

# Lab 4.1

## From Database to Data mining

***Sohrab Shah***

UBC Bioinformatics Centre

sohrab@bioinformatics.ubc.ca

<http://bioinformatics.ubc.ca/people/sohrab>

# Lab4.1 – Goals

- Load microarray data from a MySQL database into a data structure in memory
- Implement a k-means algorithm to cluster the data into 2 clusters
- Address inherent problems with k-means

# Introduction to the data – *Science* 286:531-537. (1999).

Golub\_et\_al\_1999.pdf (application/pdf Object) - Netscape

File Edit View Go Bookmarks Tools Window Help

200%

## Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring

T. R. Golub,<sup>1,2\*</sup> D. K. Slonim,<sup>1†</sup> P. Tamayo,<sup>1</sup> C. Huard,<sup>1</sup>  
M. Gaasenbeek,<sup>1</sup> J. P. Mesirov,<sup>1</sup> H. Coller,<sup>1</sup> M. L. Loh,<sup>2</sup>  
J. R. Downing,<sup>3</sup> M. A. Caligiuri,<sup>4</sup> C. D. Bloomfield,<sup>4</sup>  
E. S. Lander<sup>1,5\*</sup>

Although cancer classification has improved over the past 30 years, there has been no general approach for identifying new cancer classes (class discovery) or for assigning tumors to known classes (class prediction). Here, a generic approach to cancer classification based on gene expression monitoring by DNA microarrays is described and applied to human acute leukemias as a test case. A class discovery procedure automatically discovered the distinction between acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL) without previous knowledge of these classes. An automatically derived class predictor was able to determine the class of new leukemia cases. The results demonstrate the feasibility of cancer classification based solely on gene expression monitoring and suggest a general strategy for discovering and predicting cancer classes for other types of cancer, independent of previous biological knowledge

acute leukemia has been associated with specific chromosomal translocations—for example, the Philadelphia chromosome translocation occurs in about 25% of patients with ALL, whereas the t(8;21) translocation occurs in about 15% of patients with APL. Although the distinction between AML and ALL has been well established, the test is currently subjective. Diagnosis of APL is currently based on morphological diagnosis. Rather, diagnosis of APL involves an experienced hematologist's interpretation of the morphology, immunophenotype, and cytogenetic analysis. APL is a rare, highly specialized disease, and diagnosis is usually accurate, but the test remains imperfect and subjective.

Distinguishing APL from other acute leukemias for successful treatment requires the use of specific agents for ALL germline toxicity, such as vincristine, methotrexate, and cytarabine, whereas most of the toxicity of the backbone of daunorubicin and cytarabine is due to the toxicity of the backbone. Although remission rates for ALL therapy for APL are high, the rates are markedly different and the toxicities are more severe.

1 of 7 8.5 x 11 in Stopped

# Introduction to the data

- Golub et al Science, 1999
  - [http://www.broad.mit.edu/cgi-bin/cancer/publications/pub\\_paper.cgi?mode=view&paper\\_id=43](http://www.broad.mit.edu/cgi-bin/cancer/publications/pub_paper.cgi?mode=view&paper_id=43)
- 6817 genes tested in leukemia patients
- 2 known classes of leukemia for training data
  - ALL (acute lymphoblastic leukemia)
    - 19 samples
  - AML (acute myeloid leukemia)
    - 11 samples
- Training data are 'labeled' with these classes

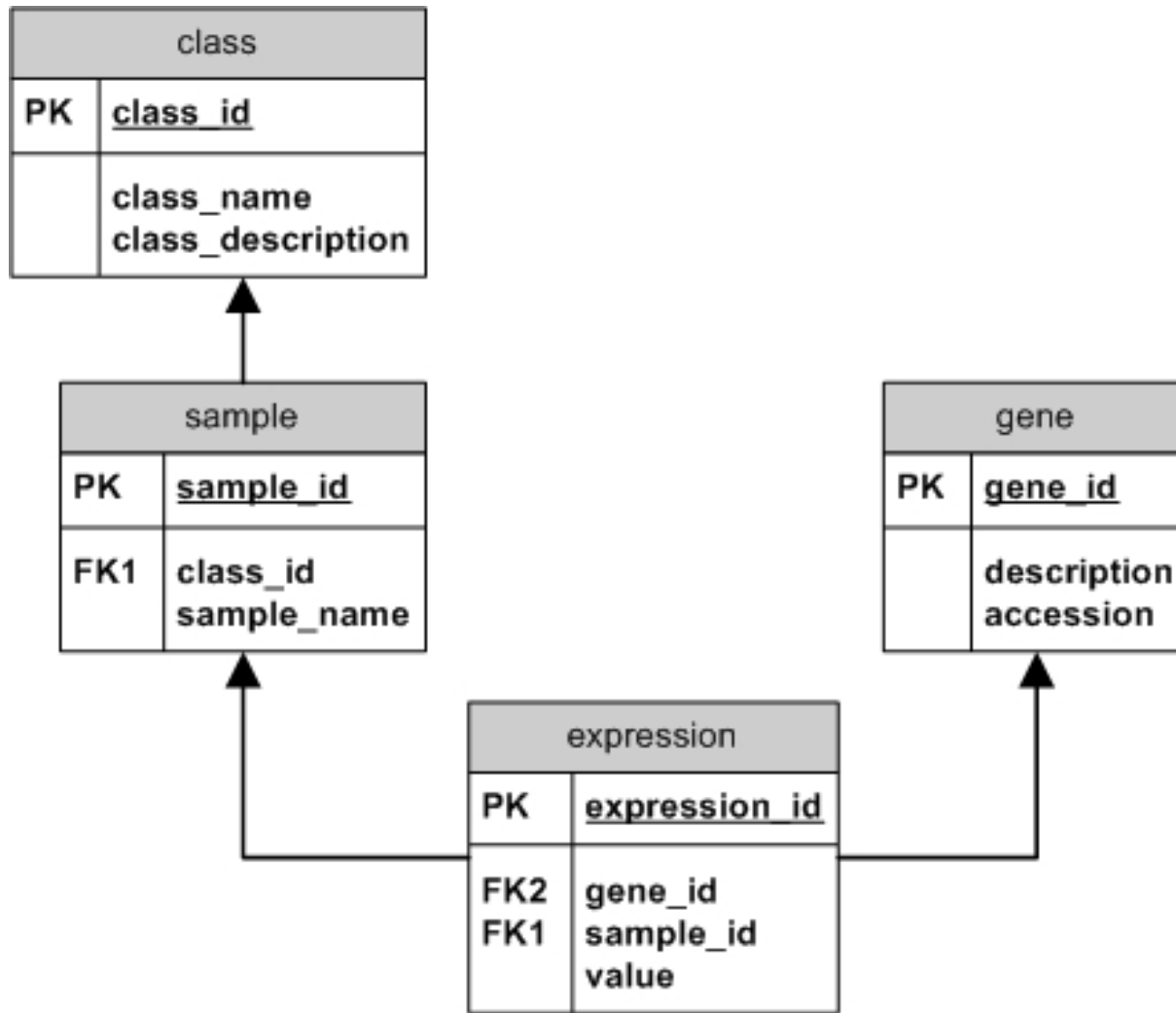
# Scientific question

- Can molecular profiles of the ~7000 genes be used to cluster the patients into 2 distinct 'groups' or classes?

# Introduction to the database

- All data are pre-loaded into a MySQL database
- 4 tables to model the data
  - class, sample, gene, expression

# Database relations



# Data Structure

- GolubSample class
  - Holds the expression data for all genes for 1 sample
  - Has a String sampleName
  - Has a String cancerClass
  - Has a HashMap geneExpressionMap
    - Keys = gene\_id's from the gene table
    - Values = value from expression table

# Database API

- GolubDb.java
  - Methods to interact with the database
    - ArrayList getAllSampleIds()
    - String sampleId2SampleName()
    - String sampleId2ClassName()
    - GolubSample sampleId2GolubSample(int sampleId)

# KMeans.java

- 'Global' variables:

```
private static int ITERATIONS = 10;
private static GolubDb golubDb;
private static HashMap sampleData;
private static HashMap clusterAssignments;
private static HashMap distanceToAssignedCluster;
private static GolubSample mean1;
private static GolubSample mean2;
private static GolubSample std1;
private static GolubSample std2;
private static ArrayList cluster1;
private static ArrayList cluster2;
```

# Exercises

## Implement

a) `KMeans.calculateMean(ArrayList cluster,  
Collection keys)`

- Take the mean of the expression values for each gene in the cluster
- Use the keys to iterate through the geneExpressionMap HashMap

b) `KMeans.calculateStandardDeviation(ArrayList cluster,  
Collection keys)`

- Take the standard deviation of the expression values for each gene in the cluster
- Use the keys to iterate through the geneExpressionMap HashMap
- $\text{Sum}(x_i - u_i)^2 / (N - 1)$

# Exercises

## Implement

c) `GolubSample.normalise(GolubSample mean,  
GolubSample standardDeviation)`

- Normalise the data in 'this' by subtracting the mean and dividing by the standard deviation

d) `GolubSample.computeDistance(GolubSample golubSample)`

- Compute the Euclidean distance from 'this' to the parameter golubSample

# Run the program

1. Use random intialisation of the centroids
2. Set the centroids manually as arguments to the program
3. Observe the differences
  - What is different and why?
4. Try different numbers of iterations
  - How many iterations are needed to converge?
  - Why is this a good/bad thing?

# Code location

- [http://www.bioinformatics.ca/dtt2004/lab4\\_1](http://www.bioinformatics.ca/dtt2004/lab4_1)