

Integrating literature-constrained and data-driven inference of signalling networks

Federica Eduati^{1,2}, Javier De Las Rivas³, Barbara Di Camillo¹, Gianna Toffolo¹ and Julio Saez-Rodriguez^{2,*}

¹Department of Information Engineering, University of Padova, Padova, 31050, Italy

²European Bioinformatics Institute (EMBL-EBI), Wellcome Trust Genome Campus, Cambridge, CB10 1SD, UK

³Bioinformatics & Functional Genomics Group, Cancer Research Center (CSIC/USAL), Salamanca, Spain

Associate Editor: Prof. Gunnar Ratsch

ABSTRACT

Motivation: Recent developments in experimental methods allow generating increasingly larger signal transduction datasets. Two main approaches can be taken to derive from these data a mathematical model: to train a network (obtained e.g. from literature) to the data, or to infer the network from the data alone. Purely data-driven methods scale up poorly and have limited interpretability, while literature-constrained methods cannot deal with incomplete networks.

Results: We present an efficient approach, implemented in the R package CNORfeeder, to integrate literature-constrained and data-driven methods to infer signalling networks from perturbation experiments. Our method extends a given network with links derived from the data via various inference methods, and uses information on physical interactions of proteins to guide and validate the integration of links. We apply CNORfeeder to a network of growth and inflammatory signalling, obtaining a model with superior data fit in the human liver cancer HepG2 and proposes potential missing pathways.

Availability: CNORfeeder is in the process of being submitted to Bioconductor and in the meantime available at www.ebi.ac.uk/~cokelaer/cnofeeder/.

Contact: saezrodriguez@ebi.ac.uk

Supplementary information: available at www.ebi.ac.uk/~cokelaer/cnofeeder/

1 INTRODUCTION

Information about signalling networks is increasingly abundant. Thanks to novel high-throughput methods, large amounts of data about the interactions among proteins is available, which is encompassed in (unsigned and undirected) protein-protein interaction networks (PINs) (Pieroni, E., et al. 2008). More precise (but with less coverage) information is derived from literature and is often described by means of signed and directed causal interactions among proteins. These give rise to what we will call here prior knowledge networks (PKN). PKNs are partially collected in differ-

ent databases (e.g. KEGG (Ogata, H., et al. 1999) Reactome (Joshi-Tope, G., et al. 2005), WikiPathways (Pico, A.R., et al. 2008); several are accessible via the portal Pathway Commons (Cerami, E.G., et al. 2011)). These databases typically contain literature-derived interactions curated with different degrees of stringency and based on experimental publications under different experimental conditions and on different cell types.

PKNs are, for example, very useful to study topological properties of networks (Ma'ayan, A., et al. 2005) or to map data (Ideker, T. and Sharan, R. 2008, Terfve, C. and Saez-Rodriguez, J. 2012). However, they are not functional in the sense that they cannot be used for simulation of a signaling process and therefore for prediction of the outcome of a certain experiment, which is fundamental to understand signal transduction and its alterations.

The most common way to model a signalling network is based on writing down its biochemistry and subsequently translating it to a mathematical form, typically differential equations (Aldridge, B.B., et al. 2006). However, information in PKNs often lacks the required mechanistic detail. In these cases, logic formalisms are a useful approach since all they need is to add logic gates to the existing (signed and directed) interactions.

One can generate logic gates by manual curation based on literature, e.g. (Calzone, L., et al. 2010, Saadatpour, A., et al. 2011, Samaga, R., et al. 2009), reviewed in (Morris, M.K., et al. 2010, Watterson, S., et al. 2008). An alternative to manual curation consists of generating a logic model from the PKN that is subsequently trained to experimental data (Saez-Rodriguez, J., et al. 2009). This method, implemented in the Bioconductor package CellNOptR (<http://www.ebi.ac.uk/saezrodriguez/software.html>), provides context-specific models with predictive power. It is efficient at handling large amounts of data as the space of possible models is limited by the prior knowledge. This key feature of the approach, however, is also its main limitation: there might be missing links due to the fact that databases are not complete, and that the effect of cross talk between pathways is often not taken into account in the canonical linear representation of the pathways. Adding to the PKN links based on the dedicated data leads to an improved goodness of fit (Saez-Rodriguez, J., et al. 2009).

*To whom correspondence should be addressed.

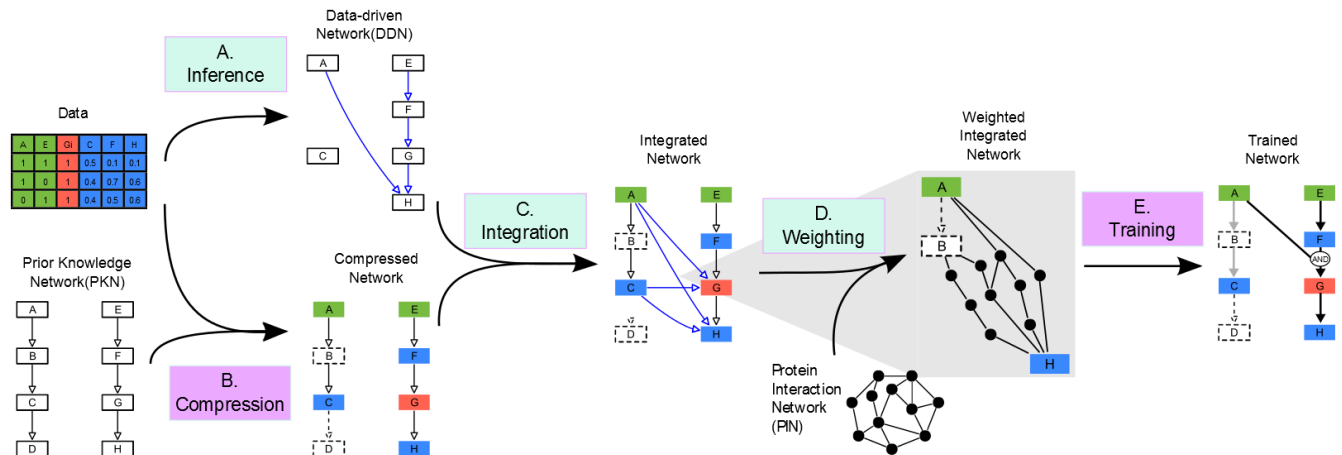


Fig. 1. Integrated pipeline of CNORfeeder (cyan boxes) and CellNOptR (magenta boxes). A. Data are used to infer, using reverse-engineering methods, a strictly data-driven network (DDN); B. the prior knowledge network (PKN) is compressed according to the data (green, red and blue nodes are respectively stimulated, inhibited and measured), removing non-identifiable nodes (dashed ones); C. the compressed network is integrated with the DDN (blue links are obtained from the DDN and black ones from the PKN); D. information derived from protein-protein interaction network (PIN) is used to support and prioritize integrated links; E. The integrated network is used as input for the training: in trained model, thick black lines denote interactions (and gates) in final model, and light-grey links present in integrated network but not in trained model.

With a different and complementary perspective, different ‘reverse engineering’ methods have been used in the literature to infer networks from perturbation experiments in a data-driven way that does not rely on prior knowledge on the network (Bansal, M., et al. 2007, Markowitz, F. 2010). Most of these methods were first developed for transcriptional data but can be applied also to signalling data. For example, in (Ciaccio, M.F., et al. 2010) Bayesian networks (Pe’er, D. 2005) were used to infer the connections between 67 proteins with high-throughput data collected using micro-western array. Two mutual information based approaches, the ‘algorithm for the reconstruction of accurate cellular networks’ (ARACNe) (Margolin, A.A., et al. 2006) and the ‘context likelihood of relatedness’ (CLR) (Faith, J.J., et al. 2007), were also applied to the same dataset to corroborate the results. Different methods were also applied in the context of the DREAM initiative (www.the-dream-project.org) for the DREAM4 Predictive Signalling Network Challenge (Prill, R.J., et al. 2011), where 12 research groups inferred signalling networks from perturbation experiments data and were evaluated based on the accuracy of their predictions of the outcome of the network under different experimental conditions. One of the methods that performed best in this task was a simple approach, strictly data-driven, that encodes in a cause-effect network significant effects of stimuli and inhibitors on measured proteins (Eduati, F., et al. 2010).

These purely data-driven methods need to consider all possible topologies, and thus in general need more data and scale-up worse than methods that rely on a give topology such as CellNOptR. Furthermore, the resulting data-driven networks (that we will call here DDNs) are limited to interactions between perturbed and measured nodes that are only a subset of the nodes involved in the pathways. Thus, DDNs are not as biologically interpretable as the PKNs and mapping DDNs to PKNs is not simple as one link in the inferred network can generally correspond to multiple links in the PKN and it is not trivial how to correctly map it.

In this paper we attempt to combine the strengths of literature-based and data-driven inference methods. We describe a procedure

(implemented in the R package CNORfeeder), to integrate prior knowledge encoded in the PKN with data-driven information obtained using reverse engineering approaches. PINs are used to prioritize links and to provide experimental support for them, and thus help to discriminate among options and add information on integrated links. The resulting network is then trained against experimental data to obtain a final refined model that has a better fit to data with respect to the PKN, highlighting plausible links that were missing in the PKN. We illustrate its application with a signalling network encompassing multiple pathways and readouts trained with data from the liver cancer cell HepG2, showing how CNORfeeder provides a significantly improved fit based on links supported by known interactions among proteins.

2 METHODS

We implemented CNORfeeder, an R package designed to be integrated with methods based on prior-knowledge such as CellNOptR as shown in Figure 1. The integrated pipeline can be summarized in the following steps:

A. Inference (CNORfeeder) A strictly data-driven network (DDN) is inferred from available data using different reverse engineering methods (so far FEED, Bayesian networks, ARACNe, CLR). This network is specific for the experiments under study, thus it only includes perturbed and measured nodes and does not exploit information available in literature.

B. Compression (CellNOptR) The prior knowledge network (PKN) is compressed according to the procedure detailed in (Saez-Rodriguez, J., et al. 2009). First, if a node has no readout downstream of it (such as D in Fig. 1), its state cannot be inferred (is non-observable), and is not considered. Similarly, if a node has no perturbation upstream of it is not included as it will not be affected. Then, nodes that are neither perturbed nor measured are bypassed so that their compression does not change the logic of the remaining nodes (e.g. B which is between A and C in Fig. 1).

C. Integration (CNORfeeder) The compressed network is expanded using the DDN in order to include links that are missing in the a priori information but seems to be supported by data.

D. *Weighting (CNORfeeder)* Protein interaction networks (PINs) are used to support and prioritize the integrated links.

E. *Training (CellNOptR)* The integrated network is finally converted in to a superstructure containing all possible logic gates compatible with the network. If a node (such as G) is affected by multiple ones (A and F), then both an OR and an AND gates are created. Then a genetic algorithm is used to search for the model contained in the superstructure which best describe data (as determined by the mean square error) with the minimum number of links. The objective function is modified with respect to the one introduced in (Saez-Rodriguez, J., et al. 2009) to include additional penalization for the integrated links using weights derived from PINs (Equation 2).

Steps performed by CNORfeeder will be detailed in the following sections.

2.1 Inference using reverse-engineering methods

CNORfeeder can in principle leverage any network inference method. So far, we have integrated the following ones:

FEED inference is the R implementation of an improved version of the algorithm described in (Eduati, F., et al. 2010). The inference of the network can be divided in two steps. First, perturbation experiments are used to infer a Boolean table for each measured protein, codifying if a particular stimulus-inhibitor combination affects the protein. A stimulus or an inhibitor significantly affects an output protein if it is able to modify its activity level

of a quantity that exceeds the uncertainty associated with its measurement. These Boolean tables are then translated into links among stimulated, inhibited and measured nodes giving rise to the inferred network (see Supplementary Material for more details).

Bayesian Network inference There are different approaches to derive causal influences between measured proteins using Bayesian networks. We have used the ‘catnet’ R package (available from <http://cran.r-project.org/web/packages/catnet/index.html>) to derive categorical Bayesian networks from static data (see Supplementary Material for more details).

Mutual information networks This class of methods computes the mutual information matrix between the measurements associated with different proteins and, based on that, infers an undirected network. In particular, ARACNe and CLR algorithms as implemented in the ‘minet’ R package (Meyer, P.E., et al. 2008) (see Supplementary Material for more details), are included in CNORfeeder.

In silico data were generated using a ‘Gold Standard’, or true network, depicted in Figure 2E to compare the four algorithms. The ‘Gold Standard’ was randomly generated and interpreted as a logic Boolean model to simulate perturbation experiments using CellNOptR by stimulating (nodes in green), inhibiting (nodes in red) and measuring (nodes in blue) some proteins. These *in silico* data were then given as input to the inference methods; resulting networks (DDNs) are shown in Figure 2 A-D. The advantage of this approach, with respect to the use of real data, is that the Gold Standard can be used to compare the performances of the different methods.

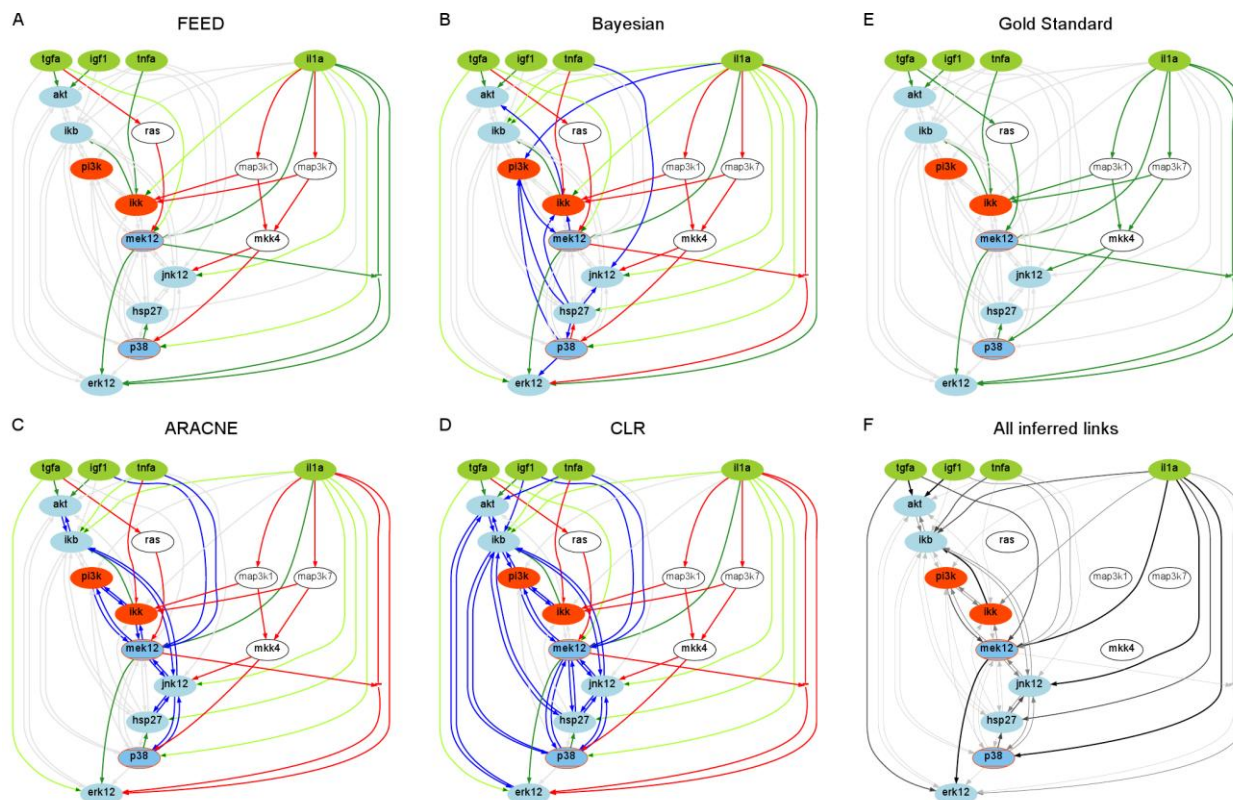


Fig. 2. Reverse engineering of a Gold Standard network (E) using four different inference methods (A-D). Green, red and blue nodes are respectively stimulated, inhibited and measured. The colour of links represent the comparison of the inferred networks (DDNs) with the Gold Standard: dark green for links in both networks, light green for links in the DDN that correspond to a path in the Gold Standard, red for links in the Gold Standard not present in the DDN and blue for links in the DDN that are not in the Gold Standard, grey for links that are not in the network under exam but are in one of the other networks. In panel F the gradation of grey represent the consistency between DDNs in panels A-D.

In Figure 2 A-D dark green links are those that are perfectly reconstructed being present both in the Gold Standard and in the inferred network. Some of the links in the Gold Standard are not inferred by the algorithm (red ones), e.g. $tgfa \rightarrow ras$ or $ras \rightarrow mek12$ for FEED, however in some cases the algorithm is still able to infer at least an indirect link (light green ones), e.g. $tgfa \rightarrow mek12$ for FEED. Blue links are those that are inferred by the algorithm but do not correspond to links in the Gold Standard even as indirect links. In this example, FEED is able to infer all links in the Gold standard, at least as indirect ones, without including any false positive link. It is important to notice that mutual information approaches do not allow for determining the directionality of the links; for light and dark green links the directionality was assessed based on comparison with the Gold Standard to simplify the Figure. A similar approach can be used in real cases by comparison with the PKN, but there is no way to assess the directionality of missing links. In Figure 2 F, links are represented with different gradations of grey according to the consistency between the analyzed inference methods: black links are reconstructed by all methods. As expected, links involving proteins that are neither perturbed nor measured (white nodes) cannot be reconstructed by any inference algorithm. However, those nodes can be important for the signalling network and often there is available literature derived information about their role. This is one of the reasons why it is fundamental to integrate the information derived from data-driven inference methods, with the prior knowledge obtained from other resources.

2.2 Integration with the PKN

Some of the links included in the DDN might be missing in the PKN, and are thus candidates to be integrated in it. However, the PKN generally includes more nodes with respect to the DDN and a link in the DDN could, in some cases, correspond to more than one link in the PKN. As shown in Figure 1 C, if there is a connection between a cue (i.e. a stimulated or an inhibited protein) and a measured protein in the DDN (e.g. from A to H), we have to connect all nodes in the different paths corresponding to that link. This means to add a link not only from the cue to the measured protein, but also from all nodes downstream of the cue, until reaching the following cue, to all nodes upstream of the measured proteins, until reaching the previous measured protein.

2.3 Protein-protein interaction network

The human protein-protein interaction network was built using a unified PPI dataset obtained as APID (Prieto,C. and De Las Rivas,J. 2006), by the combination of interactions coming from six source databases. The starting whole dataset was composed by 68488 human physical protein-protein interactions validated at least by one experimental method and reported in one article published in PubMed. From this dataset we obtained two PPI subsets with increasing confidence: a set of 28971 interactions validated by at least one "binary" experimental method (binary as defined in (De Las Rivas,J. and Fontanillo,C. 2010)); a set 6033 interactions validated by at least two experimental methods, one of them binary.

2.4 Weighting and training of integrated network

The integrated network is then optimized using CellNOptR to find the model which best describes the data using information from PIN to differently prioritize integrated links. As described in (Saez-Rodriguez,J., et al. 2009), a bipartite objective function is used to balance fit and size, thus to find models with good fit to the data but with the minimum number of links. Defining P as a Boolean vector encoding the candidate solution model (value 1 or 0 is assigned depending if the link is included or not in the model), the function that is minimized during the optimization process is the following

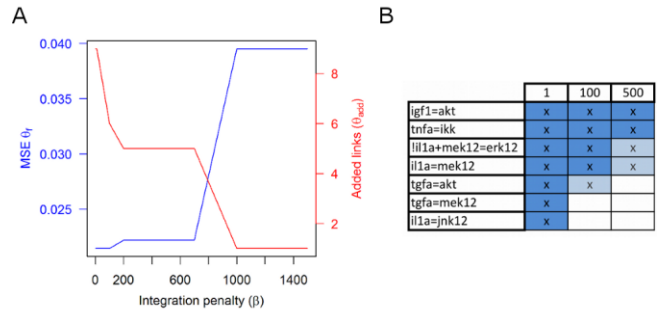


Fig. 3. Panel A: effect of tuning parameter β on the number of integrated links and on the fit (MSE). Panel B: links integrated for different values of β (1, 100, 500); a reduced number of links is selected when using PIN to prioritize links (dark blue).

$$\theta(P) = \theta_f(P) + \alpha \cdot \theta_s(P) \quad (1)$$

where $\theta_f(P) = \frac{1}{N} \sum_{n=1}^N (data_n - pred_n)^2$ is the mean square error (MSE) deviation between the normalized experimental data (continuous values between 0 and 1), and the model prediction (binary values 0 or 1), for all N measured data points. $\theta_s(P) = \sum_{m=1}^M v_m P_m$ is a term to penalize increasing model size according to a tunable parameter α . The size penalty $\theta_s(P)$ is computed as the weighted sum of the M links, which are mathematically hyperedges in the hypergraph that defines the model; see (Saez-Rodriguez,J., et al. 2009) for details. The weight (v) is given by the number of starting nodes, e.g. hyperedge A AND B \rightarrow C is weighted twice A \rightarrow C. In Equation (1) it is possible to include a tunable parameter β to allow a stronger penalization of links integrated to the PKN leading to

$$\theta_s(P) = \theta_{pkn}(P) + \beta \cdot \theta_{add}(P) \quad (2)$$

where the size penalty $\theta_s(P)$ is the sum of two terms: one for the links in the PKN ($\theta_{pkn}(P)$) and one for integrated links ($\theta_{add}(P)$). This is motivated by the fact that, being supported by literature, links in the PKN are more reliable with respect to links integrated using data-driven approaches and they should be prioritized in the training.

Additionally, integrated links can be differently prioritized based on information derived from PINs: the basic idea is that if, for a directed link A \rightarrow B integrated in the PKN, there is a corresponding path in the PIN, it is more plausible that there is a molecular pathway A \rightarrow B. Because shorter paths are more feasible, as a first approximation the shortest path length between A and B in the PIN can be used as a reliability score for the integrated link. Since the optimization is performed on a compressed version of the PKN, one link integrated in the compressed network generally corresponds to multiple possible links integrated in the PKN (Figure 1 E). Thus, the reliability score for each integrated link i is given by $\omega_i = \sum_{j=1}^i 1/d_{ji}$, where $j=1, \dots, J_i$ are the links in the PKN corresponding to the integrated link i in the compressed network. The shortest path d is computed using the Dijkstra's algorithm implemented in the igraph R package (Csardi,G. and Nepusz,T. 2006) considering the PIN as a graph where the weight of the edges is the inverse of the number of experiments (experimental evidences) that validate it.

Thus, the penalty for all A integrated links into the compressed network P, can be defined as

$$\theta_{add}(P) = \sum_{i=1}^A v_i \left(\frac{1}{\omega_i} + 1 \right) P_i \quad (3)$$

The training step, to find the P that minimizes $\theta(P)$ in Equation 1, is performed with CellNOptR using a genetic algorithm that explores the P -

space. The genetic algorithm is run multiple times, and in each run the values for the explored models is recorded, so that at the end a family of models is reported.

3 RESULTS

The method was applied to a dataset of a human liver cell line (HepG2) from DREAM 4 challenge (Prill, R.J., et al. 2011), where the phosphorylation of seven proteins (akt, erk12, ikb, jnk12, p38, hsp27, mek12) is measured 30 minutes after combinatorial stimulation with four ligands (tnfa, il1a, igf1, tgfa) and four inhibitors (pi3k, ikk, p38, mek12). The level of phosphorylation of proteins is measured using Luminex xMAP assay and provides a value of the phosphorylation in arbitrary units, that can be used to compare values at two conditions. In our case we compare the values between 0 and 30, and this change is a proxy of the induced activation of the corresponding protein. The normalization of this data into a value between 0 and 1 is achieved using a method based on a set of thresholds as described in (Saez-Rodriguez et al. 2009). According to the CellNOptR pipeline, the PKN was first compressed removing all non-observable and non-controllable nodes and then expanded as described in (Saez-Rodriguez, J., et al. 2009) to include all possible combinations of AND and OR gates compatible with the network obtaining a total of 62 hyperedges. Additionally, 18 links inferred using FEED were integrated in the network according to the procedure previously described and the integrated network was used for optimization using CellNOptR.

Fixing $\alpha=0.001$, the influence of the integration penalty (β) on the number of integrated links selected by the optimization process on the fit of the optimal model to the data (in terms of MSE) was tested as shown in Figure 3. As expected, a low value of β permits to obtain the best fit but at the price of a high number of integrated

links included in the optimal model (9 with $\beta=1$). An increase of the value of β decreases the number of selected integrated links but worsen the fit to the data. With $\beta=1000$ only the integrated link $tnfa \rightarrow ikk$ is included in the optimal model: the presence of this link is well supported by the data since it permits to lower the MSE from 0.064 (the optimal fit obtained with CellNOptR using as input the non-integrated network) to 0.040. The integrated links can be ranked as shown in Figure 3 B according to the highest value of β allowing their selection and thus according to their effect on the improvement of the fit. A lower number of links is selected when using the PIN to additionally penalize unsupported links (highlighted in dark blue in Figure 3B). Those links, combined with the information from the PIN, suggest possible missing connections in the PKN. For example, in the PIN there is an interaction between the adaptor *irs1* and the kinase *pdk* that would justify the link $igf1 \rightarrow akt$ in the compressed network since in the PKN (Fig S2 in Supplementary Material) *igf1* binds to its receptor and *pdk* regulates *akt* (links $igf1 \rightarrow igfr$ and $pdk1 \rightarrow akt$ in Fig. S2; note that in Fig. 4 the compressed networks are shown and thus intermediates *igfr*, *irs1s* and *pdk1* are not present). Therefore the path $igf1 \rightarrow igfr \rightarrow irs1s \rightarrow pdk1 \rightarrow akt$ is supported by a combination of literature and interaction data. Similarly, to support the link $tnfa \rightarrow ikk$ there is a validated interaction between the *tnfa* receptor and *cot*, a protein that activate *ikk*, leading to the combined pathway $tnf \rightarrow tnfr \rightarrow cot \rightarrow ikk$.

In Figure 4 A and B the results of CellNOptR optimization (with $\beta=700$) are shown using as input the compressed network and the integrated network respectively. In the upper panels, optimal models are shown: links selected by the optimization algorithm are represented in green if derived from the PKN and blue if integrated using FEED. In the lower panel the improvement in the fit is shown (from 0.064 to 0.022), which is particularly large for pro-

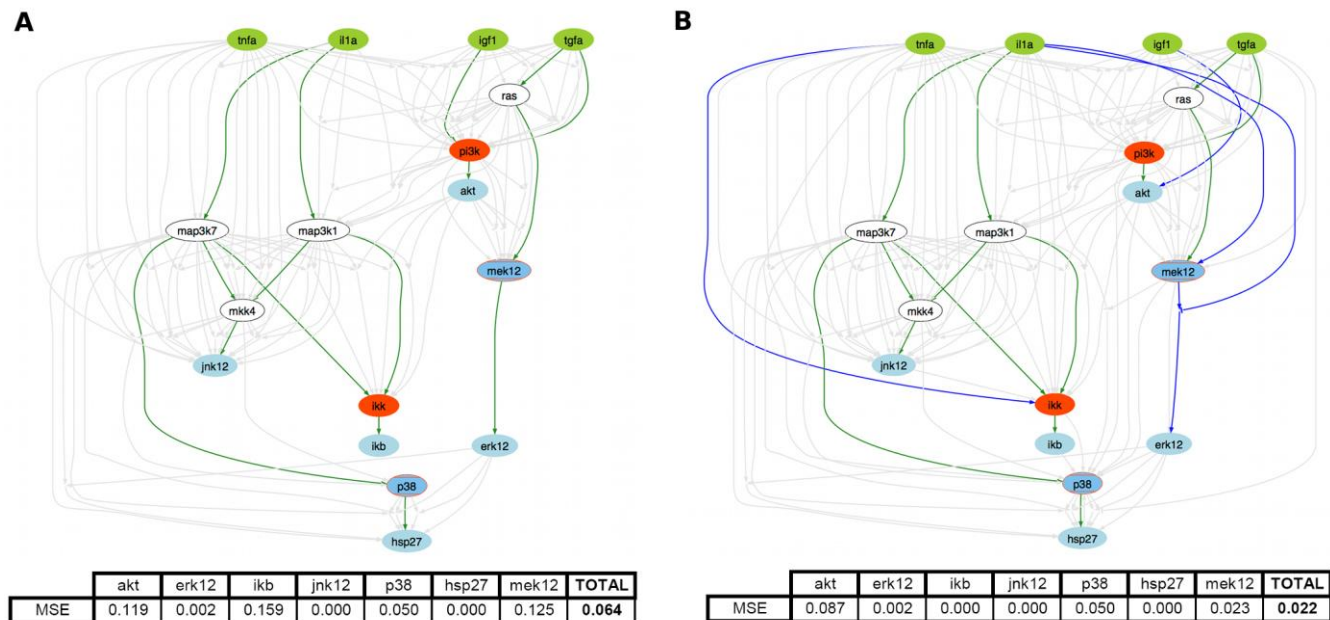


Fig. 4. Results of the training of the compressed model (A) and of the integrated network (B) against data using CellNOptR. Green, red and blue nodes are respectively stimulated, inhibited and measured. Selected links are represented in green if derived from the PKN and blue if integrated, links not selected are in grey. In the tables the fit (in terms of MSE) is reported for each measured protein along with the sum for all proteins.

teins *ikb*, *mek12* and *akt*. In this case study, using the same parameter setting ($\alpha=0.001$, $\beta=700$), networks integrated using ARACNe and CLR do not provide an improvement of the fit, while Bayesian networks permits to obtain a MSE of 0.040 (see Supplementary Information). As for the computational times, FEED, CLR, and ARACNe inferred the network in ~ 1 second while Bayesian inference took ~ 1 hour on a cluster.

To evaluate the scalability of our method, we applied CNORfeeder to a larger dataset obtained also in the cell lines HepG2, comprising 7 stimuli, 7 inhibitors and 15 readouts (Saez-Rodriguez, J., et al. 2009) obtaining comparably good results (see Supplementary Information).

Furthermore, we investigated the ability of our method to capture

feedback loops, which are fundamental in the regulation of signal transduction. We constructed a toy model containing a negative feedback loop and simulated data at 2 different time points (10 and 30 minutes). We used FEED as reverse-engineering method to retrieve, from the data, a link of the feedback that was missing in the PKN and then applied a recently implemented package of CellNOptR (www.cellnopt.org) that, looking also at the second time point, was able to select all links of the feedback loop (see Supplementary Information for further details).

4 DISCUSSION

In this paper we present an approach that integrates literature-constrained and data-driven methods to efficiently infer signalling networks from experimental data collected under perturbation experiments with different stimuli and inhibitors. The procedure is implemented in the R package CNORfeeder and consists of (i) inference of a data-derived network (DDN) using strictly data-driven reverse-engineering methods (so far FEED, Bayesian networks, and mutual information approaches), (ii) integration of the DDN with a literature-derived prior knowledge network (PKN), using protein interaction networks (PINs) to prioritize and validate integrated links, and (iii) training of the integrated network against data using CellNOptR to obtain a logic model that best describe the data with the minimum number of links.

Links that improve the fit to data with respect to the PKN alone may be missing due to the difficulty assembling all available pathway information or because of incomplete knowledge of the biology. Protein interaction networks (PINs) are used as a complementary source of information to tackle this problem. PINs contain physical interactions between proteins, including potentially those that lead to protein activations, and they typically include more nodes and many more links than those based on literature-derived pathways. For this reason they have been proposed to extend pathways (Glaab, E., et al. 2010) but they have the main limitation of a lack of directionality. PINs are also known to have high false positive and false negative rates, and we therefore used a very curated PIN that integrates different sources and experimental techniques. This PIN seems to be quite complete for the pathways we studied (canonical pro-growth and inflammatory pathways) since we verified that for links in the PKN there is generally also a direct connection in the PIN (Figure 5). Interestingly, mapping to the PIN the links integrated to the PKN we found a corresponding short path that do not pass through other nodes of the PKN. To limit the effect of false positive links in the PIN when searching for the shortest path, we weighted the edges according to the number of experimental evidences that support them. The length of shortest path is then used to differently prioritize the integrated links in the training of the network, but other metrics could be used to discriminate between links. PINs were previously shown to be potentially useful to find previously unknown modulator of signalling pathway in (Vinayagam, A., et al. 2011), where a Bayesian learning strategy was applied to assign directionality to a comprehensive PIN exploiting information on the shortest path from membrane receptors to transcription factors. In our method, we can take ad-

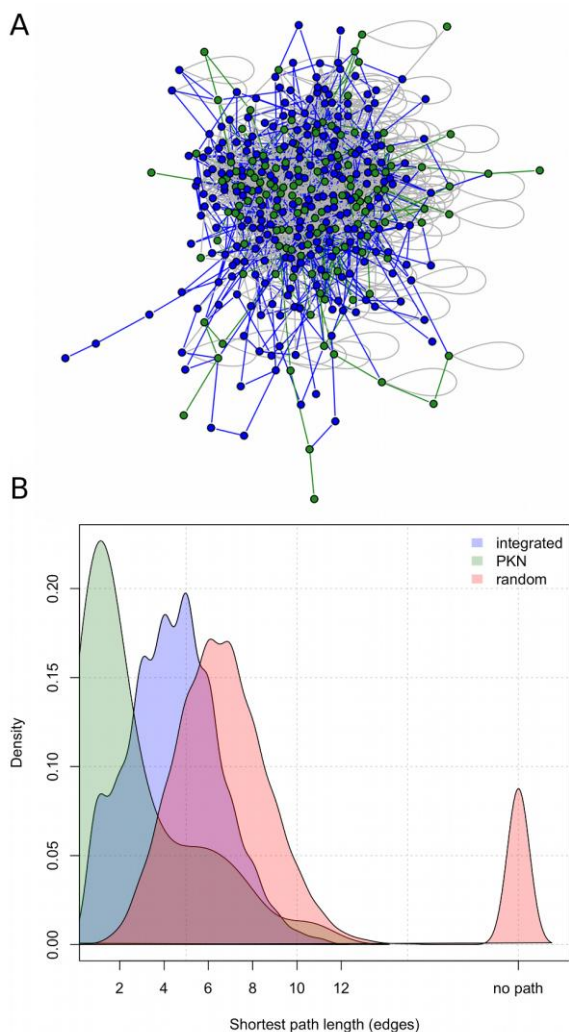


Fig. 5. Mapping of the prior knowledge network (PKN) to the protein-protein interaction network (PIN). Panel A represents the subgraph of the PIN that include only nodes belonging to the PKN (green) and nodes used in the mapping of integrated links (blue); network was plotted with R package igraph. The same colour code is used for the edges: as expected, shortest paths between nodes in the PKN (green) are generally shorter than paths used to map integrated links (blue). This is highlighted also in panel B where the density of the shortest path length (in terms of number of edges) is plotted for integrated links (blue), for links in the PKN (green) and for random links (red).

vantage of the directed links inferred via reverse engineering to limit the paths present in the PIN we integrate, limiting the search space for the optimization algorithm.

We have used different data-driven inference methods, and applied them to both *in silico* (to reverse-engineer a benchmark network with known topology) and real data (to integrate links missing in the PKN that improve the fit of the model to the data in the liver cancer cell line HepG2). Each reverse-engineering method has specific features and can be suitable for different needs: for example Bayesian networks can provide statistically rigorous results but at the price of high computational costs, while mutual information approaches are computationally fast but are limited mostly by the lack of directionality of the inferred links. FEED seems to be particularly suitable to infer causal networks from single-stimulus/single-inhibitor experiments with low computational costs but, as far as we know, does not permit to exploit data from all multiple combinatorial perturbation experiments.

It is not the purpose of this study to compare reverse-engineering methods (which would require a larger set of benchmark networks with known topology and a more realistic simulation of experimental data). The spirit of the paper is more in line with the lesson derived from the DREAM challenges (Marbach, D., et al. 2010, Prill, R.J., et al. 2011) that different approaches can provide complementary insights into the same problem, and we have thus employed various approaches and we plan to extend it to others in the future. Furthermore, some reverse-engineering methods can use prior knowledge, in particular Bayesian inference methods (Bender, C., et al. 2011, Mukherjee, S. and Speed, T.P. 2008), so that we could use the PKN or results from the training with CellNOptR to guide a further search for novel links.

To conclude, the integration of literature-constrained and data-driven inference methods allows to overcome the limitations of both: for purely data-driven inference methods, the poor scalability (as the search space increase exponentially) and limited biological interpretability (since they are limited to measured and perturbed proteins excluding intermediate ones), and for methods constrained to prior knowledge their inability to overcome incompleteness in the networks. We propose here an approach (and software package) to combine them that is effective and extensible to include other methods.

ACKNOWLEDGEMENTS

The authors thank T. Cokelaer for help coding the R package, A. MacNamara for critical reading of the manuscript and assistance with the figures, and L. Alexopoulos for useful discussions.

Funding: JSR thanks funding from EU-7FP-BioPreDyn, JdIR from EU FP7-HEALTH-2007-B (ref. 223411), Spanish ISCiii (ref. PS09/00843), and Junta Castilla y Leon (ref. CSI07A09). FE was partially supported by the “Borsa Gini” scholarship, awarded by “Fondazione Aldo Gini”, Padova, Italy.

REFERENCES

Aldridge, B.B., et al. (2006) Physicochemical modelling of cell signalling pathways. *Nat. Cell Biol.*, 8, 1195-1203.

- Bansal, M., et al. (2007) How to infer gene networks from expression profiles. *Mol. Syst. Biol.*, 3, 78.
- Bender, C., et al. (2011) Inferring signalling networks from longitudinal data using sampling based approaches in the R-package ‘ddepn’. *BMC Bioinformatics*, 12, 291.
- Calzone, L., et al. (2010) Mathematical modelling of cell-fate decision in response to death receptor engagement. *PLoS Comput. Biol.*, 6, e1000702.
- Cerami, E.G., et al. (2011) Pathway Commons, a web resource for biological pathway data. *Nucleic Acids Res.*, 39, D685-90.
- Ciaccio, M.F., et al. (2010) Systems analysis of EGF receptor signaling dynamics with microwestern arrays. *Nat. Methods*, 7, 148-155.
- Csardi, G. and Nepusz, T. (2006) The igraph software package for complex network research. *Int. J. Complex Syst.*, 1695.
- De Las Rivas, J. and Fontanillo, C. (2010) Protein-protein interactions essentials: key concepts to building and analyzing interactome networks. *PLoS Comput. Biol.*, 6, e1000807.
- Eduati, F., et al. (2010) A Boolean approach to linear prediction for signaling network modeling. *PLoS One*, 5, e12789.
- Faith, J.J., et al. (2007) Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles. *PLoS Biol.*, 5, e8.
- Glaab, E., et al. (2010) Extending pathways and processes using molecular interaction networks to analyse cancer genome data. *BMC Bioinformatics*, 11, 597.
- Ideker, T. and Sharan, R. (2008) Protein networks in disease. *Genome Res.*, 18, 644-652.
- Joshi-Tope, G., et al. (2005) Reactome: a knowledgebase of biological pathways. *Nucleic Acids Res.*, 33, D428-32.
- Ma’ayan, A., et al. (2005) Formation of regulatory patterns during signal propagation in a Mammalian cellular network. *Science*, 309, 1078-1083.
- Marbach, D., et al. (2010) Revealing strengths and weaknesses of methods for gene network inference. *Proc. Natl. Acad. Sci. U.S.A.*, 107, 6286-6291.
- Margolin, A.A., et al. (2006) ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, 7 Suppl 1, S7.
- Markowitz, F. (2010) How to understand the cell by breaking it: network analysis of gene perturbation screens. *PLoS Comput. Biol.*, 6, e1000655.
- Meyer, P.E., et al. (2008) minet: A R/Bioconductor package for inferring large transcriptional networks using mutual information. *BMC Bioinformatics*, 9, 461.
- Morris, M.K., et al. (2010) Logic-based models for the analysis of cell signaling networks. *Biochemistry*, 49, 3216-3224.
- Mukherjee, S. and Speed, T.P. (2008) Network inference using informative priors. *Proc. Natl. Acad. Sci. U.S.A.*, 105, 14313-14318.
- Ogata, H., et al. (1999) KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res.*, 27, 29-34.
- Pe’er, D. (2005) Bayesian network analysis of signaling networks: a primer. *Sci. STKE*, 2005, pl4.
- Pico, A.R., et al. (2008) WikiPathways: pathway editing for the people. *PLoS Biol.*, 6, e184.
- Pieroni, E., et al. (2008) Protein networking: insights into global functional organization of proteomes. *Proteomics*, 8, 799-816.
- Prieto, C. and De Las Rivas, J. (2006) APID: Agile Protein Interaction Data Analyzer. *Nucleic Acids Res.*, 34, W298-302.
- Prill, R.J., et al. (2011) Crowdsourcing network inference: the DREAM predictive signaling network challenge. *Sci. Signal.*, 4, m7.
- Saadatpour, A., et al. (2011) Dynamical and structural analysis of a T cell survival network identifies novel candidate therapeutic targets for large granular lymphocyte leukemia. *PLoS Comput. Biol.*, 7, e1002267.
- Saez-Rodriguez, J., et al. (2009) Discrete logic modelling as a means to link protein signalling networks with functional analysis of mammalian signal transduction. *Mol. Syst. Biol.*, 5, 331.
- Samaga, R., et al. (2009) The logic of EGFR/Erbb signaling: theoretical properties and analysis of high-throughput data. *PLoS Comput. Biol.*, 5, e1000438.
- Terfve, C. and Saez-Rodriguez, J. (2012) Modeling signaling networks using high-throughput phospho-proteomics. *Adv. Exp. Med. Biol.*, 736, 19-57.
- Vinayagam, A., et al. (2011) A directed protein interaction network for investigating intracellular signal transduction. *Sci. Signal.*, 4, rs8.
- Watterson, S., et al. (2008) Logic models of pathway biology. *Drug Discov. Today*, 13, 447-456.