

Fabian Schmich^{1,2} / Jack Kuipers^{1,2} / Gunter Merdes¹ / Niko Beerenwinkel^{1,2}

netprioR: a probabilistic model for integrative hit prioritisation of genetic screens

¹ Department of Biosystems Science and Engineering, ETH Zurich, Mattenstrasse 26, 4058 Basel, Switzerland, E-mail: niko.beerenwinkel@bsse.ethz.ch. <https://orcid.org/0000-0002-0573-6119>.

² SIB Swiss Institute of Bioinformatics, Mattenstrasse 26, 4058 Basel, Switzerland, E-mail: niko.beerenwinkel@bsse.ethz.ch. <https://orcid.org/0000-0002-0573-6119>.

Abstract:

In the post-genomic era of big data in biology, computational approaches to integrate multiple heterogeneous data sets become increasingly important. Despite the availability of large amounts of omics data, the prioritisation of genes relevant for a specific functional pathway based on genetic screening experiments, remains a challenging task. Here, we introduce *netprioR*, a probabilistic generative model for semi-supervised integrative prioritisation of hit genes. The model integrates multiple network data sets representing gene–gene similarities and prior knowledge about gene functions from the literature with gene-based covariates, such as phenotypes measured in genetic perturbation screens, for example, by RNA interference or CRISPR/Cas9. We evaluate *netprioR* on simulated data and show that the model outperforms current state-of-the-art methods in many scenarios and is on par otherwise. In an application to real biological data, we integrate 22 network data sets, 1784 prior knowledge class labels and 3840 RNA interference phenotypes in order to prioritise novel regulators of Notch signalling in *Drosophila melanogaster*. The biological relevance of our predictions is evaluated using *in silico* and *in vivo* experiments. An efficient implementation of *netprioR* is available as an R package at <http://bioconductor.org/packages/netprioR>.

Keywords: EM, expectation maximization, genetic, GMRF, hit prioritization, model, probabilistic, screen

DOI: 10.1515/sagmb-2018-0033

1 Introduction

Identifying the set of genes, or their protein products, that operate together in order to execute a certain function within the cell or that are relevant for a specific disease has been a challenging task in the post-genomic era for molecular and computational biologists alike. In particular, the prioritisation of prospective candidate genes, so called *hits*, from preliminary genetic screening experiments for follow up analyses is critical (Moreau and Tranchevent 2012). With the development and widespread application of high-throughput techniques, such as Yeast two-hybrid (Giot et al., 2003; Formstecher et al., 2005) and co-affinity purification (co-AP)/MS (Friedman et al., 2011; Guruharsha et al., 2011) screens for protein–protein interactions, gene expression profiling (Chintapalli, Wang & Dow, 2007; Horan et al., 2008; Graveley et al., 2011), or genetic interaction (Costanzo et al. 2010) screening, we have seen a steady increase of publicly available genome-wide interaction data sets. These data sets are often represented as functional linkage networks, where nodes represent genes and weighted edges represent the degree of evidence for co-functionality (Mostafavi et al. 2008).

Semi-supervised graph-based learning for gene prioritisation. Over the last decade, the availability of vast amounts of network data organised in organism-centric databases (Yu et al. 2008) fuelled the development of new integrative approaches and the adaptation of numerous algorithms from graph theory for network-based gene function prediction (Tsuda, Shin & Schölkopf, 2005; Aerts et al., 2006; Mostafavi et al., 2008; Chen et al., 2009; Kato, Kashima & Sugiyama, 2009). The common underlying principle of these methods is the consistency assumption (Zhu, Ghahramani, and Lafferty 2003), i.e. similar genes are likely to have a similar function. This principle has also been termed guilt by association (Mostafavi et al. 2008). It allows the use of interactions between genes to predict functions for uncharacterised genes by associating them with genes of known function. Guilt-by-association approaches typically take a set of seed genes with known function distributed across the network and score uncharacterised genes according to their proximity to the seed genes. It has been shown that approaches integrating multiple network data sets outperform predictions based on single data sets (Tsuda, Shin & Schölkopf, 2005; Mostafavi et al., 2008), likely because of their noisy, incomplete, and in part complementary nature.

Niko Beerenwinkel is the corresponding author.

©2019 Walter de Gruyter GmbH, Berlin/Boston.

Current state-of-the-art approaches, such as *TSS* (Tsuda, Shin, and Schölkopf 2005) and *GeneMANIA* (Mostafavi et al. 2008) integrate two types of input data: functional linkage networks and prior knowledge class labels for seed genes. *GeneMANIA* is based on the seminal work of Zhu et al. (Zhu, Ghahramani, and Lafferty 2003) and was developed with a focus on fast predictions. Suitable to be used as an online server, *GeneMANIA* weighs multiple network data sets prior to the learning task using a regularised regression model inspired by kernel-target alignment (Cristianini et al. 2002), sacrificing maximum accuracy for speed. *TSS*, in contrast, estimates network weights and prioritisation labels simultaneously solving a convex optimisation problem. However, both approaches do not allow for the integration of additional gene-based features not easily transformed into gene–gene similarities or class labels, such as, for instance, perturbation screen phenotypes. To the best of our knowledge, the first guilt-by-association method that implements this feature is *LMgraph* (Vembu and Morris 2015), a recent extension to *GeneMANIA*, which combines functional linkage networks and gene-based features using a weighting scheme in conjunction with linear classifiers.

Integrative prioritisation of perturbation screen hits. Perturbation experiments have become a common approach to screen in a genome-wide fashion for candidate genes involved in a certain biological function. Scalable screening technologies include gene knock-downs using RNA interference (RNAi) and, more recently, gene knock-outs using CRISPR/Cas9. Typical workflows for the analysis of screening data have been shown to produce high numbers of false positive and false negative candidate genes when prioritising for follow-up analyses. For RNAi screens, this limitation is often attributed to sequence-based siRNA off-target effects (Rämö et al., 2014; Schmich et al., 2015). In organisms like *Drosophila melanogaster*, however, where this problem is not likely to be caused by miRNA-like siRNA off-target effects, it has been proposed to integrate network data and prior knowledge about the biological systems for improved hit prioritisation of the screen (Wang, Tu, and Sun 2009).

Following this paradigm we developed *netprioR*, a probabilistic graphical model inspired by work from Kato and co-workers (Kato, Kashima, and Sugiyama 2009) that integrates multiple network data sets, prior knowledge on gene functions, and phenotype data or any other type of additional covariates (Figure 1). The output of *netprioR* is a list of prioritised hit genes ranked by predicted labels, as well as estimated weights for the integrated networks reflecting the importance of each data set. We demonstrate on simulated data that prioritisations from *netprioR* outperform current state-of-the-art methods with respect to different metrics. In addition, we prioritise novel regulators of Notch signalling in *Drosophila melanogaster*, integrating 22 network data sets, prior knowledge labels for true positive and true negative Notch regulators from the literature, as well as perturbation screen phenotypes from a recent RNAi screen (Saj et al. 2010).

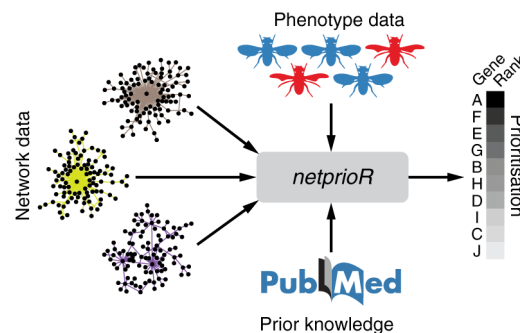


Figure 1: Integrative gene prioritisation of hits in genetic perturbation screens. The *netprioR* model integrates network data, prior knowledge in the form of true positive and true negative hits and phenotype data for robust prioritisation.

2 Results

We present the probabilistic graphical model of *netprioR*, provide a comparative evaluation of its performance on simulated data and, integrating multiple data sets for *Drosophila melanogaster*, we prioritise novel regulators of Notch signalling.

2.1 The *netprioR* model for integrative hit prioritisation

Let N be the number of genes. Let $\mathbf{Y} = [Y_1, \dots, Y_N]$ be the vector of continuous gene labels and $\mathbf{X}$ the $N \times P$ matrix of P covariates, e.g. a P -dimensional phenotype measurement. We represent the $N \times N$ gene–gene

similarity networks based on K different data sources as graph Laplacians $\mathbf{G} = (\mathbf{G}_1, \dots, \mathbf{G}_K)$. The distribution of labels is modelled hierarchically as

$$W_k | a, b \sim \text{Ga}(a, b) \quad (1)$$

$$\mathbf{R} | \mathbf{W}, \mathbf{G} \sim \text{Norm}\left(\mathbf{0}, \left(\sum_{k=1}^K W_k \mathbf{G}_k + \epsilon \mathbf{I}\right)^{-1}\right) \quad (2)$$

$$\boldsymbol{\beta} | \tau \sim \text{Norm}(\mathbf{0}, \tau \mathbf{I}) \quad (3)$$

$$\mathbf{Y} | \mathbf{R}, \mathbf{X}, \boldsymbol{\beta}, \sigma \sim \text{Norm}(\mathbf{X}\boldsymbol{\beta} + \mathbf{R}, \sigma \mathbf{I}) \quad (4)$$

The random effects $\mathbf{R}$ are modelled by a Gaussian Markov Random Field (GMRF) (Rue and Held 2005) with $N \times N$ precision matrix $\mathbf{Q} = \sum_{k=1}^K W_k \mathbf{G}_k + \epsilon \mathbf{I}$. The addition of ϵ to the diagonal is required to ensure that $\mathbf{Q}$ is always invertible. We define the following two covariance matrices to simplify the notation: $\mathbf{T} = \tau \mathbf{I}$ and $\mathbf{S} = \sigma \mathbf{I}$. W_k can be interpreted as the weight or importance of data source k , whereas $\boldsymbol{\beta}$ is the vector of fixed effects for the P covariates. Hyper-parameters $\sigma = 0.1$, $\tau = 100$ and $a = b = 0.01$ remain fixed throughout this study. They result in a flat prior on $\boldsymbol{\beta}$ and a prior for W_k that shrinks weights to zero. We characterise the Gamma distribution $\text{Ga}(W_k; a, b)$ in terms of shape a and rate b with the resulting probability density function (PDF)

$$\Pr(W_k | a, b) = \frac{b^a}{\Gamma(a)} W_k^{(a-1)} \exp(-bW_k).$$

Without loss of generality, we separate $\mathbf{Y}$ into the vector of labelled genes $\mathbf{Y}_L = [Y_1, \dots, Y_L]$ and unlabelled genes $\mathbf{Y}_U = [Y_{L+1}, \dots, Y_N]$. The structure of the complete probabilistic graphical model is shown in Figure 2. In order to estimate the parameters $\boldsymbol{\theta} = (\mathbf{W}, \boldsymbol{\beta}) = (W_1, \dots, W_K, \beta_1, \dots, \beta_P)$ and predict the missing gene labels $\mathbf{Y}_U$ for prioritisation, we developed an Expectation Maximisation (EM) algorithm (see Section 4). The pseudocode of the EM algorithm, including the update rules for the E-step (Eqs. (12), (13) and (14)) and M-step (Eqs. (5) and (6)), is provided in Algorithm 1.

Algorithm 1

Expectation Maximisation (EM) algorithm for *netprioR*. Update rules for the E-step and M-step are derived in section 4.1. $l(\cdot)$ is the likelihood function.

Input: N, G, Y_L, X

Output: Parameter estimates $\hat{\theta}$ and predicted labels $\hat{Y}_U$

Procedure:

Initialise parameters $\theta^{(0)} = \{W^{(0)}, \beta^{(0)}\}$ drawing from prior

$$H_L = Y_L - X_L \beta^{(0)}$$

$t = 1$

while not converged **do**

$$Q^{(t)} = \sum_{k=1}^K W_k^{(t-1)} G_k + \epsilon I$$

E step (solve with conjugate gradient method):

$$\mathbb{E}[R_L | H_L]^{(t)} = (Q_{LL}^{(t)} + S_{LL}^{-1})^{-1} Q_{LL}^{(t)} H_L$$

$$\mathbb{E}[R_U | H_L]^{(t)} = - (Q_{UU}^{(t)})^{-1} Q_{UL}^{(t)} (Q_{LL}^{(t)} + S_{LL}^{-1})^{-1} Q_{LL}^{(t)} H_L$$

$$\mathbb{E}[Y_U | H_L]^{(t)} = - (Q_{UU}^{(t)})^{-1} Q_{UL}^{(t)} (Q_{LL}^{(t)} + S_{LL}^{-1})^{-1} Q_{LL}^{(t)} H_L + X \beta^{(t-1)}$$

M step:

$$\beta^{(t)} = (X_L^\top S_{LL}^{-1} X_L + T^{-1})^{-1} X_L^\top S_{LL}^{-1} (Y_L - \mathbb{E}[R_L]^{(t)})$$

for $k \in \{1, \dots, K\}$ **do**

$$W_k^{(t)} = \frac{a + \frac{N}{2} - 1}{b + \frac{1}{2} (\mathbb{E}[R]^{(t)})^\top G_k \mathbb{E}[R]^{(t)}}$$

end

if $\|l(\theta^{(t)}) - l(\theta^{(t-1)})\| < 10^{-6}$ **then**

 converged = TRUE

else

$t = t + 1$

end

end

$$\hat{\theta} = \{W^{(t)}, \beta^{(t)}\}$$

$$\hat{Y}_U = \mathbb{E}[Y_U | H_L]^{(t)}$$

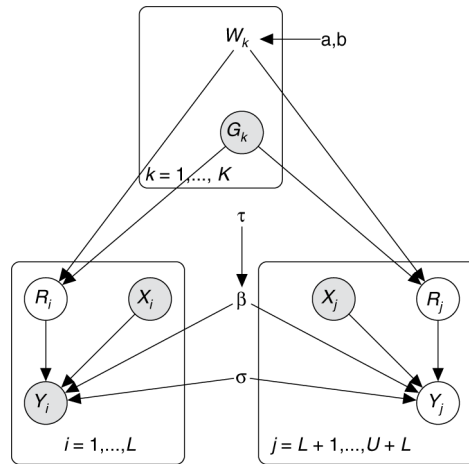


Figure 2: Graphical representation of the *netprioR* model. Shaded and non-shaded discs represent observed and latent random variables, respectively. Arrows depict conditional dependencies between random variables and parameters. We used plate layouts to distinguish between random variables for labelled data Y_L (left) and unlabelled data Y_U (right), as well as k different similarity networks G_k and corresponding weight parameters W_k . Each plate represents multiple nodes of which only a single example is shown.

2.2 Comparative performance evaluation on simulated data

Data simulation. In order to evaluate the performance of *netprioR*, we generated simulated data with known ground truth labels and phenotypes for each gene, as well as gene–gene networks according to the schema depicted in Figure 3. We simulated 1000 genes and split them into equally sized classes with labels *hit* and *non-hit* (Figure 3B). We simulated two kinds of gene–gene networks: For low-noise networks, 80% of all interactions, i.e. gene–gene similarities, lie within the same class and are simulated as a scale-free network with preferential

attachment proportional to node degrees, whereas for high-noise networks, interactions do not obey the class structure, such that the model is provided with networks of varying degrees of information content during hit prioritisation (Figure 3A). Univariate phenotypes for hits and non-hits were sampled from $\text{Norm}(\mu_{\text{pos}}, 1)$ and $\text{Norm}(\mu_{\text{neg}}, 1)$, respectively, with varying effect size $|\mu_{\text{neg}} - \mu_{\text{pos}}|$ (Figure 3C). In each iteration of the simulation study, we constructed (1) 1%, 2%, 5%, and 10% of the labels, (2) two low-noise and 0–3 high-noise gene–gene networks, and (3) phenotype effect sizes of 0, 0.25, 0.5, and 1.0. In total, we simulated 50 data sets for each of the $4 \times 4 \times 4 = 64$ parameter combinations.

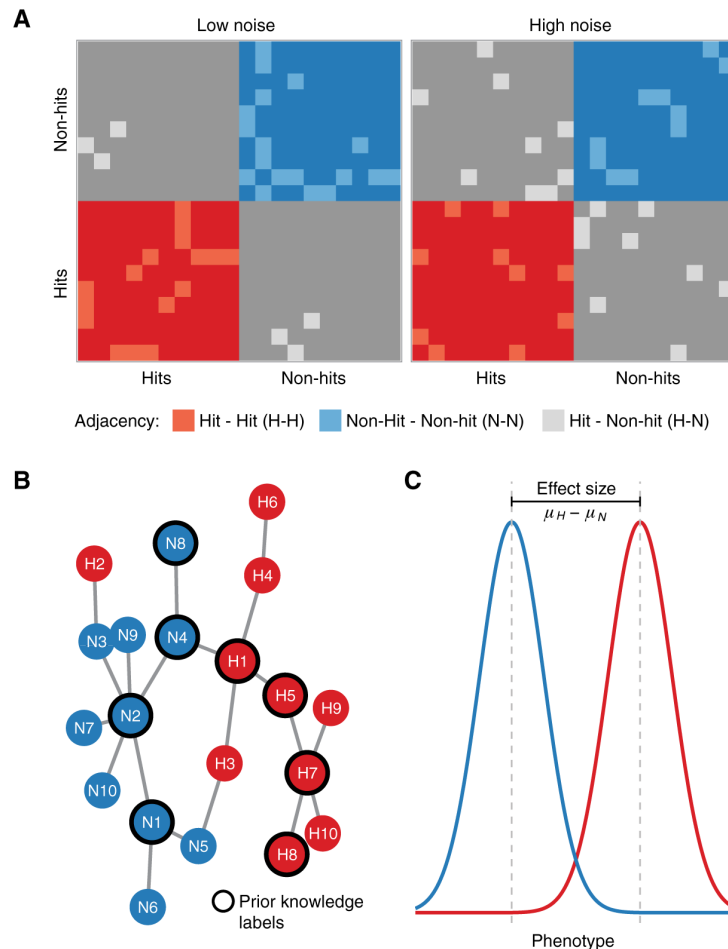


Figure 3: Schematic design of the simulation study for the prioritisation problem of 500 non-hits (N, blue) and 500 hits (H, red). (A) Simulation of two low-noise and varying amounts (0–3) of high-noise networks with identical average vertex degree (depicted as adjacency matrices). Low-noise networks (left) are scale-free and strongly obey the class structure with 80% intra-class interactions, whereas high-noise networks (right) do not obey class structure. (B) Varying amounts of a priori known class labels were sampled (1%, 2%, 5%, 10%, depicted as black circle). (C) Phenotypes were sampled from $\text{Norm}(\mu_{\text{neg}}, 1)$ and $\text{Norm}(\mu_{\text{pos}}, 1)$ with varying effect size $|\mu_N - \mu_P| \in \{0, 0.25, 0.5, 1\}$.

Benchmark methods. The performance of *netprioR* was compared to the competing methods *LMgraph* (Vembu and Morris 2015; web <http://droidb.org>) and *TSS* (Tsuda, Shin, and Schölkopf 2005; web <https://github.com/morrislab/lmgraph>), as well as the baseline of prioritising hits solely based on the phenotypic measurements (termed *phenotype-only*). We used default parameters for *TSS* ($c = 1, c_0 = 0.4, \text{const} = 0.7$) and *LMgraph* ($\text{MIN_LBLs} = 30, \text{NTRIALS} = 5$). Since *TSS* was not designed to incorporate additional covariates, we transformed simulated phenotypes with a z-score >2 and <-2 into additional hit labels for fair comparison.

Evaluation of prioritisation performance. Performance was evaluated based on how well each model could correctly separate unlabelled genes into hits and non-hits, measured by the area under the receiver-operator characteristic curve (AUC). We focused on simulated data with 10% of class labels available, as this resembled the setting we found in real data from *Drosophila melanogaster* (see Section 2.3), and present the results of the comparative analysis in Figure 4. We observed *netprioR* to outperform or be on par with both competing models, as well as the baseline of *phenotype-only* prioritisation in the classification task of separating hit from non-hit labelled genes (Figure 4A). As expected, all methods showed an increase in performance with increasing phenotypic effect size and fewer high-noise networks. However, in contrast to *netprioR*, *TSS* fell below

the *phenotype-only* baseline when a single high-noise similarity network was added and the phenotype effect size was 1. This effect was even observable for smaller effect sizes, when we increased the number of high-noise networks. The performance of *netprioR* never dropped below that of the baseline phenotype-only method, indicating that our integrative approach is favourable even if only noisy network data is available. This observation was confirmed in a larger simulation with 20 networks and 10% labelled data (Supplementary Figure 1), more closely matching our real-world application in *Drosophila melanogaster*. The performance evaluation for 1%, 2%, and 5% of labelled data available for learning was similar, except for the case with 1% labelled data and no high-noise networks, where all methods performed equally well (Supplementary Figure 2). We also investigated the distributions of estimates for the phenotype fixed effect β and found that, as expected, for increasing effect size *netprioR* inferred increasing values of β (Supplementary Figure 3).

Evaluation of inferred network weights. We compared the inferred weights for low-noise and high-noise data sources for *netprioR*, *TSS*, and *LMgraph* (Figure 4B). *netprioR* put most of the probability mass on low-noise networks, whereas *TSS* and *LMgraph* weighed high- and low-noise networks very similarly. This observation may explain the more steeply derogating performance of *TSS* for increasing number of high-noise networks, compared to *netprioR* and emphasizes the strength of *netprioR* to distinguish high-noise from low-noise network data sources for a specific prioritisation task.

Evaluation of model robustness. Both the estimated prioritisation rankings of unobserved genes and the network weights are very robust with respect to restarts of the EM algorithm. In order to demonstrate this, we performed 100 restarts on the same input data where we each time sampled initial parameters from the prior and fitted the *netprioR* model. Then, we investigated both the inferred network weights, as well as the inferred labels Y' and observed that the coefficient of variation (CV) in both predicted variables is very low (network weights: CV < 0.01, Y' : median CV = 0.11) as illustrated in Supplementary Figure 5. We also investigated *netprioR*'s robustness with respect to the hyper parameters τ , σ , a and b and found high median concordance (>0.9) between prioritisation rankings across all hyper parameters indicating *netprioR*'s ability to recover the same ranking consistently (Supplementary Figure 6).

Evaluation of runtime. *netprioR*'s robust integration of networks from data sources with varying degree of noise using a probabilistic model comes at the price of increased runtime compared to *TSS* and *LMgraph*. The runtime of *netprioR* is dominated by the convergence of the EM algorithm and depends strongly on the number of networks and available labels. The computational bottleneck in each iteration of the EM algorithm is the computation of the expectation

$$\mathbb{E}[\mathbf{R}_U | \mathbf{H}_L]^{(t)} = -(\mathbf{Q}_{UU}^{(t)})^{-1} \mathbf{Q}_{UL}^{(t)} (\mathbf{Q}_{LL}^{(t)} + \mathbf{S}_{LL}^{-1})^{-1} \mathbf{Q}_{LL}^{(t)} \mathbf{H}_L,$$

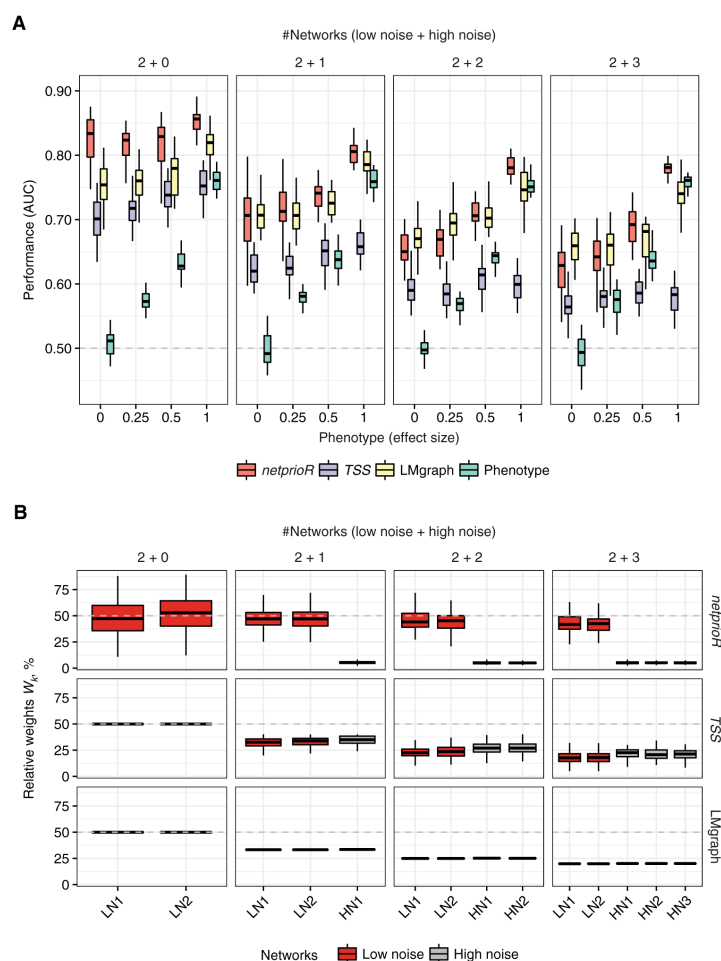


Figure 4: Comparison of *netprioR* (red), *TSS* (purple), *LMgraph* (yellow) and the *phenotype-only* baseline (green) on 50 simulated data sets with 10% a priori known labels, varying number of high-noise networks and phenotype effect size, respectively. Boxes summarise results over all simulations. (A) *netprioR* always outperforms or is on par with *TSS* and *LMgraph* in terms of the area under the receiver-operator characteristic curve (AUC, y-axis) for classifying unlabelled genes. Each panel depicts, from left to right, the addition of an increasing number of high-noise networks in addition to two low-noise networks. Within each panel, performance is depicted for increasing phenotype effect size. (B) *netprioR* emphasises low-noise networks. Relative weights (y-axis) for low- (red) and high-noise (grey) networks inferred by *netprioR* (top row), *TSS* (middle) and *LMgraph* (bottom).

because typically $U \gg L$. Using a conjugate gradient method to solve the linear equation system, the asymptotic runtime is $\mathcal{O}(m)$, where m is the sum of non-zero entries in Q_{UU} and the number of genes in U . The average runtime per restart at a convergence threshold of 10^{-6} for *netprioR* was 90.7 s. A comparative overview, illustrating the dependencies between runtime and the number of available prior knowledge labels and networks, for *netprioR*, *TSS* and *LMgraph* is shown in Supplementary Figure 4.

2.3 Prioritisation of novel regulators of Notch signalling in fly

Integrated datasets. We applied *netprioR* to Notch signalling in *Drosophila melanogaster*. Quantitative phenotypes were obtained from an *in vitro* RNA interference (RNAi) screen for regulators of Notch signalling by Saj and co-workers (Saj et al. 2010). In our application, we did not consider the direction of regulation and defined the phenotype as $|\log(\text{Firefly}/\text{Renilla})|$ based on the readout of the luminescence assay for Firefly and Renilla luciferase described in the supplement of (Saj et al. 2010). Since phenotypes were only available for 3840 genes from this study, we imputed missing phenotypes by randomly drawing values from an interval between 0 and the hit cut-off of 0.2 and repeated this procedure ten times. Prior labels were obtained from *in vivo* experiments from (Saj et al. 2010) and four additional studies summarised in (Guruharsha, Kankel, and Artavanis-Tsakonas 2012). A total of 1675 and 109 genes were labelled as positives and negatives, respectively, totalling 11.8% labelled data. The distribution of negative phenotypes exhibited a larger mode than the distribution of positive phenotypes, which is explained by the fact that labels were, as described above, derived from different experiments (Figure 5A). We obtained 22 different network data sets for *Drosophila melanogaster* from the Droid

database (Yu et al. 2008), with the type of interactions ranging from protein–protein, to co-expression, genetic, miRNA–gene, transcription factor (TF)–gene, and interolog data from human, worm and yeast. The network data sets were very heterogeneous with respect to the number of genes and interactions. Where information was available, we split interactions in each data set into high- and low-confidence (abbreviated HC and LC) subsets, following guidelines on the DroID website (web <http://tsudalab.org/files/code/eccb05.html>) and respective publications. An overview of all network datasets used for prioritisation is given in Table 1.

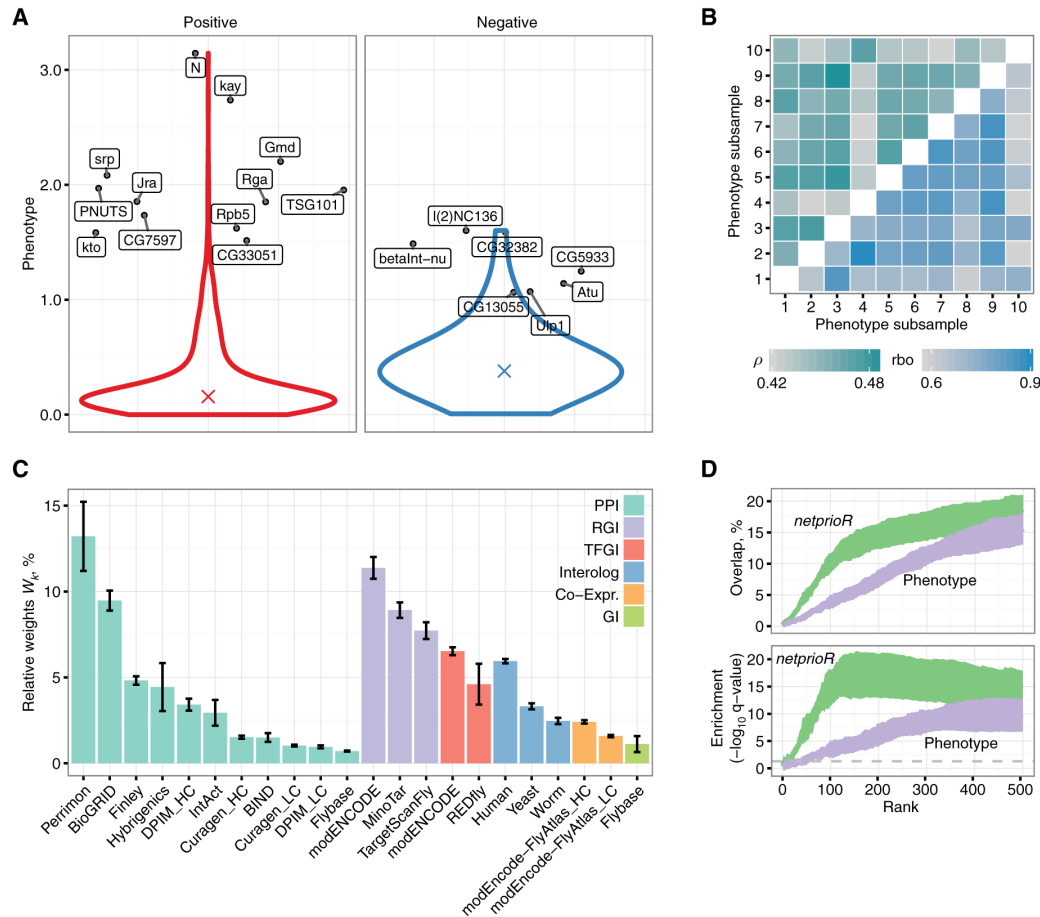


Figure 5: Gene prioritisation with *netprioR* for Notch signalling in *Drosophila melanogaster*. (A) The distribution of phenotypes for positive (P) hit genes (red) exhibits a smaller mode (red cross) but longer tail compared to negative (N) hit genes (blue). Phenotypes were jittered along the x-axis to avoid overlaps. Genes at the tails of distributions are highlighted with labels. (B) Rank similarity between ten data subsets, each time sub sampling 80% of measured phenotypes, is quantified as rank-biased overlap (Webber, Moffat, and Zobel 2010) (rbo, blue) and Spearman's rank correlation coefficient (ρ , green). Darker colour depicts higher similarity. (C) Inferred relative weights for the integration network data grouped by interaction type (colour). PPI: Protein–Protein, RGI: RNA–Gene, TFGI: Transcription Factor–Gene, Co-Expr.: Co-Expression, GI: Genetic. Error bars depict variation over ten 80% measured phenotype subsamples. (D) Rank evaluation of the leave-out set with the top 500 prioritised genes (top row) and corresponding Benjamini-Hochberg adjusted $-\log_{10}$ q-value from hyper-geometric tests for enrichment (bottom row) is higher for *netprioR* (green) compared to the baseline of *phenotype-only* (purple).

Table 1: Network data sets for *Drosophila melanogaster* from the DroID database (Yu et al. 2008).

Interaction	Source	Conf.	#Nodes	#Edges
Protein-Protein	BIND		591	1210
Protein-Protein	BioGRID		1095	4606
Protein-Protein	Curagen	high	4395	8854
Protein-Protein	Curagen	low	5210	29,336
Protein-Protein	DPIM	high	1830	5508
Protein-Protein	DPIM	low	4205	68,814
Protein-Protein	Finley		2236	12,380
Protein-Protein	Flybase		4127	31,626

Protein-Protein	Hybrigenics		1269	3648
Protein-Protein	IntAct		5163	30,224
Protein-Protein	Perrimon		252	762
Co-Expression	modEncode-FlyAtlas	high	8513	31,124
Co-Expression	modEncode-FlyAtlas	low	13,616	26,5538
Genetic	Flybase		2889	14,288
Interolog	Human		5688	13,4176
Interolog	Worm		1837	6924
Interolog	Yeast		2810	16,0984
RNA-Gene	MinoTar		2527	9410
RNA-Gene	modENCODE		1162	6288
RNA-Gene	TargetScanFly		11,925	21,0536
TF-Gene	modENCODE		12,313	31,3620
TF-Gene	REDfly		180	494

Robustness evaluation by sub-sampling. In the absence of a ground truth of Notch regulators and given the fact that we already included most of available prior knowledge in the integrative prioritisation, we evaluated the robustness of *netprioR* prioritisations with respect to missing phenotypes, as well as the predictive performance of the model by sub-sampling the data. First, we investigated the robustness of the model and constructed ten data sets from measured phenotypes, each containing 80% of all measurements. Integrating the full set of available network data and prior knowledge about labels, we fitted ten models and evaluated the pairwise stability between prioritisation ranks, as well as the robustness of inferred relative weights for each network data set. The average pairwise rank correlation between prioritisations was 0.45 and the average pairwise rank biased overlap (Webber, Moffat, and Zobel 2010) (rbo) of the top of the ranked lists was 0.75 (Figure 5B). This result indicates that top prioritised genes are highly stable with respect to missing phenotypic data, while the overall ranking is only moderately stable.

Network weight estimates were robust across phenotype subsamples with an average standard error of the mean (SEM) of 0.45 (Figure 5C). Networks spanning only a small number of genes (e.g. 180 Redfly transcription factor (TF)-gene interactions and 252 Perrimon protein-protein interactions (PPI)) exhibited, as expected, higher variation. Among the highest weighted network data sets were the PPI networks from Perrimon (13.2% relative weight) and BioGRID (9.5%), as well as the RNA-gene interaction networks from modENCODE (11.4%) and MinoTAR (8.9%). Low confidence (LC) filtered networks (Curagen_LC, DPIM_LC, modENCODE-FlyAtlas_LC) were assigned low weights, notably always lower than the respective high confidence (HC) counterparts (e.g. DPIM PPI 3.4% and 0.9%). This observation indicates that *netprioR* successfully down weighs noisy network data sets.

Next, we split the available prior knowledge about known true positive (TP) and true negative (TN) genes into ten non-overlapping subsets and evaluated the model performance in a ten-fold cross validation setting. In each iteration, we used 90% of available prior knowledge to predict labels for the remaining 10% and evaluated whether the TPs of the leave-out set were enriched at the top of the ranked prioritisation gene lists (Figure 5D). We computed the running overlap from ranks 1 to 500 of each prioritisation with the left out TPs and the corresponding Benjamini-Hochberg corrected q-values from hypergeometric tests for enrichment. Comparing the performance of *netprioR* to prioritising based on *phenotype-only*, we observed that our integrative model yielded much stronger overlaps (10.1% versus 3.9% at rank 100) and enrichments ($-\log_{10}$ q-values 14.7 versus 3.5 at rank 100).

Prioritisation of novel Notch regulators. For the prioritisation of Notch regulators, we fit the full *netprioR* model using all available network data sets, phenotype data and prior knowledge. The resulting list of the top 50 prioritised novel regulators, i.e. high-scoring unlabelled prioritised by the model, is show in Supplementary Table 1. Gene *shrb* (rank 1) and *AGO1* (rank 3) were previously both deemed hits for potential follow-up analysis based on their strong phenotype in the study by Saj and co-workers (Saj et al. 2010) and are again picked up by *netprioR*. However, also genes which did not generate a strong phenotype in the original screen were prioritised highly, for example, *Fadd* which was ranked 35. Literature search revealed that *Fadd* indeed modulates Notch signalling (Zhang et al. 2014). Furthermore, we compared a set of novel *in vivo* validated Notch regulators, hand-selected by an expert based on data from the study by Saj et al. (Saj et al. 2010) (Supplementary Table 2), to the prioritisation from *netprioR*. The *in vivo* phenotypes were obtained following the protocols described in (Saj et al. 2010). We found that the distribution of ranks of *in vivo* validated regulators in the *netprioR* prioritisation were positively skewed towards lower ranks (Figure 6A), indicating high prioritisation of true positive Notch regulators. Looking at the top 500 genes prioritised by *netprioR*, we observed an overlap of 13.2% with the set of validated regulators, which is a highly significant enrichment yielding a $-\log_{10}$ q-value of 4.6 (Figure 6B).

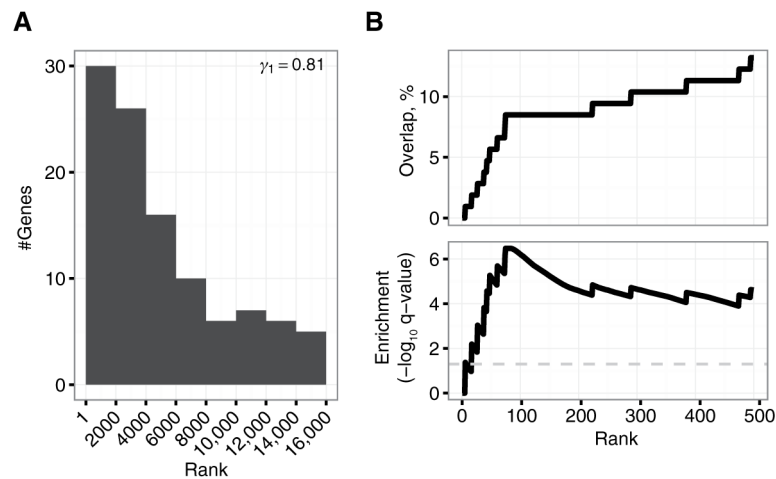


Figure 6: Set of hand-selected *in vivo* validated genes for Notch signalling is enriched at the top of the *netprioR* prioritisation ranking. (A) Distribution of ranks of *in vivo* validated genes in the *netprioR* prioritisation ranking are skewed towards low ranks (γ_1 depicts skewness). (B) Rank evaluation of *in vivo* validated genes. The percent overlap with the top 500 prioritised genes (top row) is significant, as shown by the Benjamini-Hochberg adjusted $-\log_{10}$ q-value from hypergeometric tests for enrichment (bottom row).

3 Discussion

We developed a probabilistic, generative model for integrative gene prioritisation and devised an EM algorithm for parameter inference. In contrast to many other methods in the field (Tsuda, Shin & Schölkopf, 2005; Mostafavi et al., 2008; Kato, Kashima & Sugiyama, 2009), *netprioR* allows for the integration of gene-based covariates, such as phenotypic readouts from perturbation screens, in addition to multiple network data sets and prior knowledge in the form of known true positive and true negative hits. Our comparative study to assess the performance of *netprioR* has several limitations. Like any simulation study, it is driven by the way the data is simulated. While we have not drawn the data from the *netprioR* model itself to avoid over-optimism, other simulation settings may result in other performance differences. In addition, we have only used the default parameters of all competing methods, and future, more extensive simulation studies that include systematic parameter optimisation should address this limitation.

Robustness to high-noise networks. *netprioR* showed superior performance in comparison to competing methods (Tsuda, Shin & Schölkopf, 2005; Mostafavi et al., 2008) in a simulation study (Figure 3 and Figure 4); in particular in cases with increased number of high-noise network data sources. This conclusion is particularly promising, as it permits integrating highly heterogeneous data sets of different quality without having to guess which data sets to include a priori for a certain prioritisation task. However, for a fixed number of prior knowledge gene labels and fixed phenotypic effect size the accuracy of the prioritisation, as expected, did show a decreasing trend with increasing number of high-noise networks (panels in Figure 4A). This is a consequence of the fact that *netprioR* shrinks the weights of high-noise networks, but does not set them to zero. Hence, a possible extension of *netprioR*, in order to yield a sparsifying estimation of network weights W_k , could be to use a prior distribution on W_k similar to the Laplace distribution as for the Bayesian Lasso (Park and Casella 2008).

Novel regulators of Notch signalling. Integrating 22 different network data sets, 3840 perturbation screen phenotypes and 1784 labels for a priori known true positive and true negative genes, we successfully prioritised novel regulators of Notch signalling in *Drosophila melanogaster*. While we provided evidence from *in vivo* experiments for the biological relevance of several prioritised genes, *netprioR* generated many more directly testable hypotheses for potential regulators. The list of top 50 prioritised genes (Supplementary Table 1) contained numerous ribosomal proteins. Due to strong phenotypic effects (Saj et al. 2010) and true positive hit labels (Guruharsha, Kankel, and Artavanis-Tsakonas 2012) for ribosomal proteins, this result was to be expected. However, the exact role of ribosomal proteins in Notch signalling remains unclear as *in vivo* perturbation experiments for validation are typically prohibitive due to toxic effects (Saj et al. 2010).

Wide applicability. With increasing amounts of publicly available omics data, *netprioR* will be a valuable tool for the identification of novel hit genes from perturbation screen hits based on RNAi or CRISPR/Cas9. Its availability as an R/Bioconductor package allows for smooth integration into existing data analysis pipelines. Apart from prioritisation of perturbation screen hits, *netprioR* could also be applied to disease gene prioritisation tasks, such as, for instance, the prediction of driver genes in cancer (Moreau & Tranchevent, 2012; Leiserson

et al., 2015; Dimitrakopoulos et al., 2018). While using similar network data sets, rich sources for a priori known driver genes, such as COSMIC (Forbes et al. 2015), are readily available. As additional covariates, one could integrate mutation profiles or gene expression measurements, for instance from exome sequencing experiments.

Computational complexity. A limitation of *netprioR* is the high computational cost proportional to m , the sum of the joint number of interactions and number of genes for all networks. In order to reduce m , pre-processing steps, such as constructing the n -nearest-neighbour network from each data source prior to integration, will lead to higher sparsity in Q and consequently to decreased runtime (Mostafavi et al. 2008). Nevertheless, the application to Notch signalling in *Drosophila melanogaster*, integrating as many as 22 network data sets in under two hours, showed that the current implementation of *netprioR* is in fact suitable for genome-scale problems.

4 Materials and methods

4.1 Model inference

Let the set of hidden data be $\mathbf{Z} = \{\mathbf{R}, \mathbf{Y}_U\}$, the set of observed data be $\mathbf{D} = \{\mathbf{Y}_L, \mathbf{X}, \mathbf{G}_1, \dots, \mathbf{G}_K\}$, and the set of all parameters be $\boldsymbol{\theta} = \{\mathbf{W}, \boldsymbol{\beta}\}$. The hidden log-likelihood of the model defined in Eqs. (1)–(4) is

$$\begin{aligned}\mathcal{L}_{\text{hid}}(\mathbf{W}, \boldsymbol{\beta}) &= \log \Pr(\mathbf{D}, \mathbf{Z} | \mathbf{W}, \boldsymbol{\beta}) = \log \Pr(\mathbf{Y}_L, \mathbf{Y}_U, \mathbf{X}, \mathbf{G}, \mathbf{R} | \mathbf{W}, \boldsymbol{\beta}) \\ &= \log \Pr(\mathbf{Y}_L | \mathbf{R}_L, \mathbf{X}_L, \boldsymbol{\beta}) + \log \Pr(\mathbf{Y}_U | \mathbf{R}_U, \mathbf{X}_U, \boldsymbol{\beta}) + \log \Pr(\mathbf{R} | \mathbf{G}, \mathbf{W}) \\ &= \sum_{i=1}^L \log \Pr(Y_i | \mathbf{X}_i, R_i, \boldsymbol{\beta}) + \sum_{j=L+1}^{L+U} \log \Pr(Y_j | \mathbf{X}_j, R_j, \boldsymbol{\beta}) \\ &\quad + \log \Pr(\mathbf{R} | \mathbf{G}, \mathbf{W})\end{aligned}$$

Including the priors for the model parameters and substituting the normal and gamma distributions (Eqs. (1)–(2)), the logarithm of the posterior probability of the parameters is given by

$$\begin{aligned}\log \Pr(\boldsymbol{\theta} | \mathbf{D}, \mathbf{Z}) &\propto \mathcal{L}_{\text{hid}}(\mathbf{W}, \boldsymbol{\beta}) + \sum_{k=1}^K \log \Pr(W_k | a, b) + \sum_{i=1}^P \log \Pr(\beta_i | \tau) \\ &= \log \text{Norm}(\mathbf{X}_L \boldsymbol{\beta} + \mathbf{R}_L, \mathbf{S}_{LL}) + \log \text{Norm}(\mathbf{X}_U \boldsymbol{\beta} + \mathbf{R}_U, \mathbf{S}_{UU}) \\ &\quad + \log \text{Norm}(\mathbf{0}, \mathbf{Q}^{-1}) + \sum_{k=1}^K \log \text{Ga}(a, b) + \log \text{Norm}(\mathbf{0}, \mathbf{T}).\end{aligned}$$

Substituting and removing constant terms,

$$\begin{aligned}\log \Pr(\boldsymbol{\theta} | \mathbf{D}, \mathbf{Z}) &\propto -\frac{1}{2} \left((\mathbf{Y}_L - \mathbf{R}_L - \mathbf{X}_L \boldsymbol{\beta})^\top \mathbf{S}_{LL}^{-1} (\mathbf{Y}_L - \mathbf{R}_L - \mathbf{X}_L \boldsymbol{\beta}) \right) \\ &\quad - \frac{1}{2} \left((\mathbf{Y}_U - \mathbf{R}_U - \mathbf{X}_U \boldsymbol{\beta})^\top \mathbf{S}_{UU}^{-1} (\mathbf{Y}_U - \mathbf{R}_U - \mathbf{X}_U \boldsymbol{\beta}) \right) \\ &\quad - \frac{1}{2} \left(\mathbf{R}^\top \left(\sum_{k=1}^K W_k \mathbf{G}_k \right) \mathbf{R} + \sum_{k=1}^K N \log \frac{1}{W_k} \right) \\ &\quad + \sum_{k=1}^K ((a-1) \log W_k - b W_k) - \frac{1}{2} (\boldsymbol{\beta}^\top \mathbf{T}^{-1} \boldsymbol{\beta}),\end{aligned}$$

and after re-arranging

$$\begin{aligned} \log \Pr(\boldsymbol{\theta} \mid \mathbf{D}, \mathbf{Z}) &\propto -\frac{1}{2} \left((\mathbf{Y}_L - \mathbf{R}_L - \mathbf{X}_L \boldsymbol{\beta})^\top \mathbf{S}_{LL}^{-1} (\mathbf{Y}_L - \mathbf{R}_L - \mathbf{X}_L \boldsymbol{\beta}) \right) \\ &\quad - \frac{1}{2} \left((\mathbf{Y}_U - \mathbf{R}_U - \mathbf{X}_U \boldsymbol{\beta})^\top \mathbf{S}_{UU}^{-1} (\mathbf{Y}_U - \mathbf{R}_U - \mathbf{X}_U \boldsymbol{\beta}) \right) \\ &\quad + \sum_{k=1}^K \left(\left(a + \frac{N}{2} - 1 \right) \log W_k - \left(b + \frac{1}{2} \mathbf{R}^\top \mathbf{G}_k \mathbf{R} \right) W_k \right) \\ &\quad - \frac{1}{2} \boldsymbol{\beta}^\top \mathbf{T}^{-1} \boldsymbol{\beta}. \end{aligned}$$

We aim to find the maximum a posteriori (MAP) estimate of parameters in our latent variable model and to predict the missing labels $\mathbf{Y}_U$. For this purpose, we use the Expectation Maximisation (EM) algorithm, which iteratively maximises the expected hidden log-likelihood of the data with respect to the logarithm of the posterior distribution of $(\mathbf{D}, \mathbf{Z})$ given previous parameter estimates $\boldsymbol{\theta}^{(t-1)}$. This expectation is given by

$$\mathcal{Q}(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(t-1)}) = \mathbb{E}_{\mathbf{Z} \mid \mathbf{D}, \boldsymbol{\theta}^{(t-1)}} [\log \Pr(\mathbf{D}, \mathbf{Z} \mid \boldsymbol{\theta}^{(t)})] + \log \Pr(\boldsymbol{\theta}^{(t)})$$

In each iteration of the EM algorithm, $\mathcal{Q}$ is maximised with respect to $\boldsymbol{\theta}^{(t)}$ (M-step) and the posterior of the hidden data $\Pr(\mathbf{Z} \mid \mathbf{D}, \boldsymbol{\theta}^{(t-1)})$ is re-estimated (E-step). In each iteration of the EM algorithm, the likelihood of the observed data will increase until it reaches a local maximum. In order to avoid being stuck in a local optimum, we perform multiple restarts of the EM algorithm with different initial settings for parameters and choose the parameters that yield the overall maximum likelihood over all restarts.

M-step. In the M-step, we obtain a new estimate of the parameters by maximising $\mathcal{Q}$ with respect to the previous estimate of the parameters $\boldsymbol{\theta}^{(t-1)}$. Because $\mathbb{E}[\mathbf{Y}_U] = \mathbb{E}[\mathbf{R}_U] + \mathbf{X}_U \boldsymbol{\beta}^{(t)}$, we find

$$\begin{aligned} \mathcal{Q}(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(t-1)}) &= -\frac{1}{2} \left((\mathbf{Y}_L - \mathbb{E}[\mathbf{R}_L] - \mathbf{X}_L \boldsymbol{\beta}^{(t)})^\top \mathbf{S}_{LL}^{-1} (\mathbf{Y}_L - \mathbb{E}[\mathbf{R}_L] - \mathbf{X}_L \boldsymbol{\beta}^{(t)}) \right) \\ &\quad + \sum_{k=1}^K \left(\left(a + \frac{N}{2} - 1 \right) \log W_k - \left(b + \frac{1}{2} \mathbb{E}[\mathbf{R}]^\top \mathbf{G}_k \mathbb{E}[\mathbf{R}] \right) W_k \right) \\ &\quad - \frac{1}{2} (\boldsymbol{\beta}^{(t)})^\top \mathbf{T}^{-1} \boldsymbol{\beta}^{(t)} \end{aligned}$$

The new parameter estimates

$$\boldsymbol{\theta}^{(t)} = \arg \max_{\boldsymbol{\theta}^{(t)}} \mathcal{Q}(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(t-1)})$$

are obtained by setting the derivative of $\mathcal{Q}(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(t-1)})$ with respect to each parameter to zero and solving for the parameter. For the network weights, we solve

$$0 = \frac{\partial}{\partial W_k^{(t)}} \mathcal{Q}(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(t-1)}) = \left(a + \frac{N}{2} - 1 \right) \frac{1}{W_k^{(t)}} - \left(b + \frac{1}{2} \mathbb{E}[\mathbf{R}]^\top \mathbf{G}_k \mathbb{E}[\mathbf{R}] \right)$$

to obtain

$$W_k^{(t)} = \frac{a + \frac{N}{2} - 1}{b + \frac{1}{2} \mathbb{E}[\mathbf{R}]^\top \mathbf{G}_k \mathbb{E}[\mathbf{R}]}, \quad (5)$$

and similarly, for the fixed effects,

$$\begin{aligned} 0 = \frac{\partial}{\partial \boldsymbol{\beta}^{(t)}} \mathcal{Q}(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(t-1)}) &= ((\mathbf{X}_L^\top \mathbf{S}_{LL}^{-1} (\mathbf{Y}_L - \mathbb{E}[\mathbf{R}_L]) - \mathbf{X}_L \boldsymbol{\beta}^{(t)}) + \mathbf{T}^{-1} \boldsymbol{\beta}^{(t)}) \\ &= \mathbf{X}_L^\top \mathbf{S}_{LL}^{-1} (\mathbf{Y}_L - \mathbb{E}[\mathbf{R}_L]) - (\mathbf{X}_L^\top \mathbf{S}_{LL}^{-1} \mathbf{X}_L + \mathbf{T}^{-1}) \boldsymbol{\beta}^{(t)} \end{aligned}$$

yields

$$\boldsymbol{\beta}^{(t)} = (\mathbf{X}_L^\top \mathbf{S}_{LL}^{-1} \mathbf{X}_L + \mathbf{T}^{-1})^{-1} \mathbf{X}_L^\top \mathbf{S}_{LL}^{-1} (\mathbf{Y}_L - \mathbb{E}[\mathbf{R}_L]). \quad (6)$$

E-step. In the E-step, we compute the expected values of the hidden data $\mathbf{Z} = \{\mathbf{R}, \mathbf{Y}_U\}$ given the observed data $\mathbf{D} = \{\mathbf{Y}_L, \mathbf{X}, \mathbf{G}\}$, $\mathbb{E}[\mathbf{R} | \mathbf{D}]$ and $\mathbb{E}[\mathbf{Y}_U | \mathbf{D}]$. For this purpose, we define the auxiliary random variable $\mathbf{H} = \mathbf{Y} - \mathbf{X}\boldsymbol{\beta}$, such that

$$\mathbf{H} | \mathbf{R} \sim \text{Norm}(\mathbf{R}, \mathbf{S})$$

We re-order genes, such that, without loss of generality, labelled genes appear before unlabelled genes and partition $\mathbf{H}, \mathbf{R}, \mathbf{S}$ and $\mathbf{Q}$, such that

$$\mathbf{H} = \begin{pmatrix} \mathbf{H}_L \\ \mathbf{H}_U \end{pmatrix}^\top, \mathbf{R} = \begin{pmatrix} \mathbf{R}_L \\ \mathbf{R}_U \end{pmatrix}^\top, \mathbf{S} = \begin{pmatrix} \mathbf{S}_{LL} & \mathbf{S}_{LU} \\ \mathbf{S}_{UL} & \mathbf{S}_{UU} \end{pmatrix}, \mathbf{Q} = \begin{pmatrix} \mathbf{Q}_{LL} & \mathbf{Q}_{LU} \\ \mathbf{Q}_{UL} & \mathbf{Q}_{UU} \end{pmatrix}.$$

Consequently, for the subset of labelled genes

$$\begin{aligned} \mathbf{H}_L | \mathbf{R}_L &\sim \text{Norm}(\mathbf{R}_L, \mathbf{S}_{LL}) \\ \mathbf{R}_L &\sim \text{Norm}(\mathbf{0}, \mathbf{Q}_{LL}^{-1}). \end{aligned}$$

The conditional distribution of $\mathbf{R}_L$ given $\mathbf{H}_L$ is constructed by completing the squares for the joint distribution $\text{Pr}(\mathbf{H}_L, \mathbf{R}_L)$ and conditioning on $\mathbf{H}_L$ (Bishop 2006). First, we derive the expression for the joint distribution of $\mathbf{J} = (\mathbf{R}_L, \mathbf{H}_L)$. Its logarithm is

$$\begin{aligned} \log \text{Pr}(\mathbf{J}) &= \log \text{Pr}(\mathbf{R}_L) + \log \text{Pr}(\mathbf{H}_L | \mathbf{R}_L) \\ &\propto -\frac{1}{2} \mathbf{R}_L^\top \mathbf{Q}_{LL} \mathbf{R}_L - \frac{1}{2} (\mathbf{H}_L - \mathbf{R}_L)^\top \mathbf{S}_{LL}^{-1} (\mathbf{H}_L - \mathbf{R}_L) \end{aligned} \quad (7)$$

It can be seen that $\log \text{Pr}(\mathbf{J})$ is a quadratic function of the components of $\mathbf{J}$ and consequently $\text{Pr}(\mathbf{J})$ follows a Gaussian distribution. In order to find the precision matrix of this Gaussian, we consider all second order terms in Eq. (7), which can be written as

$$\begin{aligned} \log \text{Pr}(\mathbf{J}) &\propto -\frac{1}{2} \mathbf{R}_L^\top (\mathbf{Q}_{LL} + \mathbf{S}_{LL}^{-1}) \mathbf{R}_L - \frac{1}{2} \mathbf{H}_L^\top \mathbf{S}_{LL}^{-1} \mathbf{H}_L + \frac{1}{2} \mathbf{R}_L^\top \mathbf{Q}_{LL} \mathbf{H}_L \\ &= -\frac{1}{2} \begin{pmatrix} \mathbf{R}_L \\ \mathbf{H}_L \end{pmatrix}^\top \begin{pmatrix} \mathbf{Q}_{LL} + \mathbf{S}_{LL}^{-1} & -\mathbf{Q}_{LL} \\ -\mathbf{Q}_{LL} & \mathbf{Q}_{LL} \end{pmatrix} \begin{pmatrix} \mathbf{R}_L \\ \mathbf{H}_L \end{pmatrix} = -\frac{1}{2} \mathbf{J}^\top \mathbf{M} \mathbf{J}. \end{aligned} \quad (8)$$

The covariance matrix of the Gaussian distribution of $\mathbf{J}$ is the inverse of the precision matrix $\mathbf{M}$. It is computed as

$$\text{cov}[\mathbf{J}] = \mathbf{M}^{-1} = \begin{pmatrix} \mathbf{Q}_{LL}^{-1} & \mathbf{Q}_{LL}^{-1} \\ \mathbf{Q}_{LL}^{-1} & \mathbf{S}_{LL} + \mathbf{Q}_{LL}^{-1} \end{pmatrix}. \quad (9)$$

We note that the exponent in a general Gaussian distribution $\text{Norm}(\mathbf{x} | \boldsymbol{\mu}, \boldsymbol{\Sigma})$ can be written as

$$-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) = -\frac{1}{2} \mathbf{x}^\top \boldsymbol{\Sigma}^{-1} \mathbf{x} + \mathbf{x}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu} + \text{const}. \quad (10)$$

In order to find an expression for the conditional distribution $\text{Pr}(\mathbf{R}_L | \mathbf{H}_L)$, we re-write Eq. (8), substituting $\mathbf{Q}_{LL}$ and $\mathbf{S}_{LL}$ with their respective counterparts in the form of partitions of $\mathbf{M}^{-1}$. Re-arranging yields

$$-\frac{1}{2} \mathbf{R}_L^\top \mathbf{M}_{R_L, R_L} \mathbf{R}_L + \frac{1}{2} \mathbf{R}_L^\top (\mathbf{M}_{R_L, H_L} \mathbf{H}_L) + \text{const}, \quad (11)$$

which is equivalent to the form in Eq. (10). Then, we can derive the mean and covariance of the conditional Gaussian $\mathbf{R}_L | \mathbf{H}_L \sim \text{Norm}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ from Eq. (11) as

$$\begin{aligned}\Sigma &= \mathbf{M}_{R_L, R_L}^{-1} = (\mathbf{Q}_{LL} + \mathbf{S}_{LL}^{-1})^{-1} \\ \mu &= \Sigma \mathbf{M}_{R_L, H_L} \mathbf{H}_L = \Sigma \mathbf{Q}_{LL} \mathbf{H}_L = \mathbb{E}[\mathbf{R}_L | \mathbf{H}_L].\end{aligned}\quad (12)$$

Similarly, the conditional Gaussian distribution of $\mathbf{R}_U$ given $\mathbf{R}_L$ is given by

$$\mathbf{R}_U | \mathbf{R}_L \sim \text{Norm}(-\mathbf{Q}_{UU}^{-1} \mathbf{Q}_{UL} \mathbf{R}_L, \mathbf{Q}_{UU}^{-1}).$$

This result is fundamental in the field of Gaussian Markov Random Fields (GMRFs), where unobserved vertices are typically conditioned on observed vertices (Rue and Held 2005). By the law of total expectation, we can compute the conditional expectation for the random effect of genes with unobserved labels, $\mathbf{R}_U$, as

$$\begin{aligned}\mathbb{E}[\mathbf{R}_U | \mathbf{H}_L] &= \mathbb{E}[\mathbb{E}[\mathbf{R}_U | \mathbf{R}_L] | \mathbf{H}_L] = \mathbb{E}[\mathbf{0} - \mathbf{Q}_{UU}^{-1} \mathbf{Q}_{UL} (\mathbf{R}_L - \mathbf{0}) | \mathbf{H}_L] \\ &= -\mathbf{Q}_{UU}^{-1} \mathbf{Q}_{UL} \mathbb{E}[\mathbf{R}_L | \mathbf{H}_L] = -\mathbf{Q}_{UU}^{-1} \mathbf{Q}_{UL} \Sigma \mathbf{Q}_{LL} \mathbf{H}_L\end{aligned}\quad (13)$$

and likewise the conditional expectation of the unobserved labels, $\mathbf{Y}_U$, as

$$\begin{aligned}\mathbb{E}[\mathbf{H}_U | \mathbf{H}_L] &= \mathbb{E}[\mathbb{E}[\mathbf{H}_U | \mathbf{R}_U] | \mathbf{H}_L] = \mathbb{E}[\mathbb{E}[\mathbb{E}[\mathbf{H}_U | \mathbf{R}_U] | \mathbf{R}_L] | \mathbf{H}_L] \\ &= \mathbb{E}[\mathbb{E}[\mathbf{R}_U | \mathbf{R}_L] | \mathbf{H}_L] = -\mathbf{Q}_{UU}^{-1} \mathbf{Q}_{UL} \Sigma \mathbf{Q}_{LL} \mathbf{H}_L \\ \mathbb{E}[\mathbf{Y}_U | \mathbf{H}_L] &= \mathbb{E}[\mathbf{H}_U + \mathbf{X}\beta | \mathbf{H}_L] = \mathbb{E}[\mathbf{H}_U | \mathbf{H}_L] + \mathbf{X}\beta \\ &= \mathbb{E}[\mathbf{R}_U | \mathbf{H}_L] + \mathbf{X}\beta = -\mathbf{Q}_{UU}^{-1} \mathbf{Q}_{UL} \Sigma \mathbf{Q}_{LL} \mathbf{H}_L + \mathbf{X}\beta.\end{aligned}\quad (14)$$

4.2 Implementation

We iterate the E-step and M-step of the EM algorithm until the difference in the expected hidden log likelihood of two consecutive iterations is smaller than 10^{-6} . The initial parameters for $\mathbf{W}$ and β are drawn independently from $\text{Ga}(a, b)$ and $\text{Norm}(0, \tau)$, respectively. The computation of the expectations in the E-Step of the EM algorithm are computed using a conjugate gradient solver for large problems. We perform five restarts of the EM algorithm and report the parameter estimates and imputed data that yielded the highest expected hidden log likelihood. We found that higher number of restarts did typically not give an improvement in performance. The model is implemented in the R-package *netprioR* and is available as open source software under GNU General Public licence version 3 (GPL-3) at <http://bioconductor.org/packages/netprioR>.

4.3 Construction and normalisation of gene–gene similarity networks from omics data

While the construction of similarity networks in the case of protein–protein or genetic interactions is straightforward (interaction = similarity), for other data sets, different measures of similarity exist. For co-expression networks, for instance, where each gene is associated with an expression profile over multiple conditions, a common approach is to use thresholded pairwise correlation between genes as a proxy for similarity (Stuart et al. 2003). Interolog data are predicted interactions which are based on experimental evidence for interactions between orthologous genes or proteins in other species. These interactions typically span a multitude of different omics data sets from additional databases. In this study, we used 22 distinct gene–gene similarity networks from the DroID database (Yu et al. 2008) with highly heterogeneous numbers of genes and interactions. Therefore, we normalised the network data sets by scaling each interaction by the Frobenius norm of the adjacency matrix of corresponding network. This step allows to compare *netprioR*'s weight estimates W_k between network data sets.

Author contributions: Conceived and designed the experiments: FS GM NB. Analysed the data: FS. Contributed reagents/materials/analysis tools: JK GM. Wrote the paper: FS NB.

References

- Aerts, S., D. Lambrechts, S. Maity, P. Van Loo, B. Coessens, F. De Smet, L.-C. Tranchevent, B. De Moor, P. Marynen, B. Hassan, P. Carmeliet and Y. Moreau (2006): "Gene prioritization through genomic data fusion," *Nat. Biotechnol.*, 24, 537–544.
- C. M. Bishop (2006): *Pattern recognition and machine learning (information science and statistics)*, Springer-Verlag New York, Inc., Secaucus, NJ, USA.
- Chen, J., E. E. Bardes, B. J. Aronow and A. G. Jegga (2009): "ToppGene Suite for gene list enrichment analysis and candidate gene prioritization," *Nucleic Acids Res.*, 37, W305–W311.
- Chintapalli, V. R., J. Wang and J. A. T. Dow (2007): "Using FlyAtlas to identify better *Drosophila melanogaster* models of human disease," *Nat. Genet.*, 39, 715–720.
- Costanzo, M., A. Baryshnikova, J. Bellay, Y. Kim, E. D. Spear, C. S. Sevier, H. Ding, J. L. Y. Koh, K. Toufighi, S. Mostafavi, J. Prinz, R. P. St Onge, B. VanderSluis, T. Makhnevych, F. J. Vizeacoumar, S. Alizadeh, S. Bahr, R. L. Brost, Y. Chen, M. Cokol, R. Deshpande, Z. Li, Z.-Y. Lin, W. Liang, M. Marback, J. Paw, B.-J. San Luis, E. Shuteriqi, A. H. Y. Tong, N. van Dyk, I. M. Wallace, J. A. Whitney, M. T. Weirauch, G. Zhong, H. Zhu, W. A. Houry, M. Brudno, S. Ragibizadeh, B. Papp, C. Pál, F. P. Roth, G. Giaever, C. Nislow, O. G. Troyanskaya, H. Bussey, G. D. Bader, A.-C. Gingras, Q. D. Morris, P. M. Kim, C. A. Kaiser, C. L. Myers, B. J. Andrews and C. Boone (2010): "The genetic landscape of a cell," *Science*, 327, 425–431.
- Cristianini, N., J. Kandola, A. Elisseeff and J. Shawe-Taylor (2002): "On kernel-target alignment." In: *Advances in Neural Information Processing Systems 14*. Berlin, Heidelberg: MIT Press. pp. 367–373.
- Dimitrakopoulos, C., S. K. Hindupur, L. Häfliger, J. Behr, H. Montazeri, M. N. Hall and N. Beerenwinkel (2018): "Network-based integration of multi-omics data for prioritizing cancer genes," *Bioinformatics*, 34, 2441–2448.
- Forbes, S. A., D. Beare, P. Gunasekaran, K. Leung, N. Bindal, H. Boutselakis, M. Ding, S. Bamford, C. Cole, S. Ward, C. Y. Kok, M. Jia, T. De, J. W. Teague, M. R. Stratton, U. McDermott and P. J. Campbell (2015): "COSMIC: exploring the world's knowledge of somatic mutations in human cancer," *Nucleic Acids Res.*, 43, D805–D811.
- Formstecher, E., S. Aresta, V. Collura, A. Hamburger, A. Meil, A. Trehin, C. Reverdy, V. Betin, S. Maire, C. Brun, B. Jacq, M. Arpin, Y. Bellaiche, S. Bellusci, P. Benaroch, M. Bornens, R. Chanet, P. Chavrier, O. Delattre, V. Doye, R. Fehon, G. Faye, T. Galli, J.-A. Girault, B. Goud, J. de Gunzburg, L. Johannes, M.-P. Junier, V. Mirouse, A. Mukherjee, D. Papadopoulou, F. Perez, A. Plessis, C. Rossé, S. Saule, D. Stoppa-Lyonnet, A. Vincent, M. White, P. Legrain, J. Wojcik, J. Camonis and L. Daviet (2005): "Protein interaction mapping: a *Drosophila* case study," *Genome Research*, 15, 376–384.
- Friedman, A. A., G. Tucker, R. Singh, D. Yan, A. Vinayagam, Y. Hu, R. Binari, P. Hong, X. Sun, M. Porto, S. Pacifico, T. Murali, R. L. Finley, J. M. Asara, B. Berger and N. Perrimon (2011): "Proteomic and functional genomic landscape of receptor tyrosine kinase and ras to extracellular signal-regulated kinase signaling," *Sci Signal*, 4, rs10–rs10.
- Giot, L., J. S. Bader, C. Brouwer, A. Chaudhuri, B. Kuang, Y. Li, Y. L. Hao, C. E. Ooi, B. Godwin, E. Vitols, G. Vijayadamodar, P. Pochart, H. Machineni, M. Welsh, Y. Kong, B. Zerhusen, R. Malcolm, Z. Varrone, A. Collis, M. Minto, S. Burgess, L. McDaniel, E. Stimpson, F. Spriggs, J. Williams, K. Neurath, N. Ioime, M. Agee, E. Voss, K. Furtak, R. Renzulli, N. Aanensen, S. Carroll, E. Bickelhaupt, Y. Lazovatsky, A. DaSilva, J. Zhong, C. A. Stanyon, R. L. Finley, K. P. White, M. Braverman, T. Jarvie, S. Gold, M. Leach, J. Knight, R. A. Shinkets, M. P. McKenna, J. Chant and J. M. Rothberg (2003): "A protein interaction map of *Drosophila melanogaster*," *Science*, 302, 1727–1736.
- Graveley, B. R., A. N. Brooks, J. W. Carlson, M. O. Duff, J. M. Landolin, L. Yang, C. G. Artieri, M. J. van Baren, N. Boley, B. W. Booth, J. B. Brown, L. Cherbas, C. A. Davis, A. Dobin, R. Li, W. Lin, J. H. Malone, N. R. Mattiuzzo, D. Miller, D. Sturgill, B. B. Tuch, C. Zaleski, D. Zhang, M. Blanchette, S. Dudoit, B. Eads, R. E. Green, A. Hammonds, L. Jiang, P. Kapranov, L. Langton, N. Perrimon, J. E. Sandler, K. H. Wan, A. Willingham, Y. Zhang, Y. Zou, J. Andrews, P. J. Bickel, S. E. Brenner, M. R. Brent, P. Cherbas, T. R. Gingeras, R. A. Hoskins, T. C. Kaufman, B. Oliver and S. E. Celniker (2011): "The developmental transcriptome of *Drosophila melanogaster*," *Nature*, 471, 473–479.
- Guruharsha, K. G., J.-F. Rual, B. Zhai, J. Mintseris, P. Vaidya, N. Vaidya, C. Beekman, C. Wong, D. Y. Rhee, O. Cenaj, E. McKillip, S. Shah, M. Stapleton, K. H. Wan, C. Yu, B. Parsa, J. W. Carlson, X. Chen, B. Kapadia, K. VijayRaghavan, S. P. Gygi, S. E. Celniker, R. A. Obar and S. Artavanis-Tsakonas (2011): "A protein complex network of *Drosophila melanogaster*," *Cell*, 147, 690–703.
- Guruharsha, K. G., M. W. Kankel and S. Artavanis-Tsakonas (2012): "The Notch signalling system: recent insights into the complexity of a conserved pathway," *Nat. Rev. Genet.*, 13, 654–666.
- K. Horan, C. Jang, J. Bailey-Serres, R. Mittler, C. Shelton, J. F. Harper, Zhu, J.-K., J. C. Cushman, M. Gollery and T. Girke (2008): "Annotating genes of known and unknown function by large-scale coexpression analysis," *Plant Physiol.*, 147, 41–57.
- Kato, T., H. Kashima and M. Sugiyama (2009): "Robust label propagation on multiple networks," *IEEE Trans. Neural Netw.*, 20, 35–44.
- Leiserson, M. D. M., F. Vandin, H.-T. Wu, J. R. Dobson, J. V. Eldridge, J. L. Thomas, A. Papoutsaki, Y. Kim, B. Niu, M. McLellan, M. S. Lawrence, A. Gonzalez-Perez, D. Tamborero, Y. Cheng, G. A. Ryslik, N. Lopez-Bigas, G. Getz, L. Ding and B. J. Raphael (2015): "Pan-cancer network analysis identifies combinations of rare somatic mutations across pathways and protein complexes," *Nat. Genet.*, 47, 106–114.
- Moreau, Y. and L.-C. Tranchevent (2012): "Computational tools for prioritizing candidate genes: boosting disease gene discovery," *Nat. Rev. Genet.*, 13, 523–536.
- Mostafavi, S., D. Ray, D. Warde-Farley, C. Grouios and Q. Morris (2008): "GeneMANIA: a real-time multiple association network integration algorithm for predicting gene function," *Genome Biology*, 9(Suppl 1), S4.
- Park, T. and G. Casella (2008): "The bayesian lasso," *J. Am. Stat. Assoc.*, 103, 681–686.
- Rämö, P., A. Drewek, C. Arriemerlou, N. Beerenwinkel, H. Ben-Tekaya, B. Cardel, A. Casanova, R. Conde-Alvarez, P. Cossart, G. Csúcs, S. Eicher, M. Emmenlauer, U. Greber, W.-D. Hardt, A. Helenius, C. Kasper, A. Kaufmann, S. Kreibich, A. Kühbacher, P. Kunszt, S. H. Low, J. Mercer, D. Mudrak, S. Muntwiler, L. Pelkmans, J. Pizarro-Cerdá, M. Podvinec, E. Pujadas, B. Rinn, V. Rouilly, F. Schmich, J. Siebourg-Polster, B. Snijder, M. Stebler, G. Studer, E. Szczurek, M. Truttmann, C. von Mering, A. Vonderheit, A. Yakimovich, P. Bühlmann and C. Dehio (2014): "Simultaneous analysis of large-scale RNAi screens for pathogen entry," *BMC Genomics*, 15, 1162.
- Rue, H. and L. Held (2005): *Gaussian markov random fields: theory and application*. Boca Raton: Chapman & Hall/CRC.

- Saj, A., Z. Arziman, D. Stempfle, W. van Belle, U. Sauder, T. Horn, M. Dürrenberger, R. Paro, M. Boutros and G. Merdes (2010): "A combined ex vivo and in vivo RNAi screen for notch regulators in *Drosophila* reveals an extensive notch interaction network," *Dev. Cell*, 18, 862–876.
- Schmich, F., E. Szczurek, S. Kreibich, S. Dilling, D. Andritschke, A. Casanova, S. H. Low, S. Eicher, S. Muntwiler, M. Emmenlauer, P. Rämö, R. Conde-Alvarez, C. von Mering, W.-D. Hardt, C. Dehio and N. Beerenwinkel (2015): "gespeR: a statistical model for deconvoluting off-target-confounded RNA interference screens," *Genome Biology*, 16, 220.
- Stuart, J. M., E. Segal, D. Koller and S. K. Kim (2003): "A gene-coexpression network for global discovery of conserved genetic modules," *Science*, 302, 249–255.
- Tsuda, K., H. Shin and B. Schölkopf (2005): "Fast protein classification with multiple networks," *Bioinformatics*, 21(Suppl 2), ii59–65.
- Vembu, S. and Q. Morris (2015): "An Efficient Algorithm to Integrate Network and Attribute Data for Gene Function Prediction," In: *Proceedings of the Pacific Symposium on Biocomputing*. pp. 388–399.
- Wang, L., Z. Tu and F. Sun (2009): "A network-based integrative approach to prioritize reliable hits from multiple genome-wide RNAi screens in *Drosophila*," *BMC Genomics*, 10, 220.
- Webber, W., A. Moffat and J. Zobel (2010): "A similarity measure for indefinite rankings," *ACM TOIS*, 28. DOI: 10.1145/1852102.1852106.
- Yu, J., S. Pacifico, G. Liu and R. L. Finley (2008): "DroID: the *Drosophila* Interactions Database, a comprehensive resource for annotated gene and protein interactions," *BMC Genomics*, 9, 461.
- Zhang, X., X. Dong, H. Wang, J. Li, B. Yang, J. Zhang and Z.-C. Hua (2014): "FADD regulates thymocyte development at the β -selection checkpoint by modulating Notch signaling," *Cell Death Dis*, 5, e1273.
- Zhu, X., Z. Ghahramani and J. Lafferty (2003): "Semi-supervised learning using gaussian fields and harmonic functions. ICML, 912–919.

Supplementary Material: The online version of this article offers supplementary material (DOI: <https://doi.org/10.1515/sagmb-2018-0033>).