# STA 772: Probabilistic Graphical Models (Exact Inference Part I)

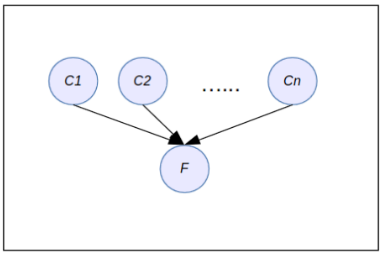
Professor O. E. Olubusoye

2023-11-23

# Basic Structures in PGM

**Causality:** When causes point to the fact in the graph

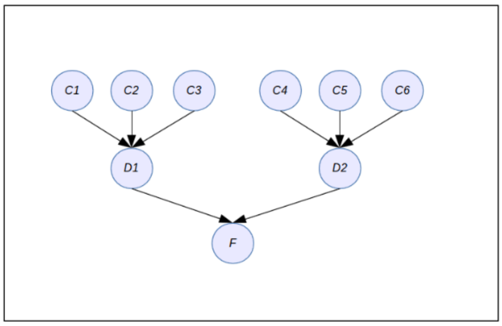
- Let $C_1$ to $C_n$ be causes and $F$ be a fact
- The probability distribution associated with this will be $P(F|C_1, C_2, \ldots C_n)$.




- The structure is very common
- If all the variables are binary, you need to represent a table with $2^{n+1}$.
- n=10 implies $2^{10+1} = 2,048$; n=31 implies $2^{31+1} = 4,294,967,296$
- It is inefficient if you have many causes for the same fact. It should be avoided except where n is small

# Basic Structures in PGM cont'd

### Hierarchy of Causes

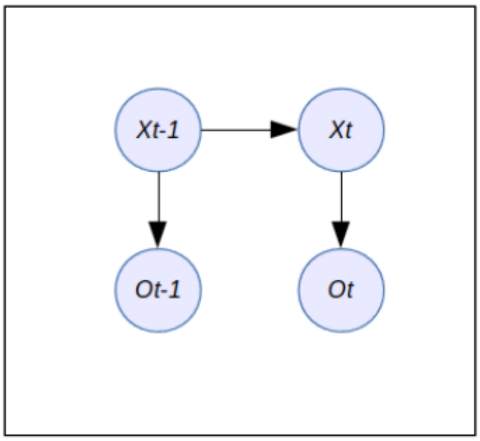



- Eight causes
- Local probability distribution such as $P(D_1|C_1, C_2, C_3)$ only involves four variables at most.

# Basic Structures in PGM cont'd

### Sequence of Variables in Time

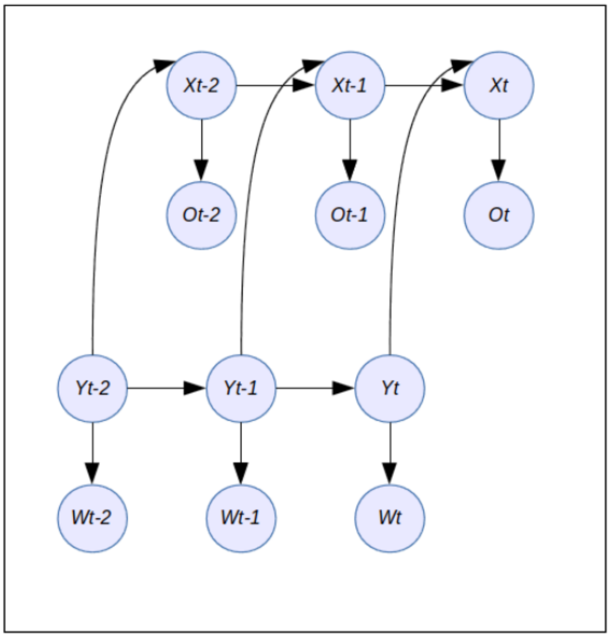
- Let's say we have a random variable representing the state of a system at time t, and let's say the state of the system predicts what the next state will be.
- Therefore, we can ask about the probability distribution of the current state of the system given the previous state $P(X_t|X_{t-1})$ where $t$ and $t-1$ denote the time




- This is a very common structure too

# Basic Structures in PGM cont'd

### Hidden Markov Model




- The graph combines two Hidden Markov models together within the same model.
- But one of the models, Y, is also considered as a cause of the state of the other model, X.
- We can write the joint probability distribution of the graph as

$$P(X) = P(Y_{t-2}).\,P(W_{t-2}|Y_{t-2})\ P(Y_{t-1}|Y_{t-2})P(W_{t-1}|Y_{t-1})\ P(Y_t|Y_{t-1}).\,P(W_t|Y_t)$$
$$P(X_{t-2}|Y_{t-2})P(O_{t-2}|X_{t-2})\ P(X_{t-1}|Y_{t-1},X_{t-2})P(O_{t-1}|X_{t-1})\ P(X_t|Y_t,X_{t-1})P(O_{t-1}|Y_{t-1}))$$

# Basic Structures in PGM cont'd

- The graph is called a **Hidden Markov Model** when the variables are discrete
- A **Hidden Markov Model** is a **Markov Model** in which the state is not directly observable (hidden).
- A **Markov Model** is a model whose current states depend only on the previous state of the system.
- When the variables follow a Gaussian distribution, it is called **Kalman Filter**

# Exact Inference

- Exact inference in probabilistic graphical models (PGMs) refers to the process of computing precise probabilistic answers to queries or probabilistic reasoning using the exact joint probability distribution (JPD) of the variables in the model.
- It involves exact calculations without approximation. Exact inference methods are typically used when the structure and size of the PGM allow for efficient computations.

**Key Characteristics of Exact Inference:**

1. **Deterministic**: Exact inference provides deterministic and precise results. It guarantees that the computed probabilities are the true probabilities based on the model's JPD.
2. **Complete Enumeration**: In some cases, exact inference involves complete enumeration of all possible combinations of variable values. This approach is feasible for small to moderately sized models.

# Exact Inference cont'd

3. **Applicability**: Exact inference is most commonly applied to models with a relatively small number of variables, acyclic Bayesian networks, and other tractable structures.

4. **Examples**: Variable Elimination, Exact Inference on Trees, Junction Tree Algorithm, Exact Inference in Discrete PGMs.

# Approximate Inference

- Approximate inference, on the other hand, involves methods and techniques for estimating probabilistic quantities when exact inference is computationally infeasible or when dealing with large, complex, or continuous models.
- In approximate inference, probabilistic answers are approximated rather than computed exactly.

**Key Characteristics of Approximate Inference:**

1. **Stochastic**: Approximate inference often provides stochastic or sampled results. Instead of exact probabilities, it offers estimates or samples from the true distribution.
2. **Efficiency**: Approximate inference methods are designed to handle large and complex models efficiently. They trade off some degree of accuracy for computational speed.

# Approximate Inference cont'd

3. **Applicability**: Approximate inference is particularly useful when exact inference is impractical due to computational limitations, as is often the case in real-world applications.

4. **Examples**: Markov Chain Monte Carlo (MCMC), Variational Inference (VI), Gibbs Sampling, Particle Filtering.

# Difference Between Exact Inference and Approxiamate Inference

1. **Precision vs. Approximation**: Exact inference provides precise, deterministic results that are guaranteed to be the exact probabilities based on the model. Approximate inference, on the other hand, provides approximations or samples that may deviate from the true probabilities but are computationally feasible.

2. **Computational Complexity**: Exact inference methods can become computationally infeasible for large or complex models due to the exponential growth in computational requirements. Approximate inference methods are designed to handle such scenarios efficiently.

3. **Stochastic vs. Deterministic**: Approximate inference often involves stochastic elements, such as sampling, which introduces randomness into the results. Exact inference is deterministic and produces consistent outcomes.

# Difference Between Exact Inference and Approxiamate Inference cont'd

4. **Model Size and Structure**: Exact inference is more suitable for smaller models with clear and tractable structures, while approximate inference is adaptable to larger and more complex models.

- In practice, the choice between exact and approximate inference depends on
  1. the nature of the problem
  2. the size and structure of the PGM, and
  3. the available computational resources.
- Researchers and practitioners often use a combination of both approaches to balance accuracy and efficiency in probabilistic modeling and reasoning.

# Variable Elimination Algorithm

### Variable Elimination

- is a powerful algorithm for exact inference in probabilistic graphical models (PGMs), particularly in Bayesian networks (BNs).
- It is used to efficiently compute marginal probabilities or conditional probabilities by systematically eliminating variables from the joint probability distribution (JPD) based on the query and the model's structure.

### Key Steps in Variable Elimination:

1. **Initialization**:
    - *Start with the original Bayesian network and the query you want to answer.*
    - *Initialize a list of factors (probability distributions) associated with each variable in the network.*

# Variable Elimination Algorithm cont'd

2. **Variable Ordering**:

    - *Choose an order in which to eliminate variables. The choice of variable ordering can greatly affect the efficiency of the algorithm.*

3. **Variable Elimination Loop**:

    - *For each variable in the chosen order:*
        - Multiply all factors that involve the current variable.
        - Sum out (marginalize) the current variable from the result.
        - Store the resulting factor.
        - Remove the factors involving the eliminated variable.

4. **Final Result**:

    - *After eliminating all variables according to the chosen order, you will be left with a single factor representing the answer to the query.*

# Example 1: BN for Medical Diagnosis

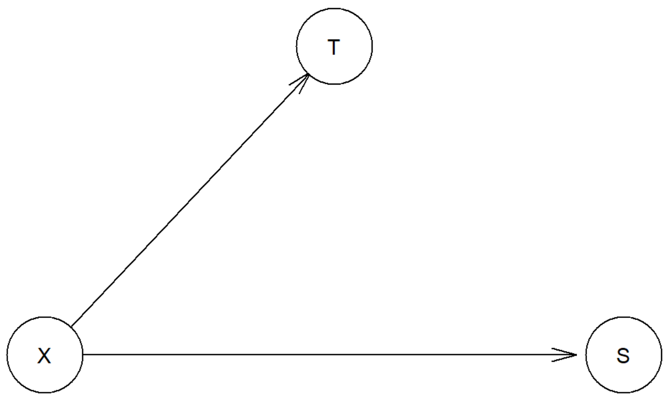
Consider a Bayesian network for medical diagnosis with random variables:

- $X$: Patient has the disease.
- $S$: Patient has a symptom.
- $T$: Patient is tested for the disease.

```r
# Install and load the bnlearn package if not already installed
# install.packages("bnlearn")
library(bnlearn)
```

```
## Warning: package 'bnlearn' was built under R version 4.2.3
```

```r
# Define the Bayesian network structure
medical_network <- empty.graph(c("X", "S", "T"))
medical_network <- set.arc(medical_network, "X", "S")
medical_network <- set.arc(medical_network, "X", "T")
# Plot the Bayesian network
plot(medical_network)
```



# Example 1: BN for Medical Diagnosis cont'd

Suppose we want to compute $P(X = \text{True}|T = \text{Positive})$, the probability of the patient having the disease given a positive test result.

**Variable Elimination Steps:**

1. Initialization: Initialize factors based on conditional probability distributions.
2. Variable Ordering: Choose an ordering, e.g., $S, T, X$.
3. Variable Elimination Loop:
   - *Multiply* $P(S)$ *and* $P(T|S)$ *to get* $P(S, T)$.
   - *Sum out* $S$ *to obtain* $P(T)$.
   - *Multiply* $P(X)$ *and* $P(T|X)$ *to get* $P(X, T)$.
   - *Sum out* $T$ *to obtain* $P(X)$.
4. Final Result: $P(X = \text{True}|T = \text{Positive})$ is computed and represents the probability of having the disease given a positive test result.

# Example 1: BN for Medical Diagnosis cont'd

Let's assume the following conditional probability distributions (CPDs):

1. $P(X = \text{True}) = 0.01$ (1% of the population has the disease).
2. $P(S = \text{True}|X = \text{True}) = 0.9$ (90% of those with the disease have the symptom).
3. $P(S = \text{True}|X = \text{False}) = 0.2$ (20% of those without the disease have the symptom).
4. $P(T = \text{Positive}|X = \text{True}) = 0.95$ (95% of those with the disease test positive).
5. $P(T = \text{Positive}|X = \text{False}) = 0.1$ (10% of those without the disease test positive).

Now, let's calculate $P(X = \text{True}|T = \text{Positive})$, the probability of having the disease given a positive test result using Variable Elimination:

# Example 1: BN for Medical Diagnosis cont'd

1. Initialization: Initialize factors based on the CPDs.
2. Variable Ordering: Choose an ordering, e.g., $S, T, X$.
3. Variable Elimination Loop:

*1. Multiply $P(S)$ and $P(T|S)$ to get $P(S, T)$:*

- $P(S) = P(S = \text{True}|X = \text{True}) \cdot P(X = \text{True}) + P(S = \text{True}|X = \text{False}) \cdot P(X = \text{False}) = 0.9 \cdot 0.($
- $P(T|S)$ *depends on the presence or absence of the symptom.*

*2. Sum out $S$ to obtain $P(T)$:*

- $P(T) = \sum_S P(S, T)$

*3. Multiply $P(X)$ and $P(T|X)$ to get $P(X, T)$:*

- $P(X) = 0.01$ *(given).*
- $P(T|X)$ *depends on the presence or absence of the disease.*

*4. Sum out $T$ to obtain $P(X)$:*

- $P(X|T = \text{Positive}) = \frac{P(X,T)}{P(T)}$

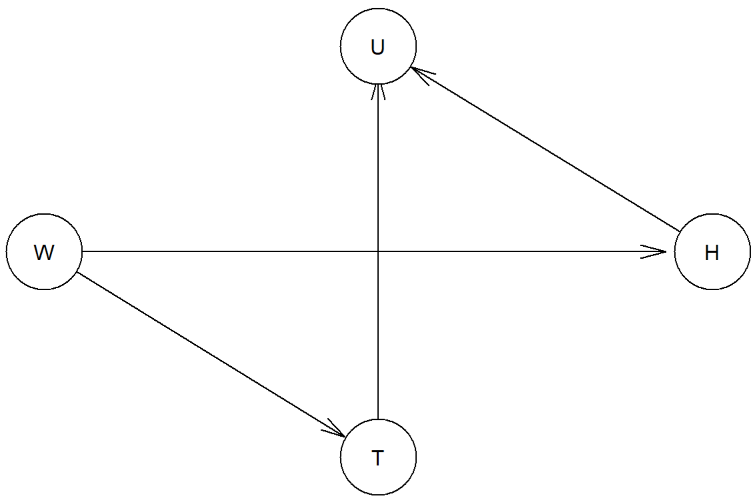
Calculating the values will provide the conditional probability $P(X = \text{True}|T = \text{Positive}.$

# Example 2: BN for Weather Prediction

Consider a Bayesian network for weather prediction with random variables:

- $W$:Weather conditions (e.g., Rain, Sun).
- $T$:Temperature (e.g., Hot, Cold).
- $H$: Humidity (e.g., High, Low).
- $U$: Umbrella status (e.g., Yes, No).

```
# Install and load the bnlearn package if not already installed
# install.packages("bnlearn")
library(bnlearn)
# Define the Bayesian network structure
weather_network <- empty.graph(c("W", "T", "H", "U"))
weather_network <- set.arc(weather_network, "W", "T")
weather_network <- set.arc(weather_network, "W", "H")
weather_network <- set.arc(weather_network, "T", "U")
weather_network <- set.arc(weather_network, "H", "U")
# Plot the Bayesian network
plot(weather_network)
```



# Example 2: BN for Weather Prediction cont'd

Suppose we want to compute $P(U = \mathrm{Yes}|W = \mathrm{Rain})$, the probability of carrying an umbrella given rainy weather.

**Variable Elimination Steps:**

1. Initialization: Initialize factors based on conditional probability distributions.
2. Variable Ordering: Choose an ordering, e.g., $T, H, U$.
3. Variable Elimination Loop:
    - *Multiply $P(T)$, $P(H)$, and $P(U|T, H)$ to get $P(T, H, U)$.*
    - *Sum out $T$ and $H$ to obtain $P(U)$.*
4. Final Result: $P(U = \mathrm{Yes}|W = \mathrm{Rain})$ is computed and represents the probability of carrying an umbrella given rainy weather.

# Example 2: BN for Weather Prediction cont'd

In this graph: - $W$ represents weather conditions. - $T$ represents temperature. - $H$ represents humidity. - $U$ represents the decision to carry an umbrella.

Let's assume the following CPDs:

1. $P(W = \text{Rain}) = 0.3$ (30% chance of rainy weather).
2. $P(W = \text{Sunny}) = 0.7$ (70% chance of sunny weather).
3. $P(T = \text{Hot}|W = \text{Sunny}) = 0.8$ (80% chance of hot weather when sunny).
4. $P(T = \text{Cold}|W = \text{Sunny}) = 0.2$ (20% chance of cold weather when sunny).
5. $P(T = \text{Hot}|W = \text{Rain}) = 0.4$ (40% chance of hot weather when rainy).
6. $P(T = \text{Cold}|W = \text{Rain}) = 0.6$ (60% chance of cold weather when rainy).
7. $P(H = \text{High}|W = \text{Sunny}) = 0.2$ (20% chance of high humidity when sunny).
8. $P(H = \text{Low}|W = \text{Sunny}) = 0.8$ (80% chance of low humidity when sunny).
9. $P(H = \text{High}|W = \text{Rain}) = 0.7$ (70% chance of high humidity when rainy).
10. $P(H = \text{Low}|W = \text{Rain}) = 0.3$ (30% chance of low humidity when rainy).
11. $P(U = \text{Yes}|T, H) = 0.9$ (90% chance of carrying an umbrella if it's cold and humid).
12. $P(U = \text{Yes}|T, H) = 0.2$ (20% chance of carrying an umbrella otherwise).

Now, calculate $P(U = \text{Yes}|W = \text{Rain})$, the probability of carrying an umbrella given rainy weather using Variable Elimination.