Structure Learning of Probabilistic Graphical Models

Causal Reasoning

Causality in Bayesian Networks

- *"... interpretation of DAGs as carriers of independence assumptions does not necessarily imply causation ...",*
- *"... some of these patterns can be given meaningful interpretation only in terms of causation."*

-- Judea Pearl, Graphical models for probabilistic and causal reasoning

?!

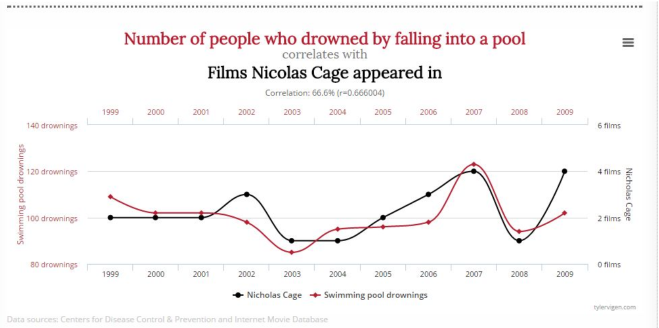
Judea Pearl

Causal relationships

- Find the right Markov blanket
- Correlation is a symmetric relation; causality isn’t
- Correlation is a precondition for causation
- Causality over time -> Cause happens before effect
- Causation often needs experimental data and profits from counter examples
- There’s usually no 100% prof for causality in real world examples

Typical wrong

- Example of random correlations
  - Drowning of people and the number of films Nicolas Cage appeared in



Correlation in Nicolas' Cage appearance
https://www.networkworld.com/article/3173856/analytics/did-nicolas-cage-cause-swimming-pool-drownings.html

- Example of correlation but wrong conclusions
  - The more firemen are sent the more damage was done
  - Scholars who are getting tutored are getting worse grades than average

LUCAS - Dataset

- LUCAS (LUng CAncer Simple set)
- Artificial dataset sampled from predetermined structure and conditional probabilities

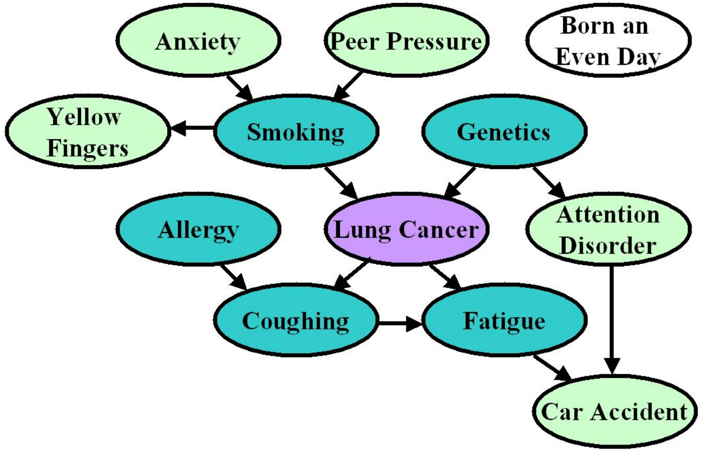
  P(Anxiety=T)=0.64277

  P(Peer Pressure=T)=0.32997

  P(Smoking=T|Peer Pressure=F, Anxiety=F)=0.43118

  P(Smoking=T|Peer Pressure=T, Anxiety=F)=0.74591

  …
- 2000 samples with binary feature values



Dataset - Overview
http://www.causality.inf.ethz.ch/data/LUCAS.html

Evaluated Frameworks / Tools

TECHNISCHE UNIVERSITÄT DARMSTADT

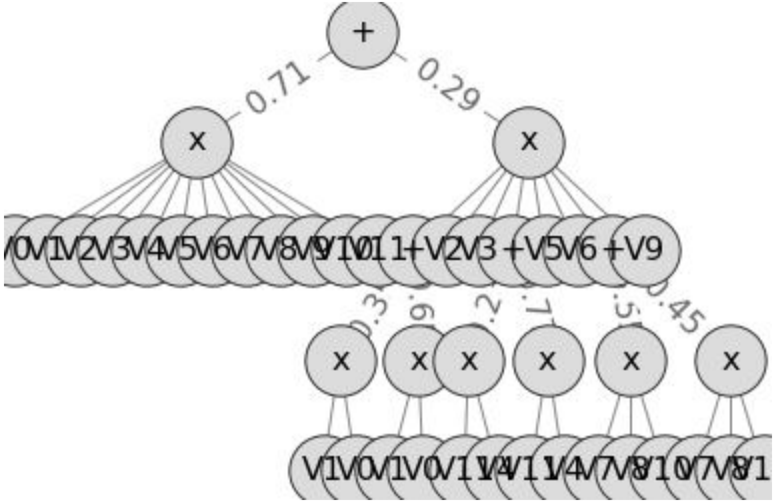
- SPFlow - https://github.com/SPFlow/SPFlow
  - An Easy and Extensible Library for Sum-Product Networks
- BNFinder - https://github.com/sysbio-vo/bnfinder
  - Tool for learning bayesian networks
- bnlearn - http://www.bnlearn.com/
  - An R package for Bayesian network learning and inference
- BayesSpy – https://github.com/bayespy/bayespy
  - Bayesian Python
- OpenGM - http://hciweb2.iwr.uni-heidelberg.de/opengm/
  - A C++ template library for discrete factor graph models and distributive operations

don't support BN structure learning

SPFlow - Results

- hardly interpretable graphs

```
learn_mspn(train_data, ds_context,
min_instances_slice=200, threshold=0.4)
```

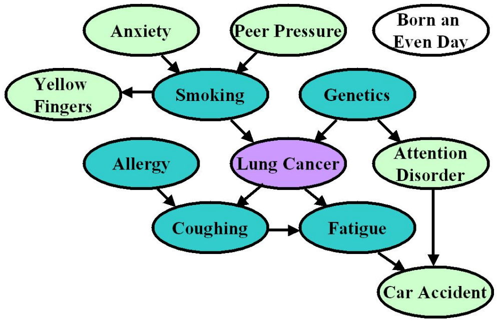


result of generated sum product network graph

BNFinder - Results

- used primarily in medical domain / bioinformatics
  - e.g. finding dependencies in genetic or protein data
- relies on given set of prior knowledge
  - `#regulators:`
    - `"Anxiety", "Peer_Pressure", "Genetics", "Born_an_Even_Day", "Car_Accident"`
  - `#parents: …`
- pos. / neg. correlations:

```
Anxiety      +    Smoking
Anxiety      +    Yellow_Fingers
Peer_Pressure     +    Smoking
Genetics     +    Attention_Disorder
Genetics     +    Lung_cancer
Car_Accident     +    Attention_Disorder
Car_Accident     +    Fatigue
Car_Accident     +    Lung_cancer
```



Dataset - Overview
http://www.causality.inf.ethz.ch/data/LUCAS.html

bnlearn - Structure learning algorithms

- **Grow-Shrink**
  - Margaritis 2003
- **Incremental Association**
  - Tsamardinoset al. 2003
- **Fast Incremental Association**
  - Yaramakala and Margaritis 2005
- **Interleaved Incremental Association**
  - Tsamardinoset al. 2003
- **Max-Min Parents and Children**
  - Tsamardinoset al. 2006

Grow-Shrink - Pseudo Code

1. [Compute Markov Blankets] (same as plain GS)

For all $X \in \mathcal{U}$, compute the Markov blanket $\mathbf{B}(X)$.

2. [Compute Graph Structure]

For each $X \in \mathcal{U}$ and $Y \in \mathbf{B}(X)$ do:

- Set $p \leftarrow \frac{1}{2}$.
- Set $\mathbf{T}$ to be the smaller of $\mathbf{B}(X) - \{Y\}$ and $\mathbf{B}(Y) - \{X\}$.
- Let $G \leftarrow G(X,Y) = 1 - (\frac{1}{2})^{|\mathbf{T}|}$.
- For each data set ξ_m, $i = 1, \ldots, M$, execute the following:
  - Set $\mathbf{S}$ to be a randomly chosen subset of $\mathbf{T}$.
  - Compute $d = \Pr(X \not\perp Y \mid \mathbf{S}, \xi_m)$.
  - Update the posterior probability p using the recursive formula
  $$p \leftarrow \frac{pd}{pd + (1-p)(G+1-d)}$$
- Set $\Pr(Y \in \mathbf{N}(X)) = \Pr(X \in \mathbf{N}(Y)) = p$.
- Assign Y to be a member of $\mathbf{N}(X)$ and X to be in $\mathbf{N}(Y)$ if and only if $p > \frac{1}{2}$.

3. [Orient Edges]

For each $X \in \mathcal{U}, Y \in \mathbf{N}(X)$ do:

- Set $Q \leftarrow \frac{1}{2}$.
- Do for each $Z \in \mathbf{N}(X) - \mathbf{N}(Y) - \{Y\}$:
  - Set $q \leftarrow \frac{1}{2}$.
  - Set $\mathcal{U}$ to be the smaller of $\mathbf{B}(Y) - \{X,Z\}$ and $\mathbf{B}(Z) - \{X,Y\}$.
  - Let $G \leftarrow G(Y,Z) = 1 - (\frac{1}{2})^{|\mathcal{U}|}$.
  - For each data set ξ_m, $i = 1, \ldots, M$, execute the following loop:
    - Set $\mathbf{S}$ to be a randomly chosen subset of $\mathcal{U}$.
    - Compute $d = \Pr(Y \not\perp Z \mid \mathbf{S} \cup \{X\}, \xi_m)$.
    - Update the posterior probability q using the recursive formula
    $$q \leftarrow \frac{qd}{qd + (1-q)(G+1-d)}$$
  - Update $Q \leftarrow \frac{Q(1-q)}{Q(1-q)+(1-Q)(1-G+q)}$.
- Set $\Pr(Y \to X) = 1 - Q$.

For each $X \in \mathcal{U}, Y \in \mathbf{N}(X)$ do:

- Assign direction $Y \to X$ if $\Pr(Y \to X) > \Pr(X \to Y)$.
- Assign direction $X \to Y$ if $\Pr(Y \to X) < \Pr(X \to Y)$.

4. [Remove Cycles]

Do the following while there exist cycles in the graph:

- Compute the set of edges $\mathbf{C} = \{X \to Y$ such that $X \to Y$ is part of a cycle$\}$.
- Remove the edge $X \to Y$ in $\mathbf{C}$ that such that $\Pr(X \in \mathbf{N}(Y)) \Pr(X \to Y)$ is minimum and put it in $\mathbf{R}$.

5. [Reverse Edges] (same as plain GS)

Insert each edge from $\mathbf{R}$ in the graph, reversed.

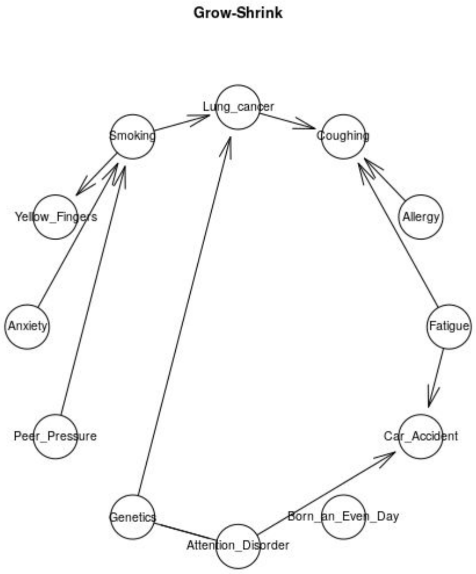
6. [Propagate Directions] (same as plain GS)

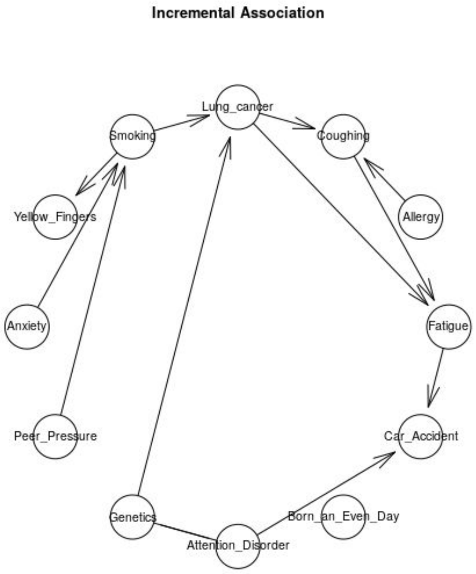
For all $X \in \mathcal{U}$ and $Y \in \mathbf{N}(X)$ such that neither $Y \to X$ nor $X \to Y$, execute the following rule until it no longer applies: If there exists a directed path from X to Y, orient $X \to Y$.

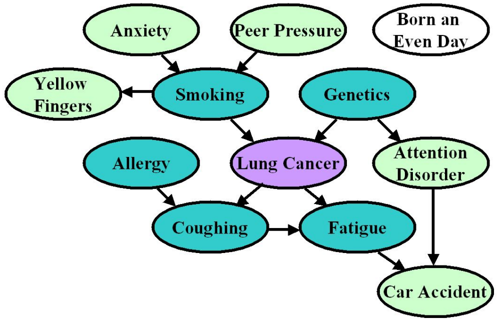
Figure 3.9: The randomized GS algorithm

Learning Bayesian Network Model Structure from Data, Phd Thesis, Dimitris Margaritis

bnlearn - Results - Constrained based



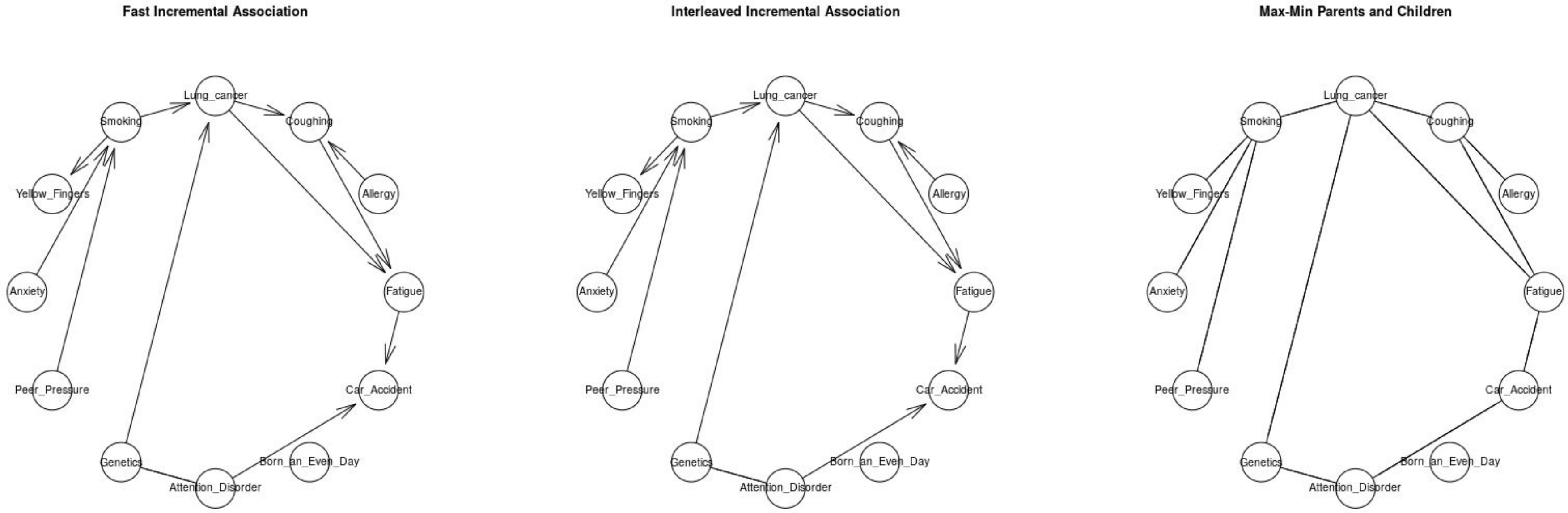




Dataset - Overview
http://www.causality.inf.ethz.ch/data/LUCAS.html

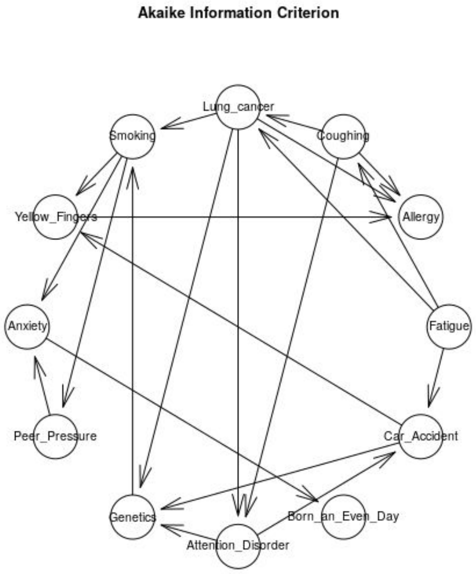
Constrained based structure learning

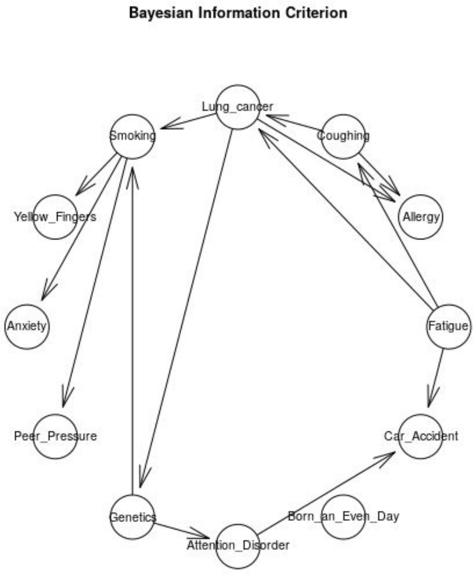
bnlearn - Results - Constrained based

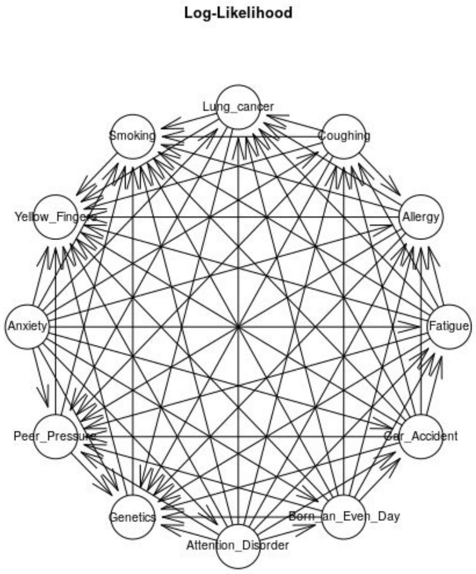


Constrained based structure learning

bnlearn - Results - Hill Climbing





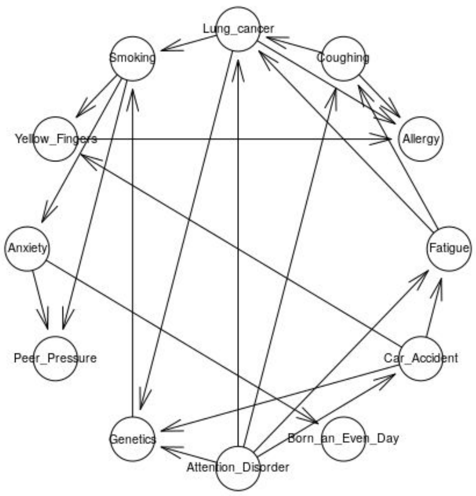


Score based methods for structure learning

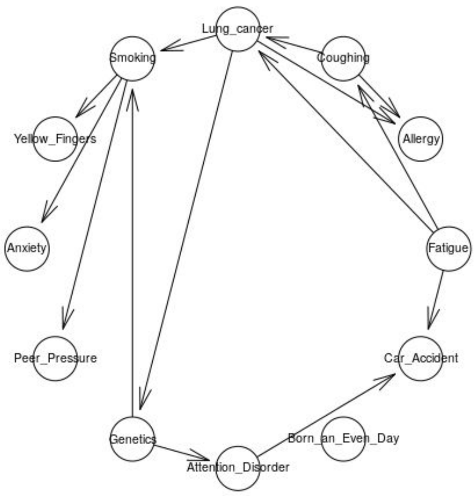
bnlearn - Results - Tabu Search

TECHNISCHE
UNIVERSITÄT
DARMSTADT

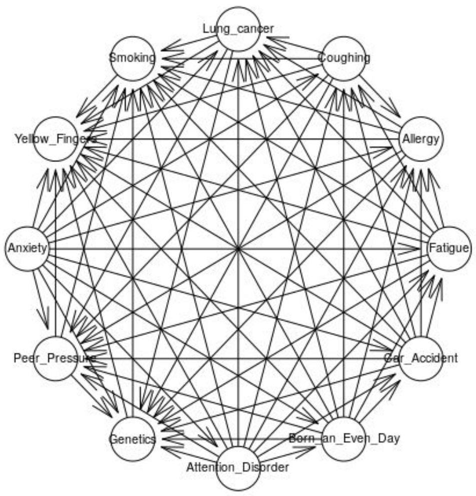
Akaike Information Criterion



Bayesian Information Criterion



Log-Likelihood



Score based methods for structure learning

Conclusion

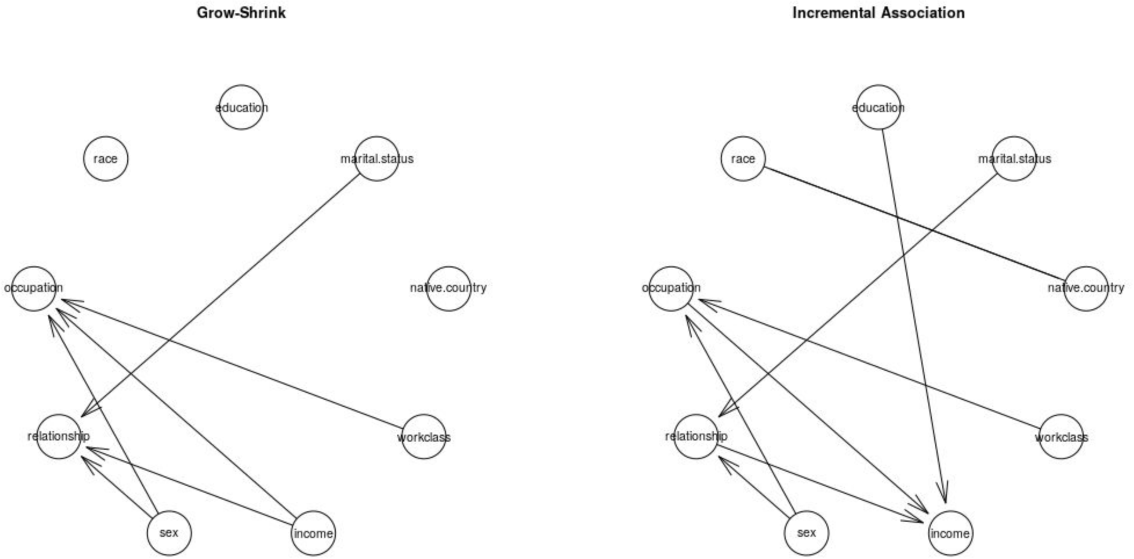
- **bnlearn** seems to be a good, out of the box framework for structure learning
- Constrained based such **Incremental Association** algorithms often outperform score based methods
- It’s possible to infer the correct structure and edge directions with constrained based methods
- If an edge is constructed by multiple algorithms in the same way, it’s more likely that this particular edge is correct
  - -> enables pseudo certainty for network components

References

- Graphical Models for Probabilistic and Causal Reasoning, Judea Pearl
- Towards A Rigorous Science of Interpretable Machine Learning, Finale Doshi-Velez, Bee Kim
- Partial orientation and local structural learning of causal networks for prediction, Jianxin Yin, You Zhou, Changzhang Wang, Ping He, Cheng Zheng, Zhi Geng
- An Exploration of Structure Learning in Bayesian Networks, Constantin Berzan
- Learning Bayesian Networks with the bnlearn R Package
- Learning Bayesian Network Model Structure from Data, Dimitris Margaritis

https://github.com/QueensGambit/PGM-Causal-Reasoning

bnlearn - Results - Constrained based

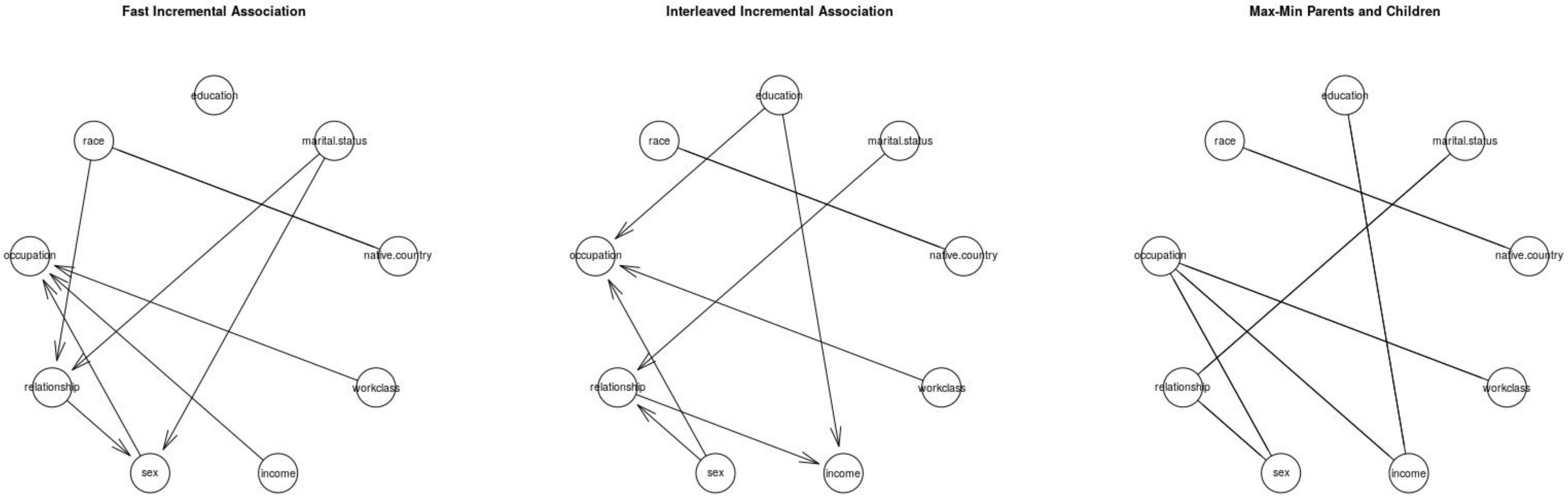


Constrained based structure learning

CINA-dataset
(**C**ensus **Is** **N**ot **A**dult dataset)

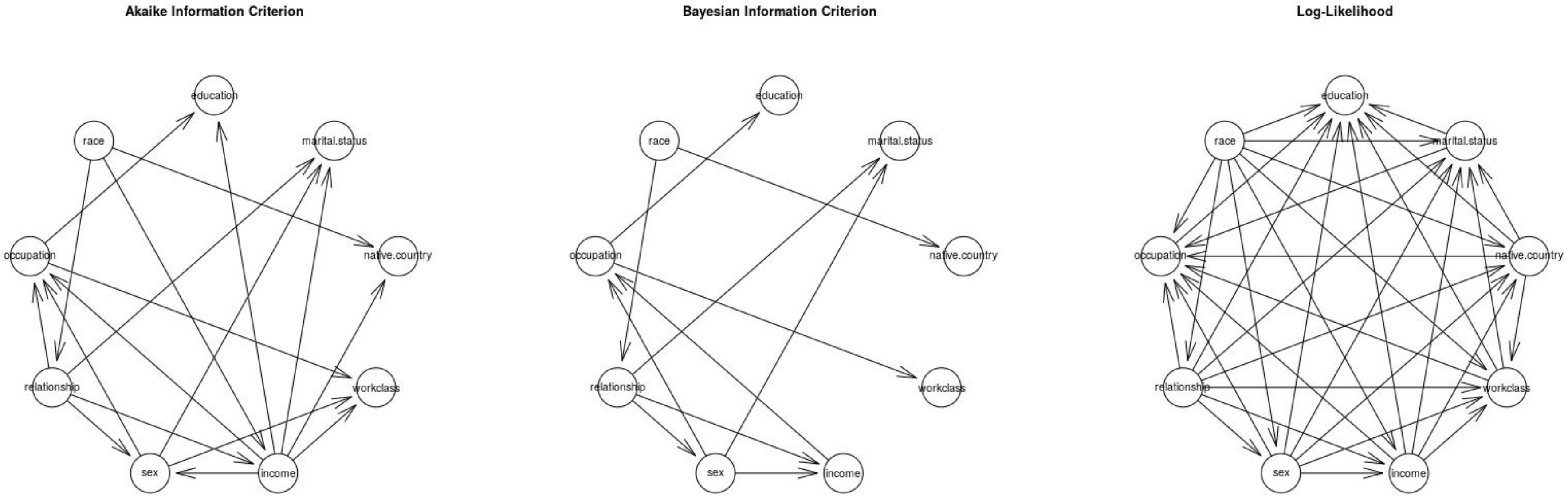
extracted from the 1994 Census database
age: continuous.
workclass: Private, Self-emp-not-inc, ...
fnlwgt: continuous.
education: Bachelors, Some-college,...
education-num: continuous.
marital-status: Married-civ-spouse, Divorced,
occupation: Tech-support, Craft-repair, …
relationship: Wife, Own-child, Husband, …
race: White, Asian-Pac-Islander, …
sex: Female, Male. ...
capital-gain: continuous.
capital-loss: continuous.
hours-per-week: continuous.
native-country: United-States, Cambodia,...
England, Puerto-Rico, Canada, Germany,…
income: >50K, <=50K

bnlearn - Results - Constrained based



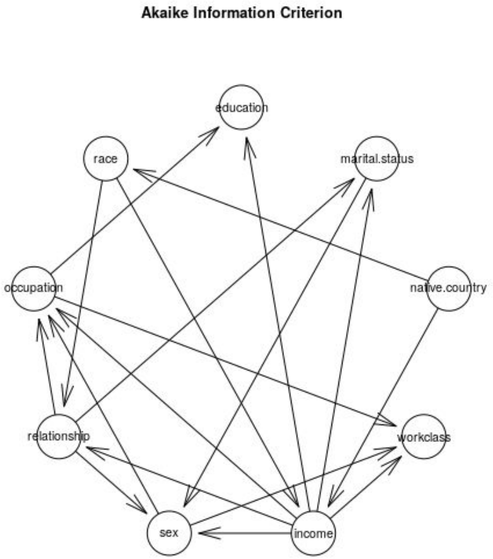
Constrained based structure learning

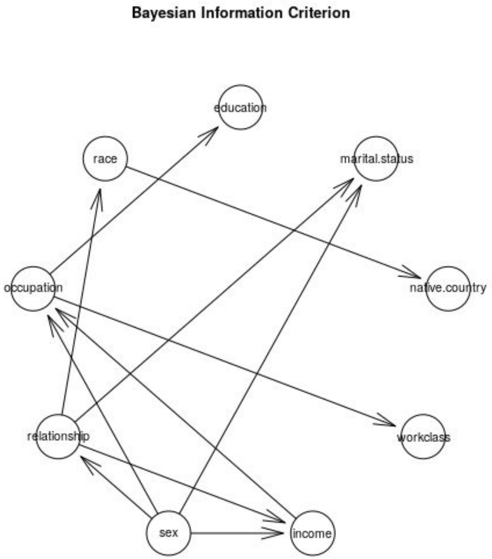
bnlearn - Results - Hill Climbing



Score based methods for structure learning

bnlearn - Results - Tabu Search





Score based methods for structure learning

Network scores (e.g. used in Hill Climbing)

- Likelihood / Log-Likelihood
- Akaike and Bayesian Information Criterion (AIC/BIC)

$$\mathrm{AIC} = \log \mathrm{L}(X_1, \ldots, X_v) - d \qquad \mathrm{BIC} = \log \mathrm{L}(X_1, \ldots, X_v) - \frac{d}{2} \log n$$

- Bayesian Dirichlet Equivalent Score (BDE)
- K2 score

$$\mathrm{K2} = \prod_{i=1}^{v} \mathrm{K2}(X_i), \qquad \mathrm{K2}(X_i) = \prod_{j=1}^{L_i} \frac{(R_i - 1)!}{\left(\sum_{k=1}^{R_i} n_{ijk} + R_i - 1\right)!} \prod_{k=1}^{R_i} n_{ijk}!$$