

Learning networks from knock-outs and RNAi



Florian Markowetz

Max-Planck-Institute for Molecular Genetics
– Computational Molecular Biology –
Berlin, Germany

<http://compdiag.molgen.mpg.de/>

Group meeting
Monday, August 04

— Roadmap —

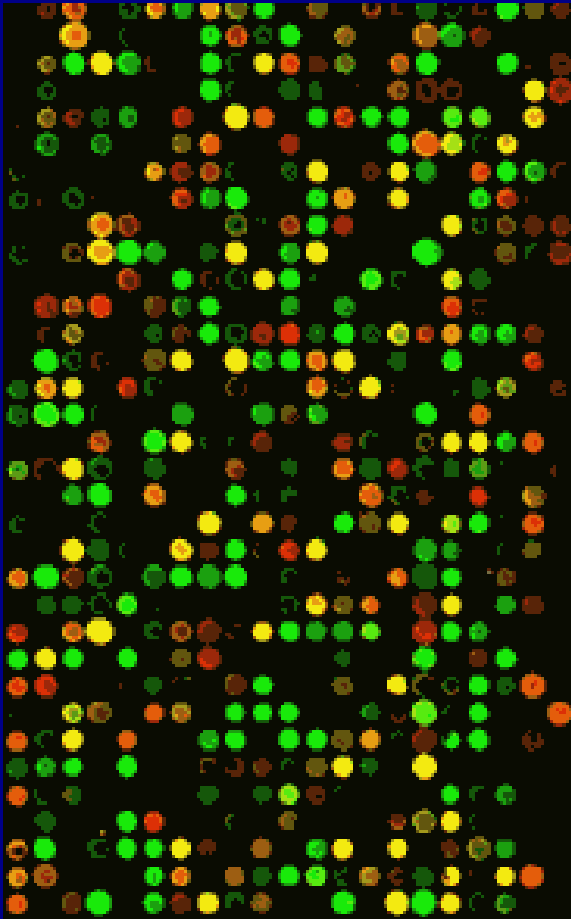
1. Bayesics

2. Modelling knock-outs and RNAi

3. Messy maths: derivation the the bayesian score



— 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, but fails to detect interactions between genes different from linear correlation.
- **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 networks —

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

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

- directed acyclic graph (DAG),
- nodes \leftrightarrow variables = genes,
- lack of arc \leftrightarrow conditional independence

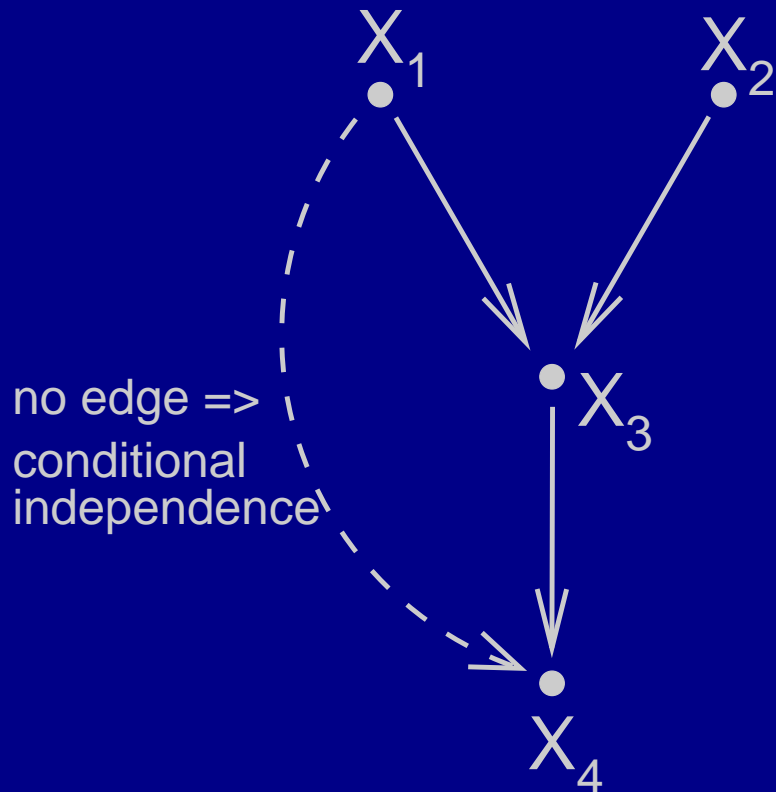
2. a set of **probability distributions** \mathcal{P}

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

$$\mathcal{P} = \{ P(X_i \mid pa_i) \}$$



— Bayesian Networks II —



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

$$P(\mathbf{X}) = \prod_{i=1}^n P(X_i \mid pa_i)$$



— Nature hides more than it shows —

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

Our goal: reconstruct the 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 Bayesian Network. But nature hides this network from us.

Our goal: reconstruct the genetic network from microarray data.

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

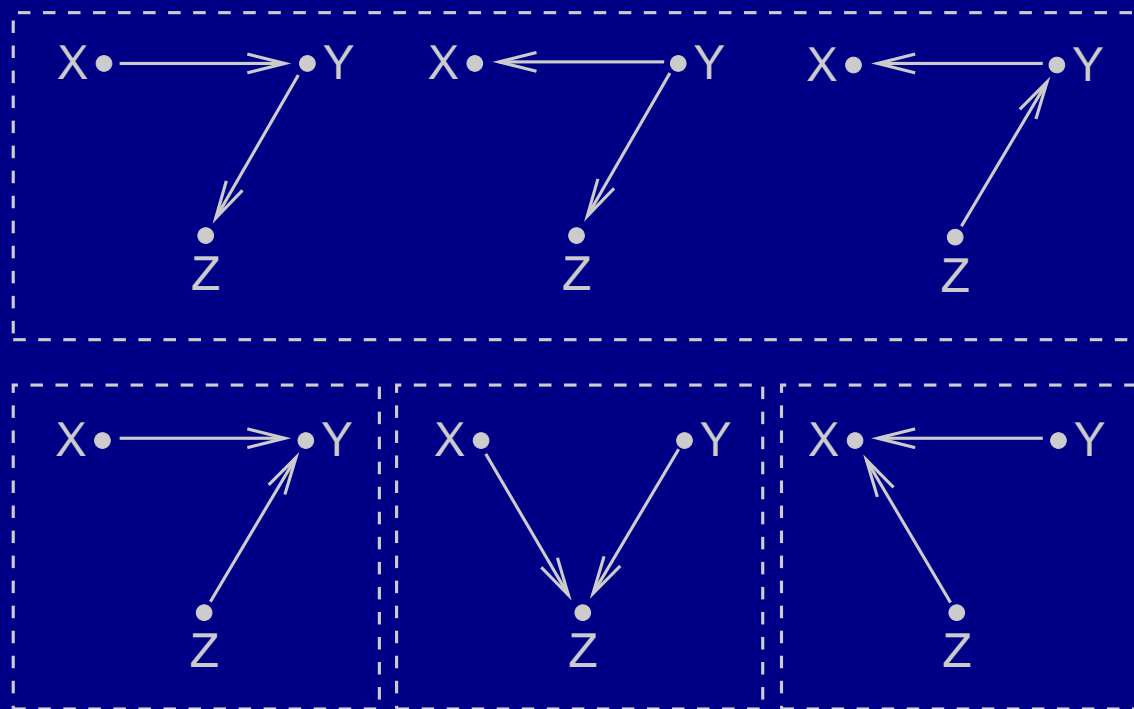
How far does this help us to resolve network structure?
Are there natural limits?

Yes, there are: some structures are statistically indistinguishable.



— Equivalence of bayesian networks —

Markov equivalence $\mathcal{G}_1 \stackrel{M}{\sim} \mathcal{G}_2$ if both structures represent the same set of independence assertions, i. e. if they are compatible with the same distribution P .



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



— Learning from observations —

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.

$$P(\text{dag} \mid \text{data}) \propto P(\text{data} \mid \text{dag}) P(\text{dag})$$

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:

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

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

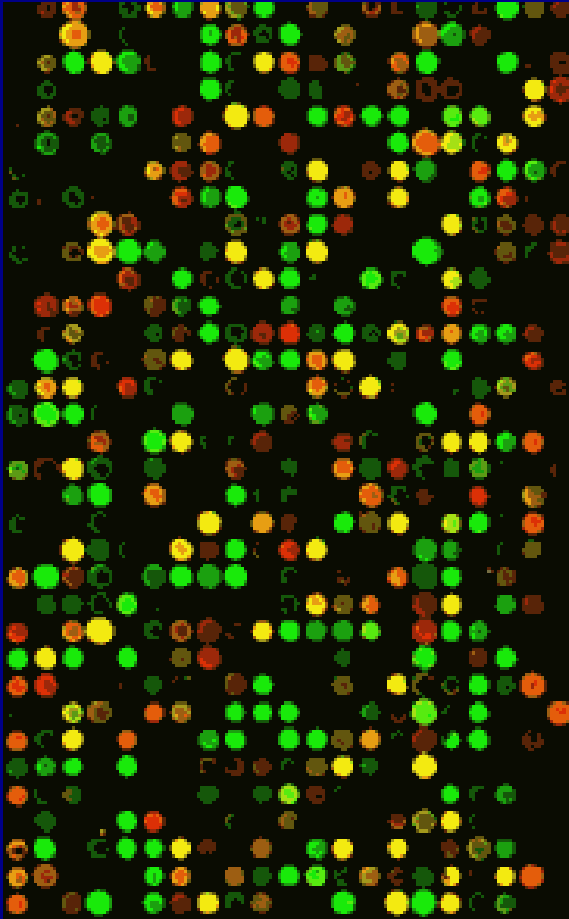
$$P(\text{data} \mid \text{dag}) = \int P(\text{data} \mid \text{dag}, \text{para}) P(\text{para} \mid \text{dag}) d\text{para}$$

Under some conditions the score is **decomposable**:

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



— 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?**

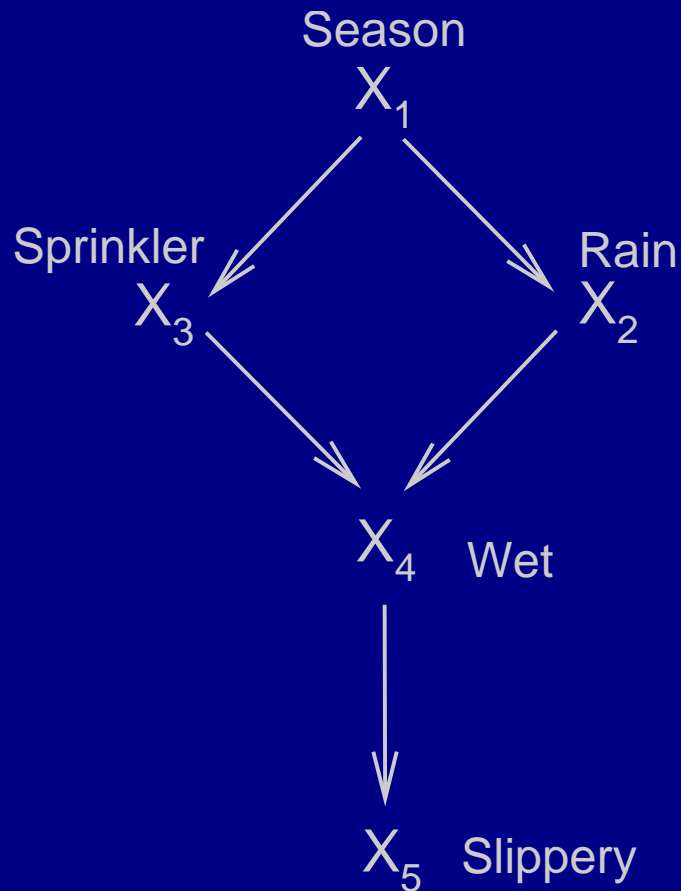


— Roadmap —

1. Bayesics
- 2. Modelling knock-outs and RNAi**
3. Messy maths: derivation the the bayesian score



— An example: the sprinkler network —



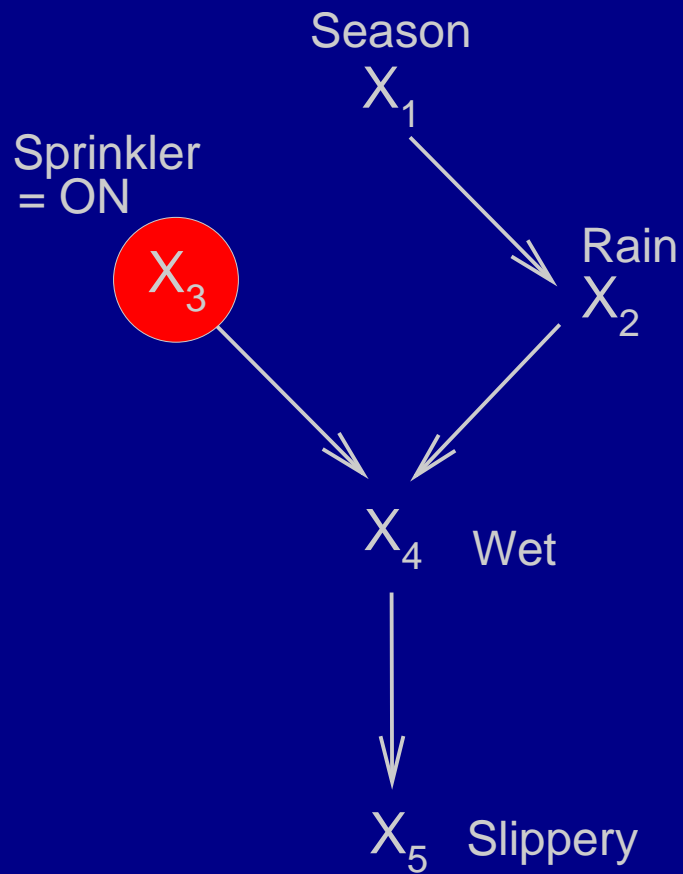
Imagine this being the true causal model we want to induce.

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.

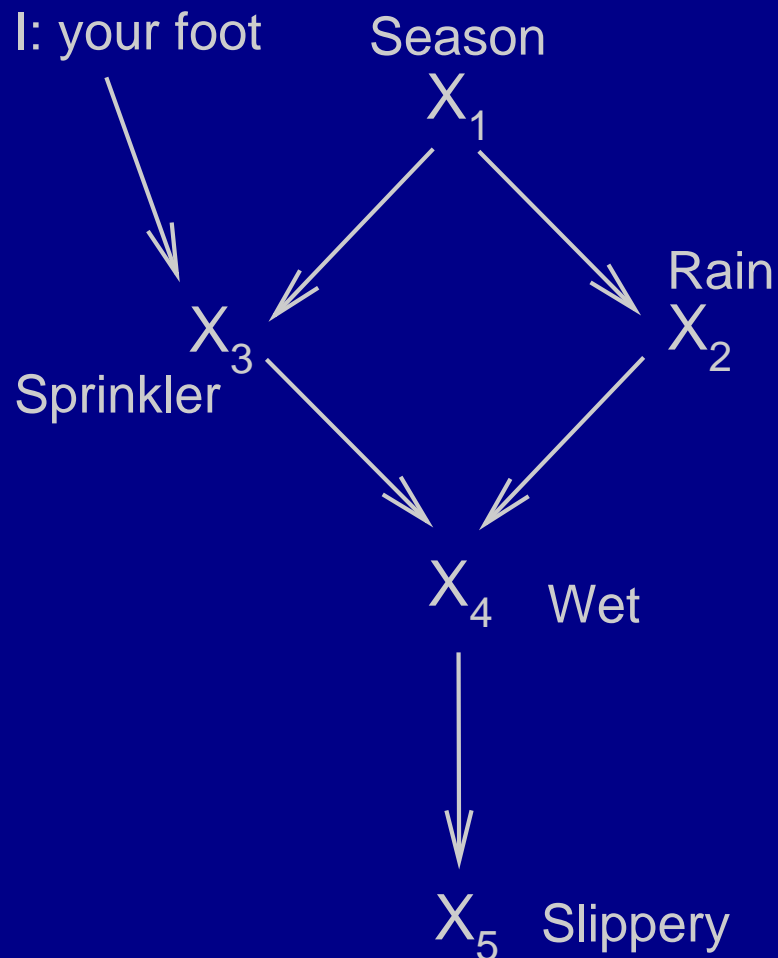
Intervention at X_3 :

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

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 set $P(X_3 \mid pa_3 \cup \{I\}) = ???$.



— 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.
[biologically reasonable?]

Since the Bayesian score decomposes in a sum of family scores we can implement interventions by exchanging one of the summands.



— Roadmap —

1. Bayesics
2. Modelling knock-outs and RNAi
- 3. Messy maths: derivation the the bayesian score**



— 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 3' —

For each experimentally manipulated variable X_i in case C_h , the probability $P(C_h \mid \theta_S, S, K)$ is modeled by ...



— 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 —

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

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 $\theta_S = (P(X_i = k \mid pa_i = j, K))_{i=1,..,n; j=1,..,q_i; k=1,..,r_i} := (\theta_{ijk})$ has a Dirichlet distribution with parameters $\alpha = (\alpha_{ijk})$ if

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



— Putting all assumptions together ... —

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



— So, what? —

Nice theory!

But does it work?

We will see: ask me in a few weeks.



— So, what? —

Nice theory!

But does it work?

We will see: ask me in a few weeks.

Next steps: hack BNT to include the “foot on the pipe” family score and validate its performance using the network simulation by Zak et al.

