Learning and Validating Bayesian Network Models of Gene Networks

Jose M. Peña[1], Johan Björkegren[2], and Jesper Tegnér[1,2]

[1] IFM, Linköping University, SE-58183 Linköping, Sweden
{jmp, jespert}@ifm.liu.se
[2] CGB, Karolinska Institute, SE-17177 Stockholm, Sweden
johan.bjorkegren@cgb.ki.se

Abstract. We propose a framework for learning from data and validating Bayesian network models of gene networks. The learning phase selects multiple locally optimal models of the data and reports the best of them. The validation phase assesses the confidence in the model reported by studying the different locally optimal models obtained in the learning phase. We prove that our framework is asymptotically correct under the faithfulness assumption. Experiments with real data (320 samples of the expression levels of 32 genes involved in *Saccharomyces cerevisiae*, i.e. baker's yeast, pheromone response) show that our framework is reliable.

1 Introduction

The cell is the functional unit or building block of all the organisms. The cell is self-contained, as it includes the information necessary for regulating its function. This information is encoded in the DNA of the cell, which is divided into a set of genes, each coding for one or more proteins. Proteins are required for practically all the functions in the cell, and they are produced through the expression of the corresponding genes. The amount of protein produced is determined by the expression level of the gene, which may be regulated by the protein produced by another gene. As a matter of fact, much of the complex behavior of the cell can be explained through the concerted activity of genes. This concerted activity is typically represented as a network of interacting genes. Identifying this network, which we call gene network (GN), is crucial for understanding the behavior of the cell which, in turn, can lead to better diagnosis and treatment of diseases. This is one of the most exciting challenges in bioinformatics. For the last few years, there has been an increasing interest in learning Bayesian network (BN) models of GNs [1,9,12,14,17,20,21,23], mainly owing to the following two reasons. First, there exist principled algorithms for learning BN models from data [3,5,19,21,27]. Second, BN models can represent stochastic relations between genes. This is particularly important when inferring models of GNs from gene expression data, because gene expression has a stochastic component [2,18], and because gene expression data typically include measurement noise [9,25].

Following the papers cited above, we view a GN as a probability distribution $p(\mathbf{U})$, where $\mathbf{U}$ is a set of random variables such that each of them

represents the expression level of a gene in the GN. And we aim to learn about the (in)dependencies in $p(\mathbf{U})$ by learning a BN model from some given gene expression data sampled from $p(\mathbf{U})$. Specifically, we define a BN model $M(G)$ as the set of independencies in G, where G is an acyclic directed graph (DAG) whose set of nodes is $\mathbf{U}$. The independencies in G correspond to the d-separation statements in G: $\mathbf{X}$ and $\mathbf{Y}$ are d-separated given $\mathbf{Z}$ in G if for every undirected path in G between a node in $\mathbf{X}$ and a node in $\mathbf{Y}$ there exists a node W in the path such that either (i) W does not have two parents in the path and $W \in \mathbf{Z}$, or (ii) W has two parents in the path and neither W nor any of its descendants in G is in $\mathbf{Z}$. The probability distributions that do not satisfy any other independence than those in G are called faithful to G.

In this paper, we follow the so-called model selection approach to learning a BN model from some given data: Given a scoring criterion that evaluates the quality of a model with respect to the data, model selection searches the space of models for the highest scoring model. Unfortunately, model selection is NP-complete [4]. For this reason, most algorithms for model selection are heuristic and they only guarantee convergence to a locally optimal model. Validating this model is crucial, as the number of locally optimal models can be large [19]. When inferring a BN model of a GN from gene expression data, validation becomes even more important: Gene expression data are typically scarce and noisy [9,25] and, thus, they may not have enough power to discriminate between those locally optimal models that are close to the set of independencies in the probability distribution of the GN and those that are not.

In this paper, we propose a framework for learning from data and validating BN models of GNs. The learning phase consists in running repeatedly a stochastic algorithm for model selection in order to discover multiple locally optimal models of the learning data and, then, reporting the best of them. The validation phase assesses the confidence in some features of the model reported by studying the different locally optimal models obtained in the learning phase. The higher the confidence in the features of the model reported, the more believable or valid it is. We prove that our framework is asymptotically, i.e. in the large sample limit, correct under the faithfulness assumption. We show with experiments on real data that our framework is reliable.

In the sections below, we describe the learning and validation phases of our framework (Sects. 2 and 3, respectively) and, then, we evaluate it on synthetic and real data (Sects. 4 and 5, respectively). We conclude in Sect. 6 with a discussion on this and related works.

2 Learning Phase

As mentioned in the previous section, the learning phase runs repeatedly a stochastic algorithm for model selection in order to obtain multiple locally

optimal models of the learning data and, then, reports the best of them. We use the k-greedy equivalence search algorithm (KES) [19] for this purpose. Like most algorithms for model selection, KES consists of three components: A neighborhood, a scoring criterion, and a search strategy. The neighborhood of a model restricts the search to a small part of the search space around the model, and it is usually defined by means of local transformations of the model. The scoring criterion evaluates the quality of a model with respect to the learning data. The search strategy selects a new model, based on the scoring criterion, from those in the neighborhood of the current best model. The paragraphs below describe these components in the case of KES.

KES uses the inclusion boundary of a model as the neighborhood of the model. The inclusion boundary of a model $M(G_1)$, $IB(M(G_1))$, is the union of the upper and lower inclusion boundaries, $UIB(M(G_1))$ and $LIB(M(G_1))$, respectively. $UIB(M(G_1))$ is the set of models $M(G_2)$ that are strictly included in $M(G_1)$ and such that no model strictly included in $M(G_1)$ strictly includes $M(G_2)$. Likewise, $LIB(M(G_1))$ is the set of models $M(G_2)$ that strictly include $M(G_1)$ and such that no model strictly including $M(G_1)$ is strictly included in $M(G_2)$. $IB(M(G_1))$ is characterized using DAGs as the set of models represented by all the DAGs that can be obtained by adding or removing a single edge from any representative DAG of $M(G_1)$, where a DAG G_2 is representative of $M(G_1)$ if $M(G_1) = M(G_2)$ [5]. Any representative DAG G_2 of a model can be obtained from any other representative DAG G_1 of the model through a sequence of covered edge reversals in G_1, where the edge $X \rightarrow Y$ is covered in G_1 if X and Y share all their parents but X in G_1 [5].[1]

KES scores a model by scoring any representative DAG of the model. Thus, KES requires that all the representative DAGs of a model receive the same score. Furthermore, KES also requires that the scoring criterion is locally consistent: Given an i.i.d sample from a probability distribution $p(\mathbf{U})$, the scoring criterion is locally consistent if the score assigned to a DAG G asymptotically increases (resp. decreases) with each edge removal that adds independencies to $M(G)$ that hold (resp. does not hold) in $p(\mathbf{U})$. The two most commonly used scoring criteria, the Bayesian Dirichlet metric with uniform prior (BDeu) [15] and the Bayesian information criterion (BIC) [24], satisfy the two requirements above and can be used with KES [5]. BDeu scores the exact marginal likelihood of the learning data for a given DAG, whereas BIC scores an asymptotic approximation to it. Finally, KES uses the following search strategy:

```
KES (k∈[0,1])
M = model of the DAG without any edge
repeat
```

[1] A more efficient, though more complex, characterization of $IB(M(G))$ using completed acyclic partially directed graphs is reported in [28,29].

```
B = set of models in IB(M) with higher score than M
if |B| > 0 then
   C = random subset of B with size max(1,|B|·k)
   M = the highest scoring model in C
else return M
```

where $|B|$ denotes the cardinality of the set B. For the sake of simplicity, KES represents each model in the search space by one of its representative DAGs. Thus, B and C are sets of DAGs. The input parameter $k \in [0, 1]$ allows to trade off greediness for randomness. This makes KES ($k \neq 1$) able to reach different locally optimal models when run repeatedly. KES ($k = 1$) corresponds to the greedy equivalence search algorithm (GES) proposed in [5].[2] We refer the reader to [19] for a thorough study of KES, including the proof of the following property.

Theorem 1. *Given a fully observed i.i.d sample from a probability distribution faithful to a DAG G, KES asymptotically returns $M(G)$.*

3 Validation Phase

In the light of the experiments in [19], the learning phase described in the previous section is very competitive. However, when the learning data are as scarce, noisy and complex as gene expression data are, the best locally optimal model discovered in the learning phase may not be reliable, because the learning data may lack the power to discriminate between those locally optimal models that are close to the set of independencies in the sampled probability distribution and those that are not. Therefore, validating the model learnt is of much importance. Our proposal for validating it consists of two main steps. First, extraction of relevant features from the model. Second, assessment of the confidence in the features extracted. The higher the confidence in these features, the more believable or valid the model is. The following sections describe these two steps.

3.1 Feature Extraction

First of all, we need to adopt a model representation scheme that allows interesting features to be extracted. Representing a model by a DAG does not seem appropriate here, because there may be many representative DAGs of the model. A completed acyclic partially directed graph (CPDAG) provides, on the other hand, a canonical representation of a model. A CPDAG represents a model by summarizing all its representative DAGs: The CPDAG

[2] To be exact, GES is a two-phase algorithm that first uses only $UIB(M(G))$ and, then, only $LIB(M(G))$. KES ($k = 1$) corresponds to a variant of GES described in [5] that uses $IB(M(G))$ in each step.

contains the directed edge $X \rightarrow Y$ if $X \rightarrow Y$ exists in all the representative DAGs, while it contains the undirected edge $X—Y$ if $X \rightarrow Y$ exists in some representative DAGs and $Y \rightarrow X$ in some others. See [5] for an efficient procedure to transform a DAG into its corresponding CPDAG.

We pay attention to four types of features in a CPDAG: Directed edges, undirected edges, directed paths, and Markov blanket neighbors. Two nodes are Markov blanket neighbors if there is an edge between them or if they have a child in common. We focus on these types of features because they suggest relevant features of the probability distribution of the learning data. A directed or undirected edge suggests an unmediated dependence. A directed path suggests a causal pathway because it appears in all the representative DAGs of the model. Finally, the Markov blanket neighborhood of a random variable suggests the minimal set of predictors of the probability distribution of the random variable, because the Markov blanket neighborhood is the minimal set conditioned on which the random variable is independent of the rest of random variables.

3.2 Confidence Assessment

Despite the fact that the different locally optimal models discovered in the learning phase disagree in some features, we expect them to share some others. In fact, the more strongly the learning data support a feature, the more frequently it should appear in the different locally optimal models found. Likewise, the more strongly the learning data support a feature, the higher the likelihood of the feature being true in the probability distribution that generated the learning data. This leads us to assess the confidence in a feature as the fraction of models containing the feature out of the different locally optimal models obtained in the learning phase. Note that we give equal weight to all the models available, no matter their scores. Alternatively, we could weight each model by its score. We prove below that this approach to confidence estimation is asymptotically correct under the faithfulness assumption. We show in Sect. 4 that it is accurate for finite samples as well.

Theorem 2. *Given a fully observed i.i.d sample from a probability distribution faithful to a DAG G, the features in $M(G)$ asymptotically receive confidence equal to one and the rest equal to zero.*

Proof. Under the conditions of the theorem, KES asymptotically returns $M(G)$ owing to Theorem 1. □

3.3 Validity Assessment

Let M^* denote the best locally optimal model found in the learning phase. Deciding on the validity of M^* on the basis of the confidence values scored by its features may be difficult. We suggest a sensible way to ease making this

decision. We call true positives (TPs) to the features in M^* with confidence value equal or above a given threshold value t. Likewise, we call false positives (FPs) to the features not in M^* with confidence value equal or above t, and false negatives (FNs) to the features in M^* with confidence value below t. In order to decide on the validity of M^*, we propose studying the trade-off between the number of FPs and FNs for each type of features under study as a function of t. The fewer FPs and FNs for high values of t, the more believable or valid M^* is. In other words, we trust M^* as a valid model of the probability distribution of the learning data if the features in M^* receive high confidence values, while the features not in M^* score low confidence values. Note that we treat on equal basis FPs and FNs. Alternatively, we can attach different costs to them according to our preferences, e.g. we may be less willing to accept FPs than FNs. The following property follows directly from Theorem 2.

Theorem 3. *Given a fully observed i.i.d sample from a probability distribution faithful to a DAG G, the number of FPs and FNs is asymptotically zero for any $t > 0$.*

Therefore, our framework for learning from data and validating BN models of GNs is asymptotically correct under the faithfulness assumption, i.e. the learning phase always returns the true model (Theorem 1) and the validation phase always confirms its validity (Theorem 3). We note that, although the faithfulness assumption may not hold in practice, the theorems above are desirable properties for any work on BN model validation to have.

4 Evaluation on Synthetic Data

We have proven in Theorems 2 and 3 that our approach to confidence estimation is asymptotically correct under the faithfulness assumption. We now show that it is also accurate for finite samples under the faithfulness assumption. The database used in the evaluation is the Alarm database [16]. This database consists of 20000 cases sampled from a BN model representing potential anesthesia problems in the operating room. The CPDAG of the BN model sampled has 37 nodes and 46 edges, and each node has from two to four states. We perform experiments with samples of sizes 1 %, 2 %, 5 %, 10 %, 25 %, 50 % and 100 % of the Alarm database. The results reported are averages over five random samples of the corresponding size.

The setting for the evaluation is as follows. We consider KES ($k = 0.6, 0.8, 0.9$) with BIC as the scoring criterion. We avoid values of k close to 0 so as to prevent convergence to poor locally optimal models [19]. For each sample in the evaluation, we first run KES 1000 independent times and use the different locally optimal models discovered to estimate the confidence in the features of interest, i.e. directed edges, undirected edges, directed paths, and Markov blanket neighbors. We give equal weight to all the models used

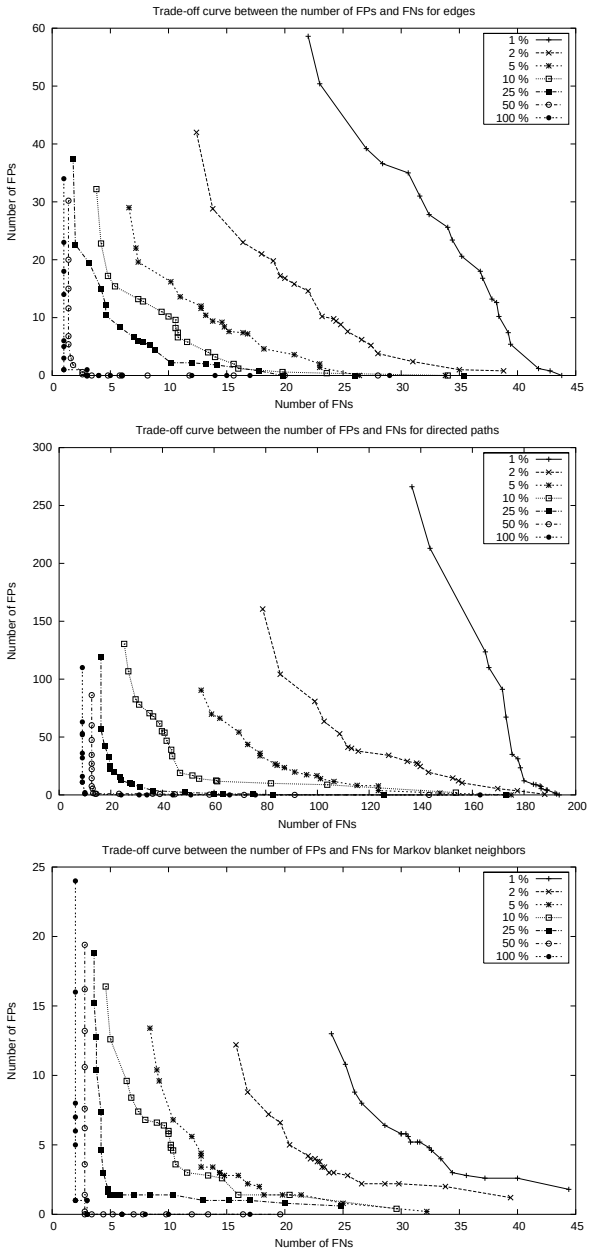


Fig. 1. Trade-off between the number of FPs and FNs for the Alarm databases ($k = 0.8$) at threshold values $t = 0.05 \cdot r$, $r = 1, \ldots, 20$. Top, directed and undirected edges. Middle, directed paths. Bottom, Markov blanket neighbors

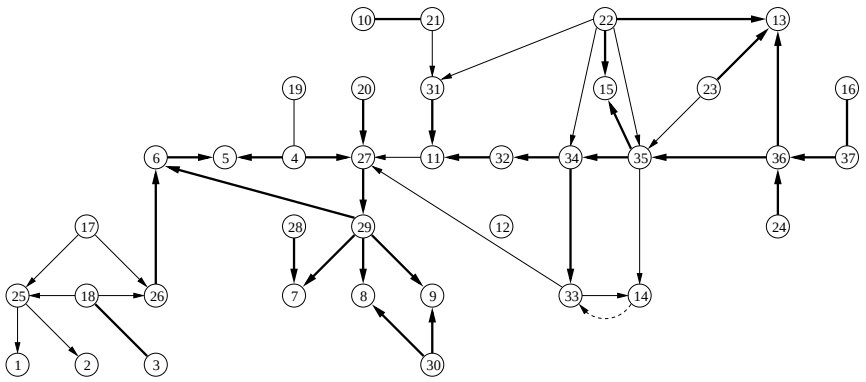


Fig. 2. Directed and undirected edges for the Alarm database of size 100 % ($k = 0.8$) when $t = 0.45$ (plain and bold edges) and when $t = 0.95$ (bold edges). Solid edges are TPs and dashed edges are FPs

for confidence estimation. Then, we compute the trade-off between the number of FPs and FNs for each type of features under study as a function of the threshold value t. We treat equally FPs and FNs when computing the trade-off. Unlike in Sect. 3.3, FPs and FNs are calculated with respect to the true model so as to assess the accuracy of our method for confidence estimation.

We report results for $k = 0.8$ and omit the rest because they all lead to the same conclusions. Out of the 1000 independent runs of KES performed for each of the samples in the evaluation, we obtained an average of 203 different locally optimal models for the sample size 1 %, 233 for 2 %, 161 for 5 %, 115 for 10 %, 119 for 25 %, 85 for 50 %, and 70 for 100 %. We note that the number of different locally optimal models obtained decreases as the size of the learning data increases, which is expected because Theorem 1 applies. Figure 1 shows the trade-off curves between the number of FPs and FNs as a function of the threshold value t. We note that the CPDAG of the true model has 42 directed edges, 4 undirected edges, 196 directed paths, and 65 Markov blanket neighbors. We do not report trade-off curves for undirected edges because they are difficult to visualize as there are only four undirected edges in the true model. Instead, the trade-off curves in Fig. 1 (top) summarize the number of FPs and FNs for both directed and undirected edges. The shape of the trade-off curves for the three types of features, concave down and closer to the horizontal axis (FNs) than to the vertical axis (FPs), indicates that our method for confidence estimation is reliable: For all the sample sizes except 1 %, there is a wide range of values of t such that (i) the number of TPs is higher than the number of FNs, and (ii) the number of FNs is higher than the number of FPs. For the sample size 1 %, these observations are true only for Markov blanket neighbors, which indicates that these features are easier to learn. This makes sense as Markov blanket neighbors are less sensitive than the other types of features to whether the edge between two nodes is

directed or undirected. The trade-off curves in the figure also show that the number of FPs and FNs decreases as the size of the learning data increases, which is expected because Theorem 3 applies. In particular, when setting t to the value that minimizes the sum of FPs and FNs for the sample size 100 %, there are 1 FP and 1 FN (45 TPs) for edges ($t = 0.45$), 1 FP and 10 FNs (186 TPs) for directed paths ($t = 0.6$), and 0 FPs and 3 FNs (62 TPs) for Markov blanket neighbors ($t = 0.7$). Figure 2 depicts the TP and FP edges for the sample size 100 % when $t = 0.45, 0.95$. Recall that $t = 0.45$ is the threshold value that minimizes the sum of FPs and FNs for edges and that it implies 1 FP and 1 FN (45 TPs). The FN edge $12 \rightarrow 32$ is reported in [6] to be not supported by the data. When $t = 0.95$, there are 0 FPs and 17 FNs (29 TPs). Therefore, our method for confidence assessment assigns to a considerable amount of TPs higher confidence than to any FP. This is also true for directed paths and Markov blanket neighbors as can be seen in Fig. 1 (middle and bottom).

5 Evaluation on Real Data

In this section, we evaluate the framework for learning from data and validating BN models of GNs that we have proposed in Sects. 2 and 3. The experimental setting is the same as in the previous section with the only exception that FPs and FNs are now calculated with respect to the best locally optimal model found in the learning phase (recall Sect. 3.3). The data used in the evaluation are the data in [14], which we call the Yeast database hereinafter. This database consists of 320 records characterized by 33 attributes. The records correspond to 320 samples of unsynchronized *Saccharomyces cerevisiae* (baker's yeast) populations observed under different experimental conditions.[3] The first 32 attributes of each record represent the expression levels of 32 genes involved in yeast pheromone response. This pathway plays an essential role in the sexual reproduction of yeast. The last attribute of each record, named MATING_TYPE, indicates the mating type of the strain of yeast in the corresponding sample, either MATa or MATα, as some of the 32 genes measured express only in strains of a specific mating type. Gene expression levels are discretized into four states. We refer the reader to [14] for details on the data collection and preparation process. Table 1 reproduces the description of the 32 genes in the database that is given in [14]. The description is based on [7,8,22]. The table also divides the genes into groups according to their function in the domain under study.

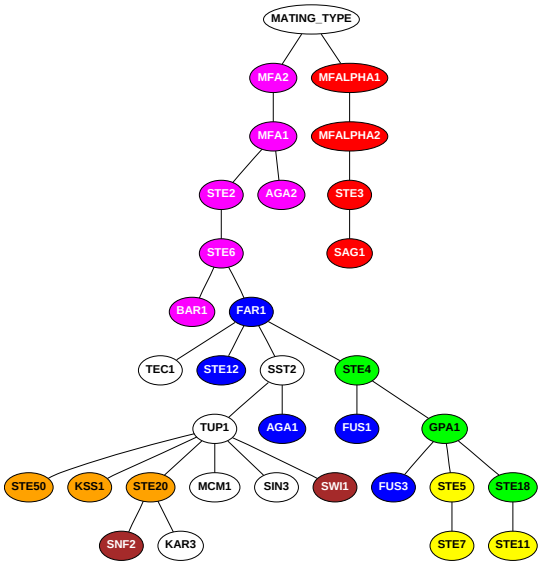
We first report the results of the learning phase. Out of the 1000 independent runs of KES performed for each value of k considered in the evaluation, we obtained 967 different locally optimal models for $k = 0.6$, 330 for

[3] Yeast is extensively studied in molecular biology and bioinformatics because it is considered an ideal organism: It is quick and easy to grow, and it provides insight into the workings of other organisms, including humans.

Table 1. Top, description of the 32 genes in the Yeast database. The genes are divided into functional groups according to the current knowledge of yeast pheromone response. Each group has a different color assigned. Bottom, description of the groups of genes

Gene	Group	Function of the protein encoded by the gene
STE2	Magenta	Transmembrane receptor peptide
MFA1	Magenta	a-factor mating pheromone
MFA2	Magenta	a-factor mating pheromone
STE6	Magenta	Responsible for the export of a-factor from MATa cells
AGA2	Magenta	Binding subunit of a-agglutinin complex, involved in cell-cell adhesion during mating by binding Sag1
BAR1	Magenta	Protease degrading α-factor
STE3	Red	Transmembrane receptor peptide
MFALPHA1	Red	α-factor mating pheromone
MFALPHA2	Red	α-factor mating pheromone
SAG1	Red	Binding subunit of α-agglutinin complex, involved in cell-cell adhesion during mating by binding Aga2 (also known as Agα1)
FUS3	Blue	Mitogen-activated protein kinase (MAPK)
STE12	Blue	Transcriptional activator
FAR1	Blue	Substrate of Fus3 that leads to G1 arrest, known to bind to STE4 as part of complex of proteins necessary for establishing cell polarity required for shmoo formation after mating signal has been received
FUS1	Blue	Required for cell fusion during mating
AGA1	Blue	Anchor subunit of a-agglutinin complex, mediates attachment of Aga2 to cell surface
GPA1	Green	Component of the heterotrimeric G-protein (Gα)
STE4	Green	Component of the heterotrimeric G-protein (Gβ)
STE18	Green	Component of the heterotrimeric G-protein (Gγ)
STE7	Yellow	MAPK kinase (MAPKK)
STE11	Yellow	MAPKK kinase (MAPKKK)
STE5	Yellow	Scaffolding peptide holding together Fus3, Ste7 and Ste11 in a large complex
KSS1	Orange	Alternative MAPK for pheromone response (in some dispute)
STE20	Orange	p21-activated protein kinase (PAK)
STE50	Orange	Unknown function but necessary for proper function of Ste11
SNF2	Brown	Implicated in induction of numerous genes in pheromone response pathway (component of SWI-SNF global transcription activator complex)
SWI1	Brown	Implicated in induction of numerous genes in pheromone response pathway (component of SWI-SNF global transcription activator complex)
SST2	White	Involved in desensitization to mating pheromone exposure
KAR3	White	Essential for nuclear migration step of karyogamy
TEC1	White	Transcriptional activator believed to bind cooperatively with Ste12 (more active during induction of filamentous or invasive growth response)
MCM1	White	Transcription factor believed to bind cooperatively with Ste12 (more active during induction of pheromone response)
SIN3	White	Implicated in induction or repression of numerous genes in pheromone response pathway
TUP1	White	Implicated in repression of numerous genes in pheromone response pathway

Group	Description of the group
Magenta	Genes expressed only in MATa cells
Red	Genes expressed only in MATα cells
Blue	Genes whose promoters are bound by Ste12
Green	Genes coding for components of the heterotrimeric G-protein complex
Yellow	Genes coding for core components of the signaling cascade (except FUS3 which is in the group Blue)
Orange	Genes coding for auxiliary components of the signaling cascade
Brown	Genes coding for components of the SWI-SNF complex
White	Others



	$k = 0.6$		$k = 0.8$		$k = 0.9$	
t	FPs	FNs	FPs	FNs	FPs	FNs
1.00	0	30	0	25	0	22
0.95	0	22	0	15	0	12
0.90	0	17	0	11	0	10
0.85	0	12	0	8	0	7
0.80	0	11	0	6	0	3
0.75	0	8	0	2	0	1
0.70	0	5	0	1	0	1
0.65	0	2	0	1	0	1
0.60	0	1	0	0	0	0
0.55	0	1	0	0	0	0
0.50	0	0	0	0	0	0
0.45	0	0	0	0	0	0
0.40	0	0	0	0	0	0
0.35	0	0	0	0	0	0
0.30	1	0	0	0	0	0
0.25	6	0	0	0	0	0
0.20	9	0	4	0	2	0
0.15	11	0	7	0	6	0
0.10	17	0	11	0	10	0
0.05	25	0	18	0	14	0

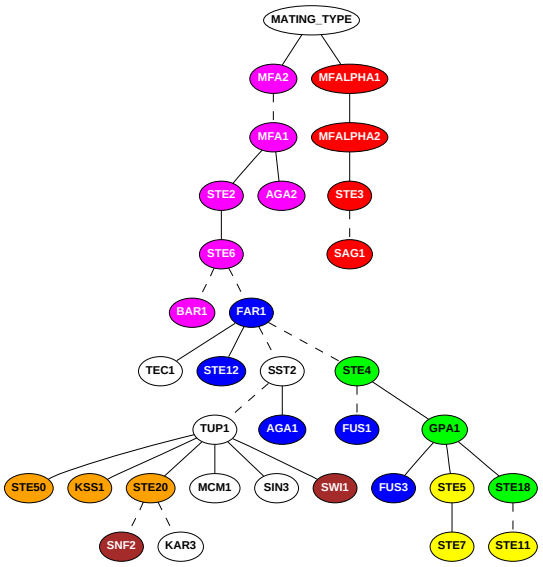


Fig. 3. Top, CPDAG of the best model learnt from the Yeast database ($k = 0.8$). Bottom left, trade-off between the number of FPs and FNs for undirected edges for the Yeast database at threshold values $t = 0.05 \cdot r$, $r = 1, \ldots, 20$. Bottom right, undirected edges for the Yeast database ($k = 0.8$) when $t = 0.60$ (solid and dashed edges) and when $t = 0.90$ (solid edges). Nodes are colored with the color of the functional group they belong to in Table 1

$k = 0.8$, and 159 for $k = 0.9$. In the three cases, the best model found was the same. Figure 3 (top) shows its CPDAG. We remark that the graph in the figure does not intend to represent the biological or physical GN, but the (in)dependencies in it. We note that all the edges in the CPDAG are undirected, meaning that each edge appears in opposite directions in at least two representative DAGs of the model. As a matter of fact, none of the CPDAGs of the different locally optimal models obtained in the 3000 runs of KES performed has directed edges. This reduces the types of features to study hereinafter to only two: Undirected edges and Markov blanket neighbors. However, when there is not any directed edge in a CPDAG, two nodes are Markov blanket neighbors if and only if they are connected by an undirected edge. Therefore, we only pay attention to undirected edges hereinafter.

We now discuss the results of the validation phase. Figure 3 (bottom left) shows the trade-off between the number of FPs and FNs for undirected edges as a function of the threshold value t. We note that the CPDAG of the best model found in the learning phase has 32 undirected edges. As can be appreciated from the figure for each value of k considered in the evaluation, FNs only happen for high values of t, while FPs only occur for low values of t. Therefore, TPs receive substantially higher confidence values than FPs. For $k = 0.8$, for instance, no TP scores lower than 0.60, while no FP scores higher than 0.25. These observations support the validity and meaningfulness of the best model discovered in the learning phase. Figure 3 (bottom right) depicts the undirected edges for $k = 0.8$ when $t = 0.60, 0.90$. We note that all the edges in the figure are TPs. As a matter of fact, there are 0 FPs and 0 FNs (32 TPs) for $t = 0.60$, and 0 FPs and 11 FNs (21 TPs) for $t = 0.90$. The figures for $k = 0.6, 0.9$ are similar to the one shown. We omit them for the sake of readability.

It is worth mentioning that we repeated the experiments in this section with a random database created by randomly reshuffling the entries of each attribute in the Yeast database. In such a database, we did not expect to find features scoring high confidence values. As a matter of fact, no edge was added in any of the 3000 runs of KES performed. This leads us to believe that the results presented above are not artifacts of the learning and validation phases but reliable findings. We give below further evidence that the (in)dependencies in the best model induced in the learning phase are consistent with the existing knowledge of yeast pheromone response. This somehow confirms the results of the validation phase, namely that the best model obtained in the learning phase is reliable.

We first discuss consistency with respect to the knowledge in Table 1. Magenta-colored genes are marginally dependent one on another as well as on MATING_TYPE. Moreover, no gene from other group mediates these dependencies. Likewise, red-colored genes are marginally dependent one on another as well as on MATING_TYPE, and no gene from other group mediates these dependencies. These observations are consistent with the fact that magenta-

colored genes express only in MATa cells, while red-colored genes express only in MATα cells. This also supports the fact that MATING_TYPE is the only node that mediates between magenta and red-colored genes. Green-colored genes are marginally dependent one on another, and no gene from other group mediates these dependencies. These observations are consistent with the fact that green-colored genes code for components of the heterotrimeric G-protein complex. Yellow-colored genes are marginally dependent one on another, which is consistent with these genes coding for core components of the signaling cascade. While STE5 and STE7 are adjacent, two green-colored genes (GPA1 and STE18) mediate between them and STE11. A similar result is reported in [14]. The authors conjecture that this finding may indicate common or serial regulatory control between green and yellow-colored genes. Orange-colored genes are marginally dependent one on another, which is consistent with the fact that they code for auxiliary components of the signaling cascade. Only TUP1 mediates these dependencies, specifically orange-colored genes are independent one of another given TUP1. As a matter of fact, TUP1 has the highest number of adjacencies in the model, which is consistent with its role as repressor of numerous genes in pheromone response pathway. We note that several nodes mediate between the core (yellow-colored) and the auxiliary (orange-colored) components of the signaling cascade. This agrees with [14]. The authors suggest that this finding may indicate that these two groups of genes have different regulatory mechanisms. Brown-colored genes are marginally dependent one on another, which is consistent with these genes coding for components of the SWI-SNF complex. However, TUP1 and STE20 mediate this dependency. A similar result is reported in [14]. Blue-colored genes are marginally dependent one on another, which is consistent with the promoters of these genes being bound by Ste12. However, several other genes mediate these dependencies.

We now discuss further evidence that does not appear in Table 1. The edges STE2—STE6, STE3—SAG1, and SST2—AGA1 are consistent with the genes connected by each edge being expressed similarly and being cell cycle-regulated [26]:[4] STE2 and STE6 peak at the M phase, while the rest of the genes peak at the M/G1 transition. Likewise, the genes connected by each of the edges MFALPHA2—STE3, MFA1—AGA2, and FAR1—TEC1 are also substantially correlated as well as cell cycle-regulated [26], though they do not peak at the same phase of the cell cycle (MFALPHA2 and MFA1 peak at the G1 phase, FAR1 at the M phase, and STE3, AGA2 and TEC1 at the M/G1 transition). The edge STE6—FAR1 is consistent with these genes being cell cycle-regulated, both peaking at the M phase [26]. The edge

[4] The cell cycle is the sequence of events by which the cell divides into two daughter cells and, thus, it is the biological basis of life. The cell cycle is divided into four main phases: G1, S, G2 and M. In G1 and G2, the cell grows and prepares to enter the next phase, either S or M. In S, the DNA is duplicated. In M, the actual cell division happens.

TUP1—MCM1 is consistent with the fact that these genes interact in the cell [13].

Finally, it is worth mentioning that most of the edges scoring high confidence values in the validation phase are supported by the existing knowledge of yeast pheromone response. For instance, most edges in Fig. 3 (bottom right) with confidence values equal of above 0.90 have been discussed in the paragraphs above. Therefore, we can conclude that the framework proposed in this paper for learning from data and validating BN models of GNs is accurate and reliable: The learning phase has produced a model that is consistent with the existing knowledge of the domain under study, and the validation phase has confirmed, independently of the existing knowledge, that the model is indeed meaningful.

6 Discussion

There exist numerous works showing that a BN model induced from gene expression data can provide accurate biological insight into the GN underlying the data [1,9,12,14,17,20,21,23]. This work is yet another example. However, learning BN models from data is a challenging problem (NP-complete and highly multimodal), specially if the learning data are as scarce and noisy as gene expression data are. For these reasons, any BN model of a GN obtained from gene expression data must be biologically validated before being accepted. Validating the model through biological experiments is expensive and, thus, the validation step typically reduces to checking whether the model agrees with the existing biological knowledge of the domain under study. Unfortunately, this way of proceeding condemns models providing true but new biological insight to be rejected. In this paper, we suggest a solution to this problem: We propose a method for checking whether the model learnt is statistically reliable, independently of the existence of biological knowledge. If the model fails to be reliable as a whole, we can instead report the features that are reliable, which are usually very informative. As a matter of fact, some of the works cited above focus on learning features with confidence value above a given threshold rather than on model selection [12,14,20]. A major limitation of this approach is that, in general, a set of features does not represent a (global) model of the probability distribution of the learning data but a collection of (local) patterns, because each feature corresponds to a piece of local information. Therefore, the reasoning about the (in)dependencies of the probability distribution of the learning data that a set of features allows is much less powerful than that of a model, e.g. a model can be queried about any (in)dependence statement but a set of features cannot. For this reason, we prefer our framework for model selection and validation and, only if the model selected does not pass the validation phase, we report features.

The works on learning features cited above use the methods in [10,14] to estimate the confidence in a feature. See [11] for yet another interesting method. Like our method, these methods assess the confidence in a feature as the fraction of models containing the feature out of a set of models. However, they differ from our method in how this set of models is obtained. In [10] it is obtained by running a greedy hill-climbing search on a series of bootstrap samples of the learning data, in [11] by Markov chain Monte Carlo simulation, and in [14] by selecting the highest scoring models visited during a simulated annealing search. No proof of asymptotic correctness is reported for any of these methods. We have proven that our method is asymptotically correct under the faithfulness assumption. The key in the proof is that our algorithm for model selection uses the inclusion boundary neighborhood, which takes into account all the representative DAGs of the current best model to produce the neighboring models. This is a major difference with the works on learning BN models of GNs cited at the beginning of this section, which use classical neighborhoods based on local transformations (single edge additions, removals and reversals) of a single representative DAG of the current best model. The inclusion boundary neighborhood outperforms the classical neighborhoods in practice without compromising the runtime, because it reduces the risk of getting stuck in a locally but not globally optimal model [3]. Moreover, unlike the classical neighborhoods, the inclusion boundary neighborhood allows to develop asymptotically optimal algorithms for model selection [3,5,19].

We are currently engaged in two lines of research. First, we are interested in replacing the faithfulness assumption by a weaker assumption such as the composition property assumption. Second, we would like to use the results of the validation phase to design informative gene perturbations, gather new data, and refine the models obtained in the learning phase accordingly. We hope that by influencing the data collection process we will reduce the amount of data required for learning a reliable model. This is important given the high cost of gathering gene expression data. Moreover, combining observational and interventional data will also provide insight into the causal relations in the GN under study.

Acknowledgements

We thank Alexander J. Hartemink for providing us with the Yeast database. This work is funded by the Swedish Foundation for Strategic Research (SSF) and Linköping Institute of Technology.

References

1. Bernard, A. and Hartemink, A. J. (2005) Informative Structure Priors: Joint Learning of Dynamic Regulatory Networks from Multiple Types of Data. In Pacific Symposium on Biocomputing 10.

2. Blake, W. J., Kærn, M., Cantor, C. R. and Collins, J. J. (2003) Noise in Eukaryotic Gene Expression. Nature 422:633-637.
3. Castelo, R. and Kočka, T. (2003) On Inclusion-Driven Learning of Bayesian Networks. Journal of Machine Learning Research 4:527-574.
4. Chickering, D. M. (1996) Learning Bayesian Networks is NP-Complete. In Learning from Data: Artificial Intelligence and Statistics V:121-130.
5. Chickering, D. M. (2002) Optimal Structure Identification with Greedy Search. Journal of Machine Learning Research 3:507-554.
6. Cooper, G. and Herskovits, E. H. (1992) A Bayesian Method for the Induction of Probabilistic Networks from Data. Machine Learning 9:309-347.
7. Costanzo, M. C., Crawford, M. E., Hirschman, J. E., Kranz, J. E., Olsen, P., Robertson, L. S., Skrzypek, M. S., Braun, B. R., Hopkins, K. L., Kondu, P., Lengieza, C., Lew-Smith, J. E., Tillberg, M. and Garrels, J. I. (2001) YPD™, PombePD™ and WormPD™: Model Organism Volumes of the BioKnowledge™ Library, an Integrated Resource for Protein Information. Nucleic Acids Research 29:75-79.
8. Elion, E. A. (2000) Pheromone Response, Mating and Cell Biology. Current Opinion in Microbiology 3:573-581.
9. Friedman, N. (2004) Inferring Cellular Networks Using Probabilistic Graphical Models. Science 303:799-805.
10. Friedman, N., Goldszmidt, M. and Wyner, A. (1999) Data Analysis with Bayesian Networks: A Bootstrap Approach. In Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence 196-205.
11. Friedman, N. and Koller, D. (2003) Being Bayesian About Network Structure. A Bayesian Approach to Structure Discovery in Bayesian Networks. Machine Learning 50:95-125.
12. Friedman, N., Linial, M., Nachman, I. and Pe'er, D. (2000) Using Bayesian Networks to Analyze Expression Data. Journal of Computational Biology 7:601-620.
13. Gavin, I. M., Kladde, M. P. and Simpson, R. T. (2000) Tup1p Represses Mcm1p Transcriptional Activation and Chromatin Remodeling of an a-Cell-Specific Gene. The EMBO Journal 19:5875-5883.
14. Hartemink, A. J., Gifford, D. K., Jaakkola, T. S. and Young, R. A. (2002) Combining Location and Expression Data for Principled Discovery of Genetic Regulatory Network Models. In Pacific Symposium on Biocomputing 7:437-449.
15. Heckerman, D., Geiger, D. and Chickering, D. M. (1995) Learning Bayesian Networks: The Combination of Knowledge and Statistical Data. Machine Learning 20:197-243.
16. Herskovits, E. H. (1991) Computer-Based Probabilistic-Network Construction. PhD Thesis, Stanford University.
17. Imoto, S., Goto, T. and Miyano, S. (2002) Estimation of Genetic Networks and Functional Structures Between Genes by Using Bayesian Network and Nonparametric Regression. In Pacific Symposium on Biocomputing 7:175-186.
18. McAdams, H. H. and Arkin, A. (1997) Stochastic Mechanisms in Gene Expression. In Proceedings of the National Academy of Science of the USA 94:814-819.
19. Nielsen, J. D., Kočka, T. and Peña, J. M. (2003) On Local Optima in Learning Bayesian Networks. In Proceedings of the Nineteenth Conference on Uncertainty in Artificial Intelligence 435-442.
20. Pe'er, D., Regev, A., Elidan, G. and Friedman, N. (2001) Inferring Subnetworks from Perturbed Expression Profiles. Bioinformatics 17:S215-S224.

21. Peña, J. M., Björkegren, J. and Tegnér, J. (2005) Growing Bayesian Network Models of Gene Networks from Seed Genes. Bioinformatics 21:ii224-ii229.
22. Ren, B., Robert, F., Wyrick, J. J., Aparicio, O., Jennings, E. G., Simon, I., Zeitlinger, J., Schreiber, J., Hannett, N., Kanin, E., Volkert, T. L., Wilson, C. J., Bell, S. P. and Young, R. A. (2000) Genome-Wide Location and Function of DNA Binding Proteins. Science 290:2306-2309.
23. Sachs, K., Perez, O., Pe'er, D., Lauffenburger, D. A. and Nolan, G. P. (2005) Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data. Science 308:523-529.
24. Schwarz, G. (1978) Estimating the Dimension of a Model. Annals of Statistics 6:461-464.
25. Sebastiani, P., Gussoni, E., Kohane, I. S. and Ramoni, M. (2003) Statistical Challenges in Functional Genomics (with Discussion). Statistical Science 18:33-60.
26. Spellman, P. T., Sherlock, G., Zhang, M. Q., Iyer, V. R., Anders, K., Eisen, M. B., Brown, P. O., Botstein, D. and Futcher, B. (1998) Comprehensive Identification of Cell Cycle-Regulated Genes of the Yeast *Saccharomyces cerevisiae* by Microarray Hybridization. Molecular Biology of the Cell 9:3273-3297.
27. Spirtes, P., Glymour, C. and Scheines, R. (1993) Causation, Prediction, and Search. Springer-Verlag, New York.
28. Studený, M. (2003) Characterization of Inclusion Neighbourhood in Terms of the Essential Graph: Upper Neighbours. In Proceedings of the Seventh European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty 161-172.
29. Studený, M. (2003) Characterization of Inclusion Neighbourhood in Terms of the Essential Graph: Lower Neighbours. In Proceedings of the Sixth Workshop on Uncertainty Processing 243-262.