Bioinformatics, 32(21), 2016, 3351–3353
doi: 10.1093/bioinformatics/btw405
Advance Access Publication Date: 4 July 2016
Applications Note

OXFORD

Systems biology

MetaCycle: an integrated R package to evaluate periodicity in large scale data

Gang Wu[1], Ron C. Anafi[2,3], Michael E. Hughes[4], Karl Kornacker[5] and John B. Hogenesch[1,*]

[1]Department of Systems Pharmacology and Translational Therapeutics, Institute for Translational Medicine and Therapeutics, [2]Division of Sleep Medicine, [3]Center for Sleep and Circadian Neurobiology, University of Pennsylvania School of Medicine, Philadelphia, PA 19104, USA, [4]Department of Biology, University of Missouri–St. Louis, St. Louis, MO 63121, USA and [5]Emeritus Professor of Sensory Biophysics, the Ohio State University, Columbus, OH 43210, USA

*To whom correspondence should be addressed.
Associate Editor: Jonathan Wren

Received on February 26, 2016; revised on June 16, 2016; accepted on June 20, 2016

Abstract

Summary: Detecting periodicity in large scale data remains a challenge. While efforts have been made to identify best of breed algorithms, relatively little research has gone into integrating these methods in a generalizable method. Here, we present MetaCycle, an R package that incorporates ARSER, JTK_CYCLE and Lomb-Scargle to conveniently evaluate periodicity in time-series data. MetaCycle has two functions, meta2d and meta3d, designed to analyze two-dimensional and three-dimensional time-series datasets, respectively. Meta2d implements N-version programming concepts using a suite of algorithms and integrating their results.

Availability and implementation: MetaCycle package is available on the CRAN repository (https://cran.r-project.org/web/packages/MetaCycle/index.html) and GitHub (https://github.com/gangwug/MetaCycle).

Contact: hogenesch@gmail.com

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Identifying periodic signals in time-series data is important in studying oscillatory systems, such as the circadian clock or cell cycle. Researchers have devised and evaluated many methods (see a short list in Doherty and Kay, 2010), each with strengths and weaknesses (e.g. Deckard *et al.*, 2013; Wu *et al.*, 2014). For example, Lomb-Scargle (Glynn *et al.*, 2006) and JTK_CYCLE (Hughes *et al.*, 2010) can differentiate between periodic and non-periodic profiles for cosine curves with high noise or low sampling rate (Deckard *et al.*, 2013). On the other hand, some popular methods, including Fisher's G Test (Fisher, 1929; Wichert *et al.*, 2004) and ARSER (Yang and Su, 2010), fail without complete, evenly sampled time series data. To avoid such 'mode failure' and capture the most value from data, one could choose the best algorithm for each experiment. However, for any single experiment the choice of optimal algorithm is often unclear and may depend on unknown features of the data, such as the shape of the oscillatory patterns or the nature of the noise. Alternatively, a fault-tolerant method that avoids mode failure would have wider applicability, insulating the user from tricky decisions.

Here, we present MetaCycle, an N-version programming (NVP) method to explore periodic data. NVP (Avizienis and Chen, 1977) is a concept from the aeronautics industry. Aircraft guidance systems employ multiple independent algorithms and a voting scheme to integrate their results. A method that performs poorly in a particular condition will be outvoted by other methods that better accommodate that condition. NVP has three elements: (i) an initial specification that addresses the problem to be solved, data formats, and methods of integrating specified variables from N-version programs, (ii) two or more independent algorithmic approaches (N-version software, or NVS) to solve the problem and (iii) an execution

environment (N-version executive, or NVX) that runs NVS and provides decision algorithms (Avizienis, 1995; Chen and Avizienis, 1978).

MetaCycle incorporates these concepts into its meta2d mode, which analyzes data from a single time series. Using the same input file, MetaCycle::meta2d implements ARSER (ARS), JTK_CYCLE (JTK) and Lomb-Scargle (LS), from dramatically different disciplines, computer science, statistics and physics, respectively. After running these algorithms, meta2d integrates their results in a common output file in the (R) environment. We show that MetaCycle::meta2d avoids mode failure while providing robust power in rhythm detection and accurate phase estimations.

Further, researchers often want to integrate results from multiple individuals, preserving individual contributions to an overall result. For analyzing data from multiple time series (e.g. several individuals), MetaCycle provides another mode, MetaCycle::meta3d, a convenient and efficient tool to integrate multiple large-scale time series datasets from individual subjects.

2 Methods

Our functional requirement, the initial specification step of NVP, is identifying periodic signals from time-series datasets. At this step, we also specify input data formats, comparison vectors (c-vectors; including *P*-values, period length and phase), the decision algorithm, and where it will be applied (cc-points). A detailed description of the decision algorithm is in the vignette. In brief, as default, Fisher's method (Fisher, 1925) is used to integrate *P*-values. Arithmetic and circular means are applied in period and phase integration, respectively. The c-vectors do not include amplitude, which is re-calculated with the ordinary least squares method by constructing a general periodic model. Three independent algorithms (ARS, JTK and LS) were then selected from best of breed methods (Deckard *et al.*, 2013; Wu *et al.*, 2014) and developed into NVS. At the third step, we used the R language as the N-version execution environment (NVX). R assures input consistency for each version, local supervision, and implementation of the decision algorithm.

3 Results and discussion

In model organism experiments, genetically similar animals are typically used, making the identification of each individual sample unimportant. Indeed, in the vast majority of such experiments, only one sample is used from any individual. With human research, though, individual subjects are usually recruited, with samples (e.g. blood, skin) collected from the same person over time. Unless samples are pooled, this generates multiple time series datasets. We classify single time-series datasets from one individual or pools as 2D, while multiple time series datasets are designated 3D (Supplementary Fig. S1). Two functions-meta2d and meta3d in MetaCycle are designed to analyze time-series datasets in these categories. MetaCycle allows users to select one or more cycling detection algorithms from ARS, JTK and LS to analyze their datasets. Using the same input data format, analyzing different kinds of time-series datasets (evenly or unevenly sampled, with or without replicates or missing values) becomes much easier. Additionally, MetaCycle reports integrated results when analyzing single time-series datasets with two or three algorithms or when analyzing multiple time series with the same algorithm.

Using simulated data generated from several different oscillatory reference waves and adding normally distributed noise, we show that NVP generally performs better than using single algorithm in reporting cycling (Supplementary Figs S2 and S3) or phase values (Supplementary Fig. S4). In some cases, NVP may not give better results than the single most suitable method, but it will rarely give the worst results. For example, in analyzing time-series datasets with high temporal resolution (every 1 h over 2 days; Supplementary Fig. S3), NVP does better than ARS but not as well as LS or JTK (Under these conditions, all three methods do relatively well but have similar mode failure, while ARS has relatively more false positive observations.). We also applied the NVP method on experimental time-series data studying mammalian circadian rhythms (Hughes et al., 2009) and the yeast cell cycle (Orlando *et al.*, 2008). In addition to finding the majority of cycling transcripts in the original paper (90.9, 83.8 and 55.6% of 24, 12 and 8 h cycling transcripts from mouse liver at FDR < 0.01; Supplementary Table S1), there are 1223 (Supplementary Fig. S5), 46 (Supplementary Fig. S6) and 13 (Supplementary Fig. S7) of 24, 12 and 8 h cycling transcripts specially identified by the NVP method. In addition, in yeast cell cycle dataset, the NVP method specifically identified 377 periodic genes that were not listed in the original analyses (Supplementary Fig. S8).

With more and different large-scale time-series datasets being produced (e.g. ChIP-seq, proteomics, metabolite profiling), tools for better handling these data are needed. Selecting the best algorithm for each dataset can be tricky, as experimental designs and biological unknowns can interact to produce a myriad of complications. Further, algorithms often require different input formats and runtime parameters and sometimes are implemented in different languages (e.g. in R, Python, or MATLAB). To address these problems, we have developed an NVP method using three popular methods—ARS, JTK and LS—in the MetaCycle package in R. To facilitate its use and further development (e.g. new periodicity detection or integration algorithms), we have made the source code available via Github and the program available through CRAN.

Acknowledgements

The authors thank Dr. Rendong Yang for his input on applying ARSER, Siqi Liu and Dr. Ben Voight for helpful discussions about Fisher's method for integrating *P*-values, Dr. Yihui Xie for helpful advice about implementing a web application of MetaCycle and Dr. Steve Haase for information on his yeast dataset. We also thank members of the Hogenesch lab for valuable discussion and advice on this project.

Funding

This work is supported by the National Institute of Neurological Disorders and Stroke [5R01NS054794-08 to JBH] and the Defence Advanced Research Projects Agency [DARPA-D12AP00025, to John Harer, Duke University].

Conflict of Interest: none declared.

References

Avizienis,A. and Chen,L. (1977) On the Implementation of N-version Programming for Software Fault Tolerance during Execution. In: *Proceedings of COMPSAC 77 (First IEEE-CS International Computer Software and Application Conference)*, pp. 149–155.

Avizienis,A.A. (1995) *The Methodology of N-Version Programming, Software Fault Tolerance* John Wiley & Sons Ltd., New York, pp. 23–46.

Chen,L. and Avizienis,A. (1978) N-version programming: a fault-tolerance approach to reliability of software operation. In: *Digest of 8th FTCS*, pp. 3–9.

Deckard,A. *et al.* (2013) Design and analysis of large-scale biological rhythm studies: a comparison of algorithms for detecting periodic signals in biological data. *Bioinformatics*, **29**, 3174–3180.

Doherty,C.J. and Kay,S.A. (2010) Circadian control of global gene expression patterns. *Annu. Rev. Genet.*, **44**, 419–444.

Fisher,R.A. (1925) *Statistical Methods for Research Workers*. Oliver and Boyd, Edinburgh.

Fisher,R.A. (1929) Tests of significance in harmonic analysis. *Proc. R. Soc. A*, **125**, 54–59.

Glynn,E.F. *et al*. (2006) Detecting periodic patterns in unevenly spaced gene expression time series using Lomb–Scargle periodograms. *Bioinformatics*, **22**, 310–316.

Hughes,M.E. *et al*. (2009) Harmonics of circadian gene transcription in mammals. *PLoS Genet.*, **5**, e1000442.

Hughes,M.E. *et al*. (2010) JTK_CYCLE: an efficient nonparametric algorithm for detecting rhythmic components in genome-scale data sets. *J. Biol. Rhythms*, **25**, 372–380.

Orlando,D.A. *et al*. (2008) Global control of cell-cycle transcription by coupled CDK and network oscillators. *Nature*, **453**, 944–947.

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

Wu,G. *et al*. (2014) Evaluation of five methods for genome-wide circadian gene identification. *J. Biol. Rhythms*, **29**, 231–242.

Yang,R. and Su,Z. (2010) Analyzing circadian expression data by harmonic regression based on autoregressive spectral estimation. *Bioinformatics*, **26**, i168–i174.