Probabilistic Methods in Cancer Biology

M. Vidyasagar

***Abstract*— Recent advances in experimental techniques have made it possible to generate an enormous amount of 'raw' biological data, with cancer biology being no exception. The main challenge faced by cancer biologists now is the generation of plausible hypotheses that can be evaluated against available data and/or validated through further experimentation. For persons trained in control theory, there is now a significant opportunity to work with biologists to create a virtuous cycle of hypothesis generation and experimental validation. Given the large number of uncertain factors in any biological experiment, probabilistic methods are a natural in this setting. In this paper, we discuss four specific problems in cancer biology that are amenable to study using probabilistic methods, namely: Reverse engineering gene regulatory networks, constructing context-specific gene regulatory networks, analyzing the significance of expression levels for collections of genes, and discriminating between drivers (mutations that cause cancer) and passengers (mutations that are caused by cancer or have no impact). Some research problems that merit the attention of the controls community are also suggested.**

I. Introduction

A. Scope of the Paper

The interplay between mathematical modeling and experimental biology dates back several decades, and 'theoretical biology' has been a well-accepted discipline for a very long time (even if the name of the area keeps 'mutating'). In recent years, many persons whose primary training was in the systems & control area have moved into biology and have made many significant contributions, and continue to do so. This is illustrated by the fact that in recent years there have been two special issues within the controls community that are devoted to systems biology [1], [2]. Apart from this, several persons with a control or system theory background publish regularly in the 'mainstream' biology literature. It would be impossible to create a comprehensive description of all these contributions, and in any case, that is not the focus of the paper. Rather, the objective of this overview paper is to present a snapshot of some research problems in cancer biology to which methods from probability and statistics may be fruitfully applied. In that sense, the scope of the paper is voluntarily limited.

Biology is a vast subject and cancer biology is a very large part of this vast subject. Moreover, our understanding of this topic is constantly shifting, and there are very few 'settled' theories.[1] Hence the choice of the specific topics discussed here is dictated by the fact of their presenting some reasonably deep challenges in probability theory and statistics, and of course by the author's personal tastes. The hope is that the paper would at least serve to present the flavor of this subject, and thus motivate interested readers to explore the literature further. In particular, there is no pretense that the article is comprehensive; rather, it should be thought of as an introduction to the topic.

Cecil & Ida Green Endowed Chair, Erik Jonsson School of Engineering & Computer Science, University of Texas at Dallas, 800 W. Campbell Road, EC38, Richardson, TX 75080, USA; email: M.Vidyasagar@utdallas.edu. This research was supported by National Science Foundation Award #1001643.

Fig. 1. The Ebers Papyrus [4]

B. Some Facts & Figures About Cancer

Cancer is one of the oldest diseases known to man. A papyrus popularly known as the 'Ebers papyrus' [4], dating to around 1500 BCE, recounts a 'tumor against the God Xenus' and suggests 'Do thou nothing there against'. The Ebers papyrus is reproduced in Figure 1.

The currently used cancer-related terms come from both Greek and Latin. In ancient times, the Greek word 'karkinos', meaning 'crab', was used to refer to the crab nebula as well as the associated zodiac sign. Supposedly Hippocrates in c.420 BCE used the word 'karkinos' to describe the *disease*, and 'karkinoma' to describe a cancerous *tumor*. One can surmise that he was influenced by the crab-like appearance of

[1] For a very readable and yet scientifically accurate description of how theories about the onset and treatment of cancer have evolved over the past hundred years or so, see [3].

a cancerous tumor, with a hard and elevated central core and lines radiating from the core. Subsequently the name for the disease was changed to the Latin word 'cancer' which also meant 'crab', while the name for the tumor was transliterated into the Roman alphabet as 'carcinoma.' In recent times, the pronunciation of the second 'c' got 'mutated' to the 's' sound instead of the 'k' sound.

Today cancer is the second leading cause of death worldwide, after heart failure, and accounts for roughly 13% of all deaths. Contrary to what one may suppose, cancer occupies the second place even in developing countries. In the USA, about 1.5 million persons will be diagnosed with cancer in a year, while around 570,000 will die from it.

Over the years, quite substantial success has been realized in the treatment of *some forms* of cancer. This can be quantified by using the so-called five-year relative survival rate (RSR), which is defined as the ratio of the fraction of those with the disease condition that survive for five years, divided by the same number for the general population. To illustrate, suppose we start with a cohort of 1,000 persons. Assuming a mortality rate of 2% per year for the general population, after five years 900 of the original population will survive (rounding off the numbers for illustrative purposes). Now suppose that amongst a cohort of 1,000 persons with a particular form of cancer, only 360 survive for five years. Then the five-year RSR is $360/900 = 40$ per cent.

The table below shows the RSR for various forms of cancer, in 1954 and in 2006. This and other information can be found on the web sites of the National Cancer Institute, specifically [5].

Primary Site	**5-Year RSR 1950–1954**	**5-Year RSR 1999–2006**
All sites	35	69.1
Childhood	20	82.9
Leukemia	10	56.2
Hodgkin lymphoma	30	87.7
Breast	60	91.2
Prostate	43	99.9
Pancreas	1	5.8
Liver	1	13.7
Lung	6	16.8

Table 1. Relative Survival Rates Over the Years

From this table, it can be seen that significant progress has been made in some forms of cancer. For instance, the RSR in Leukemia has gone up from a mere 10% to more than 50%. Prostate cancer is virtually a non-disease, as the RSR is close to 100%. On the other hand, in some forms of cancer, such as pancreas, liver, and lung, the RSR figures have remained stubbornly stuck at very low levels. Not surprisingly, these diseases form a primary focus of cancer studies.

C. Advances in Data Generation

In recent years, rapid advances in experimental methods have enabled the biologist community to amass truly vast amounts of data of various types. This data can be divided into two broad categories, namely molecular and clinical. Some of these data types that are most relevant to the present discussion are described next.

- **DNA Sequencing:** DNA stands for Deoxyribonucleic acid, and is the fundamental building block of life. For present purposes, one can think of the DNA of an organism (including humans) as just an enormously long string over the four-symbol alphabet $\{A, C, G, T\}$, where the letters represent the bases of the four nucleotides: A for Adenine, C for Cytosine, G for Guanine and T for Thymine. Thus the genome of an organism is its 'digital' description at the most basic level. Genes are the 'operative' part of the DNA that produce proteins and thus sustain life. When the first 'complete' human genome, consisting of nearly 3.3 billion base pairs[2] was published in 2001 [6], [7], it cost more than $3 billion and took several years; on top of that, it was only a 'draft' in that its error rate was roughly 2%. Today there are commercial companies that promise to sequence a complete human genome, or sell the equipment to do so, at a cost of $1,000 or so per genome. Even allowing for the ever-present hype in the biotechnology industry, this is an impressive reduction of several orders of magnitude in both the cost and the time needed. Quite apart from sequencing entire genomes, it is now feasible to sequence literally tens of thousands of cancer tissues that are available at various research laboratories. The National Institutes of Health (NIH) has embarked upon a very ambitious project called TCGA (The Cancer Genome Atlas) whose ultimate aim is to publish the DNA sequence of every cancerous tissue that is available to it [8]. By comparing (wherever possible) the DNA sequence of the normal tissue of the same individual, it is possible to isolate many mutations (referred to as polymorphisms) that accompany the onset and growth of cancer. By studying the consequences (phenotypes) of these polymorphisms, one could in principle be able to formulate predictive models for cancer growth.
- **Gene Expression Profiling:** This refers to measuring the activity level of various genes under specified experimental conditions. The experiment consists of measuring the quantum of gene product produced, where the gene products could be 'final' products such as proteins, or 'intermediate' products such as mRNA (messenger RNA), RNAi (RNA interference), etc. Often the genes are subjected to external influences (see the next two items) and the objective of the study is to quantify the effect of these influences. In general, it is very difficult to replicate these experiments, as conditions are variable from one experiment to another. Thus gene expression profiling experiments almost always have some 'control' genes whose expression levels are expected to remain constant across experiments; these values are then used to normalize the rest of the measured quanti-

[2]The phrase 'base pair' refers to the fact that DNA consists of two strands running in opposite directions, and that the two strands have 'reverse complementarity' – A occurs opposite T and C occurs opposite G.

ties. Moreover, actually taking measurements is a highly invasive activity and usually results in the termimnation of the experiment.[3] This is in sharp contrast to the situation in engineering systems, where most of the time it is possible to perform noninvasive measurements. As a consequence, 'temporal gene expression profiles' are in reality a set of ostensibly identical experimets, that are terminated in a staggered fashion at different points in time. In the author's view, since no two experiments are ever identical or close to it, it is rather problematical to treat the outcome of 'temporal gene expression profiles' as a time series and fit ODE models to the data. However, this does not prevent some researchers from doing it anyway.

- **siRNA Experimentation:** When cells reproduce, DNA gets converted to RNA (Ribonucleic acid) which in turn produces any of the roughly 100,000 proteins that sustain life. Unlike DNA which is a chemically stable molecule, RNA is somewhat unstable and can be thought of as an intermediary stage. The conversion of DNA to RNA (transcription) and of RNA to proteins (translation) is usually referred to as the 'central dogma' of biology. siRNA stands for 'small interfering RNA' (the expansion 'silencing RNA' is also used). siRNAs are small double-stranded RNA molecules, just 20-25 nucleotides long, that play a variety of roles in biology. For our purposes, the most important role is that each siRNA gets involved in the RNAi (RNA interference) pathway, and interferes with the expression of a specific gene. While originally siRNAs were naturally occurring, nowadays there are more than 21,000 siRNA molecules, many of them synthetically created, each of which silences the functioning of one specific gene. Thus, for example, one can take a cancerous cell line that is kept alive in a laboratory ('immortalized'), apply a specific siRNA, and see whether or not the application of the siRNA causes the cell line to die out. If the answer is 'yes', then we conclude that the gene which is silenced by that specific siRNA plays a key role in the reproduction of the cancerous cell.
- **Micro-RNA Experimentation:** Micro-RNAs are relatively short RNA molecules, roughly 20 nucleotides long, that bind to messenger RNA and inhibit some part of the translation aspect. At present there are about known 1,500 micro-RNAs. As a gross over-simplification, it can be said that each micro-RNA inhibits the functioning of more than one gene, while each gene is inhibited by more than one micro-RNA. A description of micro-RNAs and their functioning can be found in [10], [11], [12], [13]. An attempt to quantify the impact of each micro-RNA on the functioning of various genes is found in the program 'Targetscan', which is described in [14].

[3]It should be noted in passing that several attempts have been made to overcome this problem, e.g. to use green fluorescent protein as a marker for gene expression [9].

D. Ways in Which Controls Community Can Contribute

In this subsection we present a broad philosophical discussion of how the controls community can contribute to cancer research. A more mathematical discussion can be found in the concluding section.

It is a truism that biology is in some sense far more complex than engineering. In engineering, one first designs a system that performs satisfactorily, and then improves the design to be optimal (or nearly so), and finally, replicates the designed system as accurately as possible. In contrast, in biology, there is no standardization. Each of the 7 billion humans differ from each other in quite significant ways – clearly we are not mass-produced from a common template. Even if we focus on components of the human body and try to understand how they work together for a common purpose, there are difficulties. In designing complex engineering systems, each subsystem is designed separately, often by a dedicated design team. Then the subsystems are connected through appropriate isolators that ensure that, even after the various subsystems are interconnected, each subsystem still behaves as it was designed to. In contrast, in biology, it is very difficult if not impossible to isolate individual subsystems and analyze their behavior. Even if one could succeed in understanding how a particular subsystem would behave in isolation, the behavior of the same subsystem gets altered significantly when it is a part of a larger system. Isolation amongst subsystems is not a common feature of biology.

Because of these considerations, it is difficult for control theorists to make an impact on biology unless they work very closely with experimental biologists. In a well-established subject like aerodynamics (to pick one), the fundamental principles are known, and captured by the Navier-Stokes equation. Thus it is possible for an engineer to 'predict' how an airframe would behave to a very high degree of accuracy before metal is ever cut. In the author's view, given the lack of foundational principles for the most part, in biology control theorists must settle for a more modest role, namely 'generating plausible hypotheses' as opposed to 'making predictions'. These plausible hypotheses are then validated or invalidated by experimentation. Learning is inductive: If a hypothesis is invalidated through experiment, then the modeling paradigm used to arrive at that hypothesis must be discarded; however, a confirmation of the hypothesis through experiment can serve only to increase one's confidence in the modeling paradigm.

In order to describe specific ways in which the controls community can contribute to cancer therapy, we begin with a very high-level of how cancer treatment is approached today. Because of the need to explain to a non-specialist readership, over-simplification is unavoidable, and the reader is cautioned that the description below is only 'probably approximately correct'. Those desirous of getting a more accurate picture should study the biology literature.

In the human body, cells die and are born all the time, and a rough parity is maintained between the two processes. Occasionally, in response to external stimuli, one or the other

process gains the upper hand for a short period of time, and in a localized manner. For instance, if one gets a wound, then a scab forms to protect the wound, and then the scab itself falls off when its function is completed. In the process of cell division and DNA replication, errors do occur. However, there is a fairly robust DNA repair process that corrects the errors made during replication. In spite of this, it is possible that some mutations that occurred during DNA replication do not get corrected, but instead get passed on to the next generation and the next after that; these are called somatic mutations. If these mutated cells replicate at a faster rate than normal cells, then it is possible (though not inevitable) that eventually the mutated cells overwhelm the normal cells by grabbing the resources needed for replication. At this point the cell growth, or tumor, has gone from being benign to being malignant. If the products of the mutated DNA enter the blood stream, or the lymph system, then the mutations can then be replicated at locations that are far-removed from the site of the original mutation; this is known as metastasis.

One of the complicating factors of cancer is that, in contrast to other diseases, every manifestation of the disease is in some sense unique.[4] Hence some sort of 'personal medicine' is not only desirable, but is in some sense mandatory. Fortunately, thanks to all the advances cited in Section I-C, there is now a tremendous opportunity for cancer therapy to aspire to precisely this. Specifically, by analyzing the vast amount of molecular and clinical data that is becoming available, cancer therapists can aspire to provide prognostic information and a selection of therapies that would most benefit that particular patient. The flip side is that the availability of enormous amounts of data poses its own challenges. While there are many possible ways to exploit the flood of data, in this paper we begin focusing on approaches based on identifying the genetic regulatory networks (GRNs)[5] that would have gone awry to cause the cancer. Then we discuss two other topics.

In this GRN-based approach, cancer therapists would proceed roughly as follows:

- Identify either a 'consensus' GRN describes a cross-section of the population, or a 'personal' GRN that describes the particular patient under normal conditions. Then, compare the GRN that governs the cancerous tissue, and see how it differs from the normal (consensus or personal) GRN.
- Take a large number of cancer patients that are afflicted by a particular condition, then group them in such a way that the variation of GRNs within each group is a minimum, while at the same time the variation between groups is maxium.
- Using a combination of machine-learning (or statistical) and experimental methods, predict which treatment regimen is likely to be most effective for a particular group of patients.

So what can the controls community contribute within this broad framework? In one phrase, hypothesis generation and validation. Specifically, the community can

- Integrate available data in a rational manner that would permit the generation of all possible hypotheses about therapeutic interventions.
- When the biologists come up with some hypotheses, exclude those hypotheses that are inconsistent with the data, and rank those that are consistent in terms of their statistical significance.
- In a *suo motu* fashion, generate hypotheses that are suggested by the data, which the biologists can then validate.
- As experiments are performed to test various hypotheses, it is inevitable that there will be mismatches between the statistical predictions and the actual experimental outcomes. When this happens, the statistical models must be recalibrated to take into account the new data.

In short, by entering into a partnership with the biologists' community, the controls community can create a 'virtuous cycle' that would benefit both groups.

E. Organization of the Paper

Four specific problems are discussed here, namely:

- Reverse engineering gene regulatory networks (GRNs)
- Constructing context-specific gene regulatory networks
- Analyzing the significance of variations from gene expression studies
- Discriminating between drivers (mutations that cause cancer) and passengers (mutations that are caused by cancer or have no impact).

Out of these four topics, the first three use a fairly homogenous set of ideas from probability and statistics, such as Markov chains, graphical models, goodness of fit tests etc. The fourth topic involves only clustering, which, though it has some probabilistic foundations, is less 'deep' than the first three. It is included here for a very good reason. The discussion on the first three topics is slightly futuristic in the sense that, while some successes have been claimed in the literature, these are only suggestive of future applicability to cancer biology, and not definitive indications. In contrast, in the case of the fourth topic, some success has already been realized in the sense that several genes that were identified as possible drivers of colorectal cancer have already been found to play a role in other forms of cancer. The fact that these genes were identified only by clustering the so-called 'developmental gene expression profile' suggests a possible connection between this profile and the role of that gene (if any) in being a driver of cancer. Since our ultimate aim is to assist cancer biologists to address the challenges they face, this small 'success story' has been considered worth reporting even if the underlying theory is not very difficult.

II. Inferring Genetic Regulatory Networks

A. Problem Formulation

A gene regulatory network (GRN) is defined as a collection of genes or gene products in a cell that interact

[4]One could paraphrase the opening sentence of Leo Tolstoy's *Anna Karenina* and say that 'Normal cells are all alike; every malignant cell is malignant in its own way'.

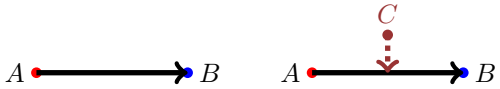
[5]This term is defined precisely later on.

with each other. The problem of inferring a GRN is that of reconstructing (or at least making a good model of) the GRN from experimental data. One of the main motivations for inferring GRNs from data is very nicely spelled out in the perspective paper [15]:

> "In the end, a good model of biological networks should be able to predict the behavior of the network under different conditions and perturbations and, ideally, even help us to engineer a desired response. For example, where in the molecular network of a tumor should we perturb with drug to reduce tumor proliferation or metastasis? Such a global understanding of networks can have transformative value, allowing biologists to dissect out the pathways that go awry in disease and then identify optimal therapeutic strategies for controlling them."

One can divide GRN models into two classes: static and dynamic. Dynamic GRN models usually consist of a system of ordinary differential equations (ODEs). See [16] and the references therein for exemplars of such an approach. Obviously, in order to generate such models, the experimental data must itself be temporally labeled. As stated earlier, 'temporal' gene expression data is in reality a collection of ostensibly identical experiments terminated in a staggered fashion at different points of time. In the author's opinion, such data is often not reliable enough to permit the construction of accurate temporal models, unless the models are particularly simple. For this reason, the discussion below is focused on static GRNs, where all quantities are in the steady-state. The paper [15] presents a set of three 'principles' and six 'strategies' for developing network models in cancer. The paper is well worth reading in its entirety. However, we note that Principle 1 is 'Molecular influences generate statistical relations in data', while Strategy 3 is 'Statistical identification of dysregulated genes and their regulators'. Given the scope of the present paper, the discussion below is guided by these two observations.

By far the most popular models of (static) GRNs are graphical, where the nodes represent individual genes or individual gene products. There are only two kinds of edges, referred to as unmediated interactions and mediated interactions respectively, as shown below.



Unmediated Interaction Mediated Interaction

The above diagram shows only one single edge within a GRN. A complete GRN is usually extremely complicated, with possibly tens of thousands of nodes, and millions of edges, often resembling a 'spider's web'. Figure 2 shows a part of the GRN corresponding to B lymphocytes, showing all the nearest neighbors of the proto-oncogene MYC, together with *some* (not all) of the neighbors of the neighbors of MYC; the figure corresponds to [17, Figure 4]. We shall return to this example later.

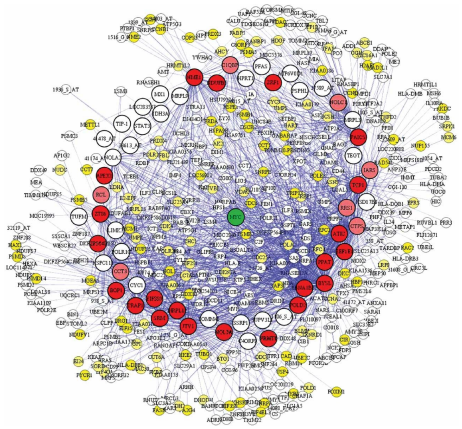


Fig. 2. The MYC Subnetwork [17, Figure 4]

GRNs have some very typical 'small world' features. For instance, simple arithmetic shows that with tens of thousands of nodes and millions of edges, the average connectivity of each node is in the double digit range. In reality however, the vast majority of nodes have connectivities in the single digit range, while a few nodes act as hubs and have connectivities in the high hundreds and possibly in the low thousands.

The problem at hand therefore is the reconstruction of a GRN on the basis of gene expression data, some (or most) of which could come from a public source such as the Gene Expression Omnibus (GEO) [18]. Even when the data has been painstakingly generated by personnel in some laboratory, the data is then immediately placed in GEO or another such publicly accessible source, so that the results can be verified by other research groups. The data would consist of expression levels of various gene products, obtained across multiple cell lines by various research teams (and all the lack of standardization that implies).[6] The data can be analyzed to study multiple genes or gene products in one cell line (lateral study), the same set of genes or gene products across multiple cell lines (longitudinal study), or both. In such studies, the number of gene products is often in the tens of thousands. However, the number of distinct cell lines rarely exceeds a few dozen, or a few hundred if one is extremely fortunate. Thus any statistical methodology must address this mismatch in dimension.

Another important aspect of the problem is that one rarely uses the 'raw' data coming out of experiments. As mentioned earlier, since biological experiments are not reproducible, every experiment includes some 'control' genes whose expression levels should be constant across experiments. Then the raw data from the various sets of experiments is normalized in such a way that the expression levels of the control genes is the same in all experiments. And then all the data is aggregated. Once this is done, the data for the remaining genes is 'smoothened' by centering, rescaling, linear to logarithmic transformation etc. The key point to note here is that each of these transformation is *one-to-*

[6]Note that the data for a single cell line could itself be a compendium of data obtained through multiple experiments carried out at different times.

one and therefore invertible. Often the transformation is also *monotone*, in that it preserves the linear ordering of real numbers. The smoothened data then forms the input to the inference problem described next.

In order to use statistical methods, let us think of the expression levels of the various genes or gene products as random variables $X_1, \ldots, X_n$, and the available data as consisting of samples $x_{ij}, i = 1, \ldots, n, j = 1, \ldots, m$. With this biological background, one can formally state the problem at hand.

Problem: Given n random variables $X_1, \ldots, X_n$ and samples $x_{ij}, i = 1, \ldots, n, j = 1, \ldots, m$, where $m \ll n$, compute the joint distribution function of the n random variables on the basis of the available data. Moreover, any technique used must have the feature of 'invariance under invertible transformations'. In other words, for any set of invertible functions $\eta_i : \mathbb{R} \to \mathbb{R}, i = 1, \ldots, n$, the technique applied to the data set $\{\eta_i(x_{ij})\}$ should produce the same result as applied to the data set $\{x_{ij}\}$.

The number of random variables n is far larger than the number of samples m and is likely to remain so for the foreseeable future. Thus, in any reasonable statistical sense, it is clearly impossible to infer the joint distribution of all n random variables, *unless one imposes some assumptions* on the nature of the joint distribution. The two specific techniques described below, namely the Markov random field approach and the Bayesian network approach, are distinguished by the assumptions they impose. It must be emphasized that the assumptions are imposed not so much because they are justified by biological realism, and more because they facilitate statistical analysis.

Irrespective of the assumptions made and the techniques used, the ultimate objective of statistical methods for inferring GRNs is to unearth dependences amongst various random variables. To put it another way, the aim is not so much to find a very precise formula for the joint distribution of the n random variables, but rather to identify whether one random variable X_i is influenced by another X_j. Let us now attempt to make precise this notion of 'being influenced'. At a very basic level, one could say that X_i is influenced by X_j if the two random variables X_i and X_j are not independent. But this is a very crude definition, so let us attempt to refine it. Suppose X_i is indeed influenced by X_j in the sense that X_i and X_j are not independent. The next level question one can ask is whether the influence is direct or indirect. In other words, is it the case that

$$\Pr\{X_i | X_k, k \neq i\} = \Pr\{X_i | X_k, k \neq i, k \neq j\}? \quad \text{(II.1)}$$

The above equation means that the conditional distribution of X_i given all other random variables $X_k, k \neq i$, is exactly the same as the conditional distribution of X_i given all random variables X_k other than X_j. So if the above equation holds, then it means that, while X_j does indeed influence X_i, the influence is indirect. For instance, suppose k is some index and that X_i, X_j are conditionally independent given X_k. Then X_j influences X_k which in turn influences X_i, but X_j does not 'directly' influence X_i. On the other hand, if (II.1) does not hold, then one can claim that X_j 'directly' influences X_i. These kinds of hypotheses can then be tested in experiments (and validated or invalidated).

B. Methods Based on Mutual Information

One way to approach the issue of whether X_j influences X_i is to compute their mutual information. Let us switch notation and suppose that X, Y are random variables assuming values in finite sets $\mathbb{A}, \mathbb{B}$ respectively. Let $\boldsymbol{\mu}, \boldsymbol{\nu}, \boldsymbol{\theta}$ denote the distribution of X, the distribution of Y, and the joint distribution of X and Y, respectively. Then the quantity

$$H(X) = H(\boldsymbol{\mu}) = -\sum_{i \in \mathbb{A}} \mu_i \log \mu_i$$

is called the **Shannon entropy** of X or $\boldsymbol{\mu}$,[7] while

$$I(X, Y) = H(X) + H(Y) - H(X, Y)$$

is called the **mutual information** between X and Y. An equivalent formula is

$$I(X, Y) = \sum_{i \in \mathbb{A}} \sum_{j \in \mathbb{B}} \theta_{ij} \log \frac{\theta_{ij}}{\mu_i \nu_j}.$$

Note that mutual information is symmetric: $I(X, Y) = I(Y, X)$. Also, $I(X, Y) = 0$ if and only if X, Y are independent random variables. Finally, if $f : \mathbb{A} \to \mathbb{A}', g : \mathbb{B} \to \mathbb{B}'$ are one-to-one and onto maps then $I(f(X), g(Y)) = I(X, Y)$. Thus in particular, monotone maps of random variables leave the entropy and mutual information invariant.

One of the first attempts to use mutual information to construct GRNs is in [19], which introduces 'influence networks.' In this approach, given m samples each for n random variables X_1 through X_n, one first computes the pairwise mutual information $I(X_i, X_j)$ for all $i, j, j \neq i$, that is, $n(n-1)/2$ pairwise mutual informations. Then X_i and X_j are said to influence each other if the computed $I(X_i, X_j)$ exceeds a certain threshold. Note that, since mutual information is symmetric, in case $I(X_i, X_j)$ does exceed the threshold, all one can say is that X_j influences X_i, or vice versa, or perhaps both. In other words, it is not possible to infer any 'directionality' to the influence if one uses mutual information (or for that matter any other symmetric quantity) to infer dependence. Another detail is note is that in fact one cannot compute the 'true' mutual information because one does not know the true joint distributions of X_i, X_j. Instead, one has to compute an 'empirical' approximation to $I(X_i, X_j)$ on the basis of the samples. In [19], this is done by grouping the observed expression levels into ten histograms, thus effectively quantizing each random variable into one of ten bins. This was possible in [19] because they were fortunate enough to have 79 samples. In cases where the number of samples is smaller, one would obviously have to use fewer bins.

The major drawback of the influence networks approach proposed in [19] is that it is not able to discriminate between direct and indirect influence. As a result, the influence

[7] We make no distinction between the entropy of a probability distribution $\boldsymbol{\mu}$ and the entropy of a random variable X having the probability distribution $\boldsymbol{\mu}$.

network constructed using mutual information *fails* to have an edge between nodes i and j if and only if X_i and X_j are independent (or if one uses empirically computed estimates for the mutual information and a threshold, nearly independent). To get more meaningful results, it is necessary to prune this first-cut influence network by deleting an edge between nodes i and j if the influence is indirect, that is, if (II.1) holds.

To achieve this objective, an algorithm called ARACNE is proposed in [20]. The basis of this algorithm is the assumption that the joint probability distribution of all n variables factors into a product of terms involving at most two variables at a time. This special feature makes it possible to invoke a bound known as the data processing inequality to prune the first-cut influence network.

Now we describe the ARACNE algorithm.[8] To make the ideas clear, let us suppose that the random variable X_i assumes values in a finite alphabet $\mathbb{A}_i$, which can depend on i. Define $\mathbb{A} = \prod_{i=1}^{n} \mathbb{A}_i$, and let $\mathbf{x}$ denote the n-tuple $(x_1, \ldots, x_n) \in \mathbb{A}$. Similarly let $\mathbf{X}$ denote $(X_1, \ldots, X_n)$. Then the joint distribution of all n random variables is the function $\phi : \mathbb{A} \to [0, 1]$ defined by

$$\phi(\mathbf{x}) = \Pr\{\mathbf{X} = \mathbf{x}\}. \tag{II.2}$$

Now let $\mathcal{N} = \{1, \ldots, n\}$, and let

$$\mathcal{D} = \{(i, j) : 1 \leq i < j \leq n\}.$$

Then the assumption that underlies the ARACNE algorithm is that the function ϕ has the form

$$\phi(\mathbf{x}) = \frac{1}{Z} \prod_{i \in \mathcal{N}} \psi_i(x_i) \cdot \prod_{(i,j) \in \mathcal{D}} \phi_{ij}(x_i, x_j), \tag{II.3}$$

where

$$Z = \sum_{\mathbf{x} \in \mathbb{A}} \left[\prod_{i \in \mathcal{N}} \psi_i(x_i) \cdot \prod_{(i,j) \in \mathcal{D}} \phi_{ij}(x_i, x_j) \right]$$

is a normalizing constant. Note that in the statistical mechanics terminology employed in [20], the quantity $\log \phi(\cdot)$ is called the 'Hamitonian,' and the assumption is that the Hamiltonian is the *sum* of terms involving only individual x_i, or pairs (x_i, x_j), but no higher order terms.

Suppose we associate an undirected graph with the distribution in (II.3) by inserting an edge[9] between nodes i and j if the function ϕ_{ij} is not identically zero. In the worst case, if every such function is not identically zero, we would wind up with a complete graph with n nodes, where every node is connected to every other node. This is clearly not desirable. So the authors of [20] set out to find a simpler representation of the data than a complete graph. In doing so, they build upon the work of [21], where the objective is to find the best possible approximation to a given probability distribution $\phi(\cdot)$ (not necessarily of the form (II.3)) in terms of a distribution of the form (II.3) where $\phi_{ij} \not\equiv 0$ for exactly $n - 1$ pairs. The criterion used to define 'best possible' is the relative entropy or the Kullback-Leibler divergence [22, p. 19]. Specifically, if ϕ is the original distribution and θ is its approximation, then the quantity to be minimized is

$$H(\phi \| \theta) = \sum_{\mathbf{x}} \phi(\mathbf{x}) \log \frac{\phi(\mathbf{x})}{\theta(\mathbf{x})}.$$

This problem has a very elegant solution, as shown in [21]. Starting with the given distribution ϕ, first compute all $n(n-1)/2$ pairwise mutual informations $I(X_i, X_j), j \neq i$. Then sort them in decreasing order. Suppose $I(X_{i_1}, X_{i_2})$ is the largest; then place an edge between nodes i_1 and i_2. Suppose $I(X_{i_3}, X_{i_4})$ is the next largest. Then create an edge between nodes i_3 and i_4. In general, at step k, suppose $I(X_{i_{2k-1}}, X_{i_{2k}})$ is the k-th largest mutual information. Then create an edge between nodes i_{2k-1} and i_{2k}, unless doing so would create a loop; in the latter case, go on to the next largest mutual information. Do this precisely $n - 1$ times. The result is a graph with n nodes, $n - 1$ edges, and no cycles – in other words, a tree.

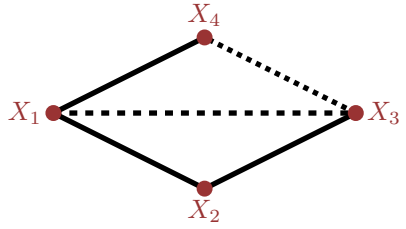
The authors of [20] build upon this approach by invoking the following result. If X_i, X_j are conditionally independent given X_k, then the so-called 'data processing inequality' [22, p. 35] states that

$$I(X_i, X_j) \leq \min\{I(X_i, X_k), I(X_j, X_k)\}. \tag{II.4}$$

Accordingly, the ARACNE algorithm initially constructs an influence network as in [19]. Then for each triplet (i, j, k) of pairwise distinct indices, the three quantities $I(X_i, X_j), I(X_i, X_k), I(X_j, X_k)$ are compared; the smallest among the three is deemed to arise from an indirect interaction, and the corresponding edge is deleted.

From the above description, it is easy to deduce the following fact: A network produced by the ARACNE algorithm will never contain a complete subgraph with three nodes. In other words, if there exist edges between nodes i and j, and between nodes j and k, then there will never be an edge between nodes i and k. From the standpoint of biology, this means that if gene i influences (or is influenced by) two other genes j and k, then perforce genes j and k must be conditionally independent given the activity level of gene i.

Note that the network that results from applying the ARACNE algorithm does not depend on where we start the pruning. To illustrate, consider a very simple-minded network with four nodes as shown below.



Suppose that

$$I(X_1, X_3) \leq \min\{I(X_1, X_2), I(X_2, X_3)\}.$$

[8] Note that language used here is not identical to that in [20] but is mathematically equivalent.

[9] Note that since the graph is undirected, it is not necessary to specify the direction.

In accordance with the algorithm, the link from X_1 to X_3 is discarded (and thus shown as a dashed line). Now suppose in addition that

$$I(X_3, X_4) \leq \min\{I(X_1, X_3), I(X_1, X_4)\}.$$

Then the edge from X_3 to X_4 is also deleted. It is easy to verify that if we had examined the triplets in the opposite order we would still end with the same final graph.

The ARACNE algorithm has been applied to the problem of reverse-engineering regulatory networks of human B cells in [17]. A total of 336 gene expression profiles for 9,563 genes were used. Only about 6,000 genes had sufficient variation in expression levels to permit the computation of mutual information. To illustrate the network that results from applying the algorithm, the authors depict how it looks in the vicinity of the proto-oncogene MYC.[10] The ARACNE algorithm showed that MYC had 56 nearest neighbors, and thes 56 neighbors had 2,007 other genes that were not neighbors of MYC. Thus at a distance of two steps, MYC contained more than 2,000 of the roughly 6,000 genes in the network. The overall network had about 129,000 interactions (edges), or about 20 per node on average. However, just 5% of the 6,000 nodes accounted for 50,000 edges, or about 40% of the total, thus demonstrating the 'small world' nature of the GRN that results from the algorithm. Figure 2 shows the 56 neighbors and another 444 most significant second neighbors of MYC.

Thus far the methods described generate GRNs with only unmediated edges. To construct GRNs with mediated edges, one follows the same approach as in ARACNE, except that instead of using the mutual information $I(X_i, X_j)$, one uses the conditional mutual information $I(X_i|X_l, X_j|X_l)$. Since the conditional mutual information also satisfies a data processing inequality of the form (II.4), the same reasoning can be applied to prune an initially overly dense network. This algorithm, based on conditional mutual information, is referred to as MINDy and is proposed in [24]. An essentially similar algorithm is proposed in [25].

In either ARACNE or MINDy, it is obvious that the most time-consuming step is the computation of all pairwise mutual informations. In [20], the authors take the given samples, and then fit them with a two-dimensional Gausian kernel for each pair of random variables. Then a copula transform is applied so that the sample space is the unit square, and the marginal probability distribution of each random variable is the uniform distribution.[11] In [28], a window-based approach is presented for computing pairwise mutual information that is claimed to result in roughly an order of magnitude reduction in the computational effort. For instance, for the B lymphocyte network studied in [17], the original ARACNE computation is claimed to take 142 hours of computation, while the method proposed in [28] is claimed to take only 23 hours.[12]

[10]Medterms [23] defines a proto-oncogene as "A normal gene which, when altered by mutation, becomes an oncogene that can contribute to cancer," and an oncogene as "A gene that played a normal role in the cell as a proto-oncogene and that has been altered by mutation and now may contribute to the growth of a tumor."

[11]The notion of a copula was introduced in [26]. See [27] for an excellent introduction to the topic.

C. Methods Based on Bayesian Networks

In this section we discuss the Bayesian network-based approach to inferring GRNs. Bayesian networks have been used in artificial intelligence for many decades, and [30] is the classic reference for that particular application. The Bayesian approach to inferring GRNs appears to have been pioneered in [31]. This was followed up by other work [32] and a survey is given in [33].

As before, the problem is to infer the joint distribution of n random variables $X_1, \ldots, X_n$, based on m independent samples of each random variable. For any set of random variables, it is possible to write their joint distribution as a product of conditional distributions. For two variables X_1, X_2, we can write

$$\Pr\{X_1, X_2\} = \Pr\{X_1\} \cdot \Pr\{X_2|X_1\},$$

and we can also write

$$\Pr\{X_1, X_2\} = \Pr\{X_2\} \cdot \Pr\{X_1|X_2\}.$$

If there are n random variables $X_1, \ldots, X_n$, then we can write

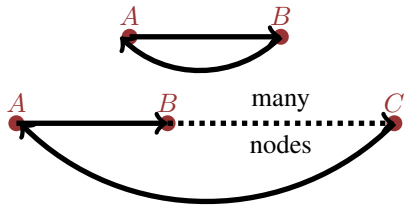
$$\Pr\{\mathbf{X}\} = \prod_{i=1}^{n} \Pr\{X_i|X_j, 1 \leq j \leq i-1\},$$

where $\mathbf{X}$ denotes $(X_1, \ldots, X_n)$. More generally, let π be any permutation on $\{1, \ldots, n\}$. Then we can also write

$$\Pr\{\mathbf{X}\} = \prod_{i=1}^{n} \Pr\{X_{\pi(i)}|X_{\pi(1),\ldots,\pi(i-1)}\}. \tag{II.5}$$

Since the above expression is valid for *every* permutation π, we should choose to order the variables in such a way that the various conditional probabilities become as simple as possible. In essence, this is the basic idea behind Bayesian networks.

Suppose now that $\mathcal{G}$ is an acyclic directed graph with n vertices. Note the total contrast with the assumptions in methods based on mutual information. In that setting, $\mathcal{G}$ is an undirected graph, so that edges can be thought of as being bidirectional. In the present setting, not only are edges unidirectional, but no cycles are permitted. In other words, both the situations shown below are ruled out in the Bayesian network paradigm.



[12]It is interesting to note that in a preprint version of [28], their method is claimed to take only 1.6 hours.

Let us think of an edge (i, j) as being *from* node i *to* node j, and let $\mathcal{E}$ denote the set of edges in $\mathcal{G}$. Since the graph is assumed to be acyclic, with each node i we can unambiguously associate its ancestor set $A(i)$ and its successor set $S(i)$ defined by

$$A(i) = \{k : (k, i) \in \mathcal{E}\}, S(i) = \{j : (i, j) \in \mathcal{E}\}. \quad \text{(II.6)}$$

Since the graph is both directed as well as acyclic, it is obvious that $A(i)$ and/or $S(i)$ may be empty for some indices i. In some circles, a node i is referred to as a 'source' if $A(i)$ is empty, and as a 'sink' if $S(i)$ is empty. Let us also adopt the notation $X_i \perp X_j$ if X_i and X_j are independent, and the notation $(X_i \perp X_j)|X_k$ if X_i and X_j are conditionally independent given X_k.

Definition 1: A set of random variables $X_1, \ldots, X_n$ is said to be a **Bayesian network** with respect to a directed acyclic graph $\mathcal{G}$ if

$$(X_i \perp X_j)|\{X_k, k \in A(i)\}, \ \forall j \notin S(i). \quad \text{(II.7)}$$

In words, a set of random variables $X_1, \ldots, X_n$ is a Bayesian network with respect to $\mathcal{G}$ if, for a fixed index i, the associated r.v. X_i is conditionally independent of X_j for all nonsuccessors X_j, given the values of X_k for all successors k of i. It is easy to see that if the set of random variables $X_1, \ldots, X_n$ forms a Bayesian network with respect to the directed acyclic graph $\mathcal{G}$, then the joint probability distribution factors as follows:

$$\Pr\{\mathbf{X}\} = \prod_{i=1}^{n} \Pr\{X_i|\{X_k, k \in A(i)\}\}, \quad \text{(II.8)}$$

where the conditional probability of X_i is taken to be the unconditional probability if the set $A(i)$ is empty. Compare (II.7) with (II.5).

The formula (II.8) demonstrates one of the main attractions of the Bayesian network model. For each source node i, the unconditional probability of X_i can be computed directly from the data. (It is obvious that if i, j are both source nodes, then $X_i \perp X_j$.) Then, using (II.8), the conditional probability computation of any intermediate X_i can be propagated along the graph. This is the feature that makes Bayesian networks so popular in AI circles.

The problem of modeling a set of expression data using a Bayesian network can be divided into two questions. First, what is the graph $\mathcal{G}$ that is used to model the data (i.e., the dependence structure among the random variables)? Second, once the graph $\mathcal{G}$ has been chosen, how can one find the best possible fit to the expression data by a suitable choice of the various conditional probabilities in (II.8)? In answering the second question, one again needs to make a distinction between parametric models, where the various conditional probabilities are specified as known functions of an unknown parameter $\theta \in \Theta$ where Θ is specified ahead of time, and nonparametric models in which case no such form is assumed. Strictly speaking, the classical Bayesian paradigm applies to the use of parametric models with the dependence structure specified beforehand. In such a case, it is assumed that the parameter θ has a known prior distribution, and that the data set, call it D, is generated using some unknown probability distribution. Then the parameter θ is chosen so as to maximize the posterior probability $\Pr\{\theta|D\}$. The Bayesian approach consists of observing that

$$\Pr\{\theta|D\} = \frac{\Pr\{D|\theta\} \cdot \Pr\{\theta\}}{\Pr\{D\}}.$$

Hence

$$\log \Pr\{\theta|D\} = \log Pr\{D|\theta\} + \log \Pr\{\theta\} - \log \Pr\{D\}.$$

In the above equation, $\Pr\{D\}$ can be treated as a constant, since it does not depend on θ. In principle, the same approach can also be extended to answer the first question as well, namely the choice of the directed graph $\mathcal{G}$ that is used to model the data. However, since the number of possible directed acyclic graphs in n nodes increases far too quickly with n, this approach may not be feasible, unless one restricts attention to a very small subset of all possible directed acyclic graphs on n nodes.

D. A Unified Interpretation

The two approaches described above can be put into some sort of common framework. Suppose $X_1, \ldots, X_n$ are random variables assuming values in finite sets $\mathbb{A}_1, \ldots, \mathbb{A}_n$ respectively. Let $\mathbf{X}$ denote $(X_1, \ldots, X_n)$, and let $\mathbb{A}$ denote $\prod_{i=1}^{n} \mathbb{A}_i$. Finally, let $\mathbf{x} \in \mathbb{A}$ denote a value that $\mathbf{X}$ can assume, and let, as before,

$$\phi(x) = \Pr\{\mathbf{X} = \mathbf{x}\}$$

denote the joint probability distribution. Then one can ask two specific questions: First, if $\phi(\mathbf{x})$ has certain product form, does this imply any kind of dependence structure on the random variables? Second, and conversely, if the random variables have some kind of dependence structure, does this imply that the joint distribution has a specific form? It turns out that the first question is very easy to answer, while the second one is more difficult.

Accordingly, suppose first that $\mathcal{G}$ is a graph with n nodes. For the moment we neither assume that the graph is symmetric nor that it is acyclic. It is a directed graph (unlike in ARACNE) and may contain cycles (unlike in the case of Bayesian networks). Let $\mathcal{N}$ denote $\{1, \ldots, n\}$, the set of nodes in the graph, and let $C_1, \ldots, C_k$ are subsets of $\mathcal{N}$ that together cover $\mathcal{N}$. In other words,

$$\bigcup_{l=1}^{k} C_l = \mathcal{N}.$$

Besides the covering property, no other assumptions are made about the nature of the C_l. For each C_l, define

$$X_{C_l} = \{X_j, j \in C_l), \mathbb{A}_{C_l} = \prod_{j \in C_l} \mathbb{A}_j.$$

The possible value $\mathbf{x}_{C_l} \in \mathbb{A}_{C_l}$ is defined analogously. Next, define

$$D(i) = \bigcup \{C_l : i \in C_l\}, S(i) = D(i) \setminus \{i\}.$$

Thus $D(i)$ consists of the union of all C_l that contain i. Note that, due to the covering property of the sets C_l, there

is at least one C_l that contains i, whence $D(i)$ is nonempty and contains i. Thus $S(i)$ is well-defined, though it could be empty. With these definitions, the following result is quite easy to prove.

Theorem 1: Suppose there exist functions $\phi_l, l = 1, \ldots, k$ such that

$$\phi(\mathbf{x}) = \frac{1}{Z} \prod_{l=1}^{k} \phi_l(\mathbf{x}_{C_l}), \tag{II.9}$$

where

$$Z = \sum_{\mathbf{x} \in \mathbb{A}} \prod_{l=1}^{k} \phi_l(\mathbf{x}_{C_l})$$

is a normalizing constant. Then

$$\Pr\{X_i | X_j, j \neq i\} = \Pr\{X_i | X_j, j \in S(i)\}. \tag{II.10}$$

An equivalent way of stating the theorem, which makes it resemble the definition of a Bayesian network is this: Suppose the joint distribution $\phi(\mathbf{x})$ can be factored as in Theorem 1. Then (II.7) is satisfied, with $S(i)$ now defined as above. Indeed the theorem is more or less a restatement of (II.5).

With suitable conventions, both the Bayesian network and the undirected graph can be put into the above dependence structure. However, the converse of the above theorem is false in general. Even if (II.10) holds, it does not readily follow that the joint distribution factors in the form (II.9). To obtain a proper converse, we introduce the notion of a Markov random field and present the Hammersley-Clifford theorem. Suppose as before that $X_1, \ldots, X_n$ are random variables assuming values in their respective finite alphabets (which need not be the same), and that $\mathcal{G}$ is an undirected graph with n nodes.[13] For each node i, let $N(i)$ denote the set of neighbors of i; thus $N(i)$ consists of all nodes j such that there is an edge between nodes i and j.

Definition 2: A set of random variables $X_1, \ldots, X_n$ is said to be a **Markov random field** with respect to a graph $\mathcal{G}$ with n nodes if

$$\Pr\{X_i | X_j, j \neq i\} = \Pr\{X_i | X_k, k \in N(i)\}, \ \forall i. \tag{II.11}$$

In words, a set of random variables $X_1, \ldots, X_n$ is a Markov random field with respect to $\mathcal{G}$ if and only if the conditional distribution of each random variable X_i depends only on its neighbors $X_k, k \in N(i)$.

A closely related notion is that of a Gibbs distribution. To define this notion, let us recall that a **clique** of an undirected graph is a maximal completely connected subgraph.

Definition 3: Suppose $X_1, \ldots, X_n$ are random variables and that $\mathcal{G}$ is an undirected graph with n nodes. Let $\mathcal{C}_1, \ldots, \mathcal{C}_k$ denote the cliques of $\mathcal{G}$. Then the random variables $X_1, \ldots, X_n$ are said to have a **Gibbs distribution** with respect to the graph $\mathcal{G}$ if their joint distribution ϕ satisfies

$$\phi(x_1, \ldots, x_n) = \prod_{l=1}^{k} \phi_l(x_j, j \in \mathcal{C}_l). \tag{II.12}$$

In words, the random variables $X_1, \ldots, X_n$ have a Gibbs distribution with respect to $\mathcal{G}$ if the joint distribution of all n variables factors as a product of simpler joint distributions, one for each clique of $\mathcal{G}$.

A fundamental result known as the Hammersley-Clifford theorem connects the two concepts.

Theorem 2: Suppose the joint distribution $\phi(x_1, \ldots, x_m)$ of a set of random variables is always strictly positive. Then they form a Markov random field if and only if the joint distribution is a Gibbs distribution.

Though this theorem is credited to Hammersley and Clifford, their original manuscript is somewhat inaccessible. A proof of this theorem can be found in [29] as well as several textbooks. Note that the proof in one direction is easy: If the joint distribution is Gibbs, then the random variables form a Markov random field, and one does not require the assumption that the joint distribution is positive in order to prove this. Therefore the real import of the theorem is in the opposite direction. To prove it in this direction, the strict positivity of the joint distribution as well as the finiteness of the alphabets in which each random variable assumes values are both essential requirements.

E. Evaluation and Validation of Competing Approaches

Given that the computational biology literature is full of various approaches for reverse-engineering GRNs, there is a lot of interest in assessing the relative performance of all the competing approaches. In the area of protein structure prediction based on the primary structure (i.e., the sequence of amino acids that constitute the protein), there is a well-established biennial competition known as CASP (Critical Assessment of Structure Prediction). In this competition, the organizers first determine the 3-D structure of a protein using x-ray crystallography or some other method, but do not share it with the community at large. Instead the community is challenged to 'predict' the structure, and the ones who come closest to the true structure are recognized as such. Perhaps drawing inspiration from this, the research community working in the area of inferring GRNs has a competition called DREAM (Dialog for Reverse Engineering Assessment and Methods). In the personal opinion of the author, the DREAM competition lacks the authenticity of the CASP competition, simply because in CASP there is an unambiguous, objective truth that everyone is striving to find, and against which any and all predictions can be compared. This is definitely not the case in DREAM. Rather, in the case of DREAM, synthetic data is generated using some model or combination of models. It should be clear that, given two algorithms, one can always generate data sets on which one algorithm outperforms the other, and other data sets on which the performance is reversed. Until and unless our knowledge of GRNs proceeds to a stage where at least a few GRNs are completely identified to constitute 'the truth' (as in CASP and protein structures), there is a danger that such competitions actually serve to confuse rather than to clarify. Again, this is the author's personal opinion.

[13] Now it is assumed that the graph is undirected.

III. Context-Specific Genomic Networks

A. An Approach to Personal Medicine

As pointed out earlier, cancer is a highly individualized disease. It is not merely that mutations in some part of the DNA cause cancer. It is also the case that mutations in other parts of the DNA have a huge impact on the responsiveness to a therapeutic regimen. Identifying which mutations cause/have caused cancer, which mutations may affect the efficacy of treatment, and tailoring the treatment appropriately, is the essence of personal medicine.

Out of the dozens of known instances, we cite just one by way of illustration [35]. The drug cetuximab is a monoclonal antibody directed against the epidermal growth factor receptor (EGFR), one of the more popular gene targets for cancer therapy. This drug is widely used as a treatment for advanced colorectal cancer, often after other forms of chemotherapy have failed. In the paper [35], the authors analyzed 394 samples of colorectal cancer to see whether they contained a mutation of the gene KRAS, which is often found to be mutated in various forms of cancer. Amongst the samples tested, 42.3% had at least one mutation in KRAS, while the rest were 'wild type'.[14] To paraphrase the findings of [35],

- Amongst the patients who were given best supportive care alone (i.e., no cetuximab), there was no significant difference between the survival of patients who had a KRAS mutation and those who did not.
- Amongst patients with wild-type KRAS tumors (meaning, no mutation in the KRAS gene), there was substantial improvement after treatment with cetuximab.
- Amongst patients with a KRAS mutation, there was no significant benefit to treatment with cetuximab.

To summarize quickly, a KRAS mutation does not affect survival prospects if colorectal cancer is left untreated. If a patient has a KRAS mutation, then cetuximab therapy is of no beneift, whereas a patient without a KRAS mutation derives significant benefit from a cetuximab treatment.

In the paper cited, the authors had a very specific hypothesis in mind, namely that KRAS mutations affected the response to cetuximab treatment. However, often the role of the computational biologists is to *generate* such hypotheses using the data at hand. This would entail examining the data at hand to examine not just one mutation (in this case KRAS) but multiple mutations, and assessing the significance of each possible combination of mutations. It is easy to see that if one examines k genes then there are 2^k possible states of mutations to be examined. With 400 patients (a large number in such studies), if one wishes to have an average of, say, 10 samples per state, then it is possible to examine at most $k = \lfloor \log_2(400/10) \rfloor = 5$ different genes at a time. When one undertakes very large studies involving siRNA knockdowns for example, it is not uncommon to have just a handful of samples, often in the single digits. Accordingly, the emphasis in this section is on methods that permit 'context-specific' genomic networks, one for each sample, that can perhaps be used to draw useful conclusions even when there are very few samples at hand.

[14]This means that the gene is not mutated.

B. Identification of Genomic Machines

The problem discussed in the previous section, namely inferring GRNs from gene expression data, presupposes that there is no prior knowledge about the structure of the GRN.[15] In order to achieve some kind of accuracy in reverse-engineering the GRN, one is forced to aggregate a very large number of gene expression profiles that provide the input data set to whatever algorithm is being used. The output of the algorithm applied to an agglomeration of multiple data sets can perhaps be referred to as a 'consensus' GRN.

However, such an *ab initio* approach is not always warranted. In reality the biology literature is full of experimental results that report the influence of one entity on another, or the presence of interactions between some biological entities (such as genes and gene products). Granted, this information is scattered throughout the literature, and each individual publication usually reveals just a very tiny bit of the overall GRN. However, there are commercial vendors who 'curate' the published literature and 'integrate' all the reported interactions into one or more giant pathway databases. Diligent researchers can add value to the commercial products, either by integrating several commercial databases and/or adding proprietary in-house data. Hence it is not unreasonable to suppose that there is available at least a first-cut approximation to the GRN. More interesting, at least to the present author, is the fact that unlike the networks studied in the earlier section, which are forced either to be undirected or acyclic (both quite unrealistic assumptions), the graphical representations of commercially available databases incorporate both directional edges and cycles. In this respect, they can perhaps be deemed to be more faithful representations of reality. Accordingly, in this section we examine a different problem to the one earlier, namely: Suppose one has available a graphical representation of which genes or gene products interact, and how; however, the strengths of the interactions are not always known. Now suppose some data is available in the form of gene expression data; how can one couple this additional information with the known (or at least, hypothesized) graphical representation to derive further insights? Since the graphical representation itself remains unchanged, and only the expression data changes, such networks are usually referred to as 'context-specific' genomic networks.

Accordingly, suppose one is given a directed graph $\mathcal{G}$ with n nodes, where each node corresponds to a gene or a gene product, and the edges represent some kind of consensus about which pairs of nodes interact, and if so in which direction. There is no restriction that the graph should be acyclic, but self-loops are not allowed. The network is modeled as a random walk on $\mathcal{G}$, that is, as a Markov process on a set of cardinality n. To describe the random walk

[15]other than the simplifying statistical assumptions of Markov random field, Bayesian network structure or the like, which are made for statistical convenience than biological realism

or Markov process, it is necessary to specify the transition probability f_{ij} defined as

$$f_{ij} = \Pr\{X_{t+1} = j | X_t = i\},$$

where $\{X_t\}$ is the Markov process. The matrix F is row-stochastic in the sense that $F\mathbf{e}_n = \mathbf{e}_n$, where $\mathbf{e}$ denotes a column vector of all 1's, and the subscript denotes its dimension.

We shall define F in stages by first defining another row-stochastic matrix P. In the absence of experimental data, one uses only the interaction pattern to determine the activity levels of a node. This is done by mimicking the well-known page rank algorithm [36]. In this algorithm, the nodes represent individual pages on the worldwide web, and each node is assigned a weight equal to the contribution from other nodes that point into it. In turn, the weight of each node is assumed to be distributed equally amongst all outgoing edges. It is easy to see that these two criteria result in the statement that the weight vector of all the nodes is the stationary distribution of a Markov chain where the state space is the set of nodes, and the transition probability p_{ij} is assumed to be equal along all outgoing edges. Thus

$$p_{ij} = \begin{cases} 1/|S(i)| & \text{if } j \in S(i), \\ 0 & \text{if } j \notin S(i) \end{cases}. \qquad \text{(III.1)}$$

The ultimate objective is to determine the stationary distribution of the Markov chain with transition matrix F, denoted by $\boldsymbol{\pi}$. The higher the value of π_i, the more active that node is taken to be.

However, if we directly set $F = P$ then two things can happen. First, the graph as a whole may not be connected. Second, there can be some nodes that do not have outgoing edges (so-called 'dangling' edges). In the first case, there may be more than one stationary distribution. In the second case, all the dangling nodes will become essential nodes while the rest will become inessential (see [37] for definitions of these terms, as well as for theorems about stationary distributions). In such a case, it is well-known [37] that any and all stationary distributions will be supported on only the set of essential, or dangling, nodes. To alleviate this difficulty, in [36] the original matrix P is augmented by a rank one matrix, in the following manner:

$$F = (1-q)P + (q/n)\mathbf{e}_n\mathbf{e}_n^T, \qquad \text{(III.2)}$$

where $q \in (0,1)$ is some arbitrarily chosen parameter. It is easy to verify that, since P is row-stochastic, so is F. In [36], the rank one correction is justified on the grounds that a person browsing a certain web page may 'jump' to an entirely unrelated web page, even if there is no direct link to that page. For instance, a person browsing a travel web site may suddenly jump to his/her bank's web site to see how much the bank balance is, to compare against the air fare. Unstated in this correction is the implicit assumption is that a 'jump' between any two pairs of nodes i and j is equally likely; we shall see shortly that this assumption is not justified in the case of biological networks. With this additional term, the matrix F is strictly positive, so that it has a unique strictly positive probability vector $\boldsymbol{\pi}$ such that $\boldsymbol{\pi} = \boldsymbol{\pi}F$. In the page rank algorithm, q is taken as 0.15, suggesting that a person browsing the web has roughly a 15% likelihood of jumping to an unconnected site.

In [38], the authors build on this idea, but with some differences. Unlike in the page rank algorithm, where the quantity of interest is the weight of a node (taken as π_i where $\boldsymbol{\pi}$ is the stationary distribution of the Markov chain), the emphasis in [38] is in identifying so-called 'genomic machines.' For the purposes of biology, a genomic machine is defined as a set of genes or gene products that work together to achieve a common purpose, even (or perhaps especially) if one does not know what this purpose might be. To make this qualitative statement precise, one first sets up a matrix P as in (III.1), and then the adjusted matrix F as in (III.2). Then one computes first the stationary distribution $\boldsymbol{\pi}$, and then the n^2-dimensional 'flow' vector $\boldsymbol{\mu}$ where

$$\mu_{ij} = \pi_i f_{ij}.$$

It is easy to see that in fact $\boldsymbol{\mu}$ is just the 'doublet frequency', or

$$\mu_{ij} = \Pr\{(X_t, X_{t+1}) = (i,j)\},$$

under steady state conditions. Now suppose a gene expression experiment takes place, in which node i has an expression value of w_i. Then the 'raw' transition probability p_{ij} is modified to

$$p_{ij}^{(r)} = \begin{cases} w_j/s_i & \text{if } j \in S(i), \\ 0 & \text{if } j \notin S(i) \end{cases},$$

where

$$s_i = \sum_{j \in S(i)} w_j.$$

It is easy to verify that the matrix $P^{(r)}$ is row-stochastic. The interpretation is that the probability of moving from node i to node j is proportional to the weight of node j, and the division by s_i serves to normalize the transitional probabilities so that they add up to one. As before, to cope with the possibility of dangling edges, the raw transition probability matrix $P^{(r)}$ is perturbed to

$$F^{(r)} = (1-q)P^{(r)} + (q/n)\mathbf{e}_n\mathbf{e}_n^T.$$

For this new Markov chain, one again computes the stationary distribution $\boldsymbol{\pi}^{(r)}$ and doublet frequency distribution $\boldsymbol{\mu}^{(r)}$. Note that if all expression weights w_i are equal, then $P^{(r)}$ and $F^{(r)}$ reduce respectively to P and F respectively.

The next step is to see which nodes are seen to be more active as a result of the gene expression experiment. For this purpose, a 'figure of merit' r_{ij} is defined for each edge as

$$r_{ij} = \log \frac{\mu_{ij}^{(r)}}{\mu_{ij}}. \qquad \text{(III.3)}$$

Thus $r_{ij} > 0$ if the flow along an edge is increased as a consequence of the gene expression experiment. Next, if one can find a cycle $\{i_0, i_1, \ldots, i_k = i_0\}$ such that $r_{i_j i_{j+1}} > 0$ for all $j = 0, \ldots, k-1$, then this set of nodes is thought of as having a common purpose – in other words, a genomic machine. Similarly, if another cycle can be found where the

figure of merit for each edge is less than one, then that set of nodes can also be thought of as a genomic machine. On the other hand, cycles consisting of edges where the figure of merit is sometimes positive and at other times negative are thought not to have any significance in this approach. As a result of this exercise, from the set of notes $\{1, \ldots, n\}$ one will be able to identify several subsets $\mathcal{S}_1, \ldots, \mathcal{S}_K$ of genes that work together in a concerted fashion and are thus genomic machines. Note that these gene sets are not necessarily disjoint. Indeed it would be natural for certain key genes to participate in multiple genomic machines.

One of the advantages of the above approach is that it is not limited to just one set of gene expression data. Suppose, as often happens, that one has a very small number cell lines, all belonging to the same form of cancer, and that gene expression studies have been carried out all of these. Then the available data consists of a set of weights $\{w_i^l, i = 1, \ldots, n, l = 1, \ldots, k\}$, where n is the number of nodes in the graph, which is typically 20,000 to 30,000 genes or gene products, and k is the number of cell lines, often of the order of a dozen or so. Now, using the accepted interactions among the nodes as captured by the graph $\mathcal{G}$, one can construct the initial flow vector μ_{ij}. The next step is to average the expression data among all cell lines to arrive at a 'consensus' set of weights for the overall expression study. This set of weights can be used to construct a consensus context-specific genomic network, together with its associated set of flows $\mu_{ij}^{(c)}$, where the superscript denotes 'consensus.' By constructing the figure of merit $r_{ij}^{(c)}$ as in (III.3) by using the consensus weights, one can determine genomic machines $\mathcal{S}_1, \ldots, \mathcal{S}_K$ as described earlier, for the entire set of cell lines. Next, one can further carry out a longitudinal study *within* the cell line population by computing the edge flows μ_{ij}^l for cell line l using the individual cell line weights, and computing a figure of merit similar to (III.3), namely

$$r_{ij} = \log \frac{\mu_{ij}^l}{\mu_{ij}},$$

where as before μ_{ij} is the edge flow associated with the unweighted graph. In this way, one can examine each of the consensus genomic machines $\mathcal{S}_1, \ldots, \mathcal{S}_K$ and test whether the machine remains intact (in the sense that the sign of the figure of merit is the same on all edges) for a specific cell lines. To repeat, the two-step process consists of first identifying genomic machines that are specific to the disease, and then examining whether a particular genomic machine still functions as such for each cell line.

C. Randomized Algorithms

From the above description, it is clear that the most time-consuming step in the construction of context-specific genomic networks is the computation of the stationary distribution $\boldsymbol{\pi}$ and the doublet frequency vector $\boldsymbol{\mu}$ for several graphs, all of them having the same topology but different sets of weights for the nodes. The baseline computation assigns the weight to each node to be its in-degree, and presumes that outbound transitions on each edge are equally likely. Further refinements are then made on the basis of actual gene expression measurements. The baseline computation is precisely that used in the page rank algorithm. In the original version of this algorithm, the stationary distribution $\boldsymbol{\pi}$ is computed using the 'power method.' Since the matrix F has all positive entries, the Perron theorm implies that $F^l \to \mathbf{e}_n \boldsymbol{\pi}$ as $l \to \infty$. In other words, F^l converges to a rank one matrix, whose rows are all equal. Consequently, for every probability vector $\mathbf{v}$, the iterated product $\mathbf{v}F^l$ converges to $\boldsymbol{\pi}$ (since $\mathbf{v}\mathbf{e}_n = 1$). In the case of the worldwide web, n is around eight billion and growing rapidly, so a direct implementation of the power method is not always practicable. The computer science community has developed various parallel algorithms for doing this computation. In contrast, in [39] a randomized approach is proposed for computing $\boldsymbol{\pi}$. The method in [39] actually pays a lot of attention to things like ensuring synchrony of updating, communication costs etc., but we ignore these factors here. Instead we point out that, unlike in the case of the page rank algorithm and the worldwide web, the precise values of the components of $\boldsymbol{\pi}$ and $\boldsymbol{\mu}$ are not directly relevant in biology. Rather, the relevant quantities are the figures of merit r_{ij} defined (III.3), and whether the figure of merit is positive or negative for a particular edge. Therefore a very germane problem in a biological context is the development of randomized algorithms for *approximate* computation of the stationary distribution and doublet frequency. The computation should be sufficiently accurate to determine the *sign* of the figure of merit for each edge (and whether its absolute value exceeds some threshold). But more is not needed, because the objects of ultimate interest are the cycles where the edges all have the same sign, as explained earlier. Since a typical biologist would want to run the above-described kinds of consensus as well as longitudinal studies on cell lines many times a day, and on simple desktops and not dedicated computing hardware, the key objective to strive for in the development of a randomized algorithm is the reduction of the computational burden.

IV. Analyzing Statistical Significance

In this section we will review some popular methods for estimating the statistical significance of various conclusions that can be drawn from gene expression data. As before, the data is assumed to consist of m samples each of n gene products. Thus the data set consists of real numbers $\{x_{ij}\}, i = 1, \ldots, n, j = 1, \ldots m$. Moreover, it is often the case that the data is labeled. Thus the m samples are grouped into K classes, where class k consists of m_k samples (and obviously $\sum_{k=1}^K m_k = m$). Usually there is a biological basis for this grouping. For instance, K could equal two, and class 1 consists tissue from patients without cancer, while class 2 consists of tumor tissue from cancer patients. Then a new $(m+1)$-st sample is generated for all n genes, and we would like to classify this new vector as belonging to one of the K classes. In order to do so, the type of questions that can be asked are the following:

- Suppose we divide the sample set into two classes consisting of m_1 and m_2 elements each, which without loss

of generality can be renumbered as $\mathcal{M}_1 = \{1, \ldots, m_1\}$ and $\mathcal{M}_2 = \{m_1+1, \ldots, m_1+m_2\}$ where $m_1 + m_2 = m$. For a specific gene (i.e., a specific index i), is it the case that the expression level of gene i for class 1 differs at a statistically significant level from that of class 2? How can this idea be extended to more than two classes?

- In biology it often happens that, in a collection of genes $\mathcal{S}$ (referred to as a genomic machine in Section III), no single gene is over-expressed in class 1 compared to class 2; however, taken together they are over-expressed. Can this notion be made mathematically precise and tested?
- Suppose some sort of classifier has been developed, which achieves a statistically significant separation between the various labeled classes. Now suppose, as before, that an $(m+1)$-st data vector consisting of n expression level measurements becomes available. Usually any classifier makes use of all n components of the data vector. Is it possible to identify a subset of $\{1, \ldots, n\}$ and a reduced-dimension classifier that more or less reproduces the classification abilities of a full-dimension classifier that uses all n components of the data?

In this section we will address each of these questions. Indeed Sections IV-B through IV-D correspond quite precisely to the three questions described above. We begin by presenting, in Section IV-A, some standard results from statistics and probability theory that will provide the underpinnings of the analysis.

A. Basic Statistical Tests

In this subsection, we describe two basic tests, namely the 'student' t distribution and the Kolmogorov-Smirnov test for goodness of fit.

The student t distribution can be used to test the null hypothesis that the means of two sets of samples are equal, under the assumption that the variance of the two sample sets is the same. Strictly speaking the t distribution is derived for the case where the samples follow a normal distribution. However, it can be shown that the distribution applies to a wide variety of situations, even without the normality assumption.

Suppose we have two classes of samples $\mathcal{M}_1, \mathcal{M}_2$, of sizes m_1, m_2 respectively. Thus the data consists of $x_1, \ldots, x_{m_1}$ belonging to the class $\mathcal{M}_1$, and $x_{m_1+1}, \ldots, x_{m_1+m_2}$ belonging to the class $\mathcal{M}_2$. Let $\bar{x}_1, \bar{x}_2$ denote the means of the two sample classes, and let S_1, S_2 denote the unbiased estimates of the standard deviations, that is,

$$S_i^2 = \frac{1}{m_i - 1} \sum_{j \in \mathcal{M}_j} (x_j - \bar{x}_i)^2, i = 1, 2.$$

Now define the 'pooled' standard deviation S_{12} by

$$\begin{aligned} S_{12}^2 &= \frac{(m_1-1)S_1^2 + (m_2-1)S_2^2}{m_1+m_2-2} \\ &= \frac{1}{m_1+m_2-2} \sum_{i=1}^{2} \sum_{j \in \mathcal{M}_j} (x_j - \bar{x}_i)^2. \end{aligned}$$

In other words, the pooled variance is just a weighted average of the two unbiased variance estimates of each class. Then the quantity

$$d_t = \frac{\bar{x}_1 - \bar{x}_2}{S_{12}\sqrt{(1/m_1) + (1/m_2)}} \tag{IV.1}$$

satisfies the t distribution with $m_1 + m_2 - 2$ degrees of freedom. Note that as the number of degrees of freedom approaches infinity, the t distribution approaches the normal distribution. In practice, the t distribution is virtually indistinguishable from the normal distribution when the number of degrees of freedom becomes 20 or larger. Explicit but complicated formulae are available in the literature for the probability density and cumulative distribution function of the t distribution.

The t test is applied as follows: Given the two sets of samples the null hypothesis is that their means are the same. Then the test statistic d_t is computed from (IV.1) for the actual samples. Using the standard tables, the likelihood that a random variable X with the t distribution exceeds d_t (if $d_t > 0$) or is less than d_t (if $d_t < 0$) is computed. If this likelihood is smaller than some prespecified level δ, then the null hypothesis is rejected at the level δ. In other words, it can be concluded with confidence $1 - \delta$ that the null hypothesis is false.

Next we describe the Kolmogorov-Smirnov test for goodness of fit. Suppose X is a real-valued random variable (r.v.). Then its cumulative distribution function (cdf), denoted by $\Phi_X(\cdot)$, is defined by

$$\Phi_X(u) = \Pr\{X \leq u\}.$$

The cdf of any r.v. has a property usually described as 'cadlag', which is an acronym formed from the French phrase 'continu à droite, limité à gauche'. In other words, the cdf is right-continuous in the sense that

$$\lim_{u \to u_0^+} \Phi_X(u) = \Phi_X(u_0),$$

and it has left limits in the sense that the limit

$$\lim_{u \to u_0^-} \Phi_X(u) =: \Phi_X^-(u_0)$$

exists and satisfies $\Phi_X^-(u_0) \leq \Phi_X(u_0)$ for all real u_0.

Suppose $\mathbf{x} = \{x_t\}_{t \geq 1}$ are independent samples of X. Based on the first l samples, we can construct an 'empirical cdf' of X, as follows:

$$\hat{\Phi}_l(u) := \frac{1}{l} \sum_{i=1}^{l} I_{\{x_i \leq u\}}, \tag{IV.2}$$

where I is the indicator function; thus I equals one if the condition stated in the subscript is true, and equals 0 if the condition stated in the subscript is false. To put it another way, $\hat{\Phi}_l(u)$ is just the fraction of the first l samples that are less than or equal to u. The quantity

$$D_l := \sup_u |\hat{\Phi}_l(u) - \Phi_X(u)|$$

gives a measure of just how well the empirical cdf approximates the true cdf. The well-known Glivenko-Cantelli

lemma states that, viewed as a function of $\mathbf{x}$, the stochastic process $\{D_l\}$ converges almost surely to zero as $l \to \infty$. The Kolmogorov theorem and the Kolmogorov-Smirnov statistic quantify the convergence, thereby leading to a test for goodness of fit. Specifically, let us think of D_l as a real-valued random variable, and let Φ_{D_l} denote the cdf of D_l. Then the K-S statistic states that

$$\sqrt{l}\Phi_{D_l} \to \Phi_K \text{ as } l \to \infty,$$

where the convergence is in the distributional sense, and Φ_K is the K-S cdf, introduced a little later. But before we proceed to that, let us recall that a sequence of random variables $\{Y_l\}$ converges to another random variable Z in the distributional sense if

$$\sup_u |\Phi_{Y_l}(u) - \Phi_Z(u)| \to 0 \text{ as } l \to \infty.$$

Thus the contribution of K-S lies in determining the exact limit distribution of the empirical cdf.

The K-S test is used to validate the null hypothesis that a given set of samples $x_1, \ldots, x_l$ are generated in an i.i.d. fashion from a specified cdf $F(\cdot)$. To apply the test, we first construct the empirical cdf $\hat{\Phi}_l$ as in (IV.2), and then compute the goodness of fit statistic

$$d_l = \sup_u |\hat{\Phi}_l(u) - F(u)|.$$

Then the null hypothesis is rejected at level δ (that is, with confidence $\geq 1 - \delta$) if

$$\sqrt{l}d_l > (\bar{\Phi}_K)^{-1}(\delta),$$

where

$$\bar{\Phi}_K(u) = 1 - \Phi_K(u)$$

is the so-called complementary distribution function. This is the so-called one-sample K-S test.

It is also possible to have a two-sample K-S test. Suppose $x_1, \ldots, x_l$ and $y_1, \ldots, y_m$ are two sets of samples, possibly of different lengths. The null hypothesis is that both sets of samples are generated from a common, but unspecified, cdf. To test this hypothesis, we form two empirical cdfs, call them $\hat{\Phi}_l$ based on the x_i samples, and $\hat{\Psi}_m$ based on the y_j samples, in analogy with (IV.2). The test statistic in this instance is

$$d_{l,m} = \sup_u |\hat{\Phi}_l(u) - \hat{\Psi}_m(u)|.$$

The null hypothesis is rejected at level δ if

$$\sqrt{\frac{lm}{l+m}}d_{l,m} > (\bar{\Phi}_K)^{-1}(\delta).$$

Now that we have seen how the K-S cdf can be used, let us specify what it is. It can be shown that

$$\Phi_K(u) = 1 - 2\sum_{k=1}^{\infty}(-1)^{k+1}\exp(-2k^2u^2),$$

$$\bar{\Phi}_K(u) = 2\sum_{k=1}^{\infty}(-1)^{k+1}\exp(-2k^2u^2).$$

As it stands, though the above formula is explicit, it is very difficult to compute $(\bar{\Phi}_K)^{-1}(\delta)$ for a given number δ. However, if we are willing to forgo a little precision, a simple estimate can be derived. Observe that $\bar{\Phi}_K(u)$ is defined by an alternating series; as a result $\bar{\Phi}_K(u)$ is bracketed by any two successive partial sums. In particular, we have that

$$\bar{\Phi}_K(u) \leq 2\exp(-2u^2) =: \bar{\Phi}_M(u), \ \forall u.$$

Therefore it follows that

$$(\bar{\Phi}_K)^{-1}(\delta) \leq (\bar{\Phi}_M)^{-1}(\delta), \ \forall \delta.$$

So to apply the one-sample K-S test, we reject the null hypothesis at level δ if

$$\begin{aligned}\sqrt{l}d_l &> (\bar{\Phi}_M)^{-1}(\delta) \iff \bar{\Phi}_M(\sqrt{l}d_l) < \delta \\ &\iff 2\exp(-2ld_l^2) < \delta \\ &\iff d_l > \left[\frac{1}{2l}\log\frac{2}{\delta}\right]^{1/2}.\end{aligned}$$

Let us define

$$\theta_M(l,\delta) := \left[\frac{1}{2l}\log\frac{2}{\delta}\right]^{1/2} \tag{IV.3}$$

to be the K-S threshold as a function of the number of samples l and the level δ. With this notation, the null hypothesis is rejected at level δ if d_l exceeds this threshold.

Now we digress briefly to discuss how the above kind of test can be applied in more general contexts. As stated, the K-S test applies strictly to real-valued random variables. Extending it even to r.v.s assuming values in $\mathbb{R}^d$ when $d \geq 2$ is not straight-forward; see [41] for one of the few results in this direction. The objective of this digression is to point out that, if one were to use recent results in statistical learning, then K-S-like tests are abundant in quite general settings. A good reference for the discussion below is [42].

We begin with the observation that the 'modern' way to prove the Glivenko-Cantelli lemma is to apply Vapnik-Chervonenkis, or VC theory, and sketch the main results of the theory next. Suppose X is some set (which need not be a subset of a Euclidean space such as $\mathbb{R}^d$), and that P is a probability measure on X. Suppose i.i.d. samples $\{x_t\}_{t\geq 1}$ are generated from X according to the law P. Let $\mathcal{A}$ denote some collection of subsets of X.[16] For each set $A \in \mathcal{A}$, we compute an empirical probability

$$\hat{P}_l(A) = \frac{1}{l}\sum_{t=1}^{l} I_{\{x_i \in A\}}.$$

In other words, $\hat{P}_l(A)$ is just the fraction of the l samples that belong to the set A. Finally, in analogy with earlier notation, define

$$D_l := \sup_{A\in\mathcal{A}} |\hat{P}_l(A) - P(A)|.$$

The collection of sets $\mathcal{A}$ has the property of 'uniform convergence of empirical means' if $D_l \to 0$ almost surely as $l \to \infty$.

[16] Strictly speaking, we should first define a σ-algebra $\mathcal{S}$ of subsets of X and assume that $\mathcal{A} \subseteq \mathcal{S}$. Such details are glossed over here but the treatment in [42] is quite precise.

Recent developments in statistical learning theory, specifically VC theory, consist of associating with each collection of sets $\mathcal{A}$ a positive integer d, called the VC-dimension of $\mathcal{A}$. One of the main results of this theory as described in [42, Theorem 7.4] states that if d is finite, then the collection does indeed has the uniform convergence property. Moreover, if $\bar{\Phi}_{D_l}$ denotes the complementary df of the random variable D_l, then it can be stated with confidence $1-\delta$ that

$$\bar{\Phi}_{D_l}(u) \leq 4\left(\frac{2el}{d}\right)^d \exp(-lu^2/8), \tag{IV.4}$$

where e denotes the base of the natural logarithm. In particular, the collection of semi-infinite intervals $\{(-\infty, u], u \in \mathbb{R}\}$ has VC-dimension one, so that for the standard K-S setting, we can state with confidence $1-\delta$ that

$$\bar{\Phi}_{D_l}(u) \leq 8el \exp(-lu^2/8).$$

In higher dimensions, say in $\mathbb{R}^d$, the collection of sets

$$\mathcal{A} = \{\prod_{i=1}^{d}(-\infty, u_i], u_i \in \mathbb{R}\ \forall i\}$$

has VC-dimension equal to d, so that (IV.4) holds.

To apply this bound in a general setting, suppose P is some probability measure on X, and that $x_1, \ldots, x_l$ are elements of X. The null hypothesis is that these samples have been generated as independent samples according to the law P. To test this hypothesis, choose any collection of subsets $\mathcal{A}$ of X with finite VC-dimension d, and form the test statistic

$$d_l = \sup_{A \in \mathcal{A}} |P(A) - \hat{P}_l(A)|.$$

If it is the case that $\bar{\Phi}_{D_l}(d_l) > \delta$, then the null hypothesis is rejected the level δ. Now we don't know $\bar{\Phi}_{D_l}(d_l)$ but we do have an upper bound in the form of (IV.4). Let $\bar{\Phi}_{VC}$ denote the right side of (IV.4). Then we reject the null hypothesis at level δ if $\bar{\Phi}_{VC}(d_l) > \delta$. This can be turned into an explicit threshold formula by simple algebra. It is easy to show that

$$\bar{\Phi}_{VC}(d_l) > \delta \iff d_l \geq \left[\frac{8}{l}\left(\log\frac{4}{\delta} + d\log\frac{2el}{d}\right)\right]^{1/2}.$$

Let us denote the right side as a new threshold function, namely

$$\theta_{VC}(l, \delta; d) := \left[\frac{8}{l}\left(\log\frac{4}{\delta} + d\log\frac{2el}{d}\right)\right]^{1/2}. \tag{IV.5}$$

Then we reject the null hypothesis if $d_l > \theta_{VC}(l, \delta; d)$.

If we compare the thresholds from K-S theory and VC theory, we see from (IV.3) and (IV.5) that for fixed confidence level δ the K-S threshold is $O(l^{-1/2})$ whereas the VC threshold is $O(l^{-1/2}\log l)$. But the VC threshold is far more general. So the slightly more conservative bound is definitely worthwhile. For fixed sample length l, both thresholds are $O(\log(1/\delta))$ so there is no difference.

B. Significance Analysis for Microarrays

In this subsection we discuss a widely use method called Significance Analysis for Microarrays (SAM), introduced in [43]. The reader is directed to that paper for discussion of earlier work in this area.

The problem considered is the following: Suppose as before that we have a gene expression data set $\{x_{ij}\}, i = 1, \ldots, n, j = 1, \ldots, m$, where n is the number of genes and m is the number of samples. Suppose further that the data is labeled and divided into two classes. Without loss of generality, suppose the first m_1 samples belong to class 1, and the remaining $m_2 = m - m_1$ belong to class 2. We would like to assess which amongst the n genes show significant variation between the two classes.

As a first-cut, we could treat each of the n genes separately, and for each index i, construct a two-sample K-S test statistic between the samples $\{x_{ij}, j = 1, \ldots, m_1\}$ and $\{x_{ij}, j = m_1 + 1, m_1 + m_2\}$. Specifically, for each index i, let $\bar{x}_{i1}, \bar{x}_{i2}$ denote the average values of the samples in the two classes, and the pooled standard deviation s_i by

$$s_i^2 = \frac{1}{m-2}\left[\sum_{j=1}^{m_1}(x_{ij} - \bar{x}_{i1})^2 + \sum_{j=m_1+1}^{m}(x_{ij} - x_{i2})^2\right].$$

Now it can happen that some genes exhibit so little variation within each class that s_i is very small, with the consequence that any quantity divided by s_i automatically becomes large. To guard against this possibility, a constant s_0 is chosen to be the same for all indices i. Next, for each index i, we define the test statistic

$$\alpha_{i0} = \frac{\bar{x}_{i1} - \bar{x}_{i2}}{(s_i + s_0)[(1/m_1) + (1/m_2)]^{1/2}}.$$

By examining the significance of α_{i0} using the t-distribution and the two-sample K-S test, we might be able to determine whether gene i exhibits a substantial variation between the two classes.

However, this alone might not give a true picture. It often happens in the case of biological data that the *inherent* variation of expression levels changes enormously from one gene to another. For instance, the expression level of one gene may show barely 10% variation across experiments, whereas that of another gene may show an order of magnitude vairation. If we were to apply the K-S test blindly, we would conclude that the second gene is far more significant than the first one. But this is potentially misleading. In biology it is often the case that the downstream consequences of variations in gene expression are also widely different for different genes.

To normalize against this possibility, in [43], the authors introduce an additional criterion. Given the integers m_1, m_2, choose an integer k roughly equal to $0.5\min\{m_1, m_2\}$. Let $\pi_1, \ldots, \pi_L$ be permutations of $\{1, \ldots, m\}$ into itself such that precisely k elements from class 1 are shifted to class 2 and vice versa. In the original paper [43], $m_1 = m_2 = 4$ so that $k = 2$, and there are $6^2 = 36$ such permutations; so they consider all of them. However, if the integers m_1, m_2 are sufficiently large, the number of such permutations will be huge, in which case one chooses, at random, a prespecified

number L of such permutations. For each permutation π_l, the first m_1 elements are labeled as 1 and the rest are labeled as 2. In other words, the elements $\pi_l(1), \ldots, \pi_l(m_1)$ are given the label 1 while the rest are given the label 2. For each labeling corresponding to the permutation π_l, let us compute a two-sample K-S test statistic, which we may denote by α_{il}. This is done for each of the n genes. Next, let us define

$$\alpha_E(i) = \frac{1}{L} \sum_{l=1}^{L} \alpha_{il}$$

to be the value of the test statistic averaged across all L permutations. Let α_{i0} denote the test statistic corresponding to the identity permutation, that is, the original labeling. For most genes (i.e., for most indices i), the test statistic α_{i0} corresponding to the original labeling will not differ much from the averaged value $\alpha_E(i)$. Those genes for which the difference is significant, in either direction, are the genes that one should examine. To implement this criterion, an absolute constant Δ is chosen, and only those genes for which $|\alpha_{i0} - \alpha_E(i)| \geq \Delta$ are studied further. One could of course argue that the threshold should be in terms of the ratio $\alpha_{i0}/\alpha_E(i)$ and that too would be a valid viewpoint. In [43], using this approach only 46 out of an original set of 6,800 genes are found to be worth examining further – a reduction of more than two orders of magnitude. What this means is that, for all except these 46 genes, the test statistic corresponding to the original labeling is not very different from what would result from a purely random assignment of labels. These short-listed genes are then examined whether indeed there is substantial variation between the two classes (which it may be noted is a different question from whether a randomly assigned label would result in a different value for the test statistic). A gene belonging to this shorter list is deemed to exhibit significant variation between classes 1 and 2 if

$$\max \left\{ \frac{\bar{x}_{i1}}{\bar{x}_{i2}}, \frac{\bar{x}_{i2}}{\bar{x}_{i1}} \right\} > R,$$

where R is another threshold. This thresholding results in a final set of genes with two attributes: (i) The test statistic corresponding to the original labeling differs substantially from that corresponding to a random assignment of labels, and (ii) there is substantial difference between the mean values of the two classes. This is the desired list of genes. Note that we could have just as easily compared $|\log(\bar{x}_{i1}/\bar{x}_{i2})|$ against a threshold. We could also apply the K-S test and choose those genes for which the difference is statistically significant at a prespecified level.

C. Gene Set Enhancement Analysis

As in the previous subsection, suppose have a gene expression data set $\{x_{ij}\}, i = 1, \ldots, n, j = 1, \ldots, m$, where n is the number of genes and m is the number of samples. Further, the data is labeled and divided into two samples. Suppose $\mathcal{M} = \{1, \ldots, m\}$ and that $\mathcal{M}_1, \mathcal{M}_2$ is a partition of $\mathcal{M}$. Further, suppose $|\mathcal{M}_i| = m_i$ for $i = 1, 2$. For example, the samples in class 1 may come from healthy tissue while those in class 2 may come from cancerous tissue. In the previous subsection, we studied the problem of identifying *individual genes* within the set of n genes that show statistically significant variation between the two classes. For this purpose, for each gene i we compared the t-statistic between the two classes against what would be obtained by randomly assigning labels to the of m samples associated with that gene. In this section, we carry the discussion to a greater level of generality. Specifically, it can happen in biological experiments that, while no single gene may show statistically a significant difference between the two classes, a collection of genes acting in concert may exhibit such statistically significant difference between the two classes. Accordingly, suppose a subset $\mathcal{S}$ of $\mathcal{N} = \{1, \ldots, n\}$ is specified beforehand as a set of genes that we expect might *collectively* exhibit different expression levels between the two classes. Note that the set $\mathcal{S}$ is specified on the basis of biological considerations, and not deduced *post facto* from the data under study. For instance, $\mathcal{S}$ could be one of the 'genomic machines' identified through the Netwalk algorithm of Section III-B.

The discussion below is essentially taken from [45] which describes an algorithm that those authors call GSA (Gene Set Analysis). In turn [45] builds on an earlier algorithm called GSEA (Gene Set Enhancement Analysis) from [44]. Along the way, the authors of [45] also relate their GSA algorithm to several earlier algorithms. In the interests of conserving space, we do not reference nor discuss all the earlier work, and the interested reader is directed to the bibliography of [45].

The main idea of the GSA algorithm is the following: In SAM (Significance Analysis for Microarrays) discussed in Section IV-B, for each index i denoting the gene, we did the following: First we computed the t-statistic of the difference between the means of the two classes. Then we assigned random labels to the m samples associated with gene i, ensuring that m_i are placed in class i, and for each random labeling, we computed the same t-statistic. That is fine so far as testing a single gene goes. To test whether a prespecified set of genes shows significant difference between the two classes, it is necessary to perform an additional step, as described next. Let $k = |\mathcal{S}|$. Then, in addition to permuting the labels of the m columns associated with each gene in the set $\mathcal{N}$, we should also do the same to a randomly selected set of k genes from the collection $\mathcal{N}$. In [45], assigning the class labels at random is referred to as 'permuation' while choosing a random set of k genes from $\mathcal{N}$ is referred to as 'randomization'. An additional complication in the randomization step is that, while the expression levels of k randomly selected genes from $\mathcal{N}$ can be thought of as being uncorrelated, the expression levels of the k genes in the specified set $\mathcal{S}$ are quite likely to be correlated (due to their having a common biological function etc.). Hence the randomized data will in general have different statistical behavior from that of the genes in the set $\mathcal{S}$. The GSA algorithm attempts to correct for this feature.

The details of the algorithm are as follows: For each gene i in $\mathcal{N}$, form a two-sample t-statistic, call it d_i. Then d_i is distributed according to the t-distribution with $m-2$ degrees

of freedom. The quantity d_i is transformed into another value z_i that has a normal distribution, by the rule

$$z_i = \Phi_{\mathrm{Nor}}^{-1}(\Phi_{t,m-2}(d_i)),$$

where Φ_{Nor} denotes the cdf of a normal r.v. and $\Phi_{t,m-2}$ denotes the cdf of a t-distributed r.v. Note that if the number of samples m is sufficiently large, then the t-distribution is virtually identical to the normal distribution, so this step can be omitted. Now suppose $S : \mathbb{R} \to \mathbb{R}$ is a scoring function.[17] In [44], the scoring function $S(z)$ equals $|z|$. For each gene i, let s_i be a shorthand for $s(z_i)$. For the gene set $\mathcal{S}$, compute the score

$$S = \frac{1}{k}\sum_{i\in\mathcal{S}} s_i. \tag{IV.6}$$

The question under study is: Is the score S sufficiently significant?

Now compute the mean μ_0 and standard deviation σ_0 of the raw samples in the familiar manner, namely:

$$\mu_0 = \frac{1}{n}\sum_{i\in\mathcal{N}} s_i, \sigma^2 = \frac{1}{n-1}\sum_{i\in\mathcal{N}}(s_i-\mu_0)^2.$$

Next, choose at random several subsets of $\mathcal{N}$ of cardinality k, compute the counterpart of the score S for each such randomly chosen gene set, and compute the mean and standard deviation of all of these scores (over all the randomly selected sets of cardinality k). Denote these by $\mu^\dagger, \sigma^\dagger$ respectively. If all the samples s_i within a set of cardinality k are independent, then we would have $\mu^\dagger = \mu_0, \sigma^\dagger = \sigma/\sqrt{k}$. But this need not be the case in general.

Next, choose a large number of permutations $\pi_1, \ldots, \pi_L$ of $\mathcal{M}$ into itself. For each permutation π_l, assign the label i to the samples in the image $\pi_l(\mathcal{M}_i)$, for $i = 1, 2$. This will generate, for each gene i, a test statistic $z_{\pi_l,i}$ and score $s_{\pi_l,i}$. Let μ_P, σ_P sdenote the mean and standard deviation of these nL numbers, where the subscript P is to remind us of 'permutation'.

The next step is called 'restandardization'. For each permutation π_l, let S_{π_l} denote the score resulting from the labeling as per the permutation π_l. Then the renormalized score corresponding to π_l is defined as

$$S_{R,\pi_l} = \mu^\dagger + \frac{\sigma^\dagger}{\sigma_P}(S_{\pi_l} - \mu_P).$$

Then a test statistic is given by the quantity

$$p_{\mathcal{S}} = \frac{1}{L}\sum_{l=1}^{L} I_{\{S_{R,\pi_l} > S\}},$$

which is the fraction of the restandardized scores that exceed the nominal score S. Clearly the smaller $p_{\mathcal{S}}$ is, the more significant is the score S. In GSEA, the cdf of the samples $\{z_i, i \in \mathcal{S}\}$ is compared to the cdf of the samples $\{z_i, i \notin \mathcal{S}\}$. This more or less corresponds to the choice $s(z) = |z|$.

[17] We mostly follow the notation in [45], in which the letter S in various fonts is used to denote various quantities. The reader is therefore urged to pay careful attention.

Finally, in [45] another statistic is introduced, known as the max-mean statistic. Define

$$(z)_+ = \max\{z, 0\}, (z)_- = -\min\{z, 0\},$$

and observe that $(z)_-$ is positive if z is negative, somewhat contrary to the usual convention. Now define

$$s^+ = \frac{1}{k}\sum_{i\in\mathcal{S}}(s_i)_+, s^- = \frac{1}{k}\sum_{i\in\mathcal{S}}(s_i)_-, s_{\max} = \max\{s^+, s^-\}.$$

D. Pattern Analysis for Microarrays

In this subsection we discuss a method for simplifying the application of nearest neighbor clustering in the context of gene expression studies. This method is known as Pattern Analysis for Microarrays (PAM) [46]. The similarity of the acronyms SAM and PAM is not coincidental, because as we shall see, the two approaches have a lot in common.

As always, suppose we are given a set of gene expression data $\{x_{ij}, i = 1, \ldots, n, j = 1, \ldots, m\}$. Suppose further that the set $\mathcal{M} = \{1, \ldots, m\}$ of samples is divided into K classes, which are denoted here as $\mathcal{M}_k, k = 1, \ldots, K$. Thus the collection $\{\mathcal{M}_1, \ldots, \mathcal{M}_K\}$ is a partition of $\mathcal{M}$. Let denote $|\mathcal{M}_k|$ by m_k. Now suppose a new data vector $\mathbf{y} \in \mathbb{R}^n$ arrives from a fresh study. We would like to classify $\mathbf{y}$ as belonging to one of the K classes. How should we go about it?

One of the most commonly used method is that of nearest neighbor classification. As before, let us define the mean values of the expression level of gene i in class k, and the overall mean value, by

$$\bar{x}_{ik} := \frac{1}{m_k}\sum_{j\in\mathcal{M}_k} x_{ij}, k = 1, \ldots, K,$$

$$\bar{x}_i = \frac{1}{m}\sum_{k=1}^{K}\sum_{j\in\mathcal{M}_k} x_{ij} = \sum_{k=1}^{K}\frac{m_k}{m}\bar{x}_{ik}.$$

Thus $\bar{\mathbf{x}}_k \in \mathbb{R}^n$ is the centroid of class k while $\mathbf{x} \in \mathbb{R}^n$ is the overall centroid. To classify the vector $\mathbf{y}$, we compute the Euclidean distance to each of the K centroids, and classify it into the class whose centroid is the closest. Applying this classification method requires the computation of

$$\|\mathbf{y} - \bar{\mathbf{x}}_k\|^2 = \sum_{i=1}^{n}(y_i - \bar{x}_{ik})^2 \tag{IV.7}$$

for each k. If, as is often the case, n is of the order of thousands if not tens of thousands, the above computation can be quite expensive. The objective of PAM is to determine a subset $\mathcal{N}_1$ of $\mathcal{N} = \{1, \ldots, n\}$ with $|\mathcal{N}_1| \ll n$ such that, if the summation is taken only over those $i \in \mathcal{N}_1$, the resulting nearest neighbor classification would be more or less the same.

The basic idea behind PAM is as follows: In [46], PAM is also referred to as the 'method of shrunken centroids'. Suppose that for some index i, it is the case that $\bar{x}_{ik}$ is the same for all values of k. In other words, suppose that the i-th component of the centroid $\bar{\mathbf{x}}_k$ is the same for all k. Then it is obvious that the index i can be dropped from the summation

in (IV.7) because the term $(y_i - \bar{x}_{ik})^2$ makes an equal contribution for all k. So the method of shrunken centroids consists of shrinking the spread amongst $\{x_{i1}, \ldots, x_{iK}\}$ to zero for as many indices i as possible, by replacing the true centroid by a synthetic centroid.

In analogy with earlier reasoning, define the pooled within class standard deviation of gene i by

$$s_i^2 = \frac{1}{m-k} \sum_{k=1}^{K} \sum_{j \in \mathcal{M}_k} (x_{ij} - \bar{x}_{ik})^2.$$

Next, as before, a small constant s_0 (independent of i) is added to each s_i to avoid division by very small numbers. Now define a test statistic d_{ik} that tests for the null hypothesis that the data in class k differs significantly from the overall data, namely

$$d_{ik} = \frac{\bar{x}_{ik} - \bar{x}_i}{(s_i + s_0)[(1/m_k) + (1/m)]^{1/2}} =: \frac{\bar{x}_{ik} - \bar{x}_i}{l_k(s_i + s_0)},$$

where

$$l_k = \left[\frac{1}{m_k} + \frac{1}{m} \right]^{1/2}.$$

Note that it would perhaps be more accurate to compare $\bar{x}_{ik}$ with the 'leave one out' mean of all the remaining $m - m_k$ entries, as opposed to the overall mean $\bar{x}_i$. But this would involve considerably more computation with relatively little benefit.

Now rewrite the above relationship as

$$\bar{x}_{ik} = \bar{x}_i + l_k(s_i + s_0)d_{ik}.$$

If we could somehow justify replacing the actual d_{ik} by zero, then it would follow that $\bar{x}_{ik} = \bar{x}_i$ for all k, and we could therefore ignore the i-th term in the summation (IV.7). This is achieved by soft thresholding. Specifically, a fixed constant Δ, independent of both i and k, is selected, Then we define

$$d'_{ik} = \text{sign}(d_{ik})(|d_{ik}| - \Delta)_+,$$

where as usual $(x)_+ = \max\{x, 0\}$. An equivalent definition of d'_{ik} is

$$d'_{ik} = \begin{cases} d_{ik} - \Delta, & \text{if } d_{ik} > \Delta, \\ d_{ik} + \Delta, & \text{if } d_{ik} < -\Delta, \\ 0, & \text{if } |d_{ik}| \leq \Delta. \end{cases}$$

Then the centroids are 'shrunk' by replacing d_{ik} by d'_{ik}, namely

$$\bar{x}_{ik} = \bar{x}_i + l_k(s_i + s_0)d'_{ik}. \tag{IV.8}$$

Note that if $d'_{ik} = 0$ for all k for a fixed i, then that term can be dropped from the summation in (IV.7).

The higher the value of Δ, the more thresholds that will be set to zero. At the same time, the higher the value of Δ, the more the likelihood of misclassification by the simplified summation. In [46], the constant Δ is chosen through ten-fold cross validation. The data set is divided vertically (in terms of the index j) into ten more or less equal-sized data sets. 90% of the data is used as training data and the remaining 10% is used to test the resulting reduced-sum classifier; this exercise is repeated by shifting the testing data through each subset of the data. The constant Δ is adjusted up or down until the cross-validation produces satisfactory results. In [46], the original data set consists of expression levels of 2,308 genes, 63 samples, classified into four forms of cancer. Thus $n = 2308, m = 63$ and $K = 4$. By using the soft thresholding technique, a subset of a mere 43 'most useful genes' are identified out of the original 2,308 – a reduction of about 98% in the computational burden.

V. Separating Drivers from Passengers

Until now we have discussed various topics that involve the use of fairly advanced methods in probability and statistics. In this section, we present some preliminary results on the problem of distinguishing drivers of cancer from passengers. As will be seen, the method used is fairly elementary, namely simple k-means clustering. However, it is included here because this approach has already led to some predictions about which genes have a role in colorectal cancer (CRC), and the existing biology literature indicates that some of these genes are already known to play a role in other forms of cancer. Thus the message of this section is two-fold: First, it is not always necessary to use very advanced methods to make interesting predictions. Second, however 'pretty' the underlying mathematics might be, unless the methods lead to hypotheses that are subsequently verified, they are not of any use.

As mentioned earlier, at present there is a massive public effort known as TCGA (The Cancer Genome Atlas) directed at sequencing every available cancerous tumor. Mutations in specific genes lead to disruptions in the associated regulatory networks, often referred to as 'lesions'. Sequencing of tumorous tissues and cells has thrown up and will continue to throw up a bewildering variety of mutations, some of which cause cancer (referred to as 'drivers' or 'causal mutations') while other mutations are caused by cancer (referred to as 'passengers' or 'coincidental mutations'). Simply detecting the frequency with which a particular gene is found to be mutated in cancerous tissue is not sufficient to distinguish the drivers of cancers from the passengers. Some additional indications need to be used to discriminate further amongst highly mutated genes. In this subsection, some preliminary results are presented to support the hypothesis that a seven-dimensional feature vector, called the 'developmental gene expression profile,' can be used to achieve such discrimination.

We begin as usual with some background. The paper [47] presents a 'landscape' of human breast and colorectal cancer by identifying every gene that has been found in a mutated state in 11 tumor tissues of colorectal and cancer and 11 tumor tissues of breast cancer. This paper builds on an earlier work, Sjöblom et al. [48], in which 13,023 genes in 11 breast and 11 colorectal cancer tissues are analyzed. In [47], A total of 18,191 genes analyzed, out of which 1,718 were found to have at least one nonsilent mutation in either a breast or a colorectal cancer.[18] Amongst these, a total of 280 genes

[18] A nonsilent mutation is a mutation that causes a change in the amino acid sequence (primary structure) of the protein(s) produced by a gene.

were identified as 'CAN-genes', that is, potentially drivers of cancer, if they had 'harbored at least one nonsynonymous mutation in both the Discovery and Validation Screens and if the total number of mutations per nucleotide sequenced exceeded a minimum threshold' [47].

It is in principle possible to carry out a very large number of experiments to test whether specific lesions are causal or not. However, in order to be definitive, it is not enough to study individual lesions – one would also have to study all possible combinations of lesions. Even if one were to focus only on the 280 CAN-genes, there would be roughly 40,000 pairs of genes, and roughly 3.6 million triplets of genes, and so on.

It is clearly impractical to carry out so many experiments. It would be preferable to have some additional indications so as to prioritize the experiments roughly in proportion to their likelihood of success. One way to achieve this is to begin with a handful of experiments where the outcomes are known, some genes being likely tumor-suppressors ('hits') while others are not likely to be so ('misses'). Then some form of pattern recognition or machine learning algorithms can be used to discriminate between the known successes ('hits') and known failures ('misses'). In the last step, this discriminating function can then be extrapoloated to all CAN-genes (or perhaps to an even larger set of genes). It must be emphasized that statistical or pattern recognition methods are not a substitute for actual experimental verification. However, by providing a high degree of separation between known hits and known misses, such methods can assist in prioritizing future experiments by increasing the likelihood of success.

Now we introduce the so-called developmental gene expression profile, and justify why it may possibly have a role in distinguishing between drivers and passengers. Development can be divided into seven stages, namely embroyd body, blastocyst, fetus, neonate, infant, juvenile, and adult. The database Unigene [49] provides, for more than 100,000 genes as well as ESTs,[19] their frequency of occurrence within the tissues tested at each of the seven developmental stages. The Unigene database is far more comprehensive than earlier efforts by individual research teams to determine this type of information; see [50] for an example of this type of effort. For instance, it has been known for some time that various genes belonging to the so-called RAS family play an important role in cancer; see [51]. Out of the genes in this family, let us focus on KRAS and HRAS for now. Their Unigene entries are as follows, in parts per million:

Gene	EB	B	F	N	I	J	A
KRAS	169	80	60	0	0	54	77
HRAS	28	16	19	0	0	0	24

Since the entries are in parts per million, it is clear that these genes are not prevalent in *any* developmental stage. This raises the question as to how statistically significant the zero entries are. We shall discuss this topic in the concluding section.

[19]ESTs (Expressed Sequence Tags) are parts of genes that were sequenced and catalogued before whole genome sequencing became commonplace.

Our hypothesis is that the developmental gene expression profile can be used to discriminate between drivers and passengers. This hypothesis is the outcome of putting together the results of a very interesting series of biological experiments. Specifically, in [52] it is shown that KRAS is essential for the development of the mouse embryo – if the KRAS gene is knocked out, then the embryo does not survive. However, as shown in [53], if the KRAS gene is not knocked out, but is instead replaced by HRAS in the KRAS locus, then the resulting HRAS-knocked in mouse embryo develops normally. Following along these lines, when HRAS was put into the KRAS locus and lung cancer was induced in these mice, the HRAS in the KRAS locus was found to be mutated, whereas the HRAS in the HRAS locus was *not mutated* [54]. Since HRAS and KRAS express themselves at different stages of the development of a mouse embryo, this observation suggests a possible relationship between the expression profile of a gene as a function of developmental stage on the one hand, and its role as a causal factor in cancer on the other hand.

To validate our hypothesis, we used another database called COSMIC (Catalogue of Somatic Mutations in Cancer) [55], that gives the observed mutation frequency of various genes in various forms of cancer. Again, COSMIC is a repository of mutation data discovered by research teams all around the world, as in [56] for example. In spite of this, since testing is expensive, not all of the roughly 30,000 known genes have been tested for mutations in all available tissues. At the moment (though of course this number keeps changing with time, albeit rather slowly), a total of 4,105 genes have been tested for mutations in any one of five forms of cancer, namely: breast, kidney, large intestine (colon), lung, and pancreas. Therefore the remaining genes were deemed not to have sufficient mutation data to permit the drawing of meaningful conclusions. Out of these 4,105 genes from COSMIC, 3,672 had entries in Unigene. These 3,672 seven-dimensional developmental gene expression profiles were clustered using the popular k-means algorithm [57]. In this approach, the given data vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^7$ where $n = 3672$ are clustered into k classes (k to be specified by the user) in such a way that the vectors in each class are closer to the centroid of its own class than to the centroids of all other classes. In symbols, if $\bar{\mathbf{x}}_1, \ldots, \bar{\mathbf{x}}_k$ denote the centroids of the clusters, and $\mathcal{N}_1, \ldots, \mathcal{N}_k$ denote the classes themselves, then

$$\|\mathbf{x}_i - \bar{\mathbf{x}}_k\| \leq \|\mathbf{x}_i - \bar{\mathbf{x}}_j\|, \ \forall j \neq k, \ \forall i \in \mathcal{N}_k.$$

Computing the optimal clusters is an NP-hard problem, so most often one uses some randomized algorithm. Also, the clusters themselves will be different depending on which norm is used. We have found that we get better segregation if we use the ℓ_1-norm than with the ℓ_2-norm.

Once the clusters are formed, the next step is to test whether any of these clusters is 'enriched' with known cancer drivers, compared to the remaining clusters. For determining this, it is necessary that at least a few of these 3,672 genes should be labeled, so that the problem is one of supervised learning. Fortunately, a recently completed

work [58] provides a good starting point. In that paper, the authors began with the 280 CAN-genes identified by [48], [47], and identified 151 of these CAN-genes for testing in an experimental test bed that roughly approximates the environment in the colon.[20] Each of these 151 genes was individually suppressed, and the effect was observed. If the suppression of the gene resulting in cell proliferation, then the gene was labeled as a 'hit' and was presumed to play some role in colorectal cancer (CRC). If on the other hand the suppression of the gene did not result in cell proliferation, then the gene was labeled as a 'miss'. Out of the 151 genes tested, 65 turned out to be hits while the remaining 86 were labeled as misses. As a point of comparison, 400 randomly chosen genes were also tested in the same way, and only 4 were hits. Thus the fact that 65 out of 151 CAN-genes, roughly 45%, are hits is clearly not due to chance, because out of the randomly chosen genes only 1% were hits.

At this stage it should be pointed out that there can in fact be some ambiguity in the miss label. Even if the suppression of a particular gene did not result in cell proliferation, it is nevertheless possible that, under a different set of experimental conditions, the gene might have turned out to be a hit. From the standpoint of machine learning, this can be thought of as a problem of learning and extrapolating from labeled data, in which a positive label is 100% accurate, whereas a negative label is treated as being inaccurate with some small probability. Learning with randomly mislabeled samples is a standard problem, and some results on this problem can be found in [42].

With the aid of these labeled genes, we then tested to see whether any of the clusters obtained by k-means was in fact enriched. Out of the 151 CAN genes tested, only 143 had entries in Unigene, so these were the labeled genes out of the 3,672 genes that were clustered. When we chose $k = 4$, the following clusters resulted.

No.	Hits	Misses	Total
C1	27	47	1,807
C2	10	4	217
C3	15	16	1,016
C4	12	12	632
Total	64	79	3,672

From these results, it is apparent that Cluster No. 2 is significantly enriched for hits. The statistical significance of this was computed in two different ways. First, the null hypothesis was that the hits and misses are uniformly distributed into the four clusters, with 74, 14, 31, and 24 elements respectively, and the likelihood of there being 10 hits and 4 misses in Cluster No. 2 was computed under the assumption that the two were distributed independently. Second, the null hypothesis was that the hits are uniformly distributed into the four clusters with 1,807, 217, 2,016, and 632 elements, and the likelihood of there being 10 hits out of 217 elements in Cluster No. 2 was tested. In both tests, the null hypothesis was rejected at a 1% level. In other words, we could assert with confidence greater than 99% that the enrichment of hits in Cluster 2 is *not* due to chance.

This then allowed us to focus on the 217 genes in Cluster No. 2 as possible candidates in causing colorectal cancer. It is worth repeating that, out of the 217 genes in this cluster, only 14 are CAN-genes and thus tested in [58]. The next step was to choose which amongst the remaining $217 - 14 = 203$ genes are to be tested. Again, since a gene that has not been found to be mutated in even one sample of CRC tissue is highly unlikely to be a CRC driver, we examined these 217 genes, and found 58 genes to be mutated in at least one CRC tissue sample, as per the COSMIC database. Out of these, 14 were already tested, thus leaving 44 candidate genes for testing as drivers for CRC.

Due to the size of the data set, a table containing the predicted and tested genes is placed at the end of the paper, after the references. In the table, the 10 genes that are hits are shown in green, while the 4 genes that are misses are shown in maroon. The mutation frequency of the genes in the samples tested as per the COSMIC database is also shown as a percentage. A careful annotation by Prof. Michael A. White of the UT Southwestern Medical Center showed that out of these 44 genes, 9 had already been mentioned in the biology literature as playing a role in other forms of cancer. This annotation is also shown in the table. This finding should be considered significant, because these 44 genes were determined purely on the basis of the developmental gene expression data, and the likelihood that 20% of them (9/44) would turn out to have a role in cancer is rather minimal. One other noteworthy point is that the gene no. 3486, IGFBP3, is known to be a tumor suppressor, but in the experiments of Eskiocak et al. [58], it turned out to be a miss! This again highlights that while a hit is definitely a hit, a miss is not always a miss.

VI. Some Research Directions

The preceding discussion barely scratches the surface of the vast array of possibilities for applying probabilistic approaches to problems in cancer biology. Rather, the emphasis has been on describing a few problems in sufficient detail to permit a description of directions for further research.

In the problem of reverse-engineering GRNs, we have discussed only methods based on viewing the gene expression data as a set of samples of random variables. There are other possible approaches; see for instance the work of Sontag and his coworkers [60], [61], [62] based on network reconstruction based on steady state data, an approach followed also in [63], [64]. Another approach is to view a regulatory network as a Boolean network, and the onset of cancer as a fault in the Boolean network; see [65], [66].

In terms of using probabilistic methods, the present situation is not entirely satisfactory. The most widely used approaches, based respectively on mutual information and on Bayesian networks respectively, each place very severe restrictions on the nature of the interactions between genes. The mutual information-based approach presumes that interactions between genes are always symmetric, whereas the

[20] As can be imagined, this is a gross over-simplification, and the interested reader is advised to read the original paper for further details.

Bayesian framework presumes that the interactions are always acyclic. In reality, neither assumption holds in practice. Hence the need of the hour is to come up with some other measure of interaction that is more biologically realistic, while being amenable to computation. In the case of the methods based on mutual information, there is a basic premise that is adopted for convenience, but is not strictly valid. Specifically, the data processing inequality states that if $X \to Y \to Z$ is a short Markov chain, then

$$I(X,Z) \leq \min\{I(X,Y), I(Y,Z)\}.$$

However, as can be seen from Section II-B, in the ARACNE algorithm the above reasoning is turned around to say that, if the data processing inequality holds, then no edge exists between X and Z. Ideally one should avoid such steps. In the case of the Bayesian approach, the choice of the 'prior', be it the graph $\mathcal{G}$ or the probability distribution of the various parameters that describe $\mathcal{G}$, is always contentious. And finally, to repeat again, the fact that real biological networks neither have symmetric interactions nor are acyclic needs to be taken into account in future research.

In the case of personal genomic networks, the amendment in (III.2) is not entirely realistic. In the original page rank algorithm, the justification for introducing an additional term is to eliminate the possibility of dangling nodes (i.e. nodes with no outgoing edges), which would cause the stationary distribution to be supported entirely on these dangling nodes. Given that a correction is required, in the page rank algorithm the correction is taken to be proportional to the rank one matrix $\mathbf{e}_n\mathbf{e}_n^T$, the reason being that a user of the web is just as likely to jump from any one web page to any other web page. In the context of genomic networks, the justification has to be that, while the original graph $\mathcal{G}$ captures the known (or suspected) interactions between genes or gene products, there may be other interactions that have not yet been detected. Hence here again a correction term is warranted. However, it is no longer possible to justify that the correction term should be proportional to $\mathbf{e}_n\mathbf{e}_n^T$. Such a correction term presupposes that every gene is equally likely to have an undetected interaction with every other gene. In reality, such an assumption is quite unrealistic. It would be far more realistic to suppose that a node with many known interactions is more likely to have undetected interactions than a node with fewer known interactions. Accordingly, instead of (III.2), it may be more realistic to use the modification

$$f_{ij} = (1 - q_i)p_{ij} + \frac{q_i}{n},$$

or equivalently

$$F = (I - Q)P + (1/n)Q\mathbf{e}_n\mathbf{e}_n^T,$$

where q_i is an increasing function of $|S(i)|$, the cardinality of the known outgoing edge set, and Q is a diagonal matrix with $q_1, \ldots, q_n$ on the diagonal. It is not difficult to modify the randomized algorithm of [39] to this alternate formulation. In terms of applying the algorithm, the termination criterion should be, not that the stationary distribution has been reached, but that the sign of each figure of merit r_{ij} can be unambiguously determined (which can happen much earlier). In short, the asymptotic theory of [39], which is based on the theory of ergodicity from [37], needs to be converted to provide finite-time estimates. One possibility may be to use the Birkhoff contraction coefficient, as discussed in [37]. With the correction, the matrix F is strictly positive as are the matrices arising from the randomized algorithm, so they all have a Birkhoff contraction strictly less than one.

There is one interesting issue in the case of context-specific genomic networks, namely invariance under monotone transformations. As pointed out earlier, the mutual information between two random variables is invariant under monotone transformations of the variables. So if we were to pre-filter the raw gene expression data by centering, scaling, or linear to logarithmic transformations, it is obvious that the resulting network would be unaffected. However, it is not clear what happens in the case of context-specific genomic networks. About all that one can say at a first glance is that if each weight w_i is replaced by αw_i where α is a fixed scalar, then the network is unaffected; but this result is quite trivial. It would be highly desirable to explore whether other types of invariance properties can be proved.

The algorithms presented in Section IV are all computationally intensive, requiring evaluation of a huge number of significance values for a given data set. In view of this feature, it is rather unfortunate that all of these algorithms are also *nonrecursive*. In other words, suppose one has run any of these algorithms on a data set of n genes and m samples, and now one more sample set becomes available. Then there is no option but to carry out the entire computation *ab initio*. It would therefore be desirable to develop some recursive algorithms for addressing the type of questions studied in Section IV.

While the approach described in Section V for discriminating between causal mutations (drivers) and coincidental mutations (passengers) does not use very advanced probability theory, it is quite possible that the simple approach described therein is only one of many ways to achieve this discrimination. It would therefore be of interest to find out whether there are other approaches that can succeed in achieving this discrimination.

ACKNOWLEDGEMENTS

I would like to express my deep gratitude to Prof. Michael A. White of the UT Southwestern Medical Center for patiently educating me in various aspects of cancer biology. I would also like to thank Tamer Başar, Aniruddha Datta, Vibhav Gogate, Steve Marcus and Roberto Tempo for useful comments and encouragement on various drafts of the paper. Any remaining errors are of course solely my responsibility.

REFERENCES

[1] Special Issue on Systems Biology, *IEEE Transactions on Automatic Control & IEEE Transactions on Circuits and Systems, Part I*, Mustafa Khammash, Claire Tomlin and M. Vidyasagar (Editors), January 2008.

[2] Special Issue on Systems Biology, *Automatica*, Frank Allgöwer and Francis J. Doyle III (Editors), June 2011.

[3] Siddhartha Mukherjee, *The Emperor of All Maladies*. Fourth Estate, London, UK, 2011.

[4] http://www.thefullwiki.org/Ancient_Egyptian_medicine

[5] http://seer.cancer.gov/statfacts/html/all.html

[6] International Human Genome Research Consortium, "Initial sequencing and analysis of the human genome," *Nature*, 409, 860-921, February 2001.

[7] J. C. Venter, M. D. Adams, E. W. Myers et al., "The sequence of the human genome," *Science*, 291, 1304-1351, 16 February 2001.

[8] http://cancergenome.nih.gov

[9] M. Chalife et al., "Green fluorescent protein as a marker for gene expression," *Science*, 263, 802-805, February 1994.

[10] David Bartel, "MicroRNAs: Genomics, biogenesis, mechanism, and function," *Cell*, 116, 281-297, 23 January 2004.

[11] Benjamin P. Lewis, Christopher B. Burge and David P. Bartel, "Conserved seed pairing, often flanked by adenosines, indicates that thousands of human genes are microRNA targets," *Cell*, 120, 15-20, 2005.

[12] Andrew Grimson et al., "MicroRNA targeting specificity in mammals: Determinants beyond seed pairing," *Molecular Cell*, 27, 91-105, 2007.

[13] David Bartel, "MicroRNAs: Target recognition and regulatory functions," *Cell*, 136, 219-236, 23 January 2009.

[14] http://www.targetscan.org

[15] D. Pe'er and M. Hacohen, "Principles and strategies for developing network models in cancer," *Cell*, 144, 864-873, 18 March 2011.

[16] Yongsoo Kim et al., "Principal network analysis: Identification of subnetworks representing major dynamics using gene expression data," *Bioinformatics*, 27(3), 391-398, February 2011.

[17] Katia Basso et al., "Reverse engineering of regulatory networks in human B cells," *Nature Genetics*, **37(4)**, 382-390, April 2005.

[18] http://www.ncbi.nlm.nih.gov/geo/

[19] A. J. Butte and I. S. Kohane, "Mutual information relevance networks: Functional genomic clustering using pairwise entropy measures," *Pacific Symposium on Biocomputing*, 418-29, 2000.

[20] Adam A. Margolin et al., "ARACNE: An algorithm for the reconstruction of gene regulatory networks in a cellular context," *BMC Bioinformatics*, 7(Supplement 1):S7, 20 March 2008.

[21] C. K. Chow and C. N. Liu, "Approximating discrete probability distributions with dependence trees," *IEEE Trans. Info. Thy.*, **14(3)**, 462-467, May 1968.

[22] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, (Second Edition), Wiley Interscience, New York, 2006.

[23] http://www.medterms.com

[24] Kai Wang et al., "Genome-wide identification of post-translational modulators of transcription factor activity in human B cells," *Nature Biotechnology*, 27(9), 829-839, September 2009.

[25] Wentao Zhao, Erchin Serpedin and Edward R. Dougherty, "Inferring connectivity of genetic regulatory networks using information theoretic criteria," *IEEE/ACM Trans. Comput. Biol. and Bioinf.*, 5(2), 262-274, April-June 2008.

[26] M. Sklar, "Fonctions de répartition à n dimension et leurs marges," *Publications de l'Institut Statistiques, Université de Paris*, 8, 229-231, 1959.

[27] F. Durante and C. Sempi, "Copula theory: An introduction," in P. Jaworski, F. Durante, W. Hrdle, and T. Rychlik (editors), *Copula Theory and its Applications*, Lecture Notes in Statistics – Proceedings. Springer, Berlin/Heidelberg, 2010.

[28] Peng Qiu, Andrew J. Gentles and Sylvia K. Plevritis, "Reducing the computational complexity of information theoretic approaches for reconstructing gene regulatory networks," *J. Comput. Biol.*, 17(2), 169-176, February 2010.

[29] Frank Spitzer, "Markov random fields and Gibbs ensembles," *American Mathematical Monthly*, 78(2), 142-154, February 1971.

[30] Judea Pearl, *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufmann, San Francisco, CA, 1988.

[31] Nir Friedman et al., "Using Bayesian networks to analyze expression data," *J. Comput. Biol.*, 7(3-4), 601-620, June-August 2000.

[32] Yoseph Barash and Nir Friedman, "Context-specific Bayesian clustering for gene expression data," *J. Comput. Biol.*, 9(2), 169-191, 2002.

[33] Nir Friedman, "Inferring cellular networks using probabilistic graphical models," *Science*, 303, 799- 805, February 2004.

[34] David Heckerman, "A tutorial on learning with Bayesian networks," in *Learning in Graphical Models*, Michael I. Jordan (Editor), MIT Press, Cambridge, MA, 1998.

[35] C. S. Karapetis et al., "K-ras mutations and benefit from cetuximab in advanced colorectal cancer", *New England Journal of Medicine*, 359(17), 1757-1765, October 23, 2008.

[36] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer Networks ISDN Systems*, 30(10), 107-117, 1998.

[37] E. Seneta, *Non-Negative Matrices and Markov Chains*, Springerl-Verlag, New York, 2006.

[38] Kakajan Komurov, Michael A. White and Prahlad T. Ram, "Use of data-biased random walks on graphs for the retrieval of context-specific networks from genomic data," *PLoS Computational Biology*, 6(8), 19 August 2010.

[39] Hideaki Ishii and Roberto Tempo, "Distributed randomized algorithms for the page rank computation," *IEEE Trans. Auto. Control*, 55(9), 1897-2002, September 2010.

[40] Warren J. Ewens and Gregory J. Grant, *Statistical Methods in Bioinformatics* (Second Edition), Springer-Verlag, New York, 2005.

[41] Ana Justel, Daniel Peña and Rubén Zamar, "A multivariable Kolmogorov-Smirnov test of goodness of fit," *Statistics & Probability Letters*, 35, 251-259, 1997.

[42] M. Vidyasagar, *Learning and Generalization: With Applications to Neural Networks and Control Systems* (Second Edition), Springer-Verlag, London, 2003.

[43] Virginia Goss Tusher, Robert Tibshirani and Gilbert Chu, "Significance analysis of microarrays applied to the ionizing radiation responses," *Proc. Nat'l. Acad. Sci.*, 98(9), 5116-5121, 24 April 2001.

[44] Aravind Subramanian et al., "Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles," *Proc. Nat'l. Acad. Sci.*, 102(43), 15545-15550, 25 October 2005.

[45] Bradley Efron and Robert Tibshirani, "On testing the significance of a set of genes,"' *The Annals of Applied Statistics*, 1(1), 107-129, 2007.

[46] Robert Tibshirani et al., "Diagnosis of multiple cancer types by shrunken centroids of gene expression," *Proc. Nat'l. Acad. Sci.*, 99(10), 6567-6572, 14 May 2002.

[47] L. D. Wood et al., "The genomic landscapes of human breast and colorectal cancers", *Science*, 318, 1108-1113, 16 November 2007.

[48] T. Sjöblom et al., "The consensus coding sequences of human breast and colorectal cancers", *Science*, 314, 268-274, Oct. 13, 2006.

[49] http://www.ncbi.nlm.nih.gov/unigene

[50] C. G. Son et al., "Database of mRNA gene expression profiles of multiple human organs", *Genome Research*, 15, 443-450, 2005.

[51] J. L. Bos, "ras oncogenes in human cancer: A review", *Cancer Research*, 49(17), 4682-4689, September 1, 1989.

[52] K. Koera et al., "K-Ras is essential for the development of the mouse embryo", *Oncogene*, 15(10), 1151-1159, September 4, 1997.

[53] N. Potenza et al., "Replacement of K-Ras with H-Ras supports normal embryonic developmentdespite inducing cardiovascular pathology in adult mice", *EMBO Reports*, 6(5), 432-437, 2005.

[54] Minh D. To *et al.*, "Kras regulatory elements and exon 4A determine mutation specificity in lung cancer", *Nature Genetics*, 40(10), 1240-1244, October 2008.

[55] http://www.sanger.ac.uk/genetics/CGP/cosmic

[56] C. Greenman et al., "Patterns of somatic mutation in human cancer genomes", *Nature*, 132, 153-158, 8 March 2007.

[57] J. B. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations", Proceedings of Fifth Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, 281-297, 1967.

[58] Ugur Eskiocak et al., "Functional parsing of driver mutations in the colorectal cancer genome reveals numerous suppressors of anchorage-independent growth," *Cancer Research*, 71, 4359-4365, July 1, 2011.

[59] Uri David Akavia et al., "An integrated approach to uncover drivers of cancer," *Cell*, 143(6), 1005-1017, 10 December 2010.

[60] Eduardo D. Sontag, "Network reconstruction based on steady-state data," *Essays in Biochemistry*, 45, 161-176, 2008.

[61] Réka Albert, Bhaskar Dasgupta and Eduardo Sontag, "Inference of signal transduction networks from double causal evidence," *Computational Biology*, 673, 239-251, 2010.

[62] Bhaskar Dasgupta, Paola Vera-Licona, Eduardo Sontag, "Reverse engineering of molecular networks from a common combinatorial approach," in *Algorithms in Computational Molecular Biology*, (Mourad Eiloumi and Albert Y. Zomaya, Eds.), Wiley, New York, 2011.

[63] Lorenzo Farina et al., "Identification of regulatory network motifs from gene expression data," *J. Math. Model. Algor.*, 9(3), 233-245, September 2010.

[64] Michael M. Zavlanos et al., "Inferring stable genetic networks from steady-state data," *Automatica*, 47(6), 1113-1122, June 2011.

[65] Ritwik Layek et al., "Cancer therapy design based on pathway logic," *Bioinformatics*, 27(4) , 548-555, 2011.

[66] Ritwik Layek, Aniruddha Datta and Edward R. Dougherty, "From biological pathways to regulatory networks," *Molecular Biosystems*, 7, 843-851, 2011.

[67] James A. Eddy et al., "Identifying tightly regulated and variably expressed networks by differential rank conservation (DIRAC)," *PLOS Computational Biology*, 5(5), May 2010.

Literature-Based Annotation of Gene Predictions[21]

Gene Id	Gene Name	Mutation Frequency	Literature Annotation
120	ADD3	2.7027027027	mutated in leukemia (translocation- oncogene)
529	ATP6V1E1	2.7027027027	
780	DDR1	2.380952381	mutated in lung cancer-oncogene
1281	COL3A1	5.4054054054	
1434	CSE1L	2.6315789474	
1499	CTNNB1	5.5152394775	mutated in cancer- oncogene
1981	EIF4G1	2.7027027027	
2030	SLC29A1	5.4054054054	mutated in cancer-drug resistance
2335	FN1	5.4054054054	
2720	GLB1	2.8571428571	
2778	GNAS_NM_016592_1	0.05	
2876	GPX1	9.0909090909	putative tumor suppressor
3417	IDH1	0.6787330317	oncogene
3486	IGFBP3	5.4054054054	tumor suppressor
3550	IK	2.8571428571	
3915	LAMC1	2.7027027027	
4131	MAP1B	5.4054054054	
4179	CD46	5.4054054054	
4313	MMP2	7.8947368421	
5591	PRKDC	7.1428571429	
5631	PRPS1	2.6315789474	
5754	PTK7	0.7518796992	
5878	RAB5C	2.6315789474	
5954	RCN1	2.7027027027	
6128	RPL6	2.7027027027	risk locus
6597	SMARCA4	7.4626865672	lung cacer-tumor suppressor
7052	TGM2	2.7027027027	
7153	TOP2A	8.33333333333	
7157	TP53	42.8651059086	
7247	TSN	2.7777777778	translocations
7358	UGDH	2.6315789474	
7385	UQCRC2	5.4054054054	
7431	VIM	2.6315789474	
8079	MLF2	2.7027027027	
8531	CSDA	2.7027027027	
8539	API5	2.8571428571	
8894	EIF2S2	2.7027027027	
9590	AKAP12	5.2631578947	
9993	DGCR2	9.0909090909	
10075	HUWE1	10.8108108108	
10291	SF3A1	2.7027027027	
10342	TFG	2.6315789474	translocations
11034	DSTN	2.7027027027	
22974	TPX2	2.6315789474	
22985	ACIN1	2.7027027027	
23020	SNRNP200	2.7027027027	
27131	SNX5	2.6315789474	
51150	SDF4	2.7027027027	
51322	WAC	2.7027027027	
51614	ERGIC3	5.4054054054	
54431	DNAJC10	2.7777777778	
55101	ATP5SL	2.8571428571	
57142	RTN4	2.7027027027	
65056	GPBP1	2.8571428571	
65125	WNK1	1.47058823529	
81887	LAS1L	2.6315789474	
83692	CD99L2	5.1282051282	
347733	TUBB2B	2.7027027027	

[21] Annotation by Prof. Michael A. White of UT Southwestern Medical Center