Lecture 26 - Introduction to Belief Networks

https://www.cs.ubc.ca/~jordon/teaching/cpsc322/2019w2/lectures/lecture26.pdf

Goals:

- Build a Belief Network for a simple domain
- Classify the types of inference
- Compute the representational saving in terms on number of probabilities required

Belief Networkse

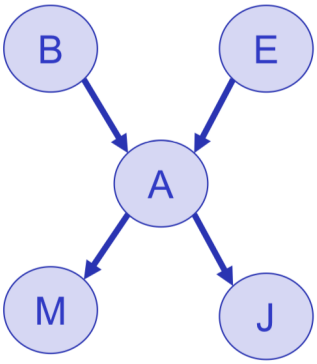
Burglary Example

Suppose we have the following scenario:

- There might be a **B**urglar in my house
- The **anti-burglar** **Alarm** in my house may go off
- I have an agreement with two of my neighbors, **John** and **Mary**, that they **call** me if they hear the alarm go off when I am at work
- **Minor** **Earthquakes** may occur and sometimes the set off the alarm.
- The **variables** are: **B, A, J, M, E**
- **Joint** has entries/probabilities

In belief networks we typically order the variables to reflect on the casual knowledge:

- A burglar (**B**) can set the alarm (**A**) off
- An earthquake (**E**) can set the alarm (**A**) off
- The alarm can cause Mary to call (**M**)
- The alarm can cause John to call (**J**)



We can then apply the chain rule to the joint distribution:

$$P(B, E, A, M, J) = P(B)P(E|B)P(A|B, E)P(M|A, B, E)P(J|M, A, E, B)$$

which lets us simlify based on the marginal and conditional independence

$$= P(B)P(E)P(A|B,E)P(M|A)P(J|A)$$

Consequently, we express the result as a network where:

- Each variable is a node.
- For each variable, the conditioning variables are its parents.
- We associate to each node its corresponding conditional probabilities.

This yields a directed acyclic graph (DAG).

In general

To define a belief network on a set of variables $\{X_1, \ldots, X_n\}$, first we need to select a definite ordering of the variables. Then we apply the chain rule, as discussed previously, where:

$$P(X_1 = v_1 \wedge \ldots \wedge X_n = v_n) = \prod_{i=1}^{n} P(X_i = v_i | X_1 = v_1 \wedge \ldots \wedge X_{i-1} = v_{i-1})$$

or in other terms:

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | X_1, \ldots, X_{i-1})$$

We subsequently define the parents of a random variable, parents(X_i) to be the minimal number of predecessors of X_i in the total ordering, such that the other predecessors of X_i are conditionally independent of X_i given parents(X_i). In other words, this means that X_i probabilistically depends on its parents, but is independent of its other predecessors. This means that: parents(X_i) $\subset \{X_1, \ldots, X_{i-1}\}$ such that:

$$P(X_i | X_1, \ldots, X_{i-1}) = P(X_i | \text{parents}(X_i))$$

There can be several numbers of *minimal* sets which may be defined as predecessors satisfying this condition, and any one of those may be chosen to be a parent. There can be more than one minimal set only when some of the predecessors are deterministic functions of others.

The chain rule a result is re-written as follows:

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | \text{parents}(X_i))$$

A belief network as a result defines a **factorization** of the joint probability distribution into a product of conditional probabilities. A belief network may be referred to as a ***Bayesian*** network, and it is a DAG where (as described above) the nodes are random variables, there is an arc from each of the elements of parents(X_i) to X_i.

We have that each node is independent from its *non-descendants* given its parents. A node is independent from the rest of the network given its *Markov blanket*, which in this case would be its set of parents.

A belief network has an associated set of conditional probability distributions that specify the probability of each variable given its parents (which includes the prior probabilities of those variables with no parents).

We also have that different ordering will result in different decompositions, as well as even fewer decompositions, and graphs which involve less arcs.

Compactness

A **Conditional probability table (CPT)** for **boolean** X_i with k **boolean** parents had 2^k rows for the combinations of the parent values. Each row requires **one number** p_i to represent X_i = true (the conjugate of which would be $1 - p_i$ to represent X_i = false).

If each variable has no more than k parents, then the complete network requires $O(n2^k)$ numbers to represent. If $k << n$, then this is a vast improvement from the previous space complexity of $O(2^n)$

Lecture 27 - Belief Networks continued

`https://www.cs.ubc.ca/~jordon/teaching/cpsc322/2019w2/lectures/lecture27.pdf`

Open Issues:

Following the previous section, we are still left with a number of issues that occur:

- **Independencies**: Does a BNet encode more independencies than the ones specified by construction?
  Yes
- **Compactness**: We reduce the number of probabilities from $O(2^n)$ to $O(n2^k)$.
  However in some domains we may need to do better than that.
- Still too many and often there are no data/experts for accurate assessment

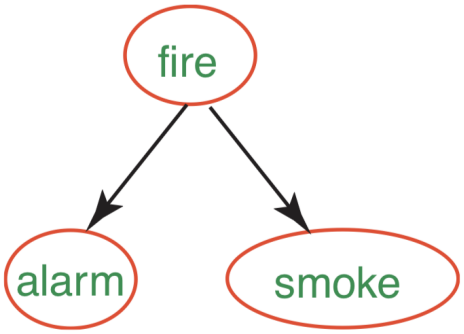
The **solution** is to make stronger (approximate) **independence assumptions**.

Conditional Independencies/Dependencies with common ancestors

In certain cases, we have that two variables' dependency will depend on the value observed of other variables.

Example 1

Consider the following example of a snapshot of a belief network:

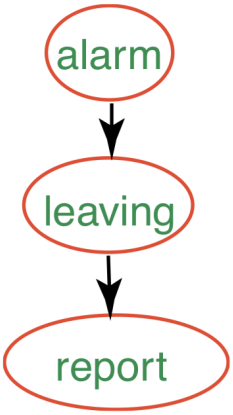


We have that:

- *Alarm* and *Smoke* are dependent, **however**
- *Alarm* and *Smoke* are independent given *Fire*
- Intuitively, *Alarm* can explain *Alarm* and *Smoke*; thus learning from one can affect the other by changing the belief in *Fire*.

Example 2

Another scenario can be pictured with the following example:

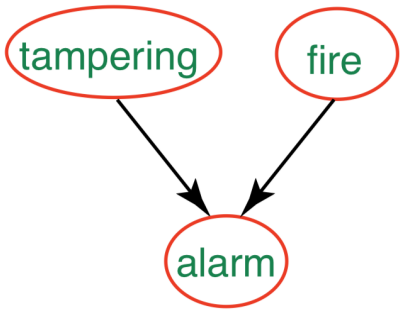


In this case we have that:

- *Alarm* and *Report* are dependent
- *Alarm* and *Report* are independent given *Leaving*
- Intuitively, the only way that the *Alarm* affects *Report* is by affecting *Leaving*.

Example 3

Lets take yet another example:



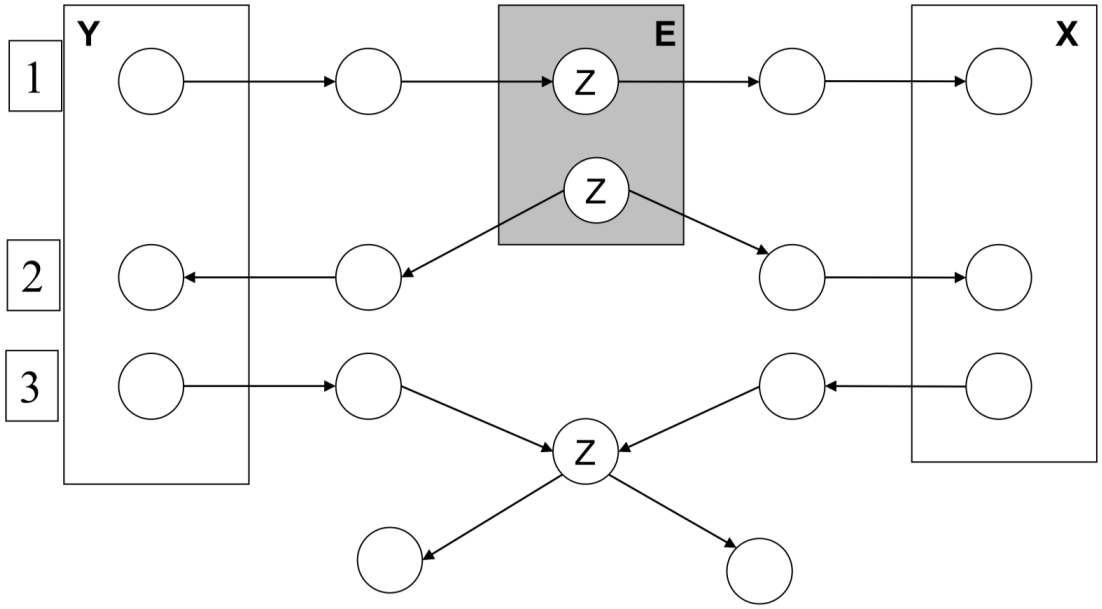
Here we have that;

- *Tampering* and *Fire* are independent
- *Tampering* and *Fire* are dependent given *Alarm*
- Intuitively, *Tampering* can explain away *Fire*

Understanding independence

In general there's three ways in which there can be a block in terms of probability propagation. Those are illustrated in the figure below, with 3 different paths, involving variables in subsets X, Y and Z:

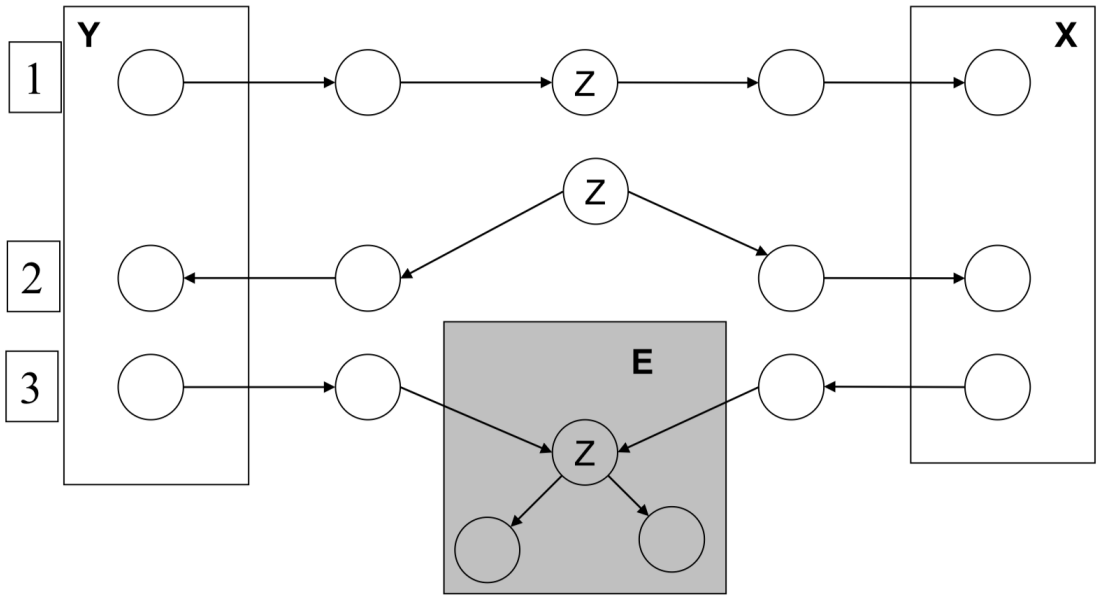
Conditional Independencies



- In paths 1 and 2, those paths become blocked as soon as we observe evidence **E**.
- In path 3, X and Y become dependent as soon as we get evidence for Z, or any of its descendants.
- Otherwise, X and Y remain independent if evidence is observed as shown above.

Conditional Dependencies

The following diagram represents cases where new evidence will cause variables to become dependent (i.e.: the opposite of the above example):



These are cases where X and Y are conditionally dependent on each other

Conditional Dependencies

In general we have the following rules:

- If we observe variable(s) $\bar{Y}$, the variables whose posterior probability is different from the prior are:
  - → The ancestors of $\bar{Y}$
  - → The descendants of the ancestors of $\bar{Y}$
- Intuitively, if we have a causal belief network:
  - → We do abduction to possible causes
  - → prediction from the causes

Compact Conditional Distributions

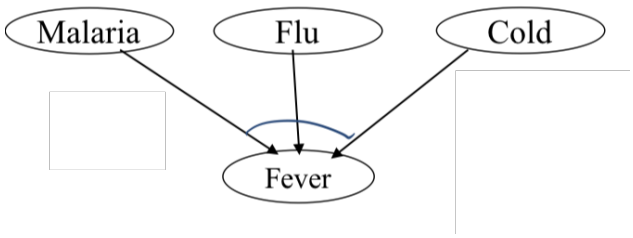
Once we have constructed the topology (structure) of a belief network, we still need to specify the conditional probabilities. This can be done using the **data** of using information from **experts**. To facilitate acquisition, we aim for compact representations for which data/experts can provide accurate assessments.

We observed that the Joint Probability table (JPT) has a size of $O(2^n)$, where n is the number of binary variables. We have that the Conditional Probability Table (CPT) has a size of $O(n2^k)$, where k is the maximum number of parents.

However this is still a problem, given the fact that the CPT size grows exponentially with the number of parents.

Effect with multiple non-interacting causes

Suppose we want to investigate the effect of multiple non-interacting causes. This is illustrated by the diagram below:



The green rown in the following table are the probabilities that experts would we able to easily provide:

Malaria	Flu	Cold	P(Fever=T | ..)	P(Fever=Fl..)
T	T	T		
T	T	F		
T	F	T		
T	F	F		
F	T	T		
F	T	F		
F	F	T		
F	F	F		

However it is more difficult to assess more complex conditioning.

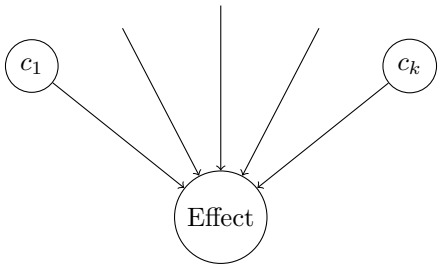
The solution is ***Noisy OR- Distributions*** This type of distribution:

- Models multiple non-interacting causes
- Uses logical OR with a probabilistic twist.
- A logic or conditional probability table is illustrated below:

Malaria	Flu	Cold	P(Fever=T | ..)	P(Fever=Fl..)
T	T	T	1	0
T	T	F	1	0
T	F	T	1	0
T	F	F	1	0
F	T	T	1	0
F	T	F	1	0
F	F	T	1	0
F	F	F	0	1

The noisy OR-model allows uncertainty in the ability of each cause to generate an effect (for example how one may have a cold without a fever). For this we have **two assumptions**:

- All possible causes are listed
- For each of the causes, whatever **inhibits** it to generate an effect is *independent* from the **inhibitors** of the other causes.



Suppose in the diagram above we have the the causes $c_1, \ldots, c_k$, which each have a probability of failure to trigger the effect of q_i, with $i \in \{1, \ldots, k\}$. Then we have that:

- $P(\text{Effect } = F | C_i = T, \text{and no other causes}) = q_i$
- $P(\text{Effect } = F | C_1 = T, \ldots, C_j = T, C_{j+1} = F, \ldots, C_k = F) = \prod_{i=1}^{j} q_i$
- $P(\text{Effect } = T | C_1 = T, \ldots, C_j = T, C_{j+1} = F, \ldots, C_k = F) = 1 - \prod_{i=1}^{j} q_i$

We go back to the disease example, where we wish to fit the table of probabilities, given only a subset of the probabilities, and using the assumptions. We are given that:

- $P(\text{Fever} = F | \text{Cold} = T, \text{Flu} = F, \text{Malaria} = F) = 0.6$
- $P(\text{Fever} = F | \text{Cold} = F, \text{Flu} = T, \text{Malaria} = F) = 0.2$
- $P(\text{Fever} = F | \text{Cold} = F, \text{Flu} = F, \text{Malaria} = T) = 0.1$

Then according to our above assumption that:

$$P(\text{Effect } = F | C_1 = T, \ldots, C_j = T, C_{j+1} = F, \ldots, C_k = F) = 1 - \prod_{i=1}^{j} q_i$$

We can fill in the table as follows:

Malaria	Flu	Cold	$P(\text{Fever} = T \mid ..)$	$P(\text{Fever} = F \mid ..)$
T	T	T		$0.6 \cdot 0.2 \cdot 0.1 = 0.012$
T	T	F		$0.2 \cdot 0.1 = 0.02$
T	F	T		$0.6 \cdot 0.1 = 0.06$
T	F	F	0.9	0.1
F	T	T		$0.2 \cdot 0.6 = 0.12$
F	T	F	0.8	0.2
F	F	T	0.4	0.6
F	F	F		1.0

In internal medicine using these kinds of assumptions can have us go from 133,931,430 probabilities to just 8,254

Naïve Bayesian Classifier

The Naïve Bayesian classifier is a very simple and successful Belief Network that allows us to classify entities into a set of classes C, given a set of attributes.

Email Spam filtering is a classic example of that:

- Determine whether an **email** is spam (we only have two classes $spam$ = T and $spam$ = F)
- We have that the useful attributes of the email are the **words** it contains

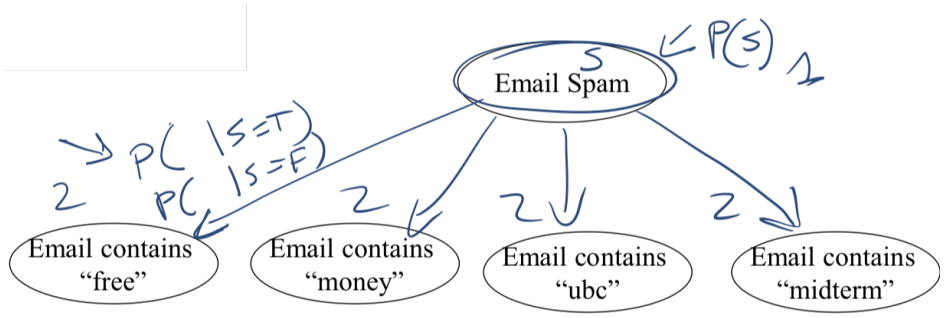
For this we make the following assumptions:

- The value of each attribute depends on the classification
- (Naïve) The attributes are independent of each other given the classification

If we have a large collection of emails, of which we know the labels (spam vs. not spam) then for each word in the English dictionary, we can compute the probability of the email containing that word, given that it's spam.

Bayes can then be used to predict on new emails, of which we do not know the label, if a email is spam or not, given the words it contains. The we classify that an email is spam if:

$$P(\text{spam} = T|\text{words it contains}) \geq P(\text{spam} = F|\text{words it contains})$$



With this type of parameters, we have that the joint probability distribution would have a space complexity around $O(2^{10^5})$ as opposed to $O(2*10^5)$ for the model described here (assuming that there are approximately 10^5 words in the English dictionary).

Summary of Compactness

In summary, we have discussed several methods where with n Boolean variables and at most k parents we can change the space complexity of our probabilities. Those are shown in the following figure: