# Functional Enrichment Analysis for Gene Set

**Yinliang Liu**

*lyl1212@gmail.com*

Institute of Applied Mathematics, AMSS,
Chinese Academy of Sciences

December 4, 2014

数系天地 勤笃求真

Bioinformatics ZHANGroup

# Outline

1. Background
2. Model
   - Model I
   - Model II
3. Discussion

# Outline

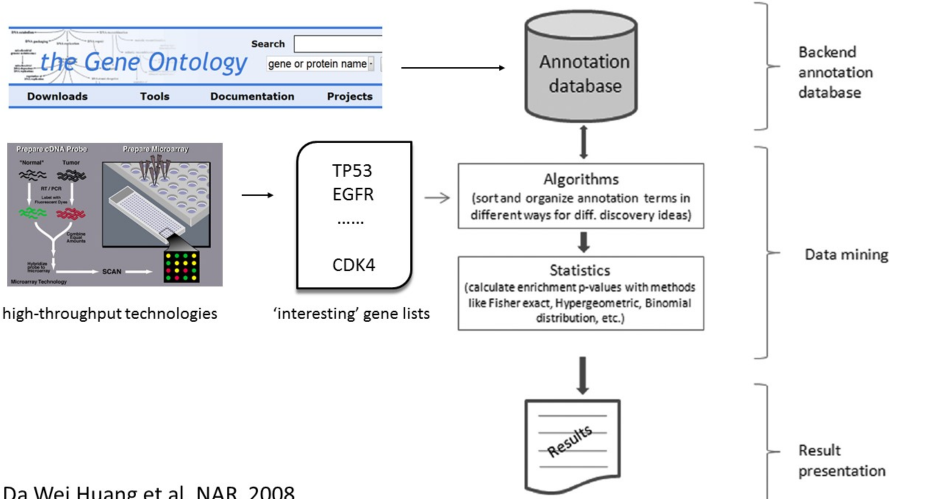
1. Background

2. Model
   - Model I
   - Model II

3. Discussion

# Background

## Workflow:



Annotation database

TP53
EGFR
……
CDK4

high-throughput technologies

'interesting' gene lists

Algorithms
(sort and organize annotation terms in different ways for diff. discovery ideas)

Statistics
(calculate enrichment p-values with methods like Fisher exact, Hypergeometric, Binomial distribution, etc.)

Backend annotation database

Data mining

Result presentation

Da Wei Huang et al. NAR. 2008

Bioinformatics ZHANGroup

# Outline

1 Background

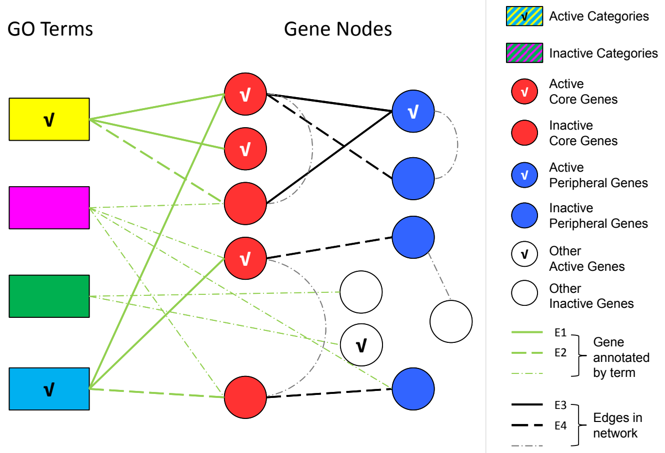
2 Model

- Model I
- Model II

3 Discussion

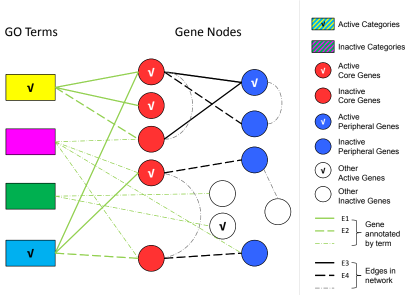
# Model I

## Network-based Functional Analysis for Gene Set



# Model I

$$
\begin{aligned}
L(C|p_1, p_2, q, G) \\
&= |E_1| \log p_1 + |E_2| \log(1 - p_1) \\
&\quad + |E_3| \log p_2 + |E_4| \log(1 - p_2) \\
&\quad + |N_A| \log q + |N_I| \log(1 - q) - \alpha |C|
\end{aligned}
\tag{1}
$$



(i) $N_A$: nodes of other active genes

(ii) $N_I$: nodes of other inactive genes

(iii) $E_1$: edges from active categories to active core genes

(iv) $E_2$: edges from active categories to inactive core genes

(v) $E_3$: edges from core genes to active peripheral genes

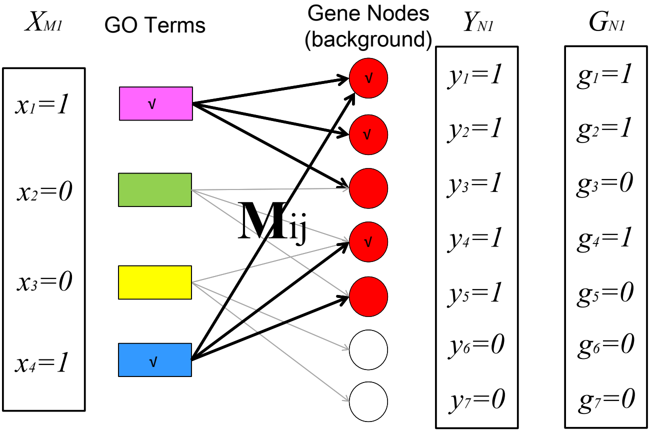
(vi) $E_4$: edges from core genes to inactive peripheral genes

# Model I

**Model I can be formulated into an integer quadratic programming:**

$$
\begin{aligned}
\max \quad & \sum_i \sum_j x_i \boldsymbol{M}_{ij} g_j \log p_1 + \sum_i \sum_j x_i \boldsymbol{M}_{ij} (1 - g_j) \log(1 - p_1) \\
& + \sum_i \sum_j y_i \boldsymbol{N}_{ij} (1 - y_j) g_j \log p_2 + \sum_i \sum_j y_i \boldsymbol{N}_{ij} (1 - y_j)(1 - g_j) \log(1 - p_2) \\
& + \sum_j (1 - y_j)(1 - z_j) g_j \log q + \sum_j (1 - y_j)(1 - z_j)(1 - g_j) \log(1 - q) \\
& - \alpha \sum_i x_i \\
\text{s.t.} \quad & y_j \leq \sum_i \boldsymbol{M}_{ij} x_i, \; j = 1, 2, \cdots, N \\
& y_j \geq \boldsymbol{M}_{ij} x_i, \qquad i = 1, 2, \cdots, M, \quad j = 1, 2, \cdots, N \\
& z_j \leq \sum_i \boldsymbol{N}_{ij} y_i, \; j = 1, 2, \cdots, N \\
& z_j \geq \boldsymbol{N}_{ij} y_i, \qquad i = 1, 2, \cdots, M, \quad j = 1, 2, \cdots, N \\
& x_i = \{0, 1\}, \qquad i = 1, 2, \cdots, M \\
& y_j = \{0, 1\}, \qquad j = 1, 2, \cdots, N \\
& z_j = \{0, 1\}, \qquad j = 1, 2, \cdots, N
\end{aligned}
\tag{2}
$$

# Model I



M为所有term的个数

N为所有gene的个数

return

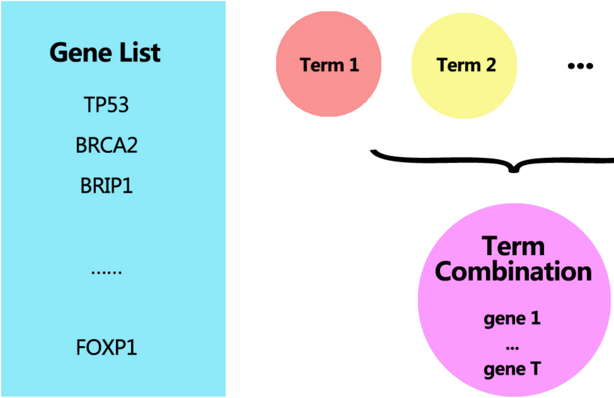
# Outline

1. Background

2. Model
   - Model I
   - Model II

3. Discussion

# Model II



Gene List

TP53

BRCA2

BRIP1

……

FOXP1

Term 1

Term 2

•••

Term x

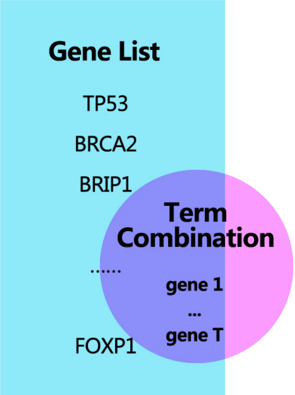
Term Combination

gene 1

...

gene T

# Model II



Gene List

TP53

BRCA2

BRIP1

......

FOXP1

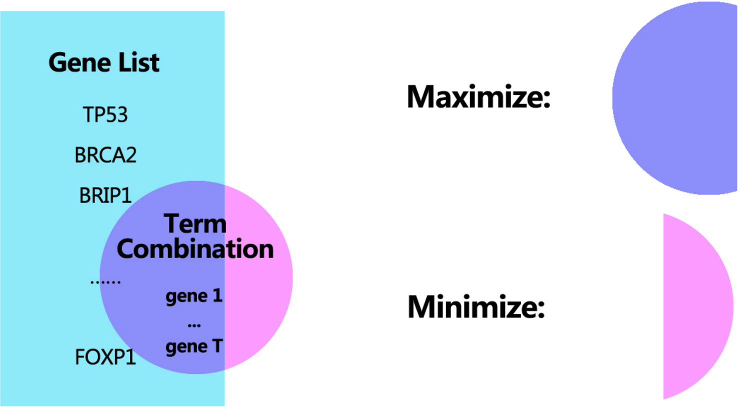
Term Combination

gene 1

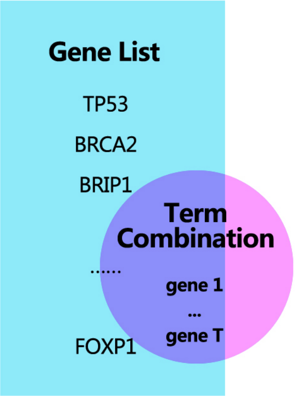
...

gene T

# Model II


Gene List
TP53
BRCA2
BRIP1
......
FOXP1
Term Combination
gene 1
...
gene T
Maximize:
Minimize:

# Model II



**A candidate modol:**

$$\begin{aligned} \min \quad & |T - t| \\ \text{s.t.} \quad & t \geq \alpha G \end{aligned} \tag{3}$$

Where $T$ denotes the amount of genes the combination term contained , $t$ denotes the size of the intersection of the 'interesting' gene list and combination term set, and $G$ denotes the size of the 'interesting' gene list. $\alpha$ is a parameter to control the degree of coverage.

# Model II

## Model II can be formulated into an integer programming:

$$
\begin{aligned}
\max \quad & \sum_j (1-g_j)y_j + \lambda \sum_i x_i \\
\text{s.t.} \quad & y_j \leq \sum_i \boldsymbol{M}_{ij} x_i, \; j = 1, 2, \cdots, N \\
& y_j \geq \boldsymbol{M}_{ij} x_i, \qquad i = 1, 2, \cdots, M, \quad j = 1, 2, \cdots, N \\
& \alpha \leq \sum_j g_j y_j \\
& x_i = \{0, 1\}, \qquad i = 1, 2, \cdots, M \\
& y_j = \{0, 1\}, \qquad j = 1, 2, \cdots, N
\end{aligned}
\tag{4}
$$

✓ Expatiation

- $x_i$, $y_j$, $g_j$ and $\boldsymbol{M}_{ij}$ are defined as before. legend
- $\lambda$ can be defined to be $1/(M+1)$ to make sure that for each $T$ and $t$, the size of optimal solution is minimum($M$ is the amount of all terms).

✓ Execution

- from $\alpha = 1$.
- for every optimal $Y$, let $\alpha = \sum_j g_j y_j + 1$, continnue.
- Untill $\alpha = G$($G$ is the size of the 'interesting' gene list.).

return

# Outline

1. Background
2. Model
   - Model I
   - Model II
3. Discussion

# Discussion

- **Models**

  Are there some advice to the models previously mentioned ?

  Model I  Model II

# Discussion

- **Models**

  Are there some advice to the models previously mentioned ?

  Model I Model II

- **Algorithm**

  Any fast algorithm can get approximate answer adequate for further using ?

# Discussion

- **Models**

  Are there some advice to the models previously mentioned ?

  Model I Model II

- **Algorithm**

  Any fast algorithm can get approximate answer adequate for further using ?

- **Meaning**

  Where to focus on for further analysis ?

# Thank you for attention!

Email: *lyl1212@gmail.com*