# Comparing gene regulatory inferring algorithms with different perspective

**Shaimaa M. Elembaby[1,*], Vidan F. Ghoneim[2], Manal Abdel Wahed[1]**

*1. Faculty of engineering, Cairo University, Giza, Egypt*

*2. Faculty of engineering, Helwan University, Cairo, Egypt*

*eng_s_elembaby@yahoo.com*

ABSTRACT. *More than hundred algorithms were developed to infer Gene Regulatory Networks (GRN) describing relations between genes. GRN construction has been a field of interest to researchers since the beginning of the current century. Many competitions were held to encourage the development of GRN inference algorithms, such competitions offer synthetic data to enable the validation of proposed algorithms. A GRN is constructed from an adjacency matrix which contains relations between genes. The developers of many of the GRN inference algorithms set a threshold on the adjacency matrix to construct GRN based on high gene-gene relation weights. This threshold strategy was followed in previous studies to increase the accuracy of any algorithm but yet based on no well-known rule. A different perspective here is to compare different GRN inference algorithms without setting any threshold. Comparison in this work is among different GRN inference algorithms by implementing all algorithms with no threshold on values of adjacency matrices: Differential Equation methods (TSNI), Granger Causality, GP4GRN, GENIE3, NIMEFI (SVR), and PLSNET. Another comparison between different distance metric equations to create adjacency matrix is also studied in an attempt to construct GRN. GP4GRN and GENIE3 participate in producing best results for dream4 InSilico_Size10 while GENIE3 produce best results for all networks of dream4 InSilico_Size100.*

RÉSUMÉ. *Plus de cent algorithmes ont été développés pour déduire des réseaux de régulation de gènes (GRN) décrivant les relations entre gènes. La construction de GRN est un domaine d'intérêt pour les chercheurs depuis le début du siècle actuel. De nombreux concours ont été organisés pour encourager le développement d'algorithmes d'inférence GRN. Ces concours offrent des données synthétiques pour permettre la validation des algorithmes proposés. Un GRN est construit à partir d'une matrice d'adjacence qui contient les relations entre les gènes. Les développeurs de nombreux algorithmes d'inférence GRN ont défini un seuil pour la matrice d'adjacence afin de construire un GRN basé sur des poids de relation gène-gène élevés. Cette stratégie de seuil a été suivie dans des études précédentes pour augmenter la précision de tout algorithme, sans pour autant s'appuyer sur aucune règle bien connue. Une autre perspective consiste à comparer différents algorithmes d'inférence GRN sans définir de seuil. La comparaison dans ce travail est faite entre différents algorithmes d'inférence GRN en implémentant tous les algorithmes sans seuil sur les valeurs des matrices d'adjacence:*

*Méthodes d'équation différentielle (TSNI), causalité de Granger, GP4GRN, GENIE3, NIMEFI (SVR) et PLSNET. Une autre comparaison entre différentes équations métriques de distance pour créer une matrice d'adjacence est également étudiée dans le but de construire un GRN. GP4GRN et GENIE3 contribuent à produire les meilleurs résultats pour dream4 InSilico_Size10, tandis que GENIE3 fournit les meilleurs résultats pour tous les réseaux de dream4 InSilico_Size100.*

KEYWORDS: *gene regulatory network, adjacency matrix, distance metrics.*

MOTS-CLÉS: *réseau de régulation de gènes, matrice d'adjacence, métriques de distance.*

DOI:10.3166/I2M.17.653-661 © 2018 Lavoisier

## 1. Introduction

Interference of Gene Regulatory Network (GRN) is essential to understand genetic changes in the cell. This makes GRN an important phase in designing drugs and vaccines in the medical field. GRN is treated in computations as a matrix (adjacency matrix). This adjacency matrix has zero diagonal if effect of each gene on itself is neglected. The elements of the adjacency matrix are the weights of the links connecting genes in a sparse network. A sparse network (GRN) can be generated as suggested by many researchers from the fully connected network by removing all edges below a definite threshold. Thus, most important links in gene regulatory sparse network can be obtained by increasing this threshold. Although threshold makes comparing different methods of inferring GRN subjected to bias.

Thinking of how to represent relation between genes, scientists in early researches used correlation and its types as Partial correlation (PCIT), (Reverter & Chan, 2008), mutual information, (Meyer *et al.*, 2008), Boolean Network, (Akutsu *et al.*, 1998; Thomas, 1991), Bayesian network, (Jing *et al*., 2010), Dynamic Bayesian Network, (Yghoobi *et al*., 2012) and network based on Linear Differential Equations, (Bansal *et al*., 2006).

Later, GENIE3 and other algorithms have been developed since 2010, where it achieved improvement in GRN inference results. GENIE3 algorithm which decomposes GRN of N genes into N different regression problems. Each subproblem is solved by tree based ensemble method, (Huynh-Thu *et al.*, 2010), ENN *et al.* gorithm is improvement of GENIE3 algorithm combines Gradient Boosting with regression Stumps to select subset of edges for building global GRN, (Sławek & Arodź, 2013), NIMEFI algorithm solve P subproblems of GRN by Support Vector Regression (E-SVR) or Ensemble Elastic Net, (Ruyssinck *et al.*, 2014), PLSN *et al.* gorithm which use Partial least squares (PLS) based feature selection method to solve P subproblems, (Guo *et al.*, 2016) nonlinear correlation coefficient derived from two-way analysis of variance(ANOVAs) between transcription factor TF and target gene TG, (Küffner *et al.*, 2012). Network Deconvolution used to improve results of inference of other methods, (Feizi *et al.*, 2013). Models depended on computational swarm intelligence (Particle Swarm Optimization PSO (Kesavan *et al*., 2016; Liang *et al.*, 2016) or Ant Colony Optimization ACO) used also to infer GRN, (Kentzoglanakis & Poole, 2012). There are many methods to infer GRN because inferring GRN is still a field of research. DREAM Challenges (DREAM3, DREAM4

and DREAM5) in Synapse site give us data which can be used for evaluation any method, computation the accuracy of this method and comparing its results with other methods results (Samee *et al.*, 2012). In this paper comparison between algorithms will be according to the area under the ROC curve only.

DREAM4 competition time series Insilco 10 and 100 gene used here to compare some algorithms based on the corresponding GRN design as a whole without ordering of links in network and taking a threshold.

## 2. Materials and methods

### *2.1. Dataset used in comparison*

InSilico_Size10 and InSilico_Size100 sub-challenges of DREAM4 were used in this comparison. Each sub-challenge consists of five networks. Here two files of each network were used. First file contains time series for genes and second file contain gold standard of network. In time series file, each simulation contains 21 time point (from t=0 to t=1000). At t=0 perturbation is applied time points show how the network response to perturbation until t=500, after t=500 perturbation is removed until t=1000 time points show how network relaxes. This simulation is repeated 5 times for network of size 10 genes (InSilico_Size10) and repeated 10 times for network of size 100 genes (InSilico_Size100), (http://wiki.c2b2.columbia.edu/dream/data/DREAM4.).

### *2.2. Distance metric equations to represent relation between genes*

Statistical method to compute distance (used as relation to get adjacency matrix). After getting adjacency matrix gold standard of each network of DREAM4 is compared with adjacency matrix of each network. Area under the ROC curve is recorded of networks of DREAM4 InSilico_Size10 in table1 and DREAM4 InSilico_Size100 in table2.

**Euclidean and Standard Euclidean Distances:** The Euclidean distance between two vectors, G1 and G2, with N samples calculated as in (Deza & Deza, 2009):

$$distance = \sqrt{\sum_{i=1}^{N}(G1_i - G2_i)^2} \quad (1)$$

Usually Euclidean distances are computed from raw data not from standardized data. Standardization is essential to balance the contributions of the variables in the computation of distance when variables are on different measurement scales. The Euclidean distance which computed on standardized variables is called the standardized Euclidean distance, (Greenacre & Primicerio, 2014).

**The city block distance** between two point G1 and G2, with N samples is defined as:

$$distance = \sum_{i=1}^{N}|G1_i - G2_i| \quad (2)$$

The City block distance (also named by Manhattan distance) is explained if there are two points in XY plane. The City block distance is calculated as the distance in x plus the distance in y (McCune & Grace, 2002),

**Chebyshev distance** Cyrus is calculated on a vector space where the distance between two vectors is the greatest of their differences along any coordinate dimension, and is defined between two point G1 and G2, with N samples or dimensions as:

$$distance_i = \max(|G1_i - G2_i|) = \lim_{k\to\infty}(\sum_{i=1}^{N}|G1_i - G2_i|^k)^{1/k} \quad (3)$$

**Cosine Distance:** is one minus the cosine of the included angle between two vectors. The cosine distances between the vector G1 and G2 are defined as follows:

$$distance = 1 - \frac{G1\acute{G}2}{\sqrt{(G1\acute{G}1)(G2\acute{G}2)}} \quad (4)$$

**Correlation Distance:** correlation distance is measuring of the dependence between random vectors (Székely *et al*., 2007). Given an M by N data matrix G, which is treated as m (1-by-n) row vectors G1, G2, ..., Gm, the correlation distances between the vector G1 and G2 are defined, (Lian *et al.*, 2017):

$$distance = 1 - \frac{(G1-\overline{G1})(G2\acute{}-\overline{G2})}{\sqrt{(G1-\overline{G1})(G1\acute{}-\overline{G1})}\sqrt{(G2-\overline{G2})(G2\acute{}-\overline{G2})}} \quad (5)$$

*Table 1. Distance between each pair of genes of dream4 InSilico_Size10 gene as adjacency matrix and its ROC*

| Distance type | Net1 | Net2 | Net3 | Net4 | Net5 |
|---|---|---|---|---|---|
| Euclidean | 0.4914 | 0.5669643 | 0.5024 | 0.55924 | 0.55303 |
| Standard Euclidean | 0.4914 | 0.581845 | 0.5039 | 0.575155 | 0.55114 |
| cityblock | **0.5259** | 0.56399 | 0.51804 | 0.56101 | 0.55303 |
| chebychev | 0.4694 | 0.583333 | 0.46471 | 0.515031 | 0.5322 |
| cosine | 0.4051 | **0.6369** | 0.63569 | 0.6088 | 0.5701 |
| correlation | 0.4145 | 0.50893 | 0.6200 | 0.564545 | 0.6572 |
| spearman | 0.4051 | 0.529762 | **0.671765** | **0.62467** | **0.6705** |

The Spearman correlation assesses monotonic relationships while Pearson's correlation assesses linear relationships. Spearman's correlation between two variables is equal to the Pearson correlation between the rank values of those two variables. Spearman's correlation (whether linear or not). A perfect Spearman correlation of +1 or −1 occurs when each of the variables is a perfect monotone function of the other If there are no repeated data values. distance of Spearman=1- the Spearman's correlation.

*Table 2. Distance between each pair of of dream4 InSilico_Size100 gene as adjacency matrix and its ROC*

| Distance type | Net1 | Net2 | Net3 | Net4 | Net5 |
|---|---|---|---|---|---|
| Euclidean | 0.55524 | 0.504 | 0.47235 | 0.4848 | 0.46046 |
| Standard Euclidean | 0.555 | 0.5036 | 0.47190 | 0.48476 | 0.4600 |
| cityblock | **0.5594** | 0.5044 | 0.47655 | 0.4883 | 0.46779 |
| chebychev | 0.5233 | 0.5055 | 0.469834 | 0.4754 | 0.45777 |
| cosine | 0.424 | **0.5098** | 0.4920 | 0.4770 | **0.5073** |
| correlation | 0.493 | 0.4706 | **0.5394** | **0.52664** | 0.5063 |
| spearman | 0.492 | 0.4708 | 0.5356 | 0.526385 | 0.4973 |

### *2.3. Algorithms used to infer GRN*

**GENIE3:**

GENIE3 decomposes the inference of GRN into different regression problems, in each there is one only (target gene) is predicted from all the other genes (input genes), GENIE3 is built on tree-based ensemble methods Random Forests or Extra-Trees. Putative regulatory links are then aggregated over all genes to provide a ranking of interactions from which the whole network is reconstructed.

**NIMEFI:**

NIMEFI (Network Inference using Multiple Ensemble Feature Importance algorithms). NIMEFI algorithm decomposes the inference of GRN into separate regression problems for each gene. NIMEFI use support vector regression, the elastic net, symbolic regression and compare it with random forest regression used in GENIE3 algorithm. NIMEFI use ensemble feature selection (EFS) method.

**PLSNET:**

PLSNET is a new ensemble GRN inference method use Partial least squares basedfeatureselectionalgorithmtakingrandompotentialregulatorygenes. PLSNET decomposes the GRN inference problem with N genes into N subproblems and solves each of the subproblems by using Partial least squares (PLS) based feature selection algorithm, (Guo *et al*., 2016).

**TSNI (Time Series Network Identification)**

It uses ordinary differential equation to represent relation of gene with other genes and other perturbation. It uses smoothed interpolating for increasing the number of samples by using piecewise cubicspline interpolation. Finally TSNI apply Principle Component Analysis (PCA) to reduce dimensionality of the problem and solve the equation (Bansal *et al.*, 2006).

**Granger causal connectivity analysis (GCCA)**

According to Granger causality, a variable G1 'Granger causes' a variable G2 if information of G1 in the past helps in predicting the future of G2 with better accuracy than is possible when considering only information of G2 in the past only (Granger, 1969) the temporal dynamics of two time series G1 (t) and G2 (t) (both of length T) is described by a bivariate autoregressive model:

$$G_1(t) = \sum_{j=1}^{p} A_{11,j} G_1(t-j) + \sum_{j=1}^{p} A_{12,j} G_2(t-j) + \varepsilon_1(t) \quad (6)$$

$$G_2(t) = \sum_{j=1}^{p} A_{21,j} G_1(t-j) + \sum_{j=1}^{p} A_{22,j} G_2(t-j) + \varepsilon_2(t) \quad (7)$$

If the variance of ε1 (or ε2) is reduced by the inclusion of the G2 (or G1) terms in the first (or second) equation, then it is said that G2 (or G1) Granger causes G1 (or G2).

If there are more than two variables then multivariate autoregressive (MVAR) models will be used. G2 Granger causes G1 if knowing G2 reduces the variance in G1's prediction error ε1 when all other variables G3……Gn are also included in the regression model (Seth, 2010).

**GP4GRN:**

It dependS on using of Bayesian analysis, ordinary differential equations (ODEs) and non-parametric Gaussian process modeling. The main differences between this method and other methods based on ODE as TSNI, (Bansal *et al*., 2006) and Inferelator, (Bonneau, 2006), are nonparametric modeling and Bayesian analysis. Bayesian approach is suitable for uncertainty of measurements and assuming normally distributed noise (Äijö & Lähdesmäki, 2009)

*Table 3. AUROC curve of some algorithms for DREAM4 Insilco 10 genes*

| Algorithms | Network1 | Network2 | Network3 | Network4 | Network5 |
|---|---|---|---|---|---|
| GEINE3 | 0.8384 | 0.6726 | ***0.7184*** | ***0.730*** | ***0.83*** |
| NIMEFI (SVR) | 0.5671 | 0.5372 | 0.5388 | 0.4235 | 0.5483 |
| TSNI (average of five simulation) | 0.6659 | 0.4807 | 0.6784 | 0.5464 | 0.4924 |
| Granger causality (average of five simulation) | 0.5333 | 0.5574 | 0.4987 | 0.3876 | 0.2938 |
| Granger causality | 0.4836 | 0.4130 | 0.3724 | 0.4505 | 0.2853 |
| GP4GRN | ***0.8873*** | ***0.6994*** | 0.7161 | 0.7165 | 0.7325 |

*Table 4. AUROC curve of some algorithms for dream4 Insilco 100 gene*

| Algorithms | Network1 | Network2 | Network3 | Network4 | Network5 |
|---|---|---|---|---|---|
| GEINE3 | 0.762 | 0.689 | 0.744 | 0.719 | 0.78 |
| NIMEFI (SVR) | 0.4696 | 0.5272 | 0.5809 | 0.5609 | 0.5514 |
| PLSNET | 0.7118 | 0.6048 | 0.6505 | 0.6787 | 0.67386 |
| TSNI | 0.6445 | 0.5320 | 0.5325 | 0.5320 | 0.5078 |
| Granger causality | 0.5578 | 0.5405 | 0.4853 | 0.5188 | 0.5024 |

## 3. Results and discussions

After applying GRN inference algorithms on Dream4 InSilico_Size10: GP4GRN gave best results for network1 and network2, GENIE3 gave best results for network 3, 4 and 5, while GENIE3 produce best results for all networks of dream4 InSilico_Size100.

In granger causality as a lot of other algorithms, the number of time points (samples) must be greater than the number of genes (unknowns variables). So, granger causality algorithm used all time samples (210) to calculate GRN of 100 genes of Dream4 competition data [InSilico_Size100].Results of all algorithms include all time samples (105 sample) of Dream4 InSilico_Size10 come from five simulations and (210 sample) for Dream4 InSilico_Size100 come from10simullation to calculate GRN directly without dividing in simulations except in results of TSNI and Granger causality of 10 genes algorithms which applied on each simulation and average is taken in results.

Bold number represent highest accuracy algorithm in each network based on area under ROC curve, GP4GRN takes lot time to infer GRN of 100 gene, it takes 15 day to infer first network of DREAM 4 InSilico_Size100 and produce AUROC=0.685 which less than GIENE 3. There are no results of PLSNET in DREAM4 InSilico_Size10 because it is designed for large scale networks.

All algorithms gave us adjacency matrix and ranked list except NIMEFI which gave us ranked list only. Here converting from ranked list results of NIMEFI to adjacency matrix has occurred without any truncation limit, this process takes some efforts but it is essential in comparison of algorithms without any threshold.

Correlation distance in table 1and2 is different with person correlation in table 3and4 as shown in previous explanation and equations.

## 4. Conclusion

A new idea of distance metric equation failed in GRN inference because it cannot compete with GIENE3, PLSNET and GP4GRN in inferring GRN. GIENE3 recorded

highest AUROC with DREAM 4 InSilico_Size100. Whereas, GP4GRN and GIENE3 recorded highest AUROC with DREAM 4 InSilico_Size10. Although NIMEFI (SVR) was introduced as improving of GIENE3 but it gave bad results with DREAM4. Although PLSNET gave best result with DREAM5 which has the largest number of genes but it gave bad results with DREAM4. Applying algorithms draw us to conclude that GP4GRN is suitable for small networks while PLSNET is best for large networks.

## References

Äijö T., Lähdesmäki H. (2009). Learning gene regulatory networks from gene expression measurements using non-parametric molecular kinetics. *Bioinformatics*, Vol. 25, No. 22, pp. 2937-2944. https://doi.org/10.1093/bioinformatics/btp511

Akutsu T., Kuhara S., Maruyama O., Miyano S. (1998). *A system for identifying genetic networks from gene expression patterns produced by gene disruptions and over expressions*. Universal Academy Press.

Bansal M., Gatta G. D., Bernardo D. (2006). Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. *Bioinformatics*, Vol. 22, No. 7, p. 815-822. http://doi.org/10.1093/bioinformatics/btl003

Bonneau R., Reiss D. J., Shannon P., Facciotti M., Hood L., Baliga N., Thorsson V. (2006). The Inferelator: An algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo. *Genome Biol.*, Vol. 7, No. 5, pp. R36. http://doi.org/10.1186/gb-2006-7-5-r36

Deza E., Deza M. M. (2009). Encyclopedia of distances. *Springer*, pp. 94. http://doi.org/10.1007/978-3-642-00234-2

Feizi S., Marbach D., Médard M., Kellis M. (2013). Network deconvolution as a general method to distinguish direct dependencies in networks. *Nature Biotechnology*, Vol. 31, No. pp. 8. http://doi.org/10.1038/nbt.2635

Granger C. W. J. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, Vol. 37, No. 3, pp. 424-438.

Guo S., Jiang Q., Chen L., Guo D. (2016). Gene regulatory network inference using PLS-based methods. *BMC Bioinformatics*, Vol. 17, No. 1, pp. 545. http://doi.org/10.1186/s12859-016-1398-6.

Huynh-Thu V. A., Irrthum A., Wehenkel L., Geurts P. (2010). Inferring regulatory networks from expression data using tree-based methods. *PLoS One*, Vol. 5, No. 9. pp. e12776. https://doi.org/10.1371/journal.pone.0012776

Jing L., Michael K. Ng, Liu Y. (2010). Construction of gene networks with hybrid approach from expression profile and gene ontology. *IEEE Transactions On Information Technology In Biomedicine*, Vol. 14, No. 1, pp. 107-118. https://doi.org/10.1109/TITB.2009.2033056

Kentzoglanakis K., Poole M. (2012). A swarm intelligence framework for reconstructing gene networks: Searching for biologically plausible architectures. *IEEE/ACM Transactions On Computational Biology And Bioinformatics*, Vol. 9, No. 2, pp. 358-371. http://doi.org/10.1109/TCBB.2011.87

Kesavan E., Gowthaman N., Tharani S., Manoharan S., Arunkumar E. (2016). Design and implementation of internal model control and particle swarm optimization based PID for heat exchanger system. *International Journal of Heat and Technology*, Vol. 34, No. 3, pp. 386-390. http://doi.org/10.18280/ijht.340306

Küffner R., Petri T., Tavakkolkhah P., Windhager L., Zimmer R. (2012). Inferring gene regulatory networks by ANOVA. *Bioinformatics*, Vol. 28, No. 10, pp. 1376-1382. http://doi.org/10.1093/bioinformatics/bts143

Lian R., Zhou C., Goertzel B. (2017). Probabilistic rank correlation - a new rank and comparison based correlation coefficient with a simple. *Pragmatic Transitivity Condition. AMSE Journals, IIETA publications, Advances A*, Vol. 54, No. 4, pp. 476-496. https://doi.org/10.18280/ama_a.540403

Liang C. H., Zeng S., Li Z. X., Yang D. G., Sherif S. A. (2016). Optimal design of plate-fin heat sink under natural convection using a particle swarm optimization algorithm. *International Journal of Heat and Technology*, Vol. 34, No. 2, pp. 275-280. http://doi.org/10.18280/ijht.340217

McCune B., Grace J. B. (2002). *Analysis of Ecological Communities*. MjM Software, Gleneden Beach, Oregon, USA. ISBN: 0-9721290-0-6

Meyer P. E., Lafitte F., Bontempi G. (2008). MiNET: A R/Bioconductor package for inferring large transcriptional networks using mutual information. *BMC Bioinformatics*, Vol. 9, pp. 461. https://doi.org/10.1186/1471-2105-9-461

Reverter A., Chan E. K. F. (2008). Combining partial correlation and an information theory approach to the reversed engineering of gene co-expression networks. *Bioinformatics*, Vol. 24, No. 21, pp. 2491-2497. https://doi.org/10.1093/bioinformatics/btn482

Ruyssinck J., Geurts P., Dhaene T., Huynh-Thu V. A., Demeester P., Saeys Y. (2014). Nimefi: gene regulatory network inference using multiple ensemble feature importance algorithms. *PLoS One*, Vol. 9, No. 3, pp. e92709. https://doi.org/10.1371/journal.pone.0092709

Seth A. K. (2010). A MATLAB toolbox for Granger causal connectivity analysis. *Elsevier, Journal of Neuroscience Methods*, Vol. 186, No. 2, pp. 262-273.

Sławek J., Arodź T. (2013). ENNET: inferring large gene regulatory networks from expression data using gradient boosting. *BMC Syst Biol*., Vol. 7, No. 1, pp. 106. https://doi.org/10.1186/1752-0509-7-106

Székely G. J., Rizzo M. L., Bakirov N. K. (2007). Measuring and testing dependence by correlation of distances. *Ann. Statist*., Vol. 35, No. 6, pp. 2313-2817.

Thomas R. (1991). Regulatory networks seen as asynchronous automata: A logical description. *Journal of Theoretical Biolgy*, Vol. 153, pp. 1-23.

Yghoobi H., Haghipour S., Hamzeiy H., Asadi- Khiavi M. (2012). A review of modelling techniques for genetic regulatory networks. *Journal of Medical Signals and sensors*, Vol. 2, No. 1, pp. 61-70. http://wiki.c2b2.columbia.edu/dream/data/DREAM4