

# Essays on Bioinformatics and Social Network Analysis

Statistical and Computational Methods for Complex Systems

George G Vega Yon

University of Southern California, Department of Preventive Medicine

November 18, 2019

## Statistical and computational methods for bioinformatics and social network analysis

- We live in a non-*IID* world.

## Statistical and computational methods for bioinformatics and social network analysis

- ▶ We live in a non-*IID* world.
- ▶ In some times, the cannot understand a process unless we look at it as a whole.

## Statistical and computational methods for bioinformatics and social network analysis

- ▶ We live in a non-*IID* world.
- ▶ In some times, the cannot understand a process unless we look at it as a whole.
- ▶ There's a reason why we usually assume *IID*.

## Statistical and computational methods for bioinformatics and social network analysis

- ▶ We live in a non-*IID* world.
- ▶ In some times, the cannot understand a process unless we look at it as a whole.
- ▶ There's a reason why we usually assume *IID*.
- ▶ *Modern* (as of today) computational tools help us coping with that.

Paper 1: On the prediction of gene functions using phylogenetic trees

Paper 2: Exponential Random Graph Models for Small Networks

Future Research

## On the prediction of gene functions using phylogenetic trees

*Joint with:* Paul D Thomas, Paul Marjoram, Huaiyu Mi, Duncan Thomas, and John Morrison

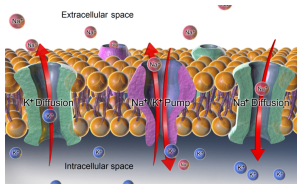
Encode the synthesis of genetic products that ultimately are related to a particular aspect of life, for example



Encode the synthesis of genetic products that ultimately are related to a particular aspect of life, for example

### Molecular function

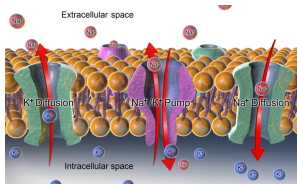
Active transport GO:0005215



Encode the synthesis of genetic products that ultimately are related to a particular aspect of life, for example

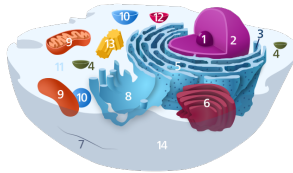
### Molecular function

Active transport GO:0005215



### Cellular component

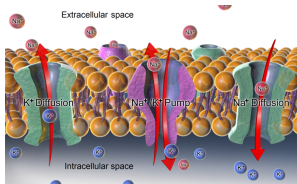
Mitochondria GO:0004016



Encode the synthesis of genetic products that ultimately are related to a particular aspect of life, for example

### Molecular function

Active transport GO:0005215



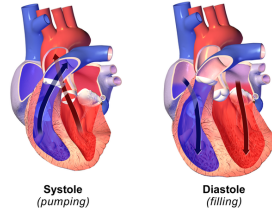
### Cellular component

Mitochondria GO:0004016



### Biological process

Heart contraction GO:0060047





- ▶ The GO project has  $\sim 44,700$  validated terms [▶ more](#),  $\sim 7.3\text{M}$  annotations on  $\sim 4,500$  species.

source: Statistics from [pantherdb.org](http://pantherdb.org) and [geneontology.org](http://geneontology.org)



- ▶ The GO project has  $\sim 44,700$  validated terms [▶ more](#),  $\sim 7.3\text{M}$  annotations on  $\sim 4,500$  species.
- ▶ About  $\sim 500,000$  are on human genes.

source: Statistics from [pantherdb.org](http://pantherdb.org) and [geneontology.org](http://geneontology.org)



- ▶ The GO project has  $\sim 44,700$  validated terms [▶ more](#),  $\sim 7.3\text{M}$  annotations on  $\sim 4,500$  species.
- ▶ About  $\sim 500,000$  are on human genes.
- ▶ Roughly half of human genes ( $\sim 10,000 / 20,000$ ) have some form of annotation.

source: Statistics from [pantherdb.org](http://pantherdb.org) and [geneontology.org](http://geneontology.org)



- ▶ The GO project has  $\sim 44,700$  validated terms [▶ more](#),  $\sim 7.3\text{M}$  annotations on  $\sim 4,500$  species.
- ▶ About  $\sim 500,000$  are on human genes.
- ▶ Roughly half of human genes ( $\sim 10,000 / 20,000$ ) have some form of annotation.
- ▶ We know something of less than 10% of known genes (near 1.7M).

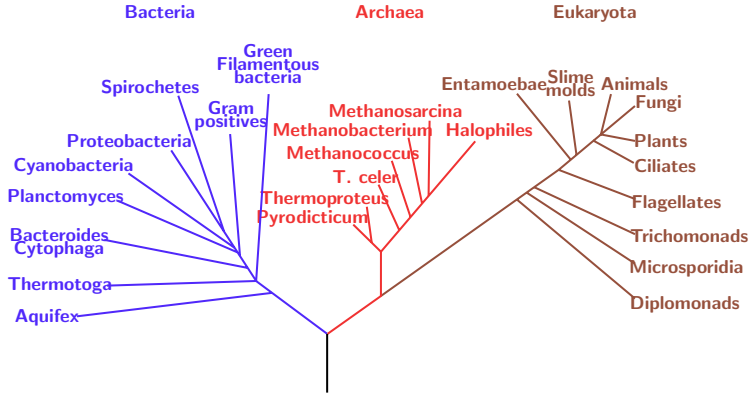
source: Statistics from [pantherdb.org](http://pantherdb.org) and [geneontology.org](http://geneontology.org)



- ▶ The GO project has  $\sim 44,700$  validated terms [▶ more](#),  $\sim 7.3\text{M}$  annotations on  $\sim 4,500$  species.
- ▶ About  $\sim 500,000$  are on human genes.
- ▶ Roughly half of human genes ( $\sim 10,000 / 20,000$ ) have some form of annotation.
- ▶ We know something of less than 10% of known genes (near 1.7M).
- ▶ An important effort of the GO has to do with phylogenetics...

source: Statistics from [pantherdb.org](http://pantherdb.org) and [geneontology.org](http://geneontology.org)





**Figure 1** A phylogenetic tree of living things, based on RNA data and proposed by Carl Woese, showing the separation of bacteria, archaea, and eukaryotes (wiki)

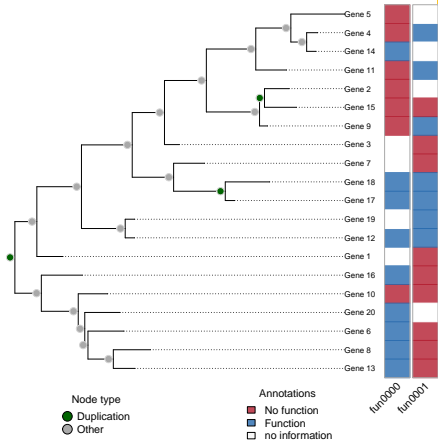


- ▶ The PANTHER project (part of GO)  
provides information about evolutionary  
structure of 1.7 million genes

- ▶ The PANTHER project (part of GO) provides information about evolutionary structure of 1.7 million genes
- ▶ These genes are grouped in 15,524 phylogenetic trees (families)

- ▶ The PANTHER project (part of GO) provides information about evolutionary structure of 1.7 million genes
- ▶ These genes are grouped in 15,524 phylogenetic trees (families)
- ▶ A single family can host multiple species

- ▶ The PANTHER project (part of GO) provides information about evolutionary structure of 1.7 million genes
- ▶ These genes are grouped in 15,524 phylogenetic trees (families)
- ▶ A single family can host multiple species



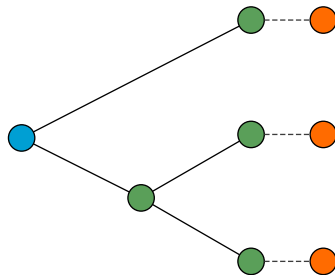
**Figure 2** Simulated phylogenetic tree and gene annotations.

We can use

evolutionary trees

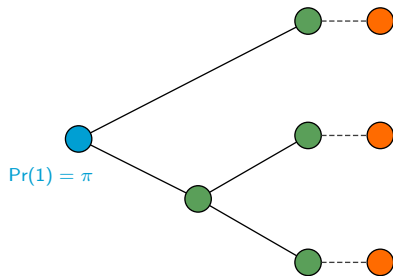
to inform a model for predicting

genetic annotations!





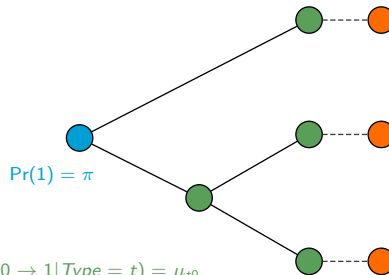
- Initial (spontaneous) gain of function.



► other models

► other view

- ▶ Initial (spontaneous) gain of function.
- ▶ Loss/gain of offspring depends on: (a) the state of their parents (**Markov process**), and (b) the type of node [▶ more](#)

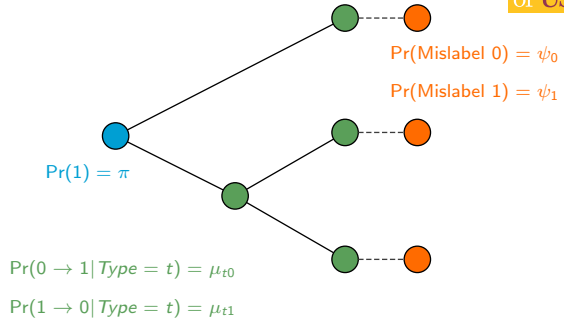


$$\Pr(0 \rightarrow 1 | Type = t) = \mu_{t0}$$

$$\Pr(1 \rightarrow 0 | Type = t) = \mu_{t1}$$

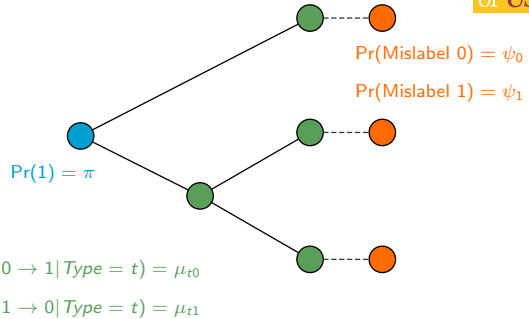
# An evolutionary model of gene functions

- ▶ Initial (spontaneous) gain of function.
- ▶ Loss/gain of offspring depends on: (a) the state of their parents (**Markov process**), and (b) the type of node [▶ more](#)
- ▶ We control for human error.



# An evolutionary model of gene functions

- ▶ Initial (spontaneous) gain of function.
- ▶ Loss/gain of offspring depends on: (a) the state of their parents (**Markov process**), and (b) the type of node [▶ more](#)
- ▶ We control for human error.



We implemented the model using Felsenstein's' pruning algorithm (linear complexity) in the R package `aphylo` [▶ more](#).

# Prediction with real data

	Prior	
	Uniform	Beta
Mislab. prob.		
$\psi_0$	0.23	0.25
$\psi_1$	0.01	0.01
Gain/Loss at dupl.		
$\mu_{d0}$	0.97	0.96
$\mu_{d1}$	0.52	0.58
Gain/Loss at spec.		
$\mu_{s0}$	0.05	0.06
$\mu_{s1}$	0.01	0.02
Root node		
$\pi$	0.81	0.45
Leave-one-out AUC		
Mean	0.69	0.67
Median	0.81	0.75

- 141 pooled functions (trees) with 7,388 genes with 0/1 annotations.

**Table 1** Parameter estimates using different priors.

# Prediction with real data

	Prior	
	Uniform	Beta
Mislab. prob.		
$\psi_0$	0.23	0.25
$\psi_1$	0.01	0.01
Gain/Loss at dupl.		
$\mu_{d0}$	0.97	0.96
$\mu_{d1}$	0.52	0.58
Gain/Loss at spec.		
$\mu_{s0}$	0.05	0.06
$\mu_{s1}$	0.01	0.02
Root node		
$\pi$	0.81	0.45
Leave-one-out AUC		
Mean	0.69	0.67
Median	0.81	0.75

- ▶ 141 pooled functions (trees) with 7,388 genes with 0/1 annotations.
- ▶ Parameter estimates are actually probabilities.

**Table 1** Parameter estimates using different priors.

# Prediction with real data

	Prior	
	Uniform	Beta
Mislab. prob.		
$\psi_0$	0.23	0.25
$\psi_1$	0.01	0.01
Gain/Loss at dupl.		
$\mu_{d0}$	0.97	0.96
$\mu_{d1}$	0.52	0.58
Gain/Loss at spec.		
$\mu_{s0}$	0.05	0.06
$\mu_{s1}$	0.01	0.02
Root node		
$\pi$	0.81	0.45
Leave-one-out AUC		
Mean	0.69	0.67
Median	0.81	0.75

- ▶ 141 pooled functions (trees) with 7,388 genes with 0/1 annotations.
- ▶ Parameter estimates are actually probabilities.
- ▶ Data driven results (uninformative prior).

**Table 1** Parameter estimates using different priors.

# Prediction with real data

	Prior	
	Uniform	Beta
Mislab. prob.		
$\psi_0$	0.23	0.25
$\psi_1$	0.01	0.01
<b>Gain/Loss at dupl.</b>		
$\mu_{d0}$	0.97	0.96
$\mu_{d1}$	0.52	0.58
<b>Gain/Loss at spec.</b>		
$\mu_{s0}$	0.05	0.06
$\mu_{s1}$	0.01	0.02
Root node		
$\pi$	0.81	0.45
Leave-one-out AUC		
Mean	0.69	0.67
Median	0.81	0.75

**Table 1** Parameter estimates using different priors.

- ▶ 141 pooled functions (trees) with 7,388 genes with 0/1 annotations.
- ▶ Parameter estimates are actually probabilities.
- ▶ Data driven results (uninformative prior).
- ▶ **Biologically meaningful results.**



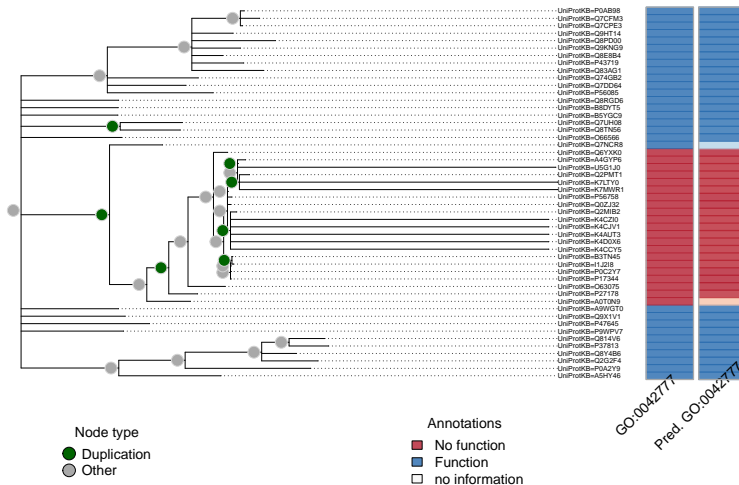
# Prediction with real data

	Prior	
	Uniform	Beta
Mislab. prob.		
$\psi_0$	0.23	0.25
$\psi_1$	0.01	0.01
<b>Gain/Loss at dupl.</b>		
$\mu_{d0}$	0.97	0.96
$\mu_{d1}$	0.52	0.58
<b>Gain/Loss at spec.</b>		
$\mu_{s0}$	0.05	0.06
$\mu_{s1}$	0.01	0.02
Root node		
$\pi$	0.81	0.45
Leave-one-out AUC		
Mean	0.69	0.67
Median	0.81	0.75

**Table 1** Parameter estimates using different priors.

- ▶ 141 pooled functions (trees) with 7,388 genes with 0/1 annotations.
- ▶ Parameter estimates are actually probabilities.
- ▶ Data driven results (uninformative prior).
- ▶ **Biologically meaningful results.**
- ▶ Took about 5 minutes each.

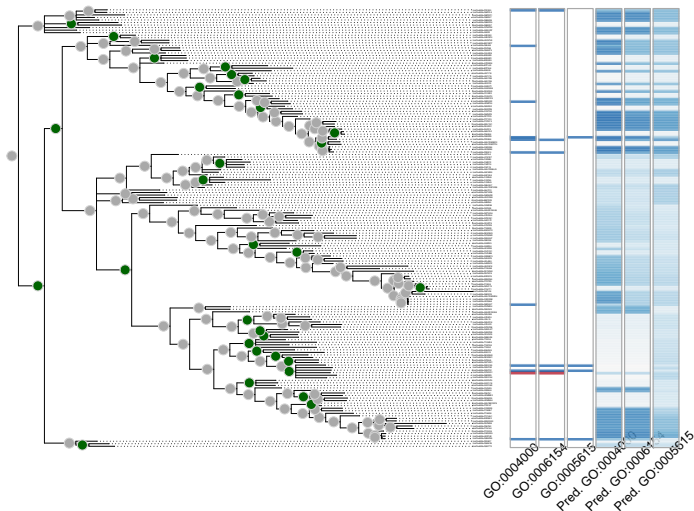
Annotated Phylogenetic Tree



# Prediction with real data: Out-of-sample prediction

Adenosine Deaminase (PTHR11409)

AUCs:={0.80, 0.67, -}



## Key takeaways

- ▶ A parsimonious model for predicting gene functions using phylogenetics.
- ▶ Computationally scalable. SIFTER (our benchmark) would take about 66 years (yes, years) to estimate a model for 100 families of size 300, we take about 5 minutes.
- ▶ Meaningful biological results.
- ▶ Preliminary accuracy results comparable to state-of-the-art phylo-based models.

## Key takeaways

- ▶ A parsimonious model for predicting gene functions using phylogenetics.
- ▶ Computationally scalable. SIFTER (our benchmark) would take about 66 years (yes, years) to estimate a model for 100 families of size 300, we take about 5 minutes.
- ▶ Meaningful biological results.
- ▶ Preliminary accuracy results comparable to state-of-the-art phylo-based models.

## Challenges

Right now the model has two big assumptions

## Key takeaways

- ▶ A parsimonious model for predicting gene functions using phylogenetics.
- ▶ Computationally scalable. SIFTER (our benchmark) would take about 66 years (yes, years) to estimate a model for 100 families of size 300, we take about 5 minutes.
- ▶ Meaningful biological results.
- ▶ Preliminary accuracy results comparable to state-of-the-art phylo-based models.

## Challenges

Right now the model has two big assumptions

- ▶ Offspring are conditional independent on their parent and

## Key takeaways

- ▶ A parsimonious model for predicting gene functions using phylogenetics.
- ▶ Computationally scalable. SIFTER (our benchmark) would take about 66 years (yes, years) to estimate a model for 100 families of size 300, we take about 5 minutes.
- ▶ Meaningful biological results.
- ▶ Preliminary accuracy results comparable to state-of-the-art phylo-based models.

## Challenges

Right now the model has two big assumptions

- ▶ Offspring are conditional independent on their parent and
- ▶ Functions evolve independently. [▶ more](#)

# Exponential Random Graph Models for Small Networks

*Joint with:* Andrew Slaughter and Kayla de la Haye



Exponential Family Random Graph Models, aka **ERGMs** are:

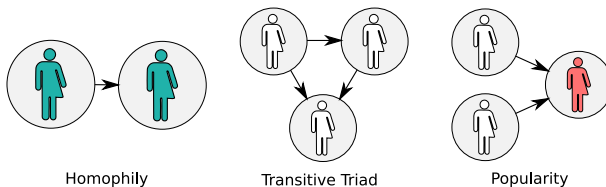
# What are Exponential Random Graph Models

Exponential Family Random Graph Models, aka **ERGMs** are:

- ▶ Statistical models of (social) networks

Exponential Family Random Graph Models, aka **ERGMs** are:

- ▶ Statistical models of (social) networks
- ▶ In simple terms: statistical inference on what network patterns/structures/motifs govern social networks



A vector of  
model parameters

A vector of  
sufficient statistics

$$\Pr(\mathbf{Y} = \mathbf{y} \mid \theta, \mathbf{X}) = \frac{\exp\{\theta^t \mathbf{s}(\mathbf{y}, \mathbf{X})\}}{\sum_{\mathbf{y}' \in \mathcal{Y}} \exp\{\theta^t \mathbf{s}(\mathbf{y}', \mathbf{X})\}}, \quad \forall \mathbf{y} \in \mathcal{Y}$$

Observed data

The normalizing constant

All possible networks

► more on terms

A vector of  
model parameters

A vector of  
sufficient statistics

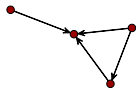
$$\Pr(\mathbf{Y} = \mathbf{y} \mid \theta, \mathbf{X}) = \frac{\exp\{\theta^t \mathbf{s}(\mathbf{y}, \mathbf{X})\}}{\sum_{\mathbf{y}' \in \mathcal{Y}} \exp\{\theta^t \mathbf{s}(\mathbf{y}', \mathbf{X})\}}, \quad \forall \mathbf{y} \in \mathcal{Y}$$

Observed data
The normalizing constant
All possible networks

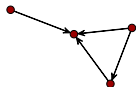
The normalizing constant has  $2^{n(n-1)}$  terms!

► more on terms

In this network

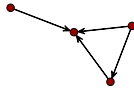


In this network



We see 4 **edges**, 1 **transitive triad** and  
**no mutual ties**.

In this network



We see 4 **edges**, 1 **transitive triad** and  
**no mutual ties**.

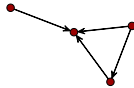
The probability function of this model  
would be

$$\mathbb{P}(\mathbf{G} = \mathbf{g} \mid \theta) = \frac{\exp \{4\theta_{edges} + \theta_{ttriads} + 0\theta_{mutual}\}}{\sum_{\mathbf{g}' \in \mathcal{G}} \exp \{\theta^t s(\mathbf{g}')\}}$$

with  $\theta = [\theta_{edges} \quad \theta_{ttriads} \quad \theta_{mutual}]^t$



In this network



We see 4 **edges**, 1 **transitive triad** and **no mutual ties**.

The probability function of this model would be

$$\mathbb{P}(\mathbf{G} = \mathbf{g} \mid \theta) = \frac{\exp \{4\theta_{edges} + \theta_{ttriads} + 0\theta_{mutual}\}}{\sum_{\mathbf{g}' \in \mathcal{G}} \exp \{\theta^t s(\mathbf{g}')\}}$$

with  $\theta = [\theta_{edges} \quad \theta_{ttriads} \quad \theta_{mutual}]^t$

This model has **MLE parameter estimates** of -0.20 (low density), 0.28 (high chance of ttriads), and -Inf (low chance of mutuality) for the parameters edges, ttriads, and mutual respectively.



Medium-large (dozens to a couple of thousand vertices) networks

- ▶ Markov Chain Monte Carlo (MCMC) based approaches like MC-MLE or Robbins-Monro Stochastic Approximation. [▶ details](#)
- ▶ Maximum Pseudo Likelihood (MPLE)

Medium-large (dozens to a couple of thousand vertices) networks

- ▶ Markov Chain Monte Carlo (MCMC) based approaches like MC-MLE or Robbins-Monro Stochastic Approximation. [▶ details](#)
- ▶ Maximum Pseudo Likelihood (MPLE)

large-huge networks (up to millions of vertices)

- ▶ Parametric bootstrap
- ▶ Conditional joint estimation (like snowball sampling, a.k.a. divide and conquer)
- ▶ Equilibrium Expectation Algorithm (millions of vertices)

Medium-large (dozens to a couple of thousand vertices) networks

- ▶ Markov Chain Monte Carlo (MCMC) based approaches like MC-MLE or Robbins-Monro Stochastic Approximation. [▶ details](#)
- ▶ Maximum Pseudo Likelihood (MPLE)

large-huge networks (up to millions of vertices)

- ▶ Parametric bootstrap
- ▶ Conditional joint estimation (like snowball sampling, a.k.a. divide and conquer)
- ▶ Equilibrium Expectation Algorithm (millions of vertices)

All of these methods are approximations!

# Do we care about small networks?

We see small networks everywhere

# Do we care about small networks?

We see small networks everywhere

- ▶ Families and friends

We see small networks everywhere

- ▶ Families and friends
- ▶ Small teams



We see small networks everywhere

- ▶ Families and friends
- ▶ Small teams
- ▶ Egocentric networks

We see small networks everywhere

- ▶ Families and friends
- ▶ Small teams
- ▶ Egocentric networks
- ▶ Online networks (sometimes)

We see small networks everywhere

- ▶ Families and friends
- ▶ Small teams
- ▶ Egocentric networks
- ▶ Online networks (sometimes)
- ▶ etc.

We see small networks everywhere

- ▶ Families and friends
- ▶ Small teams
- ▶ Egocentric networks
- ▶ Online networks (sometimes)
- ▶ etc.



- In the case of small-enough networks, computation of the likelihood becomes computationally feasible.

- ▶ In the case of small-enough networks, computation of the likelihood becomes computationally feasible.
- ▶ For example, a network with 5 nodes has 1,048,576 unique configurations.

- ▶ In the case of small-enough networks, computation of the likelihood becomes computationally feasible.
- ▶ For example, a network with 5 nodes has 1,048,576 unique configurations.
- ▶ This allow us to directly compute **the normalizing constant**.

$$\Pr(\mathbf{Y} = \mathbf{y} \mid \theta, \mathbf{X}) = \frac{\exp\{\theta^t \mathbf{s}(\mathbf{y}, \mathbf{X})\}}{\sum_{\mathbf{y}' \in \mathcal{Y}} \exp\{\theta^t \mathbf{s}(\mathbf{y}', \mathbf{X})\}}, \quad \forall \mathbf{y} \in \mathcal{Y}$$

A vector of model parameters      A vector of sufficient statistics

The normalizing constant

All possible networks

Observed data

- ▶ In the case of small-enough networks, computation of the likelihood becomes computationally feasible.
- ▶ For example, a network with 5 nodes has 1,048,576 unique configurations.
- ▶ This allow us to directly compute **the normalizing constant**.



$$\Pr(\mathbf{Y} = \mathbf{y} \mid \theta, \mathbf{X}) = \frac{\exp\{\theta^t \mathbf{s}(\mathbf{y}, \mathbf{X})\}}{\sum_{\mathbf{y}' \in \mathcal{Y}} \exp\{\theta^t \mathbf{s}(\mathbf{y}', \mathbf{X})\}}, \quad \forall \mathbf{y} \in \mathcal{Y}$$

A vector of model parameters      A vector of sufficient statistics

The normalizing constant

All possible networks

Observed data

- ▶ In the case of small-enough networks, computation of the likelihood becomes computationally feasible.
- ▶ For example, a network with 5 nodes has 1,048,576 unique configurations.
- ▶ This allow us to directly compute **the normalizing constant**.
- ▶ Using the exact likelihood opens a huge window of methodological-possibilities.

$$\Pr(\mathbf{Y} = \mathbf{y} \mid \theta, \mathbf{X}) = \frac{\exp\{\theta^t \mathbf{s}(\mathbf{y}, \mathbf{X})\}}{\sum_{\mathbf{y}' \in \mathcal{Y}} \exp\{\theta^t \mathbf{s}(\mathbf{y}', \mathbf{X})\}}, \quad \forall \mathbf{y} \in \mathcal{Y}$$

A vector of model parameters      A vector of sufficient statistics

The normalizing constant

All possible networks

Observed data

- ▶ In the case of small-enough networks, computation of the likelihood becomes computationally feasible.
- ▶ For example, a network with 5 nodes has 1,048,576 unique configurations.
- ▶ This allow us to directly compute **the normalizing constant**.
- ▶ Using the exact likelihood opens a huge window of methodological-possibilities.
- ▶ We implemented this and more in the `ergmito` R package [▶ more](#)

Sidetrack...

**ito, ita:** From the latin *-ītus*. suffix in Spanish used to denote small or affection. e.g.:

*¡Qué lindo ese perrito! / What a beautiful little dog!*

*¿Me darías una tacita de azúcar? / Would you give me a small cup of sugar?*

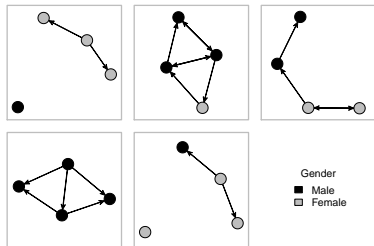
Sidetrack...

**ito, ita:** From the latin *-ītus*. suffix in Spanish used to denote small or affection. e.g.:

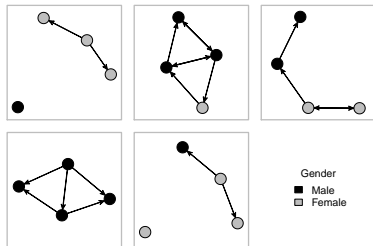
*¡Qué lindo ese perrito! / What a beautiful little dog!*

*¿Me darías una tacita de azúcar? / Would you give me a small cup of sugar?*

Special thanks to George Barnett who proposed the name during the 2018 NASN!



**Figure 3** Random sample of 5 networks simulated using the ergmito package

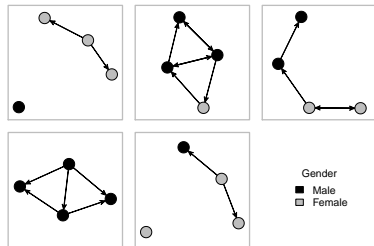


**Figure 3** Random sample of 5 networks simulated using the ergmito package

	Bernoulli	Full model
Edge-count	-0.69* (0.27)	-1.70** (0.54)
Homophily (on Gender)		1.59* (0.64)
AIC	78.38	73.34
BIC	80.48	77.53
Log Likelihood	-38.19	-34.67
Num. networks	5	5

Standard errors in parenthesis. \*\*\*  $p < 0.001$ , \*\*  $p < 0.01$ , \*  $p < 0.05$

**Table 2** Fitted ERGMitos using the fivenets dataset.



**Figure 3** Random sample of 5 networks simulated using the ergmito package

We performed a large simulation study [▶ more](#) comparing MC-MLE (ergm) with MLE (ergmito).

	Bernoulli	Full model
Edge-count	-0.69* (0.27)	-1.70** (0.54)
Homophily (on Gender)		1.59* (0.64)
AIC	78.38	73.34
BIC	80.48	77.53
Log Likelihood	-38.19	-34.67
Num. networks	5	5

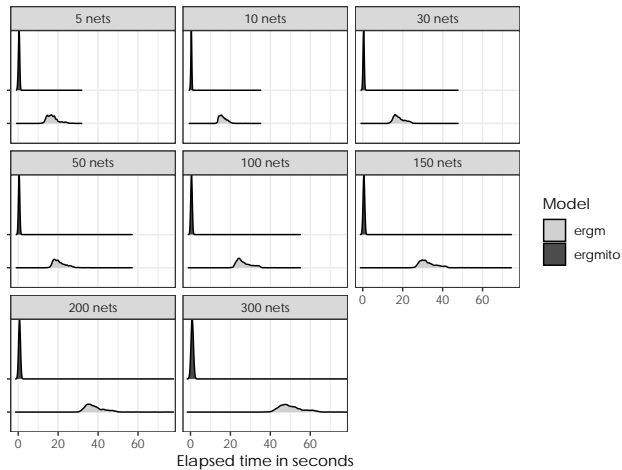
Standard errors in parenthesis. \*\*\*  $p < 0.001$ , \*\*  $p < 0.01$ , \*  $p < 0.05$

**Table 2** Fitted ERGMitos using the fivenets dataset.

Sample size	P(Type I error)		$\chi^2$
	MC-MLE (ergm)	MLE (ergmito)	
5	0.084	0.057	11.71 ***
10	0.070	0.045	12.46 ***
15	0.084	0.066	5.55 *
20	0.074	0.060	3.58
30	0.057	0.052	0.67
50	0.046	0.044	0.17
100	0.048	0.048	0.00

**Table 3** Empirical Type I error rates. The  $\chi^2$  statistic is from a 2-sample test for equality of proportions, and the significance levels are given by \*\*\*  $p < 0.001$ , \*\*  $p < 0.01$ , and \*  $p < 0.05$ .





### Key takeaways

- ▶ New extension of ERGMs using exact statistics for small networks (families, teams, etc.)
- ▶ Performance: Same (un)bias, Lower Type I error rates, (way) faster.
- ▶ Opens the door the new methods, e.g. Mixed effects, LRT, etc.

### Key takeaways

- ▶ New extension of ERGMs using exact statistics for small networks (families, teams, etc.)
- ▶ Performance: Same (un)bias, Lower Type I error rates, (way) faster.
- ▶ Opens the door the new methods, e.g. Mixed effects, LRT, etc.

### Challenges

- ▶ Computationally, we can do better in terms of speed/memory.

## Key takeaways

- ▶ New extension of ERGMs using exact statistics for small networks (families, teams, etc.)
- ▶ Performance: Same (un)bias, Lower Type I error rates, (way) faster.
- ▶ Opens the door the new methods, e.g. Mixed effects, LRT, etc.

## Challenges

- ▶ Computationally, we can do better in terms of speed/memory.
- ▶ Have a good way of assessing goodness-of-fit.

## Key takeaways

- ▶ New extension of ERGMs using exact statistics for small networks (families, teams, etc.)
- ▶ Performance: Same (un)bias, Lower Type I error rates, (way) faster.
- ▶ Opens the door the new methods, e.g. Mixed effects, LRT, etc.

## Challenges

- ▶ Computationally, we can do better in terms of speed/memory.
- ▶ Have a good way of assessing goodness-of-fit.
- ▶ Explore extending this method for (very) large networks.

## Future Research

- Make the model hierarchical when pooling trees

## Future Research: phylogenetic models

- ▶ Make the model hierarchical when pooling trees
  - ▶ Different mutation rates per class of tree/function
  - ▶ Can be complicated to fit (how many classes?)



## Future Research: phylogenetic models

- ▶ Make the model hierarchical when pooling trees
  - ▶ Different mutation rates per class of tree/function
  - ▶ Can be complicated to fit (how many classes?)
- ▶ Use a framework similar to Exponential Random Graph Models:

## Future Research: phylogenetic models

- ▶ Make the model hierarchical when pooling trees
  - ▶ Different mutation rates per class of tree/function
  - ▶ Can be complicated to fit (how many classes?)
- ▶ Use a framework similar to Exponential Random Graph Models:
  - ▶ A generalization of the model
  - ▶ Extends to account for joint dist of functions+siblings
  - ▶ Can incorporate additional information such as branch lengths
  - ▶ Yet computationally more compact compared to SIFTER (iso-statistics)

## Future Research: phylogenetic models

- ▶ Make the model hierarchical when pooling trees
  - ▶ Different mutation rates per class of tree/function
  - ▶ Can be complicated to fit (how many classes?)
- ▶ Use a framework similar to Exponential Random Graph Models:
  - ▶ A generalization of the model
  - ▶ Extends to account for joint dist of functions+siblings
  - ▶ Can incorporate additional information such as branch lengths
  - ▶ Yet computationally more compact compared to SIFTER (iso-statistics)

$$\mathbb{P}(\mathbf{X} = \{x_{n1}, x_{n2}, \dots\} \mid x_{\mathbf{p}(n1, \dots)}) = \frac{\exp\{\mu^T s(\mathbf{x} \mid x_{\mathbf{p}(\cdot)})\}}{\sum_{\mathbf{x}'} \exp\{\mu^T s(\mathbf{x}' \mid x_{\mathbf{p}(\cdot)})\}}$$

## Example 1

### Example 1

- 2 siblings 2 function involves modelling the following array:

$$\begin{bmatrix} x_{p1} \\ x_{p2} \end{bmatrix} \rightarrow \left( \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}, \begin{bmatrix} x_{j1} \\ x_{j2} \end{bmatrix} \right)$$

### Example 1

- ▶ 2 siblings 2 function involves modelling the following array:

$$\begin{bmatrix} x_{p1} \\ x_{p2} \end{bmatrix} \rightarrow \left( \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}, \begin{bmatrix} x_{j1} \\ x_{j2} \end{bmatrix} \right)$$

- ▶ Here we have  $2^2 = 4$  possible states.

## Example 1

- ▶ 2 siblings 2 function involves modelling the following array:

$$\begin{bmatrix} x_{p1} \\ x_{p2} \end{bmatrix} \rightarrow \left( \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}, \begin{bmatrix} x_{j1} \\ x_{j2} \end{bmatrix} \right)$$

- ▶ Here we have  $2^2 = 4$  possible states.

## Example 2

- ▶ If we treat siblings independent, but work with 5 functions, SIFTER needs to estimate  $2^{10} = 1,024$  parameters.

### Example 1

- ▶ 2 siblings 2 function involves modelling the following array:

$$\begin{bmatrix} x_{p1} \\ x_{p2} \end{bmatrix} \rightarrow \left( \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}, \begin{bmatrix} x_{j1} \\ x_{j2} \end{bmatrix} \right)$$

- ▶ Here we have  $2^2 = 4$  possible states.

### Example 2

- ▶ If we treat siblings independent, but work with 5 functions, SIFTER needs to estimate  $2^{10} = 1,024$  parameters.
- ▶ Our approach can reduce this number to, for example, 11 terms:



## Example 1

- ▶ 2 siblings 2 function involves modelling the following array:

$$\begin{bmatrix} x_{p1} \\ x_{p2} \end{bmatrix} \rightarrow \left( \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}, \begin{bmatrix} x_{j1} \\ x_{j2} \end{bmatrix} \right)$$

- ▶ Here we have  $2^2 = 4$  possible states.

## Example 2

- ▶ If we treat siblings independent, but work with 5 functions, SIFTER needs to estimate  $2^{10} = 1,024$  parameters.
- ▶ Our approach can reduce this number to, for example, 11 terms:
  - ▶  $5 \times 4/2 = 10$  statistics for pairwise correlation.

## Example 1

- ▶ 2 siblings 2 function involves modelling the following array:

$$\begin{bmatrix} x_{p1} \\ x_{p2} \end{bmatrix} \rightarrow \left( \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}, \begin{bmatrix} x_{j1} \\ x_{j2} \end{bmatrix} \right)$$

- ▶ Here we have  $2^2 = 4$  possible states.

## Example 2

- ▶ If we treat siblings independent, but work with 5 functions, SIFTER needs to estimate  $2^{10} = 1,024$  parameters.
- ▶ Our approach can reduce this number to, for example, 11 terms:
  - ▶  $5 \times 4/2 = 10$  statistics for pairwise correlation.
  - ▶ One statistic accounting for longest branch.

		Transitions to	
		Case 1	Case 2
Parent	$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix}$
Sufficient statistics			
# Gains		1	2
# only one offspring changes		1	0
# Swaps ( $0 \rightarrow 1, 1 \rightarrow 0$ )		2	4

### Goodness-of-fit

### Goodness-of-fit

- Is something that will need to be addressed at some point.

### Goodness-of-fit

- ▶ Is something that will need to be addressed at some point.
- ▶ The problem is not easy as we need to deal a discrete distribution.

### Goodness-of-fit

- ▶ Is something that will need to be addressed at some point.
- ▶ The problem is not easy as we need to deal a discrete distribution.
- ▶ Two key questions: What sufficient statistic to look at? what test?

### Goodness-of-fit

- ▶ Is something that will need to be addressed at some point.
- ▶ The problem is not easy as we need to deal a discrete distribution.
- ▶ Two key questions: What sufficient statistic to look at? what test?

### ERGMs for large networks



### Goodness-of-fit

- ▶ Is something that will need to be addressed at some point.
- ▶ The problem is not easy as we need to deal a discrete distribution.
- ▶ Two key questions: What sufficient statistic to look at? what test?

### ERGMs for large networks

- ▶ There is still no standard way to estimate ERGMs for large networks.

### Goodness-of-fit

- ▶ Is something that will need to be addressed at some point.
- ▶ The problem is not easy as we need to deal a discrete distribution.
- ▶ Two key questions: What sufficient statistic to look at? what test?

### ERGMs for large networks

- ▶ There is still no standard way to estimate ERGMs for large networks.
- ▶ Most attempts are still depending on simulation methods.

### Goodness-of-fit

- ▶ Is something that will need to be addressed at some point.
- ▶ The problem is not easy as we need to deal a discrete distribution.
- ▶ Two key questions: What sufficient statistic to look at? what test?

### ERGMs for large networks

- ▶ There is still no standard way to estimate ERGMs for large networks.
- ▶ Most attempts are still depending on simulation methods.
- ▶ We could use the Snowball Sampling framework together with ERGMitos.

## Goodness-of-fit

- ▶ Is something that will need to be addressed at some point.
- ▶ The problem is not easy as we need to deal a discrete distribution.
- ▶ Two key questions: What sufficient statistic to look at? what test?

## ERGMs for large networks

- ▶ There is still no standard way to estimate ERGMs for large networks.
- ▶ Most attempts are still depending on simulation methods.
- ▶ We could use the Snowball Sampling framework together with ERGMitos. (... I would call this ERGMote)

- ▶ Paper 1: Phylogenetic models of gene functional evolution

- ▶ Paper 1: Phylogenetic models of gene functional evolution
  - ▶ Parsimonious and biologically meaningful.

- ▶ Paper 1: Phylogenetic models of gene functional evolution
  - ▶ Parsimonious and biologically meaningful.
  - ▶ Computationally scalable.

- ▶ Paper 1: Phylogenetic models of gene functional evolution
  - ▶ Parsimonious and biologically meaningful.
  - ▶ Computationally scalable.
  - ▶ Performance comparable to state-of-the-art alternatives.



- ▶ Paper 1: Phylogenetic models of gene functional evolution
  - ▶ Parsimonious and biologically meaningful.
  - ▶ Computationally scalable.
  - ▶ Performance comparable to state-of-the-art alternatives.
  - ▶ **Next steps:** Use ERGMs framework to break assumptions.

- ▶ Paper 1: Phylogenetic models of gene functional evolution
  - ▶ Parsimonious and biologically meaningful.
  - ▶ Computationally scalable.
  - ▶ Performance comparable to state-of-the-art alternatives.
  - ▶ **Next steps:** Use ERGMs framework to break assumptions.
- ▶ Paper 2: ERGMs for small networks

## Concluding Remarks

- ▶ Paper 1: Phylogenetic models of gene functional evolution
  - ▶ Parsimonious and biologically meaningful.
  - ▶ Computationally scalable.
  - ▶ Performance comparable to state-of-the-art alternatives.
  - ▶ **Next steps:** Use ERGMs framework to break assumptions.
- ▶ Paper 2: ERGMs for small networks
  - ▶ An extension to a well studied models for social networks.

- ▶ Paper 1: Phylogenetic models of gene functional evolution
  - ▶ Parsimonious and biologically meaningful.
  - ▶ Computationally scalable.
  - ▶ Performance comparable to state-of-the-art alternatives.
  - ▶ **Next steps:** Use ERGMs framework to break assumptions.
- ▶ Paper 2: ERGMs for small networks
  - ▶ An extension to a well studied models for social networks.
  - ▶ Implements

- ▶ Paper 1: Phylogenetic models of gene functional evolution
  - ▶ Parsimonious and biologically meaningful.
  - ▶ Computationally scalable.
  - ▶ Performance comparable to state-of-the-art alternatives.
  - ▶ **Next steps:** Use ERGMs framework to break assumptions.
- ▶ Paper 2: ERGMs for small networks
  - ▶ An extension to a well studied models for social networks.
  - ▶ Implements
  - ▶ Opens the door to a large set of methodological innovations.

- ▶ Paper 1: Phylogenetic models of gene functional evolution
  - ▶ Parsimonious and biologically meaningful.
  - ▶ Computationally scalable.
  - ▶ Performance comparable to state-of-the-art alternatives.
  - ▶ **Next steps:** Use ERGMs framework to break assumptions.
- ▶ Paper 2: ERGMs for small networks
  - ▶ An extension to a well studied models for social networks.
  - ▶ Implements
  - ▶ Opens the door to a large set of methodological innovations.
  - ▶ **Next steps:** GOF or extensions to large networks?

## Concluding Remarks

- ▶ Paper 1: Phylogenetic models of gene functional evolution
  - ▶ Parsimonious and biologically meaningful.
  - ▶ Computationally scalable.
  - ▶ Performance comparable to state-of-the-art alternatives.
  - ▶ **Next steps:** Use ERGMs framework to break assumptions.
- ▶ Paper 2: ERGMs for small networks
  - ▶ An extension to a well studied models for social networks.
  - ▶ Implements
  - ▶ Opens the door to a large set of methodological innovations.
  - ▶ **Next steps:** GOF or extensions to large networks?

Accomplishments during the development of this work



- ▶ Paper 1: Phylogenetic models of gene functional evolution
  - ▶ Parsimonious and biologically meaningful.
  - ▶ Computationally scalable.
  - ▶ Performance comparable to state-of-the-art alternatives.
  - ▶ **Next steps:** Use ERGMs framework to break assumptions.
- ▶ Paper 2: ERGMs for small networks
  - ▶ An extension to a well studied models for social networks.
  - ▶ Implements
  - ▶ Opens the door to a large set of methodological innovations.
  - ▶ **Next steps:** GOF or extensions to large networks?

## Accomplishments during the development of this work

- ▶ 6 journal publications (Journal of Open Source Software, Stata Journal, Journal of health and social behavior, Translational behavioral medicine, Social Science & Medicine)



## Concluding Remarks

- ▶ Paper 1: Phylogenetic models of gene functional evolution
  - ▶ Parsimonious and biologically meaningful.
  - ▶ Computationally scalable.
  - ▶ Performance comparable to state-of-the-art alternatives.
  - ▶ **Next steps:** Use ERGMs framework to break assumptions.
- ▶ Paper 2: ERGMs for small networks
  - ▶ An extension to a well studied models for social networks.
  - ▶ Implements
  - ▶ Opens the door to a large set of methodological innovations.
  - ▶ **Next steps:** GOF or extensions to large networks?

### Accomplishments during the development of this work

- ▶ 6 journal publications (Journal of Open Source Software, Stata Journal, Journal of health and social behavior, Translational behavioral medicine, Social Science & Medicine)
- ▶ 11 packages/libraries built (ergmito, similR, gnet, fmcmm, slurmR, aphylo, polygons, pruner, netplot, rphyloxml, jsPhyloSVG)

# Essays on Bioinformatics and Social Network Analysis

Statistical and Computational Methods for Complex Systems





George G Vega Yon

University of Southern California, Department of Preventive Medicine

November 18, 2019



# Thanks!

-  Dodd, Diane M. B. (1989). "Reproductive Isolation as a Consequence of Adaptive Divergence in *Drosophila pseudoobscura*". In: Evolution 43.6, pp. 1308–1311. ISSN: 00143820, 15585646. URL: <http://www.jstor.org/stable/2409365>.
-  Engelhardt, Barbara E. et al. (2011). "Genome-scale phylogenetic function annotation of large and diverse protein families". In: Genome Research 21.11, pp. 1969–1980. ISSN: 10889051. DOI: 10.1101/gr.104687.109.
-  Engelhardt, Barbara E et al. (2005). "Protein Molecular Function Prediction by Bayesian Phylogenomics". In: PLOS Computational Biology 1.5. DOI: 10.1371/journal.pcbi.0010045. URL: <https://doi.org/10.1371/journal.pcbi.0010045>.
-  Jiang, Yuxiang et al. (Dec. 2016). "An expanded evaluation of protein function prediction methods shows an improvement in accuracy". In: Genome Biology 17.1, p. 184. ISSN: 1474-760X. DOI: 10.1186/s13059-016-1037-6. URL: <http://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1037-6>.



Oliver, Stephen (Feb. 2000). “Guilt-by-association goes global”. In: Nature 403.6770, pp. 601–602. ISSN: 0028-0836. DOI: 10.1038/35001165. URL: <http://www.nature.com/articles/35001165>.



Pesaranghader, Ahmad et al. (May 2016). “simDEF: definition-based semantic similarity measure of gene ontology terms for functional similarity analysis of genes”. In: Bioinformatics 32.9, pp. 1380–1387. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btv755. URL: <https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btv755>.



Piovesan, Damiano et al. (July 2015). “INGA: protein function prediction combining interaction networks, domain assignments and sequence similarity”. In: Nucleic Acids Research 43.W1, W134–W140. ISSN: 0305-1048. DOI: 10.1093/nar/gkv523. URL: <https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkv523>.



Yu, Chun et al. (Jan. 2018). “Assessing the Performances of Protein Function Prediction Algorithms from the Perspectives of Identification Accuracy and False Discovery Rate”. In: International Journal of Molecular Sciences 19.1, p. 183. ISSN: 1422-0067. DOI: 10.3390/ijms19010183. URL: <http://www.mdpi.com/1422-0067/19/1/183>.

## Example of GO term

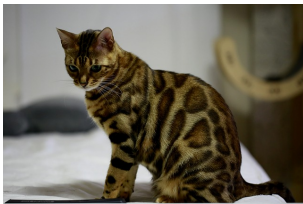
<b>Accession</b>	GO:0060047
<b>Name</b>	heart contraction
<b>Ontology</b>	biological_process
<b>Synonyms</b>	heart beating, cardiac contraction, hemolymph circulation
<b>Alternate</b>	IDs None
<b>Definition</b>	The multicellular organismal process in which the heart decreases in volume in a characteristic way to propel blood through the body. Source: GOC:dph

**Table 4** Heart Contraction Function. source: [amigo.geneontology.org](http://amigo.geneontology.org)

You know what is interesting about this function?

◀ go back

These four species have a gene with that function...



*Felis catus* pthr10037



*Oryzias latipes* pthr11521



*Anolis carolinensis* pthr11521



*Equus caballus* pthr24356

These four species have a gene with that function... and two of these are part of the same evolutionary tree!



*Felis catus* pthr10037



*Oryzias latipes* pthr11521



*Anolis carolinensis* pthr11521



*Equus caballus* pthr24356



There various approaches for this, some to highlight

- ▶ Text analysis like in Pesaranghader et al. 2016
- ▶ Protein-protein interaction networks like in Oliver 2000; Piovesan et al. 2015.
- ▶ Phylogenetic based like SIFTER Barbara E. Engelhardt et al. 2011, 2005.
  - ▶ Parameters to estimate:  $2^{2P}$ , where  $P$  is the number of functions.

(a nice literature review in Jiang et al. 2016; Yu et al. 2018)

◀ go back

# An evolutionary model of gene functions (algorithmic view)

**Data:** A phylogenetic tree,  $\{\pi, \mu, \psi\}$  (Model probabilities)

**Result:** An annotated tree

for  $n \in \text{PostOrder}(N)$  do

**Nodes gain/loss function depending on their parent;**

    switch class of  $n$  do

        case root node do

            Gain function with probability  $\pi$ ;

        case interior node do

            if Parent has the function then Keep it with prob.  $(1 - \mu_1)$ ;

            else Gain it with prob.  $\mu_0$ ;

    end

**Finally, we allow for mislabeling;**

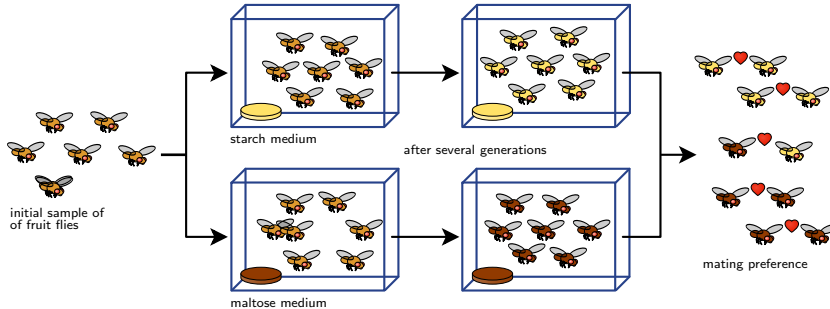
    if  $n$  is leaf then

        if has the function then Mislabel with prob.  $\psi_1$ ;

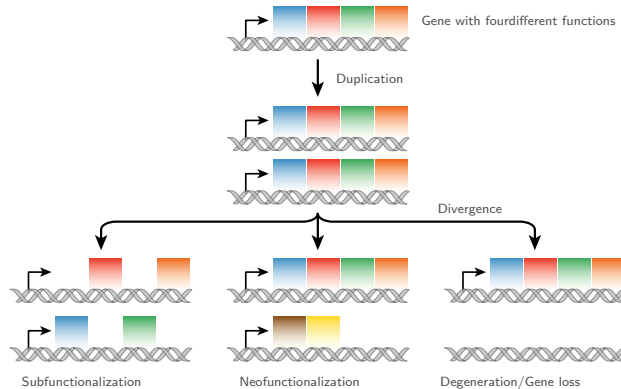
        else Mislabel with prob.  $\psi_0$ ;

end

► go back



**Figure 4** Dodd 1989: After one year of isolation, flies showed a significant level of assortativity in mating (wikimedia)




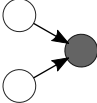



**Figure 5** A key part of molecular innovation, gene duplication provides opportunity for new functions to emerge (wikimedia)

- ▶ Simulation and visualization of annotated phylogenetic trees.
- ▶ Pruning algorithm implemented in C++ using the `pruner` template library (by-product).
- ▶ Uses metaprogramming (users can specify different formulas).
- ▶ The estimation is done using either Maximum Likelihood, Maximum A Posteriori, or MCMC.
- ▶ The MCMC estimation is done via the `fmcmc` R package using adaptive MCMC (also implemented as part of this project):
  - ▶ Automatic stop via convergence check.
  - ▶ Out-of-the-box parallel chains using parallel computing.
  - ▶ User-defined transition kernel (in our case, Adaptive Kernel).

◀ go back

## Sufficient statistics have various forms

Representation	Description
	Mutual Ties (Reciprocity) $\sum_{i \neq j} y_{ij} y_{ji}$
	Transitive Triad (Balance) $\sum_{i \neq j \neq k} y_{ij} y_{jk} y_{ik}$
	Homophily $\sum_{i \neq j} y_{ij} \mathbf{1}(x_i = x_j)$
	Covariate Effect for Incoming Ties $\sum_{i \neq j} y_{ij} x_j$
	Four Cycle $\sum_{i \neq j \neq k \neq l} y_{ij} y_{jk} y_{kl} y_{li}$

One of the most popular methods for estimating ERGMs is the MC-MLE approach (citations here)

This consists on the following steps

1. Start from a sensible guess on what should be the population parameters (usually done using pseudo-MLE estimation)
2. While the algorithm doesn't converge, do:
  - 2.1 Simulate a stream of networks with the current state of the parameter,  $\theta_t$
  - 2.2 Using the law of large numbers, approximate the ratio of likelihoods based on the parameter  $\theta_t$ , this is the objective function
  - 2.3 Update the parameter by a Newton-Raphson step
  - 2.4 Next iteration

◀ go back

In general

- ▶ Implements estimation of ERGMs using exact statistics for small networks
- ▶ Meta-programming allows specifying likelihood (and gradient) functions for joint models (a function that writes a function)
- ▶ Includes tools for simulating, and post-estimation checks
- ▶ Getting ready for CRAN!

More specific tricks

- ▶ Computes support of  $\Pr$  using `ergm::ergm.allstats`
- ▶ It includes a vectorized function doing the same
- ▶ Scales up nice (hundreds of small networks) saving space and computation (when possible)
- ▶ Highly tested (90% coverage with more than one hundred tests)



We performed a simulation study with the following features:

- ▶ Draw 20,000 samples of groups of small networks

◀ go back

We performed a simulation study with the following features:

- ▶ Draw 20,000 samples of groups of small networks
- ▶ Each group had prescribed: (model parameters, number of networks, sizes of the networks)

◀ go back

We performed a simulation study with the following features:

- ▶ Draw 20,000 samples of groups of small networks
- ▶ Each group had prescribed: (model parameters, number of networks, sizes of the networks)
- ▶ Each group could have from 5 to 300 small networks

◀ go back

We performed a simulation study with the following features:

- ▶ Draw 20,000 samples of groups of small networks
- ▶ Each group had prescribed: (model parameters, number of networks, sizes of the networks)
- ▶ Each group could have from 5 to 300 small networks
- ▶ We estimated the models using MC-MLE and MLE.

◀ go back

