Bayesian Networks

Alan Ritter

Problem: Non-IID Data

- Most real-world data is not IID
  - (like coin flips)
- Multiple correlated variables
- Examples:
  - Pixels in an image
  - Words in a document
  - Genes in a microarray
- We saw one example of how to deal with this
  - Markov Models + Hidden Markov Models

Questions

- How to compactly represent $P(X|\theta)$?
- How can we use this distribution to infer one set of variables given another?
- How can we learn the parameters with a reasonable amount of data?

The Chain Rule of Probability

$$P(x_{1:N}) = P(x_1)P(x_2|x_1)P(x_3|x_1,x_2)P(x_4|x_1,x_2,x_3)\dots P(x_N|x_{1:N-1})$$

Problem: this distribution has 2^(N-1) parameters

- Can represent any joint distribution this way
- Using any ordering of the variables…

Conditional Independence

- This is the key to representing large joint distributions
- X and Y are conditionally independent given Z
  - if and only if the conditional joint can be written as a product of the conditional marginals

$$X \perp Y | Z \iff P(X, Y | Z) = P(X | Z) P(Y | Z)$$

(non-hidden) Markov Models

- "The future is independent of the past given the present"

$$x_{t+1} \perp x_{1:t-1} | x_t$$

$$P(x_1, x_2, x_3, \ldots, x_n)$$

$$= P(x_1)P(x_2|x_1)P(x_3|x_1, x_2) \ldots P(x_n|x_1, x_2, x_3, \ldots, x_{n-1})$$

$$= P(x_1)P(x_2|x_1)P(x_3|x_2) \ldots P(x_n|x_{n-1})$$

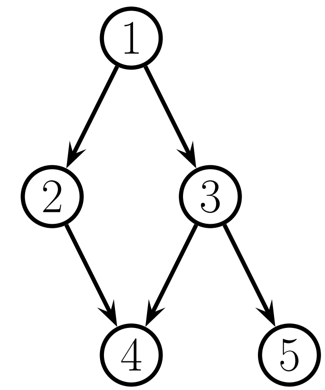
Graphical Models

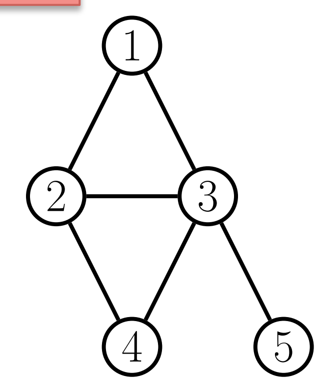
- First order Markov assumption is useful for 1d sequence data
  - Sequences of words in a sentence or document
- Q: What about 2d images, 3d video
  - Or in general arbitrary collections of variables
    - Gene pathways, etc…

Graphical Models

- A way to represent a joint distribution by making conditional independence assumptions
- Nodes represent variables
- (lack of) edges represent conditional independence assumptions
- Better name: "conditional independence diagrams"

Doesn't sound as cool





Graph Terminology

- Graph (V,E) consists of
  - A set of nodes or verticies V={1..V}
  - A set of edges {(s,t) in V}
- Child (for directed graph)
- Ancestors (for directed graph)
- Decedents (for directed graph)
- Neighbors (for any graph)
- Cycle (Directed vs. undirected)
- Tree (no cycles)
- Clique / Maximal Clique

Directed Graphical Models

- Graphical Model whose graph is a DAG
  - Directed acyclic graph
  - No cycles!
- A.K.A. Bayesian Networks
  - Nothing inherently Bayesian about them
    - Just a way of defining conditional independences
    - Just sounds cooler I guess…

Directed Graphical Models

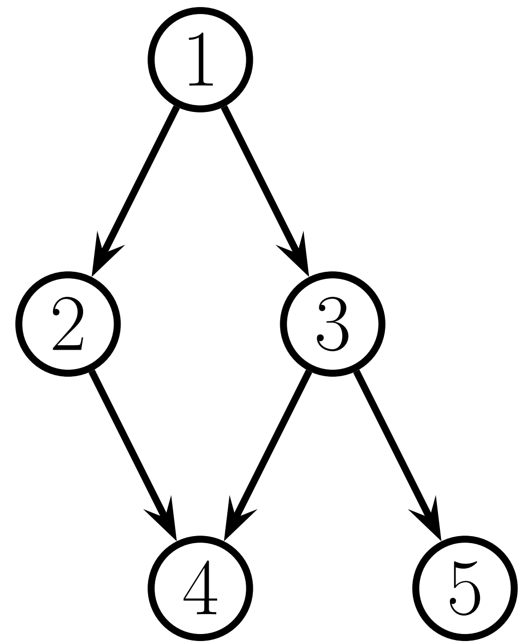
- Key property: Nodes can be ordered so that parents come before children
  - Topological ordering
  - Can be constructed from any DAG
- Ordered Markov Property:
  - Generalization of first-order Markov Property to general DAGs
  - Node only depends on it's parents (not other predecessors)

$$x_s \perp x_{\mathrm{pred}(s)-\mathrm{parents}(s)} \big| x_{\mathrm{parents(s)}}$$

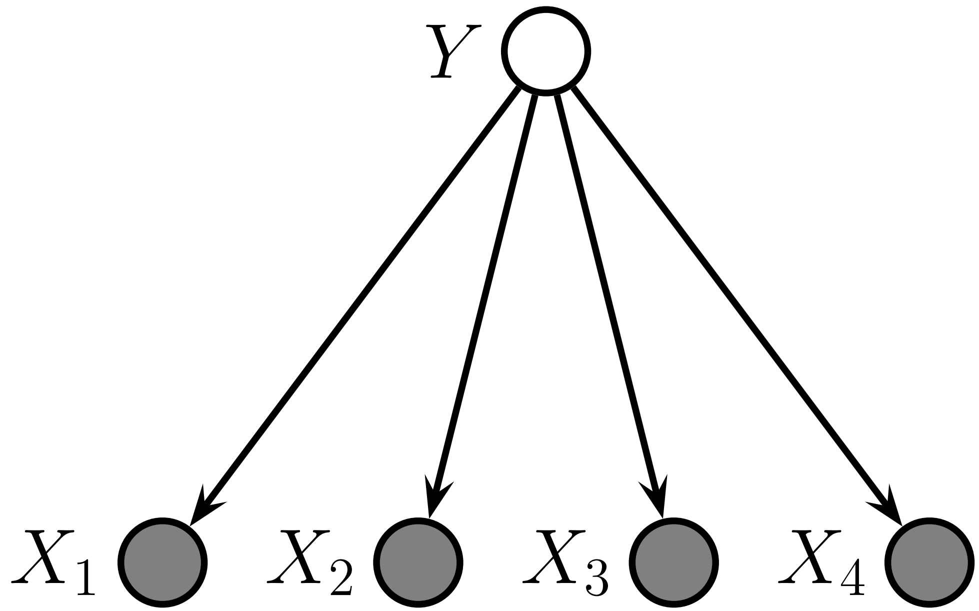
Example

$$P(x_{1:5}) = P(x_1)P(x_2|x_1)P(x_3|x_1, \mathbf{x_2})P(x_4|\mathbf{x_1}, x_2, x_3)p(x_5|\mathbf{x_1}, \mathbf{x_2}, x_3, \mathbf{x_4})$$

$$= P(x_1)P(x_2|x_1)P(x_3|x_1)P(x_4|x_2, x_3)p(x_5|x_3)$$



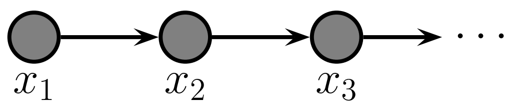
Naïve Bayes (Same as Gaussian Mixture Model w/ Diagonal Covariance)



$$P(y, x_{1:D}) = P(y) \prod_{j=1}^{D} P(x_j | y)$$

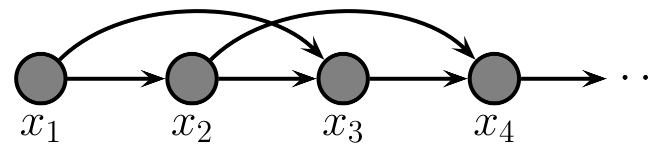
Markov Models

First order Markov Model



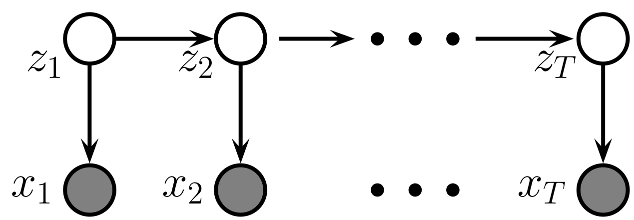
$$P(x_{1:N}) = P(x_1)\prod_{i=2}^{n} P(x_i|x_{i-1})$$

Second order Markov Model



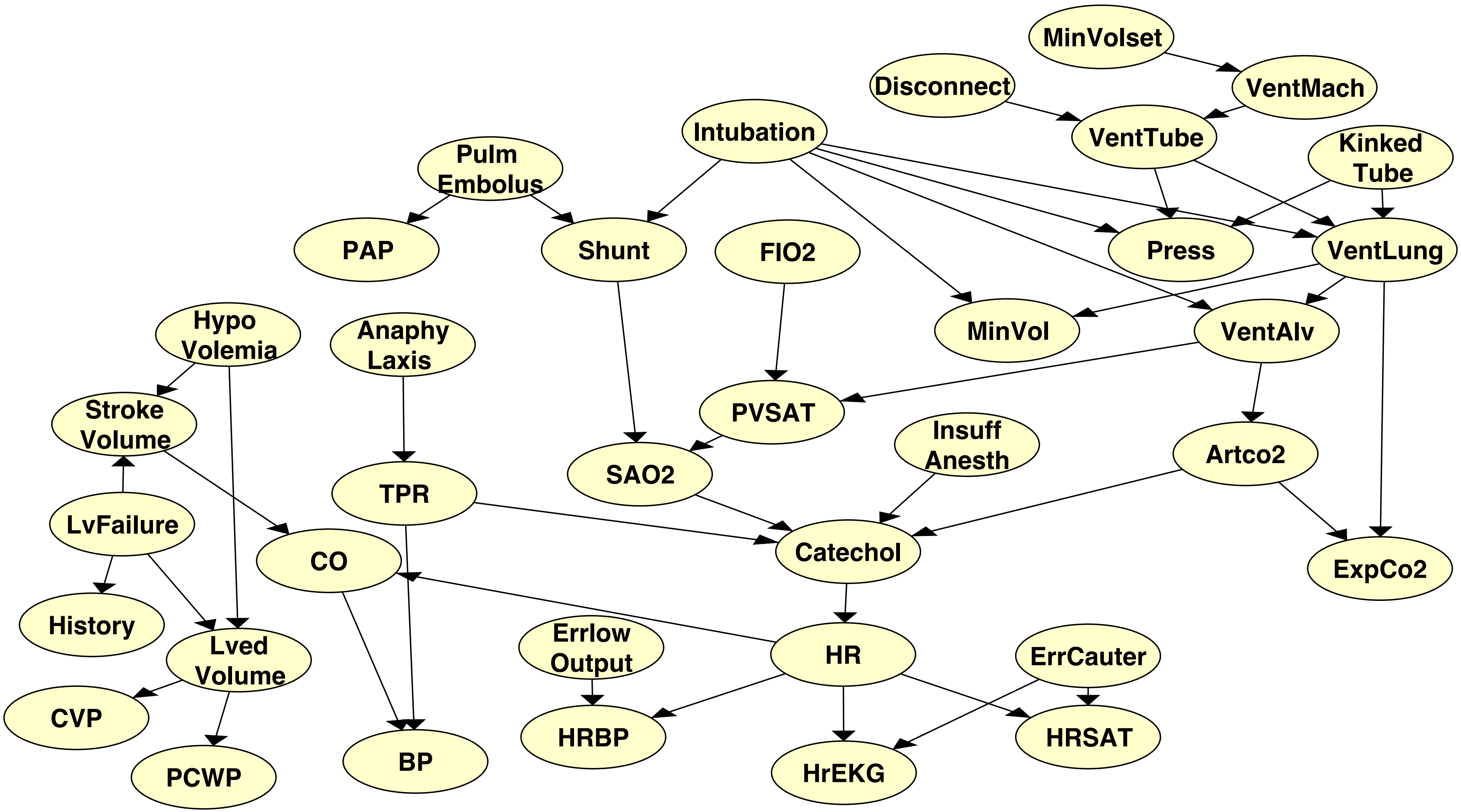
$$P(x_{1:N}) = P(x_1, x_2)\prod_{i=3}^{n} P(x_i|x_{i-1}, x_{i-2})$$

Hidden Markov Model

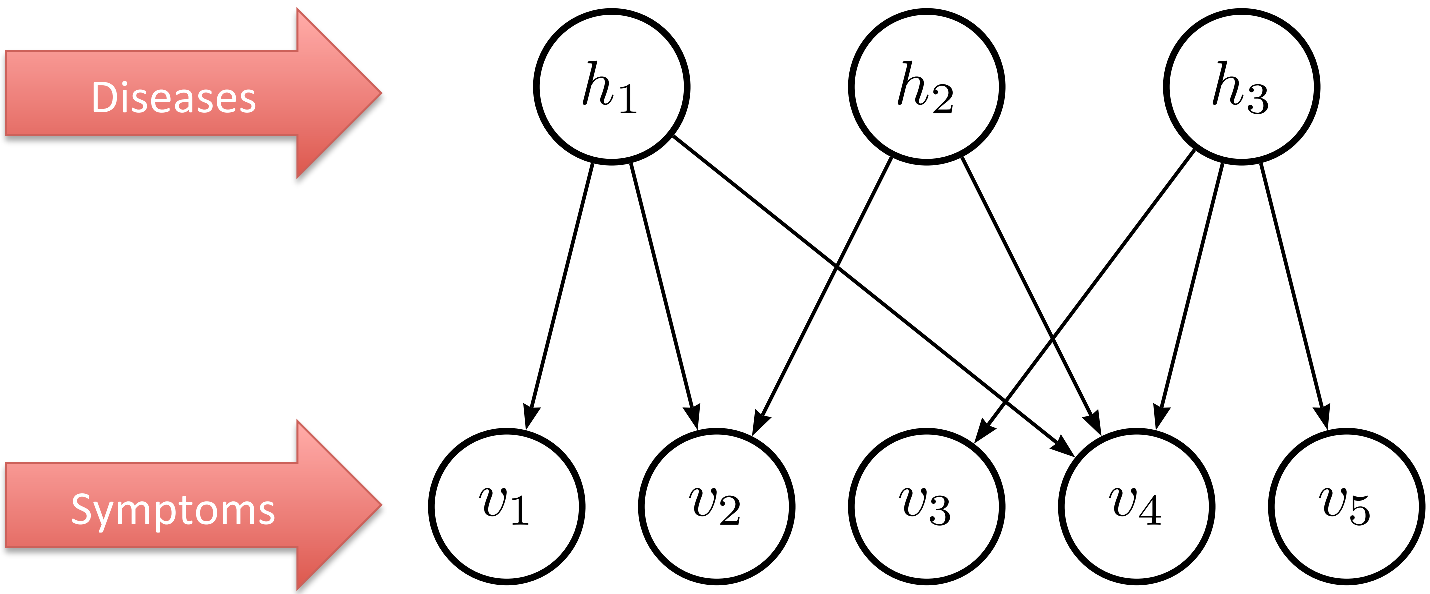


$$P(x_{1:N}) = P(z_1)P(x_1|z_1)\prod_{i=2}^{n} P(z_i|z_{i-1})P(x_i|z_i)$$

Example: medical Diagnosis
The Alarm Network



Another medical diagnosis example: QMR network



Compact conditional distributions contd.

Noisy-OR distributions model multiple noninteracting causes

1) Parents $U_1 \ldots U_k$ include all causes (can add leak node)

2) Independent failure probability q_i for each cause alone

$$\Rightarrow P(X|U_1 \ldots U_j, \neg U_{j+1} \ldots \neg U_k) = 1 - \Pi_{i=1}^{j} q_i$$

$Cold$	Flu	$Malaria$	$P(Fever)$	$P(\neg Fever)$
F	F	F	**0.0**	1.0
F	F	T	0.9	**0.1**
F	T	F	0.8	**0.2**
F	T	T	0.98	$0.02 = 0.2 \times 0.1$
T	F	F	0.4	**0.6**
T	F	T	0.94	$0.06 = 0.6 \times 0.1$
T	T	F	0.88	$0.12 = 0.6 \times 0.2$
T	T	T	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

Number of parameters **linear** in number of parents

Probabilistic Inference

- Graphical Models provide a compact way to represent complex joint distributions
- **Q:** Given a joint distribution, what can we do with it?
- **A:** Main use = Probabilistic Inference
  – Estimate unknown variables from known ones

Examples of Inference

- Predict the most likely cluster for X in R^n given a set of mixture components
  - This is what you did in HW #1
- Viterbi Algorithm, Forward/Backward (HMMs)
  - Estimate words from speech signal
  - Estimate parts of speech given sequence of words in a text

General Form of Inference

- We have:
  - A correlated set of random variables
  - Joint distribution: $P(x_{1:V}|\theta)$
    - Assumption: parameters are known
- Partition variables into:
  - Visible: x_v
  - Hidden: x_h
- Goal: compute unknowns from knowns

$$P(x_h|x_v,\theta) = \frac{P(x_h, x_v|\theta)}{P(x_v|\theta)} = \frac{P(x_h, x_v|\theta)}{\sum_{x'_h} P(x'_h, x_v|\theta)}$$

General Form of Inference

$$P(x_h|x_v,\theta) = \frac{P(x_h, x_v|\theta)}{P(x_v|\theta)} = \frac{P(x_h, x_v|\theta)}{\sum_{x'_h} P(x'_h, x_v|\theta)}$$

- Condition data by clamping visible variables to observed values.
- Normalize by probability of evidence

Nuisance Variables

- Partition hidden variables into:
  - Query Variables: x_q
  - Nuisance variables: x_u

$$P(x_q|x_v,\theta) = \sum_{x_u} P(x_q, x_u|x_v)$$

Inference vs. Learning

- Inference:
  - Compute $P(x_h|x_v,\theta)$
  - Parameters are assumed to be known
- Learning
  - Compute MAP estimate of the parameters

$$\hat{\theta} = \arg\max_{\theta} \sum_{i=1}^{N} \log P(x_{i,v}|\theta) + \log P(\theta)$$

Bayesian Learning

- Parameters are treated as hidden variables
  - **no distinction between inference and learning**
- Main distinction between inference and learning:
  - # hidden variables grows with size of dataset
  - # parameters is fixed

Conditional Independence Properties

- A is independent of B given C

$$X_A \perp_G X_B | X_C$$

- I(G) is the set of all such conditional independence assumptions encoded by G
- G is an I-map for P iff I(G) $\subseteq$ I(P)
  - Where I(P) is the set of all CI statements that hold for P
  - In other words: G doesn't make any assertions that are not true about P

Conditional Independence Properties (cont)

- Note: fully connected graph is an I-map for all distributions
- G is a **minimal I-map** of P if:
  - G is an I-map of P
  - There is no G' $\subseteq$ G which is an I-map of P
- Question:
  - How to determine if $X_A \perp_G X_B | X_C$?
  - Easy for undirected graphs (we'll see later)
  - Kind of complicated for DAGs (Bayesian Nets)

D-separation

- Definitions:
  - An undirected path P is d-separated by a set of nodes E (containing evidence) iff at least one of the following conditions hold:
    - P contains a chain *s -> m -> t* or *s <- m <- t* where *m* is evidence
    - P contains a **fork** *s <- m -> t* where *m* is in the evidence
    - P contains a **v-structure** *s -> m <- t* where *m* is **not** in the evidence, nor any descendent of *m*

D-seperation (cont)

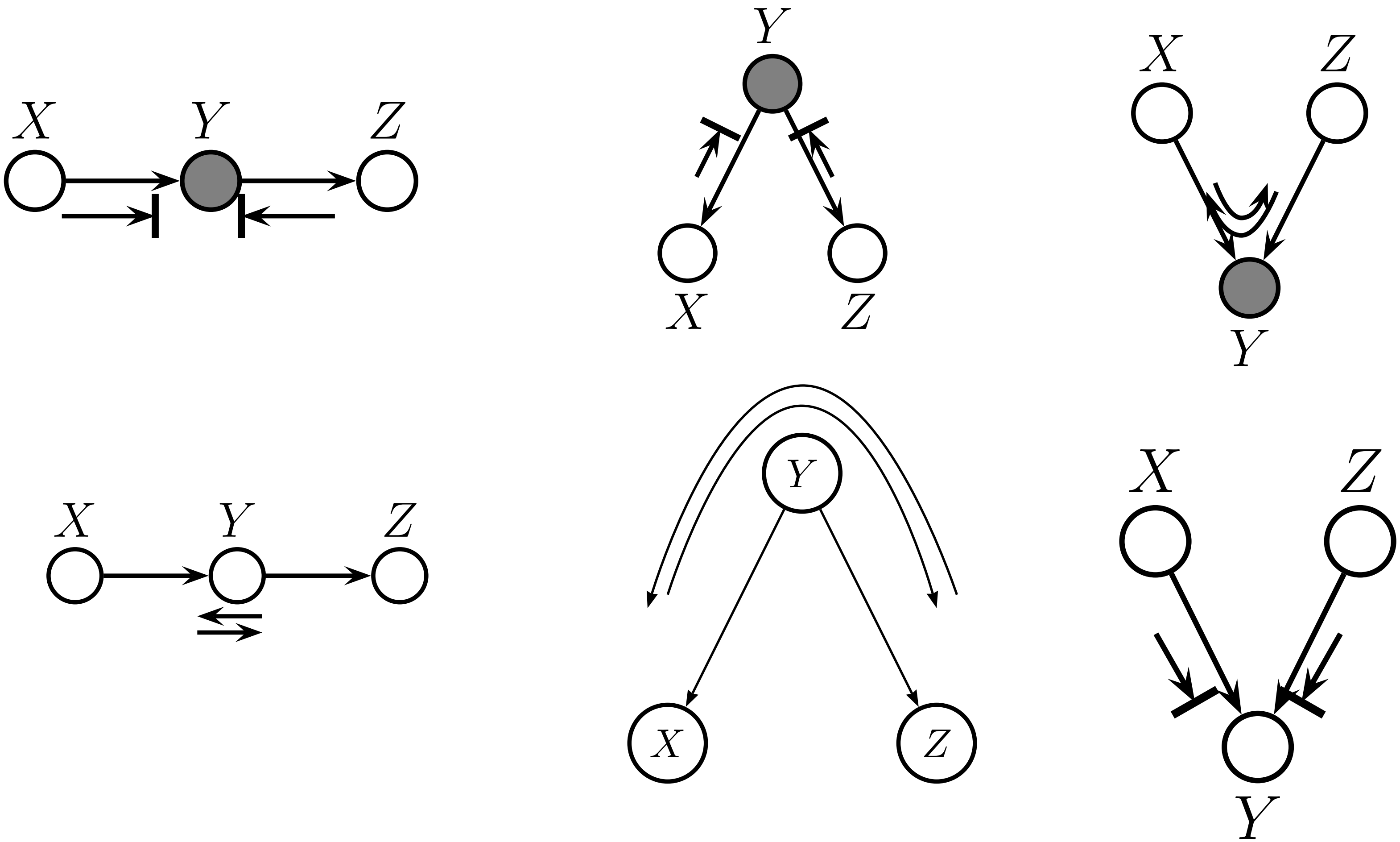
- A set of nodes A is **D-separated** from a set of nodes B, if given a third set of nodes E iff each undirected path from every node in A to every node in B is d-seperated by E
- Finally, define the CI properties of a DAG as follows:

$$X_A \perp_G X_B | X_E \iff \text{A is d-seperated from B given E}$$

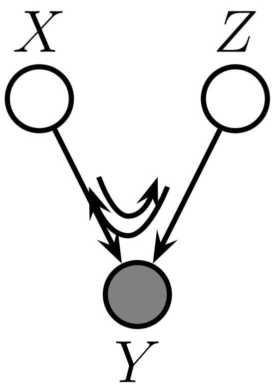
Bayes Ball Algorithm

- Simple way to check if A is d-separated from B given E
  1. Shade in all nodes in E
  2. Place “balls” in each node in A and let them “bounce around” according to some rules
     - Note: balls can travel in either direction
  3. Check if any balls from A reach nodes in B

Bayes Ball Rules



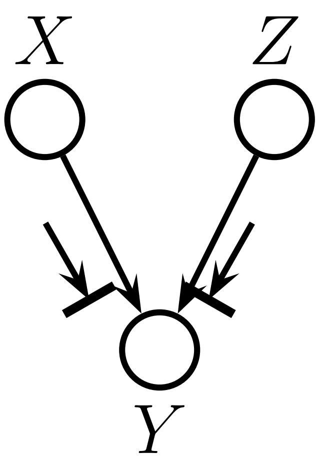
Explaining Away (inter-causal reasoning)



$$P(x, z|y) = \frac{P(x)P(z)P(y|x, z)}{P(y)}$$

$$\implies x \not\perp z|y$$

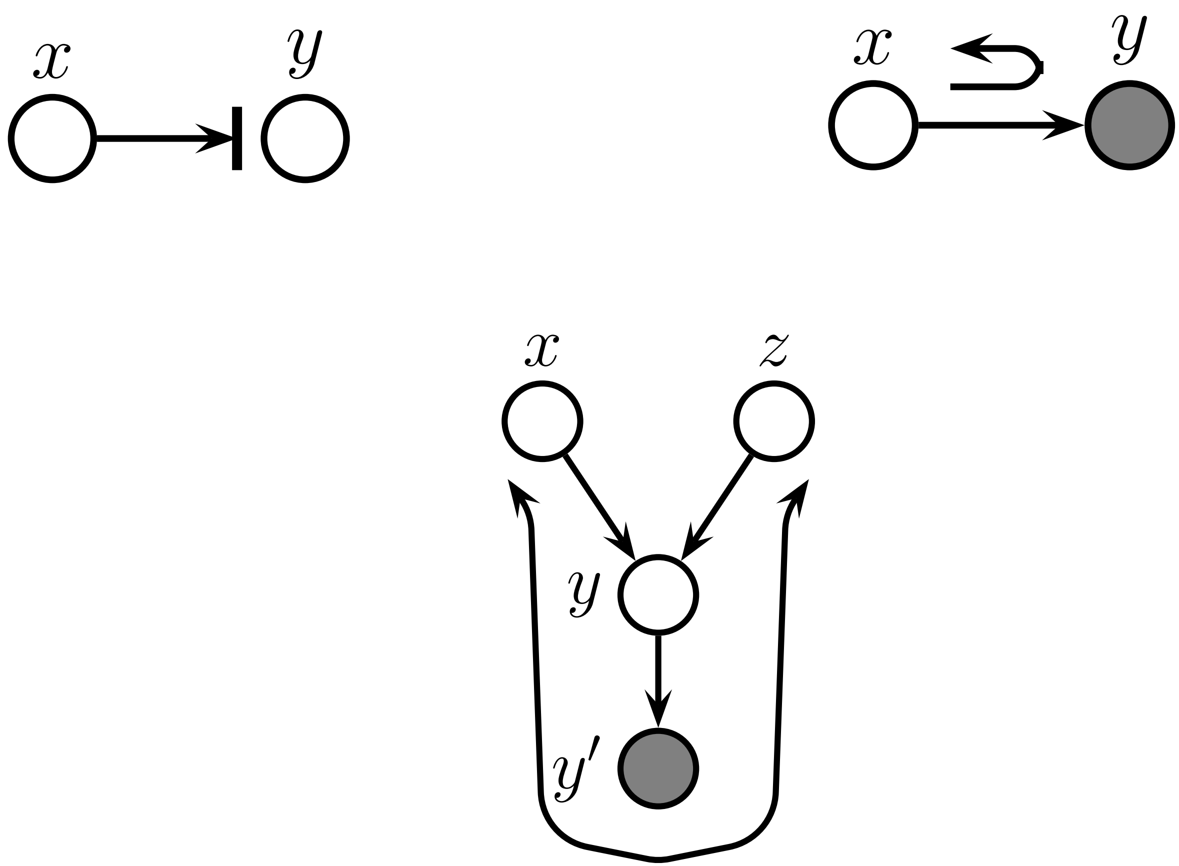
Example: Toss two coins and observe their sum



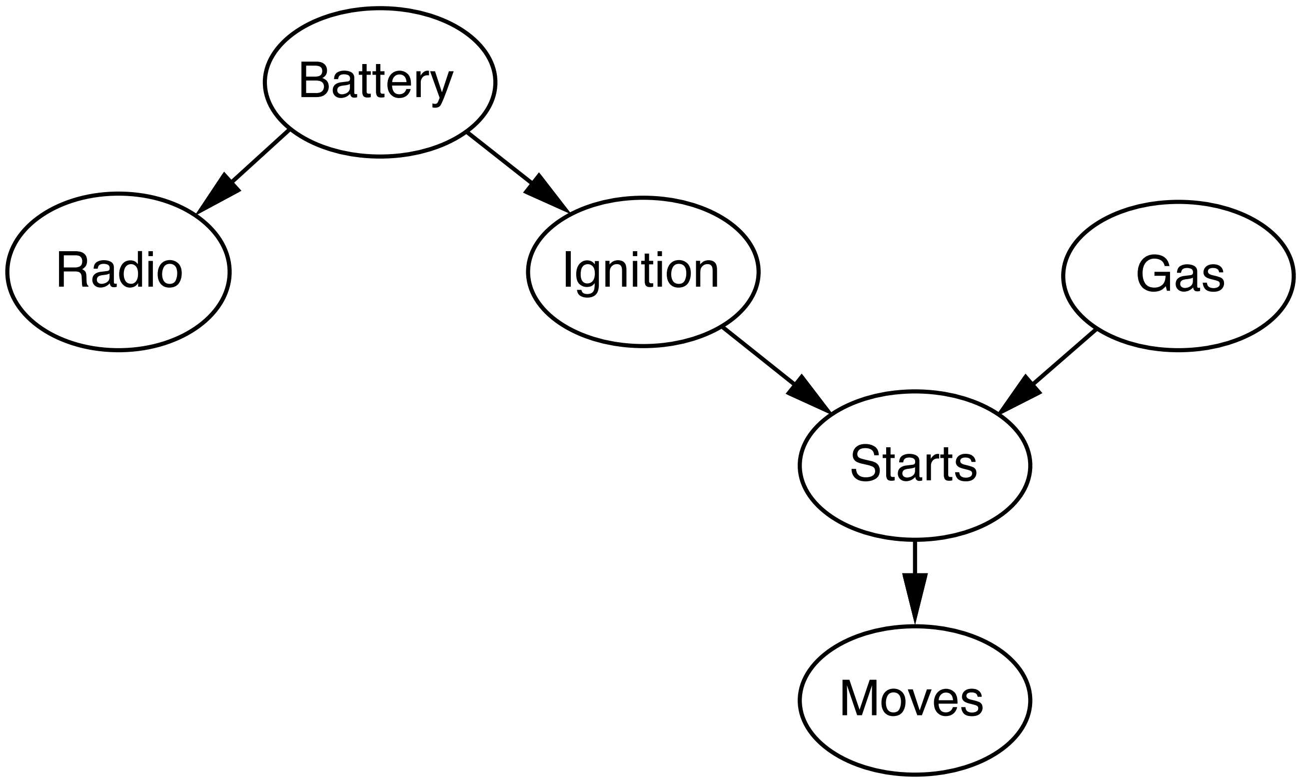
$$P(x, z) = P(x)P(z)$$

$$\implies x \perp z$$

Boundary Conditions



Example



Are Gas and Radio independent? Given Battery? Ignition? Starts? Moves?

Other Independence Properties

1. Ordered Markov Property

$$t \perp pred(t) - pa(t) | pa(t)$$

2. Directed local Markov property

$$t \perp nd(t) - pa(t) | pa(t)$$

3. D separation (we saw this already)

$$X_A \perp_G X_B | X_E \iff \text{A is d-seperated from B given E}$$

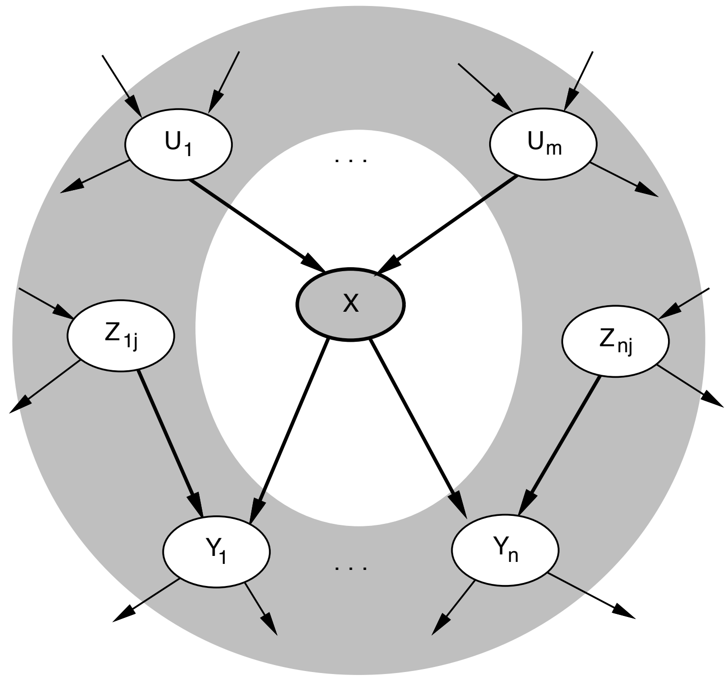
Easy to see:	$3 \implies 2 \implies 1$
Less Obvious:	$1 \implies 2 \implies 3$

Markov Blanket

- Definition:
  - The smallest set of nodes that renders a node t conditionally independent of all the other nodes in the graph.
- Markov blanket in DAG is:
  - Parents
  - Children
  - Co-parents (other nodes that are also parents of the children)

Markov blanket

Each node is conditionally independent of all others given its Markov blanket: parents + children + children's parents



Q: why are the co-parents in the Markov Blanket?

$$P(x_t|\mathbf{x}_{-t}) = \frac{P(x_t, \mathbf{x}_{-t})}{P(\mathbf{x}_{-t})}$$

All terms that do not involve x_t will cancel out between numerator and denominator

$$P(x_t|\mathbf{x}_{-t}) \propto P(x_t|x_{\mathrm{pa}(t)}) \prod_{s \in \mathrm{ch}(t)} p(x_s|\mathbf{x}_{\mathrm{pa}(s)})$$