On the performance of de novo pathway enrichment

Richa Batra 1,2,3, Nicolas Alcaraz 4, Kevin Gitzhofer 5, Josch Pauling 6, Henrik J. Ditzel 4,7, Marc Hellmuth 5,8, Jan Baumbach 5,9 and Markus List 10

De novo pathway enrichment is a powerful approach to discover previously uncharacterized molecular mechanisms in addition to already known pathways. To achieve this, condition-specific functional modules are extracted from large interaction networks. Here, we give an overview of the state of the art and present the first framework for assessing the performance of existing methods. We identified 19 tools and selected seven representative candidates for a comparative analysis with more than 12,000 runs, spanning different biological networks, molecular profiles, and parameters. Our results show that none of the methods consistently outperforms the others. To mitigate this issue for biomedical researchers, we provide guidelines to choose the appropriate tool for a given dataset. Moreover, our framework is the first attempt for a quantitative evaluation of de novo methods, which will allow the bioinformatics community to objectively compare future tools against the state of the art.

npj Systems Biology and Applications (2017)3:6 ; doi:10.1038/s41540-017-0007-2

INTRODUCTION

Modern high-throughput OMICS technologies (genomics, transcriptomics, proteomics, metabolomics, etc.) are driving the exponential growth of biological data. With continuously decreasing costs, it is now possible to determine the activity of genes, proteins and metabolites, their chemical modification (phosphorylation, methylation, etc.), as well as mutations on a genome scale. For example, the gene expression omnibus (GEO) alone hosts >5,000 study records derived from >1600 organisms. Moreover, many international consortia have gathered data from thousands of samples, for instance, the International Cancer Genome Consortium has collected data on 50 cancer types derived from several OMICS technologies. To gain a holistic view of such huge datasets, network-based analysis has become a promising alternative to traditional enrichment approaches such as gene ontology enrichment or gene set enrichment analysis.

The objective is to study condition-specific systemic changes by projecting experimental data on network modules, representing conditionally perturbed biological processes. Despite the inherent noise and incompleteness of currently available networks, network-based analysis has compelling advantages. It can capture the modular interplay of biological entities and processes overlooked in traditional enrichment methods and can further unravel molecular interactions not covered in well-defined pathways. In addition, network-based pathway enrichment methods can unravel crosstalk between pathways or sub-mechanisms which remain undetected in traditional pathway enrichment methods with known pathways such as KEGG or Reactome.

De novo pathway enrichment

Over the years, several approaches have been proposed for network-based analysis of experimental data. While some methods exploit network topology to augment scoring of known pathways, others rely on experimental data to reconstruct a condition-specific interaction network. Here, we focus on methods that perform de novo pathway enrichment. These methods integrate experimental data with a large-scale interaction network, to extract sub-networks enriched in biological entities active in a given experimental dataset. The definition of active depends on the type of dataset, for instance, differential expression in the case of gene expression data.

Diverse strategies for de novo pathway enrichment methods have been devised. To summarize the state of the art, we classify existing methods into four categories (Supplementary Fig. 1); (1) aggregate score optimization methods, (2) module cover approaches, (3) score propagation approaches, and (4) clustering-based approaches. Detailed descriptions of these classes are provided in the supplementary material. While several such methods exist (see Table 1), there are no guidelines to aid potential users in the choice of the appropriate tool. To fill this gap, we present a comparative analysis of selected tools across different datasets and parameter settings, as well as propose the first gold standard for quantitative benchmarking of de novo pathway enrichment methods (Fig. 1).

Challenges in evaluating de novo pathway enrichment tools

Pathway enrichment methods belong to the family of unsupervised statistical learning algorithms. In this context, unsupervised
Figure 2a and b illustrate that with increasing signal strength generally improves the performance of all tools. BioNet, GXNA, and KeyPathwayMiner (KPM) performed best with VM. However, if the simulation of expression profiles was based on changes in the variance (VV) between FG and BG distributions, the top performers BioNet and GXNA performed poorly. We speculate that this was due to their internal expression profile preprocessing procedures. Thus, one should carefully consider the characteristics of FG and BG expression profiles in the choice of the most suitable tool.

The effect of varying sparsity
We used two algorithms to distribute the label FG across the nodes in the network. The seed-and-extend (SAE) algorithm generates connected random sub-networks of a particular size and labels all its nodes as FG. In contrast, the average-distance-k (AVDk) algorithm distributes the FG node labels such that their average pairwise distance equals a given k. With increasing k values, the sparsity grows. All other nodes in the network were labeled as BG. We computed the sparsity of the FG nodes with three measures (a) Global FG proximity: Shortest path covering the FG set nodes in the network. (b) Global FG connectivity: Average degree of the FG nodes in the network. (c) Local FG density: Graph density of the FG nodes.

As expected and depicted in Fig. 2c and d, the performance of all tools decreased with increasing sparsity (decreasing FG proximity, i.e. increasing k using AVDk). Supplementary Fig. 2 further illustrates the performance dynamics with increasing FG density and connectivity. Evidently, all tools performed better with FG nodes selected using the SAE method (Supplementary Fig. 3). However, biological processes are complex and interaction networks are incomplete. Genes that are deregulated in a condition/disease may not be directly connected in the cellular network. Consequently, we believe that generally AVDk is a better, more natural choice for FG selection. The results for the I2D network are depicted in Supplementary Figs 4 and 5.

The effect of varying internal parameters
We can further categorize tools based on the characteristic of their internal parameters (IP). COSINE, GIGA, GXNA, and PinnacleZ restrict the solution to a user-specified size. In contrast, BioNet, DEGAS and KPM allow for noise (exceptions or outliers) in the reported solutions, such that sub-networks can have arbitrary size.
within certain constraints to allow for noise. We tested each selected method on a wide range of IP values. Overall, the trend was that with increasing IP values the size of the solution increases (Fig. 3). Lower IP values led to solutions with fewer nodes, whereas higher IP values led to larger sub-networks. As expected, the performance (F-measure) decreased towards the extreme ends of the IP range.

**Summary**

Consistently across the two tested networks (HPRD and I2D), BioNet, COSINE, GXNA, and PinnacleZ performed better with expression data generated using VM as compared to VV. DEGAS yielded better results with VV, while the performance of KPM and GiGA was independent of VV and VM. When using VM GiGA, BioNet, GXNA, and KPM showed better performance with SAE (dense FG nodes) compared to AVD (relatively sparse). COSINE, DEGAS, and PinnacleZ appeared less affected. These trends were conserved across the two networks. Using HPRD as a network base, KPM, DEGAS, and GiGA appeared less affected. These trends were conserved across the two networks. Using HPRD as a network base, KPM, DEGAS, and GiGA performed better with VV with both FG node selection algorithms. All tools performed poorly with VV and AVD when using I2D. However, when using SAE instead, KPM and GiGA performed very well.

Based on our analysis, we propose the following guidelines for biomedical researchers, (a) if computationally feasible, choose a comprehensive biological network such as I2D for capturing a wide range of known or predicted biological interactions (b) apply several tools to obtain a comprehensive picture of the information captured in the dataset. In any case, a wide range of IP should be tested. Please note that apart from gene expression profiles, some tools can be used for other types of OMICS data (molecular profiles) such as protein expression profiles or RNA sequencing data.

**DISCUSSION AND CONCLUSION**

De novo pathway enrichment is a powerful method to uncover condition-specific functional modules in molecular biological networks. Existing methods for integrating networks and experimental data have recently been reviewed in Mitra et al. Nevertheless, systematic and quantitative evaluations of the performance of existing methods and methodology to do so, have yet to be published.

To establish a basis for such a comparison in the absence of a suitable gold standard, we propose a strategy to assess the performance of de novo pathway enrichment tools on simulated data. We sampled nodes in the network (e.g. genes) that are relevant in a condition from a foreground distribution and non-relevant nodes from a background distribution. By adjusting the sparsity and signal strength of the solutions “hidden” in the network, we were able to adjust the difficulty of the de novo pathway enrichment problem and systematically compare the performance of selected tools.

We compared seven tools, representing diverse methodological approaches, to study the influence of data preprocessing and parameter choice on the quality of the solutions. In our analysis, BioNet and KPM often performed best, presumably because these tools leave the preprocessing (computation of $p$-values from...
expression data, for instance) to the user. Our results show that, in absence of a well-defined gold standard, using two or three different tools can provide a comprehensive picture of the information content of the data. In the future, we hope to extend our framework to a platform for meta or ensemble analysis using several tools in conjunction. Such a method could, for instance, combine results from different tools into concatenated (sub-)networks.

The framework presented here allows for the first standardized, well-structured and comprehensive evaluation of de novo pathway enrichment methods. However, the design of an optimal benchmark scheme relies on reasonable model assumptions and is a challenging task. Our conclusions are limited by nature, as we have no a priori gold standard datasets. While our knowledge of disease relevant genes continuously improves, we still lack sets of non-relevant genes in certain diseases. Nevertheless, we may assume that disease genes are generally closer to each other in a molecular biological network, and that their expression (in general) is differentially distributed compared to non-disease genes. Considering this, we believe to offer a reasonable test scenario for de novo pathway enrichment methods. However, we acknowledge that existing methods are diverse in methodology as well as in their expected input datatypes and formats. As their performance and ranking varies greatly with their IP and the characteristics of the expression datasets, all rankings should be carefully interpreted. While simulated data are suitable for objective comparisons, we acknowledge that we cannot fully model the complexity of biological systems. A promising way forward would be to use large-scale perturbation experiments, such as shRNA or CRISPR knockdown/knockout studies, in a benchmark.

METHODS
Candidate tool selection
For our analysis, we selected methods available as standalone applications, a prerequisite for a systematic and automated analysis across different datasets and parameters. We further restricted our comparison to tools designed for pathway enrichment using gene expression data, since it is the most common application type. The following seven tools fulfill both inclusion criteria: (a) BioNet is an aggregate score optimization method that implements an efficient integer linear programming approach to compute optimal sub-networks. (b) COSINE is a score propagation method that includes not only nodes but also edges in its scoring function. (c) DEGAS is a module cover approach that uses a shortest path heuristic to identify sub-networks with at least K “active” nodes, with “active” being defined via the proportion of active cases in the experimental data. (d) Giga is another score propagation method that...
Fig. 3  Average performance for over 80 FG sets of size 20 nodes generated using the AVDₜ algorithm. Expression profiles were simulated with VM. The HPRD network was used as the input network. Performance was assessed with the F-measure for a range of internal parameter settings for each tool, i.e. the expected pathway size (M) for GXNA, the number of allowed exceptions (outliers) in a pathways (L) for KPM, and the pathway false discovery rate (FDR) for BioNet.
first computes local minima to serve as starting points for iteratively building sub-networks with \( n \) members and a maximal rank of \( m \). (e) GXNA\(^{29}\) is a score propagation method that selects random nodes as seeds of candidate sub-networks. These are iteratively extended by adding the neighboring node with the highest score. (f) KPM\(^{30}\) is another module cover approach inspired by DEGAS. It heuristically extracts maximal connected sub-networks, where all nodes except for at most \( K \) are differentially expressed in a given set of samples but at most \( L \). (g) PinnacleZ\(^{31}\) is an aggregate score propagation method that extends a set of random seed nodes to a number of sub-networks, which are then filtered by comparing their score distribution with that of randomly generated sub-networks. We provide more details for each of these tools in the supplementary material and briefly summarize some application cases. Please note that due to unavailability of a standalone application, clustering-based approaches could not be considered in our analysis.

The network data
The human protein–protein interaction (PPI) networks from the HPRD\(^{22}\) and the I2D\(^{23}\) were selected as test networks. The HPRD network consists of 9425 nodes and 36,811 edges. I2D network consists of 17,351 nodes and 217,379 edges. The networks could be modeled as undirected graphs \( G \) consisting of a set of vertices \( V \) and a set of edges \( E \). To gain insights into how network complexity affects the performance of the selected tools, we chose two networks that differ largely in network complexity, e.g. recent interaction networks such as I2D are orders of magnitude larger than classical PPI networks such as HPRD.

Simulation of gene expression datasets
For each graph \( G \), we split the nodes into FG and BG genes. Next, we simulated expression profiles for a set of 110 samples, 100 patient samples and 10 controls. Please note, we have chosen an arbitrary number of samples in each class. This choice can also be guided by (a) random sampling or (b) GEO GDS datasets, in order to reflect the median-experiment size in real case studies. We lack it in our current study due to limited time and computational power. We will work on this in the future, even though we expect the main findings of our study to remain unaffected.

We divided the simulation into three cases: (i) of patient-FG, (ii) of control-FG, (iii) of BG for patients and controls. We used the following models for the simulation:

(a) Varying variance. We simulated the expression profiles for the patient-FG genes and—BG genes from normal distributions \( N(\mu, v_{FG}) \) and \( N(\mu, v_{BG}) \) (Fig. 4a). We used the same distribution for (ii) and (iii). \( \mu = 0 \) was used in all cases. Signal strength was defined by the ratio \( v_{FG} / v_{BG} \). See Supplementary Table 3 for the specific values selected for variances \( v_{FG} \) and \( v_{BG} \) in each scenario.

(b) Varying mean. In this case, we simulated the expression profile for the FG genes from a normal distribution \( N(\mu_{FG}, v) \), while for the expression values for BG genes we used \( N(\mu = 0, v) \) and \( N(\mu_{BG}, v) \) for control-FG (Fig. 4b). Here, we used the setting \( v = 1 \) in all cases. Signal strength was defined by the difference \( \mu_{FG} - \mu_{BG} \). See Supplementary Table 2 for the specific values selected for means \( \mu_{FG} \) and \( \mu_{BG} \).

We further transformed the raw data into tool specific input, (e.g. p-
values), the particular pre-processing steps for each tool are specified in the supplementary material.

Selection of FG and BG nodes
We used two approaches for splitting a given graph $G = (V, E)$ into FG and BG nodes.

**Seed and extend approach.** This is a simple approach, in which we (a) randomly selected a seed node $s$, (b) expanded the sub-graph by including the nodes from the neighborhood of $s$, and (c) continued expanding the neighborhood until the sub-graph $s$ had $n \pm 20$ nodes or until the number of nodes in the sub-graph remains the same as $\pm 1$ iteration. If the sub-graph exceeds $n$, we chose $n-1$ neighbors randomly from the set of neighbors.

$AVD_k$. In this approach, the sparsity is modeled for a set of FG nodes in a graph as the average pairwise shortest distance between them. By generating FG node sets of varying distance $k$, the sparseness of the sub-networks (or closeness of the FG genes) could be adjusted. Given a network modeled as graph $G$ and a sparsity level $k$, we define the problem of selecting the FG nodes, as the average distance $k \pm \alpha$ problem, or $AVD_k$ for short. The goal was to identify a set of nodes from $W \subseteq V$ such that the average shortest distance between all nodes of $W$ have an average shortest distance (in $G$) to each other of $k \pm \alpha$. If not stated otherwise, we set $\alpha = 1$. Our sparseness function $AVD_k$, for instance, would thus return a set of nodes (which we will later on treat as FG nodes) having, on average, a pairwise shortest distance to each other between 2 and 4. We introduced $\alpha$ parameter to allow for inexactness and variability. Finding solutions for $AVD_k$ is computationally difficult (NP-complete). We first tackled it by using integer-linear programming, but it proved too slow and memory intense for graphs with thousands of nodes and edges (see Supplementary Material). To deal with the size of current biological networks, we developed a greedy heuristic running in polynomial time to compute solutions for the $AVD_k$ problem. The pseudo-code and additional details can be found in the supplementary material.

Availability. All scripts and data used in this study are available online at http://patheval.compbio.sdu.dk.

**ACKNOWLEDGEMENTS**

The research is supported in part by Villum foundation, ERC grant 676858-IMCIS, and the National Experimental Therapy Partnership (NEXT), which is financed by a grant from Innovation Fund Denmark.

**COMPETING INTERESTS**

The authors declare no competing interest.

**REFERENCES**

1. Barrett, T. et al. NCBI GEO: archive for functional genomics data sets-update. Nucleic Acids Res. 41, D991–995 (2013).
2. International Cancer Genome, C. et al. International network of cancer genome projects. Nature 464, 993–998 (2010).
3. Alexa, A., Rahnenführer, J. & Lengauer, T. Improved scoring of functional groups for graphs with thousands of nodes and edges (see Supplementary material). Of selecting the FG nodes, as the average distance $k \pm \alpha$ problem, or $AVD_k$ for short. The goal was to identify a set of nodes from $W \subseteq V$ such that the average shortest distance between all nodes of $W$ have an average shortest distance (in $G$) to each other of $k \pm \alpha$. If not stated otherwise, we set $\alpha = 1$. Our sparseness function $AVD_k$, for instance, would thus return a set of nodes (which we will later on treat as FG nodes) having, on average, a pairwise shortest distance to each other between 2 and 4. We introduced $\alpha$ parameter to allow for inexactness and variability. Finding solutions for $AVD_k$ is computationally difficult (NP-complete). We first tackled it by using integer-linear programming, but it proved too slow and memory intense for graphs with thousands of nodes and edges (see Supplementary Material). To deal with the size of current biological networks, we developed a greedy heuristic running in polynomial time to compute solutions for the $AVD_k$ problem. The pseudo-code and additional details can be found in the supplementary material.

Availability. All scripts and data used in this study are available online at http://patheval.compbio.sdu.dk.

**ACKNOWLEDGEMENTS**

The research is supported in part by Villum foundation, ERC grant 676858-IMCIS, and the National Experimental Therapy Partnership (NEXT), which is financed by a grant from Innovation Fund Denmark.

**COMPETING INTERESTS**

The authors declare no competing interest.

**REFERENCES**

1. Barrett, T. et al. NCBI GEO: archive for functional genomics data sets-update. Nucleic Acids Res. 41, D991–995 (2013).
2. International Cancer Genome, C. et al. International network of cancer genome projects. Nature 464, 993–998 (2010).
3. Alexa, A., Rahnenführer, J. & Lengauer, T. Improved scoring of functional groups for graphs with thousands of nodes and edges (see Supplementary Material). Of selecting the FG nodes, as the average distance $k \pm \alpha$ problem, or $AVD_k$ for short. The goal was to identify a set of nodes from $W \subseteq V$ such that the average shortest distance between all nodes of $W$ have an average shortest distance (in $G$) to each other of $k \pm \alpha$. If not stated otherwise, we set $\alpha = 1$. Our sparseness function $AVD_k$, for instance, would thus return a set of nodes (which we will later on treat as FG nodes) having, on average, a pairwise shortest distance to each other between 2 and 4. We introduced $\alpha$ parameter to allow for inexactness and variability. Finding solutions for $AVD_k$ is computationally difficult (NP-complete). We first tackled it by using integer-linear programming, but it proved too slow and memory intense for graphs with thousands of nodes and edges (see Supplementary Material). To deal with the size of current biological networks, we developed a greedy heuristic running in polynomial time to compute solutions for the $AVD_k$ problem. The pseudo-code and additional details can be found in the supplementary material.

Availability. All scripts and data used in this study are available online at http://patheval.compbio.sdu.dk.

**ACKNOWLEDGEMENTS**

The research is supported in part by Villum foundation, ERC grant 676858-IMCIS, and the National Experimental Therapy Partnership (NEXT), which is financed by a grant from Innovation Fund Denmark.

**COMPETING INTERESTS**

The authors declare no competing interest.

**REFERENCES**

1. Barrett, T. et al. NCBI GEO: archive for functional genomics data sets-update. Nucleic Acids Res. 41, D991–995 (2013).
2. International Cancer Genome, C. et al. International network of cancer genome projects. Nature 464, 993–998 (2010).
3. Alexa, A., Rahnenführer, J. & Lengauer, T. Improved scoring of functional groups for graphs with thousands of nodes and edges (see Supplementary Material). Of selecting the FG nodes, as the average distance $k \pm \alpha$ problem, or $AVD_k$ for short. The goal was to identify a set of nodes from $W \subseteq V$ such that the average shortest distance between all nodes of $W$ have an average shortest distance (in $G$) to each other of $k \pm \alpha$. If not stated otherwise, we set $\alpha = 1$. Our sparseness function $AVD_k$, for instance, would thus return a set of nodes (which we will later on treat as FG nodes) having, on average, a pairwise shortest distance to each other between 2 and 4. We introduced $\alpha$ parameter to allow for inexactness and variability. Finding solutions for $AVD_k$ is computationally difficult (NP-complete). We first tackled it by using integer-linear programming, but it proved too slow and memory intense for graphs with thousands of nodes and edges (see Supplementary Material). To deal with the size of current biological networks, we developed a greedy heuristic running in polynomial time to compute solutions for the $AVD_k$ problem. The pseudo-code and additional details can be found in the supplementary material.

Availability. All scripts and data used in this study are available online at http://patheval.compbio.sdu.dk.

**ACKNOWLEDGEMENTS**

The research is supported in part by Villum foundation, ERC grant 676858-IMCIS, and the National Experimental Therapy Partnership (NEXT), which is financed by a grant from Innovation Fund Denmark.

**COMPETING INTERESTS**

The authors declare no competing interest.

**REFERENCES**

1. Barrett, T. et al. NCBI GEO: archive for functional genomics data sets-update. Nucleic Acids Res. 41, D991–995 (2013).
2. International Cancer Genome, C. et al. International network of cancer genome projects. Nature 464, 993–998 (2010).
3. Alexa, A., Rahnenführer, J. & Lengauer, T. Improved scoring of functional groups for graphs with thousands of nodes and edges (see Supplementary Material). Of selecting the FG nodes, as the average distance $k \pm \alpha$ problem, or $AVD_k$ for short. The goal was to identify a set of nodes from $W \subseteq V$ such that the average shortest distance between all nodes of $W$ have an average shortest distance (in $G$) to each other of $k \pm \alpha$. If not stated otherwise, we set $\alpha = 1$. Our sparseness function $AVD_k$, for instance, would thus return a set of nodes (which we will later on treat as FG nodes) having, on average, a pairwise shortest distance to each other between 2 and 4. We introduced $\alpha$ parameter to allow for inexactness and variability. Finding solutions for $AVD_k$ is computationally difficult (NP-complete). We first tackled it by using integer-linear programming, but it proved too slow and memory intense for graphs with thousands of nodes and edges (see Supplementary Material). To deal with the size of current biological networks, we developed a greedy heuristic running in polynomial time to compute solutions for the $AVD_k$ problem. The pseudo-code and additional details can be found in the supplementary material.

Availability. All scripts and data used in this study are available online at http://patheval.compbio.sdu.dk.
44. Paull, E. O. et al. Discovering causal pathways linking genomic events to transcriptional states using tied diffusion through interacting events (TieDIE). *Bioinformatics* **29**, 2757–2764 (2013).

This work is licensed under a Creative Commons Attribution 4.0 International License. The images or other third party material in this article are included in the article’s Creative Commons license, unless indicated otherwise in the credit line; if the material is not included under the Creative Commons license, users will need to obtain permission from the license holder to reproduce the material. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/

© The Author(s) 2017

Supplementary Information accompanies the paper on the *npj Systems Biology and Applications* website (doi:10.1038/s41540-017-0007-2).