Integrative Network Analysis Discovers Causal Regulatory Network of System Level Measurements

Dongze He[1], Arda Durmaz[1], and Gurkan Bebek[1⋆]

Case Western Reserve University, Cleveland OH 44106, USA
{dongze.he,arda.durmaz2,gurkan.bebek}@case.edu

Abstract. Causal regulatory network analysis explains observed gene expression changes under a given experiment. Recent approaches use pre-established knowledge to extract regulatory network which is consistent with differentially expressed genes (DEGs). However, the identification of DEGs may exclude functionally relevant regulatory gene sets due to technical artifacts or transient biological mechanisms leading to inconsistencies in regulatory network prediction. We use system level expression measurements to provide efficient and effective regulatory annotation and analysis for genomics and proteomics studies.

We report a new causal network analysis algorithm, including network-searching and local-reviewing, to infer the activity of genes or proteins in a given network and predict a causal pattern of events for the provided study. Given a directed protein-protein interaction network, we infer plausible states for genes and build causal regulatory networks for the study. We assess the predicted upstream regulators on the gene level and regulatory paths on the sample and phenotype level. Finally, we report the regulatory networks with a predicted state for each gene and path.

Our algorithm improves the false discovery rate that is prevalent in causal regulatory network analysis. Moreover, our method can predict active states for genes and paths for an individual sample, providing flexibility for downstream analysis.

Supplementary Information: Supplementary methods, figures, tables, and code are available at https://github.com/bebeklab/CRNA.

Keywords: Gene Activity · Causal Regulatory Network · Upstream Regulation.

1 Methods

1.1 Building a Knowledgebase

Protein-Protein Interaction Network: Human protein-protein interaction (PPI) data with confidence scores were obtained from STRING [**?**]. Directed and weighted *Homo sapiens* PPI data (1=stimulation, -1=inhibition) are downloaded from OmniPath database [**?**]. Gene symbol annotations are queried using biomaRt package in R. STRING and OmniPath are merged to obtain a directed PPI network with edge weight and sign. We collapse duplicated edges between genes and used their mean weight and the majority vote of signs as the fused edges weight and sign. We used this PPI network as the knowledgebase to test and develop our method. It contains 2,817 genes/proteins and 12,193 edges representing functional annotations between these genes/proteins.

⋆ Correspondence should be addressed to gurkan.bebek@case.edu.

Expression Data We use 17 beta-estradiol (E2) treated MCF-7 cell line expression data (called the E2 study in result section) where MCF-7 cells were treated by E2 after they were cultured 12 hours, 24 hours and 48 hours (retrieved from NCBI Gene Expression Omnibus (GEO) series GSE11352). We also investigated EGF treated HLF cell line expression data (called the EGF study in the result section, GEO series GSE60800). Pre-processing procedure and differential expression analysis are performed in R/Bioconductor. Genes with p-value ≤ 0.05 and absolute value of fold change ($|FC| \geq 2$ are regarded as significantly differential expressed). In the E2 study, we get 200 genes that are differentially expressed when comparing nine treated samples with nine control samples, In EGF study, we get 51 DEGs when comparing 23 control samples with 24 treated samples.

1.2 Inferring Causal Networks through Random Walks with Restarts

Random walk with restart is defined as

$$p^{t+1} = (1 - r)Wp^t + rp^0$$

where p^0 is the uniform probability vector for a given seed set to $1/V_{seed}$ for each seed gene based on the DEGs, W is the column stochastic adjacency matrix providing the transition probabilities based on the knowledgebase (the input network) and p^t is the probability vector at time t initially set to $1/V$ for all nodes in the network. At each iteration, the RWR algorithm either jumps to one of the neighboring nodes or restarts at one of the seed nodes. To compute the steady-state probability vector we have applied power iteration until the difference between p^t and p^{t+1} is no greater than $epsilon = 1E - 10$. These probabilities provide a distance measure of each gene to the seed genes in our knowledgebase.

Using DEGs as seeds to apply RWR over our knowledgebase, we create a candidate regulatory network that is closely related to seeds. By starting from the seed nodes, RWR visits connected nodes according to edges′ weights, the larger an edge weight is, the higher possibility the edge is traversed. After every visit, RWR decides whether restarting from start point at next visit according to the restart probability (if restart probability is 1, RWR returns to start point after every visit). After setting times of visit, RWR reports the probabilities for each node that it visited. As biological findings are always bi-directed, which means if gene A regulates gene B, gene B sometimes is also reported to regulate gene A, we choose to perform an undirected RWR to select potential upstream regulators.

Similar to IPA process, RWR lacks background filtering, and many of the returned significant nodes should be DEGs since DEGs are supposed to be altered under the experimental setting. After testing, we set restart probability as 75%. Average of 1000 RWR probabilities are taken, and nodes within the top 2.5% are kept as upstream regulator candidates. Thresholds at 5% and 10% returned large networks that were impractical for downstream analysis and were not considered (See Supplementary Methods and Figure S1).

1.3 Filtering Upstream Regulators With Activities Predicted with System Level Measurements

Our framework is shown in Algorithm 1. First, we use depth-first search (DFS) to filter out genes that are only downstream of DEGs since we applied an undirected RWR. We set the DEGs in RWR regulatory network as the endpoint of DFS path. We then combine all DFS results as a directed regulatory network. Here we assume each gene r can take one of three state s, which are activated, inhibited and no change, represented by 1, -1 and 0.

Algorithm 1: The Causal Regulatory Network Algorithm based on random walk restart networks.

Result: Causal Regulatory Network
G: PPI knowledgebase;
D: DEGs list;
E: Expression measurements;
G_{RWR} = Random-walk-with-restarts(D,G);
for *all nodes* $r \in G_{RWR}$ *and* $r \notin D$ **do**
  $Net = Net \cup DFS$ between r and D;
end
DAG =remove cliques in Net;
$p(r, s)$=junction-tree(DAG);
$p'(r, s) = EM(p(r, s))$ // EM calculates maximal $p(r, s)$;
$x \in \{0, 1, -1\}$ //3 possible states for each gene ;
for *all nodes* $r \in DAG$ **do**
  $logL(p'(r, s))$ =$\log\frac{P(E|x_r=s,DAG)}{P(E|x_r\neq s,DAG)}$
end
if $logL(p'(r, 1)) > logL(p'(r, 0))$ *and* $logL(p'(r, 1)) > logL(p'(r, -1))$ **then**
  $logL(r) = logL(p'(r, 1))$;
else if $logL(p'(r, -1)) > logL(p'(r, 0))$ *and* $logL(p'(r, -1)) > logL(p'(r, 1))$ **then**
  $logL(r) = -logL(p'(r, -1))$;
else
  $logL(r) = 0$;
end
return $LogL(r)$

For each gene r in our candidate regulatory network, we then use expression measurements assisted with junction tree inference algorithm [**?** **?**] to infer the probability that a gene r is in a state s,which means for each gene r, we get three probabilities corresponding to the three possible states. A similar approach was introduced for finding gene activities for pathway enrichment analysis [**?**]. As we use a tree-based discovery algorithm to infer the probability, we remove all cliques in our candidate regulatory network to build a directed acyclic graph (DAG) regulatory network. We prefer to keep positive edges when removing the cliques. As genes' expression may not be measured, we then use the expectation-maximization algo-

rithm to maximize each probability [**?**]. Next, we calculate the log likelihood ($logL(r, s)$) to assess the activity index of genes, each gene r in each sample get a $logL(r, s)$ for each state s. We only assign activity index to genes that have consistent logL across three states. To be specific, we assign $logL(r, 1)$ to a gene if the gene's $logL(r, 1) \geq logL(r, 0) \geq logL(r, -1)$ or assign $-logL(r, -1)$ to a genes only if the gene's $logL(r, -1) \geq logL(r, 0) \geq logL(r, 1)$. Otherwise, the gene's activity index is regarded as 0. We record each gene's activity index for each sample into a matrix. As a result, we get an activity index matrix for each sample and gene, corresponding to the expression measurements. Our algorithms run-time is determined by the bottleneck junction-tree algorithm, which is exponential in the treewidth. Finding the junction tree with the smallest clusters is an NP-hard problem [**?**].

Significance assessment and evaluation of individual samples To assess the significance of activity indexes, we permuted 1000 samples for each real sample, and infer the activity index for each gene for each permuted sample. To be specific, for each real sample, we randomly select an expression measurement of a gene in the candidate regulatory network of the real sample, and arrange it to a random gene in the network of a permuted sample of this real sample, till each gene in the permuted sample gets a value. We then repeat 1000 times to get 1000 permuted samples for each real sample and do the same procedure for all real samples. We create a null distribution for each gene for each real sample using the gene's activity index distribution of 1000 permuted samples for the real sample. Considering that genes have different topological structures and expression conditions, we assign a gene in a real sample is in activated state(assign as 1) if the gene's activity index in the real sample is over 90 percentile of its null distribution, and is an inhibited state (assign as -1) if the gene's activity index in the real sample is under 10 percentile of its null distribution. Otherwise, we regard the state of the gene in the sample as not-changed (assign as 0).

Evaluation of phenotype or treatment comparisons As we get an activity state for each gene for each sample, we use the majority votes of a gene across all repeated samples in a phenotype group as the activity state of the gene in the phenotype group. When predicting gene's activity state across different treatments, we use the majority votes of a gene across all repeated samples in a treatment group as the activity state of the gene in the treatment group. When predicting the activity state of a gene in a regulatory network of comparison, for example, treatment group vs. control group, take the difference of the activity state of the gene as the predicted activity state of the gene in the regulatory network.

For each gene in our candidate regulatory network, we can use DFS for the gene in RWR regulatory network to build up a local regulatory network and assess each gene's activity state in the local regulatory network to get the informative regulatory network regulated by the gene.

1.4 Comparison with other methods

To assess our method, we compared our method to state-of-the-art methodologies.

A statistical approach to Causal Regulatory Network Analysis The CRNA of IPA is an excellent example of a statistically driven approach based on a well-curated knowledge-base. The upstream regulators are predicted without integration of the expression profiles in network discovery. The final mechanistic networks which could be queried for each potential upstream regulator are local regulatory networks from its regulatory network.To better assess our methods, we implemented the core algorithm as described in **?**] in R, including finding the overlapping p-value (Algorithm S1), activation z-score (Algorithm S2) and breadth-first search (BFS) (See Supplementary Methods for more details).

A network searching approach to Causal Regulatory Network Analysis We use CausalR package with the default setting to analyze the two datasets we used in the previous step and compared the results, using the same knowledge base we used in our method. CausalR offers CRNA by searching a directed protein-protein interaction network database as the reference network with depth-first search. CausalR uses DEG list as the endpoints to for DFS and finds the nodes which regulate most DEGs within the given path-depth. This is a fast and straightforward approach. DEGs regulated by each node are found and then ranked based on how many DEGs are regulated with the node within given path-length.

Regulatory Network Enrichment Analysis RNEA first selects differentially expressed transcriptional factors(DETFs) by using over-representation analysis and arrange them into their regulatory network. Then RNEA uses BFS in their knowledge base to find upstream regulators. To be specific, RNEA uses BFS for each DETF to find the regulated genes of the DETF within two depth. Then RNEA select potential upstream regulators according to its criteria. In RNEA, as long as a gene have significant expression p-value, the gene is regarded as differentially expressed. It means that RNEA is trying to utilize more expression profiles rather than genes which have significant p-value and fold change. However, RNEA has high FDR and cannot predict the activity states of upstream regulators.