A Framework for Scalable Inference of Temporal Gene Regulatory Networks based on Clustering and Multivariate Analysis*

Ricardo de Souza Jacomini[1], David Correa Martins-Jr[2], Felipe Leno da Silva[1] and Anna Helena Reali Costa[1]
[1]Escola Politécnica da Universidade de São Paulo – São Paulo, Brazil
[2]Universidade Federal do ABC – Santo André, Brazil
{ricardo.jacomini,f.leno,anna.reali}@usp.br, david.martins@ufabc.edu.br

Abstract

Genomic and transcriptomic information have been used as a starting point for the analysis of the origin and development of diseases, which lead to the development of many methods that model the dynamics of gene expression data. Gene Networks (GN) are widely used to model such information, and many methods have been developed for GN inference from temporal gene expression data. However, this data usually results in training sets composed of a small number of temporal samples for a large amount of genes, which renders many GN inference methods unfeasible to apply in real temporal expression data composed of thousands of genes, since they are exponential in function of the number of genes. In order to improve the scalability of the GN inference problem, we propose a novel framework based on the Probabilistic Gene Networks model, in which we rely on a clustering preprocessing step to provide an approximated solution with reduced computational complexity. We compared our proposal with a similar approach without the clustering step, and our experiments show that the proposed framework achieves substantial computation time reduction, while approximately preserving the prediction accuracy.

1 Introduction

The modeling, inference and interpretation of gene regulatory networks (GRNs) from temporal gene expression data has drawn significant attention recently [Hecker *et al.*, 2009; Marbach *et al.*, 2012; Shmulevich and Dougherty, 2014], specially after the advent of large scale gene expression measurement techniques, such as cDNA microarrays [Shalon *et al.*, 1996] SAGE [Velculescu *et al.*, 1995] and, more recently, RNA-Seq [Wang *et al.*, 2009]. This interest relies on the fact that genes play a major role in the control of cell functions. The GRN inference problem involves the discovery of complex regulatory relationships between biological molecules which can describe not only diverse biological functions, but also the dynamics of molecular activities. Once the network is recovered, intervention studies can be conducted to control the dynamics of biological systems aiming to prevent or treat diseases [Shmulevich and Dougherty, 2014]. Genes and proteins usually form an intrincate complex network where, in many cases, the behavior of a given gene, measured by means of its expression level (i.e. mRNA abundance), depends on a multivariate and coordinated action of other genes and their byproducts (proteins) [Martins-Jr *et al.*, 2008]. The importance of GRN reconstruction can also be seen through many initiatives, such as DREAM (Dialogue for Reverse Engineering Assessments and Methods) [Marbach *et al.*, 2012].

There are two main approaches to model the gene interactions [Shmulevich and Dougherty, 2014]: continuous and discrete. The continuous approaches consider mainly differential equations to obtain a quantitative detailed model of biochemical networks [De-Jong, 2002; Hecker *et al.*, 2009] while discrete models measure the gene interactions from the qualitative point of view. The main discrete models include the ones based on graphs such as Bayesian Networks [Friedman *et al.*, 2000], Boolean Networks (BN) [Kauffman, 1969], and its stochastic versions Probabilistic Boolean Networks (PBN) [Shmulevich *et al.*, 2002] and Probabilistic Gene Networks (PGN), a simplified model of PBN [Barrera *et al.*, 2007]. Continuous models provide a detailed understanding of the system, but prior information about kinetic parameters and more experimental samples are demanded [Hecker *et al.*, 2009]. On the other hand, discrete models are more useful to capture the global behavior of the system dynamics, requiring less data and being easier to implement and analyse [Hecker *et al.*, 2009].

The literature that deals with the GRN inference problem is vast. Some examples of methods that deal with this problem include mutual information based feature selection [Liang *et al.*, 1998; Lopes *et al.*, 2014], relevance networks [Margolin *et al.*, 2006; Faith *et al.*, 2007], feature selection by maximum relevance/minimum redundancy [Meyer *et al.*, 2007], signal perturbation [Ideker *et al.*, 2000; Carastan-Santos *et al.*, 2016], among others. Even though there are many GRN inference methods in the literature [Markowetz and Spang, 2007; Hecker *et al.*, 2009; De-Smet and Marchal, 2010; Marbach *et al.*, 2012], GRN inference is considered an ill-posed problem, since for a given dataset of gene expression profiles,

*We gratefully acknowledge funding from: CAPES, CNPq (grants 304955/2014-0, 311608/2014-0), São Paulo Research Foundation (FAPESP) (grants 2011/50761-2, 2015/01587-0, 2015/16310-4)

there are many (if not infinity) networks capable of generating this same dataset. This problem is further hampered due to a typically limited number of samples, a huge dimensionality (number of variables, i.e., genes), and the presence of noise [Hecker *et al.*, 2009; Shmulevich and Dougherty, 2014].

In the specific context of discrete models, BNs and PBNs can generalize and capture the global behavior of biological systems [Kauffman, 1969; Shmulevich *et al.*, 2002]. The main disadvantage of these models is the information loss as a consequence of the required data quantization. However, the quantization makes the BN and PBN models simpler to implement and analyse [Styczynski and Stephanopoulos, 2005; Ivanov and Dougherty, 2006], and many methods were proposed to infer GRNs modeled as BN or PBN [Akutsu *et al.*, 1999; Lahdesmaki and Shmulevich, 2003; Liang *et al.*, 1998; Nam *et al.*, 2006].

Although PBN genes have only two possible expression values, network inferences are still difficult, since the curse of dimensionality still plays an important role. In this way, PGN provides a simplification of the inference process that allows to apply local feature selection to search for the best subsets of genes to predict the behavior of a given target gene [Barrera *et al.*, 2007]. Since exhaustive search is the only feature selection algorithm that guarantees optimality [Cover and van Campenhout, 1977], high performance computing techniques are required when using this algorithm to search for predictor subsets of three or four dimensions (for higher dimensions, this technique is still impractical) [Borelli *et al.*, 2013; Carastan-Santos *et al.*, 2016]. An alternative to reduce the computational complexity of the exhaustive search is to apply some prior dimensionality reduction technique to restrict the search space of candidate predictor subsets for a given target, which is not trivial to do, since even the worst features individually could be great when combined to predict a given target, while the best individual features could not be so good in predicting the target when combined [Pudil *et al.*, 1994; Martins-Jr *et al.*, 2008].

In order to alleviate the curse of dimensionality inherent to the GRN inference problem, and consequently its computational cost, this paper proposes a novel GRN inference framework to infer PGNs. Our proposal relies on a clustering technique to reduce the search complexity when evaluating the possible predictor subsets and thus alleviate the computational complexity of the GRN inference. Besides, an intrinsically multivariate analysis is conducted to eliminate redundant features from each predictor subset [Martins-Jr *et al.*, 2008] and, consequently, to obtain a minimal network. We experimentally compared the prediction quality of our proposal with GRN inference by executing an exhaustive search over all possible predictor subsets, which has prohibitive computational costs but is expected to achieve the best prediction quality. Our results using *in silico* data show that the approximated solution given by our framework achieves very similar prediction quality to the exhaustive search, while providing a substantial reduction of the computational complexity.

2 Probabilistic Gene Networks inference

The Probabilistic Gene Networks (PGN) model [Barrera *et al.*, 2007] assumes that the temporal gene expression samples follow a first order Markov Chain where each target gene in a given timepoint depends only on its predictor subset values in the previous time instant. The transition function is homogeneous (it does not change over time), almost deterministic (from any state, the system has a preferential state to go) and conditionally independent (i.e., the expression value of a given gene is dependent only on its predictors, following the Markov hypothesis). These assumptions are important simplifications to deal with the limited number of samples typically available in real gene expression data.

Conceptually, PGN is a restricted type of PBN [Shmulevich *et al.*, 2002]. While PBNs assume that variables are binary, PGNs assume that gene expression values can be described in two or more discrete values. For example, Barrera *et al.* (2007) considered three possible states for each gene: -1 (underexpressed), 0 (normally expressed), +1 (overexpressed). However, it is worth to note that the number of statistical parameters (configuration values of a given predictor subset) are doubled when including just one binary predictor in the subset (i.e., it grows exponentially). PGN assumes that a target gene presents several different predictor functions like PBN. However, all these functions necessarily present the same set of predictor genes as inputs in PGN, whereas a PBN target gene might be described by several transition functions that might take as input different sets of predictors. Another important restriction is the quasi-determinism assumed by PGN, which implies that it is often possible to find very good predictors for every target in terms of prediction error.

PGN-based GRN inference methods rely on three fundamental steps [Barrera *et al.*, 2007; Lopes *et al.*, 2014]: (i) data quantization; (ii) feature selection; (iii) determination of the logic function that minimizes the classification error of each target expression profile.

Feature Selection is an extremely important step in this inference procedure. A feature selection problem consists in selecting a subset of features that well represents the objects under study. In our case, a feature selection algorithm consists basically in searching the subsets of genes that best predicts a given target gene according to a criterion function, which assign a quality value for a subset according to the expression profile of the target gene at the next time instant [Barrera *et al.*, 2007; Borelli *et al.*, 2013; Lopes *et al.*, 2014].

There are many feature selection algorithms proposed in the literature, some of them are computationally efficient but suboptimal. In fact, in the general case the only algorithm that guarantees optimality is the exhaustive search [Cover and van Campenhout, 1977]. This is due to the well known nesting effect in which a feature included into the solution subset might never be removed by a suboptimal algorithm feature selection, even if that feature is not in the optimal solution set. Similarly, a previously removed feature might never be inserted again into the current subset solution, even if it belongs to the optimal solution set [Pudil *et al.*, 1994].

The GRN inference framework here proposed (see Section 4) applies an exhaustive search for subsets of a given fixed dimension k, adopting two criterion functions popularly used in feature selection-based GRN inference methods [Martins-Jr *et al.*, 2008; Lopes *et al.*, 2014]: (i) Coefficient of Determination (CoD), which is based on classification Bayesian error [Dougherty *et al.*, 2000]; and (ii) Mutual Information (MI), which is based on Shannon's entropy [Shannon, 2001].

The Coefficient of Determination (CoD) [Dougherty *et al.*, 2000] for a target gene Y given a set of candidate predictor genes $\mathbf{Z}$ is a non-linear criterion function given by:

$$CoD_Y(\mathbf{Z}) = \frac{\varepsilon_Y - \varepsilon_Y(\mathbf{Z})}{\varepsilon_Y} \tag{1}$$

where $\varepsilon_Y = 1 - \max_{y \in Y} P(y)$ and $\varepsilon_Y(\mathbf{Z}) = 1 - \sum_{\mathbf{z} \in \mathbf{Z}} \max_{y \in Y} P(\mathbf{z}, y)$. Greater CoD values lead to better feature subspaces ($CoD = 0$ means that the feature subspace does not identify the prior error, while $CoD = 1$ means that the error is totally eliminated).

In its turn, the mutual information (MI) is defined as:

$$I(\mathbf{Z}, Y) = H(Y) - H(Y|\mathbf{Z}) \tag{2}$$

where $H(.)$ is the Shannon entropy, with $H(Y) = \sum_{y \in Y} P(y) log P(y)$ and $H(Y|\mathbf{Z}) = \sum_{y \in Y, \mathbf{z} \in \mathbf{Z}} P(\mathbf{z}) P(y|\mathbf{z}) log P(y|\mathbf{z})$, $P(y)$ is the probability of $Y = y$ and $P(y|\mathbf{z})$ is the conditional probability of $Y = y$ given that $\mathbf{Z} = \mathbf{z}$.

However, it is not enough to select the best subset of predictors, since redundant genes might be present in these subsets. So it is important to perform a multivariate analysis of these predictors with the aim of reducing the number of predictors per target, thus simplifying the network.

The multivariate nature of the relationship of certain predictors with regard to the target leads to the already mentioned *nesting effect*, and can be estimated by the intrinsically multivariate prediction (IMP) phenomenon [Martins-Jr *et al.*, 2008]. A set of genes $\mathbf{Z}$ is considered IMP given a target gene Y if the target behavior (expression profile) is strongly predicted by the combined expression profiles of $\mathbf{Z}$ and, at the same time, weakly predicted by any proper subset of $\mathbf{Z}$. In this sense, the IMP score (IS) can be defined by [Martins-Jr *et al.*, 2008]:

$$IS(\mathbf{Z}, Y) = \mathcal{J}(\mathbf{Z}, Y) - \max_{\mathbf{Z}' \subset \mathbf{Z}} \mathcal{J}(\mathbf{Z}', Y), \tag{3}$$

where $\mathcal{J}(.)$ is the chosen criterion function which evaluates the dependence of a variable target Y with regard to a candidate feature set $\mathbf{Z}$ (higher values imply higher dependence). $IS(\mathbf{Z}, Y) = 0$ indicates that there is at least one redundant variable in $\mathbf{Z}$, implying that $\mathbf{Z}$ should be reduced ($\mathbf{Z}$ is definitely not IMP with regard to the target). It is also possible to define a positive threshold to decide whether a feature set is IMP or not with regard to the target. In case the pair $(\mathbf{Z}, Y)$ is not IMP, $\mathbf{Z}$ can be reduced to one of its proper subsets that presents maximum $\mathcal{J}(.)$ value. This process is recursive: the reduction is applied until the IMP score of the current pair $(\mathbf{Z}, Y)$ be positive or larger than a certain threshold.

Our proposed gene networks inference framework performs an intrinsically multivariate prediction analysis in the subsets returned by the exhaustive search algorithm to reduce the search dimensionality by discarding irrelevant features. This procedure helps to simplify the final networks.

3 Clustering

The use of clustering algorithms on gene expression data analysis can elucidate some challenging biological and genomic issues, such as identifying the functionality of genes, finding out which genes are co-regulated (which could give clues about functional annotation of genes), revealing important genes that distinguish between abnormal and normal tissues, etc [Zhao and Karypis, 2003]. Furthermore, since some genes might be strongly correlated (have almost identical profiles), this may suggest that they could be assigned to the same group in such a way that this group could be represented by one of these genes. In this way, highly correlated genes belonging to the same cluster are never considered in the same candidate predictor subset, since highly correlated genes are considered redundant. Therefore, in practice, clustering could be a useful tool to discard many candidate subsets with redundant genes which, in turn, implies in a significant reduction of both initial dimensionality and search space, which helps to alleviate scalability issues.

In this paper, we adopted the k-means clustering as an initial step of the GRN inference proposed method (see Section 4) for two reasons. First, k-means clustering gives partitions as result, i.e., each resulting cluster contains a list of genes and the intersection between different clusters lists is always null (a gene cannot belong to more than one cluster). Second, k-means clustering allows to regulate the number of desired clusters (parameter k indicates the number of clusters). This is important, since k becomes the resulting dimensionality of the GRN inference process. For instance, if k is set to a number in the order of dozens, the dimensionality of the process reduces from N initial genes in the order of thousands to k gene clusters in the order of dozens.

4 Proposed GRN inference framework

Here we propose a new framework for GRN inference that follows the PGN model assumptions (see Section 2), and applies a clustering technique before feature selection to reduce the dimensionality of possible predictor subsets. Besides, after the feature selection phase, minimal predictor subsets are found by removing redundant genes inside the predictor subset through a multivariate analysis to make networks as simple as possible. Figure 1 depicts the main steps involved, which are described as follows.

(a) Gene expression standardization: Given a gene expression data, first a transformation is applied to the input data. In the experiments of Section 5, a Z-score standardization is applied, in which the expression e_i of a given gene i becomes $e'_i = \frac{e_i - \mu_i}{\sigma_i}$, where μ_i and σ_i are average and standard deviation of the expressions of gene i, respectively. This transformation aims to change the data in such a way that expression values of a given

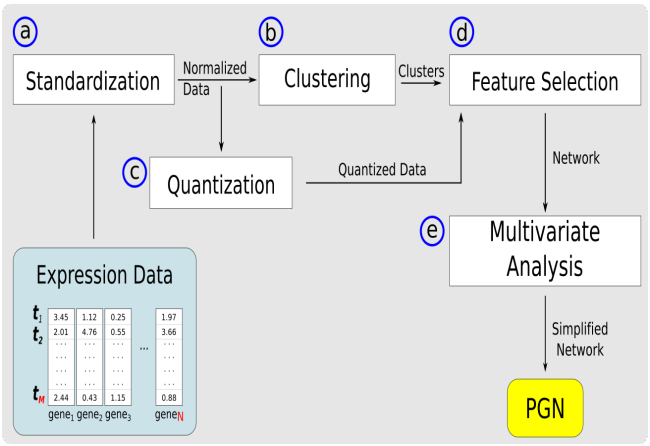

Figure 1: Block diagram with the main steps performed in this proposed framework to infer gene networks following the PGN model. The light blue box represents the gene expression profiles dataset to be taken as input by the framework, and the yellow box represents the output PGN.

gene below its own average become negative (underexpressed), while expressions above its own average become positive (overexpressed).

(b) Clustering: A clustering method is applied to group genes with similar expression profile measures. Any clustering method that returns a partition and a list of members per cluster, including their respective representative genes (genes that most represent their respective clusters according to a given criterion) can be used. It is desirable that the clustering method allows to set the number of clusters to be returned, or at least to restrict the maximum number of clusters, since this number has a crucial impact on the feature search space. Here we adopted the k-means clustering for the experiments described in Section 5.

(c) Quantization: Following the PGN model, the gene expression dataset is quantized so that each gene expression presents a finite set of possible values. We adopt the binary quantization where negative Z-scored values become 0, while positive Z-scored values become 1.

(d) Feature selection: A feature selection algorithm is applied considering each gene placed as target, aiming to achieve the best predictor subset for that target, according to a given criterion function. All gene representatives (one gene for each cluster) are taken as potential predictor genes, hence all other genes are ignored. If the defined number of clusters is small enough (one hundred at most), an exhaustive search is applicable to search for trios or even subsets with larger dimensions (4 or 5). Following the PGN model, the criterion function needs to evaluate the prediction power of a candidate predictor subset with regard to the target expression at the next timepoint (first-order Markov Chain).

(e) Multivariate analysis: This step is necessary to eliminate redundant genes from the best predictor subset achieved for a given target. This can be done by evaluating the IMP score (IS) according to Equation 3. In the experiments of Section 5, a subset $\mathbf{Z}$ selected to predict Y is reduced if $IS(\mathbf{Z}, Y) = 0$. As already mentioned in Section 2, this reduction process is recursive: the reduction continues until $IS(\mathbf{Z}, Y) > 0$, achieving the minimal predictor subset. Once defined the final predictor subset for each target, it derives the dependence logics that rule the target expression profile based on its final predictor subset. These dependence logics are retrieved from the conditional probabilities distributions $P(Y|\mathbf{Z})$ (where Y is the target and $\mathbf{Z}$ is the best predictor subset for Y), in such a way that for all $\mathbf{z} \in \mathbf{Z}$, the Y output is defined by $\{y|P(Y = y|\mathbf{Z} = \mathbf{z}) = \max_{y \in Y} P(Y = y|\mathbf{Z} = \mathbf{z})\}$ (the logic outputs are those that minimize the Bayesian classification error of Y based on $\mathbf{Z}$ values). Thus, the expression of a gene at the time $t + 1$ is given by the application of the prediction logic of the predictors aforementioned by taking its expression values from time t as inputs (these expressions are obtained from the quantized dataset). This is performed considering all timepoints.

Finally, the output of our framework is a set of predictor *clusters* for each desired target gene. If necessary, a PGN can be assembled by combining all desired target genes with its predictor set. Thus, the output PGN is composed of all target genes linked to cluster representative genes.

4.1 Computational complexity analysis

As the computational complexity of the framework is mainly given by the exhaustive search algorithm in the Feature Selection step (step d), we focus only on the analysis of the complexity of this step. Other steps have a negligible processing time, since they are processed in seconds even for very big datasets. Hence, the complexity is measured according to the number of times that the criterion function is calculated during the *Feature Selection* step (so lets assume that one criterion function calculation presents $\mathcal{O}(1)$ time, which is true for small predictor subset cardinalities and small number of possible discrete expression values). Let N be the number of genes in the dataset, p be the fixed number of predictors for a predictor subset and k be the number of clusters obtained in step b. The complexity of inferring the gene network topology using the exhaustive search is given by: $\mathcal{O}(N \times \binom{k}{p}) = \mathcal{O}(N \times k^p)$, where N is the number of genes in the original dataset, k is the number of clusters, and p is the number of predictors in each subset. Since k is expected to be much smaller than N (k is in the order of tens or hundreds while N is in the order of thousands), the gain in computational time is substantial when compared to the pure exhaustive search, which presents complexity $\mathcal{O}(N \times \binom{N}{p}) = \mathcal{O}(N^{p+1})$. For example, in a dataset with $N = 1000$ and $p = 3$, the number of criterion function evaluations is equal to 1.66×10^8 for the pure exhaustive search, and 1.62×10^5 for our proposal with $k = 100$ (three orders of magnitude below).

5 Experimental Setup

We adopted the SysGenSIM to generate datasets for our experiments. SysGenSIM is an *in silico* method that generates gene expression data from non-linear differential equations based on biochemical dynamics of yeasts [Pinna *et al.*, 2011].

The following parameters were defined when generating the datasets: 3 different expression profiles were generated with 40 samples ($M = 40$) each. The Barabási-Albert scale-free model was adopted to generate the network topology [Barabási and Albert, 1999], and the average input degree was set to 3. The number of genes was defined as $N = 100$ and $N = 1000$ (one for each experiment). The cooperativity coefficient was set to a Gamma distribution and the degradation rate was constant. The biological variance of transcription, degradation and noise was set to a Gaussian distribution, and the other parameters were set to the default values provided by the simulator. These parameters should be defined such that the distribution of estimated "heritabilities" of the traits is close to those found in real data.

In the clustering step, the k-means method was applied to group genes with similar expression profiles. The parameter k, which indicates the number of clusters, was varied among 20, 30, 40, 50 and 100, and the Euclidean distance was adopted as the distance criterion. For each cluster, the gene with minimum Euclidean distance to the cluster centroid in terms of the expression profiles was selected to be the representative gene of the cluster.

After the clustering step, we set each gene of the N genes of the input dataset as the target gene, and then we performed an exhaustive search that evaluates all possible subsets of candidate predictor genes of size $p = 3$ to retrieve the best predictor subset according to the coefficient of determination (Equation 1) and the mutual information score (Equation 2) as criterion functions. Recall that candidate predictors are only the representative genes of the clusters (one for each of the k clusters), which were retrieved in the previous step. Then, a multivariate analysis is applied to further discard redundant predictors from subsets that present null IMP score with regard to their corresponding targets.

As we are interested in evaluating the structure of the inferred gene expression profile dynamics, the expression of a gene at the time t_{i+1} is given by the application of the prediction logic of its corresponding predictors by taking its expression values from time t_i obtained from the quantized dataset as inputs. This is performed considering all timepoints. Each inferred binary gene expression profile is compared with the corresponding binary gene expression profile from the quantized dataset. The percentage of correctly predicted timepoints defines the accuracy (it is equivalent to the Hamming distance between the two binary profiles divided by the number of timepoints of each profile). The average of accuracies obtained for all target genes is taken as the overall accuracy of the inferred dataset (values between 0 and 1, where 1 means perfect accuracy and 0.5 is the expected value obtained by random guesses of the binary gene expression profile values). Figure 2 illustrates this assessment.

In our experiments, we compare both accuracy and execution time between GRN inference by pure exhaustive search and by our proposed framework. The pure exhaustive search was performed only for datasets composed of $N = 100$ genes, since this method was unfeasible to compute in our hardware for $N > 100$ (see computational complexity analysis in Section 4.1). In contrast, our proposal was executed for datasets composed of $N = \{100, 1000\}$ genes. We evaluated the performance of our proposal for $k = \{20, 30, 40, 50, 100\}$, where k is the number of clusters.

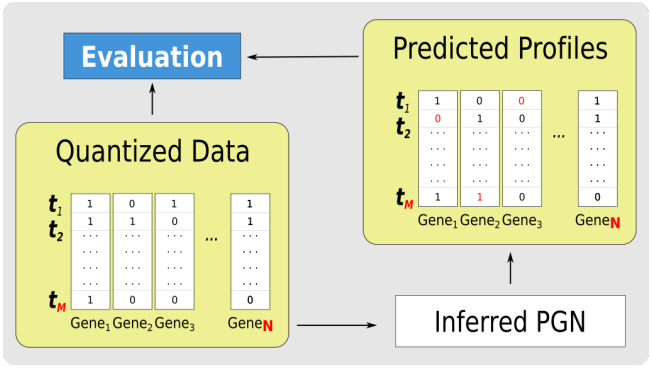

Figure 2: Evaluation of the framework. A quantized data sample corresponding to timepoint t_i is used to define the inferred sample at timepoint $t + 1$ by applying the prediction logics derived for every gene. Each inferred gene expression profile is evaluated by means of the accuracy (percentage of inferred timepoints correctly inferred). The overall accuracy of the inferred dataset is given by the average accuracy of all inferred gene expression profiles.

Our framework was implemented in R language (version 3.2.3). All experiments were executed in a computer Intel® Xeon® 8 core CPU E7- 2870 2.40 GHz with 32 GB RAM, under Linux Ubuntu 64-bit operating system.

6 Results and Discussion

Table 1 shows the average prediction accuracy of the inferred expression profiles taking the quantized dataset as ground truth for different numbers of clusters and for the pure exhaustive with $N = 100$.

It is noteworthy that the accuracy loss was very small when using our proposal, specially for $k = \{50, 100\}$, while the processing time is substantially reduced when compared to the pure exhaustive search. Note also that our proposal spent less than 30 minutes regardless of the number of total genes, which means that our proposal is scalable in terms of total number of genes. As predicted by our theoretical analysis, the execution time in our proposal is affected by the number of clusters, and the overhead introduced by the clustering step is negligible.

The exhaustive search, by its turn, is unfeasible to execute in the greater number of genes. According to our estimates, if the pure exhaustive search was fully processed for a single target gene considering $N = 1000$ genes, it would spend about 28,800 minutes (20 days) according to our estimates, which is roughly 1000 times longer than the processing time required by our proposed framework considering $N = 1000$ and $k = 100$. Finally, it is also noteworthy that real expres-

Table 1: Average precision of all N gene expression profiles and observed processing time for a single gene achieved by both our proposal and the pure exhaustive search (represented by Exha. columns) for $N = \{100, 1000\}$ genes. In our proposal the k-means clustering was adopted considering $k = \{20, 30, 40, 50, 100\}$. In both cases, the exhaustive feature selection was applied using Mutual Information (MI) and Coefficient of Determination (CoD) as criterion functions. Times denoted by m (minutes) and d (days). Times with $*$ symbol are estimated.

		***N* = 100 Genes**					***N* = 1000 Genes**					
		***k*=20**	***k*=30**	***k*=40**	***k*=50**	**Exha.**	***k*=20**	***k*=30**	***k*=40**	***k*=50**	***k*=100**	**Exha.**
MI	Acc. (%)	82.57	84.84	85.41	86.86	89.84	81.99	84.30	85.78	86.75	89.62	$--$
	Time	$<$ 1m	$\approx$ 1m	$\approx$ 3m	$\approx$ 6m	$\approx$ 29m	$<$ 1m	$\approx$ 1m	$\approx$ 3m	$\approx$ 6m	$\approx$ 29m	$*20d$
CoD	Acc. (%)	83.59	85.73	86.03	87.70	90.46	83.19	85.28	86.70	87.65	90.29	$--$
	Time	$<$ 1m	$\approx$ 1m	$\approx$ 2m	$\approx$ 5m	$\approx$ 28m	$<$ 1m	$\approx$ 1m	$\approx$ 2m	$\approx$ 5m	$\approx$ 28m	$*20d$

sion data often have greater dimensionality than $N = 1000$, which means that our proposal may be an useful method in many domains in which the exhaustive search was inapplicable.

It can also be noted in our results that the prediction accuracies obtained by the CoD and the MI scores used as search criterion functions were very similar. These experiments were repeated 10 times for $k = 100$ and $N = 1000$, the CoD score achieved an average accuracy of 90.29% with 2% of standard deviation, while the MI score achieved 89.62% with 3% of standard deviation.

7 Conclusion and future work

In this paper we describe a new framework for gene regulatory networks inference, in which a clustering method is applied to reduce the complexity of the predictor subsets search for each gene placed as target. We demonstrated the applicability of our proposal in experiments using synthetic data, for which it was able to preserve the prediction accuracy obtained by the pure exhaustive search, but substantially reduced the computational complexity of the search. In addition, it is important to highlight that the synthetic datasets were generated by a complex and detailed model (non-linear differential equations based on biochemical dynamics of yeasts [Pinna *et al.*, 2011]), while the PGN model on which our framework relies is much simpler. Even assuming a simpler model, our framework described the synthetic expression profiles with great accuracy (about 90%) considering datasets with 1000 genes and setting 100 clusters.

The next step is to evaluate the performance of our proposal in real gene expression datasets. Besides, as our proposal consists in a framework, several aspects regarding the different steps involved can be improved. For example, other clustering algorithms can be tested as well as other distance metrics and methods to define the representative genes. Also, the clustering algorithm can be applied after the quantization step, which might lead to clusters with less variability among their respective gene expression profiles.

Even though completely understanding and modeling the properties and structures of real biological systems is still an open problem, our proposal showed promise in assisting professionals of biomedicine and related areas in decision-making regarding the control of the gene regulatory systems dynamics. Our proposal also provides a viable system in environments with limited computing resources, which was not possible considering previous works that applied exhaustive search as a way to guarantee the best predictor subset for each target.

Finally, our proposal showed to be scalable, since we were able to increase in ten times the number of genes in the input expression data without increase in the processing time of exhaustive feature selection for a single target gene, which implies that the processing time linearly increases with the number of genes in the whole network.

References

[Akutsu *et al.*, 1999] T. Akutsu, S. Miyano, S. Kuhara, et al. Identification of Genetic Networks from a small number of Gene Expression Patterns under the Boolean Network Model. In *Proceedings of the Pacific Symposium on Biocomputing (PSB)*, volume 4, pages 17–28, 1999.

[Barabási and Albert, 1999] A. L. Barabási and R. Albert. Emergence of scaling in Random Networks. *Science*, 286(5439):509–512, 1999.

[Barrera *et al.*, 2007] J. Barrera, R. M. Cesar-Jr, D. C. Martins-Jr, R. Z. N. Vencio, E. F. Merino, M. M. Yamamoto, F. G. Leonardi, C. A. B. Pereira, and H. A. del Portillo. Constructing Probabilistic Genetic Networks of *Plasmodium falciparum* from Dynamical Expression Signals of the Intraerythrocytic Development Cycle. In *Methods of Microarray Data Analysis V*, chapter 2, pages 11–26. Springer, 2007.

[Borelli *et al.*, 2013] F. F. Borelli, R. Y. de Camargo, D. C. Martins-Jr, and L. C. S. Rozante. Gene Regulatory Networks Inference using a multi-GPU Exhaustive Search algorithm. *BMC Bioinformatics*, 14(S5), 2013.

[Carastan-Santos *et al.*, 2016] D. Carastan-Santos, R. Y. Camargo, D. C. Martins-Jr, S. W. Song, and L. C. S. Rozante. Finding Exact Hitting Set Solutions for Systems Biology applications using heterogeneous GPU Clusters. *Future Generation Computer Systems*, 2016. (in press).

[Cover and van Campenhout, 1977] T. M. Cover and J. M. van Campenhout. On the Possible Orderings in the Measurement Selection Problem. *IEEE Transactions on Systems, Man and Cybernetics*, 7(9):657–661, 1977.

[De-Jong, 2002] H. De-Jong. Modeling and Simulation of Genetic Regulatory Systems: A Literature Review. *Journal of Computational Biology*, 9(1):67–103, 2002.

[De-Smet and Marchal, 2010] R. De-Smet and K. Marchal. Advantages and Limitations of Current Network Inference Methods. *Nature Reviews Microbiology*, 8(10):717–729, 2010.

[Dougherty *et al.*, 2000] E. R. Dougherty, S. Kim, and Y. Chen. Coefficient of Determination in Nonlinear Signal Processing. *Signal Processing*, 80:2219–2235, 2000.

[Faith *et al.*, 2007] J. Faith, B. Hayete, J. Thaden, I. Mogno, J. Wierzbowski, G. Cottarel, S. Kasif, J. Collins, and T. Gardner. Large-scale mapping and validation of *Escherichia coli* transcriptional regulation from a compendium of expression profiles. *PLoS Biology*, 5:259–265, 2007.

[Friedman *et al.*, 2000] N Friedman, M Linial, I Nachman, and D Pe'er. Using Bayesian Networks to Analyze Expression Data. *Journal of Computational Biology*, 7(3-4):601–20, January 2000.

[Hecker *et al.*, 2009] M. Hecker, S. Lambeck, S. Toepfere, E. Van-Someren, and R. Guthke. Gene Regulatory Network Inference: Data Integration in Dynamic Models: a Review. *Biosystems*, 96(1):86–103, 2009.

[Ideker *et al.*, 2000] T. Ideker, V. Thorsson, and R. M. Karp. Discovery of Regulatory Interactions through Perturbation: Inference and Experimental Design. In *Proceedings of the Pacific Symposium on Biocomputing (PSB)*, volume 5, pages 302–313, 2000.

[Ivanov and Dougherty, 2006] I. Ivanov and E. R. Dougherty. Modeling Genetic Regulatory Networks: Continuous or Discrete? *Journal of Biological Systems*, 14(2):219–229, 2006.

[Kauffman, 1969] S. A. Kauffman. Metabolic Stability and Epigenesis in Randomly Constructed Genetic Nets. *Journal of Theoretical Biology*, 22(3):437–467, 1969.

[Lahdesmaki and Shmulevich, 2003] H. Lahdesmaki and I. Shmulevich. On Learning Gene Regulatory Networks under the Boolean Network Model. *Machine Learning*, 52:147–167, 2003.

[Liang *et al.*, 1998] S. Liang, S. Fuhrmane, and R. Somogyi. Reveal, a General Reverse Engineering Algorithm for Inference of Genetic Network Architectures. In *Proceedings of the Pacific Symposium on Biocomputing (PSB)*, volume 3, pages 18–29, 1998.

[Lopes *et al.*, 2014] F. M. Lopes, D. C. Martins-Jr, J. Barrera, and R. M. Cesar-Jr. A Feature Selection Technique for Inference of Graphs from their Known Topological Properties: Revealing Scale-free Gene Regulatory Networks. *Information Sciences*, 272:1–15, 2014.

[Marbach *et al.*, 2012] Daniel Marbach, James C Costello, Robert Küffner, Nicole M Vega, Robert J Prill, Diogo M Camacho, Kyle R Allison, Manolis Kellis, James J Collins, and Gustavo Stolovitzky. Wisdom of Crowds for Robust Gene Network Inference. *Nature Methods*, 9(8):796–804, 2012.

[Margolin *et al.*, 2006] A. A. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. D. Favera, and A. Califano. ARACNE: An Algorithm for the Reconstruction of Gene Regulatory Networks in a Mammalian Cellular Context. *BMC Bioinformatics*, 7(Suppl 1):S7, 2006.

[Markowetz and Spang, 2007] F. Markowetz and R. Spang. Inferring Cellular Networks – A Review. *BMC Bioinformatics*, 8(Suppl 6):S5, 2007.

[Martins-Jr *et al.*, 2008] D. C. Martins-Jr, U. M. Braga-Neto, R. F. Hashimoto, M. L. Bittner, and E. R. Dougherty. Intrinsically Multivariate Predictive Genes. *IEEE Journal of Selected Topics in Signal Processing*, 2(3):424–439, 2008.

[Meyer *et al.*, 2007] P. Meyer, K. Kontos, F. Lafitte, and G. Bontempi. Information Theoretic Inference of Large Transcriptional Regulatory Networks. *EURASIP Journal on Bioinformatics and Systems Biology*, 2007:1–9, 2007.

[Nam *et al.*, 2006] D. Nam, S. Seo, and S. Kim. An Efficient Top-down Search Algorithm for Learning Boolean Networks of Gene Expression. *Machine Learning*, 65:229–245, 2006.

[Pinna *et al.*, 2011] A. Pinna, N. Soranzo, I. Hoeschele, and A. de-la Fuente. Simulating Systems Genetics Data with SysGenSIM. *Bioinformatics*, 27(17):2459–2462, 2011.

[Pudil *et al.*, 1994] P. Pudil, J. Novovicová, and J. Kittler. Floating Search Methods in Feature Selection. *Pattern Recognition Letters*, 15:1119–1125, 1994.

[Shalon *et al.*, 1996] D. Shalon, S. J. Smith, and P. O. Brown. A DNA Microarray System for Analyzing Complex DNA Samples Using Two-color Fluorescent Probe Hybridization. *Genome Res*, pages 639–45, 1996.

[Shannon, 2001] C. E. Shannon. A Mathematical Theory of Communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55, 2001.

[Shmulevich and Dougherty, 2014] I. Shmulevich and E. R. Dougherty. *Genomic Signal Processing*. Princeton University Press, 2014.

[Shmulevich *et al.*, 2002] I. Shmulevich, E. R. Dougherty, S. Kim, and W. Zhang. Probabilistic Boolean Networks: A Rule-based Uncertainty Model for Gene Regulatory Networks. *Bioinformatics*, 18(2):261–274, 2002.

[Styczynski and Stephanopoulos, 2005] M. P. Styczynski and G. Stephanopoulos. Overview of Computational Methods for the Inference of Gene Regulatory Networks. *Computers & Chemical Engineering*, 29(3):519–534, 2005.

[Velculescu *et al.*, 1995] V. E. Velculescu, L. Zhang, B. Vogelstein, and K. W. Kinzler. Serial Analysis of Gene Expression. *Science*, 270:484–487, 1995.

[Wang *et al.*, 2009] Z. Wang, M. Gerstein, and M. Snyder. RNA-Seq: A Revolutionary Tool for Transcriptomics. *Nat Rev Genet*, 10(1):57–63, 2009.

[Zhao and Karypis, 2003] Y. Zhao and G. Karypis. Clustering in Life Sciences. *Functional Genomics: Methods and Protocols*, pages 183–218, 2003.