

Causal graphical models and interventions

Florian Markowetz

Max Planck Institute for Molecular Genetics
– Computational Molecular Biology –
Berlin, Germany
<http://compdiag.molgen.mpg.de/>



Group meeting
Monday, January 12

— Classical Graphical Models —

Graphical models are undirected graphs encoding statements of conditional independence.

A missing edge between two variables X_i and X_j is interpreted as:

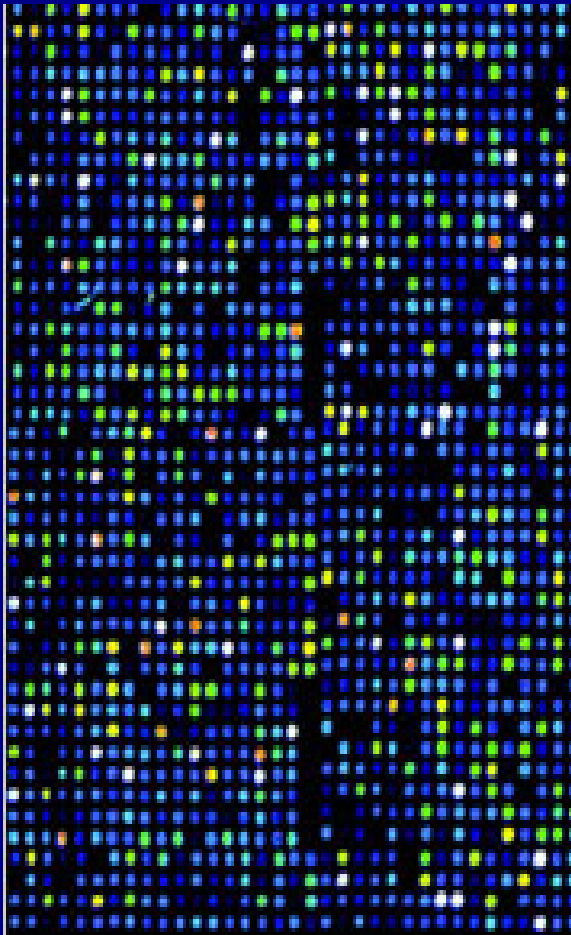
$$X_i \perp X_j \mid (\text{the rest})$$

In the Gaussian case, building a graphical model amounts to testing, which entries of the precision matrix Σ^{-1} are zero.

Graphical models can be used to elucidate the dependence structure between gene expression levels measured in microarray experiments.



— Genetic networks —



- Microarrays provide a snapshot of gene expression in a cell. Genes are not expressed independently, they regulate each others activity.
- **Goal: Reconstruct the gene regulation network.**
 - Clustering points to functional relationships.
 - Undirected graphical models evaluate conditional independence restrictions.
- **Causality, not correlation!** Is the effect of a mutated gene on a target direct, or mediated by other genes? What is the nature of the interaction between genes (e.g. does gene A inhibit gene B)?



— Bayesian network —

A **Bayesian Network** for $\mathbf{X} = \{X_1, \dots, X_n\}$ consists of

1. a **network structure** \mathcal{S}

- directed acyclic graph (DAG),
- nodes \leftrightarrow variables = genes,

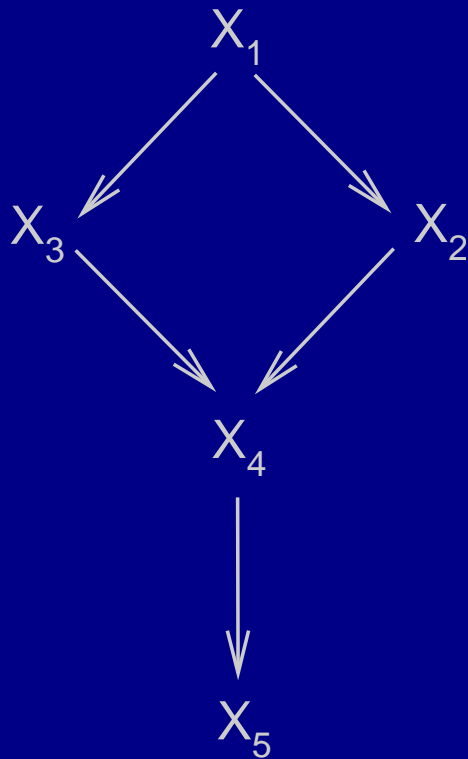
2. a set of **probability distributions** \mathcal{P}

- locally: conditional distribution of a variable X_i given its parents pa_i in the graph \mathcal{S} :

$$\mathcal{P} = \{ P(X_i \mid X_{pa(i)}) \}$$



— Bayesian networks *cont'd* —



$(\mathcal{S}, \mathcal{P})$ encode the joint distribution:

$$\begin{aligned} P(\mathbf{X}) &= \prod_{i=1}^n P(X_i \mid X_1, \dots, X_{i-1}) \\ &= \prod_{i=1}^n P(X_i \mid X_{pa(i)}) \end{aligned}$$

The DAG structure \mathcal{S} depends on the ordering of the variables.



— Causal Models —

In a causal model we use DAGs in two ways:

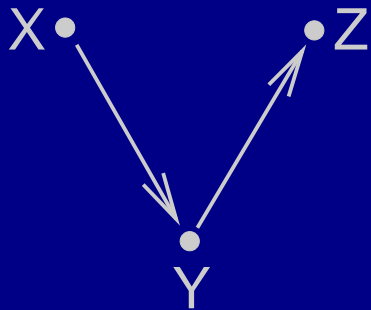
1. as a carrier of **independence assumptions**, and
2. as a representation of **direct functional relationships**.

The structure of a Bayesian network depends on the ordering of the variables.

Not every Bayesian network is a representation of causal mechanisms.

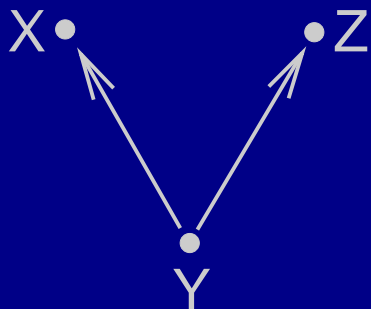


— Conditional Independence —



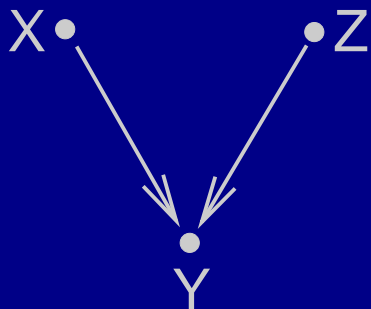
Chain/linear

$$X \perp Z \mid Y \quad \text{and} \quad X \not\perp Z \mid \emptyset$$



Fork/diverging

$$X \perp Z \mid Y \quad \text{and} \quad X \not\perp Z \mid \emptyset$$



Collider/converging

$$X \perp Z \mid \emptyset \quad \text{and} \quad X \not\perp Z \mid Y$$



— d-separation —

Definition: A path p in a DAG G is said to be d-separated (or blocked) by a set of nodes Y if and only if

1. p contains a **chain** $i \rightarrow m \rightarrow j$ or a **fork** $i \leftarrow m \rightarrow j$ such that the middle node m is in Y , or
2. p contains a **collider** $i \rightarrow m \leftarrow j$ such that the middle node m is NOT in Y and such that no descendent of m is in Y .

A set Y is said to d-separate X from Z if and only if Y blocks every path from a node in X to a node in Z .



— SGS algorithm —

Step 1: Build the skeleton.

- Form the complete undirected graph \mathcal{C} on the node set $\mathbf{X} = \{X_1, \dots, X_n\}$.
- For each pair of variables X_i and X_j :
 - if** there exists a subset $\mathbf{S} \subset \mathbf{X} \setminus \{X_i, X_j\}$ such that $X_i \perp X_j \mid \mathbf{S}$,
then remove the edge $X_i - X_j$ from \mathcal{C} .

Step 2: Direct the edges.

- SGS and Pearl give three simple rules.



— Nature hides more than it shows —

We assume that the interaction of genes is organised as a Causal Bayesian network. But Nature hides this network from us.

Our goal: reconstruct a genetic network from microarray data.

We only observe data from the joint distribution of the genes.

How far does this help us to resolve network structure?
Are there natural limits?



— Nature hides more than it shows —

We assume that the interaction of genes is organised as a Causal Bayesian network. But Nature hides this network from us.

Our goal: reconstruct a genetic network from microarray data.

We only observe data from the joint distribution of the genes.

How far does this help us to resolve network structure?
Are there natural limits?

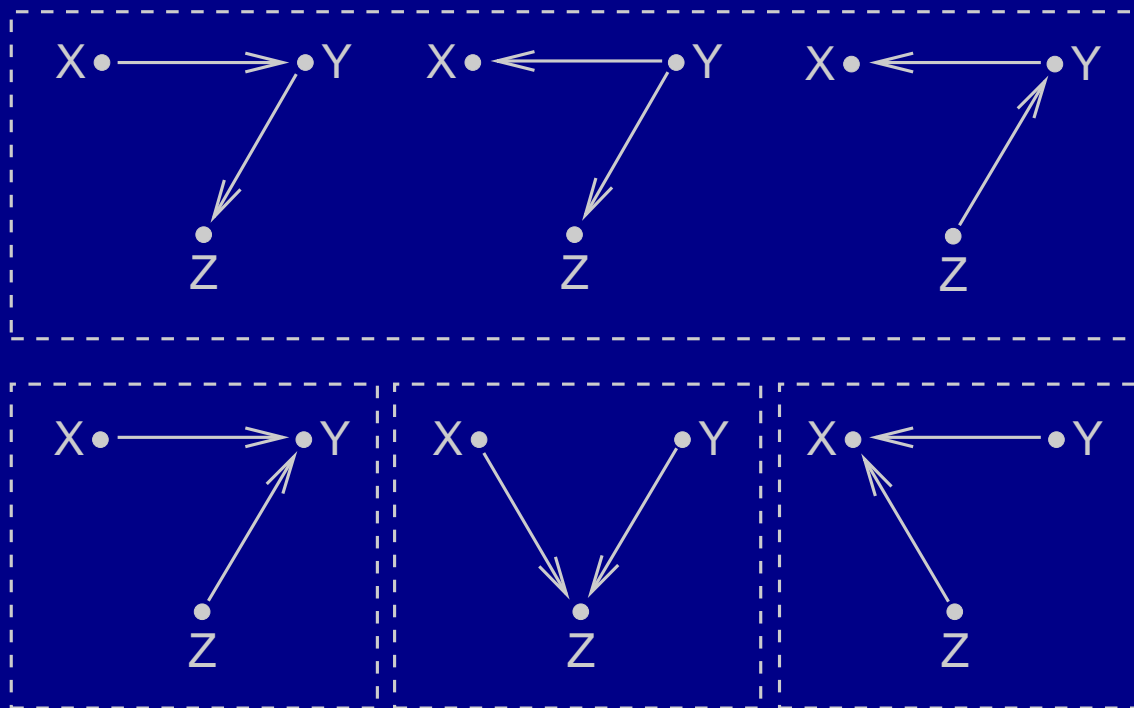
Yes, there are: some structures are statistically indistinguishable.



— Equivalence of Bayesian networks —

Markov equivalence

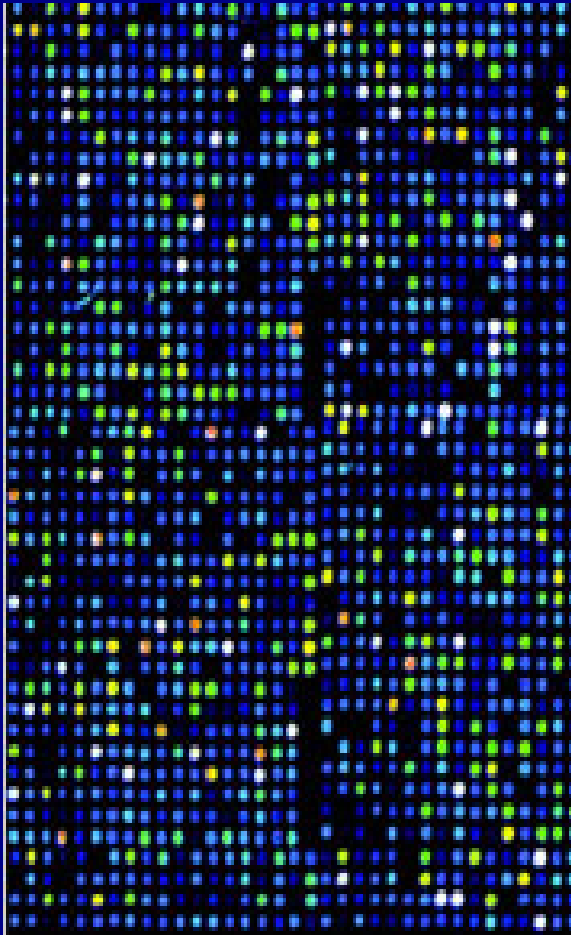
$\mathcal{S}_1 \stackrel{M}{\sim} \mathcal{S}_2$ if both structures represent the same set of independence assertions.



Even with infinitely many observations we cannot decide between the DAGs in the same equivalence class.



— Observation and Intervention —

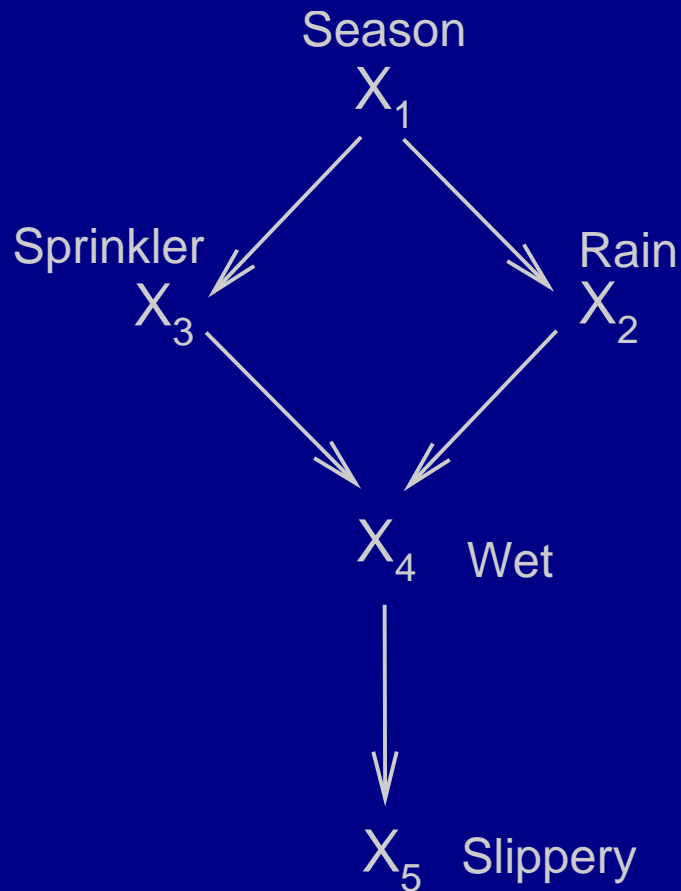


What effect does this result have on the reconstruction of genetic networks by BN?

- Arrows in the BN do not necessarily represent causal influence! From observations alone we can only learn whole equivalence classes, in general not a single DAG.
- But biologists not only observe, they also **intervene, perturb, disrupt** the gene network e. g. by knock-out experiments.
- **How can we model interventions in Bayesian networks?**



— An example: the sprinkler network —



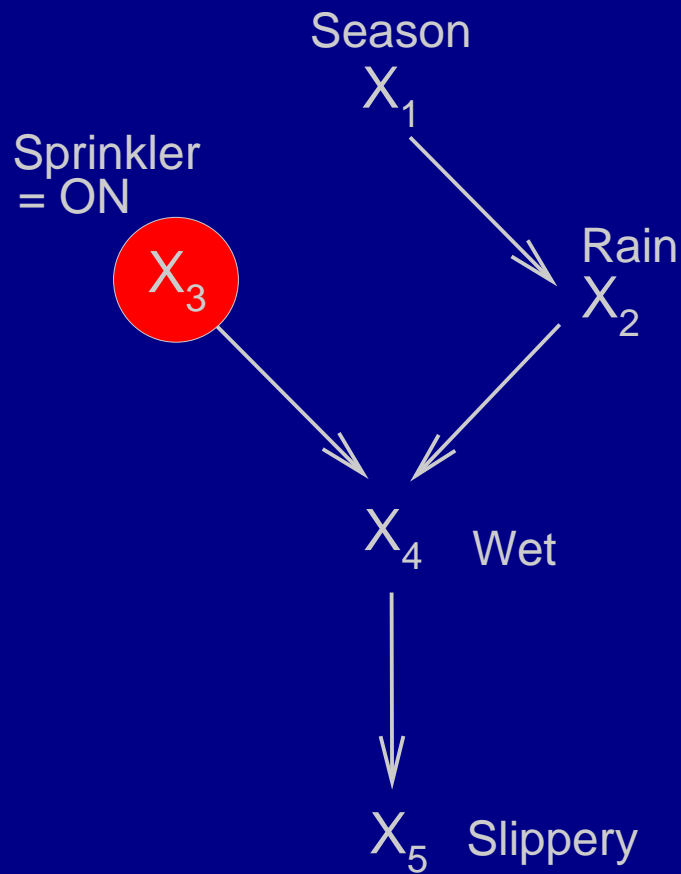
Imagine this being the true causal model we want to induce.

By looking out of the window we can collect observations of the states of all five variables.

What happens if we go out and turn the sprinkler on?



— Human manipulation —



Human intervention is a deterministic manipulation by forces outside the causal network model.

The sprinkler is no longer under the influence of any variables in the model, and thus, the arcs into it should be removed.

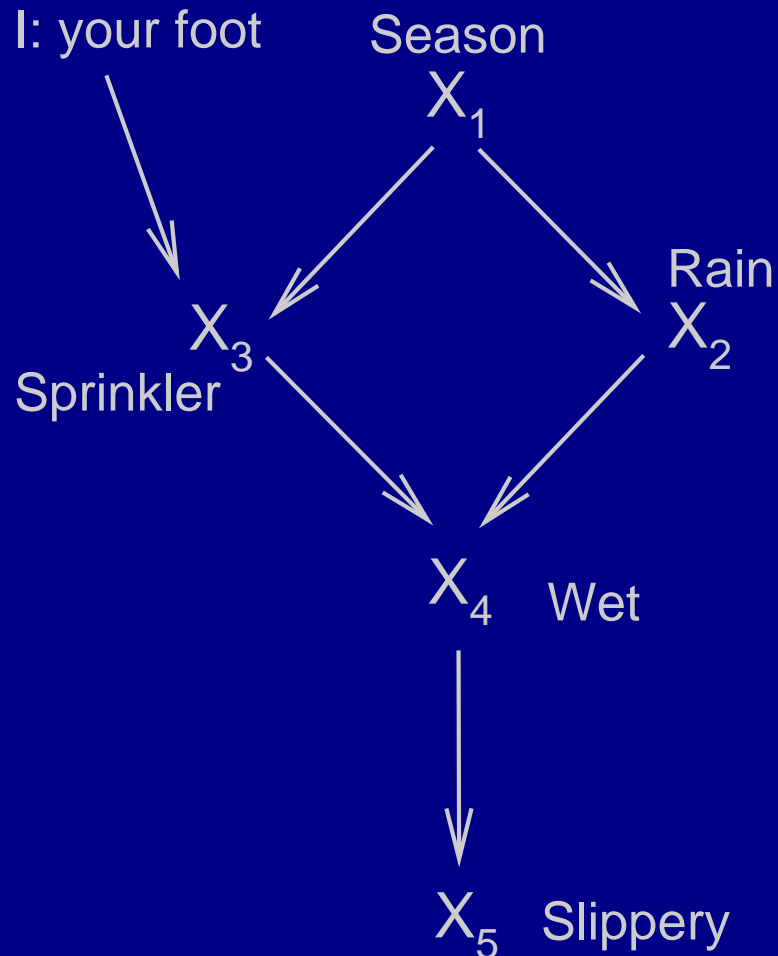
Intervention at X_3 :

- cut the edges from the pa_3 to X_3
- and set $P(X_3 \leftarrow \text{ON}) = 1$.

This manipulation is quite hard, let's be softer ...



— A foot on the water pipe —



Now imagine, you don't want to fix the state of the sprinkler, you just want to reduce it's output. You put your foot on the pipe connecting the sprinkler to the tap.

We model this by introducing a new variable I which represents the inhibiting force (your foot!).

Inhibition at X_3 :

- add I as a new parent to X_3
- and "push the distribution" to the lower state.



— Knock-outs and RNAi —

How does the sprinkler example relate to biology?

- Forcing a node to a state deterministically \sim knock-out
- Putting a foot an the pipe \sim RNAi

In this model, both knock-out and RNAi only involve single nodes and their parents.



— Bayesian model selection —

The bayesian way of learning a model structure from data:

1. **Scoring:** introduce a scoring function that evaluates each network with respect to the training data.
2. **Searching:** search for the optimal network according to this score.

The number of graphs grows exponentially in the number of nodes. For more than 5 nodes an exhaustive search is intractable.

Use heuristics: hill-climbing (with random restarts), simulated annealing, ...



— The Bayesian score —

The score evaluates the **posterior probability** of a graph given the data:

$$\begin{aligned} \text{Score}(\text{dag} : \text{data}) &= \log P(\text{dag} | \text{data}) \\ &= \log P(\text{data} | \text{dag}) + \log P(\text{dag}) + C \end{aligned}$$

where C is a constant and $P(\text{data} | \text{dag})$ is the **marginal likelihood**:

$$P(\text{data} | \text{dag}) = \int P(\text{data} | \text{dag}, \text{para}) P(\text{para} | \text{dag}) d\text{para}$$

For **Gaussian or multinomial models**, the posterior can be given explicitly and it is **decomposable**:

$$\text{Score}(\text{dag} : \text{data}) = \sum_i \text{FamilyScore}(\text{node}_i, \text{parents}_i : \text{data}).$$



— Learning from observations —

For **multinomial models**, the marginal likelihood can be derived as:

$$P(\text{data} \mid \text{dag}) = \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij+})}{\Gamma(\alpha_{ij+} + N_{ij+})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})}$$

r_i = number of states of node X_i ,

q_i = number of joint states of parents of X_i ,

N_{ijk} = number of times we observe node X_i in state k given parental state j ,

α_{ijk} = parameter of the Dirichlet prior $P(\text{para} \mid \text{dag})$.

Multiplying $P(\text{data} \mid \text{dag})$ with a prior $P(\text{dag})$ reflecting your prior knowledge on network structure yields a scoring metric $\text{Score}(\text{dag} : \text{data})$.



— Learning by hard interventions: knock-outs —

The likelihood for data from hard interventions looks only slightly different:

$$P(\text{data} \mid \text{dag}) = \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij+})}{\Gamma(\alpha_{ij+} + N_{ij+}^{\text{obs}})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk}^{\text{obs}})}{\Gamma(\alpha_{ijk})}$$

Here N_{ijk}^{obs} is the number of **passive observations** $X_i = k \mid X_{pa(i)} = j$.

The interventions vanish in the calculations, because there $P(X_i \leftarrow k) = 1$.



— Learning by soft interventions: RNAi —

For soft interventions it's a bit more complicated, though:

$$\begin{aligned}
 P(\text{data} \mid \text{dag}) &= \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij+})}{\Gamma(\alpha_{ij+} + N_{ij+}^{\text{obs}})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk}^{\text{obs}})}{\Gamma(\alpha_{ijk})} \\
 &\times \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij+}^{\text{int}})}{\Gamma(\alpha_{ij+}^{\text{int}} + N_{ij+}^{\text{int}})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk}^{\text{int}} + N_{ijk}^{\text{int}})}{\Gamma(\alpha_{ijk}^{\text{int}})}
 \end{aligned}$$

We used the decomposition $N_{ijk} = N_{ijk}^{\text{obs}} + N_{ijk}^{\text{int}}$, with

$$\begin{array}{ll}
 N_{ijk}^{\text{obs}} &= \text{number of passive observations} & X_i = k \mid X_{pa(i)} = j \\
 N_{ijk}^{\text{int}} &= \text{number of experimental interventions} & X_i \leftarrow k \mid X_{pa(i)} = j
 \end{array}$$

The Dirichlet parameters α_{ijk} are now different for passive observations and experimental interventions.



— References —

1. David Edwards: *Introduction to Graphical Modelling*, Springer, 2000
2. Judea Pearl: *Causality: Models, Reasoning and Inference*, Cambridge University Press, 2000
3. Spirtes, Glymour, and Scheines: *Causation, Prediction, and Search*, MIT Press, 2000
4. Cooper and Yoo, *Causal discovery from a mixture of experimental and observational data*, UAI 1999



— Thank you! Questions? —



— Appendix —

1. Three rules to direct the edges in the SGS algorithm,
2. Derivation of the Bayesian score



— SGS algorithm Step 2: direct the edges. —

Let \mathcal{D} be the undirected graph resulting from Step 1 (the skeleton of the DAG).

- **Find v-structures:** For each triple (X_i, X_j, X_k) , such that $X_i - X_j - X_k$ and $X_i \not\sim X_k$:
 - iff there is no $S \subset \{X_j\} \cup \mathbf{X} \setminus \{X_i, X_k\}$ that d-separates X_i and X_k ,
 - then orient $X_i - X_j - X_k$ as $X_i \rightarrow X_j \leftarrow X_k$.
- **If** $X_i \rightarrow X_j$, $X_j - X_k$, $X_i \not\sim X_k$, and there is no arrowhead at X_k ,
then orient $X_j - X_k$ as $X_j \rightarrow X_k$.
- **If** there is a directed path from X_i to X_j , and $X_i - X_j$,
then orient $X_i - X_j$ as $X_i \rightarrow X_j$.



— The problem —

Learn the dependency structure S of n variables X_1, \dots, X_n .

A dataset D is a collection of m cases: $D = \{C_1, \dots, C_m\}$.

Each case C_h consists in realisations of all the variables: $C_h = (x_1^h, \dots, x_n^h)$.

Our background knowledge is denoted by K . It contains e. g. the information, how the cases were experientially collected.

COOPER AND YOO, *Causal discovery from a mixture of experimental and observational data*, UAI 1999

COOPER AND HERSKOVITS, *A Bayesian method for the induction of probabilistic networks from data*, Machine Learning, 9, 309-347 (1992)



— Assumption 1 —

Causal relationships are represented using causal Bayesian networks.



— Assumption 1 —

Causal relationships are represented using causal Bayesian networks.

$$\begin{aligned} P(S \mid D, K) &\propto P(S, D \mid K) \\ &= P(S \mid K) P(D \mid S, K) \\ &= P(S \mid K) \int \underline{P(D \mid S, \theta_S, K)} P(\theta_S \mid S, K) d\theta_S \end{aligned}$$



— Assumption 2 —

The cases in D are a random sample from the joint distribution given by a causal Bayesian network B with structure S and parameters θ_S



— Assumption 2 —

The cases in D are a random sample from the joint distribution given by a causal Bayesian network B with structure S and parameters θ_S

This implies that cases are independent *conditioned on the generating model*:

$$P(D \mid S, \theta_S, K) = P(C_1, \dots, C_m \mid S, \theta_S, K) = \prod_{h=1}^m P(C_h \mid S, \theta_S, K),$$



— Assumption 3 —

For each experimentally manipulated variable X_i in case C_h , the probability $P(C_h \mid S, \theta_S, K)$ is modeled by removing from S the arcs into X_i , and setting $P(X_i = k \mid K) = 1$, where k is the value to which X_i was manipulated.

Consider a case C_h that contains a variable X_i that is manipulated to state k . $P(C_h \mid \theta_S, S, K)$ is inferred by:

1. modify S by removing the arcs into X_i ,
2. remove the parameters θ_S that correspond to the removed arcs in S ,
3. set $P(X_i = k \mid K) = 1$,
4. use this mutilated Bayesian network to infer the probability of the state of the variables in $C_h - \{X_i\}$



— Assumption 4 —

There are no missing data or hidden variables.



— Assumption 4 —

There are no missing data or hidden variables.

$$\begin{aligned} P(D \mid S, \theta_S, K) &= \prod_{h=1}^m P(C_h \mid S, \theta_S, K) \\ &= \prod_{h=1}^m \prod_{i=1}^n P(x_i^h \mid pa_i^h, \theta_S, K) \end{aligned}$$

since $C_h = (x_1^h, \dots, x_n^h)$ where x_i^h is the realisation of node X_i in case h and pa_i^h denotes the state of the parents of X_i in case h .



— Assumption 5 and 6 —

Variables are discrete.

Parameter independence:

Global: For each causal Bayesian network structure, the parameters (probabilities) associated with one node are prob. independent of the parameters associated with other nodes.

Local: The parameters associated within a node given one instance of its parents are independent of the parameters of that node given other instances of its parent nodes.



— Assumption 5 and 6 —

Variables are discrete.

Parameter independence:

Global: For each causal Bayesian network structure, the parameters (probabilities) associated with one node are prob. independent of the parameters associated with other nodes.

Local: The parameters associated within a node given one instance of its parents are independent of the parameters of that node given other instances of its parent nodes.

Conditional distributions can be written in tabular form.

$$\theta_S = \prod_{i=1}^n \theta_i = \prod_{i=1}^n \prod_{j=1}^{q_i} \theta_{ij}$$



— Assumption 5 and 6 —

$$\begin{aligned} P(D \mid S, \theta_S, K) &= \prod_{h=1}^m \prod_{i=1}^n P(x_i^h \mid pa_i^h, \theta_S, K) \\ &= \prod_{i=1}^n \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} P(x_i = k \mid pa_i = j, \theta_S, K)^{N_{ijk}} \\ &=: \prod_{i=1}^n \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} \theta_{ijk}^{N_{ijk}} \end{aligned}$$

where

r_i = number of states of X_i ,

q_i = number of joint states of parents of X_i , and

N_{ijk} = number of cases in D in which node X_i is passively observed to have state k when its parents have states as given by j .



— Assumption 7 and 8 —

Parameter modularity: If a node has the same parents in two distinct networks, then the distribution of the parameters associated with this node are identical in both networks.

The prior distribution of parameters associated with each node is Dirichlet.



— Assumption 7 and 8 —

Parameter modularity: If a node has the same parents in two distinct networks, then the distribution of the parameters associated with this node are identical in both networks.

The prior distribution of parameters associated with each node is Dirichlet.

The parameter $\theta_{ij} = (\theta_{ijk})_{k=1}^{r_i}$ has a Dirichlet distribution with parameters $\alpha = (\alpha_{ijk})$ if

$$\begin{aligned} P(\theta_{ij1}, \dots, \theta_{ijr_i} \mid S, K) &= \text{Dir}(\theta_{ij1}, \dots, \theta_{ijr_i} \mid \alpha) \\ &= \frac{\Gamma(\sum_{k=1}^{r_i} \alpha_{ijk})}{\prod_{k=1}^{r_i} \Gamma(\alpha_{ijk})} \prod_{k=1}^{r_i} \theta_{ijk}^{\alpha_{ijk}-1} \\ &= \frac{(\alpha_{ij+} - 1)!}{\prod_{k=1}^{r_i} (\alpha_{ijk} - 1)!} \theta_{ij1}^{\alpha_{ij1}-1} \dots \theta_{ijr_i}^{\alpha_{ijr_i}-1}. \end{aligned}$$



— Putting all assumptions together ... —

$$\begin{aligned}
 P(D \mid S, K) &= \int P(D \mid S, \theta_S, K) P(\theta_S \mid S, K) d\theta_S \\
 &= \int \prod_{i=1}^n \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} \theta_{ijk}^{N_{ijk}} \cdot \frac{\Gamma(\alpha_{ij+})}{\prod_{k=1}^{r_i} \Gamma(\alpha_{ijk})} \prod_{k=1}^{r_i} \theta_{ijk}^{\alpha_{ijk}-1} d\theta_S \\
 &= \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij+})}{\prod_{k=1}^{r_i} \Gamma(\alpha_{ijk})} \int \prod_{k=1}^{r_i} \theta_{ijk}^{N_{ijk} + \alpha_{ijk} - 1} d\theta_S \\
 &= \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij+})}{\prod_{k=1}^{r_i} \Gamma(\alpha_{ijk})} \cdot \frac{\prod_{k=1}^{r_i} \Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ij+} + N_{ij+})} \\
 &= \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij+})}{\Gamma(\alpha_{ij+} + N_{ij+})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})}
 \end{aligned}$$

