Adaptive least absolute regression network analysis improves genetic network reconstruction by employing prior knowledge

James Yong-A-Poi[1,*], Eugene van Someren[2], Domenico Bellomo[1] and Marcel Reinders[1]

[1]Information and Communication Theory group, Faculty of EEMCS, Delft University of Technology, 2600 GA, Delft, The Netherlands

[2]Molecular Design and Informatics, N.V. Organon, part of Schering-Plough Corporation, 5340 BH, Oss, The Netherlands

ABSTRACT

Motivation: The inference of genetic regulatory networks from time-series gene expression data has been performed with linear models. The challenge of this inference problem is solving an underdetermined system in which the number of genes is far greater than the number of measurements. LARNA (least absolute regression network analysis) tackles this problem by employing the LASSO (least absolute shrinkage and selection operator) technique for simultaneous estimation and variable selection. However, with the availability of different data sources (e.g. literature network, transcription factor binding information) capturing different parts of the true network, integration of this type of prior knowledge with expression data into LARNA can potentially improve the variable selection and eventually the reconstructed network.

Results: We propose to integrate prior knowledge into LARNA by modifying the L_1 LASSO penalty. The performance of our scheme is evaluated on synthetic and real datasets. The evaluation focused on part of the network for which no prior knowledge was available. Results indicate that the integration of prior knowledge improved in reconstructing part of the network without prior knowledge.

1 INTRODUCTION

The behavior of a cell largely depends on the complex interactions between genes, proteins and metabolites. The activity (a.k.a. expression) of a gene is regulated by special proteins, called transcription factors (TFs) that bind to the promoter of the gene. Activated genes produce specific mRNA molecules that are in turn translated into proteins. Proteins perform all kinds of functions, e.g. they can act as TFs for other genes, as enzymes catalyzing metabolic reactions or as structural elements in the cell. An example of a network of molecular interactions is depicted in Figure 1.

The complete cellular system can be simplified by considering only gene, protein or metabolic interactions. Gene networks focuses on gene interactions and are phenomenological models of how the expression level of each gene is influenced by the expression level of all other genes (Brazhnik, 2002). Here, connections between genes are characterized as either direct or indirect (see caption to Figure 1 for details).

Gene networks are important, because they might contain valuable information for the pharmaceutical and biotechnology industries to design novel drugs for complex diseases. They are able to describe how cells react to external influences in a concise way by gene connections, which implicitly capture regulatory mechanisms at the protein and metabolite space (Gardner, 2005).

Linear models have been proposed to infer genetic regulatory networks from time-series gene expression data in many papers (e.g. Chen, 1999; D'Haeseleer, 1999; van Someren, 2006; Cosentino, 2007). Linear models assume that the future expression level of a gene is a linear combination of the past expression levels of all genes. This assumption is only valid near equilibrium, because the Hartman-Grobman theorem states that the behavior of a (biological) nonlinear system around a steady-state is similar to a linearised system (Grobman, 1959; Hartman, 1960).

Despite the fact that linear models are already a strong simplification of the underlying biological system, they are still underdetermined as the number of genes is far greater than the number of measured time points. The LASSO method (Tibshirani, 1996) is an elegant shrinkage method, which performs feature selection and parameter estimation simultaneously. It has previously been applied to overcome the dimensionality problem during the inference of a 100 gene osteoblast differentiation network from eleven time points (van Someren, 2006). The approach was called LARNA.

In this paper, we propose to integrate prior knowledge into the linear regression model LARNA (van Someren, 2006) by adjusting its shrinkage method such that it can utilize partial information about the topology of the true network. A drawback of traditional LARNA when trying to infer the right regulator from a group of genes that contains highly correlated expression profiles, is that it may arbitrarily select one of these genes. Prior knowledge about the likelihood of each of these genes as the potential regulator of the current target should aid LARNA in selecting the correct gene by taking this prior knowledge into account.

Prior knowledge can be defined as any additional information about connections between genes. Transcription factor binding information identifies the binding of transcription factors to promoters of genes. Literature networks are build from searching articles for co-occurrences of pair of genes. Many resources can serve as prior knowledge and it is important to integrate them.

The main goal of this paper is to evaluate whether our approach of adding prior knowledge to LARNA has a positive effect in discovering the original network. We evaluate the effect on the parts of the networks for which no prior knowledge was given. Moreover, we evaluate whether our approach is able to infer the original network when the prior knowledge contains errors.

Evaluation of our new approach, called adaptive LARNA (aLARNA), on real data is difficult, because the underlying true gene network is unknown. Therefore, we first estimate its performance on synthetic datasets. In this way, we can compare the inferred networks with the original network and include (part of) the

*To whom correspondence should be addressed.

original network as prior knowledge. We also evaluate how its performance is influenced when incorrect connections are put into the prior knowledge.

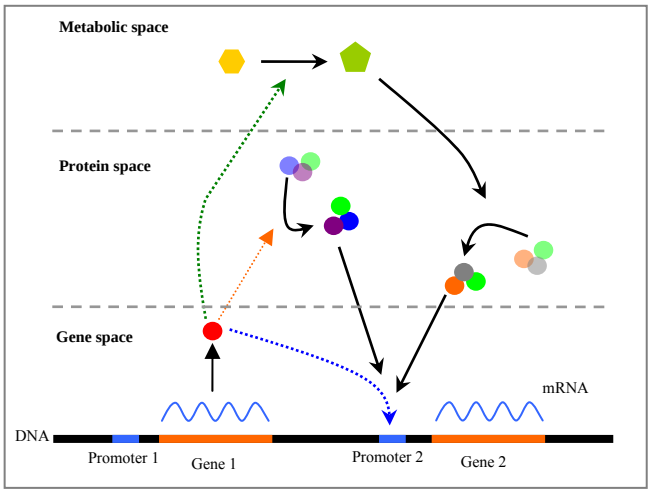


Fig. 1. Gene networks model how the expression level of each gene is influenced by the expression level of all other genes at the gene space. Gene connections between genes are either direct (physical) or indirect (influential). Direct connections between two genes occur when the product of one gene acts as a transcription factor and binds to the promoter of the other gene in order to regulate its expression level (blue arrow). Indirect connections between genes are mediated by protein and metabolic reactions and occur in two ways: (1) the product of one gene activates a complex of transcription factors, which regulate the expression level of the other gene (orange arrow); (2) the product of one gene catalyzes a metabolic reaction in which the transformed metabolite activates a set of TFs and these bind to the upstream region of another gene (green arrow). Protein networks model the interaction between proteins at the protein space. Protein interactions refer to the formation of protein complexes and protein modification by signaling enzymes. Metabolic networks model the chemical transformations of metabolites at the metabolic space.

We also evaluate aLARNA on a real dataset consisting of microarray measurements that follow murine stem cells undergoing osteoblast differentiation (development towards bone cells). Here, the underlying network is unknown and we used a literature network to benchmark aLARNA's performance. Here, two experimental setups were designed; one to use the literature network as prior knowledge and the other to use transcription factor binding information as prior knowledge.

We first introduce the LARNA model and the integration of prior knowledge. Next, results on synthetic and real datasets are presented and some discussion points are presented. Furthermore, we discuss methods to evaluate the performance. Finally, we conclude and give directions for future work.

2 APPROACH

A linear genetic network model assumes that current gene expression levels can be predicted by a linear combination of the gene expression levels at the previous time point. In matrix notation this means that output matrix **Y** can be predicted by multiplication of connectivity matrix **A** with input matrix **X**, as follows:

$$\hat{\mathbf{Y}} = \mathbf{A} \cdot \mathbf{X} + \boldsymbol{\varepsilon}, \tag{1}$$

where $\boldsymbol{\varepsilon} \in \mathbb{R}^m$ represents the error, $\mathbf{X} \in \mathbb{R}^{p\times(n-1)}$, $\mathbf{Y} \in \mathbb{R}^{m\times(n-1)}$, $\mathbf{A} \in \mathbb{R}^{m\times m}$, p is the number of regulatory genes, m is the number of target genes and n is the number of time points. Input matrix **X** consists of rows of gene expression profiles, with x_{jk} the expression of the j-th gene at the k-th time point. Output matrix **Y** consists of rows of gene expression profiles, with y_{ik} the expression of the i-th gene at the $(k+1)$-th time point. Connectivity matrix **A** consists of coefficients a_{ij}, which represents the influence of gene j on gene i, and are only non-zero when gene j has an activating ($a_{ij} > 0$) or an inhibiting ($a_{ij} < 0$) effect on gene i.

LARNA infers the linear genetic network by estimating matrix **A** from input matrix **X** and output matrix **Y**. The coefficients of matrix **A** are found row-wisely by minimizing the penalized sum of squares via LASSO (Tibshirani, 1996), as follows:

$$\begin{aligned}\hat{\mathbf{a}}_i &= \arg\min_{\mathbf{a}_i}\Big(\frac{1}{n}\sum_{k=1}^{n}(\hat{y}_{i(k-1)} - y_{ik})^2 + \lambda\sum_{j=1}^{m}|a_{ij}|\Big)\\ &= \arg\min_{\mathbf{a}_i}\Big(\frac{1}{n}\sum_{k=1}^{n}(\sum_{j=1}^{m} a_{ij}x_{j(k-1)} - y_{ik})^2 + \lambda\sum_{j=1}^{m}|a_{ij}|\Big)\\ &\text{for } i = 1\ldots m.\end{aligned} \tag{2}$$

Here $\mathbf{a}_i$ represents the coefficients of the i-th row of matrix **A** and $\lambda \geq 0$ is a complexity parameter that controls the amount of shrinkage. For increasing values of λ, the effect of the L_1 LASSO penalty $\sum|a_{ij}|$ causes coefficients to become smaller and consequently to become exactly zero. Thus LASSO performs simultaneous parameter estimation and variable selection. Here, we estimated matrix **A** by adopting the algorithm from (Grandvalet, 1998), in which LASSO is proven to be equivalent to adaptive ridge regression and this is solved by an EM-algorithm.

Traditionally, the L_1 LASSO penalty applies equal penalties to each coefficient (see Supplement S.1), which is undesirable when the likelihood of being zero is known beforehand for some particular coefficients. Therefore, we propose adaptive Least Absolute Network Analysis (aLARNA), which exploit this knowledge by assigning different weights to these coefficients in the L_1 penalty, as follows:

$$\hat{\mathbf{a}}_i = \arg\min_{\mathbf{a}_i}\Big(\frac{1}{n}\sum_{k=1}^{n}(\hat{y}_{ik} - y_{ik})^2 + \lambda\sum_{j=1}^{m}\theta_{ij}|a_{ij}|\Big), \tag{3}$$

where $\theta_{ij} \in (0,\infty]$ represent the weights. The weights give the coefficients different penalties to express the a priori influences. A strong indication of an existing influence of gene j on gene i receives a small weight such that the coefficient is less penalised and therefore has a higher change to remain in the model. Coefficients that are likely to have no influence receive larger weights in order to shrink them more strongly towards zero and as a result more likely to be removed from the model.

3 RESULTS

We evaluate the performance of aLARNA by means of synthetic and real datasets. As the underlying networks of synthetic datasets

are known this presents an ideal situation for evaluation, because we can compare the overlap between the networks inferred by aLARNA with the underlying networks. Methods section 5.3 describes the used performance measures in more detail. The synthetic datasets are used to study the effect of different amounts of prior knowledge and different percentages of errors introduced in the prior knowledge. The real dataset is used to test aLARNA in a realistic situation with severe undersampling, significant measurement noise and an unknown underlying model.

3.1 Synthetic datasets

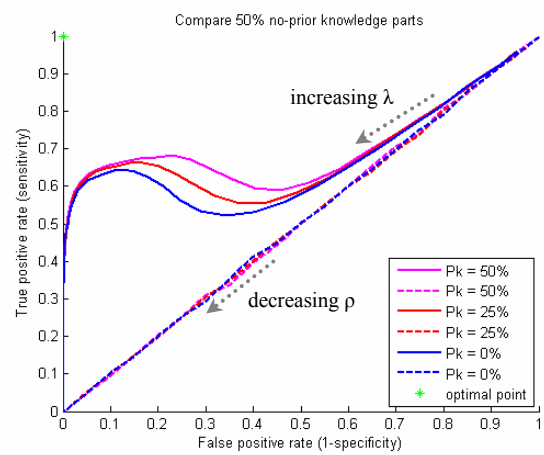
3.1.1 More prior knowledge leads to better sparse networks We only present results on the artificial scale-free gene networks from Mendes *et al* (2003). (CenturySF). Experiments have shown that regulatory networks are sparse and have a scale-free topology (Albert, 2005; Featherstone, 2002; Jeong, 2000). Results on other artificial gene networks were similar and can be found in the Supplement S.5. Methods section 5.1 describes in detail how the synthetic data was generated.

This experiment evaluates how aLARNA performs when increasing amounts of prior knowledge (0%, 25% and 50% of the underlying network) are given. The performance of aLARNA is compared to a random reconstruction algorithm (RRA) that randomly assigns connections between genes (see Methods section 5.7). For both approaches, the performance measures are calculated on the part of the network about which aLARNA and RRA received no prior knowledge. We call this the no-prior knowledge part, which represents 50% of the whole network structure (see Methods section 5.6).

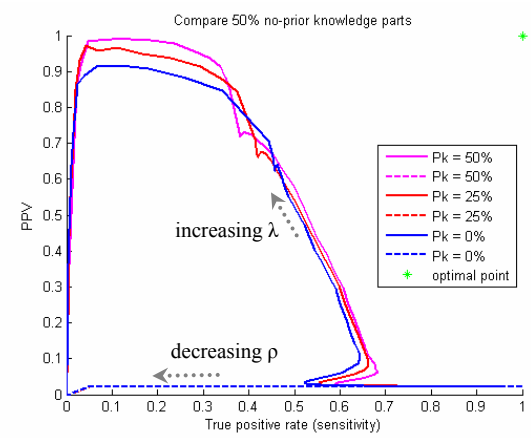
aLARNA and RRA were applied on the 50 CenturySF datasets (Methods section 5.1.1) with values of λ between 10^{-4} and 10^{7} and ρ between 0.00 and 1.00, respectively. Figure 2a shows the performance of both approaches by plotting the specificity (SP) against the sensitivity (SE). The better a method performs the closer its performance will be to the upper-left corner (SP = 1 and SE = 1). As expected, the performance of RRA remains on the 45 degrees line regardless of the amount of prior knowledge. Clearly, the use of prior knowledge on one part of the network does not affect the performance of RRA on the no-prior knowledge part. aLARNA returns a fully connected network when λ is very small and therefore SE = 1 and SP = 0. aLARNA sets more coefficients to zero as λ increases and this has effects on the sensitivity and specificity. One, if the correct coefficients are set to zero, the specificity increases. Two, if the wrong coefficients are set to zero, the sensitivity decreases. Clearly, LARNA (i.e. aLARNA with 0% prior knowledge) outperforms RRA especially for large values of λ.

When specificity is between 0.50 and 0.90, aLARNA performs increasingly better for increasing values of prior knowledge. In this interval, we observe that aLARNA correctly sets 50% to 90% of the coefficients to zero at a sensitivity of 60%. We observe that the sensitivity of the pink line is higher in this interval, which indicates a better performance with more prior knowledge.

Figure 2b shows the sensitivity against the positive predictive value for the same conditions as in Figure 2a. Clearly, the PPV of RRA is constantly 2% regardless the amount of prior knowledge. aLARNA's performance is not always better with more prior knowledge. Initially, different percentages of prior knowledge do not effect the performance (SE = [0.70, 1.00]) and performance is close to that of RRA. In the next interval SE = [0.40, 0.70], different percentages of prior knowledge do not have a significant effect on aLARNA's performance. However, when λ is very large and sparse networks are inferred, aLARNA with 50% prior knowledge reaches a PPV close to 1.00 and outperforms LARNA (PPV = 0.90). From Figure 2b, we learn that prior knowledge is most helpful when sparse networks are inferred by aLARNA.



(a)



(b)

Fig. 2. Scale-free gene networks: specificity versus sensitivity (a) and sensitivity versus PPV (b). 0% (blue lines), 25% (red lines) and 50% (pink lines) of the underlying network were given as prior knowledge to aLARNA (solid lines) and RRA (dashed lines). aLARNA was evaluated with a λ between 10^{-4} and 10^{7}.

3.1.2 Erroneous prior knowledge In the previous section, aLARNA receives faultless prior knowledge to reconstruct the original connectivity matrix. Unfortunately, such prior knowledge is unlikely to exist when we apply aLARNA in a realistic setting. Therefore, we evaluated its performance when errors are introduced into the prior knowledge. The experiment is set up with 50%

prior knowledge and with 0%, 5%, 10% and 20% errors, which are randomly introduced into the prior knowledge.

Figure 3 shows that the performance of aLARNA decreases with more errors as we would expect. It is interesting to observe that with erroneous prior knowledge aLARNA still managed to achieve a good PPV score when it returns very spare networks. In Figure 3b, aLARNA has a PPV score above 0.95 with 5% and 10% errors in the prior knowledge when SE = 0.10, while LARNA only scores a PPV of 0.90 when SE = 0.10 (Figure 2b).

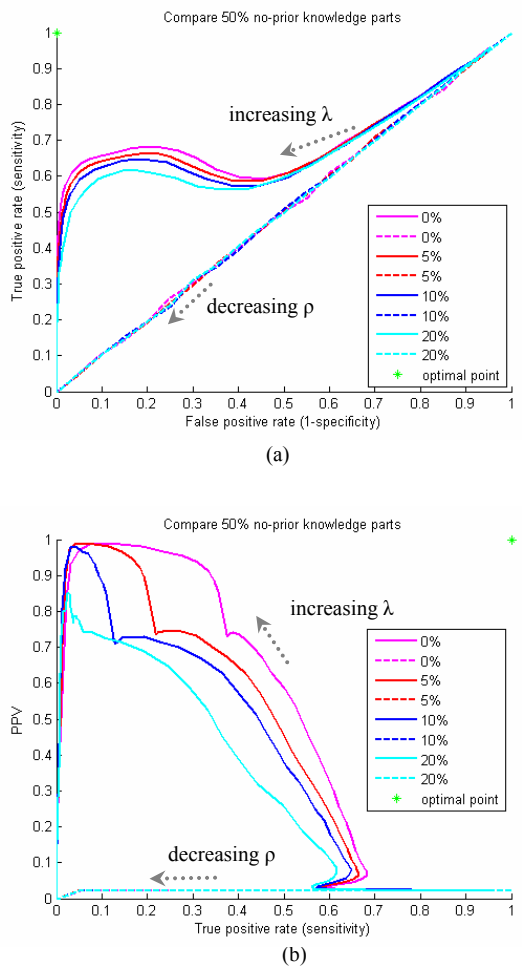


Fig. 3. Performance evaluation of erroneous prior knowledge. (a) sensitivity versus specificity; (b) and sensitivity versus PPV. Solid lines represents the aLARNA's performance with 0% (pink), 5% (red), 10% (blue) and 20% (cyan) errors in the prior knowledge. RRA's performance is shown by dashed lines.

3.2 Real dataset

Results from two experimental setups are presented to understand the effects of different types of prior knowledge (see Methods section 5.2).

3.2.1 Experimental setup 1 aLarna is applied to the osteoblast differentiation dataset consisting of 334 genes measured over 52 time points. As no underlying network is available, a literature network was constructed to serve as benchmark as well as (partial) prior knowledge (see Methods section 5.4.2). This is a challenging task, because the literature network is not precisely the underlying network (incomplete and erroneous). Here we also test aLARNA with different percentages of prior knowledge.

aLARNA is evaluated with a λ between 10^{-5} and 10^{15}. Contrary to the synthetic dataset, the osteoblast differentiation dataset was not normalized. Therefore, a larger λ was needed in order to infer sparse networks. We were unable to choose a smaller λ to infer a fully connected network, because of computation limitations in Matlab. Therefore, a large part of the performance (SP = [0.00, 0.90] and SE = [0.43, 1.00]) could not be calculated (Figure 4). Since we know from the previous experiments on synthetic data that aLARNA with prior knowledge performs best when it returns sparse networks, it should not be a problem that we fail to evaluate aLARNA's performance for very small λ values.

.

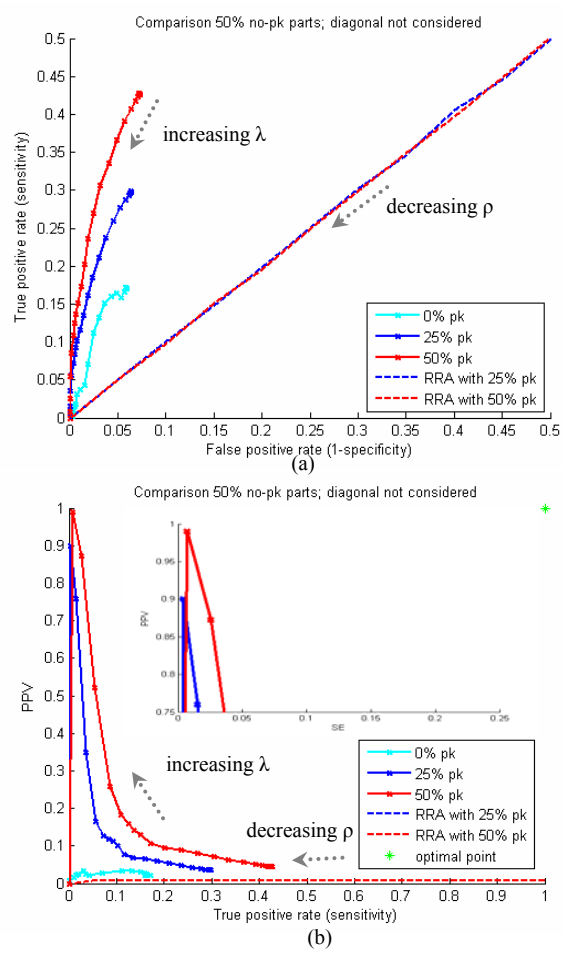


Fig. 4. Performance evaluation on real dataset with prior knowledge (experimental setup 1). Sensitivity versus specificity (a) and sensitivity versus PPV (b). 0% (cyan lines), 25% (blue lines) and 50% (red lines) of the literature network were given as prior knowledge to aLARNA (solid lines) and RRA (dashed lines). aLARNA was evaluated with a λ between 10^{-5} and 10^{15}.

In the interval of SE = [0.90, 1.00], aLARNA achieves a higher SE with more prior knowledge. So, aLARNA has put fewer coefficients incorrectly to zero with more prior knowledge. Figure 4b shows that aLARNA performs poorly without prior knowledge (solid cyan line) though better than RRA. With 25% prior knowledge the performance improves significantly (solid blue line). A PPV of 0.90 is reached with 25% prior knowledge. Moreover, a PPV close to 1.00 is reached with 50% prior knowledge (solid red line)

3.2.2 Experimental setup 2 In this section, aLarna is applied time profiles of 334 target genes and 335 regulatory genes (see Methods section 5.2.2). This setup is even more challenging than the previous one, because the benchmark is different from the prior knowledge. Here, we integrate transcription factor binding information and gene expression data and evaluate with a literature network.

Figure 5 shows poor performance from aLARNA with 25% and 50% prior knowledge. There is very little difference between the performance of RRA and aLARNA. Thus, aLARNA does not improve with transcription factor binding information as prior knowledge.

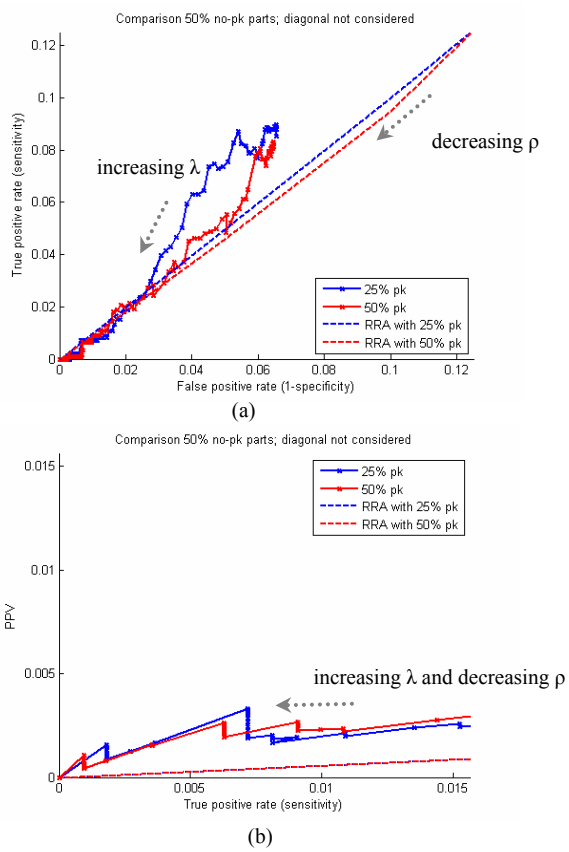


Fig. 5. Performance evaluation on real dataset with prior knowledge (experimental setup 2). Sensitivity versus specificity (a) and sensitivity versus PPV (b). 0% (cyan lines), 25% (blue lines) and 50% (red lines) of the literature network were given as prior knowledge to aLARNA (solid lines) and RRA (dashed lines). aLARNA was evaluated with a λ between $10^{\wedge -5}$ and $10^{\wedge 15}$.

4 DISCUSSION

The true network of a real biological system reveals itself trough e.g. gene expression data, transcription binding information, protein-protein interaction data, biological literature. Integration of these different data sources is needed, but the problem arises when the reconstructed network has to be evaluated. Generally, researchers use literature networks as the ground truth, because it offers a generic network encompassing all gene connections. However, there are two problems with literature networks. One, in literature two genes may be connected when e.g. they share similar protein sequences even though it is not necessarily a genetic connection. Two, an inferred gene connection not contained in the literature network requires costly gene knock out experiments in order to validate if the inferred connection is indeed false or a new discovered connection (Husmeier, 2003).

We experienced poor results when we attempted to integrate gene expression data with transcription binding information as prior knowledge (Section 3.2.2). The problem is caused in several ways. Firstly, the literature network is constructed by searching through all biological literature and not only osteoblast specific literature. This makes the literature network less specific to osteoblast differentiation. Secondly, by restricting the regulatory genes to only transcription factors even makes the inferred osteoblast networks more specific, because most general regulatory genes are omitted. Thirdly, the gene expression profiles of the regulatory genes are not predictive to the gene expression profiles of the target genes. It is reasonable that TF activity profile is very far from its gene expression profile. Fourthly, transcription factor binding information contains like the literature network more general connections between genes. Therefore, it adds little improvement to aLARNA.

The determination of the weights θ_{ij} is not straightforward (Supplement S.4). On the one hand, the weight assigned to coefficients known to be non-zero has most effect on how large the range of the sensitivity is when a PPV score close to 1.00 (Figure S4 and S5). On the other hand, the weight assigned to coefficients known to be zero has smaller effects on the performance of aLARNA, especially when errors are present in the prior knowledge (Figure S5). It seems that different weights have small effects on the performance of aLARNA when it infers sparse networks. Nevertheless, when prior knowledge is available weighting the coefficients will always result in better performance than LARNA.

5 METHODS

5.1 Data sets

5.1.1 Synthetic datasets To evaluate the performance of aLARNA, artificial gene expression datasets were generated based on the artificial gene networks proposed by Mendes *et al.* (2003). The artificial gene networks are defined by sets of ordinary differential equations (ODEs, see Supplement S.2) and the Jacobians (see Supplement S.3) at the steady-state. The change in mRNA concentration of the *i*-th gene over time is described by an ODE, which describes the rate of synthesis and degradation of the corresponding mRNA concentration, as follows:

$$\frac{dG_i}{dt} = s(G_1, ..., G_n) - b(G_i), \quad (4)$$

where G_i represents the abundance of the mRNA of gene i, $s(G_1,...,G_n)$ the rate of transcription and $b(G_i)$ the mRNA breakdown rate.

The set of ODEs were numerically solved in Matlab to generate synthetic datasets, each containing n time profiles (one per gene). Each time profile consists of 5 subseries of 30 interpolated time points. Each subseries was created with initial concentrations all near the steady-state concentrations, i.e. they were drawn from a uniform distribution with the steady-state concentrations as means, as follows:

$$G_i(0) = U(-0.25, 0.25) + G_i(t_{st}) \quad (5)$$

where $G_i(t_{st})$ represents the mRNA abundance of gene i at the steady state.

Furthermore, to each generated expression value Gaussian noise was added. The noise had zero mean and a standard deviation that was 5% of the standard deviation of the entire time profile, as follows:

$$G_i^{noise}(t) = G_i(t) + N(0.00, 0.05 * \delta_i) \quad (6)$$

where δ_i represents the standard deviation of the i-th gene's time profile.

Before network inference, each gene expression profile was normalized by subtracting the mean and dividing by the standard deviation of the gene's entire expression profile, as follows:

$$G_i^{norm}(t) = \frac{G_i^{noise}(t) - \mu_i^{noise}}{\delta_i^{noise}} \quad (7)$$

where μ_i^{noise} and δ_i^{noise} represent the mean and standard deviation of the i-the gene's noisy time profile, respectively.

Mendes *et al.* (2003) proposed several different series of artificial genetic networks for evaluation of inference algorithms. We used the Century series, because it contains networks with three different topologies (random, scale-free and small-world) and a large number of network realizations per topology set. For each topology, the Century series consists of a set of 50 networks with 100 genes and 200 connections. The first set, CenturyRND, contains random networks (Erdös, 1959) in which each node has an equal probability to be connected to any other node in the network. The degree distribution follows a Poisson distribution, which indicates that each node has a degree close to the average degree of the network. The nodes of the scale-free networks (Albert, 2000) in the second set, called CenturySF, follow a power-law distribution. A few nodes have a significantly high degree, whereas most nodes have only a few connections. The third set of networks, called CenturySW, have small-world topologies (Watts, 1998), where each pair of nodes has a small averaged distance. An overview of the main properties of the generated datasets that were used is depicted in Table 1.

According to Mendes *et al* (2003). it is meaningless to use a single network per topology due to the fact that the connectivity is randomly generated. Therefore, the Century series contains 50 networks per topology. Thus, for each network (150 in total) we generated a synthetic dataset with 100 genes and 150 time points (in 5 subseries) and corrupted it with measurement noise followed by a gene-wise normalization step.

Table 1. Properties of the employed artificial gene expression datasets. Three different sets of synthetic datasets were employed for performance evaluation, i.e. random Erdös networks, scale-free networks and small-world networks.

Name	Topology	Models	Genes (n)	Links	Datasets
CenturyRND	Erdös	50	100	200	50
CenturySF	Scale-free	50	100	200	50
CenturySW	Small-world	50	100	200	50

5.1.2 Real dataset The biological example involves a gene expression dataset measured from murein stem-cells undergoing osteoblast differentiation stimulated by BMP2 (Bone Morphogenetic Protein 2). Embryonic cells treated with BMP2 differentiate into bone cells. BMP2 belongs to a group of growth factor (BMPs) responsible for cell type differentiation, skeleton formation and development of the nervous system.

The osteoblast dataset consist of gene expression data of 45101 genes measured with the Affymetrix GeneChip Mouse Genome 430 2.0 Array (Mouse430-2). Murine stemcells were grown in standard differentiation medium with BMP2 which was refreshed every 3 days, consequently the amount of active BMP2 peaked at time points 0, 73 and 145 hours. Measurements (mixture of triplos and duplos) were taken at exponentially increasing time intervals after each refreshment covering a total of 34 unique time points during a time window between 0 and 200 hours. Hence, we linearly interpolated the time points to obtain equal sampling. Data was normalized using Rosetta Resolver. Genes were ranked according to how significantly they changed over time using an ANOVA with each time point as a separate group.

Time profiles from this dataset were not normalized, because this would influence the significance of the time profiles. The highest ranked genes have the most informative time profiles. Thus, normalization would remove this information from the time profiles.

5.2 Experimental setups with real data

5.2.1 Experimental setup 1 In the first experimental setup, the literature network serves as benchmark and as prior knowledge. This is similar to the experiments with synthetic data where the benchmark and prior knowledge come from the same resource, namely the Jacobian.

We initially selected the first 500 time profiles representing the target and regulatory genes, but ended with 334 time profiles due to mapping the time profiles to CoPub identifiers. The CoPub identifiers are necessarily in order to make use of the literature network as benchmark.

5.2.2 Experimental setup 2 In the second experimental setup, the benchmark and prior knowledge come from different resources. The benchmark is the literature network and the prior knowledge comes from transcription factor binding information. Here, the weights θ_{ij} contain information about the affinity of binding calculated by the Segalscore (see Supplement S.6)

We selected 321 target genes and 335 regulatory genes based on the ranking. The number of target genes was chosen in a similar way as in the previous section. However, 13 genes were omitted which were transcription factors. Furthermore, the number of regulatory genes was determined by selecting those genes from the osteoblast dataset, which have CoPub identifiers and are contained as transcription factors with a binding motif in the TRANSFAC® 11.2 database. In order to integrate the prior knowledge most effectively, we decided to select regulatory genes acting as transcription factors.

5.3 Performance measures

Discovering the correct regulators of target genes is already a difficult task, thus we did not focus on the strength and type of the regulators. The performance is measured by calculating the sensitivity, specificity and the positive predictive value (table 2). The sensitivity measures how well the algorithm can retrieve all the nonzero coefficients of the true network. The specificity measures how well the algorithm can retrieve all the zero coefficients of the true network. The positive predictive value measures how much of the inferred nonzero coefficients are true nonzero coefficient of the true network. Since networks are often sparse, it is important that the few inferred non-zero coefficients are indeed non-zero coefficient of the true network.

Table 2. Performance measures. True positive (TP) occurs when $J_{ij} \neq 0$ and $A_{ij} \neq 0$. True negative (TN) is when $J_{ij} = 0$ and $A_{ij} = 0$. False-positive (FP) is when $J_{ij} = 0$ and $A_{ij} \neq 0$. False-negative (FN) is when $J_{ij} \neq 0$ and $A_{ij} = 0$.

Sensitivity	Specificity	PPV
true positive rate	false positive rate	positive predictive value
$\frac{TP}{TP + FN}$	$\frac{TN}{TN + FP}$	$\frac{TP}{TP + FP}$

5.4 Benchmarks

5.4.1 Synthetic benchmark The Jacobians calculated at the steady-states serve as benchmarks for evaluation. They represent the connectivity matrices J of the original networks and are compared to the reconstructed connectivity matrices A inferred by aLARNA. The search for non-zero coefficients is already hard, so we binarize both the Jacobians and inferred networks with a threshold of 10^{-3} and lose all information about strength and type of regulation.

5.4.2 Literature benchmark The literature network **L** is constructed by means of the tool CoPub Mapper (Alako, 2005). It searches publications in MEDLINE to calculate how often pairs of genes are mentioned together by returning two measures: (1) a literature count, which represents the number of co-occurrences of a pair of genes; (2) a R-scaled value, which specifies how specific a pair of genes is.

We binarized the literature network and the inferred networks before calculating the performance. Positive connections between genes pairs are assigned if their literature counts ≥ 3 and if their R-scaled values ≥ 37. The literature network contains no information about the direction of regulation and therefore its connectivity matrix will be symmetric. For example, a positive connection will be represented by two ones in the connectivity matrix: l_{ij}=1 and l_{ji}=1. The inferred networks are thresholded with a value of 10^{-3} as described before. Furthermore, they are symmetrized in order to fit the literature network. Finally, the diagonal entries of both networks are omitted, as the literature network contains no self-feedback loops, and this could influence the results.

5.5 The values of the weights

Coefficients in the experiments with synthetic datasets and real dataset are penalized with weights θ respectively chosen from the sets {0.50, 1.00, 1.50} and {0.05, 1.00, 1.95}, dependent on whether the coefficient should receive prior knowledge and if so the kind of the prior knowledge (connection or non-connection). More details about the determination of the sets are in Supplement S.4. The weights in the weighted L_1 penalty are set as follows:

- When we include only part of the Jacobian or literature network as prior knowledge, then some coefficients will have to be weighted as having no prior knowledge. We penalize such a coefficient a_{ij} with a neutral weight $\theta_{ij} = 1.0$. In this way a_{ij} will be penalized indifferently to a connection or non-connection in the original network.
- When $J_{ij} \neq 0$ we know gene *i* is regulated by gene *j* and thus we penalize the corresponding coefficient a_{ij} with a small $\theta_{ij} = 0.50$. For the literature network we used: $\theta_{ij} = 0.05$.
- When $J_{ij} = 0$ we know gene *i* is not regulated by gene *j* and thus we penalize the corresponding coefficient a_{ij} with a large $\theta_{ij} = 1.50$. For the literature network we used: $\theta_{ij} = 1.95$.

5.6 Performance evaluation

If we would consider the performance of aLARNA on the entire network when 50% prior knowledge is given, then it is difficult to see whether the inclusion of the prior knowledge improves the recovery in more than just the given part of the connectivity matrix. In other words, do coefficients weighted with 0.50 and 1.50 have a positive effect in discovering the coefficients weighted with neutral weights of 1.00.

In order to clarify this, we compare the performance measures of different parts of the connectivity matrix. For convenience we take a four-gene network as example. One dataset of the four-gene network is evaluated three times with 0%, 25% and 50% prior knowledge and inferred networks are shown in Figure 6. The left matrix (Figure 6) is inferred by aLARNA without prior knowledge. Then aLARNA is given 25% prior knowledge (4 blue elements in middle matrix) to infer the original connectivity matrix. Independent

from the values in the Jacobian, we randomly assign one coefficient per row to be weighted with knowledge from the Jacobian. In the right matrix, we randomly assign per row an additional coefficient to be weighted with prior knowledge in order to reach 50% prior knowledge (8 blue elements).

The red elements in the matrices are the coefficients selected for calculating the performance measures. The red elements represent the no-prior knowledge part of the matrix. We start by selecting all the coefficients without prior knowledge in the 50% case (red elements in right matrix). Consequently, the same coefficients are selected in the 25% (middle matrix) and 0% (left matrix) cases. Thus, part of the coefficients without prior knowledge is left out in the 25% and 0% case to make the comparison fair, because we now only consider the same coefficients without prior knowledge in all three setups.

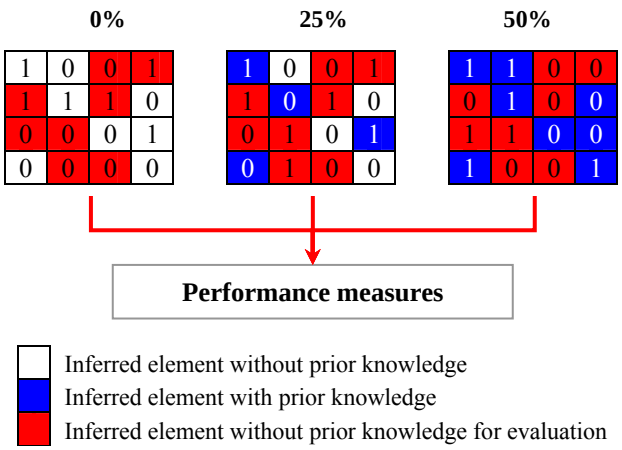


Fig. 6. Performance evaluation on no-prior knowledge part. Top row shows three connectivity matrices of a four-gene network with different percentages of prior knowledge. The row below shows the same matrices with red elements, which are used to calculated the performance measures. The red elements represent the no-prior knowledge part, which are used to calculate the specificity, the sensitivity and the positive predictive value. In the left and right matrices, white elements are left out of the evaluation of the no-prior knowledge part even thought they did not receive any prior knowledge in order to make the calculated performance measures comparable.

5.7 Random reconstruction algorithm

In order to determine if the inferred networks by aLARNA are better than randomly generating networks, we also obtain results from a random reconstruction algorithm (RRA).

The RRA generates a connectivity matrix randomly, but takes prior knowledge into account when given. We also introduce a probability parameter ρ, which controls the connectivity of the inferred networks. ρ is in the interval of [0.00 … 1.00] and indicates the probability of a one in the inferred connectivity matrix.

A full connectivity matrix is returned with ρ = 1.00, because all entries of the matrix receives ones (Figure 7: left matrix). As ρ→0.00, the RRA randomly adds zeros to the matrix to the no-prior knowledge part. 50% of the entries in the no-prior knowledge part is zero when ρ = 0.50 and 100% of no-prior knowledge entries are zero if ρ = 0.00.

Entries in the prior knowledge part are only set to zero if they are zero in the true network. When ρ = 0.50, RRA have found 50% of the zeros in the prior knowledge part (Figure 7: middle matrix). RRA infers the prior knowledge perfectly when ρ = 0.00 (Figure 7: right matrix).

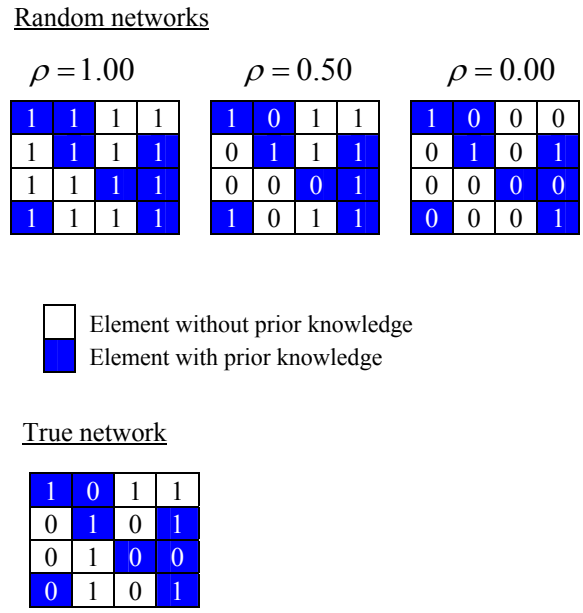


Fig. 7. Random reconstruction algorithm (RRA). The top matrices shows how RRA sets ones to zeros. The matrix below represents the true network. When ρ = 1.00, a fully connected network is inferred. As ρ → 0.00 more zeros are added to the matrix. Elements without prior knowledge are randomly set to zero. Elements with prior knowledge are only set to zero if they are zero in the true network.

6 CONCLUSION AND FUTURE WORK

We have shown that integration of prior knowledge into LARNA can improve the reconstruction of the underlying network. Moreover, when aLARNA is presented with partial information of the underlying network, it effectively infers the part of the network for which no prior information was given. aLARNA proves to be most effective with sparse networks and it even performs well when incorrect connections are present in the prior knowledge.

As future work, the integration of transcription factor binding data was not successful due to several reasons discussed in Section 4. Prior knowledge that contains more connections related to osteoblast differentiation should result in better performance. For example, ChIP-chip technology measures the binding of proteins and DNA *in vivo*. Data collected by ChIP-chip during osteoblast differentiation therefore reveals more related gene connections.

In the experiments with synthetic datasets, the Jacobian served as benchmark and as prior knowledge. In practice, the benchmark should be different than the prior knowledge, because prior knowledge can be acquired from different resources. So, there is a need

for more realistic synthetic datasets where the biological process is modeled at different levels. Zak et al. (2001) introduced a more realistic genetic network simulator where the biological process was modeled with a hysteric oscillator, a switch and a cascade. Here, prior knowledge could be extracted from the transcription, translation or post-translational modification level. The size of the network is the only disadvantage, because it contains only 10 genes whereas real gene networks contain far more genes.

The time between synthesis and activation of a transcription factor can be large due to different intermediate regulation mechanisms e.g. post-translation modification. The performance of aLARNA on the experimental setup of Section 5.2.2 could be improved if the time profiles of the regulatory genes were delayed. For example, the expression level of a target gene at time point *k* is a linear combination of the expression levels of all regulatory genes at time point *k*-5. The determination of the previous time point can however be a difficult task.

7 REFERENCES

Alako, B.T., et al. (2005). "CoPub Mapper: mining MEDLINE based on search term co-publication." BMC Bioinformatics. 2005;6:51. doi: 10.1186/1471-2105-6-51.

Albert, R. ,et al (2000). "Topology of evolving networks: local events and universality." Phys. Rev. Lett., 85, 5234-5237.

Albert, R. (2005). "Scale-free networks in cell biology." Journal of Cell Science 118, 4947-4957.

Brazhnik, P., et al. (2002) "Gene networks: how to put the function in genomics." Trends in Biotechnology 20, 467-472.

Bussemaker, H. J., et al. (2001). "Regulatory element detection using correlation with expression." Nature Genetics 27: 167-174.

Chen, T., et al. (1999). "Modeling gene expression with differential equataions." Pacific symp. biocomputing. p.29-40

Clements, M., et al. (2007). "Integration of Known Transcription Factor Binding Site Information and Gene Expression Data to Advance from Co-Expression to Co-Regulation." Genomics, Proteomics & Bioinformatics, vol. 5, no. 2, pp. 86 - 101, 2007.

Cosentino, C., et al. (2007). "Linear Matrix Inequalities Approach to Reconstruction of Biological Networks." IET Systems Biology 1(3) (2007) 164-173.

D'Haeseleer, P. et al. (1999). "Linear modeling of mRNA expression levels during CNS development and injury." Proc Pacific symp. biocomp. p.4:41

Erdös, P., et al. (1959). "On random graphs." Publ. Math. Debrecen, 6, 290-297.

Featherstone, D.E., et al. (2002). "Wrestling with pleiotropy: genomic and topological analysis of the yeast gene expression network." Bioessays, 24, 267-274.

Gardner, T.S., et al. (2005). "Reverse-engineering transcription control networks" Physics of Life Reviews 2 (2005) 65-88.

Grandvalet, Y (1998). "Least absolute shrinkage is equivalent to quadratic penalization. " ICANN'98, Perspectives in Neural Computing, pages 201-206, Springer.

Grobman, D.M. (1959). "Homeomorphisms of systems of differential equations". Dokl. Akad. Nauk SSSR 128: 880-881.

Hartman, P. (1960). "A lemma in the theory of structural stability of differential equations". Proc. A.M.S. 11 (4): 610-620. doi:10.2307/2034720. Retrieved on 2007-03-09.

Husmeier, D. (2003). "Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks." Bioinformatics, Vol. 19 no. 17 2003, pages 2271-2282.

Jeong, H., et al. (2000). "The large-scale organization of metabolic networks." Nature, 407, 651-654.

Mendes, P., et al. (2003). "Artificial gene networks for objective comparison of analysis algorithms." Bioinformatics, 2003, 19, (Suppl. 2), pp. ii122-ii129.

Schmidt, H., et al. (2005). "Identification of small scale biochemical networks based on general type system perturbations." FEBS Journal 272 (2005) 2141-2151.

Segal, E. (2003). "Genome-wide discovery of transcriptional modules from DNA sequence and gene expression." Bioinformatics 19(90001): 273-282.

Stamey, T., et al. (1989), "Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate II. Radical prostatectomy treated patients." Journal of Urology 16: 1076-1083.

Tibshirani, R. (1996). "Regression shrinkage and selection via the LASSO. " J. Royal. Statist. Soc B., Vol. 58, No. 1, pages 267-288).

van Someren, E. P., et al. (2006). "Least absolute regression network analysis of the murine osteoblast differentiation network." Bioinformatics 22(4): 477-484.

Watts, D.J., et al. (1998). "Collective dynamics of 'small-world networks." Nature, 393, 440-442.

Zak, D.E., et al. (2002). "Simulation Studies for the Identification of Genetic Networks from cDNA Array and Regulatory Activity Data." Proceedings of the Second International Conference on Systems Biology, pp. 231-238.

SUPPLEMENT

First, the shrinkage effects imposed by weighting the L_1 LASSO penalty are analyzed. Next, the ODEs by Mendes *et al*. (2003) are described in detail and how a linearised system is derived from them. The determination of weights is described in the next section. Furthermore, results with synthetic random and small-world networks are presented. Finally, the computation of Segalscore is described in detail.

S.1 Weighting the L1 LASSO penalty

We used the prostate cancer dataset (Stamey, 1989) to study the effects of weighting the L_1 LASSO penalty. LASSO applies equal penalties to each coefficient. Figure S1 shows how the coefficients shrink when LASSO is applied to the dataset. The *lweight* predictor (green dashed line) is shrunk to zero before the *svi* predictor (pink solid line).

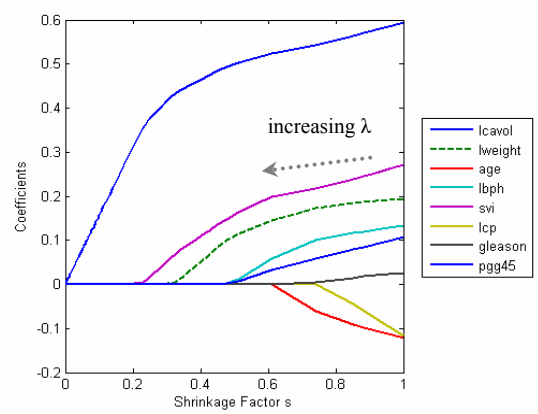


Fig. S1. Shrinkage of coefficients from the prostate cancer dataset, as λ is varied. Coefficients are plotted against shrinkage factor s (standardized λ). All predictors are weighted equally with 1.00 in the L_1 LASSO penalty.

When the *lweight* predictor is weighted with a value of 0.60, we penalize this predictor less and hope that it will be shrunk to zero at a later stage. In Figure S2, we observe that the lweight predictor (green dashed line) is shrunk towards zero after the svi predictor (pink solid line).

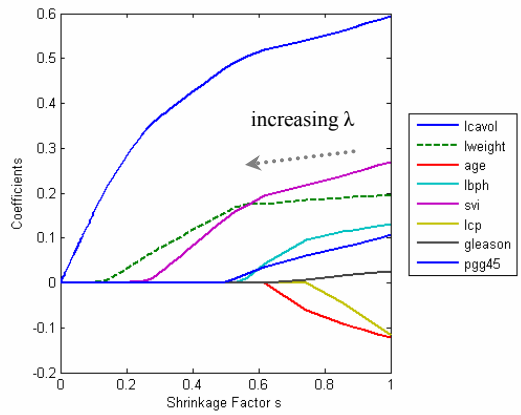


Fig. S2. Shrinkage of coefficients from the prostate cancer dataset, as λ is varied. Coefficients are plotted against shrinkage factor s (standardized λ). The *lweight* predictor is weighted with 0.60 in the L_1 LASSO penalty, whereas all other predictors are weighted with 1.00. As expected, the *lweight* predictor is shrunk to zero after the *svi* predictor.

S.2 Ordinary differential equations

The ODEs from the artificial gene networks (Mendes, 2003) are defined as:

$$\frac{dG_i}{dt} = s(G_1, ..., G_n) - b(G_i), \qquad (S1)$$

where G_i represents the abundance of the mRNA of gene i, $s(G_1,...,G_n)$ the rate of transcription and $b(G_i)$ the mRNA breakdown rate.

Regulatory effects of all regulatory genes are represented by the rate function s as follows:

$$s(G_1, \ldots, G_n) = V_i \prod_j \left(\frac{\beta_j^{m_j}}{I_j^{m_j} + \beta_j^{m_j}}\right) \times \prod_k \left(1 + \frac{A_k^{m_k}}{A_k^{m_k} + \alpha_k^{m_k}}\right), \qquad (S2)$$

where V_i symbolizes the basal rate of transcription, I_j the inhibitor, A_j the activator, α and β the saturation constants of the activators and inhibitors and m_j regulates the sigmoidicity of the curve. The number of regulatory genes per i-th gene is dependent on the network topology. When this number is determined, 50% of the regulatory genes are selected as activators and 50% are inhibitors.

S.3 Jacobian

This section describes the calculation and importance of the Jacobian for a linear system. Consider the following nonlinear system of ODEs:

$$\frac{dG}{dt} = f(G). \qquad (S3)$$

From this nonlinear system we can calculate the Jacobian by taking the partial derivatives, as follows:

$$J = Df = \begin{pmatrix} \frac{\partial f_1}{\partial G_1} & \cdots & \frac{\partial f_1}{\partial G_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial G_1} & \cdots & \frac{\partial f_n}{\partial G_n} \end{pmatrix}, \qquad (S4)$$

where n is the number of ODEs.

The behavior of a nonlinear system around a steady-state is similar to a linearised system. The Jacobian at the steady-state needs to be calculated in order to define a linearised system. The nonlinear system is linearised as follows:

$$\frac{dG}{dt} = J_{st} \cdot G, \qquad (S5)$$

where J_{st} is the calculated Jacobian at the steady-state. Since LARNA models a biological system as a linear, the inferred connectivity matrices **A** are compared to the Jacobian.

S.4 Determining the values of the weights

S.4.1 Scale-free gene network The weights θ_{ij} from equation (3) have to be chosen carefully in order to obtain good performance. When all coefficients receive weights equal to 1, then equation (3) becomes equation (2), which is LARNA without any prior knowledge about the coefficients. Therefore, coefficients without prior knowledge will receive weights equal to 1.00 (table S1).

Coefficients known to be nonzero in the original network receive weights equal to a value from the open interval (0.00, 1.00), because a weight below 1.00 causes the coefficients to be shrunk less and thus have a higher probability to be selected. Weights close to 0.00 cause the coefficients to be shrunk less, than weights close to 1.00.

Coefficients known to be zero in the original network receive weights equal to a value from the interval (1.00, 5.00]. Higher weights than 5.00 seems inappropriate, because the coefficients would penalised too severe. Weights with a value above 1.00 cause the coefficients to be shrunk more and as a result are not selected.

Table S1. Performance measures.

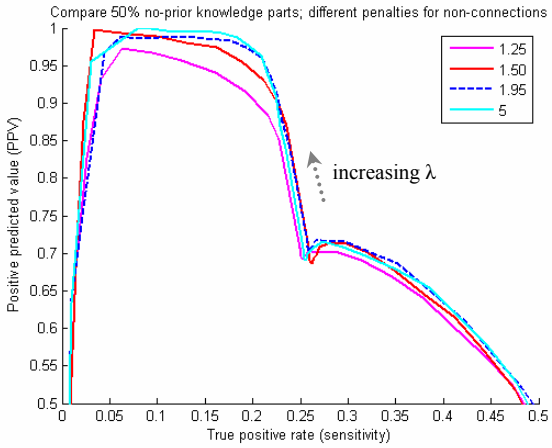
	Evaluated values	Optimal value
No prior knowledge	$\theta_{ij}^{no-pk} = 1.00$	$\theta_{ij}^{no-pk} = 1.00$
Nonzero element in true network	$\theta_{ij}^{conn} = (0.00, 1.00)$	$\theta_{ij}^{conn} = 0.50$
Zero element in true network	$\theta_{ij}^{no-conn} = (1.00...5.00]$	$\theta_{ij}^{no-conn} = 1.50$

Here, we look for the optimal values to assign the three different weights. Coefficients known to be nonzero received weights equal to 0.05 and 0.50 in Figure S3a and Figure S3b, respectively. Coef-

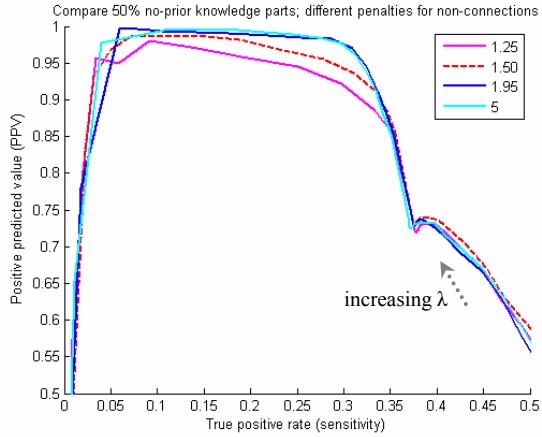
ficients known to be zero receive weights equal to 1.25, 1.50, 1.95 or 5.00.

Figure S3b shows better performance than Figure S3a. In Figure S3b, there is a wider range (SE = [0.05, 0.37]) where the positive predictive value is close to 1.00. Thus, the optimal weight for coefficient known to be nonzero is 0.50.

It is not straightforward to select the most optimal weight for coefficients known to be zero. Figure S3b shows that the performance with weights 1.95 and 5.00 are similar and also weight 1.50 results in a good performance. In order to decide which weight to select for the coefficients known to be nonzero, the performance is evaluated with errors in the prior knowledge and with weight 0.50 for coefficients known to be nonzero.



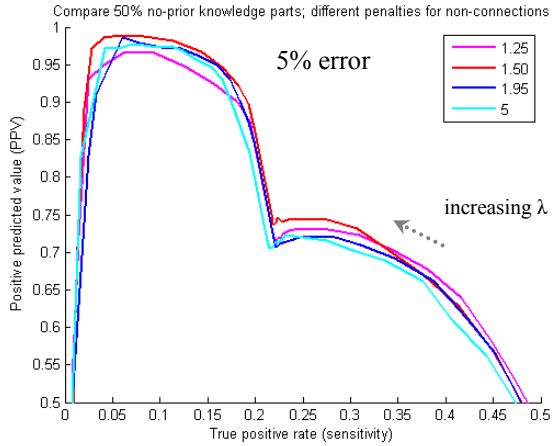
(a)



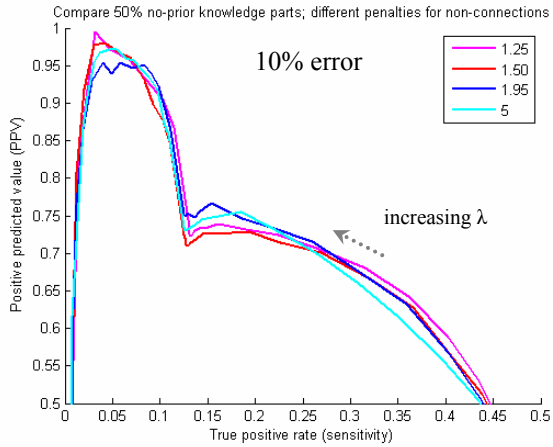
(b)

Fig. S3. Performances on the scale-free gene networks with different values for weighting the coefficients. Both plots consider only the no-prior knowledge part. The two plots show 4 lines, which are the performances with 4 different values for the weights of coefficients known to be zero. Furthermore, the two plots differ in the values for the weights of the coefficients known to be nonzero; (a) the coefficients known to be nonzero receive a weight equal to 0.05; (b) the coefficients known to be nonzero receive a weight equal to 0.50. Both plots receive 50% of the original network as prior knowledge without errors.

Figures S4a and S4b are similar to Figure S3b, but with respectively 5% and 10% errors in the prior knowledge. Weight 1.50 results in the best performance with 5% errors (red line in Figure S4a). Over the whole range (SE = [0.00, 0.50], it has the highest PPV score. In Figure S4b, weight 1.50 results in the second highest PPV score in the interval SE = [0.02, 0.05]. Overall, weight 1.50 gives aLARNA the best performance.



(a)



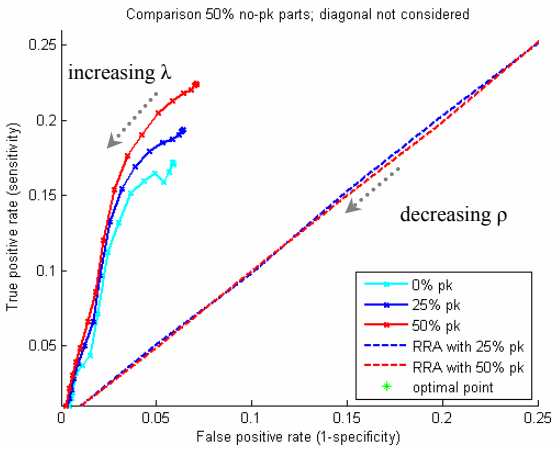
(b)

Fig. S4. Performances on the scale-free networks with different values for weighting the coefficients. Plots (a) and (b) show the performance on the no-prior knowledge part. Each plot represents the performance of aLARNA with different values for weighting the coefficients. The different lines in each plot represent different values for weighting coefficients known to be zero in the original network. Coefficients known to nonzero are weighted with 0.50. All plots receive 50% of the original network as prior knowledge. However, aLARNA received prior knowledge with 5% errors in plot (a) and with 10% errors in plot (b).

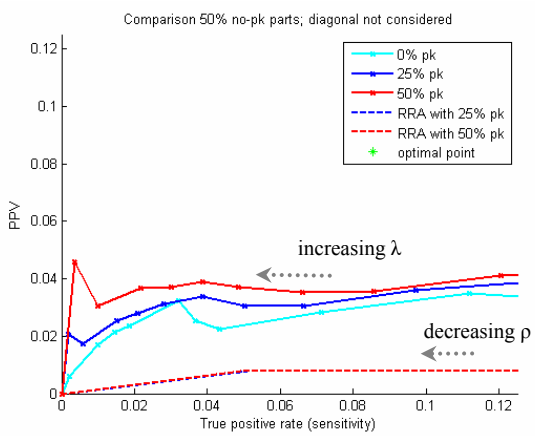
S.4.2 Osteoblast dataset The same range of weights {0.50, 1.00, 1.50} was initially used for the osteoblast dataset, but this resulted in poor performance (Figure S5). The solid red line always obtains a higher SE and PPV score in respectively Figure S5a and

S5b, which shows that more prior knowledge results in better performance. However, the highest PPV score of 0.05 is still a poor performance.

We tried another range of weights (0.05, 1.00, 1.95) and this resulted in good performance (see Figure 3). Here, the coefficients are penalized more severely and that has a positive influence. We think that the networks inferred with the osteoblast dataset are rather different than the literature network. The literature network contains general relationships between genes, while networks inferred from the osteoblast dataset contains more specific relationships. By penalizing aLARNA with more stringent weights, we force aLARNA to find these general relationships. However, since we only take consideration of the no-prior knowledge part, it is interesting that general relationships are found without prior knowledge about them. So, the osteoblast dataset also contains information about general relationships between genes, which are apparently important for osteoblast differentiation.



(a)

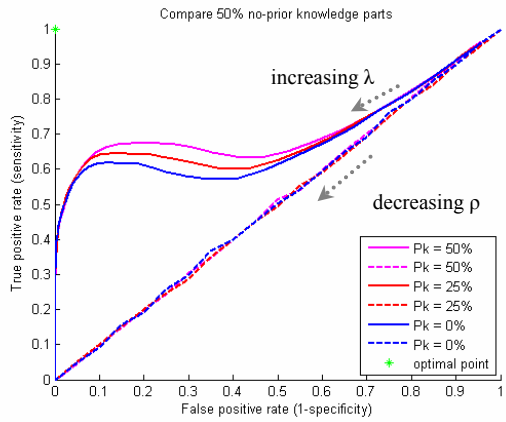


(b)

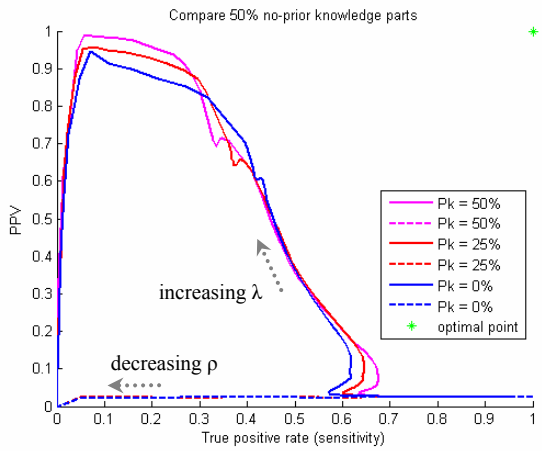
Fig. S5. Performances on the osteoblast dataset. Sensitivity versus specificity (a) and PPV versus sensitivity (b). Weights selected from the set {0.50, 1.00, 1.50} proved to result in poor performance.

S.5 Results on CenturyRND and CenturySW

We present the performance of aLARNA on the random (Figure S6) and small-world networks (Figure S7). The differences in performance are minimal compared to scale-free networks. Figure S7b shows that for small-world networks there is hardly any difference between 25% and 50% prior knowledge (solid pink and red lines).

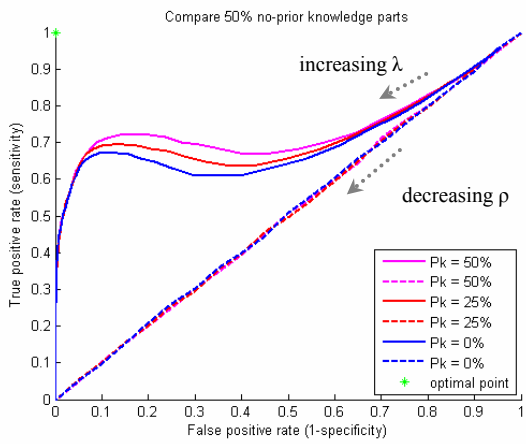


(a)

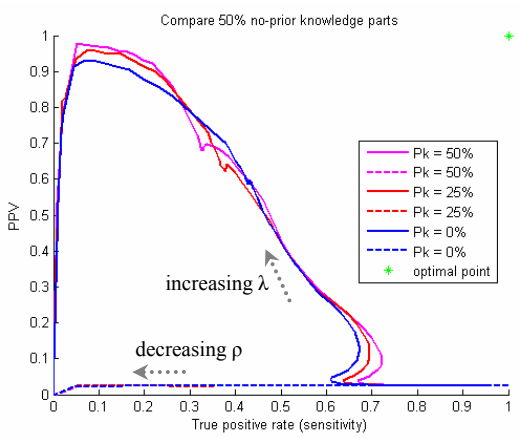


(b)

Fig. S6. Results on CenturyRND: specificity versus sensitivity (a) and sensitivity versus PPV (b). 0% (blue lines), 25% (red lines) and 50% (pink lines) of the underlying network were given as prior knowledge to aLARNA (solid lines) and RRA (dashed lines). aLARNA was evaluated with a λ between 10^{-4} and 10^{7}.



(a)



(b)

Fig. S7. Results on CenturySW: specificity versus sensitivity (a) and sensitivity versus PPV (b). 0% (blue lines), 25% (red lines) and 50% (pink lines) of the underlying network were given as prior knowledge to aLARNA (solid lines) and RRA (dashed lines). aLARNA was evaluated with a λ between 10^{-4} and 10^{7}.

S.6 Segalscore

The binding sites of transcription factors are represented by nucleotide distribution matrices. These are called motif matrices (Figure S8). For each transcription factor the motif matrix was obtained from the TRANSFAC® 11.2 database.

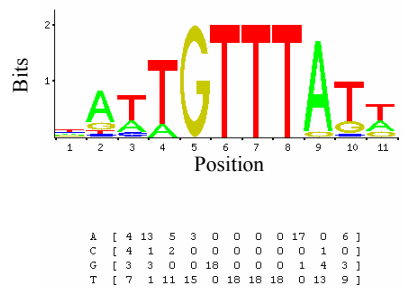


Fig. S8. Example of a motif logo (above) and nucleotide distribution matrix (under).

Position weight matrices (PWM) are computed from the motif matrices in order to scan upstream sequences of target genes, as follows:

$$W_{b,j} = \ln \frac{f_{b,j}}{p_b}, \qquad (S6)$$

where $f_{b,j}$ is the number of occurrences of the possible bases b at position j and p_b is the background frequency of base b.

The upstream sequence of a target gene can be scored by a PWM at each position of the sequence.

$$Sc_i = \sum_{j=1}^{p} W_j[S_{i+j-1}], \qquad (S7)$$

Where S_i is the base at position i in the upstream sequence, p is the size of the motif and W is the PWM.

The Segalscore scores each gene-motif combination by a value between 0 and 1. The larger the Segalscore the more likely a transcription factor binds to the upstream sequence of the target gene. This score provides better predictions, because it avoids the problems arising from high-frequency but meaningless motifs that are common in many upstream sequences (Clements, 2007; Segal, 2003). The Segalscore is computes as follows:

$$P(g.M = true \mid S_1, \ldots, S_n) = \zeta(\log(\frac{1}{n-p+1} \sum_{i=1}^{n-p+1} \exp\{\sum_{j=1}^{p} W_j[S_{i+j-1}]\})), \qquad (S8)$$

where ζ is the sigmoid function and n the length of the upstream sequence.

We threshold the calculated Segalscore with a value of 0.95 (Figure S9). At this threshold, the target gene have a median number of 6 motifs (TFs) and 77 target genes don't have motifs. This is motivated by [3]: in vertebrates targets are approximately regulated by 6 TFs and the number of target genes without motifs should be a small as possible.

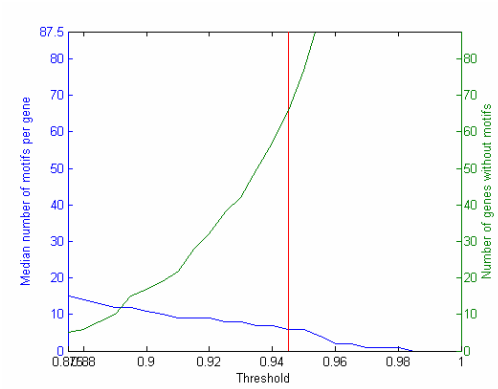


Fig. S9. Median number of motifs per gene and number of genes without motifs.