Introduction to cycle package

Matthias E. Futschik
SysBioLab, University of Algarve, Faro, Portugal
URL: *http://cycle.sysbiolab.eu*

October 17, 2016

Contents

1 Overview

Periodicity is an important phenomenon in molecular biology and physiology. Prominent examples are the cell cycle and the circadian clock. Microarray array technology has enabled us to screen complete sets of transcripts for possible association with such fundamental periodic processes on a system-wide level. To assess the significance of the identified periodic expression, several approaches for detection have been proposed based on time series analysis and statistical modeling. Most of the proposed methods rely on data normality or the extensive use of permutation tests. However, this neglects the fact that time series data exhibit generally a considerable autocorrelation i.e. correlation between successive measurements. Therefore, neither the assumptions of data normality nor for randomizations may hold.

This failure can substantially interfere with the significance testing, and that neglecting autocorrelation can potentially lead to a considerable overestimation of the number of periodically expressed genes [1]. Notably, randomized and Gaussian background models neglect the dependency structure within the observed data. In contrast, the use of autoregressive AR(1) background models gave a more accurate representation of correlations between measurements. More importantly, the choice of background model has drastic effects on the number of genes detected as significantly periodically expressed. A study of expression data of yeast cell cycle showed clearly that randomized and Gaussian background models tend to overestimate the number of significant periodically expressed genes [1]. Strikingly, the use of the more

accurate AR(1)-background led to a considerable reduction of the number of periodic genes. Most importantly, AR(1)-based models achieve superior accuracy in determining periodically expressed genes as a subsequent assessment using benchmark datasets demonstrated.

This vignette gives a short introduction to the *cycle* package which can be employed to assess the significance of periodic expression using Fourier analysis and different background models. Its usage is illustrated by the re-analysis of yeast cell cycle data. More information and references can be found at the *cycle* webpage:
http://cycle.sysbiolab.eu

2 Installation requirements

Following software is required to run the *cycle* package:

- R ($>$ 2.0.0). For installation of R, refer to *http://www.r-project.org*.
- Bioconductor packages: Biobase and Mfuzz. Refer to *http://www.bioconductor.org* for installation.

If these requirements are fulfilled, the *cycle* add-on R-package can be installed. To see how to install add-on R-packages on your computer system, start *R* and type in *help(INSTALL)*. Optionally, you may use the R-function *install.packages()*. Once the *cycle* package is installed, you can load the package by

```
>   library(cycle)
```

3 Case study: Yeast cell cycle

To illustrate the impact of different background models on the significance of periodic expression, we apply re-analyse yeast cell cycle expression data by Cho *et al.* [2]. 6178 genes were monitored at 17 time points over a span of 160 minutes using Affymetrix chips. Note that we include here only the first 100 genes for illustration purpose.

```
> data(yeast)
>   yeast <- yeast[1:200,]
```

3.1 Missing values

As a first step, we exclude genes with more than 25% of the measurements missing. Note that missing values should be denoted by `NA` in the gene expression matrix.

```
> yeast <- filter.NA(yeast, thres=0.25)
```

```
3 genes excluded.
```

The calculation of the Fourier scores does not allow for missing values. Thus, we replace remaining missing values by the average values expression value of the corresponding gene.

```
> yeast <- fill.NA(yeast,mode="mean")
```

Alternatively (and recommended), the (weighted) k-nearest neighbour method can be used (`mode='knn'/'wknn'`). These methods perform usually favourable compared to the simple method above, but are computationally intensive.

3.2 Standardisation

For the calculation of Fourier scores that the expression values are standardised i.e. have a mean value of zero and a standard deviation of one.

```
> yeast <- standardise(yeast)
```

3.3 Statistical assessment of periodicity

3.3.1 Choice of background model

Microarray data comprise the measurements of transcript levels for many thousands of genes. Due to the large number of genes, it can be expected that some genes show periodicity simply by chance. To assess therefore the significance of periodic signals, it is necessary first to define what distribution of signals can be expected if the studied process exhibits no true periodicity. In statistical terms this is equivalent with the definition of a null hypothesis of non-periodic expression.

The most simple model for non-periodic expression is based on randomization of the observed times series. A background distribution can then be constructed by (repeated) random permutation of the sequentially ordered measurements in the experiment. Alternatively, non-periodic expression can be derived using a statistical model. A conventional approach is based on the assumption of data normality and to use the normal distribution.

However, these two approaches neglect the fact that time series data exhibit generally a considerable autocorrelation i.e. correlation between successive measurements. In our case, we find

```
> auto.corr <- 0
> for (i in 2:dim(exprs(yeast))[[2]]){
+  auto.corr[i] <- cor(exprs(yeast)[,i-1],exprs(yeast)[,i])
+  }
> auto.corr

 [1]  0.0000000  0.1378803  0.3759064  0.5166133  0.2395612  0.3109068
 [7]  0.3839816  0.2586308  0.2486221 -0.1037879  0.3458338  0.3515935
[13]  0.2119740  0.2732031  0.2348118  0.2650634  0.4173739
```

Therefore, neither the assumptions of data normality nor for randomizations may hold. As we have showed for yeast cell cycle data [1], this failure can substantially interfere with the significance testing, and that neglecting autocorrelation can potentially lead to a considerable overestimation of the number of periodically expressed genes.

A more suitable model is based on autoregressive processes of order one (AR(1)), for which the value of the time-dependent variable X depends on its previous value up to a normally distributed random variable Z. The autocorrelation of X and variance of Z is estimated for each feature of the ExpressionSet object separately. More details can be found in the given reference [1].

It is important to note in this context, that AR(1) processes cannot capture periodic patterns except for alternations with period two. Since Z is a random variable, we can readily generate a collection of time series with the same autocorrelation as in the original data set. Therefore, although AR(1) processes constitute random processes, they allow us to

construct a background distribution that captures the autocorrelation structure of original gene expression time series without fitting the potentially included periodic pattern.

3.3.2 Fourier analysis to detect periodic expression

To detect periodic signals within the large datasets, several different approaches have been put forward ranging from simple visual inspection to elaborated statistical models. Recently, an extensive comparison showed that a relatively simple method using Fourier analysis performs better than other approaches [3].

Thus, the detection of periodic signals is based here on the calculation of Fourier scores. The closer a gene's expression follows a (possibly shifted) cosine curve of cycle period, the larger is the Fourier score. Mathematical details can be found in the reference [1]. It should be noted that the influence of the background model is not restricted to Fourier analysis, but is equally prominent for other approaches neglecting autocorrelation.

To calculate the Fourier score, the cycle period has to be given. For the yeast cell cycle data, the value were taken from the original publication i.e. T = 85 min.

```
> T.yeast <- 85
```

Additionally, the times of measurement have to be stated. In our case, they are are already included in the phenoData slot of the ExpressionSet object (*yeast*):

```
> times.yeast <-  pData(yeast)$time
> times.yeast
```

```
 [1]   0  10  20  30  40  50  60  70  80  90 100 110 120 130 140 150 160
```

3.3.3 Derivation of false discovery rates

To assess the significance of the Fourier score obtained for the original gene expression time series, the probability has to be calculated of how often such a score would be observed by chance based on the chosen background distribution. The statistical significance is given by the calculated false discovery rate. It is defined here as the expected proportion of false positives among all genes detected as periodically expressed. Mathematical details can be found in the given reference [1].

The main function of the *cycle* package is *fdrfourier*. It calculates the Fourier scores for the observed expression data and derives the corresponding false discovery rates based on the comparison with the Fourier scores obtained for the background data. As the number of generated background data sets, we choose e.g.

```
> NN <- 100
```

for illustration purpose. It may be necessary to choose a larger number i.e. $NN = 1000$. Note that the calculation of FDRs employing empirical background distributions for large expression data sets can require considerable time (up to several days).

First, we calculate the false discovery rates using a random permutation to generate the background data:

```
> fdr.rr <- fdrfourier(eset=yeast,T=T.yeast,times=times.yeast,background.model="rr",N=NN,p
```

For larger datasets or *NN*, it may be helpful to set the *progress* argument to TRUE allowing the user to monitor the progress of the calculations.

Subsequently, we derive the false discovery rates using AR1 models to generate the background data

```
> fdr.ar1 <- fdrfourier(eset=yeast,T=T.yeast,times=times.yeast,background.model="ar1",N=NN
```

(Note that this function evaluates soley the *exprs* matrix and no information is used from the *phenoData*. In particular, the ordering of samples (arrays) is the same as the ordering of the columns in the *exprs* matrix. Also, replicated arrays in the *exprs* matrix are treated as independent i.e. they should be averagered prior to analysis or placed into different distinct *ExpressionSet* objects.)

The comparison of the number of significant gene (for e.g. FDR < 0.25)

```
> sum(fdr.rr$fdr < 0.25)

[1] 33

> sum(fdr.ar1$fdr < 0.25)

[1] 7
```

indicates that neglecting of the observed autocorrelation can potentially lead to a considerable overestimation of the number of periodically expressed genes.

Finally, we list the genes with significant periodicity and the corresponding false discovery rate

```
> fdr.ar1$fdr[which(fdr.ar1$fdr < 0.25)]

  YMR296C   YGL101W   YGL114W   YOR319W   YDL105W   YOL112W   YBR275C
0.1350000 0.1966667 0.1980000 0.2228571 0.1975000 0.2425000 0.2300000
```

References

[1] Matthias E. Futschik and Hanspeter Herzel (2008) Are we overestimating the number of cell-cycling genes? The impact of background models on time series analysis, *Bioinformatics*, 24(8):1063-1069

[2] Cho RJ, Campbell MJ, Winzeler EA, Steinmetz L, Conway A, Wodicka L, Wolfsberg TG, Gabrielian AE, Landsman D, Lockhart DJ, Davis RW, A genome-wide transcriptional analysis of the mitotic cell cycle, *Mol Cell*, **2**:65–73, 1998

[3] de Lichtenberg, U., L.J. Jensen, A. Fausboll, T.S. Jensen, P. Bork, and S. Brunak. 2005. Comparison of computational methods for the identification of cell cycle-regulated genes. *Bioinformatics* 21: 1164-1171.