0.2	0.8

CPTs of some variables. We assume variables are binary.

$\boxed{P(T \mid C)}$

	C_1	C_2
T_1	0.7	0.1
T_2	0.3	0.9



$\boxed{P(F \mid T, G)}$

	$T_1\,G_1$	$T_1\,G_2$	$T_2\,G_1$	$T_2\,G_2$
F_1	0.8	0.6	0.5	0.1
F_2	0.2	0.4	0.5	0.9

this can become difficult with no. of parameters in a CPT increasing exponentially with the increase in no. of parents in a node.

Two solution:

Canonical models } for CPTs
Graphical models

Canonical models

Noisy OR model:

When several variables or causes can produce an effect

- if one of them is TRUE
- and as more of the causes are TRUE, the probability of the effect increases.

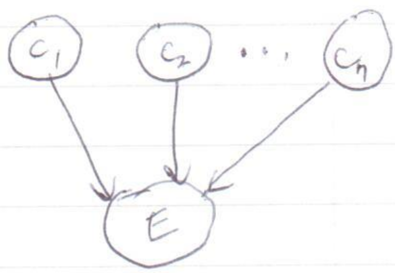
Two condition:

Responsibility: the effect is FALSE if all possible causes are FALSE

Independence of occurrence:

Probability that the effect E is inhibited

$$q_i = P(E = FALSE \mid C_i = TRUE)$$



$$P(E = FALSE \mid C_1 = TRUE, \cdots, C_m = TRUE) = \prod_{i=1}^{m} q_i$$

$$P(E = TRUE \mid C_1 = TRUE, \cdots, C_m = TRUE) = 1 - \prod_{i=1}^{m} q_i$$

CPT for Noisy OR with 3 parents and parameters
$q_1 = q_2 = q_3 = 0.1$.

C_1	0	0	0	0	1	1	1	1
C_2	0	0	1	1	0	0	1	1
C_3	0	1	0	1	0	1	0	1
$P(E=0)$	1	0.1	0.1	0.01	0.1	0.01	0.01	0.001
$P(E=1)$	0	0.9	0.9	0.99	0.9	0.99	0.99	0.999

TRUE = 1, FALSE = 0

other representation - decision diagram

Inference:

Single connected networks: Belief propagation (BP)

Pearl's tree propagation algorithm:

Based on Bayes rule:

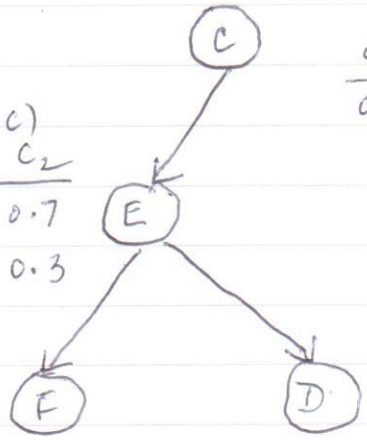
Given certain evidence E, the posteriori probability for a value i of a variable B

$$P(B_i | E) = P(B_i) P(E | B_i) / P(E)$$

Evidence is propagated using message passing mechanism.
Each node sends messages to sons and parent in the tree.

Example: BP algorithm

Consider the BN with 4 binary variables (each with T and F).



$P(C)$

C_1	C_2
0.8	0.2

$P(E|C)$

	C_1	C_2
E_1	0.9	0.7
E_2	0.1	0.3

$P(D|E)$

	E_1	E_2
D_1	0.7	0.4
D_2	0.3	0.6

$P(F|E)$

	E_1	E_2
F_1	0.9	0.5
F_2	0.1	0.5

Consider that the only evidence is F = FALSE.

Then, initial condition for the leaf nodes are:

$\lambda_F = [1, 0]$ and $\lambda_D = [1, 1]$ (no evidence).

Propagating to the parent node E

$$\lambda_F(E) = [1, 0] \begin{bmatrix} 0.9 & 0.5 \\ 0.1 & 0.5 \end{bmatrix} = [0.9, 0.5]$$

$$\lambda_D(E) = [1, 1] \begin{bmatrix} 0.7 & 0.4 \\ 0.3 & 0.6 \end{bmatrix} = [1, 1]$$

Then, $\lambda(E)$ is obtained by combining the messages from its two sons: $\lambda(E) = [0.9, 0.5] \times [1, 1] = [0.9, 0.5]$

Now, propagate to parent C:

$$\lambda_E(C) = [0.9, 0.5] \begin{bmatrix} 0.9 & 0.7 \\ 0.1 & 0.3 \end{bmatrix} = [0.86, 0.78]$$

In this case, $\lambda_E(C) = \lambda(C) = [0.86, 0.78]$ [$\because$ C has only one son]

We complete the bottom-up propagation.

We now proceed to the top-down message passing.

Given that C is not instantiated, $\pi(C) = [0.8, 0.2]$. We propagate to its son E.

$$\pi(E) = [0.8,\ 0.2] \begin{bmatrix} 0.9 & 0.7 \\ 0.1 & 0.3 \end{bmatrix} = [0.86,\ 0.14]$$

Now, we propagate to son D; however E has another son F, so we need to consider the λ message from F also. Thus,

$$\pi(D) = [0.86,\ 0.14] \times [0.9,\ 0.5] \begin{bmatrix} 0.7 & 0.4 \\ 0.3 & 0.6 \end{bmatrix} = [0.57,\ 0.27]$$

This completes the top-down propagation.

Given the λ and π vectors we find the posterior probabilities (following normalization):

$$P(C) = [0.80,\ 0.20] \times [0.86,\ 0.78] = [0.69,\ 0.16]\alpha_1 = [0.815,\ 0.185]$$
$$P(E) = [0.86,\ 0.14] \times [0.9,\ 0.5] = [0.77,\ 0.07]\alpha_2 = [0.917,\ 0.083]$$
$$P(D) = [0.57,\ 0.27] \times [1,\ 1] = [0.57,\ 0.27]\alpha_3 = [0.67,\ 0.33]$$

Remarks:

1. Time complexity ~ diameter of the network.
2. It can be extended to polytrees. If a node has multiple parents, λ messages should be sent to all parents. Time complexity is of the same order as ~~for~~ the tree structures.