

Advanced Prediction Models

Deep Learning, Graphical Models and Reinforcement
Learning


Today's Outline

- Motivation
- Primer on Graphs
- Directed Graphical Models
- Undirected Graphical Models


Recent Turing Award (highest in the CS discipline)




MORE ACM AWARDS



A.M. TURING CENTENARY CELEBRATION WEBCAST

 Search





A.M. TURING AWARD WINNERS BY...

ALPHABETICAL LISTING

YEAR OF THE AWARD

RESEARCH SUBJECT



 **Photo-Essay**


BIRTH:
September 4, 1936, Tel Aviv.


EDUCATION:
B.S., Electrical Engineering (Technion, 1960); M.S., Electronics (Newark College of Engineering, 1961); M.S., Physics (Rutgers University, 1965); Ph.D., Electrical Engineering (Polytechnic Institute of Brooklyn, 1965).


EXPERIENCE:
Research Engineer, New York University Medical School (1960–1961); Instructor,


JUDEA PEARL
United States – 2011


CITATION
For fundamental contributions to artificial intelligence through the development of a calculus for probabilistic and causal reasoning.

 SHORT ANNOTATED BIBLIOGRAPHY

 ACM DL AUTHOR PROFILE

 ACM TURING AWARD LECTURE VIDEO

 RESEARCH SUBJECTS

 ADDITIONAL MATERIALS

Judea Pearl created the representational and computational foundation for the processing of information under uncertainty.

He is credited with the invention of *Bayesian networks*, a mathematical formalism for defining complex probability models, as well as the principal algorithms used for inference in these models. This work not only revolutionized the field of artificial intelligence but also became an important tool for many other branches of engineering and the natural sciences. He later created a mathematical framework for *causal inference* that has had significant impact in the social sciences.

Judea Pearl was born on September 4, 1936, in Tel Aviv, which was at that time administered under the British Mandate for Palestine. He grew up in *Bnei Brak*, a Biblical town his grandfather went to reestablish in 1924. In 1956, after serving in the Israeli army and joining a Kibbutz, Judea decided to study engineering. He attended the Technion, where he met his wife, Ruth, and received a B.S. degree in Electrical Engineering in 1960. Recalling the Technion faculty members in a 2012 interview in the *Technion Magazine*, he emphasized the thrill of discovery:

¹Reference: David Sontag, (2013)

Why Graphical Models

- We have seen deep learning techniques for unstructured data
 - Predominantly vision and text/audio
 - We will see control in the last part of the course
 - (Reinforcement Learning)

Why Graphical Models

- We have seen deep learning techniques for unstructured data
 - Predominantly vision and text/audio
 - We will see control in the last part of the course
 - (Reinforcement Learning)
- For structured data, graphical models are the most versatile framework
 - Successfully applications:
 - Kalman filtering in engineering
 - Decoding in cell phones (channel codes)
 - Hidden Markov models for time series
 - ...

Graphical Models vs Deep Learning

Graphical Models

- Probabilistic
- Dependencies btw. RVs
- Low capacity
- Domain knowledge: easy to encode

Deep Neural Networks

- Deterministic
- Input/Output Mapping
- High capacity
- Domain knowledge: hard

Graphical Models Landscape

- Three parts to the story:
 - Representation ([this lecture](#))
 - Capture uncertainty (joint distribution)
 - Capture [conditional independences](#) (metadata)
 - Visualization of metadata for a distribution

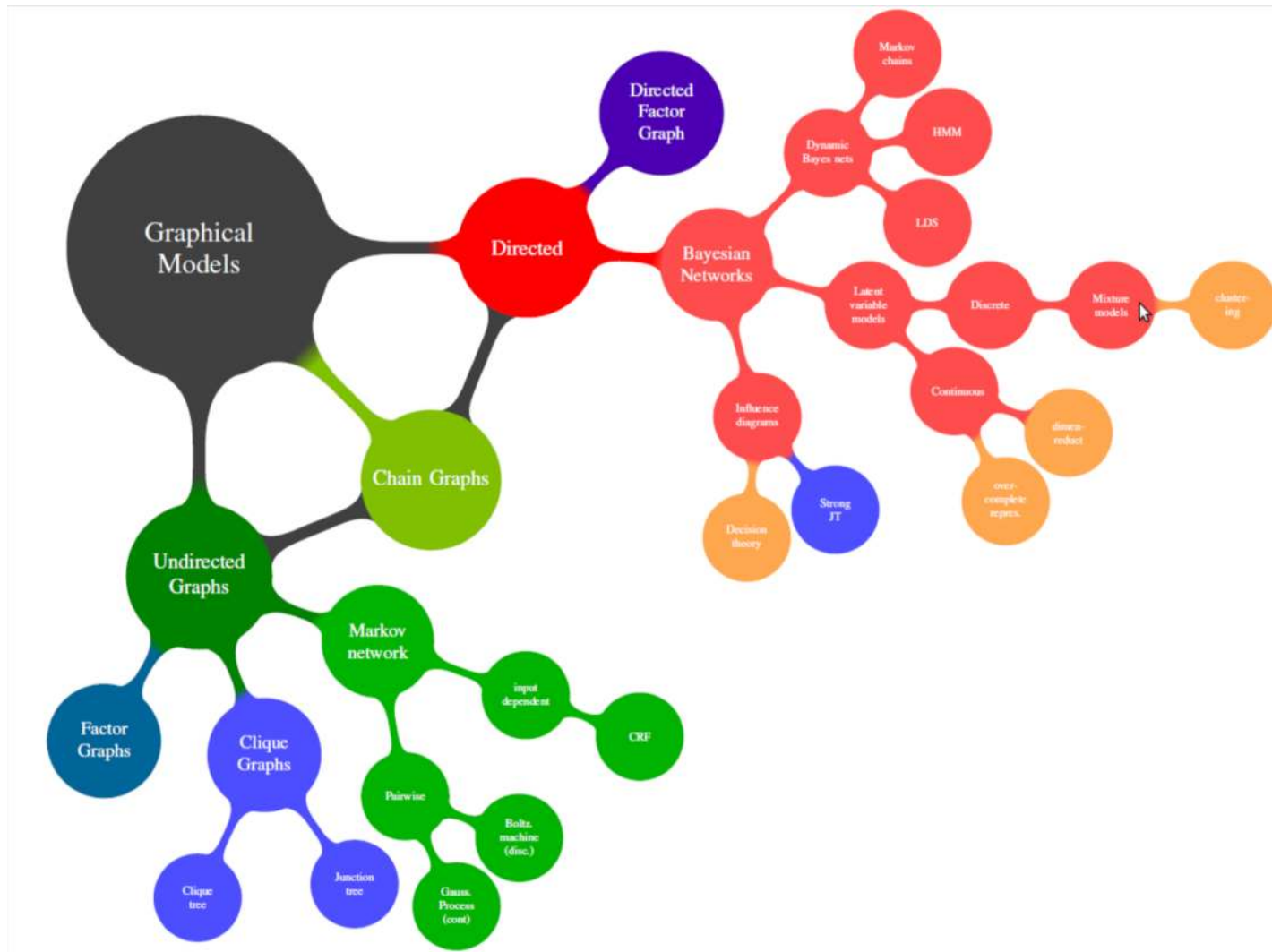
Graphical Models Landscape

- Three parts to the story:
 - Representation ([this lecture](#))
 - Capture uncertainty (joint distribution)
 - Capture [conditional independences](#) (metadata)
 - Visualization of metadata for a distribution
 - Inference
 - Create data structures for computing marginal or conditional distributions [quickly](#)

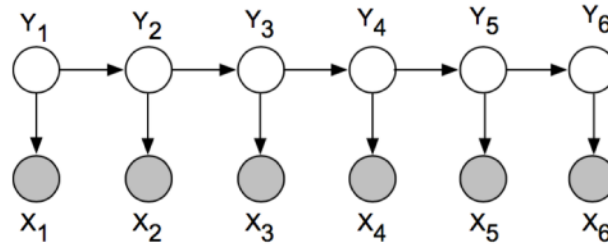
Graphical Models Landscape

- Three parts to the story:
 - Representation ([this lecture](#))
 - Capture uncertainty (joint distribution)
 - Capture [conditional independences](#) (metadata)
 - Visualization of metadata for a distribution
 - Inference
 - Create data structures for computing marginal or conditional distributions [quickly](#)
 - Learning
 - Learning the [parameters of the distribution](#) can be aided by graph techniques

Graphical Models Landscape



Application 1: Hidden Markov Model

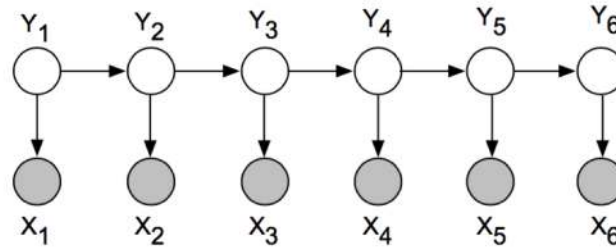


- Frequently used for speech recognition and part-of-speech tagging
- Joint distribution factors as:

$$p(\mathbf{y}, \mathbf{x}) = p(y_1)p(x_1 | y_1) \prod_{t=2}^T p(y_t | y_{t-1})p(x_t | y_t)$$

- $p(y_1)$ is the distribution for the starting state
 - $p(y_t | y_{t-1})$ is the *transition* probability between any two states
 - $p(x_t | y_t)$ is the *emission* probability
- What are the conditional independencies here? For example,
 $Y_1 \perp \{Y_3, \dots, Y_6\} \mid Y_2$

Application 1: Hidden Markov Model



- Joint distribution factors as:

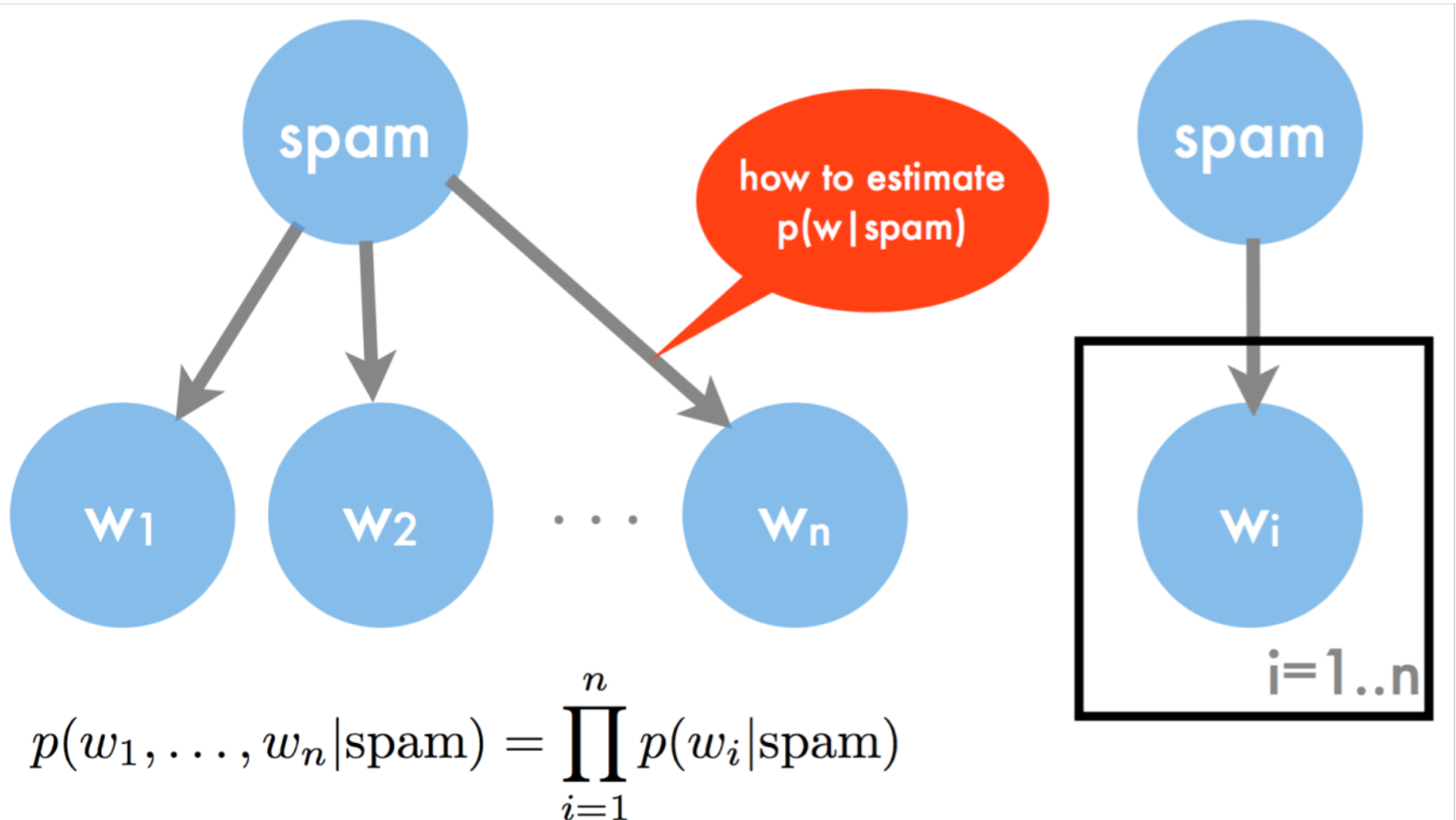
$$p(\mathbf{y}, \mathbf{x}) = p(y_1)p(x_1 | y_1) \prod_{t=2}^T p(y_t | y_{t-1})p(x_t | y_t)$$

- A **homogeneous** HMM uses the same parameters (β and α below) for each transition and emission distribution (**parameter sharing**):

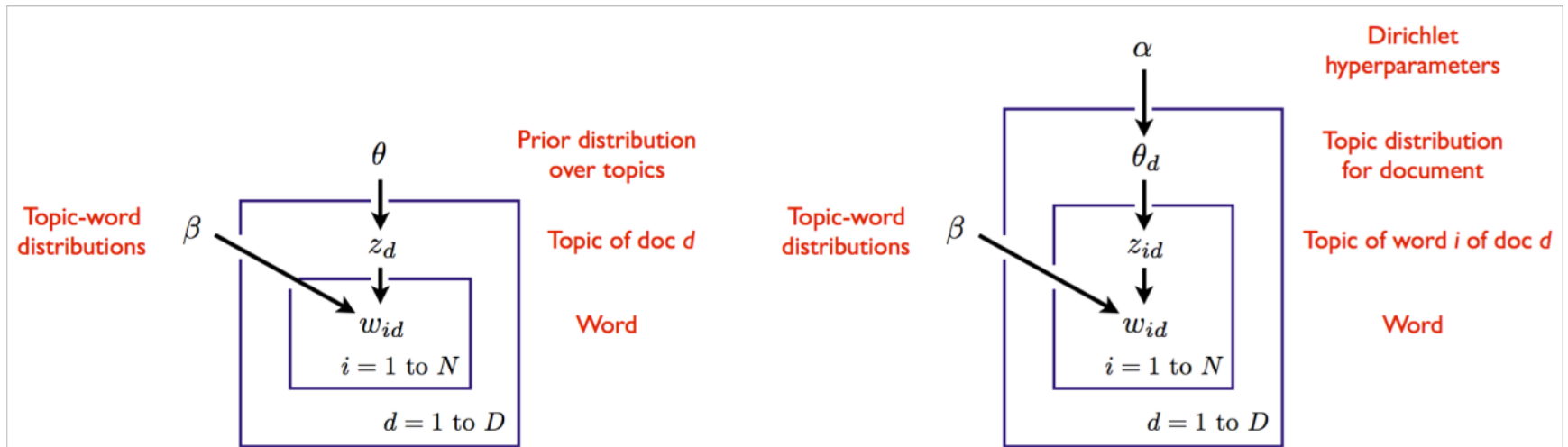
$$p(\mathbf{y}, \mathbf{x}) = p(y_1)\alpha_{x_1, y_1} \prod_{t=2}^T \beta_{y_t, y_{t-1}} \alpha_{x_t, y_t}$$

How many parameters need to be learned?

Application 2: Naïve Bayes Spam Filter



Application 3: Latent Dirichlet Allocation



- Model on left is a **mixture model**
 - Called *multinomial* naive Bayes (a word can appear multiple times)
 - Document is generated from a single topic
- Model on right (LDA) is an **admixture model**
 - Document is generated from a distribution over topics

Application 4: Conditional Random Field

- **Conditional random fields** are undirected graphical models of conditional distributions $p(\mathbf{Y} \mid \mathbf{X})$
 - \mathbf{Y} is a set of **target variables**
 - \mathbf{X} is a set of **observed variables**
- A CRF is a Markov network on variables $\mathbf{X} \cup \mathbf{Y}$, which specifies the conditional distribution

$$P(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{c \in C} \phi_c(\mathbf{x}_c, \mathbf{y}_c)$$

with partition function

$$Z(\mathbf{x}) = \sum_{\hat{\mathbf{y}}} \prod_{c \in C} \phi_c(\mathbf{x}_c, \hat{\mathbf{y}}_c).$$

Questions?

Today's Outline

- Motivation
- Primer on Graphs
- Directed Graphical Models
- Undirected Graphical Models

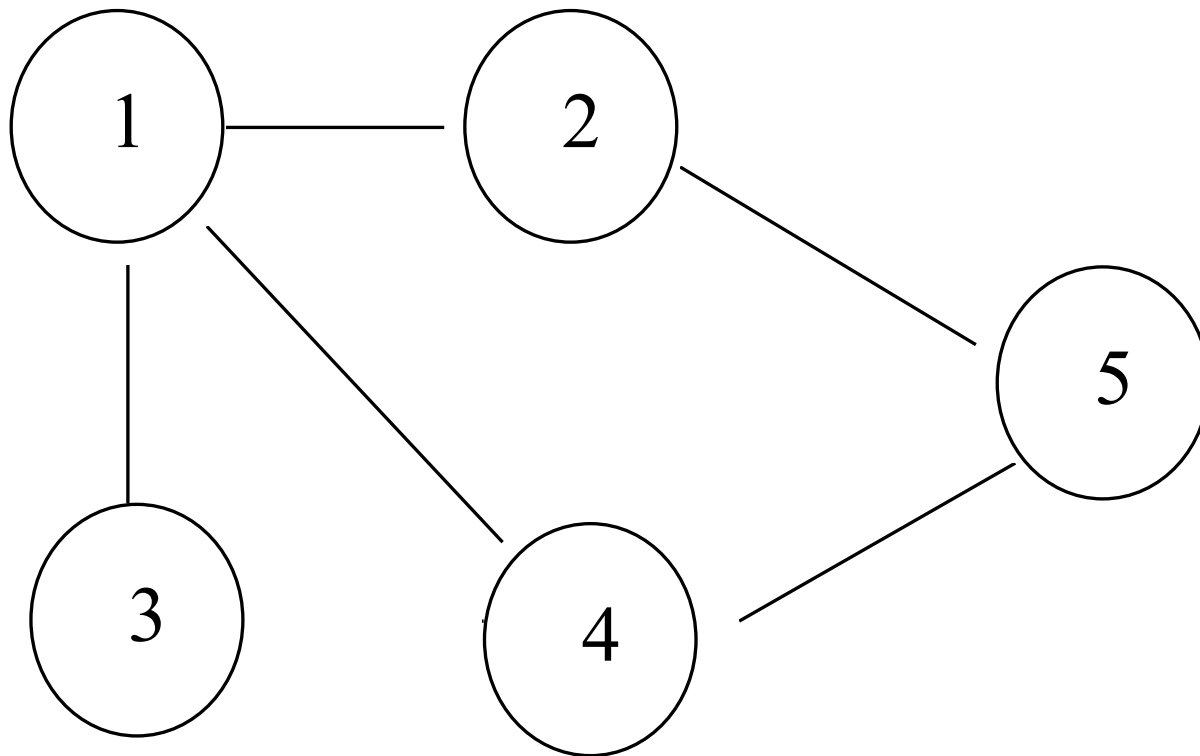
Primer on Graphs

Graph

- A network with
 - Edges (links)
 - Vertices (nodes)
- Heavily used in Computer Science for algorithms and data structures
- Here, we will only need the terminology of graphs.
 - As we will see, their primary purpose will be visualization and encoding domain knowledge

Undirected Graph

- An undirected graph
 - Edges have no direction information

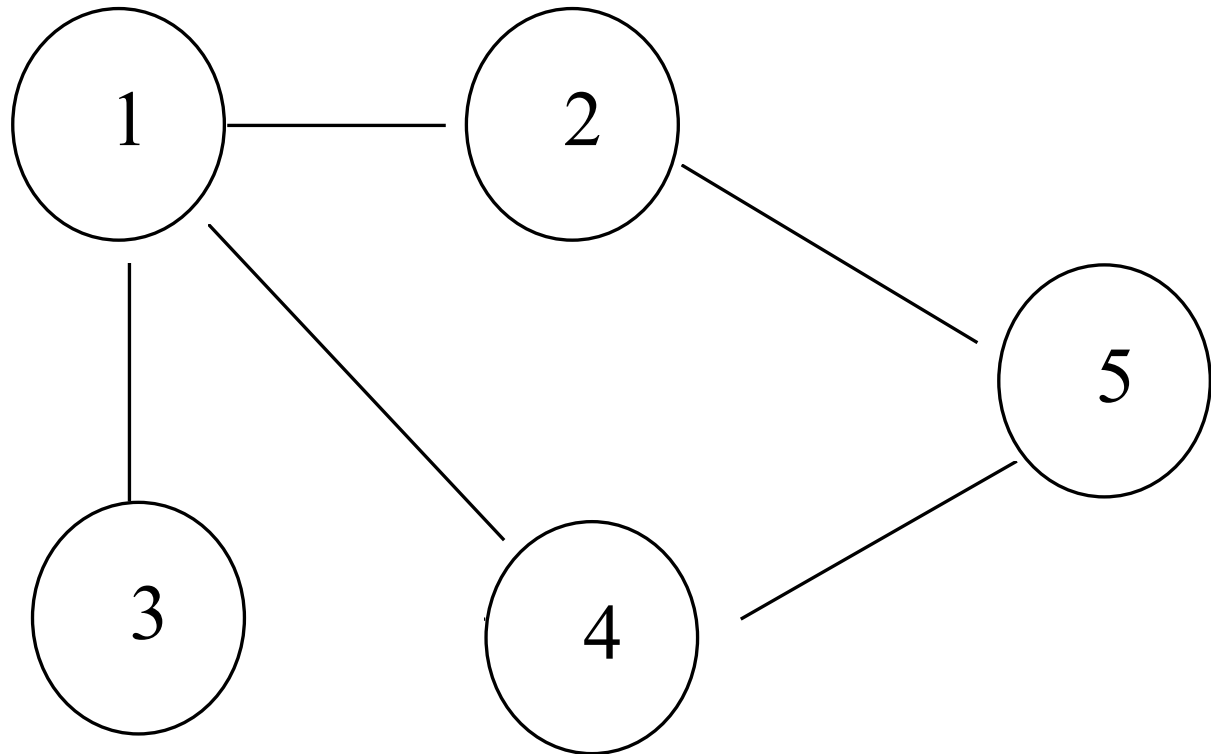


Notation for Undirected Graphs

- Set of vertices denoted $1, \dots, N$
- Size of graph is N
- Edge is an (unordered) pair (i, j)
 - (i, j) is the same as (j, i)
 - indicates that i and j are directly connected
- Maximum number of edges: $N(N - 1)/2$ (order N^2)
- i and j connected if there is a path of edges between them
- Subgraph of G :
 - restrict attention to certain vertices and edges between them

Path

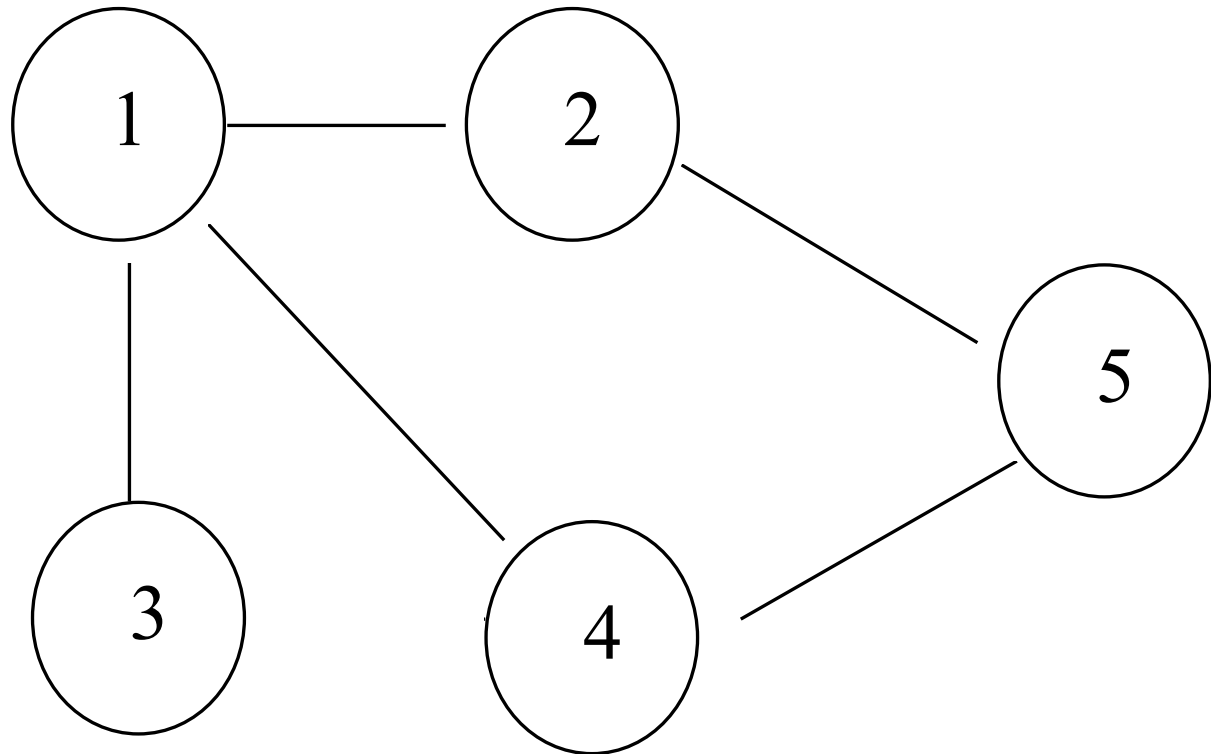
- A sequence of vertices where each successive pair are connected by an edge



- For example, $(3,4,5)$ is not a path. $(3,1,4,5)$ is a path

Neighbor

- All vertices that share an edge with the node are its neighbors. Denote as $nbhd(X)$



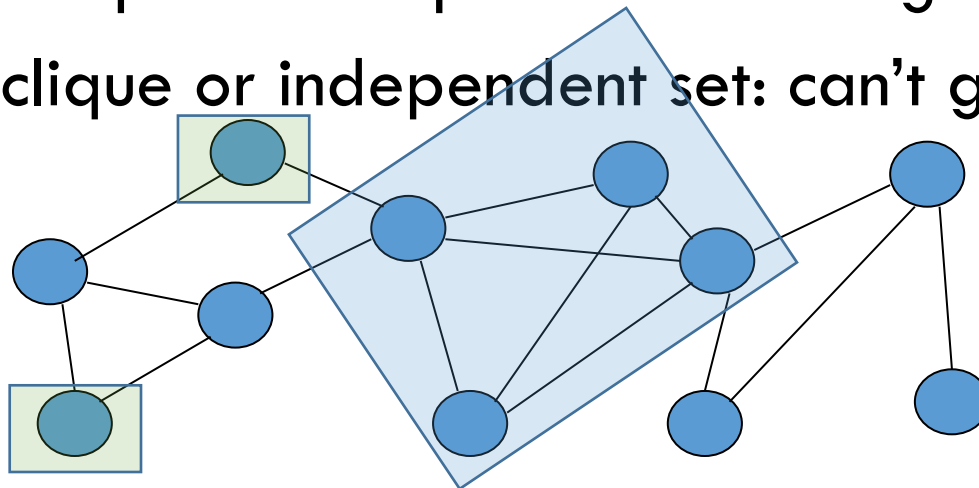
- For example, (3,4,2) are neighbors of 1.
 - $nbhd(1) = (3,4,2)$

Cliques and Independent Sets

- A clique in a graph G is a set of vertices:
 - informal: that are all directly connected to each other
 - formal: whose induced subgraph is complete
 - an edge is a clique of just 2 vertices

Cliques and Independent Sets

- A clique in a graph G is a set of vertices:
 - informal: that are all directly connected to each other
 - formal: whose induced subgraph is complete
 - an edge is a clique of just 2 vertices
- Independent set:
 - set of vertices whose induced subgraph is empty (no edges)
- Maximum clique or independent set: largest in the graph
- Maximal clique or independent set: can't grow any larger

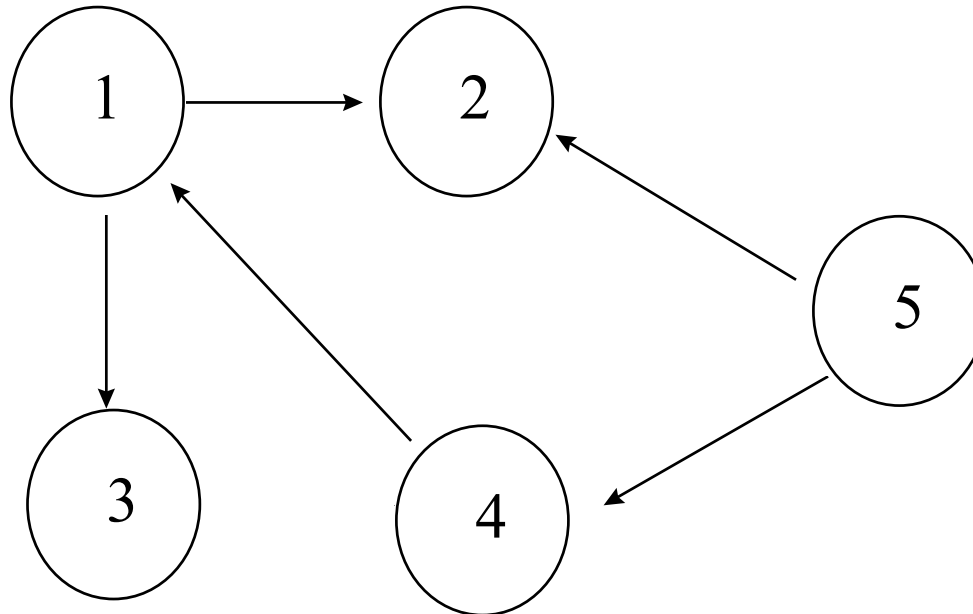


Directed Acyclic Graph

- A directed graph
 - Edges have directions or orientations
 - Edge (u,v) means $u \rightarrow v$
 - May also have edge (v,u)
 - Common for capturing asymmetric relations
- A directed acyclic graph (DAG)
 - No directed cycles
 - No way to follow the oriented edges and come back to the starting node

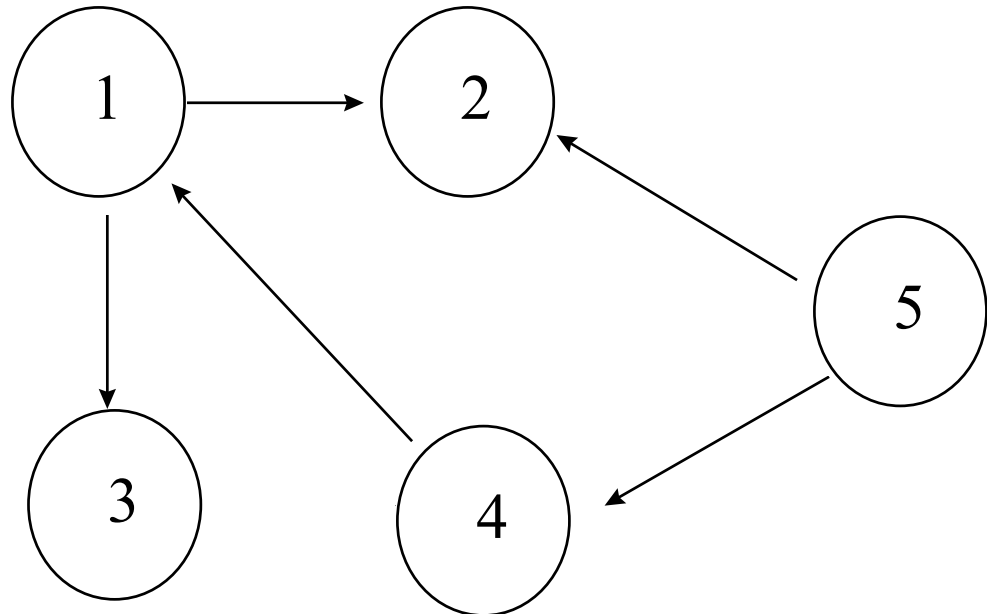
DAG

- A directed acyclic graph (DAG)



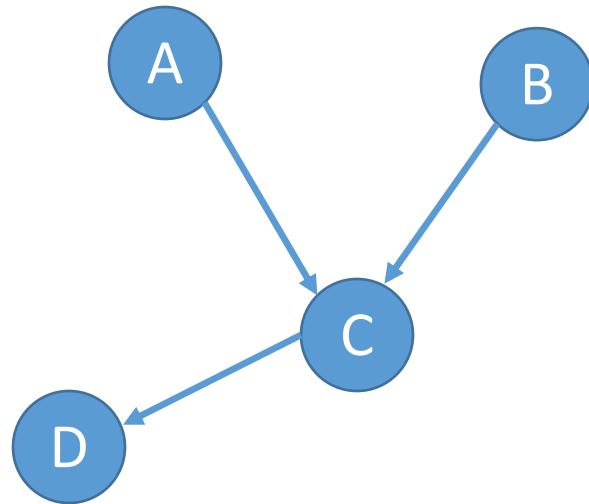
DAG Paths vs Directed Paths

- Path:
 - Same as undirected graph. Ignore directions
 - Example: (3,1,2) is a path
- Directed path
 - Take direction into account. E.g., (5,4,1,3) is a directed path



Parents of a Node

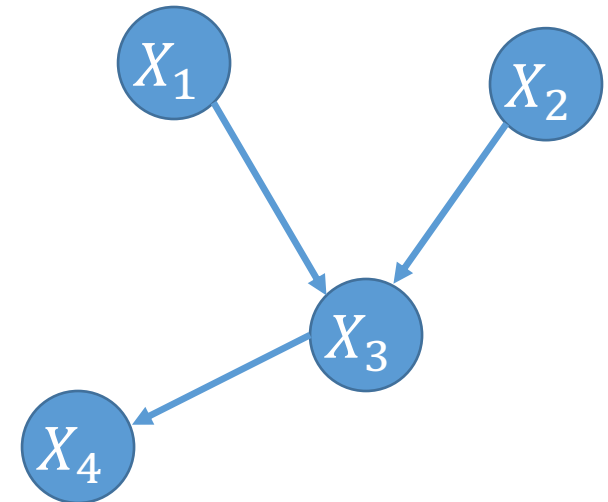
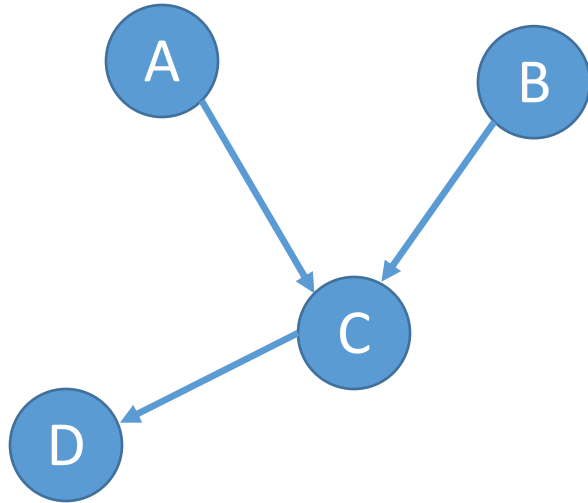
- Notation:
 - $pa(J)$ = Parents of node J



- In this graph, parents of C are (A, B)
 - Neighbors of that vertex that point to that vertex

Parents of a Node

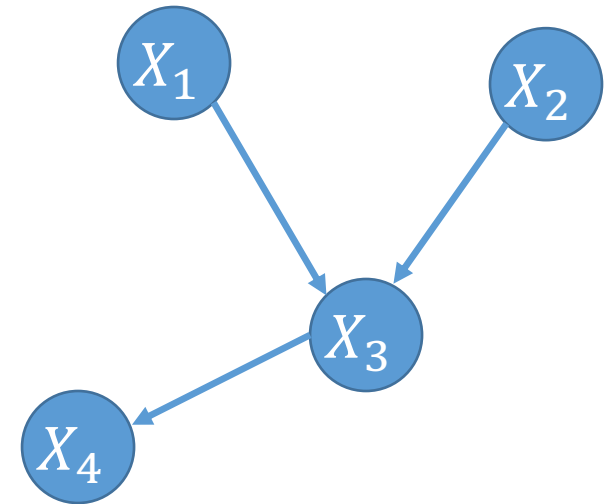
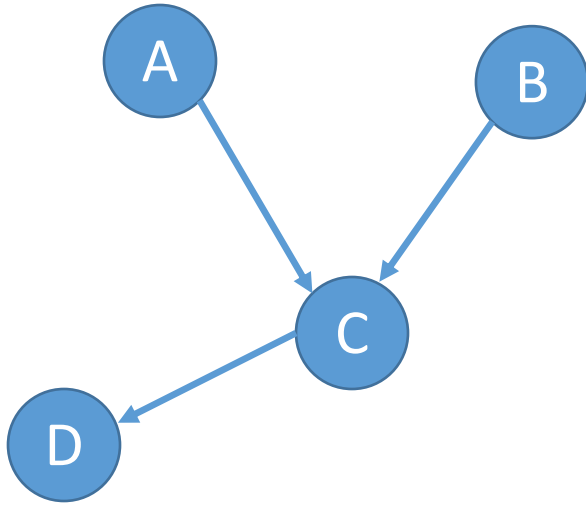
- In the below graph, parents of A is the empty set ϕ



- In the graph on the right, $pa(X_4) = (X_3)$

Descendants of a Node

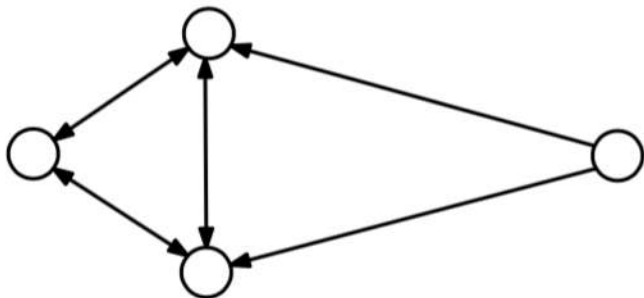
- All nodes that can be reached by following the arrow directions
- In the below graph, descendants of A are $\{C, D\}$



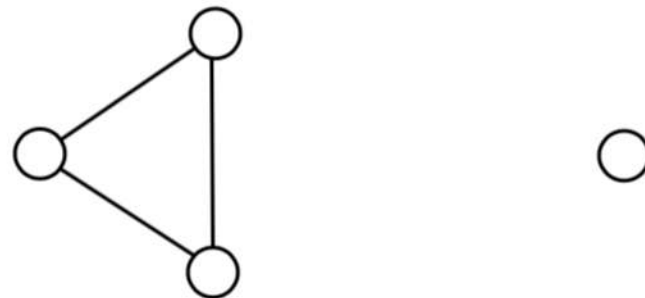
- In the graph on the right, $\text{Desc}(X_3) = (X_4)$

Connected Graphs

- G (directed or undirected) is connected if there is a path between any two vertices.
- Otherwise, we have connected components.
 - subgraphs determined by mutual connectivity
- Complete graph: edge between all pairs of vertices



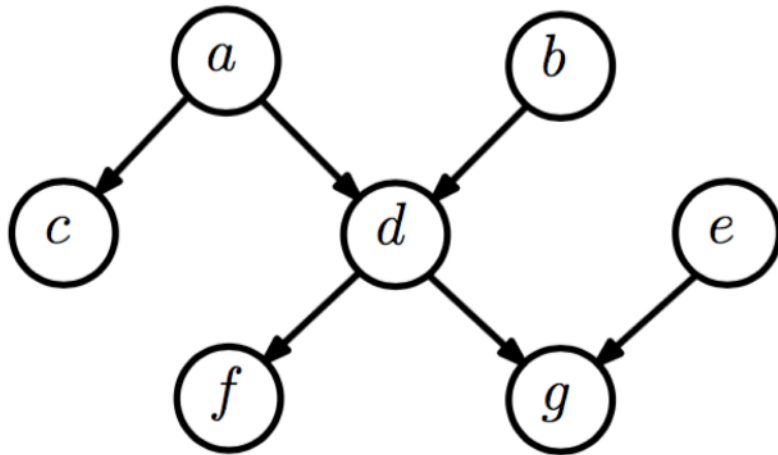
connected graph



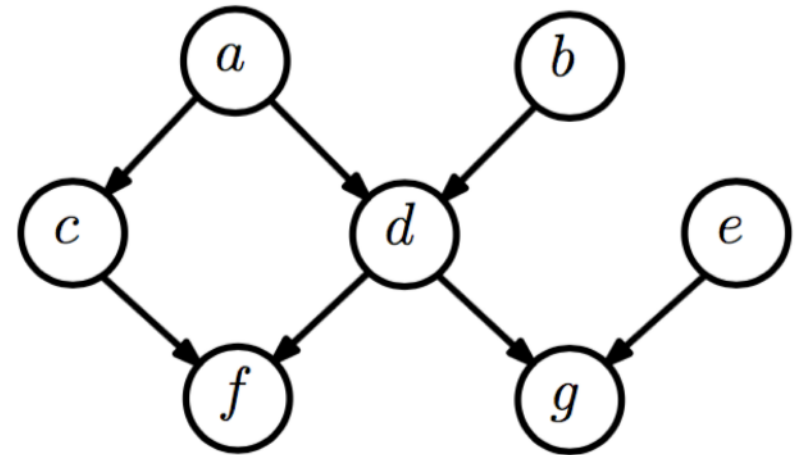
graph with
two connected components

Tree Graph (Singly-Connected)

- If for any vertex pair, there is no more than one path between them. This is also called a **tree**.



singly-connected



multiply-connected

- Otherwise, it is multiply-connected. Also called **loopy**.
- Similar definition for undirected graphs as well.

Questions?

Today's Outline

- Motivation
- Primer on Graphs
- Directed Graphical Models
- Undirected Graphical Models

Directed (Probabilistic) Graphical Models

Based on notes by *MathematicalMonk*¹

¹Reference: <https://www.youtube.com/playlist?list=PLD0F06AA0D2E8FFBA>

DPGM

- DPGM: Directed Probabilistic Graphical Model
- Also called a Bayesian Network or Belief Net
 - Nothing Bayesian here
- Directed graphs tell us about conditional independence properties of a probability distribution

Why Conditional Independence?

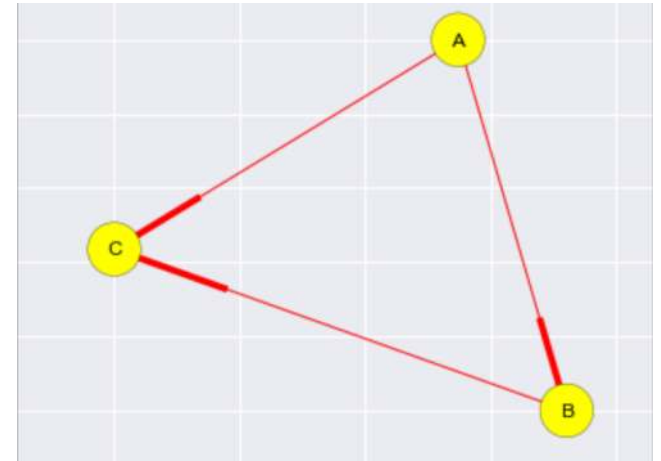
- Why do we care about conditional independence?
- Because we can perform tractable or efficient inference (we will address this next lecture!)

Joint Distribution

- Let A, B, C be RVs
- Joint distribution
 - $P(A = a, B = b, C = c)$
 - $= P(c|a, b)P(a, b)$
 - $= P(c|a, b)P(b|a)P(a)$
 - This is a factorization
 - We can always do this

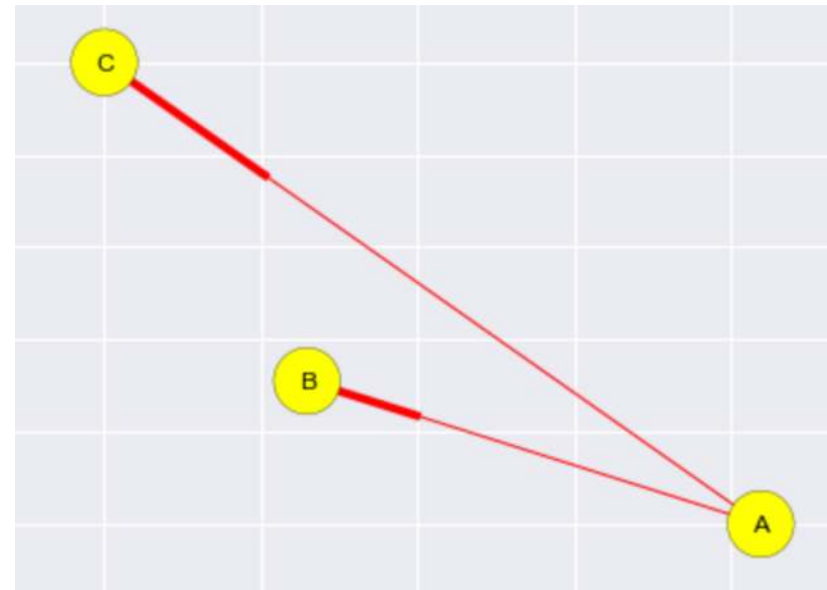
Factorizations are not Unique

- $P(c|a, b)P(b|a)P(a)$
- Create a node for each factor
- Graph has directed edges
- No cycles
 - Can't return to a node
- Nothing special about this factorization
 - We could have factored in a completely different way



Conditional Independence Changes the Graph

- If C is conditionally independent of B given A
- Use notation $C \perp B \mid A$
- Then $P(c|a, b) = P(c|a)$
- So we got a different graph



- Not every distribution could have lead to this graph.

Non-unique Graphs

- Given $X = (X_1, \dots, X_n) \sim P$, and a DAG G
- We say X respects G (or P respects G) if
 - $P(x_1, \dots, x_n) = \prod P(x_i | pa(x_i)) \quad \forall i = 1, \dots, n$

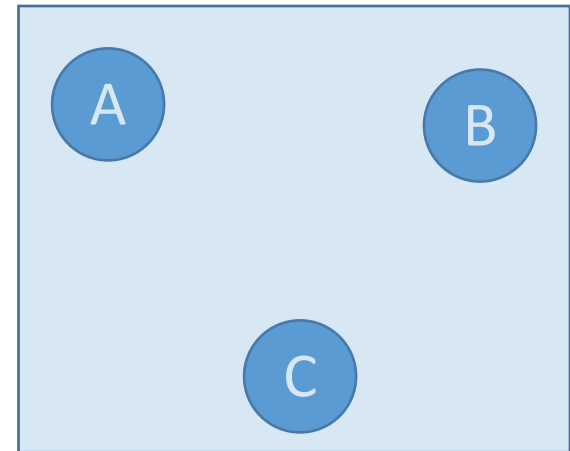
Non-unique Graphs

- Given $X = (X_1, \dots, X_n) \sim P$, and a DAG G
- We say X respects G (or P respects G) if
 - $P(x_1, \dots, x_n) = \prod P(x_i | pa(x_i)) \quad \forall i = 1, \dots, n$
- The graph G does not imply that any RVs are conditionally dependent.
 - At most, it will imply is conditional independence
- The graph G does not uniquely determine the probability distribution P

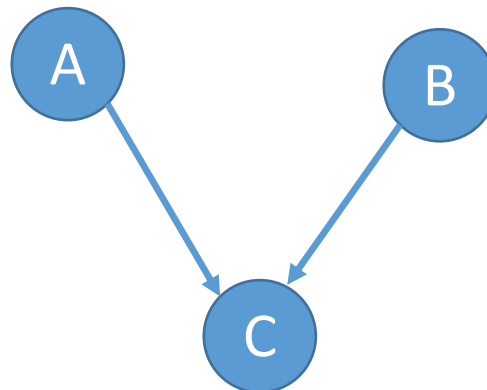
Non-unique Graphs: Example

- Say A,B,C are independent
 - $P(a,b,c) = P(a)P(b)P(c)$

- Let $X = (A,B,C)$
- Then X respects the graph G

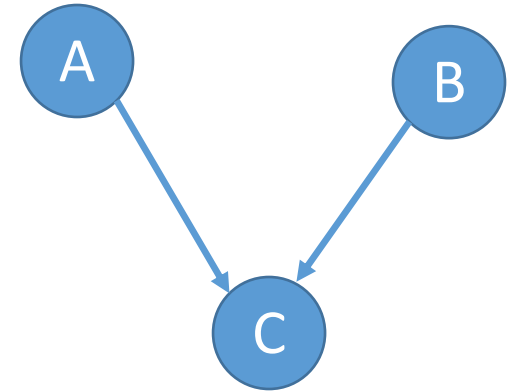


- X also respects G'

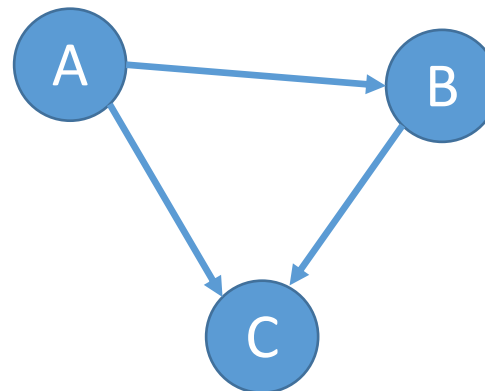


Non-unique Graphs: Example

- Graph G' is not saying C depends on A and B



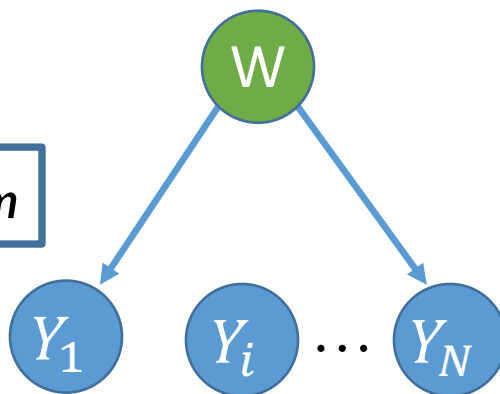
- It only says that the distribution of $X = (A, B, C)$ factors in a way that can be represented by G'
- X also respects G''



Example: Linear Regression

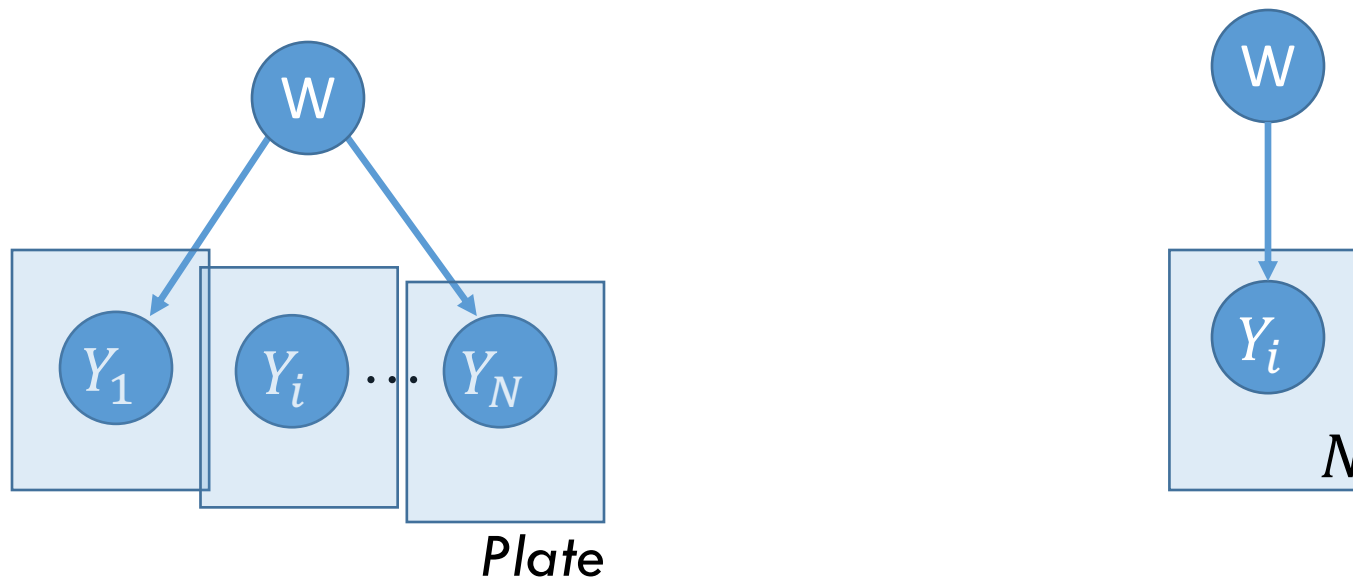
- Graphical model for (Bayesian) linear regression
 - Data: $\{x_i, y_i\}_{i=1}^N$ where x_i is d dimensional
 - Model: $f_W(x) = W^T \phi(x)$
 - Linear in W (not a matrix, a random vector)
- Let $W \sim N(0, \sigma_0^2 I)$
- Let $Y_i \sim N(W^T \phi(x_i), \sigma^2)$
 - Let Y_i be conditionally independent of Y_j given W

σ_0, σ and x_i are not random



Example: Linear Regression

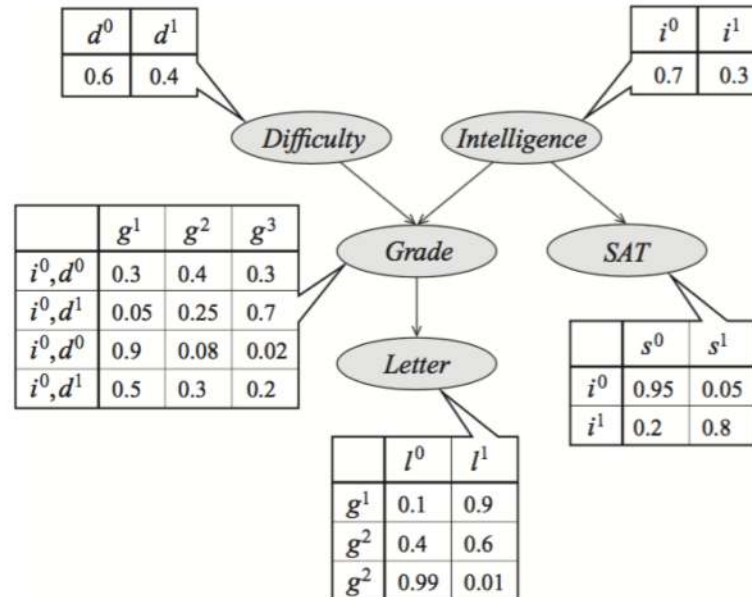
- $P(w, y_1, \dots, y_N) = P(w) \prod P(y_i | w)$
- Can also use a plate notation
 - Stack the plates on top of each other



- Variable W is called a latent or hidden variable
- Variables Y_i are called observed variables

Example: Student Network

- Consider the following Bayesian network:



- What is its joint distribution?

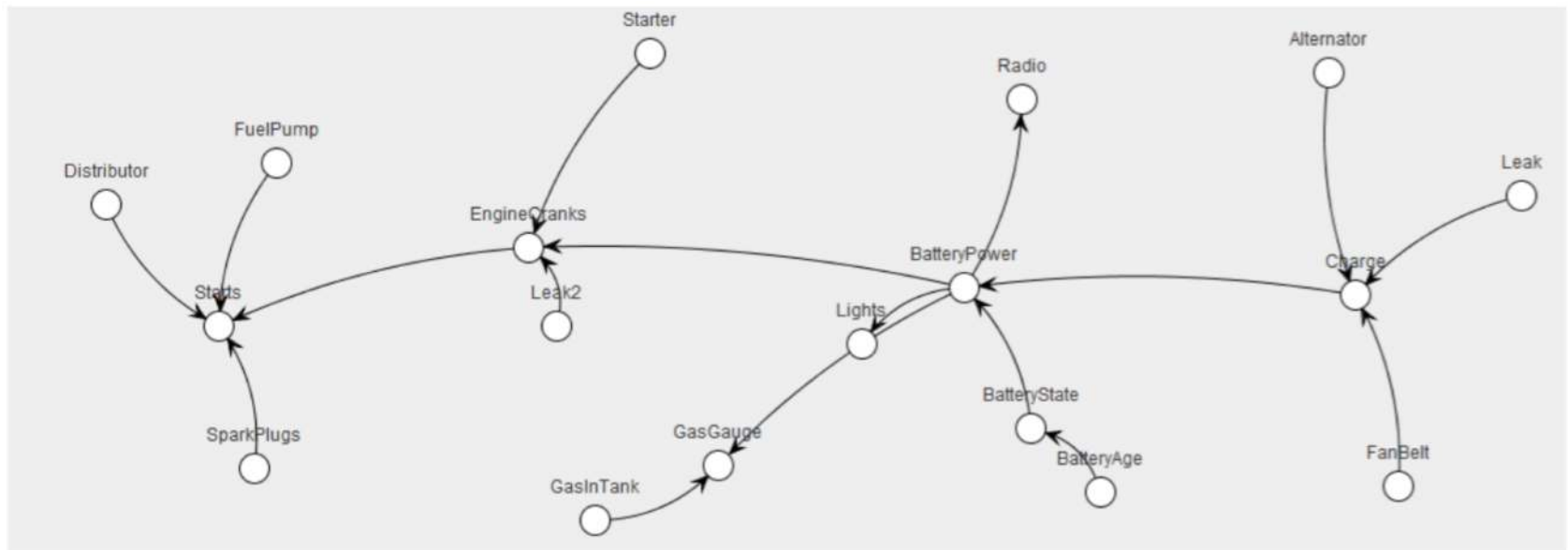
$$p(x_1, \dots, x_n) = \prod_{i \in V} p(x_i \mid \mathbf{x}_{\text{Pa}(i)})$$

$$p(d, i, g, s, l) = p(d)p(i)p(g \mid i, d)p(s \mid i)p(l \mid g)$$

Example: Car Network

$$p(x_1, \dots, x_n) = \prod_{i \in V} p(x_i \mid \mathbf{x}_{\text{Pa}(i)})$$

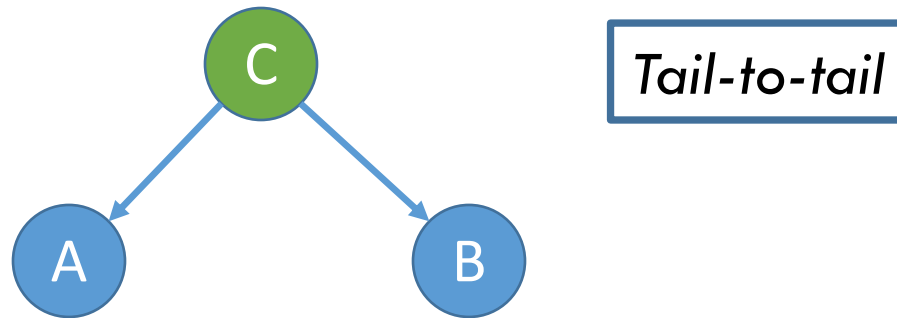
Will my car start this morning?



Heckerman *et al.*, Decision-Theoretic Troubleshooting, 1995

Conditional Independence (I)

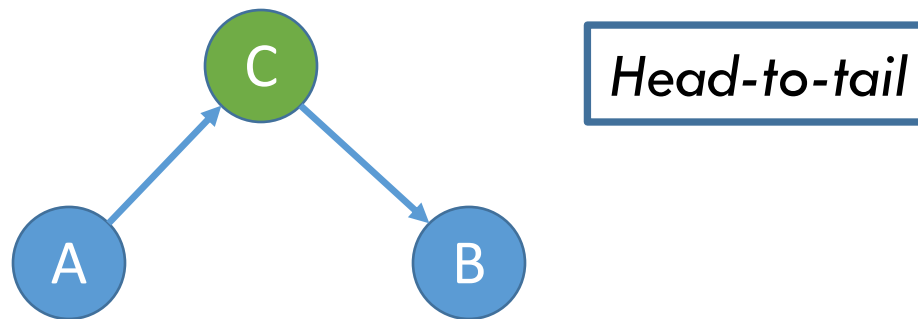
- Given a graphical model, we can determine if two sets of RVs are conditionally independent or not



- $P(a, b, c) = P(a|c)P(b|c)P(c)$ is a joint distribution that respects this graph
- What happens when we condition on C?
 - $P(a, b|c) = \frac{P(a, b, c)}{P(c)} = P(a|c)P(b|c)$
 - Thus, A and B are conditionally independent given C
 - Use notation $A \perp B \mid C$

Conditional Independence (II)

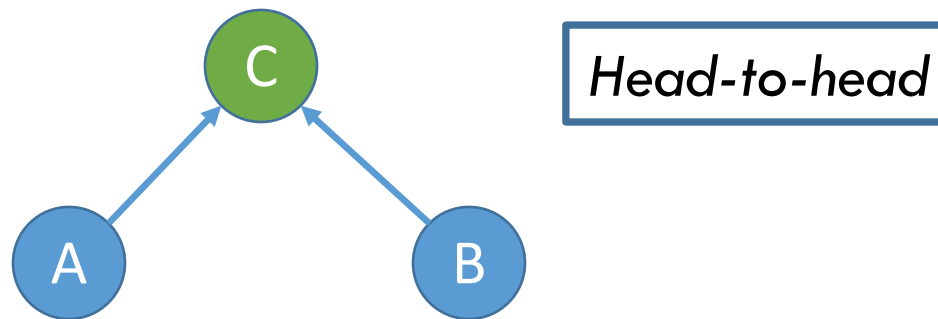
- Given a graphical model, we can determine if two sets of RVs are conditionally independent or not



- $P(a, b, c) = P(a)P(c|a)P(b|c) = [P(a|c)P(c)]P(b|c)$ is the joint distribution that respects this graph
- What happens when we condition on C?
 - $P(a, b|c) = \frac{P(a, b, c)}{P(c)} = P(a|c)P(b|c)$
 - Thus, A and B are conditionally independent given C
 - Use notation $A \perp B \mid C$

Conditional Independence (III)

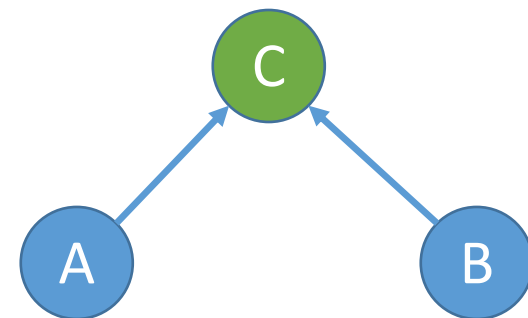
- Given a graphical model, we can determine if two sets of RVs are conditionally independent or not



- $P(a, b, c) = P(a)P(b)P(c|a, b)$ is the joint distribution that respects this graph
- What happens when we condition on C?
 - $P(a, b|c) = \frac{P(a, b, c)}{P(c)} \neq P(a|c)P(b|c)$
 - Cannot say A & B are conditionally independent given C

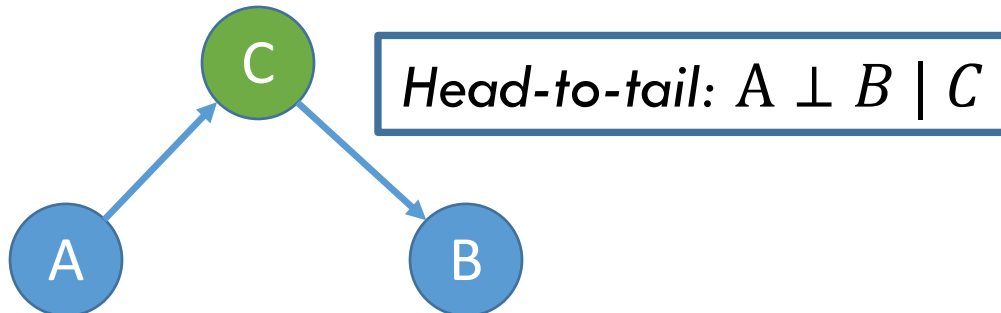
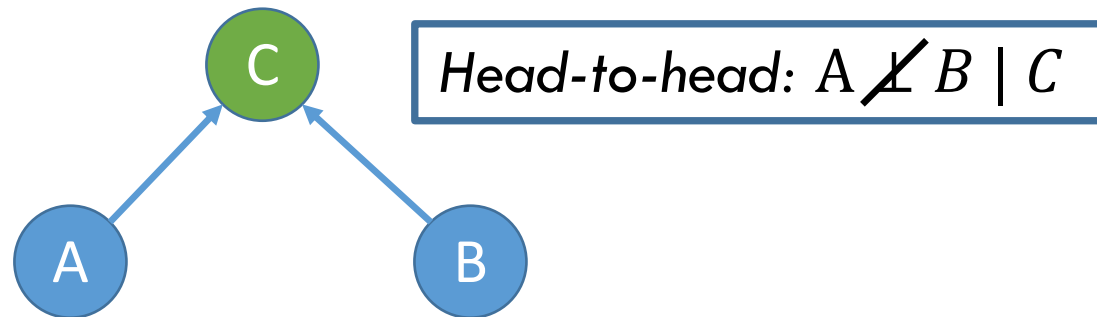
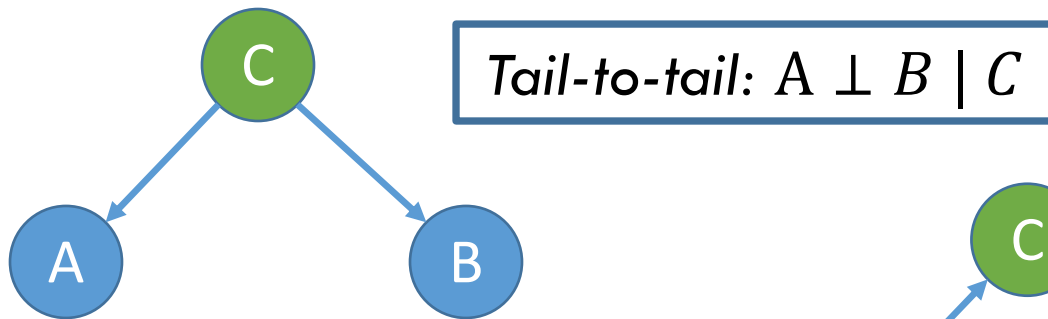
Head-to-Head Example

- Say $A \sim \text{Bern}\left(\frac{1}{2}\right), B \sim \text{Bern}\left(\frac{1}{2}\right)$
- Say $C = 1$ if $A = B$ and 0 otherwise
- Conditioned on C
 - If we know A, we know B.
 - They are dependent!
 - Similarly, if we know B, we know A.
- Hence, $A \not\perp B \mid C$ (i.e., not true for every distribution that respects the graph)
- But unconditionally, $A \perp B$
 - $P(a, b) = \sum_c P(a, b, c) = \sum_c P(a)P(b)P(c|a, b)$
 - $= P(a)P(b) \sum_c P(c|a, b) = P(a)P(b)$



Conditional Independence: Summary for 3 Node Graphical models

- Given a graphical model, we can determine if two sets of RVs are conditionally independent or not



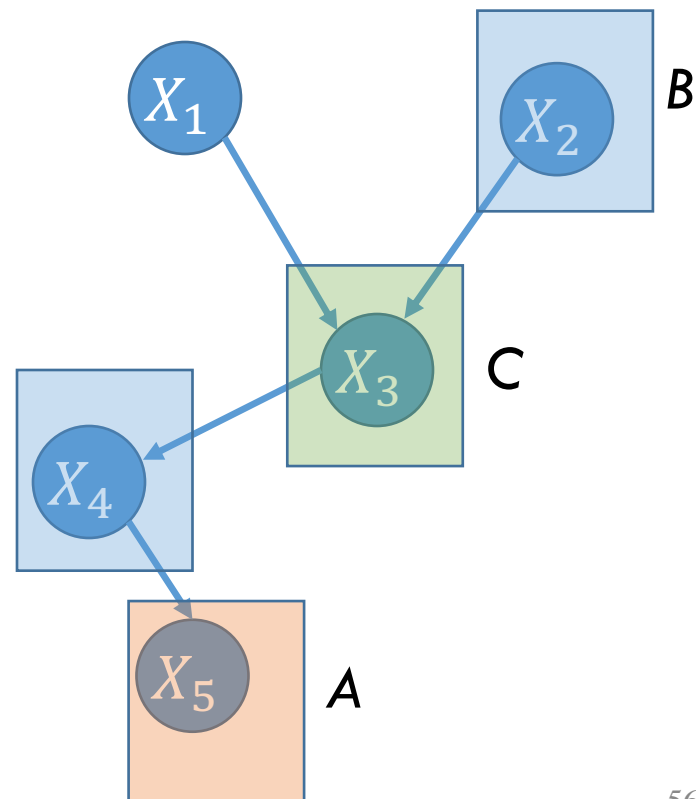
D-Separation Criterion: N Node Setting

- We saw how conditional independence properties unfold due to graph structure
 - This was only for three node graphs
- We will now move to larger DAGs
- We will look at the general idea of d-separation

D-Separation (I)

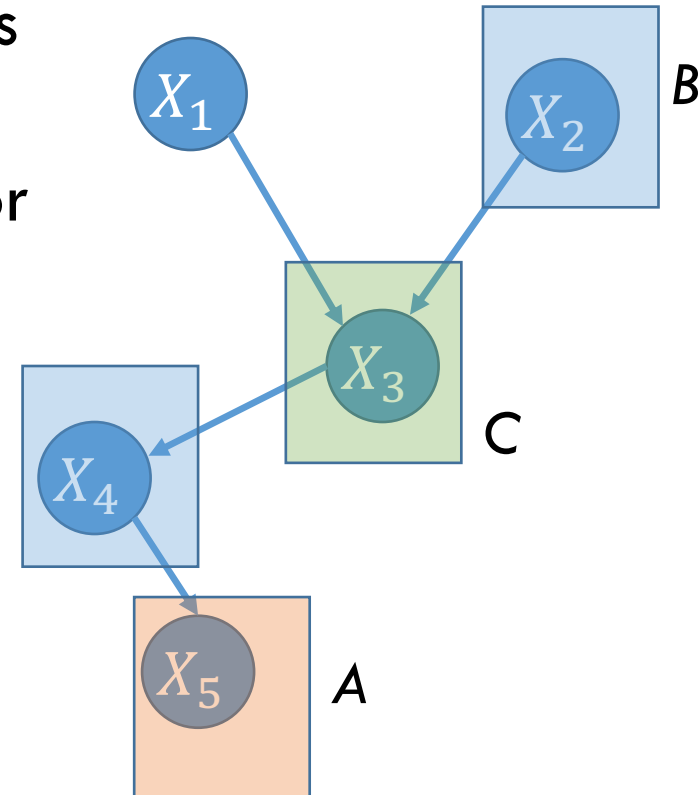
- Helps you read off the conditional independence properties

- Notation
 - Sets of RVs A, B and C
 - Disjoint
 - Not necessarily covering all



D-Separation (II)

- A path between two vertices is **blocked** with respect to \mathcal{C} if it passes through a node v such that
 - $v \in \mathcal{C}$, arrows are head-to-tail or tail-to-tail
 - OR, $v \notin \mathcal{C}$, arrows are head-to-head, and $\text{Descendants}(v) \notin \mathcal{C}$
- Example
 - X_4, X_3 and X_2 are in head-tail
 - So path is blocked

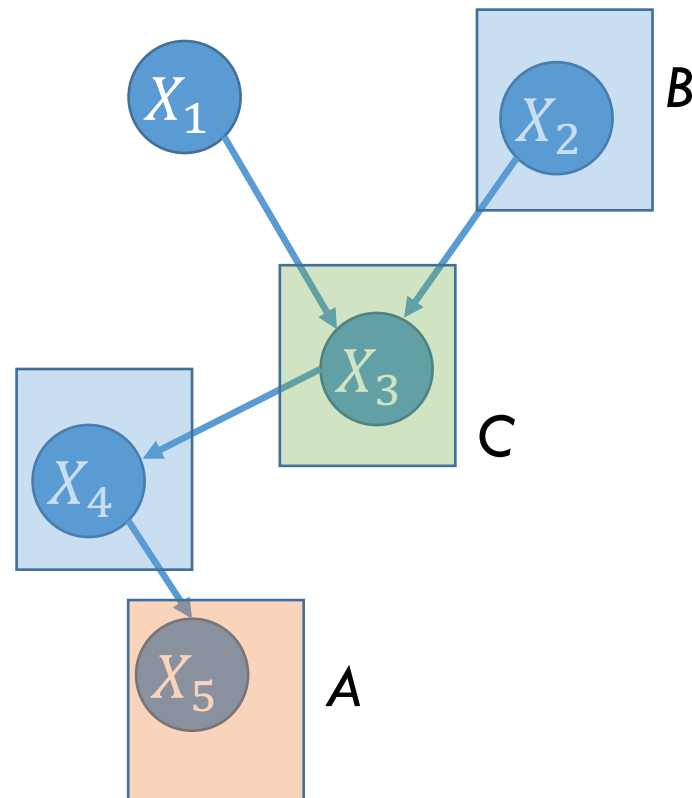


D-Separation (III)

- Definition of D-Separation
 - A and B are d-separated by C if all paths from vertices in A to vertices in B are blocked with respect to C
- Key result
 - If A and B are d-separated by C, then $A \perp B \mid C$
- Note: the above result is only 'necessary' not 'sufficient'

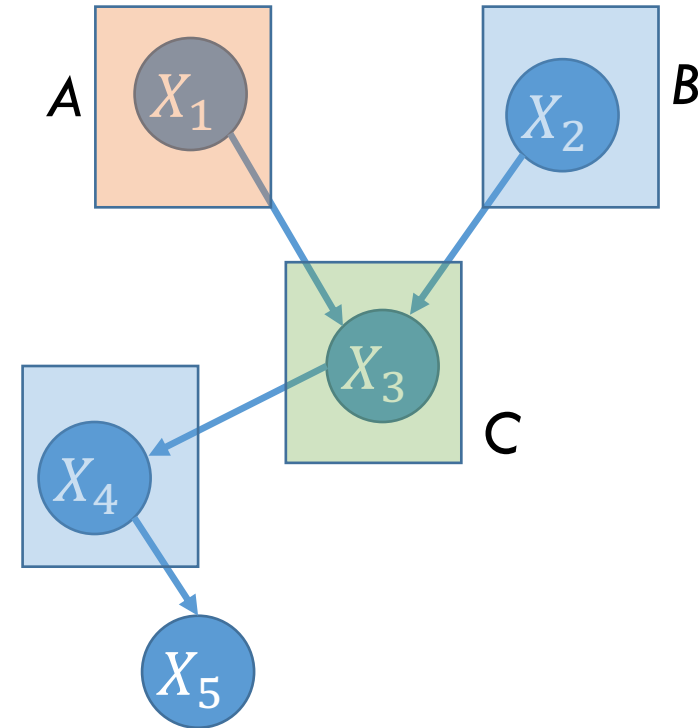
D-Separation Example I

- Let $C = \{X_3\}$
- Is $A \perp B \mid C$?
- We can check that by checking d-separation for all pairs of vertices $X_i \perp X_j \mid C$?
 - $i = \{5\}$
 - $j = \{2,4\}$
- Easy to see that
 - X_2, X_5 are blocked by C
 - X_4, X_5 are not blocked by C
- Hence, not d-separated
- Hence cannot say $A \perp B \mid C$



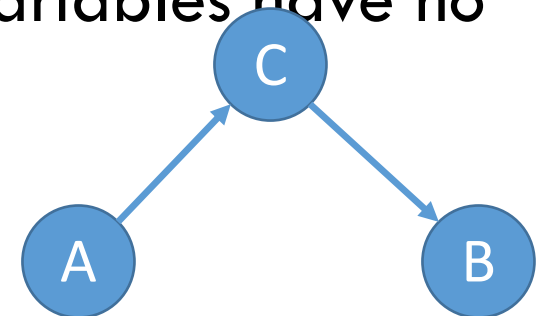
D-Separation Example II

- Let $C = \{X_3\}$
- Is $A \perp B \mid C$?
- We can check that by checking d-separation for all pairs of vertices $X_i \perp X_j \mid C$?
 - $i = \{1\}$
 - $j = \{2,4\}$
- We can see that
 - X_1, X_2 are not blocked by C
 - X_1, X_4 are blocked by C
- Hence, not d-separated
- Hence cannot say $A \perp B \mid C$



DAG and Probability (I)

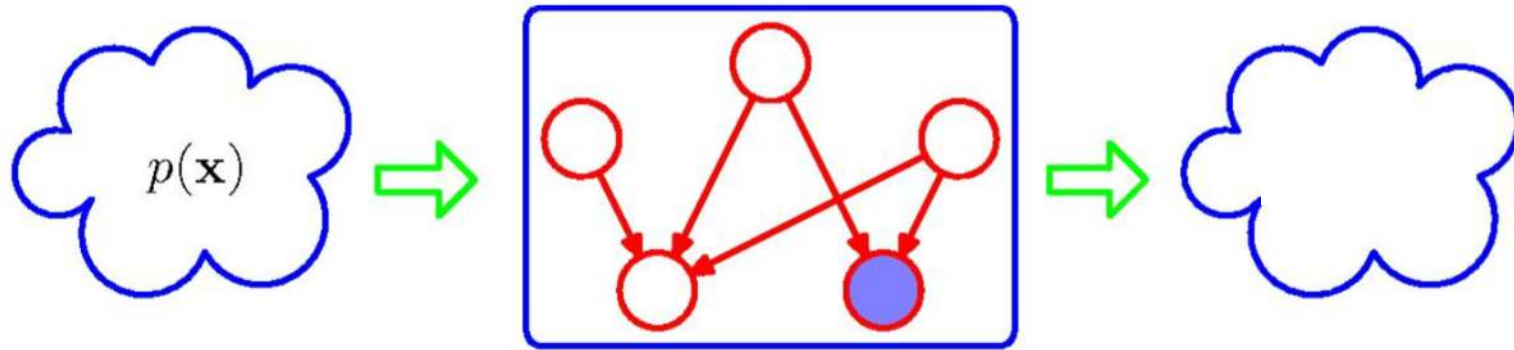
- We have showed that the structure of the DAG corresponds to a set of conditional independence assumptions
 - We can read conditional independence easily!
- We just need to specify $P(X_i|pa(X_i))$
- This does not mean that non-parent variables have no influence
 - Thus, the DAG does not imply
 - $P(c|a, b) = P(c|a)$



DAG and Probability (II)

- DPGMs are good for representing independence, not for representing dependence
- We have seen this
 - Multiple graphs for the same distribution
 - D-separation only says conditional independence if true. If not true, then no conclusion is drawn.

Filter view of DPGM



- Only distributions that satisfy conditional independences are allowed to pass
- One graph can describe many probability distributions
- Edge cases:
 - When DAG is fully connected, all distributions pass
 - When DAG is fully disconnected, only the product distribution ($\prod_i P(X_i)$) passes

Continuous Distributions

- We never had to state whether $P(X|Y)$ was continuous or discrete
- The graph is agonistic to the support of the random variables!

Questions?

Today's Outline

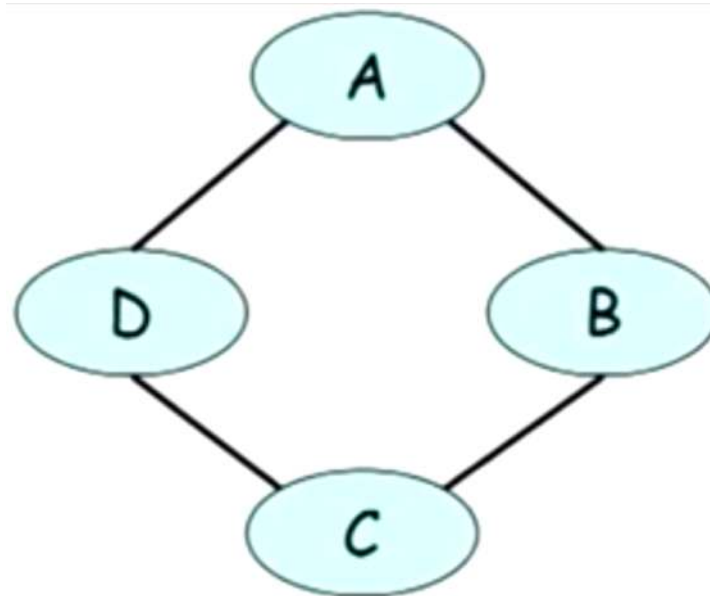
- Motivation
- Primer on Graphs
- Directed Graphical Models
- Undirected Graphical Models

Undirected (Probabilistic) Graphical Models

Based on notes from Bjoern Andres and Bernt Schiele (2016)

UPGM

- Also called Markov Networks or Markov Random Fields
- No edge directions
- Again, diagrams of probability distributions that capture conditional independences

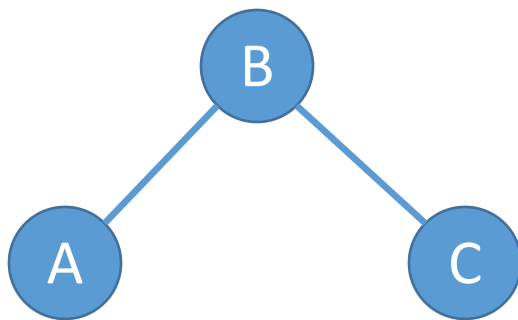


UPGM vs DPGM

- DPGMs have been used in data analytics, ML, statistics
- UPGMs have been used in computer vision and physics, and have applications in data analytics as well
- DPGM
 - Factor of the distribution was a (cond.) distribution
- UPGM
 - Factor (also called **potential**) need not be a distribution
 - Let $P(a, b, c) = \frac{1}{Z} \phi_1(a, b) \phi_2(b, c)$
 - Here Z is the normalization constant or **partition function**.
$$Z = \sum_{a,b,c} \phi_1(a, b) \phi_2(b, c)$$

Notion of a Potential

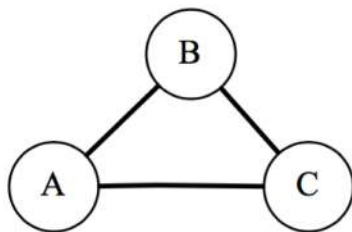
- Potential $\phi(x)$ is a non-negative function of variable x . Joint potential $\phi(x_1, \dots, x_D)$ is a non-negative function of a set of variables.
- Let $P(a, b, c) = \frac{1}{Z} \phi_1(a, b) \phi_2(b, c)$



Potentials Over Cliques

- For RVs X_1, \dots, X_D , an UPGM is defined as a product of potentials over the **cliques** of graph G
- $P(X_1, \dots, X_D) = \frac{1}{Z} \prod_c \phi_c(\mathcal{X}_c)$
 - Here $Z = \sum_{x_1, \dots, x_D} \prod_c \phi_c(\{x_i: X_i \in \mathcal{X}_c\})$
- Special cases:
 - When cliques are of size 2: the UPGM is called a pairwise UPGM
 - When all potentials are strictly positive: the distribution is called a Gibbs distribution

Example Potentials



$$\phi_{A,B}(a,b) = \begin{array}{c} \text{B} \\ 0 \quad 1 \\ \begin{array}{c} \text{A} \\ 0 \\ 1 \end{array} \end{array} \begin{array}{|c|c|} \hline 10 & 1 \\ \hline 1 & 10 \\ \hline \end{array}$$

$$\phi_{B,C}(b,c) = \begin{array}{c} \text{C} \\ 0 \quad 1 \\ \begin{array}{c} \text{B} \\ 0 \\ 1 \end{array} \end{array} \begin{array}{|c|c|} \hline 10 & 1 \\ \hline 1 & 10 \\ \hline \end{array}$$

$$\phi_{A,C}(a,c) = \begin{array}{c} \text{C} \\ 0 \quad 1 \\ \begin{array}{c} \text{A} \\ 0 \\ 1 \end{array} \end{array} \begin{array}{|c|c|} \hline 10 & 1 \\ \hline 1 & 10 \\ \hline \end{array}$$

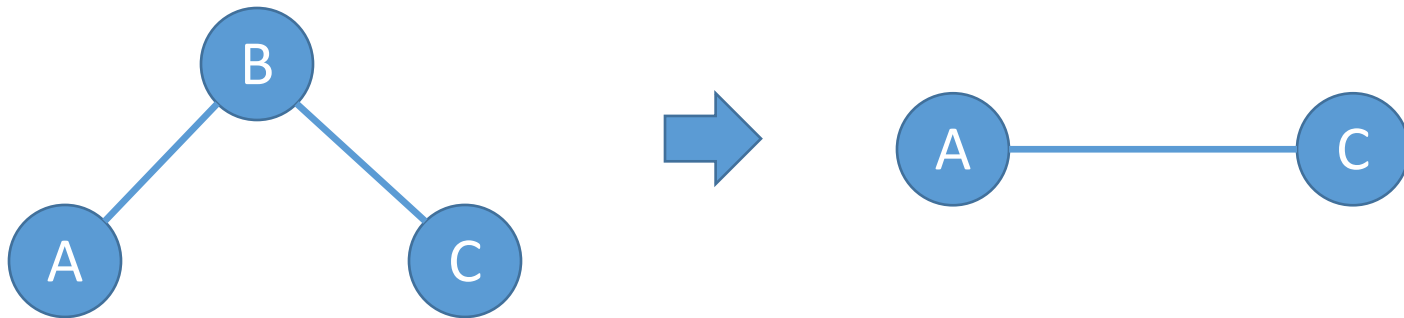
$$p(a, b, c) = \frac{1}{Z} \phi_{A,B}(a, b) \cdot \phi_{B,C}(b, c) \cdot \phi_{A,C}(a, c),$$

where

$$Z = \sum_{\hat{a}, \hat{b}, \hat{c} \in \{0,1\}^3} \phi_{A,B}(\hat{a}, \hat{b}) \cdot \phi_{B,C}(\hat{b}, \hat{c}) \cdot \phi_{A,C}(\hat{a}, \hat{c}) = 2 \cdot 1000 + 6 \cdot 10 = 2060.$$

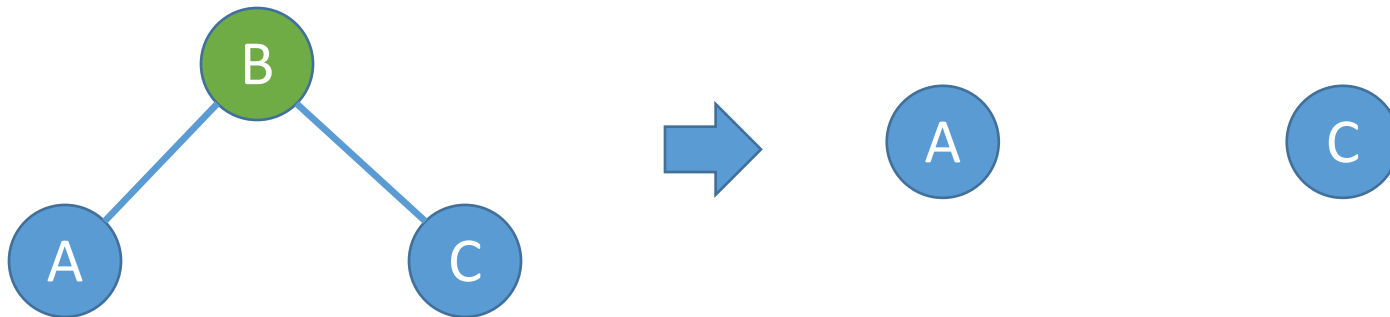
Marginalization

- Marginalizing over B makes A and C graphically dependent
- $P(a, c) = \sum_b P(a, b, c) = \frac{1}{Z} \phi_3(a, c)$



Conditional Independence (I)

- Conditioning on B makes A and C independent
- $P(a, c|b) = P(a|b)P(c|b)$



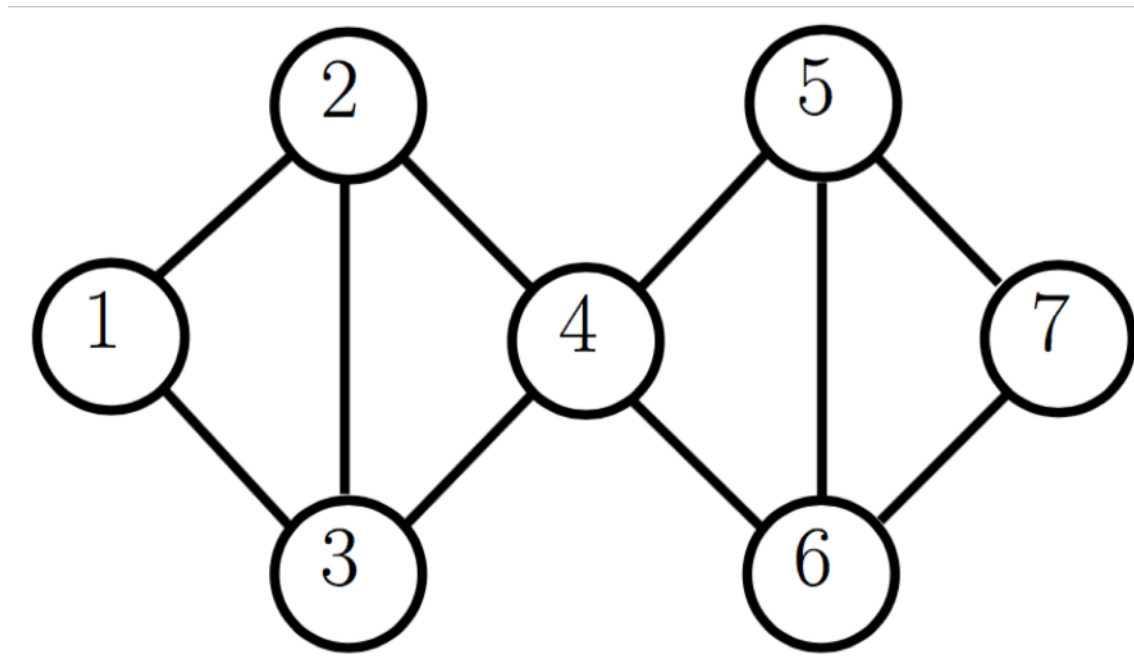
- Key: This is different from the head-to-head directed graph example, where conditioning introduced dependency!

Conditional Independence (II)

- Global Markov property
 - Two sets of nodes (say A and B) are conditionally independent given a third set C if
 - All nodes in A and B are connected through nodes in C
- Local Markov property
 - Conditioning on the neighbors of X makes X independent of the rest of the graph.
 - $P(X_i | X_1, \dots, X_{i-1}, X_{i+1}, \dots, X_D) = P(X_i | nbhd(X_i))$

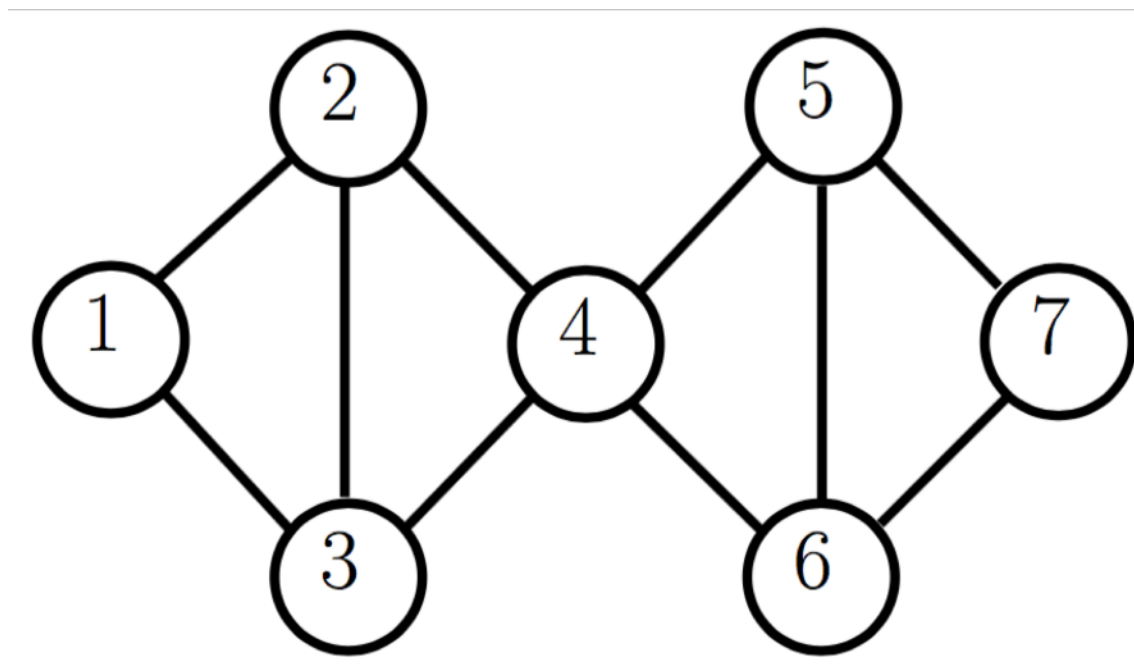
Global Markov Property

- In the following graph G , as a consequence of global Markov property:
 - $\{X_1, X_2, X_3\} \perp \{X_5, X_6, X_7\} | X_4$



Local Markov Property

- In the following graph G , as a consequence of local Markov property:
 - $X_4 \perp \{X_1, X_7\} | \{X_2, X_3, X_5, X_6\}$
 - $X_1 \perp \{X_4, X_5, X_6, X_7\}$

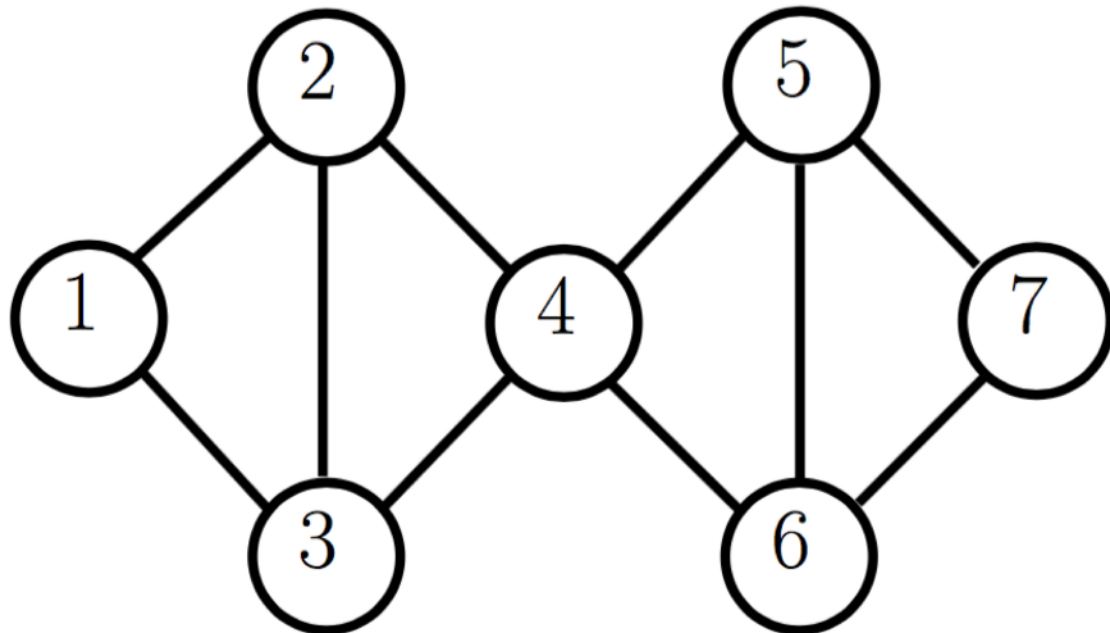


Graph to Distribution

- So, the undirected graph specifies a set of conditional independence statements
- We can write down a joint distribution using the graph
- For example, we may consider a factorization involving maximal cliques.

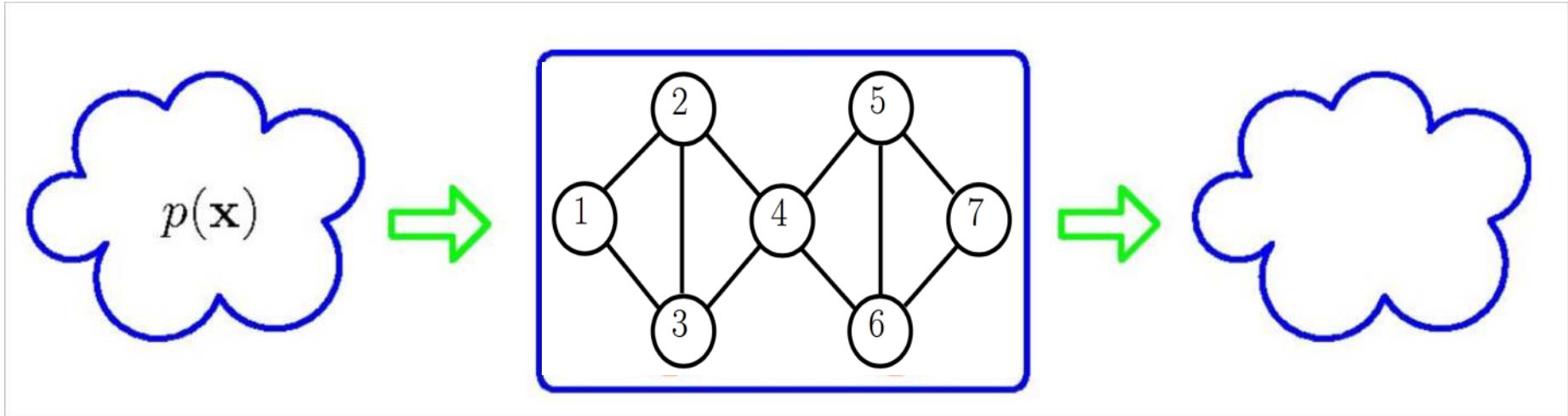
Graph to Distribution

- $P(x_1, \dots, x_7) = \frac{1}{Z} \phi_1(x_1, x_2, x_3) \phi_2(x_2, x_3, x_4) \phi_3(x_4, x_5, x_6) \phi_4(x_5, x_6, x_7)$



- But, we could have also considered some other factorization

Filter view of UPGM



- Only distributions that satisfy conditional independences are allowed to pass

Limitations of DPGM and UPGM

- Cannot always represent all conditional independences of a given joint distribution
- Example: we cannot draw a DPGM for the following distribution
 - $P(A, B, C, D)$ with $A \perp C | \{B, D\}$ and $B \perp D | \{A, C\}$
- Another example: we cannot represent the following using a UPGM
 - $P(A, B, C)$ with $A \not\perp C | \{B\}$ and $A \perp C$
- Homework: verify the above two statements!

DPGM vs UPGM

Property	UPGMs	DPGMs
Form	Prod. potentials	Prod. potentials
Potentials	Arbitrary	Cond. probabilities
Cycles	Allowed	Forbidden
Partition func.	$Z = ?$	$Z = 1$
Indep. check	Graph separation	D-separation
Indep. props.	Some	Some
Inference	MCMC, BP, etc.	Convert to UPGM

Questions?

Summary

- What are graphical models good at?
 - Capture complexity and uncertainty
 - Capture conditional independences
 - We can visualize what's happening with a distribution
- They unify many probabilistic techniques: mixture models, factor analysis, hidden Markov models, Kalman filters etc.
- Today we saw: visualization, conditional independence properties
- Next: computations (**inference** and **learning**)

Sample Exam Questions

- What is the need for graphical models?
- What is the significance of 'hidden' and 'Markov' in a HMM?
- What is the use of a Latent Dirichlet Allocation model?
- What is a clique?
- Which distributions respect a graph?
- What is the difference between a head-to-head and a tail-to-tail configuration in DPGMs?
- How is the factorization in a UPGM different from the factorization in a DPGM?
- How would you find conditional independence relationships in a UPGM?

Appendix

Additional Resources

- Book 1: *Graphical models, exponential families, and variational inference* by Martin J. Wainwright and Michael I. Jordan
 - See https://people.eecs.berkeley.edu/~wainwrig/Papers/WaiJor08_FTML.pdf
- Book 2: *Bayesian Reasoning and Machine Learning* by David Barber
 - See <http://web4.cs.ucl.ac.uk/staff/D.Barber/pmwiki/pmwiki.php?n=Brml.Online>

Review: Probability

Based on Sam Roweis's slides (2002)

Probability

- We use probabilities $p(x)$ to represent our beliefs $B(x)$ about the states x of the world.
- There is a formal calculus for manipulating uncertainties represented by probabilities.

Probability

- We use probabilities $p(x)$ to represent our beliefs $B(x)$ about the states x of the world.
- There is a formal calculus for manipulating uncertainties represented by probabilities.
- Any consistent set of beliefs obeying the *Cox Axioms* can be mapped into probabilities.
 1. Rationally ordered degrees of belief:
if $B(x) > B(y)$ and $B(y) > B(z)$ then $B(x) > B(z)$
 2. Belief in x and its negation \bar{x} are related: $B(x) = f[B(\bar{x})]$
 3. Belief in conjunction depends only on conditionals:
 $B(x \text{ and } y) = g[B(x), B(y|x)] = g[B(y), B(x|y)]$

Probability

- An **outcome space** specifies the possible outcomes that we would like to reason about, e.g.

$$\Omega = \{ \text{, \text{} \} \quad \text{Coin toss}$$

$$\Omega = \{ \text{, , , , , } \} \quad \text{Die toss}$$

- We specify a **probability** $p(\omega)$ for each outcome ω such that

$$p(\omega) \geq 0, \quad \sum_{\omega \in \Omega} p(\omega) = 1$$

$$\text{E.g., } p(\text{) = .6$$

$$p(\text{) = .4$$

Probability

- An **event** is a subset of the outcome space, e.g.

$$E = \{ \text{die 1}, \text{die 2}, \text{die 3} \} \quad \text{Even die tosses}$$

$$O = \{ \text{die 4}, \text{die 5}, \text{die 6} \} \quad \text{Odd die tosses}$$

- The **probability** of an event is given by the sum of the probabilities of the outcomes it contains,

$$p(E) = \sum_{\omega \in E} p(\omega)$$

$$\begin{aligned} \text{E.g., } p(E) &= p(\text{die 1}) + p(\text{die 2}) + p(\text{die 3}) \\ &= 1/2, \text{ if fair die} \end{aligned}$$

Random Variables

- Random variables X represents outcomes or states of world.
Instantiations of variables usually in lower case: x
We will write $p(x)$ to mean probability($X = x$).
- Sample Space: the space of all possible outcomes/states.
(May be discrete or continuous or mixed.)

Random Variables

- Random variables X represents outcomes or states of world.
Instantiations of variables usually in lower case: x
We will write $p(x)$ to mean $\text{probability}(X = x)$.
- Sample Space: the space of all possible outcomes/states.
(May be discrete or continuous or mixed.)
- Probability mass (density) function $p(x) \geq 0$
Assigns a non-negative number to each point in sample space.
Sums (integrates) to unity: $\sum_x p(x) = 1$ or $\int_x p(x)dx = 1$.
Intuitively: how often does x occur, how much do we believe in x .
- Ensemble: random variable + sample space + probability function

Expectation

- Expectation of a function $a(x)$ is written $E[a]$ or $\langle a \rangle$

$$E[a] = \langle a \rangle = \sum_x p(x) a(x)$$

e.g. mean = $\sum_x x p(x)$, variance = $\sum_x (x - E[x])^2 p(x)$

Expectation

- Expectation of a function $a(x)$ is written $E[a]$ or $\langle a \rangle$

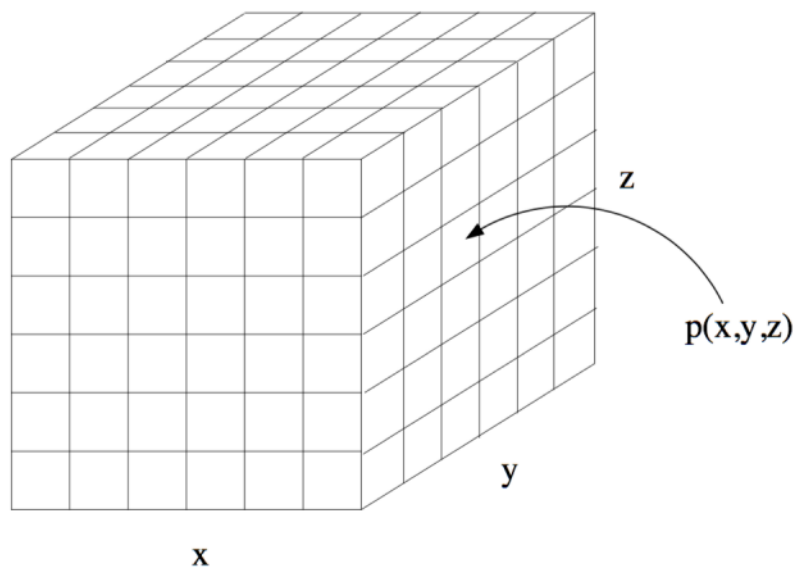
$$E[a] = \langle a \rangle = \sum_x p(x) a(x)$$

e.g. mean = $\sum_x x p(x)$, variance = $\sum_x (x - E[x])^2 p(x)$

- Moments are expectations of higher order powers.
(Mean is first moment. Autocorrelation is second moment.)
- Centralized moments have lower moments subtracted away
(e.g. variance, skew, kurtosis).
- Deep fact: Knowledge of all orders of moments completely defines the entire distribution.

Joint Probability

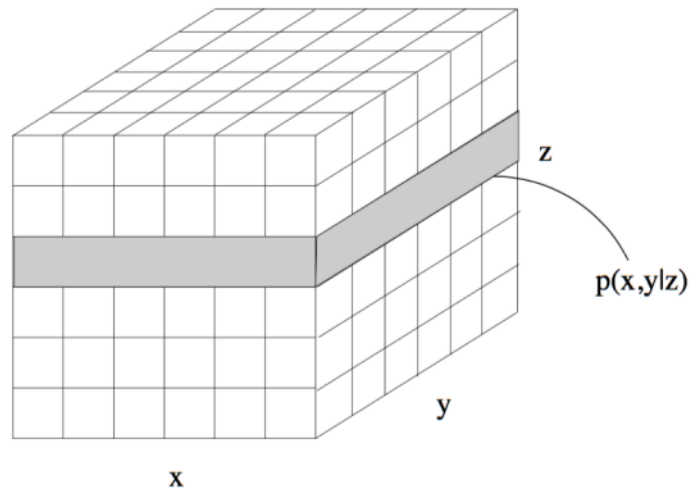
- Key concept: two or more random variables may interact. Thus, the probability of one taking on a certain value depends on which value(s) the others are taking.
- We call this a joint ensemble and write
$$p(x, y) = \text{prob}(X = x \text{ and } Y = y)$$



Conditional Probability

- If we know that some event has occurred, it changes our belief about the probability of other events.
- This is like taking a "slice" through the joint table.

$$p(x|y) = p(x, y) / p(y)$$

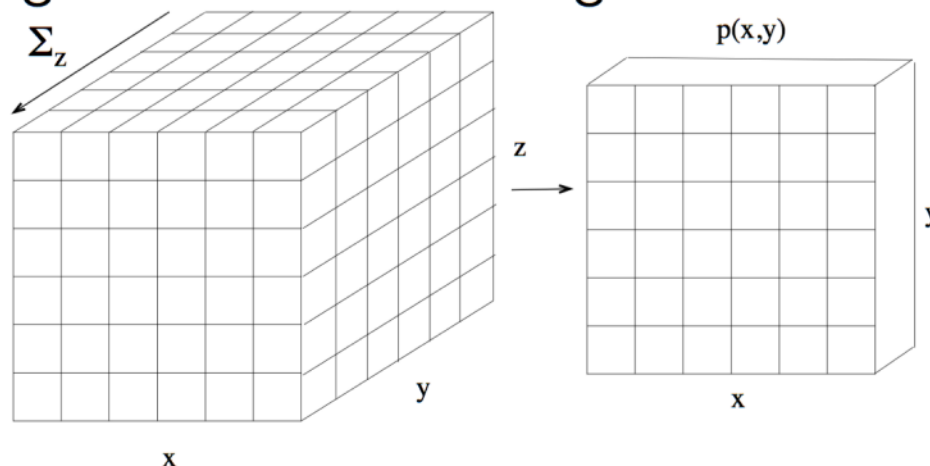


Marginal Probability

- We can "sum out" part of a joint distribution to get the *marginal distribution* of a subset of variables:

$$p(x) = \sum_y p(x, y)$$

- This is like adding slices of the table together.



- Another equivalent definition: $p(x) = \sum_y p(x|y)p(y)$.

Bayes Rule

- Manipulating the basic definition of conditional probability gives one of the most important formulas in probability theory:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} = \frac{p(y|x)p(x)}{\sum_{x'} p(y|x')p(x')}$$

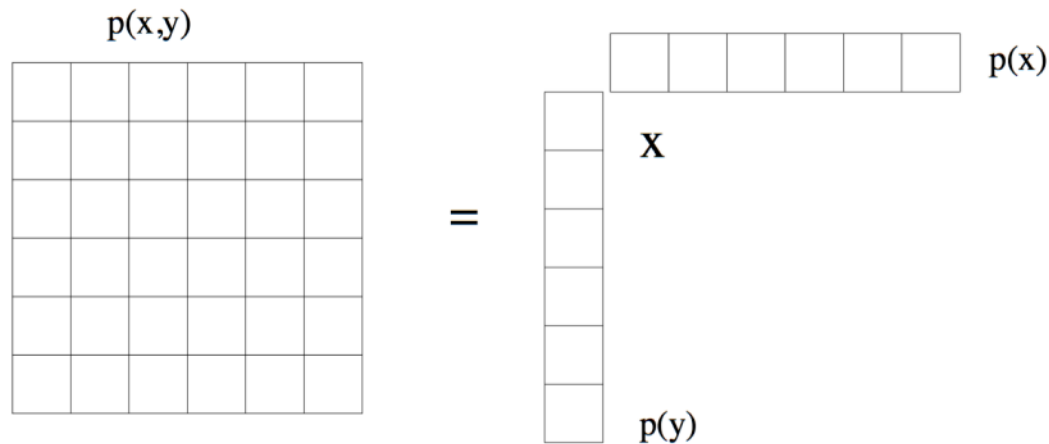
- This gives us a way of "reversing" conditional probabilities.
- Thus, all joint probabilities can be factored by selecting an ordering for the random variables and using the "chain rule":

$$p(x, y, z, \dots) = p(x)p(y|x)p(z|x, y)p(\dots |x, y, z)$$

Conditional Independence

- Two variables are independent iff their joint factors:

$$p(x, y) = p(x)p(y)$$



- Two variables are conditionally independent given a third one if for all values of the conditioning variable, the resulting slice factors:

$$p(x, y|z) = p(x|z)p(y|z) \quad \forall z$$

Independent Event Examples

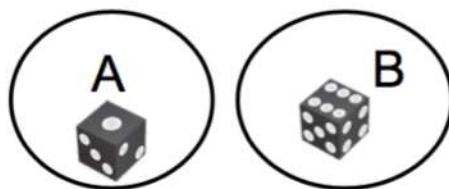
- Independent event example
 - Hardware failures events in different data centers
- Dependent event examples
 - Queries to a search engine and news
 - Tweets and news
 - IM and email communications

Independent Event Examples

- Two events A and B are **independent** if

$$p(A \cap B) = p(A)p(B)$$

- Are these two events independent?



No! $p(A \cap B) = 0, \quad p(A)p(B) = \left(\frac{1}{6}\right)^2$

- Now suppose our outcome space had two different die:

$$\Omega = \{ \text{brown die}, \text{blue die}, \text{brown die}, \text{blue die}, \text{brown die}, \text{blue die}, \dots, \text{brown die}, \text{blue die} \} \quad \text{2 die tosses}$$

$6^2 = 36 \text{ outcomes}$

and the probability distribution is such that each die is independent,

$$p(\text{brown die}, \text{blue die}) = p(\text{brown die}) p(\text{blue die})$$

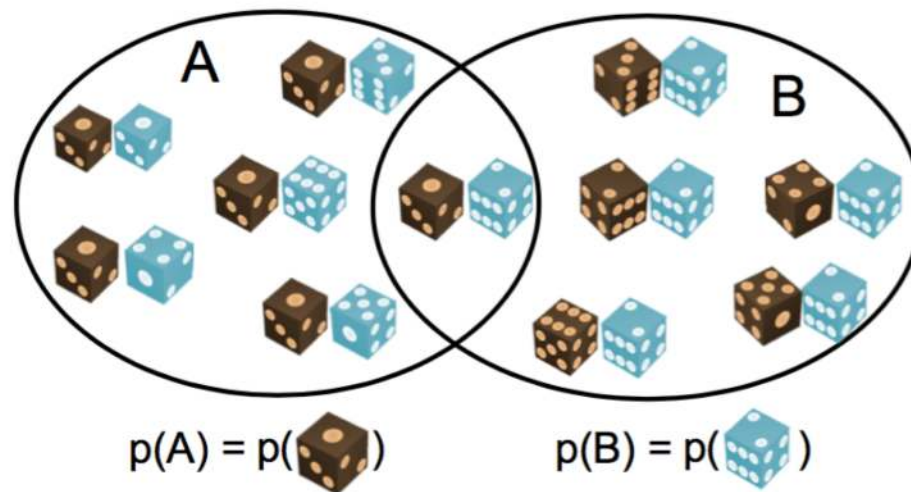
$$p(\text{brown die}, \text{blue die}) = p(\text{brown die}) p(\text{blue die})$$

Independent Event Examples

- Two events A and B are **independent** if

$$p(A \cap B) = p(A)p(B)$$

- Are these two events independent?



Yes!

$$p(A \cap B) = p(\text{brown die with 1 on top and blue die with 1 on side})$$

$$p(A)p(B) = p(\text{brown die with 1 on top}) p(\text{blue die with 1 on side})$$

Relation to Statistics

- Probability: inferring probabilistic quantities for data given fixed models (e.g. prob. of events, marginals, conditionals, etc).

Relation to Statistics

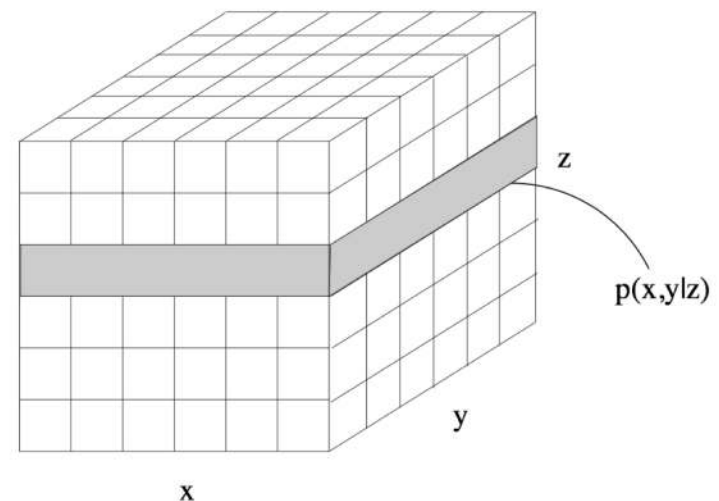
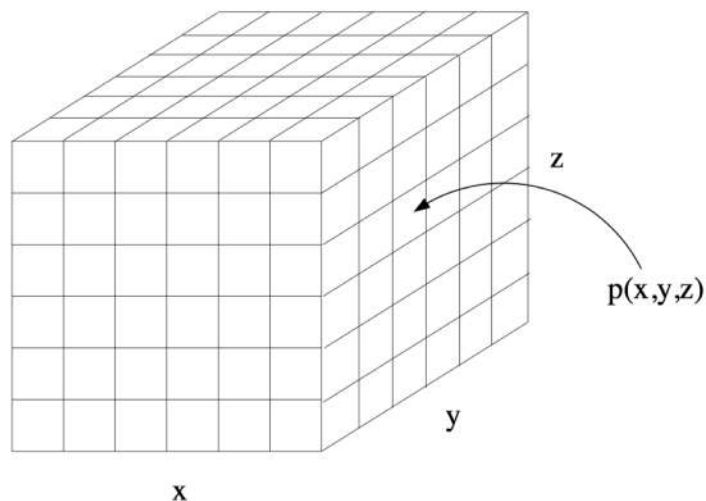
- Probability: inferring probabilistic quantities for data given fixed models (e.g. prob. of events, marginals, conditionals, etc).
- Statistics: inferring a model given fixed data observations (e.g. clustering, classification, regression).
- Many approaches to statistics:
frequentist, Bayesian, decision theory, ...

Conditional Probability Table

- For discrete (categorical) quantities, the most basic parametrization is the probability table which lists $p(x_i = k^{th} \text{ value})$.
- Since PTs must be nonnegative and sum to 1, for k -ary variables there are $k - 1$ free parameters.

Conditional Probability Table

- For discrete (categorical) quantities, the most basic parametrization is the probability table which lists $p(x_i = k^{th} \text{ value})$.
- Since PTs must be nonnegative and sum to 1, for k -ary variables there are $k - 1$ free parameters.
- If a discrete variable is conditioned on the values of some other discrete variables we make one table for each possible setting of the parents: these are called *conditional probability tables* or CPTs.



Likelihood Function

- So far we have focused on the (log) probability function $p(\mathbf{x}|\theta)$ which assigns a probability (density) to any joint configuration of variables \mathbf{x} given fixed parameters θ .
- But in learning we turn this on its head: we have some fixed data and we want to find parameters.
- Think of $p(\mathbf{x}|\theta)$ as a function of θ for fixed \mathbf{x} :

$$L(\theta; \mathbf{x}) = p(\mathbf{x}|\theta)$$

$$\ell(\theta; \mathbf{x}) = \log p(\mathbf{x}|\theta)$$

This function is called the (log) “likelihood”.

- Chose θ to maximize some cost function $c(\theta)$ which includes $\ell(\theta)$:

$$c(\theta) = \ell(\theta; \mathcal{D}) \quad \text{maximum likelihood (ML)}$$

$$c(\theta) = \ell(\theta; \mathcal{D}) + r(\theta) \quad \text{maximum a posteriori (MAP)/penalized ML}$$

(also cross-validation, Bayesian estimators, BIC, AIC, ...)

Complete Data, IID Sampling

- A single observation of the data \mathbf{X} is rarely useful on its own.
- Generally we have data including many observations, which creates a *set of random variables*: $\mathcal{D} = \{\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^M\}$
- Two very common assumptions:
 1. Observations are independently and identically distributed according to joint distribution of graphical model: IID samples.
 2. We observe all random variables in the domain on each observation: complete data.

Maximum Likelihood

- For IID data:

$$p(\mathcal{D}|\theta) = \prod_m p(\mathbf{x}^m|\theta)$$

$$\ell(\theta; \mathcal{D}) = \sum_m \log p(\mathbf{x}^m|\theta)$$

- Idea of maximum likelihood estimation (MLE): pick the setting of parameters most likely to have generated the data we saw:

$$\theta_{\text{ML}}^* = \operatorname{argmax}_{\theta} \ell(\theta; \mathcal{D})$$

- Very commonly used in statistics.
Often leads to “intuitive”, “appealing”, or “natural” estimators.

What to do with a Distribution

- *Generate data*: draw samples from the distribution. This often involves generating a uniformly distributed variable in the range $[0,1]$ and transforming it. For more complex distributions it may involve an iterative procedure that takes a long time to produce a single sample (e.g. Gibbs sampling, MCMC).
- *Compute log probabilities*.
When all variables are either observed or marginalized the result is a single number which is the log prob of the configuration.

What to do with a Distribution

- *Generate data*: draw samples from the distribution. This often involves generating a uniformly distributed variable in the range $[0,1]$ and transforming it. For more complex distributions it may involve an iterative procedure that takes a long time to produce a single sample (e.g. Gibbs sampling, MCMC).
- *Compute log probabilities*.
When all variables are either observed or marginalized the result is a single number which is the log prob of the configuration.
- *Inference*: Compute expectations of some variables given others which are observed or marginalized.
- *Learning*.
Set the parameters of the density functions given some (partially) observed data to maximize likelihood or penalized likelihood.

Aside: Observed vs Hidden Variables

- Observed variables:
 - For example, inputs in regression or classification
- Unobserved variables:
 - Also called hidden or latent
 - Can be marginalized out
 - Can make the modeling of observed variables easier (e.g., Gaussian mixture models)