中央研究院

Clustering and Characterizing Data

Statistics in GeneSpring 7

吳漢銘

2005年9月21日

中央研究院 統計科學研究所
Institute of Statistical Science, Academia Sinica

hmwu@stat.sinica.edu.tw
http://www.sinica.edu.tw/~hmwu

Outlines

- Principal Components Analysis (PCA)
- Similarity Measures
- Clustering
  - K-means
  - Gene Tree
  - Condition Tree
  - Self-Organizing Map (SOM)
  - QT Clustering
- Classification (Class Prediction Analysis)
  - K-Nearest Neighbors (KNN)
  - Support Vector Machines (SVM)

Chapter 15

GeneSpring Manual 7.2

Principal Component Analysis

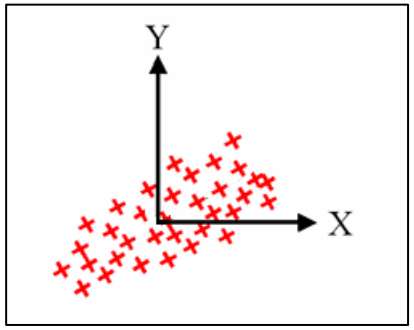
GeneSpring Tutorials:
Principal Components Analysis
http://www.chem.agilent.com/cag/bsp/SiG/Downloads/Tutorial/principal_component_analysis.viewlet/principal_component_analysis_viewlet_swf.html

Analysis Guides: Principal Components Analysis
http://www.chem.agilent.com/cag/bsp/SiG/Downloads/pdf/pca.pdf

Principal Component Analysis

Microarray Data Matrix

MA Table	exp01	exp02	exp03	exp04	exp05	exp•••	exp p
gene001	-0.48	-0.42	0.87	0.92	0.67		-0.35
gene002	-0.39	-0.58	1.08	1.21	0.52		-0.58
gene003	0.87	0.25	-0.17	0.18	-0.13		-0.13
gene004	1.57	1.03	1.22	0.31	0.16		-1.02
gene005	-1.15	-0.86	1.21	1.62	1.12		-0.44
gene006	0.04	-0.12	0.31	0.16	0.17		0.08
gene007	2.95	0.45	-0.40	-0.66	-0.59		-0.76
gene008	-1.22	-0.74	1.34	1.50	0.63		-0.55
gene009	-0.73	-1.06	-0.79	-0.02	0.16		0.03
gene010	-0.58	-0.40	0.13	0.58	-0.09		-0.45
gene011	-0.50	-0.42	0.66	1.05	0.68		0.01
gene012	-0.86	-0.29	0.42	0.46	0.30		-0.63
gene013	-0.16	0.29	0.17	-0.28	-0.02		-0.04
gene014	-0.36	-0.03	-0.03	-0.08	-0.23		-0.21
gene015	-0.72	-0.85	0.54	1.04	0.84		-0.64
gene016	-0.78	-0.52	0.26	0.20	0.48		0.27
gene017	0.60	-0.55	0.41	0.45	0.18		-1.02
gene018	-0.20	-0.67	0.13	0.10	0.38		0.05
gene019	-2.29	-0.64	0.77	1.60	0.53		-0.38
gene020	-1.46	-0.76	1.08	1.50	0.74		-0.70
gene021	-0.57	0.42	1.03	1.35	0.64		-0.40
gene022	-0.11	0.13	0.41	0.60	0.23		0.19
gene•••							
gene n	-1.79	0.94	2.13	1.75	0.23		-0.66



- Make a visual inspection of the relationship between genes or conditions in a multi-dimensional matrix.
- PCA is a method that reduces data dimensionality by performing a covariance analysis between factors.

- In GeneSpring, PCA can be performed based on gene expression profiles, or based on samples or conditions.

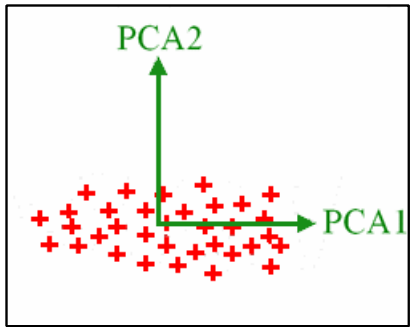
When to use PCA?

- As an exploratory tool to uncover unknown trends in the data.
- PCA on genes provide a way to identify predominant gene expression patterns.
- PCA on conditions explore correlations between samples or conditions.
- PCA is to ‘summarize’ the data, it is not considered a clustering tool.

PCA

(Pearson 1901; Hotelling 1933; Jolliffe 2002)

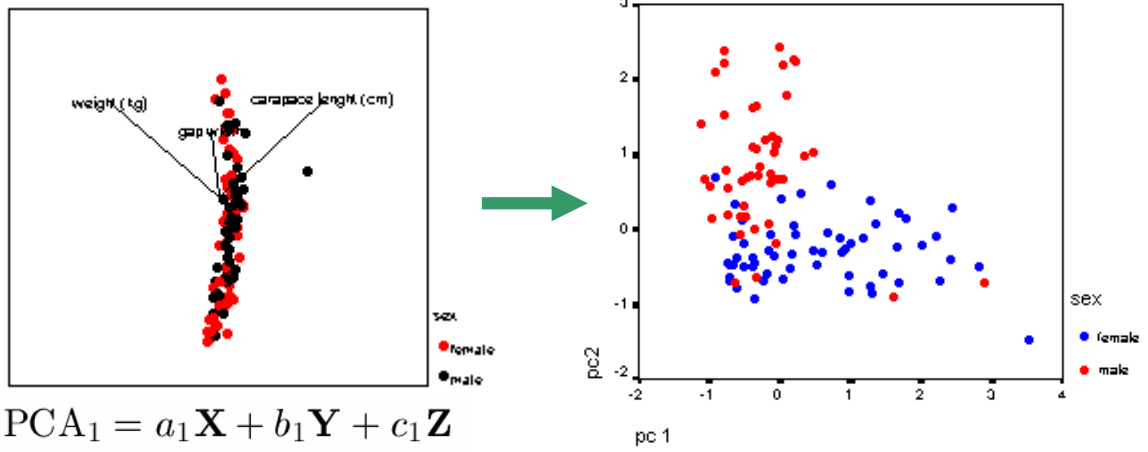
PCA is a method that reduces data dimensionality by finding the new variables (major axes, principal components).



$$\mathrm{PCA}_1 = a_1\mathbf{X} + b_1\mathbf{Y}$$

$$\mathrm{PCA}_2 = a_2\mathbf{X} + b_2\mathbf{Y}$$

Image source: 61BL4165 Multivariate Statistics, Department of Biological Sciences, Manchester Metropolitan University



$$\mathrm{PCA}_1 = a_1\mathbf{X} + b_1\mathbf{Y} + c_1\mathbf{Z}$$

$$\mathrm{PCA}_2 = a_2\mathbf{X} + b_2\mathbf{Y} + c_2\mathbf{Z}$$

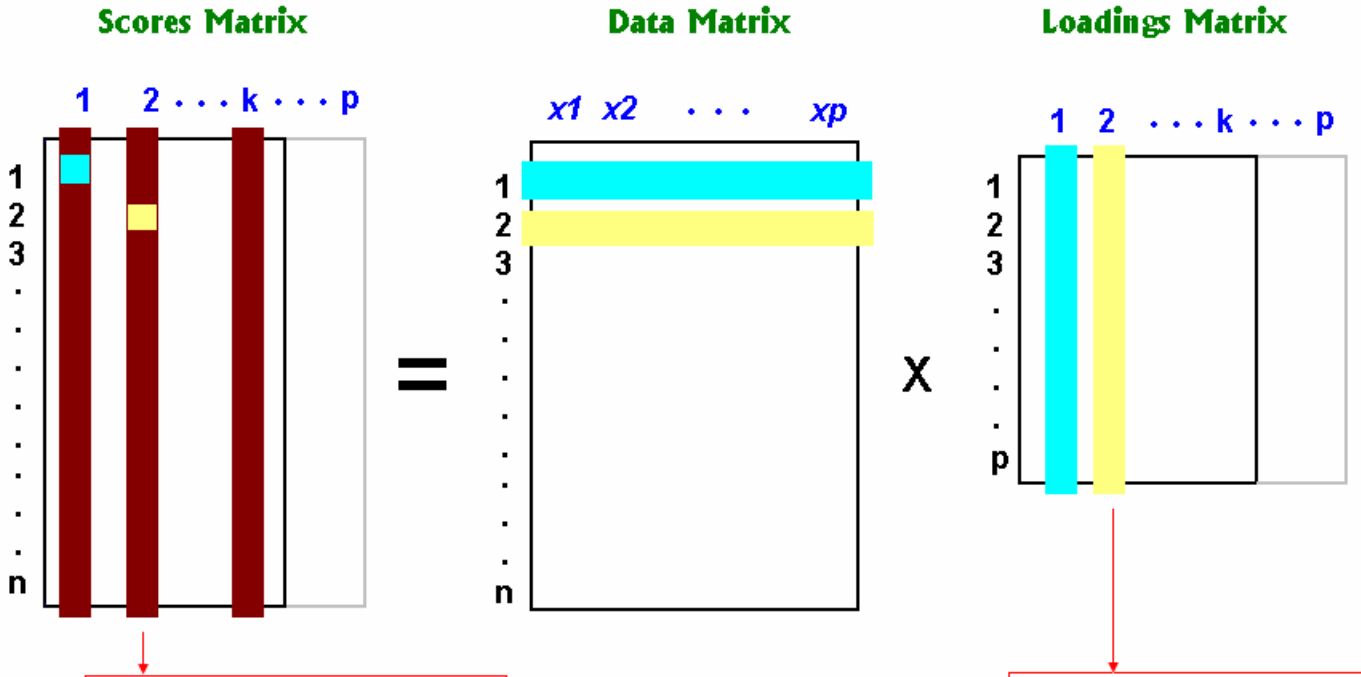
Amongst all possible projections, PCA finds the projections so that the maximum amount of information, measured in terms of variability, is retained in the smallest number of dimensions.

$$\mathrm{PCA}_1 = a_{11}\mathbf{X}_1 + a_{12}\mathbf{X}_2 + \cdots + a_{1p}\mathbf{X}_p$$

$$\mathrm{PCA}_2 = a_{21}\mathbf{X}_1 + a_{22}\mathbf{X}_2 + \cdots + a_{2p}\mathbf{X}_p$$

PCA: Loadings and Scores

$$\mathbf{U} = \mathbf{X}\,\mathbf{V}$$



The ith principal component of $\mathbf{X}$ is $\mathbf{X}\,\mathbf{v}_i$, where $\mathbf{v}_i$ is the ith normalized eigenvector of $\Sigma_{\mathbf{x}}$ corresponding to the ith largest eigenvaules.

Eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$

$$\text{proportion} = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{p} \lambda_i}$$

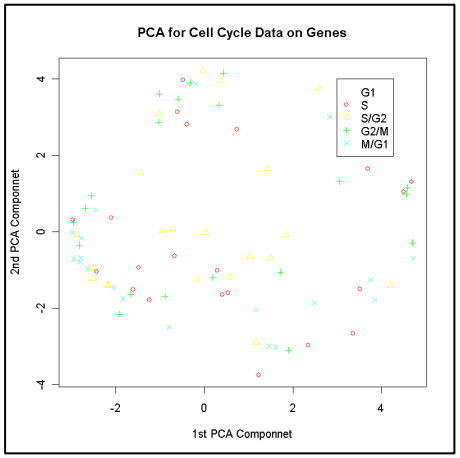
- The eigenvalue corresponding to an eigenvector represents the amount of variability explained by that eigenvector.

Interpretation of the PCA Results

Microarray Data Matrix

MA Table	exp01	exp02	exp03	exp04	exp05	exp•••	exp p
gene001	-0.48	-0.42	0.87	0.92	0.67		-0.35
gene002	-0.39	-0.58	1.08	1.21	0.52		-0.58
gene003	0.87	0.25	-0.17	0.18	-0.13		-0.13
gene004	1.57	1.03	1.22	0.31	0.16		-1.02
gene005	-1.15	-0.86	1.21	1.62	1.12		-0.44
gene006	0.04	-0.12	0.31	0.16	0.17		0.08
gene007	2.95	0.45	-0.40	-0.66	-0.59		-0.76
gene008	-1.22	-0.74	1.34	1.50	0.63		-0.55
gene009	-0.73	-1.06	-0.79	-0.02	0.16		0.03
gene010	-0.58	-0.40	0.13	0.58	-0.09		-0.45
gene011	-0.50	-0.42	0.66	1.05	0.68		0.01
gene012	-0.86	-0.29	0.42	0.46	0.30		-0.63
gene013	-0.16	0.29	0.17	-0.28	-0.02		-0.04
gene014	-0.36	-0.03	-0.03	-0.08	-0.23		-0.21
gene015	-0.72	-0.85	0.54	1.04	0.84		-0.64
gene016	-0.78	-0.52	0.26	0.20	0.48		0.27
gene017	0.60	-0.55	0.41	0.45	0.18		-1.02
gene018	-0.20	-0.67	0.13	0.10	0.38		0.05
gene019	-2.29	-0.64	0.77	1.60	0.53		-0.38
gene020	-1.46	-0.76	1.08	1.50	0.74		-0.70
gene021	-0.57	0.42	1.03	1.35	0.64		-0.40
gene022	-0.11	0.13	0.41	0.60	0.23		0.19
gene•••							
gene n	-1.79	0.94	2.13	1.75	0.23		-0.66

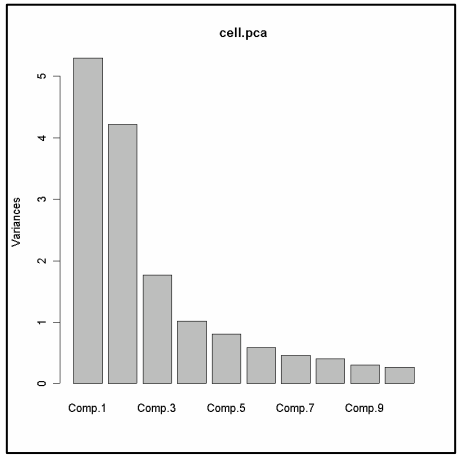
PCA on Conditions



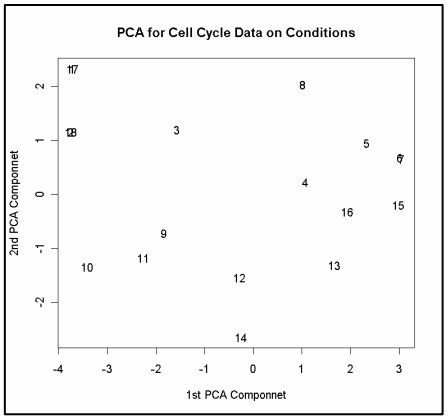
Scree plot

Eigenvalues

Variances



PCA on Genes



Loadings Matrix

Loadings:	Comp.1	Comp.2	Comp.3	Comp.4
alpha14	-0.283	-0.21	0.283	0.136
alpha21	-0.374	0.211	-0.135	-0.16
alpha28	-0.26	0.298	0.161	-0.168
alpha35	-0.102	0.372	0.165	-0.321
alpha42	0.161	0.355	0.2	-0.317
alpha49	0.287	0.167	0.116	-0.515
alpha56	0.35	0.172	-0.274	-0.115
alpha63	0.251	-0.258	-0.275	-0.37
alpha70	-0.372	-0.217	-0.382	-0.159
alpha77	-0.253	-0.221	-0.321	-0.32
alpha84	-0.249	-0.437	-0.309	-0.256
alpha91	-0.115	0.279	-0.436	0.114
alpha98	0.36	-0.284	0.186	-0.138
alpha105	0.16	0.257	-0.283	-0.125
alpha112	0.347	0.319	-0.178	-0.276
alpha119	0.348	-0.164	-0.201	0.11

Loadings Plot


```
> summary(cell.pca)
Importance of components:
                          Comp.1     Comp.2     Comp.3     Comp.4     Comp.5       Comp.15
Standard deviation     2.3012110 2.0542795 1.3300507 1.00895544 0.90053289    0.308577283
Proportion of Variance 0.3309732 0.2637540 0.1105647 0.06362444 0.05068497 ••• 0.005951246
Cumulative Proportion  0.3309732 0.5947272 0.7052919 0.76891637 0.81960134    1.000000000
```

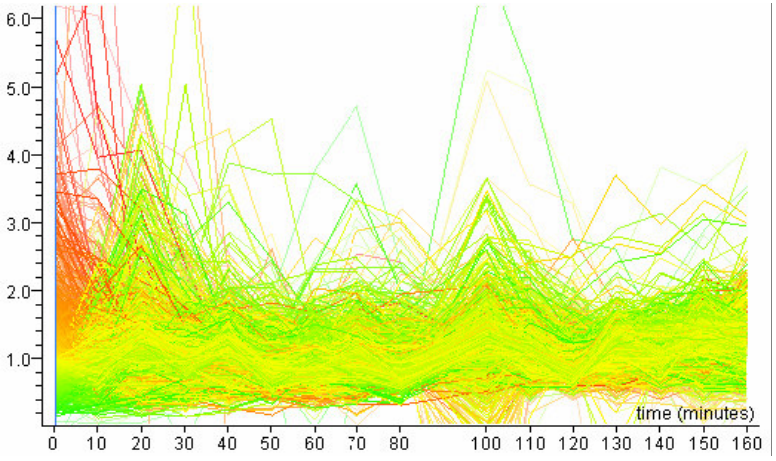
PCA on Genes and Conditions

PCA on Genes

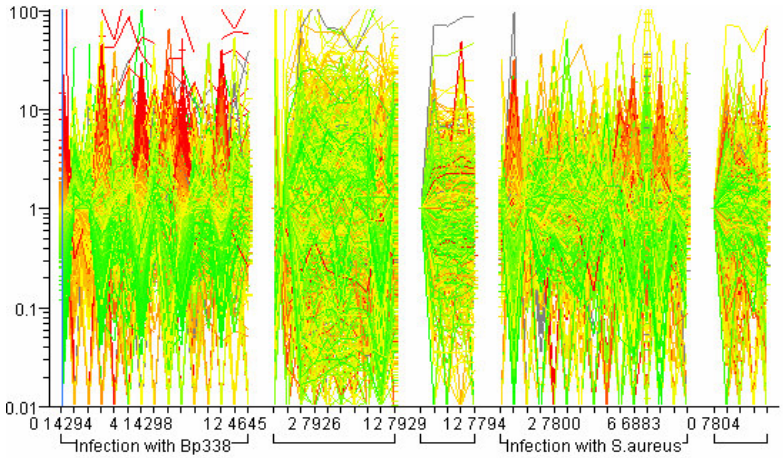
- If you have a serial type or dose experiment with one main parameter, such as time or concentration, you will more likely be interested in finding principal gene expression profiles.

PCA on Conditions

- your experiment may consist of only a few different conditions but with a large quantity of replicates.
- You may primarily be interested in identifying prevalent expression profiles among samples regardless of individual genes' expression patterns.
- PCA on conditions will identify the key sample profiles.

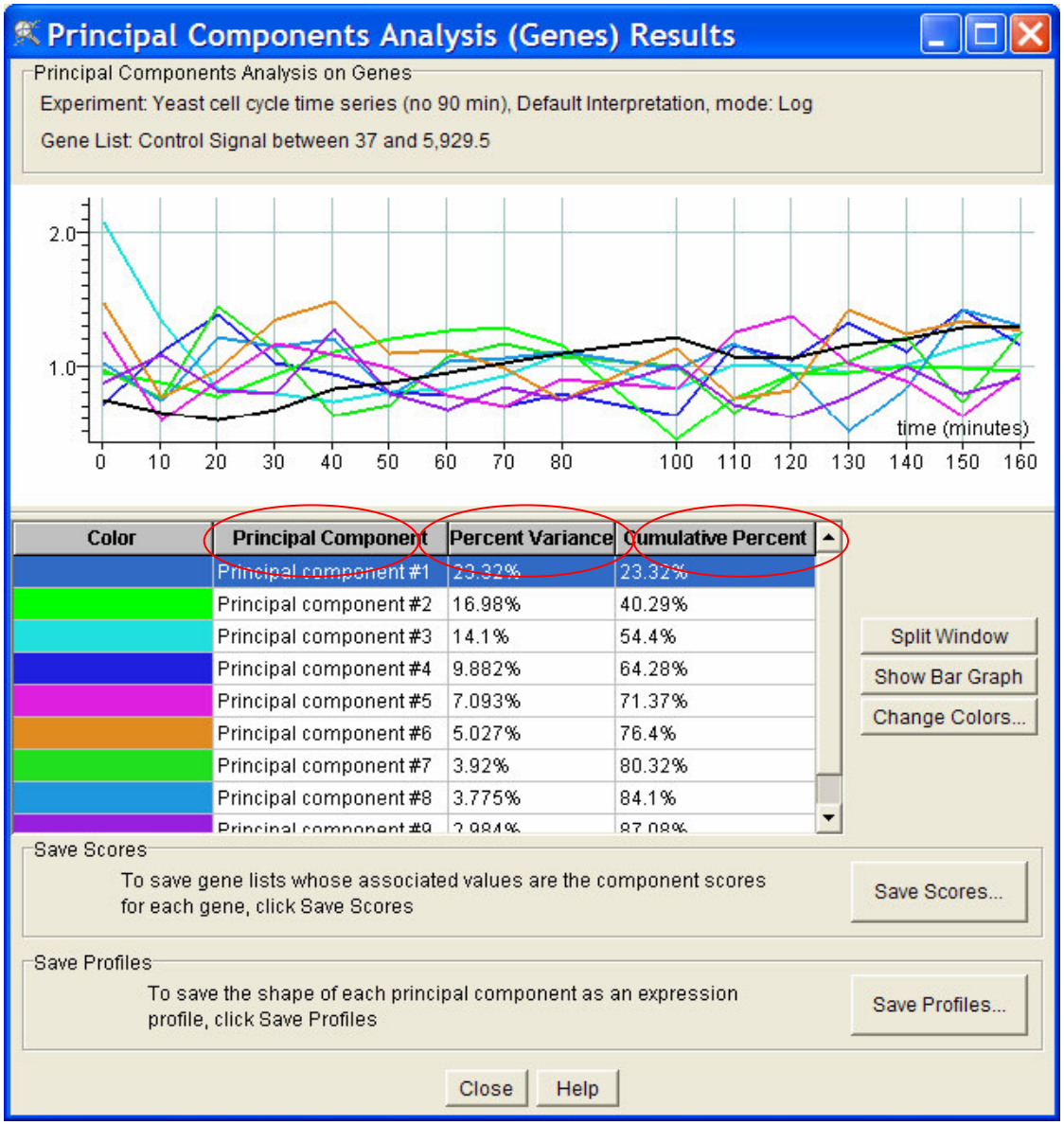


Source: Analysis Guides: PCA

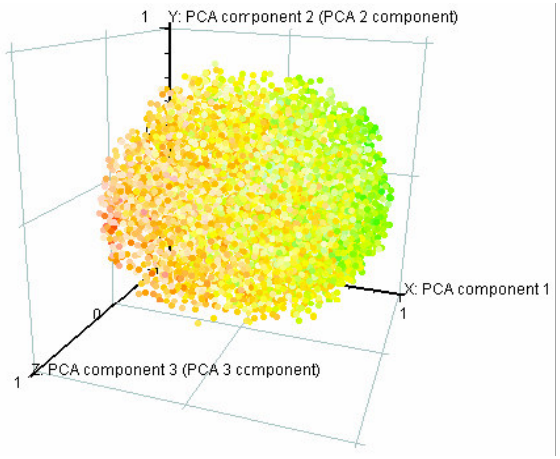


Source: Analysis Guides: PCA

Viewing PCA on Genes Results

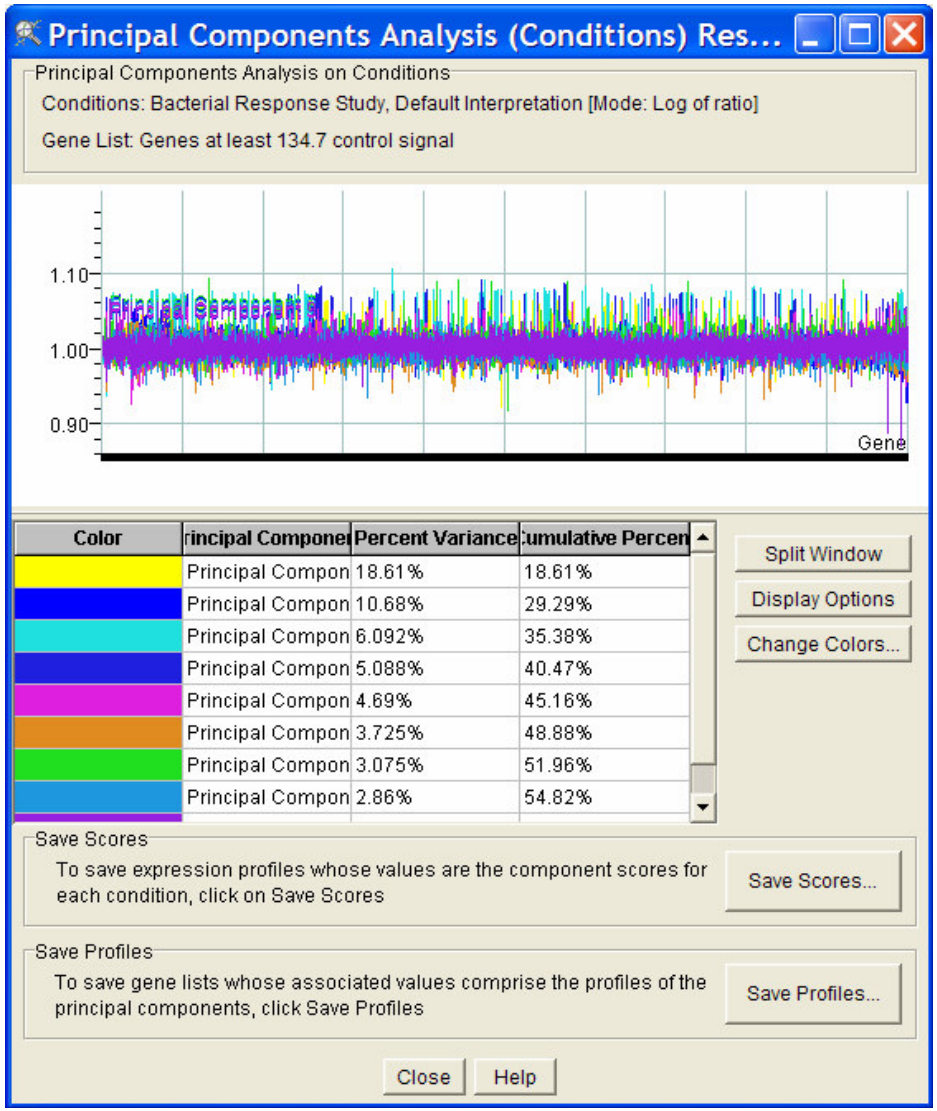


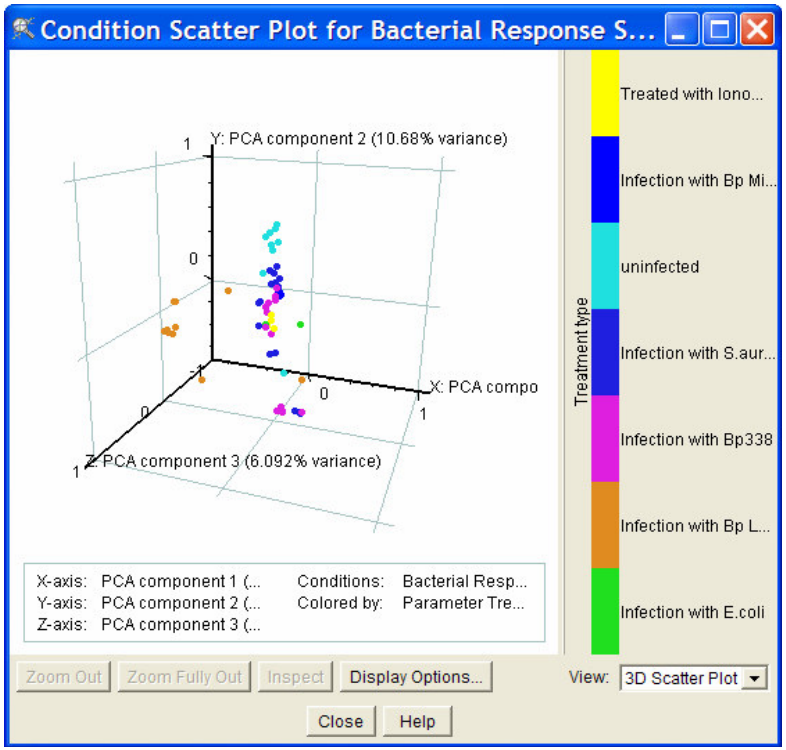
PCA on genes will find relevant components, or patterns, across gene expression data.



Source: Analysis Guides: PCA

Viewing PCA on Conditions Results

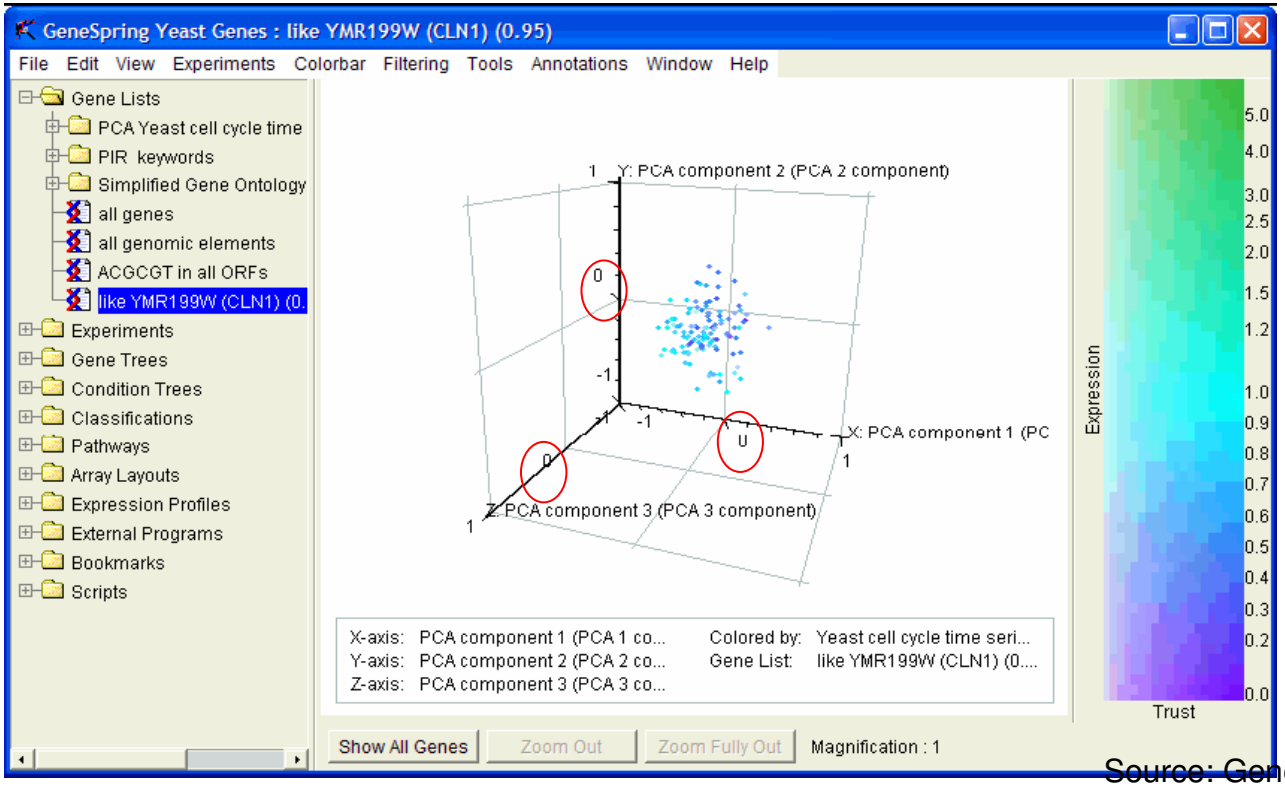




Source: Analysis Guides: PCA

Viewing Principal Component Loadings in a Scatter Plot

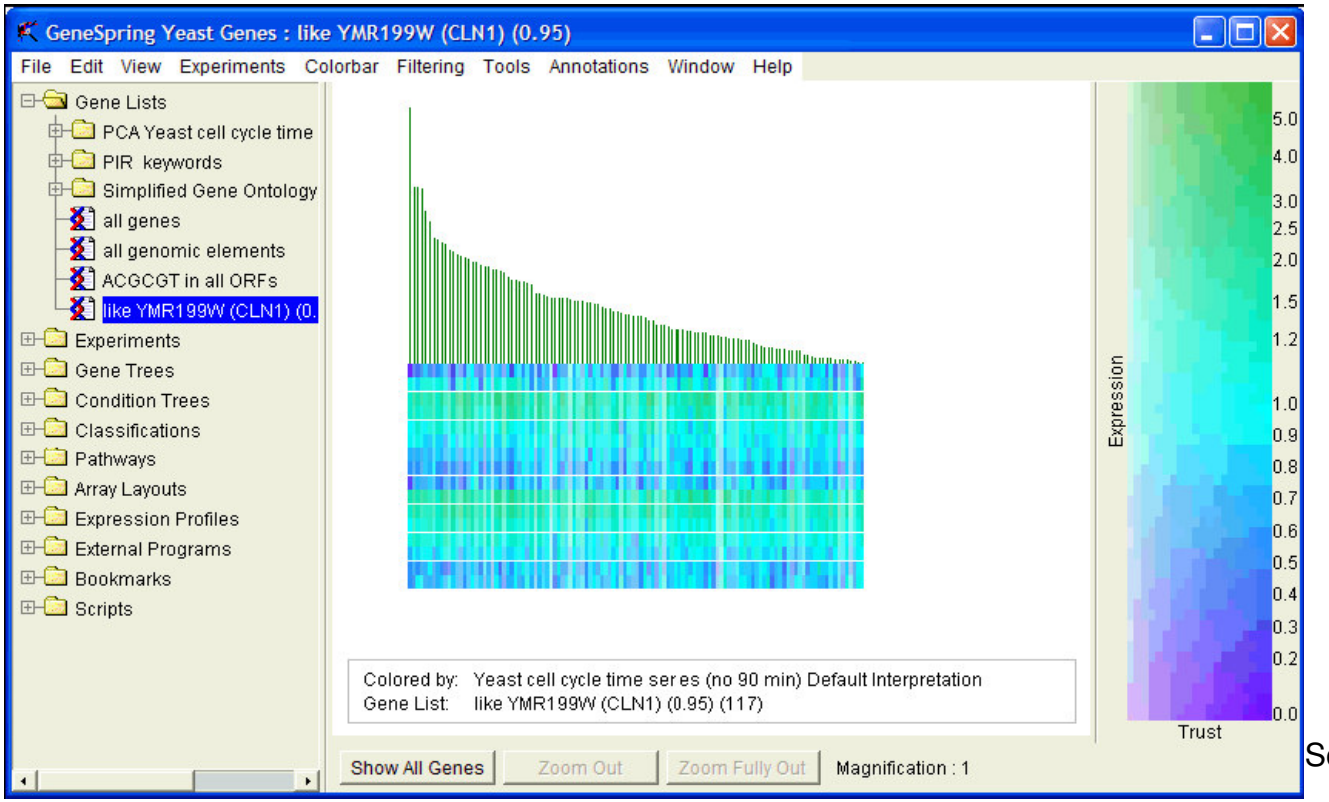
- Each point represents a single gene.
- Useful for selecting and making lists of genes that exhibit high levels of one or two principal components.
- Genes that exhibit high levels of the first principal component and low levels of the second principal component are displayed in the lower right corner of the plot.
- Genes exhibiting equal levels of the two components lie along the diagonal.



Source: GeneSpring Manual 7.2

Viewing Principal Components in an Ordered List (PCA on Genes)

- The best way to visualize the genes that exhibit the highest levels of an individual component is to use the Ordered List view
- Genes exhibiting the highest levels of the selected principal component are displayed on the left side of the Genome Browser and have the longest lines extending upward from them.
- Genes will be ordered according to their correlation to the principal component .



Source: GeneSpring Manual 7.2

Similarity Measure

GeneSpring Tutorials:
Gene Tree & Condition Tree (Hierarchical Clustering) View
http://www.chem.agilent.com/cag/bsp/SiG/Downloads/Tutorial/gene_tree_and_condition_tree.viewlet/gene_tree_and_condition_tree_viewlet_swf.html

Similarity Measures

Standard Correlation	Measures and **B** ar *Result* =
Smooth Correlation	Make a n consecut old value by a line **b**. *Result* =
Change Correlation	Make a n pair of ele connecte two value vector **B** *Result* =
Upregulated Correlation	Make a n pair of elements of **a**. Do this for each pair of elements that would be connected by a line in the graph window. The value created between

Pearson Correlation	Calculate the mean of all elements in vector **a**. Then subtract that value from each element in **a**. Call the resulting vector **A**. Do the same for **b** to make a vector **B**. *Result* = **A**.**B**/(|**A**||**B**|)
Distance	Distance is not a correlation at all, but a measurement of dissimilarity. Distance is the measurement of Euclidian distance between the expression profile for gene **A** (defined by its expression values for each point in N-dimensional space, where N is the number of conditions with data in your experiment) and the expression profile for gene **B**. *Result* = |**a**-**b**| divided by the square root of the number of conditions with data
Spearman Correlation	Order all the elements of vector **a**. Use this order to assign a rank to each element of **a**. Make a new vector **a'** where the i^{th} element in **a'** is the rank of a_i in **a**. Now make a vector **A** from **a'** in the same way as **A** was made from **a** in the Pearson Correlation. Similarly, make a vector **B** from **b**. *Result* = **A**.**B**/(|**A**||**B**|)

Spearman Confidence	Compute a value *r* of the spearman correlation as described above. *Result* =1-(probability you would get a value of *r* or higher by chance.)
Two-sided Spearman Confidence	Compute a value *r* of the spearman correlation as described above. *Result* =1-(probability you would get a value of |*r*| or higher, or -|r| or lower, by chance.)

Source:
Chapter 14, GeneSpring Manual 7.2

Distance and Similarity Measure

Cov	x1	x2	x3	x4	...	x p
x1	1.00	0.48	0.10	-0.10		-0.28
x2	0.48	1.00	0.41	0.22		-0.23
x3	0.10	0.41	1.00	0.36		-0.05
x4	-0.10	0.22	0.36	1.00		0.10
...						
x p	-0.28	-0.23	-0.05	0.10		1.00

Proximity Matrix

Pearson Correlation Coefficient

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Euclidean Distance

$$d_{xy} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

Data Matrix x y

Data	x1	x2	x3	x4	...	x p
subject01	-0.48	-0.42	0.87	0.92		-0.18
subject02	-0.39	-0.58	1.08	1.21		-0.33
subject03	0.87	0.25	-0.17	0.18		-0.44
subject04	1.57	1.03	1.22	0.31		-0.49
subject05	-1.15	-0.86	1.21	1.62		0.16
subject06	0.04	-0.12	0.31	0.16		-0.06
subject07	2.95	0.45	-0.40	-0.66		-0.38
subject08	-1.22	-0.74	1.34	1.50		0.29
subject09	-0.73	-1.06	-0.79	-0.02		0.44
subject10	-0.58	-0.40	0.13	0.58		0.02
subject11	-0.50	-0.42	0.66	1.05		0.06
subject12	-0.86	-0.29	0.42	0.46		0.10
subject13	-0.16	0.29	0.17	-0.28		-0.55
subject14	-0.36	-0.03	-0.03	-0.08		-0.25
subject15	-0.72	-0.85	0.54	1.04		0.24
subject16	-0.78	-0.52	0.26	0.20		0.48
subject17	0.60	-0.55	0.41	0.45		-0.66
⋮						
subject n	-2.29	-0.64	0.77	1.60		0.55
mean	0.07	-0.04	0.44	0.31	...	-0.21

$$x = (x_1, x_2, \cdots, x_n)$$

$$y = (y_1, y_2, \cdots, y_n)$$

- The standard transformation from a similarity matrix C to a distance matrix D is given by $d_{rs} = (c_{rr} - 2c_{rs} + c_{ss})^{1/2}$.
- (Eisen *et al.* 1998) $d_{rs} = 1 - c_{rs}$
- Other transformations (Chatfield and Collins 1980, Section 10.2)

Raw Data Matrix $\mathbf{X}$
Dispersion Matrix $\mathbf{S}_X^2 = \mathbf{X}^T\mathbf{X}$
Centered Data $\mathbf{C} = \mathbf{X} - \mu$
Covariance Matrix $\mathbf{\Sigma}_X = \mathbf{C}^T\mathbf{C}$
Scaled Data $\mathbf{Z} = \frac{\mathbf{X}-\mu}{\sigma}$
Correlation Matrix $\mathbf{R}_X = \mathbf{Z}^T\mathbf{Z}$

Similarity Measures

Dissimilarity/Similarity Measure for Quantitative Data

Kendall's tau

Two pairs of observation (x_i, y_i) and (x_j, y_j)

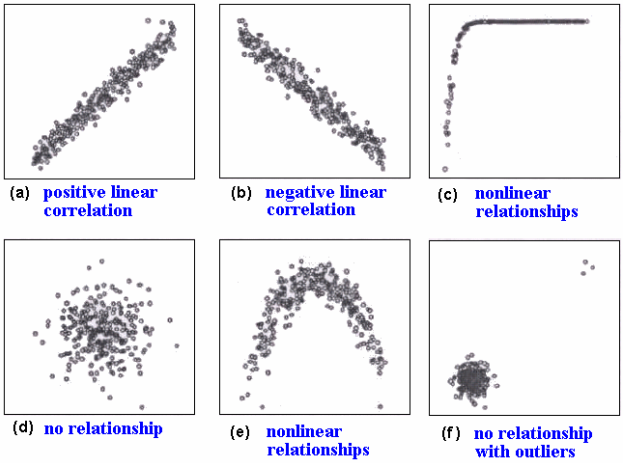
- C: concordant pair: $(x_j - x_i)(y_j - y_i) > 0$
- D: discordant pair: $(x_j - x_i)(y_j - y_i) < 0$
- tie:

E_y: extra y pair in x's: $(x_j - x_i) = 0$

E_x : extra x pair in y's: $(y_j - y_i) = 0$

$$\tau = \frac{C - D}{\sqrt{C + D - E_y}\sqrt{C + D - E_x}}$$

Similarity	Formula
Pearson correlation	$s(i,j) = \frac{\text{cov}(x_i, x_j)}{\sqrt{\text{var}(x_i)\,\text{var}(x_j)}}$
Spearman correlation (r_i is ranked x_i)	$s(i,j) = \frac{\text{cov}(r_i, r_j)}{\sqrt{\text{var}(r_i)\,\text{var}(r_j)}}$
Kendall's Tau	$s(i,j) = \frac{1}{\binom{p}{2}} \sum_{k \neq k'} sign\ [(x_{ik} - x_{ik'})(x_{jk} - x_{jk'})]$



(a) positive linear correlation (b) negative linear correlation (c) nonlinear relationships

(d) no relationship (e) nonlinear relationships (f) no relationship with outliers

- Pearson's rho measures the strength of a linear relationship [(a), (b)].
- Spearman's rho and Kendall's tau measure any monotonic relationship between two variables [(a), (b) ,(c)].
- If the relationship between the two variables is non-monotonic, all three correlation coefficients fail to detect the existence of a relationship [(e)].
- Both Spearman's rho and Kendall's tau are rank-based non-parametric measures of association between variable X and Y.
- The rank-based correlation coefficients are more robust against outliers.

Data	*Pearson's rho*	*Spearman's rho*	*Kendall's tau*
(a)	0.98	0.98	0.87
(b)	-0.98	-0.98	-0.87
(c)	0.50	0.99	0.98
(d)	-0.02	-0.03	-0.02
(e)	-0.06	-0.02	-0.02
(f)	0.68	0.00	0.00

Algorithm they use different logic for computing the correlation coefficient, they seldom lead to markedly different conclusions (Siegel and Castellan, 1988).

Clustering Analysis

GeneSpring Tutorials:
Gene Tree & Condition Tree (Hierarchical Clustering) View
http://www.chem.agilent.com/cag/bsp/SiG/Downloads/Tutorial/gene_tree_and_condition_tree.viewlet/gene_tree_and_condition_tree_viewlet_swf.html

Clustering Analysis

What is Clustering?

Cluster analysis is the organization of a collection of patterns into clusters based on similarity. The problem is to group a given collection of unlabeled patterns into meaningful clusters.

Clustering Methods in GeneSpring 7.2

- K-mesns
- Hierarchical Clustering: Gene Tree and Condition Tree
- Self-Organizing Map
- QT Clustering

Two important properties of a clustering definition:

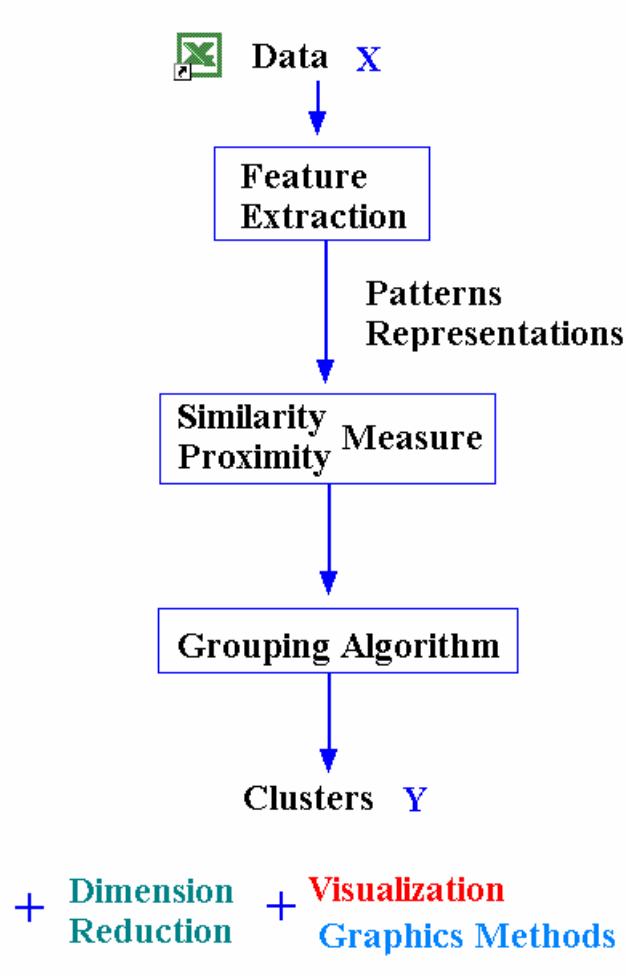
1. Most of data has been organized into non-overlapping clusters.
2. Each cluster has a within variance and one between variance for each of the other clusters. A good cluster should have a small within variance and large between variance.

Data types

- *binary / discrete / continuous*

Data scales

- *Qualitative: nominal / ordinal*
- *Quantitative: interval / ratio*



Clustering Analysis in Microarray Experiments

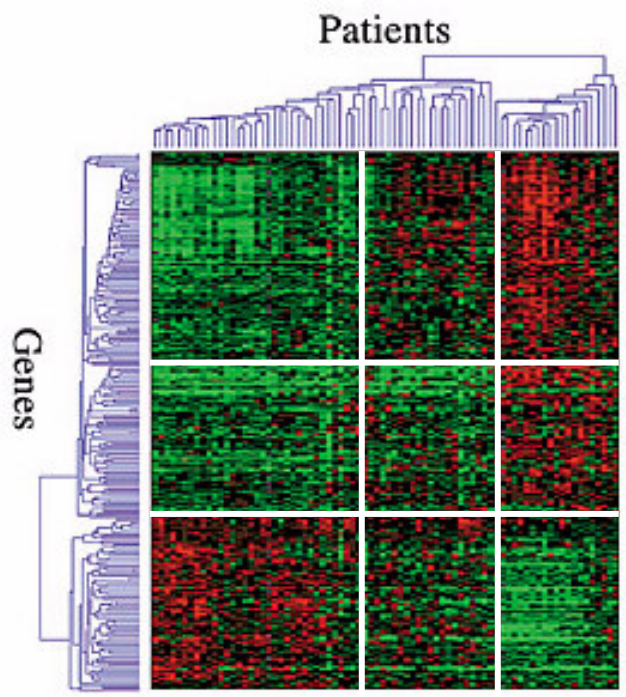
Goals

- Find natural classes in the data
- Identify new classes/gene correlations
- Refine existing taxonomies
- Support biological analysis/discovery

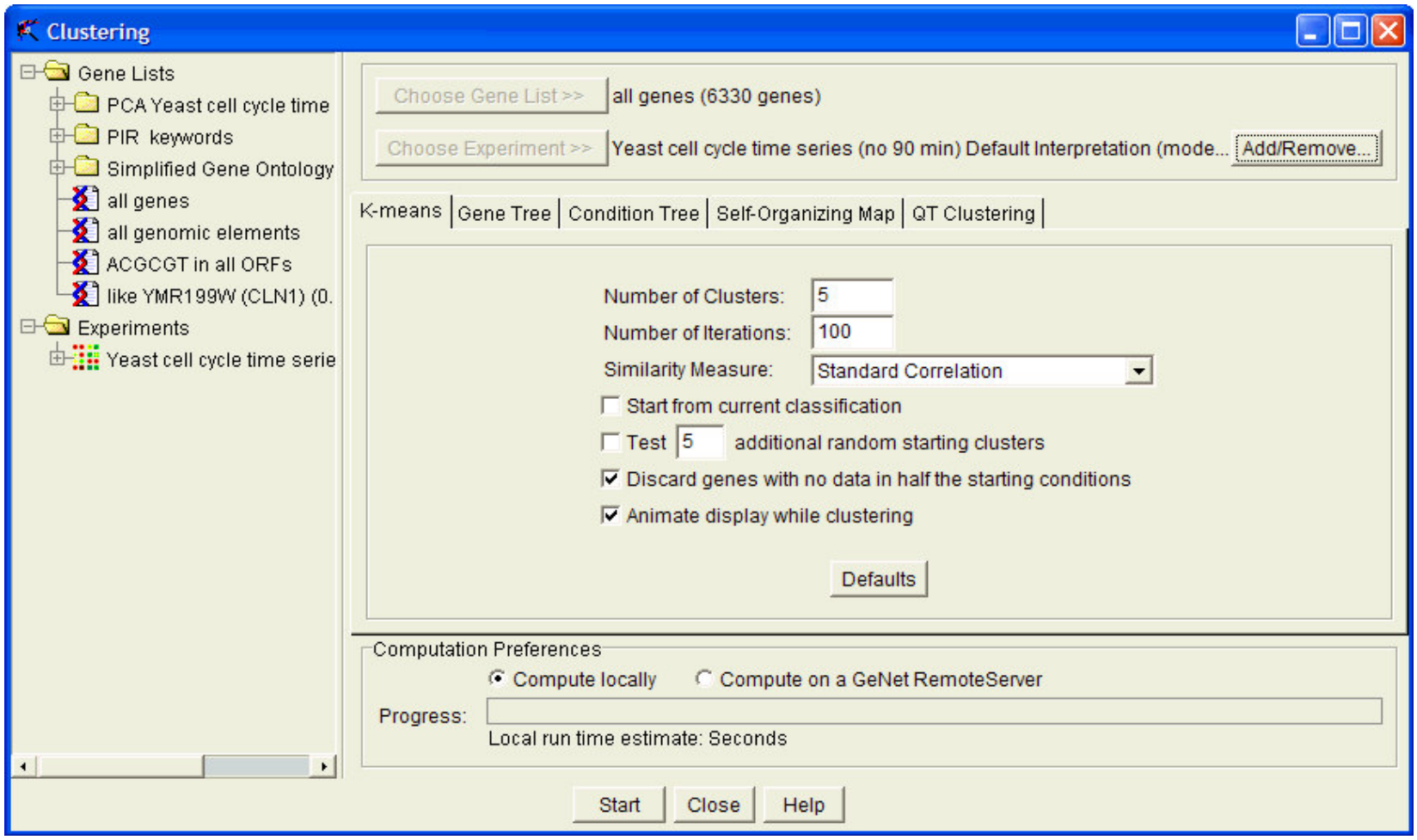
- cluster genes based on samples profiles
- cluster samples based on genes profiles

Hypothesis:

- genes with similar function have similar expression profiles



Clustering Analysis in GeneSpring 7.2



Source: GeneSpring Manual 7.2

K-Means Clustering

- K-means is a partition methods for clustering.
- Data are classified into k groups as specified by the user.
- Two different clusters cannot have any objects in common, and the k groups together constitute the full data set.

Optimization problem:

Minimize the sum of squared within-cluster distances

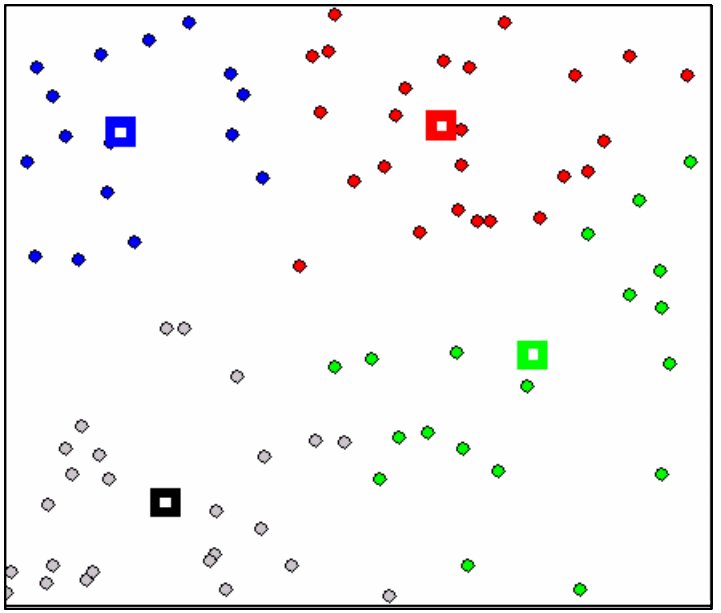
$$W(C) = \frac{1}{2}\sum_{k=1}^{K} \sum_{C(i)=C(j)=k} d_E(x_i, x_j)^2$$

The K-Means Algorithm

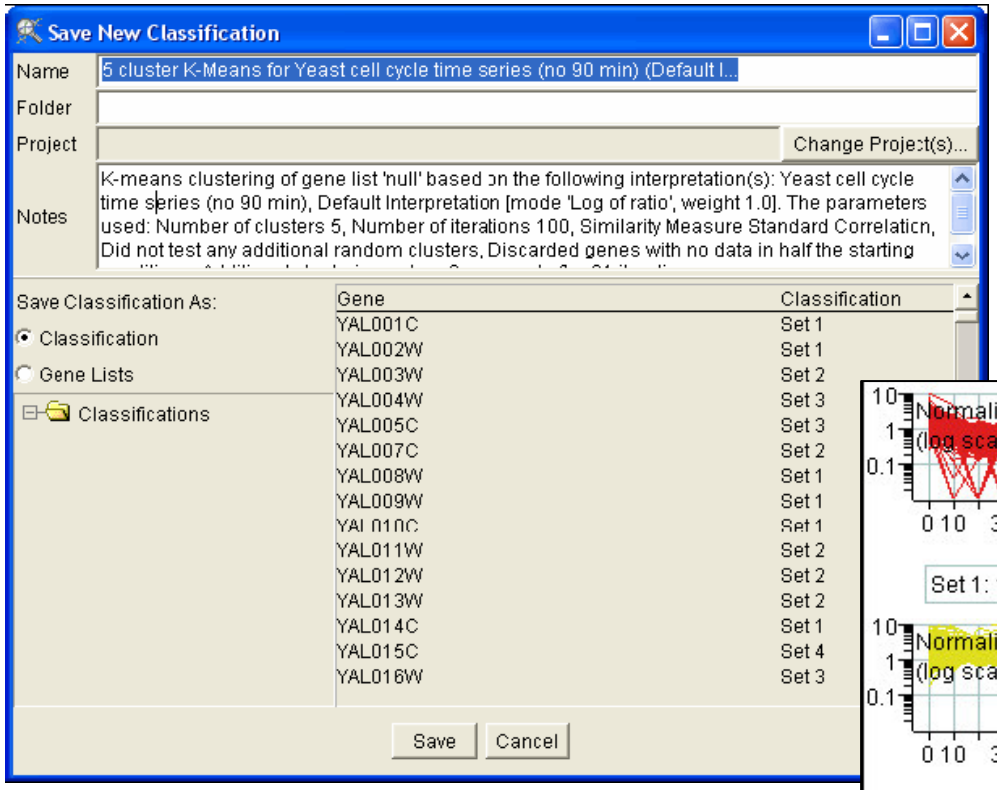
1. The data points are randomly assigned to one of the K clusters.
2. The position of the K centroids are determined (initial group centroids).
3. For each data point:
   - Calculate the distance from the data point to each cluster.
   - Assign data point to the cluster that has the closest centroid.
4. Repeat the above step until the centroids no longer move.

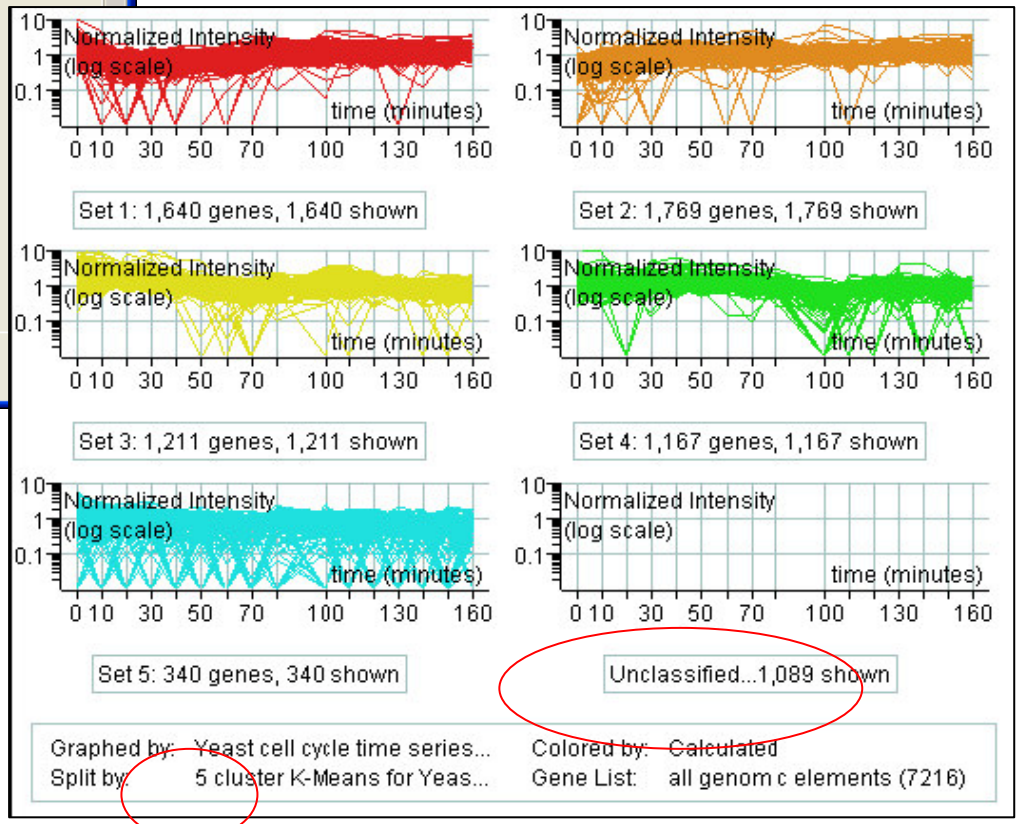
The choice of initial partition can greatly affect the final clusters that result.

Converged



Saving/Viewing K-means Cluster Results



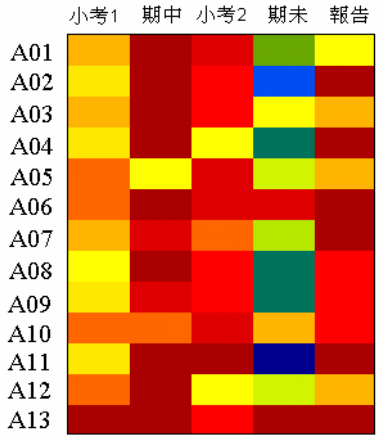


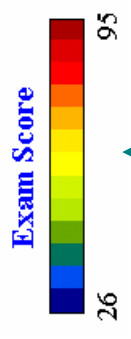
Source: GeneSpring Manual 7.2

Classifications

The Classifications folder in the Navigator contains genes that have been grouped or classified into groups as defined by K-means or SOM clustering.

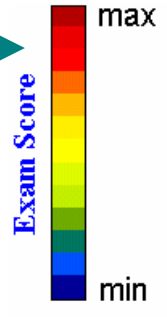
Heat Map

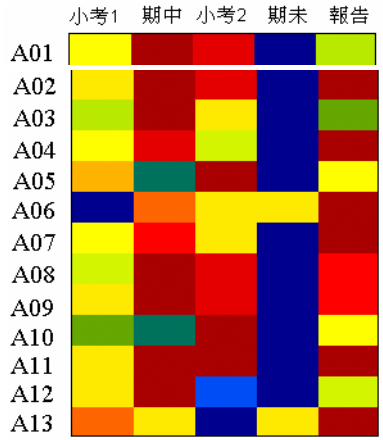




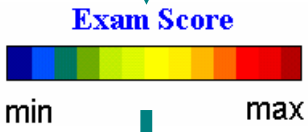
Range Matrix Condition

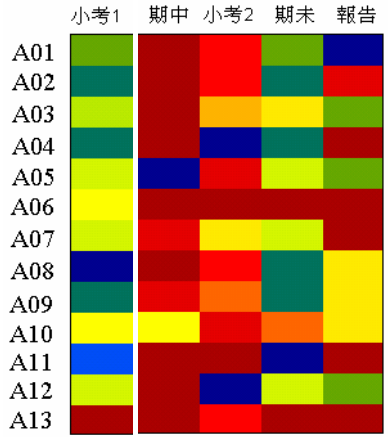
	A	B	C	D	E	F
1	學號	小考1	期中考	小考2	期末考	報告
2	A01	69	92	85	45	62
3	A02	66	90	83	36	90
4	A03	72	92	80	62	70
5	A04	68	90	60	37	95
6	A05	74	60	86	54	70
7	A06	77	90	88	88	95
8	A07	73	88	77	51	95
9	A08	61	90	84	40	82
10	A09	66	88	82	39	80
11	A10	76	75	87	72	80
12	A11	64	90	90	26	95
13	A12	75	90	60	55	70
14	A13	92	90	83	90	95





Range Raw Condition

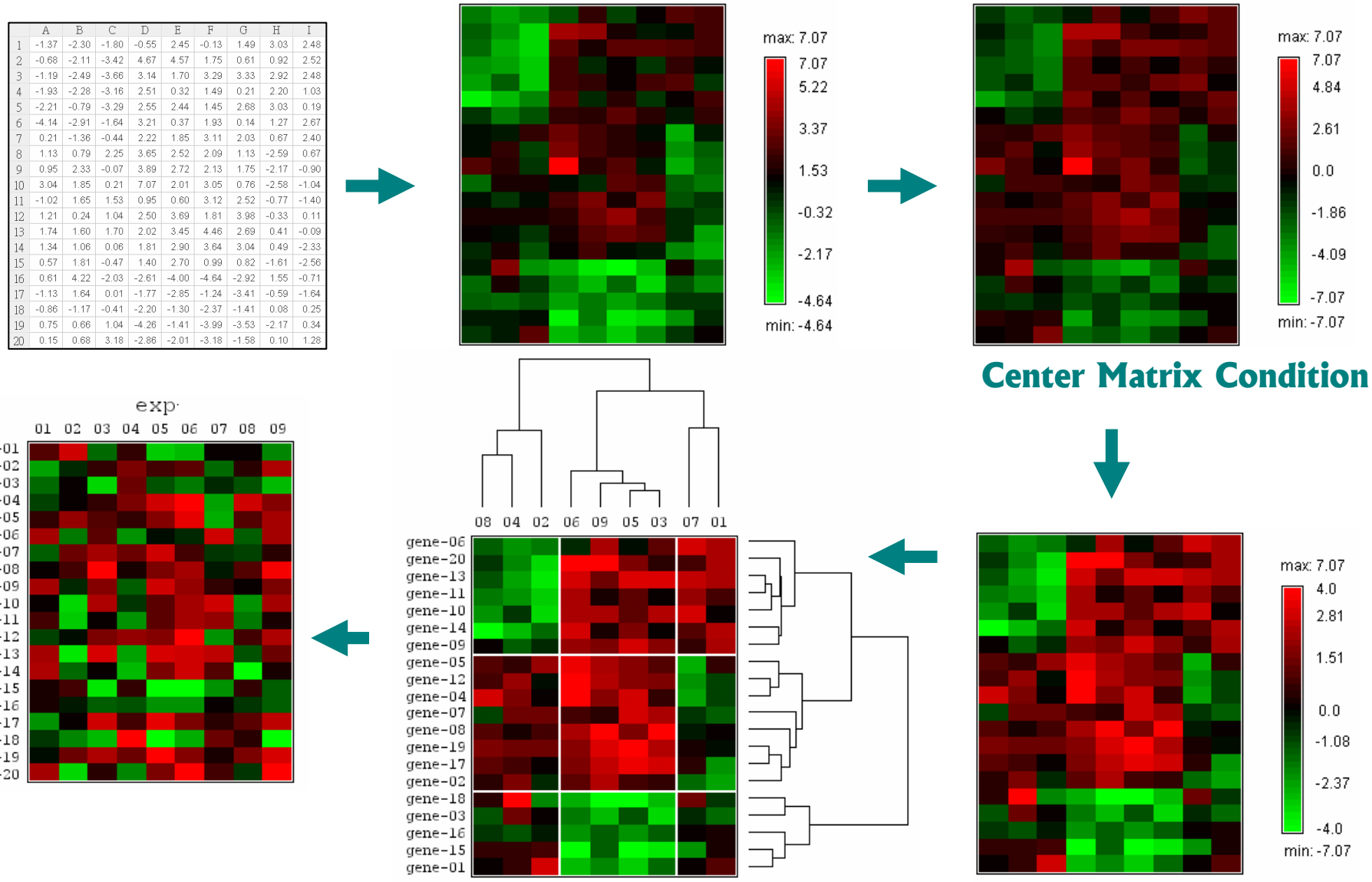




Range Column Condition

What about this one?

Heat Map (conti.)



Hierarchical Clustering

Hierarchical clustering can be perform using agglomerative and divisive approaches. The result is a tree that depicts the relationships between the objects.

- **Divisive clustering:** begin at step 1 with all the data in one cluster, in each subsequent step a cluster is split off, until there are n clusters.

- **Agglomerative clustering:** all the objects start apart. There are n clusters at step 0, each object forms a separate cluster. In each subsequent step two clusters are merged, until only cluster is left.

Non-Hierarchical clustering

- k-means
- The EM algorithm
- Nearest Neighbor
- …

Hierarchical Clustering and Dendrogram

(Kaufman and Rousseeuw, 1990)

Example:

Average-Linkage

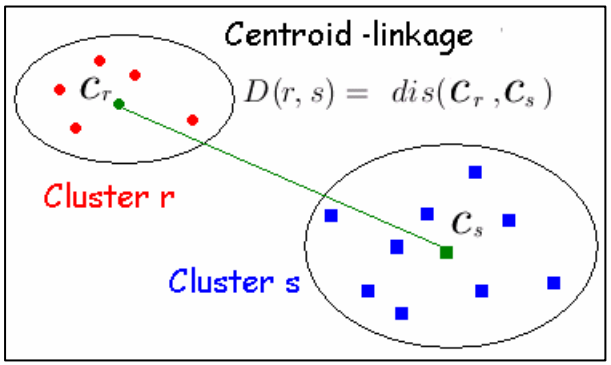
	a	b	c	d	e
a	0	2	6	10	9
b		0	5	9	8
c			0	4	5
d				0	3
e					0

	{a, b}	c	d	e
{a, b}	0	5.5	9.5	8.5
c		0	4	5
d			0	3
e				0

	{a,b}	c	{d, e}
{a, b}	0	5.5	9.0
c		0	4.5
{d, e}			0

	{a, b}	{c, d, e}
{a, b}	0	7.83
{c, d, e}		0

UPGMC (Unweighted Pair-Groups Method Centroid)



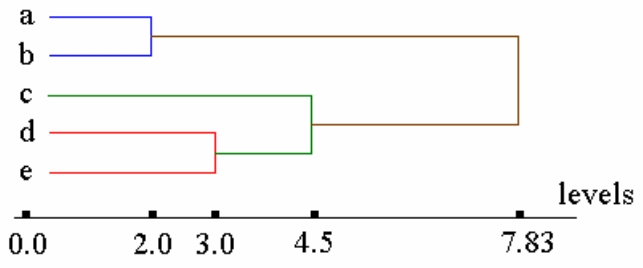
$$D(\{a,b\},\{c\}) = \frac{1}{2}[D(a,c) + D(b,c)]$$

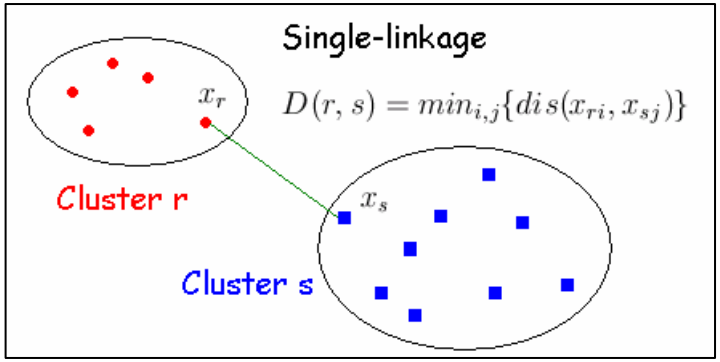
$$= \frac{1}{2}(6+5) = 5.5$$

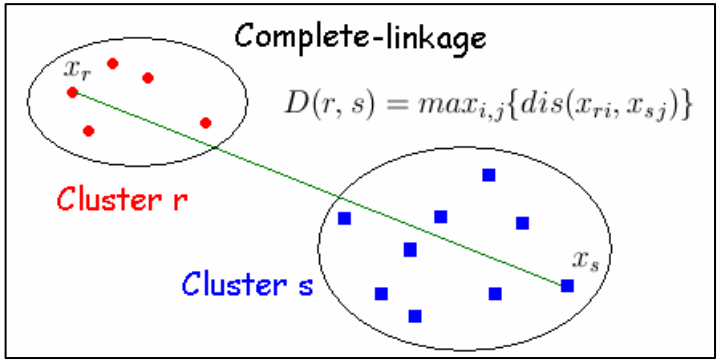
$$D(\{a,b\},\{d,e\})$$

$$= \frac{1}{4}[D(a,d)+D(a,e)+D(b,d)+D(b,e)]$$

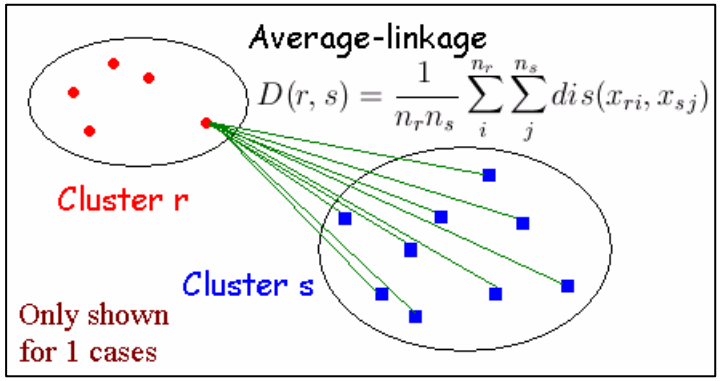
$$= \frac{1}{4}(10+9+9+8) = 9$$



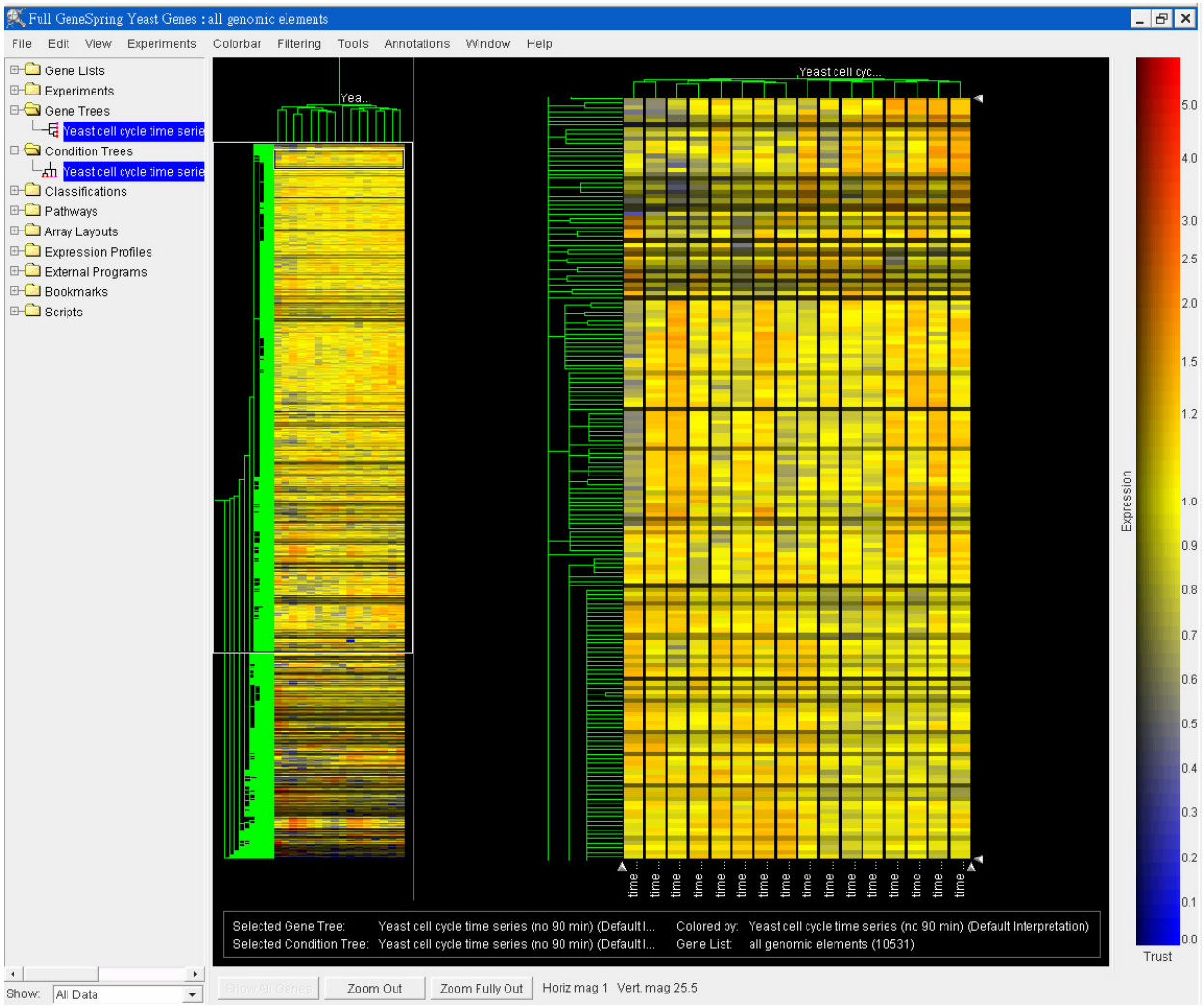




UPGMA (Unweighted Pair-Groups Method Average)



Viewing Gene/Condition Tree Clustering Results



- GeneSpring uses the 'centroid' clustering method. In this method, the distance between two clusters is the distance between the averages of the data points under one branch and the averages of the data points under another

Source: GeneSpring Manual 7.2

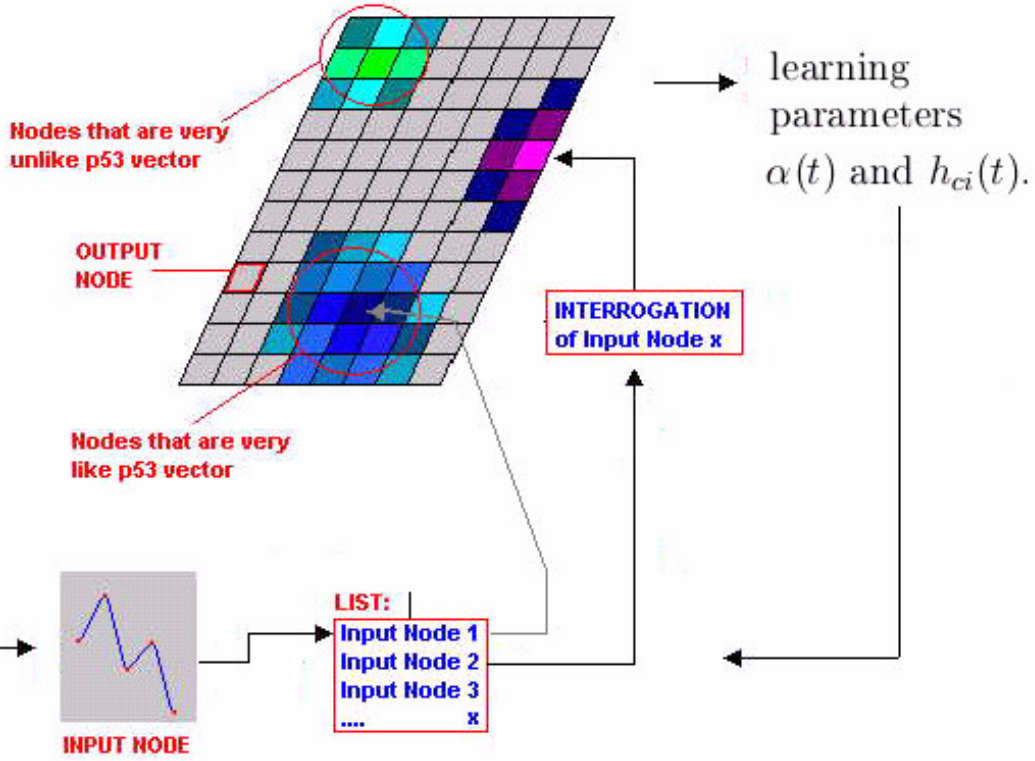
Self-Organizing Maps (SOM)

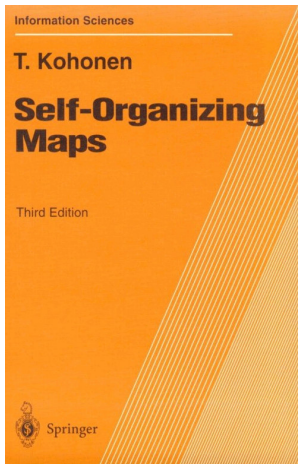
■ SOMs were developed by Kohonen in the early 1980's, original area was in the area of speech recognition.

■ ***Idea:*** Organise data on the basis of similarity by putting entities geometrically close to each other.

12x8=96群

■ SOM is unique in the sense that it combines both aspects. It can be used at the same time both to reduce the amount of data by clustering, and to construct a nonlinear projection of the data onto a low-dimensional display.





1995, 1997, 2001

Images:SCIpath

Algorithm of SOM

Step 0: Initialize weights $\mathbf{w}_i(t)$.
Set topological neighborhood parameters $N_c(t)$.
Set learning rate parameters $\alpha(t)$ and $h_{ci}(t)$.

Step 1: For each input vector $\mathbf{x}(t)$, do

a. Finding a BMU: $\|\mathbf{x}(t) - \mathbf{w}_c(t)\| = \min_i \|\mathbf{x}(t) - \mathbf{w}_i(t)\|$

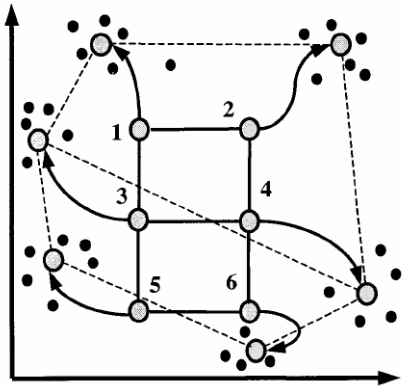
b. Learning process:

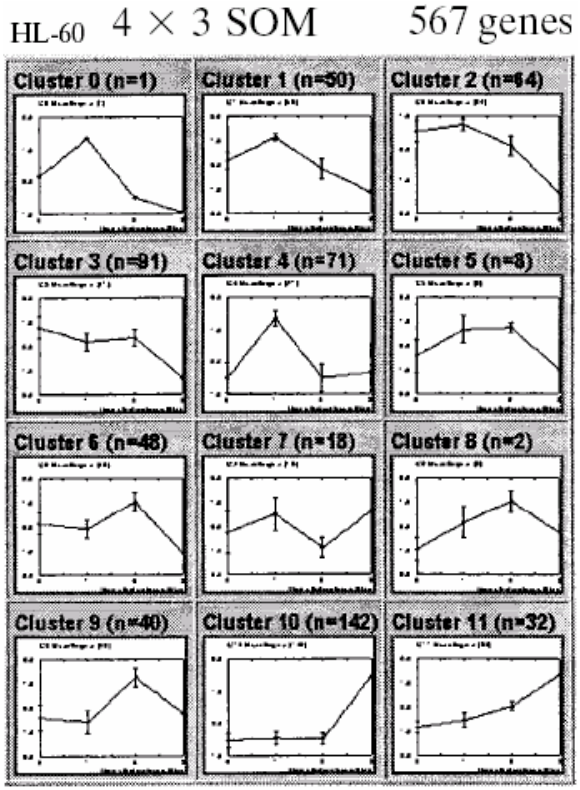
$$\mathbf{w}_i(t+1) = \begin{cases} \mathbf{w}_i(t) + h_{ci}(t)[\,\mathbf{x}(t) - \mathbf{w}_i(t)\,], & i \in N_c(t) \\ \mathbf{w}_i(t), & \text{o.w.} \end{cases}$$

c. Go to the next unvisited input vector. If there are no unvisited input vector left then go back to the very first one and go to Step 2.

Step 2: Incrementally decrease the learning rate and the neighborhood size, and repeat Step 1.

Step 3: Keep doing Steps 1 and 2 for a sufficient number of iterations.

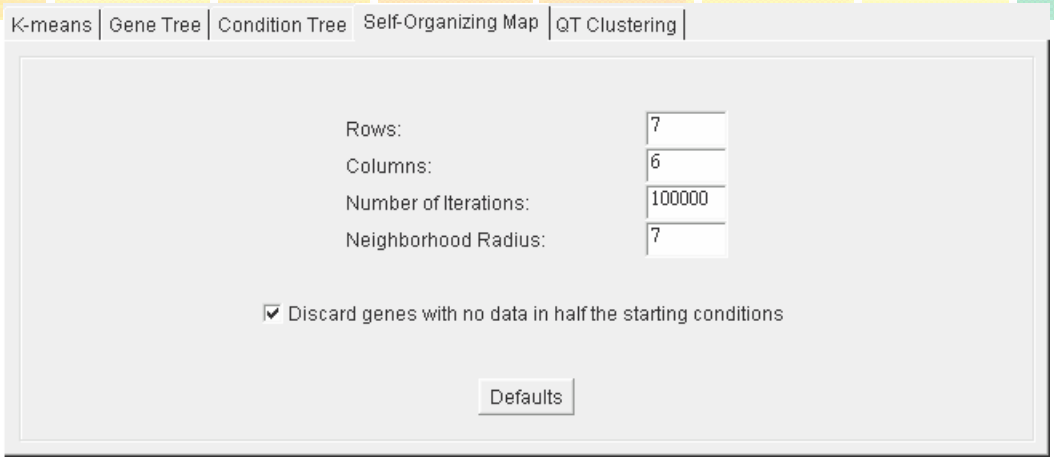


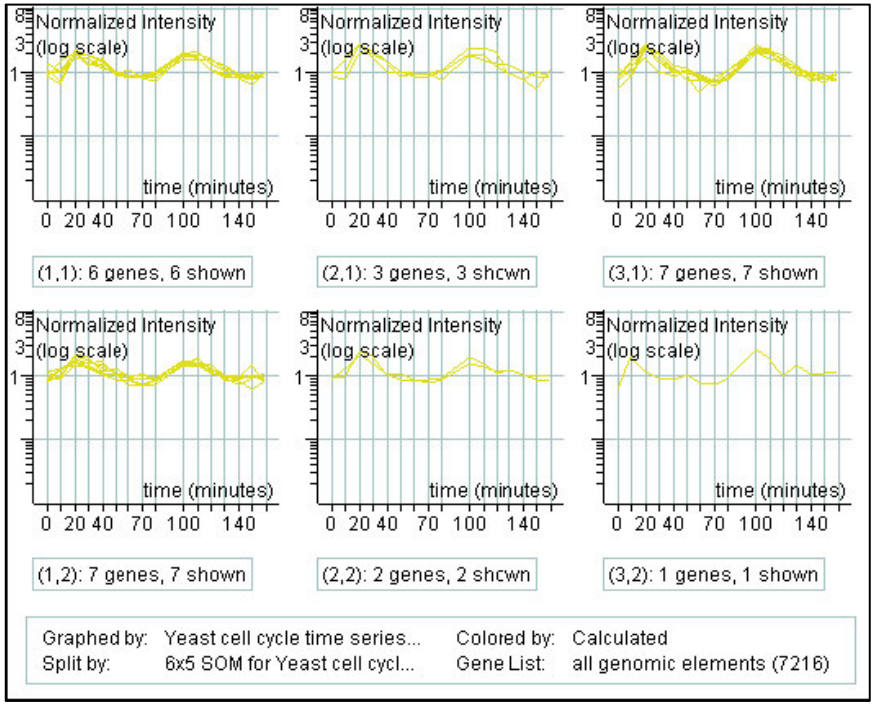


Macrophage Differentiation in HL-60 cells

Tamayo, P. et al. (1999). Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. *Proc Natl Acad Sci* 96:2907-2912.

SOM

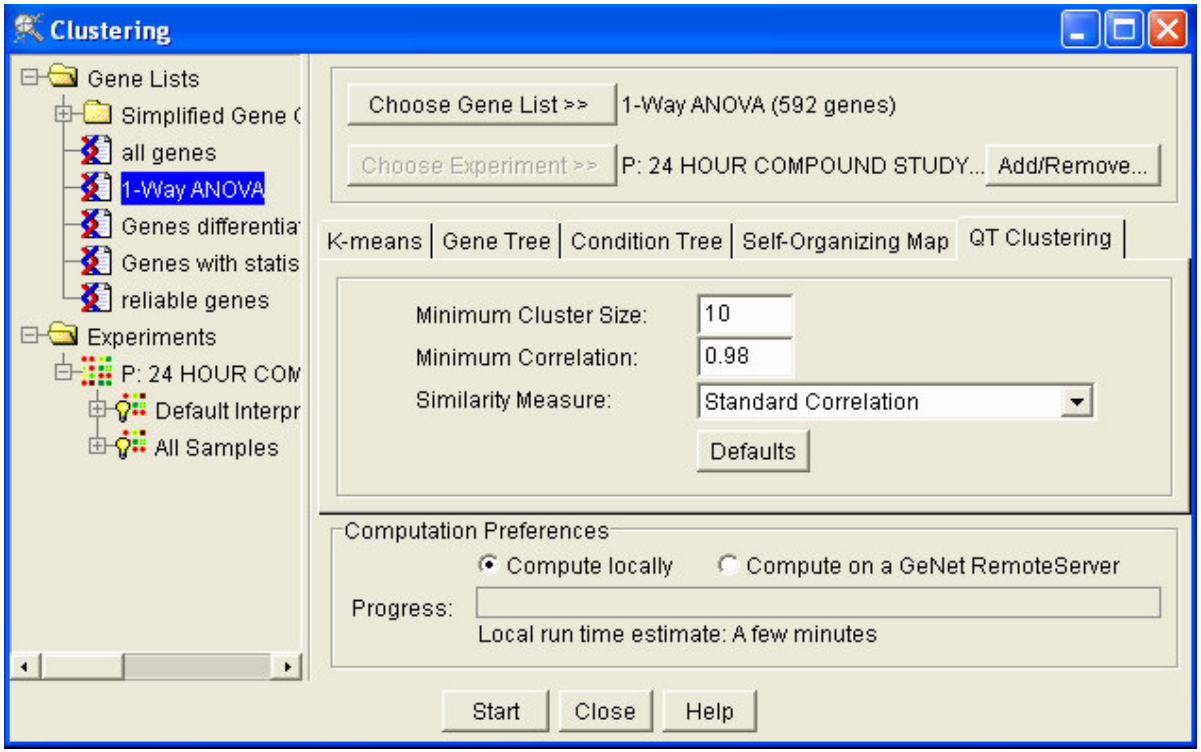




- **Rows**—The number of rows in your grid. The default setting is based on the number of genes and conditions in the selected experiment(s). •
- **Columns**—The number of columns in your grid. The default setting is based on the number of genes and conditions in the selected experiment(s).
- **Number of Iterations**—How many times each gene is examined. For example, if there are 10,000 genes and 60,000 iterations are specified, each gene is examined six times.
- **Neighborhood Radius**—How many nodes move toward a data point at the beginning of the iteration, and therefore how similar the profiles are for each node.

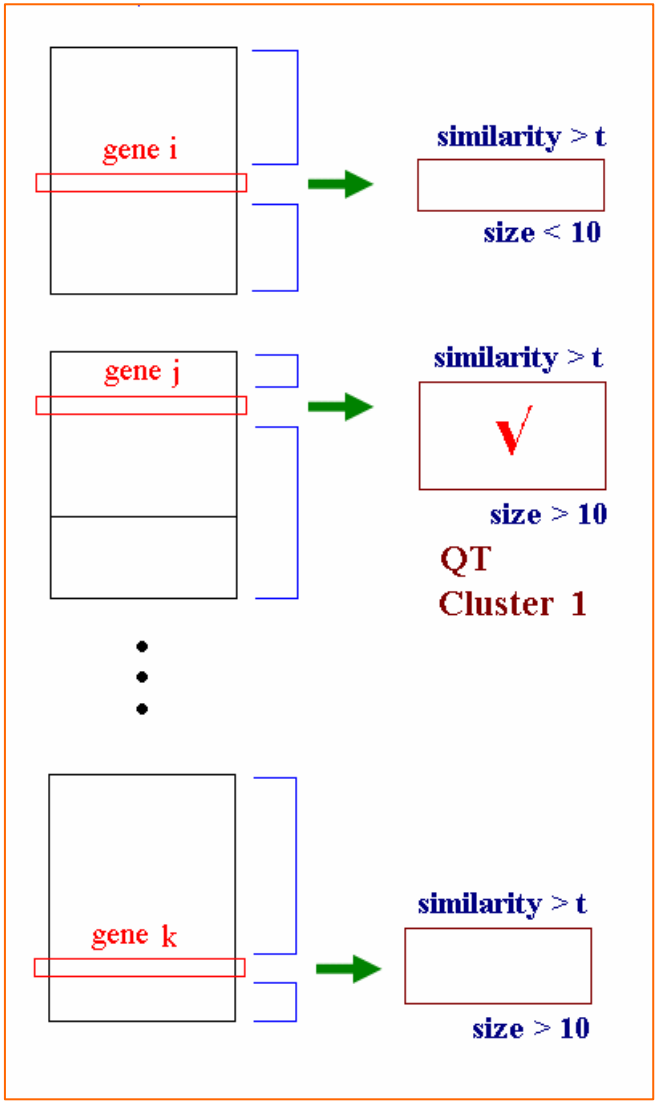
Source: GeneSpring Manual 7.2

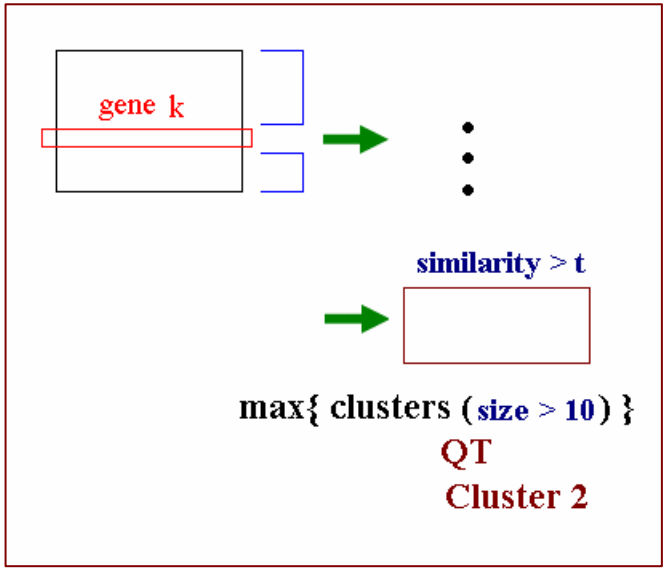
QT (Quality Threshold) Clustering



- **Minimum Cluster Size:** Minimum number of genes that you would like to have in each cluster.
- **Minimum Correlation:** Minimum correlation that genes within each cluster must have to one another.
- The diameter is the equivalent of 1 minus the minimum correlation.

Algorithm of QT Clustering

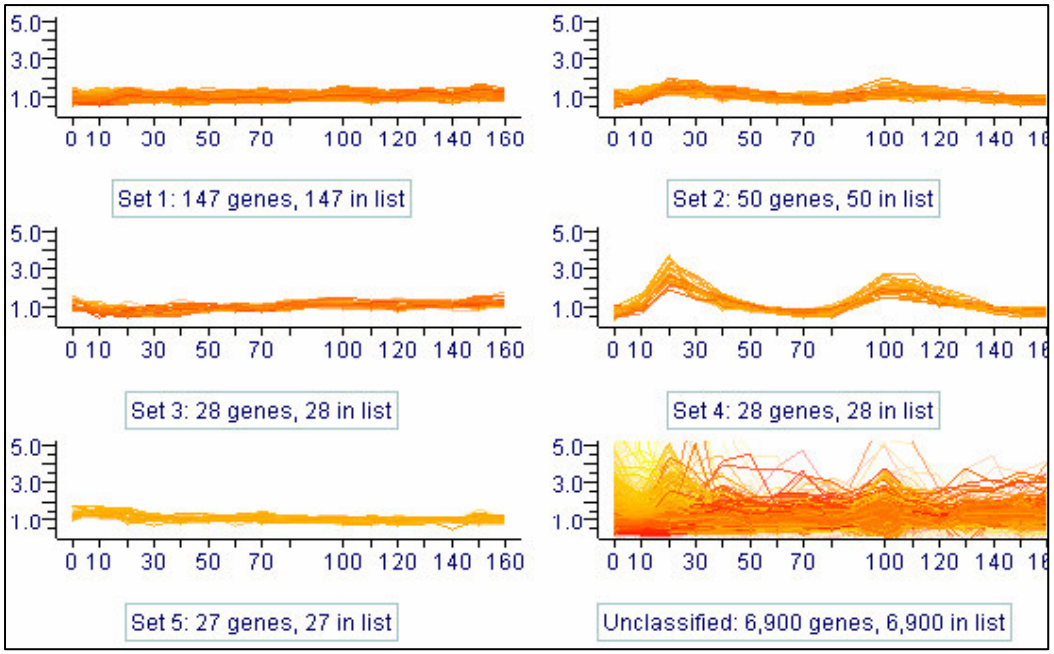




- The result is a set of non-overlapping QT clusters that meet quality threshold for both size, with respect to number of genes, and similarity, with respect to maximum allowable diameter.
- Genes that do not belong in any clusters will be grouped under the “unclassified” group.

Interpreting the Results

- QT Clusters are displayed according to the cluster size, from the largest to the smallest.
- Set 1 is the largest cluster, followed by set 2, etc…
- All sets will have **at least** the user-defined minimum cluster size and the minimum correlation (diameter).
- For example, all 147 genes in Set 1 below are at least 0.98 correlated to each other.
- Genes that did not meet the minimum quality are grouped under the "unclassified" category.



QT Clustering (conti.)

Advantages

- Quality Guarantee
- Number of clusters is not specified a priori
- All possible clusters are considered

Disadvantages

- Computationally Intensive/Time Consuming

Main differences between QT clustering and K-means clustering?

	K-means	QT clustering	Consequence
Need to specify cluster number?	Yes	No	K-means: if users specify too few clusters, genes that are not similar will be forced to group together.
Very computationally intensive?	No	Yes	QT clustering: may be too computationally intensive, depending on available RAM and number of genes in starting gene list, for some desktop computer.
Every gene must be clustered?	Yes	No	K-means: every gene on the selected gene list must belong to a cluster. This could potentially group genes that are not very similar into the same cluster. QT clustering: only cluster with user-specified quality will be formed.

Class Prediction Analysis

Analysis Guides: Class Prediction: K-Nearest Neighbors

http://www.chem.agilent.com/cag/bsp/SiG/Downloads/pdf/class_prediction.pdf

GeneSpring Tutorials: Viewing Genes by Classification View

http://www.chem.agilent.com/cag/bsp/SiG/Downloads/Tutorial/view_gene_by_classification.viewlet/view_gene_by_classification_viewlet_swf.html

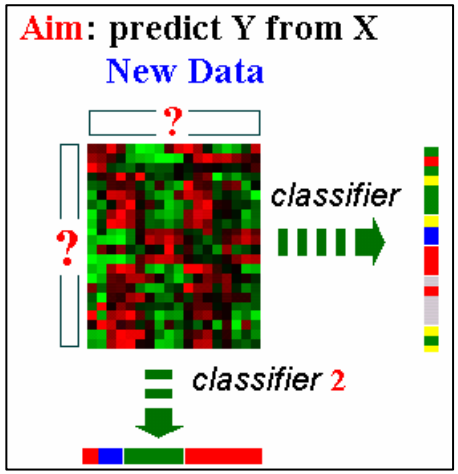
Class Prediction Analysis

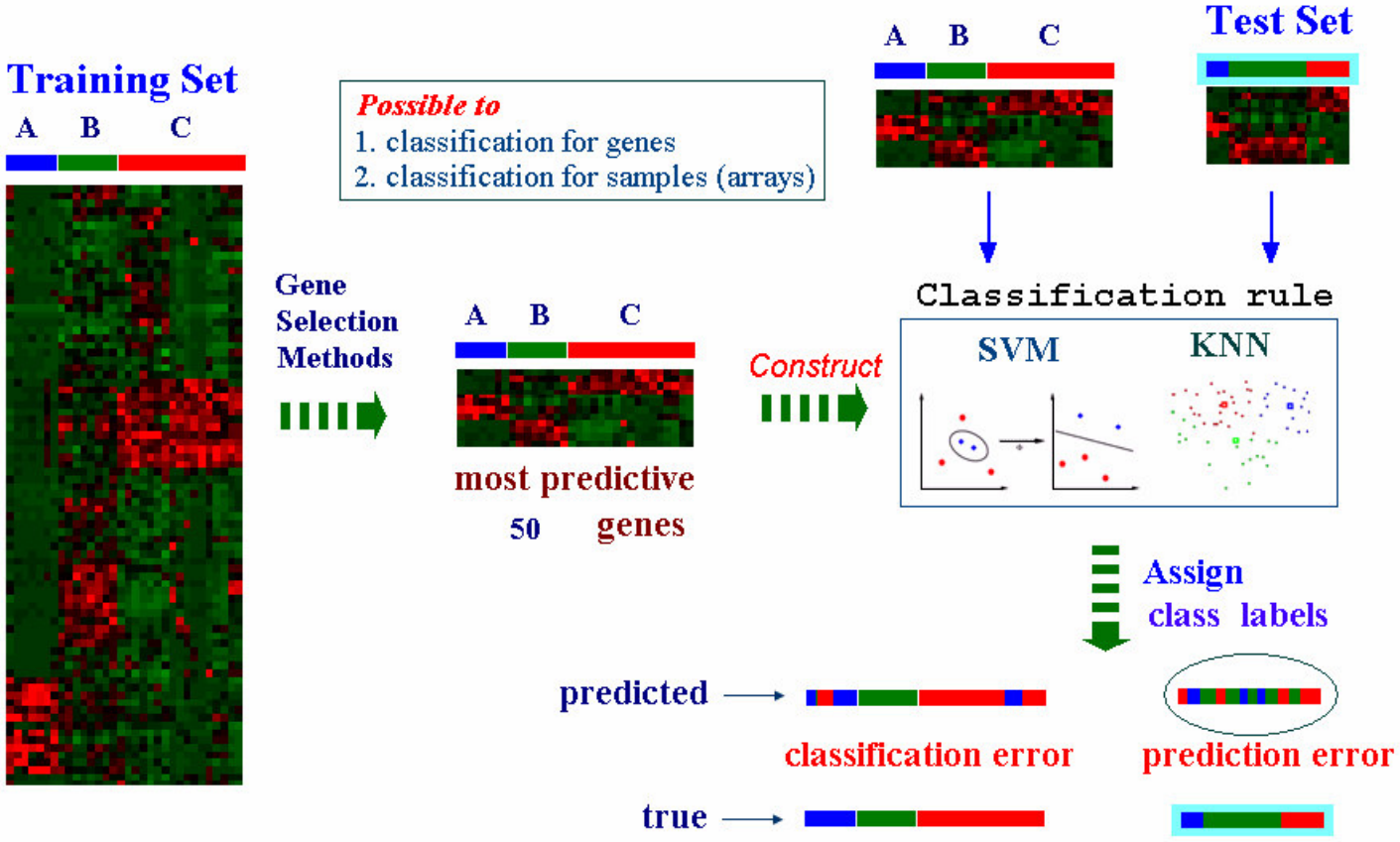
- Class prediction analysis is designed to predict the value, or “class”, of an individual parameter in an uncharacteristic sample or set of samples.

- The Class Predictor tool in GeneSpring performs this analysis in two steps.
  - First, the Class Predictor algorithm examines all genes in the training set individually and ranks them on their power to discriminate each class from all the others.
  - Next it uses the most predictive genes to classify the “test set” .

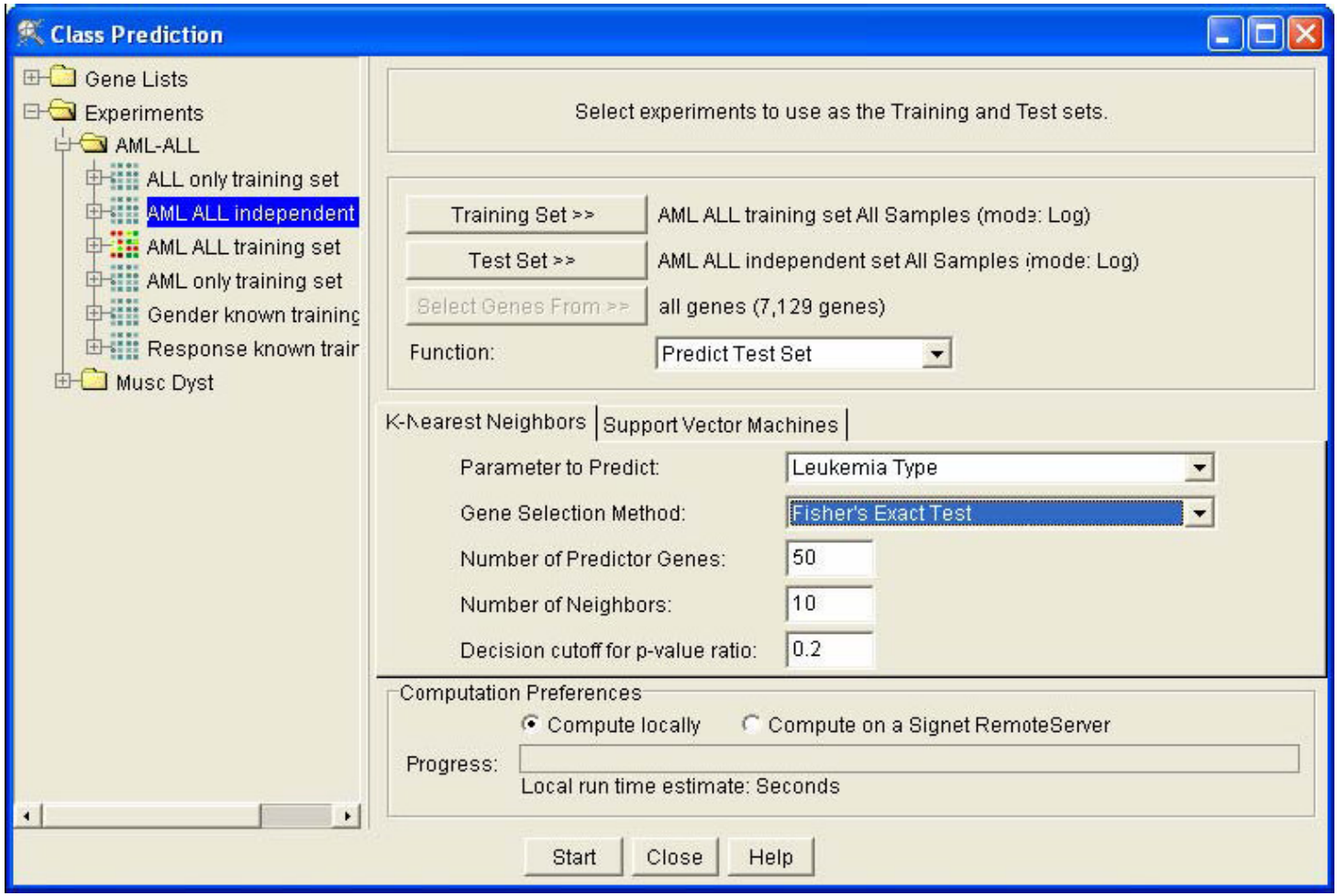
- This method can be used, for instance, to predict cancer types using genomic expression profiling.
  - Predict the class/phenotype/parameter of a sample
  - Identify genes that discriminate well among classes
  - Identify samples that could be potential outliers
- This technique is best used with at least 20 samples or conditions per class.

Classification of Genes, Tissues or Samples (Supervised Learning)





K-Nearest Neighbors



Gene Selection

Fisher's Exact Test

		First Variable		
		Type A	Type B	total
Second Variable	Category One	*a*	*b*	*a+b*
	Category Two	*c*	*d*	*c+d*
	totals	*a+c*	*b+d*	*n*

		Guess Poured First		
		Milk	Tea	total
Poured First	Milk	*3*	*1*	4
	Tea	*1*	*3*	4
	Totals	4	4	8

$$P(a) = \frac{\binom{a+b}{a}\binom{c+d}{c}}{\binom{n}{a+c}}$$

a	probability	p-value
0	0.014	1.000
1	0.229	0.986
2	0.514	0.757
3	0.229	0.243
4	0.014	0.014

$$P(3) = \frac{\binom{4}{3}\binom{4}{1}}{\binom{8}{4}} = 0.229 \quad P(4) = \frac{\binom{4}{4}\binom{4}{0}}{\binom{8}{4}} = 0.014$$

one-sided p-value = P(3) + P(4) = 0.243 (no evidence)

two-sided p-value = sum the hypergeometric probabilities of all outcomes y for which $\{P(y) \leq P(a)\}$

two-sided p-value = P(0) + P(1) + P(3) + P(4) = 0.486 (no evidence)

Golub Method

$$\left|\frac{\mu_1 - \mu_2}{\sigma_1 + \sigma_2}\right|$$

each gene is tested for its ability to discriminate between the classes using a signal-to-noise score

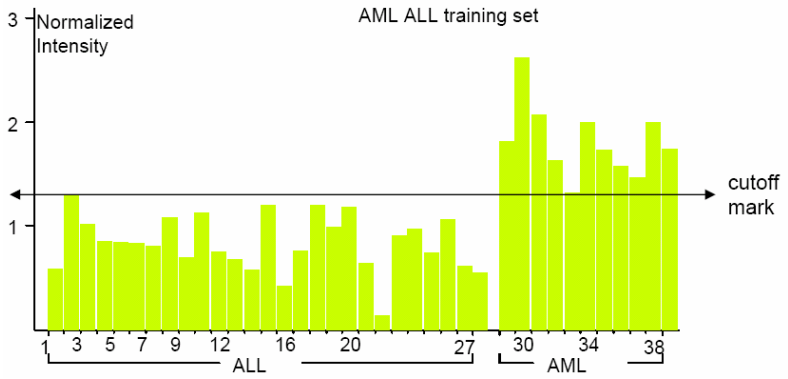
Gene Selection (conti.)

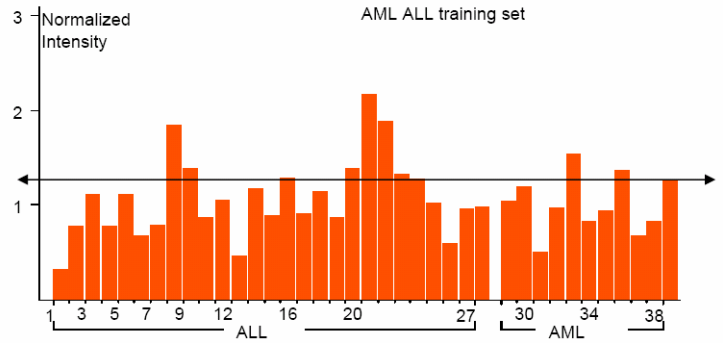
1. The class prediction isolates a gene.
2. For each sample, it calculates the probability of obtaining the observed number of samples from each class above and below that cutoff mark by chance, using Fisher's exact test.
3. Selects the smallest p-value calculated in step 2.

prediction strength = - log(p-value)

Repeats steps 1 to 3 for all genes.

4. Ranks the genes according to their predictive strength for each class.
5. Genes with highest predictive strength for each class are selected equally to generate a final list of best predictor genes. The final number of best predictors is user-specified.





Decision Cutoff for P-value Ratio

A rule indicating how the algorithm should make a prediction for the test sample.

- A p-value ratio of 0.2 (equivalent to 1/5) indicates that the algorithm will make a prediction if the p value (probability that the test sample is predicted as belonging to one class by chance) of the first best class is at least 5 times smaller than the p-value of the next best class.
- If the actual p value ratio is less than the cutoff, a prediction will be made.
- If the ratio is higher, no prediction will be made.
- Setting the p value cutoff to 1 will force the algorithm to always make a prediction but may result in more prediction errors.

K-Nearest Neighbors

- The number of k-nearest neighbors is user-defined.

1.Counts the k-nearest samples (in Euclidean distance) in the training set to the new sample to be classified.

2. Determines the proportion of neighbor samples from each class and makes a ‘vote’ for each class.

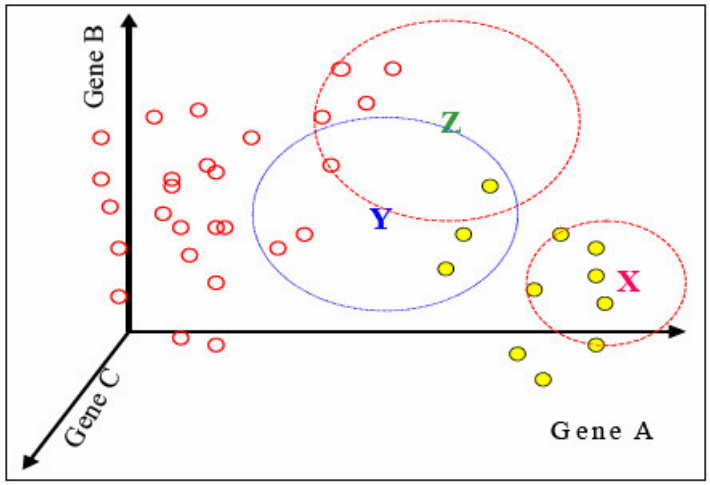
3. Calculates p-values for the likelihood of observed representation of each class.

4. Computes the ratio between the p-value of the most highly represented class and the p-value of the next most highly represented class.

5. Allows “no prediction” result if differential between p-values is above Decision cutoff for P-value ratio

K-Nearest Neighbors (conti.)

Number of Predictor genes: 3
Number of neighbors: 6
Decision cutoff P-value ratio: 0.2
There are 27 ALL training samples (represented by red circles) and 11 AML samples (represented by yellow spheres)

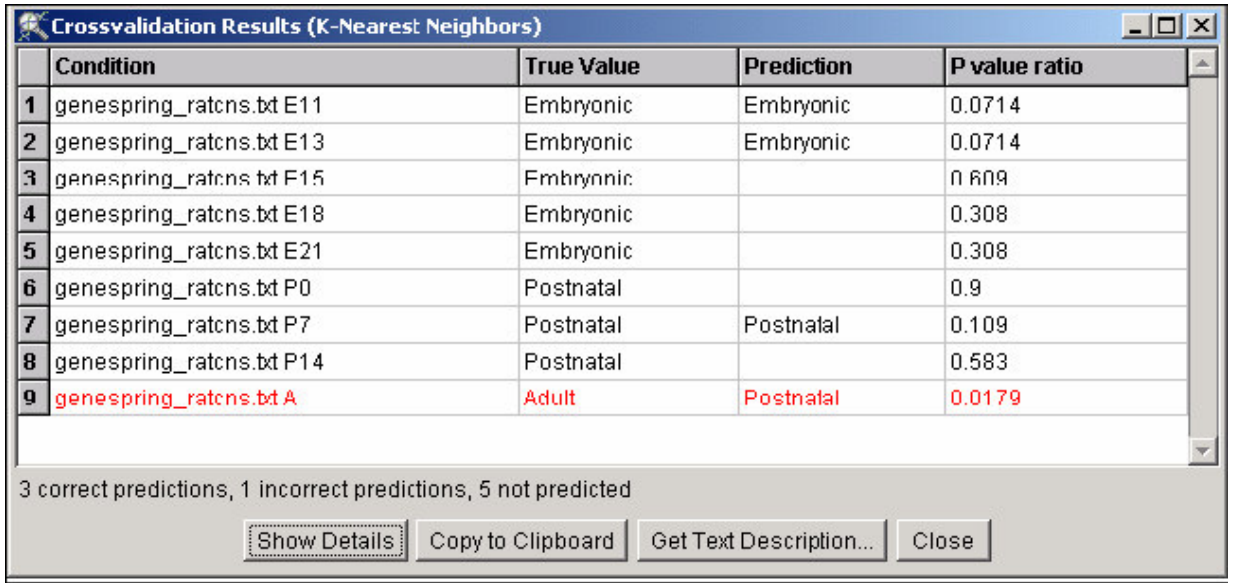


Test Sample	ALL vote	ALL p-value	AML vote	AML p-value	P value ratio	Prediction
X	0	1	6	0	0 (0.0 is less than decision cutoff p-value ratio, 0.2 ⇒ predicts sample X as AML)	AML
Y	3	.953**	3	.221**	.232 (0 .232 is higher than decision cutoff p-value ratio, 0.2 ⇒ do not make prediction for sample Y)	Not predicted
Z	5	.429	1	.893	.480 (0.480 is higher than decision cutoff p-value ratio, 0.2 ⇒ do not make prediction for sample Z)	Not predicted

Recommendations

- The class prediction analysis for k-nearest neighbors is designed for experiments with at least 20 or so samples in each class.
- It is possible to use the Class Predictor when you have very small sample sizes if you disable the p-value cutoff function.
- For sample sizes of less than 5, specify 1 or 2 number of neighbors and specify 1 in the p-value cutoff field.

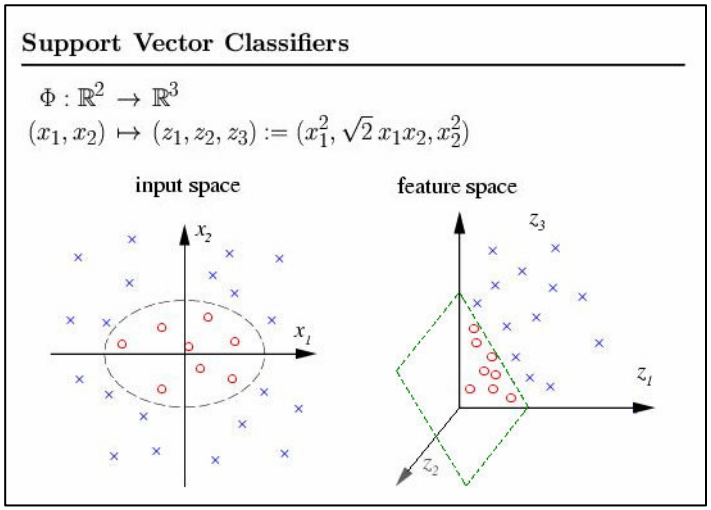
K-Nearest Neighbors (conti.)

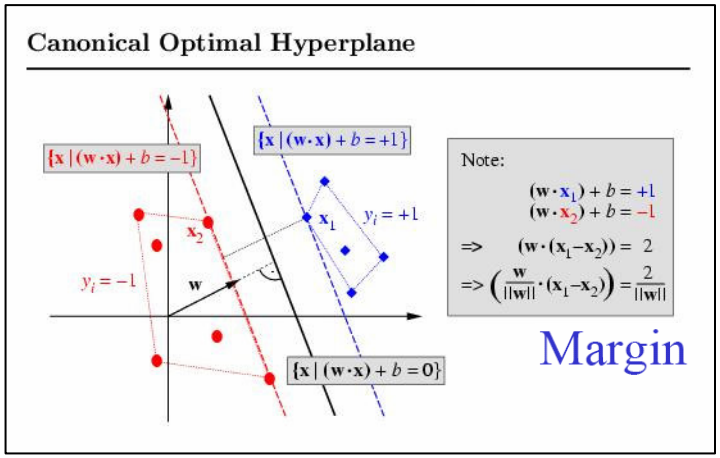


- P-values are computed for testing the likelihood of seeing at least the observed number of neighborhood members from each class based on the proportion in the whole training set.
- The class with the smallest p-value is given as the predicted class. The column labeled “P-value ratio” is the ratio of the p-value for the best class to that of the second-best class.
- The predictor will make a prediction if this ratio is less than the “P-value Cutoff” specified on the initial panel, and will not make a prediction if the ratio is above this cutoff.

Support Vector Machine (SVM)

SVMs (Vapnik, 1995) map the data (input space) into high dimensional space (feature space) through a kernel function ϕ and then find a hyperplane $\mathbf{w}$ to separate two groups (binary classification).





Quadratic Optimization Problem

- To find the optimal hyperplane
  (solve the quadratic optimization problem)
  To minimize the quadratic form $|W|^2 = (W * W)$ subject to the linear constraints $y_i((x_i * W) + b_0) \geq 1$

Multi-class problem

Two approaches for multi-class classification:

- **one-against-others**: The kth SVM model is constructed with all of the samples in the kth class with one group, and all other samples with the other group.
- **one-against-one**: The SVM trained model is constructed by using any two of classes. Therefore, there are total $K(K-1)/2$ classifiers.

decision function

$$f(\mathbf{X}) = \text{sign}\,((\mathbf{X} * W) + b_0)$$

SVM

- **Polynomial Dot Product 1**—The simplest form of similarity metric that provides the dot product between two vectors.

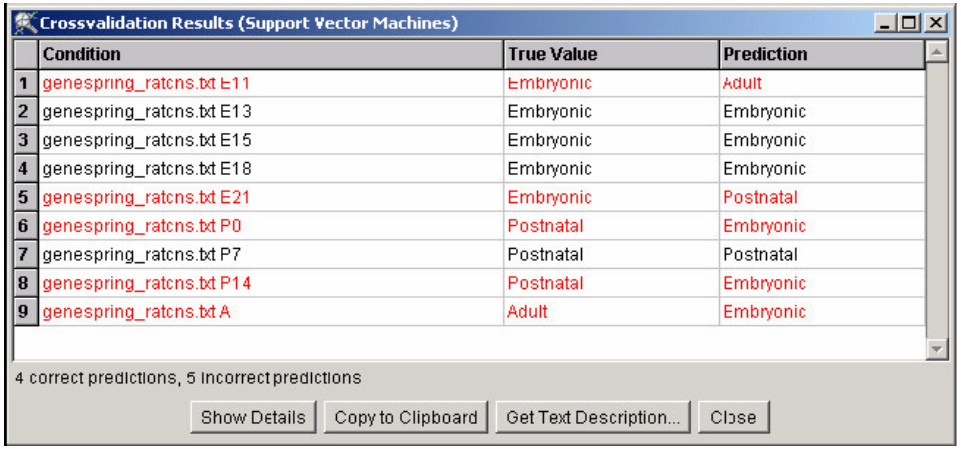
  `k(x,z)=dot(x,z)`, where `dot(x,z)` is the normal vector inner product operator.

- **Polynomial Dot Product 2, 3, or 4**—a higher-order polynomial kernel. The higher the user-defined parameter, the higher the capacity model that SVM can provide.

  `k(x,z)=(dot(x,z)+1)^d`, where `d` is user defined parameter

- **Gaussian Kernel**—This kernel has an infinite dimensional feature space and produces models that are good "univeral approximators." As v gets larger, the capacity gets lower.

  `k(x,z)=exp(-(||x-z||^2)/v)`, where `v` is user defined variance.



Specify a **Diagonal Scaling Factor**.

This option is used to control the misclassification rate. It corrects for unbalanced class sizes. A value of 0 assumes that the class sizes are equal.

END

hmwu@stat.sinica.edu.tw

http://www.sinica.edu.tw/~hmwu/Talks/index.htm

Silicon Genetics

GeneSpring
User Manual

Version 7.0