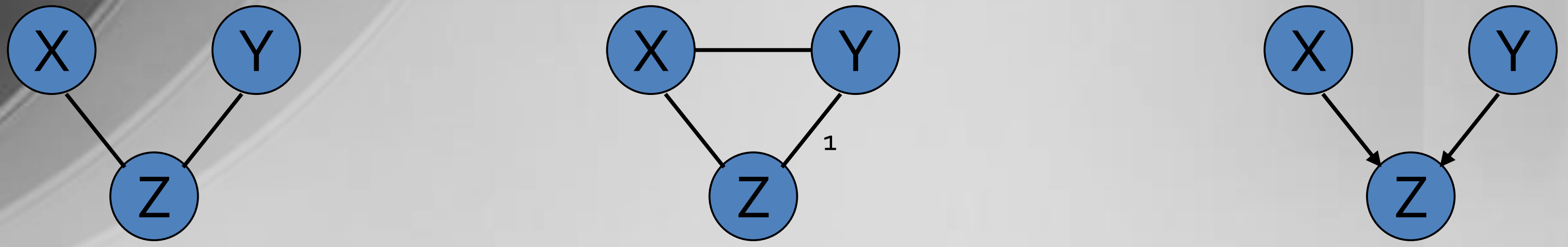
Stochastic Bayesian learning algorithm for graphical models

Many different types of graphs can be used as graphical models

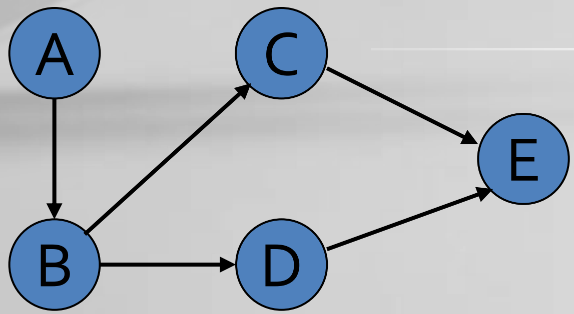
- Undirected
- Labeled
- Directed (Bayesian Networks)



I will today focus on Bayesian Networks

Bayesian Networks

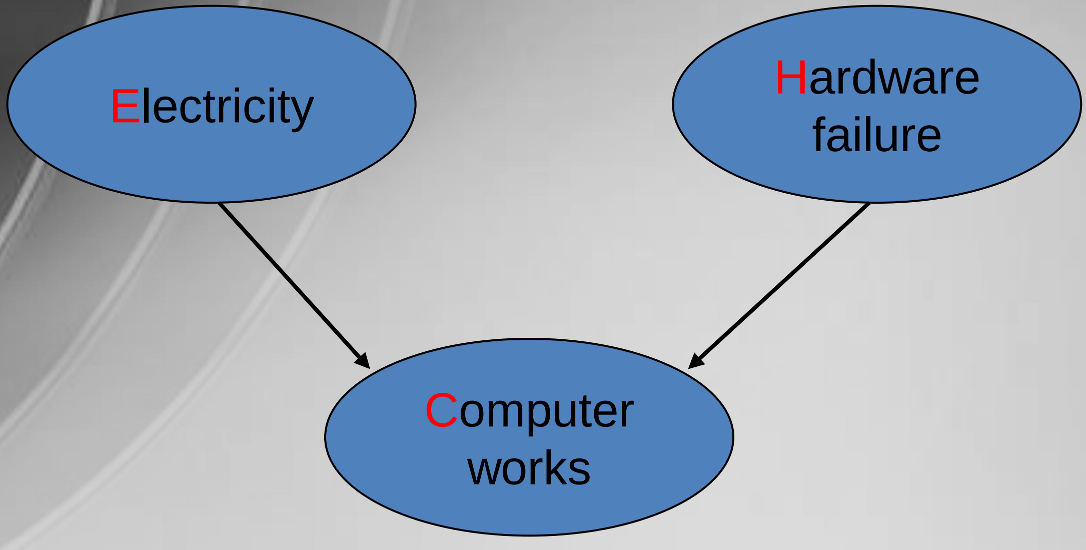
- Represented by Directed Acyclic Graphs (DAG)
- G=(V,E) where V is a set of nodes (or vertices) and E is a set of edges
- The nodes represent Random Variables
- The edges represent dependencies between the variables



V={A, B, C, D, E}

E={(A,B), (B,C), (B,D), (C,E), (D,E)}

Bayesian Network

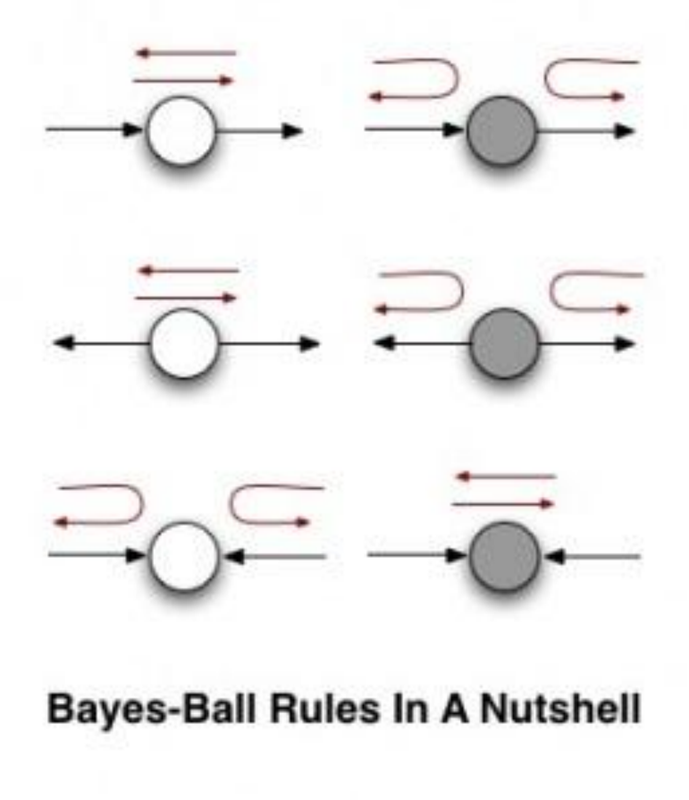


- Three binary variables E, H and C
- 1 represents a positive outcome, 0 a negative outcome
- Given a Bayesian Network you can easily determine which variables are conditionally independent

How conditional independences are determined given a Bayesian Network

- Two variables are independent if they are Dependency-separated (D-separated)
- D-separation can be determined using Bayes-ball

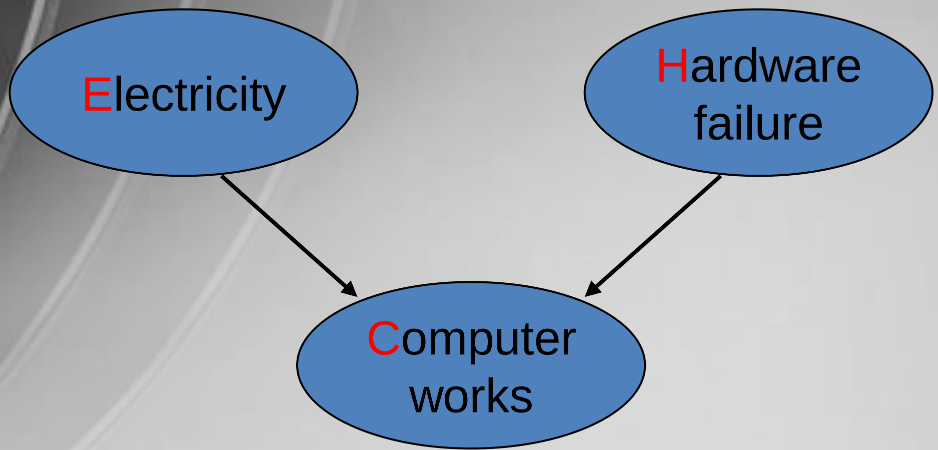
Bayes-ball Rules



- Shows how information is passed between variables
- Observed nodes are shaded
- Two variables X and Y are conditionally independent if you can't “bounce a ball” from X to Y using these rules

Graphic borrowed from http://ergodicity.net/2009/12/08/bays-ball-in-a-nuttshell/

D-separation example



- C is unknown leads to independence between E and H ($E \perp H$)
- If C is known E and H are not independent ($E \not\perp H \mid C$)

Structural learning of Bayesian Networks

- Which DAG is best suited to represent a given data set **X**?
- Would like to find G which optimizes p(G|**X**) over the set of all possible models *S*
- The size of *S* grows exponentially as the number of nodes increase
- For undirected graphs |*S*| = $2^{\binom{k}{2}}$,where k is the number of nodes in the considered graphs
- Exhaustive searches are impossible even for a relatively small number of nodes

Structural learning of Bayesian Networks

- We have to design a method for a selective search
- We use a Markov chain Monte Carlo (MCMC) style approach
- The idea is to create a Markov chain that traverses through different DAGs eventually finding the optimal DAG

MCMC approach to structural learning

- State space of the Markov chain is *S*
- There are methods available to calculate p(**X**|G)
- Define a "proposal mechanism" which proposes a new state G' given the current state G_t with probability $q(G', G_t)$
- Set $G_{t+1} = G'$ with probability

$$\min\left(1, \frac{p(G')\,p(X \mid G')\,q(G_t, G')}{p(G_t)\,p(X \mid G_t)\,q(G', G_t)}\right) \qquad *$$

Else $G_{t+1} = G_t$

MCMC approach to structural learning

- Where p(G) is the so called prior probability of G
- * is called the acceptance probability
- It can be shown that the stationary distribution of the created Markov chain equals the distribution p(G|**X**), for every G in *S*

MCMC approach to structural learning

- To apply this method we need to be able to analytically calculate the probabilities q(G’, Gt)
- This adds heavy restrictions to the way the proposal mechanism is constructed
- If we remove the factor

$$\frac{q(G_t, G')}{q(G', G_t)}$$

from the acceptance probability, the proposal mechanism can be constructed much more freely as we no longer need to be able to calculate q(G’, Gt)

MCMC approach to structural learning

- The stationary distribution of the Markov chain does not equal p(G|**X**)
- But

$$\arg\max_{G \in S_T} p(X \mid G) = \arg\max_{G \in S} p(G \mid X)$$

when $T \to \infty$ and p(G) = $\frac{1}{|S|}$

Here S_T is defined as the subset of S that the Markov Chain has discovered by time T

Future research

- Clustering of data using MCMC methods, within each cluster specify a Bayesian Network
- Labeled Graphical Models (LGM)
  - New type of graphical model
  - Subclass of context specific independence models (CSI)
  - Try to find relation between LGMs and Bayesian Networks