Algorithms for Big Data Analysis in Biology and Medicine **Fall, 2017-8**

Lecture 10: December 26, 2017

Lecturer: Prof. Ron Shamir *Scribe: Dan Coster*

10.1 Biological Networks

10.1.1 Preface - The Challenge of Cancer

Two of the key challenges of cancer are understanding its biological process and having the ability to differentiate between subtypes of it. Individuals have different genetic profiles and hence tend to react in different ways to cancer treatments. For example, the gene ERBB2 underlies amplification and over-expression in 25% of the breast cancer patients, but only less than 50% of them show an improvement due to treatment of Trastuzumab (a monoclonal antibody used to treat breast cancer for patients with ERBB2 amplification).

10.1.2 Biological Background

Genome Variation in Cacner

The genome of cacer cells undergoes changes of two types: The first, is first numerical changes which are duplication or deletion event that affect a large chromosome segment (or even of the whole chromosome). It can be visualized via copy number (CN) profile, a graph whose X axis presents the chromosomes sorted by their sizes, and the Y axis presents the number of copies of each chromosomal segment. The second type is a structural rearrangement, which changes the order of the segments in a chromosome or even exchanges of segments between different chromosomes. Both types can be observed experimentally (e.g. by cytogenetic

techniques or deep sequencing). Such techniques can identify and evaluate the size, shape, and number of chromosomes in a sample.

Biological Networks[1]

An ultimate goal of a molecular biologist is to reveal fundamental cellular processes, and understand their impact on complex organisms. In order to achieve this goal one has to study how complex systems of many genes and proteins function and interact. A biological network models such a system as a set of molecular components (such as genes, proteins and other molecules) and interactions between them that collectively carry out some cellular function. Below we show some examples of such networks.

Expression of the Gene proB: Figure 10.1 depicts the gene's expression and its role in catalyzing a specific chemical reaction in the cell. The proB gene is expressed and produces the gamma-glutamylkinase protein, which catalyzes a reaction that takes as inputs (substrats) glutamate and ATP and produces gamma-glutamyl-phosphate and ADP.

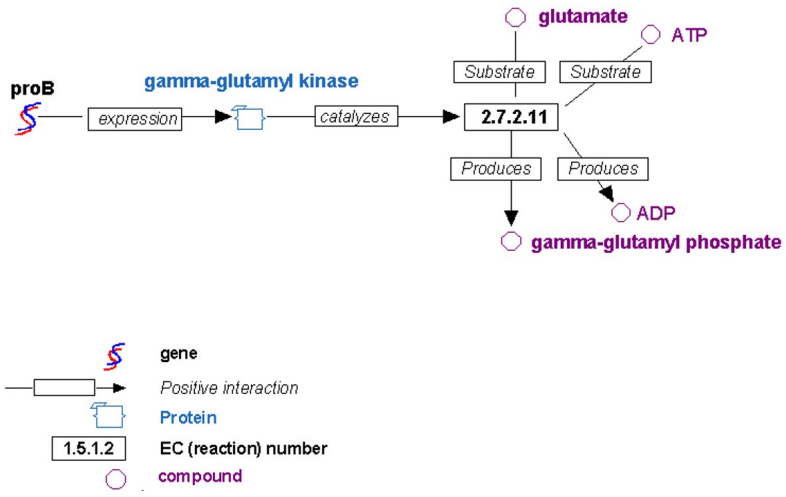

Figure 10.1: An example of the role of gene expression in catalyzing chemical reactions [4].

[1]based in part on a scribe by Karin Inbar and Anat Lev-Goldstein, March 2005

A Simple Metabolic Pathway - Proline Biosynthesis: The next example is part of a simple metabolic pathway, involving a chain of generated compounds, which is shown in Figure 10.2. One of the final products of the chain, the amino acid proline, inhibits the initial reaction that started the whole process. This negative feedback pattern is very common in biological networks, and serves to regulate the process execution rate.

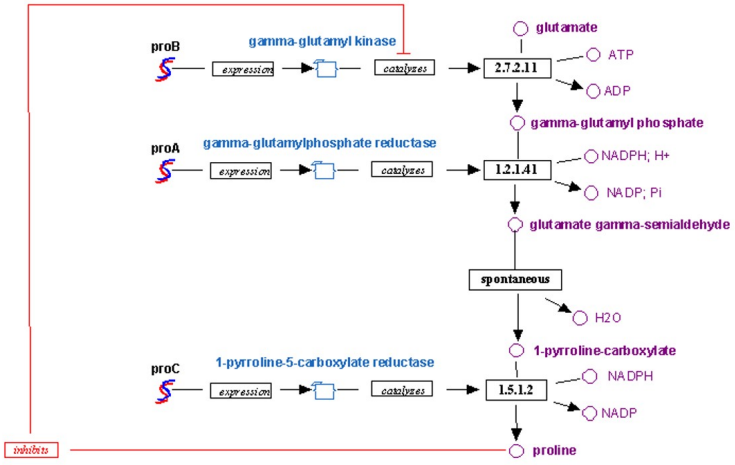

Figure 10.2: An example of a metabolic pathway: Proline biosynthesis [4].

Methionine Biosynthesis in E-coli: The next example is part of a more complex metabolic pathway, involving a chain of generated proteins (Figure 10.3). Hence a longer chain of reactions creates the amino acid methionine and multiple feedback steps are involved. As larger networks are harder to visualize, a simplified version with less details is shown in Figure 10.4.

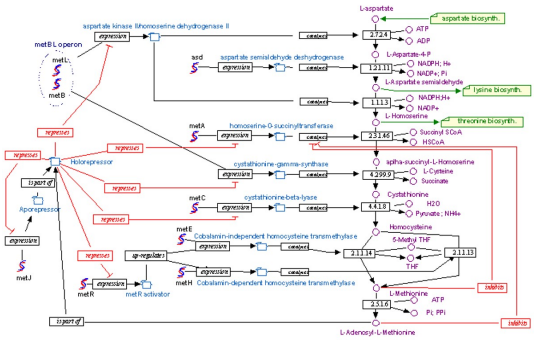

Figure 10.3: Methionine biosynthesis network in E-coli [4].

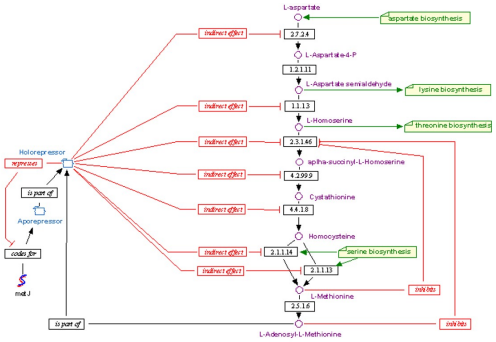

Figure 10.4: A simplified representation of the biosynthesis pathway presented in Figure 10.3.[4].

Signal Transduction Network: This example, depicted in Figure 10.5, is that of signal transduction - a cellular process whereby signals are relayed. By chain of interactions of proteins' complexes and small molecules, a process starting with molecule from outside the cell eventually affects gene expression inside the nucleus.

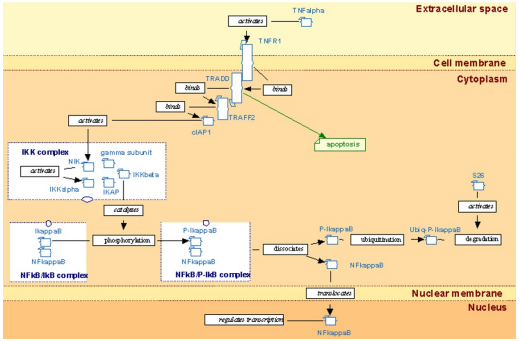

Figure 10.5: Methionine biosynthesis network in E-coli [4].

G1/S signaling pathway: This example, depicted in Figure 10.6, shows a specific signaling pathway that activates the transition between G1 phase to S phase as part of the cell cycle. The red dot in an element represents that more interactions involving the element are known but not shown, as they are not relevant to the specific G1/S pathways. The directed edges represent a known directed interaction (e.g. TF-target). Undirected edges represent two elements that interact.

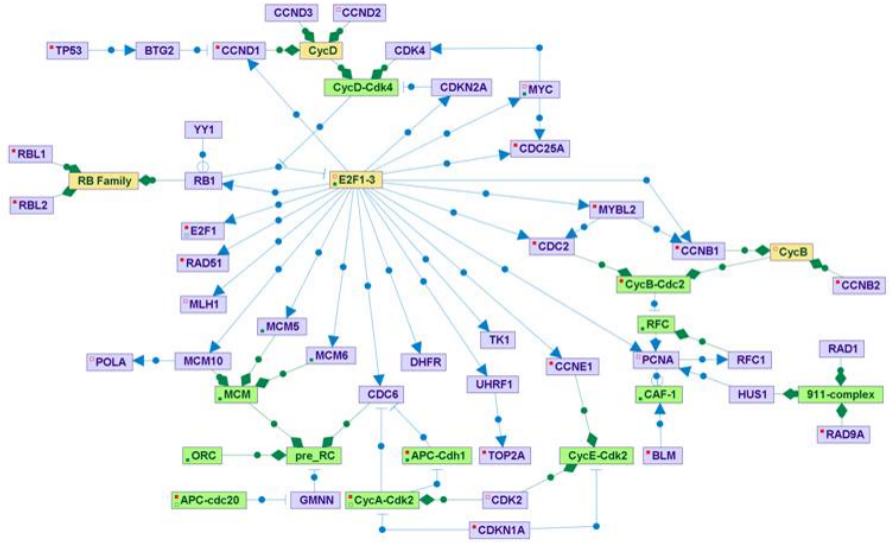

Figure 10.6: A signaling pathway of G1/S transition [3].

Sea urchin endomesoderm development: The following example, depicted in Figure 10.6, shows a genetic network controlling early development of the sea urchin endomesoderm. Here arrows to gene promotes indicate regulation of the gene's expression by a transcription factor. When multiple TFs regulate a gene, the logic of their regulation was also inferred but is not shown. This is one of the best understood and reverse engineered biological systems.

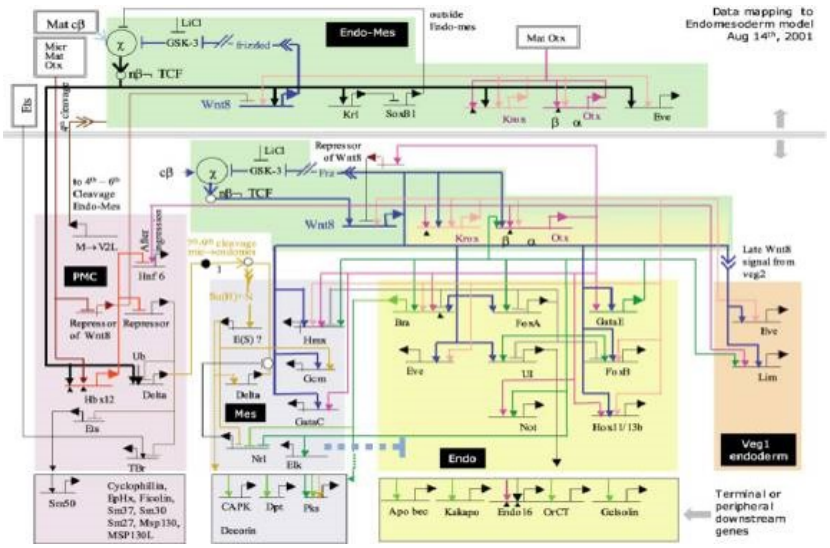

Figure 10.7: A genetic network that controls early development of sea urchin endomesoderm. [5].

10.1.3 Biological Network Resources

STRING: One of the biggest databases for protein-protein interactions is STRING, it contains data on more than 2000 organisms, more than 9 million proteins, and more than 1.3 billion interactions, some of the interactions contain confidence values [2].

10.2 Factor Graphs

10.2.1 Introduction

A factor graph is a bipartite graph representing the factorization of a function. It can be used to represent factorization of a probability distribution function (For an intuitive explanation, see the slides of Carl Edward Rasmussen [1]). Factor graphs will allow us to represent the product structure of a function and compute it efficiently (*Kschischang et al.* 2001 [6]).

Let $x_1 \in A_1, x_2 \in A_2, ..., x_n \in A_n$ be a set of variables. Consider the function $g(x_1, x_2, ..., x_n) : A_1 \times A_2 \times ... \times A_n \to \mathbb{R}$, such that $g(x_1, x_2, ..., x_n) = \prod_j f_j(X_j)$. $]X_j$ stands for the j^{th} subset of $\{x_1, x_2, ..., x_n\}$.$f_j \in F$ stands for the j^{th} **local function** which g factors into its product. $f_j(X_j)$ contains only parameters from X_j.
The graph contains two types of nodes:

- A variable node for each variable $\{x_1, x_2, ..., x_n\}$ (usually marked with $\bigcirc$)
- A factor node for each local function f (usually marked with $\blacksquare$).

The edges connect all the local functions to the variables that the local function depends on. Namely, they represent the dependency of factors on variables. Figure 10.8 shows a simple factor graph.

The **marginal function** represents the marginal of a variable x_k and is defined as:

$g_i(x_i) = \sum_{\tilde{x_i}} g(x_1, x_2, ..., x_n) = \sum_{x_1 \in A_1} \sum_{x_2 \in A_2} ... \sum_{x_{i-1} \in A_{i-1}} \sum_{x_{i+1} \in A_{i+1}} ... \sum_{x_n \in A_n} g(x_1, x_2, ..., x_n)$

$\tilde{x_i}$ means that the summation goes over all the variables, except x_i. When g represents the joint probability distribution, $g_i(x_i)$ is proportional to the marginal probability.

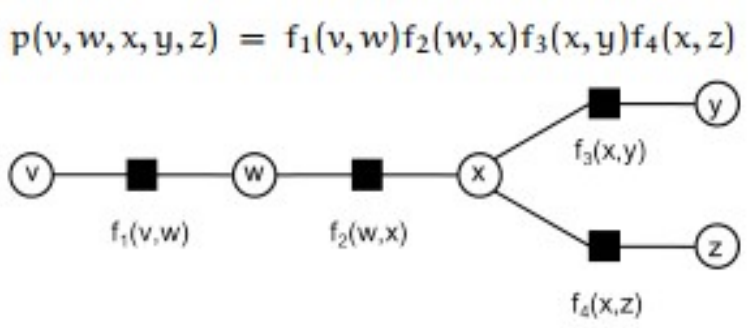

Figure 10.8: An illustration of Factor Graph [1].

10.2.2 The Sum-Product Algorithm

We describe an efficient algorithm to compute all the marginal functions in a factor graph. The marginals of each variable can be represented as a sum on product of all the incoming messages from neighbor factors. We denote two types of messages:

- A message sent from a node of variable x to a node of local function f_i as $\mu_{x\to f_i}(x)$
- A message sent from a node of local function f_i to a node of variable x as $\mu_{f_i\to x}(x)$

Assume that our factor graph of function g does not contain cycles (a tree form) and represents a joint probability mass function. We are interested in computing all the marginals $g_i(x_i)$. $n(x)$ stands for the neighbors of a node (variable or a local function), X stands for the set of arguments of the function f_i.
The sum-product update rule for a message from a node v on an edge e is the product of the local function at v (or unit function if v is a variable) with all messages received at v on edges other than e, summarized for the variable associated with e, Hence, we denote:

$$\mu_{x\to f_i}(x) = \prod_{h\in n(x)\setminus\{f\}} \mu_{h\to x}(x) \tag{10.1}$$

$$\mu_{f_i \to x}(x) = \sum_{\tilde{x}} \left(f_i(X) \prod_{y \in n(f) \setminus \{x\}} \mu_{y \to f_i}(y) \right) \tag{10.2}$$

In order to calculate the messages, we use a 'bottom-up' approach. First we compute the message at the leaf nodes, and then propagate the message up on the tree, until we reach the root variable. Specifically, each leaf variable node sends a trivial identity function message to its parent, and each leaf factor node sends a description to its parent. Each vertex waits for messages from all of its children before computing the message to be sent to its parent.

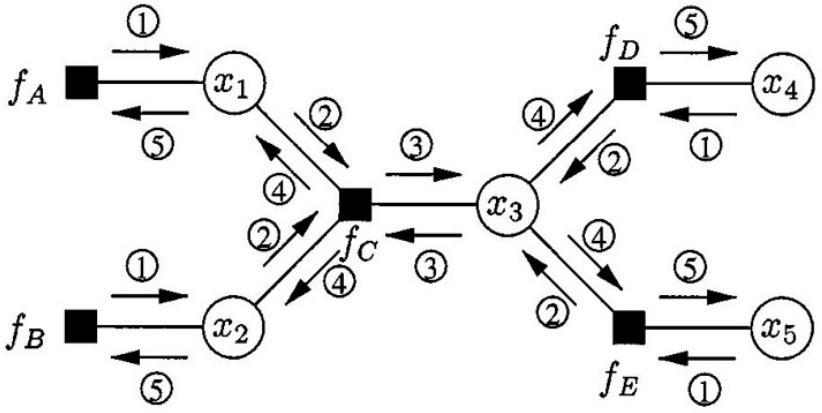

Figure 10.9: An illustration messages transactions in A Factor Graph

After computation of all the above messages (based on the above formulation) the marginal of a variable can be computed by the product of all the messages sent to the variable, namely:

$$g_i(x_i) = \prod_{f_j \in n(x_i)} \mu_{f_j \to x_i}(x_i) \tag{10.3}$$

The algorithm also known by the names 'sum-product algorithm', or 'belief propagation algorithm' or 'passing message algorithm'.Using the same procedure, marginals for all nodes of a tree can be computed. The message on each edge in each direction is passed as soon as it is computed (see Figure 10.9).

The complexity of the algorithm for trees depends on the number of variables involved in each local function. Suppose each of the n variable attains k levels, and there are m local function. Let d be the maximum node degree in the factor graph. Computing (10.1) takes $O(d)$ per node. Computing (10.2) takes $O(k^{d+1})$ products and $O(k^d)$ sums per node. Hence the total time is $O(md + nk^{d+1})$

10.2.3 Factor Graph with Cycles

The above procedure is not guaranteed to terminate on cyclic graphs. There are two practical approaches to deal with factor graphs with cycles. The first one is to keep iterating and computing the messages until the changes between its results on consecutive iteration are smaller than a constant ϵ. The second approach limits the number N of iterations. In practice often both ϵ and N limits are used.

10.3 Paradigm

10.3.1 Introduction

The PARADIGM algorithm tries to infer patient specific genetic activities incorporating omic data and curated pathway interactions among genes (*Vaske et al.* 2010 [9]). PARADIGM's input is composed of (1) patient's gene expression profile, (2) patient's copy number variation profile, (3) curated pathways that are related to cancer. Paradigm's model is based on factor graphs. A diagram of the model inputs and outputs is shown in Figure 10.10.

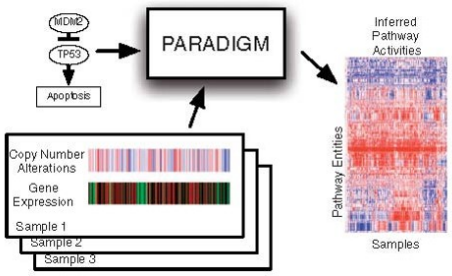

Figure 10.10: An illustration of paradigm algorithm

PARADIGM makes some assumptions on its input arguments, under the general premise that gene expression is not a direct indicator of the gene activity. The first assumption is that in the case that gene A is upstream from gene B in a pathway, and gene A is an activator of gene B, their gene expression tend to be correlated (since gene A is a direct or indirect regulator of gene B). The second, if A has high copy number then B may show high expression even if not regulated by A. In other words, if gene A is upstream from gene B in a pathway, and gene A is an activator of gene B, the CN of gene A, tends to be correlated to gene B. Those assumptions are illustrated on Figure 10.11.

10.3.2 PARADIGM model

The model of PARADIGM is based on a factor graph. The factor graph encodes the state of the cell using random variables for its entities $X = \{x_1, x_2, ..., x_n\}$ where $x_i \in \{-1, 0, 1\}$. Those variables represent the differential state of each entity in comparison with a normal level (rather than direct concentrations of the entities). In this manner each entity is in one of three possible states: activated, nominal and deactivated, which are interpreted differently depending on the type of entity (CN, GE etc.).

The local functions in the factor graph represent interactions between two or more variables (e.g. entities). There are m non-negative functions that constrain the entities to take

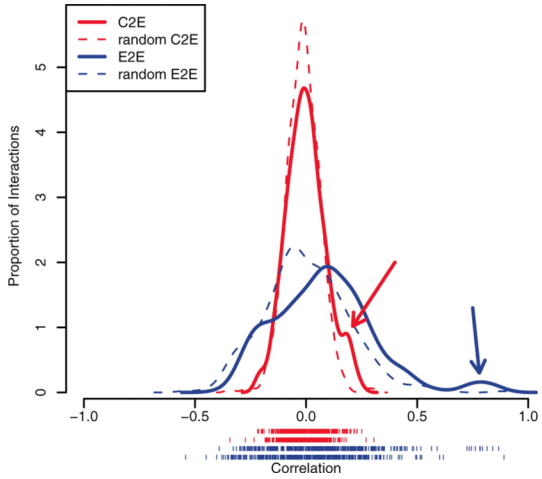

Figure 10.11: The following graph is based on the TCGA GBM data, of 462 individuals where gene A is upstream activator of gene B. X axis stands for the correlation, and Y axis presents the density. The red line (C2E) presents the correlation of A's CN and B's GE. The blue line (E2E) presents the correlation of A's GE and B's GE. The dashed lines present the correlation between random paired genes

biologically meaningful values. The j^{th} factor ϕ_j defines a probability distribution over a subset of entities $X_j \subset X$. We denote the joint probability distribution of all the entries as (Z is a normalization constant):

$$P(X) = \frac{1}{Z} \prod_{j=1}^{m} \phi_j(X_j) \tag{10.4}$$

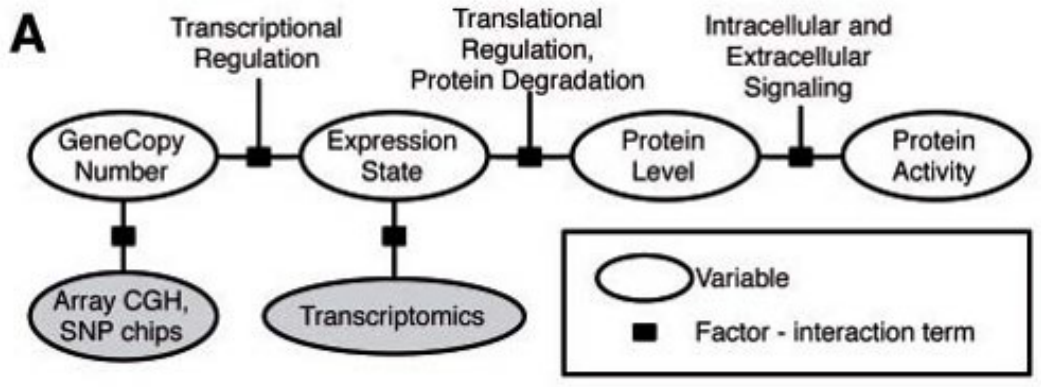

Figure 10.12: An illustration of a gene's sub-model in PARADIGM

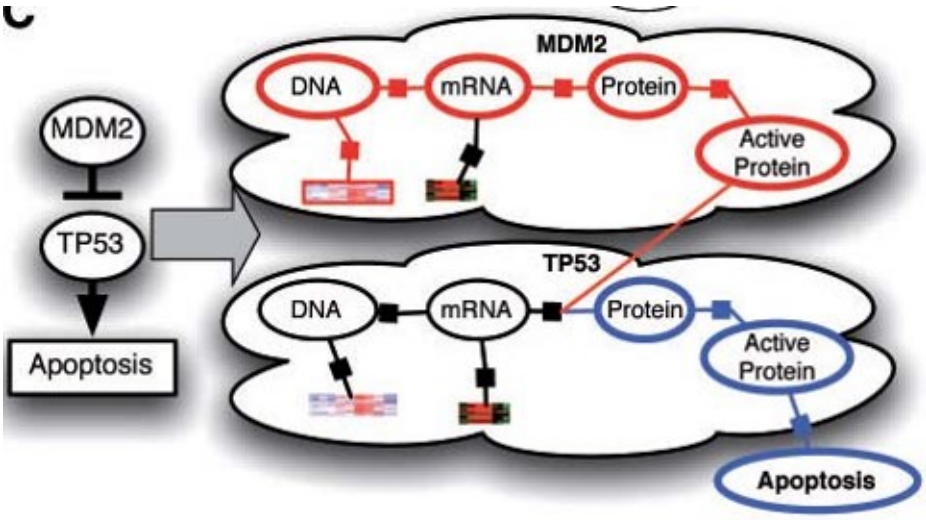

Figure 10.13: Toy example of a small sub-pathway involving P53, an inhibitor MDM2 and the high level process of apoptosis

The model generation is made by creating a directed graph. The typical information flow is from DNA to mRNA then to Protein and then to Protein Activity. There will be positive edges between pairs and their direction follows the typical biological order (as presented in Figure 10.12 from left to right). Other edges in the pathway have a positive or negative label according to their function. For example, protein-protein interactions (PPIs)

can have both negative and positive edges, depending on the pathway information. We assign a factor $\phi_i(X_i)$ to each variable x_i, which consists of the variable node and its parents, $X_i = \{x_i\} \cup \{parents(x_i)\}$ and represents their joint probability. The expected value of ϕ_i is based on a majority vote of the parent variables. If a parent is connected by positive edge, it contributes a vote of (+1)×(its state), and symmetrically (-1)×(its state) for a negative edge. Hence, we denote:

$$\phi_i(x_i, Parents(x_i)) = \begin{cases} 1-\epsilon & x_i \text{ is the expected state from Parents}(x_i) \\ \frac{\epsilon}{2} & \text{otherwise} \end{cases} \tag{10.5}$$

Our inputs' values , e.g. gene's CN and GE are rescaled as follows. Each input is ranked based on all the samples from smallest to largest, and afterwards values are mapped into the interval $[0, 1]$. Then the values will be discretized into three values. Then, for a given individual the observed data is denoted as $D = \{x_1 = s_1, ..., x_k = s_k\}$, while ϕ is a fully specified factor graph. $S \sqsubset_D X$ stands for all the possible assignments to a set of variables X that are consistent with the assignments in D (e.g. any observed variable x_i is fixed to its assignment in D, while hidden variables can be varied). We estimate the prior probability of the hidden variable a, given our factor graph ϕ, using the sum-product algorithm. $A_i(a)$ stands for the singleton assignment set $\{x_i = a\}$

$$P(x_i = a|\phi) = \frac{1}{Z} \prod_{j=1}^{m} \sum_{S \sqsubset_{A_i(a)} X_j} \phi_j(S)) \tag{10.6}$$

In the same manner, the joint probability of a and an individual data (D) is:

$$P(x_i = a, D|\phi) = \frac{1}{Z} \prod_{j=1}^{m} \sum_{S \sqsubset_{A_i(a) \cup D} X} \phi_j(S) \tag{10.7}$$

We can estimate the values of the parameters of the model (such as ϵ) using the Expectation-

Maximization algorithm (details now shown).

We compute the log-likelihood ratio based on equations 10.6, 10.7 in order to estimate the level to which a patient's data increase our belief that entity i's activity is up/down using Bayes rule:

$$L(i,a) = log\left(\frac{P(x_i = a, D|\phi)}{P(x_i \neq a, D|\phi)}\right) - log\left(\frac{P(x_i = a|\phi)}{P(x_i \neq a|\phi)}\right) = log\left(\frac{P(D|x_i = a, \phi)}{P(D|x_i \neq a, \phi)}\right) \tag{10.8}$$

Then, we compute the integrated pathway activity (IPA) score of gene i based on the $L(i,a)$ result, in a manner that keeps the IPA score's sign analog to its $L(i,a)$.

$$IPA(i) = \begin{cases} L(i,1) & L(i,1) > L(i,-1) \text{ and } L(i,1) > L(i,0) \\ -L(i,-1) & L(i,-1) > L(i,1) \text{ and } L(i,-1) > L(i,0) \\ L(i,0) & \text{otherwise} \end{cases} \tag{10.9}$$

10.3.3 Significance Assessment

In order to estimate an IPA score's significance, two different methods for generating randomized data were used. In the 'within' method, datasets contain tuples of GE and CN from a random individual sample chosen from a random gene within the same pathway. In the 'any' method, datasets contain tuples of GE and CN from a random individual sample, chosen from a random gene within the whole genome. We compute the empirical p-values based on these distributions. For each method 1000 datasets are generated, PARADIGM computes IPA scores for their genes, and the distribution of the scores is formed. Figure 10.14 shows a plot using the true and randomized IPA scores.

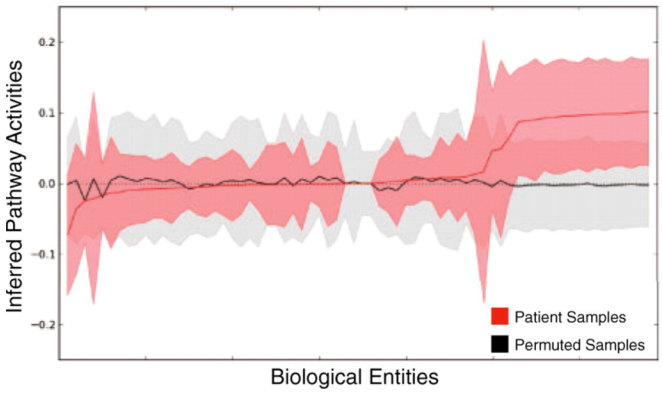

Figure 10.14: An example of a cancer breast patient's IPAs (Based on a pathway that is related to cancer) compared with 'within' permutations. The red and blue line represent the mean IPA, and the colored area stands for the standard deviation. The biological entities on the right side of the X axis are the ones that are associated to cancer.

10.3.4 Decoy pathways

For each type of cancer, a set of decoy pathways was generated. A decoy pathway maintains the original pathway's topology but every gene was substituted with a random gene. A significance analysis for both PARADIGM and SPIA (*Tarca et al.* 2009, an alternative approach) was executed on both real and decoy pathways, and then the pathways were ranked

according to their p-values. The ROC curve in Figure 10.15 presents the fraction of real pathways versus the fraction of total pathways for each method.

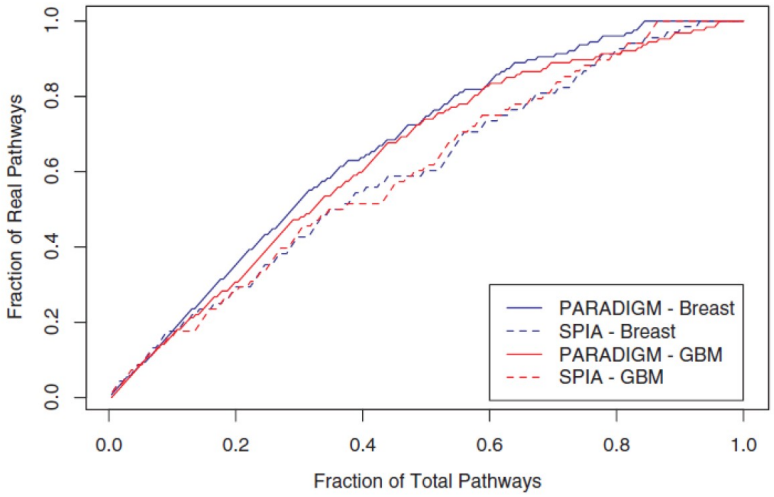

Figure 10.15: The computed ROC curve.

10.3.5 Circle Maps

Vaske et al. developed a novel display method, that, for each entity in the pathway, presents the patients' data types in concentric circles. Each patient corresponds to a section in the circle. Figure 10.16 shows the ERBB2 pathway circle Map. The ER status (Estrogen Receptor, a typical bio-marker of breast cancer), IPA scores, GE data and CN data are displayed as concentric circles, from innermost to outermost, respectively.

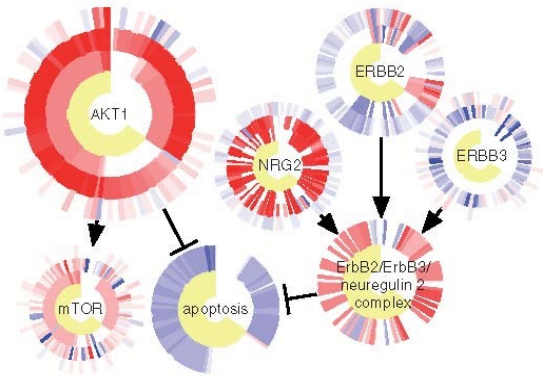

Figure 10.16: Circle Map display of the ErBB2 pathway.

10.3.6 Clustering IPAs

The authors chose to use the IPAs to cluster the patients. Figure 10.17 shows a hierarchical clustering of GBM IPA scores. Four clear clusters can be seen. Remarkably, patients in one of the clusters had a significantly better survival rate than the others (Figure 10.18). Additional applications of PARADIGM can be seen, e.g. in Spellman et al. 2011[7].

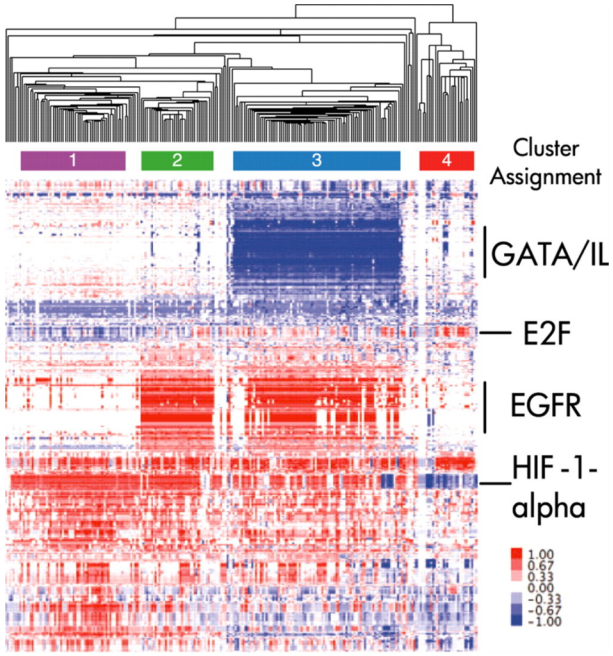

Figure 10.17: Hierarchical clustering of IPAs scores of 229 patients with Glioblastoma (GBM). Only entities with $0.25 < IPAScore$ in at least 75 of 229 individuals were taken.

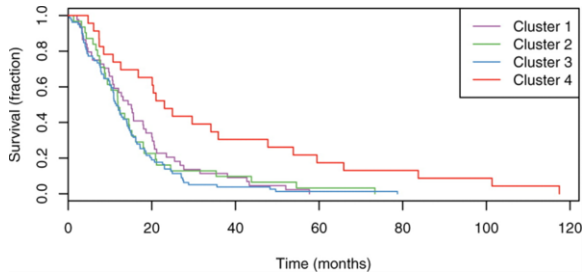

Figure 10.18: The Kaplan-Meier curves corresponds to the hierarchical clustering results

10.3.7 PARADIGM-SHIFT

Mutations within genes can cause gain-of-function (GOF), which adds a new functionality to a gene, or loss-of-function (LOF), which deactivates a particular activity of the gene. Such mutations can be inferred by their effect on a pathway. For example, in the case of LOF mutation, even high gene activity will not effect its target gene's expression (see Figure 10.19).

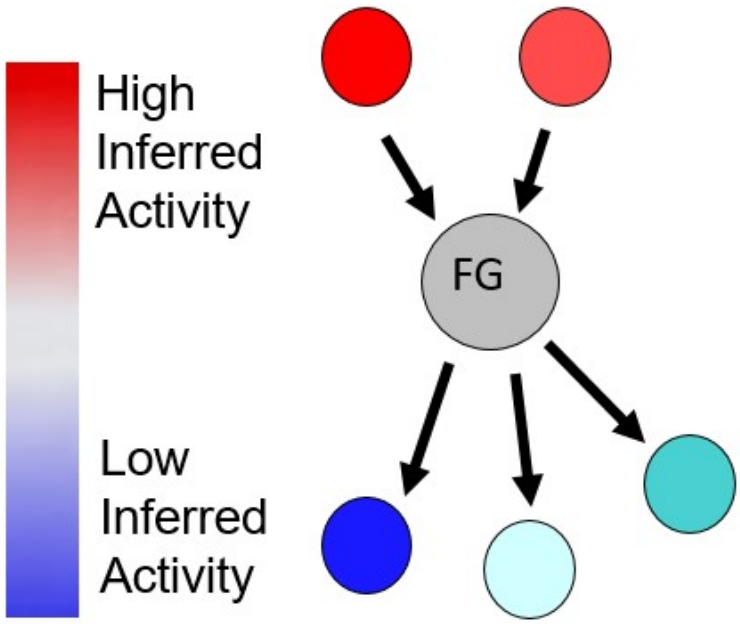

Figure 10.19: An illustration of loss of function mutation, FG stand for the focus (target) gene

The PARADIGM-SHIFT approach uses the GE, CN and mutation data in order to predict the impact of the observed mutations on genetic pathways in patient tumors (*Ng et al.* 2012[8]). The idea is to use the IPA scores in order to assess the expected activity of a focus gene, based on its upstream regulators. Similarly we assess the observed gene activity based on its downstream targets. To do that, we calculate the IPA score of a focus gene twice, once with its regulators only and once with its targets only. Based on both IPAs scores, we calculate

the P-shift score of gene f, $PS(f)$. Figure 10.20 shows an overview of the process.

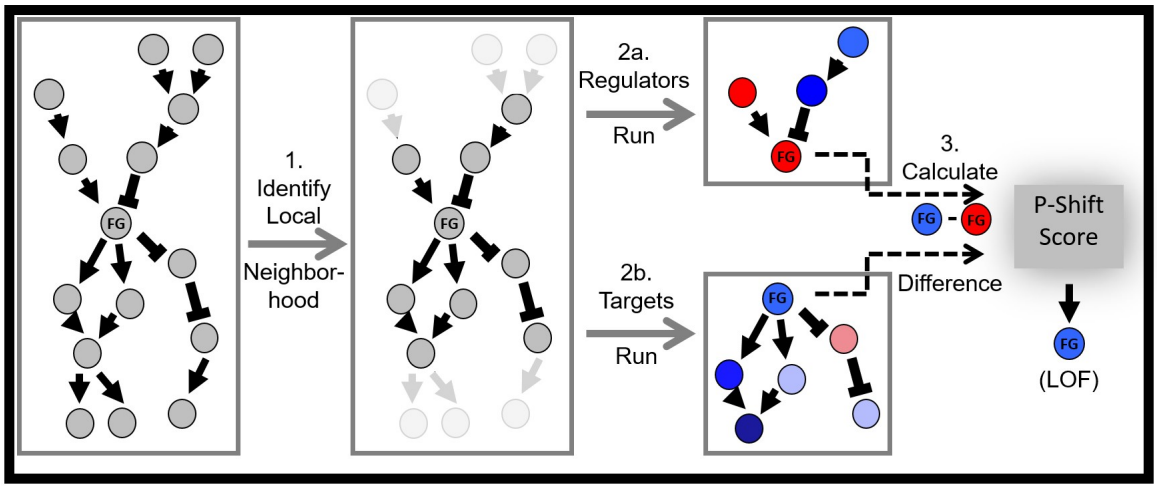

Figure 10.20: An illustration the PARADIGM-SHIFT calculation process

We denote f as a focus gene. $R-run$ stands for PARADIGM's execution using upstream regulators only, and $\phi(R)$ represents the model including gene f and its regulators. $T-run$ stands for PARADIGM's execution using downstream targets only and $\phi(T)$ represents its model including gene f and its targets. $D(R)$, $D(T)$ and $D(f)$ denote the observed data of the upstream regulators, the downstream targets, and the focus gene, respectively. We can write the likelihood ratio of the data Y given $x = a$ and the alternative, namely $x \neq a$, and a model Z based on equation 10.8:

$$LR(Y|x^a, Z) = \frac{P(Y|x^a, Z)}{P(Y|x^{\neg a}, Z)} \tag{10.10}$$

We generate $PS(f)$ score as the delta between the the $IPA_{T-run}(f)$ and $IPA_{R-run}(f)$, Hence:

$$PS(f) = IPA_{T-run}(f) - IPA_{R-run}(f)$$

$$= log\left(\frac{P(D(T)|x_f^a, \phi(T)}{P(D(T)|x_f^{\neg}a, \phi(T)}\right) - log\left(\frac{P(D(R)|x_f^a, \phi(R)}{P(D(R)|x_f^{\neg a}, \phi(R)}\right) = log\left(\frac{LR(D(T)|x_f^a, \phi(T))}{LR(D(L)|x_f^a, \phi(L))}\right)$$

$$= log\left(\frac{P(D(T), x_f^a|\phi(T)}{P(D(T), x_f^{\neg a}|\phi(T)}\right) - log\left(\frac{P(D(R), x_f^a|\phi(R)}{P(D(R), x_f^{\neg a}|\phi(R)}\right) - log\left(\frac{P(x_f^a|\phi(T)}{P(x_f^{\neg a}|\phi(T)}\right) - log\left(\frac{P(x_f^a|\phi(R)}{P(x_f^{\neg a}|\phi(R)}\right)$$

$$= log\left(\frac{P(D(T), x_f^a|\phi(T)}{P(D(T), x_f^{\neg a}|\phi(T)}\right) - log\left(\frac{P(D(R), x_f^a|\phi(R)}{P(D(R), x_f^{\neg a}|\phi(R)}\right) - prior \qquad (10.11)$$

Hence, for K mutated genes, we need to execute PARADIGM $2K$ times in order to calculate $PS(f)$ for each gene f. Transformation of $PS(f)$ into $Z - score$ provided better results in practice to assign significance to the results. The authors construct 100 random samples for each gene by shuffling data, and calculate its $PS(f)$. Then the score is normalized by subtracting the mean and dividing by the standard deviation (computed on the random sample).

Figure 10.21 shows a circle map for RB1 mutations in GBM. Figure 10.22 shows RB1 together with its regulators and targets. Figure 10.23 shows the significance of the P-shift score of RB1 by comparison to randomized models.

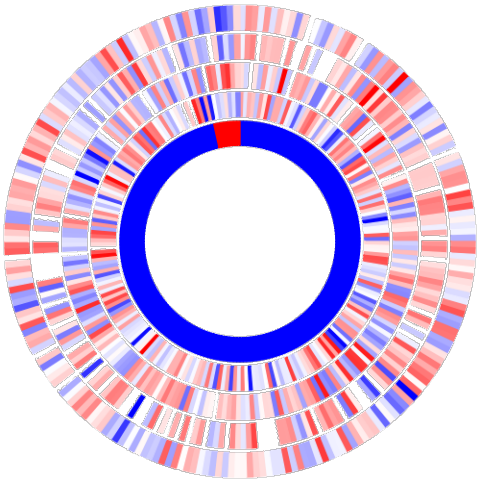

Figure 10.21: CircleMap display shows **RB1** results on GBM. From innermost to outermost ring: mutation status, expression level, activity derived from the Regulators-Run, activity from the Targets-Run and the P-Shift score. The red sector corresponds to the mutation. We can see that the P-Shift score highlights the importance of the mutation.

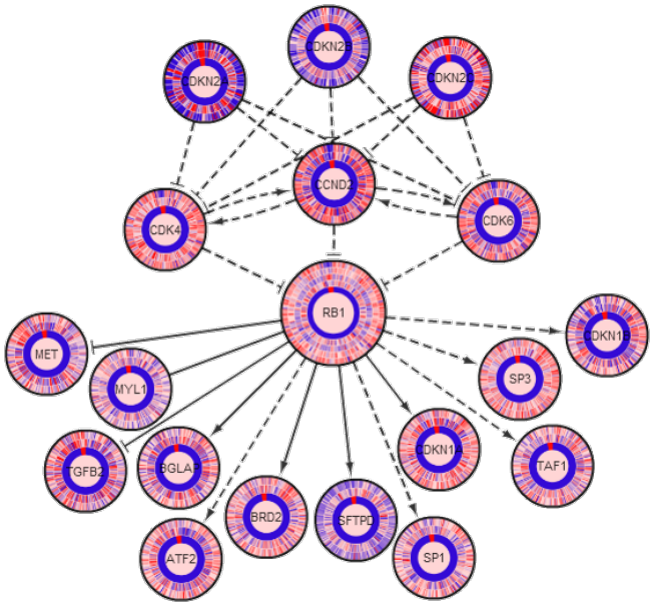

Figure 10.22: The CircleMap of the full pathway of RB1

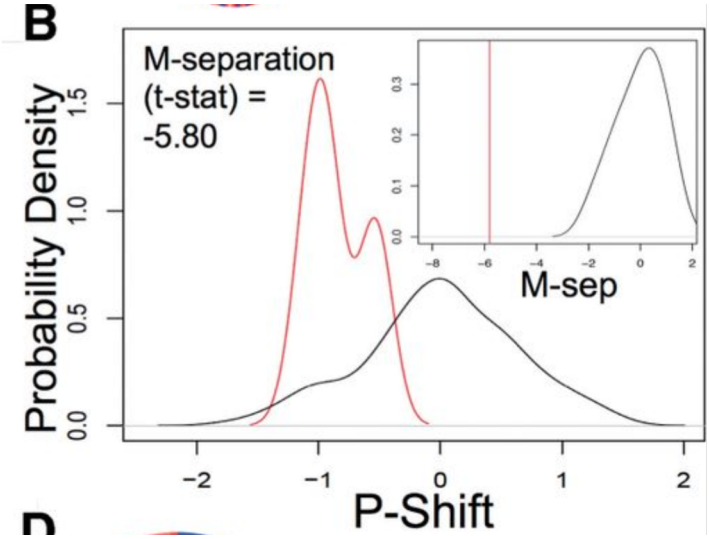

Figure 10.23: P-Shift scores (X-axis) calculated for samples harboring mutations in RB1 (red histogram) plotted alongside P-Shift scores calculated for samples lacking a reported RB1 mutation (black histogram). Y-axis shows probability density for each population. The right upper figure is the M-separation score, which shows that the difference between mutant and non-mutant (ored line) is significantly lower than the difference obtained in 1000 randomized background simulation models with the same pathway structure but with random genes.

A key advantage of the method is its ability to detect genes that are involved in LOF or GOF even if the dataset shows them mutated in a small number of patients. Figure 10.24 demonstrates this convincingly.

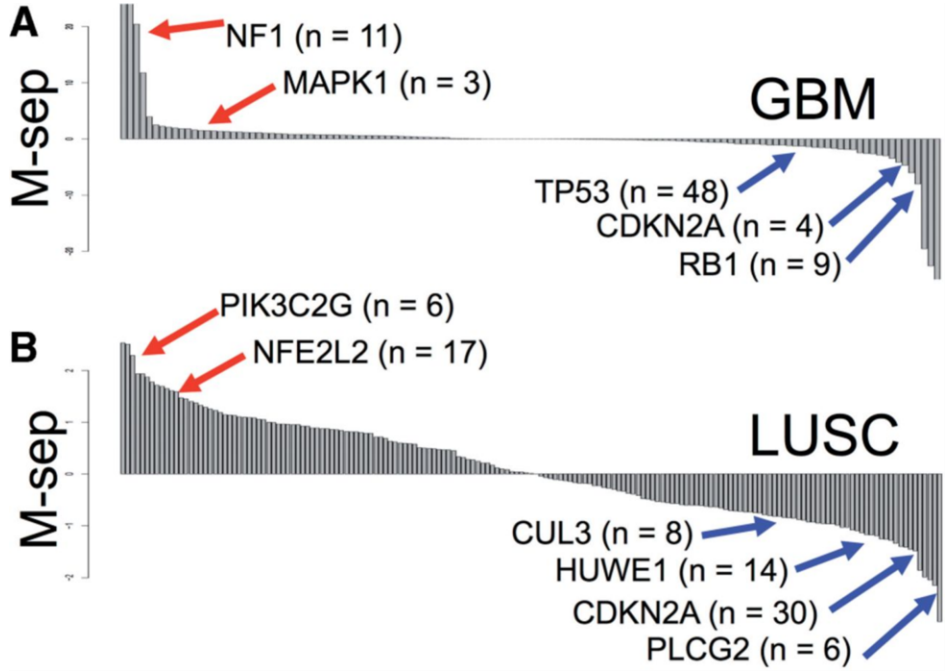

Figure 10.24: The figure shows that PARADIGM-SHIFT can also predict mutations that occur in a tiny number of individuals, both on GBM cancer and in lung cancer (LUSC) solely based on there m-separation score. The pointed genes are well known genes associated with those types of cancer. Some of them have extreme M-seperation scores even though they occur in small number of patients (n)

10.4 ResponseNet

10.4.1 Genetic Screens

In genetic screen, one or more genes of the organism are altered or deleted and the resulting phenotype is observed (e.g. growth rate). In the yeast deletion project, a set of 6,000 homozygous diploid (a genotype consisting of two identical alleles at each locus) single deletion mutants was generated, namely, each mutant contained a deletion of one gene. Each mutant was identified by a special sequence signature ("barcode"). 1,100 mutants did not survive when grown on a rich culture. Those genes are defined as **essential** genes, the rest are the **non-essential** genes.

To study a particular condition , all the non-essential deletion strains can be grown in that condition, and the phenotype (e.g. growth rate) is measured. The strains that are changed by the condition can be identified by their barcodes. Their genes are called **genetic hits** of the condition. This process enables testing all genes simultaneously. In parallel, one measures the expression of all the genes of the yeast. In that manner we are able to detect a set of differentially expressed genes (DEG). For example, if we alter 10 genes and cause a LOF mutation, those mutations are supposed to affect various processes in the cell, and specifically cause (directly or indirectly) different GE of other genes. The biological assumptions of the study is that some of the hits will be connected through regulatory pathways to DEGs.

10.4.2 ResponseNet Model

The model is based on a graph with two types of nodes, genes and proteins ($Yeger\ et\ al.$ 2011[10]). Instead of the genetic hits, we use their corresponding protein as nodes. Moreover, we add two types of edges, undirected edges represent known PPIs, and directed edge represents a protein-DNA interaction (PDI), for instance an interaction between a TF and its target gene (see Figure 10.25). The cost of each edge is its confidence level.

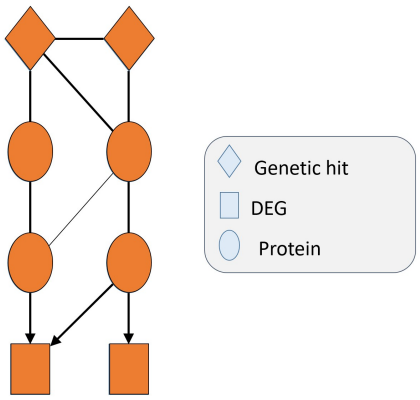

Figure 10.25: An illustration of the graph.

The idea is to formulate a 'Minimun-Cost Flow' (MCF) problem (adding a source and a sink), and use it to detect pathways with strong flow that connect between genetic hits and DEGs, and thus tend to be influential to the process.The overall process is outlined in Figure 10.26. The detailed formulation is described later in this section.

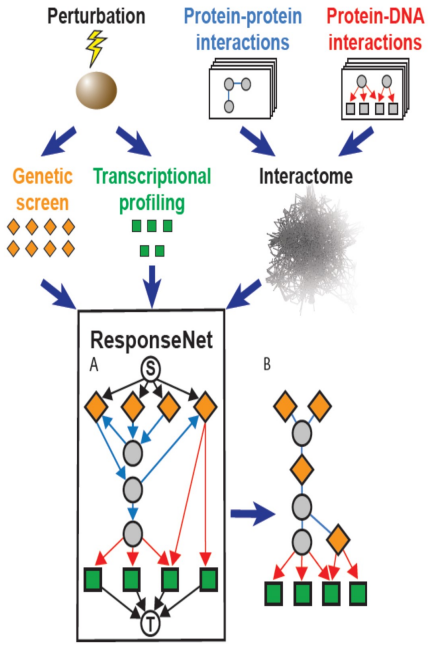

Figure 10.26: An illustration of the process of ResponseNet algorithm

In order to represent the confidence level (cost) of a PPI and PDI a Bayesian weighting scheme was developed based on the information sources supporting each interaction. Each edge $(i, j) \in E$ is characterized by a weight w_{ij}. For each edge, a vector $X = \{x_1, x_2, ..., x_k\}$ was generated, where entry x_l is an indicator of specific interaction of type l (for example high-throughput two-hybrid experiment), $x_e = 1$ if the edge is detected by this type of interaction. Eventually, the weight w_{ij} stands for the probability of the interaction between node i and node j in a randomly selected pathway. We use Bayes rule to assess the probability ($P(i, j[+])$ represents the probability that i and j interact):

$$w_{ij} = P(i, j[+]|X) = \frac{P(X|i, j[+]) \cdot P(i, j[+])}{P(X)} \tag{10.12}$$

$$P(X) = P(X|i, j[+]) \cdot P(i, j[+]) + P(X|i, j[-]) \cdot P(i, j[-]) \tag{10.13}$$

By assuming independence between different types of evidence, we can compute:

$$P(X|i, j[+]) = \prod_k P(X_k|i, j[+]) \tag{10.14}$$

In order to calculate $P(X_k|i, j[+])$ and $P(i, j[+])$ we use data from well studied interaction pathways (obtaining confident non-interaction edges as well).

different information sources on PPIs vary dramatically in their confidence. The authors capped edge weights at 0.7 in order to avoid the dominant impact of edges with high probability on the model. For PDIs a similar cap was used. We are now ready to formulate the MCF problem: Add a source S and connect it by a directed edge to all hits, and add a sink T and connect it by a directed edge from each DEG node. Call the resulting graph $G' = (V', E')$. Suppose the strength of the genetic hit is m_x for gene x. We assign each edge from S to hit x the capacity and cost $\frac{m_x}{\sum m_i}$. The ration of the logarithm of the expression of DEG x in the condition vs. normal is n_x. We assign each edge from target y to T capacity

and cost $\frac{n_y}{\sum n_i}$. All the other edges have capacity of 1 and cost as equal $1 - w_{ij}$ (see Equation 10.12 ?).

A linear programming formulation is as follows. f_{ij} stands for the flow on the edge (i, j):

$$min_f \left(\left(\sum_{i \in V', j \in V'} -log(w_{ij}) \cdot f_{ij} \right) - \left(\gamma \cdot \sum_{i \in Gen} f_{Si} \right) \right)$$

$$\forall i \in V' - \{S, T\} : \sum_{j \in V'} f_{ij} - \sum_{j \in V'} f_{ji} = 0$$

$$\forall (i, j) \in E' : c_{ij} \geq f_{ij} \geq 0 \tag{10.15}$$

The left term of the objective stands for the MCF objective. The right term tries to regularize the number of genetic hits that are used in the model.γ is a calibration parameter.

10.4.3 Results

The results are divided into two sections, the first one validates the algorithm based on well known pathways. The second, tries to use the algorithm in order to detect new pathways.The MAPK pathway consists of a set of proteins that activate a cascade of signals from a receptor on the cell membrane to the DNA in the cell's nucleus. STE5 is a MAPK scaffold protein involved in the mating of yeast and is activated by pheromone. This network was created by taking the induced subnetwork of all nodes at distance ≤ 3 edges from STE5 (Figure 10.27). It contains 193 nodes and 778 edges. It was used as the underlying model for ResponseNet on the data of genetic hits and DGEs for yeast STE5 deletion strain.

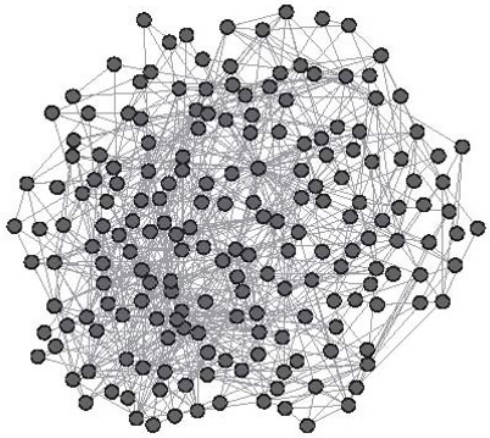

Figure 10.27: The STE5-centered MAPK subnetwork used by ResponseNet

ResponseNet identified a subnetwork of 23 nodes and 96 edges (Figure 10.28). Importantly, many of the central players in MAPK pathways were included in it.

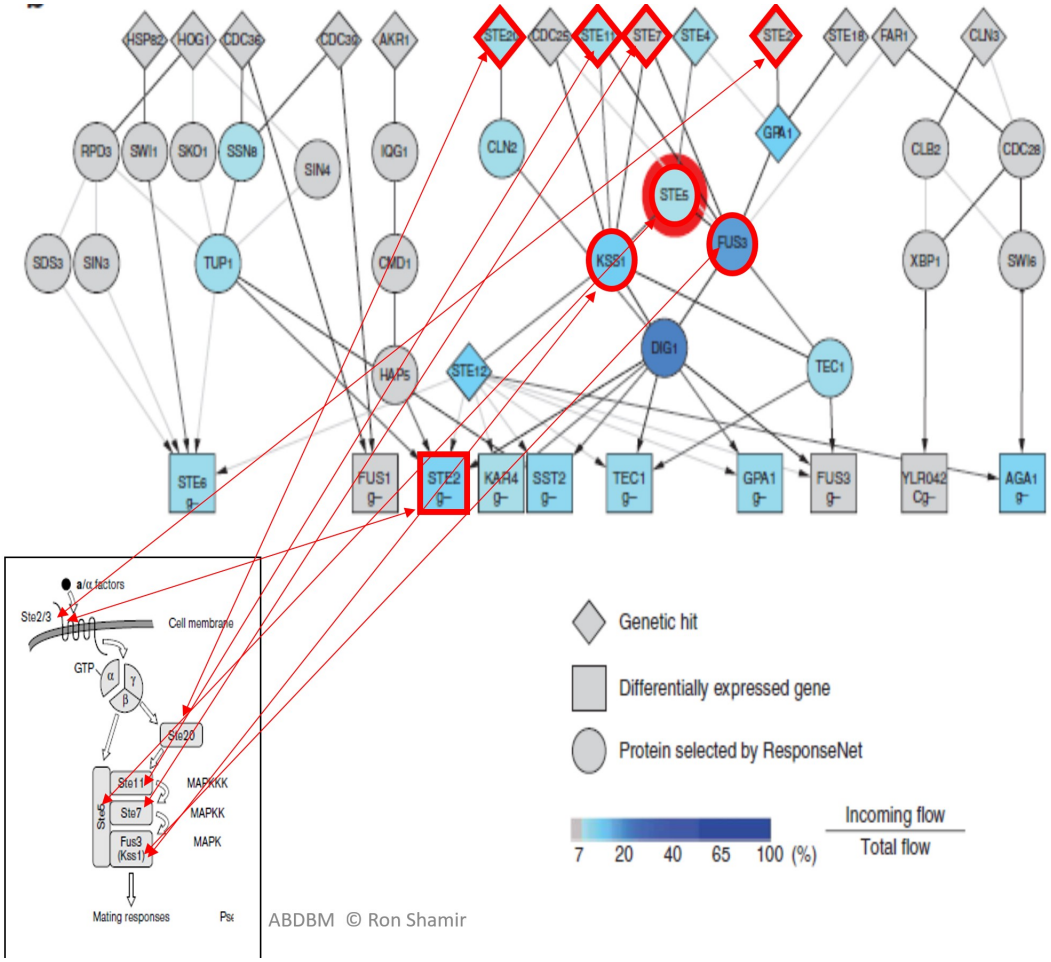

Figure 10.28: A 'Zoom in' of the network produced by ResponseNet for the STE5 data.It contains 23 nodes and 96 edges. Higher ranked nodes, as determined by ResponseNet appear in darker shades of blue. The red nodes present the core components of the typical MAPK pathway including STE5.The arrows show the mapping of discovered nodes to genes known to be involved in MAPK pathway (left bottom).

The algorithm is also applied to a large DNA damage response network with 1448 genetic hits and 198 DEGs in response to the DNA damage in agent MMS. One of the subnetwork identified is shown in Figure 10.29.

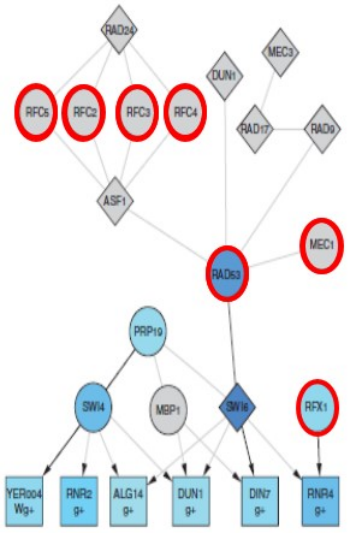

Figure 10.29: One main component in the highly-ranked part of the network created by ResponseNet upon connecting genetic hits to DNA damage signature genes identified in yeast treated with the DNA damaging agent MMS. The highest ranking intermediate nodes predicted by ResponseNet include core components of the DNA damage response pathway such as MEC1,RAD53, RFC2, RFC3, RFC4,RFC5, and RFX1, Moreover all those genes and the gene RFX1 essential genes and could not have been detected via genetic screening. The colors scale is the same as in figure 10.28.

Finally, the algorithm was applied to data related to Parkinson disease (PD). The protein Alpha-Synuclein, a main actor in PD, was over expressed in the deletion yeast library and 77 hits were observed. Many insights on PD were identified via the subnetworks found (Figure 10.30), and some were experimentally verified. The results provided functional explanations for many genes whose relation to Alpha-Synuclein was previously unexplained.

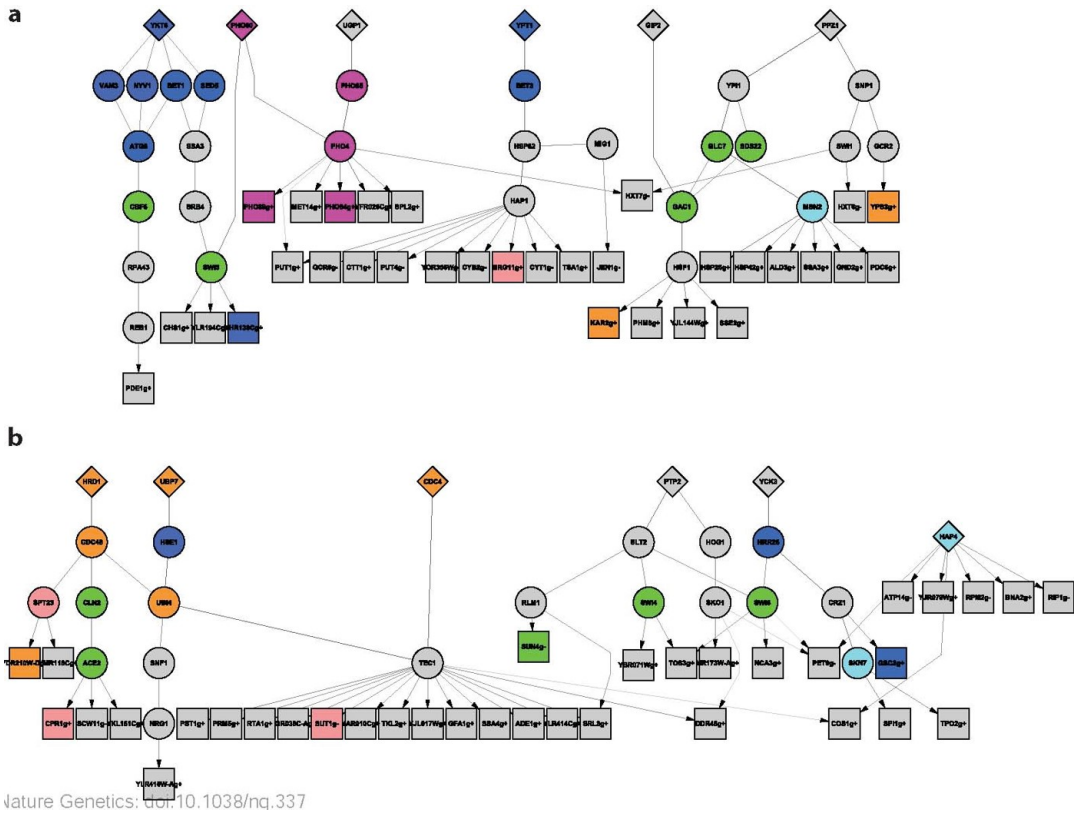

Figure 10.30: One of the results of ResponseNet on the pathway of the Alpha-Synuclein protein, which is related to Parkinson disease. The Alpha-Synuclein protein was expressed in the yeast model using genetic engineering techniques, and 77 hits were observed. Then ResponseNet found a sub-network of 34 genetic-hits and 166 DEGs.

Bibliography

[1] http://mlg.eng.cam.ac.uk/teaching/4f13/1718/factor%20graphs.pdf.

[2] https://string-db.org/.

[3] http://www.cs.tau.ac.il/\~spike/.

[4] http://www.ebi.ac.uk/research/pfmp.

[5] www.its.caltech.edu/~mirsky/endomeso.html.

[6] Frank R Kschischang, Brendan J Frey, and H-A Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on information theory*, 47(2):498–519, 2001.

[7] Cancer Genome Atlas Research Network et al. Integrated genomic analyses of ovarian carcinoma. *Nature*, 474(7353):609–615, 2011.

[8] Sam Ng, Eric A Collisson, Artem Sokolov, Theodore Goldstein, Abel Gonzalez-Perez, Nuria Lopez-Bigas, Christopher Benz, David Haussler, and Joshua M Stuart. Paradigm-shift predicts the function of mutations in multiple cancers using pathway impact analysis. *Bioinformatics*, 28(18):i640–i646, 2012.

[9] Charles J Vaske, Stephen C Benz, J Zachary Sanborn, Dent Earl, Christopher Szeto, Jingchun Zhu, David Haussler, and Joshua M Stuart. Inference of patient-specific path-

way activities from multi-dimensional cancer genomics data using paradigm. *Bioinformatics*, 26(12):i237–i245, 2010.

[10] Esti Yeger-Lotem, Laura Riva, Linhui Julie Su, Aaron D Gitler, Anil G Cashikar, Oliver D King, Pavan K Auluck, Melissa L Geddie, Julie S Valastyan, David R Karger, et al. Bridging high-throughput genetic and transcriptional data reveals cellular responses to alpha-synuclein toxicity. *Nature genetics*, 41(3):316–323, 2009.