Bioinformatics, 31, 2015, i197–i205
doi: 10.1093/bioinformatics/btv268
ISMB/ECCB 2015

Integrative random forest for gene regulatory network inference

Francesca Petralia, Pei Wang, Jialiang Yang and Zhidong Tu*

Department of Genetics and Genomic Sciences, Icahn School of Medicine at Mount Sinai, New York, NY 10029, USA

*To whom correspondence should be addressed.

Abstract

Motivation: Gene regulatory network (GRN) inference based on genomic data is one of the most actively pursued computational biological problems. Because different types of biological data usually provide complementary information regarding the underlying GRN, a model that integrates big data of diverse types is expected to increase both the power and accuracy of GRN inference. Towards this goal, we propose a novel algorithm named iRafNet: integrative random forest for gene regulatory network inference.

Results: iRafNet is a flexible, unified integrative framework that allows information from heterogeneous data, such as protein–protein interactions, transcription factor (TF)-DNA-binding, gene knock-down, to be jointly considered for GRN inference. Using test data from the DREAM4 and DREAM5 challenges, we demonstrate that iRafNet outperforms the original random forest based network inference algorithm (GENIE3), and is highly comparable to the community learning approach. We apply iRafNet to construct GRN in *Saccharomyces cerevisiae* and demonstrate that it improves the performance in predicting TF-target gene regulations and provides additional functional insights to the predicted gene regulations.

Availability and implementation: The R code of iRafNet implementation and a tutorial are available at: http://research.mssm.edu/tulab/software/irafnet.html

Contact: zhidong.tu@mssm.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Understanding the complex gene regulatory network (GRN) has an important role in current biomedical research. With the advent of high-throughput technologies such as transcriptomic and proteomic profiling, computational inference of GRN at genome scale are feasible and a large number of models have emerged (Karlebach and Shamir, 2008). Common challenges exist for inferring GRNs, including the non-linearity of regulatory relationships among genes, the incompleteness and noisiness characterizing genomic data, the presence of relatively fewer samples compared with the number of genes (the 'large *p* small *n* problem'). Therefore, successful GRN inference algorithms need to be sensitive to capture the non-linearity among genes, robust to avoid over-fitting and resilient to the limited number of samples. Among the various approaches developed so far, random forest has emerged as a strong player. As an ensemble learning algorithm, random forest performs extensive bootstrap sampling and random feature selection and relies on combining the outputs from a collection of non-linear learners to derive the final model. Such practice allows the delivery of excellent performance with moderate sample size requirement. Random forest has been used as an efficient and flexible tool to predict disease phenotypes (Bureau, 2005; Sun, 2009) and its utility has been demonstrated in a variety of biological applications (Yang *et al.*, 2010). Recently, Huynh-Thu *et al.* (2009) introduced GENIE3, a random forest based algorithm for the construction of GRN. The performance of GENIE3 was evaluated on the DREAM 4 *in-silico* size 100 challenge, the DREAM 4 *in-silico* multifactorial challenge, and the DREAM 5 network inference challenge. In particular, GENIE3 was the best performer in both the DREAM 4 *in-silico* multifactorial challenge (Greenfield *et al.*, 2010) and DREAM 5 network inference challenge (Marbach *et al.*, 2012). In order to better capture the direction of regulatory relationships, Maduranga *et al.* (2013) proposed a random forest based algorithm which infers GRN from time-series data. In particular, they demonstrated the superior performance of their algorithm over other existing methods such as dynamic Bayesian network and ordinary differential equations models. Although being very successful, these random forest based models derive GRN from a single data type, namely, the gene expression data. There is no direct way to integrate multiple genomics data such as protein–protein interactions and expression from perturbation experiments in current implementations.

The importance of integrating multiple genomics information has been well recognized by the research community. One reason of doing so is that different data types usually provide non-redundant information about regulatory relationships; e.g. protein–protein interactions are particularly informative for the topological structure of the network and the functions of neighboring genes (Deng *et al.*, 2004; Jeong *et al.*, 2001; Maslov and Sneppen, 2002); while gene expressions from perturbation and time-series experiments often provide more insights on the directionality or the causality of regulatory relationships. Multiple integrative models have been developed so far, such as Bayesian networks models (Bernard and Hartemink, 2005; Werhli and Husmeier, 2007; Zhu *et al.*, 2003, 2008), sparse structural equation models (Cai *et al.*, 2012; Logsdon and Mezey, 2010), and consensus techniques (Shojaie *et al.*, 2014; Yip *et al.*, 2010). The limitations of many existing methods are the linearity and normality assumptions often made on gene regulations. For example, due to the computational complexity of the models, gene dependence structure is often approximated via a linear regression, which may not perform well under the presence of higher-order interactions in the data. On the other hand, algorithms making no assumption on the linearity or normality can easily become computationally intractable when the number of genes significantly increases and, therefore, their applicability is limited to the construction of relatively small networks (Friedman and Goldszmidt, 1996; Imoto *et al.*, 2003; Kim *et al.*, 2004).

In this article, we propose iRafNet—a new algorithm in which different data types are integrated under a unified random forest framework. The key idea of iRafNet is to introduce a weighted sampling scheme within random forest to incorporate information from other source of data. Specifically, the model considers the expression of each gene as a function of the expression of other genes. For each node in the tree ensemble, instead of randomly sampling N genes from the entire gene set as done by GENIE3 (Huynh-Thu *et al.*, 2009), iRafNet samples genes (the potential regulators) according to the information provided by other data such as protein–protein interactions or expression data from perturbation experiments, so that genes supported by other data as potential regulators will be favorably sampled. By doing so, information embedded in other datasets is integrated into the network construction, while the effective search space of potential regulators is significantly reduced.

To demonstrate the advantage of integrating multiple data in the construction of GRN, we consider synthetic data from the DREAM 4 (Greenfield *et al.*, 2010) and the DREAM 5 (Marbach *et al.*, 2012) challenges, which have been used as gold test data sets for objectively comparing the performance of various GRN inference models. We show that iRafNet performs better than previous models in most considerations. As a real data application, we apply iRafNet to the inference of yeast GRN by integrating multiple public data sets. We show that our new approach has an improved performance in predicting transcription factor (TF) regulations and it also provides additional functional insights to the predicted gene regulations.

2 Methods

2.1 Overview of random forest-based GRN inference

Random forest is an ensemble algorithm based on learning a collection of decision trees. Each decision tree is learned independently on a group of bootstrapped samples. Starting from the root node containing all observations, each tree recursively splits observations into more homogeneous subsets. This allocation process is obtained by determining and applying certain splitting rules depending on the predictor variables. Specifically, at each node in the tree ensemble, N candidate predictors ($N < p$, with p being the number of all genes) are randomly sampled and the final predictor to be used for the splitting rule is chosen to minimize a certain cost function (Breiman *et al.*, 1984). Finally, outputs from individual trees are averaged to obtain the ultimate outcome.

Recently, Huynh-Thu *et al.* (2009) introduced GENIE3, a random forest based model which infers GRN by solving p independent regression problems. Specifically, the expression of a particular gene g_j is modeled as a function of the expression of other genes via random forest and genes that are strong predictors for the expression of g_j are considered as g_j's regulators. Genes are ranked based on the measure of importance resulting from random forest. The importance score $S_{k,j}$ of gene g_k for predicting gene g_j is defined as the total decrease in node impurity due to splitting the samples based on gene g_j (Breiman *et al.*, 1984). Let τ denote a node in the tree ensemble and let (τ_L, τ_R) denote its left and right children nodes. Then, the decrease in node impurity $I(\tau)$ from splitting τ based on gene g_k is defined as

$$I(\tau) = c_\tau v(\tau) - c_{\tau_L} v(\tau_L) - c_{\tau_R} v(\tau_R),$$

where $v(\tau)$, $v(\tau_L)$ and $v(\tau_R)$ are the variances of observations allocated to τ, τ_L and τ_R; while c_τ, c_{τ_L} and c_{τ_R} are the number of samples allocated to τ, τ_L and τ_R. Let $\mathbb{V}_k$ be the set of nodes in the tree ensemble that use g_k for the splitting rule. Then, the importance score $S_{k,j}$ of gene g_k for predicting gene g_j is calculated as the average of node impurities across all trees, i.e. $S_{k,j} = \sum_{\tau \in \mathbb{V}_k} I(\tau)/T$ where T is the number of trees.

2.2 iRafNet algorithm design

In this article, we introduce a weighted sampling scheme under the framework of random forest to allow the integration of heterogeneous data types. As shown in Figure 1, first, iRafNet processes supporting data to derive the prior belief of regulatory relationships among genes, then, it integrates such prior information to the main dataset via random forest to construct the final GRN. We consider different genomic data including gene expression data from steady-state experiments, time-series experiments, knockout experiments and other biological data such as protein–protein interactions. As shown in Figure 1, one data source is considered as main input data for random forest inference while other D datasets (supporting data) are utilized to derive prior information. iRafNet can be summarized in the following major steps, and detailed information regarding each step is provided in later sections:

Step A1. For the dth supporting data, with $d \in \{1, ..., D\}$, we derive scores $\{s^{d}_{k\to j}\}$ which measure the likelihood of regulatory events $\{g_k \to g_j\}$ based on the dth genomic data. Then, scores $\{s^{d}_{k\to j}\}$ are transformed into sampling weights $\{w^{d}_{k\to j}\}$, which are utilized in the next step for data integration;

Step A2. For each target gene g_j, with $j = \{1, ..., p\}$, we model the expression value of gene g_j as a function of the expression value of potential regulators via random forest using the main input dataset. Particularly, at each node, we randomly choose an integer $I \in \{1, ..., D\}$ with equal probability and sample N potential regulators according to weights $\{w^{I}_{k\to j}\}$;

Step A3. Potential regulators are ranked based on the importance score resulting from random forest (see Section 2.1).

In Step A1, sampling weights are derived from $\{s^{d}_{k\to j}\}$ which can be any score measuring how likely regulatory relationships $\{g_k \to g_j\}$ are based on the dth genomic data. In particular, when scores $\{s^{d}_{k\to j}\}$

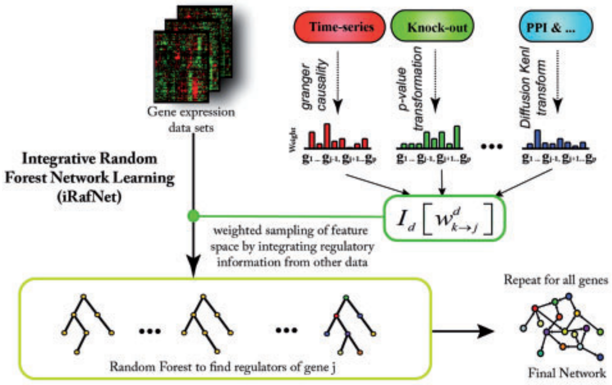
Gene expression data sets
Time-series
Knock-out
PPI & ...
granger causality
p-value transformation
Diffusion Kenl transform
Weight
g1 .. gj-1, gj+1 .. gp
g1 .. gj-1, gj+1 .. gp
g1 .. gj-1, gj+1 .. gp
Integrative Random Forest Network Learning (iRafNet)
$I_d\left[w^d_{k\to j}\right]$
weighted sampling of feature space by integrating regulatory information from other data
Random Forest to find regulators of gene j
Repeat for all genes
Final Network

Fig. 1. iRafNet schematics. For each gene $g_j \in \{1, ..., p\}$, we determine a ranked list of potential regulators via iRafNet. Based on each data $d \in \{1, ..., D\}$, we derive weights $\{w^{\mathrm{d}}_{k\to j}\}$ measuring the prior belief of regulatory relationships $\{g_k \to g_j\}$. Using expression data, we run random forest to find genes regulating g_j. At each node, instead of sampling a random subset of genes from the entire set of genes; we randomly choose an integer $l \in \{1, ..., D\}$ and we sample genes according to weights $\{w^{l}_{k\to j}\}$. The final network is derived by ranking potential regulators based on the random forest importance score

consist of *P*-values, weights are calculated as $w^{\mathrm{d}}_{k\to j} = (1/s^{d}_{k\to j} - 1)$. The procedure utilized to sample potential regulators in Step A2 is an extension of the one introduced by Amaratunga et al. (2008). As we described in Section 2.1, the importance score for a given predictor is derived by averaging the decrease in node impurities across all trees. Under the standard random forest algorithm, at each node, *N* potential regulators are proposed as candidates for the splitting rule via random sampling. When the number of potential regulators is large, relevant variables will less likely be sampled as candidates for establishing splitting rules. Consequently, for each tree, the total decrease in node impurity of relevant variables will be reduced. iRafNet overcomes this problem by sampling potential regulators according to prior information so that variables supported as relevant by other data will be more frequently sampled as candidates for the splitting procedure.

Generally, potential regulators of a target gene g_{j} consist of any other gene g_{k} with $k \neq j$; in some cases, the set of potential regulators may be set as a smaller subset based on certain prior knowledge. It is worth noting that Step A2 can be performed using either steady-state (Huynh-Thu *et al.*, 2009) or time-series gene expression data (Maduranga *et al.*, 2013). Because steady-state gene expression data usually contains more samples than time-series data, we use the former as main dataset for random forest construction and the latter to derive prior weights in Step A1.

2.3 Construction of sampling weights

One key step of iRafNet is to transform information embedded in supporting data into indicators of potential gene regulations. In this section, we focus on some commonly used data types which include steady-state gene expression, time-series gene expression, protein-protein interactions and gene expression from knockout experiments. For each data type, we provide detailed information on how weights are derived.

2.3.1 Weights based on protein–protein interactions

We use diffusion kernel to capture and transform the protein–protein interaction information (Lee *et al.*, 2005). We define the diffusion matrix as $F = e^{\mathrm{H}}$ with H being a $p \times p$ symmetric matrix with:

a. $h_{\mathrm{j,\,k}}$ equals one if genes g_{j} and g_{k} interact, and zero otherwise for $j \neq \mathrm{k}$(off-diagonal element)
b. $h_{\mathrm{k,k}} = -i_{\mathrm{k}}$, where i_{k} is the total number of interactions of gene g_{k} (diagonal element).

Given the diffusion matrix F, regulatory weights are constructed as $w^{\mathrm{PPI}}_{k\to j} = F_{\mathrm{k,j}}$, i.e. the element (k, j) of F. Because protein–protein interactions are bi-directional, the following identity holds $w^{\mathrm{PPI}}_{k\to j} = w^{\mathrm{PPI}}_{j\to k} = F_{\mathrm{k,j}} = F_{\mathrm{j,k}}$.

2.3.2 Weights based on time-series gene expression

In contrast to protein–protein interactions, time series data can provide information on the directionality of regulatory relationships. According to the definition of Granger causality, a gene g_{k} is causal for gene g_{j} if past values of g_{k} are predictive for future values of g_{j} (Lozano *et al.*, 2009). For a pair of genes $(g_{\mathrm{j}}, g_{\mathrm{k}})$, the expression value of gene g_{j} at future time $(t+1)$ is modeled as a linear function of the expression value of gene g_{k} at current time (t) and the significance of regulation $g_{\mathrm{k}} \to g_{\mathrm{j}}$ is tested via a standard *t*-test. The resulting *P*-values $\{P^{\mathrm{TS}}_{k\to j}\}$ are, then, utilized to derive sampling weights as follows $w^{\mathrm{TS}}_{k\to j} = (1/p^{\mathrm{TS}}_{k\to j} - 1)$.

2.3.3 Weights based on knockout data

We denote $x^{\mathrm{wt}}_{\mathrm{j}}$ the expression of gene g_{j} in wild-type condition, and $x^{\mathrm{KO}}_{k\to j}$ the expression of gene g_{j} after knocking out gene g_{k}. Similarly to time-series data, weights $w^{\mathrm{KO}}_{k\to j}$ are derived as $w^{\mathrm{KO}}_{k\to j} = (1/P^{\mathrm{KO}}_{k\to j} - 1)$ with $P^{\mathrm{KO}}_{k\to j}$ being the *P*-value testing the regulatory relationship $g_{\mathrm{k}} \to g_{\mathrm{j}}$ based on knockout data. Specifically, $P^{\mathrm{KO}}_{k\to j}$ is computed via a two-tailed *t*-test on the difference $(x^{\mathrm{wt}}_{j} - w^{\mathrm{KO}}_{k\to j})$. In real world applications, only a small subset of genes is generally knocked-out and only some regulatory relationships could be inferred by this approach. To overcome this problem, we propose a method that imputes causal relationships by borrowing information from other knocked-out genes. Let K be the set of knocked-out genes; then, missing causal relationships are inferred based on the following steps:

Step B1. For any gene g_{k} with $g_k \in \mathrm{K}$, we derive *P*-values $\{P^{\mathrm{KO}}_{k\to j}\}$ and we consider the regulatory event $g_{\mathrm{k}} \to g_{\mathrm{j}}$ true if $P^{\mathrm{KO}}_{k\to j}$ is smaller than 0.01;

Step B2. For each pair of genes $(g_{\mathrm{h}}, g_{\mathrm{k}})$, a measure of similarity is obtained as follows:

- We derive the sets of genes which are functionally related to genes g_{h} and g_{k} based on knockout data. In particular,
  a. when both g_{h} and g_{k} belong to K, we compute $(E_{\mathrm{h}}, E_{\mathrm{k}})$, the sets of genes affected by knocking-out genes $(g_{\mathrm{h}}, g_{\mathrm{k}})$, and $(\mathrm{C_h}, \mathrm{C_k})$, the sets of knocked-out genes which affect genes $(g_{\mathrm{h}}, g_{\mathrm{k}})$;
  b. otherwise, we compute only $(\mathrm{C_h}, \mathrm{C_k})$, the sets of knocked-out genes which affect genes $(g_{\mathrm{h}}, g_{\mathrm{k}})$;
- Letting $J(A, B)$ be the Jaccard index between sets A and B, the similarity measure $\mathrm{G_{h,k}}$ between genes g_{h} and g_{k} is derived as
  a. $\mathrm{G_{h,k}} = (J(E_{\mathrm{h}}, E_{\mathrm{k}}) + J(\mathrm{C_h}, \mathrm{C_k}))/2$;
  b. $\mathrm{G_{h,k}} = J(\mathrm{C_h}, \mathrm{C_k})$;

Step B3. For genes $\{g_s, s \notin \mathrm{K}\}$, we impute missing weights as follows:

$$\frac{w^{\mathrm{KO}}_{s\to j} = \sum_{\ell\in \mathrm{K}} \mathrm{G}_{\mathrm{s},\ell} w^{\mathrm{KO}}_{\ell\to j}}{\sum_{\ell\in \mathrm{K}} \mathrm{G}_{\mathrm{s},\ell}} \qquad (1)$$

Table 1. Comparison between iRafNet, GENIE3 and the best performer in the challenge in terms of the AUC and AUPR for experiments from the DREAM 4 *in-silico size 100* challenge

Method	GENIE3		iRafNet		Pinna *et al.* (2010)	
	AUC	AUPR	AUC	AUPR	AUC	AUPR
Net 1	0.864	0.338	0.901 (0.870,0.932)	0.552 (0.548,0.556)	0.914	0.536
Net 2	0.748	0.309	0.799 (0.765,0.834)	0.337 (0.333,0.341)	0.801	0.377
Net 3	0.782	0.277	0.835 (0.798,0.873)	0.414 (0.410,0.418)	0.833	0.39
Net 4	0.808	0.267	0.847 (0.813,0.881)	0.421 (0.417,0.426)	0.842	0.349
Net 5	0.720	0.114	0.792 (0.751,0.832)	0.298 (0.294,0.301)	0.759	0.213

For iRafNet, 95% confidence intervals are provided under the corresponding AUC and AUPR values in brackets.

According to Equation (1), missing weights $\{w^{\mathrm{KO}}_{s\to j}\}$ are found via a weighted average of scores $\{w^{\mathrm{KO}}_{\ell\to j}\}_{\ell\in\mathrm{K}}$ with weights proportional to the corresponding similarity measures. The similarity measure derived in Step B3 is based on the assumption that when two genes are functionally related they are more likely to be affected by a similar set of genes (Peleg et al., 2010).

2.4 iRafNet implementation and availability

iRafNet package is an extension of the original package Random Forest available in R Cran (Liaw and Wiener, 2002). Specifically, the original random forest code was modified to allow weighted random sampling. The computational complexity of iRafNet is the same as GENIE3, i.e. $O(pTNn\log(n))$, where n is the sample size, p *is* the number of genes, N is the number of variables sampled at each node and T is the number of trees. iRafNet can be easily parallelized as the network inference consists of p independent sub-problems. iRafNet requires users to specify T, the number of trees and N, the number of potential regulators to be sampled at each node. As the number of trees increases, the tree ensemble generally provides more accurate results. For this reason, the number of trees is usually chosen sufficiently large ($T > 500$). The choice of N is less straightforward, since large values of N usually result in predictions with high-bias; while low values result in predictions with high-variance (Breiman *et al.*, 1984). However, it is a common practice to set $N = r^{1/2}$ with r being the number of potential regulators (either the number of genes -1, or a customized smaller number) (Shi and Horvath, 2006).

3 Results

3.1 Application of iRafNet to the DREAM 4 and DREAM 5 network inference challenges

To evaluate the performance of iRafNet, experiments from the DREAM 4 (Greenfield *et al.*, 2010) *in-silico size 100* and the DREAM 5 (Marbach *et al.*, 2012) challenge are considered. Both challenges provide gene expression and other biological data such as time-series and perturbation data. Each team participating in the challenge had access to all different data types, and the exact data to be used for GRN inference was a decision made by each team. For each synthetic data, iRafNet was compared with GENIE3. In particular, GENIE3 results were obtained directly from The Dream Project website (http://www.the-dream-project.org/).

The DREAM 4 *in-silico size 100* challenge consists of five networks involving $p = 100$ genes. For all experiments, sampling weights were computed from P-values as described in *Method*s section. For each network, knockout data and time-series expression were provided. In particular, time-series data consisted of ten different experiments with 21 time points each; while knockout data included wild-type gene expression and gene expression after knocking out each one of the p genes. iRafNet infers the five networks from time-series data utilizing knockout data as prior information. Because knockout data are based on one experiment, P-values could not be computed via standard t-test. Alternatively, we derived P-values $P^{\mathrm{KO}}_{k\to j}$ as $P^{\mathrm{KO}}_{k\to j} = 2(1 - \Phi(|x^{\mathrm{KO}}_{k\to j} - x^{\mathrm{wt}}_{j}|/\sigma_{\mathrm{j}}))$, where Φ is the distribution function of the standard normal distribution with mean of 0 and SD of 1; and σ_{j} is the SD of the expression of gene g_{j} based on all samples in the knockout data. Following the original algorithm GENIE3, random forest parameters are set as $T = 1000$ and $\mathrm{N} = r^{1/2}$, with r being the total number of genes minus one, i.e. $r = 99$.

Table 1 compares GENIE3 and iRafNet in terms of the area under the receiver operating characteristic curve (AUC) and the precision-recall curve (AUPR). Both GENIE3 and iRafNet provide a ranking of regulatory relationships based on importance scores resulting from random forest. For different thresholds on the importance scores, we computed receiver operating characteristic and precision-recall curves using the R package 'ROCR'. For each network, knockout data and time-series gene expression are provided. GENIE3 is implemented using only time-series gene expression; the best performer utilizes only knockout data; while iRafNet integrates both knockout data and time-series gene expression. For iRafNet, we provide 95% confidence intervals for both AUC and AUPR values. The confidence interval for AUC was derived using 'ci.auc()' function in R package 'pROC' (Robin *et al.*, 2011), which implements a method developed by DeLong *et al.* (1988). The confidence interval for AUPR was computed using a logit transformation approach (Boyd *et al.*, 2013).

As shown in Table 1, our algorithm achieves better predictive performance compared with GENIE3 in terms of both AUC and AUPR for all the five networks involved in the DREAM 4 challenge. Furthermore, as shown in Table 1, iRafNet performs similarly to the best performer in the DREAM4 *in-silico size 100* challenge, which inferred GRN from knock-out data alone (Pinna *et al.*, 2010).

The DREAM 5 challenge consists of four networks with one being derived by *in–silico* simulation and the other three being obtained experimentally from three species. We report results concerning Network 1 and Network 3, involving 1643 and 4511 genes, respectively. We decided to focus on these two networks mainly because limited knockout data are provided for Network 2 and Network 4. In addition, Network 1 and Network 3 are the networks where teams participating in the challenge scored the highest predictive performance. Incomplete knockout data, time-series expression and steady-state gene expression are available for both Network 1 and Network 3. The total number of knocked out genes was 19 and 15 for Network 1 and Network 3, respectively; and missing knockout relationships were inferred using the method described in *Method*s section. As mentioned in *Method*s section, since time-series data is usually characterized by small sample sizes,

Table 2. Comparison between iRafNet, GENIE3, Meta 1 and COMMUNITY in terms of the AUC and AUPR with corresponding 95% confidence intervals for synthetic experiments from the DREAM 5 challenge

Method	Data	Network 1		Network 3	
GENIE3	Exp	0.815 (0.807,0.823)	0.291 (0.289,0.295)	0.617 (0.607,0.627)	0.093 (0.091,0.106)
Meta 1	KO	0.736 (0.727,0.745)	0.276 (0.274,0.277)	0.614 (0.604,0.624)	0.087 (0.085,0.089)
Community	Exp, KO, TS	0.809 (0.801,0.817)	0.327 (0.326,0.329)	0.65 (0.639,0.660)	0.09 (0.090,0.105)
iRafNet	Exp, KO	0.812 (0.804,0.82)	0.364 (0.361,0.364)	0.638 (0.629,0.651)	0.113 (0.110,0.115)
	Exp, KO, TS	0.813 (0.804,0.819)	0.364 (0.360,0.366)	0.641 (0.63,0.651)	0.112 (0.109,0.114)

when both time-series and steady-state expression are available we suggest using the former to compute sampling weights in Step A1 and the latter to implement Step A2. Therefore, for networks involved in the DREAM 5 challenge, iRafNet estimated GRN from steady-state expression utilizing knockout and time-series data as prior information. The list of potential regulators was pre-determined by the DREAM 5 and contained $r = 195$ genes under Network 1 and $r = 334$ genes under Network 3.

Similarly to the comparison procedure used by the DREAM5 challenge, for each model, receiver operating characteristic and precision-recall curves were computed considering the top 100 000 regulations. Table 2 compares iRafNet, GENIE3 and COMMUNITY in terms of AUC and AUPR. COMMUNITY is a more generalized ensemble model, which derives a consensus network by combining the results of all 35 teams participating in the challenge (Marbach *et al.*, 2012). The DREAM 5 challenge provides predicted networks for all teams participating in the challenge; based on this information, we compute confidence intervals of the area under the ROC and precision recall curve for all models and include the results in Table 2. Although COMMUNITY outperformed each single team participating in the challenge, iRafNet results in better AUPR than both GENIE 3 and COMMUNITY. Specifically, the AUPR of iRafNet is ~9% larger than that of COMMUNITY and ~21% larger than that of GENIE3 for Network 1. For Network 3, the AUPR of iRafNet is ~11% larger than that of COMMUNITY and GENIE3. The three methods scored similar performance in terms of AUC; however, as shown in Figure 2, iRafNet outperforms the other two methods in the most critical region of the ROC curve characterized by small values of false positive rates.

It is worth noting that the DREAM4 challenge is based on *in-silico* data and provides complete knockout data. As illustrated in Table 1, for this challenge, the performance of Pinna *et al.* (2010) is comparable to iRafNet. However, with data input more similar to real-world applications (i.e. only a small set of genes are knocked out), the model that estimates GRN solely from knockout data may perform poorly. As an illustration, we reported in Table 2 the results from a team participating in the DREAM 5 challenge who considered only knockout data for GRN inference. This algorithm is referred to as Meta 1 by Marbach *et al.* (2012) and resembles the best performing algorithms from the DREAM 4 challenge (Greenfield *et al.*, 2010). As shown in Table 2, iRafNet outperforms this algorithm in terms of both AUC and AUPR.

3.2 Application of iRafNet to GRN inference in *Saccharomyces cerevisiae*

To further demonstrate the applicability of iRafNet to real biological data, we apply iRafNet to construct GRN in *S. cerevisiae*. In particular, we considered knockout data from Hu *et al.* (2007), protein–protein interactions from three databases [BioGRID (Chatr-Aryamontri *et al.*, 2012), DIP (Xenarios *et al.*, 2000) and MINT (Zanzoni *et al.*, 2002)], time-series data (Spellman *et al.*, 1998), and steady-state expression for $p = 3665$ genes from Zhu *et al.* (2008). iRafNet constructed GRN from steady-state gene expression data and used protein–protein interactions, knockout gene expression and time-series gene expression as prior information. For testing purposes, iRafNet was compared to GENIE3 which estimates GRN from gene expression data alone.

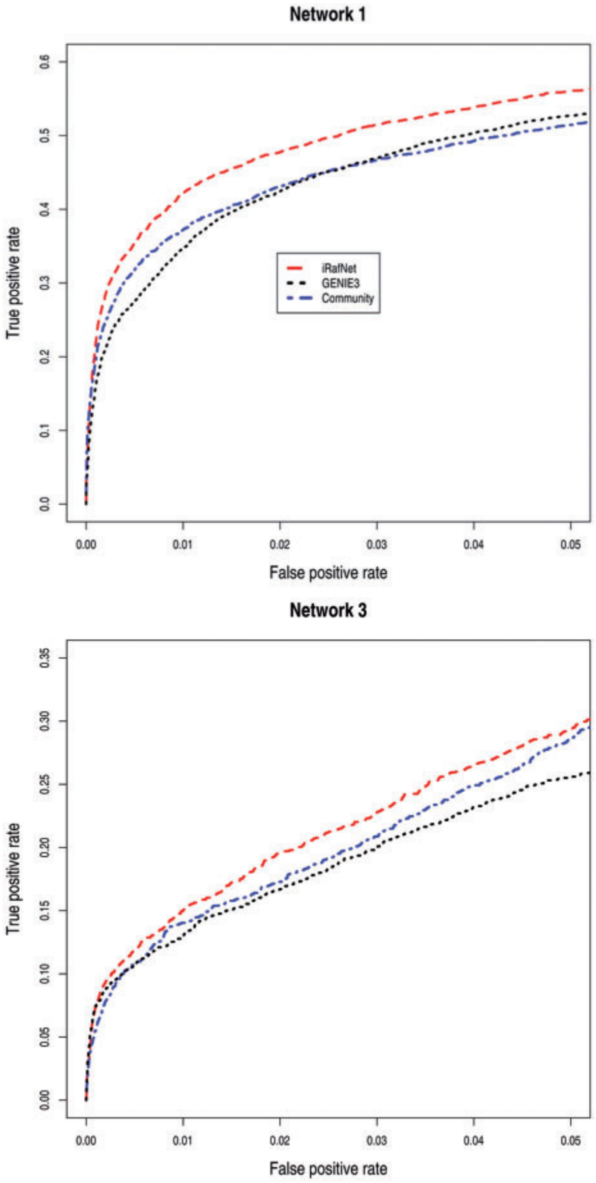

Fig. 2. ROC curves resulting from various methods for the estimation of Network 1 and Network 3 from the DREAM 5 challenge. Community is an ensemble algorithm which derives a consensus network by integrating predictions of GENIE3 and the other 34 teams participating in the challenge. iRafNet infers GRN by integrating all knockout, time-series and steady-state gene expression data

Table 3. Networks output from GENIE3 and iRafNet

	No of edges	No of directed edges	No of shared edges	No of shared directed edges	No of enriched GO terms	
					0.05	0.01
GENIE3	156 359	200 000	102 501	126 009	51	44
iRafNet	163 886	200 000	102 501	126 009	61	51

For both GENIE3 and iRafNet, we consider the set of 200 000 highest scored directed edges, referred to as $\mathbb{D}$. As shown, the number of unique undirected edges $a - b$ was 156 359 and 163 886 for GENIE3 and iRafNet, respectively. For each method, we show the number of GO categories with significant enrichment for different P-value thresholds (0.05 and 0.01).

For the time-series data (Spellman *et al.*, 1998), we selected time-course gene expression data from cdc28 cell cycle arrest which consists of 17 time points. For the knockout data, we considered P-values provided by Hu *et al.* (2007). The total number of knockout genes from this experiment was 169 and missing causal relationships were inferred as described in *Methods* section. Further details on how sampling weights were derived are provided in *Methods* section. The random forest parameters were set as $T = 1000$ and $\mathrm{N} = r^{1/2}$, respectively, where $r = 3664$. In order to evaluate the performance of both models, the following criteria were considered: GO terms enrichment and prediction of TF regulations.

3.2.1 iRafNet results in more enriched GO categories

We compared networks resulting from GENIE3 and iRafNet based on GO terms enrichment. We focused on 58 GO Slim terms obtained from the Saccharomyces Genome Database (Cherry *et al.*, 1998) containing from 20 to 200 genes. For each model, importance scores for all regulatory relationships were derived and we focused on the 200 000 highest scored regulatory relationships (in the DREAM5 challenge, only the first 100 000 predicted regulations were considered for the competition, we relax the cutoff so that true predictions are less likely to be excluded due to this parameter setting). For each GO term, the enrichment score was computed via a one-sided Kolmogorov-Smirnov test (Aravind *et al.*, 2005). Specifically, for each GO term, we considered every undirected edge between all pairs of genes contained in the GO category and calculated the Kolmogorov-Smirnov statistics based on importance scores of undirected edges resulting from each method.

The importance score of each undirected edge $(g_s - g_k)$ was defined as the mean between importance scores of the two directed edges $(g_s \rightarrow g_k)$ and $(g_s \leftarrow g_k)$. The Kolmogorov-Smirnov statistics reflects the degree to which a gene ontology (GO) category is overrepresented at the top of the ranked list of importance scores. Table 3 shows the number of GO terms with significant enrichment. As shown, iRafNet results in more enriched GO categories than the original algorithm which relies on a single data type. Supplementary Table S1 in the supplementary material shows the list of GO categories and corresponding P-values under each method.

3.2.2 iRafNet better predicts TF regulations

In this section, we evaluate the ability of our model to predict TF regulations. For this purpose, we consider results from Lee *et al.* (2002) which used chromatin immuno-precipitation techniques to detect TF-gene interactions and provided P-values of regulations between 72 TFs and 3644 genes. Based on these P-values, we derive the 'true' network. Specifically, an edge between TF g_k and gene g_j $(g_k \rightarrow g_j)$ is considered true if the corresponding P-value is smaller than 0.01; while the edge of the opposite direction $(g_j \rightarrow g_k)$ is used as negative control. Table 4 shows the AUC and AUPR for GENIE3 and iRafNet. Specifically, for iRafNet, we used different set of weights derived from either knockout, time-series or protein–protein interactions data, as well as used all these weights simultaneously.

Table 4. Prediction performance of TF regulations

Method	Data	AUC	AUPR
GENIE3	Expression	0.547 (0.537,0.566)	0.542 (0.537,0.548)
iRafNet	Multiple weights	0.624 (0.613,0.636)	0.565 (0.561,0.569)
	Expression and KO	0.657 (0.645,0.673)	0.567 (0.562,0.574)
	Expression and TS	0.543 (0.528,0.557)	0.536 (0.530,0.541)
	Expression and PPI	0.574 (0.562,0.591)	0.557 (0.551,0.561)

For each model, the AUC and the AUPR and corresponding 95% confidence intervals are reported.

Overall, iRafNet results in better predictive performance than GENIE3. The best predictive performance is achieved when sampling weights were obtained from knockout data alone. This result is not completely surprising since knockout data is considered one of the most informative data for inferring regulatory relationships (Marbach *et al.*, 2012). The slightly less optimal performance resulting from integrating all data types may be due to the inconsistency among different datasets (e.g. some datasets could have less optimal quality). This result suggests that a careful selection of input data is very important regardless the underlying algorithms.

We perform another comparison based on the ability to predict TF regulations. Let R_{t_h} be the top t_h directed regulations with the largest importance scores. Then, we derive $R_e \subseteq R_{t_h}$, defined as the set of directed edges belonging to set R_{t_h} which were found to be significant ($P < 0.01$) by Lee *et al.* (2002) and $R_d \subseteq R_e$, defined as the set of directed edges for which the opposite direction is not included in set R_{t_h}. A higher cardinality of R_e indicates that the algorithm is more capable of revealing the regulatory relationships as detected by Lee *et al.* (2002); while the higher cardinality of R_d indicates the algorithm is more accurate in excluding the 'wrong' directed edges. As shown in Table 5, for different values of t_h, iRafNet consistently identifies larger R_e and R_d than GENIE3. Supplementary Table S2 in the supplementary material provides a list of regulations identified by iRafNet but not recovered by GENIE3. Multiple regulations are supported by independent experiments, suggesting the validity of the predictions. For example, Chou *et al.* (2006) showed that Dig1 forms a complex with Ste12, Tec1 or Dig2. Dig1 knockout caused up-regulation of Fus1 gene expression, the effect was particularly significant when both Dig1 and Dig2 were knocked out (Chou *et al.*, 2006). As another example, Santangelo and Tornow (1990) showed that the transcription of ADH1 was sensitive to GCR1 disruption, which is consistent with our prediction.

4 Discussion

In this article, we develop iRafNet, a unified framework based on random forest which constructs GRNs by integrating information from multiple data types. Specifically, information from different data sources is used to derive a series of weights, which, then, are utilized for sampling potential regulators during the tree construction. This weighting scheme provides multiple benefits compared with the sampling procedure adopted by the standard random

Table 5. Prediction of TF regulations using different cutoffs

	$t_k = 60\ 000$		$t_k = 80\ 000$		$t_k = 1000\ 000$	
Method	*ne*	*nd*	*ne*	*nd*	*ne*	*nd*
iRafNet	64	49	85	64	103	77
GENIE3	28	7	34	11	44	13

Cardinality (n_e, n_d) of sets (R_e, R_d). Let R_{t_k} be the set of the first t_h directed edges with highest scores, with $t_h = \{60\ 000;\ 80\ 000;\ 100\ 000\}$. Then, $R_e \subseteq R_{t_h}$ is defined as the set of directed edges found to be significant ($P < 0.01$) by Lee *et al.* (2002), while $R_d \subseteq R_e$ is defined as the set of directed edges in R_e for which the opposite direction is not included in set R_{t_k}.

Table 6. GRN inference performance using different numbers of trees in random forest learning

Number of trees	500	1000	5000
AUC	0.810	0.815	0.813
AUPR	0.290	0.291	0.294

Network 1 from the DREAM 5 challenge is considered and performance measured in terms of AUC and the AUPR.

forest. In the original random forest algorithm, at each node, sampling is done by randomly selecting N potential regulators and, among them, the predictor maximizing the decrease in node impurity is chosen for the splitting rule. This strategy may be less effective when the number of informative predictors is small compared with the total number of genes; in such case, informative predictors have small chance of being chosen as candidates for the splitting rules. This will cause the importance scores of informative predictors to be lower compared to the cases when they have larger chance of being chosen as potential regulators.

Simply increasing the number of trees may not resolve the aforementioned problem. In order to better illustrate this point, we applied GENIE3 to infer Network 1 from the DREAM 5 challenge using different tree numbers. As shown in Table 6, the predictive performance is relatively unchanged when we increase the tree number from 500 to 5000. This result is not surprising since a larger tree number is not going to significantly change the average score of a feature across all trees when T is already large. Our approach of weighted sampling allows potential regulators identified as informative by other genomic data to be more favorably selected. As a result, the corresponding importance score will be more favorably measured compared with other regulators with no prior support. The importance of appropriate prioritizing relevant features in high-dimensional learning has been recognized by multiple works. For example, a recent work showed that using a bootstrap ranking to derive a robust prioritization of SNPs could significantly increase the performance of disease risk prediction (Manor and Segal, 2013). Our work provides another demonstration on this concept and points out a direction to further improve the prediction performance under random forest framework.

Besides reducing the curse of dimensionality, iRafNet is expected to increase accuracy and coverage of GRN inference through integrating diverse information using its weighting scheme. Of course, this may not always be the case, as we have seen that when datasets of different quality and different origin are combined, the overall performance may actually reduce. In this work, we only illustrate the integration of limited types of genomic data. However, iRafNet can be utilized to integrate almost any data type as long as its information can be transformed into prior knowledge regarding potential regulatory relationships.

When compared with other integrative models like Bayesian network, the advantage of iRafNet relies on its computational efficiency and the robust predictive performance resulting from its non-parametric nature. In fact, a limitation of many existing methods such as Bayesian networks are the linearity and normality assumptions often made to reduce the computational complexity of the algorithm. Although non-parametric Bayesian networks have been proposed in literature (Friedman and Goldszmidt, 1996; Imoto *et al.*, 2003; Kim *et al.*, 2004), they are computationally intensive and generally require very large sample size. The computational complexity of Bayesian networks is amplified by the difficulties encountered in parallelizing the algorithm. In contrast to Bayesian networks, iRafNet can flexibly model non-linearity and higher-order interactions while being efficient on large-scale applications as it can be easily executed with parallelization. Moreover, the performance of iRafNet is in general robust to the number of trees in the random forest model upon our investigation.

One open question in the estimation of GRN is whether the best performance is achieved by single models or by COMMUNITY methods which derive a consensus network combining results from different models (Shojaie *et al.*, 2014; Yip *et al.*, 2010). Recently, Marbach *et al.* (2012) claimed that COMMUNITY methods perform better than single methods. Despite their predictive performance, COMMUNITY methods remain computationally demanding algorithms which require the estimation of many different models followed by the estimation of the consensus network. For the DREAM5 challenge, our algorithm was compared to COMMUNITY, which integrated the predictions of 35 teams participating in the challenge (Marbach *et al.*, 2012). As we showed in the manuscript, iRafNet is comparable to if not better than the COMMUNITY and, therefore, represents an efficient alternative (for the datasets used in this work, iRafNet can finish in less than an hour on a cluster running in parallel).

In our implementation of iRafNet, we treated time-series and steady-state gene expression data separately, one for deriving prior information regarding potential regulators, while the other as main input for random forest construction. Alternatively, they could be combined as a single dataset and be used as input for a single source-based random forest algorithm. Although combining the two datasets would increase the sample size and generally provide greater power in detecting regulations; problems may arise when the sample sizes of the two data sets are imbalanced. In such situation, the construction of tree ensemble may be largely driven by the dataset with more samples while the signals embedded in the smaller dataset may be concealed. This would be less an issue when the two datasets are used separately to train different models. For this reason, we decided to integrate time-series and steady-state gene expression in two stages. A rigorous test should be considered to evaluate and compare the performance of either combining datasets or treating them separately in a two-stage learning procedure.

It is also worth noting that prior weights may be computed using alternative methods. For example, in Section 2.3.3, instead of using the Jaccard index, other methods may be utilized to measure the similarity between genes. Because the 'best' methods for prior weights calculation may depend on the data inputs, users are encouraged to explore different options when calculating the prior information.

Additional work is needed to improve the way that different data types are integrated. In its current implementation, prior biological data, such as protein–protein interactions and expression from perturbation experiments, are equally weighted. This characteristic is appealing when different source of data provide equally important information about regulatory relationships. However, in real world applications, some experiments may be less informative about the network structure and an equal weighting procedure may penalize the overall performance. To overcome this problem, as future work, we consider to design a new model where the contribution of each data source is estimated and appropriately weighted within the unified random forest framework.

Acknowledgements

We would like to thank the authors of GENIE3 (Huynh-Thu *et al*., 2009) for providing important information about software implementation.

Funding

This work was supported in part through the computational resources and staff expertise provided by the Department of Scientific Computing at the Icahn School of Medicine at Mount Sinai. Z.T. received financial support from Berg Pharma as a consultant and support from National Institutes of Health (U01AG046170) and Leducq Foundation Transatlantic Networks of Excellence Program grant. F.P. and P.W. are partially supported by National Institutes of Health grant (SUB-R01GM108711). P.W. is also supported by National Institutes of Health grant (R01GM082802, sub-P01CA53996 and SUB-CA160034).

Conflict of Interest: none declared.

References

Amaratunga,D. *et al*. (2008) Enriched random forests. *Bioinformatics*, **24**, 2010–2014.

Aravind,S. *et al*. (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc. Natl. Acad. Sci. USA*,**102**, 15545–15550.

Bernard,A. and Hartemink,A.J. (2005) Informative structure priors: joint learning of dynamic regulatory networks from multiple types of data. In: Campos,L.M.D. and Castellano,J.G. (eds.) *Pacific Symposium on Biocomputing*, World Scientific, New Jersey, **Vol. 10**, pp. 459–470.

Boyd,K. *et al*. (2013) Area under the precision-recall curve: point estimates and confidence intervals. In: *Machine Learning and Knowledge Discovery in Databases*. Springer, 2013. pp. 451–466.

Breiman,L. (2001) Random forests. *Mach. Learn*, **45**, 5–32.

Breiman,L. *et al*. (1984) *Classification and Regression Trees*. CRC press, Boca Raton.

Bureau,A. *et al*. (2005) Identifying SNPs predictive of phenotype using random forests. *Genet. Epidemiol.*, **28**, 171–182.

Cai,X. *et al*. (2012) Inference of gene regulatory networks with sparse structural equation models exploiting genetic perturbations. *PLoS Comput. Biol.*, **9**, e1003068.

Chatr-Aryamontri,A. *et al*. (2012) The BioGRID interaction database: 2013 update. *Nucleic Acids Res.*, **41** (Database issue), 23.

Cherry,J.M. *et al*. (1998) SGD: Saccharomyces genome database. *Nucleic Acids Res.*, **26**, 73–79.

Chou,S. *et al*. (2006) Regulation of mating and filamentation genes by two distinct Ste12 complexes in *Saccharomyces cerevisiae*. *Mol. Cell. Biol.*, **26**, 4794–4805.

DeLong,E.R. *et al*. (1988) Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics*, **44**, 837–845.

Deng,M. *et al*. (2004) Mapping gene ontology to proteins based on protein–protein interaction data. *Bioinformatics*, **20**, 895–902.

Friedman,N. and Goldszmidt,M. (1996) Discretizing continuous attributes while learning Bayesian networks. In: *Proceedings of the 13th* International Conference on Machine Learning (ICML), 1996, pp. 157–165.

Greenfield,A. *et al*. (2010) DREAM4: combining genetic and dynamic information to identify biological networks and dynamical models. *PloS One* **5**, e13397.

Hu,Z. *et al*. (2007) Genetic reconstruction of a functional transcriptional regulatory network. *Nat. Genet.*, **39**, 683–687.

Huynh-Thu,V.A. *et al*. (2009) Inferring regulatory networks from expression data using tree-based methods. *PloS One* **5**, pii: e12776.

Imoto,S. *et al*. (2003) Bayesian network and nonparametric heteroscedastic regression for nonlinear modeling of genetic network. *J. Bioinform. Comput. Biol.*, **1**, 231–252.

Jeong,H. *et al*. (2001) Lethality and centrality in protein networks. *Nature*, **411**, 41–42.

Karlebach,G. and Shamir,R. (2008) Modelling and analysis of gene regulatory networks. *Nat. Rev. Mol. Cell Biol.*, **9**, 770–780.

Kim,S. *et al*. (2004) Dynamic Bayesian network and nonparametric regression for nonlinear modeling of gene networks from time series gene expression data. *Biosystems*, **75**, 57–65.

Lee,H. *et al*. (2005) Diffusion kernel-based logistic regression models for protein function prediction. *OMICS*, **10**, 40–55.

Lee,T.I. *et al*. (2002) Transcriptional regulatory networks in *Saccharomyces cerevisiae*. *Science*, **298**, 799–804.

Liaw,A. and Wiener,M. (2002) Classification and regression by randomForest. *R. News*, **2**, 18–22.

Logsdon,B.A. and Mezey,J. (2010) Gene expression network reconstruction by convex feature selection when incorporating genetic perturbations. *PLoS Comput. Biol*.**6**, e1001014.

Lozano,A.C. *et al*. (2009) Grouped graphical Granger modeling for gene expression regulatory networks discovery. *Bioinformatics*, **25**, i110–i118.

Maduranga,D. *et al*. (2013) Inferring gene regulatory networks from time-series expressions using random forests ensemble. In: *Pattern Recognition in Bioinformatics*. Springer, Berlin, pp. 13–22.

Manor,O. and Segal,E. (2013) Predicting disease risk using bootstrap ranking and classification algorithms. *PLoS Comput. Biol.*, **9**, e1003200.

Marbach,D. *et al*. (2012) Wisdom of crowds for robust gene network inference. *Nat. Methods*, **9**, 796–804.

Maslov,S. and Sneppen,K. (2002) Specificity and stability in topology of protein networks. *Science*, **296**, 910–913.

Peleg,T. *et al*. (2010) Network-free inference of knockout effects in yeast. *PLoS Comput. Biol.*, **6**, e1000635.

Pinna,A. *et al*. (2010) From knockouts to networks: establishing direct cause-effect relationships through graph analysis. *PloS One*, **5**, e12912.

Robin,X. *et al*. (2011) pROC: an open-source package for R and S+ to analyze and compare ROC curves. *BMC Bioinformatics*, **12**, 77.

Santangelo,G.M. and Tornow,J. (1990) Efficient transcription of the glycolytic gene ADH1 and three translational component genes requires the GCR1 product, which can act through TUF/GRF/RAP binding sites. *Mol. Cell. Biol.*, **10**, 859–862.

Shi,T. and Horvath,S. (2006) Unsupervised learning with random forest predictors. *J. Comput. Graph. Stat.*, **15**, 118–138.

Shojaie,A. *et al*. (2014) Inferring regulatory networks by combining perturbation screens and steady state gene expression profiles. *PloS One*, **9**, e82393.

Spellman,P.T. *et al*. (1998) Comprehensive identification of cell cycle–regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell*, **9**, 3273–3297.

Sun,Y.V. (2009) Multigenic modeling of complex disease by random forests. *Adv. Genet.*, **72**, 73–99.

Werhli,A.V. and Husmeier,D. (2007) Reconstructing gene regulatory networks with Bayesian networks by combining expression data with multiple sources of prior knowledge. *Stat. Appl. Genet. Mol. Biol.*, **6**, article 15.

Xenarios,I. *et al*. (2000) DIP: the database of interacting proteins. *Nucleic Acids Res.*, **28**, 289–291.

Yang,P. *et al*. (2010) A review of ensemble methods in bioinformatics. *Curr. Bioinformatics*, **5**, 296–308.

Yip,K.Y. *et al.* (2010) Improved reconstruction of in silico gene regulatory networks by integrating knockout and perturbation data. *PloS One,* **5**, e8121.

Zanzoni,A. *et al.* (2002) MINT: a Molecular INTeraction database. *FEBS Lett.,* **513**, 135–140.

Zhu,J. *et al.* (2003) An integrative genomics approach to the reconstruction of gene networks in segregating populations. *Cytogenet. Genome Res.,* **105**, 363–374.

Zhu,J. *et al.* (2008) Integrating large-scale functional genomic data to dissect the complexity of yeast regulatory networks. *Nat. Genet.*, **40**, 854–861.