**LISTED RESPONSE TO COMMENT/SUGGESTIONS OF**

***REVIEWER 1***

|  |  |  |  |
| --- | --- | --- | --- |
| ***No*** | ***Reviewer’s Comment/Suggestion*** | ***Author’s Comment*** | ***Action Taken\*\*\**** |
| 1 | In Section "D. Gene Regulatory Network Inference", the authors show that BANJO and ARACNE produce more accurate networks when generated data is used, as opposed to using the original data. It is not clear if this is result of the data generation or the strength of the models used in data generation. The authors can show their point better if they used the same methods for data generation and inference; e.g., can a (single or) multi-model data generation approach using one or both BANJO and ARACNE improve the results, compared to when they are used individually on the original data? | Thank you for your comment and suggestion. Since ARACNE & BANJO are specifically designed for network inference rather than data generation methods, the available software for both cannot be used for data generation purposes. It is generally the case that data generation techniques are different than network inference algorithms. For example, the study that ARACNE algorithm is presented [1] uses Mendes models [2] for synthetic data generation which employs Hill kinetics to approximate transcriptional relationships. The study that BANJO algorithm is presented [3] uses Boolean glass gene model for generating synthetic gene expression data [4]. The study of the synthetic gene expression data generation tool GeneNetWeaver (GNW) [5], on the other hand, employs stochastic and deterministic differential equations to generate synthetic gene expression samples, and measures the quality of the generated samples based on several network inference algorithms including ARACNE and GENIE3, etc [1,6]. Hence, we observe that generally network inference algorithms are different than data generation methods, and not used for data generation purposes. Hence, we designed our Multi-Model framework as a data generation framework, and evaluated it with respect to the two well-known network inference algorithms, ARACNE & BANJO.  Another reason for using different methods for data generation and network inference is to make a fair evaluation of the generated samples. That is, we want to measure the quality of the generated data with respect to unseen method during the data generation. This is similar to using separate samples for training and testing purposes. We use different methods for generating samples, and different methods for measuring the quality of the generated samples. Otherwise, it may lead to over-fit the network since the inference algorithm would already find its required system dynamics in the generated data.  However, your point that the successful result of GRN inference is due to whether the success of our multi-model data generation framework, or the strength of the models is a very true point. In order to clarify this, we have compared the inferred network from the data generated by the Multi-Model framework, with the inferred networks from the data generated by each single model separately. We have seen that, though the data generated by each single model cannot predict more accurate networks than the original data, the data generated by the Multi-Model framework can.  In Section V-D of the old version “Multi-Model Justification”, we had tried to evaluate the effect of combining different generative models. Figure 23 – 26\* show that the metric values, i.e., compatibility, diversity, and coverage values are multi-objectively better for the data generated by Multi-Model framework than for the data generated by each single generative model. However, it is true that, though metric values are higher for the Multi-Model framework, and Multi-Model framework produces data predicting more accurate networks, it may still be the case that each single model could have improved the network inferred from the original data, as well. However, as we explained in the previous paragraph, we hope it is now clear that combining gene expression samples from different generative models has a significant effect on the quality of the generated samples. The qualities of the generated samples from the individual generative models are not good enough to predict networks better than the one predicted by the original data. However, combining generative models multi-objectively increases the quality of the generated data in such a way that we can predict better networks than the original data predicts. | We put Section V-D of the old version “Gene Regulatory Network Inference” in front of Section V-C of the old version “Multi-Model Justification” for better logical flow.  The new experiments on Multi-Model justification over GRN inference is put at the end of the Section V-D. Figures 27 – 30, and Table III are newly inserted. |
| 2 | It appears the authors use generated data separately in their evaluation. Why not use generated data \*with\* the original data, and how would that fare? | When we use the original data with the generated data, we get worse results than we get only by the generated data. When we combine the original and generated samples used in Section V-C, and feed them to the ARACNE network inference algorithm, we get 0.4324 precision, 0.4211 recall, and 0.4267 f-measure values for the inferred network. The network is shown in Figure 1\*\* in Appendix below. There are 16 correct edges over all proposed 37 edges. There were 24 correct edges out of 49 edges proposed by the network inferred from only the *generated* data; and there were 3 correct edges in the all 14 edges for the network inferred from only the *original* data. For the sake of readability the results are reported in Table I in Appendix below. As it is seen, the combined original and generated data produces results between the original data and the generated data. This is an expected result, indeed. ARACNE employs Mutual Information (MI) over the expression patterns of each pair of genes to understand the statistical relationships between the genes. While these statistical relationships are strong in the generated data since the network inferred from the generated data has high precision, recall, and f-measure values, the relationships are weak in the original data since the network inferred from the original data has low precision, recall, and f-measure values. When we combine the two datasets, the relationships between the pair of genes are stronger than that in the original data, and weaker than that in the generated data. Note that ARACNE produces the same network when we append the original samples to the generated samples, and when we append the generated samples to the original samples.  When we feed the combined original and generated data to the BANJO network inference algorithm, this time we get worse result than both we get only by generated data, and we get only by original data. If we append generated samples to the original samples, the inferred network includes 9 total edges in which 3 of them are correct. The network is shown in Figure 2 in Appendix below. The precision value is 0.3333, recall value is 0.0577, and the f-measure value is 0.0984. If we append original samples to the generated samples, the inferred network includes 10 total edges in which 3 of them are correct. The network is shown in Figure 3 in Appendix below. The precision value is 0.3000, recall value is 0.0577, and the f-measure value is 0.0968. Precision, recall, and f-measure values are shown in Table II in the Appendix. As it is seen, unlike the ARACNE, the network inferred by the BANJO algorithm from the combined data of the original and generated samples is worse than both the network inferred only from original data, and the network inferred from only generated data. Again, this is an expected result, indeed. Unlike the ARACNE, BANJO infers a Dynamic Bayesian Network (DBN) from time-series gene expression data. By concatenating the original and generated datasets, we ruin the time series nature of the original dataset and that of generated dataset, resulting in inferring worse networks. Note that, unlike the ARACNE, the network inferred from the original data appended to the generated data is different than the network inferred from the generated data appended to the original data. This is consistent with our interpretation that time-series nature of the data affects the inferred network by BANJO. Since the order of combination of the samples affects the time-series characteristics of the data, the resulting inferred networks are different. | No action taken. |
| 3 | Most of the analysis focuses on varying number of original or generated samples. e.g., it is suggested that 240 samples is sufficient for original experiments. However, there must be a strong connection between number of genes and the number of samples. It is not clear how the results depend on varying number of genes. It appears that the application of the approach is focused on cases where one knows the genes involved in a select number of (sub)networks. Some guidance or discussion is needed when one is analyzing a dataset containing tens of thousands of genes. | Thank you for your careful comment. The reason for us not to investigate the relation between the number of required samples and the number of genes is that PBN construction algorithm does not scale well for large number of genes. For example, building a PBN from a 300-gene data set takes almost 30 hours. Since we are generating PBNs again and again for each of the training sample sets, it takes significant amount of time to test the number of required training samples for a regulation system of large number of genes. For instance, building a PBN would roughly take 1000 hours (~17 days) for a 1000-gene data. Motivated by your comment, we have decided to investigate the relation between the number of genes and the number of training samples by excluding the PBN from our multi-model framework (using only ODE, HIMM, and Multi-objective genetic algorithm). We have done experiments with 25, 40, 60, 80, 100, 200, 300, 400, 500, 1000, 2000, 5000, and 10000 genes. We see that as the number of genes in the data set increases, the number of required samples to train our multi-model framework increases, as well. Moreover, depending on the desirable precision on the quality of the generated data sets from our multi-model framework, the number of required training samples can be chosen differently.  Note that, after 2000 genes, ODE could not scale, hence we exclude ODE in experiments with 5000 and 10000 genes. After 10000 genes, none of the computational models we employed could scale. Therefore, we could not provide results for several tens of thousands of genes as you recommended due to scalability problem of the computational models we are using. However, we believe the available results up to 10000 genes provide considerable insight on the relation between the number of genes in the training data set and the number of required samples to train our multi-model data generation framework.  Note also that, in our previously done experiments on real life HUVECs dataset composed of 379 genes, we had used PBN in the evaluation of the number of required training samples. When we compare the results of HUVECs dataset and the results of the synthetic datasets including 300-400 genes, we observe that the quality of the generated samples from HUVECs converges faster. Hence, we conclude that exclusion of PBN from the multi-model framework affect the synthetic dataset experiments in a way that they converge slower than they would with PBN. Therefore, another fact revealed from our experiments for this action item is that depending on the computational models used in our multi-model framework, the quality of the generated sample sets, and the number of required samples to train our framework may change, as well. | The experiments in Section V-E are repeated with varying number of genes. Section V-E is extended in accordance with the author reply. Figure 31 is newly inserted. |
| 4 | In t-test results, it would be more meaningful to present "number of genes with less than the (0.05) p-value threshold", rather than the average p-value of all genes. Data generation should not introduce significantly differentially expressed genes, which can be captured by such a "count". | Thank you for your enlightening comment. You are right that, though the average of the p-values of all the genes is greater than 0.05, there may be many genes having less p-value than 0.05. In addition to the average p-value results, we have put the number of genes having less p-value than 0.05. For melanoma and yeast data sets, we have seen the number of genes having p-value less than 0.05 is not significant. For HUVECs data set, we have seen that, on the average almost half of the genes have larger p-values than 0.05, which we believe is due to the high number of genes, 379, and the high ratio between the number of genes and the number of samples in HUVECs data set. The ratio is 7/31 = 0.2258 for melanoma, 25/77 = 0.3247 for yeast, and 379/400 = 0.9475 for HUVECs. We observe that it becomes hard to simulate large number of genes if the ratio between the number of genes and the number of samples is high, as well. | Section V-A and V-B is extended with respect to the newly added experiments on the number of genes with p-value less than 0.05. Figures 7, 8, and 17 are newly inserted. |
| 5 | In some figures (e.g., Fig 5, 6), there is a strong fluctuating behavior for the melanoma dataset as more samples are generated. What is this attributed to? Perhaps, showing a PCA plot of the original and generated data would help describe this. e.g., it may be due to the data forming two distinct clusters and each new generated sample alternating between these two clusters. | Thank you for your enlightening comment. We have checked the PCA plot of the generated sample sets from the melanoma data set. When we overlay the PCA plot of the generated sample sets, interestingly they continuously form a spiral. Once, we analyze observed spiral and the fluctuations, we see that as the coefficients of the 1st principal component increase, the results of melanoma in Figure 5 and 6 increase, as well. Similarly, as the coefficients of the 1st principal component decreases, the results of melanoma in Figure 5 and 6 decrease, as well. Moreover, the sample sets produced from neither yeast nor HUVECs datasets have such a specific pattern in their PCA plots. Hence, we believe the reason for fluctuations in the results of the melanoma dataset is the spiral structure of PCA plot of the generated sample sets.  We have also checked PCA plot of the original melanoma, yeast and HUVECs datasets. However, we could not get any specific pattern in the original datasets. Figure 4, 5, 6 in Appendix below show the PCA plot of melanoma, yeast, and HUVECs datasets, respectively. | We extended Section V-A by adding the PCA plots of the generated datasets from melanoma, and their explanations. Figures 9 – 13 are newly inserted. |
| 6 | \* similar genes in general -> genes in common \* hypothetic[+al] \* training[=,] set \* highly good -> awkward \* below [-to] \* but flows -> varies | We have done all the suggested minor corrections and suggestions. | Corrections are done in accordance with author reply. |
| 7 | \* fortunately a perfect result -> too strong and informal | We have relaxed our claim as “reasonably successful result”. | Corrections are done in accordance with author reply. |
| 8 | \* Their sample selection mechanism are unsatisfiable -> I couldn't tell what is meant by "unsatisfiable" | We have changed the sentence as “Their sample selection mechanism is weak”. | Corrections are done in accordance with author reply. |
| 9 | \* Our main observation is that although there are several gene regulation models in the literature, none of them is able to perfectly capture the system dynamics of the gene regulation mechanism. -> No evidence is given for this claim. | We have removed the referred sentence. | Corrections are done in accordance with author reply. |
| 10 | \* is about %3.2 on average -> Does this have to be %1 | The average diversity value is 1.032 for original datasets ([1.06+1.005+1.03]/3 = 1.032), implying %3.2 information contribution. We have corrected the confusing explanations. | Corrections are done in accordance with author reply. |

*\* Throughout the letter, the section and figure numbers are referred to our revised version of the paper, unless it is specified as in old version.*

*\*\* Throughout the letter, all figures and tables included in the Appendix of this letter are specified as “in Appendix below”.*

*\*\*\* In the revised manuscript, all textual insertions are marked with red, deleted ones with green.*

**References**

[1] A. Margolin, I. Nemenman, K. 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, pp. S7+, 2006.

[2] P. Mendes, W. Sha, and K. Ye, “Artificial gene networks for objective comparison of analysis algorithms”. Bioinformatics 2003,

vol. 19, no Suppl 1, pp. 122- pp. 129.

[3] A. Bernard and A. Hartemink, “Informative structure priors: Joint learning of dynamic regulatory networks from multiple types of data,” in Pacific Symposium on Biocomputing 2005 (PSB05), A. R., D. A.K., H. L., J. T., and K. T., Eds. World Scientific: New Jersey, 2005.

[4] R. Edwards and L. Glass, “Combinatorial explosion in model gene networks,” Chaos 10, 691–704 (2000).

[5] T. Schaffter, D. Marbach, and D. Floreano, “GeneNetWeaver: In silico benchmark generation and performance profiling of network inference methods,” Bioinformatics, vol. 27, no. 16, pp. 2263–2270, 2011.

[6] V. A. Huynh-Thu, A. Irrthum, L. Wehenkel, and P. Geurts, “Inferring regulatory networks from expression data using tree-based methods”, PLoS ONE, 5(9):e12776, 2010.

**Appendix**

|  |  |  |  |
| --- | --- | --- | --- |
| Data Set | Precision | Recall | F-measure |
| Original | 0.2143 | 0.0789 | 0.1161 |
| Generated | 0.4898 | 0.6316 | 0.5517 |
| Original + Generated | 0.4324 | 0.4211 | 0.4267 |
| Generated + Original | 0.4324 | 0.4211 | 0.4267 |

Table I: Network evaluation for ARACNE algorithm

|  |  |  |  |
| --- | --- | --- | --- |
| Data Set | Precision | Recall | F-measure |
| Original | 0.3846 | 0.0962 | 0.1538 |
| Generated | 0.4211 | 0.1538 | 0.2253 |
| Original + Generated | 0.3333 | 0.0577 | 0.0984 |
| Generated + Original | 0.3000 | 0.0577 | 0.0968 |

Table II: Network evaluation for BANJO algorithm

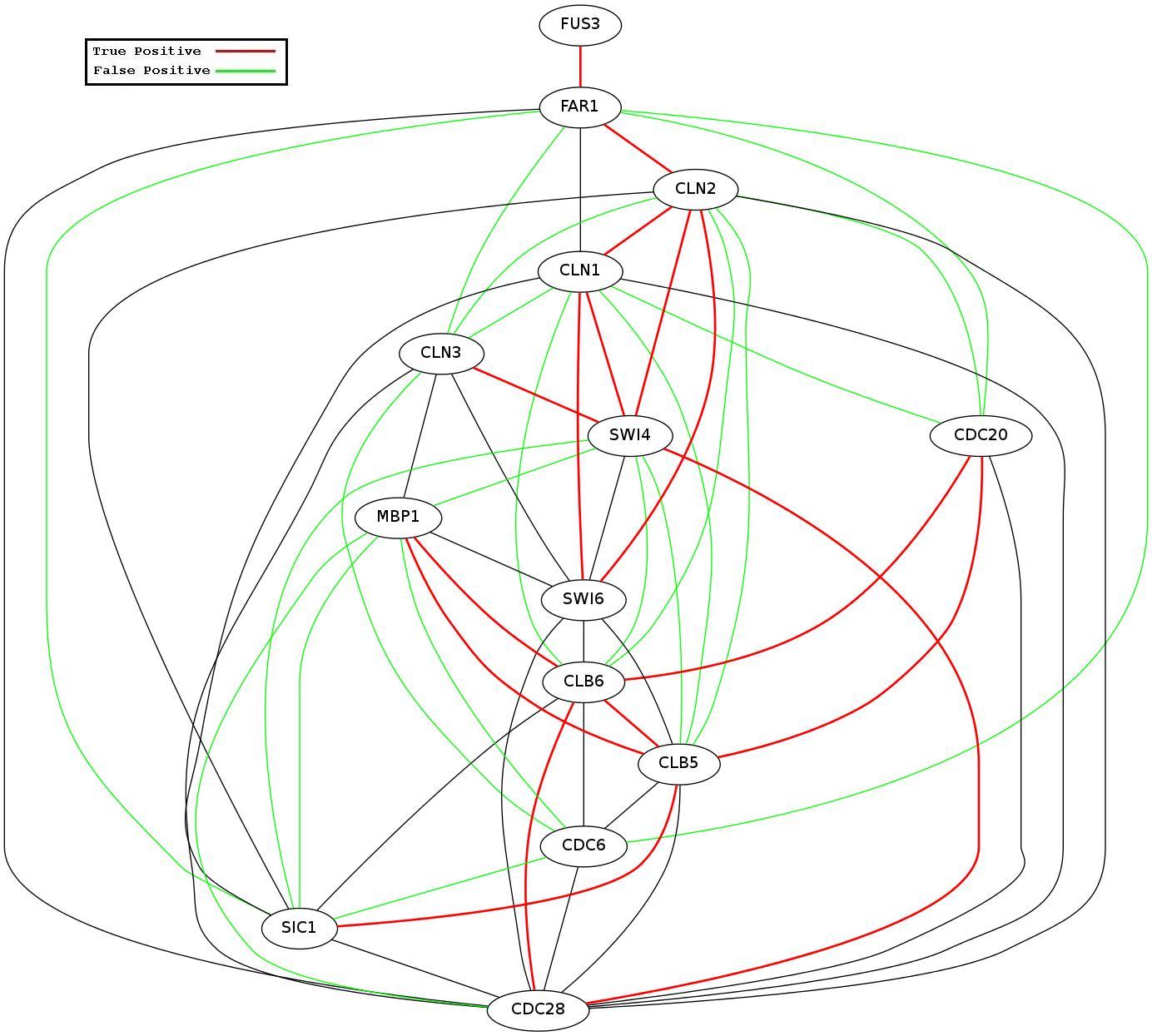


Figure 1: The regulatory network obtained from original and generated samples using ARACNE

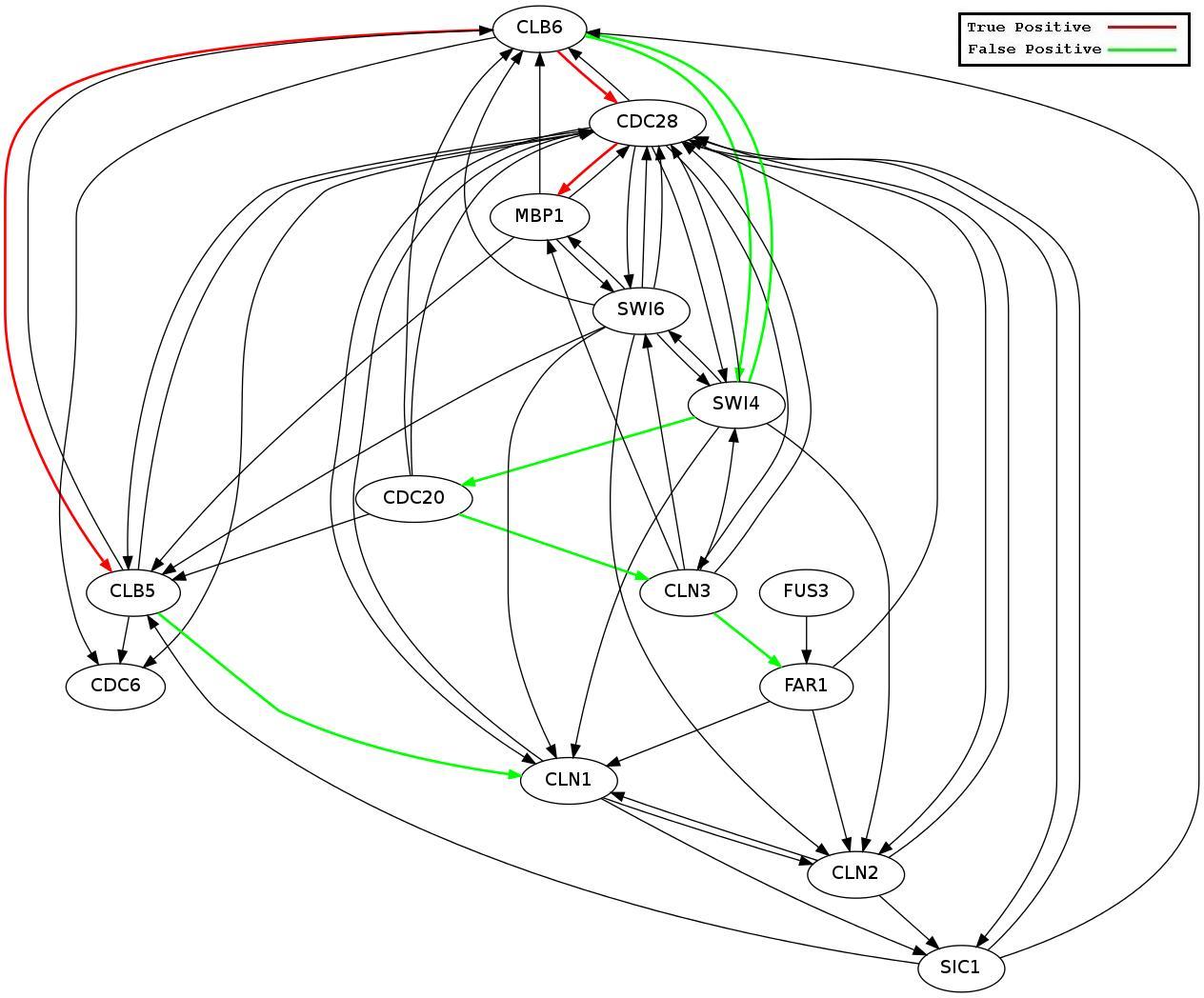


Figure 2: The regulatory network obtained from *original and generated* samples (in order) using BANJO

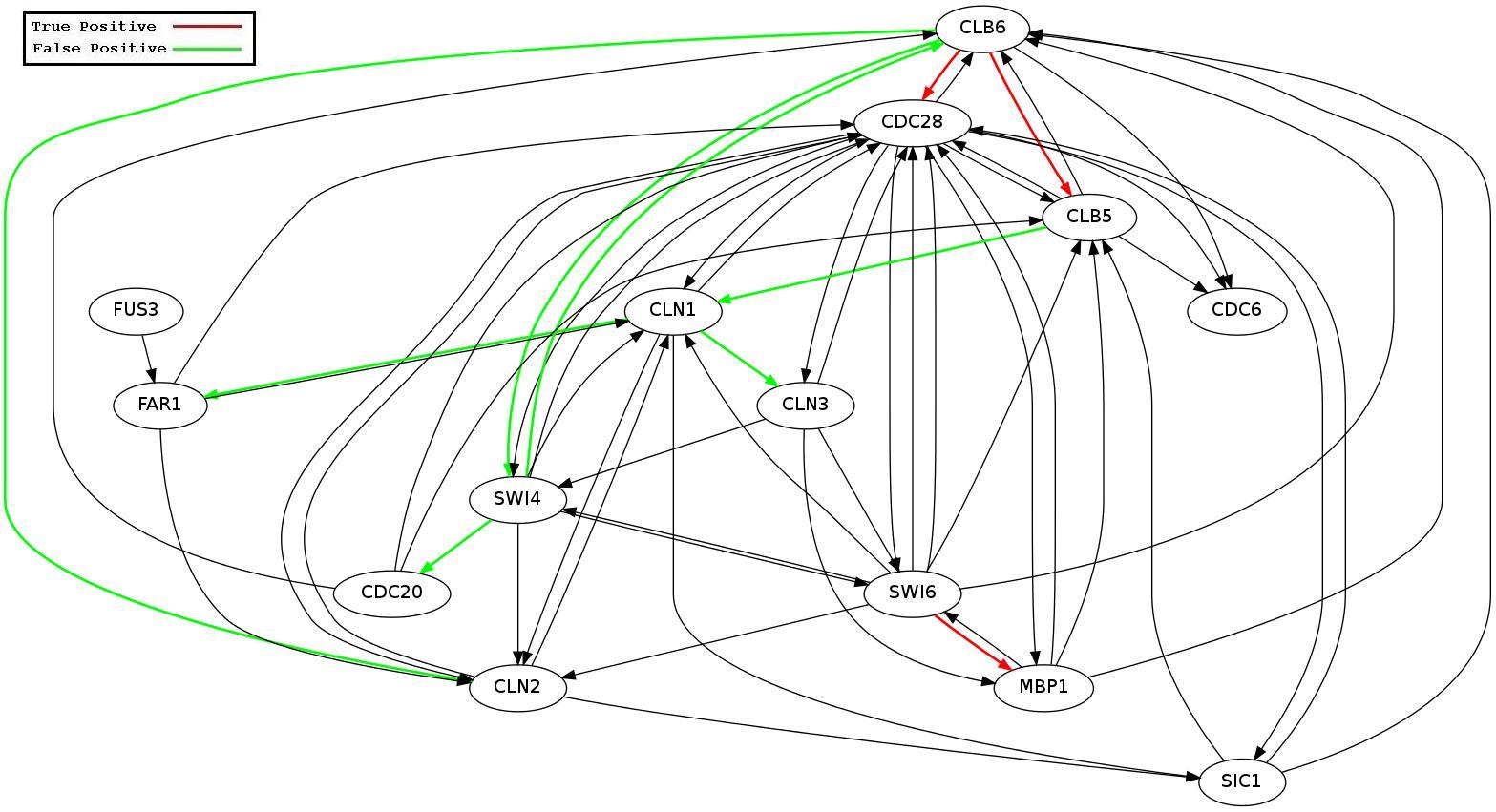


Figure 3: The regulatory network obtained from *generated and original* samples (in order) using BANJO

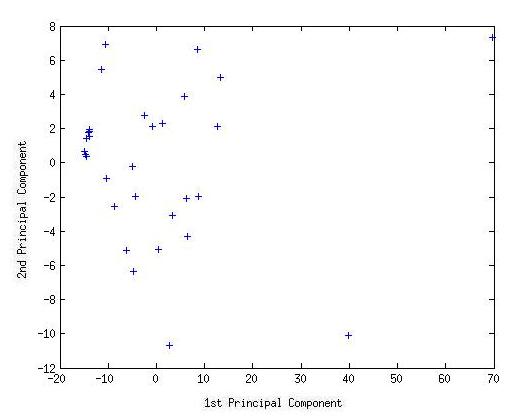


Figure 4: PCA plot of real life melanoma dataset

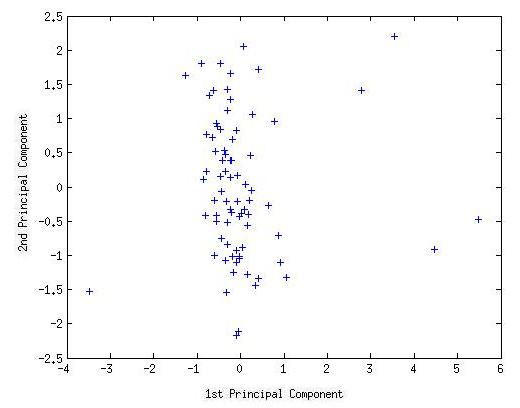


Figure 5: PCA plot of real life yeast dataset

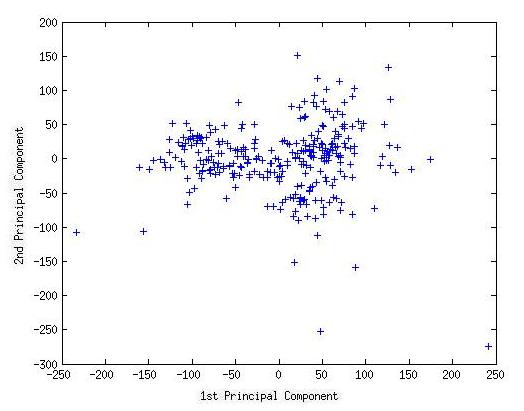


Figure 6: PCA plot of real life HUVECs dataset