In *Functional Plant Genomics*, JF Morot-Gaudry, P Lea, JF Briat ed.

Statistical Design and the Analysis of Gene Expression Determined by Independent Component Analysis

Alessandra Riva [a, *], Anne-Sophie Carpentier [a], Bruno Torrésani [b], Alain Hénaut [a]

[a] Laboratoire Génome et Informatique UMR 8116 Tour Evry2, 523 Place des Terrasses, 91034 Evry, FRANCE

[b] LATP, CMI, Université de Provence, 39 rue Joliot-Curie, 13453 Marseille Cedex 13, FRANCE

* To whom correspondence should be addressed.

Tel: (+33) 1-60-87-38-63 Fax: (+33) 1-60-87-38-97

E-mail: gucki@genopole.cnrs.fr

1 First Part

Expression data experiments create a large amount of quantitative data and the biologist is not necessarily used to or trained for such a situation. Also, the number of methods proposed for their analysis is enormous and is increasing, still. However, all these methods can be grouped into a few, fundamental types. The first part of this chapter aims to explain their nature and their principles, whilst the second part considers the important aspects in the experimental design, based on the analysis methods discussed.

Note that we talk almost exclusively about the transcriptome. However, all which is said can also be applied to the proteome and the metabolome, *mutatis mutandis*.

1.1 The data used

Throughout this chapter, we will refer to a set of experimental data, which stem from experiments on the sulphur metabolism of *B. subtilis* (Sekowska et al., 2001). These data are freely available at http://195.221.65.10:1234/~carpenti/ .

The aim of these experiments was to identify the genes differentially expressed when the bacteria are grown with methionine (“met”) or methyl-thioribose (“mtr”) as sulphur source. The experiments followed a fully crossed factorial design with four factors (sulphur source, day of experiment, type of protocol (here: amount of RNA) used and duplicate of each spot). The data (raw levels of expression) were gathered in an array of 4107 rows (all *B. subtilis* genes) and 16 columns (experimental conditions). The minimum value was 213, the maximum value 13455, with two thirds of the data having a value below 800. Note that each factor has only two states: all factors are binary (see Figure 1).

1.2 The data table and some preliminary considerations and manipulations

The reader may be surprised to find that we do not talk about taking the ratio, a rather popular pre-processing step. We refer the reader to the excellent work of Kerr and Churchill (2001) who discuss the issue in a very clear and lucid manner.

1.2.1 The raw data

Surprising as it may seem, you may find yourself in the situation of not having access to the “real” or truly “raw” data. You are given a set of data, where the machine has replaced all values below a certain threshold with one or very few arbitrary values. In other words, it has subtracted the background noise for you.

This operation, a translation, does not in itself pose a problem, but will complicate things when analyzing your data. Firstly, your data will contain a large number of very small or zero values and this is a problem, if you decide to take the log, something frequently done and described below.

Secondly, the data distribution will be far from being Gaussian (or just unimodal), a fundamental pre-requisite for the analysis of the microarray data. The only solution to this dilemma is to try to “restore” the Gaussian distribution by replacing the smallest values with random values (see Chiappetta et al., 2004).

As for the usefulness of eliminating the background noise in general, please refer to section 1.2.3.

1.2.2 Missing values

These pose a problem, as many data analysis methods require full sets of data. The missing values have generally two possible origins: (i) the microarray contains a defect resulting in the impossibility of taking a reading or (ii) the machine eliminates the measurement as the value is very close to the noise level (in this case it would be advisable to change the setup of the machine). The most radical solution is to eliminate the genes with missing entries, which is obviously far from ideal. A more moderate solution is to fill in the gaps with estimate values. The easiest is to use the row average; the two most common methods, however, are:

a) looking with whom the “missing gene” associates with in the other experimental conditions, i.e. determining that gene’s “neighbours”, then presuming that in the missing experiment this gene still associates with them and filling the gap with the median value (a method known under “K-nearest neighbours”),

b) variations around the Principle Component Analysis (described below). Examples here are the Singular Value Decomposition and Bayesian Principal Component Analysis (Oba et al., 2003).

The interested reader is referred to the works of Ouyang et al. (2004), Kim et al. (2004) and Zhou et al. (2003) for the comparison of some currently used estimation methods.

1.2.3 The correction of the background noise on the membranes, glass plates or silicon chips

Researchers often proceed to the correction of the background noise, before starting with the analysis of their data. A lot of effort is devoted to this, but recent studies suggest that the problem tends to be given too much importance and that the time devoted is not justified (see for example Chen et al., 2004). There is also a second aspect to be considered, namely that finding a reliable method to correcting background noise is not easy. As Lawrence et al. (2004) point out, the human component, difficult to quantify and correct, plays an important role. Also, the assumption that the background level is consistent between the DNA spot and the surrounding space, frequently used for background quantification, is not correct (Konishi, 2004). Using "designated" household genes for the background determination is in itself a good idea, but finding out who the household genes are, is posing problems (Stoyanova et al., 2004).

Regarding Affymetrix's GeneChips, the common practice of subtracting the mismatch (MM) probe intensities from the perfect match (PM) ones is "unjustifiable", according to Sasik et al. (2002), as the target sequence hybridizes not only with the PM but also with the MM probe.

A useful way to assess the utility of an anti-noise measure taken is to check on the change of the *eigenvalue* of the first axis in a PCA (should increase) or the F value in an ANOVA (should increase), discussed below.

1.2.4 N-dimensional graphs, translation and normalization

It is natural to want to plot the data in a **graph**. Each gene (*at least normally you talk about genes rather than EST or similar*) has as many coordinates as there are experimental conditions. If we have N experimental conditions, each gene will have N coordinates. To represent the data graphically, we will have a reference frame with N axes, each axis corresponding to one experimental condition. We will obtain one (and only one!) graph, with N dimensions, and with as many points as there are genes. These points form a "cloud". However, as we are not good at coping with drawings having more than two dimensions (three still works well on a computer screen), we are obliged to examine this cloud in little "portions" of two dimensions at a time: we look at one experimental condition versus another). In other words, we choose two experimental conditions and look at the projection of the cloud on this particular plane.

Note that when you draw a graph by hand, you will automatically try to maximize the use of the paper: you look at the minimum and maximum values for both variables, and define the scale accordingly. The machine will do the same. In both cases, the data are transformed through a change of variable: 1 cm on the graph corresponds to X units of the original variable (a linear transformation).

Translation. This is an operation which in itself does not pose a problem, as one is interested in the relative position of the points to each other: the aim is to find the points that are far away relative to the main body of the cloud, which means that the reference frame used to look at the cloud doesn't really have much importance. However, the translation may create complications when it consists in bringing a lot of the values close to zero followed by taking the log of the data, something discussed in the next section.

Normalization. Note that drawing a graph or letting a spreadsheet (like MS-Excel) draw the graph, implicitly presumes that the sum of the signal does not change in function of the experimental conditions; one allows the data to be normalized. By doing this, one has presumed that the total of the signal in each column is the same: total signal of column 1 = total signal of column 2. This is justified when three conditions are fulfilled: firstly, more than 90% of the genes don't care about the experiment, i.e. do not change expression in function of the different experimental conditions; in that case one can indeed presume that the total quantity of cDNA (and therefore of the mRNA) is the same. Secondly, the number of genes analyzed has to be large: this is a way to make sure that the majority of the genes does not change expression in function of the different experimental conditions)[1]. Thirdly, the overall intensity change of up- and down-regulated genes is similar. The three conditions are fulfilled in our example, but they would not be in, say, the temporal analysis of mRNA decay. The reader is referred to the work of Stoyanova et al. (2004) for some interesting considerations on this subject, as well as to the work of Zhao et al. (2005) who propose a normalization procedure for data not fulfilling the above conditions.

[1] This maxim should be kept in mind when tempted to work with "specialized" microarrays.

Instead of just looking at the minimal and maximal values in order to best represent the graph, it is advisable to calculate the means and variance for each experimental condition: In the first case the estimates are based on 2 points only (min and max) per experimental condition, in the second case the estimate is made using all points. If these are numerous, the result is more stable.

1.3 Graphic exploration

1.3.1 Preliminary considerations

As we said, we are forced to take the columns 2 by 2, which means that we will look at PROJECTIONS OF OUR SINGLE CLOUD ON THE DIFFERENT PLANES.

What are we looking for? Presuming that the three above-mentioned conditions are fulfilled, at least 90% of the genes analyzed will not change expression under the different experimental conditions. This means that on the graph one would see them all lying on one line, if it wasn't for the noise: the noise is responsible for making those points look more like a cigar which is the wider the more noise there is. The remaining 10% of the genes will change expression; they have an atypical behaviour and will not lie on the line (the cigar) but be apart. These genes that are apart from the main body of the cloud are the ones the biologist is interested in. Note that having the 90% of the point lying on a line is an ideal case, the "cigar' being the reality; so one tries to find that line (which describes 90% of the genes) somehow.

How do we describe those 90% of the data? How do we determine the line? Various options are available:

a) One can try to draw it by hand.

b) Calculate the linear regression. This is not such a good idea as there are two lines of regression (x-axis versus y-axis and vice versa) and they are not identical except when all the points lie on the same line.

c) Use methods that are more sophisticated.

The methods all presume that the cloud follows a Gaussian distribution, or at least an unimodal and symmetrical one. They also need some preprocessing of the data, for two reasons:

a) The fact that the data often consist of a very large amount of small values and a few, extreme points, something which affects most data analysis techniques strongly (Chiappetta et al., 2004).

b) Some effects being studied may have a multiplicative behaviour.

To solve the first of these problems, taking the log, the square (or cubic or fifth etc) root or the hyperbolic tangent are all possible and generally accepted methods (see Figure 2), whilst for the second problem taking the log is preferable (Chiappetta et al., 2004; Hoyle et al., 2002; Thygesen et al., 2004; Tusher et al., 2001).

As we mentioned in the section above, the reference frame used to look at the cloud doesn't really matter and making a simple translation does not in itself pose a problem. One does need to be careful, though: making a simple translation is indeed no problem, nor is taking the log. However, when executing both operations, one needs to be cautious: if the translation consists in bringing a lot of the values close to zero, taking the log afterwards will create a distortion in the cloud of points: one has just created a whole package of data with values going towards minus infinity. This means that in trying to take care of the problem of the points at the far right (few points with very large values) by taking the log, the result is worse than the starting point. Note that when executing the two operations in the inverse order (first log, then translation) the problem is not created.

Some general considerations about the graphs. We come back to the graphs, which are just many projections on different planes of ONE cloud. A brief look at the general shape of each cloud projection is worthwhile. If a cloud resembles a fat cigar, a lot of genes have considerably changed expression. If, on the other hand, the cloud resembles a line, the great majority has not changed expression (see Figure 3 for two examples). The "cigar" may also be bent or twisted, which means that the readings were taken outside the linear range of the machine. The first and obvious recommendation is to make sure that at the moment of taking the readings, the scanning settings are correct, which they are often not (Stoyanova et al., 2004). The second is to check that one is not just working at one extreme of the linear range. If that is the case, a change of concentration in the hybridization solution is a good option. If the entire linear range is taken up, two solutions can be proposed: using two different voltage settings for the photomultiplier or using different exposure times, when working with radioactively labelled samples. Algorithms for subsequently combining the different readings are readily available (see for example Querec et al., 2004; Lyng et al., 2004). The article by Lyng et al. (2004) shows the relationship between the type of incorrect setting and the resulting cloud shape.

Numerous authors propose "remedies" if the above suggestions prove impossible to follow, but none will give you the "perfect" data back you would have had if the experiments had been executed correctly.

We can be faced with a problem: taking the columns 2 by 2, the number or graphs increases very rapidly when increasing the number of experimental conditions. In our example we have 16 columns which means we need to look at 16x15/2 i.e. 120 graphs. Evaluating them all in detail becomes a bit tedious.

Thus, we need to find ways to reduce the number of graphs we have to examine. To do this, we need to decide, from which point of view we want to look at the cloud, which has to be translated into a mathematical criterion. This implies that there will be a change (rotation) of the reference frame.

1.3.2 By hand (with a spreadsheet)

With "by hand", we refer to the fact that the calculations are extremely simple. As the calculations have to be repeated for each gene, though, the number of calculations is such as to make handing the job over to a spreadsheet a practical alternative.

The only reasonable option to reduce the 120 little pictures means concentrating on the expression changes caused by each single factor being studied, in our case four. For this we calculate the mean expression for each gene; this will be the x-value. Then, for a given factor like sulphur, we calculate the sum of all met values and subtract from it the sum of all mtr values, which gives us the y-value.

This is done for all four factors. Note that we have changed the reference frame; this calculation, which is done instinctively by hand, can be formalized and done via a matrix, called "mixing matrix": it allows to change from the old reference frame to the new one and is shown in Table 1.

We obtain four graphs, one for each factor; we then look for genes that are far away from the main body of the cloud. Figure 4 shows the graph obtained for the factor sulphur. Executing this operation, each experimental condition is given the same weight and the criterion chosen to look at the cloud is "one factor per graph".

1.3.3 PCA

A more sophisticated approach is the Principal Component Analysis. Pearson first introduced it in 1901. The reader may refer to the work by Stoyanova et al. (2004) for a comprehensive introduction to the subject and to Kendall et al. (1983) for a technical presentation.

Here, the criterion chosen to look at the cloud is to maximise the variances along the axes of the reference frame. There are numerous softwares that do this job and which supply us with the mixing matrix, which in PCA's case is called *eigenvector matrix*, shown in Table 1. This matrix allows us to change from the old reference frame to the new one; it gives us for each of the new axes (in the table: the columns) the coefficient with which we have to multiply each gene's value in a given experimental condition (in the table: the lines) in order to obtain its new coordinates (see legend of Table 1).

The *eigenvector matrix* gives us also another information: the variance or *eigenvalue* for each axis, expressed in percentage. This provides an indication of the cloud's dispersion along the axis (the bigger the value, the more the genes are dispersed along this axis). The fundamental idea is that if the dispersion is great, the image is easier to interpret than if all the points were packed together. If an experimental factor influences the expression of some genes, the factor will contribute to the dispersion of the cloud and may coincide with one of the axes determined by PCA.

The *eigenvector matrix* gives a wealth of information. Looking at our matrix, we see that for the first axis all the sixteen coefficients have basically the same value; this means that for the first axis, all experimental conditions have the same weight, in other words, the first axis gives us the total expression of each gene, just like with a spreadsheet. This observation is generally true (see Stoyanova et al., 2004).

In each of the other columns (axes), the experimental conditions can be grouped together according to the sign of their coefficient (positive or negative). For some axes, this coincides with a separation of the two states of a factor. In our case, axis two separates well the two protocols (1 microgram RNA -all values are negative- and 10 micrograms RNA -all values are positive-); axis three separates the day (A and B), axis five the sulphur source (met and mtr) and axis seven the two spots (a and b). Other axes, on the other hand correspond to combinations of the experimental conditions, whose interpretation is not evident: axis four is an example. It singles out the ribosomal proteins; a biologically speaking coherent result, which is waiting for an interpretation. This is something frequently found when analyzing microarray data.

The *eigenvector matrix* deserves a little more attention: the values it contains can be looked at from a different point of view. If we take up our example, each line represents an experimental condition and the values in the 16 columns for a given line give us the position of that particular experimental condition in the 16-dimensional space. (To be precise, each value has to be multiplied with the root of the variance of that axis, in order to obtain the coordinate.) This means that instead of looking at the *eigenvector matrix*, we can look at the different projections of the experimental conditions in order to figure out which axes separate well the different states of our factors. Once we have established which planes deserve being examined in details, we come back to the projections of the cloud on these planes and pinpoint those genes, which are far away from the main body of the cloud. Figure 5 shows the cloud projection on the plane formed by axis one versus axis five.

Note: the normalization of the data is an integral part of PCA.

To resume, with PCA the experimental conditions are not given the same weight (contrary to a spreadsheet) and the criterion chosen to look at the cloud is to maximise the variances along the axes.

1.3.4 ICA

"ICA tries to find a linear representation of non-Gaussian data so that the components (or factors, or sources) are statistically independent, or as independent as possible" (Hyvärinen and Oja, 2000).

This search for statistical independence is generally very difficult and therefore an approximation is made: One looks for the directions that maximize the criterion of non-Gaussian distribution. As "non-Gaussian" is a "non-property", numerous possibilities exist for defining such a distribution. One criterion that seems to work quite well is to look for distributions with a positive kurtosis (distributions with "heavy tails"). ICA can be seen as a close relative of PCA. Whilst PCA looks at which directions maximize the variance, ICA approaches the question of finding genes with an "atypical behaviour" more directly, by defining "atypical" as "following a non-Gaussian distribution". The new reference frame will maximise the criterion of "non-Gaussianity". With this criterion, one increases the weight of points that had only small deviations from the main body of the cloud and thus allows them to be detected as potentially interesting.

A latent difficulty with ICA is that there is no analytical solution (contrary to PCA): we look for the numerical solutions. There is the danger that the algorithm finds a direction with a solution, but that this direction is not the best solution in absolute terms: the algorithm gets stuck with a local maximum (Chiappetta et al., 2004). Launching ICA a large number of times, typically100, circumvents this problem and only those directions or solutions that have been most frequently found are kept. As with PCA we have a mixing matrix that allows us to change from the old reference frame to the new one. Again, the different experimental conditions do not have the same weight; the weight attributed, though, varies slightly from PCA. Once we have determined the axes, the procedure is the same as with PCA. Figure 6 shows the cloud projection on the plane that separates well the sulphur sources.

The applications of ICA in microarray analysis include the identification of groups of genes implicated in cancer, the study of the cell cycle (Liebermeister, 2002; Martoglio, 2002), the identification of genes that are potentially co-regulated (Chiappetta et al., 2004), as well as metabolome studies (Scholz et al., 2004). Chiappetta et al. (2004) and Carpentier et al. (2004) have applied both PCA and ICA to the sulphur metabolism data and shown that the two methods perform similarly well, with ICA slightly outperforming PCA.

1.3.5 A brief remark

We have said that whilst a spreadsheet attributes to each experimental condition the same weight, PCA and ICA do not (Figure 7 shows a comparison between the three methods from this point of view).

The spreadsheet simply calculates the mean expression. This choice is not optimal when certain experimental conditions contain more information than others. Ideally, the weight attributed should be proportional to the information contained. PCA is a good choice when the signal follows a Gaussian distribution, whilst ICA imposes itself when the distribution is non-Gaussian.

You might wonder what happens if you use, say, PCA on data that follow a non-Gaussian distribution. The answer is that you are likely to miss out on potentially interesting genes; you do not, however, risk finding "wrong" genes. Using more than one tool amounts to examining the cloud from different angles; the results obtained with the different tools are complementary.

1.4 Statistical tests

1.4.1 Preliminary considerations

Our experience shows that some confusion reigns regarding the statistical tools in general and their application to microarrays in particular. Hence this rather long introductory section.

When approaching microarray data from a statistical point of view, people seem to worry a lot about the fact that the data are "relative" and whether they should or not take ratios.

Microarrays give us "relative data": The interesting information regarding a gene is "relative" as one compares the expression of a gene under condition A with that of the same gene under condition B. Microarray technology is quite recent; however, dealing with relative data is not and taking the ratio results in a reduction and a falsification of the information offered (Kerr and Churchill, 2001). It is Fisher who first tackled and solved the problem at the very beginning of the 20th century, resulting in ANOVA. For a more detailed discussion of this issue, the reader is referred to the work of Kerr and Churchill (2001). At about the same time, Gosset ("Student") came up with the t-test as a solution to the problem.

Statistics help us to answer the question whether the expression differences observed are real. The answer is given indirectly, as the statistical tools give us the probability of having a false positive. A false positive is a gene whose expression difference surpasses by chance a threshold value, which has been fixed in advance. "By chance" means that if the experiment were repeated, you would not find again such a large expression change.

The statistical analysis is used to evaluate the probable percentage of false positives beyond a given threshold value: 40 genes will surpass by chance the threshold value of 1% if the experiment was carried out on 4000 genes.

The estimation of the number of false positives is only the first step. Beyond the threshold value we not only find false positives, but also genes whose expression change is "real" (we would find it again if the experiment were repeated). The key information is the proportion of false positives on the total, because it measures the risk of being on the wrong track when deciding to work on one of the genes from this group (Benjamini et al., 1995). One generally chooses the threshold in order to have less than 5% of false positives in the group. Take for example an experiment carried out on 4000 genes with 80 lying beyond the threshold of 0.1%. As there are on average 4 false positives beyond the 0.1% threshold (4000 x 0.001), the percentage of false positives is 4 / 80, or 5% of the selected genes.

The literature sometimes refers to the Bonferroni correction. This correction is not pertinent for the analysis of microarray data, as it is too restrictive.

The numerical criterion used in the statistical tests is always the ratio between the deviations observed for the factor of interest (the signal) and the deviations due to all the causes one chooses to ignore (the noise). The statistical tests differ from each other in the way they define the noise and the probability function they use to estimate the probability of false positives. In the past, the function used was the Gaussian. Nowadays one tends to employ the probability function, estimated on the data using permutations (see Tusher et al., 2001)

1.4.2 ANOVA

ANOVA is a tool that allows us to analyze simultaneously the effect of more than one factor on a variable, in our case the genes' expression levels. The method is based on the calculation of the sum of squares, degrees of freedom, mean square (short for mean square deviation from the mean) and F-statistics2 (see Zar (1998) for details). As we use ANOVA in a somewhat reductive manner, the reader may refer to the work of Zar (1998) for a full appreciation and pedagogic explanation of the possibilities offered.

Various quantities are used simultaneously in order to decide whether the expression of a gene varies significantly for the factor of interest.

1. V1, the variance for the total of the observations made on the gene,
2. V2, the variance for the observations made for the factor of interest,
3. V3, the variance for the observations made for those factors whose influence one wishes to subtract.

[2] Sometimes referred to as F-test

The signal is equal to V2, the noise to V1 – (V2 + V3). The possibility to calculate the term V3 is a particularity of ANOVA and it allows a finer control of the noise's composition. In our example, V3 corresponds to the expression change caused by the day, the duplicate and the protocol used. The noise encompasses all which causes the difference between the actual expression level and the sum of the expression levels of the four factors.

In the case of the sulphur metabolism data, the equation used for each gene is the following:

$$Y_{ijkl} = \mu + S_i + J_j + C_k + D_l + \varepsilon_{ijkl}$$

Where

Y_{ijkl} is the total expression level measured

μ is the mean of the expression levels measured for the gene

S_i, J_j, C_k and D_l are, respectively, the effects of sulphur source i, experiment day j, protocol used k and duplicate l on the expression level

ε_{ijkl} is the residual error.

Note that the residual error ε_{ijkl} encompasses all interactions: between two factors (6), between three factors (4) and between four factors (1). The interactions are grouped together under "error" for the following reason: it is information with which we cannot work, unless we have a very precise idea of the nature of the interaction (linear, sinusoidal or other).

The *F-test* is calculated in the following manner: *F* = "mean square of the sulphur source"/"mean square of the residual error". We are interested in genes that posses a high *F*-value (*p-value*) for the factor sulphur source. The calculations are done for all genes and the results can be represented in a graphical form. The variance of the factor of interest is given on the x-axis, the *p-value* on the y-axis. The *p-value* is used to calculate how many false positives will lie below a chosen threshold value (see Figure 8).

Note that we are not interested, whether the expression levels of the thus identified genes also vary in function of the other factors. One does not preclude the other and has no impact on our analysis.

ANOVA has some advantages when the experimental factors are not binary; in that case, it basically becomes the only tool which is easy to use.

1.4.3 Paired *t-test*

We have said that ANOVA quantifies the contribution given by each factor to the total expression of a gene, permitting us to isolate the contribution of our factor of interest. The paired *t-test* also allows this, but the approach is different, and we can only use it for binary factors. The paired *t-test* eliminates the influence of all the factors we are not interested in by calculating the difference between pairs of values. The members of each pair differ from each other only with respect to the factor of interest (state 1 vs. state 2), all other experimental conditions being equal.

For example, we calculate the difference between the value obtained on met with the value obtained on mtr, both obtained on day A, with one microgram mRNA and spot a. Then we calculate the difference of met versus mtr on day B, with one microgram mRNA and spot a and so forth. This is done for each gene and we thus obtain 8 comparisons, or differences per gene. V1 is calculated on these 8 comparisons, the term V3 has disappeared.

However, as the paired *t-test* takes pairs of "similar conditions", systemic biases due to, e.g. "day" or "duplicate", are eliminated, therefore still allowing for a reasonable estimation of the error.

1.4.4 t-test

The t-test corresponds to an ANOVA with one factor and is the least favourable option. The t-test only considers the expression difference due to one factor, ignoring that there are pairs of measurements which have more or less in common (like the day, protocol and spot), unlike ANOVA and the paired t-test. Thus, we cannot separate the contribution made by our factor of interest from the contribution made by the other factors and the interaction between them; the expression difference due to our factor risks being drowned by the rest.

In terms of V1, V2 and V3: V1 is calculated on the total of the 16 observations made (as with ANOVA), but as the term V3 has disappeared, the noise risks being much larger.

1.4.5 In conclusion

The biggest difficulty is to estimate the noise with accuracy. The best solution is to repeat the experiment a large number of times. As this is not always possible, statisticians try to improve the estimation of the noise by working on groups of genes having more or less the same level of noise. A considerable amount of literature is dedicated to this effort. Numerous are the solutions proposed, none is perfect. Generally, the grouping is done a posteriori, after a first estimation of the noise for all the genes separately. One speaks in this case of a Bayesian approach. The reader is referred to the work of Neuhäuser and Senske (2004) for an introduction into the subject and to the work of Kutalik et al. (2004) for the comparison of some methods proposed.

Regarding the three approaches discussed above: Table 2 shows the measurements obtained for ytmJ and the results obtained from ANOVA, the paired t-test and the t-test. It shows that though ANOVA and the paired t-test both identify the gene as interesting, the t-test results inconclusive. The observation made on this particular example can be generalized. Table 3 shows a comparison of the number of genes detected by the three methods. Although ANOVA detects the highest number of genes, the paired t-test performs comparably well, whilst the t-test lags far behind.

1.5 Graphic exploration and statistical tests in comparison

We have chosen to talk about the typical representatives of the two approaches. They are not the only ones proposed in the literature: the number of tools is continuously increasing and no one, definitive method has so far emerged, as is exemplified by the web-site maintained by Li, which has a steadily growing collection of articles on microarray data analysis (http://www.nslij-genetics.org/microarray/). Conceptually, all these tools are based on one of the methods described above or they fall into the category "cluster analysis", described below.

Some methods will use the term "distance", whilst others may talk about "correlation". In mathematical terms, it boils down to the same thing: second order statistics, yielding the same type of information. As the methods all differ more or less from each other, it is normal that they do not come up with exactly the same results.

Which method is the best? Carpentier et al. (2004) have examined this issue and developed a protocol that allows the comparison of the different methods, in terms of their reliability. They conclude that each of the methods analyzed gave some information not provided by the others, suggesting once more the advantage of analyzing one's data with more than one statistical tool.

ANOVA, one of the methods tested, did not excel on the sulphur metabolism data. However, all factors were binary and ANOVA has the great advantage of being easily applicable in cases where the factors are non-binary. It also has another important property: ANOVA is the only method that forces the experimenter from the beginning to give the experimental set up some thought, to plan it carefully. It is therefore a good practice to think of an experimental set up in terms of ANOVA, even if the data are then exploited by another method (see 4.2).

1.6 Some alternative approaches

Instead of working with the raw data –our original spreadsheet- we can also decide to look at the expression similarity of the genes. For this we need to decide what kind of relationship we want to investigate. If we choose to work with second order statistics like the linear correlation or the Spearman's rank correlation, we are sure to find pairs of genes which fulfil this criterion. Genes, which are part of the same multi-enzymatic complex or which are part of the same metabolic pathway are two examples. From a biological point of view, however, these may not be the most interesting pairs to work with. It is therefore worthwhile to consider working with a more general similarity measure like the mutual information (see Daub et al., 2004). With this method we will not only find the genes also detected by the second order statistics but also pairs displaying a more complex behaviour, for example pairs of genes where the expression of one is only activated beyond a certain threshold expression of the other.

If you are working with N genes you end up with a table that gives you all the distances between the genes taken two by two. You can decide to represent this graphically (the same way you can express the distances between cities in a table or graphically, with a map). In other words, you can exploit your table with any of the methods described in section 1.3. Alternatively, you can use methods, which will give a more concise representation, like the two examples described below.

1.6.1 Self Organizing Maps (SOM)

Generally speaking, only the outskirts of the cloud are visually exploitable. The internal organization is hidden by the superposition of thousands of genes on the same image. The analysis would be easier if it were possible to give a faithful representation of the genes' density in each region of the cloud, with only *k* points. A rather naïve solution consists in choosing these *k* genes at random. This is unlikely to give satisfactory results, though. Calculating the optimal position of the *k* points is a difficult problem. A number of programmes exists proposing approximate solutions. An example is SOM (Self Organizing Maps), which chooses the *k* genes and provides a list of the genes close to the *k* genes. The interested reader may refer to the work by Kaski et al. (2003) for an introduction to SOM as well as a comparison of its merits compared to some classic classification methods.

Note that all the programmes proposed necessitate the adjustment of numerous parameters for which you do not necessarily have a rational basis to make your choice. This carries the risk that you only believe those results which tell you something you already know: not the best way to discover new things.

1.6.2 Clustering

The principle is to group and/or to classify the genes in function of the expression profile obtained under the various experimental conditions.

The cloud is thus divided into a number of clusters, the idea being that a cluster corresponds to a functional class. Choosing a gene of unknown function, one can look to which cluster it belongs and thus draw conclusions about its possible role.

This approach poses various problems.

From a biological point of view, we have to define what a functional class is and how many there are. These are not banal questions, as exemplified by the fact that even for well-studied organisms numerous classifications are proposed (for example SwissProt, MetaCyc, Kegg). Secondly, the functional classes found in the literature tend to be rather large, containing dozens or hundreds of genes, making them too large to permit their exploitation in the wet lab. Thirdly, the clustering methods normally do not allow a gene to be part of more than one cluster, which goes against biological intuition and experience.

From a technical point of view, we have to choose amongst a myriad of (family of) clustering techniques. As the biological question is not clearly defined, we do not have a criterion to select the pertinent and coherent method for our needs[3]. At this point one has to make do with a data-driven attitude. This necessitates a thorough knowledge of the different families of clustering techniques in order to make the best choice in function of the data set to be analyzed (Somorjai et al., 2003), as all the clustering techniques require many prior decisions (Chiappetta et al., 2004). In addition, as Somorjai et al. (2003) point out: “the maxim ‘simpler is better’ has mostly been ignored”.

As clustering methods are well-liked tools (see for example the popular software proposed by Eisen et al., 1998), various attempts have been made to circumvent the various technical problems. The reader, who would like to have a critical introduction to different families of clustering techniques, may refer to the works of Datta and Datta (2003), De Smet et al. (2002) and Somorjai et al. (2003).

1.6.3 Biological pertinence of these approaches

The methods described in section 1.6 allow grouping the genes into sets, which form networks. As Schäfer and Strimmer (2005) point out, these are NOT the “true model” for genetic networks. The biological interpretation of the networks obtained with these methods is complex when working with transcriptomic data. The complexity increases when working with proteomic or metabolomic data (Weckwerth, 2003).

2 Second part

2.1 How to plan one’s experiment

It is quite usual to find that a rather large number of genes, typically around 10 %, change expression considerably between two experimental conditions. This number is too large to be directly exploitable and we will have to extract a short and pertinent list of genes to work with. This task is greatly facilitated by an adequate and well thought-through experimental set up.

The following discussion does not include time series experiments, as they are rather idiosyncratic regarding the experimental set up (as well as the data analysis). The reader is referred to the work of Bar-Joseph (2004) who reviews this subject with great clarity.

2.1.1 The type of factors

An experiment is made up of three types of factors, each providing specific information.

The first factor corresponds to the phenomenon studied. The study concern two or more states (two culture conditions, for example, or a certain number of samples taken during a time course experiment). The aim is to narrow down to a maximum the target genes, in other words to have only few genes who change expression considerably between the different experimental states. For this, the experimental states should be as close as possible, for example:

a) In the case of the sulphur metabolism experiments, the two sulphur sources were metabolically speaking closely related.

b) When trying to isolate genes typical of a certain illness, one should study different subtypes, all closely related to the one of interest.

[3] An example is the definition of the distance between clusters. This is not a banal problem. Take for example the problem of having to define the distance between two countries: do you take the two capitals? The two biggest cities? The shortest distance (0 if the countries are adjoining)?

If this maxim is not observed, too many genes will change expression considerably and the identification of target genes will become near impossible.

The second type of factor serves to verify whether the observations made hold true if the biological parameters are changed. Do we find the same candidate genes if we work with a different ecotype? Or if the experiment is carried out on a different day? Note that even repeating the experiment on a different date introduces a biological variability, as the experimental conditions will never be exactly the same (see Sekowska et al., 2001). This verification is extremely important as the most interesting genes are those which come up whatever the biological background. They are most likely the genes at the heart of the phenomenon studied, as their behaviour is not bound to a particular context (genetic or physiological). The reader is referred to the works of Turk et al. (2004) and Whitehead and Crawford (2005) who discuss this issue.

The third type of factor is a technical one. Examples are the type of protocol used (choosing the quantity of mRNA for the RT-PCR in the case of the sulphur data, or using the red rather than the green dye when working with fluorescence-based microarrays) or choosing to work with two spots for each gene on the array. This type of factor increases the workload without adding any biologically pertinent information. The experimental protocols have become highly reproducible and it is advisable to stick to just one protocol (with its systemic biases) and increase the number of states of the two other factors.

2.1.2 The ideal situation: a fully crossed factorial design

The best experimental setup is to follow a fully crossed experimental design (exemplified by ANOVA) as it

a) allows a good exploitation of the information given.

b) allows a precise estimation of the error variance (see Fisher for the original discussion or Mather (1943), Zar (1998) and Kerr et al. (2000) for a more user-friendly approach).

Setting up a fully crossed factorial design means that each level (state) of one factor is found in combination with each level of the other factors, as shown in Figure 1. Note that carrying out twice the experiment on strain 1 on day A and twice the experiment on strain 2 on day B would not be adequate as it would be impossible to separate the effect of the day from the effect of the strain.

2.1.3 The reality

A fully crossed factorial design may not be possible. The reason is often the limited availability of material. Typically, the work is carried out in a single lab with material originating from this lab only. This material represents the learning set. To confirm the results a validation set would be needed, and to avoid finding genes specific to a particular genetic background only, one should work with different ecotypes.

This may prove to be unfeasible if not impossible. A different option is to give up on the fully crossed factorial design altogether and take a completely different approach: One can exploit all the experimental data available in the literature (freely available on the web) by pooling them together. This is not as bizarre an idea as it may seem; the aim can be to increase the number of patients (Jiang et al., 2004) or to get information about the co-expression and co-regulation of genes (Lee et al., 2004; Yeung et al., 2004). Especially for the two latter issues, this is the only approach, as a very large amount of data is needed, which a single lab could not possibly come up with.

Various authors propose statistical models to help extracting the maximum information from these pooled data (see for example Shen et al. (2004) and Statnikov et al. (2005)). Note that when working with pooled date, their analysis will have to be carried out with methods which do not need the definition of the factors *a priori*, like PCA or ICA.

2.1.4 The combination of factors

If we want to obtain useful information from our microarray experiment, we are forced to formulate precise questions. This means that we cannot combine two factors in one question, as this is equivalent to measuring the interaction between the factors, which is not separable from the error (unless we know in detail the relationship between the two factors, for example linear or sinusoidal).

This is one more reason to follow a fully crossed factorial experimental set up, as exemplified by ANOVA. It forces us to spell out in detail what we want to measure and what will be part of the error or interaction component. It has the great advantage of permitting the identification of the sources of variability and their magnitude; this allows making the improvements to the set up at the right sources, which will generally be a modification of the experimental protocol and an increase in the number of biological replications (Chen et al., 2004).

2.1.5 The use of microarrays to find genes for an accurate diagnosis of a disease

In general, the data will come from one lab and the observations will have been made on a limited amount of material. This material represents the learning set and the analysis of the data will always come up with some candidate genes. To validate the results, however, we need a validation set. It is wise to have five to six times more observations in this set than candidate genes, which may prove to be difficult, if not impossible.

To avoid finding genes that are only specific to a particular genetic background, different ecotypes should be studied. This is equivalent to increasing the states of the second type of factor, described in section 2.1.1. Again, this may pose a problem, from a financial as well as logistical point of view.

From a theoretical point of view, the use of microarrays for the diagnosis of a diseases poses two fundamental problems, the first one being Bellman's "curse of dimensionality" (too many features or dimensions, e.g. thousands of genes), the second one being the "curse of dataset sparsity" (too few samples) (Somorjai et al., 2003); this means that we end up analyzing a space with a great number of dimensions which is nearly empty: whatever method is applied to the analysis of the data, the result is unlikely to be statistically sound, the biological interpretation risks being inconclusive.

Somorjai et al. (2003) discuss this problem in detail. Hwang et al. (2002) propose a power analysis method in order to determine the minimum sample size for the - statistically reliable- discrimination of distinct disease states.

3 The answer to all our questions?

Microarrays are sometimes seen as the miracle tool, which will give all the answers to all the questions. Paying considerable attention to the experimental set up is a necessary condition, but not a sufficient one.

The preliminary phase should already take into consideration the different analysis options available to the experimenter by pulling in statisticians. This should be an exchange, not a handing over the job to a statistician, as the biological question has to stay at the front. As Vingron (2001) points out in his editorial, bioinformaticians should "go back to school and learn more statistics. Not so much with the goal of mastering all of statistics but with the goal of sufficiently educating ourselves on order to pull in statisticians."

A careful analysis of the data should follow (we suggest using more than one method, as they tend to give complementary information).

The complex nature of biological phenomena means that it is near impossible to isolate the candidate genes only through a microarray experiment (Curtis and Brand, 2004; Somorjai et al., 2003; Sontag et al., 2004), meaning that the list of candidate genes obtained will have to be further worked on. This may be done by further theoretical work (integration of all available biological knowledge) or additional experiments in the wet-lab.

Sometimes the a priori biological knowledge about the phenomenon of interest may be very limited. In this case, it can happen that despite careful planning and execution, the genes identified as interesting are not actually the cause of the phenomenon. This was for example the case in the genome-wide analysis undertaken by Oshima et al. (2001). In these cases a new experimental set-up may be solution. However, the conclusion may also be that a transcriptome analysis is not the adequate tool for the study of the phenomenon (Riva et al., 2004).

It is therefore important to realize that a microarray analysis will generally not be THE answer to all your questions. It is a complement to other approaches.

4 Software and Data used

The sulphur metabolism data from Sekowska et al. (2001) are freely accessible at http: http://195.221.65.10:1234/~carpenti/

PCA and ANOVA were performed using GeneANOVA, freely available on request for non-commercial use. Please contact Gilles Didier at didier@genopole.cnrs.fr .

ICA was adapted to gene expression analysis by Bruno Torrésani and Pierre Chiappetta (see http://www.cmi.univ-mrs.fr/~torresan/publi.html).

5 Acknowledgments

The authors wish to thank J. - L. Risler for critical reading of the manuscript.

This work was supported in part by the French Ministry of the Economy, Finance and Industry (contract ASG n° 01 4 90 6093).

References

- Bar-Joseph, Z., 2004. Analyzing time series gene expression data. Bioinformatics 20, 2493-2503.
- Benjamini, Y., Hochberg, Y., 1995. Controlling the false discovery rate: a practical and powerful approach to multiple testing. J. R. Stat. Soc. Ser. B 57, 289-300.
- Carpentier, A.-S., Riva, A., Tisseur, P., Didier, G., Hénaut, A., 2004. The operons, a criterion to compare the reliability of transcriptome analysis tools: ICA is more reliable than ANOVA, PLS and PCA. Computational Biology and Chemistry 28, 3–10.
- Chen, J., Delongchamp, R., Tsai, C.-A., Hsueh, H.-M., Sistare, F., Thompson, K.L., Desai, V.G., Fuscoe, J.C., 2004. Analysis of variance components in gene expression data. Bioinformatics 20, 1436–1446.
- Chiappetta, P., Toubaud, M.C., Torrésani, B., 2004. Blind Source Separation and the Analysis of Microarray Data. J. Comput. Biology 11, 1090-109.
- Curtis, R.K., Brand, M.D., 2004. Analysing microarray data using modular regulation analysis. Bioinformatics 20, 1272-1284.
- Datta, S., Datta, S., 2003. Comparisons and validation of statistical clustering techniques for microarray gene expression data. Bioinformatics 19, 459–466.
- Daub, C.O., Steure, R., Selbig, J., Kloska, S., 2004. Estimating mutual information using B-spline functions – an improved similarity measure for analysing gene expression data. BMC Bioinformatics, 5: 118, e-pub.
- De Smet, F., Mathys, J., Marchal, K., Thijs, G., De Moor, B., Moreau, Y., 2002. Adaptive quality-based clustering of gene expression profiles. Bioinformatics 18, 735-746.
- Eisen, M.B., Spellman, P.T., Brown, P.O., Botstein, D., 1998. Cluster analysis and display of genome-wide expression patterns. PNAS 95, 14863-14868.
- Fisher, R.A., 1951. The Design of Experiments, sixth ed. Oliver and Boyd, London.
- Hoyle, D., Rattray, M., Jupp, R., Brass, A., 2002. Making sense of microarray data distributions. Bioinformatics 18, 576–584.
- Hwang, D., Schmitt, W., Stephanopoulos, G., Stephanopoulos, G., 2002. Determination of minimum sample size and discriminatory expression patterns in microarray data. Bioinformatics 18, 1184-1193.
- Hyvärinen, A., Oja, E., 2000. Independent component analysis: algorithms and applications. Neural Networks 13, 411-430.
- Jiang, H., Deng, Y., Chen, H.-S., Tao, L., Sha, Q., Chen, J., Tsai, C.-J., Zhang, S., 2004. Joint analysis of two microarray gene-expression data sets to select lung adenocarcinoma marker genes. BMC Bioinformatics 5: 81, e-pub.
- Kaski, S., Nikkilä, J., Oja, M., Venna, J., Törönen, P., Castrén, E., 2003. Trustworthiness and metrics in visualizing similarity of gene expression. BMC Bioinformatics 4: 48, e-pub.
- Kendall, M., Stuart, A., Ord, J.K., 1983. The advanced theory of statistics – Vol. 3 Design and analysis, and time-series. Charles Griffin & Co.
- Kerr, K., Churchill, G., 2001. Statistical Design and the Analysis of Gene Expression Microarray Data. Genetical Research **77**, 123–128.

- Kerr, K., Martin, M., Churchill, G., 2000. Analysis of variance for gene expression microarray data. J. Comput. Biol., **7**, 819–837.
- Kim, K.-Y., Kim, B.-J., Yi, G.-S., 2004. Reuse of imputed data in microarray analysis increases imputation efficiency. BMC Bioinformatics 5: 160, e-pub.
- Konishi, T., 2004. Three-parameter lognormal distribution ubiquitously found in cDNA microarray data and its application to parametric data treatment. BMC Bioinformatics 5: 5, e-pub.
- Kutalik, Z., Inwald, J., Gordon, S.V., Hewinson, R.G., Bucher, P., Hinds, J., Cho, K.-H., Wokenhauer, O., 2004. Advanced significance analysis of microarray data based on weighted resampling: a comparative study and application to gene deletions in Mycobacterium bovis. Bioinformatics 20, 357-363.
- Lawrence, N., Milo, M., Niranjan, M., Rashbass, P., Soullier, S., 2004. Reducing the variability in cDNA microarray image processing by Bayesian inference. Bioinformatics 20, 518–526.
- Lee, H., Hsu, A., Sajdak, J., Qin, J., Pavlidis, P., 2004. Coexpression Analysis of Human Genes Across Many Microarray Data Sets. Genome Research 14, 1085–1094.
- Liebermeister, W., 2002. Linear modes of gene expression determined by independent component analysis. Bioinformatics 18, 51–60.
- Lyng, J., Badiee, A., Svendsrud, D., Hovig, E., Myklebost, O., Stokke, T., 2004. Profound influence of microarray scanner characteristics on gene expression ratios: analysis and procedure for correction. BMC Genomics 5: 10, e-pub.
- Martoglio, A.-M., Miskin, J., Smith, S., MacKay, D., 2002. A decomposition model to track gene expression signatures: preview on observer-independent classification of ovarian cancer. Bioinformatics 18, 1617–1624.
- Mather, K., 1943. Statistical Analysis in Biology, first ed. Methuen.
- Neuhäuser, M., Senske, R., 2004. The Baumgartner-Weiß-Schindler test for the detection of differentially expressed genes in replicated microarray experiments. Bioinformatics 20, 3553-3564.
- Oba, S., Sato, M.-A., Takemasa, I., Monden, M., Matsubara, K.-I., Ishii, S., 2003. A Bayesian missing value estimation method for gene expression profile data. Bioinformatics 19, 2088–2096.
- Oshima, T., Wade, C., Kawagoe, Y., Ara, T., Maeda, M., Masuda, Y., Hiraga, S., Mori, H., 2002. Genome-wide analysis of deoxyadenosine methyltransferase-mediated control of gene expression in Escherichia coli. Mol. Microbiol. 45, 673-695.
- Ouyang, M., Welsh, W.J., Georgopoulos, P., 2004. Gaussian mixture clustering and imputation of microarray data.Bioinformatics 20, 917-923.
- Querec, T., Stoyanova, R., Ross, E., Patriotis, C., 2004. A novel approach for increasing sensitivity and correcting saturation artifacts of radioactively labeled cDNA arrays. Bioinformatics 20, 1955–1961.
- Riva, A., Delorme, M.-O., Chevalier, T., Guilhot, N., Henaut, C., Henaut, A., 2004. The difficult interpretation of transcriptome data: the case of the GATC regulatory network. Computational Biology and Chemistry 28, 109–118.
- Sasik, R., Calvo, E., Corbeil, J., 2002. Statistical analysis of high-density oligonucleotide arrays: a multiplicative noise model. Bioinformatics 18, 1633–1640.
- Schäfer, J., Strimmer, K., 2005. An emprirical Bayes approach to ingerring large-scale gene association networks. Bioinformatics 21: 754-764.
- Scholz, M., Gatzek, S., Sterling, A., Fiehn, O., Selbig, J., 2004. Metabolite fingerprinting: detecting biological features by independent component analysis.
- Sekowska, A., Robin, S., Daudin, J.J., Henaut, A., Danchin, A., 2001. Extracting biological information from DNA arrays: an unexpected link between arginine and methionine metabolism in Bacillus subtilis. Genome Biology. 2, Research0019, e-pub.

- Somorjai, R.L., Dolenko, B., Baumgartner, R., 2003. Class prediction and discovery using gene microarray and proteomics mass spectroscopy data: curses, caveats, cautions. Bioinformatics 19, 1484–1491.

- Sontag, E., Kiyatkin, A., Kholodenko, B., 2004. Inferring dynamic architecture of cellular networks using time series of gene expression, protein and metabolite data. Bioinformatics 20, 1877–1886.

- Stoyanova, R., Querec, T., Brown, T., Patriotis, C., 2004. Normalization of single-channel DNA array data by principal component analysis. Bioinformatics 20, 1772–1784.

- Thygesen, H., Zwinderman, A., 2004. Comparing transformation methods for DNA microarray data. BMC Bioinformatics 5:77, e-pub.

- Turk., R., t'Hoen, P.A., Sterrenburg, E., de Menezes, R.X., de Meijer, E.J., Boer, J.M., van Ommen, G.-J.B., den Dunnen, J.T., 2004. Gene expression variation between mouse inbred strains. BMC Genomics 5: 57, e-pub.

- Tusher, V., Tibshirani, R., Chu, G., 2001. Significance analysis of microarrays applied to the ionizing radiation response. PNAS 98, 5116–5121.

- Vingron, M., 2001. Bioinformatics needs to adopt statistical thinking. Bioinformatics 17, 389-390.

- Weckwerth, W., 2003. Metabolomics in systems biology. Annu. Rev. Plant Biol. 54: 669-689.

- Whitehead, A. and Crawford, D.L., 2005. Variation in tissue-specific gene expression among natural populations. Genome Biology 6: R13, e-pub.

- Yeung, K.Y., Medvedovic, M., Bumgarner, R., 2004. From co-expression to co-regulation: how many microarray experiments do we need? Genome Biology 5:R48, e-pub.

- Zar, J.H., 1998. Biostatistical Analysis, fourth ed. Pearson Education.

- Zhao, Y., Li, M.-C., Simon, R., 2005. An adaptive method for cDNA microarray normalization. BMC Bioinformatics 6: 28, e-pub.

- Zhou, X., Wang, X., Dougherty, E.R., 2003. Missing-value estimation using linear and non-linear regression with Bayesian gene selection. Bioinformatics 19, 2302-2307.

Legends

Figure 1 : Experimental design of the transcriptome analysis on *Bacillus subtilis* (Sekowska et al., 2001)

The experimental set up follows a fully crossed factorial design. In the case of Sekowska et al. (2001) the quantity of RNA used for the RT-PCR differed between the two protocols. Note that changing the protocol (a different quantity of RNA or labelling with Cy3 rather than Cy5) or having duplicats for each gene on the array are all technical factors which increase the workload without adding any biologically pertinent information. It is preferable to increase the number of states for the biological factors, in the above case an additional sulphur source or an additional experimental day.

Figure 2: Effect of different pre-processing methods on the data distribution

The figure shows the effect different pre-processing methods have on the data distribution. Shown are the distributions a) of the raw data, b) after having taken the log and c) after having taken the fifth root. As can be seen, either operation brings the distribution closer to a Gaussian one.

Figure 3: Projections of the data on different planes.

In all four figures, each axis corresponds to an experimental condition. Figures 3a and c show metA1a versus mtrA1a, Figures 3b and d show metA1a versus metA10a (see Figure 1 for the nomenclature). Figures a and b show projections of the raw data, in Figures c and d the data are log centre-reduced. log centre-reducing the data has brought the few points which are far away from the main body in Figures a and b closer in Figures c and d. Note how the space is more efficiently used in Figures c and d. The points in the two left hand pictures form a narrower “cigar”, indicating that fewer genes have changed expression than on the right hand side.

Table 1: The mixing matrix calculated by the spreadsheet (MS-Excel) and the *eigenvector matrix* calculated by PCA

The mixing matrix at the top was calculated by the spreadsheet, the mixing matrix (or *eigenvector matrix*) at the bottom by PCA. The arrows indicate the columns which separate well the effects of the same factors. The matrices allow us to change from the old to the new reference frame: they give us for each of the new axes (the columns) the coefiicient with which we have to multiply each gene’s value in a given experimental condition (the lines) in order to obtain the new coordinates. The first line in the *eigenvector matrix* contains the *eigenvalue* for each axis (in %), providing an indication of the cloud’s dispersion along that axis. Note that for the first axis all the sixteen coefficients have basically the same value; this means that for the first axis, all experimental conditions have the same weight, in other words, the first axis gives us the total expression of each gene, which is generally true (see Stoyanova et al., 2004). **An example** for the calculation of the new coordinates with the *eigenvector matrix*: in the original (or “old”) reference frame, the gene *galK* has the coordinates (5.431; 5.432; 5.092; 5.068; 4.893; 4.744; 3.763; 3.661; 5.333; 5.265; 5.329; 5.249; 4.607; 4.444; 3.806; 3.737). To obtain *galK*’s coordinate on the new axis 1. the calculations are as follows: (5.431 x 0.250) + (5.432 x 0.249) + …. + (3.737 x 0.251) = 19.0. The other coordinates are obtained accordingly.

Figure 4: The expression change in function of the factor sulphur as calculated by a spreadsheet (MS-Excel)

The figure shows the genes’ expression change in function of the sulphur source against their mean expression. The potentially interesting genes are those away from the main body of the cloud. The highlighted genes are the ones which proved to be of particular interest for the problem investigated by Sekowska et al. (2001); the reader may refer to their work for a detailed discussion. Note that not all of these genes would have been detected using the spreadsheet.

Figure 5: The data cloud projected on the plane formed by axis 1 against axis 5 (PCA)

The figure shows the genes’ expression change in function of the sulphur source (axis 5) against their mean expression (axis 1), as calculated by PCA. . The potentially interesting genes are those away from the main body of the cloud. The highlighted genes are the ones which proved to be of particular interest for the problem investigated by Sekowska et al. (2001); the reader may refer to their work for a detailed discussion. Note that not all of these genes would have been detected using PCA.

Figure 6: The data cloud projected on the plane that separates well the sulphur source (ICA)

The figure shows the genes’ expression change in function of the sulphur source, as determined by ICA. . The potentially interesting genes are those away from the main body of the cloud. The highlighted genes are the ones which proved to be of particular interest for the problem investigated by Sekowska et al. (2001); the reader may refer to their work for a detailed discussion. Note that not all of these genes would have been detected using ICA.

Figure 7: The “weight” attributed to each experimental condition by a spreadsheet (MS-Excel), PCA and ICA

The figure shows that a spreadsheet (MS-Excel) attributes the same “weight”, or importance, to each experimental condition, whilst PCA and ICA do not.

Figure 8: The graphical representation of the results obtained with ANOVA for the factor sulphur

The potentially interesting genes are those with a small *p-value* and a large variance, genes which are therefore in the bottom right part of the image. They are away from the main body of the cloud. The highlighted genes are the ones which proved to be of particular interest for the problem investigated by Sekowska et al. (2001); the reader may refer to their work for a detailed discussion. As can be seen, not all genes of interest would have been identified by the sole use of ANOVA.

Table 2: Comparison between ANOVA, the paired *t-test* and the *t-test*, an example

Table 2a shows the measurements obtained for *ytmJ*, Tables 2b-d the calculations and results obtained with the *t-test*, the paired *t-test* and ANOVA, respectively. ANOVA and the paired *t-test* both identify the gene as potentially interesting, whilst the *t-test* results inconclusive (see the relative –log (*p-value*)). DF = Degrees of Freedom; SS= Sum of Squares. See section 3.4 for details.

Table 3:Overall comparison between ANOVA, the paired *t-test* and the *t-test*

The table shows a comparison of the number of genes detected by the three methods. In Table 3a the threshold for detection was –log (*p-value*) = 3, in Table 3b it was equal to 4. Although ANOVA detects in both cases the highest number of genes, the paired *t-test* performs comparably well, whilst the *t-test* lags far behind.

Tables

Table 1: The mixing matrix calculated by the spreadsheet (MS-Excel) and the *eigenvector matrix* calculated by PCA

		Axis				
		Mean expression	Effect of protocol	Effect of day	Effect of sulphur source	Effect of duplicate
Experimental condition	metA1a	0.250	0.250	-0.250	0.250	-0.250
	metA1b	0.250	0.250	-0.250	0.250	0.250
	metB1a	0.250	0.250	0.250	0.250	-0.250
	metB1b	0.250	0.250	0.250	0.250	0.250
	metA10a	0.250	-0.250	-0.250	0.250	-0.250
	metA10b	0.250	-0.250	-0.250	0.250	0.250
	metB10a	0.250	-0.250	0.250	0.250	-0.250
	metB10b	0.250	-0.250	0.250	0.250	0.250
	mtrA1a	0.250	0.250	-0.250	-0.250	-0.250
	mtrA1b	0.250	0.250	-0.250	-0.250	0.250
	mtrB1a	0.250	0.250	0.250	-0.250	-0.250
	mtrB1b	0.250	0.250	0.250	-0.250	0.250
	mtrA10a	0.250	-0.250	-0.250	-0.250	-0.250
	mtrA10b	0.250	-0.250	-0.250	-0.250	0.250
	mtrB10a	0.250	-0.250	0.250	-0.250	-0.250
	mtrB10b	0.250	-0.250	0.250	-0.250	0.250

		Axis															
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
	eigenvalue	94.64	2.26	1.01	0.56	0.36	0.29	0.26	0.18	0.16	0.05	0.05	0.05	0.04	0.04	0.03	0.02
Experimental condition	metA1a	0.250	0.307	-0.164	-0.114	0.336	-0.004	-0.310	0.179	0.349	-0.146	-0.130	0.406	-0.228	0.429	-0.020	-0.017
	metA1b	0.249	0.312	-0.166	-0.131	0.366	0.075	0.157	0.252	0.347	0.131	0.078	-0.429	0.262	-0.413	0.001	0.026
	metB1a	0.252	0.179	0.115	0.365	0.124	0.306	-0.301	-0.280	-0.158	0.172	0.629	-0.094	-0.019	0.132	0.052	0.028
	metB1b	0.251	0.181	0.107	0.369	0.153	0.425	0.211	-0.246	-0.203	-0.170	-0.592	0.072	0.003	-0.140	-0.036	-0.027
	metA10a	0.249	-0.256	-0.326	0.211	0.200	-0.429	-0.183	-0.129	-0.186	0.372	-0.309	-0.254	0.193	0.264	0.082	0.028
	metA10b	0.249	-0.255	-0.328	0.192	0.260	-0.303	0.315	-0.027	-0.146	-0.388	0.325	0.261	-0.198	-0.276	-0.081	-0.031
	metB10a	0.250	-0.254	0.263	-0.232	0.126	0.086	-0.322	0.311	-0.322	-0.041	-0.070	-0.072	-0.146	-0.163	-0.330	0.505
	metB10b	0.250	-0.253	0.259	-0.236	0.178	0.180	0.170	0.358	-0.283	0.064	0.067	0.092	0.171	0.180	0.335	-0.508
	mtrA1a	0.248	0.334	-0.083	-0.365	-0.224	-0.182	-0.173	-0.256	-0.270	0.108	0.003	0.432	0.363	-0.310	0.093	0.018
	mtrA1b	0.248	0.331	-0.087	-0.370	-0.189	-0.093	0.321	-0.159	-0.277	-0.089	0.029	-0.407	-0.399	0.305	-0.086	-0.028
	mtrB1a	0.251	0.156	0.220	0.311	-0.360	-0.291	-0.164	0.272	0.049	-0.540	0.002	-0.211	0.315	0.102	0.061	-0.029
	mtrB1b	0.252	0.149	0.214	0.322	-0.290	-0.206	0.256	0.332	0.099	0.540	-0.012	0.236	-0.301	-0.101	-0.066	0.029
	mtrA10a	0.249	-0.239	-0.346	-0.031	-0.376	0.270	-0.285	0.035	0.151	-0.042	-0.089	-0.114	-0.350	-0.261	0.475	-0.045
	mtrA10b	0.249	-0.240	-0.352	-0.030	-0.342	0.369	0.195	0.052	0.150	0.053	0.087	0.108	0.352	0.262	-0.471	0.045
	mtrB10a	0.251	-0.226	0.335	-0.131	-0.003	-0.151	-0.196	-0.375	0.341	0.014	-0.054	-0.094	-0.118	-0.174	-0.385	-0.484
	mtrB10b	0.251	-0.225	0.327	-0.140	0.041	-0.057	0.310	-0.322	0.358	-0.037	0.036	0.067	0.101	0.164	0.376	0.491

Table 2: Comparison between ANOVA, the paired *t-test* and the *t-test*, an example

a) Measurements obtained for *ytmJ*

	met	**mtr**	**met-mtr**
A1a	1,170	1,520	-0,3494
A1b	1,176	1,580	-0,4048
B1a	0,950	1,566	-0,6158
B1b	0,891	1,541	-0,6496
A10a	1,939	2,049	-0,1096
A10b	1,565	2,048	-0,4827
B10a	0,893	1,523	-0,6296
B10b	1,007	1,485	-0,4772

b) *t-test*

Numerator	-0,465
Denominator	0,313
DF	14
t-test	**-2,971**
-log (*p-value*)	**1,99**

c) paired *t-test*

Numerator	-0,465
Denominator	0,064
DF	7
paired t-test	**-7,290**
-log (*p-value*)	**3,78**

d) ANOVA

	State 1	State 2	SS
Sulphur	9,592	13,311	0,864
Day	13,048	9,855	0,637
RNA	10,394	12,508	0,279
Spot	11,610	11,293	0,006
Residual			0,448
Total			2,235

Factor	SS	DF	Variance	F	-log (*p-value*)
Sulphur	0,864	1	0,864	**21,21**	**3,12**
Day	0,637	1	0,637	15,64	2,65
RNA	0,279	1	0,279	6,86	1,62
Spot	0,006	1	0,006	0,15	0,00
Residual			0,041		
Total			0,149		

Table 3:Overall comparison between ANOVA, the paired *t-test* and the *t-test*

a) Number of genes detected with a threshold of –log (*p-value*) = 3

ANOVA only	62
t-test only	5
Both	24
ANOVA only	35
Paired *t-test* only	30
Both	51
t-test only	12
Paired *t-test* only	64
Both	17
Total ANOVA	86
Total paired *t-test*	81
Total *t-test*	29

b) Number of genes detected with a threshold of –log (*p-value*) = 4

ANOVA only	16
t-test only	0
Both	9
ANOVA only	17
Paired *t-test* only	12
Both	8
t-test only	14
Paired *t-test* only	15
Both	5
Total ANOVA	25
Total paired *t-test*	20
Total *t-test*	9

Figures

Figure 1 : Experimental design of the transcriptome analysis on *Bacillus subtilis* (Sekowska et al., 2001)

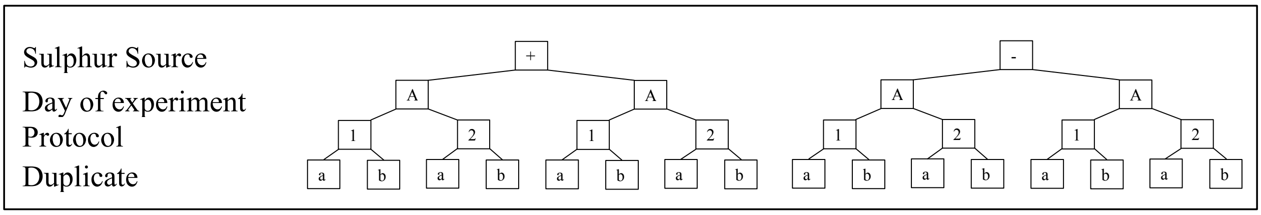


Figure 2: Effect of different pre-processing methods on the data distribution

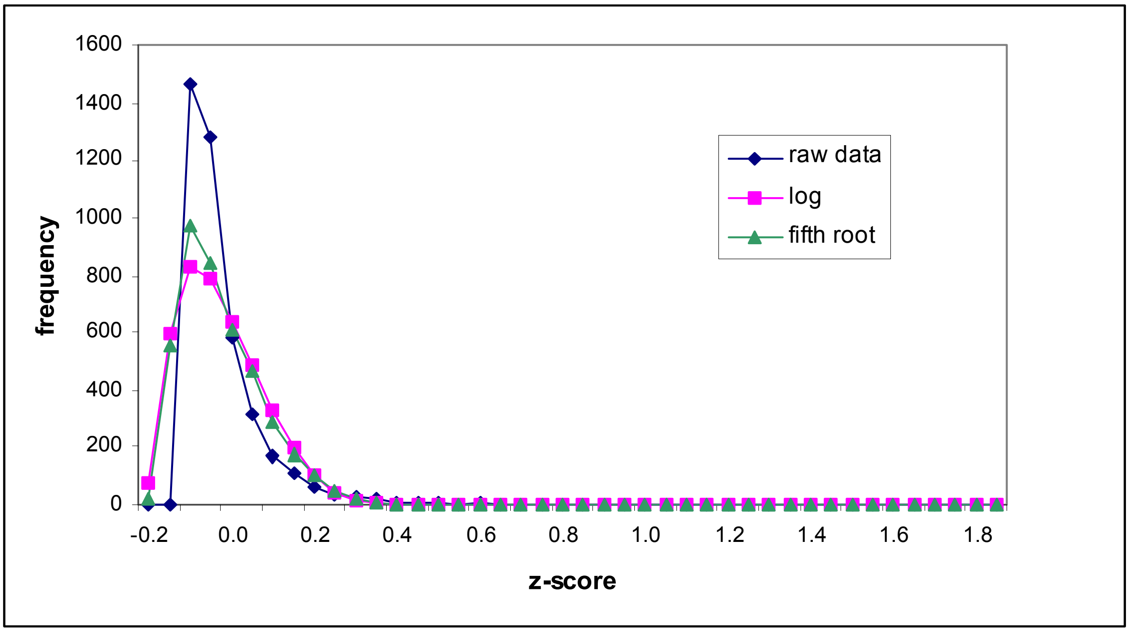


Figure 3: Projections of the data on different planes.

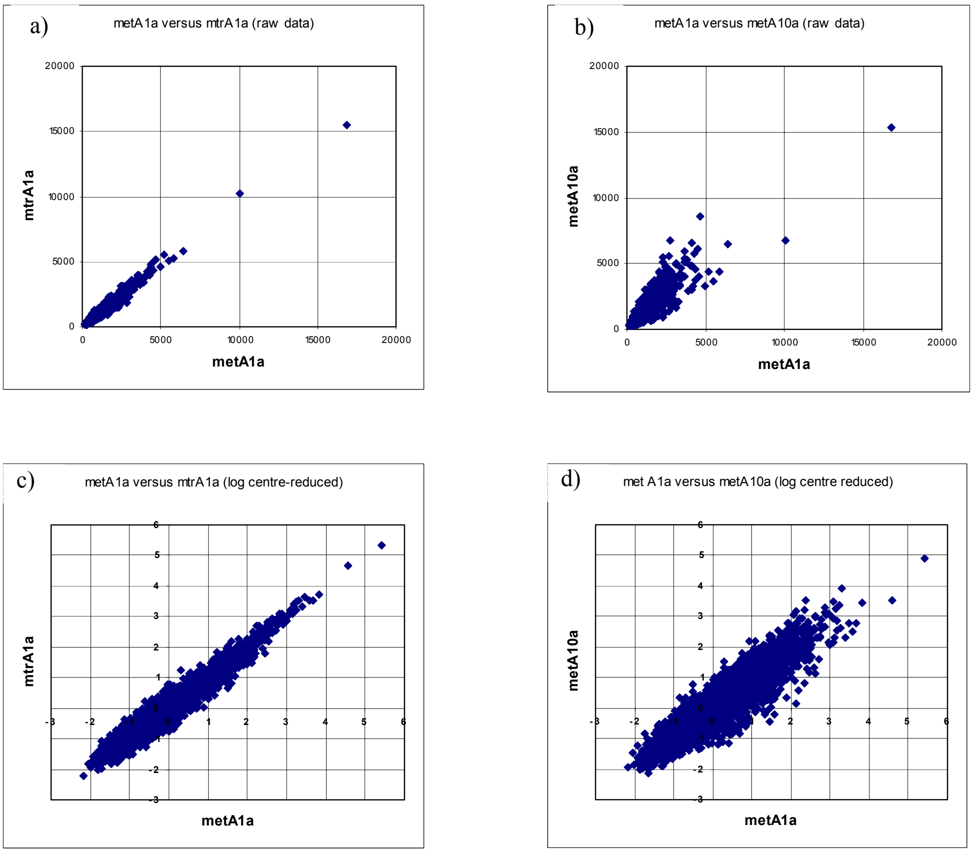


Figure 4: The expression change in function of the factor sulphur as calculated by a spreadsheet (MS-Excel)

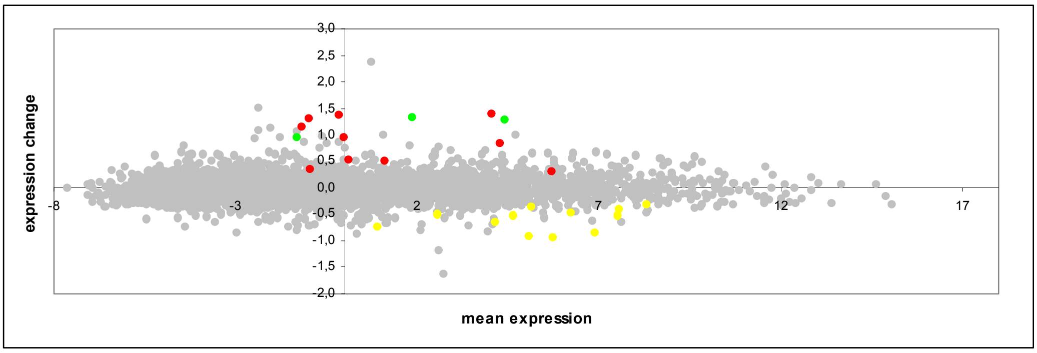


Figure 5: The data cloud projected on the plane formed by axis 1 against axis 5 (PCA)

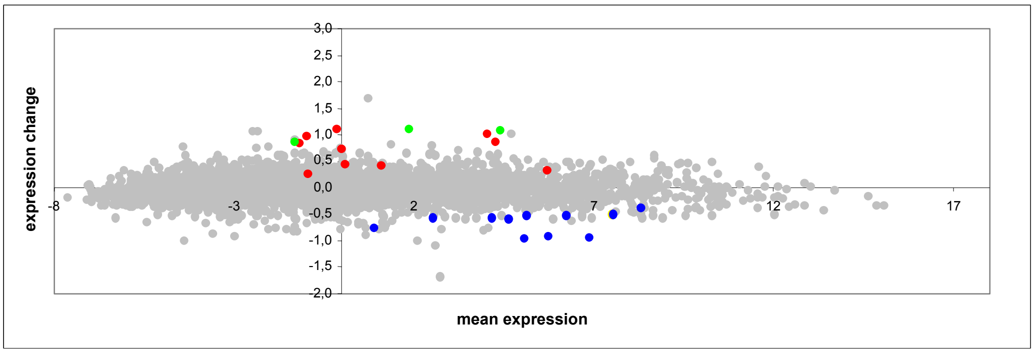


Figure 6: The data cloud projected on the plane that separates well the sulphur source (ICA)

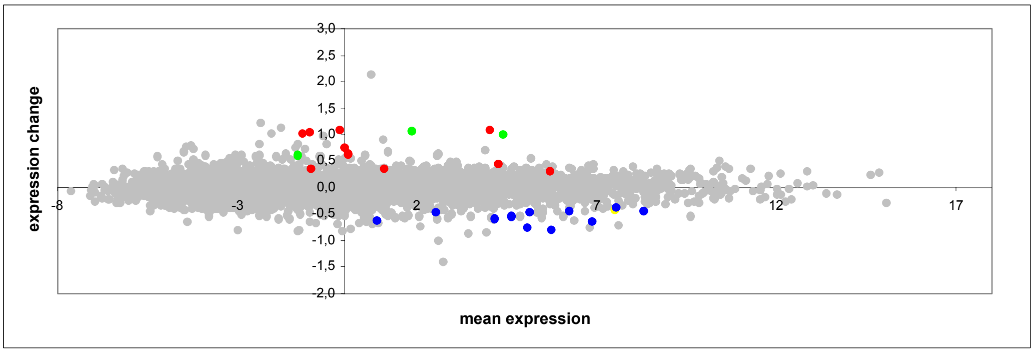


Figure 7: The "weight" attributed to each experimental condition by a spreadsheet (MS-Excel), PCA and ICA

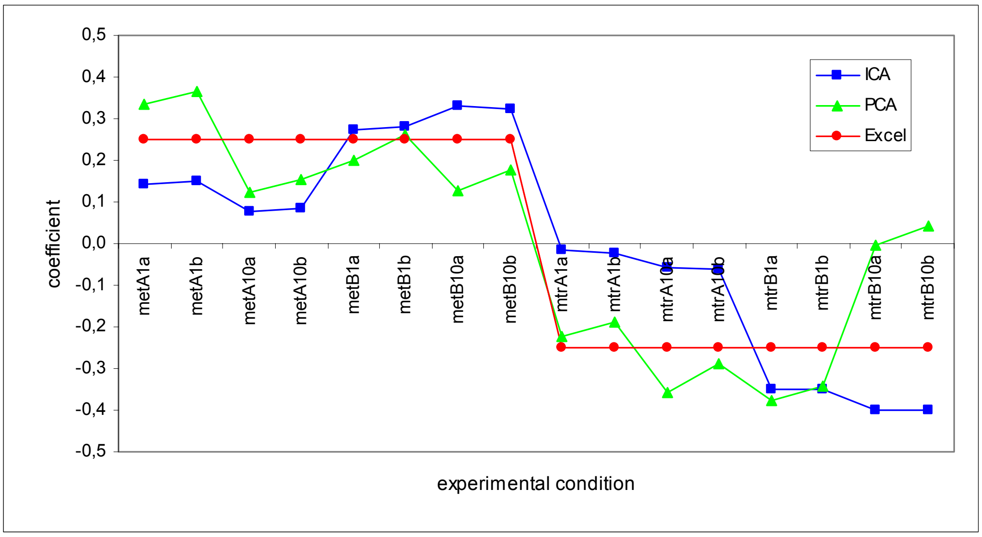
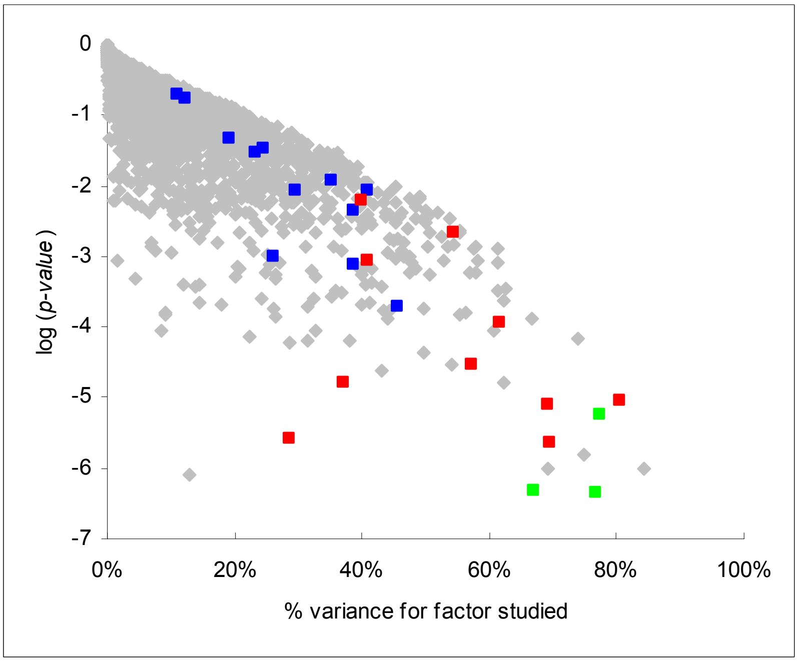


Figure 8: The graphical representation of the results obtained with ANOVA for the factor sulphur


log (*p-value*)
% variance for factor studied