Using Bayesian networks to predict some regulatory systems components of Escherichia coli from microarray gene expression data

Diego Garcia and Irene Tischer

November 10, 2017

Abstract

Identification and analysis of signature genes and its interactions are keys on pharmacist design process, improve physiological of plants and research of disease as oncogenesis, among others. it is possible to analyze the collective behavior of genes using models that represent the interaction between them. Some of these models are gene co-expression networks and gene regulatory networks; where we can observe the expression patterns that follow gene groups under given biological conditions and the impact of the behavior of one group of genes over other. The networks in question have been represented with models like Bayesian network and we can infer its structure and parameters using statistical methods like Markov Chain Monte Carlo simulation (MCMC) and Bayesian inference.

Purpose of this work is to valid the useful of Bayesian networks to predict several bacterial regulatory system components: First, we obtain gene expression data of 4297 genes of Escherichia coli and 466 observations in several biological conditions. Selection genes process was done using gene coexpression networks and identifying signature genes across of caracterization of modules to functional level and topological, so we use a subset of modular genes and then discretize the data into three categories. Second, we learned Bayesian networks of gene interactions adopting MCMC simulation and Bayesian inference methods. Specifically, we use Metropolis Hasting algorithm and Posterior distribution to approximate the parameters and structure of network. To avoid overfitting in the learning we include two constraints during simulations: maximum ingrade in nodes and minimum dependence in edges of network. Third, we compare the genes interactions that we found with some published on ABASY for modular genes in bacterial regulatory systems specifically, Escherichia coli. Finally, we found Bayesian networks that correspond to components of modular genes like lactose, sodium ion transport and phosphorelay signal transduction system in Escherichia coli.

In this work, we see that Bayesian networks can help us to understand some bacterial regulatory systems components, mainly components of modular genes, however, only its learning method from microarray gene

expression data can't say us about other components such as global regulators, inter-modular regulators and basal machinery.

1 Background

1.1 Bayesian networks

A Bayesian network is a directed acyclic graph (DAG), in which nodes represent random variables and edges denote dependencies between them (see [22, 23]). Such graph can be defined as a pair $B = (G, \Theta)$; where $G = (V, E)$, V denote a set of nodes, E a set of edges and Θ a vector of conditional probabilities. We can say that v_p is parent of v_i for all $v_c, v_p \in V$ with $c \neq p$ if there is an edge from v_p to v_c. The set of all parents of v_i will be denote $Pa(v_i)$ and the joint distribution of the variables $V = \{\nu_1, ..., \nu_d\}$ can be specified by decomposition:

$$P(V) = \prod_{i=1}^{d} P(\nu_i | Pa(\nu_i)) \qquad (1)$$

On the other hand, DNA microarrays are a technology that allows monitoring broad levels of gene expression for organism given; these experiments concurrently measure the expression level of thousands of genes. An improtant problem in computational biology is to discover, from such measurements, gene interaction networks and key biological features of cellular systems (see [5]). To learn a Bayesian networks from gene expression data is often used to address this problem (see [22]).

Now, let v_i a random variable such that:

$$v_i = \begin{cases} 0 & ,if \quad v_i < Q_{i,\frac{1}{3}} \\ 1 & ,if \quad Q_{i,\frac{1}{3}} \leq v_i < Q_{i,\frac{2}{3}} \\ 2 & ,if \quad v_i \geq Q_{i,\frac{2}{3}} \end{cases}$$

where $Q_{i,\frac{1}{3}}$ y $Q_{i,\frac{2}{3}}$ are $\frac{1}{3}$ and $\frac{2}{3}$ quantiles of expression values in i^{th} gene; for $i = 1, ..., d$, so each variable v_i will have a conditional probability table (CPT) with $q_i = \prod_{v_p \in Pa(v_i)} 3$ rows, each entry correspond to a different combination of values in v_i parent variables, in addition, each row contain a probabilities vector $\theta_{i,j,k}$ with the probability value that $v_i = j$ given than $Pa(v_i)$ variables take the values of k^{th} entry of the table. Therefore, Bayesian network for a fixed structure G will be parametrized with $\Theta = \{\theta_{ijk} > 0 : \sum_j \theta_{ijk} = 1\}$.

Given N independent observations $X = \{X_1, ..., X_N\}$ obtained from (1), the sufficient statistics for Θ is the set of counts $\{N_{ijk}\}$ where N_{ijk} equals the number of times when $v_i = j$ given than $Pa(v_i)$ variables take values specificated in k^{th} entry in CPT of v_i (see [6]), so counts $\{N_{ijk}, j = 0, 1, 2\}$ follows a multinomial distribution with parameters $N_{i.k} = N_{i0k} + N_{i1k} + N_{i2k}$ (trials) and $\{\theta_{i0k}, \theta_{i1k}, \theta_{i2k}\}$ (probability vector). Therefore, likelihood function of our Bayesian network model is a product of multinomials

$$P\left(X|G,\Theta\right) = \prod_{i=1}^{d}\prod_{k=1}^{q_i} Multinomial(N_{i.k}, \theta_{i0k}, \theta_{i1k}, \theta_{i2k}) \tag{2}$$

where $Multinomial(N_{i.k}, \theta_{i0k}, \theta_{i1k}, \theta_{i2k}) = \binom{N_{i.k}}{N_{i0k}, N_{i1k}, N_{i2k}} \theta_{i0k}^{N_{i0k}}.\theta_{i1k}^{N_{i1k}}.\theta_{i2k}^{N_{i2k}}$.

In Bayesian probability theory, if the posterior distributions $P(\Theta|X)$ are in the same family as the prior probability distribution $P(\Theta)$, the prior and posterior are then called conjugate distributions, and the prior is called a conjugate prior for the likelihood function (see [25, 3, 9]). So for a multinomial likelihood with parameters $N_{i.k}$ (trials) and $\{\theta_{i0k}, \theta_{i1k}, \theta_{i2k}\}$ (probability vector) conjugate prior distribution is Dirichlet with hyperparameters $\alpha_{i.k} = \{\alpha_{i0k}, \alpha_{i1k}, \alpha_{i2k}\}$ and posterior hyperparameters $\{N_{i0k} + \alpha_{i0k}, N_{i1k} + \alpha_{i1k}, N_{i2k} + \alpha_{i2k}\}$. Therefore, posterior of our Bayesian network model is a product of Dirichlets

$$P(G,\Theta|X) = \prod_{i=1}^{d}\prod_{k=1}^{q_i} Diritchlet(N_{i0k} + \alpha_{i0k}, N_{i1k} + \alpha_{i1k}, N_{i2k} + \alpha_{i2k}) \tag{3}$$

where

$$Diritchlet(N_{i0k} + \alpha_{i0k}, N_{i1k} + \alpha_{i1k}, N_{i2k} + \alpha_{i2k}) =$$
$$\frac{\theta_{i0k}^{N_{i0k}+\alpha_{i0k}-1}.\theta_{i1k}^{N_{i1k}+\alpha_{i1k}-1}.\theta_{i2k}^{N_{i2k}+\alpha_{i2k}-1}}{\beta(N_{i0k} + \alpha_{i0k}, N_{i1k} + \alpha_{i1k}, N_{i2k} + \alpha_{i2k})},$$

$$\theta_{i0k} \approx E[\theta_{i0k}] = \frac{N_{i0k} + \alpha_{i0k}}{(N_{i0k} + \alpha_{i0k} + N_{i1k} + \alpha_{i1k} + N_{i2k} + \alpha_{i2k})},$$

$$\theta_{i1k} \approx E[\theta_{i1k}] = \frac{N_{i1k} + \alpha_{i1k}}{(N_{i0k} + \alpha_{i0k} + N_{i1k} + \alpha_{i1k} + N_{i2k} + \alpha_{i2k})},$$

$$\theta_{i2k} \approx E[\theta_{i2k}] = \frac{N_{i2k} + \alpha_{i2k}}{(N_{i0k} + \alpha_{i0k} + N_{i1k} + \alpha_{i1k} + N_{i2k} + \alpha_{i2k})}$$

and

$$\beta(N_{i0k} + \alpha_{i0k}, N_{i1k} + \alpha_{i1k}, N_{i2k} + \alpha_{i2k}) =$$
$$\frac{\Gamma(N_{i0k} + \alpha_{i0k})\Gamma(N_{i1k} + \alpha_{i1k})\Gamma(N_{i2k} + \alpha_{i2k})}{\Gamma(N_{i0k} + \alpha_{i0k} + N_{i1k} + \alpha_{i1k} + N_{i2k} + \alpha_{i2k})}.$$

1.2 MCMC simulation

Markov Chain Monte Carlo (MCMC) is an area of statistics wich provides answer to problem of simulations in high dimensional probability distributions. A Markov chain is defined on terms of graph states over which a stochastic process carry out a random walk. This process often tend to balance and its states follow a probability distribution. MCMC techniques allow simulation of a probability distribution embed in the distribution of one Markov chain and then carry out simulation of such chain until tends to balance. MCMC simulation are used as approach to Bayesian learning in wich a viable Bayesian nerworks set are explored across of a random walk that converges to a optimal networks set that satisfy a accept criteria determined (see [8, 16, 20, 17, 18, 10]).

Metropolis Hasting (MH, see [11]) is an MCMC algorithm which is based in a Markov chain whose dependence with its previous states has two parts: proposed distribution and acceptance of proposal. Proposed probability distribution suggest to next step of the walk in the Markov chain and acceptance of proposal keeps the suitable course and rejects undesirable moves in such chain. We used the approach MCMC simulations to learn Bayesian network based on [15] and we simulate a Markov chain over the space of feasible network structures whose stationary distribution is the posterior distribution of the network (see equation 3).

1.3 Gene coexpression networks

Before of pass to selection of genes across this method, we review networks definition in biological context and gene coexpression networks. A network or graph is a set of points called nodes linked across of lines called edges which denote some type of relation. In biological ambit is possible to represent relations between biomolecules through networks, so that each node correspond for instance to a gene or protein and edges some interaction between them. Such networks has been used on biology to represent metabolic pathways, gene coexpressions, gene regulatory relations, protein interactions and others [1, 26]. A gene coexpression networks (GCN) is a graph undirected , in which nodes represent genes (specifically, its expression profiles) and edges denote coexpression relations, that is, expression of two or more genes, simultaneously, so two genes are connected if behavior of its expression profiles shows relation. This networks provides information about association relations, expression similarity and neighborhood among genes, so allows us to infer interactions between them. Commonly, such networks are constructed from expression data microarrays of DNA and it can be of two categories depending on connection type between nodes: unweighted networks which edges denote association or not association among nodes and weighted networks which quantify us the grade of association among nodes across of attribute called weight often in range [0,1].

An application of gene coexpression networks is identification of node groups highly coexpressed called *modules*. Such modules indicates common functions among its genes. Zhang and Horvath [28] define a method to construct weighted networks known as "Weighted Gene Coexpression Network Analysis" (WGCNA) which uses Pearson correlation coefficient like measure of coexpression among genes. This method consists on creation of a matrix that saves the grade of connection between nodes pairs:

1. Preparation of expression data.
2. Definition of similarity measure among nodes.
3. Definition of adjacency function among modes.
4. Definition of adjacency function parameters.
5. Definition of unsimilarity measure among nodes.
6. Identification of modules.

The method authors have devoloped a package on R language that has different functions to construct and analysis of gene coexpression networks. This networks can be visualized with applications like Cytoscape (`http://www.cytoscape.org`), Gephi (`http://gephi.github.io`) o gViz (`http://urbm-cluster.urbm.fundp.ac.be/webapps/gviz`). Recently, WGCNA has been applying over RNA-Seq data to selection of genes (see [27, 4, 12]).

2 Methods

2.1 Selecting cluster of genes from gene co-expression network

Before applying Bayesian learning, we must get a gene co-expression network then select one or more network's cluster. Selection can be queried by biologist depending of study or also we can do it randomly or deterministically.

Firsts, we need obtain microarray gene expression data from a public source like GEO [2], ArrayExpress [14], Colombos [19] or [7], among others. After, we can calculate correlation coefficient (like Pearson or Kendall) between each pair of expression vector of genes selected then we joint it to get a correlation matrix that will represent a gene co-expression network. Last, we must choose a cutoff for correlation coefficient values and each cluster will be conformed with those pair genes that exceeds cutoff threshold chosen. Constructing of gene co-expression network and modules detection was done using Weighted Gene Co-expression Network Analysis (WGCNA) method by Zhang and Horvath [28].

Selection genes process using gene coexpression networks consists of identify signature genes across of caracterization of modules to functional level and its topology. For instance, *connectivity* denote connection grade of genes and *betweenness centrality* denote grade that a gene serves of bridge among others [24], allow us detect *hubs* (genes highly connected) and bottlenecks, respectively, such genes are often relationed tofunctional process. So, we can select genes more interesting topological and biologically.

2.2 *MCMC simulation to learn Bayesian networks from data*

MH algorithm (see Algorithm 1) was adapted to learn Bayesian networks of the following manner:

- The states (x) of Markov chain correspond to possible network structures (G).
- Stationary distribution ($\tilde{p}(x)$) is the posterior distribution of the network (see equation 3).
- Proposed distribution ($q(x'|x)$) is a uniform distribution calculated over the length of the search space.
- Now we describe how to find a new structure in *The Search Space.* First, we define the connectivity of our search space in terms of operators such as:

  1. Edge addition to current structure.
  2. Edge deletion to current structure.
  3. Edge deletion + Edge addition (double operation) to current structure.

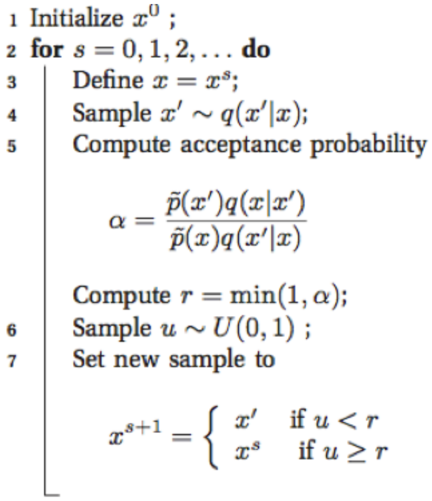
  Calculus of search space: For get valid structures (acyclic), we suppose that A is the adjacency matrix that represent a structure G of Bayesian network.

  1. *For Edge addition*: positions in P_A *matrix* with value 0 will indicate us nodes pairs of the network where is feasible add a edge to get a acyclic graph G' from G.
     $P_A = IdentityMatrix + A + Transpose(A) + Transpose(A^2) + .. + Transpose(A^{N-1})$
  2. *For Edge deletion*: position in A with value 1 will indicate us nodes pairs of the network where is feasible delete a edge from G.

  Once we have defined the search space, *search procedure* will allow us to explore it and search for feasibles structures, so MH algorithm adapted (see algorithm 1) as follow:

  1. We pick an initial network structure G as a starting point; this network can be empty one $\mathcal{G}_\emptyset$.
  2. We compute equation 3 from G , this is $\tilde{p}(G) = P(G, \Theta|X)$.
  3. We then consider all of the neighbors of G in the space (all of the legal networks obtained by applying a single operator to G) and take a random walk between them, this is we get G'. We employ a uniform distribution calculated with the length of set of positions obtained from the search space, previously:
     $q(G'|G) = \frac{1}{length(P_A)+length(P_B)}$

Algorithm 1 Metropolis Hastings algorithm [20].



```
1 Initialize $x^0$ ;
2 for $s = 0, 1, 2, \ldots$ do
3     Define $x = x^s$;
4     Sample $x' \sim q(x'|x)$;
5     Compute acceptance probability
          $\alpha = \frac{\tilde{p}(x')q(x|x')}{\tilde{p}(x)q(x'|x)}$
      Compute $r = \min(1, \alpha)$;
6     Sample $u \sim U(0, 1)$ ;
7     Set new sample to
          $x^{s+1} = \begin{cases} x' & \text{if } u < r \\ x^s & \text{if } u \geq r \end{cases}$
```

4. We compute equation 3 from G' , this is $\tilde{p}(G') = P(G', \Theta|X)$.
5. Evaluate proposal (criteria MH, step 5 in algorithm 1).
6. Go to step 2. (now with $G = G^{s+1}$), until when step 2 (in algorithm 1) has finished.

2.3 Applying Bayesian framework to each genes cluster

Now, for each cluster of genes selected, we proceed loading its samples from data of expression, later we apply our Bayesian learning method to the samples loaded and next we take the results and graph its networks and traces, respectively.

In this work, each gene expression vector must be submitted to a discrete process before Bayesian learning stage. We will use quantiles discretization method with three levels (quantiles). See [5].

Our Bayesian learning method consist on an adaptation of Metropolis Hasting (MH) algorithm. Such adaptation consists on doing a random walking over search space of possible structures and evaluate it from posterior distribution to data with accept criteria of MH algorithm [15].

Once finished MCMC simulation, we analyze results of each iteration looking those graphs that have greatest posterior density function value (score), among other properties (like minimum in-grade in its nodes and maximum conditional dependence in its edges).

Finally, we convert the sub-set of graphs selected to a weighted network and gene interactions proposed was compared with gene interactions on ABASY for Escherichia coli [13]. i.e. we do it averaging occurrence of each edge over cardinality of sub-set and again we choose a cutoff for weight values and weighted network will be conformed with those edges that exceeds cutoff threshold chosen.

3 Results

3.1 Gene co-expression network

.Here, we select a subset of 16 genes that correspond to 5 different modules and its global regulators, see (Strong Evidences) Escherichia coli str. K-12 substr. MG1655 [13]:

- Module 34 – Lactose transport (GO:0015767)
  lacA – galactoside O-acetyltransferase monomer
  lacI – LacI DNA-binding transcriptional repressor
  lacY – lactose / melibiose:H+ symporter LacY
  lacZ – β-galactosidase monomer
  Global regulators:
  rpoD – RNA polymerase, sigma 70 (sigma D) factor
  hns – H-NS DNA-binding transcriptional dual regulator
  crp – CRP transcriptional dual regulator

- Module 37 – Zinc ion transport (GO:0006829)
  znuA – Zn2+ ABC transporter - periplasmic binding protein

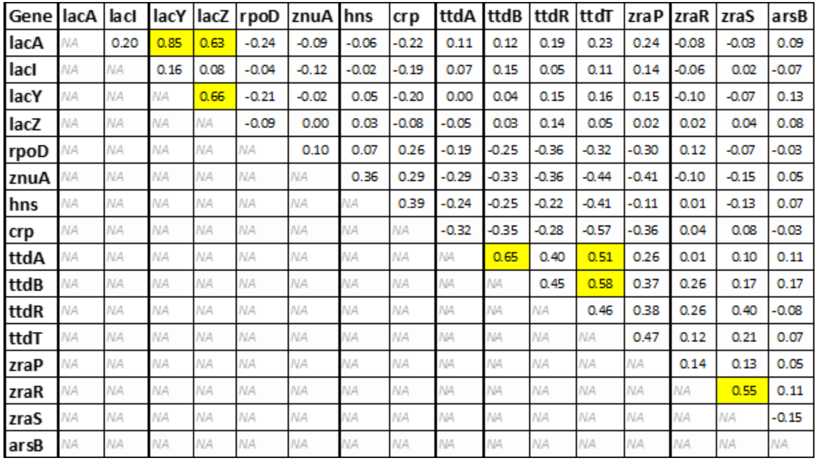
- Module 40 – Sodium ion transport (GO:0006814)
  ttdA – L-tartrate dehydratase, α subunit
  ttdB – L-tartrate dehydratase, β subunit
  ttdR – Dan
  ttdT – tartrate:succinate antiporter

- Module 52 – Phosphorelay signal transduction system (GO:0000160)
  zraP – zinc homeostasis protein
  zraR – ZraR transcriptional activator
  zraS – ZraS sensory histidine kinase

- Module 55 – Response to arsenic-containing substance (GO:0046685)
  arsB – ArsB

Constructing of gene co-expression network was done jointing the correlation coefficient values calculated using the function “cor” of R with “pearson” argument. Gene expression data was taken from M^{3D} [7] and the resulting matrix is the following: The values highlighted in yellow (see table 1) correspond to a cutoff of 0.5, i.e. values greater than 0.5 are into threshold and its genes will conform the clusters, so clusters selected are the following:

Table 1: Adjacency matrix of gene co-expression network.



Gene	lacA	lacI	lacY	lacZ	rpoD	znuA	hns	crp	ttdA	ttdB	ttdR	ttdT	zraP	zraR	zraS	arsB
lacA	NA	0.20	0.85	0.63	-0.24	-0.09	-0.06	-0.22	0.11	0.12	0.19	0.23	0.24	-0.08	-0.03	0.09
lacI	NA	NA	0.16	0.08	-0.04	-0.12	-0.02	-0.19	0.07	0.15	0.05	0.11	0.14	-0.06	0.02	-0.07
lacY	NA	NA	NA	0.66	-0.21	-0.02	0.05	-0.20	0.00	0.04	0.15	0.16	0.15	-0.10	-0.07	0.13
lacZ	NA	NA	NA	NA	-0.09	0.00	0.03	-0.08	-0.05	0.03	0.14	0.05	0.02	0.02	0.04	0.08
rpoD	NA	NA	NA	NA	NA	0.10	0.07	0.26	-0.19	-0.25	-0.36	-0.32	-0.30	0.12	-0.07	-0.03
znuA	NA	NA	NA	NA	NA	NA	0.36	0.29	-0.29	-0.33	-0.36	-0.44	-0.41	-0.10	-0.15	0.05
hns	NA	NA	NA	NA	NA	NA	NA	0.39	-0.24	-0.25	-0.22	-0.41	-0.11	0.01	-0.13	0.07
crp	NA	NA	NA	NA	NA	NA	NA	NA	-0.32	-0.35	-0.28	-0.57	-0.36	0.04	0.08	-0.03
ttdA	NA	NA	NA	NA	NA	NA	NA	NA	NA	0.65	0.40	0.51	0.26	0.01	0.10	0.11
ttdB	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	0.45	0.58	0.37	0.26	0.17	0.17
ttdR	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	0.46	0.38	0.26	0.40	-0.08
ttdT	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	0.47	0.12	0.21	0.07
zraP	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	0.14	0.13	0.05
zraR	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	0.55	0.11
zraS	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	-0.15
arsB	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA

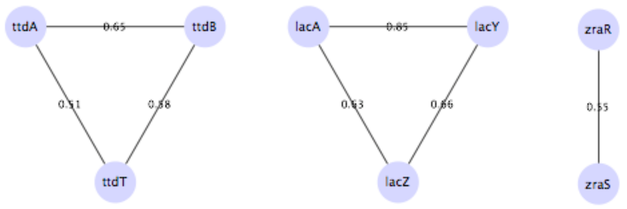


Figure 1: Clusters selected from gene co-expression network.

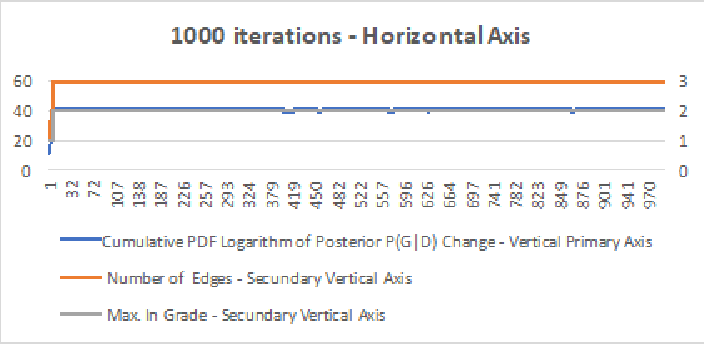


Figure 2: Simulation of 1000 iterations for lacA, lacY and lacZ genes.

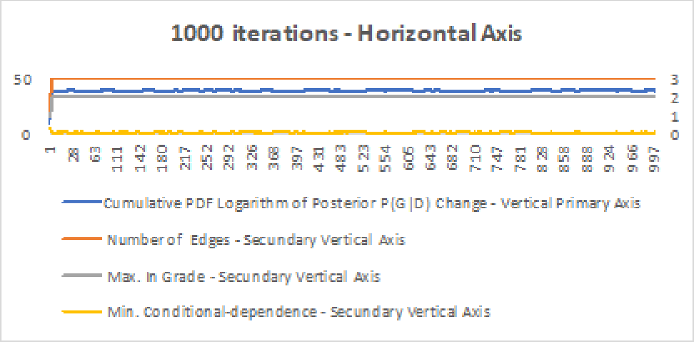


Figure 3: Simulation of 1000 iterations for ttdA, ttdB and ttdT genes.

3.2 Bayesian learning

Clusters showed in Figure 1 are the input for Bayesian learning but before we must discretize its expression values with the “discretize” function in the “bnlearn” package of R. See [21]. Later, we execute three simulations of 1000 iterations, one per cluster, and we obtain the following results: We can see in the figures of above that optimal level of the density function posterior to the data is achieved during the firsts iterations in all simulations. Also, we can observe that the number of edges and the maximum in-grade are constant during simulations.

In summary, we obtained 6 different Bayesian networks in the first and second simulation and 2 Bayesian networks in third simulation.

3.3 Weighted network

In previous sub-section, we obtained several Bayesian networks as result of learning process. Now we take the sub-set of networks and averaging each edge over total of networks obtained, we get a weighted network per simulation. The

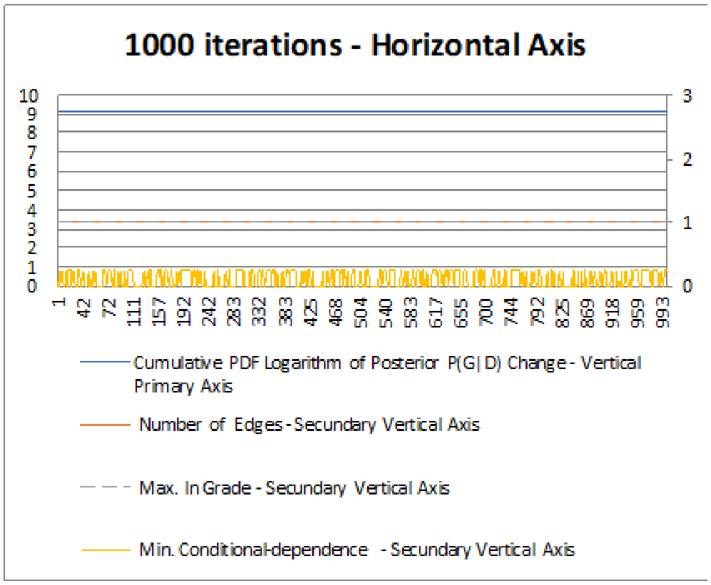


Figure 4: Simulation of 1000 iterations for zraR and zraS genes.

result is shown to continuation:

4 Discussion

4.1 Relation between co-expression network and Bayesian network

We can observe in table 1 that co-expressed genes inside each cluster also it present dependence relations between them, as it is shown in figure 5. If we see this from point of view mathematics seems logic that two variables whose

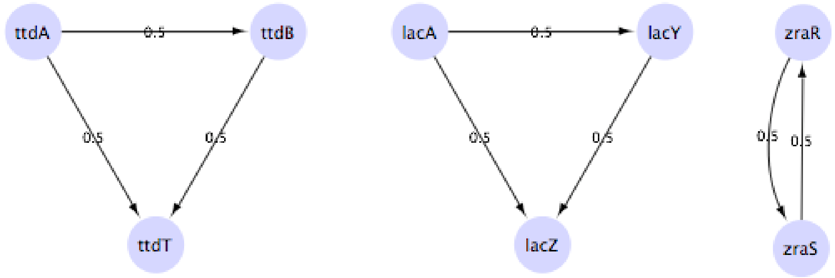


Figure 5: Weighted networks learned corresponding to modules 40, 34 and 52 of E. coli.

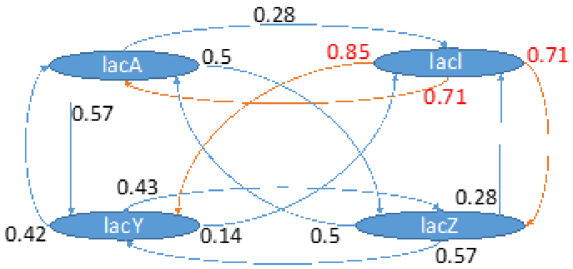


Figure 6: Weighted network of module 34 after of adding lacI regulator gene.

values present high correlation, also it shows conditional dependence, talking probabilistically, i.e. these results are an effect numeric and no biological. May be this the reason because from Bayesian paradigm of statistics is not enough for to predict other bacterial regulatory system component like regulators or basal machinery if not there are high numeric correlation between them from expression data. In conclusion, we recommend articulate other machine learning approaches to complement this paradigm and achieve better predictions.

4.2 Trend to overfitting when we try select more than one cluster

Other observation is trend to overfitting in the Bayesian learning process. For instance, we try adding other component (lacI regulator gene) in the simulation of the lacA, lacY and lacZ genes and we saw the addition of edges from lacI to each modular gene. The figure 6 shows the weighted network of the simulation. The same effect is observed when adding rpoD and znuA genes (see figure 7). Overfitting can become chronic if we include more than one cluster in one simulation. The figure 8 shows this situation. In conclusion, we recommend highly to apply Bayesian learning process only to clusters of gene co-expression networks to avoid overfitting of Bayesian networks.

Conclusions and future work

- We found Bayesian networks that correspond to some components of modular genes like Lactose in Escherichia coli, among others. Predictions was successfully confirmed with respect to gene interactions published.

- We see that Bayesian networks can help us to understand some bacterial regulatory systems components, mainly components of modular genes, however, its learning method only from microarray gene expression data cannot say us something about other components such as global regulators, inter-modular regulators or basal machinery.

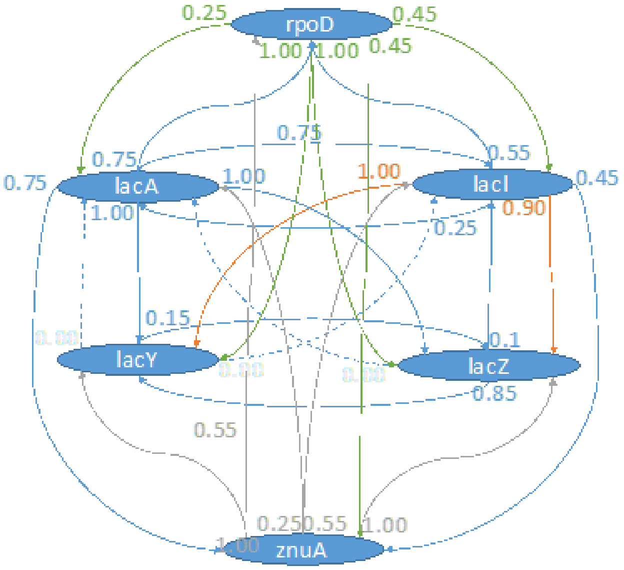


Figure 7: Weighted network of module 34 after adding rpoD and znuA genes.

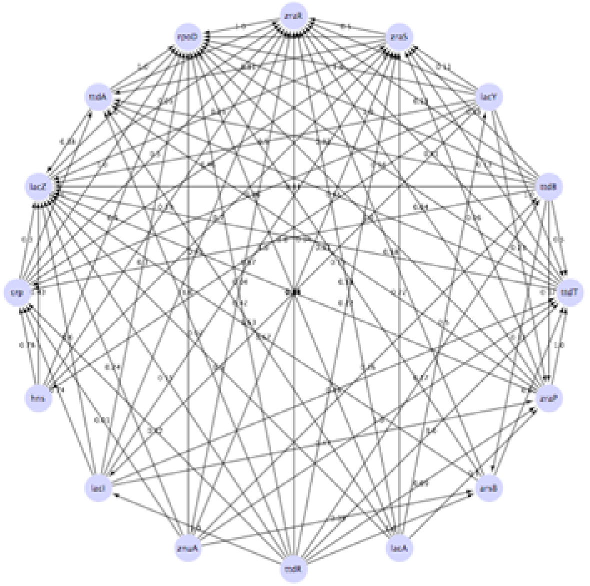


Figure 8: Trend to overfitting when several clusters are included inside one MCMC simulation.

- A future work may be analyzing possibility of articulate Bayesian learning with other learning approaches like network science, neural networks, fractal analysis, contact maps or dependences networks (based on constraints) to do prediction of other bacterial regulatory system components.

References

[1] Albert-László Barabási and Zoltán N. Oltvai. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5:101 EP –, 02 2004.

[2] Tanya Barrett, Stephen E Wilhite, Pierre Ledoux, Carlos Evangelista, Irene F Kim, Maxim Tomashevsky, Kimberly A Marshall, Katherine H Phillippy, Patti M Sherman, Michelle Holko, Andrey Yefanov, Hyeseung Lee, Naigong Zhang, Cynthia L Robertson, Nadezhda Serova, Sean Davis, and Alexandra Soboleva. Ncbi geo: archive for functional genomics data sets—update. *Nucleic Acids Research*, 41(Database issue):D991–D995, 01 2013.

[3] J.M. Bernardo and A.F.M. Smith. *Bayesian Theory*. Wiley Series in Probability and Statistics. John Wiley & Sons Canada, Limited, 2006.

[4] Rebecca Davidson, Candice N. Hansey, Malali Gowda, Kevin L. Childs, Haining Lin, Brieanne Vaillancourt, Rajandeep Sekhon, Natalia de Leon, Shawn Kaeppler, Ning Jiang, and C Robin Buell. Utility of rna sequencing for analysis of maize reproductive transcriptomes. 4:191–203, 08 2011.

[5] Dey, D.K. and Ghosh, S. and Mallick, B.K. *Bayesian Modeling in Bioinformatics*. Chapman & Hall/CRC Biostatistics Series. CRC Press, 2010.

[6] Byron Ellis and Wing Hung Wong. Learning causal bayesian network structures from experimental data. *Journal of the American Statistical Association*, 103(482):778–789, 2008.

[7] Jeremiah J Faith, Michael E Driscoll, Vincent A Fusaro, Elissa J Cosgrove, Boris Hayete, Frank S Juhn, Stephen J Schneider, and Timothy S Gardner. Many microbe microarrays database: uniformly normalized affymetrix compendia with structured experimental metadata. *Nucleic Acids Research*, 36(Database issue):D866–D870, 01 2008.

[8] Nir Friedman and Daphne Koller. Being bayesian about network structure. a bayesian approach to structure discovery in bayesian networks. *Machine Learning*, 50(1-2):95–125, 2003.

[9] Andrew Gelman, John B. Carlin, Hal S. Stern, and Donald B. Rubin. *Bayesian Data Analysis, Second Edition (Chapman & Hall/CRC Texts in Statistical Science)*. Chapman and Hall/CRC, 2 edition, July 2003.

[10] P Giudici and PJ Green. Decomposable graphical gaussian model determination. *Biometrika*, 86(4):785–801, 1999.

[11] W. K. HASTINGS. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[12] Courtney Hollender, Chunying Kang, Omar Darwish, Aviva Geretz, Benjamin Matthews, Janet Slovin, Nadim Alkharouf, and Zhongchi Liu. Floral transcriptomes in woodland strawberry uncover developing receptacle and anther gene networks. 165, 05 2014.

[13] Miguel A. Ibarra-Arellano, A. I. Campos-G., Luis G. T.-Quintanilla, Andreas Tauch, and Julio A. Freyre-G. Abasy atlas: a comprehensive inventory of systems, global network properties and systems-level elements across bacteria. *Database*, 2016:baw089, 2016.

[14] Nikolay Kolesnikov, Emma Hastings, Maria Keays, Olga Melnichuk, Y Amy Tang, Eleanor Williams, Miroslaw Dylag, Natalja Kurbatova, Marco Brandizi, Tony Burdett, Karyn Megy, Ekaterina Pilicheva, Gabriella Rustici, Andrew Tikhonov, Helen Parkinson, Robert Petryszak, Ugis Sarkans, and Alvis Brazma. Arrayexpress update—simplifying data submissions. *Nucleic Acids Research*, 43(Database issue):D1113–D1116, 01 2015.

[15] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[16] K.B. Korb and A.E. Nicholson. *Bayesian Artificial Intelligence, Second Edition*. Chapman & Hall/CRC Computer Science & Data Analysis. CRC Press, 2010.

[17] David Madigan and Adrian E Raftery. Model selection and accounting for model uncertainty in graphical models using occam's window. *Journal of the American Statistical Association*, 89(428):1535–1546, 1994.

[18] David Madigan, Jeremy York, and Denis Allard. Bayesian graphical models for discrete data. *International Statistical Review/Revue Internationale de Statistique*, pages 215–232, 1995.

[19] Marco Moretto, Paolo Sonego, Nicolas Dierckxsens, Matteo Brilli, Luca Bianco, Daniela Ledezma-Tejeida, Socorro Gama-Castro, Marco Galardini, Chiara Romualdi, Kris Laukens, Julio Collado-Vides, Pieter Meysman, and Kristof Engelen. Colombos v3.0: leveraging gene expression compendia for cross-species analyses. *Nucleic Acids Research*, 44(Database issue):D620–D623, 01 2016.

[20] Kevin P Murphy. *Machine Learning: A Probabilistic Perspective*. The MIT Press, Cambridge, MA, 2012.

[21] Radhakrishnan Nagarajan, Marco Scutari, and Sophie Lbre. *Bayesian Networks in R: With Applications in Systems Biology.* Springer Publishing Company, Incorporated, 2013.

[22] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988.

[23] E. Prestat, S. R. de Morais, J. A. Vendrell, A. Thollet, C. Gautier, P. A. Cohen, and A. Aussem. Learning the local Bayesian network structure around the ZNF217 oncogene in breast tumours. *Comput. Biol. Med.*, 43(4):334–341, May 2013.

[24] Edi Prifti, Jean-Daniel Zucker, Karine ClÃ©ment, and Corneliu Henegar. Interactional and functional centrality in transcriptional co-expression networks. *Bioinformatics*, 26(24):3083–3089, 2010.

[25] H. Raiffa and R. Schlaifer. *Applied statistical decision theory.* Studies in managerial economics. Division of Research, Graduate School of Business Adminitration, Harvard University, 1961.

[26] Francisco G. Vital-Lopez, Vesna M., and Bhaskar Dutta. Tutorial on biological networks. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(4):298–325, 2012.

[27] Zhigang Xue, Kevin Huang, Chaochao Cai, Lingbo Cai, Chun-yan Jiang, Yun Feng, Zhenshan Liu, Qiao Zeng, Liming Cheng, Yi E. Sun, Jia-yin Liu, Steve Horvath, and Guoping Fan. Genetic programs in human and mouse early embryos revealed by single-cell rna sequencing. *Nature*, 500:593 EP –, 07 2013.

[28] Bin Zhang and Steve Horvath. A general framework for weighted gene coexpression network analysis. In *STATISTICAL APPLICATIONS IN GENETICS AND MOLECULAR BIOLOGY 4: ARTICLE 17*, 2005.