Elucidating Transcriptional Regulatory Networks from heterogeneous gene-expression compendia

Andrea Pinna[1], Nicola Soranzo[1], Vincenzo De Leo[1,2], Alberto de la Fuente[1]

[1]CRS4 Bioinformatica, Loc. Piscina Manna, 09010 Pula (CA), Italy
[2]Linkalab, Complex Systems Computational Laboratory, 09100 Cagliari (CA), Italy

1) Overview: Transcriptional Regulatory Network inference

The goal of the DREAM5 Network Inference Challenge is to reverse engineer Transcriptional Regulatory Networks from gene expression datasets, i.e. to identify direct targets of transcription factors. With the development of mRNA measurement techniques, a constantly increasing amount of data is available to researchers, so methods and algorithms are needed for analyzing such huge and heterogeneous compendia.

2) DREAM5: Network Inference challenge

- Four microarray compendia were available to the participants. One is based on simulated data, while the remaining are obtained from *S. aureus*, *E. coli*, and *S. cerevisiae* (microorganisms revealed after the submission deadline).
- Each compendium contains gene expression microarrays (either steady states or time series) obtained from hundreds of experiments performed in different laboratories and with several combinations of genetic (knockout, knockdown, over-expression), drug and environmental perturbations.
- A list of transcription factors was provided too, and participants were requested to rank each possible link between transcription factors and target genes.

3) Approaches and Techniques

In order to exploit the **different types of information** included in these mixed datasets, we applied three distinct approaches to **subsets of data**, and then combined the results in a final prediction.

4) Approach 1: Perturbation-response analysis

Applied to: **wild-type** and **TF knockout** or **over-expression** experiments. We used steady state data, but also the time series from which we considered the final time-point.

Goal: identify the potential targets of a strongly perturbed TF_i.

Method: we calculated for all potential target genes the normalized deviation (superscript indicates the perturbation):

$$R_{TF_i \to T_j} = \frac{T_j^{TF_i} - T_j^{WT}}{T_j^{WT}}$$

In addition we employed double knockout or over-expression: to gain more confidence in $TF_i \to T_j$ we calculated (when possible):

$$R_{TF_i \to T_j} = \frac{T_j^{TF_i,TF_k} - T_j^{WT}}{T_j^{WT}} - \frac{T_j^{TF_k} - T_j^{WT}}{T_j^{WT}}$$

The double knockout or over-expression can have extra information for $TF_i \to T_j$ however the response to the double perturbation might be explained by the effect of TF_k, so this must be subtracted. In some cases TF_k was not perturbed and then we simply used:

$$R_{TF_i \to T_j} = \frac{T_j^{TF_i,TF_k} - T_j^{WT}}{T_j^{WT}}$$

(large values might be due to the TF_k perturbation, but we accept making some mistakes as a trade off for also identifying real targets of TF_i).
In case of triple knockouts, we proceeded in a similar way.

S_{ij}^1 (confidence in $TF_i \to T_j$) was obtained by averaging the $R_{TF_i \to T_j}$ values from single, double and triple knockout and/or over-expression experiments.

5) Approach 2: Full-order partial correlation analysis of non time-series data subset

Applied to: **steady state data** (i.e. removed the time series).

Goal: exploit the overall correlation in expression between a TF_i and its targets.

Method: we applied full order partial correlation using the GeneNet R package, available for download at: `http://strimmerlab.org/software/genenet`.

S_{ij}^2 (confidence in $TF_i \to T_j$) is the absolute value of partial correlations ω_{TF_i,T_j}.

6) Approach 3: Co-deviation analysis of aspecific perturbation data

Applied to: chips with **drug perturbations**, and chips featuring **non-TF single gene perturbations**.

Goal: check if target T_j consistently makes large deviations in expression when TF_i makes large deviations.

Method: we converted the gene expression values into z-scores.
Then, for each transcription factor TF_i:

1. We split the data into two subsets: one group of observations D_i^H with $z_i > d$ (TF_i is *high*) and another group of observations D_i^L with $z_i < -d$ (TF_i is *low*); we used $d = 0.5$.
2. To identify the potential targets T_j of TF_i, we performed for each T_j a two-sided t-test to check whether its mean in D_i^H is significantly different from its mean in D_i^L.

S_{ij}^3 (confidence in $TF_i \to T_j$) is the absolute value of the t-statistic t_{ij} (test performed for T_j when datasets are formed based on deviation of TF_i).

7) Combining the approaches

We combined the approaches through a **weighted mean**. When approach 1 is applicable (i.e. if at least one experiment where the only perturbation is the knockout and/or over-expression of TF_i is available), then the overall ranking is:

$$S_{ij} = a \cdot S_{ij}^1 + b \cdot S_{ij}^2 + (1 - a - b) \cdot S_{ij}^3$$

Otherwise:

$$S_{ij} = c \cdot S_{ij}^2 + (1 - c) \cdot S_{ij}^3$$

The scores of the individual approaches were replaced by their ranks, so that S_{ij} is a weighted average of the ranks. The values of the weights ($a = 0.4$, $b = 0.5$, $c = 0.8$) were obtained through an optimization process on **simulated data** generated by our in-house simulator, which we adapted to perform simulations of the experiments considered in this challenge.

8) Simulation and method evaluations

We employed SysGenSIM, our in-house simulation toolbox, to generate the gene regulatory networks where DREAM5-like experiments have been performed and gene-expression values have been simulated. Several inference algorithms and methods have been evaluated by calculation of AUC(ROC) and AUC(PvsR).

Network generation: large Gene Networks were generated in order to represent real modular networks, i.e. showing exponential in- and power law out-degree distributions. Networks with different average degrees have been considered too.

Model equation:

$$\frac{\mathrm{d}G_i}{\mathrm{d}t} = \zeta_i \cdot \tau_i \cdot \delta_i^{syn} \cdot \theta_i^{syn} \cdot V_i \cdot \prod_{j \in \mathbf{R_i}} \left(1 + A_{ji} \frac{G_j^{h_{ji}}}{G_j^{h_{ji}} + (K_{ji}/\pi_j)^{h_{ji}}} \right) - \delta_i^{deg} \cdot \theta_i^{deg} \cdot d_i \cdot G_i$$

Dataset parameters: $\mathbf{R_i}$ is a set containing the indexes of all regulators (both activators and inhibitors) j of gene i; G_i is the mRNA concentration (gene activity) of gene i, V_i is its basal transcription rate, while d_i is its degradation rate constant. K_{ji} is the interaction strength of G_j on G_i, h_{ji} is a cooperativity coefficient, and A_{ji} is an element of the matrix A encoding the signed network structure. For simplicity, parameters V_i, K_{ji}, and d_i were kept constant, and set equal to 1. Cooperativity coefficients h_{ji} are set to 1, 2, or 4 with probabilities 60%, 30%, and 10%, respectively. The elements of the adjacency matrix are $A_{ji} = 1$ when gene j is an activator for gene i, or is $A_{ji} = -1$ when gene j is an inhibitor for gene i; $A_{ji} = 0$ otherwise, i.e. when a direct regulation from gene j to gene i does not exist. Once these parameters have been set, they are used to simulate all the chips of the dataset.

Experiment parameters: biological variances θ^{syn} and θ^{deg} are both sampled from the Gaussian distribution $N(1, 0.1)$, and are constant for the chips belonging to the same experiment.

Chip parameters: ζ, π, δ^{syn}, δ^{deg} and τ are generated for the simulation of each single chip. Parameter ζ is different from 1 in case of knockout and/or over-expression (i.e., $\zeta_i = 0$ if gene i is knocked-out, and $\zeta_j = 5$ if gene j is over-expressed), while π enables for drug perturbations of about 10% of targets, and is sampled from $N(1, 0.4)$; δ^{syn} and δ^{deg}, sampled from $N(1, 0.025)$, represent stochasticity in both transcription and degradation rates; τ, sampled from $N(1, 0.1)$, gives variability in time series simulations. Then, measurement noise ϵ is added to the gene-expression values.

Experiments: similarly to DREAM5 data, we simulated gene knockouts and over-expressions, and mimicked multi-factorial and drug perturbations, to obtain either steady states and time series gene expression measurements similar to those provided by the challenge organizers, to be used as test-benches for inference algorithms.

9) Results

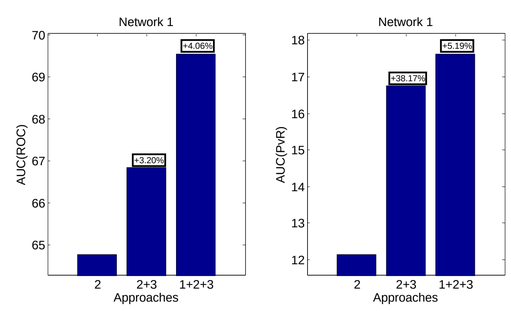
The submitted approach has achieved different performances, depending on which networks are considered for the score:

- Network Inference challenge (Networks 1, 3 and 4): 10$^{\text{th}}$ position
- Real Networks sub-challenge (Networks 3 and 4): 2$^{\text{nd}}$ position
- Yeast Network sub-challenge (Network 4): 1$^{\text{st}}$ position

Therefore, our algorithm proved to be efficient in reverse engineering transcriptional regulatory networks from real expression compendia.

The submitted predictions consist of the combination of three approaches. These figures show how combining different methods improves the performances, i.e. the AUC(ROC) and the AUC(PvR), for predictions given by:

- the application of Approach 2 by itself
- the combination of Approaches 2 and 3
- the combination of Approaches 1, 2 and 3



These results reveal that the combination of different inference techniques is indeed useful, especially with heterogeneous compendia, provided that these data are correctly subdivided. Moreover, even if gene expression simulations are more and more realistic, it clearly emerges that inference methods having good performances on synthetic datasets can not yet be expected to obtain the same results on real microarrays.

This work is (also) supported by the operating program of Regione Sardegna (European Social Fund 2007-2013), L.R.7/2007, *Promotion of scientific research and technological innovation in Sardinia*.

For further questions please contact Alberto de la Fuente: `alf@crs4.it`

`http://www.bioinformatica.crs4.it` `http://www.linkalab.it`