

Discovering Classes in Microarray Data Using Island Counts

New Mexico Bioinformatics Symposium, March 2005

Brendan Mumey

Computer Science, Montana State University

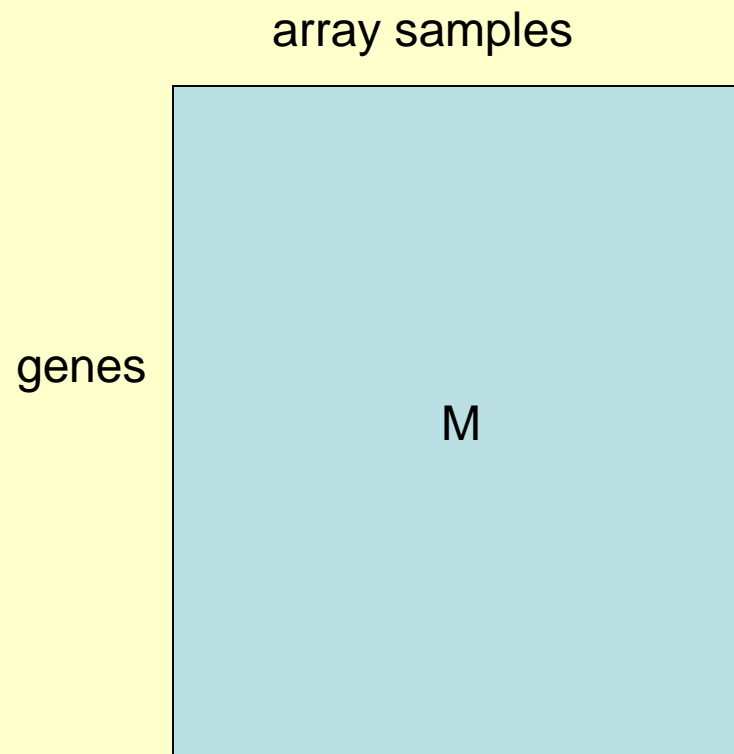
Joint work with Mike and Louise Showe,

Wistar Institute



Microarray Data

- Large matrix: *array samples x genes*



Biclustering...

- We are typically interested in determining a **partition of samples** based on a variety of potential criteria:
 - *Disease condition*
 - *Tissue type*
 - *Developmental stage*
- These partitions should be well characterized by a **set of genes** whose expression levels are correlated with the sample partitions

The Color Island statistic

- For a particular gene **g**, color the values of **g** according to the tentative class assignment.
- Then **sort** the colored values for gene **g**
- And **count** the number of *color islands*...

	Tissue sample					
	1	2	3	4	5	6
Gene g values:	1.7	6.2	1.3	5.7	0.9	7.2
Assigned class:	1	2	1	2	2	1
Values sorted:	0.9	1.3	1.7	5.7	6.2	7.2

=> # of color islands = 3

Why count color islands?

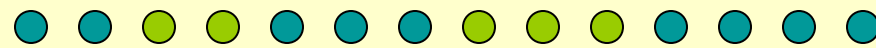
- A **low** number of color changes indicates that **g** supports and is informative of the given partitioning.
- *If we find a set of genes that all have low color changes, then this is evidence that:*
 - *We have found a good partitioning of the array samples*
 - *The genes selected are relevant in the biological processes that discriminate between the sample classes*

Significance Scoring

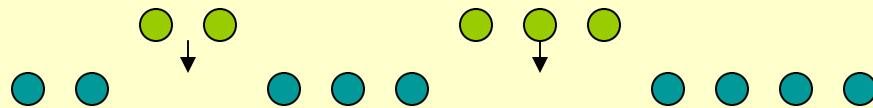
- Given a color island statistic for a gene, how significant is it?
- Compute a **p-value**...
 - We need to count the number of ways that n objects partitioned into (s_1, \dots, s_k) distinguishable classes can be arranged to form

Calculating a P-value (2 color case)

- We have s_1 blue values and s_2 green value ($s_1 + s_2 = n$).



- Let c be the number of color islands.



$$CI(\{s_1, s_2\}, c) = \binom{s_1 + 1}{c} \binom{s_2 - 1}{c - 1}$$

Calculating a P-value (general case)

- Let $CI(\{\mathbf{s}_1, \dots, \mathbf{s}_k\}, \mathbf{c}) = \#$ of distinguishable permutations with \mathbf{c} color islands...

$$CI(\{s_1, \dots, s_k\}, c) = \sum_{\substack{d+e+2f=c \\ d,e,f \geq 0}} CI(\{s_1, \dots, s_{k-1}\}, d) \binom{d+1}{e} \binom{n-s_k-d}{f} \binom{s_k-1}{e+f-1}$$

P-value computation

- Significance = probability of occurrence by chance = ...

$$\Pr(\text{island_count}(g) = c \mid P) = CI(\{s_1, \dots, s_k\}, c) / \left(\frac{n!}{s_1! s_2! \dots s_k!} \right)$$

- Take *log* and approximate to simplify computation:

$$\text{p_score}(g) = -\sum_i \log \binom{s_i + 1}{c_i} + \log \binom{n - s_i - 1}{c_i - 1} - \log \left(\frac{n!}{s_i! (n - s_i)!} \right)$$

(note: tolerates missing values)

Formal GENEPART Optimization Problems

- **Input:** The input consists of an $m \times n$ expression matrix E , k , the number of partitions, s , a minimum class size, and p , the desired size of the selected gene set G .
- **Output:** The output is the partition $P = \{P_1, \dots, P_k\}$ and gene set G subject to the constraints $|P_i| \geq s$ for $i = 1 \dots k$ and $|G|=p$ and that (P, G) is the solution to:

$$\arg \min_{(P, G)} \sum_{g \in G} \text{island_count}(g) \quad \text{(Version 1)}$$

OR

$$\arg \max_{(P, G)} \sum_{g \in G} \text{p_score}(g) \quad \text{(Version 2)}$$

GENEPART Version 1 is NP-complete

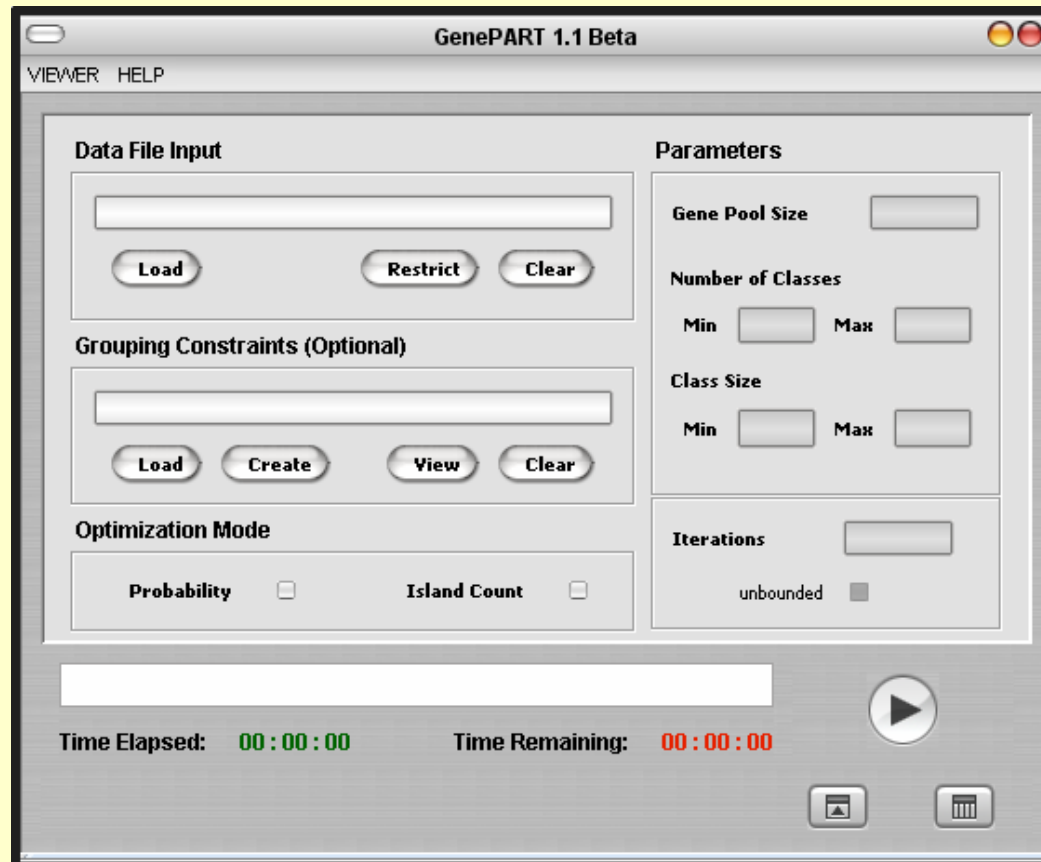
- The reduction is from a graph partitioning problem; the problem of deciding whether a graph G contains a *complete bipartite subgraph* of a given size.

Experimental Results

- We have tested the algorithm on a 30 sample Cutaneous T-cell Lymphoma data set (and others)
- It was able to almost perfectly discriminate short-term survivors from long-term survivors and normal controls.
- Genes include ones previously identified with discriminant analysis plus new ones that also appear informative
- [\(see accompanying GenePart Report\)](#)

Software

- An efficient branch-and-bound algorithm for solving GENEPART has been implemented in Java:



To Download

- GenePart can be downloaded from:

www.cs.montana.edu/mumey/genepart/

- Requirements:
 - Java 1.5 runtime
 - 256+ Mb RAM