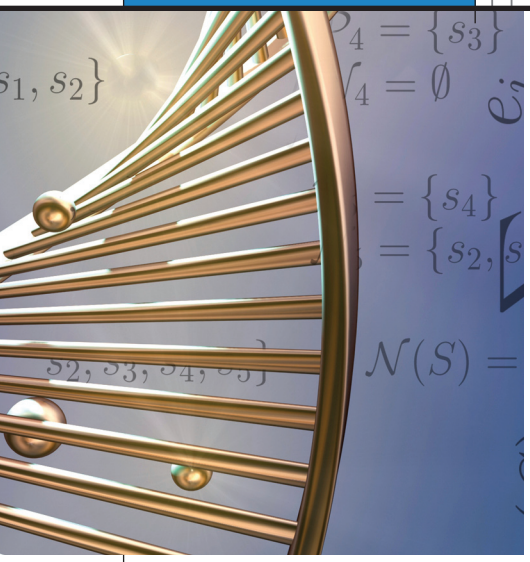

Algorithms and Genome Sequencing: Identifying Driver Pathways in Cancer

Fabio Vandin, Eli Upfal, and Benjamin J. Raphael, *Brown University*

Two proposed algorithms predict which combinations of mutations in cancer genomes are priorities for experimental study. One relies on interaction network data to identify recurrently mutated sets of genes, while the other searches for groups of mutations that exhibit specific combinatorial properties.

Cancer is a disease driven by somatic mutations in an individual's DNA sequence, or genome, that accumulate during the person's lifetime. These mutations arise during DNA replication, which occurs as cells grow and divide into two daughter cells. Mutations arise as errors in the DNA replication process and distinguish the DNA in the daughter cells from the parental cells. They take place on a continuum of scales—ranging from single "character" substitutions (the nucleotides A, C, T, and G of DNA) to structural variants that duplicate, delete, or rearrange larger genome segments. Single-nucleotide substitutions occur at a rate of approximately 10^{-9}, so that on average each daughter cell contains around six somatic mutations. Most are benign, or inconsequential for the organism. However, in certain circumstances, dangerous somatic mutations can accumulate in a collection of cells and lead to cancer.

Theodor Boveri first articulated the idea that mutations cause cancer in 1914, a remarkable insight as the structure of DNA, or even the concept of a gene, was not yet known. Decades later, cytogenetic techniques that researchers use to directly visualize chromosomes in cells led to the discovery of chromosomal abnormalities in cancer cells, resulting from large-scale rearrangements of the DNA sequence. In some types of leukemia, for example, chromosomes 9 and 22 undergo a translocation that swaps DNA between these chromosomes. Unfortunately, finding other important large-scale rearrangements has been a challenge. Many cancer cells contain dozens of chromosomal abnormalities, and these differ among individuals with the same type of cancer. A natural question is whether some or all of these rearrangements contribute to cancer or are merely random occurrences.

ADVANCES IN DNA SEQUENCING

The emergence of DNA sequencing enabled biologists to measure single-nucleotide mutations with increasing speed and accuracy. These studies showed that the "typical" cancer genome might have hundreds to thousands of somatic mutations of different types. However, most of the somatic mutations in a cancer cell are benign *passenger mutations*. A much smaller fraction of *driver mutations* are important for cancer development, with current estimates ranging from 10 to 20 driver mutations per tumor.

Because cancer cells have a large variety of relatively rare mutations, genome-wide studies for identifying cancer driver mutations require sequencing numerous patients. This task became feasible in the past five years with the development of next-generation sequencing technologies such as Roche's 454, Illumina's Genome Analyzer, and Applied Biosystems' SOLiD (Sequencing by Oligonucleotide Ligation and Detection), which provide low-cost, high-throughput sequencing through massive parallelism.[1] While each has unique characteristics, all of these technologies collect dozens or hundreds of millions of

0018-9162/12/$31.00 © 2012 IEEE Published by the IEEE Computer Society

short DNA sequences or *reads* simultaneously, corresponding to billions of DNA nucleotides. Improvements in these technologies are continuing at a rapid pace and are nearing the goal of producing a human (or cancer) genome at extremely low cost (less than $1,000).

Next-generation DNA sequencing has enabled large cancer-sequencing efforts including The Cancer Genome Atlas (TCGA; http://cancergenome.nih.gov) in the US and many others worldwide through the International Cancer Genome Consortium (ICGC; www.icgc.org). These projects identify somatic mutations in hundreds to thousands of patients with different types of cancer by sequencing each patient's tumor and, in some cases, the healthy tissue as well. In particular, TCGA aims to comprehensively identify genomic changes—including somatic mutations and other types of data—from about 20 different cancer types by 2014. For each cancer type, researchers will collect and analyze some 500 samples.

A key question for such projects is how to use the resulting DNA sequence to understand the mutations that cause specific properties of cancer cells.

The ability to measure mutations far exceeds the capacity to experimentally evaluate each mutation's function.

COMPUTATIONAL CHALLENGES

There are presently two main computational challenges in applying next-generation DNA sequencing to cancer genomes.

The first is how to derive catalogs of mutations in a genome from the data generated by a DNA sequencing machine. Although these machines produce a remarkable number of DNA sequences, these sequences are only reads (about 30 to 1,000 nucleotides), not full-length genomes. Obtaining the catalog of somatic mutations from such short sequences requires algorithmic techniques, an active area of investigation in recent years.[2]

Assuming we have obtained a list of all somatic mutations in the cancer genome, the second challenge is to distinguish the functional driver mutations from the random passenger mutations. The ultimate determinant of function is a biological experiment, but the ability to measure mutations far exceeds the capacity to experimentally evaluate each mutation's function. One way to predict candidate driver mutations is to examine the somatic mutations measured in a large population of cancer patients and identify *recurrent* mutations that occur more frequently than expected by chance, or, alternatively, recurrently mutated genes, which are genes that are mutated more frequently than expected.

CANCER GENE IDENTIFICATION

To formalize the problem of predicting recurrently mutated genes, we represent the measured somatic mutations as a binary *mutation matrix* with patients on the rows and genes on the columns, where a 1 in an entry indicates that the corresponding gene is mutated in the corresponding patient. Given a mutation matrix, the goal is to find genes that are mutated in more patients than expected by chance.

Suppose we were to predict driver genes as the genes that are mutated in the largest number of patients. For example, if we examine mutation data from 316 patients in a recent large-scale sequencing study of ovarian cancer,[3] the most frequently mutated gene is TP53 (tumor protein 53), a well-known cancer gene involved in DNA repair and other functions. The second most frequently mutated gene, TTN, is also special, but not because of a biological function in cancer: it is the largest gene in the human genome. Thus, TTN's high mutation frequency is explained by its exceptional length, not by its function. Distinguishing such cases requires a probabilistic model.

PROBABILISTIC MODEL

Our probabilistic model for cancer mutations, like many probabilistic models, is based on a coin-flipping experiment. Suppose that mutations occurred randomly with probability q. Then, for a given gene in one patient, the status of a gene (mutated or not) is an experiment with two possible outcomes. We model this as a coin flip resulting in H or T, with $Pr[H] = q$ and $Pr[T] = 1 - q$. For simplicity, assume that the coin is fair—that is, $q = 1/2$.

Given a set of N patients, the total number of heads in N coin flips is described by a binomial random variable of parameters N and $q = 1/2$. Since our goal is to find genes that harbor nonrandom mutations, we want to identify coins that are not fair, and to reject the claim of the coin being fair only if we are fairly certain. In mathematical terms, this means rejecting the claim if the probability of observing numerous heads is small.

We construct an algorithm, CoinFlip, that decides whether or not to reject the claim that the coin is fair based on the observed number R of heads, and a threshold α for rejecting the claim. CoinFlip is a special case of *hypothesis testing*. Formally, we define the *null hypothesis* that the coin is fair and the *alternative hypothesis* that the coin is biased toward heads. For a given threshold α, we reject the null hypothesis in favor of the alternative if the tail probability—the probability of obtaining at least R heads assuming the null hypothesis—is less than α. The smallest threshold for which we would reject the null hypothesis for a given observed value R is called the p-value of the test.

We now have a model of "expected by chance" to identify cancer genes. We run the CoinFlip algorithm for each gene, and thereby compute the probability, or p-value,

that the observed number of mutated patients or more is obtained under the null hypothesis. We assume a null hypothesis where all mutations are passenger mutations with probability q and then compute the tail probability using a binomial model. In particular, we examine a single gene g and model the presence or absence of a mutation in a patient as the outcome of a coin flip, with a fixed probability q of becoming heads; q depends on the gene's length and on the rate r that passenger mutations occur, which must be estimated from the data.

Obtaining an accurate estimate of r is challenging because the passenger mutation rate depends on many parameters whose values are not easily determined (for example, the times that tumor cells have divided into daughter cells). Moreover, r is usually assumed to be the same for all patients and all genes but in reality differs among patients and possibly among genes. Current methods approximate r, but the details of this approximation and other issues related to single-gene tests of recurrence are subjects of debate.[4]

We use the CoinFlip algorithm to determine whether the number of mutated patients is significantly higher than expected, as Figure 1 shows. If so, the gene is a candidate driver gene in cancer. This test (with a more detailed model for the passenger mutation rate) is essentially current practice in cancer genome projects.[5]

MULTIPLE HYPOTHESIS TESTING PROBLEM

We examine the same ovarian cancer data using the CoinFlip test, where for a given gene the passenger mutation probability is determined by a mutation rate per nucleotide and the gene's length. Three genes have p-values less than 0.01 and thus could be considered surprising: TP53, BRCA1, and RB1 (note that TTN is no longer statistically significant). These are among the most well-known cancer genes. In addition to the previously mentioned TP53, genetic variants in BRCA1 give increased risk of breast cancer and RB1 is mutated in retinoblastoma, a childhood cancer. Thus, a modern-day sequencing machine and straightforward statistical analysis have rediscovered decades of cancer research!

Unfortunately, the analysis has overlooked one small detail, what statisticians refer to as the *multiple hypothesis testing problem*.

To motivate this problem, we return to the CoinFlip algorithm. Suppose we choose a fixed threshold α to reject the null hypothesis that the coin is fair. Then, we make an error if we reject the hypothesis when the coin is fair. The probability of making an error is exactly α. If we run the algorithm on 50 fair coins, with a probability of error α for each coin, then the probability of *not* making a single error is $(1 - \alpha)^{50}$. Alternatively, Pr[at least one error] = $1 - (1 - \alpha)^{50}$, which is approximately 0.4 when $\alpha = 0.01$. Thus, if we apply CoinFlip to 50 fair coins, there is a 40

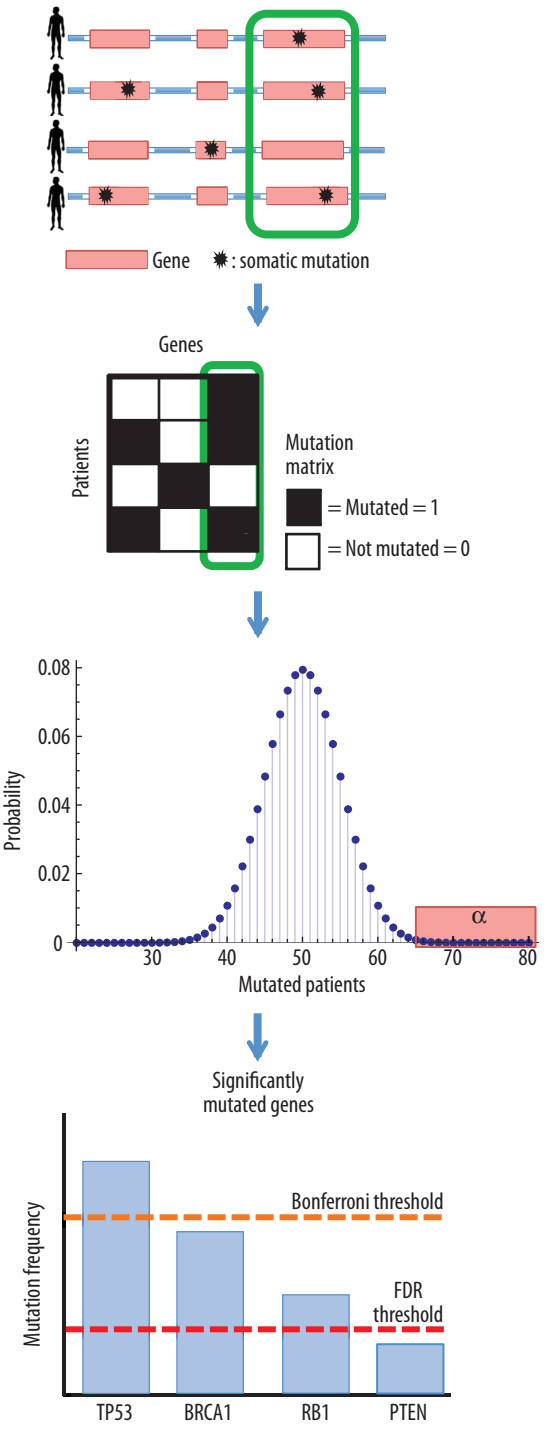

Figure 1. Given the observed somatic mutations in many cancer patients, CoinFlip—a simple algorithm based on the binomial distribution—finds those genes that are mutated in more patients than expected by chance.

percent probability that we will falsely reject the fair coin claim at least once. Moreover, the more fair coins we test, the less likely that all of the coins are called fair. Thus, even if the threshold α for deciding when to reject the claim of the coin being fair is small, if we test enough coins, CoinFlip will eventually reject the claim for one of the coins.

A more general, but in this case not as accurate, analysis is to see that the event of an error on at least one coin is the union of the events of errors on individual coins. Although these events are not disjoint, the sum of the probabilities of the single events bounds the probability of their union. This constitutes the Bonferroni correction, which in our scenario states that if we are testing n coins and want a bound α on the probability of incorrectly calling one or more coins as not fair, we can test each single coin with the CoinFlip algorithm using as error threshold α/n.

The resulting algorithm accounts for the fact that we use n coins. If we run this algorithm with a Bonferroni correction on the ovarian cancer data, the only statistically significant gene is TP53. This is somewhat disappointing given our knowledge about the importance of BRCA1 and RB1 in cancer, but statistical significance and biological significance are not always the same, particularly since statistical techniques might require many more samples than the 316 patients here. In fact, using more sophisticated techniques developed in the past 20 years that make multiple hypotheses corrections based on the *false discovery rate* (FDR),[6] we recover these two genes.

Methods that identify groups of genes with a significant number of mutations but do not restrict attention to only known pathways are desirable.

However, this data presents a larger problem. Even with the FDR technique we predict only a total of nine driver genes, many of which are mutated in only a small number of patients, and not enough to explain cancer in all patients. This phenomenon is not unique to the ovarian cancer data. Single-gene analysis techniques are inherently too weak to identify most driver mutations. This is due in part to the number of patients that were sequenced and errors in the mutation data. However, there is also a biological reason: driver mutations target groups of genes, or *pathways*.

DRIVER PATHWAYS

Genes do not act in isolation, but rather interact with other genes (and the proteins these genes produce) in complex signaling and regulatory networks. Cancer is often called a disease of pathways, as it is pathways, or groups of genes, that are mutated to perturb a particular function in cancer. There are many ways to deregulate a given pathway by mutating one of its genes, and each cancer patient might have mutations in a different subset of genes in an important pathway. Testing genes independently does not take into account the fact that genes interact with one another. Rather than test individual genes, we should test groups of genes.

Unfortunately, testing groups of genes is difficult to do exhaustively because there are too many groups to test. For example, there are about 10^{22} groups of six genes in the human genome. Not only must we perform a p-value calculation for each group, but we also must account for the number of hypotheses or groups that we test, using one of the multiple hypothesis correction procedures above. Therefore, standard practice in cancer genome studies is to assess enrichment of mutated genes only in pathways known to perform a certain function. Typically, this is done by treating a known pathway as a "bag of genes" (without considering the interactions between genes) and assessing whether mutations are enriched in the "concatenation of genes" using variations of the single-gene test.

However, this approach will not discover any new group, nor does it account for the fact that signaling pathways are interconnected in larger signaling networks. Pathways cannot be viewed in isolation, as the different pathways interact. Rather, the genes involved in cancer "affect multiple pathways that intersect and overlap."[7]

Thus, methods that identify groups of genes with a significant number of mutations but do not restrict attention to only known pathways are desirable. We recently developed two algorithms for this purpose. One considers all interactions within a cell, represented as a network or graph, and finds subnetworks that are mutated more than expected. The second algorithm uses no prior information about interactions between genes, but rather exploits some properties of the patterns of mutations that are expected for interacting genes.

HOTNET: MUTATED SUBNETWORKS

The first algorithm, HotNet,[8] considers a large-scale interaction network and mutation data from many patients, as Figure 2 shows. HotNet finds *subnetworks*, or clusters of interacting genes, that are mutated in a significant number of patients. The algorithm thus generalizes the analysis of recurrent mutations in single genes.

Human interaction network

The human interaction network is not presently known. However, researchers have assembled several large-scale interaction networks from various data sources including well-characterized experimental pathways, high-throughput interaction experiments, and computational predictions.[9] While these networks are incomplete

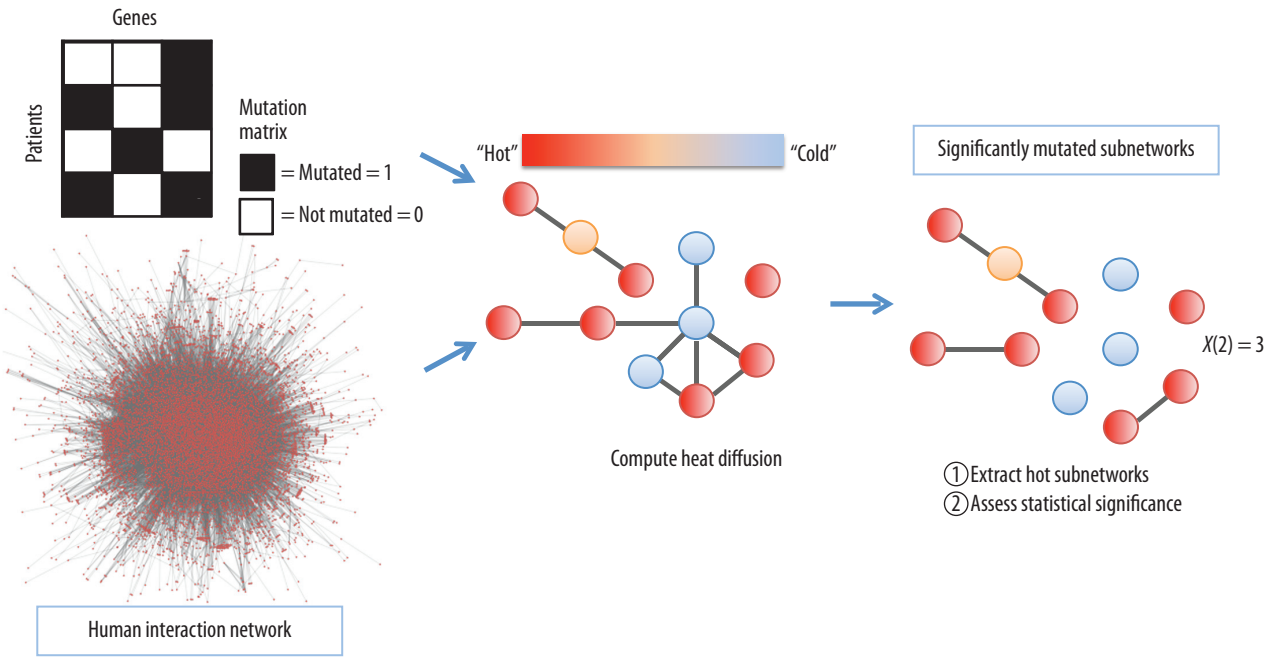

Figure 2. The HotNet algorithm combines mutation data and protein-protein interaction network information to find hot subnetworks, or clusters of interacting genes, that are mutated in a significant number of cancer patients. On each gene, HotNet places a source of heat proportional to the number of mutations on the gene. The heat diffuses on the network for a fixed time, revealing the hot subnetworks. Finally, a statistical test assesses the significance of the list of observed subnetworks.

and inaccurate, they encode useful information that researchers can combine with mutation data to identify genes important in cancer.

As the bottom left of Figure 2 shows, an interaction network can be represented as an undirected graph with nodes representing genes and edges representing interactions between them. Connected subgraphs constitute subnetworks.

One way to find subnetworks that are mutated in a significant number of patients is to test each possible subnetwork using an appropriate statistical test. However, there are two problems with this approach. First, testing many subnetworks is computationally difficult and reduces statistical power, as the test's p-value must be corrected for the number of subnetworks tested. This can be quite large for most interaction networks. For example, the number of subnetworks with at most six nodes in the network obtained from the Human Protein Reference Database exceeds 10^{10}. Second, subnetworks are not independent. An extreme example is provided by nodes with high degree, or hubs, in an interaction network. If these hub genes are mutated, a large number of subnetworks containing them will be flagged as significant.

HotNet addresses these problems in two ways. First, it uses a diffusion process on the interaction network to retain information about a gene's local topology while minimizing spurious connections from hubs. Second, it employs a new multihypothesis test that bounds the FDR of hot subnetworks.

Heat diffusion model

To understand the diffusion process, consider two scenarios. In one scenario, two mutated genes are connected by a single low-degree node in the network, while in the other, a high-degree node connects the mutated genes. Because there are many paths through the high-degree node, it is more surprising to see mutated genes connected by a path through a low-degree node in the network than mutated genes connected by a path through a high-degree node.

To formalize this intuition, we use a model of heat diffusion. Each mutation on a gene is a source of heat on the network and diffuses this heat to its neighbors. We place an amount of heat on a gene in proportion to the frequency of the gene's mutation and allow heat to diffuse over the edges for some length of time.

If we place the heat source on a low-degree node, heat will diffuse to the small number of neighbors in the graph, and thus these neighbors will remain hot for an appreciable length of time. On the other hand, heat placed on a high-degree node will diffuse to the many neighbors, and thus none of the nodes will be very hot. After allowing heat to diffuse for a fixed length of time, highly mutated

subnetworks will thus become hot spots on the graph; HotNet breaks the graph by removing cold edges, thus dividing the network into subnetworks. The algorithm assesses statistical significance by comparing the size of the resulting subnetworks to those obtained by performing the same procedure using an appropriate random model for mutations. HotNet uses a Laplacian matrix to compute heat diffusion.

The heat diffusion model is equivalent to a certain random walk on the graph and thus somewhat resembles the PageRank algorithm that Google originally used to rank webpages. However, a key difference is that PageRank and related algorithms examine only the graph's topology, while HotNet considers both the topology and the nodes' values.

Multihypothesis test

A two-stage multihypothesis test that bounds the FDR of the entire set of identified hot subnetworks circumvents the multiple hypothesis testing problem that arises if all subnetworks are tested as individual hypotheses. Our statistic is the number of subnetworks with at least a certain number of genes. In the first step, we assess the significance of the number $X(s)$ of subnetworks of a certain minimize size s. The number of measured genes now bounds the number of hypotheses to test, which is much smaller than the number of pathways. We can thus determine an s such that the number $X(s)$ of connected components of size $\geq s$ is significant. However, the fact that $X(s)$ is significant does not imply that any of the individual subnetworks is significant. Thus, we add a second step that rigorously bounds the FDR of the list of hot subnetworks.

Example application

We applied the HotNet algorithm to mutation data from the 316 ovarian cancer patients whose genes were sequenced as part of the TCGA project.[3] Using a large protein interaction network with more than 37,000 interactions, we found 33 hot subnetworks whose genes were mutated in a significant number of patients. Based on our statistical test, around one-third of these subnetworks would be expected to be true discoveries. Moreover, nearly one-third of these subnetworks corresponded to groups of proteins with known biological function.

DENDRIX: DE NOVO DRIVER EXCLUSIVITY

Because biological interaction networks are far from complete, as the number of patients increases, it might become possible to identify groups of mutated genes without the network. Indeed, the network's primary utility is to reduce the number of groups (hypotheses) to test. However, without the network, there are too many groups of genes to test exhaustively, since considering all the groups up to a reasonable size would be computationally inefficient and result in a loss of statistical power.

Mutual exclusivity and coverage

Current knowledge of mutations in cancer provides two constraints on groups of genes to examine. First, because a driver mutation is rare, if a group of genes (or a pathway) is important for cancer, typically only a single gene in the group will be mutated in a patient. Thus, there is a pattern of *mutual exclusivity* between driver mutations. Second, an important cancer pathway will be mutated in most patients. Thus, the mutations in a pathway important for cancer show high patient *coverage*.

These constraints led us to examine particular mutation patterns in the mutation matrix. First, mutual exclusivity implies that we want to identify a group of genes, or columns, in the matrix such that each patient (row) has at most one mutation. We refer to this as an *exclusive submatrix*. We define the coverage of a submatrix as the number of rows with at least one mutation in the group of columns (genes). We are interested in exclusive submatrices that cover many patients—that is, for which many patients have a mutation in at least one gene.

We thus define the *maximum coverage exclusive submatrix problem*: find the exclusive submatrix with k columns with maximum coverage. This problem is NP-hard, therefore no algorithm efficient in all instances is expected to exist for its solution. Perhaps more importantly, the exclusivity constraint is too restrictive for real data where errors or passenger mutations might result in a pathway important for cancer to present nonexclusive mutations.

We thus focus on finding approximately exclusive sets of genes (columns) that cover many patients. Let $\Gamma(g)$ denote the set of patients with a mutation in gene g. We define the *coverage overlap* of a set M of genes to be the difference between the sum of the coverages of the single genes in the set and the coverage of the set

$$\gamma(M) = \sum_{g \in M} |\Gamma(g)| - |\Gamma(M)|.$$

Our goal is to simultaneously maximize coverage and minimize coverage overlap. There is an inherent tradeoff in these criteria, so we define the weight of a set of genes as the difference between coverage and coverage overlap:

$$W(M) = |\Gamma(M)| - \gamma(M) = 2|\Gamma(M)| - \sum_{g \in M} |\Gamma(g)|.$$

We thus define the *maximum weight submatrix problem*: find the submatrix with k columns with maximum weight. This problem is also NP-hard.

MCMC-based solution

We developed two algorithms to solve the maximum weight submatrix problem. The first is a simple greedy algorithm that yields maximum weight submatrices with high probability when the data comes from a generative model well suited for single-nucleotide mutations. The

second algorithm uses a *Markov chain Monte Carlo* method to sample submatrices in proportion to their weight. The MCMC method does not require any assumptions about the data.

The greedy algorithm iteratively adds columns (genes) that increase the weight. This algorithm will return the driver pathway provided the mutations follow a particular independent-genes model that is reasonable for some types of cancer data and the number of samples is reasonably large. Unfortunately, the bounds obtained require a number of samples that is an order of magnitude larger than what is currently available.

Thus, we also developed an MCMC approach. With this approach, we consider different gene sets of fixed size k as states of a Markov chain, and transitions are substitutions of a gene in a set. We use the Metropolis-Hastings procedure to define the transition probabilities between states so that the Markov chain converges to the desired distribution where the probability of a set M is proportional to its weight $W(M)$. According to Markov chain convergence theory, under certain reasonable assumptions, running the chain long enough converges to a stationary distribution. In general, the Metropolis-Hastings procedure is guaranteed to converge to the desired distribution, but the time to convergence can be very long. Thus, MCMC approaches use various heuristics to determine how many transitions are necessary before outputting a state.

In our case, we prove that the Markov chain converges rapidly, making it possible to efficiently sample from the distribution of gene sets. A major advantage of the MCMC approach is that it samples from distributions of sets rather than identifying a single optimal set. Moreover, unlike the greedy algorithm, it does not require any assumptions about the mutations.

We implemented the resulting MCMC method as the De novo Driver Exclusivity (Dendrix) algorithm[10] and ran it on simulated data and real cancer sequencing data. The sets of genes that Dendrix sampled with high frequency interact or have common interacting partners in well-known cancer signaling pathways. On brain cancer (glioblastoma) data, our method identifies three sets of genes that form parts of signaling pathways: one consists of three genes, as Figure 3 shows, and the other two consist of two genes. Thus, Dendrix automatically discovers groups of interacting genes solely from the pattern of mutations in the genes.

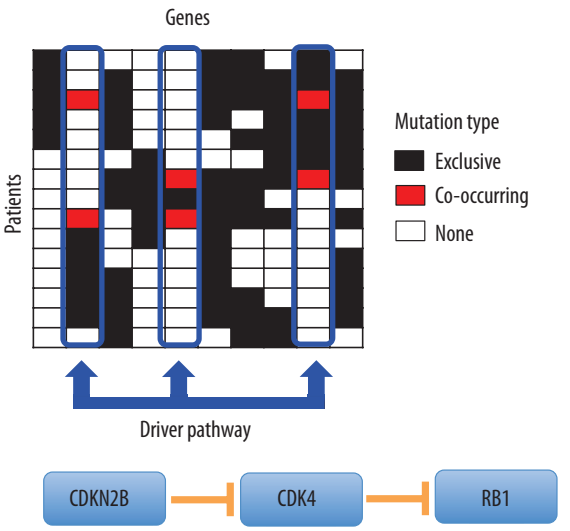

Figure 3. The De novo Driver Exclusivity (Dendrix) algorithm uses a Markov chain Monte Carlo method to sample submatrices of the mutation matrix with high coverage and exclusivity. In this case, Dendrix identifies a set of three genes—CDKN2B, CDK4, and RB1—in brain cancer data corresponding to a known important pathway.

The analysis of cancer genome data presents many computational challenges. Here, we have focused on one of the key challenges: to distinguish driver mutations relevant for cancer development from passenger mutations that do not have functional implications and are not important for cancer. While an experiment provides the ultimate evidence that a mutation is functional, the large amount of cancer sequences now available demand new computational approaches to prioritize mutations for biological validation. Going a step further, such approaches are also useful to predict which combinations of mutations are driver mutations.

Appropriate algorithms can significantly reduce the number of combinations of possible driver mutations that need to be tested in expensive and time-consuming wet lab experiments. These algorithms need to be time and space efficient to handle massive datasets, and they must rely on rigorous statistical methods to reduce the number of candidate driver mutations (false positives) while also not eliminating true driver mutations from consideration (false negatives).

Both of the algorithms that we designed to find groups of genes that are functionally redundant for the development of cancer generalize the single-gene test that is commonly used to identify driver genes by their recurrence in many cancer patients. Moreover, neither algorithm restricts groups of genes to those already known to be involved in cancer, thus allowing the discovery of novel combinations of mutations. The HotNet algorithm relies on prior knowledge of the interactions between genes, represented as a graph, to restrict the search space of possible combinations. The Dendrix algorithm exploits some combinatorial properties of the patterns of mutations that are expected for driver pathways.

More research is required to develop better algorithms for the identification of driver genes and driver pathways,

and to use the resulting information to improve cancer treatments. While here we focused on mutation data, a wealth of other types of genomic and epigenomic data—on gene expression, DNA methylation, and so on—can be combined to make more accurate predictions. The Cancer Genome Atlas and other similar projects are collecting multiple data types on the same patients that can be used for such research. Finally, identifying the driver mutations and pathways is only a first step toward understanding how these mutations affect a particular patient's prognosis and treatment.

The data to address all of these questions is being produced at a rapid pace, and the major challenge for computational biologists going forward is to interpret this data.

References

1. S.C. Schuster, "Next-Generation Sequencing Transforms Today's Biology," *Nature Methods*, Jan. 2008, pp. 16-18.
2. M. Meyerson, S. Gabriel, and G. Getz, "Advances in Understanding Cancer Genomes through Second-Generation Sequencing," *Nature Reviews Genetics*, Oct. 2010, pp. 685-696.
3. The Cancer Genome Atlas Research Network, "Integrated Genomic Analyses of Ovarian Carcinoma," *Nature*, 30 June 2011, pp. 609-615.
4. G. Parmigiani et al., "Response to Comments on 'The Consensus Coding Sequences of Human Breast and Colorectal Cancers,'" *Science*, 14 Sept. 2007, p. 1500.
5. N. Stransky et al., "The Mutational Landscape of Head and Neck Squamous Cell Carcinoma," *Science*, 26 Aug. 2011, pp. 1157-1160.
6. Y. Benjamini and Y. Hochberg, "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing," *J. Royal Statistical Soc., Series B (Methodological)*, vol. 57, no. 1, 1995, pp. 289-300.
7. F. McCormick, "Signalling Networks That Cause Cancer," *Trends in Cell Biology*, Dec. 1999, pp. M53-M56.
8. F. Vandin, E. Upfal, and B.J. Raphael, "Algorithms for Detecting Significantly Mutated Pathways in Cancer," *J. Computational Biology*, Mar. 2011, pp. 507-522.
9. T. Ideker and R. Sharan, "Protein Networks in Disease," *Genome Research*, Apr. 2008, pp. 644-652.
10. F. Vandin, E. Upfal, and B.J. Raphael, "De Novo Discovery of Mutated Driver Pathways in Cancer," *Genome Research*, Feb. 2012, pp. 375-385.

Fabio Vandin *is a research assistant professor in the Department of Computer Science at Brown University. His research interests include the design of algorithms and applications for cancer genomics, structural proteomics, biological sequence analysis, and database mining. Vandin received a PhD in information engineering from the University of Padua, Italy. He is a member of the International Society for Computational Biology and the European Association for Theoretical Computer Science. Contact him at vandinfa@cs.brown.edu.*

Eli Upfal *is a professor in the Department of Computer Science at Brown University. His research focuses on the design and probabilistic analysis of algorithms, and on statistical methods in computation. Upfal received a PhD in computer science from the Hebrew University of Jerusalem, Israel. He is a Fellow of IEEE and ACM. Contact him at eli@cs.brown.edu.*

Benjamin J. Raphael *is an associate professor in the Department of Computer Science at Brown University. His research focuses on the design of combinatorial and statistical algorithms for the interpretation of genomes. Raphael received a PhD in mathematics from the University of California, San Diego, and completed postdoctoral training in bioinformatics and computer science at UCSD. Contact him at braphael@cs.brown.edu.*