WebPropagate: A Web Server for Network Propagation

Hadas Biran[1], **Tovi Almozlino**[1], **Martin Kupiec**[2] **and Roded Sharan**[1]

*1 - **Blavatnik School of Computer Science,** Tel Aviv University, Tel Aviv 69978, Israel*
*2 - **Department of Molecular Microbiology and Biotechnology,** Tel Aviv University, Tel Aviv 69978, Israel*

Correspondence to Roded Sharan: *roded@post.tau.ac.il*
https://doi.org/10.1016/j.jmb.2018.02.025
Edited by Michael Sternberg

Abstract

Network propagation is a powerful tool for genetic analysis which is widely used to identify genes and genetic modules that underlie a process of interest. Here we provide a graphical, web-based platform (http://anat.cs.tau.ac.il/WebPropagate/) in which researchers can easily apply variants of this method to data sets of interest using up-to-date networks of protein–protein interactions in several organisms.

Introduction

Proteins do not work in isolation; rather, they interact to drive cellular processes. Network propagation is a powerful approach for exploiting the proximity of proteins in a network to infer their functional roles [1]. Network propagation has diverse applications in the biomedical domain, including gene prioritization [2–5], identification of protein complexes [3], detection of disease-related subnetworks [6–8], clustering cancer patients to disease subtypes [9–11], and identification of potential drug targets in personalized medicine [12,13].

In its basic form, network propagation is used to enrich limited prior knowledge about the set of proteins involved in a process of interest by identifying novel proteins that are proximal in the network to the prior set. However, the scores given by the network propagation process cannot be readily assigned statistical significance levels. This is because they are highly correlated with the size of the prior set [14] and are also biased by the degrees of proteins in the network [15].

Network propagation has been previously implemented by two Cytoscape Apps, Propagate and Diffusion. These apps implement two basic variants of propagation, both producing a list of all the network proteins, ranked based on their relative association with the prior set [3,16]. None of these previous methods allow assigning *p*-values to the resulting candidate proteins.

Here we report on a new web server, WebPropagate, for network propagation that allows the user to pre-set a desired false discovery rate (FDR) threshold, avoiding the need to arbitrarily determine a significance threshold for every propagation run. Furthermore, the server lets the user control the propagation parameters, choose a protein–protein interaction (PPI) network to operate on, apply single or integrative variants, and obtain a graphical visualization of the results.

Results

WebPropagate facilitates easy and fast PPI-based gene prioritization. It applies the well-known network propagation method, normalizes its results to deal with biases that stem from the size of the input set and network hubs, and generates a list of proteins that are significantly close in the network to the user defined prior or seed set. Its usage, example applications and comparison to previous tools are described below.

How to use WebPropagate?

WebPropagate has five input fields: (i) *Species*, in which the user selects the PPI network that is suitable for his/her data. (ii) *Integrative Propagation* checkbox, which should be blank if the user wishes to propagate from a single seed set, and checked if the user wishes to propagate simultaneously from two different seed

sets. In the latter case, the output will be the intersection of the sets of significant proteins from both computations. (iii) *Propagation alpha*, which weighs the relative importance of the network vs. the seed set in the propagation (ranges between 0.5 and 0.9). A larger value gives more weight to the network smoothness. (iv) *FDR threshold*, which determines the value under which the corrected *p*-values are classified as significant (ranges between 0.01 and 0.5). And (v) *Seeds*, which are the list of proteins that are chosen based on prior knowledge to serve as seeds for the propagation computations. The seeds can be entered by their gene identifiers (entrez ids), symbols or locus tags. They can also be imported through an Excel file rather than be entered manually.

WebPropagate outputs a *Seeds* table, in which only the input seeds that were recognized as proteins in the chosen network are listed; *Significant Proteins* table, in which the discovered significant proteins are listed, with their initial and FDR-corrected *p*-values; and a sub-network display in which the seeds and the significant proteins are presented in different colors (seeds in green, significant proteins in pink). Direct neighbors of a protein can be added to the view (in cyan) by clicking on its node. Clicking anywhere in the view afterward will make them disappear. The sub-network view assists in interpreting the molecular mechanisms that underlie the process being analyzed. Both output tables can be exported to Excel.

Case study

We demonstrate the utility of WebPropagate by analyzing a data set of proteins that regulate telomere length in yeast. We use 10 telomere-binding proteins [17] as a seed set (Fig. 1). WebPropagate outputs 10 significant proteins as potentially involved in telomere length maintenance (TLM). Remarkably, most of them are indeed linked to telomeres and telomerase activity: four proteins (RFA1, POL12, SIR4, PXR1) are known to be TLM genes [17–20]. In addition, ZDS2 interacts with telomeric "core" proteins and affects silencing of genes at telomeres [21,22], FUN30 (SMARCAD1 in humans) is a chromatin remodeller that has a role in DNA processing and affects silencing at telomeres [23,24], and MRX1, although annotated as a mitochondrial protein, has strong negative genetic interactions with cdc13–1, a mutant defective in telomere maintenance, as well as with tel2, another important TLM gene [25]. Of the three remaining proteins, two have only scarce annotation information (YKR051w and YML020w) and thus remain as "uncharacterized open reading frames." With respect to the merged list of TLM-related proteins from Refs. [17–27] (339

Table 1. Telomere-binding proteins

Name	Entrez ID	Symbol
EST1	850934	EST1
EST2	851028	EST2
EST3	854806	EST3
YKU70	855328	YKU70
YKU80	855132	YKU80
STN1	851655	STN1
TEN1	850696	TEN1
CDC13	851306	CDC13
EXO1	854198	EXO1
RAP1	855505	RAP1

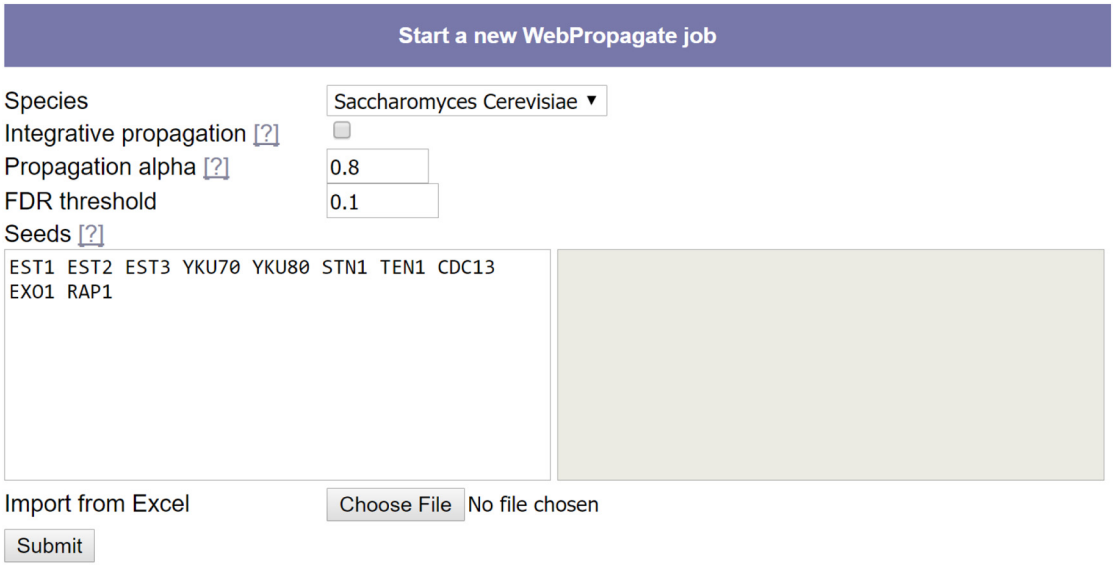

Fig. 1. Input parameters for the telomere length maintenance case study in yeast.

Table 2. Output of WebPropagate on TLM proteins

Significant proteins			
Entrez ID	Symbol	*p*-Value	FDR Corrected
851813	SIR4	0.00001	0.00791
852245	POL12	0.00001	0.00923
853925	YKR051W	0.00001	0.01108
856810	MRX1	0.00001	0.01384
854988	YML020W	0.00001	0.01846
854836	FLO11	0.00001	0.02769
854931	ZDS2	0.00004	0.02769
853197	PXR1	0.00001	0.05538
851266	RFA1	0.00014	0.07753
851214	FUN30	0.00013	0.07998

proteins in total that appear in our *Saccharomyces cerevisiae* network), WebPropagate's output yields a notable hypergeometric *p*-value of 3.1e−7. Table 1 displays the seed list, Table 2 displays the list of significant proteins reported by WebPropagate, and Fig. 2 displays the subnetwork induced by these proteins.

Comparison with the Propagate Cytoscape App

To show the advantage of WebPropagate's normalization method over the basic propagation scheme, as implemented in Propagate, we used a data set of gene-disease associations [28]. The data set was curated by classifying diseases and their linked genes from Online Mendelian Inheritance in Man and from genome-wide association study to 299 diseases defined by the Medical Subject Headings ontology. Here we use the more reliable Online Mendelian Inheritance in Man part of the data set and focus on diseases that are associated with at least 20 genes which appear in our *Homo sapiens* network (171 diseases in total). For each disease, we evaluated the performance of both tools in predicting true associations in a fivefold cross-validation setting. Each iteration, the hidden associations served as positive samples and a set of genes of the same size and similar degrees served as the negative samples. To draw a protein with weighted degree similar to some given value w, we chose the smallest integer r such that there are at least 100 proteins in the network (excluding the prior set, the positive samples, and the already chosen negative samples) with weighted degree in the range $[w - r, w + r]$. We then randomly picked a protein from this group to be used as a negative association. We averaged the resulting AUCs over the five cross-validation iterations for each disease. For 150 out of the 171 diseases, the WebPropagate resulting AUC outperformed the Propagate AUC ($p < 1.1e-16$ based on a cumulative binomial distribution with parameter 0.5; see Fig. 3). When averaging over all the diseases, the average AUC for basic propagation was 0.67, while the average AUC for WebPropagate was 0.71.

An advantage of WebPropagate over the basic propagation scheme is that the output scores are normalized to account for the degree of a given node. To verify that this normalization corrects for potential degree biases in the scores and implied protein ranks, we computed the sample Pearson correlation coefficient between the ranks assigned to the network's proteins by each method and their weighted degrees. For WebPropagate, we observed a weak correlation of −0.009 across the 171 diseases. In contrast, the Propagate plugin yielded a strongly negative correlation of −0.544.

Methods

Networks

Our website applies the propagation on up-to-date protein–protein interaction networks from four organisms (*H. sapiens*, *Arabidopsis thaliana*, *Drosophila melanogaster*, and *S. cerevisiae*) that were derived from the BioGrid database as described in Ref. [29].

Basic propagation

The basic propagation procedure can be thought of as a diffusion process in which each of the seed proteins is a source of heat and this heat is diffused in a protein–protein interaction network. When the process converges the score $F(v)$of every protein v respects the equation.

$F(v) = \alpha[\sum_{u \in N(v)} F(u) w'(v,u)] + (1 - \alpha) Y(v)$, where $Y(v)$ denotes whether the protein is part of the seed set; $w'(v,u)$ is the weight of the interaction between u and v, normalized by the square root of the weighted degrees of both vertices, that is, $w'(v,u) = \frac{w(v,u)}{\sqrt{\sum_{z \in N(v)} w(v,z) \sum_{t \in N(u)} w(u,t)}}$; and α is a network smoothing parameter [3]. In matrix notation, it can be shown (see, e.g., Ref. [1]) that the score vector F is a linear transformation of the prior vector Y: $F = \alpha(I - (1 - \alpha)W)^{-1} Y$, an observation that we use below.

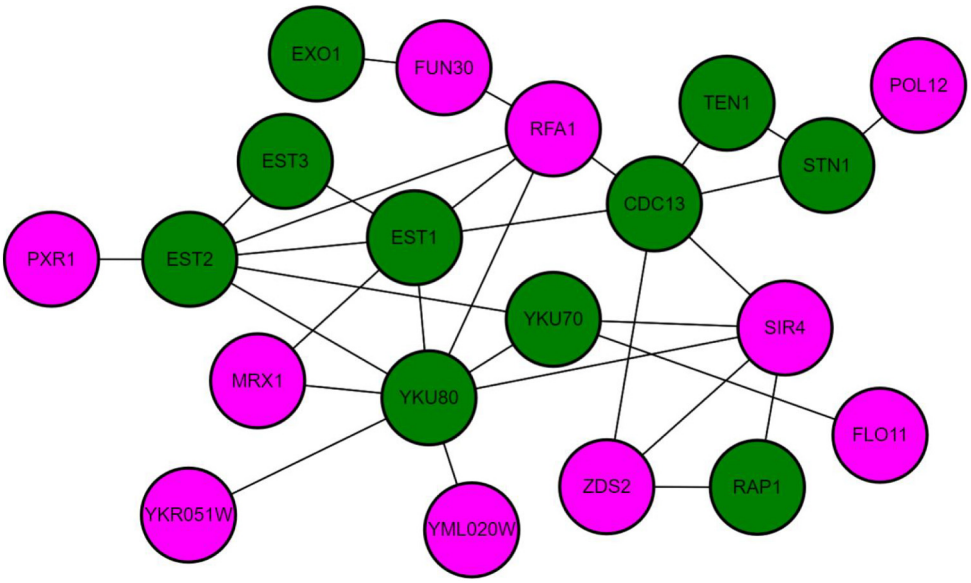

Fig. 2. The subnetwork output on the telomere length maintenance case study. The seeds are displayed in green; significant proteins, pink.

p-value computation and FDR correction

Given the results of network propagation on some seed set, we would like to assign *p*-values to the scores to control for the size of the prior set and the degree of each protein. To this end, we randomly generate 100,000 prior sets of the same size as the user-defined set and compute propagation scores for each. Thus, for every protein P we have a "real" score $X^{\mathrm{P}}_{\mathrm{real}}$ and 100,000 "random" scores X^{P}_i $(0 \le i \le 99{,}999)$ obtained by using random prior sets. This allows us to estimate the empirical *p*-value of P as the percent of its random scores that exceed the real score (excluding prior sets that contain P), that is:

$$p\_\mathrm{value} = \frac{\left|\left\{i \middle| \left(X^{\mathrm{P}}_i \ge X^{\mathrm{P}}_{\mathrm{real}} \text{ and P was not part of the } i\text{-th random seed set}\right)\right\}\right| + 1}{\left|\left\{i \middle| \mathrm{P} \text{ was not part of the } i\text{-th random seed set}\right\}\right| + 1}$$

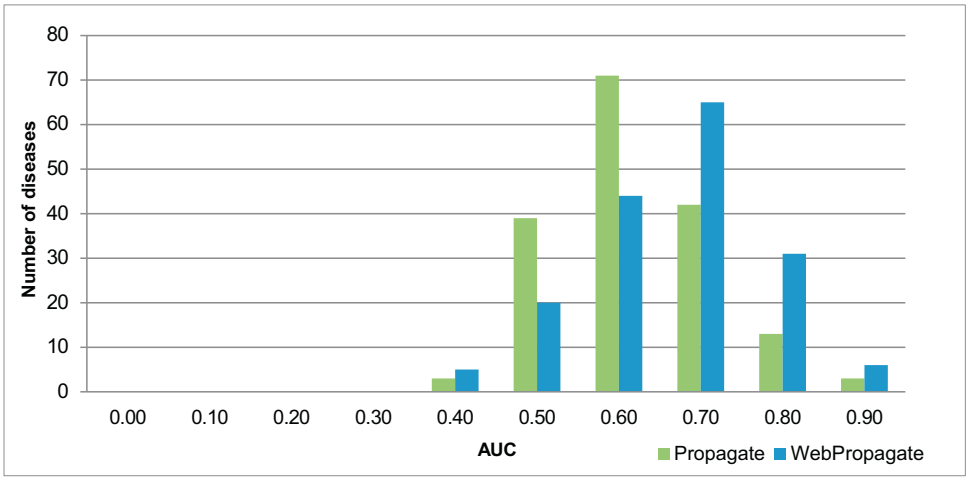

Fig. 3. Histograms of AUCs produced by WebPropagate and the Propagate plugin on a data set of 171 diseases in human.

Expectedly, for random seed sets, these *p*-values are uniformly distributed (data not shown). We further FDR correct these *p*-values for multiple testing by using the Benjamini–Hochberg procedure [30], namely, $p\_value_{BH} = \frac{p\_value \times M}{k}$, where *k* is P's position in a sorted ascending list of *p*-values of size *M* (which is equal to the number of network proteins excluding the prior set). As suggested by Yekutieli and Benjamini [31], we further adjust these initial *p*-values by assigning each protein in position *i* in the list with the smallest p_value_{BH} that appears at position $j \geq i$ in the list. We then retain those final *p*-values that are smaller than a user-defined threshold.

As the computation of 100,000 network propagations is very costly, we rely on the linearity of the propagation scores to speed it up. As a preliminary step, we propagate each of the network's proteins separately (a seed set of size 1) and save the propagation scores. Now for any seed set $S = \{g_i\}_{i=1}^{|S|}$ of size $|S|$, denote by $F(S)$, the vector of propagation scores of all network's proteins when propagating from *S*. By the linearity of the propagation scores (as discussed in the Basic propagation section), $F(S) = \sum_{i=1}^{|S|} F(\{g_i\})$. This allows us to compute the 100,000 "random" scores required for the *p*-value computation efficiently (less than a minute for seed sets of size up to 100).

Conflicts of Interest Statement

None.

Acknowledgments

RS was supported by a research grant from the I-CORE Program (grant 757/12).

Received 1 December 2017;
Received in revised form 26 February 2018;
Accepted 27 February 2018
Available online 7 March 2018

Keywords:
network diffusion;
protein–protein interaction network;
gene prioritization;
p-value computation;
subnetwork inference

Abbreviations used:
FDR, false discovery rate; PPI, protein–protein interaction; TLM, telomere length maintenance.

References

[1] L. Cowen, T. Ideker, B.J. Raphael, R. Sharan, Network propagation: a universal amplifier of genetic associations, Nat. Rev. Genet. 18 (2017) 551, https://doi.org/10.1038/nrg.2017.38.

[2] Y. Qian, S. Besenbacher, T. Mailund, M.H. Schierup, Identifying disease associated genes by network propagation, BMC Syst. Biol. 8 (Suppl. 1) (2014) S6, https://doi.org/10.1186/1752-0509-8-S1-S6.

[3] O. Vanunu, O. Magger, E. Ruppin, T. Shlomi, R. Sharan, Associating genes and protein complexes with disease via network propagation, PLoS Comput. Biol. 6 (2010), e1000641. https://doi.org/10.1371/journal.pcbi.1000641.

[4] I. Lee, U.M. Blom, P.I. Wang, J.E. Shim, E.M. Marcotte, Prioritizing candidate disease genes by network-based boosting of genome-wide association data, Genome Res. 21 (2011) 1109–1121, https://doi.org/10.1101/gr.118992.110.

[5] S. Köhler, S. Bauer, D. Horn, P.N. Robinson, Walking the interactome for prioritization of candidate disease genes, Am. J. Hum. Genet. 82 (2008) 949–958, https://doi.org/10.1016/j.ajhg.2008.02.013.

[6] F. Vandin, E. Upfal, B.J. Raphael, Algorithms for detecting significantly mutated pathways in cancer, J. Comput. Biol. J. Comput. Mol. Cell Biol. 18 (2011) 507–522, https://doi.org/10.1089/cmb.2010.0265.

[7] E.O. Paull, D.E. Carlin, M. Niepel, P.K. Sorger, D. Haussler, J.M. Stuart, Discovering causal pathways linking genomic events to transcriptional states using Tied Diffusion Through Interacting Events (TieDIE), Bioinformatics (Oxf. Engl.) 29 (2013) 2757–2764, https://doi.org/10.1093/bioinformatics/btt471.

[8] M.D.M. Leiserson, F. Vandin, H.-T. Wu, J.R. Dobson, J.V. Eldridge, J.L. Thomas, A. Papoutsaki, Y. Kim, B. Niu, M. McLellan, M.S. Lawrence, A. Gonzalez-Perez, D. Tamborero, Y. Cheng, G.A. Ryslik, N. Lopez-Bigas, G. Getz, L. Ding, B.J. Raphael, Pan-cancer network analysis identifies combinations of rare somatic mutations across pathways and protein complexes, Nat. Genet. 47 (2015) 106–114, https://doi.org/10.1038/ng.3168.

[9] M. Hofree, J.P. Shen, H. Carter, A. Gross, T. Ideker, Network-based stratification of tumor mutations, Nat. Methods 10 (2013) 1108–1115, https://doi.org/10.1038/nmeth.2651.

[10] Z. Liu, S. Zhang, Tumor characterization and stratification by integrated molecular profiles reveals essential pan-cancer features, BMC Genomics 16 (2015), https://bmcgenomics.biomedcentral.com/articles/10.1186/s12864-015-1687-x, Accessed date: 10 February 2018.

[11] X. Zhong, H. Yang, S. Zhao, Y. Shyr, B. Li, Network-based stratification analysis of 13 major cancer types using mutations in panels of cancer genes, BMC Genomics 16 (2015), https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4474538/, Accessed date: 10 February 2018.

[12] O. Shnaps, E. Perry, D. Silverbush, R. Sharan, Inference of personalized drug targets via network propagation, Pac. Symp. Biocomput. 21 (2016) 156–167.

[13] J.M. Drake, E.O. Paull, N.A. Graham, J.K. Lee, B.A. Smith, B. Titz, T. Stoyanova, C.M. Faltermeier, V. Uzunangelov, D.E. Carlin, D.T. Fleming, C.K. Wong, Y. Newton, S. Sudha, A.A. Vashisht, J. Huang, J.A. Wohlschlegel, T.G. Graeber, O.N. Witte, J.M. Stuart, Phosphoproteome integration reveals

patient-specific networks in prostate cancer, Cell 166 (2016) 1041–1054, https://doi.org/10.1016/j.cell.2016.07.007.
[14] A. Mazza, K. Klockmeier, E. Wanker, R. Sharan, An integer programming framework for inferring disease complexes from network data, Bioinformatics (Oxf. Engl.) 32 (2016) i271–i277, https://doi.org/10.1093/bioinformatics/btw263.
[15] S. Erten, G. Bebek, R.M. Ewing, M. Koyutürk, DADA: degree-aware algorithms for network-based disease gene prioritization, BioData Min. 4 (2011) 19, https://doi.org/10.1186/1756-0381-4-19.
[16] D.E. Carlin, B. Demchak, D. Pratt, E. Sage, T. Ideker, Network propagation in the cytoscape cyberinfrastructure, PLoS Comput. Biol. 13 (2017), e1005598. https://doi.org/10.1371/journal.pcbi.1005598.
[17] R. Shachar, L. Ungar, M. Kupiec, E. Ruppin, R. Sharan, A systems-level approach to mapping the telomere length maintenance gene circuitry, Mol. Syst. Biol. 4 (2008) 172, https://doi.org/10.1038/msb.2008.13.
[18] S. Grossi, A. Puglisi, P.V. Dmitriev, M. Lopes, D. Shore, Pol12, the B subunit of DNA polymerase alpha, functions in both telomere capping and length regulation, Genes Dev. 18 (2004) 992–1006, https://doi.org/10.1101/gad.300004.
[19] T. Gatbonton, M. Imbesi, M. Nelson, J.M. Akey, D.M. Ruderfer, L. Kruglyak, J.A. Simon, A. Bedalov, Telomere length as a quantitative trait: genome-wide survey and genetic mapping of telomere length-control genes in yeast, PLoS Genet. 2 (2006), e35. https://doi.org/10.1371/journal.pgen.0020035.
[20] J. Lin, E.H. Blackburn, Nucleolar protein PinX1p regulates telomerase by sequestering its protein catalytic subunit in an inactive complex lacking telomerase RNA, Genes Dev. 18 (2004) 387–396, https://doi.org/10.1101/gad.1171804.
[21] C.-L. Hsu, Y.-S. Chen, S.-Y. Tsai, P.-J. Tu, M.-J. Wang, J.-J. Lin, Interaction of *Saccharomyces* Cdc13p with Pol1p, Imp4p, Sir4p and Zds2p is involved in telomere replication, telomere maintenance and cell growth control, Nucleic Acids Res. 32 (2004) 511–521, https://doi.org/10.1093/nar/gkh203.
[22] N. Roy, K.W. Runge, The ZDS1 and ZDS2 proteins require the Sir3p component of yeast silent chromatin to enhance the stability of short linear centromeric plasmids, Chromosoma 108 (1999) 146–161.
[23] A. Neves-Costa, W.R. Will, A.T. Vetter, J.R. Miller, P. Varga-Weisz, The SNF2-family member Fun30 promotes gene silencing in heterochromatic loci, PLoS One 4 (2009), e8111. https://doi.org/10.1371/journal.pone.0008111.
[24] Q. Yu, X. Zhang, X. Bi, Roles of chromatin remodeling factors in the formation and maintenance of heterochromatin structure, J. Biol. Chem. 286 (2011) 14659–14669, https://doi.org/10.1074/jbc.M110.183269.
[25] TheCellMap—Tabular view, (n.d.). http://thecellmap.org/tabular/?n=4850 (accessed November 27, 2017).
[26] S.H. Askree, T. Yehuda, S. Smolikov, R. Gurevich, J. Hawk, C. Coker, A. Krauskopf, M. Kupiec, M.J. McEachern, A genome-wide screen for *Saccharomyces cerevisiae* deletion mutants that affect telomere length, Proc. Natl. Acad. Sci. U. S. A. 101 (2004) 8658–8663, https://doi.org/10.1073/pnas.0401263101.
[27] L. Ungar, N. Yosef, Y. Sela, R. Sharan, E. Ruppin, M. Kupiec, A genome-wide screen for essential yeast genes that affect telomere length maintenance, Nucleic Acids Res. 37 (2009) 3840–3849, https://doi.org/10.1093/nar/gkp259.
[28] J. Menche, A. Sharma, M. Kitsak, S.D. Ghiassian, M. Vidal, J. Loscalzo, A.-L. Barabási, Disease networks. Uncovering disease-disease relationships through the incomplete interactome, Science 347 (2015) 1257601, https://doi.org/10.1126/science.1257601.
[29] Y. Almozlino, N. Atias, D. Silverbush, R. Sharan, ANAT 2.0: reconstructing functional protein subnetworks, BMC Bioinf. 18 (2017), https://doi.org/10.1186/s12859-017-1932-1.
[30] Y. Benjamini, Y. Hochberg, Controlling the false discovery rate—a practical and powerful approach to multiple testing, J. R. Stat. Soc. Ser. B Methodol. 57 (1995) 289–300.
[31] D. Yekutieli, Y. Benjamini, Resampling-based false discovery rate controlling multiple test procedures for correlated test statistics, J. Stat. Plan. Infer. 82 (1999) 171–196, https://doi.org/10.1016/S0378-3758(99)00041-5.