BIOINFORMATICS APPLICATIONS NOTE Vol. 31 no. 2 2015, pages 277–278
doi:10.1093/bioinformatics/btu612

Systems biology Advance Access publication September 21, 2014

NAIL, a software toolset for inferring, analyzing and visualizing regulatory networks

Daniel G. Hurley[1,2,3,4,*], Joseph Cursons[1,4], Yi Kan Wang[1,5], David M. Budden[4], Cristin G. Print[2,3,6] and Edmund J. Crampin[1,4,7,8]

[1]Auckland Bioengineering Institute, University of Auckland, Auckland 1001, New Zealand, [2]Department of Molecular Medicine and Pathology, School of Medical Sciences, Faculty of Medical and Health Sciences,University of Auckland, Auckland 1001, New Zealand, [3]Bioinformatics Institute, University of Auckland, Auckland 1001, New Zealand, [4]Maurice Wilkins Centre, University of Auckland, Auckland 1001, New Zealand, [5]Systems Biology Laboratory, Melbourne School of Engineering, University of Melbourne, Victoria 3010, Australia, [6]Department of Mathematics and Statistics, University of Melbourne [7]School of Medicine, University of Melbourne, Victoria 3010, Australia and [8]Department of Molecular Oncology, British Columbia Cancer Agency, Vancouver, Canada

Associate Editor: Igor Jurisica

ABSTRACT

Summary: The wide variety of published approaches for the problem of regulatory network inference makes using multiple inference algorithms complex and time-consuming. Network Analysis and Inference Library (NAIL) is a set of software tools to simplify the range of computational activities involved in regulatory network inference. It uses a modular approach to connect different network inference algorithms to the same visualization and network-based analyses. NAIL is technology-independent and includes an interface layer to allow easy integration of components into other applications.

Availability and implementation: NAIL is implemented in MATLAB, runs on Windows, Linux and OSX, and is available from SourceForge at https://sourceforge.net/projects/nailsystemsbiology/ for all researchers to use.

Contact: daniel.hurley@unimelb.edu.au

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

Received on May 11, 2014; revised on August 22, 2014; accepted on September 11, 2014

1 INTRODUCTION

Modelling biological systems as networks (graphs) is becoming a common approach in the life sciences. Specifically, inferring regulatory networks from genomic, transcriptomic and proteomic data is now giving insight into how biological systems function in healthy and pathological states (Ideker and Krogan, 2012). In a clinical context, ‘network medicine’ is becoming increasingly important in developing treatments for complex multifactorial conditions (Barabasi *et al*., 2011).

Many algorithms or methods have been published to reverse-engineer regulatory networks from genomic and transcriptomic data; comprehensive reviews of different inference algorithms and methods can be found in De Smet and Marchal (2010). However, different algorithms typically use different input and output data types, are implemented using different technologies and are demonstrated by application to different biological problems. In addition, individual algorithms often evaluate and visualize output in different ways, and tools for evaluation and visualization are generally not included in the released package for each algorithm. These range of technologies, data formats and invocation methods make it complex and labour-intensive to deploy and use multiple algorithms on a single dataset.

Motivated by results from the DREAM reverse-engineering consortium suggesting that combinations of network inference algorithms can produce more biologically relevant networks than individual algorithms (Prill *et al*., 2011), we developed an extensible framework to simplify the use of multiple network inference methods simultaneously, and to visualize and compare their results. In a previous study (Hurley *et al*., 2011), we described the initial development of this work and illustration of different biological meanings for regulatory networks inferred using different methods. In this article, we describe a new framework developed from our previous methods, this time intended for researchers in systems biology and bioinformatics, called NAIL (Network Analysis and Inference Library). NAIL is intended to perform network analyses using a number of published methods and to visualize the results. Additional functionality of NAIL includes the generation of networks from bootstrapped and clustered datasets, and the evaluation of predictive power using both cross-validation and resampling methods.

We stress that NAIL is not a standard project, and does not confine researchers to specific algorithms or approaches. Instead, it is a pragmatic and extensible collection of tools designed to address a current barrier to the broader use of network inference algorithms.

2 IMPLEMENTATION

Within NAIL, each major task in network inference is defined as a *component*. A component is a separable piece of code performing a defined task with input and output that are meaningful to the researchers—for instance, log transforming the data, or taking in a dataset and inferring a network from it. Figure 1 shows a list of stages of network inference and common components used

[*]To whom correspondence should be addressed.

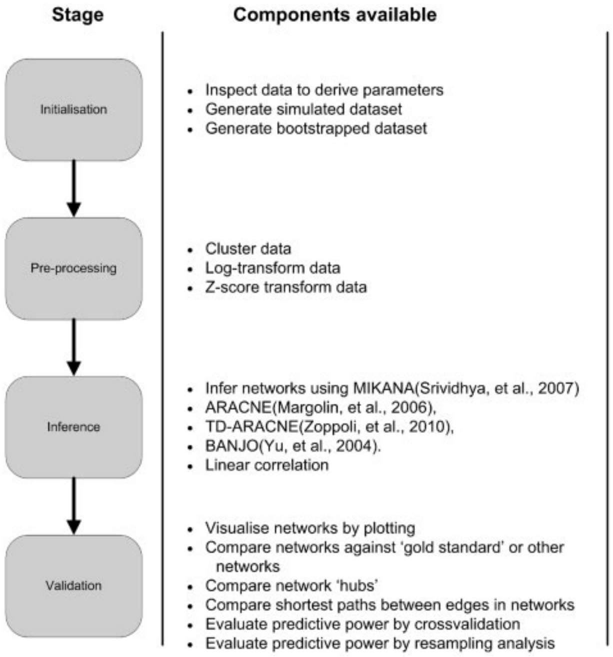
Stage
Components available
Initialisation
• Inspect data to derive parameters
• Generate simulated dataset
• Generate bootstrapped dataset
Pre-processing
• Cluster data
• Log-transform data
• Z-score transform data
Inference
• Infer networks using MIKANA(Srividhya, et al., 2007)
• ARACNE(Margolin, et al., 2006),
• TD-ARACNE(Zoppoli, et al., 2010),
• BANJO(Yu, et al., 2004).
• Linear correlation
Validation
• Visualise networks by plotting
• Compare networks against 'gold standard' or other networks
• Compare network 'hubs'
• Compare shortest paths between edges in networks
• Evaluate predictive power by crossvalidation
• Evaluate predictive power by resampling analysis

Fig. 1. High-level schematic of NAIL components and their uses. Third-party inference methods integrated into NAIL include MIKANA (Srividhya *et al.*, 2007), ARACNE (Margolin *et al.*, 2006), TD-ARACNE (Lizier, 2014; Yu *et al.*, 2004; Zoppoli *et al.*, 2010). A complete list can be found in the Supplementary Data

in those stages. Each component is controlled by a plain-text configuration file; any tool with the ability to read and write plain-text files, and to write to a file system, can interface easily with any of the functions in the framework. To use a component, a user will make changes to its configuration file, then invoke the component using another language, or from the command line. Components in the NAIL are divided into four separate categories: *initialization*, *preprocessing, inference* and *evaluation*.

The table above lists components arranged by type. Components have full access to the file system, so they can call other components, and write to their configuration files, allowing a set of components to be called in sequence from any technology that can execute code from the command line.

The packaged 'validation' components are a key feature of NAIL, as they are not included in currently available tools. When used with data bootstrapping/clustering, and network cross-validation, these components allow networks generated from different inference methods to be compared using a single set of metrics. Components are also included to compare and visualize the relative connectivity of hubs between networks, and to compare and visualize the shortest paths between individual nodes (examples shown in Supplementary Data*)*. These comparison tools produce a richer understanding of the differences between different network inference methods that can be gained by comparing edges alone.

3 EXAMPLES OF USE

A typical workflow using NAIL components could involve selecting a subset of an experimental dataset (initialization stage), then applying a log transformation to the data (preprocessing stage), inferring networks using different algorithms (inference stage) and finally comparing the inferred networks in terms of their connectivity and degree of edgewise overlap, as well as the relative connectivity of their hubs (validation stage). Documentation demonstrating the use of these components, along with input and output from each step in this workflow is shown in Supplementary Data, and we encourage readers to recreate these using the settings provided.

4 CONCLUSION

NAIL is a technology-independent and format-neutral platform for performing a variety of regulatory network inference activities. We believe that integrative toolsets like NAIL can be relevant to the work of experimentally focused researchers, and can become a regular part of genomic and proteomic research.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the assistance of K. Wong and M. Pan with review, feedback and software testing.

Funding: This work was supported by University of Auckland Doctoral Scholarship (to D.H, J.C, Y,W.); Top Achiever Doctoral Scholarship (to J.C.); Maurice Wilkins Centre for Molecular Biodiscovery Studentship (to D.H.); Auckland Bioengineering Institute (to D.H.); Australian Postgraduate Award (to D.M.B.); Marsden Fund of the Royal Society of New Zealand [06-UOA-182 to E.J.C.]; Health Research Council of New Zealand International Investment Opportunities Fund [06/581 to C.P.]; University of Auckland Faculty of Medical and Health Sciences Research Development Fund [#8303460 to C.P.]; and University of Auckland Vice Chancellor's Development Fund [#23285 to C.P.].

Conflict of interest: none declared.

REFERENCES

Barabasi,A.L. *et al.* (2011) Network medicine: a network-based approach to human disease. *Nat. Rev. Genet.*, **12**, 56–68.

De Smet,R. and Marchal,K. (2010) Advantages and limitations of current network inference methods. *Nat. Rev. Microbiol.*, **8**, 717–729.

Hurley,D. *et al.* (2011) Gene network inference and visualization tools for biologists: application to new human transcriptome datasets. *Nucleic Acids Res.*, **40**, 2377–2398.

Ideker,T. and Krogan,N.J. (2012) Differential network biology. *Mol. Syst. Biol.*, **8**, 565.

Lizier,J.T. (2014) JIDT: an information-theoretic toolkit for studying the dynamics of complex systems, arXiv:1408.3270.

Margolin,A.A. *et al.* (2006) ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, **7** (**Suppl. 1**), S7.

Prill,R.J. *et al.* (2011) Crowdsourcing Network Inference: the DREAM predictive signaling network challenge. *Sci. Signal.*, **4**, mr7.

Srividhya,J. *et al.* (2007) Reconstructing biochemical pathways from time course data. *Proteomics*, **7**, 828–838.

Yu,J. *et al.* (2004) Advances to Bayesian network inference for generating causal networks from observational biological data. *Bioinformatics*, **20**, 3594–3603.

Zoppoli,P. *et al.* (2010) TimeDelay-ARACNE: reverse engineering of gene networks from time-course data by an information theoretic approach. *BMC Bioinformatics*, **11**, 154.