Pattern Mining in Microarray

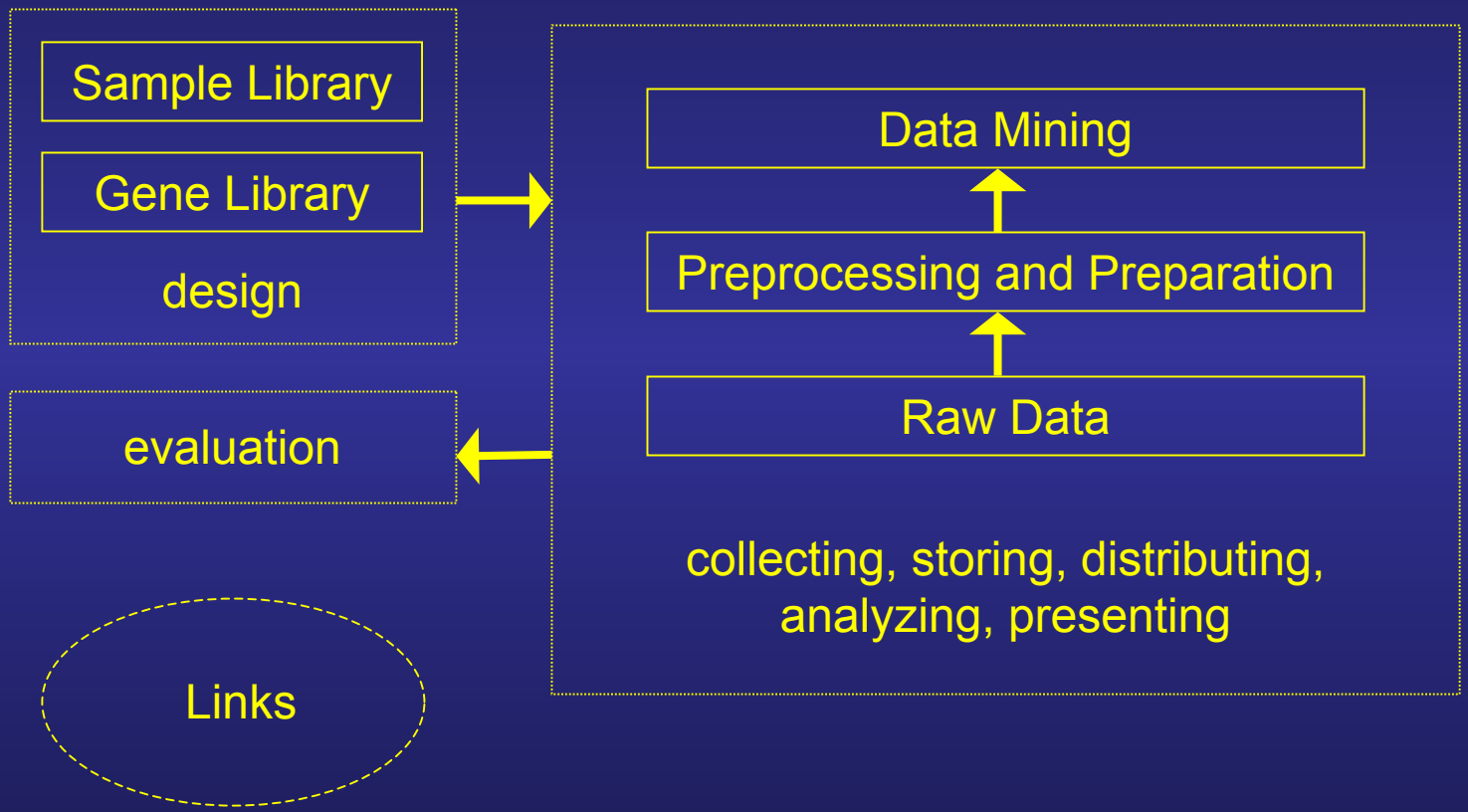
Eric Y. Wu
Zhengfan Dai

Florida International University

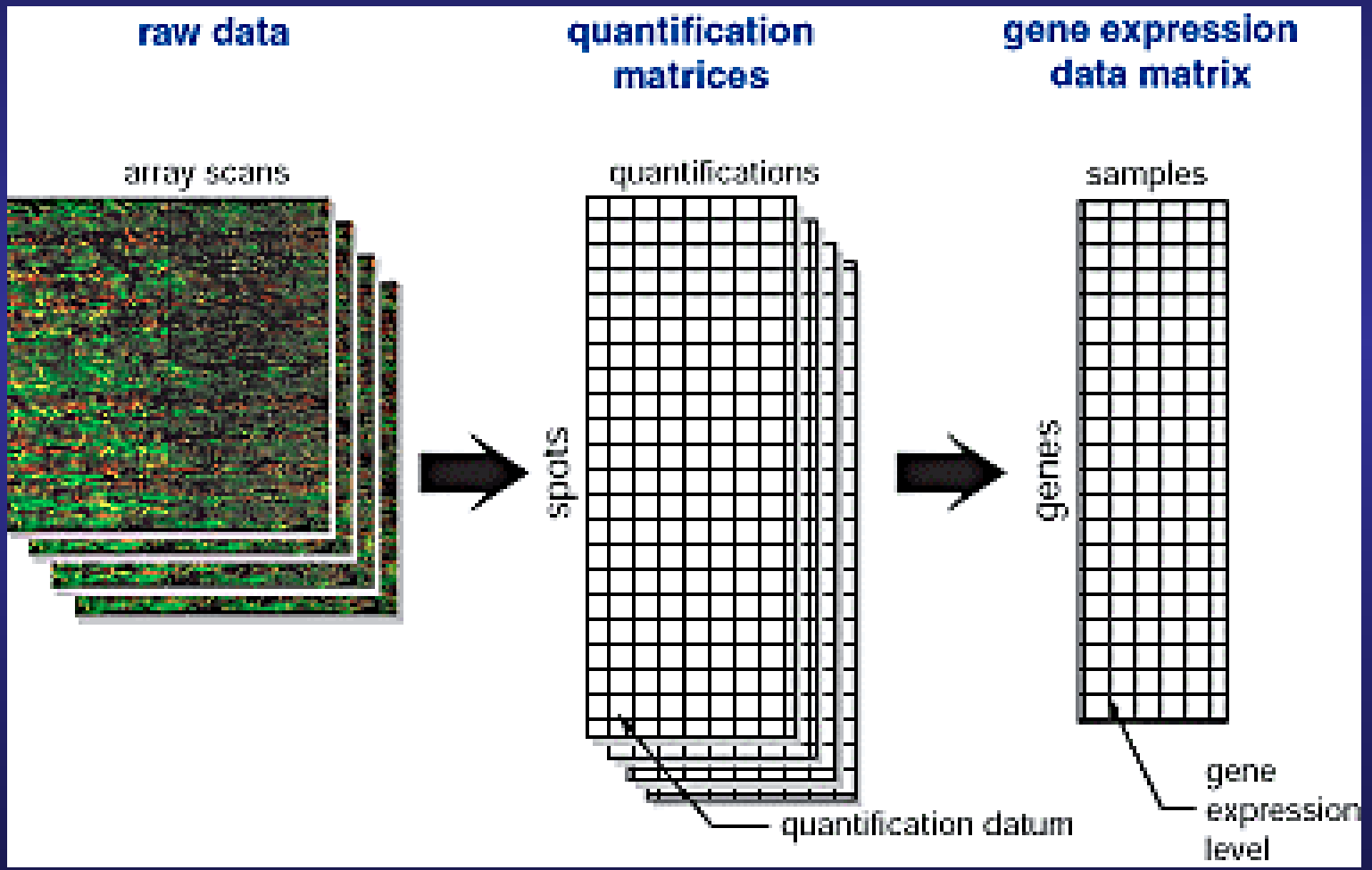
Outline

- Introduction
  - Microarray Informatics
  - Our Data and Objectives
- Our Data Structures & Algorithm
- Demonstration & Results
- Future Work

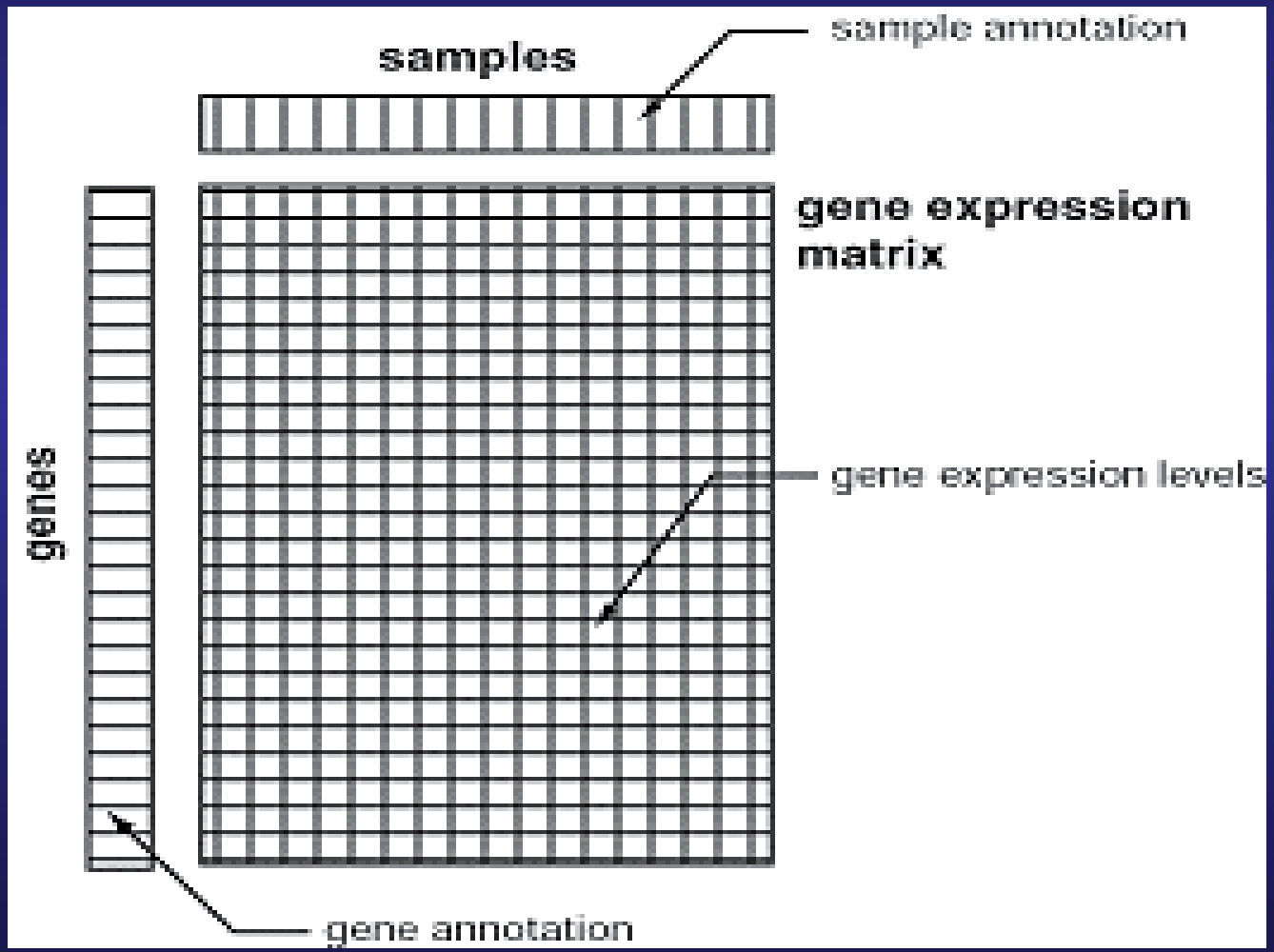
Microarray Informatics - a Schematic View



Three Levels of Data Processing



Conceptual Expression Matrix



Some of the Problems

- Lack of standards and units
- Slow algorithms (and slow programs)
- Algorithms sensible to noises

Possible (and partial) solutions

- Lack of standards and units
- Algorithms sensible to noises
- Slow algorithms (and slow programs)

- Conceptual gene expression matrix
- Preprocess data to filter noises
- New algorithms

The Objectives

- Implement conceptual GEM
- Provide a flexible platform for various categorizing strategies
- Try new pattern discovery algorithms

The Data Used in the Project

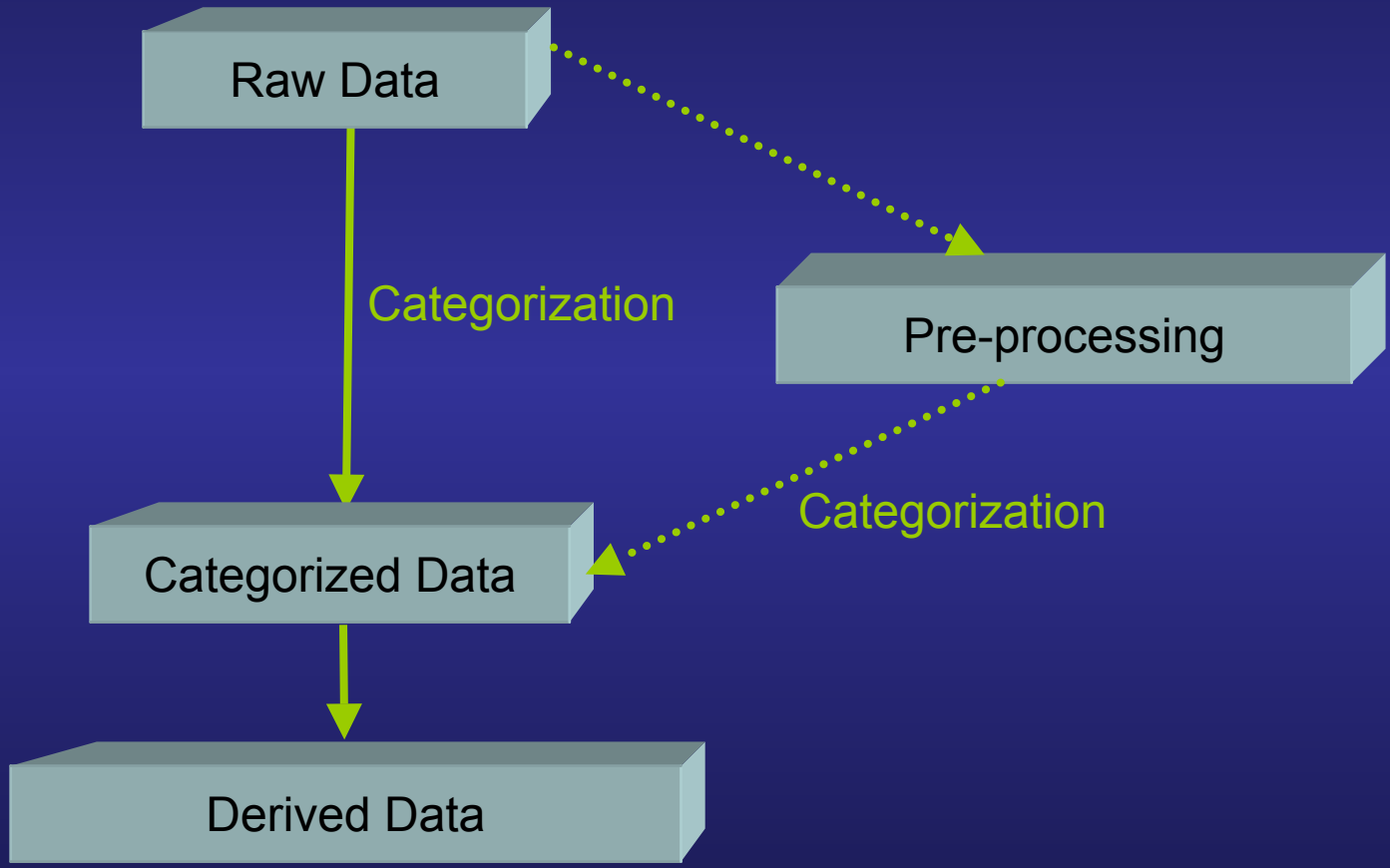
Time series data
(Claridge-Chang et al, 2001)

- Published data close to GEM
- Patterns related to genes only
- Data structures implementing rises and falls
- Clustering results of only 159 genes

Our Algorithm

- Data Structures
- Similarity Function
- The Algorithm

The Data Transformation



An Example of Raw Data

GC022	S021	S089	S236	S705	S887
G0055	564	1000	136	210	351
G0101	550	956	129	209	259
G0383	150	352	30	69	89
G3298	207	92	789	510	260

Categorizing

- Choosing Ranges
  - Even distribution by values
  - Customized distribution
- Assigning Categories
  - Integers

An Example of Categorized Data

Raw Data:

GC022	S021	S089	S236	S705	S887
G0055	564	1000	136	210	351
G0101	550	956	129	209	259
G0383	150	352	30	69	89
G3298	207	92	789	510	260

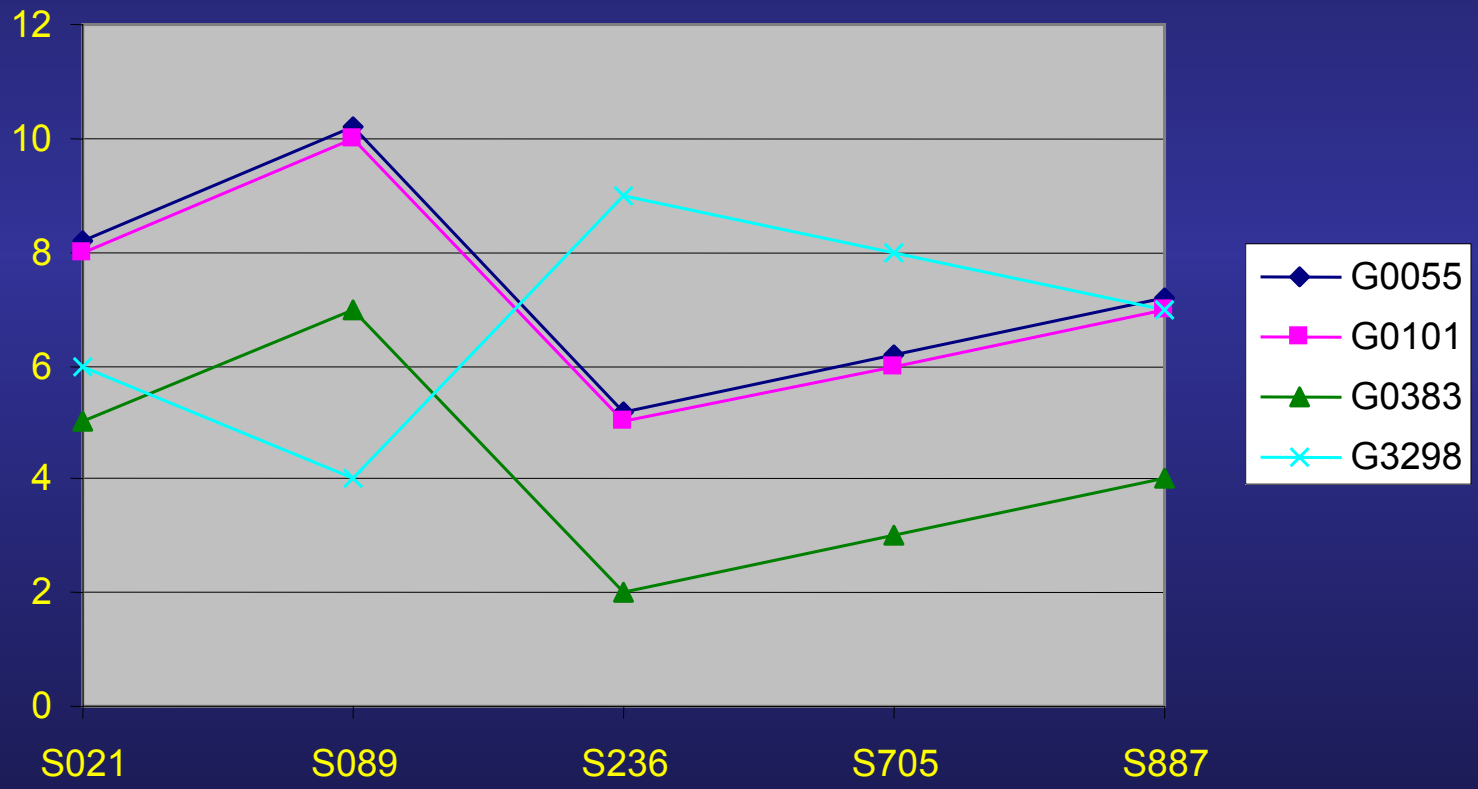
Categorized Data: (for example, 400 – 600: 8; 601 – 800: 9, etc)

GC022	S021	S089	S236	S705	S887
G0055	8	10	5	6	7
G0101	8	10	5	6	7
G0383	5	7	2	3	4
G3298	6	4	9	8	7

Derived Data

- Relative difference
  - r[i] - r[i-1]
- Relative ratio
  - (r[i] - r[i-1])/r[i-1]

Patterns to Be Discovered



The Similarity Function corr(row i, c)

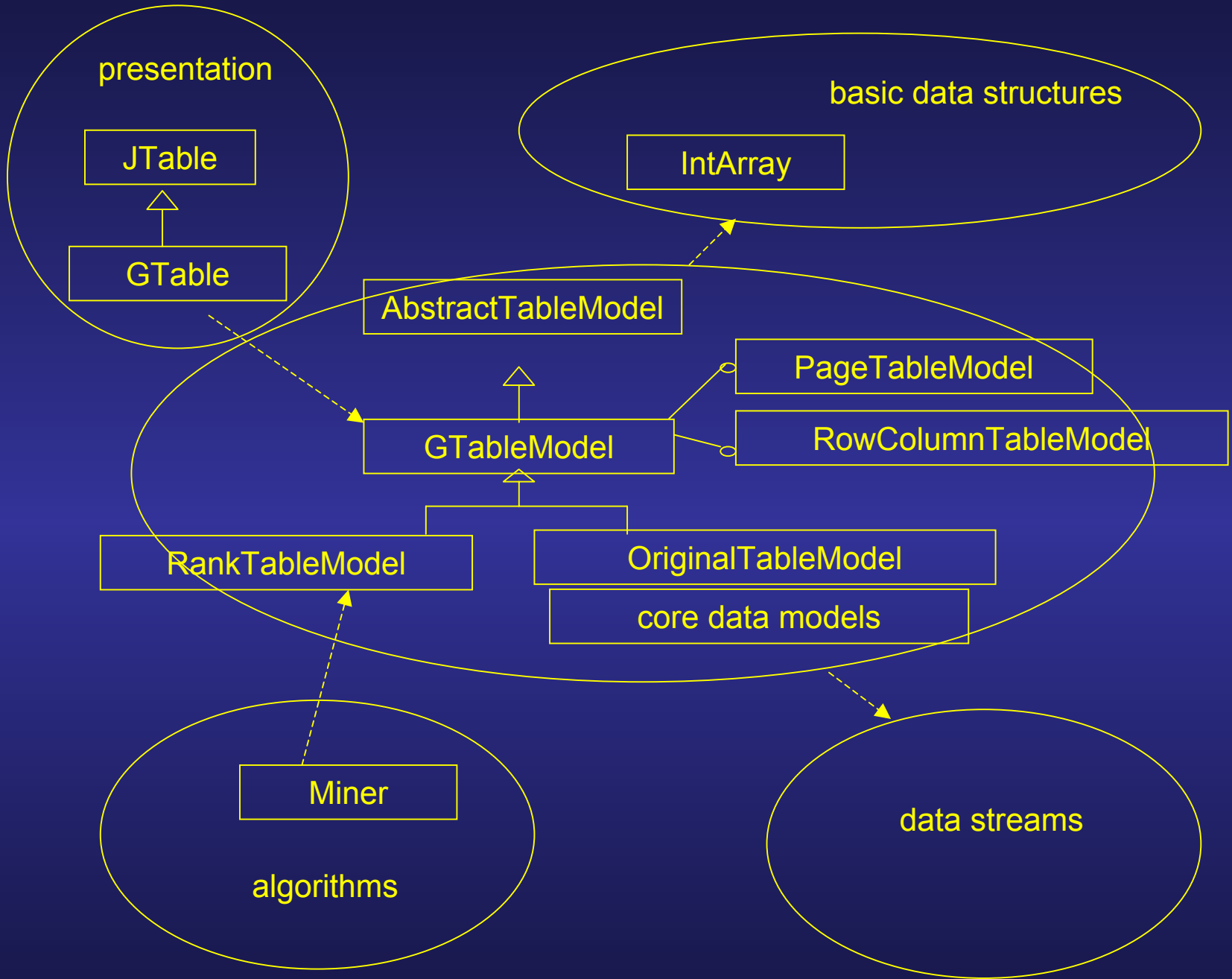
- Similarity functions:
  - Euclidean distance; Mahalanobis distance; Manhattan metric; Minkowski metric; Canberra metric; one minus correlation; Pearson Correlation
- Pearson Correlation m:
  - -1 <= m <= 1
- Vector of Cluster c:
  - The average of the coordinates of the vectors (rows) in the cluster c

The Algorithm

So the algorithm looks like this: Let **N** be the total number of rows, M be the total number of columns, **A** be the ArrayList of clusters, and T be a threshold.

```
1.  Put row i = 1 into A
2.  For i = 2 to N Do
3.        For each cluster c in A
4.            if |corr(row i, c)| > T
5.                put row i into cluster c
6.            else
7.                put row i into a new cluster d
8.                add d to A
9.  Output the A
```

Demonstration


presentation
JTable
GTable
basic data structures
IntArray
AbstractTableModel
PageTableModel
GTableModel
RowColumnTableModel
RankTableModel
OriginalTableModel
core data models
Miner
algorithms
data streams

Results

Results

Cluster #16

Cluster rise and fall pattern:	0.0	11.4	-35.8	41.0	-21.5	7.6
+ AFFX-DapX-3_atrRNA_3_at	23	24	24	24	24	24
- AFFX-BioDn-5_atRNA_3_at	70	46	51	51	58	64
- 155152_atn-5_atRNA_3_at	30	27	30	29	28	28
+ 155127_atn-5_atRNA_3_at	25	26	25	26	26	26

Cluster #17

Cluster rise and fall pattern:	0.0	-35.5	94.2	-92.0	61.0	-29.7
+ AFFX-CreX-5_attRNA_3_at	411	224	301	271	323	380
+ AFFX-CreX-3_attRNA_3_at	512	288	358	298	393	483
- 155141_at-3_attRNA_3_at	27	29	27	28	27	26

Future Work

- Fully functional interface
- Implement additional categorizing and ranking strategies
- Add preprocessing functions
- Design and implement new algorithms