Supplementary Results For: Combining Static and Time Series Data to Determine the Quality of Expression Profiles in Time Series Experiments

Itamar Simon[*1], Siegfried Zahava[1], Jason Ernst[2] and Ziv Bar-Joseph[*,2,3,4]

[1]Dept. Molecular Biology, Hebrew University Medical School, Jerusalem, Israel 91120

[2] School of Computer Science and [3]Department of Biology, Carnegie Mellon University

5000 Forbes Ave. Pittsburgh, PA, 15213

[4]To whom correspondence should be addressed: zivbj@cs.cmu.edu

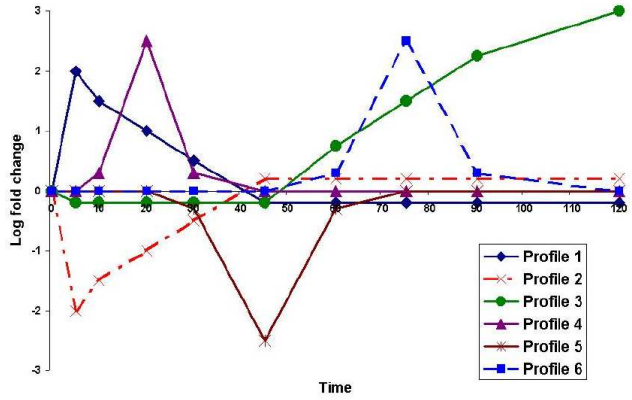
I. Simulated data

We have first tested our algorithm on simulated data to determine its ability to detect genes that are missed due to inadequate sampling rates. We generated six different profiles with 10 time points each (see Supplementary Fig. 2). One of these profiles (profile 6) contained a peak at the 8th time point, while the other five profiles where either flat or monotone at that point. We set M_i for each profile i to be the integral of a highly sampled set of points from the curve. We generated 100 sampled sets from each profile resulting in 600 simulated genes and added random noise to each of the values in these profiles. Following Cui $et\ al$ [1] the added noise was normally distributed with mean 0 and a *value dependent* standard deviation that was based on real microarray experiments (in this case, the alpha time series experiment in [2]).

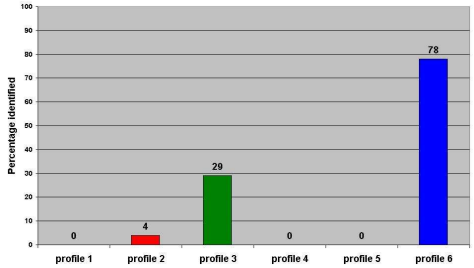
In order to simulate inadequate sampling, we have hidden the 8th time point (75 minutes) for all genes. Since most profiles are flat at this time point, an interpolation algorithm (such as the splines method described in Box 3) can easily overcome the missing point and generate an accurate representation of the profile using the 9 other time points. However, for profile 6, it is impossible to accurately reconstruct its true expression profile from the nine remaining time points. Thus, for this profile the reconstructed temporal expression curve will deviate significantly from its true underlying expression profile. This is exactly what we would like CheckSum to detect using the available average data.

Using a p-value cutoff of $.01$ CheckSum correctly identified 78% of the genes of profile 6 as genes lacking an important measurement (Supplementary Fig. 3b) . A few other genes that are expressed highly at the 8th time point (profile 3) were also detected, though most of these genes were correctly represented using splines due to their monotonous increase before and after the 8th time point. On the other hand, almost all genes that are flat at the 8th time point were not detected by CheckSum, indicating that these genes were correctly explained by the nine points used.

Supplementary Figure 2: Simulated data



(a) Expression profiles



(b) Genes identified for each profile.

Figure 2: Using synthetic data to test our algorithm. **(a)** The six profiles generated. Only profile 6 peaks at the 8th time point (75 minutes). The rest are either flat or monotone at that point. **(b)** Percentage of genes identified for each of the profiles. As can be seen, CheckSum correctly identified most of the genes in profile 6 as missing an important measurement while determining that most other genes are accurately represented.

II. Yeast cell cycle expression data

Raw yeast cell cycle expression data for the alpha synchronization experiment were downloaded from the Stanford Microarray Database (SMD, http://genome-www5.stanford.edu/). We have downloaded the values for the two channels (synchronized and unsynchronized samples) and computed their ratio for each time point. As mentioned above, noise models for individual time points were based on the values of the synchronized cells at that time point (channel 2 values).

Based on the analysis in previous papers [2, 3] we have used 63 minutes as the cell cycle duration. Since this dataset was sampled every 7 minutes, every ten consecutive time points represent the expression over one complete cycle (though the starting points change). For our analysis we have used all points between 28 to 91 minutes. This range was selected to eliminate arrest and release response artifacts. Similar results were obtained when using different start and end points.

Using a p-value of 0.01 Checksum identified 47 genes (32 had a p-value less than 0.001). To inspect this list we once again divided these genes to genes that were expressed at lower than expected levels ($S1$) and genes expressed at higher than expected levels ($S2$). Unlike with the human data, for yeast $S2$ contained more genes than $S1$ (27 and 20 respectfully) indicating that synchronization loss was not a problem with this dataset. Interestingly, $S1$ significantly intersected the list of cycling genes identified by Spellman $et\ al$ (12 of 20, p-value = $3 * 10^{-7}$). Note that the Spellman $et\ al$ list was derived based on two other datasets which could have compensated for minor errors in the alpha dataset. As Supplementary Figure 3 shows, at least for some of these genes it seems like noise or sampling problems prevented the correct identification of one of the peaks in that genes expression profile. Almost half of the genes in $S2$ where annotated as transposable elements, perhaps indicating that these genes are more noisy than most other genes.

Supplementary Figure 3: Yeast genes

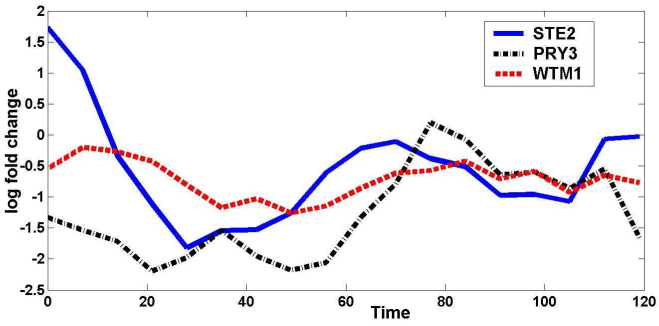


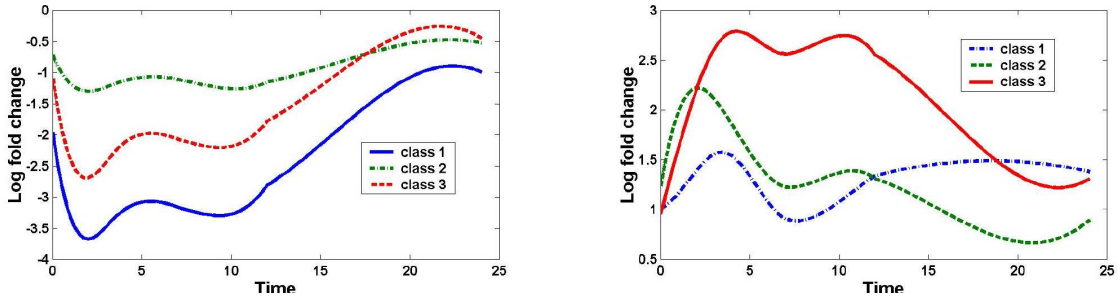
Figure 3: Genes identified in the yeast cell cycle experiment. These genes were determined by Checksum to be significantly underrepresented in the measured time series data (p-value of $8*10^{-8}$ for PRY3, 10^{-4} for STE2 and $3 * 10^{-4}$ for WTM1). All three genes were determined to cycling genes by Spellman $et\ al$ which relied on two other expression experiments. Since yeast cells were shown to be synchronized for two cycles, it is likely that the reason for this discrepancy between the time series and average data is due to noise or sampling problems. This conclusion if further strengthen by the fact that these genes contain peak values in one of the two cycles but not in the other (for example, STE2 is much higher in the first cycle compared with the second cycle while PRY3 is higher in the second.

The supporting website [4] contains the complete list of genes identified by Checksum along with a figure presenting the 12 genes in $S1$ that were determined to be cycling by Spellman $et\ al$. It is interesting to note that while the figures in the paper and on the website have the exact same shape as the results presented in Spellman $et\ al$, for each gene each time point differs from their figures by a constant value. The reason for this difference is that Spellman $et\ al$ normalized each row (gene) to sum to 0 where as we used the original measured values (and have only relied on global normalization among microarrays). By normalizing rows researchers make the implicit assumption that genes are accurately represented in the time series experiment. As we have shown in this paper, this is not always the case and so we believe that it is better to first use Checksum and only afterwards, when it is determined that the temporal profiles are accurate, to normalize the rows.

III. Human cell cycle expression data

Human cell cycle expression data was downloaded from the Stanford Microarray Database (SMD, http://genome-www5.stanford.edu/). We have downloaded the values for the two channels (synchronized and unsynchronized samples) and computed their ratio for each time point. Since our method only relies on this ratio, we did not have to explicitly compute the unsynchronized average values, which is an advantage when working with cDNA arrays. However, noise models for individual time points were based only on the values of the synchronized cells at that time point, and are thus value dependent as discussed above. We have used 24 hours as the estimate of the

Supplementary Figure 4: Profiles of $S1$ and $S2$ genes



(a) $S1$ classes (b) $S2$ classes

Figure 4: Clustering of $S1$ (left) and $S2$ genes. The number of genes for the three $S1$ clusters were: 42, 227 and 133 respectfully. The number of genes for the $S2$ clusters were: 46, 89 and 27. Note that while genes in $S1$ where selected so that they are mostly down regulated, the shape of their curves (initially down and than rising) is not a result of the requirements of the algorithm. The fact that most genes display this profile indicates that they are significantly involved in a specific function. As we discuss, many of these genes are known to be late acting (S, G2 and M phase) cycling genes. Similarly, for $S2$ there was no requirement that the expression be up early and than decline. As we show in the text, $S2$ genes are significantly associated with response to external stimulus and to serum.

duration of the cell cycle for this data. However, due to the probabilistic nature of our algorithm, the results do not drastically change when assuming that the cell cycle period is between 22 and 26 hours.

In Supplementary Figure 4 we present the expression profiles for genes in the two sets identified by CheckSum. We have used K-means (with $K = 3$) to cluster the continuous spline representation of each set. As described in the Results section the two sets had very different expression profiles. $S1$ genes were mainly down for the first half and then gradually rose to their average (unsynchronized) expression levels. $S2$ genes peaked early on, and then most of these genes declined drastically.

VI. Comparison of spline and LOESS interpolation methods

We use a spline based interpolation method to represent genes using continuous curves. This preprocessing step uses a method we have outlined in a previous paper [3]. Unlike traditional spline assignment, this method relies on co-expressed genes to overcome noise and missing values in individual genes. Thus, spline assignment is combined with a clustering algorithm and profiles benefit from both, local gene based information (via splines) and global class information (via clustering). In that paper we have carried out many tests to compare this method to all previously suggested methods for interpolating time series expression data. Using a cross validation test it was shown that our method outperforms all such methods by at least 15%.

In response to a comment from one of the reviewers of this paper we have carried out addi-

Supplementary Figure 5: Comparison of spline and LOESS interpolation

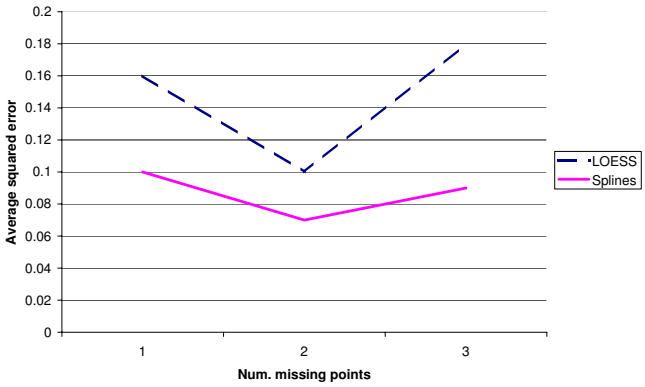


Figure 5: Comparison between spline interpolation (solid) and LOESS interpolation (dashed) using a cross validation test. The x axis represents the number of consecutive points hidden and the y axis lists the average square difference between the predicted and actual value for the points hidden. As can be seen, in all cases our spline based approach outperforms the LOESS method. For one missing point the difference is bigger than 50%.

tional tests to compare our spline assignment method to a LOESS method which relies on a locally weighted least square method to assign continuous values to expression profiles. We have downloaded a Matlab implementation of LOESS from Datatool (http://www.datatool.com) and used the recommended values for testing (a second degree polynomial and a smoothing value of .5). Other values yielded similar results.

We have used the yeast alpha dataset discussed in this paper to a carry out a cross validation test to compare the two approaches. We have used the 612 cycling genes (from the 800 genes identified by Spellman $et\ al$) that did not have any missing values. Next, we have selected 50 genes at random from this list and have hidden one of the measured values for each of these genes. We used our method and the LOESS method to interpolate the genes with missing values and computed the average square difference between their predictions and the values that were hidden. This was repeated 10 time using a different set of 50 genes, and results where averaged. We have also repeated the same test holding two and three consecutive points for each gene.

The results are summarized in Supplementary Figure 5. As can be seen, in all three cases (1, 2 and 3 consecutive points) the spline based approach, which relies on global similarity top co-expressed genes outperform the LOESS method by as much as 50%. These results agree with the results published in our original paper and in other papers [5] indicating that relying on co-expressed genes is a useful method for correcting noise in expression data.

References

[1] X. Cui, J.T. Hwang, J. Qiu, N.J. Blades, and G.A. Churchill. Improved statistical tests for differential gene expression by shrinking variance components estimates. *Biostatistics*, 6(1):59–75, 2005.

[2] P. T. Spellman, G. Sherlock, M.Q. Zhang, V.R. Iyer, K. Anders, and $et\ al$. Comprehensive identification of cell cycle-regulated genes of the yeast $saccharomyces\ cerevisia$ by microarray hybridization. *Mol. Biol. of the Cell*, 9:3273–3297, 1998.

[3] Z. Bar-Joseph, G. Gerber, T.S. Jaakkola, D.K. Gifford, and I. Simon. Continuous representations of time series gene expression data. *Journal of Computational Biology*, 3-4:341–356, 2003.

[4] *CheckSum for Time Series Expression Data*. URL: http://www.cs.cmu..edu/∼zivbj/checksum/checksum.html.

[5] O. Troyanskaya, M. Cantor, and $et\ al$. Missing value estimation methods for dna microarrays. *Bioinformatics*, 17:520–525, 2001.