On the Ensemble Prediction of Gene Regulatory Networks: a Comparative Study

Mariana R. Mendoza
PPGC, Instituto de Informática
Universidade Federal do Rio Grande do Sul
Porto Alegre, RS, Brazil
mrmendoza@inf.ufrgs.br

Ana L. C. Bazzan
PPGC, Instituto de Informática
Universidade Federal do Rio Grande do Sul
Porto Alegre, RS, Brazil
bazzan@inf.ufrgs.br

***Abstract*—A common trend in several areas of knowledge is to combine distinct methods or algorithms expertise in order to provide more accurate results. This approach has its basis on the theory of wisdom of crowds, which claims that the information drawn from collective decisions is usually more precise than the individual ones. Many classification tasks have profit from this idea. However, its application to gene regulatory networks inference is recent and still not deeply explored. In the current work, we perform a comparative study between several widely used combination methods in machine learning. We analyze their performance for artificially generated gene networks and observe that ensemble predictions yield more accurate results than individual ones, thus being an interesting strategy for improving inference on gene regulatory networks.**

***Keywords*-Gene Network; Inference; Ensemble Predictions; Wisdom of Crowds; Reverse Engineering**

I. Introduction

One of the most challenging and complex problems in bioinformatics refers to the analysis of gene expression data for the inference of regulatory interactions between genes. The main motivation is to characterize the genetic regulation responsible for cellular development and function performed by the organisms' underlying gene regulatory networks (GRNs). The construction of a global model for GRNs enables the investigation of the control and function of its distinct parts considering their effects over the system as a whole. This is particularly interesting, for instance, for drugs and treatments development. Additionally, this model represents an important tool for in silico experiments.

A wide range of statistical and machine learning methods have been already applied in the reverse engineering process, as recently reviewed in [1]. In general, these methods receive a data set of gene expression measurements as input and try to reveal the regulatory interactions that compose the networks' topology from implicit information embedded on experimental biological data, providing a network model consistent with the data. However, there is an important challenge involved in this context: as experiments usually yield sparse data sets, the interactions between hundreds of genes must be learned from a few available samples [2], which not only raises the difficulty of the inference problem, but also impairs its performance.

Indeed, in the comparative study by Hache and colleagues [1], it was demonstrated that the performance of the reverse engineering methods is still not good enough for practical applications with large networks. None of the tested methods outperforms all other methods or is capable of reconstructing the entire true network structure for all data sets. Furthermore, values for sensitivity, specificity, and precision were always low, suggesting that reverse engineering methods need further improvements to be truly reliable for biological applications.

An alternative approach for inference improvement was proposed in [3], which has its basis on the concepts of wisdom of crowds [4]. In contrast to the classical method of selecting a single network from the output of an algorithm, usually the one that best fits the data, Marbach and colleagues suggested integrating the information contained within a set of plausible solutions, relying on the fact that many reverse engineering algorithms naturally provide an ensemble of networks that are consistent with the biological data, e.g., genetic algorithms, simulated annealing, among others. By applying majority voting as the integration mechanism, authors found that ensemble predictions boost the performance compared to individual members of an ensemble provided by an evolutionary approach. A similar strategy was explored in [5], in which authors applied the idea behind the wisdom of crowds to compose the final solution of the inference process based on the set of candidate networks pointed out by the last generation of a genetic algorithm.

Moreover, the effects of building an ensemble from the output of a set of distinct reverse engineering algorithms were investigated by Marbach and colleagues in [6]. Given the set of predictions submitted by the participants of the DREAM3 challenge [7], which are given as ranked lists of predicted edges, authors have shown that inference methods have particular strengths and weaknesses, and that predictions drawn from the integration of multiple methods benefit from the variance in expertise. The so-called community predictions are formed following the Borda count voting method: the edge lists are reranked according to the average rank for each edge considering the top two methods, the top three methods, and so on, until all methods are included in the ensemble. The findings suggest that

1522-4899/12 $26.00 © 2012 IEEE
DOI 10.1109/SBRN.2012.27

IEEE computer society

community predictions are as accurate as the predictions made by best-performer methods and that the inclusion of methods with very low scores do not compromise the good performance of the ensemble.

The idea of combining the expertise of distinct algorithms is not new. In fact, recent studies on classification methods discuss the improvements introduced by the use of classifier ensembles. This approach exploits the idea that the use of different classifiers may offer complementary information about the patterns to be classified, so that the pattern recognition process becomes more effective [8]. In addition, since no classifier can be considered entirely satisfactory for a particular task, the combination strategy aims at increasing the capacity of generalization and, consequently, the performance of the system as a whole. However, even though the benefits of the ensembles for classification tasks are elucidated, ensemble solutions have been just recently applied in the context of inference of GRNs so that its potential was not fully explored yet. There is still the lack of a broad comparative study about inference of GRNs based on ensemble approaches, applying different types of combination methods to the same data set and evaluating its particular strengths, as well as research on the development of more robust combination methods. The present study aims at exploring the first niche and providing the reader with some insight on the benefits of using ensemble predictions.

In what follows, we introduce the concepts of wisdom of crowds and explain our materials and methods. Next, we describe our experiments and findings, which reinforce the idea of the improvement capacity associated to ensemble predictions in relation to currently available reverse engineering methods.

II. Materials and Methods

A. Wisdom of Crowds

The idea of building ensembles is motivated by the observation that, in general, individuals who decide collectively are more likely to make better decisions and predictions in regard to individuals or even experts [4]. This phenomena is known as wisdom of crowds and has been observed in many real situations, such as predictive markets. The Iowa Electronic Market[1] (IEM) is a speculative market based on users' opinion about an upcoming event and has been used to forecast the outcome of future events such as presidential elections. Previous events, such as the USA Presidential Race of 2008, have shown that IEM predictions have generally outperformed the major national polls and were more accurate than these, even months in advance of the actual election. Thus, predictive markets, as well as other situations in which information of a collective is aggregated, have shown better capability.

[1]http://tippie.uiowa.edu/iem/

B. Data

Due to the restricted knowledge about the biological systems from which gene expression profiles are obtained, the use of artificial networks and simulated expression signals is still a common practice to assess algorithms performance. Thus, in the present paper, we resort to an Artificial Gene Network (AGN) validation and simulation model[2] [9], based on the Boolean network approach, to compose an artificial set of networks. We generate 30- and 50-nodes AGNs, five instances of each, following the Barabási-Albert (BA) [10] model, which is currently known to be one of the most similar models to real gene regulatory networks [11].

Following the upper limit of stability for Boolean networks discussed in [12], we set the upper bound of nodes' average connectivity to $\langle k \rangle = 3$. Also, we consider a probabilistic approach for the Boolean modeling, which relax the deterministic rigidity of deterministic Boolean networks by allowing each gene to have more than one Boolean function, each of which have a particular usage probability [13]. For each network, we simulated 10 temporal expression signals of length 10, each one starting from a randomly chosen initial state. The dynamics of the AGN is obtained by applying the Boolean transition functions to the network's initial state and yields binary gene expression signals. Furthermore, we have concatenated these signals generating a single time series of size 100, which is used for network inference.

C. Reverse Engineering Algorithms

As the goal of this study is to focus on the combination of the results provided by multiple reverse engineering methods, and not on the methods themselves, we did not implement any inference algorithm. Instead, we resort to R packages designed for GRNs reconstruction based on two widely used modeling frameworks, Boolean networks and Bayesian networks, implemented by the **boolnet** [14] and **bnlearn** [15] packages respectively. We extracted two types of information from the gene expression profile: an adjacency matrix and a confidence matrix. While the first provides the network wiring, the second give us an additional information about the confidence attached to each link by the inference method. In addition, we also create a ranked list of interactions based on the confidence matrix for the use of non-linear voting-based combination methods.

The algorithms used for network inference, as well as their respective parameters, are summarized on Table I. A total of 13 possible combinations of algorithms and parameters were tested: three based on the Boolean formalism and ten based on the Bayesian formalism. Regarding the former, we applied the best-fit algorithm [13] and obtained, for each gene, a set of Boolean functions that explain its behavior within the gene expression profile, each of which associated to a probability. In contrast to algorithms that only find

[2]http://code.google.com/p/jagn/

Table I
REVERSE ENGINEERING ALGORITHMS AND PARAMETERS.

Network Formalism	Inference Algorithm	Algorithm's Parameter
Boolean	Best-fit	maxK = 2 maxK = 3 maxK = 4
Bayesian	Tabu	score = "aic" score = "bde" score = "loglik"
	HC	score = "aic" score = "bde" score = "loglik"
	MMHC	score = "aic" score = "bde" score = "loglik"
	ARACNE	—

functions that perfectly explain all measurements, the best-fit is a more suitable option since real gene expression data are known to be noisy. The use of the best-fit algorithm implemented by the **boolnet** R package involves a single parameter, `maxK`, which determines the maximum number of input genes, or fan-in, to be tested per gene. Following previous studies [2], we limit the fan-in to values `maxK` $= \{2, 3, 4\}$, incorporating the notion that a gene's expression is controlled by a small set of active regulators.

From the output of the best-fit algorithm we derive the confidence matrix for the inference process, which is computed by summing the probabilities attached to all Boolean functions that comprise a particular predictor. For instance, assume that four Boolean functions were inferred for `gene1`, each of which attached to a probability of 0.25. Suppose also that three of the Boolean functions for `gene1` include `gene4` as its predictor. Thus, the level of confidence on link `gene4` $\rightarrow$ `gene1` will be equal to 0.75. Once the confidence matrix is available, we obtain the adjacency matrix by assigning 1 to all links with confidence greater than zero.

Regarding the Bayesian formalism, we have tested three distinct classes of algorithms available in the **bnlearn** R package: two greedy score-based structure learning algorithms, Hill Climbing (HC) and Tabu Search (Tabu); a constraint-based local discovery algorithm, ARACNE [16]; and a hybrid structure learning algorithm, Max-Min Hill Climbing (MMHC) [17]. The score-based algorithms are general purpose heuristic optimization algorithms that rank network structures with respect to a goodness of fit score and try to maximize it. The search for the best structures is performed by the algorithms HC and Tabu, which are both greedy algorithms with the difference that the latter allows an eventual decrease in the score function in order to escape local optima. Due to the discrete characteristic of data, we applied the multinomial log-likelihood, the Akaike Information Criterion (AIC), and the equivalent Dirichlet posterior density (BDe) scores for structure evaluation.

Constraint-based algorithms, such as ARACNE, use conditional independence tests to detect the Markov blankets[3] of the variables and compute the structure of the Bayesian network. ARACNE identifies candidate interactions by estimating pairwise gene expression profile mutual information and, afterwards, refines the model by removing the vast majority of indirect candidate interactions using a well-known information theoretic property, the Data Processing Inequality (DPI). Finally, hybrid algorithms such as the Max-Min Hill-Climbing (MMHC) [17] combine aspects of score- and constraint-based algorithms: while the latter aims at restricting the search space by detecting the Markov blanket through conditional independence tests, i.e., a local search, the score-based algorithms are used to explore the restricted search space.

The default data structure returned by the algorithms available in the **bnlearn** R package is the adjacency matrix, obtained through the function `amat()`. Therefore, to estimate the degree of confidence attached to each interaction, or network link, we follow the approach proposed in [18], which is implemented under the package function `arc.strength()`. There, a simple way of measuring Bayesian networks' arc strength based on bootstrap is introduced: multiple network structures are generated by applying non-parametric bootstrap to the data and the confidence level of a particular feature is estimated by examining in how many of the bootstrap samples it appeared. Once the confidence matrix is obtained, a ranked list of interactions according to their respective confidence levels is constructed.

D. Combination Methods

Several combination methods have been already discussed in the literature, especially regarding supervised learning. In general, these methods are classified as selection-based and fusion-based methods. While selection-based methods nominate a single method to provide the output for the process according to an analysis of methods' competence, fusion-based methods integrate the information contained within a set of predictions to compose an single output. In the current paper we will focus our analysis on the latter, more specifically in five distinct methods, classified as follows [8]:

- **Linear methods**: sum and average of confidence levels
- **Non-linear methods**: Borda count and majority voting
- **Statistical methods**: Dempster-Shafer (D-S)

The linear and statistical methods are applied to the confidence matrix. The linear methods simply compute the sum and mean of the confidence levels inferred by each of the reconstruction methods, obtaining the degree of confidence of the ensemble prediction. The Dempster-Shafer method is a generalization of the Bayesian theory for

[3]The Markov blanket for a given node is the set composed by its parents, its children and its children's other parents.

propositions combination and allows one to merge evidence from different sources to obtain a single degree of belief that takes into account all the available evidence. The general combination rule of D-S theory is given by:

$$m^{1,2}(C) = \frac{\sum_{A \cap B = C} m^1(A) m^2(B)}{\sum_{A \cap B \neq \varnothing} m^1(A) m^2(B)} \quad (1)$$

in which $m^X \in [0, 1]$ is the belief mass of a given element according to method X, $m^{1,2}$ is the resulting ensemble mass, and the denominator term is a normalization factor.

Considering the two possible states for the system under consideration, i.e., the status of an interaction between two genes, to be absent (A) or present (P), such that $\theta = \{A, P\}$, the universal set of our problem containing all and only the states in which the proposition is true is defined by $2^\theta = \{\{A, P\}, A, P, \varnothing\}$. Since state $\{A, P\}$ is not allowed in our problem (an edge can be either present or absent, but not both), it is removed from this set. Recalling the two main properties of the belief mass functions, I) $m(\varnothing) = 0$ and II) $\sum_{S \in 2^\theta} m(S) = 1$, we have derived the following combination rules for the ensemble prediction of GRNs by D-S technique based on two distinct methods, m^1 and m^2:

$$m^{1,2}(A) \propto m^1(A)m^2(P) + m^1(P)m^2(P) + m^1(A)m^2(A) \quad (2)$$

$$m^{1,2}(P) \propto m^1(P)m^2(A) + m^1(A)m^2(P) + m^1(P)m^2(P) \quad (3)$$

Regarding the non-linear methods, the majority voting requires every method on the ensemble to vote on the classification of a given interaction as absent or present, according to their respective adjacency matrices. The inclusion of the interaction in the final GRN model will be decided by the majority of the votes. Moreover, the model's confidence is computed as the average confidence level of the majority group. In contrast, the Borda count considers the confidence level attached to each interaction by ranking the predictions made by each of the individual methods in respect to the degree of confidence and combining the ranked lists by averaging the rank of each interaction [6]. The confidence level of each interaction is then recomputed as its position in the ranked list.

III. Experiments and Results

In order to perform a comparative study on the combination methods for ensemble predictions discussed in the previous section, we have run the algorithms outlined in Table I, varying their respective parameters, for each of the AGNs comprising our data set, i.e., five networks of size 30 and five networks of size 50. Each algorithm yields an individual solution that is used to compose the ensemble prediction. We follow the approach discussed in [6] to build the ensembles: starting from the top scored method, we systematically aggregate the other methods according to a decreasing order of score, forming an ensemble of the top two methods, the top three methods, and so on, until an ensemble comprising all the methods is built. This approach aims at providing a better picture about the effect of each inference method on the ensembles' performance.

After the algorithms have been applied for all networks and the ensembles have been built, we average their respective results, obtaining a single score for each reverse engineering algorithm and combination method, for both network sizes. The performance of the distinct combination methods were twofold assessed and compared. First, we have evaluated results in terms of the area under the ROC curve (AUC score). Since the reverse engineering methods return a matrix with the degree of confidence of all inferred connections, we have computed the specificity and sensitivity for each possible threshold of this matrix, starting from the lowest value, and subsequently we obtain the AUC score.

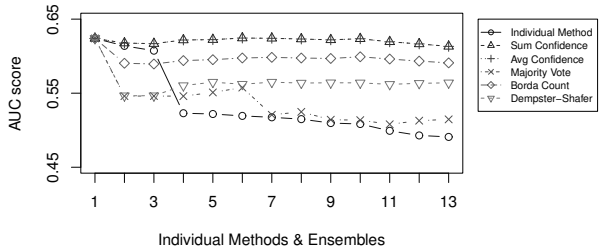
Moreover, we compute the similarity between an inferred network (N_I) and the target network (N_T). Similarity is a measure that combines both true positive rate (TPR, sensitivity) and true negative rate (TNR, specificity), as defined in Equation 4. While the AUC score evaluates the performance across the range of possible thresholds, the similarity considers a single and specific threshold of occurrence, based on which an adjacency matrix is extracted from the confidence matrix for comparison purposes. Due to space constraints, only the results for a threshold equal to 0.1 are shown in this paper.

$$Similarity(N_I, N_T) = \sqrt{TPR \times TNR} \quad (4)$$

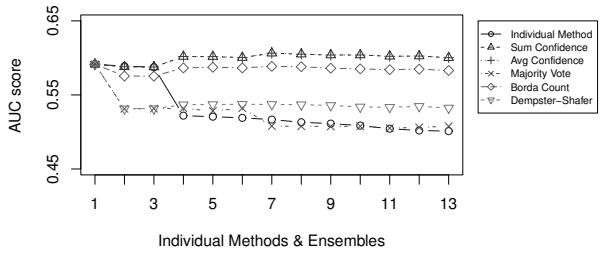
In general, Boolean-based methods have performed better than the Bayesian approach. However, since our goal is to evaluate combination methods and not reverse engineering methods, we do not identify them in the graphs. Instead, methods and ensembles receive a numeric identification (x axis), in which 1 refers to the best performing method, 2 to the second best method and the ensemble of the top two methods, and so on until 13, which refers to both the worst performing method and the ensemble composed by the whole set of methods.

Results in terms of the AUC score are depicted in Figures 1(a) and 1(b) for the 30-node and 50-node networks. These values corroborate the advantages of ensemble predictions over individual predictions. For both network sizes, all combination methods have yielded more accurate results, with the most meaningful improvement being observed for the linear methods, namely the sum and average, followed by Borda count. Little to none performance difference was observed between the linear methods. Regarding D-S method, the improvement was moderate.

In contrast, the combination by means of majority voting produced the poorest results. For the 30-node networks, it modestly outperformed most of the ensembles. Also, its

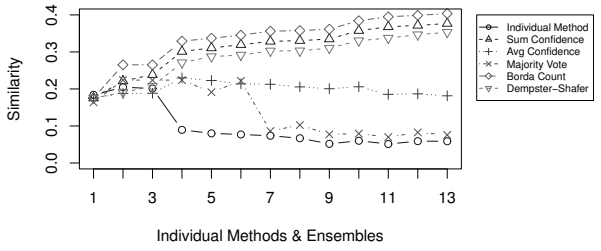


(a) 30-node networks

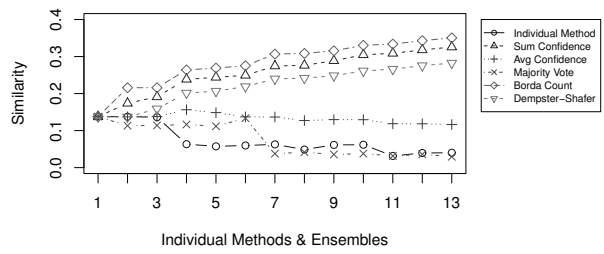


(b) 50-node networks

Figure 1. Average AUC scores of inferred networks.



(a) 30-node networks



(b) 50-node networks

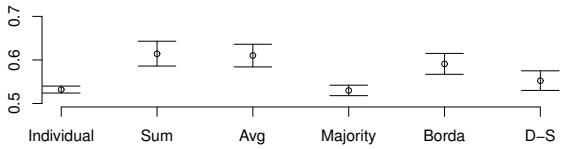
Figure 2. Average similarity between inferred and target networks.

performance for the 50-node networks was mostly worse and only in few cases higher than individual scores. This is probably due to the fact that in majority voting, the information carried by a set of methods that predict a given edge is completely discarded when this set is the minority. In this case, the edge's confidence based on the ensemble prediction is equal to zero, regardless the relevance and confidence of the methods that predicted it.

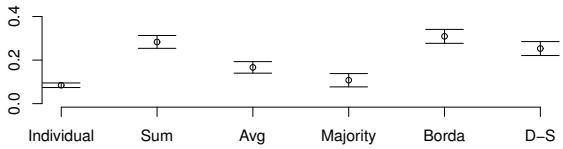
Furthermore, one can notice that the performance improvement given by the combination methods in relation to individual methods becomes more significant as the size of the ensemble increases. However, for the 30-node networks, linear methods have provided better results even for small ensembles, composed solely by the top three methods, while the other methods have improved results only for ensembles composed by four or more solutions. For the 50-node networks, all combination methods have shown some improvement only from the forth ensemble.

The benefits of ensemble predictions are even clearer for the analysis based on the similarity measure (Figures 2(a) and 2(b)). When comparing the adjacency matrices obtained under a threshold of occurrence equal to 0.1, we observe a substantial gain in accuracy. The best ensemble predictions, provided by the sum of interactions confidence, D-S and Borda count, outperform all individual predictions. Moreover, the performance of D-S method has improved, while the performance of the average confidence method has dropped to an intermediate level. Once again, the majority voting was the least robust combination method to low-performing algorithms: its performance deteriorates from the sixth ensemble for both 30- and 50-node networks.

Figures 3(a) and 3(b) depict the 95% confidence intervals for the tested methods, confirming the superiority of the ensemble approach in relation to individual algorithms. A statistically significant improvement in terms of AUC score and the similarity measure has been observed for the sum and average of interactions confidence, as well as for the Borda count method. Considering solely the similarity measure, the Dempster-Shafer method is also significantly better. In contrast, no statistically significant difference is observed between majority voting and individual solutions. Based on this comparison, one can conclude that the overall performance of ensemble approaches was satisfactory and it seems this is a promising strategy for inference of GRNs.



(a) AUC score



(b) Similarity measure

Figure 3. 95% confidence intervals.

IV. Conclusion

In the present work we have investigated the effects of several combination methods of reverse engineering algorithms on the ensemble prediction of GRNs. We have tested two linear methods – the sum and average of confidence levels, two non-linear methods – the majority voting and Borda count, and Dempster-Shafer theory with AGNs. The results corroborate previous studies [3], [6]: predictions drawn from a community of methods are indeed more accurate than individual ones. However, differently from these, we perform a comparative study between distinct combination methods for ensemble predictions, assessing their performance for a common data set.

Our results suggest that Borda count and the sum or average of confidence levels are the most reliable combination methods for ensemble prediction of GRNs between the tested methods. Most of the ensembles composed by their predictions have outperformed all other combination methods, as well as individual algorithms. In contrast, the majority voting presented the poorest performance and was not very robust to low-performing methods. The findings of this comparative study meet the ideas discussed in [4]: under the right circumstances, such as independent decision making, groups are remarkably intelligent and often perform better than the smartest people in them. However, only known combination methods were investigated in this study such that we hypothesized that more robust combination methods, as well as ensembles built over more accurate algorithms, could provide even more remarkable gains.

Acknowledgment

We thank the Brazilian National Research Council (CNPq) for its support to the authors.

References

[1] H. Hache, H. Lehrach, and R. Herwig, "Reverse Engineering of Gene Regulatory Networks: A Comparative Study," *EURASIP Journal on Bioinformatics and Systems Biology*, vol. 2009, pp. 8:1–8:12, 2009.

[2] D. Husmeier, "Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks," *Bioinformatics*, vol. 19, pp. 2271–2282, 2003.

[3] D. Marbach, C. Mattiussi, and D. Floreano, "Combining Multiple Results of a Reverse Engineering Algorithm: Application to the DREAM Five Gene Network Challenge," *Ann N Y Acad Sci*, vol. 1158, pp. 102–113, 2009.

[4] J. Surowiecki, *The Wisdom of Crowds*. New York, NY: Random House, 2005.

[5] M. R. Mendoza and A. L. C. Bazzan, "Evolving Random Boolean Networks with Genetic Algorithms for Regulatory Networks Reconstruction," in *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*, ser. GECCO '11. New York, NY, USA: ACM, July 2011, pp. 291–298. [Online].

[6] D. Marbach, R. J. Prill, T. Schaffter, C. Mattiussi, D. Floreano, and G. Stolovitzky, "Revealing strengths and weaknesses of methods for gene network inference," *PNAS*, vol. 107, no. 14, pp. 6286–6291, 2010.

[7] R. Prill, D. Marbach, J. Saez-Rodriguez, P. Sorger, L. Alexopoulos, X. Xue, N. Neil Clarke, G. Altan-Bonnet, and G. Stolovitzky, "Towards a rigorous assessment of systems biology models: The DREAM3 challenges," *PLoS ONE*, vol. 5, no. 2, p. e9202, 02 2010.

[8] A. M. Canuto, M. C. Abreu, L. de Melo Oliveira, J. C. X. Jr., and A. de M. Santos, "Investigating the influence of the choice of the ensemble members in accuracy and diversity of selection-based and fusion-based methods for ensembles," *Pattern Recognition Letters*, vol. 28, pp. 472–486, 2007.

[9] F. M. Lopes, R. M. Cesar-Jr, and L. da F. Costa, " AGN Simulation and Validation model," in *Proceedings of Advances in Bioinformatics and Computational Biology*, vol. 5167 of Lecture Notes in Bioinformatics. Springer-Verlag Berlin, 2008, pp. 169–173.

[10] A. L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[11] R. Albert, "Scale-free networks in cell biology," *J Cell Sci*, vol. 118, no. 21, pp. 4947–4957, 2005.

[12] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets." *J Theor Biol*, vol. 22, no. 3, pp. 437–467, March 1969.

[13] I. Shmulevich, E. R. Dougherty, S. Kim, and W. Zhang, "Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks," *Bioinformatics*, vol. 18, no. 2, pp. 261–274, 2002.

[14] C. Müssel, M. Hopfensitz, and H. A. Kestler, "BoolNet – an R package for generation, reconstruction and analysis of Boolean networks," *Bioinformatics*, vol. 26, no. 10, pp. 1378–1380, 2010.

[15] M. Scutari, "Learning Bayesian networks with the bnlearn R package," *Journal of Statistical Software*, vol. 35, no. 3, pp. 1–22, 2010.

[16] A. Margolin, K. N. Basso, C. Wiggins, G. Stolovitzky, R. Favera, and A. Califano, "ARACNE: An algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context," *BMC Bioinformatics*, vol. 7, no. Suppl 1, p. S7, 2006.

[17] I. Tsamardinos, L. E. Brown, and C. F. Aliferis, "The Max-Min Hill-Climbing Bayesian network structure learning algorithm," *Machine Learning*, vol. 65, no. 1, pp. 31–78, 2006.

[18] N. Friedman, M. Goldszmidt, and A. Wyner, "Data Analysis with Bayesian Networks: A Bootstrap Approach," in *UAI '99: Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, 1999, pp. 196–20.