Chapter 11

An Introduction to Microarray Data Analysis

M. Madan Babu

Abstract

This chapter aims to provide an introduction to the analysis of gene expression data obtained using microarray experiments. It has been divided into four sections. The first section provides basic concepts on the working of microarrays and describes the basic principles behind a microarray experiment. The second section deals with the representation and extraction of information from images obtained from microarray experiments. The third section addresses different methods for comparing expression profiles of genes and also provides an overview of different methods for clustering genes with similar expression profiles. The last section focuses on relating gene expression data with other biological information; it will provide the readers with a feel for the kind of biological discoveries one can make by integrating gene expression data with external information.

1. INTRODUCTION

Functional genomics involves the analysis of large datasets of information derived from various biological experiments. One such type of large-scale experiment involves monitoring the expression levels of thousands of genes simultaneously under a particular condition, called gene expression analysis. Microarray technology makes this possible and the quantity of data generated from each experiment is enormous, dwarfing the amount of data generated by genome sequencing projects. This chapter is a brief overview of the basic concepts involved in a microarray experiment; it gives a feeling for what the data actually represents, and will provide information on the various computational methods that one can employ to derive meaningful results from such experiments.

1.1 What are microarrays and how do they work?

Microarray technology has become one of the indispensable tools that many biologists use to monitor genome wide expression levels of genes in a given organism. A microarray is typically a glass slide on to which DNA molecules are fixed in an orderly manner at specific locations called spots (or features). A microarray may contain thousands of spots and each spot may contain a few million copies of identical DNA molecules that uniquely correspond to a gene (Figure 1A). The DNA in a spot may either be genomic DNA or short stretch of oligo-nucleotide strands that correspond to a gene. The spots are printed on to the glass slide by a robot or are synthesised by the process of photolithography.

Microarrays may be used to measure gene expression in many ways, but one of the most popular applications is to compare expression of a set of genes from a cell maintained in a particular condition (condition A) to the same set of genes from a reference cell

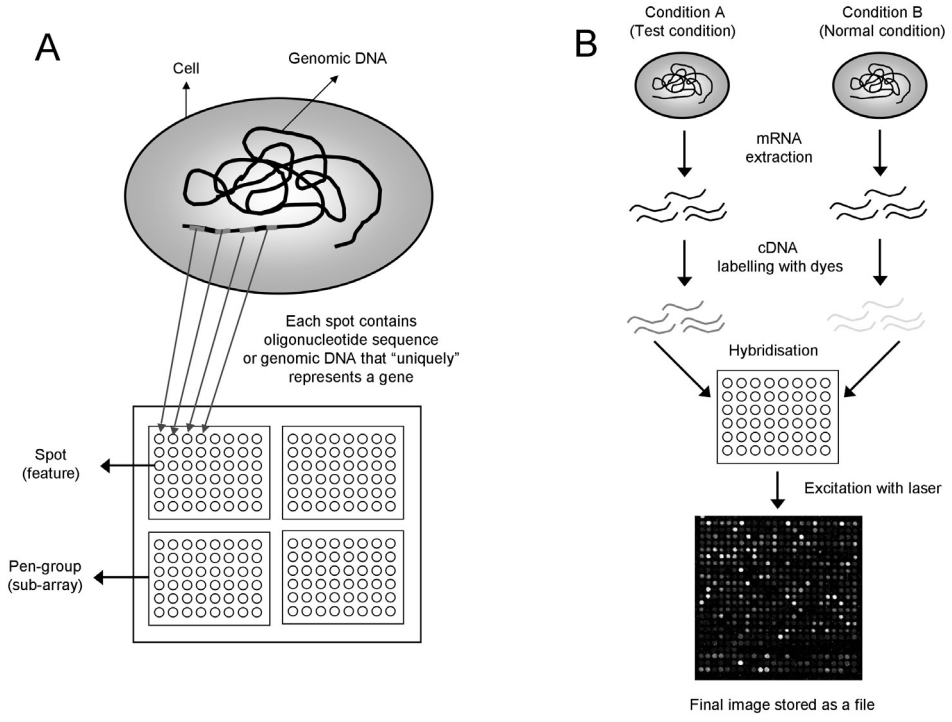

Figure 1. (A) A microarray may contain thousands of 'spots'. Each spot contains many copies of the same DNA sequence that uniquely represents a gene from an organism. Spots are arranged in an orderly fashion into Pen-groups. (B) Schematic of the experimental protocol to study differential expression of genes. The organism is grown in two different conditions (a reference condition and a test condition). RNA is extracted from the two cells, and is labelled with different dyes (red and green) during the synthesis of cDNA by reverse transcriptase. Following this step, cDNA is hybridized onto the microarray slide, where each cDNA molecule representing a gene will bind to the spot containing its complementary DNA sequence. The microarray slide is then excited with a laser at suitable wavelengths to detect the red and green dyes. The final image is stored as a file for further analysis. Colour figure at: http://www.mrc-lmb.cam.ac.uk/genomes/madanm/microarray/.

maintained under normal conditions (condition B). Figure 1B gives a general picture of the experimental steps involved. First, RNA is extracted from the cells. Next, RNA molecules in the extract are reverse transcribed into cDNA by using an enzyme reverse transcriptase and nucleotides labelled with different fluorescent dyes. For example, cDNA from cells grown in condition A may be labelled with a red dye and from cells grown in condition B with a green dye. Once the samples have been differentially labelled, they are allowed to hybridize onto the same glass slide. At this point, any cDNA sequence in the sample will hybridize to specific spots on the glass slide containing its complementary sequence. The amount of cDNA bound to a spot will be directly proportional to the initial number of RNA molecules present for that gene in both samples.

Following the hybridization step, the spots in the hybridized microarray are excited by a laser and scanned at suitable wavelengths to detect the red and green dyes. The amount of fluorescence emitted upon excitation corresponds to the amount of bound nucleic acid. For instance, if cDNA from condition A for a particular gene was in greater abundance than

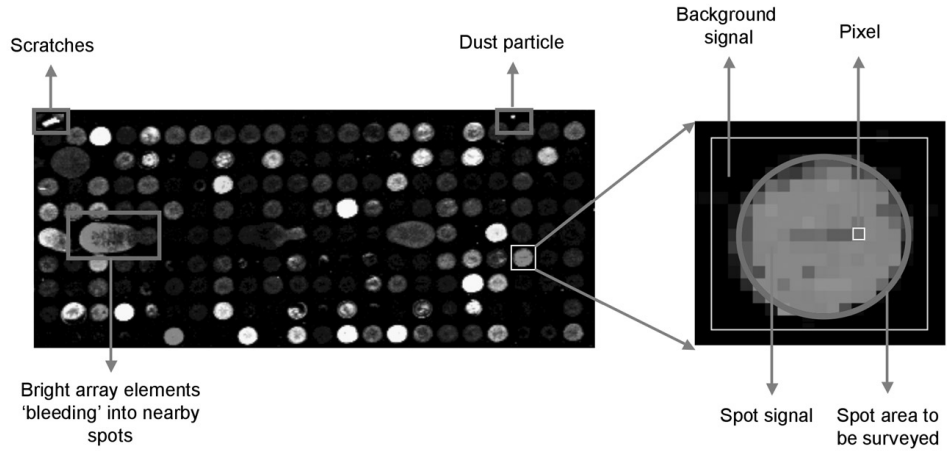

Figure 2. Zooming onto a spot on the microarray slide. The spot area and the background area are depicted by a blue circle and a white box, respectively. A pixel in the spot area is also shown. Any pixel within the blue circle will be treated as a signal from the spot. Pixels outside the blue circle but within the white box will be treated as a signal from the background. One can see that the images are not perfect, as it is often the case, which leads to many problems with spurious signals from dust particles, scratches, bright arrays, etc. This image was retrieved from Stanford Microarray Database. Colour figure at: http://www.mrc-lmb.cam.ac.uk/genomes/madanm/microarray/.

that from condition B, one would find the spot to be red. If it was the other way, the spot would be green. If the gene was expressed to the same extent in both conditions, one would find the spot to be yellow, and if the gene was not expressed in both conditions, the spot would be black. Thus, what is seen at the end of the experimental stage is an image of the microarray, in which each spot that corresponds to a gene has an associated fluorescence value representing the relative expression level of that gene.

2. OVERVIEW OF IMAGE PROCESSING, TRANSFORMATION AND NORMALIZATION

2.1 Image processing and analysis

In the previous section, we saw that the relative expression level for each gene (population of RNA in the two samples) can be stored as an image. The first step in the analysis of microarray data is to process this image. Most manufacturers of microarray scanners provide their own software; however, it is important to understand how data is actually being extracted from images, as this represents the primary data collection step and forms the basis of any further analysis.

Image processing involves the following steps:

1. *Identification of the spots and distinguishing them from spurious signals.*
   The microarray is scanned following hybridization and a TIFF image file is normally generated. Once image generation is completed, the image is analysed to identify spots. In the case of microarrays, the spots are arranged in an orderly manner into sub-arrays or pen groups (Figure 1A), which makes spot identification straightforward. Most image

processing software requires the user to specify approximately where each sub-array lies and also additional parameters relevant to the spotted array. This information is then used to identify regions that correspond to spots.

2. *Determination of the spot area to be surveyed, determination of the local region to estimate background hybridization.*
   After identifying regions that correspond to sub-arrays, an area within the sub-array must be selected to get a measure of the spot signal and an estimate for background intensity (Figure 2). There are two methods to define the spot signal. The first method is to use an area of a fixed size that is centred on the centre of mass of the spot. This method has an advantage that it is computationally less expensive, but a disadvantage of being more error-prone in estimating spot intensity and background intensity. An alternative method is to precisely define the boundary for a spot and only include pixels within the boundary. This method has an advantage that it can give a better estimate of the spot intensity, but also has a disadvantage of being computationally intensive and time-consuming.

3. *Reporting summary statistics and assigning spot intensity after subtracting for background intensity.*
   Once the spot and background areas have been defined, a variety of summary statistics for each spot in each channel (red and green channels) are reported. Typically, each pixel (Figure 2) within the area is taken into account, and the mean, median, and total values for the intensity considering all the pixels in the defined area are reported for both the spot and background. Most approaches use the spot median value, with the background median value subtracted from it, as the metric to represent spot intensity. The median intensity is a value where half the measured pixels have intensities greater than this value and the other half of the measured pixels have intensities less than this value. The "background subtracted median value" approach has an advantage of being relatively insensitive to a few pixels with anomalous fluorescent values in one or both channels, but has a disadvantage of being sensitive to misidentification of spot and background areas. The other method is to use total intensity values, which has an advantage of being insensitive to misidentification of spots (as few more pixels with zero value in the background will not affect the total intensity), but has a disadvantage of being prone to be skewed by a few pixels with extreme intensity values.

Another consideration in image processing is the number of pixels to be included for measurement in the spot image. For many scanners, the default pixel size is 10μm. This means that an average spot of diameter of 200μm will have ~314 pixels. However, for a smaller spot diameter, it is better to use a smaller pixel size to ensure enough pixels are sampled. Most scanners now allow the pixel size of 5μm. Even though using a smaller pixel size increases our confidence in the measurement, the only disadvantage is that the image file size tends to be much bigger when compared with image file sizes created using larger pixel sizes.

2.2 Expression ratios: the primary comparison

We saw that the relative expression level for a gene can be measured as the amount of red or green light emitted after excitation. The most common metric used to relate this information

is called expression ratio. It is denoted here as T_k and defined as:

$$T_k = \frac{R_k}{G_k}$$

For each gene k on the array, where R_k represents the spot intensity metric for the test sample and G_k represents the spot intensity metric for the reference sample. As mentioned above, the spot intensity metric for each gene can be represented as a total intensity value or a background subtracted median value. If we choose the median pixel value, then the median expression ratio for a given spot is:

$$T_{median} = \frac{R_{median}^{spot} - R_{median}^{background}}{G_{median}^{spot} - G_{median}^{background}}$$

where R_{median}^{spot} and $R_{median}^{background}$ are the median intensity values for the spot and background respectively, for the test sample.

2.3 Transformations of the expression ratio

The expression ratio is a relevant way of representing expression differences in a very intuitive manner. For example, genes that do not differ in their expression level will have an expression ratio of 1. However, this representation may be unhelpful when one has to represent up-regulation and down-regulation. For example, a gene that is up-regulated by a factor of 4 has an expression ratio of 4 (R/G = 4G/G = 4). However, for the case where a gene is down regulated by a factor of 4, the expression ratio becomes 0.25 (R/G = R/4R = 1/4). Thus up-regulation is blown up and mapped between 1 and infinity, whereas down-regulation is compressed and mapped between 0 and 1.

$$Up-regulation \xrightarrow{mapped} [1,\infty]$$

$$Down-regulation \xrightarrow{mapped} [0,1]$$

To eliminate this inconsistency in the mapping interval, one can perform two kinds of transformations of the expression ratio, namely, inverse transformation and logarithmic transformation.

Inverse or reciprocal transformation

The inverse or reciprocal transformation converts the expression ratio into a fold-change, where for genes with an expression ratio of less than 1 the reciprocal of the expression ratio is multiplied by -1. If the expression ratio is ≥ 1 then the fold change is equal to the expression ratio. The advantage of such a transformation is that one can represent up-regulation and down-regulation with a similar mapping interval.

$$Fold\ change = \begin{cases} T_k & if\ T_k \geq 1 \\ -1/T_k & if\ T_k < 1 \end{cases} \quad e.g.: \quad Fold\ change = \begin{cases} 4 & when\ T_k = 4 \\ -4 & when\ T_k = 0.25 \end{cases}$$

However, this method also has a problem in that the mapping space is discontinuous between –1 and +1 and hence becomes a problem in most mathematical analyses downstream of this step.

Logarithmic transformation

A better transformation procedure is to take the logarithm base 2 value of the expression ratio (*i.e.* $\log_2$ (expression ratio)). This has the major advantage that it treats differential up-regulation and down-regulation equally, and also has a continuous mapping space. For example, if the expression ratio is 1, then log2 (1) equals 0 represents no change in expression. If the expression ratio is 4, then $\log_2$ (4) equals +2 and for expression ratio of $\log_2$ ($^1/_4$) equals -2. Thus, in this transformation the mapping space is continuous and up-regulation and down-regulation are comparable.

Having explained the advantages of using expression ratios as a metric for gene expression, it should also be understood that there are disadvantages of using expression ratios or transformations of the ratios for data analysis. Even though expression ratios can reveal patterns inherent in the data, they remove all information about absolute expression levels of the genes. For example, genes that have R/G ratios of $^{400}/_{100}$ and $^4/_1$ will end up having the same expression ratio of 4, and associated problems will surface when one tries to reliably identify differentially regulated genes.

2.4 Data normalization

In the last section, it was shown that expression ratios and their transformations is a reasonable measure to detect differentially expressed genes. However, when one compares the expression levels of genes that should not change in the two conditions (say, housekeeping genes), what one quite often finds is that an average expression ratio of such genes deviates from 1. This may be due to various reasons, for example, variation caused by differential labelling efficiency of the two fluorescent dyes or different amounts of starting mRNA material in the two samples. Thus, in the case of microarray experiments, as for any large-scale experiments, there are many sources of systematic variation that affect measurements of gene expression levels.

Normalization is a term that is used to describe the process of eliminating such variations to allow appropriate comparison of data obtained from the two samples. There are many methods of normalization and discussing each one of them is beyond the scope of this chapter.

The first step in a normalization procedure is to choose a gene-set (which consists of genes for which expression levels should not change under the conditions studied, that is the expression ratio for all genes in the gene-set is expected to be 1. From that set, a *normalization factor,* which is a number that accounts for the variability seen in the gene-set, is calculated. It is then applied to the other genes in the microarray experiment. One should note that the normalization procedure changes the data, and is carried out only on the background corrected values for each spot. Figure 3 shows expression data before and after the normalization procedure.

Total intensity normalization

The basic assumption in a total intensity normalization is that the total quantity of RNA for the two samples is the same. Also assuming that the same number of molecules of RNA

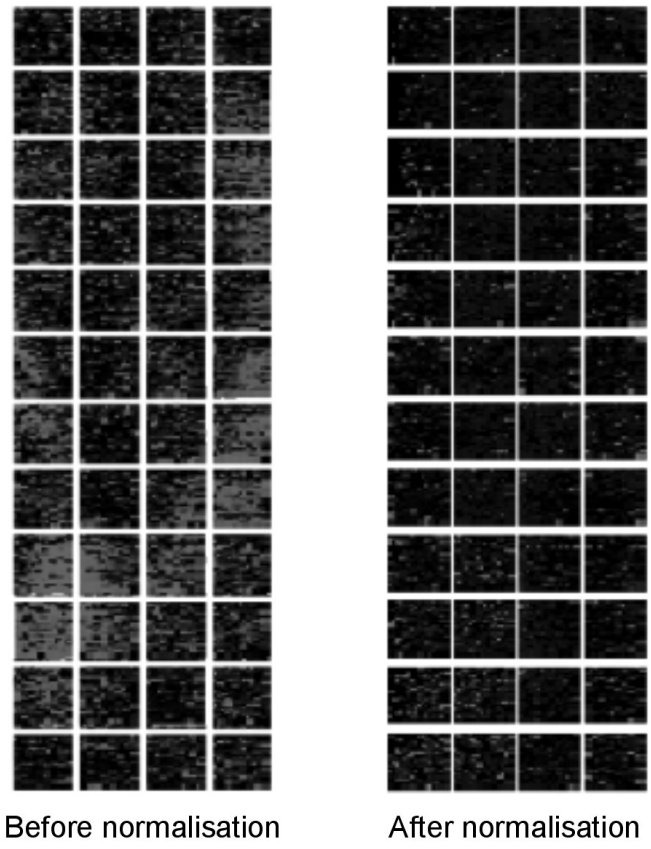

Figure 3. Gene expression data before and after the normalization procedure. Note that before normalization the image had many spots of different intensities, but after normalization only spots that are really different light up. This image was kindly provided by N. Luscombe. Colour figure at: http://www.mrc-lmb.cam.ac.uk/genomes/madanm/microarray/.

from both samples hybridize to the microarray, the total hybridization intensities for the gene-sets should be equal. So, a normalization factor can be calculated as:

$$N_{total} = \frac{\sum_{k=1}^{N_{gene-set}} R_k}{\sum_{k=1}^{N_{gene-set}} G_k}$$

The intensities are now rescaled such that $G_k^{'} = G_k \times N_{total}$ and $R_k^{'} = R_k$. The normalized expression ratio becomes:

$$T_k^{'} = \frac{R_k^{'}}{G_k^{'}} = \frac{R_k}{G_k \times N_{total}} = \frac{T_k}{N_{total}}$$

Which is equivalent to:

$$\log_2 (T_k^{'}) = \log_2 (T_k) - \log_2 (N_{total})$$

This now adjusts the ratio such that the mean ratio for the gene set is equal to 1.

Mean log centring

In this method, the basic assumption is that the mean $\log_2$ (expression ratio) should be equal to 0 for the gene-set. In this case, the normalization factor can be calculated as:

$$N_{mlc} = \frac{\sum_{k=1}^{N_{gene-set}} \log_2 \left(\frac{R_k}{G_k} \right)}{N_{gene-set}}$$

The intensities are now rescaled such that $G_k^{'} = G_k \times (2^{N_{mlc}})$ and $R_k^{'} = R_k$. The normalized expression ratio becomes:

$$T_k^{'} = \frac{R_k^{'}}{G_k^{'}} = \frac{R_k}{G_k \times (2^{N_{mlc}})} = \frac{T_k}{2^{N_{mlc}}}$$

Which is equivalent to:

$$\log_2 (T_k^{'}) = \log_2 (T_k) - \log_2 (2^{N_{mlc}}) = \log_2 (T_k) - N_{mlc}$$

This adjusts the ratio such that the mean $\log_2$ (expression ratio) for the gene-set is equal to 0.

Other normalization methods include: linear regression, Chen's ratio statistics and Lowess normalization. The next step following the normalization procedure is to filter low intensity data using specific threshold or relative threshold imposed according to the background intensity. If the experimental procedure included a replicate, averaging the values using the replicate data is the next step to be performed after data filtering. Finally, differentially expressed genes are identified. For an excellent review of normalization procedures, filtering methods and averaging procedures using replicate data, please refer to Quackenbush, (2002) and references therein.

3. ANALYSIS OF GENE EXPRESSION DATA

One of the reasons to carry out a microarray experiment is to monitor the expression level of genes at a genome scale. Patterns could be derived from analysing the change in expression of the genes, and new insights could be gained into the underlying biology. In this section, basic terminologies, representations of the microarray data and the various methods by which expression data can be analysed will be introduced.

The processed data, after the normalization procedure, can then be represented in the form of a matrix, often called gene expression matrix (Table 1A). Each row in the matrix corresponds

Table 1. A: Gene expression matrix that contains rows representing genes and columns representing particular conditions. Each cell contains a value, given in arbitrary units, that reflects the expression level of a gene under a corresponding condition. B: Condition C4 is used as a reference and all other conditions are normalized with respect to C4 to obtain expression ratios. C: In this table all expression ratios were converted into the $\log_2$ (expression ratio) values. This representation has an advantage of treating up-regulation and down-regulation on comparable scales. D: Discrete values for the elements in Table 1.C. Genes with $\log_2$ (expression ratio) values greater than 1 were changed to 1, genes with values less than –1 were changed to –1. Any value between –1 and 1 was changed to 0.

Table 1.A: Absolute measurement

	C1	C2	C3	C4
Gene A	10	80	40	20
Gene B	100	200	400	200
Gene C	30	240	60	60
Gene D	20	160	80	80

Table 1.B: Relative measurement

	C1/C4	C2/C4	C3/C4
Gene A	0.50	4.00	2.00
Gene B	0.50	1.00	2.00
Gene C	0.50	4.00	1.00
Gene D	0.25	2.00	1.00

Table 1.C: $\log_2$(relative measurement)

	$\log_2$ (C1/C4)	$\log_2$ (C2/C4)	$\log_2$ (C3/C4)
Gene A	-1	2	1
Gene B	-1	0	1
Gene C	-1	2	0
Gene D	-2	1	0

Table 1.D: Discrete values

	D [$\log_2$ (C1/C4)]	D [$\log_2$ (C2/C4)]	D [$\log_2$ (C3/C4)]
Gene A	0	1	0
Gene B	0	0	0
Gene C	0	1	0
Gene D	-1	0	0

to a particular gene and each column could either correspond to an experimental condition or a specific time point at which expression of the genes has been measured. The expression levels for a gene across different experimental conditions are cumulatively called the gene expression profile, and the expression levels for all genes under an experimental condition are cumulatively called the sample expression profile. Once we have obtained the gene expression matrix (Table 1A), additional levels of annotation can be added either to the gene or to the sample. For example, the function of the genes can be provided, or the additional details on the biology of the sample may be provided, such as ‘disease state’ or ‘normal state’.

Depending on whether the annotation is used or not, analysis of gene expression data can be classified into two different types, namely supervised or unsupervised learning. In the case of a supervised learning, we do use the annotation of either the gene or the sample, and create clusters of genes or samples in order to identify patterns that are characteristic for the cluster. For example, we could separate sample expression profiles into ‘disease state’ and ‘normal state’ groups, and then look for patterns that separate the sample profile of the ‘disease state’ from the sample profile of the ‘normal state’.

In the case of an unsupervised learning, the expression data is analysed to identify patterns that can group genes or samples into clusters without the use of any form of

annotation. For example, genes with similar expression profiles can be clustered together without the use of any annotation. However, annotation information may be taken into account at a later stage to make meaningful biological inferences. Throughout this section, set of genes or set of experimental conditions that have similar expression profiles will be referred to as a 'cluster'. Thus, a cluster consists of 'objects' with similar expression profiles, where an object may either refer to genes or samples.

3.1 Representation of gene expression data

To make any meaningful comparison or biological analysis, one should know what the data in the gene expression matrix represents. Expression data can be represented in five different ways, which are described below:

Absolute measurement

In the case of an absolute measurement, each cell in the matrix will represent the expression level of the gene in abstract units. Note that it is not meaningful to compare expression levels of genes across two different conditions in absolute units, because the starting amounts of mRNA could be different. Table 1A shows a sample gene expression matrix with each cell containing the expression level in abstract units.

Relative measurement or expression ratio

In the case of a relative measurement or representations involving expression ratio, the expression level of a gene in abstract units is normalized with respect to its expression in a reference condition. This gives the expression ratio of the gene in relative units. Note that in such cases, a ratio of $^{4000}/_{100}$ will lead to the same result as $^{40}/_{10}$. Thus any information on absolute measurement will be lost in such a representation, but now meaningful comparison across different conditions can be made as long as the same reference condition is used to get the expression ratio. As mentioned before, this representation does not treat up-regulation and down-regulation in a comparable manner. Table 1B shows the gene expression matrix with each cell representing the expression ratio normalized with respect to a reference condition.

$\log_2$(expression ratio)

In the case of tables representing the $\log_2$ (expression ratio) values, information on up-regulation and down-regulation is captured and is mapped in a symmetric manner. For example, 4-fold up-regulation maps to $\log_2 (4) = 2$ and a 4-fold down-regulation maps to $\log_2 (1/4) = -2$. Thus, from this table the fold-change for a differentially regulated gene under any condition can be easily recognised. Table 1C shows the $\log_2$ (expression ratio) values of the genes under different conditions.

Discrete values

Another way of representing information is to convert to discrete numbers the values in the tables mentioned above. In the case of converting the absolute measurement to discrete numbers, a binary expression matrix of 1 and 0 can be used, where 1 means that the gene is expressed above a user defined threshold, and 0 means that the gene is expressed below this threshold. In the case of making the relative expression tables or $\log_2$ (expression ratio) tables discrete, values can be divided into 3 classes, +1, 0 and –1, where +1 represents a gene that is positively regulated, 0 represents a gene that is not differentially regulated and –1 represents

a gene that is repressed. The process of making the values discrete loses a lot of information, but is useful to analyse expression profiles using algorithms that cannot handle real value expression matrices, for example algorithms calculating mutual information between genes or samples. Table 1D shows discrete values for the $\log_2$ (expression ratio) table.

Representation of expression profiles as vectors
So far we have seen how individual cells in the gene expression matrix can be represented. Similarly, an expression profile (of a gene or a sample) can be thought of as a vector and can be represented in vector space. For example, an expression profile of a gene can be considered as a vector in *n* dimensional space (where n is the number of conditions), and an expression profile of a sample with *m* genes can be considered as a vector in *m* dimensional space (where *m* is the number of genes). In the example given below, the gene expression matrix X with *m* genes across *n* conditions is considered to be an *m* x *n* matrix, where the expression value for gene *i* in condition *j* is denoted as x_{ij}:

$$X = \begin{bmatrix} x_{11} & x_{12} & .. & x_{1n} \\ x_{21} & x_{22} & .. & x_{2n} \\ .. & .. & .. & .. \\ x_{m1} & x_{m2} & .. & x_{mn} \end{bmatrix}$$

The expression profile of a gene *i* can be represented as a row vector:

$$G_i = \begin{bmatrix} x_{i1}, & x_{i2}, & x_{i3}, & .., & x_{in} \end{bmatrix}$$

The expression profile of a sample *j* can be represented as a column vector:

$$G_i = \begin{bmatrix} x_{1j} \\ x_{2j} \\ .. \\ x_{mj} \end{bmatrix}$$

In the next section, we will see how expression profiles that are represented as vectors can be used to compare how similar or different are the pairs of objects (remember, an object may refer to a gene or a sample).

3.2 Distance measures

Analysis of gene expression data is primarily based on comparison of gene expression profiles or sample expression profiles. In order to compare expression profiles, we need a measure to quantify how similar or dissimilar are the objects that are being considered. A variety of distance measures can be used to calculate similarity in expression profiles and these are discussed below.

Euclidean distance

Euclidean distance is one of the common distance measures used to calculate similarity between expression profiles. The Euclidean distance between two vectors of dimension 2, say $A=[a_1, a_2]$ and $B=[b_1, b_2]$ can be calculated as:

$$D_{Euc}(A,B) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2}$$

For instance two genes with expression profiles in two conditions $G_1=[1,2]$ and $G_2=[2,3]$, the Euclidean distance can be calculated as:

$$D_{Euc}(G_1,G_2) = \sqrt{(1\ -2\ )^2 + (2\ -3\ )^2} = \sqrt{2}$$

Thus for genes with expression data available for n conditions, represented as $A=[a_1, .., a_n]$ and $B=[b_1, .., b_n]$, Euclidean distance can be calculated as:

$$D_{Euc}(A,B) = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2}$$

In other words, the Euclidean distance between two genes is the square root of the sum of the squares of the distances between the values in each condition (dimension).

A more general form of the Euclidean distance is called the *Minkowski distance*, calculated as:

$$D_{Min}(A,B) = \sqrt[p]{\sum_{i=1}^{n}(a_i - b_i)^p}$$

A special case of the Minkowski distance when p=1 is called rectilinear distance. When applied to binary expression profiles (*i.e.* expression levels changed to 1 and 0), it is called hamming distance.

Pearson correlation coefficient

One of the most commonly used metrics to measure similarity between expression profiles is the Pearson correlation coefficient (PCC) (Eisen *et al.* 1998). Given the expression ratios for two genes under three conditions $A=[a_1, a_2, a_3]$ and $B=[b_1, b_2, b_3]$, PCC can be computed as follows:

Step1: Compute mean

$$\bar{a} = \frac{a_1 + a_2 + a_3}{3} \quad and \quad \bar{b} = \frac{b_1 + b_2 + b_3}{3}$$

Step2: "Mean centre" expression profiles

$$\overline{A} = (a_1 - \bar{a},\ a_2 - \bar{a},\ a_3 - \bar{a}) \quad and \quad \overline{B} = (b_1 - \bar{b},\ b_2 - \bar{b},\ b_3 - \bar{b})$$

Step3: Calculate PCC as the cosine of the angle between the mean-centred profiles

$$PCC = \frac{\overline{A} \circ \overline{B}}{\left|\overline{A}\right| \left|\overline{B}\right|}$$

Where,

$$\overline{A} \circ \overline{B} = \sum_{i=1}^{n} (a_i - \bar{a}) \times (b_i - \bar{b})$$

$$\left|\overline{A}\right| = \sqrt{\sum_{i=1}^{n} (a_i - \bar{a})^2} \qquad \left|\overline{B}\right| = \sqrt{\sum_{i=1}^{n} (b_i - \bar{b})^2}$$

and

The reason why we "mean centre" the expression profiles is to make sure that we compare 'shapes' of the expression profiles and not their magnitude. Mean centring maintains the shape of the profile, but it changes the magnitude of the profile as shown in Figure 4.

A PCC value of 1 essentially means that the two genes have similar expression profiles and a value of –1 means that the two genes have exactly opposite expression profiles. A value of 0 means that no relationship can be inferred between the expression profiles of genes. In reality, PCC values range from –1 to +1. A PCC value ≥ 0.7 suggests that the genes behave similarly and a PCC value ≤ -0.7 suggests that the genes have opposite behaviour. The value of 0.7 is an arbitrary cut-off, and in real cases this value can be chosen depending on the dataset used. An example calculation is shown below:

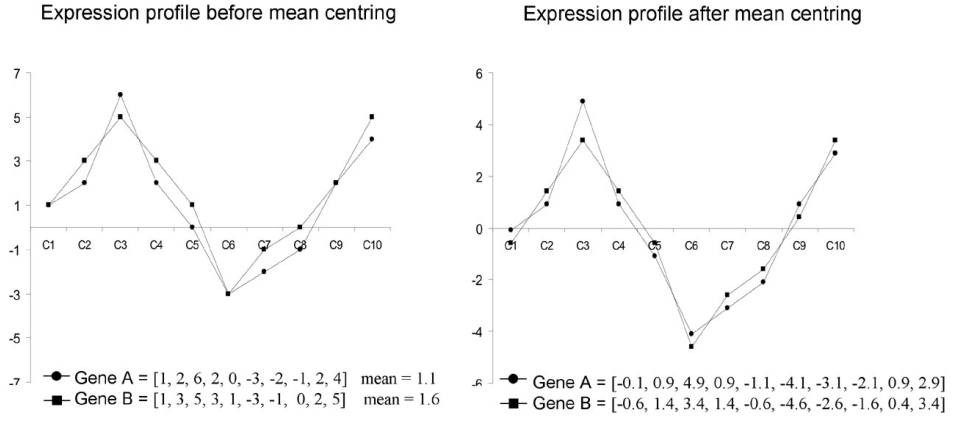

Figure 4. Expression profile before and after 'mean centring'. Note that after mean centring, the relative 'shapes' of the expression profiles are still maintained, but the magnitude changes. The graphs do not show the actual result of PCC analysis.

Consider two genes with expression profiles A = [*1*, *3*, *5*, *6*, *9*] and B=[*2*, *6*, *9*, *12*, *19*]. The PCC can be calculated as follows:

$\bar{a}$ = 4.8 and $\bar{b}$ = 9.6, the mean centred expression profiles become:

$$\bar{A} = [\text{-}3.8, \text{-}1.8, 0.2, 1.2, 4.2] \text{ and } \bar{B} = [\text{-}7.6, \text{-}3.6, \text{-}0.6, 2.4, 9.4]$$

Therefore,
$$PCC = \frac{\bar{A} \circ \bar{B}}{\left|\bar{A}\right|\left|\bar{B}\right|} = \frac{77.6}{\sqrt{36.8} \times \sqrt{165.2}} = 0.995$$

Where,

$$\bar{A} \circ \bar{B} = (\text{-}3.8 \times 7.6) + (\text{-}1.8 \times 3.6) + (0.2 \times 0.6) + (1.2 \times 2.4) + (4.2 \times 9.4) = 77.6$$

$$\left|\bar{A}\right| = \sqrt{(\text{-}3.8)^2 + (\text{-}1.8)^2 + (0.2)^2 + (1.2)^2 + (4.2)^2} = \sqrt{36.8}$$

$$\left|\bar{B}\right| = \sqrt{(\text{-}7.6)^2 + (\text{-}3.6)^2 + (\text{-}0.6)^2 + (2.4)^2 + (9.4)^2} = \sqrt{165.2}$$

Rank correlation coefficient

Rank correlation coefficient (RCC) is a distance measure that does not take into account the actual magnitude of the expression ratio in each condition, but takes into account the 'rank' of the expression ratio. For example, consider two genes A = [*2*, *3*, *9*, *15*, *8*] and B = [*2*, *7*, *15*, *25*, *13*]. When we consider the rank of the values for different conditions for gene A, we get the following:

2 (rank = 1) < 3 (rank = 2) < 8 (rank = 3) < 9 (rank = 4) < 15 (rank = 5) which is equivalent to A = [*1*, *2*, *4*, *5*, *3*].

Similarly, for gene B, we get the ranks for the values for the different conditions as:

2 (rank = 1) < 7 (rank = 2) < 13 (rank = 3) < 15 (rank = 4) < 25 (rank = 5), which is equivalent to B = [*1*, *2*, *4*, *5*, *3*].

Rank correlation coefficient is the PCC calculated on the expression profiles converted into their rank profiles. In the above case the two genes have exactly the same rank profile, thus rank correlation coefficient becomes 1. However, PCC is not applicable when two values within a rank profile are repeated. In this case, the rank correlation coefficient can be directly computed as:

$$D_{rank}(A,B) = 1 - 6 \times \sum_{i=1}^{n} \frac{d_i^{\,2}}{n(n^2-1)}$$

Where n is the number of conditions (dimension of the profile) and d_i is the difference between ranks for the two genes at condition i. An advantage of RCC is that it is not sensitive to outliers in the data.

Mutual information

A distance measure to compare genes whose profiles have been made discrete can be calculated using an entropy notion, called Shannon's entropy. This measure gives us a metric that is indicative of how much 'information' from the expression profile of one gene can be obtained to predict the behaviour of the other gene.

Consider the discrete expression profiles for two genes, $A = [1, 1, 0, 1, -1]$ and $B = [1, -1, 0, 1, -1]$. We know that at any condition, the values that have been made discrete can be 1, 0 or –1. Thus, the probability for each state to occur in the profile for the two genes can be computed as follows:

Genes	Probability			
	P(1)	*P(0)*	*P(-1)*	*P(1)+P(0)+P(-1)*
A	$\frac{3}{5}$ (3 occurrences in 5 conditions)	$\frac{1}{5}$ (1 occurrence in 5 conditions)	$\frac{1}{5}$ (1 occurrence in 5 conditions)	$\frac{(3+1+1)}{5}=1$
B	$\frac{2}{5}$ (2 occurrences in 5 conditions)	$\frac{1}{5}$ (1 occurrence in 5 conditions)	$\frac{2}{5}$ (2 occurrences in 5 conditions)	$\frac{(2+1+2)}{5}=1$

From this table, the Shannon's entropy for the genes can be calculated as:

$$H(gene) = -\sum_{i=1}^{3} P_i \times \log_2 P_i$$

Note that i runs from 1 to 3 because there are three possible states (1, 0 and –1).

$$H(A) = -1\times(\tfrac{3}{5}\times \log_2 \tfrac{3}{5} + \tfrac{1}{5}\times\log_2 \tfrac{1}{5} + \tfrac{1}{5}\times\log_2 \tfrac{1}{5}) = 1.371$$

$$H(B) = -1\times(\tfrac{2}{5}\times \log_2 \tfrac{2}{5} + \tfrac{1}{5}\times\log_2 \tfrac{1}{5} + \tfrac{2}{5}\times\log_2 \tfrac{2}{5}) = 1.522$$

The next step in our calculation is to consider how often gene A and gene B have the same state (1, 0, or -1) across given conditions. There are 9 possible pairwise combinations of states, and they are calculated for our example in the following manner:

P(A,B)	Occurrence
P(1,1)	$\frac{2}{5}$
P(1,0)	$\frac{0}{5}$
P(1,-1)	$\frac{1}{5}$

P(A,B)	Occurrence
P(0,1)	$\frac{0}{5}$
P(0,0)	$\frac{1}{5}$
P(0,-1)	$\frac{0}{5}$

P(A,B)	Occurrence
P(-1,1)	$\frac{0}{5}$
P(-1,0)	$\frac{0}{5}$
P(-1,-1)	$\frac{1}{5}$

The number of conditions in which both gene A and gene B have their values equal to 1 over all conditions is 2 out of 5 conditions, and so on.

Another parameter we will need to calculate mutual information is joint entropy H(A,B):

$$H(A,B) = -\sum_{i,j=1}^{3} P_{ij} \times \log_2 P_{ij}$$

when both *i* and *j* independently run from 1 to 3, corresponding to the three states (1, 0 and –1).

$$H(A,B) = -1 \times (2/5 \times \log_2 2/5 + 1/5 \times \log_2 1/5 + 1/5 \times \log_2 1/5 + 1/5 \times \log_2 1/5) = 1.923$$

For the above example, the mutual information between the two expression profiles, which provides a measure of the similarity between the two genes can be calculated as:

$$M(A,B) = H(A) + H(B) - H(A,B) = 1.371 + 1.522 - 1.923 = 0.970$$

In general, the higher the mutual information score, the more similar are the two profiles. However, precise state and consequently, interpretation of the observed score would depend on the number of conditions for which measurements were available. For our case of 5 conditions, the obtained score of 0.97 is high. But reader is advised to consult more specialised sources for understanding of states associated with mutual information distance measure (Shannon, 1949). One should note that a distance measure has to be chosen only after considering the data to be analysed and that there is no single distance measure that is appropriate for all types of data.

3.3 Clustering methods

One of the goals of microarray data analysis is to cluster genes or samples with similar expression profiles together, to make meaningful biological inference about the set of genes or samples. Clustering is one of the unsupervised approaches to classify data into groups of genes or samples with similar patterns that are characteristic to the group. Clustering methods can be hierarchical (grouping objects into clusters and specifying relationships among objects in a cluster, resembling a phylogenetic tree) or non-hierarchical (grouping

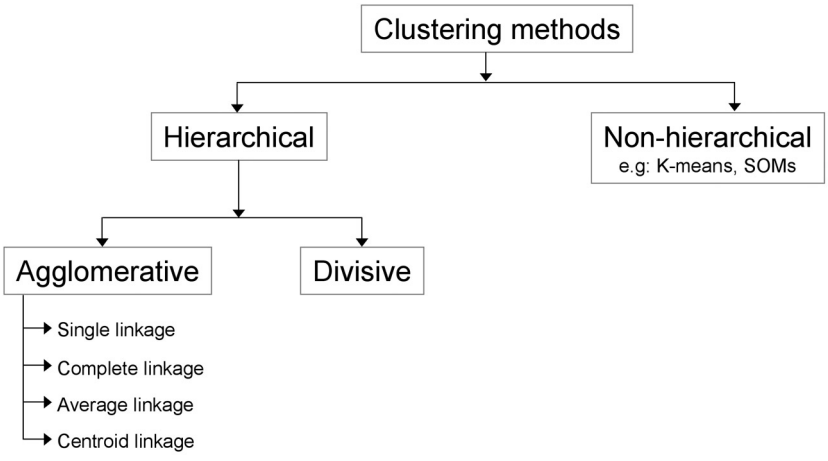

Figure 5. An overview of the different clustering methods.

into clusters without specifying relationships between objects in a cluster) as schematically represented in Figure 5. Remember, an object may refer to a gene or a sample, and a cluster refers to a set of objects that behave in a similar manner.

Hierarchical clustering

Hierarchical clustering may be agglomerative (starting with the assumption that each object is a cluster and grouping similar objects into bigger clusters) or divisive (starting from grouping all objects into one cluster and subsequently breaking the big cluster into smaller clusters with similar properties). The basic idea behind agglomerative and divisive hierarchical clustering is shown in Figure 6. There are many different types of clustering methods and a few commonly used ones are described below.

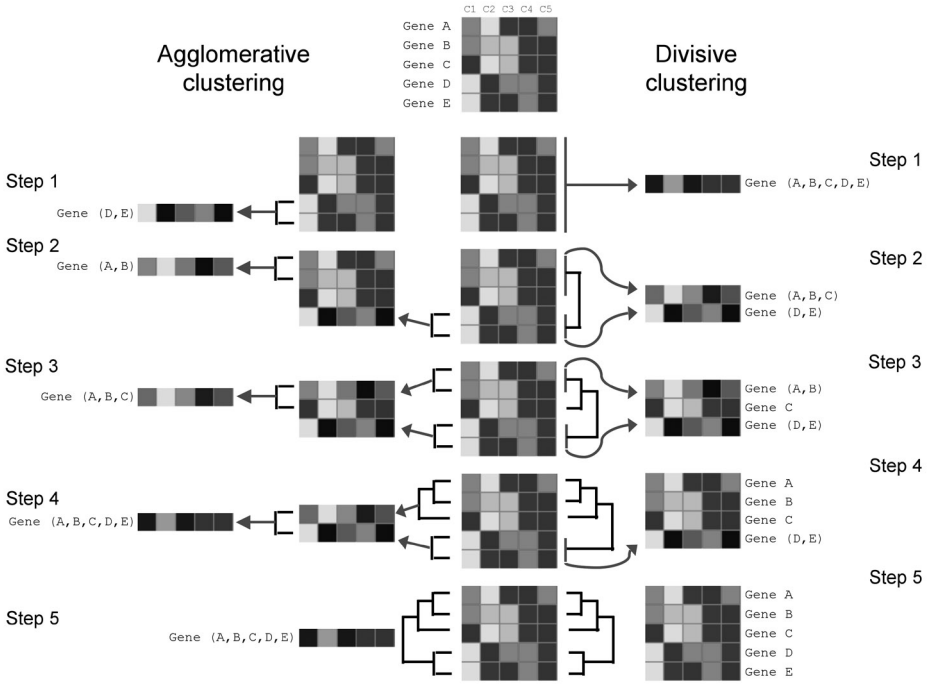

Figure 6. Schematic diagram showing the principle behind agglomerative and divisive clustering. The colour code represents the $\log_2$ (expression ratio), where red represents up-regulation, green represents down-regulation, and black represents no change in expression. In agglomerative clustering, genes that are similar to each other are grouped together, and an average expression profile is calculated for the group by using the average linkage algorithm. This step is performed iteratively until all genes are included into one cluster. In the case of divisive clustering, the whole set of genes is considered as a single cluster and is broken down iteratively into sub-clusters with similar expression profiles until each cluster contains only one gene. This information can be represented as a tree, where the terminal nodes represent genes and all branches represent different clusters. The distance from the branch point provides a measure of the distance between two objects. This image was adapted from Dopazo *et al*., (2001). Notice that the ordered matrix at the top is the actual product of either agglomerative or divisive clustering, and genes A to E are given in the final order for the simplicity of illustration; initially rows corresponding to genes A to E could be arranged in any order and it is the task of the methods to arrange them meaningfully. Colour figure at: http://www.mrc-lmb.cam.ac.uk/genomes/madanm/microarray/.

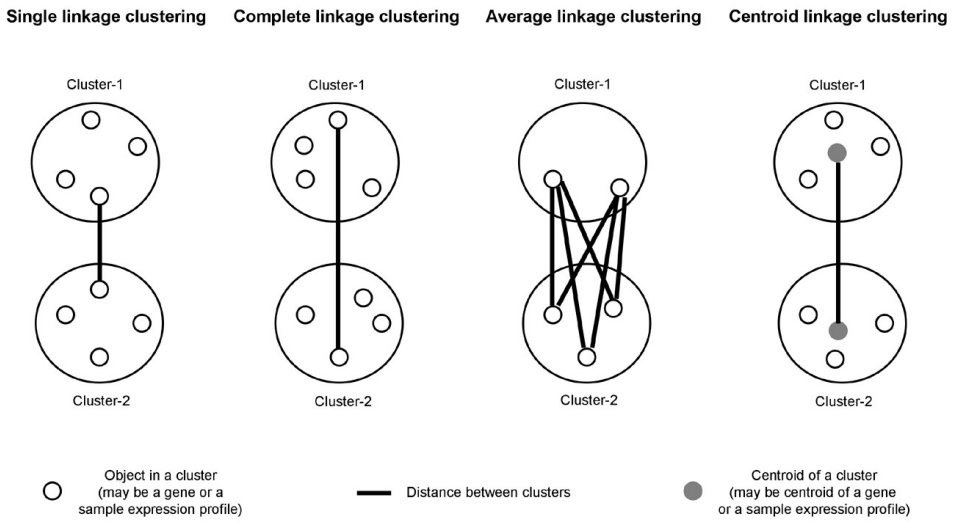

Figure 7. Different algorithms to find distance between two clusters.

Hierarchical clustering: agglomerative

In the case of a hierarchical agglomerative clustering, the objects are successively fused until all the objects are included. For a hierarchical agglomerative clustering procedure, each object is considered as a cluster. The first step is the calculation of pairwise distance measures for the objects to be clustered. Based on the pairwise distances between them, objects that are similar to each other are grouped into clusters. After this is done, pairwise distances between the clusters are re-calculated, and clusters that are similar are grouped together in an iterative manner until all the objects are included into a single cluster. This information can be represented as a dendrogram, where the distance from the branch point is indicative of the distance between the two clusters or objects.

Comparison of clusters with another cluster or an object can be carried out using four approaches (Figure 7).

Single linkage clustering (Minimum distance)

In single linkage clustering, distance between two clusters is calculated as the minimum distance between all possible pairs of objects, one from each cluster. This method has an advantage that it is insensitive to outliers. This method is also known as the nearest neighbour linkage.

Complete linkage clustering (Maximum distance)

In complete linkage clustering, distance between two clusters is calculated as the maximum distance between all possible pairs of objects, one from each cluster. The disadvantage of this method is that it is sensitive to outliers. This method is also known as the farthest neighbour linkage.

Average linkage clustering

In average linkage clustering, distance between two clusters is calculated as the average of distances between all possible pairs of objects in the two clusters.

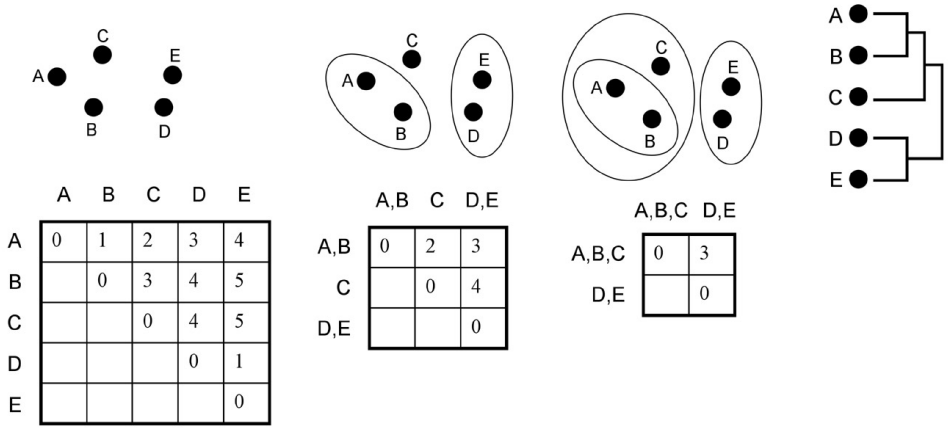

Figure 8. An example of a hierarchical clustering using single linkage algorithm. Consider five genes and the distances between them as shown in the table. In the first step, genes that are close to each other are grouped together and the distances are re-calculated using the single linkage algorithm. This procedure is repeated until all genes are grouped into one cluster. This information can be represented as a tree (shown to the right), where the distance from the branch point reflects the distance between genes or clusters. This image was adapted from Causton *et al*. (2003).

Centroid linkage clustering

In centroid linkage clustering, an average expression profile (called a centroid) is calculated in two steps. First, the mean in each dimension of the expression profiles is calculated for all objects in a cluster. Then, distance between the clusters is measured as the distance between the average expression profiles of the two clusters.

An example of the hierarchical agglomerative clustering using single linkage clustering is shown in Figure 8.

Hierarchical clustering: divisive

Hierarchical divisive clustering is the opposite of the agglomerative method, where the entire set of objects is considered as a single cluster and is broken down into two or more clusters that have similar expression profiles. After this is done, each cluster is considered separately and the divisive process is repeated iteratively until all objects have been separated into single objects. The division of objects into clusters on each iterative step may be decided upon by principal component analysis which determines a vector that separates given objects. This method is less popular than agglomerative clustering, but has successfully been used in the analysis of gene expression data by Alon *et al*. (1999).

Non-hierarchical clustering

One of the major criticisms of hierarchical clustering is that there is no compelling evidence that a hierarchical structure best suits grouping of the expression profiles. An alternative to this method is a non-hierarchical clustering, which requires predetermination of the number of clusters. Non-hierarchical clustering then groups existing objects into these predefined clusters rather than organizing them into a hierarchical structure.

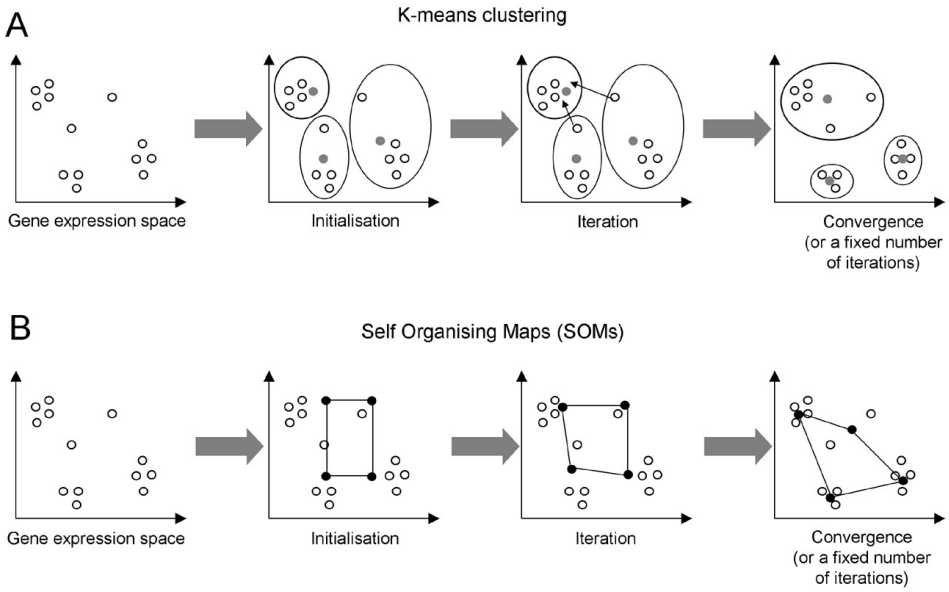

Figure 9. A: The principle behind K-means clustering. Objects are grouped into a predefined number of clusters during the initialization step. Centroid for each cluster is calculated, and objects are re-grouped depending on how close they are to available centroids. This step is performed iteratively until convergence or is performed for a fixed number of iterations to get final clusters of objects. B: The principle behind SOMs. During the initialization step, a grid of nodes is projected onto the expression space and each gene is assigned its closest node. Following this step, one gene is chosen at random and the assigned node is 'moved' towards it. The other nodes are moved towards this gene depending on how close they are to the selected gene. This step is performed iteratively until convergence or is performed for a fixed number of iterations to get a final map of nodes.

Non-hierarchical clustering: K-means

K-means is a popular non-hierarchical clustering method (Figure 9A). In K-means clustering, the first step is to arbitrarily group objects into a predetermined number of clusters. The number of clusters can be chosen randomly or estimated by first performing a hierarchical clustering of the data. Following this step, an average expression profile (centroid) is calculated for each cluster, this is called initialization. Next, individual objects are reattributed from one cluster to the other depending on which centroid is closer to the gene (or sample). This procedure of calculating the centroid for each cluster and re-grouping objects closer to available centroids is performed in an iterative manner for a fixed number of times, or until convergence (state when composition of clusters remains unaltered by further iterations). Typically, the number of iterations required to obtain stable clusters ranges from 20,000 to 100,000. However, there is no guarantee that the clusters will converge. This method has an advantage that it is scalable for large datasets.

Non-hierarchical clustering: Self Organizing Maps

Self Organizing Maps (SOMs) work in a manner similar to K-means clustering (Figure 9B). In K-means clustering, one chooses the number of clusters to fit the data, whereas with SOM the first step is to choose the number and orientation of the clusters with respect to each other. For example, a two-dimensional grid of 'nodes' (which may end up being clusters)

could be the starting point. The grid is projected onto the expression space, and each object is assigned a node that is nearest to it – this is called initialization. In the next step, a random object is chosen and the node (called a reference vector) which is in the 'neighbourhood' of the object is moved closer to it. The other nodes are moved to a small extent depending on how close they are to the object chosen. In successive iterations, with randomly chosen objects, the positions of the nodes are refined and the 'radius of neighbourhood' becomes confined. In this way, the grid of nodes (initially a two-dimensional grid) is deformed to fit the data. The advantage of this method, unlike K-means, is that SOM does not force the number of clusters to be equal to the number of starting nodes in the chosen grid. This is because some nodes may have no objects associated with them when the map is complete. Other advantages of SOM include providing information on the similarity between the nodes, and the ability of SOM to produce reliable results even with noisy data.

4. RELATING EXPRESSION DATA TO OTHER BIOLOGICAL INFORMATION

Gene expression profiles can be linked to external information to gain insight into biological processes and to make new discoveries. Some of the possible questions that can be addressed after analysing gene expression data will be discussed in this section.

4.1 Predicting binding sites

It is reasonable to assume that genes with similar expression profiles are regulated by the same set of transcription factors. If this happens to be the case, then genes that have similar expression profiles should have similar transcription factor binding sites upstream of the coding sequence in the DNA. Various research groups have exploited this assumption. Brazma *et al*. (1998) and others (Bussemaker *et al*., 2001; Conlon *et al*., 2003) have studied the occurrence of sequence patterns and discovered 'putative binding sites' in the promoter regions of genes that are co-expressed. The steps involved in such studies are the following: (1) Find a set of genes that have similar expression profiles. (2) Extract promoter sequences of the co-expressed genes. (3) Identify statistically over-represented sequence patterns. (4) Assess quality of the discovered pattern using statistical significance criteria.

4.2 Predicting protein interactions and protein functions

Integrating expression data with other external information, for example evolutionary conservation of proteins, have been used to predict interacting proteins, protein complexes, and protein function. Work by Ge *et al*. (2001) and Jansen and Gerstein (2000) have shown that genes with similar expression profiles are more likely to encode proteins that interact. When this information is combined with evolutionary conservation of proteins, meaningful predictions can be made. In a recent work by Teichmann and Madan Babu (2002) it was shown that proteins that are evolutionarily conserved in yeast and worm and that have similar expression profiles in both organisms tend to be a part of the same stable complex or interact physically. Noort *et al*. (2003) have also shown that the encoded proteins of conserved, co-expressed gene pairs are highly likely to be part of the same pathway. Such studies enable us to predict specific gene functions. The steps involved in such studies are the following: (1) Identify co-expressed genes in the two studied organisms. (2) Identify conserved (orthologous) proteins. (3) Find instances where conserved (orthologous) proteins are co-expressed in both organisms. (4) Map information on protein interaction or metabolic pathway available for one organism to predict interacting proteins or function of the proteins in the other organism.

4.3 Predicting functionally conserved modules

Genes that have similar expression profiles often have related functions. Instead of studying co-expressed pairs of genes, one can view sets of co-expressed genes that are known to interact as a functional module involved in a particular biological process (Madan Babu *et al*., 2004). This information, when integrated with the evolutionary conservation of proteins in more than two organisms, provides knowledge of the significance of the functional modules that have been conserved in evolution. Stuart *et al*. (2003) have addressed this issue in great detail and have identified one evolutionarily conserved functional module which belongs to an as yet unknown biological process. For other modules that are known to be involved in previously well-studied biological process, new module members were discovered by Stuart *et al*. (2003), providing clues about unknown candidates involved in the processes. The steps involved in such studies are similar to those discussed in the previous section. Instead of two organisms, one has to consider three or more organisms, and should also address other issues related to identifying orthologous proteins.

4.4 'Reverse-engineering' of gene regulatory networks

Gene expression data can also be used to infer regulatory relationships. This approach is known as reverse engineering of regulatory networks. Research by Segal *et al*. (2003) and Gardner *et al*. (2003) clearly highlights that we are now in a good position to use expression data to make predictions about the transcriptional regulators for a given gene or sets of genes. Segal *et al*. (2003) have developed a probabilistic model to identify modules of co-regulated genes, their transcriptional regulators and conditions that influence regulation. This new knowledge allowed them to generate further hypotheses, which are experimentally testable. Gardner *et al*. (2003) described a method to infer regulatory relationships, called NIR (Network Identification by multiple Regression), which uses non-linear differential equations to model regulatory networks. In this method, a model of connections between genes in a network is inferred from measurements of system dynamics (*i.e*. response of genes and proteins to perturbations).

5. WEBSITE REFERENCES, ACADEMIC SOFTWARE AND WEB SUPPLEMENT

5.1 Website references

Some websites that provide a reference to various aspects of microarrays are given below:

Portals:
http://ihome.cuhk.edu.hk/%7Eb400559/array.html
A comprehensive web portal on microarrays.

http://www.bioinformatics.vg/biolinks/bioinformatics/Microarrays.shtml
A web-portal on microarrays.

http://www.hgmp.mrc.ac.uk/GenomeWeb/nuc-genexp.html
A collection of gene expression and microarray links at the HGMP (Human Genome Mapping Project).

Table 2. List of software available for academic use.

	Software	URL
1	**Express Yourself** - An automated, online microarray data processing platform, where you can upload image files and carry out data processing and data analysis.	http://array.mbb.yale.edu/analysis/
2	**Expression Profiler** - A set of tools for clustering, analysis and visualization of gene expression and other genomic data. Tools in the Expression Profiler allow to perform cluster analysis, pattern discovery, pattern visualization, study and search Gene Ontology categories, generate sequence logos, extract regulatory sequences, study protein interactions, as well as to link analysis results to external databases.	http://ep.ebi.ac.uk/EP/
3	**Cluster & Treeview** - **Cluster** performs a variety of types of cluster analysis and other types of processing on large microarray datasets. Currently includes hierarchical clustering, self-organizing maps (SOMs), K-means clustering, principal component analysis. **Treeview** can be used to graphically browse results of clustering and other analyses from Cluster.	http://rana.lbl.gov/EisenSoftware.htm
4	**Xcluster** - cross platform software for analysing microarray data.	http://genetics.stanford.edu/~sherlock/cluster.html
5	**J-Express** - A Java implementation of hierarchical clustering, self organized maps, and principal component analysis, with several different viewing options and output formats.	http://www.microarrays.org/software.html
6	**TM4** - A package of Open Source software programs for microarray analysis	http://www.tigr.org/software/
7	**GeneXPress** - A visualization and analysis tool for gene expression data, integrating clustering, gene annotation, and sequence information.	http://genexpress.stanford.edu/
8	**GEPAS** - Gene Expression Pattern Analysis Suite.	http://gepas.bioinfo.cnio.es/tools.html
9	**GenMAPP** - A computer application designed to visualize gene expression data on maps representing biological pathways, and other biologically meaningful groups of genes.	http://www.genmapp.org/
10	**OligoArray** - An application which computes gene specific oligonucleotides for genome-scale oligonucleotide microarray construction.	http://berry.engin.umich.edu/oligoarray/

Tutorials:
http://www.ucl.ac.uk/oncology/MicroCore/HTML_resource/tut_frameset.htm
A website that provides a tutorial on the various aspects of microarray data analysis discussed in this chapter.

Software links:
http://genome-www5.stanford.edu/restech.html
This website provides a list of software available for microarray data analysis with a brief description of what the software does and the platform on which it can be run.

Microarray databases:
http://genome-www5.stanford.edu/
Stanford Microarray Database (SMD) – contains raw and normalized data from microarray experiments as well as their image files. SMD also provides interfaces for data retrieval, analysis and visualisation.

http://www.ebi.ac.uk/arrayexpress/
ArrayExpress – public repository for microarray data at the EMBL-EBI.

http://info.med.yale.edu/microarray/
Yale Microarray Database (YMD)

http://www.ncbi.nlm.nih.gov/geo/
Gene Expression Omnibus at the NCBI, NIH.

5.2 Software available for non-commercial use

The list of software provided here is by no means exhaustive. The readers are urged to visit the reference websites provided above to get a more comprehensive list of available programs (Table 2).

5.3 Supplementary material on the web

The web-supplement is available at:
http://www.mrc-lmb.cam.ac.uk/genomes/madanm/microarray/

It has PERL scripts to calculate the following statistics:

1. Euclidean distance.
2. Pearson correlation coefficient.
3. Rank correlation coefficient.

Expression datasets for the yeast genome from Cho *et al*. (1998) and Spellman *et al*. (1998) are also provided.

Acknowledgements

I would like to thank Siarhei Maslau, Janki Shah and the others in the group for reading the chapter. I would also like to acknowledge the Medical Research Council, Cambridge Commonwealth Trust and Trinity College, Cambridge for financial support.

References

Alon, U., Barkai, N., Notterman, D.A., Gish, K., Ybarra, S., Mack, D., and Levine, A.J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proc. Natl. Acad. Sci. USA *96*, 6745-6750.

Brazma, A., Jonassen, I., Vilo, J., and Ukkonen, E. (1998). Predicting gene regulatory elements in silico on a genomic scale. Genome Res. *8*, 1202-1215.

Bussemaker, H. J., Li, H., and Siggia, E. D. (2001). Regulatory element detection using correlation with expression. Nat. Genet. *27*, 167-171.

Causton, H., Quackenbush, J., and Brazma, A. (2003). Microarray Gene Expression Data Analysis: A Beginner's Guide. (UK: Blackwell Science)

Cho, R. J., Campbell, M. J., Winzeler, E. A., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T. G., Gabrielian, A. E., Landsman, D., Lockhart, D. J., and Davis, R. W. (1998). A genome-wide transcriptional analysis of the mitotic cell cycle. Mol. Cell. *2*, 65-73.

Conlon, E. M., Liu, X. S., Lieb, J. D., and Liu, J. S. (2003). Integrating regulatory motif discovery and genome-wide expression analysis. Proc. Natl. Acad. Sci. USA *100*, 3339-3344.

Dopazo, J., Zanders, E., Dragoni, I., Amphlett, G., and Falciani, F. (2001). Methods and approaches in the analysis of gene expression data. J. Immunol. Meth. *250*, 93-112.

Eisen, M. B., Spellman, P. T., Brown, P. O., and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. Proc. Natl. Acad. Sci. USA *95*, 14863-14868.

Gardner, T. S., di Bernardo, D., Lorenz, D., and Collins, J. J. (2003). Inferring genetic networks and identifying compound mode of action via expression profiling. Science *301*, 102-105.

Ge, H., Liu, Z., Church, G. M., and Vidal, M. (2001). Correlation between transcriptome and interactome mapping data from Saccharomyces cerevisiae. Nat. Genet. *29*, 482-486.

Jansen, R., and Gerstein, M. (2000). Analysis of the yeast transcriptome with structural and functional categories: characterizing highly expressed proteins. Nucl. Acids Res. *28*, 1481-1488.

Luscombe, N. M., Royce, T. E., Bertone, P., Echols, N., Horak, C. E., Chang, J. T., Snyder, M., and Gerstein, M. (2003). ExpressYourself: A modular platform for processing and visualizing microarray data. Nucl. Acids Res. *31*, 3477-3482.

Madan Babu, M., Luscombe, N., Aravind, L., Gerstein, M., Teichmann, S.A. (2004). Structure and evolution of transcriptional regulatory networks. Curr. Opin. Struct. Biol. *14*, in press.

Quackenbush, J. (2002). Microarray data normalization and transformation. Nat. Genet. *32*, 496-501.

Segal, E., Shapira, M., Regev, A., Pe'er, D., Botstein, D., Koller, D., and Friedman, N. (2003). Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. Nat. Genet. *34*, 166-176.

Shannon, C. (1949). A mathematical theory of communication. Bell. Sys. Tech. J. *17*, 379-423; 623-656.

Spellman, P. T., Sherlock, G., Zhang, M. Q., Iyer, V. R., Anders, K., Eisen, M. B., Brown, P. O., Botstein, D., and Futcher, B. (1998). Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization. Mol. Biol. Cell *9*, 3273-3297.

Stuart, J. M., Segal, E., Koller, D., and Kim, S. K. (2003). A gene-coexpression network for global discovery of conserved genetic modules. Science *302*, 249-255.

Teichmann, S. A., and Madan Babu, M. (2002). Conservation of gene co-regulation in prokaryotes and eukaryotes. Trends Biotechnol. *20*, 407-410; discussion 410.

van Noort, V., Snel, B., and Huynen, M. A. (2003). Predicting gene function by conserved co-expression. Trends Genet. *19*, 238-242.