Mixture tree model for network inference

JOBIM

Raphaëlle Momal

UMR518 AgroParis Tech/INRA

July 3, 2018

Context

Rising interest in jointly analysed species abundances:

- Metagenomics
- Microbiology
- Ecology

Ecological network

Tool to better understand species interactions (direct/indirect, nature), eco-systems organizations (clusters ?)

Allows for resilience analyses, pathogens control, ecosystem comparison, reaction prediction...

Data

- Species: bacteria, fungi...
- Abundances: read counts from Next-Generation Sequencing technologies (metabarcoding)
- Covariates: coverage, temperature, water depth...

Repeated signal : n samples of p abundances.

Notations

$Y = [Y_{ij}]_{(i,j)\in\{1,\dots,n\}\times\{1,\dots,p\}}$

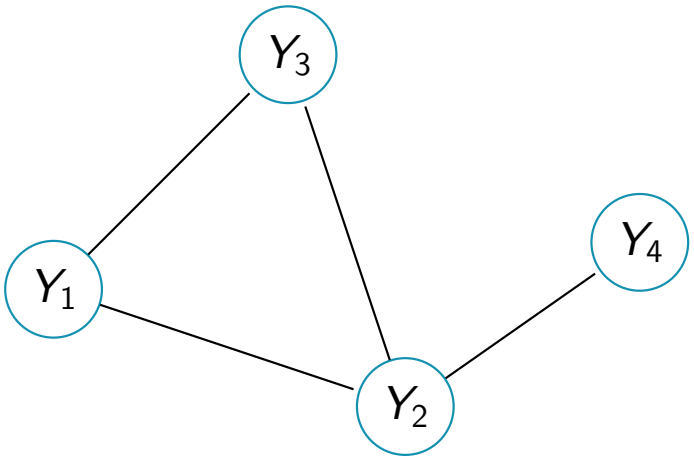
- Y_{ij} : abundance of the j^{th} species in the i^{th} sample

Infer the species interaction network from Y

Challenges

- Statistical network inference
- Count data
- Offsets and covariates

Graphical models



- All variables are dependant
- Some are conditionnally independent (i.e. indirectly dependeant)

  Y_4 is independent from (Y_1, Y_3) conditionally to Y_2

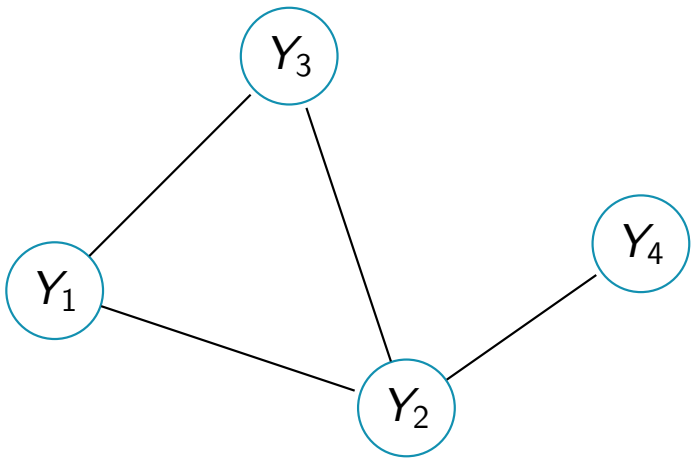
Graphical models

Definition [?]

The joint distribution P is faithful to the graph G iff

$$P(Y_1, \ldots, Y_p) \propto \prod_{C \in \mathcal{C}_G} \psi_C(Y_C)$$

where $\mathcal{C}_G$ = set of cliques of G.



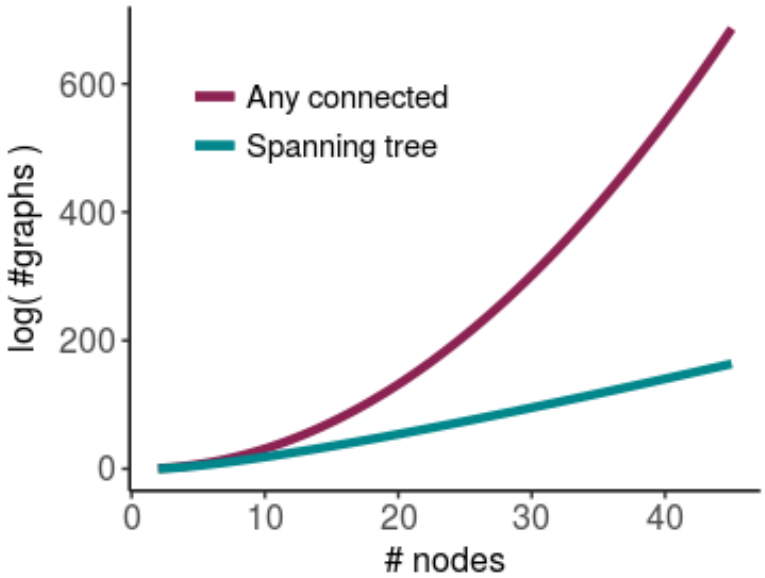
$$P(Y_1, Y_2, Y_3, Y_4) \propto \psi_1(Y_1, Y_2, Y_3) \times \psi_2(Y_3, Y_4)$$

Spanning trees

Very large space to explore: $\#\mathcal{G}_{\surd} = 2^{\frac{p(p-1)}{2}}$

Spanning trees are a sparse solution :

$$\left.\begin{array}{l} G \text{ is connected} \\ G \text{ has no cycle} \end{array}\right\} \; G \text{ has } (p-1) \text{ edges}$$



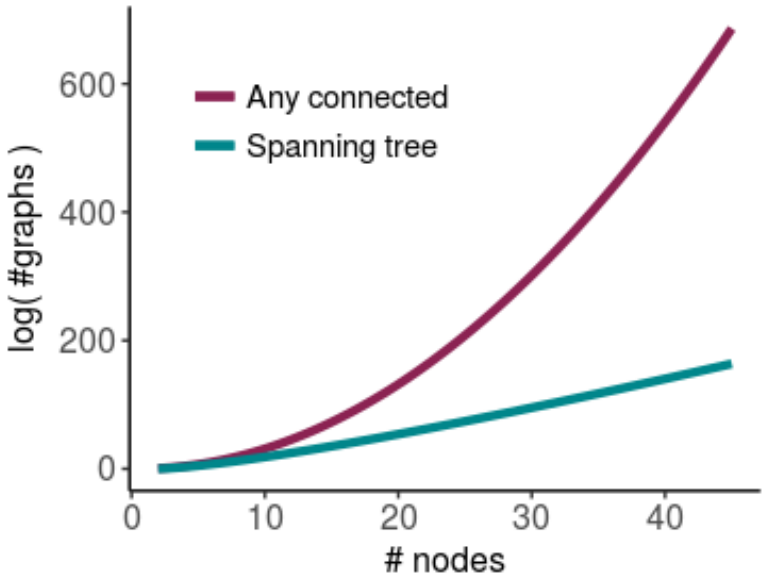
Much smaller space to explore:
$\#\mathcal{T}_{\surd} = p^{(p-2)}$

Spanning trees

Very large space to explore: $\#\mathcal{G}_{\surd} = 2^{\frac{p(p-1)}{2}}$

Spanning trees are a sparse solution :

$$\left.\begin{array}{l} G \text{ is connected} \\ G \text{ has no cycle} \end{array}\right\} \; G \text{ has } (p-1) \text{ edges}$$



Much smaller space to explore:
$\#\mathcal{T}_{\surd} = p^{(p-2)}$

Still a huge complexity...

Algebra tools

Maximum spanning tree Kruskall's algorithm

$$\hat{T} = \underset{T}{\operatorname{argmax}} \left\{ \prod_k q_k \prod_{(k,l)(Y)\in T} q_{k,l}(Y) \right\} \to \Theta((\#\text{ nodes})^2)$$

Tree averaging Matrix tree theoem [**?**]

$$\mathbb{P}(Y) = \sum_T \mathbb{P}(T)\mathbb{P}(Y|T) \to \Theta((\#\text{ nodes})^3)$$

Algebra tools

Maximum spanning tree Kruskall's algorithm

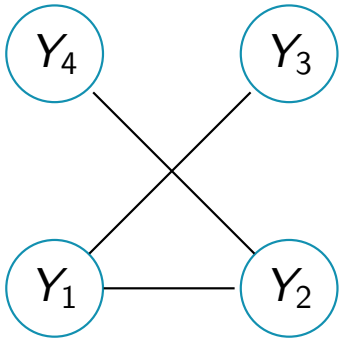
$$\hat{T} = \underset{T}{\operatorname{argmax}} \left\{ \prod_k q_k \prod_{(k,l)(Y)\in T} q_{k,l}(Y) \right\} \to \Theta((\#\text{ nodes})^2)$$

Tree averaging Matrix tree theoem [**?**]

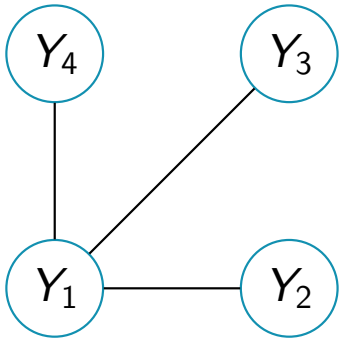
$$\mathbb{P}(Y) = \sum_T \mathbb{P}(T)\mathbb{P}(Y|T) \to \Theta((\#\text{ nodes})^3)$$

Idea: infer the network by averaging spanning trees

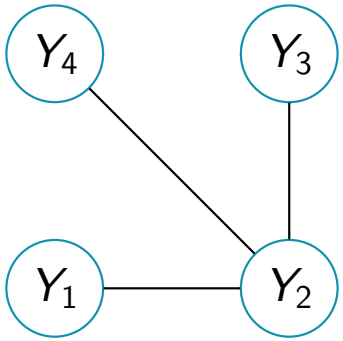
Tree averaging



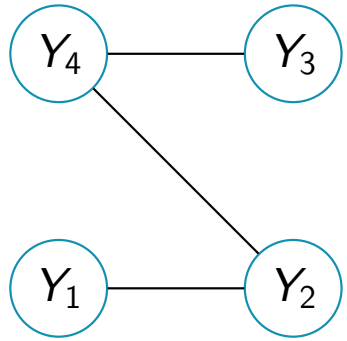
$P\{T = T_1|Y\}$



$P\{T = T_2|Y\}$



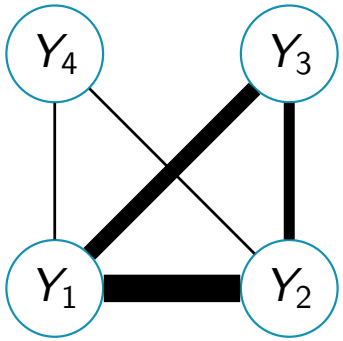
$P\{T = T_3|Y\}$



$P\{T = T_4|Y\}$

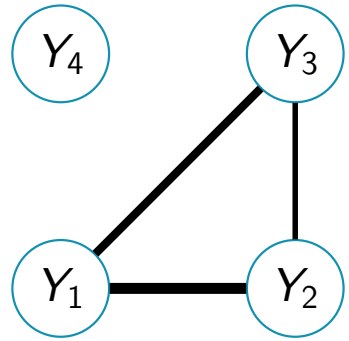
...

Compute edge probabilities:



$P\{(j,k) \in T|Y\}$

Thresholding probabilities:



$P\{(j,k) \in T|Y\}$

PLN model

Poisson log-Normal distribution [?]

$$\left.\begin{array}{ll} Z_i \ iid & \sim \mathcal{N}_d(0, \Sigma) \\ & (Y_{ij})_j \perp\!\!\!\perp | Z_i \\ Y_{ij}|Z_{ij} & \sim \mathcal{P}(e^{Z_{ij}}) \end{array}\right\} Y \sim \mathcal{PLN}(0, \Sigma)$$

- Gaussian latent layer
- Easy handling of multi-variate data (contrary to Negative binomial distribution)

PLN model

Poisson log-Normal distribution [?]

$$\left.\begin{array}{ll} Z_i \ iid & \sim \mathcal{N}_d(0, \Sigma) \\ & (Y_{ij})_j \perp\!\!\!\perp | Z_i \\ Y_{ij}|Z_{ij} & \sim \mathcal{P}(e^{o_{ij} + x_i^\top \Theta_j + Z_{ij}}) \end{array}\right\} Y \sim \mathcal{PLN}(O + x^\top \Theta, \Sigma)$$

- Gaussian latent layer
- Easy handling of multi-variate data (contrary to Negative binomial distribution)
- Allow adjustment for covariates and offsets

PLN model

Poisson log-Normal distribution [?]

$$\left.\begin{array}{ll} Z_i \ iid & \sim \mathcal{N}_d(0, \Sigma) \\ & (Y_{ij})_j \perp\!\!\!\perp | Z_i \\ Y_{ij}|Z_{ij} & \sim \mathcal{P}(e^{o_{ij} + x_i^\top \Theta_j + Z_{ij}}) \end{array}\right\} Y \sim \mathcal{PLN}(O + x^\top \Theta, \Sigma)$$

- Gaussian latent layer
- Easy handling of multi-variate data (contrary to Negative binomial distribution)
- Allow adjustment for covariates and offsets

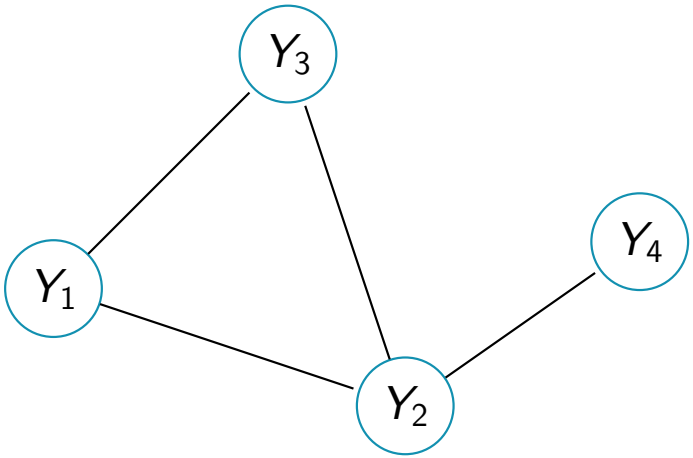
Idea: Infer the latent Gaussian network with an EM algorithm.

Gaussian Graphical Models (GGM) & SpiecEasi

Gaussian distribution.

$$Y_r \sim \mathcal{N}_p(\mu, \Sigma),\ \mu = \text{vector of means},\ \Sigma = \text{covariance matrix}.$$

A nice property.



Inverse covariance matrix

$$\Sigma^{-1} = \Omega \propto \begin{bmatrix} 1 & .5 & .5 & 0 \\ .5 & 1 & .5 & .5 \\ .5 & .5 & 1 & 0 \\ 0 & .5 & 0 & 1 \end{bmatrix}$$

Glasso.

On gaussian data : $\widehat{\Omega}_\lambda = \arg\min_{\Omega \in \mathcal{S}_d^+} \left\{ L(Y, \Omega) + \lambda \sum_{i \neq j} |\omega_{ij}| \right\}$

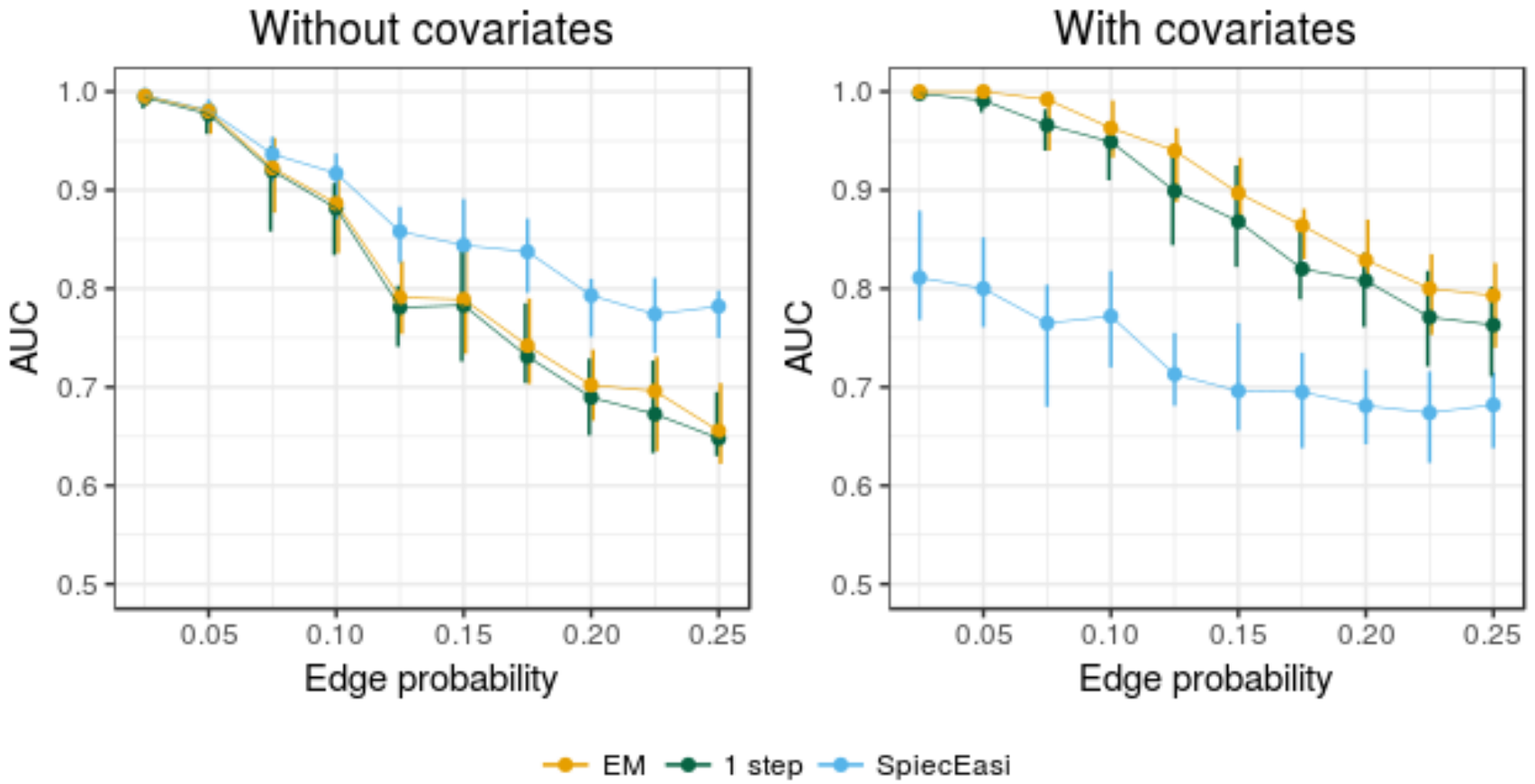
$\Rightarrow$ SpiecEasi method [?]: glasso on transformed counts

Simulation design

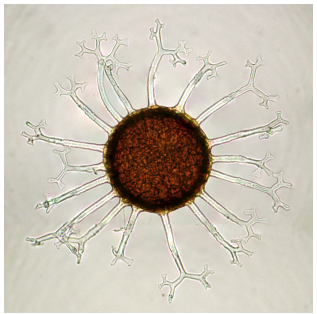
For each parameter settings, repeat 40 times:

1. Draw G
2. Derive Ω from the adjacency matrix
3. Generate count data Y under PLN model with parameter Ω and possible covariates
4. Infer the network with PLN + mixture tree EM and SpiecEasi
5. Compare results with AUC (presence/absence of edges)

Results: Erdös, 20 nodes



Oak Mildew



Erysiphe alphitoides.

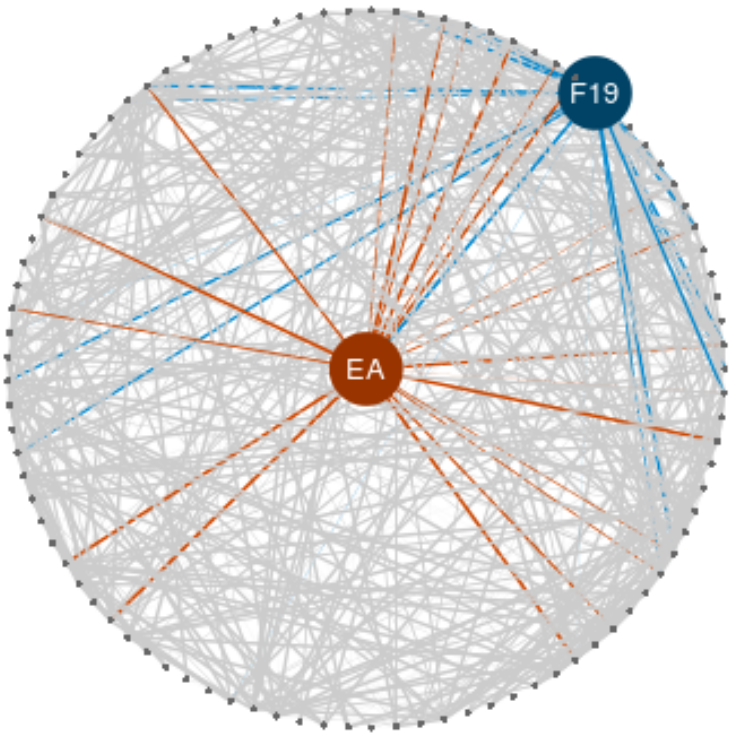
Oak leaf with powdery mildew.

Data: metabarcoding of oak tree leaves microbiome.

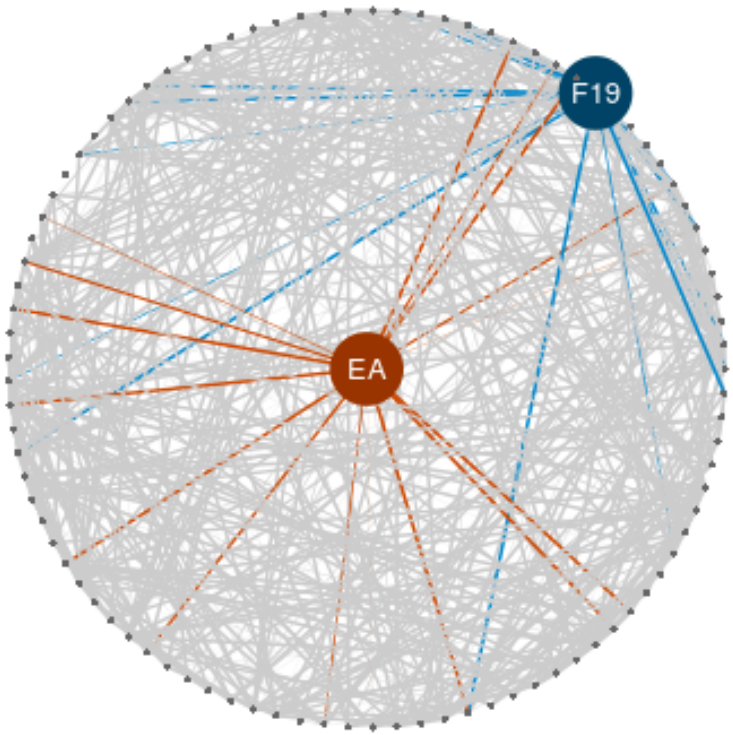
- 114 sample of 94 microbial species counts (bacteria/fungi)
- Different read depth for bacteria and fungi
- 3 distances covariates (quantitative)

Inferred networks

Offset



Adding distance covariables



Including distance covariates

Regression coefficients

	Offset	Adding distances			
	Int.	Int.	to base	to trunk	to ground
EA	-4.39	0.710	-0.0200	0.0215	-0.0251
F19	-4.37	-8.52	0.0219	-0.0172	0.0143

Degree estimation

	Offset	Distances
EA	2.20	1.86
F19	3.03	2.80

Perspectives

We provide:

- Formal probabilistic model for network inference with count data
- Estimation algorithm
- Allows to account for offsets and covariates

What is next:

- Method for determining the threshold
- Network comparison
- Inference in the observed counts layer
- Missing major entity (species/covariable)

Thank you

Thank you for your attention.