Structure Learning

Dr. Xiaowei Huang

https://cgi.csc.liv.ac.uk/~xiaowei/

Up to now,

- Overview of Machine Learning
- Traditional Machine Learning Algorithms
- Deep learning
- Probabilistic Graphical Models
    - Introduction
    - I-Map, Perfect Map
    - Reasoning Patterns
    - D-Seperation

Topics

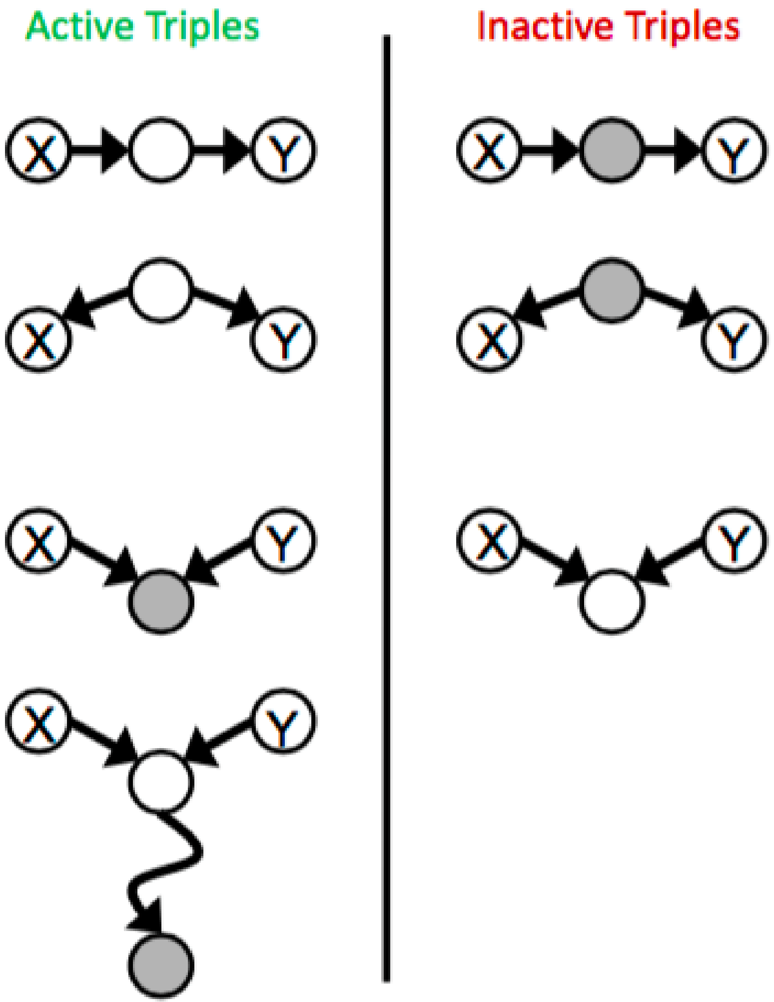
- Example of D-separation

- Why do we need structure learning?
- Goal of structure learning?
- Caution in establishing a connection between two variables?
- Overview of methods

Example of D-separation

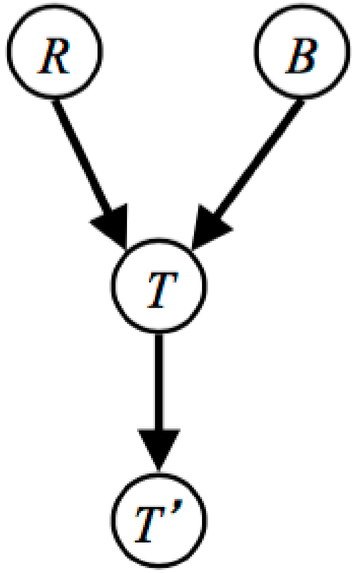
Active / Inactive Paths

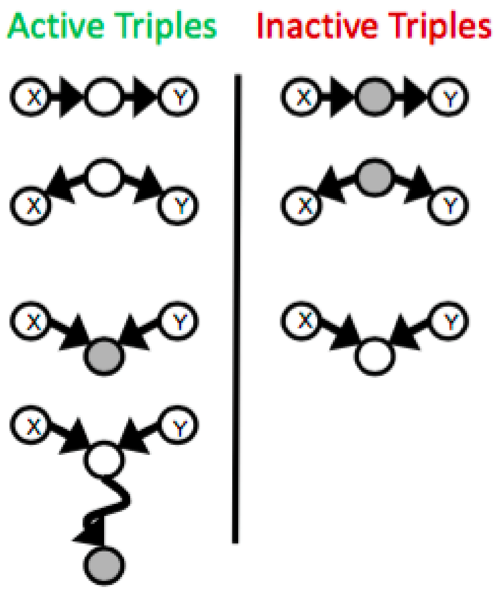
- Question: Are X and Y conditionally independent given evidence variables {Z}?
  - Yes,if X and Y “d-separated” by Z
  - Consider all (undirected) paths from X to Y
  - If no path is active -> independence!
- A ***path*** is active if every triple in path is active:
  - Causal chain A -> B -> C where B is unobserved (either direction)
  - Commoncause A <- B -> C where B is unobserved
  - Common effect (aka v-structure)
    A -> B <- C where B *or one of its descendants* is observed
- All it takes to block a path is a ***single*** inactive segment
  - (But ***all*** paths must be inactive)



Example

$R \perp\!\!\!\perp B$ ***Yes, Independent!***

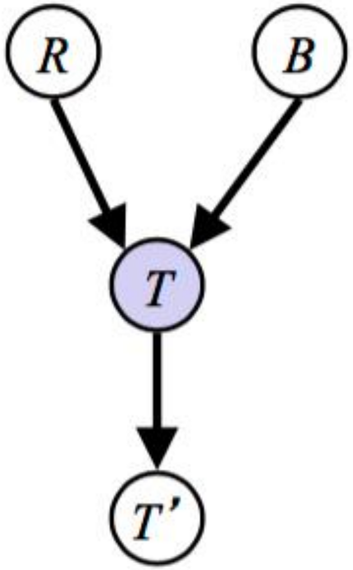


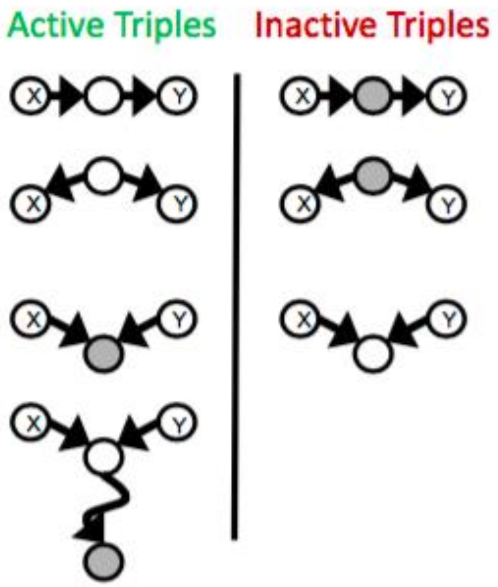


Example

$R \perp\!\!\!\perp B$ *Yes, Independent!*

$R \perp\!\!\!\perp B | T$ *No*



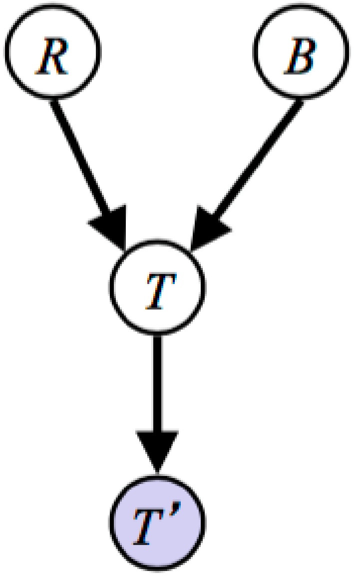


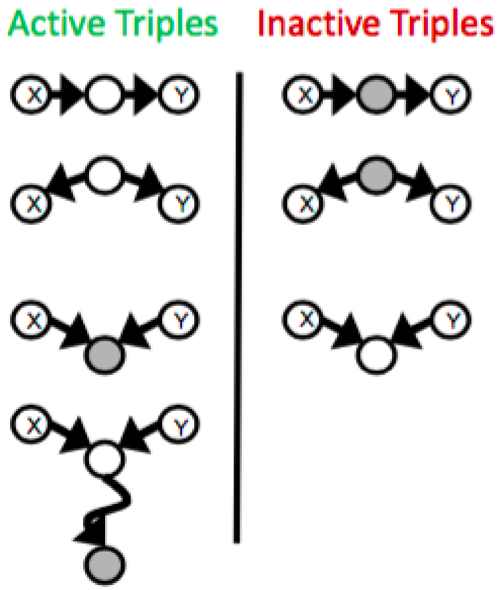
Example

$R \perp\!\!\!\perp B$ *Yes, Independent!*

$R \perp\!\!\!\perp B | T$ *No*

$R \perp\!\!\!\perp B | T'$ *No*

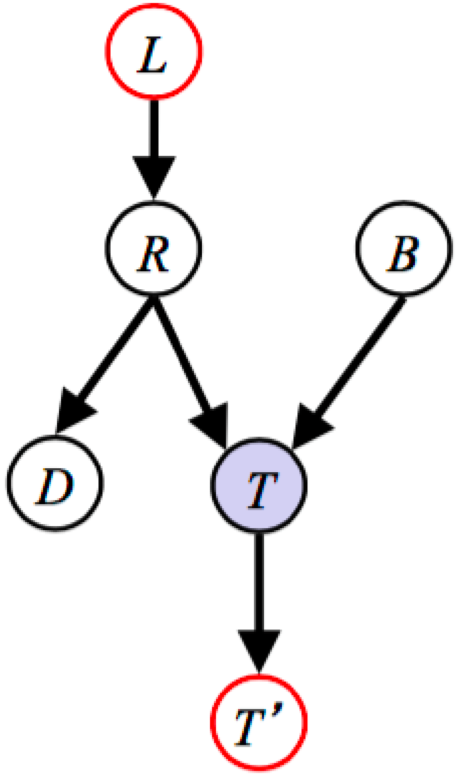


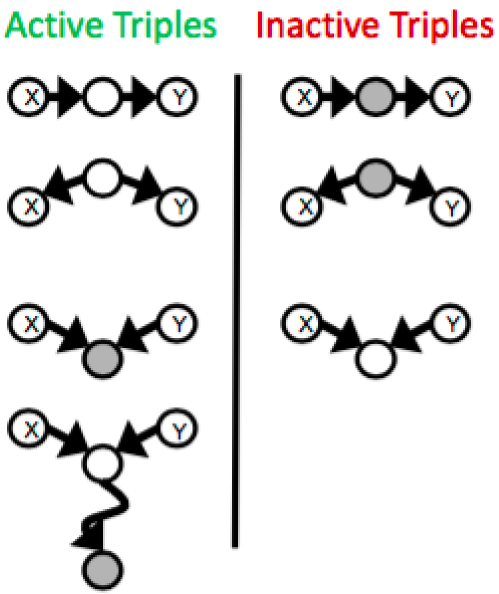


Example

$L \perp\!\!\!\perp T' | T$

Yes, Independent

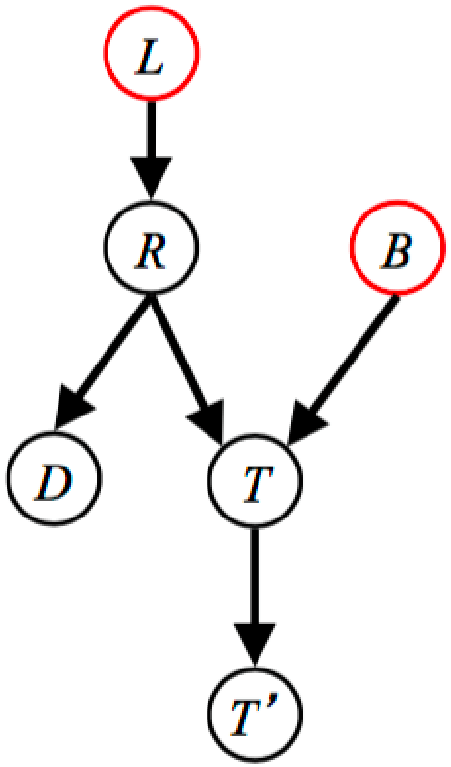


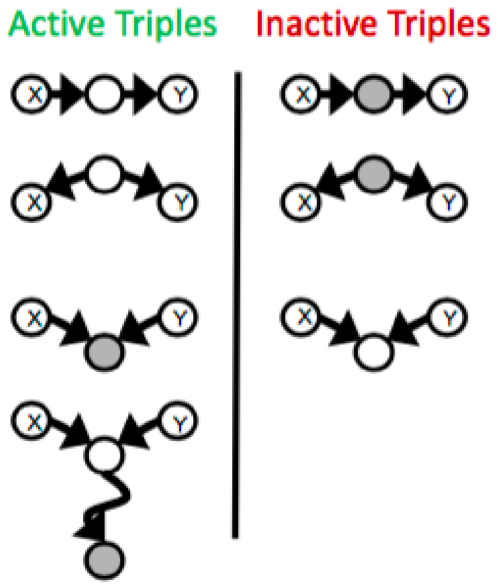


Example

$L \perp\!\!\!\perp T' | T$ *Yes, Independent*

$L \perp\!\!\!\perp B$ *Yes, Independent*



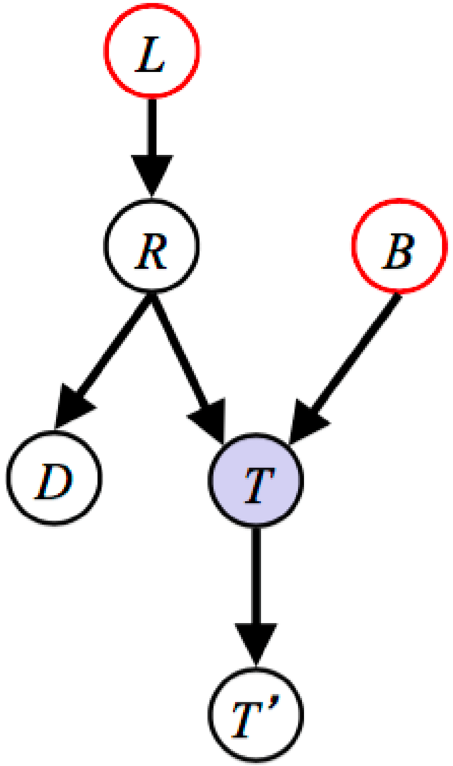


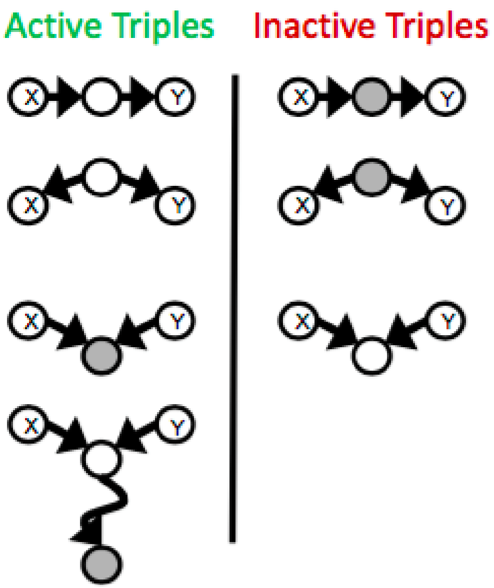
Example

$L \perp\!\!\!\perp T' | T$ *Yes, Independent*

$L \perp\!\!\!\perp B$ *Yes, Independent*

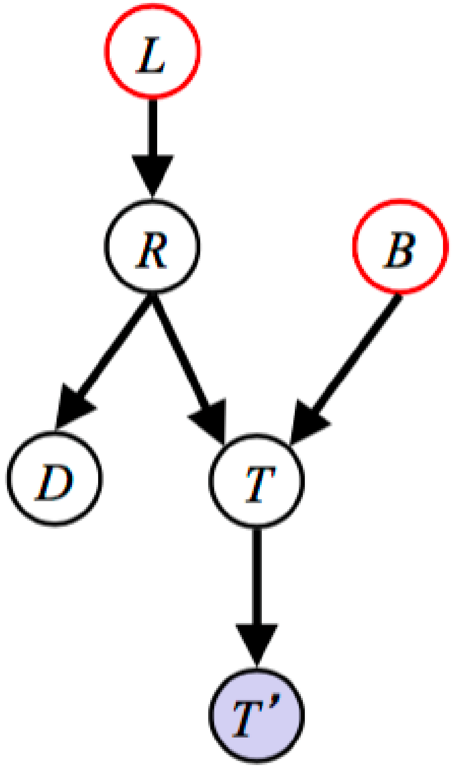
$L \perp\!\!\!\perp B | T$ *No*

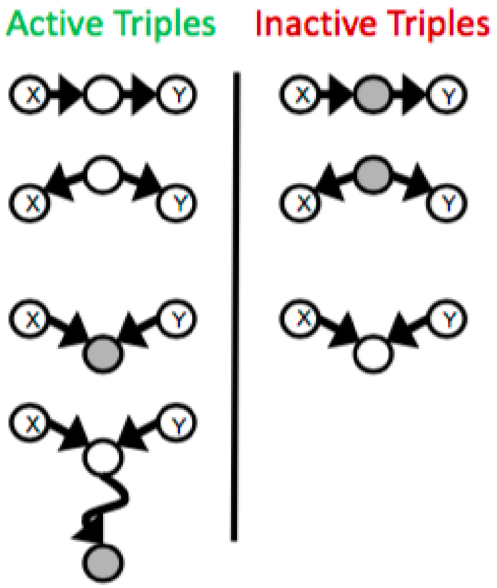




Example

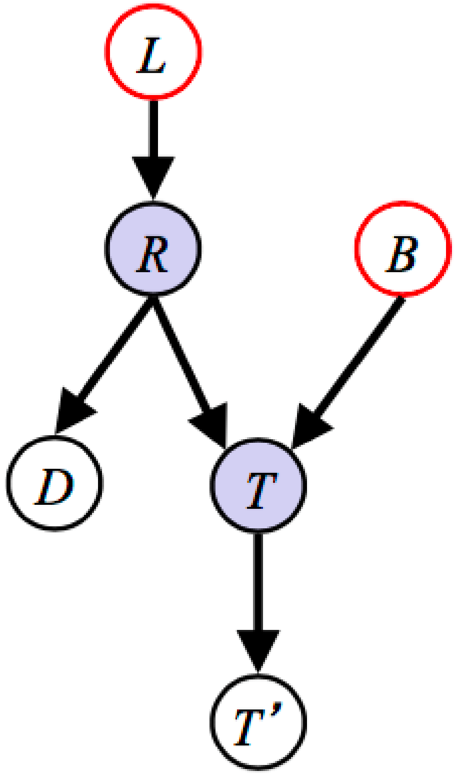
$L \perp\!\!\!\perp T' \mid T$	*Yes, Independent*
$L \perp\!\!\!\perp B$	*Yes, Independent*
$L \perp\!\!\!\perp B \mid T$	*No*
$L \perp\!\!\!\perp B \mid T'$	*No*

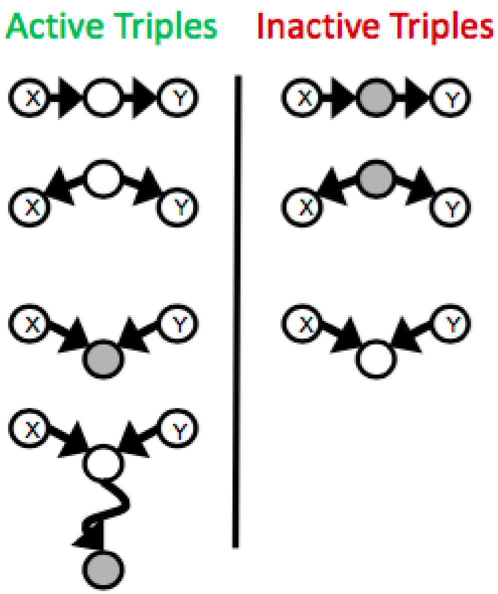




Example

$L \perp\!\!\!\perp T' \mid T$	*Yes, Independent*
$L \perp\!\!\!\perp B$	*Yes, Independent*
$L \perp\!\!\!\perp B \mid T$	*No*
$L \perp\!\!\!\perp B \mid T'$	*No*
$L \perp\!\!\!\perp B \mid T, R$	*Yes, Independent*





Example

$L \perp\!\!\!\perp T' | T$ *Yes, Independent*

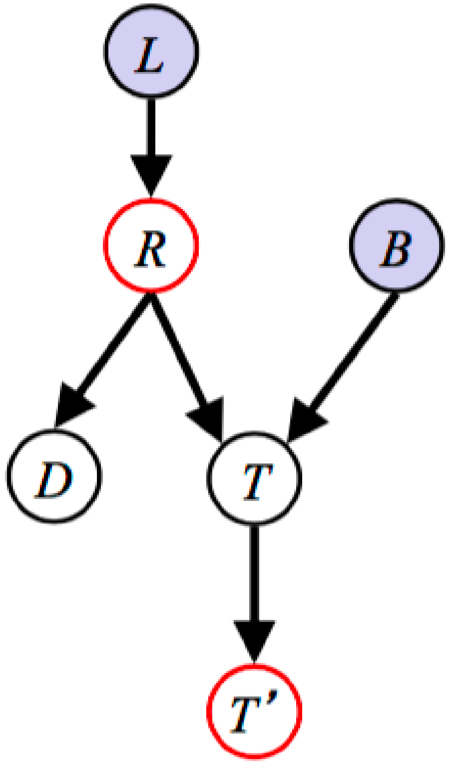
$L \perp\!\!\!\perp B$ *Yes, Independent*

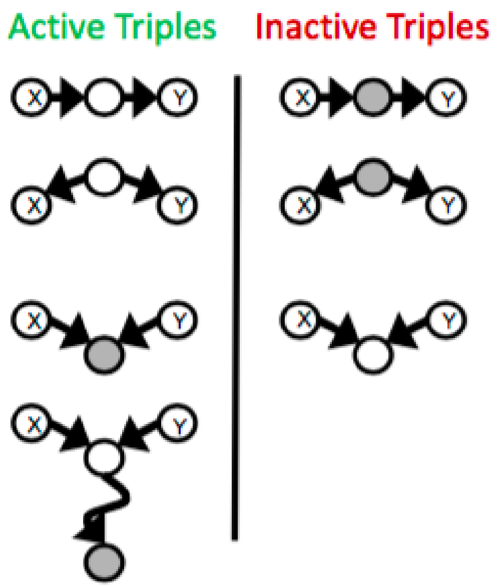
$L \perp\!\!\!\perp B | T$ *No*

$L \perp\!\!\!\perp B | T'$ *No*

$L \perp\!\!\!\perp B | T, R$ *Yes, Independent*

$R \perp\!\!\!\perp T' | L, B$ *No*

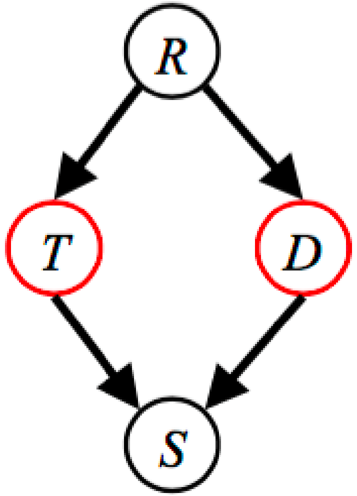


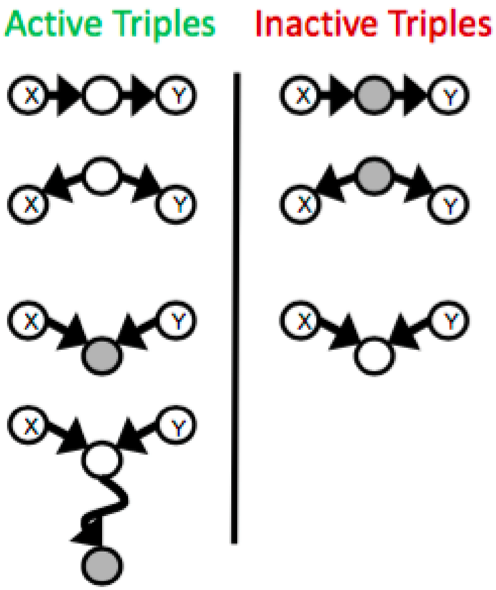


Example

- Variables:
  - R: Raining
  - T: Traffic
  - D: Roof drips
  - S: I'm sad
- Questions:

$T \perp\!\!\!\perp D$ *No*



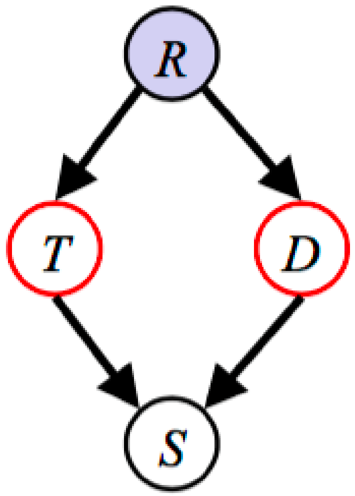


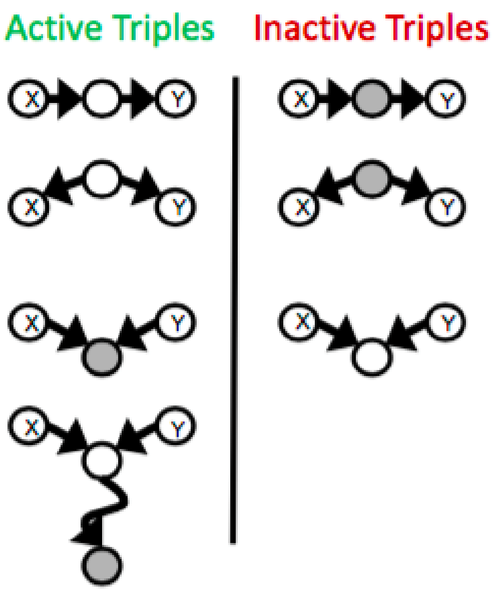
Example

- Variables:
  - R: Raining
  - T: Traffic
  - D: Roof drips
  - S: I'm sad
- Questions:

$T \perp\!\!\!\perp D$ *No*

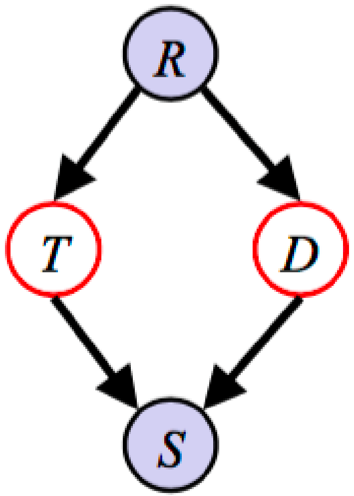
$T \perp\!\!\!\perp D | R$ *Yes, Independent*

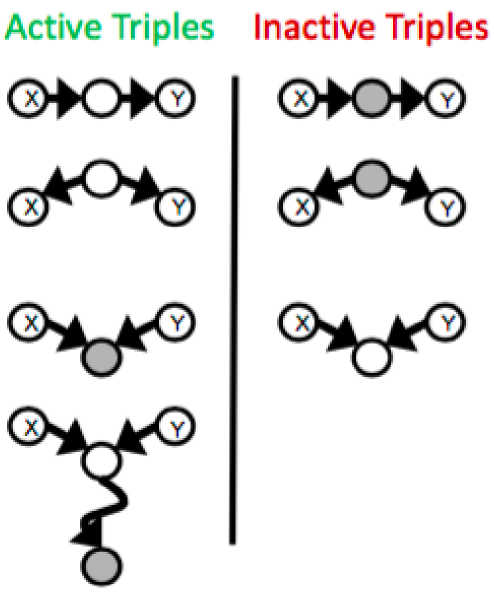




Example

- Variables:
  - R: Raining
  - T: Traffic
  - D: Roof drips
  - S: I'm sad
- Questions:





$T \perp\!\!\!\perp D$ *No*

$T \perp\!\!\!\perp D | R$ *Yes, Independent*

$T \perp\!\!\!\perp D | R, S$ *No*

Why do we need structure learning?

Two approaches to task of acquiring a model

- 1. Knowledge Engineering
  - Construct a network by hand with expert's help
- 2. Machine Learning
  - Learn model from a set of instances

Knowledge Engineering vs ML

- Knowledge Engineering Approach
  - Pick variables, pick structure, pick probabilities
  - Too much Effort
    - Simple ones require hours of effort, complex one: months
  - Significant testing of model by evaluating results of typical queries yield plausible answers
- Machine Learning Approach
  - Instances available from distribution we wish to model
  - Easier to get large data sets rather than human expertise

Difficulties with Manual Construction

- In some domains:
  - Amount of knowledge required too large
  - No experts who have sufficient understanding
  - Cost: expert time is valuable
- Properties of distribution change from one site to another
- Change over time
  - Expert cannot redesign every few weeks
- Modeling mistakes have serious impact on quality of answers

Advantage of ML approach

- We are in the Information Age
  - Easier to obtain even large amounts of data in electronic form than to obtain human expertise
- Example Data
  - Medical Diagnosis
    - Patient records
  - Pedigree Analysis (Genetic Inheritance)
    - Family trees for disease transmission
  - Image Segmentation
    - Set of images segmented by a person

Example: Medical Diagnosis Task

- Collection of patient records
  - History:
    - Age, sex, history, medical complications
  - Symptoms
  - Results of tests
  - Diagnosis
  - Treatment
  - Outcome
- Task: Use data to model distribution of patients
    - Pathologist diagnoses disease of lymph nodes (Pathfinder 1992)

Goal of Structure Learning

Goal of Structure Learning: Knowledge Discovery

- A tool for discovering knowledge about P^*
  - What are the direct/indirect independencies?
  - Nature of dependencies
    - E.g., positive or negative correlation
  - Example: in medical domain, which factors lead to a disease
- Bayesian network reveals much finer structure
  - Distinguish between direct and indirect independencies, both of which lead to correlations

Problem Assumptions

- We do not know the structure
- Dataset is fully observed
  - A strong assumption
- Assume data D is generated i.i.d. from distribution $P^*(X)$
- Assume that P^* is induced by BN G^*

Caution in establishing a connection between two variables?

Knowledge Discovery Goal

- Goal: recover $G*$
- Since there are many I-maps for P* we cannot distinguish them from D
- Thus $G*$ is not *identifiable*
- Best we can do is recover $G*$s equivalence class

Too few or too many edges in $G*$

- Even learning equivalence class of networks is hard
- Data sampled is noisy
- Need to make decisions about including edges we are less sure about
  - Too few edges means missing out on dependencies
  - Too many edges means spurious dependencies

To what extent do independencies in $G*$ manifest in D?

- Two coins X and Y tossed independently
- We are given data set of 100 instances
- Learn a model for this scenario
- Typical data set:
  - 27 head/head
  - 22 head/tail
  - 25 tail/head
  - 26 tail/tail
- Are the coins independent?

Coin Tossing Probabilities

- Marginal Probabilities
  - *P(X=head)=.49, P(X=tail)=0.51, P(Y=head)=.52, P(Y=tail)=.48*
- Products of marginals*:*
  - *P(X=head)* x *P(Y=head) =.49 x .52 =.25*
  - *P(X=head)* x *P(Y=tail) =.49 x .48 =.24*
  - *P(X=tail)* x *P(Y=head) =.51 x .52 =.27*
  - *P(X=tail)* x *P(Y=tail) =.51 x .48 =.24*
- Joint Probabilities
  - *P(XY=head-head)=.27*
  - *P(XY=head-tail)=22*
  - *P(XY=tail-head)=.25*
  - *P(XY=tail-tail)=.26*
- But we suspect independence
  - since probability of getting exactly 25 in each category is small (approx. 1 in 1,000)

27 head/head

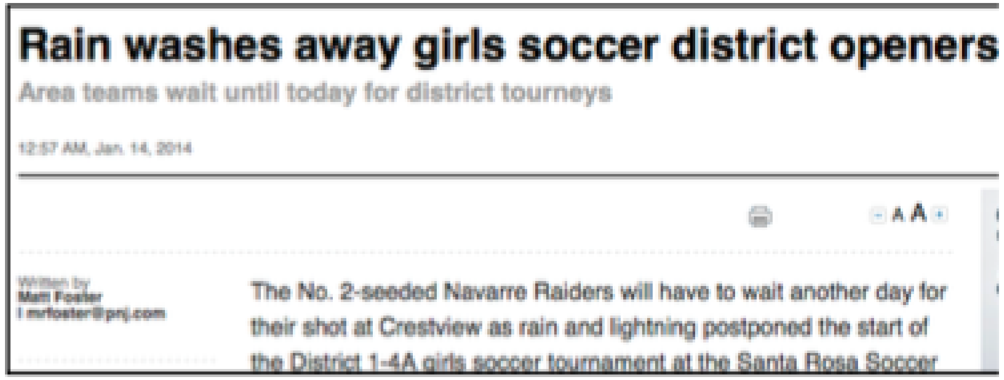
22 head/tail

25 tail/head

26 tail/tail

According to empirical distribution: not independent

Rain-Soccer Probabilities


Rain washes away girls soccer district openers

Area teams wait until today for district tourneys

12:57 AM, Jan. 14, 2014

The No. 2-seeded Navarre Raiders will have to wait another day for their shot at Crestview as rain and lightning postponed the start of the District 1-4A girls soccer tournament at the Santa Rosa Soccer

- Scan sports pages for 100 days
- Select an article at random to see
  - If there is mention of rain and soccer
- Marginal Probabilities
  - *P(X=rain)=.49, P(X=no rain)=.51, P(Y=soccer)=.48, P(Y=no soccer)=. 52*
- Joint Probabilities
  - *P(XY=rain-soccer)=.27*
  - *P(XY=rain-no soccer)=.22*
  - *P(XY=no rain-soccer)=.25*
  - *P(XY=no rain-no soccer)=.26*

According to empirical distribution: not independent → We suspect there is a weak connection (not independent)

It is hard to be sure whether the true underlying model has an edge between X and Y

Data Fragmentation with spurious edges

- In a table CPD no of bins grows exponentially with no of parents
- Cost of adding a parent can be very large
- Cost of adding a parent grows with no of parents already there
- It is better to obtain a sparser structure
- We can sometimes learn a better model by learning a model with fewer edges even if it does not represent the true distribution.

Overview of methods

Structure Learning Algorithms

- Constraint-based
  - Find structure that best explains determined dependencies
  - Sensitive to errors in testing individual dependencies
- Score-based
  - Search the space of networks to find high-scoring structure
  - Since space is super-exponential, need heuristics

Finds a Bayesian network structure whose implied independence constraints "match" those found in the data.

Find the Bayesian network structure that can represent distributions that "match" the data (i.e. could have generated the data).

Elements of BN Structure Learning

- Local: Independence Tests
  - Measures of *Deviance*-from-independence between variables
  - Rule for accepting/rejecting hypothesis of independence
- Global: Structure Scoring
  - Goodness of Network

Independence Tests

Idea: measure the independence between two variables with a value, and then determine whether to establish the connection based on the score.

- For variables x_i, x_j in data set D of M samples
  - Pearson's Chi-squared (χ^2) statistic

$$d_{\chi^2}(\mathcal{D}) = \sum_{x_i, x_j} \frac{\left(M[x_i, x_j] - M \cdot \hat{P}(x_i) \cdot \hat{P}(x_j)\right)^2}{M \cdot \hat{P}(x_i) \cdot \hat{P}(x_j)}$$

    - Independence $d_\chi(D)=0$, larger value when Joint $M[x,y]$ and expected counts (under independence assumption) differ
  - Mutual Information (K-L divergence) between joint and product of marginals

$$d_I(\mathcal{D}) = \frac{1}{M} \sum_{x_i, x_j} M[x_i, x_j] \log \frac{M[x_i, x_j]}{M[x_i] M[x_j]}$$

    - Independence $d_I(D)=0$, otherwise a positive value
- Decision rule
  - False rejection probability due to choice of t is its p-value

$$R_{d,t}(\mathcal{D}) = \begin{cases} \text{Accept} & d(\mathcal{D}) \le t \\ \text{Reject} & d(\mathcal{D}) > t \end{cases}$$

Structure Scoring

Idea: measure each graph with a score, and select the graph with best score as G

- Log-likelihood Score for *G* with *n* variables

$$score_L(\mathcal{G} : \mathcal{D}) = \sum_{\mathcal{D}} \sum_{i=1}^{n} \log \hat{P}(x_i \mid pax_i)$$ Sum over all data and variables x_i

- 2. Bayesian Score

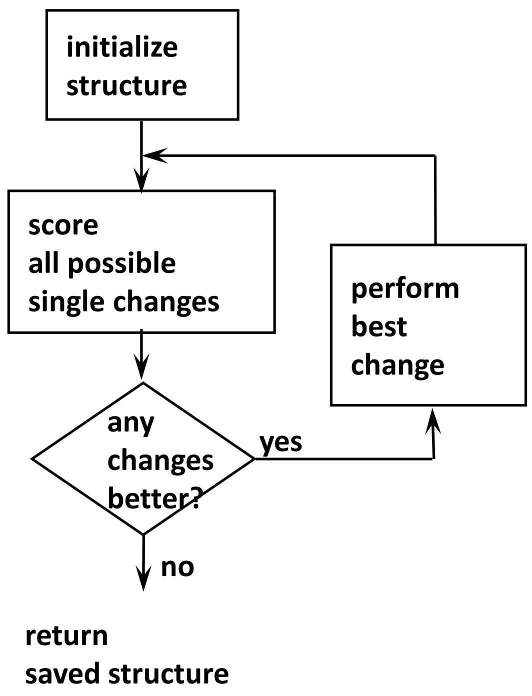
$$score_B(\mathcal{G} : \mathcal{D}) = \log p(\mathcal{D} \mid \mathcal{G}) + \log p(\mathcal{G})$$

- 3. Bayes Information Criterion
  - With Dirichlet prior over graphs

$$score_{BIC}(\mathcal{G} : D) = l(\hat{\theta}_G : D) - \frac{\log M}{2} Dim(\mathcal{G})$$

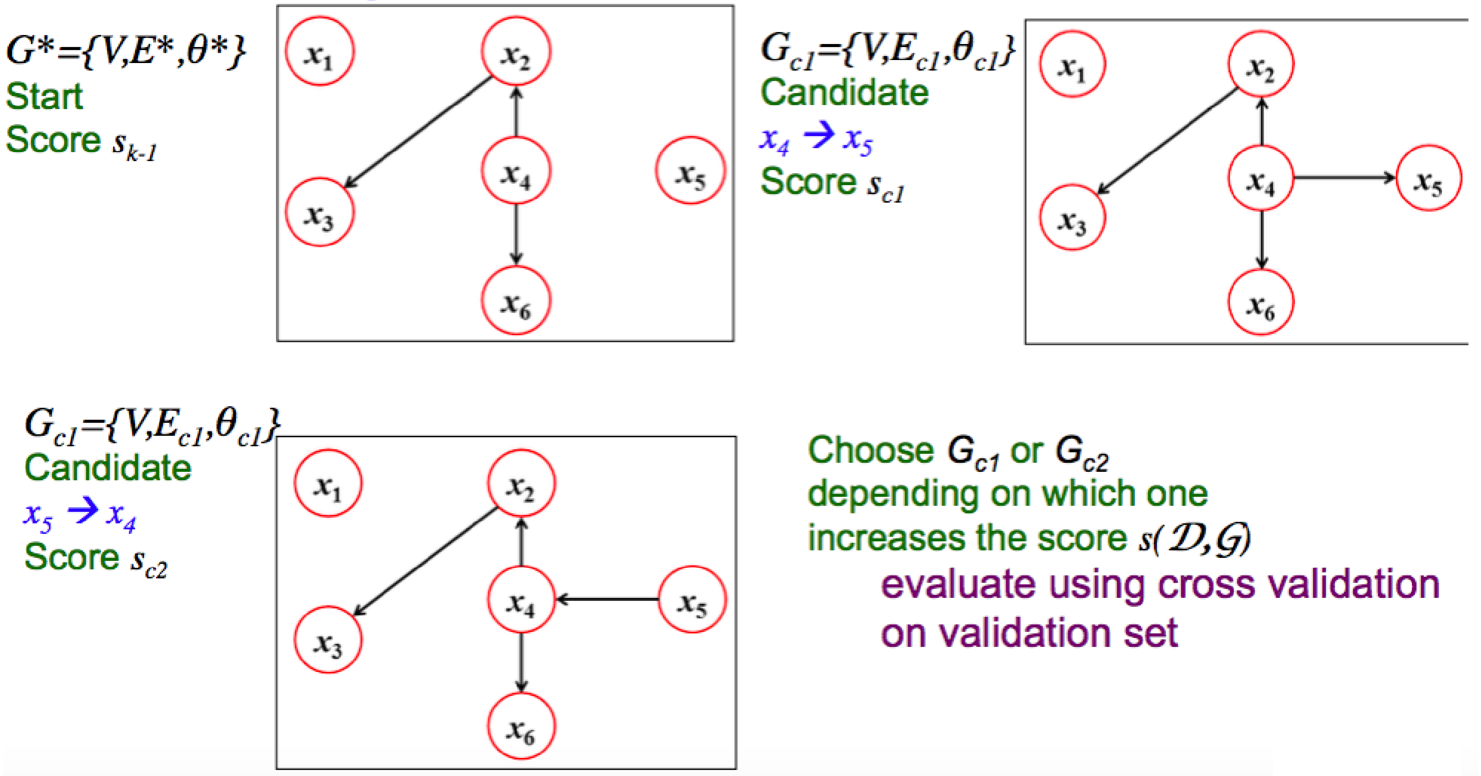
Model search

- Finding the BN structure with the highest score among those structures with at most *k* parents is NP hard for *k*>1 (Chickering, 1995)
- Heuristic methods
  - Greedy
  - Greedy with restarts
  - MCMC methods



Heuristic for BN Structure Learning

- Consider pairs of variables ordered by χ^2 value
- Add next edge if score is increased



Summary of Bayesian Networks

Summary of Bayesian Networks

- Bayesian network specifies a set of independencies
- Distributions have multiple minimal I-maps
  - Minimal I-map does not capture all independence properties of *P*
- P-map: not every distributions has a P-map
  - This motivates the use of Markov networks
- I-equivalence is when graphs capture same independences