Systems Biology

pyNBS: A Python implementation for network-based stratification of tumor mutations

Justin K. Huang[1] *, Tongqiu Jia[2], Daniel E. Carlin[2], Trey Ideker[1,2]

[1] Bioinformatics and Systems Biology Program, UC San Diego, La Jolla, California, USA, [2] Department of Medicine, UC San Diego, La Jolla, California, USA

*To whom correspondence should be addressed.

Abstract

Summary: We present pyNBS: a modularized Python 2.7 implementation of the network-based stratification (NBS) algorithm for stratifying tumor somatic mutation profiles into molecularly and clinically relevant subtypes. In addition to release of the software, we benchmark its key parameters and provide a compact cancer reference network that increases the significance of tumor stratification using the NBS algorithm. The structure of the code exposes key steps of the algorithm to foster further collaborative development.

Availability and Implementation: The package, along with examples and data, can be downloaded and installed from the URL http://www.github.com/huangger/pyNBS/.

Contact: jkh013@ucsd.edu

1. Introduction

The biomedical community increasingly relies on genomic information to diagnose and treat many different complex diseases, including cancer (Frampton 2013; Johnson 2014). In parallel, developments in molecular interaction mapping technologies and network analysis algorithms have enabled the systematic elucidation of pathways involved in cancer and other complex diseases (Schaefer 2008). These two technologies - genomics and network analysis - have been recently combined to contextualize somatic mutations in tumors against the knowledge contained in molecular interaction networks and disease pathway maps. For example, numerous algorithms now use molecular network information to discover significantly mutated pathways in particular cohorts of patients (Vaske 2010; Ciriello 2012; Vandin 2011; Vandin 2011; Leiserson 2013; Paull 2013; Leiserson 2014; Drake 2016).

Recently, we introduced an algorithm that uses molecular network information to guide the stratification of tumor somatic mutation profiles into clinically relevant subtypes (Hofree 2013). Such mutation profiles have been notoriously difficult to stratify (i.e. cluster) due to their extreme heterogeneity from patient to patient. Our algorithm, called Network-Based Stratification (NBS), relies upon aggregating these mutations in molecular network neighborhoods to gain power in separating patients. The underlying assumption is that cancer arises due to disruptions in specific molecular pathways, not only disruptions in isolated genes. It is commonly observed that similar cancer types arise from mutations that affect different genes that are participants in common pathways. However, traditional gene-wise clustering methods fail to capture similarities that are observed only on the pathway level since mutations do not necessarily fall on the same genes and therefore do not contribute to any measure of similarity between patients despite affecting the same pathway. The information of each somatic mutation is smoothed across its network neighborhood, spreading the signal to other functionally related genes in network space. It is then possible to obtain robust clusters of patients based on the similarity of these network-smoothed mutation profiles.

In the original publication of NBS, the code used to develop the project was provided in Matlab, a proprietary programming language, making open access to this software difficult. Additionally, the code lacked modularization, making individual steps of the algorithm difficult to control, analyze and test. In what follows, we implement and organize the NBS algorithm as an installable Python package, which we call pyNBS. This package modularizes and exposes the major steps in the algorithm to better control, analyze, and improve the approach in future studies.

2. Methods

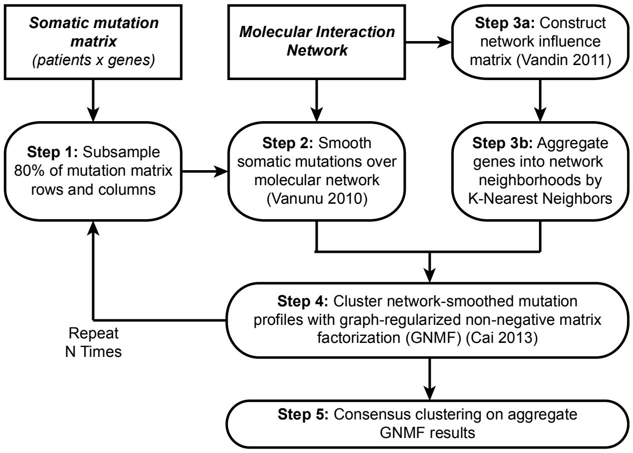
Somatic mutation matrix
(patients x genes)
Molecular Interaction Network
Step 3a: Construct network influence matrix (Vandin 2011)
Step 1: Subsample 80% of mutation matrix rows and columns
Step 2: Smooth somatic mutations over molecular network (Vanunu 2010)
Step 3b: Aggregate genes into network neighborhoods by K-Nearest Neighbors
Repeat
N Times
Step 4: Cluster network-smoothed mutation profiles with graph-regularized non-negative matrix factorization (GNMF) (Cai 2013)
Step 5: Consensus clustering on aggregate GNMF results

Figure 1. Overview and stepwise factorization of the NBS algorithm.

The NBS algorithm requires two inputs: a matrix of binary values describing all somatic tumor mutations found within a cohort of cancer patients (patients x genes) and a second file describing the gene-gene interactions defining a reference molecular network. Given these inputs, the NBS algorithm clusters the tumor mutation profiles into molecular subtypes as seen in **Figure 1**. Additional details of the algorithm are described in the original NBS manuscript (Hofree 2013).

3. Results

3.1 pyNBS Usage and Validation

The NBS algorithm can be executed using the pyNBS package in two modes: using a wrapper script via the command line, or by running the provided Jupyter Notebooks. Documentation for both code execution modes are provided within a GitHub repository, which can be found at: http://www.github.com/huangger/pyNBS/.

It should be noted that each full run of pyNBS does not necessarily produce the exact same cluster assignments on the same cohort. This variation is due to the stochastic nature of the sub-sampling step as well as the non-unique nature of matrix factorization (Cai 2013). However, this variance is largely controlled by the final consensus clustering step.

We tested the pyNBS package by generating patient subtypes in ovarian and uterine cancer using the data and corresponding networks released with the original Hofree et al. manuscript. PyNBS recovered the original Hofree patient cluster assignments for ovarian and uterine cancer (X^2 p-value: 2.3×10^{-107} and 5.3×10^{-88}, respectively). These two test examples are provided, along with the required datasets (re-formatted for usage with pyNBS), as Jupyter Notebooks in the GitHub repository.

3.2 A Cancer-Specific Network for pyNBS

In addition to reconstructing the original NBS algorithm, we also explored alternative reference networks for their ability to separate tumor cohorts into clinically relevant subtypes. The outcome of this exploratory research was a compact cancer reference network that contained only high-confidence interactions specific to cancer. To construct this network, we began with a high-quality network assembled in a previous study containing 19,781 genes with each of its 2,724,724 protein interactions supported by multiple lines of evidence (Huang and Carlin in press). We filtered this network to retain only cancer genes as documented in at least one of four collections (Hanahan 2011; Vogelstein 2013; Iorio 2016; Forbes 2017). We found that this cancer reference network more effectively clusters tumor samples from several different cancer types, as measured by the clusters' ability to predict patient survival, in comparison to the HumanNet network used in the original NBS study (**Figure 2A**). This cancer reference network, as well as directions on constructing this network and analysis on the effect of different network models on pyNBS are presented as supplemental Jupyter Notebooks located in the Github repository.

3.3 Practical Benchmarking and Parameter Tuning

The pyNBS algorithm can be expensive in both memory and in run time for large networks, or if many iterations of the sub-sampling and matrix factorization are required. However, we found that 1,000 iterations of sub-sampling and consensus clustering, as originally performed by Hofree et al., could be markedly decreased with little reduction in performance, with only 100 iterations being sufficient for the consensus clustering to converge. This reduction can offer about 90% run time savings with no appreciable deviation in the results (**Figure 2B**). For example, to stratify the TCGA head and neck cancer data using the filtered HumanNet (HN90, as described by Hofree et al.), we reduced the runtime of pyNBS from approximately 21.5 hours to 2.2 hours.

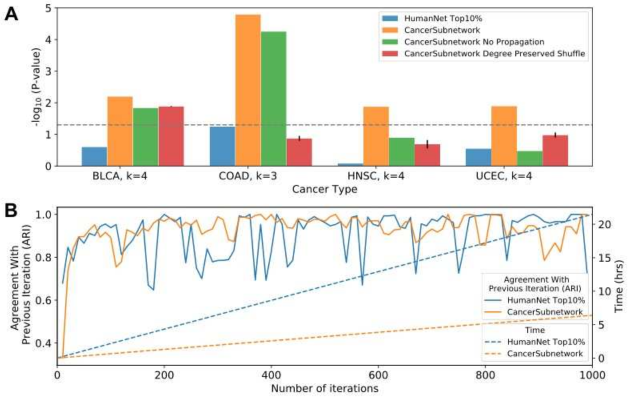

Figure 2. Benchmarking and pyNBS stratification performance.
A. Significance of survival separation between subtypes in bladder (BLCA), Colon (COAD), head and neck (HNSC) and uterine (UCEC) cancer as discovered by pyNBS. Cohorts were stratified using the top 10% of edges in HumanNet (HN90, blue) (Hofree et al. 2013), our cancer subnetwork from high-confidence network interactions (gold) (Huang and Carlin in press), without network propagation (green), and with propagation over a randomized cancer subnetwork (red).
B. Consensus clustering convergence rate and runtime performance of pyNBS on TCGA head and neck cancer data with HN90 (blue) and the Cancer Subnetwork (gold). By measuring the agreement of consensus clustering results at each step and the consensus clustering result using 10 less sub-sampling iterations, it is clear that the consensus clustering is fairly stable at just 100 sub-sampling iterations.

In addition, using the filtered Cancer Subnetwork (see above), which only has 2,291 nodes compared to the 7,939 nodes in HN90, we see that pyNBS not only runs much faster, but by reducing the consensus clustering iterations, this also reduces the overall runtime of pyNBS in this scenario from 6.5 hours to approximately 40 minutes (**Figure 2B**). Due to the NBS algorithm requiring many matrix multiplications, we recommend running pyNBS on a machine with at least four threads and 4GB of RAM per thread.

While we only sought to recreate the original procedure and parameter space for running pyNBS here, we performed an additional exploration on the effect of varying several parameters and algorithmic decisions on the final consensus clustering results in pyNBS. We present some of these results in a supplemental Jupyter Notebook in the Github repository.

Acknowledgements

We would like to thank Matan Hofree for his continued support and consultation regarding the NBS algorithm. We would also like to thank The Cancer Genome Atlas for providing the somatic mutation data used in the original NBS manuscript as well as in code examples.

Funding

This project was supported by grants from the National Institutes of Health (U24 CA184427, U54CA209891, T32 GM008806) and DARPA (W911-NF-14-1-0397).

Conflict of Interest: Trey Ideker is co-founder of Data4Cure, Inc. and has an equity interest. Trey Ideker has an equity interest in Ideaya BioSciences, Inc. The terms of this arrangement have been reviewed and approved by the University of California, San Diego in accordance with its conflict of interest policies.

References

Cai,D., *et al.* (2011) Graph Regularized Nonnegative Matrix Factorization for Data Representation. *IEEE Trans Pattern Anal Mach Intell.* 33(8), 1548-1560.

Ciriello,G., *et al.* (2012) Mutual exclusivity analysis identifies oncogenic network modules. *Genome Res*. 22(2), 398-406.

Drake,J.M., *et al*. (2016) Phosphoproteome Integration Reveals Patient-Specific Networks in Prostate Cancer. *Cell*. 166(4), 1041-1054.

Forbes,S.A., *et al*. (2017) COSMIC: somatic cancer genetics at high-resolution. *Nucleic Acids Res*. 45(D1), D777-D783.

Frampton,G.M., *et al*. (2013) Development and validation of a clinical cancer genomic profiling test based on massively parallel DNA sequencing. *Nat Biotechnol.*, 31(11), 1023-1031.

Hanahan,D. and Weinberg,R.A. (2011) Hallmarks of Cancer: The Next Generation. *Cell*. 144(5), 646-674.

Hofree,M. *et al*. (2013) Network-based stratification of tumor mutations. *Nat Methods.*, 10(11), 1108-1115.

Huang,J.K., Carlin,D.E., et al. (in press) Systematic evaluation of gene networks for discovery of disease genes.

Iorio,F., *et al*. (2016) A Landscape of Pharmacogenomic Interactions in Cancer. *Cell*. 166(3), 740-754.

Johnson,D.B., *et al*. (2014) Enabling a genetically informed approach to cancer medicine: a retrospective evaluation of the impact of comprehensive tumor profiling using a targeted next-generation sequencing panel. *Oncologist.*, 19(6), 616-622.

Leiserson,M.D.M. *et al*. (2014) Pan-Cancer Network Analysis Identifies Combinations of Rare Somatic Mutations across Pathways and Protein Complexes. *Nat Genet.*, 47(2), 106-114.

Leiserson,M.D.M. *et al*. (2013) Simultaneous Identification of Multiple Driver Pathways in Cancer. *PLoS Comput Biol.*, 9(5), e1003054.

Paull,E.O., *et al.* (2013) Discovering causal pathways linking genomic events to transcriptional states using Tied Diffusion Through Interacting Events (TieDIE). *Bioinformatics*. 29(21), 2757-2764.

Schaefer,C.F., *et al*. (2009) PID: Pathway Interaction Database. *Nucleic Acids Res.*, 37(Database issue), D674-D679.

Vandin,F. *et al*. (2011) Algorithms for detecting significantly mutated pathways in cancer. *J. Comput. Biol.*, 18(3), 507-522.

Vandin,F. *et al*. (2011) De novo discovery of mutated driver pathways in cancer. *Genome Res..*, 22(2), 375-385.

Vanunu,O., *et al*. (2010) Associating Genes and Protein Complexes with Disease via Network Propagation. *PLoS Comput Biol.*, 6(1), e1000641.

Vaske,C.J., *et al*. (2010) Inference of patient-specific pathway activities from multi-dimensional cancer genomics data using PARADIGM. *Bioinformatics*. 26(12), 234-245.

Vogelstein,B., *et al*. (2013) Cancer genome landscapes. *Science*. 339(6127), 1546-1558.