CSCI E-82a
Probabilistic Programming and AI
Lecture 1
Directed Graphical Models

Steve Elston

Why Directed Graphical Models?

How can we make inferences in an uncertain environment?

- Example; **sensor fusion** for a self-driving car
- Example; Locate a robot while mapping environment
- Example; make an optimal decision given **uncertain information**
- Must compute the **posterior of multivariate probability distribution**:

  $p(x_1, x_2, \dots x_n)$
- Or, compute **maximum a posteriori** (MAP)
- Process of **inference**

Why Directed Graphical Models?

Are directed graphical models interpretable?

- Allow experts to include **prior information**!
- Provide a language to describe the relationship between variables
- Can easily query **conditional probability tables** (CPTs)
- But, can non-experts really understand these models?
  - Making the model results transparent to non-experts is a research area

Why Directed Graphical Models?

We want the **posterior distribution** of one or more variables

- How to compute this **query** efficiently?
- Direct tabular representation **$O(n^3)$ complexity**
- Can do (much) better **factoring distribution by conditional independencies**
  - Discuss complexity of inference later
- **Directed graphical model is a representation of a distribution factored by conditional independencies**

Why Directed Graphical Models?

Directed graphical models are generative

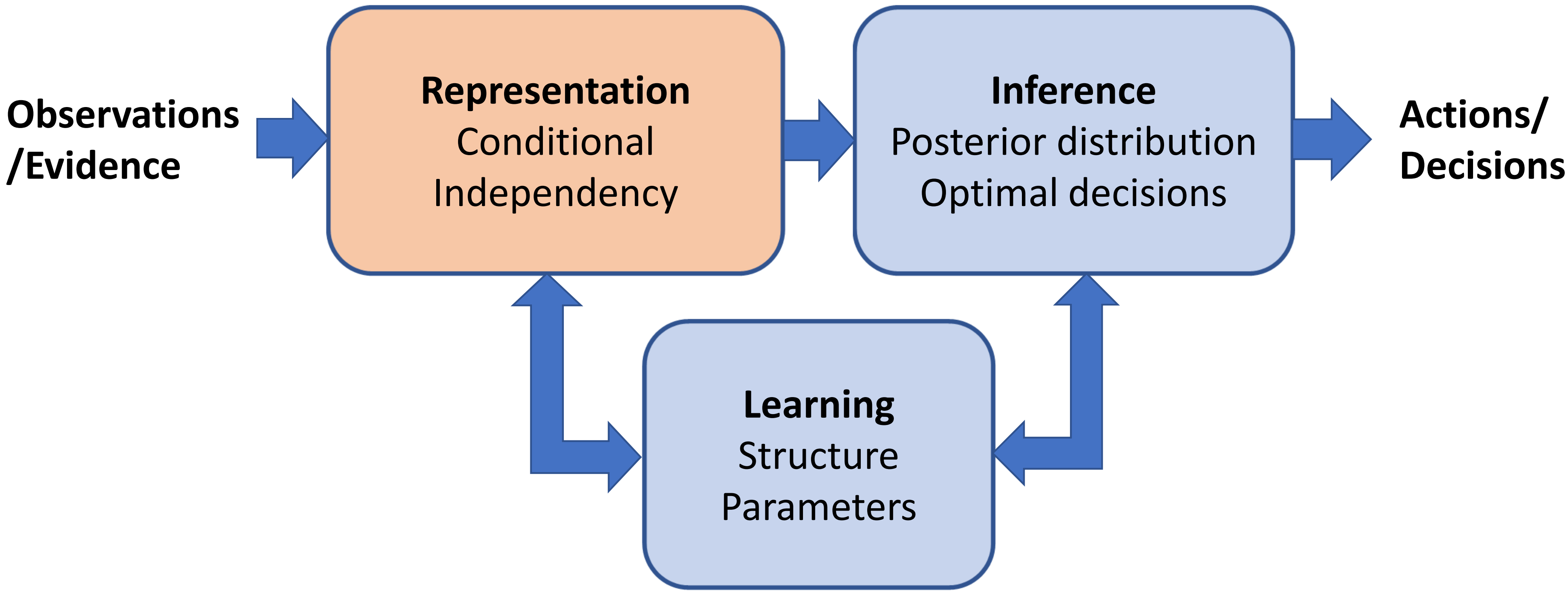
- Define probability distribution with DAG
- DAG is expressive in terms of independency structure
- Generate realizations from that distribution

Why Directed Graphical Models?

How much data do we need for learning?

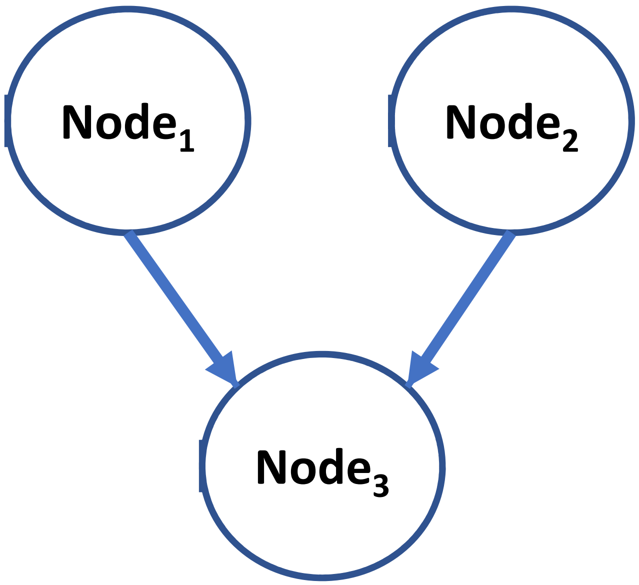
- Learning is both hard and easy
- Model parameters (CPTs) improve with data
    - Does not take very much data
    - Can use prior or expert information
    - Not all variables need to be observable
    - Can deal with missing values
- Learning model structure is hard!
    - Requires massive data
    - Or, expert input
- Discuss learning in another lesson

Focus on representation with graphical models



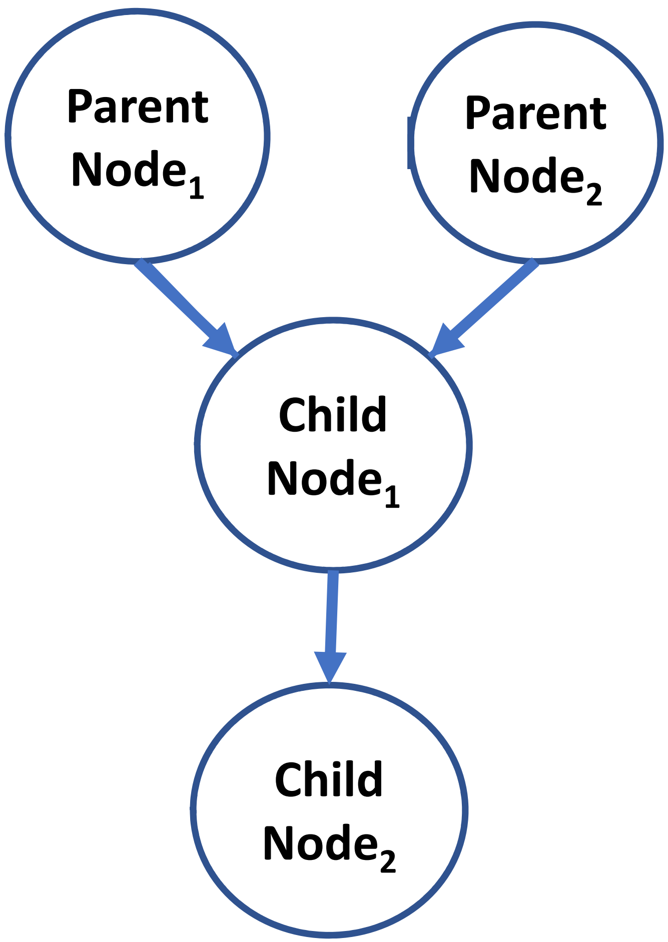
Schematic of intelligent agent using directed graphical model

Some Graph Terminology: Nodes and edges



- **Nodes or vertices** contain **conditional probability distributions (CPD)**
- **Undirected edges** define the connectivity between the nodes or the **skeleton of the graph**
- Nodes connected by edges are **neighbors**
- Information or **influence** flows along **directed edges**

Some Graph Terminology: Relationships



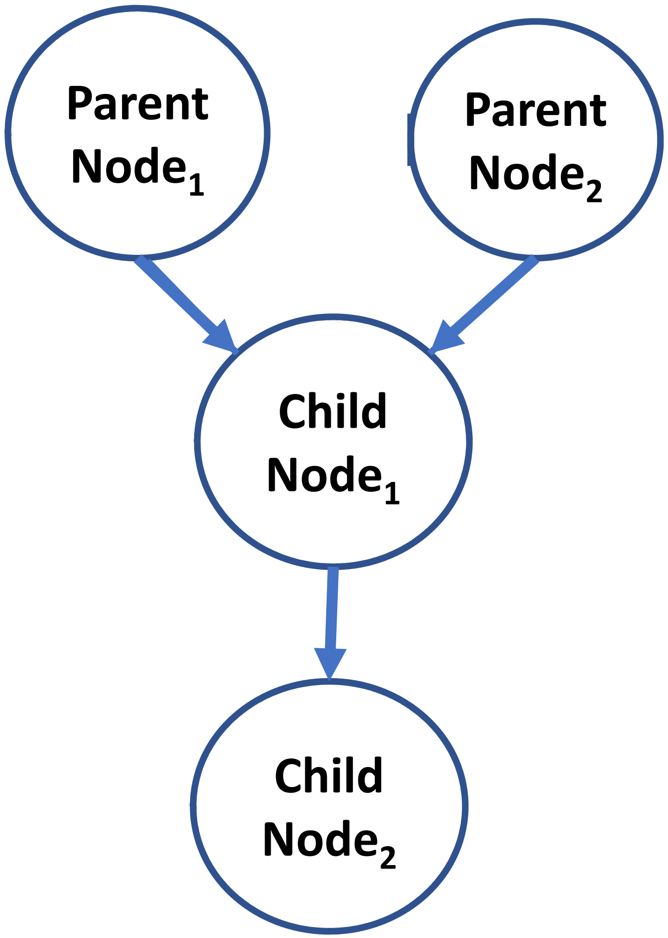
- **Parent nodes** have directed edges to, and **influence**, child nodes
- **Child nodes** are descendants of parents
- **Ancestors** are parents, grand parents, etc.
- Example; express ancestors of child node 2:

  $\mathbf{PA(CN_2) = \{PN_1, PN_2, CN_1\}}$

- **Decedents** are nodes receiving influence
- Example; express decedents of parent node 1:

  $\mathbf{DE(PN_1) = \{CN_1, CN_2\}}$

Some Graph Terminology: Degree of nodes

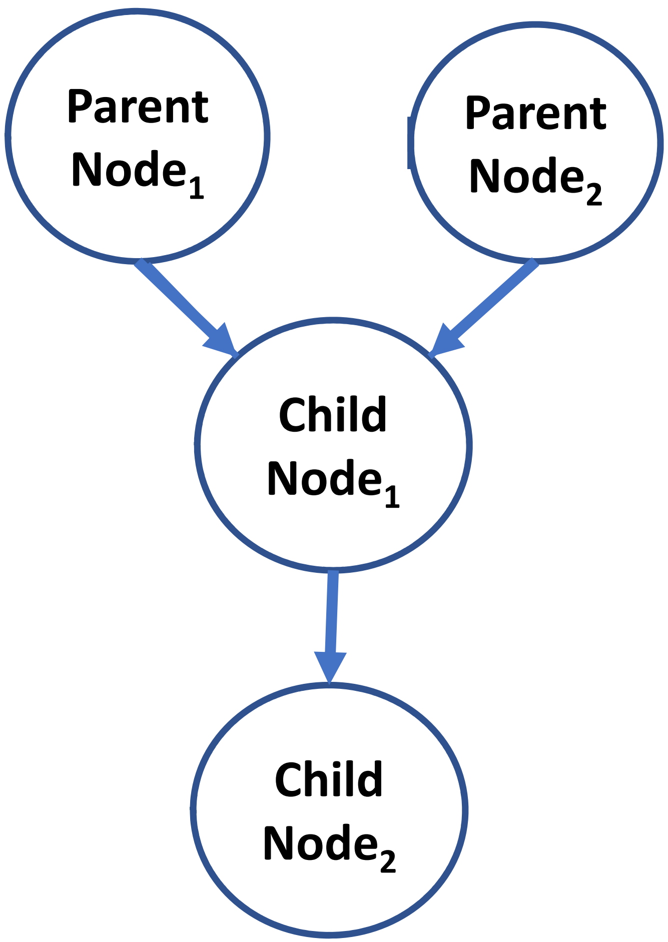


- **Degree** of a node is the **number of neighbors**
- Example **in degree** of a child node 1 can be expressed:

  IN(CN$_1$) = 2
- Example **out degree** of a child node 1 can be expressed:

  OUT(CN$_1$) = 1

Some Graph Terminology: Special nodes

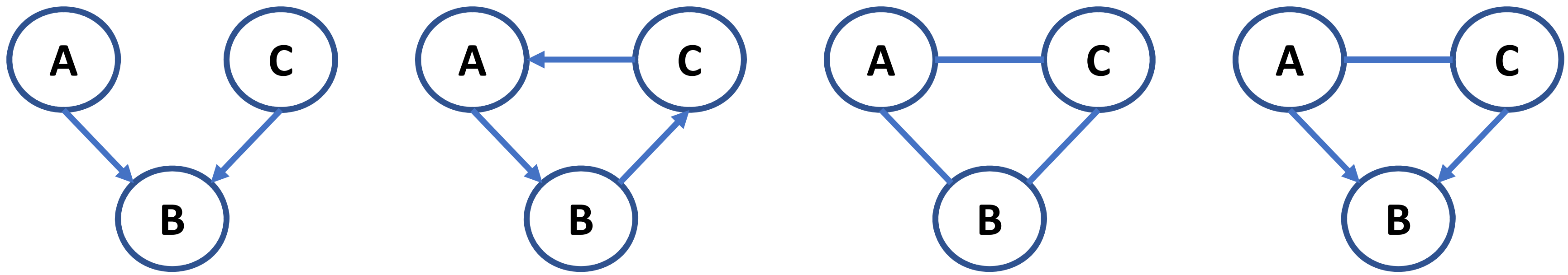


- A **root node** has **no ancestors**

  Root nodes = {PN$_1$, PN$_2$}
- A **Leaf node** has **no children**

  Leaf node = {CN$_2$}

Some Graph Terminology: Types of graphs



Directed Acyclic Graph | **Directed Cyclic Graph** | **Undirected Graph** | **Partially Directed Graph**

Independencies in Directed Graphical Models

We need to model the **independency structure of a multivariate probability distribution**

$$p(x_1, x_2, \ldots x_n)$$

- We use the notation **I(p)** to represent the independency structure of the distribution
- We use the notation **I(G)** to represent the independency structure of the graphical model
- Is it possible to have:

$$\mathbf{I(p) = I(G)}$$

- **In general no,** but we can still get a useful model!

Independencies in Directed Graphical Models

Factorizing a distribution greatly reduces computational complexity

- A **bivariate distribution can be factored** as a conditional probability distribution and an unconditional probability distribution:

$$P(A, B) = P(A|B)P(B) = P(B|A)P(A)$$

- Notice that the **factorization is not unique**

Independencies in Directed Graphical Models

Factorizing a distribution greatly reduces computational complexity

- The **chain rule of probability** is the key to factoring a multivariate distribution
- First, a multivariate distribution can be factored:

$$P(A_1, A_2, A_3, A_4 \ldots, A_n) = P(A_1|A_2, A_3, A_4, \ldots, A_n)\ P(A_2, A_3, A_4 \ldots, A_n)$$

- Continue factoring on the right hand side eventually yields:

$$= P(A_1|A_2, A_3, A_4, \ldots, A_n)\ P(A_2|A_3, A_4 \ldots, A_n)\ P(A_3|A_4 \ldots, A_n) \ldots P(A_n)$$

Independencies in Directed Graphical Models

Factorizing a distribution greatly reduces computational complexity

- Factorization using the **chain rule of probability is not unique**
- For example, the multivariate distribution can be factored:

$$
\begin{aligned}
&P(A_1, A_2, A_3, A_4 \ldots, A_n) \\
&= P(A_n|A_{n-1}, A_{n-2}, A_{n-3}, \ldots, A_1)\ P(A_{n-1}|A_{n-2}, A_{n-3}, \ldots, A_1)\ P(A_{n-2}|A_{n-3}, \ldots, A_1) \\
&\qquad\qquad\qquad \ldots p(A_1)
\end{aligned}
$$

Independencies in Directed Graphical Models

Factorizing a distribution greatly reduces computational complexity

- Factorization can be performed in any other order, which can be expressed generally as the product of conditional distributions:

$$P\left(\bigcap_{k=1}^{n} A_k\right) = \prod_{k=1}^{n} p\left(A_k \,\middle|\, \bigcap_{j=1}^{k-1} A_j\right)$$

- Unfortunately, finding the **best factorization is an NP complete problem**

Independencies in Directed Graphical Models

Factorizing a distribution greatly reduces computational complexity

- For a directed graph the **choice of parents determines the semantics**
- The factorization of the entire distribution are defined by **global semantics** of the DAG
- The global semantics are expressed:

$$P(X) = \prod_{i=1:d} P(X_i | \{parents(X_i)\})$$

Distribution Factorization Example

Example of Factorizing a distribution for probabilities that a student getting a job based on GRE score and letter of recommendation

- A student is seeking a job as a machine learning engineer
- The employer will make a hiring decision based on her GRE score and the quality of a letter of recommendation from her machine learning course professor
- The student’s GRE score is only dependent on her intelligence
- The student’s grade in the machine learning course is dependent on both her intelligence and the difficulty of the course
- Unfortunately, the professor is absent minded and will base her letter only on the student’s grade

Distribution Factorization Example

Example of Factorizing a distribution for probabilities that a student getting a job based on GRE score and letter of recommendation

- The joint discrete distribution is: P(D, I, S, G, L)

  D = {0,1}, difficulty of the student's machine learning course

  I = {0,1}, intelligence of the student

  S = {0,1}, student's GRE score

  G = {0,1,2}, student's grade in the machine learning course

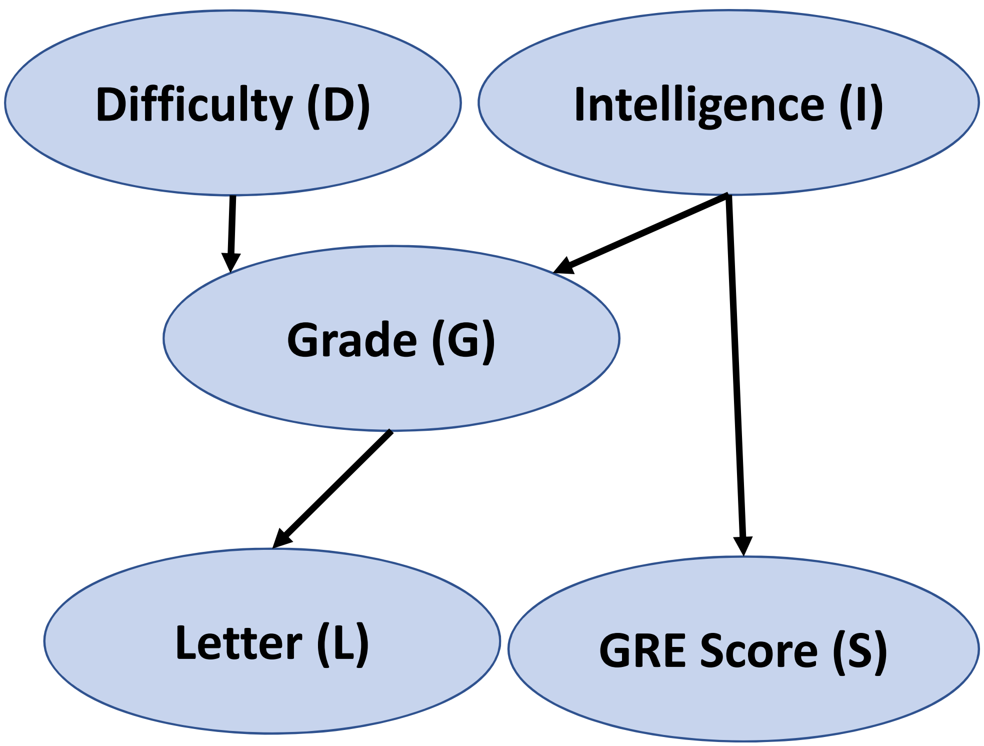
  L = {0,1}, quality of machine learning professor's recommendation letter

- The full table of the join discrete distribution is comparatively large:

  2 x 2 x 2 x 3 x 2 = 48 cases

Distribution Factorization Example

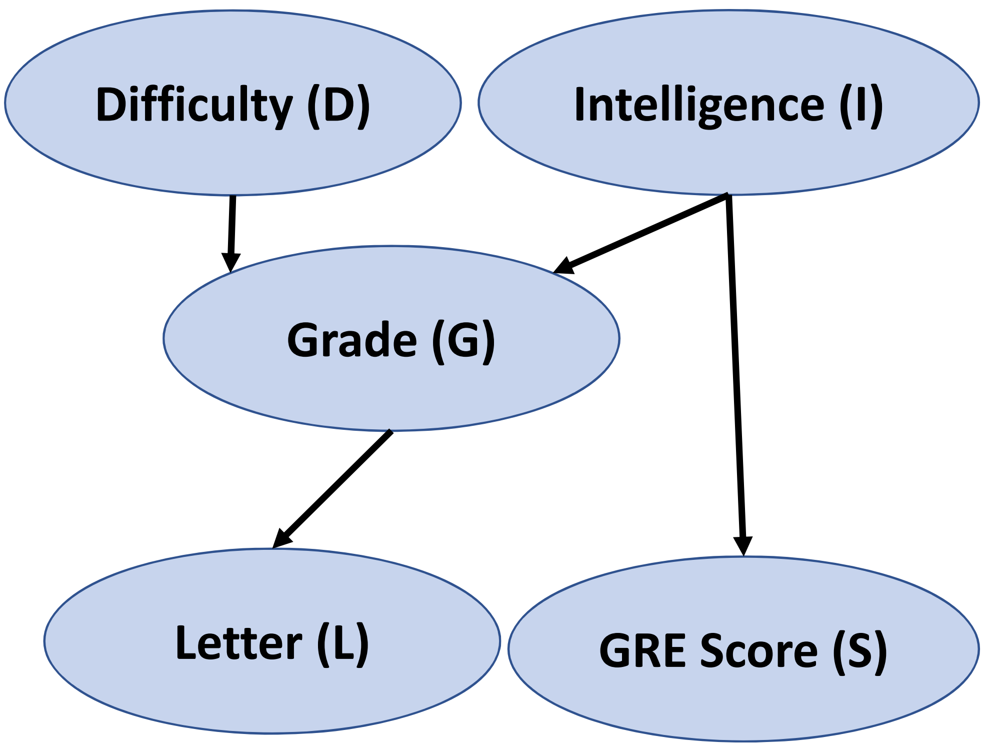
Use **conditional probability tables (CPTs)** to factor distribution



- Difficulty, D, and Intelligence, I, are independent of all other variables
- Grade, G, depends on Difficulty and Intelligence
- GRE Score, S, depends only on intelligence
- Quality of Letter, L, depends only on the grade

Distribution Factorization Example

Use **conditional probability tables (CPTs)** to factor distribution



The **semantics** of the graphical model are:

$$D \perp I, S, G, L$$

$$I \perp D, S, G, L$$

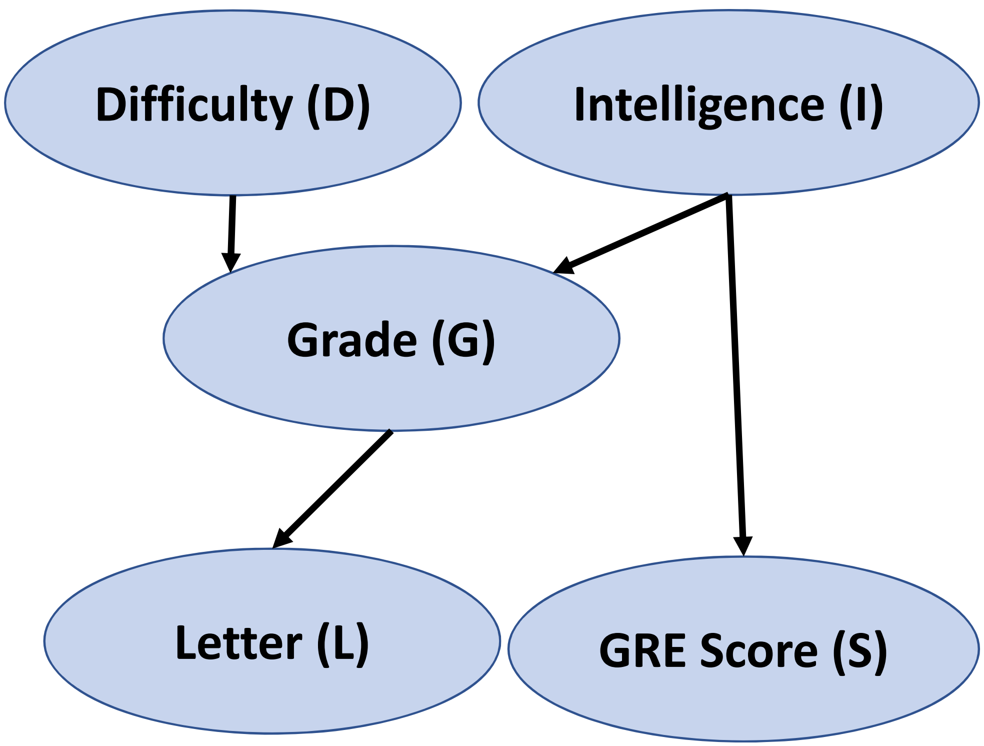
$$G \perp S, L \mid I, D$$

$$S \perp G, D, L \mid I$$

$$L \perp I, S, D \mid G$$

Distribution Factorization Example

What have we gained by factoring the distribution?



- The joint distribution can now be expressed:

  $$P(D, I, S, G, L) =$$
  $$P(D)\, P(I)\, P(S|I)\, P(G|I, D)\, P(L|G)$$

- One **joint table replaced with 5 CPTs**
- Largest table is now:

  2 x 2 x 3 = 12 cases

- Total cases in all tables:

  2 + 2 + 12 + 6 + 4 = 26

- For large scale problem reduction in table size can be orders of magnitude!

Distribution Factorization Example

For example of using Bayes network for self driving car

https://www.youtube.com/watch?v=avLyV7SC22I&app=desktop

Pay attention to the posterior distributions of the variables

https://www.youtube.com/watch?v=D1jds-KxXJA

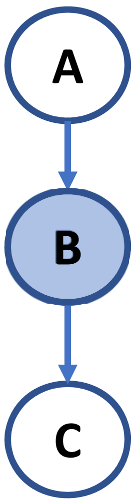
NVIDIA is clearly promoting deep learning technology. Some type of Bayes network is used to integrate the uncertain outputs of the multiple sensors. Check out the probability of a clear path at an intersection starting at 0:07, posterior distribution of sensors displayed starting at 0:20 and the noisy LIDAR starting at 0:28 and uncertainty in path planning staring at 0:48

Independencies Structures in Directed Graphical Models

- How can we define the local directed graph structures that define independencies?
- There are only 4 types of structure that determine independency properties
- These are considered **local independencies** expressed by **local semantics**
- **Definition:** The **local semantics** of a DAG specifies that each node is conditionally independent of its non-descendants given its parents
- Understanding these local semantics enables analysis of complex graphs globally

Independencies Structures in Directed Graphical Models

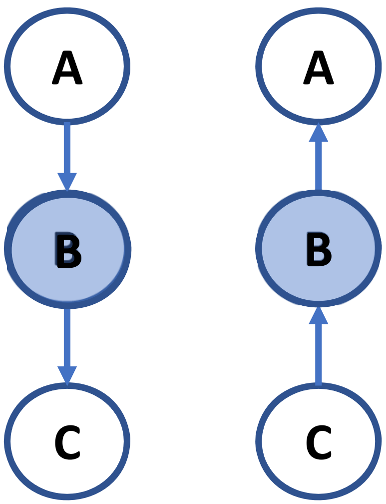
There are 4 types of graph structures that govern independencies



- A **cascade structure**, or **causal relationship:**
$$A \rightarrow B \rightarrow C$$
- In general A and C are not independent, $A \not\perp C$
- However, if **B is observed** then A and C are **conditionally independent,** $A \perp C \mid B$

Independencies Structures in Directed Graphical Models

There are 4 types of graph structures that govern independencies



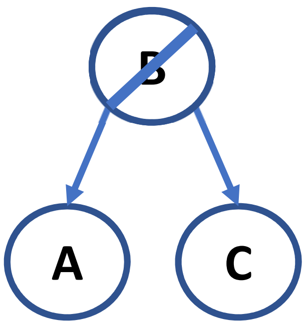
- Reversing the arrows **does not change the independencies**
- Gives an **evidential structure:**

$$C \rightarrow B \rightarrow A$$

- In general A and C are not independent**,** $A \not\perp C$
- However, if **B is observed** then A and C are **conditionally independent**, $A \perp C \mid B$
- **Independencies**, I(G), for a graphical models are **not unique**

Independencies Structures in Directed Graphical Models

There are 4 types of graph structures that govern independencies



- A **common cause**, or **common parent relationship:**

  $B \rightarrow A$ and $B \rightarrow C$

- In general A and C are **independent,**

  $A \perp C$ or $P(A, B, C) = P(A \mid B)\, P(C \mid B)\, P(B)$

- However, if B is observed then A and C cannot be independent, $A \not\perp C \mid B$

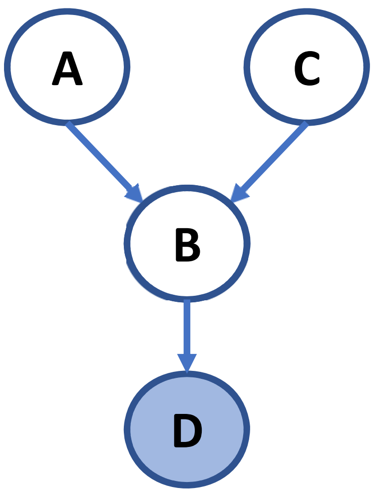
- If **B is marginalized out**, then:

$$\sum_B p(A, B, C) = p(A)\, p(B)$$

Or $A \perp B \mid marginal(B)$

Independencies Structures in Directed Graphical Models

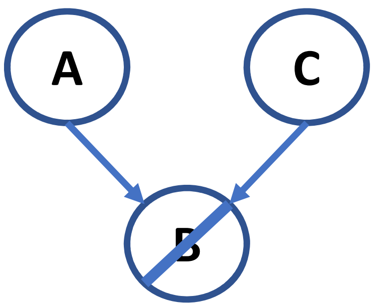
There are 4 types of graph structures that govern independencies



- A **common evidence**, **v-structure,** or **collider**
- In general A and C are unconditionally independent**,**
  $A \perp C$ or $P(A,B,C) = P(B \mid A,C)\, P(A)\, P(C)$
- However, if B is observed then A and C cannot be independent, $A \not\perp C \mid B$ or,
  $P(A,B,C) = P(A,C \mid B)P(B) = P(A \mid B)\, P(C \mid B)P(B)$
- The above **independcies applies to ancestors of the v-structure**

Independencies Structures in Directed Graphical Models

There are 4 types of graph structures that govern independencies



- A **common evidence**, **v-structure,** or **collider**
- If B is marginalized out, then:

$$\sum_B p(A, B, C) = p(A)\ p(B)$$

Or $A \perp B \mid marginal(B)$

Mapping Independencies in Directed Graphical Models

- We have looked at simple rules for local independencies
- How do we extend these concepts to complex graphs?
- Use the concept of **D-separation**:

  > **Definition:** Given subsets *X*, *Y* and *Z*, *X* and *Y* are conditionally independent or **D-separated** conditioned on the subset *Z* if they are separated on the moralized graph.

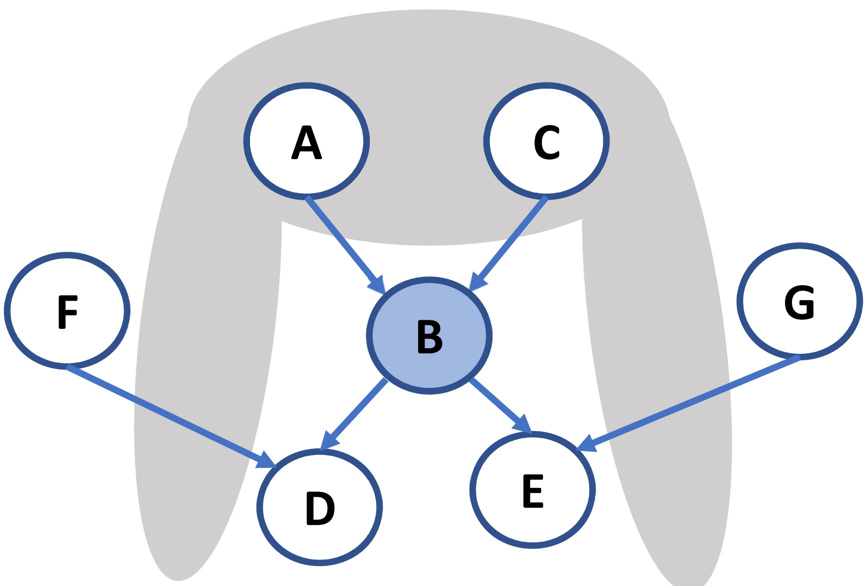
- We can say that, *X* and *Y* are **D-separated if all paths** between them in *Z* are **blocked**
- We can state d-separation in terms of the **local Markov assumption**:

  $$I(G) = \{X \perp Z \mid Y : \ dsep_G(X : Z|Y)\}$$

  Note: we will discuss moralization of graphs in the next lecture

Mapping Independencies in Directed Graphical Models

Relating local semantics to global semantics



- By local semantics the a node is conditionally independent of its non-descendants.
- The local semantics map to the global semantics and vice versa

$$\textit{local semantics} \iff \textit{global semantics}$$

Mapping Independencies in Directed Graphical Models

Independence map

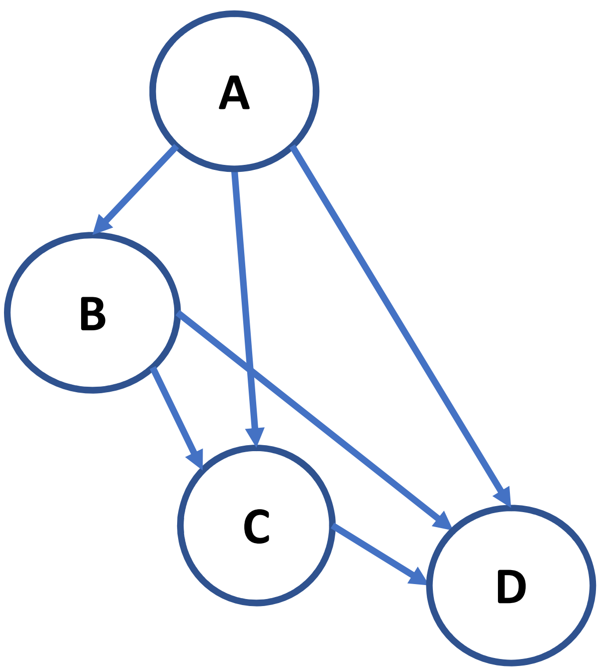
Definition: A DAG, G, is an **independence map** or **I-map** of a distribution P if $I_l(G) \subseteq I(P)$, where $I(P)$ is the set of independencies of the distribution P and $I_l(G)$ is the set of independencies of the DAG. We can express this relationship as:

$$(X \perp Y \mid Z_G) \Rightarrow (X \perp Y \mid Z_P)$$

- This relationship is not unique and there are can be multiple graphs for which $I_l(G) \subseteq I(P)$

Mapping Independencies in Directed Graphical Models

Fully connected DAG is an I-map for any distribution



- This means we can **always create a DAG**, G, which is an **I-map of any distribution**, P, such that, $I_l(G) \subseteq I(P)$!
- But, a more compact representation is a **minimal independence map**

  Definition: A DAG, G, is a **minimal I-map** for a distribution P if removal of even a single edge renders G not an I-map.

Mapping Independencies in Directed Graphical Models

Dependency map

Definition: A graph G is a **dependency map** or **D-map** of a distribution P if the graph contains every conditional independence in P.

We can represent this relationship as:

$$(X \perp Y \mid Z_G) \Leftarrow (X \perp Y \mid Z_P)$$

Mapping Independencies in Directed Graphical Models

Perfect map

Definition: If a graph G is **both an I-map and a D-map** of a distribution P we say that G is a **prefect map** of P.

We can write this relationships as:

$$(X \perp Y \mid Z_G) \Leftrightarrow (X \perp Y \mid Z_P)$$

- It would be nice if a graph were a perfect map of a distribution
- This will rarely be the case in real world problems.
- Thus, a perfect map is mostly useful as a reference point in developing probabilistic graphical models.

Paths and Bayes’ Ball Algorithm

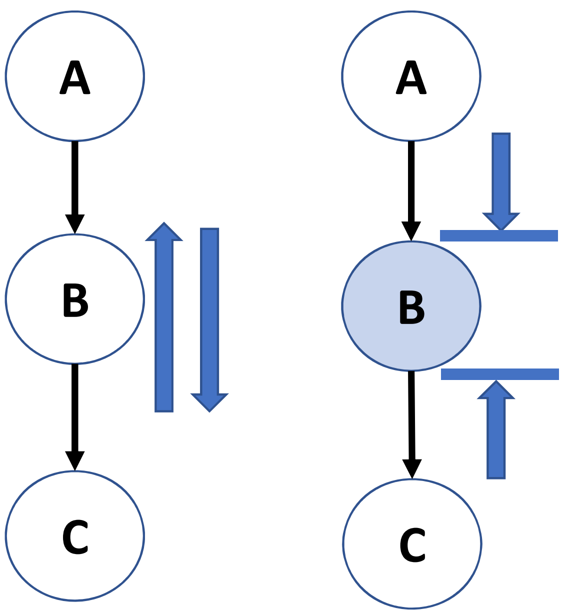
Two subsets of variables, X and Y are **D-separated if there is no active trail or path** though the subset Z between them

Is there a way to determine if a **path or trail is blocked or active**?

- The **Bayes’ ball algorithm** determines if a path is **active**
- The ball rolls along a path
  - If the ball can complete the path, it is active
  - If the ball is blocked so is the path
- There are only a few simple rules to the Bayes’ ball algorithm

Paths and Bayes' Ball Algorithm

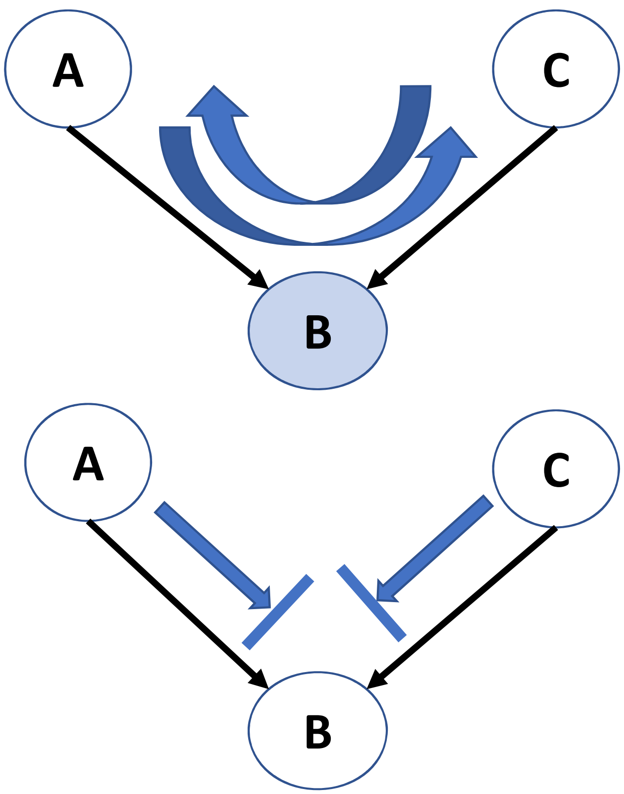
Rules for the Bayes' ball algorithm: **Causal trail or evidential trail**



- For a **causal relationship** or **evidential structure:**
- The ball can roll through: if **B is not observed the path is active**
- If **B is observed the path is blocked**

Paths and Bayes' Ball Algorithm

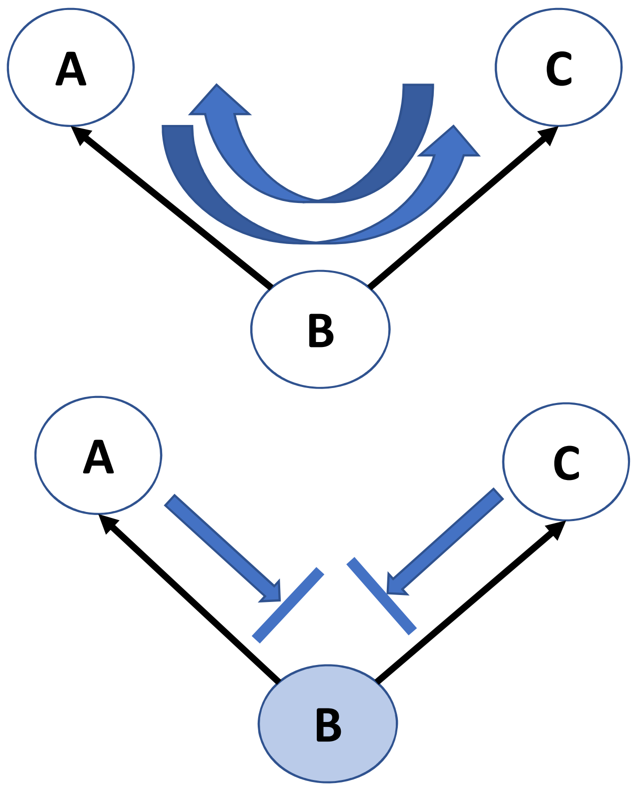
Rules for the Bayes' ball algorithm: **Common cause trail**



- For a **V-structure** or **common evidence:**
- The ball can roll through: if **B is observed the path is active**
- If **B not is observed the path is blocked**

Paths and Bayes' Ball Algorithm

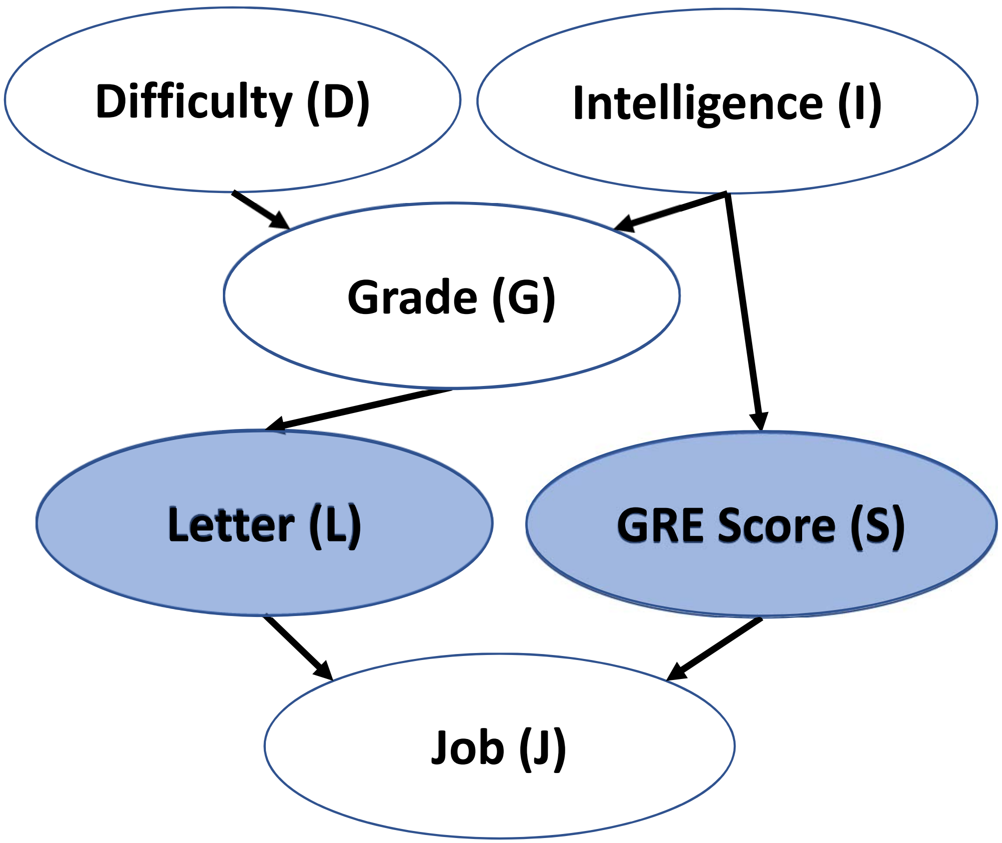
Rules for the Bayes' ball algorithm: **Common effect trail**



- For a **common cause** or **common parent:**
- The ball can roll through: if **B is not observed the path is blocked**
- If **B is observed the path is blocked**

Mapping Independencies in Directed Graphical Models

Examples of D-separation on a graph

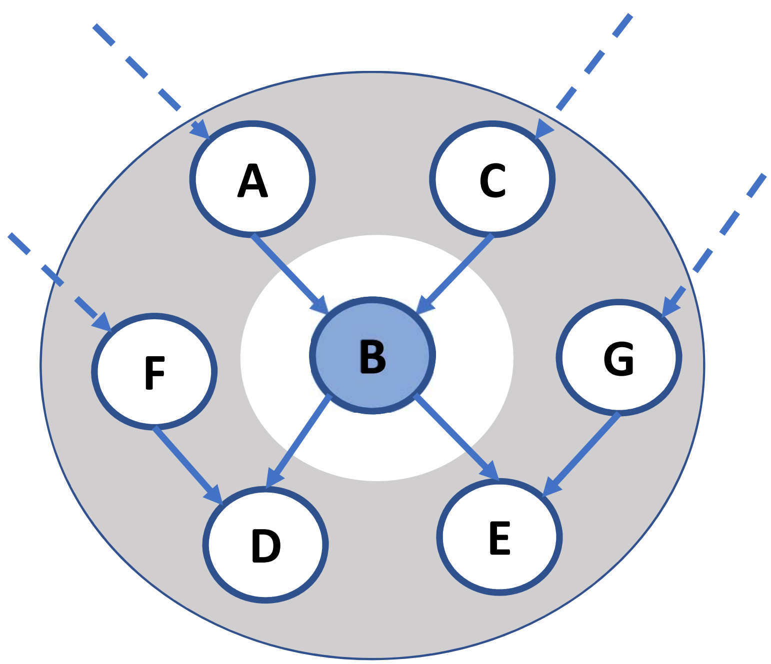


- Let X = {D}, Y = {I}, Z = {G}
- If G is not observed, X and Y are D-separated

- Or, let X = {D, I, G}, Y = {J}, and Z = {L,S}
- If L and S are observed then X and Y are D-separated

Mapping Independencies in Directed Graphical Models

For a DAG, any node is conditionally independent or all others given its **Markov Blanket**



The **Markov blanket** includes**:**

- Parents
- Children
- Parents of children
- Above define Markov blanket

Summary

In many cases, can factorize a distribution, P, or a directed graph, G

> **Definition**: Given a distribution P and a Bayesian network G, P factorizes as a set of CDPs specified as the nodes of G

- Graph can be an I-map of P
- P can be a D-map of G
- A perfect map is both an I map and a D-map
- Useful model is rarely a perfect map