METHODOLOGY

Efficient randomization of biological networks while preserving functional characterization of individual nodes

Francesco Iorio[1†^], Andrea Gobbi[2†], Thomas Cokelaer[3†], Marti Bernardo Faura[1†], Giuseppe Jurman[2] and Julio Saez-Rodriguez[1,4*^]

Abstract

Background: Networks are popular and powerful tools to describe and model biological processes. Many computational methods have been developed to infer biological networks from literature, high-throughput experiments, and combinations of both. Additionally, a wide range of tools has been developed to calibrate experimental data onto reference biological networks, in order to extract meaningful modules. Many of these methods assess results' significance against null distributions of randomized networks. However, these standard unconstrained randomizations do not preserve the functional characterization of the nodes in the reference networks (i.e. their degrees and connection signs), hence including potential biases in the assessment.

Results: Building on our previous work about rewiring bipartite and undirected-unweighted networks, we propose a method for rewiring any type of unweighted networks. In particular we formally demonstrate that the problem of rewiring a signed and directed network preserving its functional connectivity (F-rewiring) reduces to the problem of rewiring two induced bipartite networks. Additionally, we reformulate the lower bound to the iterations' number of the switching-algorithm to make it suitable for the F-rewiring of networks of any size. Finally, we present *BiRewire3*, an open-source Bioconductor software enabling the F-rewiring of any type of unweighted network. We illustrate its application to a case study about the identification of modules from gene expression data mapped on protein interaction networks, and a second one focused on building logic models from more complex signed-directed reference signaling networks and phosphoproteomic data.

Conclusions: *BiRewire3* it is freely available at `https://www.bioconductor.org/packages/BiRewire/`, and it should have a broad application as it allows an efficient and analytically derived statistical assessment of results from any network biology tool.

Keywords: networks; pathways; rewiring

1 Background

Representing and modeling biological processes as networks, in particular signaling and gene regulatory relations, is a widely used practice in bioinformatics and computational biology, bridging these research fields to the vast repertoire of tools and formalisms provided by graph- and complex-network-theory. Furthermore, these representations facilitate an integrative analysis of experimental observations, either by derivation of these networks from the data, or by mapping the latter on the former. Hence, network-based approaches have become a popular paradigm in computational biology ([1, 2]).

In the last few years this has allowed the design of a broad assortment of algorithms and tools whose aim ranges from providing an interpretative framework for the modeled biological relations, to the identification of network-modules able *to explain* phenotypic traits and experimental data from large reference signaling graphs ([3, 4]). Many methods in this last class aim at identifying a sub-network, for example, composed by the most differentially expressed or significantly mutated genes ([5, 6, 7, 8, 9]), or that it is targeted by a given external perturbation ([10, 11, 12, 13, 14]). Toward this aim different optimization procedures have been used to analyze experimental data, identifying a

*Correspondence: iorio@ebi.ac.uk; saezrodriguez@gmail.com
[1]European Molecular Biology Laboratory - European Bioinformatics Institute, Wellcome Genome Campus, CB10 1SD Cambridge, UK
Full list of author information is available at the end of the article
[†]Equal contributors ^Co-corresponding authors

pathway that is deregulated in a given disease or whose activity is perturbed upon a given drug treatment.

In many approaches, directed signed networks (DSNs, formally defined in the following sections) are used to model pathways and to interlink pathways from a given collection. In these networks, nodes represent biological entities (typically proteins) while edges represent the biological relationships between them (e.g., the activity of protein A affects that of protein B). These edges have a direction to discriminate effectors and affected nodes in a modeled relation, and a sign to specify whether the modeled relation is an activation (positive sign) or an inhibition (negative sign). Unsigned/undirected edges modeling generic interactions can be also present. When available, sign and direction allow a more detailed detection of the nature of the interaction between the nodes. In this study, the number, sign and direction of a node's connections are cumulatively denoted by the *functional characterization level* (FCL) of the corresponding modeled biological entity (from now entity).

In a reference network modeling a set of interlinked pathways or protein-protein-interactions, the FCL might be high for a node that models a *functional hub*. For example a kinase phosphorylating a large number of substrate proteins will have a high number of outgoing edges with positive signs. Similarly, a gene activated by a large number of transcription factors will have a high number of positive in-coming edges. On the other hand the FCL might be strongly biased by the relevance of an entity in a given research field and the corresponding resource from which the network has been assembled. For example, in a cancer-focused reference network it is reasonable to find nodes that have a high FCL just because they have oncogenetic or tumor-suppressive properties, thus have been studied more than others. As a consequence, solutions to the network optimization problems tackled in bioinformatics (and mentioned above) can be strongly influenced by the topology of the initial network, and by the FCL of its nodes.

In an attempt to overcome this issue, some tools assess this bias by comparing their provided sub-network solutions with those that would be obtained (using the same experimental data and the same algorithm) across a large number of trials starting from a reference network that is a randomized version of the original one. Many other tools neglect this aspect and the significance of the solution is computed by randomizing the experimental data only. For both options, the expectation of some topological properties (for example the inclusion of a given edge or node) of the sub-network solutions is estimated by analyzing the random solutions obtained across the trials. In this way, the significance of these properties is quantified as the divergence from their expectation, testing against the null hypothesis that there is no association between the analyzed experimental data and the outputted sub-network solutions.

To our knowledge, all the existing methods assessing their solution significance through reference network randomizations make use of a simple edge shuffling. This means that in a randomization trial each edge of the network is simply set to link two randomly selected nodes. This implicitly means that null models resulting from this randomization strategy are totally unconstrained with regards to the degree of the nodes and the way they are linked to each other in the original network. Therefore, the impact of the FCL of the nodes in the original reference network on the outputted sub-network solution is not considered. In order to take this into account and to comprehensively avoid biases in the results, a constrained randomization strategy preserving the FCL of all the nodes in the original network must be used.

The problem of randomizing an undirected and unweighted network while preserving the degree of its nodes, i.e. the total number of incident edges for each node, is known in graph theory as *network rewiring* and unfortunately presents itself with analytical and numerical challenges ([15]). With the additional constrain that the network to rewire is bipartite (i.e. nodes can be partitioned into two sub-sets such that there are no edges linking nodes in the same set), this problem reduces to randomizing a binary matrix preserving its marginal totals, i.e. its row-wise and column-wise sums. Several algorithms exist to solve this problem ([16, 17]) but the computationally efficient randomization of moderately large matrices (therefore the rewiring of large bipartite networks) is still challenging. Additionally, to our knowledge, none of the methods published is formally shown to be able to actually simulate samplings from the uniform distributions of all the possible binary matrices with prescribed marginal totals. Such proof exists for methods rewiring directed binary networks based on *swap-and-fill* strategies applied to their adjacency matrices [18] but not dealing with DSNs. Finally, some recent methods have been proposed to solve the related (but yet different from FCL preserving rewiring) problem of randomizing metabolic networks in a mass-balanced way [19].

In [20] we showed how an algorithm based on a Monte Carlo procedure known as the *switching-algorithm* (SA) ([21]) can be used to efficiently randomize large cancer genomics datasets preserving the

mutation burdens observed across patients and the number of mutations harbored by individual genes, hence to efficiently rewire large bipartite networks. To this aim we derived a novel lower bound for the number of steps required by the SA in order for its underlying Markov chain to reach a stationary distribution. Additionally, we implemented the SA in the R package *BiRewire* (publicly available on Bioconductor ([20])) and we showed a massive reduction in computational time requirements of our package and bound with respect to other existing R implementations ([22]) and bounds ([21]).

Here (i) we introduce the problem of rewiring a DSN modeling a biological network in a way that the FCL of all the modeled entities is preserved: *F-Rewiring*, (ii) we formally show how this problem reduces to rewiring 2 bipartite networks, (iii) we provide a generalized bound to the SA for bipartite networks of any size, and (iv) we show the validity of the Markov chain convergence criteria used in our previous work for F-rewiring DSNs.
Finally, we provide an overview of the functions included in a new version of *BiRewire* for F-Rewiring, and we show results from two case studies where solutions obtained with two network optimization methods (BioNet ([9]), and CellNOpt ([23])) are assessed for statistical significance and intial reference network biases against constrained null models generated with *BiRewire*.

2 Methods

2.1 Preliminary notations

The problem we are tackling is the computationally efficient randomization of a directed and signed network (DSN) (formally defined below) in a way that some local features of its individual nodes are preserved.
In such a network $\mathcal{G} = (V, E)$, the edges in E can be encoded as triplets $(a, b, *)$ where a is called source node, b is called target node and $*$ is a label denoting the sign of the relation occurring among them, which could be positive, $* = +$, or negative, $* = -$.
According to this definition, in a DSN the edge $(a, b, +)$ is different from the edge $(a, b, -)$, thus making this formalism more flexible than that provided by a directed weighted network (with weights $\in \{+1, -1\}$). In fact, differently from such a model, in a DSN two edges with same terminal nodes and direction but different sign can coexist. In addition, a DSN is different and less general than a multidigraph (a directed multigraph), because only two possible edges with the same direction can coexist between the same couple of nodes.
Given an edge $e \in E$, we define the function $\lambda(e) : E \to \{+, -\}$, mapping each edge to its sign label.
Given a node $v \in V$, we define its *in-bound-star* $I(v)$ as the set of edges in E having v as destination, $I(v) = \{e \in E : e = (a, v, *)\}$. Similarly, considering the edges having v as source defines its *out-bound-star*, $O(v) = \{e \in E : e = (v, b, *)\}$. Imposing as additional condition for an edge to be included in these sets that of having a fixed sign label, defines positive and negative *in-bound* and *out-bound stars*. Formally, the v *positive-* (respectively *negative*) *in-bound-star* is the set of edges in G having v as destination and positive (respectively negative) label, $I^+(v) = \{e \in I(v) : \lambda(e) = +\}$ (respectively $I^-(v) = \{e \in I(v) : \lambda(e) = -\}$). Analogously, the v *positive-* (respectively *negative*) *-out-bound-star* is the set of edges in G having v as source and positive (respectively negative) label, $O^+(v) = \{e \in O(v) : \lambda(e) = +\}$ (respectively $O^-(v) = \{e \in O(v) : \lambda(e) = -\}$).
By naturally extending the definition of *node degree* (i.e. the number of edges connected to a node) to these formalisms, we call positive-in-degree of a node v the quantity $|I^+(v)|$ equal to the number of edges with positive label having v as destination. Similarly we define the v *negative-in-degree*, *positive-out-degree* and *positive-in-degree*, the quantities $|I^-(v)|$, $|O^+(v)|$ and $|O^-(v)|$, respectively.
In the light of the introduced notation, the object of this study can be redefined as the randomization of the edges of a DNS $\mathcal{G}$ while preserving not only its general node-degrees (*network rewiring*), but also all the signed degrees defined above, for all the nodes: *network F-rewiring*.
A biological pathway can be naturally represented through a DNS $\mathcal{G} = (V, E)$. In this case the nodes in V would represent biological entities, and the edges in E would represent functional relationships occurring among them, whose type would be defined by the sign label ($+$ for *activatory* and $-$ for *inhibitory* interactions), with directions indicating effector/affected roles (source/destination of the edges). In this case the signed degrees introduced above would define the functional characterization level (FCL) of the individual biological entities considering all the possible roles that they can assume within a given pathway.
Particularly the positive-out-degree of a node v would correspond to the level of characterization of the corresponding biological entity as *activator* of other entities; the negative-out-degree would correspond to its characterization as *inhibitor*; finally, the positive-, respectively negative-, in-degree of a node would correspond to the level of characterization of the corresponding entity as *activated*, respectively *inhibited*, by other entities in the same DSN.
As a consequence, the ultimate goal of this study is

to efficiently randomize a pathway (or a collection of interlinked pathways) in a way the functional characterization levels of its individual entities are preserved.

2.2 F-rewiring of a directed signed networks is reducible to the rewiring of two bipartite networks: reduction proof

Let us consider a directed signed network (DSN) $\mathcal{G} = (V, E)$, with $\lambda(e) \in \{-, +\}, \forall e \in E$ and a transforming function $f(\mathcal{G})$, from the set of all the possible DSNs to the set of all the possible pairs of bipartite networks (B_+, B_-), such as $B_* = (S_*, D_*, E_*)$, whose node sets are defined as $S_* = \{v \in V : \exists(v, x, *) \in E\}$, and $D_* = \{v \in V : \exists(x, v, *) \in E\}$, with $* \in \{+, -\}$. Worthy of note is that the same node of $\mathcal{G}$ can be both a source (therefore belonging the set S_*) for some edge in E, and a destination (therefore belonging to the set D_*) for some other edge in E. As a consequence f should also relabel the nodes (for example adding a subscript to the nodes in D_*). Here, for simplicity we will neglect this relabeling.

As a conclusion, the function f maps $\mathcal{G}$ to two bipartite networks (BNs) (B_+, B_-) such that $B_+ = (S_+, D_+, E_+)$ is the BN induced by the positive edges of $\mathcal{G}$, where all the sources of these edges are included in the first node set S_+, all the destinations in the second set D_+ and two nodes across these two sets are connected by an undirected edge if they are connected in the original network $\mathcal{G}$ by a positive edge that goes from the node in the first set to that in the second one. The second bipartite network of the pair B_- is similarly induced by the negative edges of $\mathcal{G}$. Formally $E_* = \{(s, d) : s \in S_*, d \in D_* \text{ and } \exists(s, d, *) \in E\}$, with $* \in \{+, -\}$. An example of this transformation is shown in Figure 1A.

It can be shown that such a function f realizes a bijection between the set of all the possible DNSs and the set of all the possible pairs of BNs ([24]). As a consequence its inverse f^{-1} is a function from the set of all the possible pairs of BNs to the set of all the possible DSNs, and it is defined as $f^{-1}(B_1, B_2) = \mathcal{G} = (V, E)$, where

$$V = S_1 \cup S_2 \cup D_1 \cup D_2,$$

$$E = \{(s, d, +) : (s, d) \in E_1 \text{ with } s \in S_1 \text{ and } d \in D_1\}$$

$$\cup\{(s, d, -) : (s, d) \in E_2 \text{ with } s \in S_2 \text{ and } d \in D_2\}.$$

With a little abuse of notation, we assume that f^{-1} re-assignes to the nodes their original labels before constructing the node/edge sets of G, if they were relabeled by the function f. An example of this inverse transformation is shown in Figure 1B.

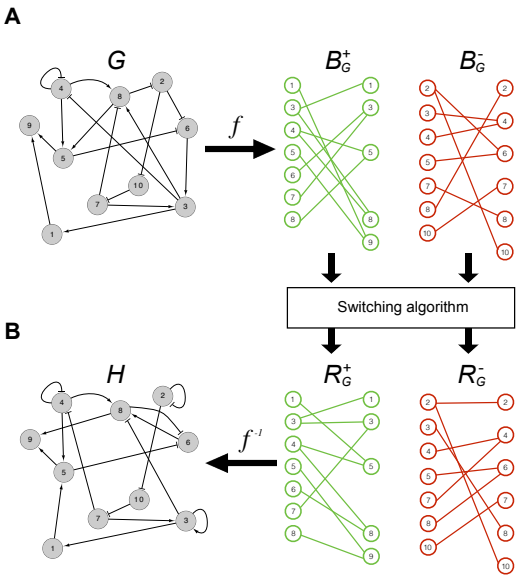

Figure 1 F-rewiring of directed signed networks is reducible to the rewiring of two bipartite graphs: (A) Scheme of the transformation function mapping a directed signed network (DSN) to two bipartite networks (BNs) induced by the positive, respectively negative, edges of the original network; (B) scheme of the inverse function that, after the two BNs induced by the edges of the original network have been rewired via the switching-algorithm, maps back the resulting rewired BNs to a DSN.

Proposition 2.1. *Let be $\mathcal{G} = (V, E)$ a DSN modeling a pathway (or a set of interlinked pathways) P, and f the transformation function described above $f(\mathcal{G}) = (B_+, B_-)$. If R_+ and R_- are rewired versions of B_+ and B_- respectively, then $f^{-1}(R_+, R_-) = \mathcal{H}$ is a randomized version of $\mathcal{G}$ in which the signed-directed degrees of all the nodes $v \in V$, i.e. the quantities $|I^+(v)|$, $|I^-(v)|$, $|O^+(v)|$, $|O^-(v)|$, are kept equal to their original values. This implies that $\mathcal{H}$ is an F-rewired version of G, hence a randomization of P in which the functional characterizations of the individual entities are preserved.*

Proof. First of all we need to show that $\mathcal{H}$ is a randomized version of $\mathcal{G}$, in other words that $\mathcal{H}$ is a directed signed network with the same nodes set and number of edges of $\mathcal{G}$ and the same signed-directed node degrees but a different edge set.

To this aim let be $\mathcal{H} = (W, F) = f^{-1}(R_+, R_-)$. Since a rewiring does not affect the node set of the transformed network, R_+ has the same node set of B_+, and R_- has the same node set of B_-. On the other hand, B_+ and B_- are the two bipartite networks induced by

the positive and negative edges (respectively) of $\mathcal{G}$. For construction, the union of their nodes gives V. Taken together these observations imply that $W = V$

From the definition of f, B_+ contains the positive edges in E and B_- the negative edges of E (whose terminal nodes have been possibly relabeled). From the definition of rewiring, the edge set of R_+ contains the same number of edges of B_+ but at least one edge not contained in B_+. Similarly the edge set of R_- contains the same number of edges of B_- and at least one edge not contained in B_-. Therefore, from the definition of f^{-1}, $|F| = |E|$ and F contains at least two edges that are not included in E. This imply that $F \neq E$.

As a conclusion $\mathcal{G}$ and $\mathcal{H}$ have the same set of nodes and number of edges but different edge sets.

Secondly we need to show that the signed degrees of all the nodes of $\mathcal{H}$ are equal to those of all the nodes in $\mathcal{G}$.

Let us assume that the positive-in-degrees of $\mathcal{H}$ are different from those of $\mathcal{G}$. From the f^{-1} definition, this implies that R_+ contains at least a node in the source set for which the degree is different from that of its counterpart in B_+. However, this contradicts R_+ being a rewired version of B_+. With the same argument it is possible to prove that all the signed-directed node degrees of $\mathcal{H}$ are equal to those of $\mathcal{G}$. □

2.3 Switching-algorithm lower bound for bipartite networks of any size

To rewire a bipartite network $B = (S, D, E)$, the switching-algorithm (SA) ([21]) performs a cascade of switching-steps (SS). In each of these SS two edges (a, b) and (c, d) are randomly selected from E and replaced with (a, d) and (c, b) if these two new edges are not already contained in E. In this case the SS under consideration is said *successful*.

Underlying the SA is a Markov chain whose states are different rewired versions of the initial network $\mathcal{G}$ and a transition between states is realized by a successful SS.

In [20] we prove that if executing a sufficiently large number of SS the SA can efficiently simulate samplings from the uniform distribution of all the possible bipartite networks with predefined node sets and prescribed node degrees. Therefore it can be used to rewire a bipartite network B that it is on average no more similar to B than are similar to each other two bipartite networks B_1 and B_2 sampled from the real uniform distribution of all the possible bipartite networks, with the same node sets and node degrees of B. To this aim, the number of SS to be performed before sampling the $(k + 1)$-th rewired network must be large enough to assure that the algorithm has *forgotten* the k-th sampled rewired network (the starting network $\mathcal{G}$ for $k = 0$). Formally, the number of SS between two following samplings must be at least equal to the burn-in time of the Markov chain underlying the SA, which is needed to reach a stationary distribution ([25, 26]). An example of this is shown in Figure 2: the 5 plots show results from a simulation study in which the SA has been used to rewire a synthetic bipartite network of 50 + 50 nodes and an edge density of 20% generated on purpose, and rewired versions of this network have been sampled at different intervals of SSs. A sampling interval of 1 SS produces sampled networks that are strongly related to each other (first plot). Gradually increasing the sampling interval (from 5 to 20 SS, 2nd to 4th plot), reduces the sampled network similarities but some local dependencies are maintained. At a sampling interval of 300 SS (5th plot) the Markov chain underlying the SS has reached its stationary distribution, the sampled network are completely unrelated and there are no dependencies. Therefore, for the bipartite network under consideration, a number of SS ≥ 300 is sufficient to simulate samplings from the uniform distribution of all the possible bipartite networks with 50 + 50 nodes and node degrees equal to those of the original network.

An empirical bound N' for the minimal number of SS to be performed by the SA between two consecutive samplings has been proposed in [21] as being equal to 100 times the number of edges of the bipartite network to rewire, thus making the rewiring of moderately large networks computationally very expensive.

By analyzing the trend of similarity to the original network along the sample path of the Markov chain simulation implemented by the SA, in [20] we proposed a novel lower bound to the number of SS needed to rewire large bipartite networks equal to

$$N = \frac{|E|}{2(1-d)} \ln\left[(1-d)|E|\right], \tag{1}$$

where E is the set of edges of the network to rewire $B = (S, D, E)$ and $d = |E|/(|S||D|)$ is its edge density. In [20] we show that this bound is much lower than N' and that our SA implementation and bound provide a massive reduction of the computational time required to rewire large bipartite networks (with thousands of nodes and tens of thousands of edges) with respect to other SA implementations ([22]) and the bound N'.

Here we provide a generalization of the lower bound N making the SA effective and computationally efficient in rewiring bipartite networks of any size. This is led by the observation that a DSN modeling a pathway (and the two bipartite networks induced by its

positive and negative edges, respectively) can be even composed by a modest number of nodes and edges.

As shown in the supplementary data of [20] (from now on going, equations from this paper will have GSD, for Gobbi supplementary data, as prefix), Equation 1 follows from the GSD-Equation 11 (page 20) and it is a simplified form of

$$N = \frac{|E|(1-d)\ln\left(\frac{|E|}{\epsilon} - \frac{|E|^2}{\epsilon t}\right)}{2p_r} \quad (2)$$

where $t = |S||D|$ is the total number of possible edges of the original network, $d = |E|/t$ is its edge density, p_r is the probability of a SS to be successful. ϵ is the accuracy of the bound in terms of distance (quantified through the convergence metric that we used to monitor the Markov chain underlying the SA, based on the number of edge shared by the original network and its rewired version at the generic k-th SS, and defined in GSD-Equation 9, page 19) from the real fixed point $\bar{x}$.

Under the assumption of a uniform degree distribution[1] we showed that $p_r = (1-d)^2$ (GDS-Equation 4, page 16). As a consequence Equation 2 can be rewritten as:

$$N = \frac{|E|\ln\left(\frac{|E|}{\epsilon} - \frac{|E|^2}{\epsilon t}\right)}{2(1-d)}, \quad (3)$$

which for $\epsilon = 1$, gives Equation 1.

Equation 3 expresses the lower bound of the number of SS as a function that accounts for the network topology and the estimated distance of the Markov chain underlying the SA from its steady-state, according to the convergence metric used in [20]. More detailed, this distance is equal to $|x^{(k)} - \bar{x}|$, where $x^{(k)}$ is the number of common edges between the original network and its rewired version after k SS, and $\bar{x}$ is the expected number of common edges between the original network and its rewired version after the Markov chain underlying the SA has reached its stationary distribution.

In our previous bound definition ϵ was defined in terms of number of edges, and N defined as in Equation 1 in order to have $|x^{(k)} - \bar{x}| \leq 1$ for $k \geq N$.

For large bipartite networks, i.e. $|E| > 10000$, a value of $\epsilon = 1$ guarantees a relative error $\delta < 0.01\%$ of edges for a number of SS $k \geq N$. However, for relatively smaller networks, for example when $|E| = 100$, a value of $\epsilon = 1$ implies a substantially increase in the relative error to $\delta = 1\%$, making the estimated lower bound N increasingly suboptimal with respect to the estimated real fixed point.

For this reason here we redefine the lower bound N for the number of SS as a function of its relative error δ, which quantifies its sub-optimality with respect to the estimated real fixed point. Through the simple substitution $\epsilon = |E|\delta$, Equation 3 can be rewritten as:

$$N = \frac{|E|(1-d)\ln\left(\frac{1-d}{\delta}\right)}{2p_r} = \Omega|E|$$

where $\Omega = \frac{(1-d)(\ln(1-d) - \ln\delta)}{2p_r}$ depends only on the level of accuracy δ, the density d of the original network and the probability p_r of a successful SS. For uniformly distributed degrees[1], i.e. $p_r = (1-d)^2$, this bound reads as:

$$N = \frac{|E|\ln\left(\frac{1-d}{\delta}\right)}{2(1-d)}. \quad (4)$$

A value of $\delta = 0.00005$ (corresponding to $\epsilon = 1$ edge when $|E| \sim 20000$), is used by default by our new implementation of the SA in the new version of the package *BiRewire* but this parameter can also be set to a user defined value, making our tool and bound suitable for the rewiring of bipartite networks of any size. Additionally, the choice of a suitable value for this parameter can be determined by visually inspecting the SA Markov chain convergence with a new dedicated function (described in Section 3.1)

2.4 Convergence criteria for signed directed networks

In [20] we showed that the convergence criteria we used to estimate our lower bound N for the number of switching-steps (SS) needed to rewire bipartite networks can be applied also to the more generic case of undirected networks.

To show the validity of this criteria for F-rewiring of directed signed networks (DSNs) let us observe that the Jaccard Index (J) used to assess the similarity between two DSN with the same set of nodes and same number of edges: $\mathcal{G} = (V, E)$ and $\mathcal{H} = (V, F)$ is defined as

$$J(\mathcal{G}, \mathcal{H}) = \frac{|E \cap F|}{|E \cup F|} = \frac{x}{2|E| - x}$$

where $x = |E \cap F|$ is the number of common edges and the last equivalence holds because the two DSNs have the same number of edges. When estimated for bipartite networks, our N guarantees that the number

[1] Our proof applies also to non uniform degree distributions, leading to the same conclusions for the case of directed signed networks. Here we use the uniform case for simplicity.

of common edges between an initial network B and its rewired version at the N-switching-step is asymptotically minimized.

Proposition 2.2. *Let be R_+ and R_- the rewired versions of two bipartite networks B_+ and B_- obtained through a number of switching-steps respectively equal to N_+ and N_- (both computed using Equation 4), and such that $(B_+, B_-) = f(\mathcal{G})$ (where f is the transformation function defined in section 2.2 and $\mathcal{G}$ a DSN). Then the Jaccard similarity between $\mathcal{G}$ and $\mathcal{H} = f^{-1}(R_+, R_-)$ is minimized.*

Proof. $J(\mathcal{G}, \mathcal{H})$ reaches a minimum when the number of common edges x between $\mathcal{G}$ and $\mathcal{H}$ reaches a minimum. x is given by the sum of the number of common positive and negative edges across the two networks, namely $x = x_+ + x_-$. Given that $\mathcal{H} = f^-1(R_+, R_-)$, x_+ is the number of common edges between B_+ and R_+. Analogously x_- is the number of common edges between B_- and R_-. Since R_+ and R_- are rewired version of B_+ and B_- computed through N_+ and N_- (minimizing x_+ and x_-, respectively) also $x = x_+ + x_-$ is minimized. □

3 Results

3.1 Overview of the new functions included in *BiRewire* v3.0.0

The R-package *BiRewire* (http://bioconductor.org/packages/BiRewire/) was originally designed to efficiently rewire large bipartite networks ([20]). We have performed a major update, by including functions to:

- read/write directed signed networks (DSN) from/to simple interaction format (SIF) files;
- perform the transformation f (and its inverse f^{-1}) to derive bipartite networks induced by positive and negative edges of a DSN (and vice-versa);
- F-rewire a DSN by applying the switching-algorithm (SA) to the two corresponding induced bipartite networks with numbers of switching-steps automatically determined for both networks individually, using Equation 3;
- sample K rewired versions of a network: this function runs K instances of the SA in cascade; each of these instances performs a number of switching-steps (SS) determined using Equation 3. This function can take in input a bipartite network, an undirected network or a DSN (in this case Equation 3 is used individually for the two bipartite networks induced by the positive and negative edges of the DSN, respectively);
- monitor the convergence of the Markov chain underlying the SA on user defined networks. This routine samples a user-defined number of networks at user defined intervals of SS. For each of these intervals, it computes a Jaccard similarity [27] pair-wisely comparing the sampled networks to each other; finally it plots the sampled networks in a plane where points proximities reflect Jaccard similarities of the corresponding networks and point coordinates are computed through the generalized multidimensional scaling method *t-SNE* ([28]); this function gives in output the network coordinates of such scale reductions and produce the plots shown in Figure 2. Also in this case the inputted graph can be a bipartite network, an undirected network or a DSN;
- perform an analysis of the trends of Jaccard similarity across SS. This function performs a user-defined number of independent runs of the SA, computing the mean value and a confidence intervals of the observed pairwise Jaccard similarities between the obtained rewired networks. The result is a dataset containing the Jaccard similarity scores computed and sampled at user-defined intervals of SS, and a plot similar to that showed in Figures 3A and 4A. This function takes in input a bipartite network or an undirected network or a DSN.

Worthy of note is that, supporting the analysis of DSNs, our package can handle also generic directed graphs, therefore with *BiRewire3* it is now possible to rewire any kind of unweighted networks.
We have developed also a cython wrapper of the corresponding C library for Python users. A first release (with some basic functions) can be found in https://github.com/andreagobbi/pyBiRewire.

3.2 Case study 1: BioNet

The R package *BioNet* [29] provides a set of methods to map gene expression data onto a large reference biological network, and to identify (with a heuristic method) a maximal scoring sub-network (MSS), which a is a set of connected nodes (or *module*) with unexpectedly high levels of differential expression ([30]). Several other methods moving along the same lines exist (as, among others, EnrichNet ([6])). Here we focus on BioNet because it can be considered a typical example among these methods, and we show how *BiRewire3* can be used to estimate the impact of the reference network topology and the functional characterization level (FCL) of its node on the optimal module outputted by this tool.

The initial reference network used by BioNet (the *Interactome*) is a large undirected protein-protein-interaction network assembled from HPRD ([31]) and encompassing 9392 nodes and 36504 edges. In [29], the authors show an application of BioNet to gene expression data from a diffuse large B-cell lymphoma patient dataset, with corresponding survival data. After determining gene-wise P-values for differential expression and risk-association, the authors aggregate them and fit a beta-uniform mixture model to the distribution of aggregated P-values that yields a final score (accounting for both considered factors) for each gene: the higher this score the more a gene is differentially expressed across the contrasted groups of patients. Then the methods proceeds with mapping these scores onto the Interactome nodes and applying a heuristic method ([9]) it identifies a sub-network (referred to as a module) that is a sub-optimal estimate of the MSS. This module is shown in Figure 3C and the BioNet package vignette contains detailed instructions on how to reproduce this result.

To evaluate the impact of the FCLs of the Interactome nodes on the module outputted by BioNet when used on the DLBC dataset, we generated 1000 F-rewired versions of the Interactome with *BiRewire3* and used each of them as initial reference network in 1000 individual BioNet runs, using the DLBC dataset as input.

To this aim we first conducted a *BiRewire3* analysis (using the dedicated function of our package) to determine the number of switching-steps (SS) to be performed by the switching-algorithm (SA) in order to F-rewire the Interactome. This function makes use of the convergence criteria we designed in [20], which is based on the estimated time, in terms of SS, in which the Jaccard similarity (JS) between the original network and its rewired version at the k-th SS reaches a plateau (Figure 3A). In [20] we showed that this criteria is equivalent to other established methods to monitor Markov chain convergence when the states are networks. In addition its relatively simple formulation consents the analytical derivation of an estimated plateau time, i.e. our bound N. Neverthless, our package allows also a visual inspection of the optimality of the estimated bound N showing how independent are F-rewired versions of an initial network sampled at a number of user-defined SS intervals as well as every N SS (Figure 2).
These preliminary analyses resulted in a required number of SS equal to $N = 170491$ (Figure 3A) and showed that this number of SS is actually sufficient to generate unrelated F-rewired versions of the Interactome,

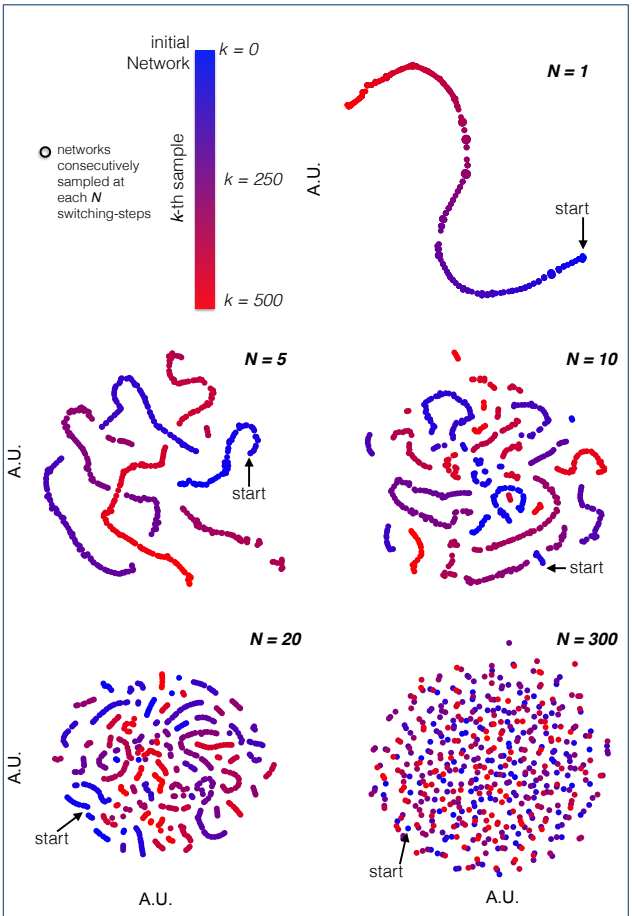

Figure 2 Rewired network samplings using the switching-algorithm (SA) at different sampling intervals, in terms of switching-steps (SS). Points represent sampled networks, arrows indicate a starting synthetic network, and colors indicate the sampling order. Point proximities reflect corresponding network similarities quantified through the Jaccard index. Point coordinates have been obtained with a generalized multi-dimensional scaling procedure (t-SNE).

thus to simulate samplings from the uniform distribution of all the possible networks with the same number of nodes and FCLs of the Interactome (Figure 3B). Generating 1000 F-rewired versions of the Interactome sampled each N SS required $\sim$ 2 hours on a 4 core 2.4 Ghz computer with 8GB memory.

Running 1000 independent instances of BioNet using each of these F-rewired Interactome as reference network and the DLBC dataset in input resulted into 1000 different module solutions (rewired solutions). For each of the nodes included in the original BioNet module solution (Figure 3C), we quantified the ratio of rewired solutions including them and we investigated how this quantity related to the corresponding

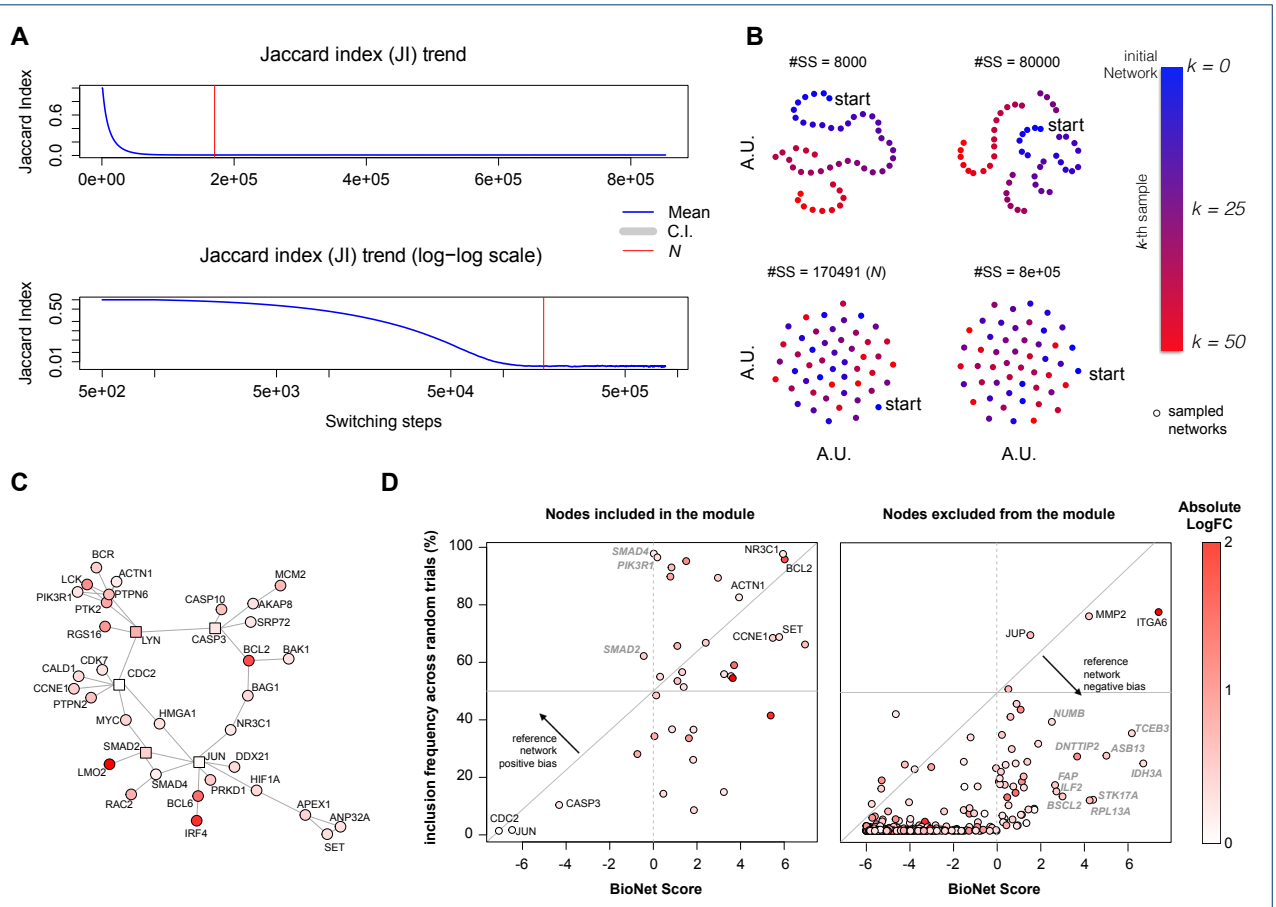

Figure 3 BioNet study case. (A) Analysis of the Jaccard index trend across switching-steps (SS) while rewiring the BioNet reference Interactome and estimation of the lower bound N; (B) visual inspection of the switching-algorithm Markov chain convergence to verify the suitability of the estimated bound (see Figure 2 legend for further details); (C) Interactome module outputted by BioNet while analyzing the DLBC dataset; (D) scatter plots of BioNet scores vs. frequency of inclusion in the rewired solutions for all the nodes included in the BioNet module (left plot) and for all the other Interactome nodes contained in the DLBC dataset (right plot).

BioNet scores (3D). As expected, we observed a significant correlation ($R = 0.51$, $p = 0.001$). In fact, as per the definition of the MSS, it is reasonable that nodes with high scores (such as, for example *NR3C1* and *BCL2*) tend to be included in the module outputted by BioNet regardless their edges and degree in the reference Interactome. Similarly, nodes with large negative scores (such as *CDC2* and *JUN*) are included in the module only because they link high scored nodes and it is obvious that they do not tend to be included in the rewired solutions, as in this case the way they are interlinked to other nodes is crucial.

Nevertheless, a number of nodes (such as, *SMAD4*, *SMAD2* and *PIK3R1*) have modest score but tend to be included very frequently in the rewired solutions. This hints that what leads the inclusion of such nodes in the BioNet module is their high FCL. As a confirmation of this, *SMAD4*, *SMAD2* and *PIK3R1* fall over the 99th percentile when sorting all the nodes in the Interactome (and included in the DLBC) based on their FCL (which in this case corresponds to their degree). This is a proof that the reference network provides the BioNet outputted module with a *positive bias*, and that at least some nodes are included in the solution because of their high FCL.

When extending this analysis to the nodes of the Interactome (included in the DLBC dataset) that are not present in the module outputted by BioNet we observed again an expected significant correlation ($R = 0.51$, $p < 10^{-16}$), and some nodes (such as *JUP*, *MMP2* and *ITGA6*) with high scores frequently included in the rewired solutions (the fact that these nodes do not appear in the BioNet outputted module is due to the sub-optimality of the used heuristic). However we also observed a large number of nodes

(such as *RPL13A*, *STK17A* and *IDH3A*) scored high but relatively infrequently included in the rewired solutions. This hints that these nodes are penalized by their low FCL in the reference Interactome, thus proving the existence of a *negative bias* provided by the reference Interactome to the BioNet outputted module, and that at least some nodes are not included in the solution because of their low FCL.

An indication of both these biases, together with diagnostic plots and statistics would complement and complete the output of many valuable and widely used tools, such as BioNet.

3.3 Case study 2: CellNOpt

CellNOpt (`www.cellnopt.org`) is a tool used to train logic models of signal transduction starting from a reference directed signed network (DSN) called a prior knowledge network (PKN), describing causal interactions among signaling species (obtained typically from literature), and a set of experimental data (typically phosphorylation), obtained upon various perturbatory conditions ([23]).
CellNOpt converts the PKN into a logic model and identifies the set of interactions (logic gates) that best explain the experimental data. This is performed through a set of Bioconductor packages supporting a number of mathematical formalisms from Boolean models to ordinary differential equations.
Through a built-in genetic algorithm CellNOpt identifies a family of subnetworks from the reference DSN (from now, models) together with the value of the objective function (the *model score*) quantifying at what extent each model is able to explain the experimental data (the lower this value the better is the fit of the model to the data). By default, the best model with the lowest score denoted $\hat{\delta}$ is returned to the end-users. Note, however, that multiple models may be returned if they cannot be discriminated given the experimental evidence. Besides, to account for experimental noise, users may also provide a parameter, which is called tolerance (in percentage), that will keep all models below a threshold defined as $\lambda = \hat{\delta}(1 + tolerance)$.
Setting this tolerance parameter is non-trivial and depends largely on the experimental error.

Here we show how *BiRewire3* can be used to identify such a threshold as the maximal value whose deviance from expectation is statistically significant. Similarly to the previous case study, this expectation can be empirically estimated by running a large number of independent CellNOpt runs using F-rewired versions of the initial reference signaling network and the same experimental data. Thus accounting for the effect of the node FCLs on both scores and outputted models. To this aim, we used the same reference PKN network and phosphoproteomic data used in [23], which has about 80 nodes and 120 directed and signed edges. This was a study on human liver cell and hence the network is called liver-PKN hereafter. With the *BiRewire3* package we generated (in less than 10 seconds, on a standard unix laptop) 1000 F-rewired versions of the liver-PKN, visually inspecting (as in the previous case study) the optimality of our estimated lower bound N for the number of switching-steps (SS) to be performed by the switching-algorithm (SA) (Figure 4AB) between one sampled F-rewired network and the following one. Subsequently we run 1000 independent instances of CellNOpt (using the CellNOptR package ([23]), v1.16 available on Bioconductor at `https://www.bioconductor.org/packages/CellNOptR/`) on each of these F-rewired liver-PKN networks and the same phosphoproteomic dataset (obtaining one *rewired model* per each analysis), as well as a final run using the original liver-PKN network (obtaining a family of 1000 different models). When comparing the two populations of CellNOpt scores obtained from these two analyses we observed, as expected, a notably statistically significant difference (t-test p-value $< 10^{-16}$, Figure 4C). Finally, using the distribution of scores of the rewired models we computed empirical p-values for the CellNOpt scores for the entire model family outputted by the final run (making use of the original liver-PKN).
For a given score δ_i corresponding to the $i-$th model of the family, a p-value was set equal to the number of rewired models m such that $\delta_m \geq \delta_i$ divided by 1000 (the number of tested f-rewired liver-PKNs). More than 90% of the models in the outputted family had a CellNOpt score significantly divergent from expectation (p-value < 0.05) and the estimated score threshold guaranteeing this (or a greater) divergence from expectation, thus a minimal impact of the initial liver-PKN FCLs, was equal to 0.06.

In summary, *BiRewire3* could be effectively used to determine a score threshold on an analytical ground, based on which meaningful models could be selected from the family outputted by CellNOpt for further analyses, and finally assemble a consensual model solution.

4 Conclusion

BiRewire3 is a one-stop tool to rewire in a meaningful way any type of unweighted networks (undirected, directed, and signed) currently used to model different datasets and relations in computational biology (including presence-absence matrices, genomics datasets,

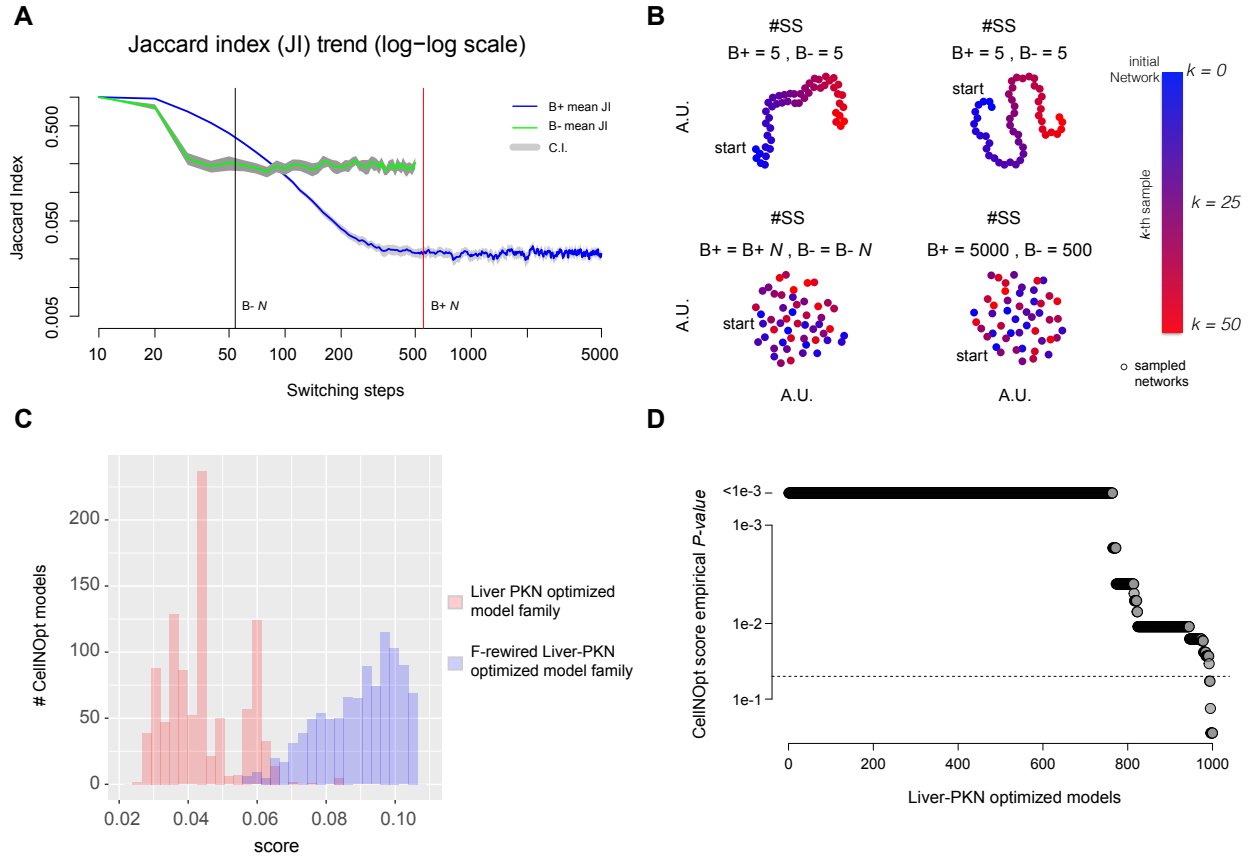

Figure 4 CellNOpt study case. (A) Analysis of the Jaccard index trend across switching-steps (SS) while rewiring the two bipartite network induced by the positive (respectively negative) edges of the reference DSN (liver-PKN) and estimation of the lower bounds for the number of switching-steps; (B) visual inspection of the switching-algorithm Markov chain convergence to verify the suitability of the estimated bounds (see Figure 2 legend for further details); (C) Comparison of the CellNOpt scores and the rewired scores; (D) Empirical p-values of the CellNOpt scores across the entire family of models.

pathways and signaling networks) in an computationally efficient way.

Our package is available as free open source software on Bioconductor and, as we showed in our case studies, it can be easily combined into computational pipelines together with a wide range of existing bioinformatics tools aiming at integrating signaling networks with experimental data.

This will allow existing software and tools to be complemented with a powerful and robust framework to compute a wide range of constrained null models, useful for testing the significance of their solutions, and to investigate how the topology of used reference networks can potentially bias their results.

Moreover, the range of applicability of *BiRewire3* goes beyond computational biology, and includes all those fields making use of tools from network theory, from operative research, to microeconomy, and ecological research (an example of the application of *BiRewire* application in a micro-economy and technology patent study can be found at http://arxiv.org/abs/1509.07285).

Competing interests
The authors declare that they have no competing interests.

Acknowledgements
FI has been partially supported by the European Bionformatics Institute and Wellcome Trust Sanger Institute post-doctoral (ESPOD) program.

Author details
[1]European Molecular Biology Laboratory - European Bioinformatics Institute, Wellcome Genome Campus, CB10 1SD Cambridge, UK. [2]Fondazione Bruno Kessler, I-38122 Povo (Trento), Italy. [3]Institut Pasteur - Bioinformatics and Biostatistics Hub - C3BI, USR 3756 IP CNRS, Paris, France. [4]Joint Research Centre for Computational Biomedicine (JRC-COMBINE), RWTH Aachen University, Faculty of Medicine, MTI2 Wendlingweg 2, 52074 Aachen, Germany.

References

1. Ma'ayan, A.: Introduction to network analysis in systems biology. Science signaling **4**(190), 5 (2011)
2. Iorio, F., Saez-Rodriguez, J., Bernardo, D.d.: Network based elucidation of drug response: from modulators to targets. BMC systems biology **7**(1), 139 (2013)

3. Saez-Rodriguez, J., MacNamara, A., Cook, S.: Modeling Signaling Networks to Advance New Cancer Therapies. Annual review of biomedical engineering **17**, 143–163 (2015)
4. Pavlopoulos, G.A., Secrier, M., Moschopoulos, C.N., Soldatos, T.G., Kossida, S., Aerts, J., Schneider, R., Bagos, P.G.: Using graph theory to analyze biological networks. BioData mining **4**, 10 (2011)
5. Mitra, K., Carvunis, A.-R., Ramesh, S.K., Ideker, T.: Integrative approaches for finding modular structure in biological networks. Nat Rev Genet **14**(10), 719–732 (2013). doi:10.1038/nrg3552
6. Glaab, E., Baudot, A., Krasnogor, N., Schneider, R., Valencia, A.: EnrichNet: network-based gene set enrichment analysis. Bioinformatics **28**(18), 451–457 (2012)
7. Vandin, F., Upfal, E., Raphael, B.J.: Algorithms for detecting significantly mutated pathways in cancer. Journal of Computational Biology **18**(3), 507–522 (2011)
8. Wang, X., Terfve, C., Rose, J.C., Markowetz, F.: HTSanalyzeR: an R/Bioconductor package for integrated network analysis of high-throughput screens. Bioinformatics **27**(6), 879–880 (2011)
9. Dittrich, M.T., Klau, G.W., Rosenwald, A., Dandekar, T., Müller, T.: Identifying functional modules in protein-protein interaction networks: an integrated exact approach. Bioinformatics **24**(13), 223–31 (2008)
10. Hill, S.M., Heiser, L.M., Cokelaer, T., Unger, M., Nesser, N.K., Carlin, D.E., Zhang, Y., Sokolov, A., Paull, E.O., Wong, C.K., Graim, K., Bivol, A., Wang, H., Zhu, F., Afsari, B., Danilova, L.V., Favorov, A.V., Lee, W.S., Taylor, D., Hu, C.W., Long, B.L., Noren, D.P., Bisberg, A.J., HPN-DREAM Consortium, Mills, G.B., Gray, J.W., Kellen, M., Norman, T., Friend, S., Qutub, A.A., Fertig, E.J., Guan, Y., Song, M., Stuart, J.M., Spellman, P.T., Koeppl, H., Stolovitzky, G., Saez-Rodriguez, J., Mukherjee, S.: Inferring causal molecular networks: empirical assessment through a community-based effort. Nature Methods **13**(4), 310–318 (2016)
11. Kulbe, H., Iorio, F., Chakravarty, P., Milagre, C.S., Moore, R., Thompson, R.G., Everitt, G., Canosa, M., Montoya, A., Drygin, D., Braicu, I., Sehouli, J., Saez-Rodriguez, J., Cutillas, P.R., Balkwill, F.R.: Integrated transcriptomic and proteomic analysis identifies protein kinase CK2 as a key signaling node in an inflammatory cytokine network in ovarian cancer cells. Oncotarget (2016)
12. Melas, I.N., Sakellaropoulos, T., Iorio, F., Alexopoulos, L.G., Loh, W.-Y., Lauffenburger, D.A., Saez-Rodriguez, J., Bai, J.P.F.: Integrative Biology. Integrative Biology, 1–17 (2015)
13. Woo, J.H., Shimoni, Y., Yang, W.S., Subramaniam, P., Iyer, A., Nicoletti, P., Martínez, M.R., López, G., Mattioli, M., Realubit, R., Karan, C., Stockwell, B.R., Bansal, M., Califano, A.: Elucidating Compound Mechanism of Action by Network Perturbation Analysis. Cell **162**(2), 441–451 (2015)
14. Lecca, P., Priami, C.: Biological network inference for drug discovery. Drug Discovery Today, 1–9 (2012)
15. Bender, E., Canfield, E.: The asymptotic number of labelled graphs with given degree sequences. Journal of Combinatorial Theory, Series A **24**, 296–307 (1978)
16. Strona, G., Nappo, D., Boccacci, F., Fattorini, S., San-Miguel-Ayanz, J.: A fast and unbiased procedure to randomize ecological binary matrices with fixed row and column totals. Nature communications **5**, 4114 (2014)
17. Gotelli, N., Entsminger, G.: Swap and fill algorithms in null model analy-sis: rethinking the knight's tour. Oecologia **129**, 281–291 (2001)
18. Roberts, E.S., Coolen, A.C.C.: Unbiased degree-preserving randomization of directed binary networks. Physical Review E **85**(4 Pt 2), 046103 (2012)
19. Basler, G., Ebenhöh, O., Selbig, J., Nikoloski, Z.: Mass-balanced randomization of metabolic networks. Bioinformatics **27**(10), 1397–1403 (2011)
20. Gobbi, A., Iorio, F., Dawson, K.J., Wedge, D.C., Tamborero, D., Alexandrov, L.B., López-Bigas, N., Garnett, M., Jurman, G., Saez-Rodriguez, J.: Fast randomization of large genomic datasets while preserving alteration counts. Bioinformatics **30**(17), 617–623 (2014). doi:10.1093/bioinformatics/btu474
21. Milo, R., Kashtan, N., Itzkovitz, S., Newman, M.E.J., Alon, U.: On the uniform generation of random graphs with prescribed degree sequences. Arxiv preprint cond-mat/0312028 (2003)
22. Csardi, G., Nepusz, T.: The igraph software package for complex network research. InterJournal, Complex Systems **1695**, 38 (2006)
23. Terfve, C., Cokelaer, T., Henriques, D., MacNamara, A., Goncalves, E., Morris, M.K., van Iersel, M., Lauffenburger, D.A., Saez-Rodriguez, J.: CellNOptR: a flexible toolkit to train protein signaling networks to data using multiple logic formalisms. BMC Systems Biology **6**(1), 133 (2012). doi:10.1186/1752-0509-6-133
24. Chen, W.-K.: Graph Theory and Its Engineering Applications. World Scientific Publishing Co Pte Ltd, ??? (1997)
25. Ray, J., Pinar, A., Seshadhri, C.: Are We There Yet? When to Stop a Markov Chain while Generating Random Graphs. In: Algorithms and Models for the Web Graph, pp. 153–164. Springer, Berlin, Heidelberg (2012)
26. Stanton, I., Pinar, A.: Constructing and sampling graphs with a prescribed joint degree distribution. Journal of Experimental Algorithmics **17**(1), 3–1 (2012)
27. Jaccard, P.: Etude comparative de la distribution florale dans une portion des Alpes et du Jura. Bulletin de la Socit Vaudoise des Sciences Naturelles **37**, 142 (1901)
28. van der Maaten, L., Hinton, G.E.: Visualizing High-Dimensional Data Using t-SNE. Journal of Machine Learning Research **9**, 2579–2605 (2008)
29. Beisser, D., Klau, G.W., Dandekar, T., Müller, T., Dittrich, M.T.: BioNet: an R-Package for the functional analysis of biological networks. Bioinformatics **26**(8), 1129–1130 (2010)
30. Ideker, T., Ozier, O., Schwikowski, B., Siegel, A.F.: Discovering regulatory and signalling circuits in molecular interaction networks. Computer applications in the biosciences: CABIOS **18 Suppl 1**, 233–40 (2002)
31. Mishra, G.R., Suresh, M., Kumaran, K., Kannabiran, N., Suresh, S., Bala, P., Shivakumar, K., Anuradha, N., Reddy, R., Raghavan, T.M., Menon, S., Hanumanthu, G., Gupta, M., Upendran, S., Gupta, S., Mahesh, M., Jacob, B., Mathew, P., Chatterjee, P., Arun, K.S., Sharma, S., Chandrika, K.N., Deshpande, N., Palvankar, K., Raghavnath, R., Krishnakanth, R., Karathia, H., Rekha, B., Nayak, R., Vishnupriya, G., Kumar, H.G.M., Nagini, M., Kumar, G.S.S., Jose, R., Deepthi, P., Mohan, S.S., Gandhi, T.K.B., Harsha, H.C., Deshpande, K.S., Sarker, M., Prasad, T.S.K., Pandey, A.: Human protein reference database–2006 update. Nucleic Acids Research **34**(Database issue), 411–4 (2006)