Bioinformatics, 2016, 0–0
doi: 10.1093/bioinformatics/xxxxx
Advance Access Publication Date: DD Month 2016
Applications Note

Systems biology

sfinx: an R package for the elimination of false positives from affinity purification – mass spectrometry datasets

Kevin Titeca[1,2], Pieter Meysman[3,4], Kris Laukens[3,4], Lennart Martens[1,2,5], Jan Tavernier[1,2] and Sven Eyckerman[1,2,*]

[1] VIB Medical Biotechnology Center, A. Baertsoenkaai 3, B-9000 Ghent, Belgium, [2] Department of Biochemistry, Ghent University, B-9000 Ghent, Belgium, [3] Advanced Database Research and Modelling (ADReM), Department of Mathematics and Computer Science, University of Antwerp, Antwerp, Belgium, [4] Biomedical Informatics Research Center Antwerp (biomina), University of Antwerp/Antwerp University Hospital, Edegem, Belgium, [5] Bioinformatics Institute Ghent, Ghent University, B-9052 Zwijnaarde, Belgium.

*To whom correspondence should be addressed.

Associate Editor: Prof. Alfonso Valencia

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Summary: We describe sfinx, an R package providing access to the straightforward filtering index (SFINX) for the separation of true positive from false positive protein interactions in affinity purification – mass spectrometry datasets. This package maintains the reliability and user-friendliness of the SFINX web site interface but is faster, unlimited in input size, and can be run locally within R.

Availability and implementation: The sfinx R package is available for download at the comprehensive R archive network (CRAN) https://cran.r-project.org/web/packages/sfinx/ under the Apache License 2.0.

Contact: sven.eyckerman@vib-ugent.be or kevin.titeca@gmail.com

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

1 Introduction

All functions in the cell result from the complex and dynamic collaboration of biomolecules, with a central role for the interactions between proteins. The analysis of protein-protein interactions (PPIs) connects genotypes to phenotypes, and it answers questions about biology and disease (Kuzmanov and Emili, 2013).

Affinity purification – mass spectrometry (AP-MS) is one of the most used technologies to perform large scale PPI analysis. In AP-MS, a protein of interest is genetically fused to an epitope tag and expressed in the cells of interest, followed by lysis of these cells and purification of the protein based on the epitope tag, ultimately enabling the detection of copurifying proteins by mass spectrometry. The protein of interest and the copurifying protein are here respectively called bait and prey.

Although AP-MS yields rich biologically relevant data, the resulting datasets typically contain large amounts of false positives, a fact which is exacerbated with increasing sensitivity of contemporary MS instruments. The false positives result from unspecific binding, potentially caused by sticking to the affinity purification matrix or affinity tag, by high abundance, or by binding to unfolded polypeptides (Gingras, et al., 2007). Furthermore, most experiments include only a few baits, replicates and negative controls, which makes the elimination of false positives even more difficult.

The ideal software approach to eliminate these false positives is fast, user-friendly, highly accurate, and independent of external data and extensive parameter optimization, making it more objective and reproducible. Different approaches already exist (Choi, et al., 2011; Meysman, et al., 2015; Sardiu, et al., 2008; Sowa, et al., 2009), but none of them fit the previous description completely. To address this need, we developed the straightforward filtering index (SFINX) (Titeca, et al., 2016).

The original version of SFINX is aimed at end-users and is accessible as an online tool at http://sfinx.ugent.be. This web site provides the research community with a user-friendly interface to SFINX but comes with some limitations. The speed of analysis is largely dependent on the speed of connection and server load. Therefore, input data set size is capped at 20 MB to avoid server overloading. Because more advanced users might

© The Author (2017). Published by Oxford University Press. All rights reserved. For Permissions, please email: journals.permissions@oup.com

instead wish to integrate SFINX in their local, automated data processing pipelines, we have now made SFINX available as an R package called sfinx. The R package allows local use of the SFINX algorithm at considerably increased speed and without limitations, although it should be noted that the performance of sfinx and the input file size of course remains limited by the specifications of the user's available computational resources.

2 Description

The sfinx R package is built on the same core algorithm as the online tool, and yields identical results, as demonstrated in Supplementary data 1 by the parallel analysis of the included example file. The description of the SFINX algorithm and its performance evaluation has been detailed before (Titeca, et al., 2016). Benchmarking on several different AP-MS datasets demonstrated up to 19.4% improved accuracy of the algorithm over alternative approaches. Similar to the online tool, the R package eliminates false positive interactions and then ranks the retained true positives by their individual certainties. It only retains interactions that are sufficiently exceptional according to a binomial distribution combined with an automatically determined cutoff that also corrects for multiple testing.

The sfinx package, which can be installed directly from the comprehensive R archive network (CRAN), provides straightforward one-step access to the SFINX algorithm through the *sfinx(InputData, BaitVector)* function. InputData is a numerical matrix containing the experimental data. The column names correspond to unique projects, and the row names to unique proteins found at least once in one of the projects. The matrix itself contains only peptide counts (or zeros in the absence of peptide counts). BaitVector is a character vector containing the bait proteins of interest to the user. Note that the algorithm is case sensitive, and that entries in BaitVector should exactly match row names of the InputData matrix. Example files for the InputData and BaitVector variables are included in the package as DataInputExampleFile and BaitIdentityExampleFile, respectively (Sardiu, et al., 2008).

The *sfinx* function outputs a list of two elements. The first element contains retained interactions ranked according to certainty scores. The second element is a string that contains basic information about the performed filtering, warnings about potential issues, and often suggested enhancements. Likewise, the function *sfinx(DataInputExampleFile, BaitIdentityFile)* yields the filtered interactions of Supplementary data 1 and the message of Supplementary data 2. The most typical warning messages generally result from the absence of the bait proteins in the dataset and from insufficient amounts of negative controls or data.

The sfinx function permits detailed tuning via the input of extra parameters, although alteration of these parameters is discouraged and users should always report any changes upon publication. The five tunable parameters are BackgroundRatio, BackgroundIdentity, BaitInfluence, ConstantLimit, and FWERType. The first three parameters influence the selection of surplus negative controls, while the last two parameters influence parts of the applied cut-off. In standard settings, the SFINX algorithm only considers five times more projects that the amount of bait-specific projects, and it preferably selects non-bait projects with the most peptide counts as negative controls. The BackgroundRatio parameter allows to control the ratio of the total amount of considered projects over the amount of bait projects. It standardly equals five but can be changed to any natural number higher than one. The BackgroundIdentity parameter allows the user to define the background projects of preference, instead of the automatically determined ones. However, for this option, column headers are obligatory for the input data matrix, and the package will switch back to automatic project selection upon inconsistencies. The BaitInfluence parameter is a logical that directs the algorithm to only use negative control projects that are associated with none of the baits, if possible. The ConstantLimit parameter is also a logical, but it influences the internal cut-off rather than the choice of negative control projects. When false, the binomial equivalent of the cut-off is calculated for every potential interaction between proteins, instead of using an approximate constant, which makes the analysis stricter but also potentially slower. The FWERType parameter accepts the following strings as input "B", "HolmB" and "Sidak", corresponding to respectively Bonferroni, Holm-Bonferroni and Šidák corrections to control the family wise error rate (FWER).

Although the sfinx package yields identical results to the online version of SFINX on input datasets that are smaller than 20 MB (Supplementary data 1), it is not limited in the size of the input dataset unlike the online version. The ability to analyze any size of dataset creates opportunities, because PPI technologies are increasingly capable of generating large interactome datasets. The dataset that resulted from the work of Hein et al. is a good example of such a large interactome dataset (Hein, et al., 2015), with a matrix of 3990 projects and 11152 proteins in a file of more than 88 MB. The analysis of this dataset is impossible with the online version of SFINX, but the package is able to process two baits in about 14s.

On datasets that are small enough for the analysis by the online version of SFINX, the package also functions considerably faster than the online version. To compare the speed of the package to the online version on data of different dimensions, we extended the included example data by repetitive concatenation. Upon the input of more projects, we observed a linear increase in the analysis time, with the package on average only requiring 57.8% of the analysis time of the online SFINX (Supplementary figure 1). Upon the input of more unique proteins, both versions can analyze the data very fast with less than 1.7s and 3.2s needed by respectively the package and the website interface to analyze 26 baits in a dataset of 22134 proteins. Because this is about the largest amount of unique proteins that will conceivably be present in any dataset, the speed increase for this kind of extension is probably less relevant. However, because users do not have to define the bait that was used in each of the experiments, the SFINX algorithm can also analyze proteins that were not the original baits, and it thus allows to employ virtually all proteins in the dataset as baits. Such a strategy can reveal associations between proteins that were not in the original study design. Upon the analysis of all proteins in datasets with rising amounts of proteins, the needed time increases at a faster than linear rate, but the package still only requires 82.8% of the analysis time of the online version of SFINX (Supplementary figure 2). Moreover, the sfinx package is able to analyze the previously discussed large dataset of Hein et al. with each of the 11152 proteins as a bait in 17 hours and 38 minutes.

Note that the sfinx package can also conceivably analyze output from other PPI technologies. The main candidates are technologies that involve tagging of only maximally one protein, like immunoprecipitation –

mass spectrometry (IP-MS) (Malovannaya, et al., 2011), tandem affinity purification – mass spectrometry (TAP-MS) (Rigaut, et al., 1999), BioID (Roux, et al., 2012) and Virotrap (Eyckerman, et al., 2016). However, measurement of the bait is essential for the algorithm, and proteins with very few detectable peptides are more difficult to analyze. Furthermore, the automatic bait detection contraindicates the comparison between different treatments with the same bait proteins or between very similar bait proteins such as isoforms and mutants.

The sfinx R package does not provide its own visualization interface, since network visualization and downstream analysis is achievable through other R packages such as ggplot2 (Wickham, 2009) and igraph (Csardi and Nepusz, 2006), or dedicated network visualization software like Cytoscape (Meysman, et al., 2015; Shannon, et al., 2003).

3 Conclusion

The sfinx R package enables local use of the SFINX algorithm in R pipelines, allowing considerable increases in speed and input capacity, and providing maximal data processing flexibility.

Acknowledgements

We thank K. Gevaert, S.J. Wodak, D. Maddelein and D. Ratman for discussions, and M. Washburn for providing additional information about the example datasets.

Funding

This work was supported by the ERC Advanced Grant CYRE 340941 to J.T., and K.T. is a PhD student with the Agency for Innovation by Science and Technology (IWT). L.M. acknowledges support from Ghent University Concerted Research Action grant BOF12/GOA/014, and VLAIO SBO grant 'INSPECTOR' (120025)

Conflict of Interest: none declared.

References

Choi, H., *et al.* (2011) SAINT: probabilistic scoring of affinity purification-mass spectrometry data, *Nat Methods*, **8**, 70-73.

Csardi, G. and Nepusz, T. (2006) *The igraph software package for complex network research.* InterJournal.

Eyckerman, S., *et al.* (2016) Trapping mammalian protein complexes in viral particles, *Nat Commun*, **7**, 11416.

Gingras, A.C., *et al.* (2007) Analysis of protein complexes using mass spectrometry, *Nat Rev Mol Cell Biol*, **8**, 645-654.

Hein, M.Y., *et al.* (2015) A human interactome in three quantitative dimensions organized by stoichiometries and abundances, *Cell*, **163**, 712-723.

Kuzmanov, U. and Emili, A. (2013) Protein-protein interaction networks: probing disease mechanisms using model systems, *Genome Med*, **5**, 37.

Malovannaya, A., *et al.* (2011) Analysis of the human endogenous coregulator complexome, *Cell*, **145**, 787-799.

Meysman, P., *et al.* (2015) Protein complex analysis: From raw protein lists to protein interaction networks, *Mass Spectrom Rev doi:10.1002/mas.21485*.

Rigaut, G., *et al.* (1999) A generic protein purification method for protein complex characterization and proteome exploration, *Nat Biotechnol*, **17**, 1030-1032.

Roux, K.J., *et al.* (2012) A promiscuous biotin ligase fusion protein identifies proximal and interacting proteins in mammalian cells, *J Cell Biol*, **196**, 801-810.

Sardiu, M.E., *et al.* (2008) Probabilistic assembly of human protein interaction networks from label-free quantitative proteomics, *Proc Natl Acad Sci U S A*, **105**, 1454-1459.

Shannon, P., *et al.* (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks, *Genome Res*, **13**, 2498-2504.

Sowa, M.E., *et al.* (2009) Defining the human deubiquitinating enzyme interaction landscape, *Cell*, **138**, 389-403.

Titeca, K., *et al.* (2016) SFINX: Straightforward Filtering Index for Affinity Purification-Mass Spectrometry Data Analysis, *J Proteome Res*, **15**, 332-338.

Wickham, H. (2009) *ggplot2: Elegant Graphics for Data Analysis.* Springer-Verlag New York.