# 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

Keck School of Medicine of USC

Keck School of Medicine of USC

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

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

Keck School of Medicine of USC

**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.

Keck School of Medicine of USC

**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.

# Contents

## On the prediction of gene functions using phylogenetic trees

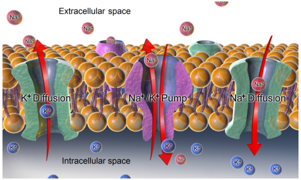
*Joint with*: Paul D Thomas, Paul Marjoram, Huaiyu Mi, Duncan Thomas, and John Morrison

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**

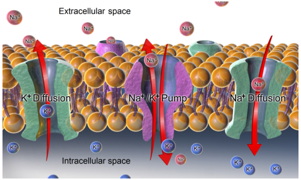
Active transport GO:0005215



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

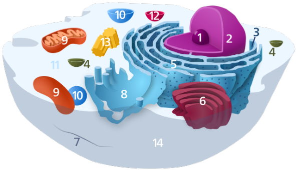
**Molecular function**

Active transport GO:0005215




**Cellular component**

Mitochondria GO:0004016

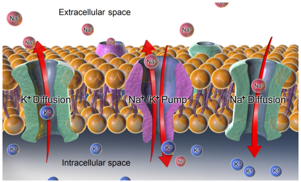


Keck School of Medicine of USC

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

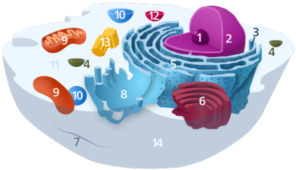
**Molecular function**

Active transport GO:0005215



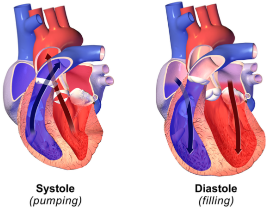

**Cellular component**

Mitochondria GO:0004016




**Biological process**

Heart contraction GO:0060047



1. Understanding genes means understanding biology
2. Far more than simply persuing knowledge, this means that we can actually use this information towards

GENEONTOLOGY
Unifying Biology

- $\sim$ 44,700 validated terms, $\sim$ 7,300,000 annotations on $\sim$ 4,500 species.

**source**: Statistics from `pantherdb.org` and `geneontology.org`

- $\sim$ 44,700 validated terms, $\sim$ 7,300,000 annotations on $\sim$ 4,500 species.
- About $\sim$ 500,000 are on human genes.

**source**: Statistics from `pantherdb.org` and `geneontology.org`

- $\sim$ 44,700 validated terms, $\sim$ 7,300,000 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` and `geneontology.org`

# The Gene Ontology Project

Keck School of Medicine of USC

GENEONTOLOGY
Unifying Biology

- $\sim$ 44,700 validated terms, $\sim$ 7,300,000 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` and `geneontology.org`

- The Gene Ontology Project, which is an international scientific effort to develop a knowledge base of biology from molecular level up to organism-level systems.

  > *[...] develop an up-to-date, comprehensive, computational model of biological systems, from the molecular level to larger pathways, cellular and organism-level systems.*
  > *It has a large collection of genetic annotations from various types of evidence including: experimentally, human curated information, and machine inferred.*

- A long way since 1999 (20 years), there's still a lot to learn
- This information has been crucial for biomedical research (e.g. translating GWAS to treatment.)
- Let me show you an example of a GO annotaiton

Keck School of Medicine of USC

Example of GO term

| | |
|---|---|
| **Accession** | GO:0060047 |
| **Name** | heart contraction |
| **Ontology** | biological_process |
| **Synonyms** | heart beating, cardiac contraction, hemolymph circulation |
| **Alternate** | IDs None |
| **Definition** | 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 1** Heart Contraction Function. source: amigo.geneontology.org

Example of GO term

| | |
|---|---|
| **Accession** | GO:0060047 |
| **Name** | heart contraction |
| **Ontology** | biological_process |
| **Synonyms** | heart beating, cardiac contraction, hemolymph circulation |
| **Alternate** | IDs None |
| **Definition** | 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 1** Heart Contraction Function. source: amigo.geneontology.org

You know what is interesting about this function?

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

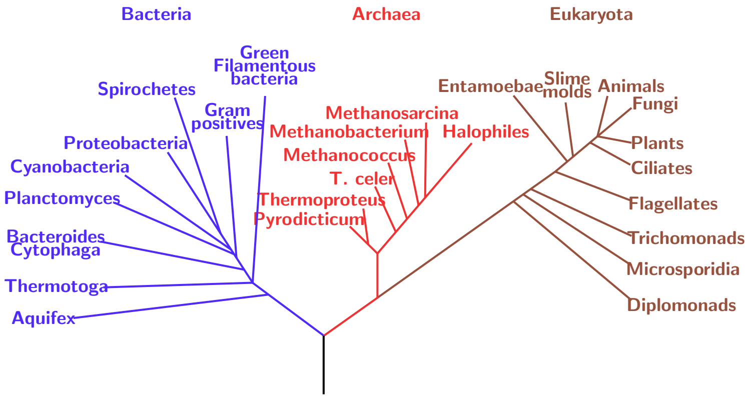



**Figure 1** A phylogenetic tree of living things, based on RNA data and proposed by Carl Woese, showing the separation of bacteria, archaea, and eukaryotes (wiki)

1. Phylogenetic trees show evolutionary relationships between species
2. Traditionally, we think about these based on say physical features, nowadays we build trees based on genetic distances between species.

- 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 (like the lizard and the fish)

# Phylogenetic Trees: The PANTHER classification system

- 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 (like the lizard and the fish)

- 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 (like the lizard and the fish)

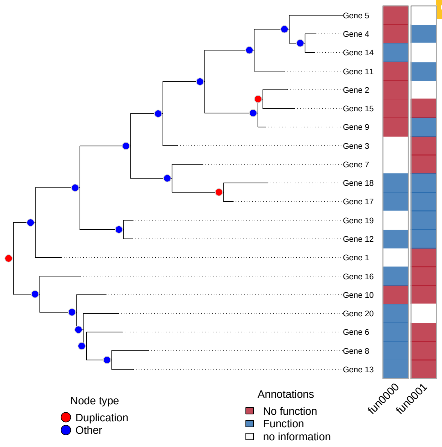



**Figure 2** Simulated phylogenetic tree and gene annotations.

1. Here we have an example of a (simulated) phylogenetic tree.
2. We will see this a couple of times during the presentation
3. This figure summarizes the information that I will be using to infere gene functions:
    - The tip nodes (leafs) are modern (known) genes
    - In general, The color bars next to each gene represent genetic annotations (GO terms) in three different states: Has the function (blue), does not have the function (red), and no information (white)
    - Each interior node represent ancestors which are classify as duplication/speciation/or horizontal transfer nodes
    - This is an hypothesis regarding to what type of event lead to a split in the family.
    - we mostly care about whether these are duplication nodes not since we believe that functional gain and loses are more likely to happen at this stage.

We can use

**evolutionary trees**

to inform a model for predicting

**genetic annotations!**

Keck School of Medicine of USC

There various approaches for this, some to highlight

- Text analysis like in Pesaranghader et al. 2016

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

# Predicting gene functions

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.

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

# Predicting gene functions

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.

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

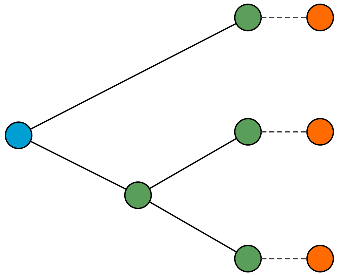
There various approaches for this, some to highlight

- 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)

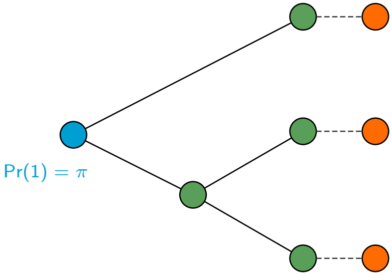
1. The last one being the most closely related to what we propose here (details to be shown).
2. In SIFTER, functions are modeled using a transition matrix in a Markov continuous model.
3. The main problem with this is that the computational complexity of the model grows horribly (estimating a model with a 100 functions) takes literally infinite time.
4. B/c of this, they truncate some of their modelling and work with small sets of up to 5 functions in a single tree (for example).
5. One key point of most of these models is that these provide a point estimate rather than a distribution, and mostly a binary estimate of the annotation (yes/no).

# An evolutionary model of gene functions



# An evolutionary model of gene functions

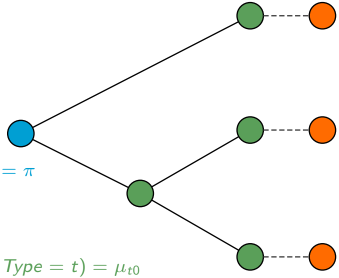
- Initial (spontaneous) gain of function.



# An evolutionary model of gene functions

Keck School of Medicine of USC

- Initial (spontaneous) gain of function.
- Loss/gain of offspring depends on: (a) the state of its' parents (**Markov process**), and (b) the type of node ▸ more



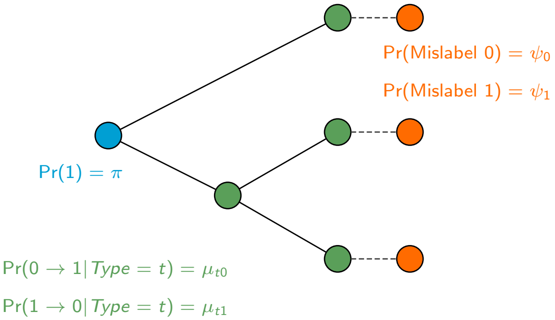
$\mathrm{Pr}(1) = \pi$

$\mathrm{Pr}(0 \to 1 | \mathit{Type} = t) = \mu_{t0}$

$\mathrm{Pr}(1 \to 0 | \mathit{Type} = t) = \mu_{t1}$

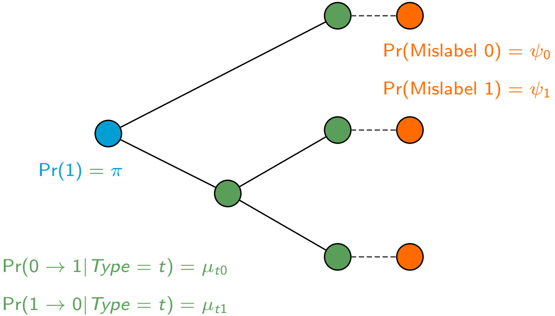
# An evolutionary model of gene functions

- Initial (spontaneous) gain of function.
- Loss/gain of offspring depends on: (a) the state of its' 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 its' 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.

1. In the version of the qual document you saw an implementation of the model that did not incorporated information regarding the node types, but that is trivially added by just adding a separate gain/loss parameter per type
2. Another venue we have explored is accounting for publicaction bias, most annotations are of the positive type (has function), but few are (no function).
3. we have failed in the last tests.
4. The model has been throughly tested. In particular, we did a large scale simulation study in which we used all 15,000 trees from panther to simulate annotations and then fitted our model using MCMC to check for bias and coverage probabilities (which are available in the paper)
5. The experiment was carriedout using USC's High Performance Computing cluster with the R package slurmR (described in the document).
6. Now, I will show you more recent information in which we take data from PantherDB with GO annotations and fit a large pooled model.

Keck School of Medicine of USC

| | (1) | (2) |
|---|---|---|
| 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 |
| Prior | Uniform | Beta |
| AUC (mean) | 0.69 | 0.67 |
| AUC (median) | 0.81 | 0.75 |

**Table 2** Parameter estimates using different priors.

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

# Prediction with real data

Keck School of Medicine of USC

| | (1) | (2) |
|---|---|---|
| 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 |
| Prior | Uniform | Beta |
| AUC (mean) | 0.69 | 0.67 |
| AUC (median) | 0.81 | 0.75 |

**Table 2** Parameter estimates using different priors.

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

# Prediction with real data

| | (1) | (2) |
|---|---|---|
| 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 |
| Prior | Uniform | Beta |
| AUC (mean) | 0.69 | 0.67 |
| AUC (median) | 0.81 | 0.75 |

**Table 2** Parameter estimates using different priors.

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

# Prediction with real data

Keck School of Medicine of USC

| | (1) | (2) |
|---|---|---|
| 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 |
| Prior | Uniform | Beta |
| AUC (mean) | 0.69 | 0.67 |
| AUC (median) | 0.81 | 0.75 |

**Table 2** Parameter estimates using different priors.

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

# Prediction with real data

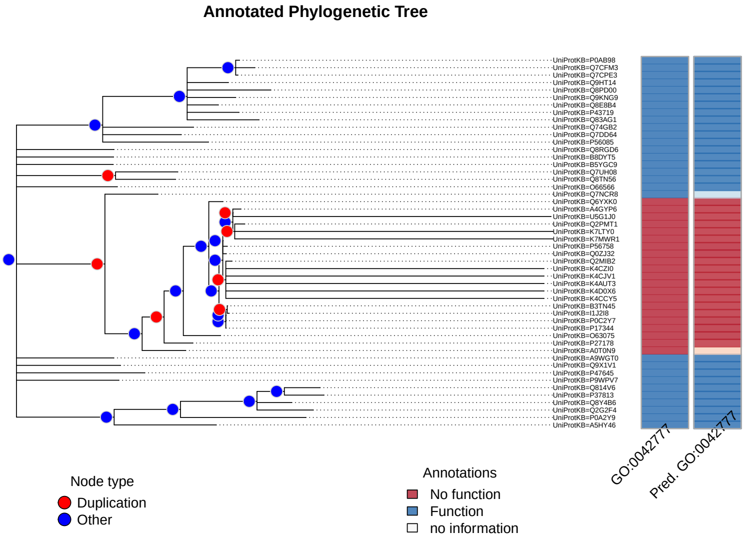
| | (1) | (2) |
|---|---|---|
| Mislab. prob. | | |
| $\psi_0$ | 0.23 | 0.25 |
| $\psi_1$ | 0.01 | 0.01 |
| **Gain/Loss at dupl.** | | |
| $\mu_{d0}$ | 0.97 | 0.96 |
| $\mu_{d1}$ | 0.52 | 0.58 |
| **Gain/Loss at spec.** | | |
| $\mu_{s0}$ | 0.05 | 0.06 |
| $\mu_{s1}$ | 0.01 | 0.02 |
| Root node | | |
| $\pi$ | 0.81 | 0.45 |
| Prior | Uniform | Beta |
| AUC (mean) | 0.69 | 0.67 |
| AUC (median) | 0.81 | 0.75 |

**Table 2** Parameter estimates using different priors.

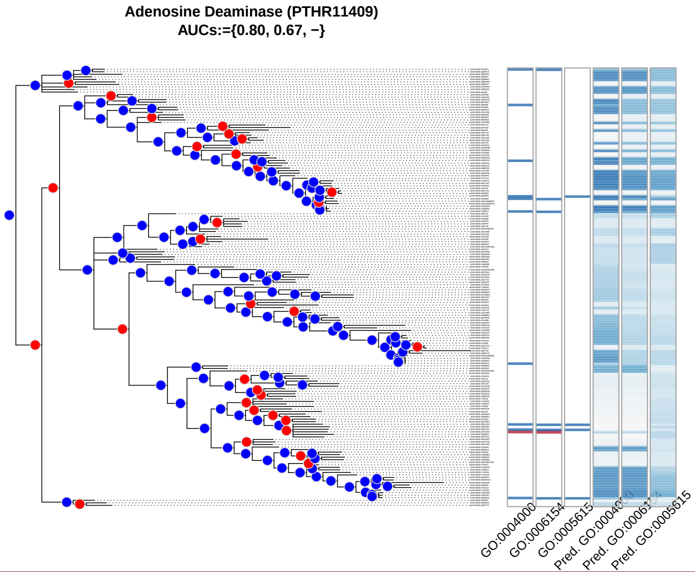
- 141 pooled functions (trees) with 7,388 genes with 0/1 annotations.
- Parameter estiamtes are actually probabilities.
- Data driven results (uninformative prior).
- Biologically meaningful results.
- Took about 5 minutes each.

1. The data used here corresponds to a subset of the trees.
2. Right now, the main criteria was: (1) must have at least one annotation of each type, and (2) must not have large sets of siblings (this due to numerical underflow issues, WIP)

# Prediction with real data: Leave-one-out



Keck School of Medicine of USC



## 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.

## Next steps

- Make the model hierarchical when pooling trees

## Next steps

- Make the model hierarchical when pooling trees
  - Different mutation rates per class of tree.
  - It can also account for mutation rate as a function of type of function
  - Can be complicated to fit (how many classes?)

Keck School of Medicine of USC

## Next steps

- Make the model hierarchical when pooling trees
  - Different mutation rates per class of tree.
  - It can also account for mutation rate as a function of type of function
  - Can be complicated to fit (how many classes?)
- Use a framework similar to Exponential Random Graph Models:

## Next steps

- Make the model hierarchical when pooling trees
  - Different mutation rates per class of tree.
  - It can also account for mutation rate as a function of type of 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 aditional information such as branch lenghts.
  - Yet computationally more compact compared to SIFTER (iso-statistics).

Keck School of Medicine of USC

## Next steps

- Make the model hierarchical when pooling trees
  - Different mutation rates per class of tree.
  - It can also account for mutation rate as a function of type of 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 aditional information such as branch lenghts.
  - Yet computationally more compact compared to SIFTER (iso-statistics).

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

**Example 1**

**Example 1**

- 2 siblings 2 function involves modelling the following array:

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

Keck School of Medicine of USC

### Example 1

- 2 siblings 2 function involves modelling the following array:

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

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

**Example 1**

- 2 siblings 2 function involves modelling the following array:

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

- 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.

Keck School of Medicine of USC

**Example 1**

- 2 siblings 2 function involves modelling the following array:

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

- 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:

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

- 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:

$$\left[\begin{array}{c} x_{p1} \\ x_{p2} \end{array}\right] \rightarrow \left(\left[\begin{array}{c} x_{i1} \\ x_{i2} \end{array}\right], \left[\begin{array}{c} x_{j1} \\ x_{j2} \end{array}\right]\right)$$

- 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.

# Contents

## Exponential Random Graph Models for Small Networks

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

## What are Exponential Random Graph Models

Exponential Family Random Graph Models, aka ERGMs are:

## What are Exponential Random Graph Models

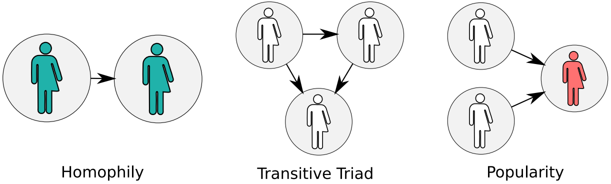
Exponential Family Random Graph Models, aka ERGMs are:

- Statistical models of (social) networks

# What are Exponential Random Graph Models

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



Keck School of Medicine of USC

A vector of model parameters

A vector of sufficient statistics

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

Observed data

The normalizing constant

All possible networks

more on terms

Keck School of Medicine of USC

A vector of model parameters

A vector of sufficient statistics

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

Observed data

The normalizing constant

All possible networks

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

more on terms

## ERGMs: State of the Art

Keck School of Medicine of USC

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)

# ERGMs: State of the Art

Keck School of Medicine of USC

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 the millions of vertices)

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

# ERGMs: State of the Art

Keck School of Medicine of USC

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 the millions of vertices)

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

What about small networks?

# 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

# Do we care about small networks?

We see small networks everywhere

- Families and friends
- Small teams

# Do we care about small networks?

We see small networks everywhere

- Families and friends
- Small teams
- Egocentric networks

# Do we care about small networks?

We see small networks everywhere

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

# Do we care about small networks?

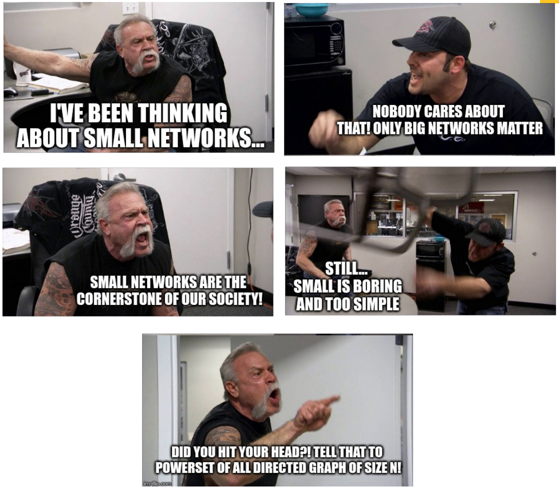
We see small networks everywhere

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

# Do we care about small networks?

We see small networks everywhere

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



From the methodological point of view, current methods are great, but:

From the methodological point of view, current methods are great, but:

- Possible accuracy issues (error rates)

From the methodological point of view, current methods are great, but:

- Possible accuracy issues (error rates)
- Prone to degeneracy problems (sampling and existence of MLE)

From the methodological point of view, current methods are great, but:

- Possible accuracy issues (error rates)
- Prone to degeneracy problems (sampling and existence of MLE)
- It is not MLE...

- 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**.

# ERGMs for Small Networks

A vector of model parameters

A vector of sufficient statistics

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

Observed data

The normalizing constant

All possible networks

- 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**.

Keck School of Medicine of USC

A vector of model parameters

A vector of sufficient statistics

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

Observed data

The normalizing constant

All possible networks

- 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.

A vector of model parameters

A vector of sufficient statistics

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

Observed data

The normalizing constant

All possible networks

- 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

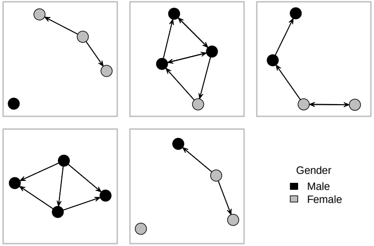



**Figure 3** Random sample of 5 networks simulated using the ergmito package

Keck School of Medicine of USC

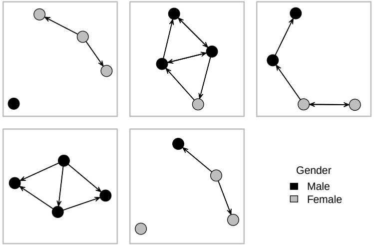



**Figure 3** Random sample of 5 networks simulated using the ergmito package

| | Bernoulli | Full model |
|---|---|---|
| Edge-count | $-0.69^{*}$ | $-1.70^{**}$ |
| | (0.27) | (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 3** Fitted ERGMitos using the fivenets dataset.

Keck School of Medicine of USC

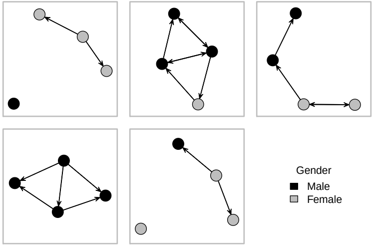



**Figure 3** Random sample of 5 networks simulated using the ergmito package

| | Bernoulli | Full model |
|---|---|---|
| Edge-count | $-0.69^{*}$ | $-1.70^{**}$ |
| | (0.27) | (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 3** Fitted ERGMitos using the fivenets dataset.

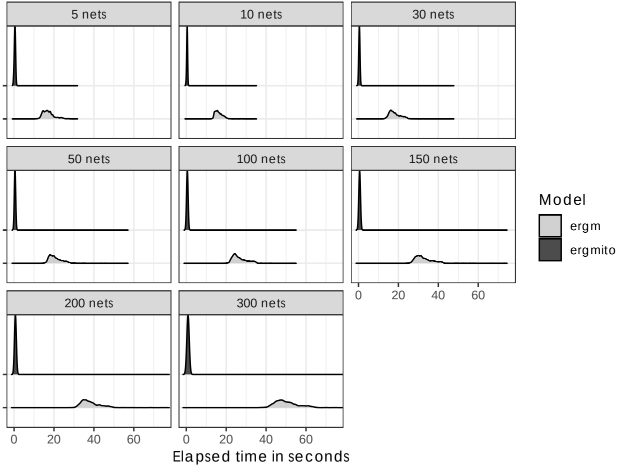
We performed a large simulation study ▸ more comparing MC-MLE (ergm) with MLE (ergmito).

Keck School of Medicine of USC

| Sample size | N. Simulations | P(Type I error) MC-MLE (ergm) | P(Type I error) MLE (ergmito) | $\chi^2$ |
|---|---|---|---|---|
| 5 | 2,189 | 0.084 | 0.057 | 11.71 *** |
| 10 | 2,330 | 0.070 | 0.045 | 12.46 *** |
| 15 | 2,395 | 0.084 | 0.066 | 5.55 * |
| 20 | 2,430 | 0.074 | 0.060 | 3.58 |
| 30 | 2,460 | 0.057 | 0.052 | 0.67 |
| 50 | 2,495 | 0.046 | 0.044 | 0.17 |
| 100 | 2,499 | 0.048 | 0.048 | 0.00 |

**Table 4** 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 * $p < 0.05$. The lack of fitted samples in some levels is due to failure of the estimation method.

# Paper 2 Simulation Studies: Elapsed time




more results

Keck School of Medicine of USC

### 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.

Keck School of Medicine of USC

### 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.

### Next steps

- Revisit measurement of goodness-of-fit.
- Explore extending this method for (very) large networks.

# 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

Keck School of Medicine of USC

# Thanks!

# References I


- 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.

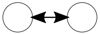
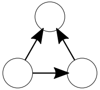
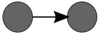
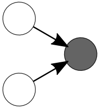
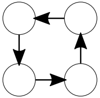
Keck School of Medicine of USC

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.

Here are some by-products of my research here at USC

- The slurmR R package
- The pruner C++ library
- The fmcmc R package

## **Sufficient statistics** have various forms

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

go back

# ERGMs: The MC-MLE approach

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

# The `ergmito`

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)

go back

# Paper 2 Simulation Studies

We performed a simulation study with the following features:

- Draw 20,000 samples of groups of small networks

go back

Keck School of Medicine of USC

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

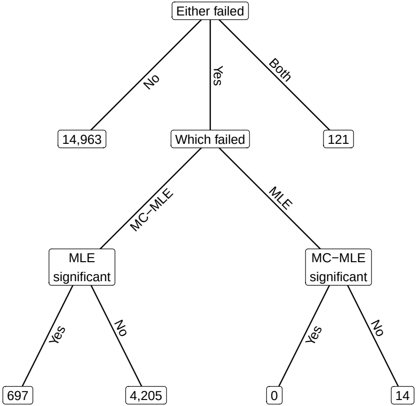
Keck School of Medicine of USC

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.

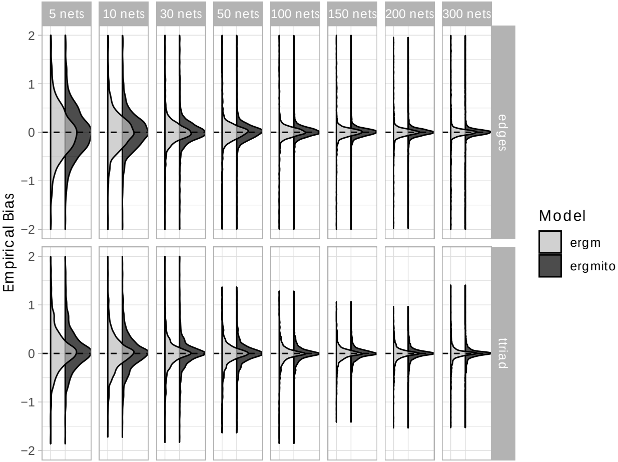
go back

## Paper 2 Simulation Studies: Error rate




go back

# Paper 2 Simulation Studies: Empirical Bias




go back

# An evolutionary model of gene functions (algorithmic view)

Keck School of Medicine of USC

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

```
for n ∈ PostOrder(N) do
    Nodes gain/loss function depending on their parent;
    switch class of n do
        case root node do
            Gain function with probability π;
        case interior node do
            if Parent has the function then Keep it with prob. (1 − μ1);
            else Gain it with prob. μ0;
    end
    Finally, we allow for mislabeling;
    if n is leaf then
        if has the function then Mislabel with prob. ψ1;
        else Mislabel with prob. ψ0;
end
```

go back

Keck School of Medicine of USC

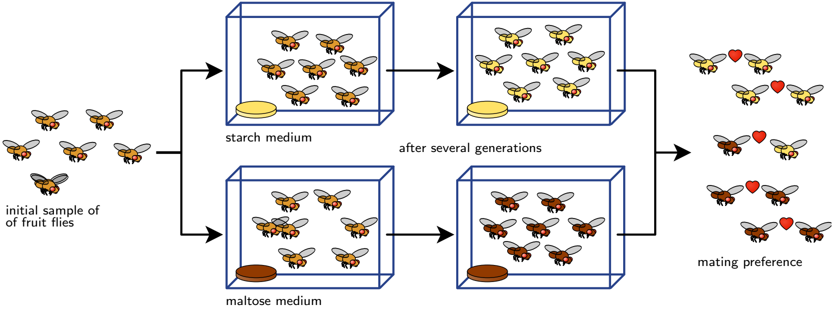



**Figure 4** Dodd 1989: After one year of isolation, flies showed a significant level or assortativity in mating (wikimedia)

go back

Keck School of Medicine of USC

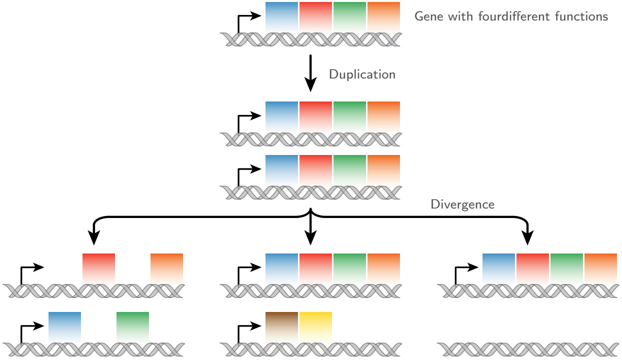



**Figure 5** A key part of molecular innovation, gene duplication provides opportunity for new functions to emerge (wikimedia)

go back

# The `aphylo` R package

- 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 Posteriory, 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