BIOINFORMATICS ORIGINAL PAPER Vol. 22 no. 12 2006, pages 1471–1476
doi:10.1093/bioinformatics/btl121

Gene expression

Unbiased pattern detection in microarray data series

S. E. Ahnert[1,4,*], K. Willbrand[2], F. C. S. Brown[3] and T. M. A. Fink[1,4]

[1]Theory of Condensed Matter, Cavendish Laboratory, Cambridge CB3 0HE, UK, [2]Laboratoire de Physique Statistique, [3]Départment de mathématiques et applications, Ecole Normale Supérieure, 75231 Paris Cedex 05, France and [4]Institut Curie, CNRS UMR 144, 75248 Paris Cedex 05, France

Received on January 3, 2006; revised on March 24, 2006; accepted on March 25, 2006
Advance Access publication April 3, 2006
Associate Editor: John Quackenbush

ABSTRACT

Motivation: Following the advent of microarray technology in recent years, the challenge for biologists is to identify genes of interest from the thousands of genetic expression levels measured in each microarray experiment. In many cases the aim is to identify pattern in the data series generated by successive microarray measurements.

Results: Here we introduce a new method of detecting pattern in microarray data series which is independent of the nature of this pattern. Our approach provides a measure of the algorithmic compressibility of each data series. A series which is significantly compressible is much more likely to result from simple underlying mechanisms than series which are incompressible. Accordingly, the gene associated with a compressible series is more likely to be biologically significant. We test our method on microarray time series of yeast cell cycle and show that it blindly selects genes exhibiting the expected cyclic behaviour as well as detecting other forms of pattern. Our results successfully predict two independent non-microarray experimental studies.

Contact: sea31@cam.ac.uk

1 INTRODUCTION

Microarrays provide a powerful tool for measuring thousands of gene expression levels in parallel (Chipping Forecast II, 2002). By making several such measurements this method can be used to create a data series for every gene. Examples of such series are measurements taken during the progression of disease, during development of an organism or over increasing drug dose. The aim is to retrieve from the large amount of data indications as to which genes play an important role in the underlying process studied.

In the few years since the conception of microarray technology, data generated from microarray experiments has often been analysed with the aim to identify a specific type of pattern in the expression level data, such as periodicity or monotonous increases or decreases (Spellman *et al*., 1998; Cho *et al*., 1998; Whitfield *et al*., 2002). In general however it is often not clear which type of pattern is to be expected. Moreover, in the case where a particular pattern is expected, an approach to find such pattern has the disadvantage of ignoring potentially significant data series with unexpected kinds of pattern. It is important to recognize that the presence of pattern, or non-randomness, in itself implies a dependence of the gene expression level on the underlying variable of the data series, such as time.

Here we introduce a new measure for detecting pattern in microarray data series which is independent of the type of pattern present. We do so by defining the notion of pattern in terms of non-randomness, using concepts related to the field of algorithmic information theory. Our method is not biased toward anticipated pattern (such as periodicity), it does not filter the data for outliers, nor does it require any free parameters to be adjusted.

In order to test our method we apply it to a well-known dataset of measurements taken over two cell cycles in yeast (Spellman *et al*., 1998). We then compare the results of our pattern identification process to two independent sets of genes which are believed to be involved in cell-cycle (Simon *et al*., 2001, http://genome-www.stanford.edu/cellcycle/data/rawdata/KnownGenes.doc). These two sets were experimentally derived without the use of microarray technology. Our method predicts the genes in both these sets well, regardless of whether they exhibit two cycles (and therefore seem most obviously related to cell cycle) or other types of pattern, such as a monotonic increase or decrease. In addition it finds genes with expression patterns similar to these, but which are not in the two experimental sets. Finally, it also identifies a distinct pattern which is the result of a systematic bias in the experiment. These results illustrate the unbiased nature of our approach, which allows it to distinguish many different types of pattern from randomness.

This method is a significant generalization of earlier work by some of us (Willbrand *et al*., 2005), in which microarray data is analyzed by studying patterns of successive increases and decreases. The present paper puts the previous ideas into a much broader and more useful framework.

2 MEASURING PATTERN AND NON-RANDOMNESS

In order to establish a rigorous measure of pattern in a given microarray data series, we introduce the following two-step procedure:

First we convert all microarray curves into their rank permutations. For example a curve of five data points with values 0.23, 0.54, 0.33, 0.78, 0.91 would be translated into the sequence 1, 3, 2, 4, 5, as 0.23 is the lowest data point, 0.54 is the third lowest, 0.33 the second lowest, etc. The conversion to permutations segregates the entire space of possible microarray curves of a given resolution (meaning a given number of significant digits in the data point values) into equally sized volumes, each associated with a particular permutation. The rank permutation of a data series satisfies three

*To whom correspondence should be addressed.

useful properties: (1) The permutation is invariant (up to rank inversion) under any one-to-one transformation of the data, such as linear or logarithmic tranformations or normalization. (2) Any series of independently and identically distributed random data points will yield a uniformly distributed random permutation, whatever the distribution of data oints might be. (3) Consider any function of a permutation to a scalar. The distribution of this scalar over random data series (and hence random permutations) is independent of the distribution from which each data point is drawn. A corollary of (3) is that the distribution of the said scalar is also independent of the addition of background noise to the data series. These properties are key to defining an unbiased measure of pattern in a curve, and although they are introduced at the expense of coarse graining the data by their rank order, we will find it is a price worth paying.

In the second step of our procedure we collect these volumes into groups of various different sizes. We do this by choosing a simple map γ which acts upon a permutation and gives as its output a real number. Permutations which are associated with the same number are grouped together. As the size of these groups varies, the amount of information necessary to locate a particular curve varies too. This information—the address of a curve—consists of two parts. The first part is of constant length (for a given γ map), indicating which group the curve belongs to. The second part however varies. For large groups we need to give a long address to specify an individual permutation and, through it, a particular curve. For small groups, on the other hand, a short address is sufficient. A simple illustration of this address structure can be given in the form of trees as shown in Figure 1. Because these addresses are a result of the application of a simple map, the combination of the map and a short address indicates a high compressibility $k(f)$ (see Discussion for details). More formally, the compressibility $k(f)$ given a particular map γ is bounded by the difference of the Shannon entropy S of the curve f—which is constant if we consider all curves of a given resolution—and the total address length for the curve f. This bound $k_\gamma(f)$ reads:

$$k(f) \geq k_\gamma(f) = S - \log_2 N_\gamma - \log_2 M(f),$$

where N_γ is the number of groups generated by the map γ (also known as the image of γ) and $M(f)$ is the number of curves located inside the group which contains the curve f. As $S = \log_2 M_{\text{total}}$, where M_{total} is the total number of possible curves of a given resolution, we can rewrite k as:

$$k(f) \geq k_\gamma(f) = -\log_2 p(f) - \log_2 N_\gamma,$$

where $p(f) \in [0, 1]$ is the probability of a random curve having the same γ value that f has. Thus we can translate the probability $p(f)$ into a bound $k_\gamma(f)$ on the true algorithmic compressibility $k(f)$ of the curve f in terms of bits. Perhaps surprisingly, there is no unique choice of the map which converts a given permutation to a number γ. This is because our approach calculates a bound on the algorithmic compressibility $k(f)$, not the quantity itself, which in fact is fundamentally uncomputable (Cover and Thomas, 1991). The simplicity of the map however is paramount, as only a concise map will yield a useful bound. As can be seen in Section 4, completely different simple maps give broadly similar results, underlining the argument that the bound is useful as long as the description of the map is short (see Discussion for details). At first sight our approach might be reminiscent of hash maps. An optimal hash map however would be one which distributes hash keys evenly among values, while a useful γ map does the opposite, namely produce a wide range of group (or, in hash table terminology, 'bucket') sizes.

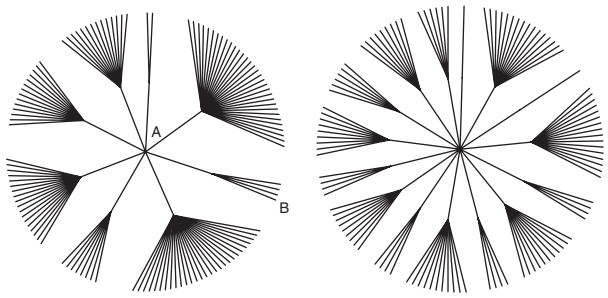

Fig. 1. The tree representations of the structure of the groups created by the maps γ_{Δ_1} (left) and γ_{+-} (right). Locating a given curve using the two-part address is equivalent to starting at the centre of the tree (A) and finding a particular exit at the edge (e.g. B). If the first part of the address leads to a small group, little more information is needed to find a specific exit. If however the initial branch leads to a big group and there are many branches to choose from, a larger amount of information is required to reach a particular endpoint in this group. The endpoint B, being in a group of four, could correspond to the permutation (4, 5, 3, 2, 1) of the example given in the text. To find it, someone starting at A would require $k_\gamma(f) = -\log_2 p(f) - \log_2 N_\gamma = 1.92$ bits of information less than required for transmitting a number between 1 and 120 (= 5!), which would specify the permutation (4, 5, 3, 2, 1) directly. The additional information to get from the permutation to a specific curve f is constant for all curves of a given resolution and thus not of interest in this representation.

The algorithmic compressibility k is a measure of the significance of a given curve in relation to the underlying variable of the series. If the curve is a time series of measurements of gene expression across the duration of a cell cycle, then an algorithmically compressible curve is more likely than others to be related to the cell cycle. If instead the series consisted of microarray data samples taken from different medical patients ordered by an external parameter of the sample, e.g. the size of a tumour in the case of cancer patients, then one would be able to identify gene expression curves which are much more likely to be related to this external parameter than others. We have performed analyses for cell cycle data of the yeast genome, the results of which are presented in Section 4 below. A major advantage of our approach is that it provides a universal currency in which the significance of a particular curve with respect to the underlying parameter can be expressed. Thus one can, for instance, compare the expression of the same gene across different experiments with different numbers of data points, different noise and different types of pattern in the data.

Some simple maps from permutations to numbers γ which we use in our analyses are

γ_{long}: the length of the longest increasing or decreasing subsequence;
γ_{opt}: the number of local optima (Warren and Seneta, 1996);
γ_{+-}: the number of permutations with the same pattern of rises and falls (Willbrant *et al*., 2005)
γ_{Δ_1}: the sum of the absolute value of the first difference operator $\sum_{i=1}^{N-1} |a_{i+1} - a_i|$ (see Methods section);

$\gamma_{\Delta_2} : \sum_{i=1}^{N-2} |\, a_{i+2} - 2a_{i+1} + a_i \,|\,;$
$\gamma_{\Delta_3} : \sum_{i=1}^{N-3} |\, a_{i+3} - 3a_{i+2} + 3a_{i+1} - a_i \,|\,;$

where a_i are the values of a given permutation sequence. Note that γ_{Δ_1} is also used in the context of the measure on functions known as bounded variation (Weisstein *et al*., http://mathworld.wolfram.com/BoundedVariation.html). Despite the simple appearance of these maps, their properties can be difficult to calculate and lead to open questions (Warren and Seneta, 1996; Willbrand *et al*., 2005). The distributions of γ values are most easily calculated by analyzing randomly generated data. For the case of γ_{+-}, we give theoretical details on this in (Willbrand *et al*., 2005).

3 METHODS

Here we give a straightforward recipe for determining the compression in bits k for a given curve. We take the map γ (the only free parameter) to first be γ_{Δ_1} and then γ_{+-}. (In Fig. 1 we show the tree structures for γ_{Δ_1} and γ_{+-}, both for $N = 5$.)

Consider the $N = 5$ point curve $f = (0.77, 0.84, 0.51, 0.30, 0.26)$, which translates into the permutation (4, 5, 3, 2, 1). In the case of the map γ_{Δ_1}, which is the sum of the absolute values of the differences of consecutive points, $\gamma_{\Delta_1}(4, 5, 3, 2, 1) = 1 + 2 + 1 + 1 = 5$. The probability that a random curve gives the same value is $p(f) = 4/120$, since 4 of the 120 permutations, namely (4, 5, 3, 2, 1), (5, 4, 3, 1, 2), (1, 2, 3, 5, 4) and (2, 1, 3, 4, 5), cause γ_{Δ_1} to take the value 5. For five data points, γ_{Δ_1} can take all values between 4 and 11, so that the total number of possible values is $N_\gamma = 8$. Hence

$$k_\gamma(f) = -\log_2 p(f) - \log_2 N_\gamma = 1.92,$$

which is the number of bits by which we can compress f using γ_{Δ_1}.

The map γ_{+-} is the number of permutations that have the same sequence of increases (+) and decreases (−) between consecutive data points (Willbrand *et al*., 2005). The up-down signature for (4, 5, 3, 2, 1) is $+---$, and only 3 other permutations have the same signature. Therefore $\gamma_{+-}(4, 5, 3, 2, 1) = 4$ and $p(f) = 4/120$. The total number of values that γ_{+-} can take is $N_\gamma = 15$, and

$$k_\gamma(f) = -\log_2 p(f) - \log_2 N_\gamma = 1.02,$$

Thus this particular curve f is algorithmically compressible for both maps γ_{Δ_1} and γ_{+-}, as both values are greater than zero. Note that many more permutations lie in one of the large groups corresponding to lower $k_\gamma(f)$, than in the smaller groups with high $k_\gamma(f)$. This means that a random change in a permutation of a curve with a high k-value is much more likely to decrease its k-value, while such a random change in a low-k permutation is unlikely to increase the k-value.

As the number of possible permutations grows as $N!$ for N data points, the probabilities $p(f)$ are most easily determined using a Monte Carlo simulation over random permutations. The probability of a random permutation generating some particular γ value is simply the fraction of permutations tested that give that value. This should be run long enough for the probabilties to have converged to a reasonable accuracy.

4 RESULTS FOR YEAST MICROARRAY DATA

We applied our analysis method to the time series data from the microarray experiments on the cell cycle of yeast measured by Spellman *et al*. (1998). The dataset contains 6073 curves of 18 points each, sampled over 2 cell cycles. These series are synchronized using the mating pheromone α-factor. Because the yeast genome and in particular this dataset have been studied extensively (Bar-Joseph *et al*., 2003; Wichert *et al*., 2004), we can compare our predictions for this data to experimental results as well as to an analysis of a random dataset of equal size.

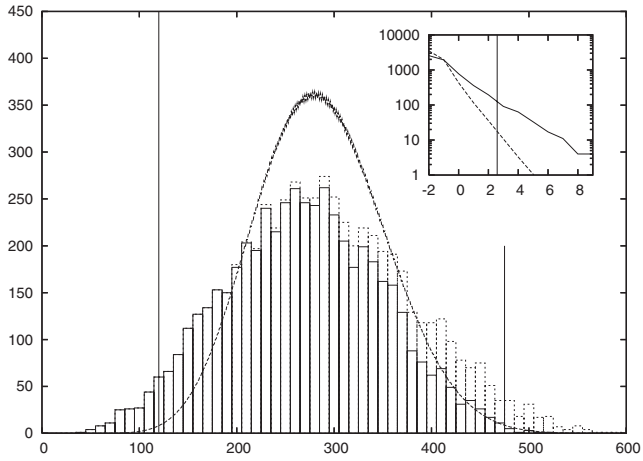

Fig. 2. MAIN: The distribution of $\gamma_{\Delta_3}(f)$ for 6073 random curves (Gaussian-shaped line) and all 6073 cell cycle curves (histogram bars). The dotted bars show the total set of 6073 curves, while the solid bars show the 5241 non-zig-zag curves. Almost all of the 832 zig-zag curves are located in the right hand tail and most of the cell cycle associated curves lie in the left hand tail. The vertical lines mark the cutoff outside of which 0.2% of the random curves lie. 150 (3%) of the non-zig-zag cell cycle curves lie outside this threshold. Inset, the distribution of $k_{\gamma_{\Delta_3}}(f)$ for random curves (dotted) and cell cycle curves (solid). The vertical line marks the k value equivalent to the 0.2% threshold in the large plot.

Among the 6073 curves we have found a class of curves which exhibit purely oscillatory (zig-zag) behaviour in step with the measurement frequency. As such curves occur far more frequently in this data set than would have been the case by chance we propose that this class of curves represents a systematic bias in the preparation of even and odd data points. It is highly unlikely that any cell-cycle related process would lead to an excess of curves exactly in step with the measurement frequency of the experiment. Therefore, in the following analysis 832 such zig-zag curves were excluded, leaving us with a set of 5241 curves. Figure 2 shows the distribution of γ_{Δ_3} and k values for all yeast genes and, as a comparison, the distribution expected from random data. It also shows the location of the removed zig-zag curves in the right hand tail of the γ_{Δ_3} distribution.

4.1 Comparison with independent experimental datasets

We ranked all 5241 non-zig-zag cell cycle curves using the six maps listed in Section 2 and compared our results with two sets of genes from non-microarray experiments. The two sets are

(I) 104 genes collected from the literature by Spellman (1998)

(II) 140 genes found to be regulated by at least one cell cycle transcription factor using genome-wide location analysis (Simon *et al*., 2001)

These two sets have an overlap of 49 genes, so that their union contains 195 genes.

The results of our analysis are shown in Figure 3, in the form of histograms. From these histograms it is evident that the majority of genes already shown to be relevant from the two experimental sets occupy positions near the top of our ranking. This demonstrates that

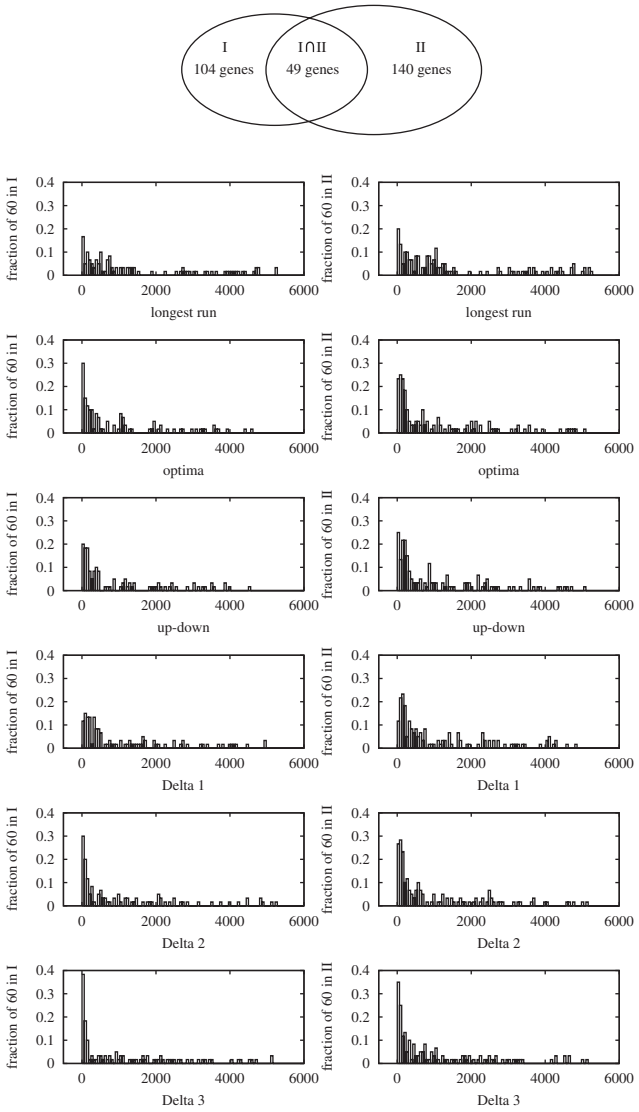

Fig. 3. The two sets of experimentally determined cell cycle genes of Spellman (I) and Simon (II) (Simon *et al*., 2001) only partially overlap, which means that at least one of them has many false positives or false negatives (see Discussion for more details on this point). We compare our ordering of Spellman's cell cycle data using the six maps γ (rows) listed in Section 2 with the two sets I and II (columns). The height of each histogram bar is the fraction of the 60 genes (the histogram width) that are included in the respective set of experimentally determined genes. Interestingly our method predict both sets equally well.

our method identifies a significant proportion of known cell cycle genes. While all our maps were successful in this respect, γ_{Δ_3} gave the best results. The top six genes in the γ_{Δ_3} ranking and their expression curves are shown in Figure 4 and Table 1.

There are 150 genes in the $k(f \mid \gamma_{\Delta_3})$ ranking—this corresponds to 2.5% of the total—that are more compressible than the 0.2% most compressible genes of the randomly generated curves. The distributions of γ_{Δ_3} and $k(f \mid \gamma_{\Delta_3})$ for both cell cycle and random curves are shown in Figure 2.

Of these 150 genes which generated highly compressible time series, 52 were identified in either of the experimental studies or (Simon *et al*., 2001), which means that >25% of the union of these two experimental sets lie in the top 2.5% of our ranking.

4.2 Compression compared to random data

For our second test, we compared the compression of the same 5241 cell cycle curves with the compression of an equal number of randomly generated curves each of which consisted of 18 data points, sampled from a uniform distribution.

Using the same six γ maps, we obtained maximum compression values between 3.8 and 8.1 bits for the randomly generated data curves. For the cell cycle curves the equivalent maximum values lay between 10.6 and 14.2 bits. The distributions of $\gamma_{\Delta_3}(f)$ and $k(f \mid \Delta_3)$ are shown in Figure 2 for both the random and the cell cycle data. From these results it is clear that the cell cycle curves are significantly more compressible than random curves. Furthermore, in the case of the six γ maps we counted 82–329 genes which were more compressible than 99.8% of the randomly generated curves—by chance one would only expect 10 curves (0.2%).

4.3 Classes of pattern

Our method attempts to detect the presence of pattern, or non-randomness, regardless of the type of pattern present. The presence of pattern can imply several different things: First that the curve is functionally (i.e. biologically) associated with the underlying variable of the series, second that the curve contains systematic experimental bias (such as drift) or third that the curve is compressible by chance. Microarray studies do not allow us to distinguish between these alternatives. However, if at least some genes in the system are thought to be biologically associated with the independent variable, the genes exhibiting the most pattern are the most likely candidates. Figure 5 shows examples of various curves exhibiting different types of pattern. Some of these are in the independent datasets (Simon *et al*., 2001) mentioned above, but others with similar types of pattern are not.

5 DISCUSSION

There are two principal advantages which differentiate our method from others (Bar-Joseph *et al*., 2003; Wichert *et al*., 2004; Wallace and Dowe, 1999) as it provides (1) a rigorous and unbiased measure of pattern and secondly, it also creates a universal currency which allows the comparison of curves from different datasets and experiments.

5.1 The choice of the map γ

The defining characteristic of our choices for the γ maps is that their descriptions are short (in an algorithmic sense) compared with the length of an arbitrary map. An arbitrary map is any assignment of the permutations to c distinct groups, where c is the size of the map's image. For example it would be possible to construct a specific map γ' which gives the same distribution of partition sizes $|S_{\gamma\prime(\pi)}|$ as γ_{Δ_3}, but with the 195 experimentally derived genes uniformly distributed throughout this ranking. However the description of this map γ' is very likely to be much longer than the simple map for γ_{Δ_3}, given in Section 2. If we were to compare several k bounds derived using different γ maps, we would have to include descriptions of the γ maps as part of the map to generate a given data series. This explains why a long and convoluted map such as γ' is not useful for finding a bound on the algorithmic compressibility.

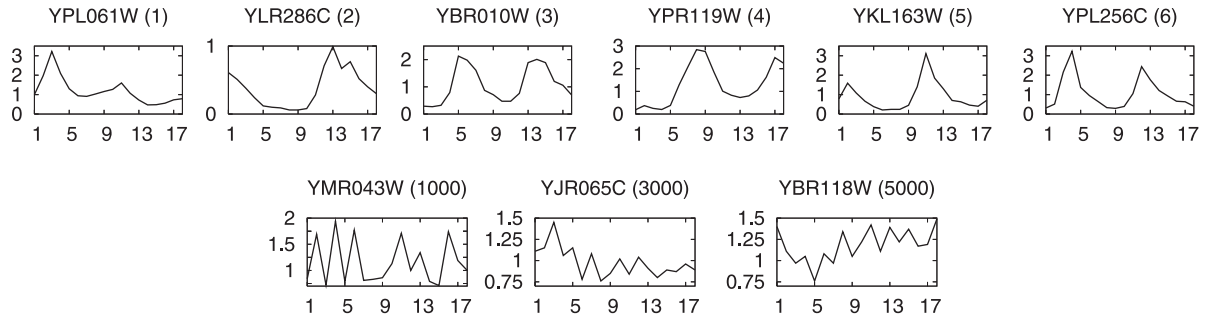

Fig. 4. The top six cell cycle expression curves and curves at positions 1000, 3000 and 5000 when ranked by $k(f \mid \gamma_{\Delta_3})$. Note that curves with lower signal amplitudes tend to end up lower down in the ranking. This is striking since the information about absolute signal levels is lost in the transformation to permutations. The reason why the absolute signal levels nevertheless play an indirect role is because low levels are more susceptible to noise and hence are more likely to appear random.

Table 1. The same nine genes as shown in Figure 4 and their ranks according to other maps γ

Gene name	γ_{inc}	γ_{opt}	γ_{+-}	γ_{Δ_1}	γ_{Δ_2}	γ_{Δ_3}	$k(f \mid \gamma_{\Delta_3})$
YPL061W	1476	21	72	21	8	1	16.2
YLR286C	3	5	1	8	1	2	13.8
YBR010W	28	12	25	104	13	3	13.6
YPR119W	492	51	82	11	2	4	13.1
YKL163W	67	7	11	125	26	5	12.2
YPL256C	486	2	5	207	17	6	11.9
YDR263C	1249	2478	1865	1598	1265	1000	0.1
YLR072W	2125	4926	4926	3772	4415	3000	−1.1
YKR083C	1607	4839	4806	3909	4994	5000	−1.3

In almost all cases the top six genes of γ_{Δ_3} are found in the top few percent of the other five rankings. This illustrates that all six γ maps provide useful bounds on the algorithmic compressibility. The last column is the compression in bits $k(f \mid \gamma_{\Delta_3})$.

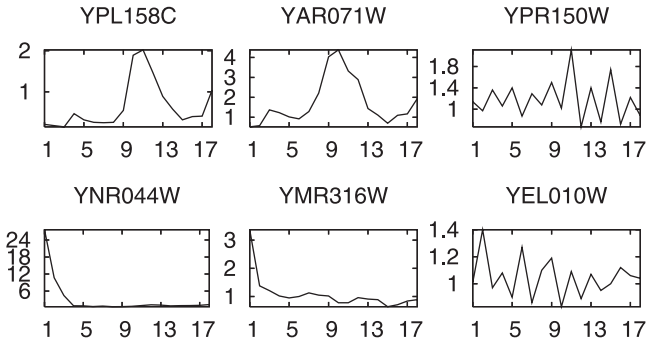

Fig. 5. Left, Two examples of cell cycle expression curves of genes inside our top 150, which were also experimentally identified by Spellman (I) (http://genome-www.stanford.edu/cellcycle/data/rawdata/knownGenes.doc) or Simon (II) (Simon *et al*., 2001). Neither exhibits two cycle behaviour. Middle, Two further examples of expression curves from our top 150, but which are not experimentally identified by Spellman (I) or Simon (II). Right: Two examples from the class of zig-zag curves which we believe to be the result of a systematic error in the experiment. These two curves are located in the right-hand side tail of the γ_{Δ_3} distribution in Figure 2, while the four other curves lie in the left-hand side tail of the same diagram.

5.2 Venn diagram analysis of histogram results

Our success in identifying yeast cell cycle genes is even more marked than Figure 3 suggests. The reason is as follows: Our goal, and the goal of experimental efforts behind sets I and II, is to approximate the unknown set of true cell cycle genes T as accurately as possible. However, we cannot validate our 150 predicted genes O with the true set T, but only with another approximation to T, in this case the 195 experimentally derived genes E (the union of I and II).

As a Venn diagram analysis (Fig. 6) shows, the overlap of O and E necessarily underestimates the overlap of O and T. Explicitly, the probability that a gene in O is in T is

$$P(O_i \in T) = \frac{P(O_i \in E) - P(F_i \in E)}{P(T_i \in E) - P(F_i \in E)},$$

where F is the set of false genes, the complement of T. If we take $|T| \ll |F|$ and $|E \cap T|$ on the order of $|E \cap F|$, then $P(F_i \in E) \ll 1$ and $P(O_i \in T)$ reduces to

$$P(O_i \in T) \simeq \frac{|E \cap O|}{|O|}\frac{|T|}{|E \cap T|} = \frac{P(O_i \in E)}{P(T_i \in E)}.$$

Our measured success rate $P(O_i \in E)$ is thus amplified by one over the probability that a gene in T is in E, the experimental efficacy. By this reckoning, to obtain the fraction of 60 in true genes in the histograms of Figure 3, the histogram bars should be divided by $P(T_i \in I)$ in the left column and by $P(T_i \in II)$ in the right column. That the efficacy $P(T_i \in E)$ is very likely to be less than one can be

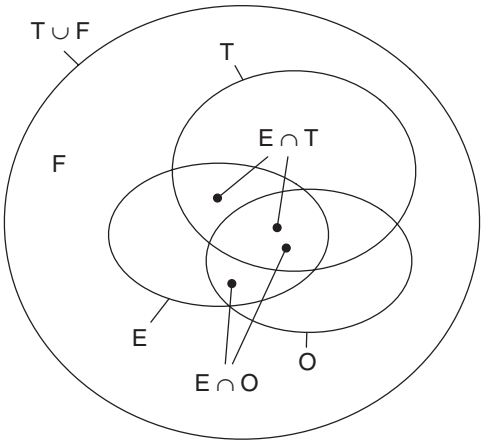

Fig. 6. An illustration of the sets mentioned in Section 5.2: (T)rue genes (relevent to cell cycle). (F)alse genes (ones which are not relevant to cell cycle). (E)xperimentally verified genes (such as in the sets I and II in Section 4) and (O)ur predicted genes.

seen from the Venn diagram of the sets I and II, shown at the top of Figure. 3. The rather small overlap of 49 genes between the two sets of 104 and 140 genes respectively implies the presence of false negatives (for a large 'true' set) or false positives (for a small 'true' set) in both these sets. The former case would give rise to a reduced $P(T_i \in E)$ which would lead to the amplification of our histograms as described. The latter case, which would imply a $P(T_i \in E)$ close to one, is less likely, as most experimental investigations as well as our analysis appear to indicate a number of cell cycle genes significantly larger than the size of the overlap, namely 49 genes. It has been estimated that 10% of all 6073 genes in yeast are involved in cell cycle (Payne and Garrels, 1997).

6 CONCLUSION

Our method of calculating a bound on the algorithmic compressibility of a given microarray data series provides an unbiased measure of pattern present in the series. This in turn gives an indication of the significance of the data series with respect to its underlying variable and thus creates a useful tool for the analysis of large collections of microarray data. A future application of this approach might also be to calculate a measure of mutual algorithmic compressibility between two data curves. This could be achieved using the permutation ordering of one data series as the underlying variable of the other. Such a measure would give rise to a weighted network of all data curves which then could be analysed in the context of regulatory genetic networks.

ACKNOWLEDGEMENTS

S. E. Ahnert was supported by Sidney Sussex College, Cambridge, UK, and the Association pour la Recherche sur le Cancer (ARC), France. K. Willbrand was supported by the Comite de Paris Ligue Nationale Contre le Cancer, France.

Conflict of Interest: A US patent of this method has been filed.

REFERENCES

Bar-Joseph,Z. *et al*. (2003) Comparing the continuous representation of time-series expression profiles to identify differentially expressed genes. *Proc. Natl Acad. Sci. USA*, **100**, 10146–10151.

Chaitin,G.J. (1966) On the lengths of programs for computing binary sequences. *J. Assoc. Comput. Mach.*, **13**, 547–569.

The Chipping Forecast II (2002), *Nat. Genet.*, **32** (suppl.), 461–552.

Cho,R.J. *et al*. (1998) A Genome-wide transcriptional analysis of the mitotic cell cycle. *Mol. Cell*, **2**, 65–73.

Cover,T.M. and Thomas,J.A. (1991) *Elements of Information Theory*. Wiley, New York.

Kolmogorov,A.N. (1965) Three approaches to the quantitative definition of information. *Prob. Inform. Transmission*, **1**, 4–7.

Weisstein,E.W. *et al*. 'Bounded Variation.' From MathWorld—A Wolfram Web Resource.

Payne,W.E. and Garrels,J.I. (1997) Yeast Protein Database (YPD): a database for the complete proteome of *Saccharomyces cerevisiae*. *Nucleic Acids Res*., **25**, 57–62.

Shannon,C.E. (1948) A mathematical theory of communication. *Bell Syst. Tech. J*., **27**, 379–423.

Simon,I. *et al*. (2001) Serial regulation of transcriptional regulators in the yeast cell cycle. *Cell*, **106**, 697–708.

Spellman,P.T. *et al*. (1998) Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol. Bio. Cell*, **9**, 3273–3297.

Wallace,C.S. and Dowe,D.L. (1999) Minimum message length and kolmogorov complexity. *Comput. J*., **4**, 270–283.

Warren,D.I. and Seneta,E. (1996) Peaks and eulerian numbers in a random sequence. *J. Appl. Prob*., **33**, 101–114.

Whitfield,M.L. *et al*. (2002) Identification of genes periodically expressed in the human cell cycle and their expression in tumors. *Mol. Biol. Cell*, **13**, 1977–2000.

Wichert,S. *et al*. (2004) Identifying periodically expressed transcripts in microarray time series data. *Bioinformatics*, **20**, 5–20.

Willbrand,K. *et al*. (2005) Identifying genes from up-down properties of microarray expression series. *Bioinformatics*, **21**, 3859–3864.