Učení struktury

- Strom: Chow–Liu Tree
- Prohledávání 'všech' struktur
- PC–algoritmus, Orientace hran

Definition (KL–divergence)

KL–divergence dvou pravděpodobnostních rozložení P, Q na stejné doméně $sp(P) = sp(Q)$ je definovaná jako: $D_{KL}(P||Q) = \sum_{i \in sp(P)} P(i) log \frac{P(i)}{Q(i)}$.

Definition (Entropie)

Entropie pravděpodobnostního rozložení P na doméně $sp(P)$ je definovaná jako: $H(P) = -\sum_{i \in sp(P)} P(i) log P(i)$.

Definition (Vzájemná informace (Mutual Information))

Mutual Information dvou veličin X, Y na doménách $sp(X), sp(Y)$ je definovaná jako: $I(X; Y) = \sum_{i \in sp(X)} \sum_{j \in sp(Y)} P(X = i, Y = j) log \frac{P(X=i, Y=j)}{P(X=i)P(Y=j)}$.

Chow–Liu Tree

Lemma

Pro modely s interakcemi maximálně druhého řádu, tj. stromy, platí pro D_{KL} aproximace P' a vzoru P

$$D_{KL}(P||P') = -\sum I(X_i, X_{j(i)}) + \sum H(X_i) - H(X_1, \ldots, X_n)$$

kde $X_{j(i)}$ je rodič vrcholu X_i.

Algoritmus Chow–Liu

Calculate $I(A, B)$ for each pair of nodes
Find maximal spanning tree (kostru)
Orient edges (no head–to–head connection)
Learn parameters.

```
 library('gRapHD')
bf=minForest(sim.data)
```

Prohledávání 'všech' BN

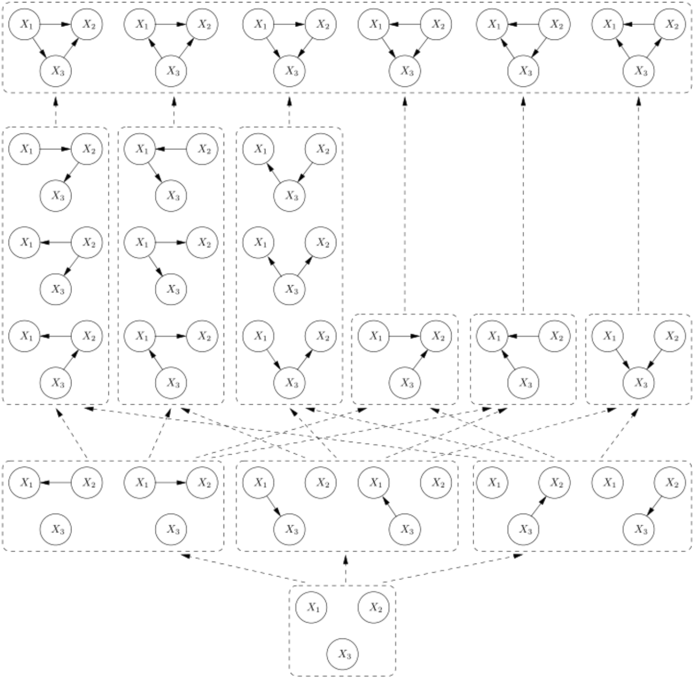
- Budeme prohledávat prostor všech BN.
- Lepší: prohledávat třídy ekvivalence BN – esenciální grafy.
- nebo: triangulované, rozložitelné (neorientované) grafy, orientaci dodat časem.
- Maximalizujeme většinou BIC kriterium, případně AIC a jiná.

Definition (BIC,AIC, loglik)

A BIC criterium for a model S with parameters θ_S compared to N data samples $\mathcal{D}$ is

- Bayesian $BIC(S, \mathcal{D}) = logP_S(\mathcal{D}|\hat{\theta_S}) - \frac{1}{2}|\theta_S|log(N)$
- Akaike's IC $AIC(S, \mathcal{D}) = logP_S(\mathcal{D}|\hat{\theta_S}) - |\theta_S|$
- $loglik(S, \mathcal{D}) = logP_S(\mathcal{D}|\hat{\theta_S})$.

- Kriteria BIC, AIC můžeme vnímat jako maximální věrohodnost penalizovanou složitostí modelu.



X1
X2
X3

Algoritmus **Učení struktury BN**

hc=function($\mathcal{D}$)
  Zvol strukturu S (vše nezávislé, Chow–Liu Tree)
  opakuj
    odhadni parametry θ_s (max. lik., smooth, bayes)
    $currscore = BIC(S, \mathcal{D})$
    S=modifikuj(S) (přidej nejslibnější hranu, uber nejzbytečnější,...)
  dokud chceš (*currscore* se zlepšuje, vyčerpán čas, ...)
  return(S)

```
 chick.bn=stepw(bf,sim.chest) # z lib. gRapHD
chest.net=hc(sim.chest,blacklist=bl) # z library('bnlearn')
```

Orientace hran

- Ne všechny orientace se dají naučit z dat.
- Můžeme použít statistický test na určení podmíněné ne–závislosti veličin,
- pomocí čtyř pravidel (Thomas Nielsen, PC–Algorithm) doplníme orientaci.

Constraint based learning

Some notation

- To denote that A is conditionally independent of B given $\mathcal{X}$ in the database we shall use

$$I(A, B, \mathcal{X}).$$

Some assumptions

- The database is a faithful sample from a Bayesian network M: A and B are d-separated given $\mathcal{X}$ in M if and only if $I(A, B, \mathcal{X})$.
- We have an oracle that correctly answers questions of the type:
  "Is $I(A, B, \mathcal{X})$?"

The algorithm

Use the oracle's answers to first establish a skeleton of a Bayesian network:

- The skeleton is the undirected graph obtained by removing directions on the arcs.

Next, when the skeleton is found we then start looking for the directions on the arcs.

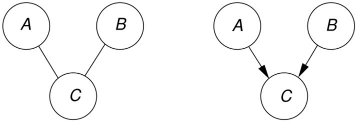
Finding the skeleton II

The idea

If there is a link between A and B in M then they cannot be d-separated, and as the data is faithful it can be checked by asking questions to the oracle:

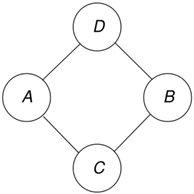
- The link $A - B$ is part of the skeleton if and only if $\neg I(A, B, \mathcal{X})$, for all $\mathcal{X}$.

Assume that the only conditional independence found is $I(A, B)$:



The skeleton   The only possible DAG

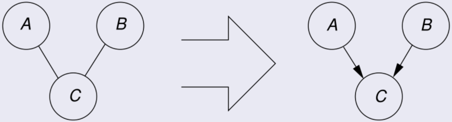
Assume that the conditional independences found are $I(A, B, D)$ and $I(C, D, \{A, B\})$:



Setting the directions on the links I

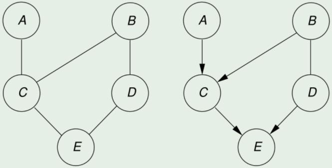
Rule 1

If you have three nodes, A, B, C such that $A - C$ and $B - C$, but not $A - B$, then introduce the v-structure $A \rightarrow C \leftarrow B$ if there exists an $\mathcal{X}$ (possibly empty) such that $I(A, B, \mathcal{X})$ and $C \notin \mathcal{X}$.



Example

Assume that we get the independences $I(A, B)$, $I(A, D)$, $I(A, B, D)$, $I(A, D, B)$, $I(C, D, B)$, $I(A, D, \{B, C\})$, $I(C, D, \{A, B\})$, $I(B, E, \{C, D\})$, $I(A, E, \{C, D\})$, $I(A, D, \{B, C, E\})$, $I(A, E, \{B, C, D\})$, $I(B, E, \{A, C, D\})$.



Setting the directions on the links II

Rule 2 [Avoid new v-structures]

When Rule 1 has been exhausted, and you have $A \rightarrow C - B$ (and no link between A and B), then direct $C \rightarrow B$.

Rule 3 [Avoid cycles]

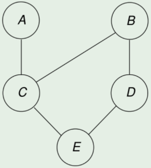
If $A \rightarrow B$ introduces a directed cycle in the graph, then do $A \leftarrow B$.

Rule 4 [Choose randomly]

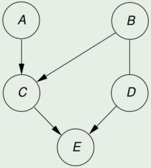
If none of the rules 1-3 can be applied anywhere in the graph, choose an undirected link and give it an arbitrary direction.

Example

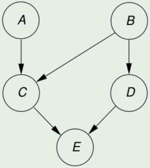
Skeleton:



Rule 1:

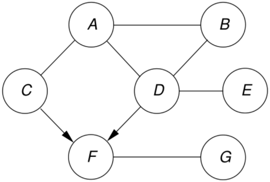


Rule 4:



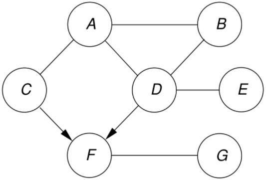
Another example

Consider the graph:

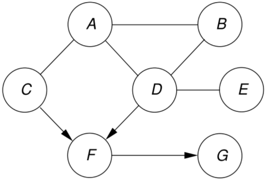


Apply the four rules to learn a Bayesian network structure

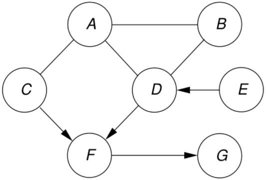
Another example I



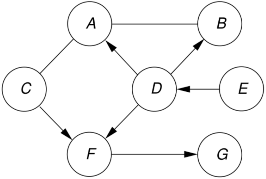
Step 1: Rule 1



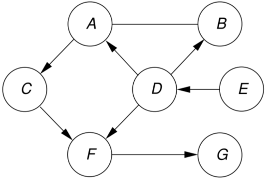
Step 2: Rule 2



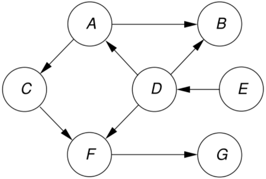
Step 3: Rule 4



Step 4: Rule 2



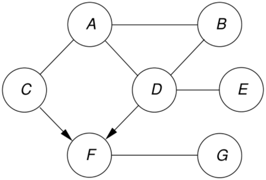
Step 5: Rule 2



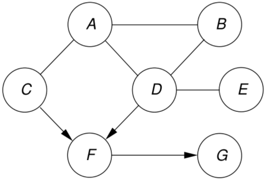
Step 6: Rule 4

However, we are not guaranteed a unique solution!

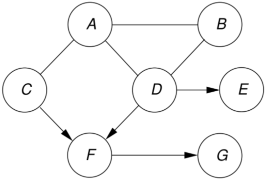
Another example II



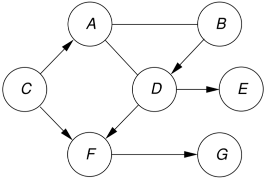
Step 1: Rule 1



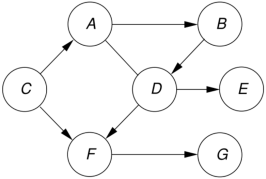
Step 2: Rule 2



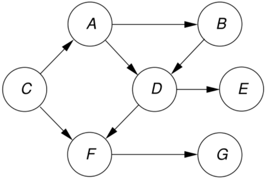
Step 3: Rule 4



Step 4: Rule 4



Step 5: Rule 2



Step 6: Rule 2+3

Although the solution is not necessarily unique, all solutions have the same d-separation properties!

From independence tests to skeleton

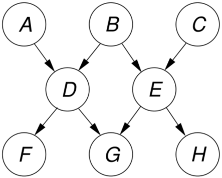
Until now, we have assumed that all questions of the form "Is $I(A, B, \mathcal{X})$?" can be answered (allowing us to establish the skeleton). However, questions come at a price, and we would therefore like to ask as few questions as possible.

To reduce the number of questions we exploit the following property:

Theorem

The nodes A and B are not linked if and only if $I(A, B, \mathrm{pa}(A))$ or $I(A, B, \mathrm{pa}(B))$.

It is sufficient to ask questions $I(A, B, \mathcal{X})$, where $\mathcal{X}$ is a subset of A's or B's neighbors.



An active path from A to B must go through a parent of B.

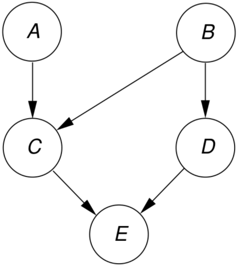
The PC algorithm

The PC algorithm:

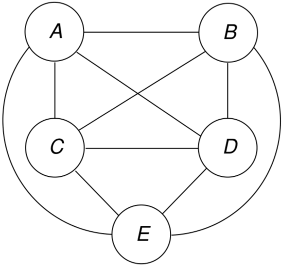
1. Start with the complete graph;
2. $i := 0$;
3. **while** a node has at least $i + 1$ neighbors
   - **for all** nodes A with at least $i + 1$ neighbors
     - **for all** neighbors B of A
       for all neighbor sets $\mathcal{X}$ such that $|\mathcal{X}| = i$ and $\mathcal{X} \subseteq (\mathrm{nb}(A) \setminus \{B\})$
       if $I(A, B, \mathcal{X})$ **then** remove the link $A - B$ and store "$I(A, B, \mathcal{X})$"
   - $i := i + 1$

Example

We start with the complete graph and ask the questions $I(A, B)$?, $I(A, C)$?, $I(A, D)$?, $I(A, E)$?, $I(B, C)$?, $I(B, D)$?, $I(B, E)$?, $I(C, D)$?, $I(C, E)$?, $I(D, E)$?.



The original model



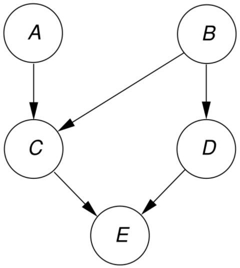
The complete graph

We get a "yes" for $I(A, B)$? and $I(A, D)$?:

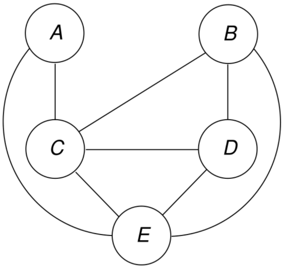
- the links $A - B$ and $A - D$ are therefore removed.

Example

We now condition on one variable and ask the questions $I(A, C, E)$?, $I(A, E, C)$?, $I(B, C, D)$?, $I(B, C, E)$?, $I(B, D, C)$?, $I(B, D, E)$?, $I(B, E, C)$?, $I(B, E, D)$?, $I(C, B, A)$?, ..., $I(C, D, A)$?, $I(C, D, B)$?.



The original model



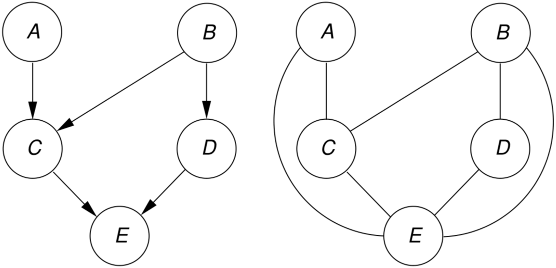
After one iteration

The question $I(C, D, B)$? has the answer "yes":

- we therefore remove the link $C - D$.

Example

We now condition on two variables and ask questions like $I(B, C, \{D, E\})$?.



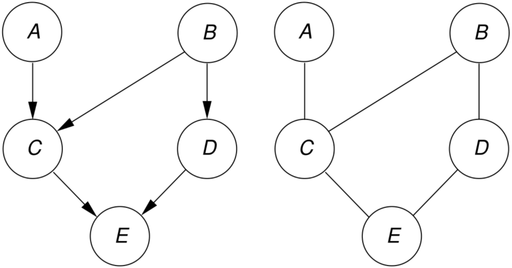
The original model After two iterations

The questions $I(B, E, \{C, D\})$? and $I(E, A, \{D, C\})$? have the answer "yes":

- we therefore remove the links $B - E$ and $A - E$.

Example

We now condition on three variables, but since no nodes have four neighbors we are finished.



The original model After three iterations

The identified set of independence statements are then $I(A, B)$, $I(A, D)$, $I(C, D, B)$, $I(A, E, \{C, D\})$, and $I(B, E, \{C, D\})$. They are sufficient for applying rules 1-4.

Real world data

The oracle is a statistical test, e.g. conditional mutual information:

$$CE(A, B|X) = \sum_{X} P(X) \sum_{A,B} P(A, B|X) \log \frac{P(A, B|X)}{P(A|X)P(B|X)}.$$

$$I(A, B, X) \Leftrightarrow CE(A, B|X) = 0.$$

However, all tests have false positives and false negatives, which may provide false results/causal relations!

Similarly, false results may also be caused by hidden variables: