Bioinformatics, 32(24), 2016, 3717–3728
doi: 10.1093/bioinformatics/btw546
Original Paper

OXFORD

Genome analysis

A new correlation clustering method for cancer mutation analysis

Jack P. Hou[1,2,†], Amin Emad[3,4,†], Gregory J. Puleo[3,4], Jian Ma[1,5,6,*] and Olgica Milenkovic[3,4,*]

[1]Department of Bioengineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA, [2]Medical Scholars Program, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA, [3]Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA, [4]Coordinated Science Lab, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA, [5]Carl R. Woese Institute for Genomic Biology, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA and [6]Computational Biology Department, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA

*To whom correspondence should be addressed.
†These two authors contributed equally.
Associate Editor: John Hancock

Received on May 4, 2016; revised on June 14, 2016; accepted on August 16, 2016

Abstract

Motivation: Cancer genomes exhibit a large number of different alterations that affect many genes in a diverse manner. An improved understanding of the generative mechanisms behind the mutation rules and their influence on gene community behavior is of great importance for the study of cancer.

Results: To expand our capability to analyze combinatorial patterns of cancer alterations, we developed a rigorous methodology for cancer mutation pattern discovery based on a new, constrained form of correlation clustering. Our new algorithm, named C^3 (Cancer Correlation Clustering), leverages mutual exclusivity of mutations, patient coverage and driver network concentration principles. To test C^3, we performed a detailed analysis on TCGA breast cancer and glioblastoma data and showed that our algorithm outperforms the state-of-the-art CoMEt method in terms of discovering mutually exclusive gene modules and identifying biologically relevant driver genes. The proposed agnostic clustering method represents a unique tool for efficient and reliable identification of mutation patterns and driver pathways in large-scale cancer genomics studies, and it may also be used for other clustering problems on biological graphs.

Availability and Implementation: The source code for the C^3 method can be found at https://github.com/jackhou2/C3

Contacts: jianma@cs.cmu.edu or milenkov@illinois.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Rapid advances in high-throughput sequencing technologies have provided unique opportunities for analyzing large numbers of cancer genomes. However, the complexity of genomic alterations in cancer causes significant analytical and computational challenges that have to be overcome in order to fully characterize the functional roles of various mutations. In particular, as cancer genomes tend to contain a large number of diverse mutations (e.g. point mutations or copy number changes) most of which are neutral, one problem of significant importance is to identify a small set of mutations that perturb key biological pathways and have significant impact on tumorigenesis (Hanahan and Weinberg, 2011). Hence, a central question in

cancer genomics is how to distinguish 'driver' mutations, which contribute to tumorigenesis, from functionally neutral 'passenger' mutations.

Many computational methods have been developed to facilitate the discovery of driver genes (Carter *et al.*, 2009; Dees *et al.*, 2012; Gonzalez-Perez and Lopez-Bigas, 2012; Lawrence *et al.*, 2013; Manolakos *et al.*, 2014), most of which rely on mutation counts. Due to the high level of inter-tumor heterogeneity, two patients with the same cancer may have vastly different drivers and as a result many cancer mutations occur with low frequency in the patient population. Therefore, approaches relying on simple estimates of recurrence or frequency of mutations usually do not work well in practice. To mitigate this problem, several recent approaches have integrated frequency analysis with pathway-based and network-based models in order to ensure high accuracy of common driver mutation discovery (Bashashati *et al.*, 2012; Hou and Ma, 2014; Ng *et al.*, 2012; Paull *et al.*, 2013; Pe'er and Hacohen, 2011). Such methods have an advantage in so far that in addition to mutation analysis, they take into account gene interactions as an added source of prior knowledge.

In parallel, methods have been proposed to identify driver pathways, i.e. groups of genes that may interact together in combinatorial patterns to promote tumorigenesis. Ciriello *et al.* (2012) described a method called MEMo, and subsequently used it to show that mutually exclusive modules based on known networks can aid in determining groups of genes that contribute to tumorigenesis. These gene groups, or modules, are jointly highly recurrent, have similar pathway impact in terms of biological processes, and their corresponding mutations tend to be mutually exclusive, meaning that very often only one gene in each gene group is mutated at a given time in any given patient. This mutual exclusivity rule in cancer pathways is supported by the observations that, in general, one mutated gene suffices to perturb the function of its corresponding pathway. Multiple mutations would require significantly higher energy investments on the part of cancer cells, and are hence selected against. Zhang *et al.* (2013) expanded the ideas behind the concept of MEMo with iMCMC, and provided a framework to integrate mutation data, copy number and expression information into cancer network weights which they used to identify modules; they also performed multiple types of integrative cancer perturbation data analysis. Dendrix (Vandin *et al.*, 2012) was developed to identify driver pathways *de novo* using mutual exclusivity and coverage (patient coverage) principles, without relying on known network information that has the potential to improve the discovery process of new modules. MDPFinder from (Zhao *et al.*, 2012) expanded on the overall framework of Dendrix by incorporating gene expression information to ensure that genes in discovered mutually exclusive pathways were also co-expressed. Multi-Dendrix (Leiserson *et al.*, 2013) and CoMDP (Zhang *et al.*, 2014) improved on the limitations of Dendrix and MDPFinder, respectively, by allowing their algorithms to find multiple co-occuring modules. More recently, CoMEt (Leiserson *et al.*, 2015a) was proposed to address an inherent bias in Dendrix and Multi-Dendrix that resulted in high frequency mutations being significantly more likely to be included in mutually exclusive modules.

However, while methods such as Dendrix, Multi-Dendrix and CoMEt all have the ability to identify mutually exclusive modules *de novo*, they still have significant limitations. The aforementioned methods are typically inefficient when applied to large-scale datasets with large values of their relevant parameters. Also, some of these methods are randomized in nature and no guarantees exist that multiple runs of the methods will produce compatible results. Furthermore, almost all methods are able to identify only a small number of modules of *limited size*, as cluster sizes are critical algorithmic parameters from the perspective of computational tractability. Most importantly, they have to be redesigned or restructured whenever new biological information is included in the discovery process.

To overcome these and other shortcomings of existing methods, we introduce a novel method called Cancer Correlation Clustering (C^3) to directly tackle the problems of integrating diverse sources of evidence regarding driver pattern behavior and eliminating computational bottlenecks associated with large cluster sizes or cluster numbers. The C^3 method uses a *new* agnostic optimization framework specifically developed and rigorously analyzed for the driver discovery task, in which patient data is converted into a simple set of weights used in the objective function that do not require the algorithm to change upon incorporation of new data sources. In addition to this flexibility, C^3 has low computational cost, and it allows for adding relevant problem constraints while retaining good theoretical performance guarantees. Furthermore, the algorithm outperforms CoMEt in *three out of four evaluation criteria*, where the three criteria depend on which weights are 'emphasized' in the optimization problem: tuning the weights allows one to select which features to improve or emphasize. What the relevant constraints features are may be chosen by the user, although our analysis included coverage, mutual exclusivity, expression data and network pathway information. We also point out that the weights may be chosen so as to cater to the need of many other computational biology problems that involve optimization on graphs.

To test C^3, we ran extensive simulations for seven cancer types (including breast cancer, kidney cancer, ovarian cancer, glioblastoma, etc.). Unfortunately, the patient sample set sizes for all except two cancers—breast cancer and glioblastoma—did not allow for accurate and statistically significant driver identifications for any of the used methods. We hence report results for these two cancers only, although a pan-cancer study is easy to conduct once sufficiently many samples become available.

The paper is organized as follows. A basic introduction of the principles of correlation clustering is provided in section Approach. Section Methods contains a description of how to transform patient data into clustering weights used for the computations, the algorithmic clustering approach based on the computed weights, and the evaluation criteria used to compare C^3 and CoMEt. Section Results contains the main results of our analysis, a comparison of the performance of C^3 and CoMEt on breast cancer and glioblastoma data. A discussion of our findings and concluding remarks are given in Discussions. A rigorous mathematical performance analysis of C^3 may be found in the Supplementary Materials, along with more extensive software evaluations and explanations of relevant concepts.

2 Approach

The basic idea behind the C^3 approach is *correlation clustering*, an agnostic learning technique first proposed in Bansal *et al.* (2004). In the most basic form of the clustering model, one is given a set of objects and, for all or some pairs of objects, one is also given an assessment as to whether the objects are 'similar' or 'dissimilar'. This information is described using a complete graph with labeled edges: each object is represented by a vertex of the graph, and the assessments are represented by edges labeled with either a '+' symbol, for similar objects, or a '-' symbol, for dissimilar objects. The goal is to partition the objects into clusters so that the edges within clusters

are mostly positive and the edges between clusters are mostly negative. Unlike in many other clustering models, such as *k*-means (Hartigan and Wong, 1979), the number of clusters is not fixed ahead of time and finding the optimal number of clusters is part of the problem. Furthermore, the assignment of positive and negative edges does not have to be mutually consistent: for example, if the graph contains a triangle with two positive edges and one negative edge, then we must either group the endpoints of the negative edge together, erroneously putting a negative edge inside a cluster, resulting in a 'negative error' or else we must group them separately, forcing one of the positive edges to erroneously go between clusters, resulting in a 'positive error'. An illustrative example is shown in Supplemental Figure S1. When a perfect clustering is not possible, we seek an *optimal* clustering: one that minimizes the total number of 'errors'. This form of correlation clustering is known to be NP-hard, but depending on the graph topology, various constant or logarithmic approximation guarantees exist.

Bansal *et al.* (2004) also proposed a weighted version of the correlation-clustering problem. A more general weighted formulation was introduced in Charikar *et al.* (2003), and this is the formulation we subsequently generalize. In this model, each edge e is assigned two nonnegative weights, w_e^+ and w_e^-. A clustering incurs cost w_e^+ if e is placed between clusters, and incurs cost w_e^- if e is placed within a cluster.

If no restrictions are placed on the weights w_e^+ and w_e^-, then it is possible to have edges with $w_e^+ = w_e^- = 0$; these edges are effectively absent from the graph, so there is no loss of generality in assuming that the graph is a complete graph. Nevertheless, in order to arrive at problems that have efficient constant approximation algorithms, one needs to place certain restrictions on w_e^+ and w_e^-. The *probability constraints* give a natural restriction on the edge weights $w_e^+ + w_e^- = 1$ for every edge e. Another restriction involves the *triangle inequality*, and one requires that $w_{uw}^- \leq w_{uv}^- + w_{vw}^-$ for all distinct vertices u, v and w.

The analytic approach pursued in this work operates on the following model: genes which show sufficiently large mutation prevalence in cancer patients represent vertices of a *complete (fully connected) graph* whose vertices are to be clustered according to similarity criteria and weights to be described in detail in the next section. Note that we only use the top 5% of mutated genes in cancer patients, ordered by mutation frequency, as vertices. The reasoning behind our approach is as follows: First, low-frequency mutations require *specialized statistical and network analysis methods* which have to be developed in parallel and for which not sufficiently many patient samples are yet available (Torkamani and Schork, 2009; Vogelstein *et al.*, 2013); Second, even when restricting our attention to the most frequently mutated genes we outperform all known methods, which illustrates that one can significantly scale down the set of genes under consideration and at the same time improve identification performance. The low-frequency trimming approach results in 170 genes in glioblastoma (GBM) and 130 genes in breast cancer (BRCA). Although these numbers may appear prohibitively small given that more than a hundred cancer driver genes are reported, usually only a very small number of driver genes are needed to initiate the process of tumorigenesis. (For example, in Tomasetti *et al.* (2015), it was shown that only three driver gene mutations are required for the development of lung and colorectal cancers.)

The weights w_e^+ and w_e^- assigned to an edge e connecting two genes u and v are weighted sums of weights capturing driver gene features, such as mutual exclusivity, coverage strength, network distance and expression similarity. More precisely, the negative weights w_e^- are chosen to be relatively small if the endpoint genes describing the edge are deemed to be mutually exclusive in cancer patients. A small negative weight encourages placing mutually exclusive genes *within the same cluster*, as the penalty paid for placement in the same cluster is small. The positive weights jointly depend on the coverage, network distance and expression correlation of the endpoint genes: The larger the joint coverage, co-expression and inverse of the network distance of the endpoint genes, the larger the positive weight and the more likely the genes will end up in the same cluster so as to avoid paying a large cross-cluster cost. Precise mathematical formulations of the weight functions will be provided in the next section.

To control the size of the resulting clusters so as to discourage uninformative singleton and giant clusters, we developed two new correlation clustering algorithms that use cluster sizes as problem parameters that may be chosen by the users. These cluster size bounds also allow for more accurate comparison with other methods which operate with inherent cluster size constraints. Furthermore, as pointed out in Vandin *et al.* (2012), driver pathways obeying mutual exclusivity and coverage constraints are usually smaller than most pathways annotated in the literature. This observation provides another reason for using bounded cluster sizes as well. Note that unlike in the aforementioned known methods, the cluster sizes have no bearing on the complexity of our algorithm nor on their overall approximation quality, and they may be completely removed by the user if so desired.

The driver discovery approaches closest to C^3 are Multi-Dendrix (Leiserson *et al.*, 2013) and CoMEt (Leiserson *et al.*, 2015a). Multi-Dendrix is an integer linear programming clustering algorithm that ensures that the genes within a cluster have mutation patterns that satisfy mutual exclusivity and coverage: In a nutshell, for any two genes in a cluster, the number of patients in which these genes are mutated at the same time is relatively small; in addition, a large portion of the patients has at least one mutation in each cluster. CoMEt uses a statistical score for mutation exclusivity that is conditioned on the frequency of each alteration, alleviating the inherent bias caused by frequently mutated genes. Compared to Multi-Dendrix and CoMEt, C^3 uses a *weighted* linear programming relaxation instead of an integer linear program which significantly improves the versatility and running time of the algorithm. Furthermore, the weights allow for straightforward incorporation of heterogeneous sources of evidence into the clustering method and the algorithm itself remains unchanged with the addition of new data. On the other hand, Multi-Dendrix cannot be easily adapted to new problem constraints. This flexibility comes at the cost of C^3 providing only an approximate solution, but the approximate solutions exhibit large overlap with the exact solutions for a number of tested smaller synthetic networks. In addition, given the inherently approximate nature of optimization criteria, the weight selection and parametrization of both algorithms, this does not appear to be a significant shortcoming. Also, empirical evaluations on real data suggest that the approximation algorithms produce results very close to the optimal solution.

3 Methods

Before rigorously describing our algorithmic methods, we introduce some relevant notation and explain how to estimate appropriate clustering weights based on available data. The weights are defined separately for each combination of datasets in order to better explain the trade-offs between different choices of weights and to allow the user to restrict her/his attention to the combination for

which the best and largest collection of data points and samples is available. In Results section, we describe the performance of C^3 for all combinations of datasets and corresponding weight choices.

3.1 Clustering weights

Let $G(V, E)$ be a complete graph, where $V(G)$ denotes the set of vertices and $E(G)$ denotes the set of edges of the graph G, respectively. As the graph is complete, an edge exists between any pair of vertices and the only relevant edge property is its pair of weights. Note that we follow this formalism as it is an established approach in correlation clustering. The symbol $e \in E(G)$ or $e = uv$ with $u, v \in V(G)$ is used to denote a generic edge. Each edge is assigned a positive weight w_e^+ and a negative weight w_e^-. Recall the interpretation of these weights: For two distinct vertices $u, v \in V(G)$, w_{uv}^+ is the cost of placing u and v in different clusters; consequently, by making the positive weight of an edge large, one can discourage placing the corresponding two genes into different clusters. Similarly, w_{uv}^- is the cost of placing u and v in the same cluster, and hence making this weight large discourages placing the corresponding two genes into the same cluster. In the rest of this section, we will explore different ways of defining the weights w_{uv}^- and w_{uv}^+; in order to avoid confusion between the different definitions, each weight we define will include a parenthetical abbreviation, so that, for example, $w^+(\mathrm{c})_{uv}$ will refer to the positive weight of uv defined according to the coverage criteria, while $w^+(\mathrm{c},\mathrm{n})_{uv}$ will refer to the positive weight of uv according to the coverage and network criteria. The weights are computed using four types of datasets: gene mutation data, copy number variation (CNV), network information (NI) and gene expression (GE) data. As each data type carries information of different importance to the driver discovery process (for example, CNVs are directly connected to driver gene properties, while GE data may only carry indirect information; CNV may also causally influence GE), when fusing different sources of information we allow for linear combinations of the weights w_{uv}^- and w_{uv}^+ corresponding to individual data sources based on their importance or accuracy (for example, there are still a number of unresolved issues in computing the exact CNV of a gene, while a large number of GE datasets are very noisy). To illustrate these points, if one source were to suggest positive and negative weights $w_{uv}^-(1)$ and $w_{uv}^+(1)$, while another source were to suggest $w_{uv}^-(2)$ and $w_{uv}^+(2)$, where the second source is deemed twice more important or accurate, the resulting weights would be obtained through a linear parameter fusion equation

$$w_{uv}^- = (1/3)w_{uv}^-(1) + (2/3)w_{uv}^-(2),$$

$$w_{uv}^+ = (1/3)w_{uv}^+(1) + (2/3)w_{uv}^+(2).$$

In our weight assignment process, we also make frequent use of the notions of coverage and mutual exclusivity which we roughly described in previous sections. Coverage refers to the number of patients in which the same mutation is observed. High coverage postulates that important driver pathway should be mutated in as many patients as possible. Mutual exclusivity refers to the property that mutated driver genes in a patient tend to belong to different pathways, and that the number of patients with more than one mutated driver gene per given pathway is small.

Let n_p denote the number of samples (i.e. patient genomes available) and let n_g denote the number of genes. Also, let $\mathbf{A} \in \{0,1\}^{n_g \times n_p}$ denote the matrix containing mutation data: If gene i is mutated in sample (patient) j, we set $\mathbf{A}(i,j) = 1$; otherwise, we set $\mathbf{A}(i,j) = 0$. Also, let $\mathbf{C}$ be an $n_g \times n_p$ matrix representing the CNV data: we set $\mathbf{C}(i,j) = 0$ if there is no change in the copy number of gene i in sample j; otherwise, we choose an integer value reflecting the deviation of the CNV number from its baseline. Hence, the CNV matrix contains both positive and negative values corresponding to the copy number changes of the corresponding gene in each sample.

To combine CNV and mutations, we combine the matrices A and C as follows: We form a new *binary matrix* $\mathbf{M} \in \{0,1\}^{n_g \times n_p}$ such that

$$\mathbf{M}(i,j) = 0, \text{ if } \mathbf{A}(i,j) = 0 \text{ AND } l_{cnv} < \mathbf{C}(i,j) < h_{cnv}, \qquad (1)$$

and $\mathbf{M}(i,j) = 1$ otherwise. In this formulation, l_{cnv} and h_{cnv} are lower and upper bounds on copy numbers that may be chosen by the user. These bounds determine what is deemed to be a significant CNV change. In our tests, we set $l_{cnv} = -1$ and $h_{cnv} = 3$, although other options are clearly possible. It is worth pointing out that more conservative CNV thresholds tend to decrease coverage, while more relaxed CNV assumptions tend to decrease mutual exclusivity. Based on the procedure above, we arrive at one 'combined mutation' matrix $\mathbf{M}$ which we use instead of the matrices $\mathbf{A}$ and $\mathbf{C}$, as it captures both mutations and CNVs. A positive entry in row i and column j of the mutation matrix $\mathbf{M}$ indicates that gene i is deemed mutated in sample j. A zero entry indicates that no mutation is recorded. Note that we are going to use this matrix in our future evaluations only for the purpose of indicating mutations, and counting mutations per gene (through row entry summation, or equivalently, by counting the number of patients that are deemed to have the gene mutated); or mutations per patient (through column entry summation, or equivalently, by counting the number of genes that are deemed to be mutated in the patient).

Finally, let $\mathbf{Z} \in R^{n_g \times n_p}$ be the matrix corresponding to z-scores of gene expression data: Here, $\mathbf{Z}(i,j)$ denotes the z-score of the expression of gene i in sample j. More precisely, if the raw expression of gene i in sample j equals x_{ij}, then $\mathbf{Z}(i,j) = \frac{x_{ij}-\mu_i}{\sigma_i}$; μ_i denotes the average expression of gene i and σ_i denotes its standard deviation. The entries of this matrix will be used to incorporate the expression information into the clustering analysis, as described in the next section.

Observe that some datasets are clearly correlated with each other while others may have very little correlation (e.g. CNV and gene expression are clearly correlated); nevertheless, different datasets provide different expert opinions that contain potential errors and noise sources and hence combining them one expects to get significantly improved inference results.

3.1.1 Clustering weights determined based on mutual exclusivity and coverage (ME-CO)

The idea behind our approach is to impose the mutual exclusivity constraint through the weights w_e^- and coverage constraint through the weights w_e^+. We remind the reader that high coverage postulates that important driver pathway should be mutated in many patients as possible, while mutual exclusivity postulates that drivers should be mutually exclusive within the same pathway.

For each gene (i.e. vertex) u, let $\mathcal{S}(u)$ denote the set of patients in which u is altered. Note that we use the matrix $\mathbf{M}$ to determine if a mutation in the gene exists, either due to sequence mutation or CNV. Then, for any $u, v \in V(G)$, the negative weights are chosen according to

$$w_{u,v}^-(\mathrm{e}) = a \times \frac{|\mathcal{S}(u) \cap \mathcal{S}(v)|}{\min(|\mathcal{S}(u)|, |\mathcal{S}(v)|)}, \qquad (2)$$

where a is a scaling parameter to be chosen by the user, and the label e in the weight refers to 'exclusivity'. The intuition behind the choice of the weight is as follows: the smaller the number of patients in

which both u and v are mutated, the smaller the weight and the more likely that u and v are mutually exclusive. The reason behind the use of the factor a is that we can strengthen or weaken the importance of mutual exclusivity through a: If a is large (e.g. empirically, a value of $a > 3$ is deemed large), mutual exclusivity is enforced strictly, while if a is small, (e.g. $a < 3$), the genes in each cluster will not be highly mutually exclusive. Also, note that

$$0 \leq \frac{|\mathcal{S}(u) \cap \mathcal{S}(v)|}{\min(|\mathcal{S}(u)|, |\mathcal{S}(v)|)} \leq 1.$$

To capture the coverage property through the positive weights, observe that if two genes increase the coverage significantly, their positive weight should be large so that they are encouraged to be placed in the same cluster. To determine the positive weights, we first form the set $\mathcal{D} = \{D(u,v)\}$, for all $u, v \in V(G)$, where $D(u,v) = |\mathcal{S}(u) \Delta \mathcal{S}(v)|$ and Δ denotes the symmetric difference of two sets. A large value for the symmetric difference $D(u, v)$ suggests that the vertices u and v should be placed in the same cluster, since they increase the coverage of the cluster.

Given the set $\mathcal{D}$, we define $T(J)$ to be the Jth percentile of the values in $\mathcal{D}$. In all our runs, we used the default value of $J = 95$, although this choice may be governed by the user as well. The positive weights are chosen according to:

$$w^{+}_{uv}(\mathrm{c}) = \begin{cases} 1, & \text{if } D(u,v) > T(J) \\ \dfrac{1}{T(J)} \times D(u,v) & \text{otherwise.} \end{cases} \quad (3)$$

Note that by this definition, $0 \leq w^{+}_{uv}(\mathrm{c}) \leq 1$ for any $uv \in E(G)$. Also, we used the index c in the weight label to indicate 'coverage.'

In order to ensure that the positive and negative weights meet the constraints imposed by our constant approximation algorithm, we also require that for all $u, v \in V(G)$, $w^{-}_{uv}(\mathrm{e}) + w^{+}_{uv}(\mathrm{c}) \geq 1$. This leads to the additional constraints:

$$\text{if } w^{+}_{uv}(\mathrm{c}) + w^{-}_{uv}(\mathrm{e}) < 1, \quad (4)$$

$$\text{rescale } w^{-}_{uv}(\mathrm{e}) \text{ to } \frac{w^{-}_{uv}(\mathrm{e})}{w^{+}_{uv}(\mathrm{c}) + w^{-}_{uv}(\mathrm{e})}, \text{ and let } w^{+}_{uv}(\mathrm{c}) = 1 - w^{-}_{uv}(\mathrm{e}).$$

3.1.2 Clustering weights determined based on mutual exclusivity, coverage and network information (NI-ME-CO)

The comprehensive results of pan-cancer studies reported in a number of recent papers (Garcia-Alonso *et al.*, 2014; Leiserson *et al.*, 2013, 2015b; Porta-Pardo *et al.*, 2015) have revealed the important connection between network topology and cancer driver distribution patterns. More precisely, the effect of deleterious mutations on the phenotype may be suppressed through a particular configuration of the corresponding protein complexes, and at the same time, the strength of the effect of a mutation may be emphasized through another configuration. As an example, most of the variants observed in healthy individuals seem to appear at the periphery of the interactome, and they do not seem to influence network connectivity. In contrast, cancer driver somatic mutations tend to occur in central, internal regions of the interactome and within highly co-integrated components. It appears that no previous attempts were made to more precisely quantify the network distances between driver variants, which prompted us to perform the following analysis. We first computed the pairwise (shortest) network distances between genes in a large pathway comprising 8726 genes from (Ciriello *et al.*, 2012) via an implementation of the standard Dijkstra algorithm (Skiena, 1990). In this test, we randomly selected 1000 pairs in order to reduce the computational burden of running Dijkstra's algorithm $O(8726^2)$ times. By using the most complete known driver list from the Cancer Gene Census (CGC) (Futreal *et al.*, 2004), we computed the same distances for driver genes, this time for all pairs of genes. The resulting distribution of shortest paths is depicted in Figure 1. One can clearly observe that the average shortest distance between drivers is significantly smaller than the average shortest distance between two randomly selected genes. A permutation test confirms this observation, and we calculated a P-value of less than 0.001.

These findings suggest that when determining potential driver mutations, one should make use of network distance and connectivity information. This may be accomplished within our approach by adjusting the positive weight of edges connecting two genes: If both endpoint genes were to be drivers, they should be sufficiently central to a given pathway, close to other known drivers or to each other.

For the purpose of studying this problem, we consider an undirected graph corresponding to the gene network, denoted by G'; in this graph, which is assumed to be known a priori and which in this work was retrieved from the KEGG Database, each vertex corresponds to a gene. The graph is not complete, but rather relatively sparse, and each edge represents an interaction between genes. As before, we let n_p and $n_g = |V(G)| = |V(G')|$ denote the total number of patients and the total number of genes in our dataset, respectively. For each vertex $u \in V(G')$, we let $\mathcal{N}(u)$ denote the set of neighbors of u and let $\mathcal{N}'(u) = \mathcal{N}(u) \cup \{u\}$. The first step in assigning the positive weights is to determine the set $\mathcal{F} = \{f(u,v)\}$, where for any pair of vertices $u, v \in V(G')$,

$$f(u,v) = \frac{|\mathcal{N}'(u) \cap \mathcal{N}'(v)|}{|\mathcal{N}'(u) \cup \mathcal{N}'(v)|}. \quad (5)$$

Note that $0 \leq f(u,v) \leq 1$ for all u, v. In a nutshell, $f(u, v)$ captures the shared neighborhood of two genes, normalized by the size of their joint network neighbors. In the statistics literature, the function is known as the *Jaccard similarity coefficient* between two sets. A large value of the Jaccard coefficient $f(u, v)$ suggests that the genes u and v are well connected and likely to be involved in the same pathway (Ciriello *et al.*, 2012), and that the corresponding genes should be clustered together.

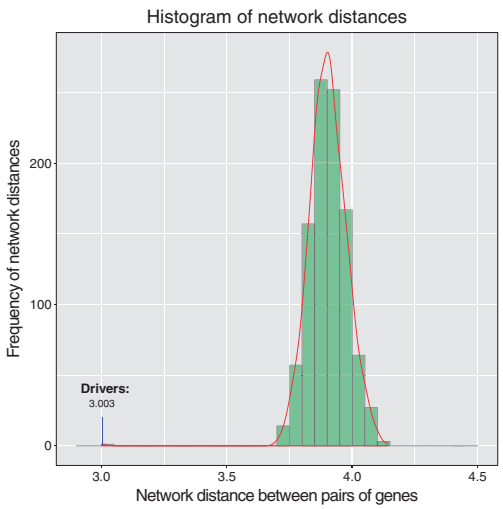

Fig. 1. Histogram of shortest distances between randomly selected genes and driver genes in the network

Given the set $\mathcal{F}$, we define $T'(J')$ to be the J'th percentile of the values in $\mathcal{F}$. For any $u, v \in V(G)$, the positive weights are then chosen according to the following formula:

$$w^+_{uv}(\mathrm{c},\mathrm{n}) = w_1\, w^+_{uv}(\mathrm{c}) + w_2\, w^+_{uv}(\mathrm{n}), \tag{6}$$

where $w_1, w_2 \geq 0$, $w_1 + w_2 = 1$, and where the indices c and n indicate 'coverage' and 'network'. The coverage weight, as before, equals

$$w^+_{uv}(\mathrm{c}) = \begin{cases} 1, & \text{if } \ D(u,v) > T(J) \\ \dfrac{1}{T(J)} \times D(u,v) & \text{otherwise,} \end{cases} \tag{7}$$

and the network weight equals

$$w^+_{uv}(\mathrm{n}) = \begin{cases} 1, & \text{if } \ f(u,v) > T'(J') \\ \dfrac{1}{T'(J')} \times f(u,v) & \text{otherwise.} \end{cases} \tag{8}$$

Again, in order to ensure that for all $u, v \in V(G)$, $w^-_{uv}(\mathrm{e}) + w^+_{uv}(\mathrm{c},\mathrm{n}) \geq 1$, we add the additional constraints

$$\text{if } \ w^+_{uv}(\mathrm{c},\mathrm{n}) + w^-_{uv}(\mathrm{e}) \ < \ 1, \tag{9}$$

$$\text{set } w^-_{uv}(\mathrm{e}) = \frac{w^-_{uv}(\mathrm{e})}{w^+_{uv}(\mathrm{c},\mathrm{n}) + w^-_{uv}(\mathrm{e})}, \text{ and } w^+_{uv}(\mathrm{c},\mathrm{n}) = 1 - w^-_{uv}(\mathrm{e}).$$

The weights w_1, w_2 may be chosen in such a way as to emphasize the importance of either coverage or network information. We suggest using $w_1 = w_2 = 1/2$ in a coverage/network only test, although our analysis reveals that emphasizing one criterion over the other offers improved algorithm performance on some datasets.

3.1.3 Clustering weights determined based on mutual exclusivity, coverage and gene expression data (EX-ME-CO)

Similar to network information, expression data may be incorporated through the positive weights, using the assumption that co-expressed genes may be involved in the same function or cancer pathway. Hence, highly (positively or negatively) co-expressed genes should be encouraged to cluster together.

To explain how to incorporate gene expression data into the clustering procedure, assume that $\mathbf{z}(u)$ and $\mathbf{z}(v)$ denote the vectors of time-evolving expression values corresponding to genes u and v, respectively. The first step in assigning the positive weights is to determine the set $\mathcal{G} = \{g(u,v)\}$, where for every pair of genes u, v,

$$g(u,v) = \frac{|\langle \mathbf{z}(u), \mathbf{z}(v)\rangle|}{||\mathbf{z}(u)||\,||\mathbf{z}(v)||}. \tag{10}$$

Here, $\langle \mathbf{a}, \mathbf{b}\rangle$ denotes the classical inner product of the vectors $\mathbf{a}$ and $\mathbf{b}$, while $||\mathbf{a}||$ stands for the ℓ_2 norm. A large value for $g(u, v)$ indicates that the expression vectors of u and v are highly correlated and hence should be clustered together (we used absolute values to capture both positive and negative correlations). Also, note that $0 \leq g(u,v) \leq 1$ for all u and v.

Given the set $\mathcal{G}$, we let $T''(J'')$ denote the J''th percentile of the values in $\mathcal{G}$. For any $u, v \in V(G)$, the positive weights are chosen according to the following formula:

$$w^+_{uv}(\mathrm{c},\mathrm{x}) = w_1\, w^+_{uv}(\mathrm{c}) + w_2\, w^+_{uv}(\mathrm{x}), \tag{11}$$

where $w_1, w_2 \geq 0$, $w_1 + w_2 = 1$, and

$$w^+_{uv}(\mathrm{c}) = \begin{cases} 1, & \text{if } \ D(u,v) > T(J) \\ \dfrac{1}{T(J)} \times D(u,v) & \text{otherwise,} \end{cases} \tag{12}$$

and

$$w^+_{uv}(\mathrm{x}) = \begin{cases} 1, & \text{if } \ g(u,v) > T''(J'') \\ \dfrac{1}{T''(J'')} \times g(u,v) & \text{otherwise.} \end{cases} \tag{13}$$

Hence, all the algorithmic conditions required are satisfied for the weights, except possibly the third one. In order to make sure that for all $u, v \in V(G)$, $w^-_{uv}(\mathrm{e}) + w^+_{uv}(\mathrm{c},\mathrm{x}) \geq 1$, we include an additional condition that

$$\text{if } w^+_{uv}(\mathrm{c},\mathrm{x}) + w^-_{uv}(\mathrm{e}) \ < \ 1, \tag{14}$$

$$\text{set } w^-_{uv}(\mathrm{e}) = \frac{w^-_{uv}(\mathrm{e})}{w^+_{uv}(\mathrm{c},\mathrm{x}) + w^-_{uv}(\mathrm{e})}, \text{ and } w^+_{uv}(\mathrm{c},\mathrm{x}) = 1 - w^-_{uv}(\mathrm{e}).$$

Note that other combinations of datasets may be used, with appropriate changes in the weights. For example, incorporating coverage, network information as well as expression information into a positive weight may be accomplished by setting

$$w^+_{uv}(\mathrm{c},\mathrm{n},\mathrm{x}) = w_1\, w^+_{uv}(\mathrm{c}) + w_2\, w^+_{uv}(\mathrm{n}) + w_3\, w^+_{uv}(\mathrm{x}), \tag{15}$$

where $w_1, w_2, w_3 \geq 0$, $w_1 + w_2 + w_3 = 1$.

Figure 2 illustrates how the various data sources were integrated into positive and negative clustering weights.

3.2 Clustering algorithms

The classical formulation of correlation clustering does not include cluster size restrictions. On the other hand, all known driver identification methods operate with de facto cluster size bounds, as the cluster sizes govern the computational complexity of the method. For example, comprehensive testing of CoMEt reveals that the algorithm fails to operate beyond cluster sizes of 10–12. In order to perform a fair comparison, we introduce a cluster size constraint in our algorithm, by assuming that all clusters are of size K. Clearly, setting K equal to the number of vertices (genes) removes the cluster size constraint, hence our algorithm has a large flexibility in cluster size selection. An additional reason for choosing a restricted cluster size is that we expect driver genes of specific cancer types to be grouped together within clusters, and as already remarked, a number of recent results suggest that only a few drivers are actually present in any cancer type. Making the clusters excessively large would potentially lead to inclusions of multiple cancer type drivers in the same cluster, thereby obscuring the fine partition of the drivers. Nevertheless, the user of the method may choose K according to her/his own requirements. Yet another reason for introducing cluster sizes is to avoid the shortcomings of many known clustering

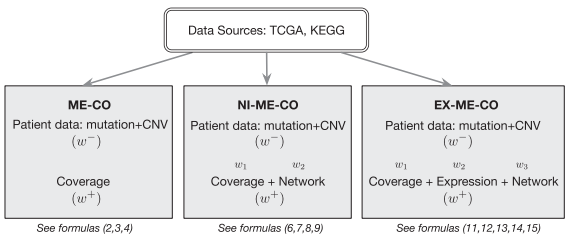

Fig. 2. Heterogenous data sources converted into different clustering weights

algorithms which tend to produce non-informative 'giant clusters' and singleton clusters.

The bounded cluster size correlation clustering problem for driver gene inference may be formulated as follows. As already described, let K be a 'hard' bound on the size of the driver clusters, and let the positive w^+ and negative weights w^- be chosen according to a desired combination of datasets, as explained in the previous section. The optimum clustering may be found by solving the integer linear program (ILP) below.

$$\underset{x}{\text{minimize}} \sum_{e \in E(G)} (w_e^+ x_e + w_e^-(1 - x_e)) \tag{16}$$

$$\text{subject to } x_{uv} \leq x_{uz} + x_{zv} \,(\text{for all distinct } u, v, z \in V(G)) \tag{17}$$

$$\sum_{v \neq u} (1 - x_{uv}) \leq K \;\; (\text{for all } u \in V(G)) \tag{18}$$

$$x_e \in \{0, 1\} \;\; (\text{for all } e \in E(G)). \tag{19}$$

In this formulation, and for a fixed edge $e = uv$, $x_{uv} = 1$ implies that u and v should belong to different clusters and $x_{uv} = 0$ implies that the two vertices should belong to the same cluster. Note that the triangle inequality (17) ensures that if u and z are in the same cluster and z and v are in the same cluster, then u and v are also in the same cluster. Any clustering of the vertices can be described using the variables x_e. For a fixed clustering, the objective function is the cost associated with that clustering.

Solving the ILP is NP-hard. We hence relax the problem by changing the integer constraint $x_e \in \{0, 1\}$ to an interval constraint $x_e \in [0, 1]$. This relaxation leads to a classical linear program (LP), the solution of which may be fractional. To obtain a valid clustering, the fractional solutions have to be subsequently *rounded* to produce integer solutions. Unfortunately, known rounding algorithms we previously developed in Puleo and Milenkovic (2015) tend to produce very small clusters, often as small as single-vertex clusters that are not meaningful. For our study, we hence slightly modify the algorithm by *moving the cluster size constraint (18) from the LP to the rounding procedure*. The new rounding procedure is described in Algorithm 1. Hence, the clustering algorithm involves solving 16 without the constraint $\sum_{v \neq u}(1 - x_{uv}) \leq K$ and then applying the rounding procedure of Algorithm 1.

Algorithm 1 is closely based on the rounding algorithm described in Charikar *et al.* (2003). The idea behind the rounding algorithm is to pivot on one vertex, examine its closest neighbors, where closeness is governed by the value of the output variables x_e of the LP, and partition large neighborhoods if needed to get clusters of size at most $K+1$. In the Appendix of the Supplementary Materials, we prove that the LP and Rounding Algorithm 1 provides a 9-approximation for the ILP problem, given that the parameter α is set to 2/7 and given that the weights obey the following constraints:

- $w_e^+ \leq 1$ for every edge e, and
- $w_e^+ + w_e^- \geq 1$ for every edge e.

The above inequalities were addressed as described in the previous section, and we remind the reader that they were imposed on the weights through proper normalization.

Note that we only used high frequency mutations for our clustering problem, and hence did not encounter any computational issues with the LP solvers. On the other hand, if one were to use all 25 000 genes in the analysis, the LP solver implemented in Gurobi (https://www.gurobi.com/) would inevitably break down due to the large number of constraints, which is quadratic in the number of genes. In this case, a much simpler scalable solution is to use *approximate LP solvers*, akin to those described in Sridhar *et al.* (2013). The approximate solver is guaranteed to produce a solution that does not exceed the LP solution by more than a factor $1 + \epsilon$, for some small value of ϵ, by using gradient descent methods that are highly scalable.

Algorithm 1
Input: $\{x_e\}_{e \in E(G)}$, α and K
Let $S = V(G)$.
while $S \neq \varnothing$ **do**
  Let the 'pivot vertex' u be an arbitrary element of S.
  Let $T = \{w \in S - \{u\} : x_{uw} \leq \alpha\}$.
  if $\sum_{w \in T} x_{uw} \geq \alpha |T|/2$ **then**
    Output the singleton cluster $\{u\}$.
    Let $S = S - \{u\}$.
  else if $|T| \leq K$ **then**
    Output the cluster $\{u\} \cup T$.
    Let $S = S - (\{u\} \cup T)$.
  else
    Partition T as $T = T'_0 \cup T_1 \cup \cdots \cup T_p$, where $|T'_0| = K$ and each $|T_i| = K + 1$ for $0 < i < p$ and $|T_p| \leq K + 1$.
    Let $T_0 = \{u\} \cup T'_0$.
    Output the clusters $T_0, T_1, T_2, \ldots, T_p$.
    Let $S = S - (\{u\} \cup T)$.
  end if
end while

3.3 Evaluation methods

We evaluated the performance of both C^3 and CoMEt in terms of their ability to detect *mutually exclusive, high-coverage and biologically relevant gene clusters*. At this point, it is important to observe that the inference and evaluation strategies may appear to involve circular arguments: Mutual exclusivity, coverage and network distance, used to predict the clusters, are also used to evaluate the performance of the clustering method. But this is clearly not the case, as mutual exclusivity, coverage and network distance are *optimization constraints*, and one always needs to test the quality of a (approximate) solution to an optimization problem based on how well the constraints are accounted for. Other driver discovery tools, such as CoMEt, use the same constraint modeling and evaluation criteria. Furthermore, we added one more evaluation criteria, related to biological significance and pathway enrichment analysis, which is independent on the optimization criteria. As will be shown in the subsequent section, this evaluation criteria confirms the quality of the C^3 analysis for cancer driver gene inference and its improvements over CoMEt.

We ran both the C^3 and CoMEt methods using mutation and CNV data collected from TCGA, pertaining to breast cancer (BRCA) (Network *et al.*, 2012) and glioblastoma (GBM) (Brennan *et al.*, 2013). In addition to GBM and BRCA, we also considered kidney cancer (KIRC) and ovarian cancer (OV), but the available patient data appeared limited at this stage to allow for statistically significant and comprehensive results. We accessed the TCGA provisional data using the cBioPortal platform (Gao *et al.*, 2013) on August 14, 2015. We ran both methods using the same alteration dataset. We evaluated both point mutations and indels, and for CNVs, we used the GISTIC thresholds (Mermel *et al.*, 2011) of –1 and 3 as our cut-offs (as already pointed out in the previous section).

To focus on mutations with high frequency, we only selected genes in the top 95 percentile of alteration frequencies, thereby obtaining 130 genes spanning 959 patient samples in BRCA and 170 genes spanning 291 patient samples in GBM.

To test the effects of cluster sizes and the quality of our results, we ran both C^3 and CoMEt to find clusters of sizes upper bounded by 5, 6, 7, 10 and 15. As already pointed out, larger cluster sizes are easily accommodated for by C^3, but since CoMEt failed to produce solutions for clusters of sizes roughly greater than ten, we restricted our attention to the aforementioned range of values. Due to the fact that correlation clustering and CoMEt will cluster all genes in a dataset, and hence produce a partition of the gene set, a large number of clusters will contain neutral mutations only and will hence have no biological significance. This is why we only compared the top ten most mutually exclusive gene sets generated by C^3 with those of CoMEt.

We ran CoMEt with 1 000 iterations each and 3 initialization points to ensure both timely and consistent runs. For C^3, we ran the C^3 clustering method for all combinations of weights $w_1, w_2, w_3 \in \{0, 0.25, 0.5, 0.75, 1\}$ that satisfy $w_1 + w_2 + w_3 = 1$, but selected to report only results for the weight parameters $w_1 = 0.167$ (coverage), $w_2 = 0.333$ (network information) and $w_3 = 0.333$ (expression data). Our choice is governed by the fact that coverage seems to be a biologically much less important criteria then network information or expression. Hence, high weights for expression and network information increase the ability of the C^3 algorithm to detect biologically significant clusters. Furthermore, the *patient coverage* criteria appears to be less relevant than *pathway coverage* and some other coverage properties that have not been explicitly investigated in the literature. Nevertheless, we observe that the choice of the weights may be completely governed by the user, and that the increase in one weight may produce better results in one performance category while reducing the performance in another category.

We used four statistical methods to assess the performance of the algorithms which reflect both the statistical and biological significance of the clusters found.

Mutual Exclusivity. To evaluate the degree of mutual exclusivity in a cluster, we performed a Fisher's exact test (Fisher, 1922) for each pair of genes in the cluster. The Fisher's exact test uses a hypergeometric distribution to calculate the probability of observing a 2×2 contingency table of a total of n samples, with a samples that have an alteration in two genes (say, g_i and g_j), b samples with an alteration in gene g_i only, and c samples with an alteration in gene g_j only. If d is the number of samples with no alteration in either gene, then the probability of co-mutation is evaluated according to

$$P(g_i, g_j) = \frac{\binom{a+b}{a}\binom{c+d}{c}}{\binom{n}{a+c}}. \tag{20}$$

We also evaluated the overall exclusivity of a cluster as the median value of each pairwise exclusivity test, for each pair of genes g_i, g_j in the network. The pairwise Fisher's method has also been used by the Mutex suite to establish mutual exclusivity (Babur *et al.*, 2015). However, because in our context the Fisher's exact test is used as an evaluation rather than as a discovery tool, we used the median pairwise P-value rather than the maximum P-value to get a better sense of the overall exclusivity of genes within a cluster. It is also important to note that while CoMEt has a built-in method that generalizes the exclusivity test to a 2^k contingency table for a cluster size $k \geq 2$, the exponential size of their test set makes evaluation for large cluster sizes computationally impractical. An alternative test for overall mutual exclusivity is a permutation test, as implemented by MEMo, which compares the exclusivity of a gene set by sampling random gene sets and patients with multiple alterations.

Coverage. To compare and evaluate the overall coverage of a cluster found by C^3 or CoMEt, we calculated and reported the proportion of patients with at least one alteration in a gene belonging to the given cluster.

Network Clustering. We performed an additional pathway analysis for the potential cancer gene drivers. As pointed out in the previous section, driver genes tend to be, on average, closer to each other in a pathway compared to randomly selected genes. Our tests involved assessing the shortest network distance of genes within the discovered clusters. We remind the readers that the distances were evaluated using Dijkstra's Algorithm on 8726 genes from Ciriello *et al.* (2012).

Biological Significance. In addition to testing the quality of the algorithm in terms of optimizing mutual exclusivity and coverage, we also investigated the biological significance of the C^3 and CoMEt methods from the perspective of gene discovery and pathway analysis. Although there is no overarching gold standard to determine biological significance, a commonly accepted metric employed by MEMo, Dendrix, Mutex, CoMEt and other similar tools is to count the number of known driver genes found within *the best clusters* according to the given criteria. These clusters usually contain known driver genes. To determine the driver gene-based biological significance, we calculated the proportion of drivers found in the ten most mutually exclusive C^3 and CoMEt clusters using a comprehensive, curated list of known drivers from the CGC.

It is important to point out that while the four test benchmarks we introduced are a reliable way to test the optimization quality and performance of CoMEt and C^3, no perfect benchmark exists for detecting mutually exclusive and biologically significant genes clusters. The hope is that multiple evaluation methods taken together may provide a better understanding of which methods outperform others in a given parameter and criteria setting.

4 Results

In what follows, we demonstrate that C^3 outperforms CoMEt in almost all of the aforementioned benchmarking criteria, or more precisely, for three out of the four chosen criteria. This is achieved without any special parameter tuning or optimization. As a rule of thumb, C^3 can be made to outperform CoMEt *in any chosen single, pair of triple of criteria* by adjusting the weights. This observation may be explained by the fact that the weights trade off the strengths of different modeling assumptions. We supplement our statistical analysis with a discussion of the biological relevance of our findings, and explore the role of the new potential drivers found by C^3 within their driver gene communities. In particular, we discuss the significance of large mutually exclusive clusters that cannot be recovered by other methods. Recall that we restrict our attention to the ten best performing clusters according to mutual exclusivity, as this approach was used in the original evaluation process of the CoMEt algorithm.

4.1 Performance evaluation

The results of our extensive comparison between C^3 and CoMEt, regarding mutual exclusivity, coverage, driver identification and pathway-level evaluation, are shown in Figure 3. Both algorithms were tested on the same server with a 256 GB RAM memory. Both methods ran uninterruptedly when the cluster sizes were constrained

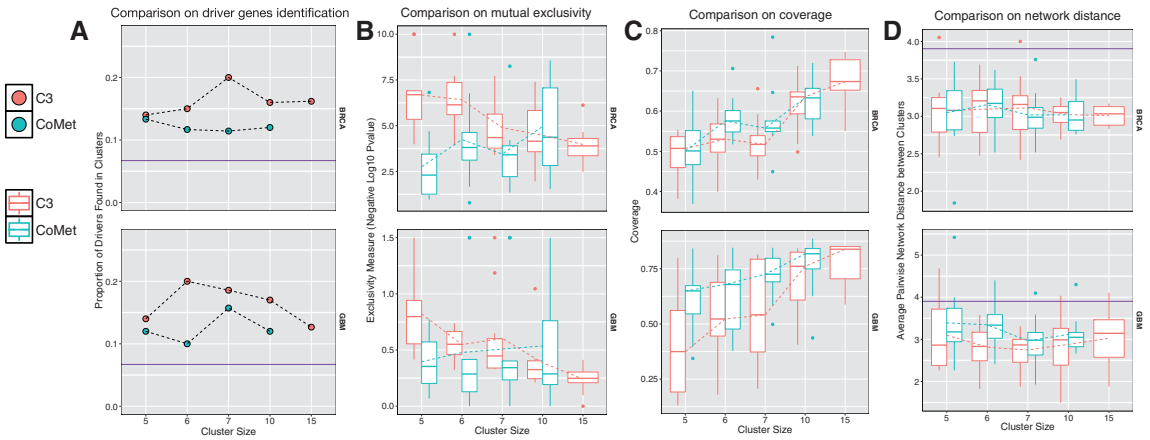

Fig. 3. A comparative analysis of C^3 (Red) and CoMEt (Blue) based on four evaluation criteria. We used five cluster sizes (5, 6, 7, 10 and 15) that index the *x*-axis in each benchmark test. (**A**) The results based on the driver gene evaluation criteria. The *y*-axis represents the proportion of drivers found by each method, contained within the best ten clusters found. The purple line represents the expected value of drivers detected if clusters are randomly selected. (**B**) The pairwise mutual exclusivity of each run. The *y*-axis represents the negative log transform of the mutual exclusive *P*-value such that larger values are more mutually exclusive than smaller ones. The boxplots illustrate the distribution of exclusivity results concerning each of the top ten individual clusters for C^3 and CoMEt. (**C**) The distribution of coverage, measured by proportion of samples with at least one alteration in a given cluster (the *y*-axis). The boxplot illustrates the distribution of coverage results for individual top ten cluster results. (**D**) The network connectivity results of C^3 and CoMEt. The *y*-axis measures the average pairwise network distance between all genes in a cluster, and the distribution of each cluster is shown in the boxplot. The purple line represents the average pairwise distance of random clusters

to $k = 5, 6, 7$ and 10. CoMEt reported segfault memory errors for $k = 15$, and for this case, only C^3 was benchmarked.

To assess the biological significance of the two methods in terms of their ability to cluster high-impact drivers from the CGC repository together, we compared the results of C^3 and CoMEt both to each other and to a 'baseline' value equal to the average proportion of drivers in the ten most mutually-exclusive clusters found, in this case 0.067, using uniform random sampling of genes (see Fig. 3A). In BRCA, we found that C^3 detected a median driver proportion of 0.160 and CoMEt detected a median driver proportion of 0.117 in the top ten clusters. C^3 outperformed CoMEt for each cluster size. We also used a Mann–Whitney Rank Sum test (Rosner and Grove, 1999) to compare the overall performance of the algorithms with respect to mutual exclusivity, for all cluster sizes. We chose a rank-sum test because it is unclear that the drivers are following a normal distribution due to the small amount of data points available. The results show that C^3 outperforms CoMEt (*P*-value of 0.0079) in terms of amount of drivers in clusters. C^3 also outperforms CoMEt on GBM, with a median proportion of drivers per cluster equal to 0.170, compared to a 0.12 proportion of drivers per cluster found by CoMEt. This finding holds for every cluster size, with a rank-sum test *P*-value of 0.0361. Both methods succeed in finding biologically significant drivers within clusters exhibiting high mutual exclusivity, and both methods significantly outperform the expected number of drivers per cluster in the random setting (*P*-value 1.594×10^{-5} and *P*-value 1.312×10^{-3} for C^3 and CoMEt, respectively).

We next tested the clusters found by each method based on their mutual exclusivity (see Fig. 3B). To do so, we used the previously described pairwise Fisher's exact test to obtain a *P*-value for each of the top ten clusters of the two methods. For better visualization, we performed a negative log transform on the *P*-values, and plotted the transformed *P*-value distribution. Hence, in this system, larger values indicate more mutually exclusivity. We again used a Mann–Whitney rank-sum test to evaluate the performance of C^3 and CoMEt. For BRCA, one can see that while both methods have significant median exclusivity values ($P = 7.541 \times 10^{-6}$ for C^3 and $P = 3.337 \times 10^{-4}$ for CoMEt), C^3 has an overall more significant p-values for each cluster size. The median *P*-value of C^3 for each cluster size is lower than its CoMEt counterpart except for the case $k = 10$. However, C^3 does have superior performance overall with a rank-sum *P*-value of $P = 4.0202 \times 10^{-4}$. For GBM, the median exclusivity results are not as strong as for the BRCA set, for both the C^3 and CoMEt method. C^3 has a median *P*-value of 0.3795 as opposed to CoMEt's 0.5022. The general drop in significance may be attributed to a lower confidence of the Fisher's test due to a small number of samples available; recall that the GBM set involved 291 samples, compared to 959 BRCA samples. This indicates that one should look at individual significant clusters to evaluate mutual exclusivity. Even for the reduced median *P*-value regime, C^3 outperforms CoMEt in significance, having lower median *P*-values for each cluster size. Overall, the C^3 *P*-values are consistently and significantly lower than those produced by CoMEt for mutual exclusivity (the rank-sum test *P*-value equals 0.04401).

The results of the coverage tests are depicted in Figure 3C. In the coverage benchmark, CoMEt outperforms C^3 for GBM, but neither method outperforms the other for BRCA. In BRCA, both methods show comparable performance, with a median result for the fraction of samples covered equal to 0.5505 for C^3, and 0.5662 for CoMEt. This rather poor performance of both methods is observed for all values of *k*, with no *P*-value based on Student's T-test (Zimmerman, 1987) being less than 0.05. The largest difference in coverage recorded for the two methods is present for $k = 6$. In conclusion, there appears to be no statistical difference between C^3 and CoMEt in terms of BRCA coverage percentage (*P*-value of 0.5127). In GBM, the median *P*-value for coverage difference is more pronounced. The median coverage of C^3 is 0.632 and the median coverage of CoMEt is 0.696. CoMEt finds significantly higher-coverage clusters according to Student's *t*-test, with *P*-value 0.0345, and the most pronounced coverage percentage differences exist for small values of *k* (0.3745 versus 0.6495 for $k = 5$ C^3 and CoMEt, respectively). It is also important to note the wide distribution of coverage score values produced by C^3 for small *k*: the IQR (Interquartile range) value is roughly 0.35 for $k = 5$. The most likely reason behind this result is that our test weights were chosen to boost the relevance of mutual-

exclusivity and biological significance rather than coverage. Mutual exclusivity accounts for 100% of the negative weights of edges, while coverage accounts for only 16.7% of the positive weights. We justify this weight choice by the fact that it leads to multiple significant cluster discovery and with our assumption that coverage is a less significant driver property compared to mutual exclusivity. We also point out that it appears that a biologically more relevant coverage constraint is pathway coverage, rather than patient sample coverage.

As already mentioned in the previous sections, one advantage of C^3 is that the user can adjust the weights according to her/his own belief about the significance of patient coverage. For example, by changing the averaging weights in our GBM run to $w_1 = 0.60$ (coverage), $w_2 = 0.20$ (network) and $w_3 = 0.20$ (expression), we obtain a coverage percentage of 0.7903 for $k = 5$. However, this excellent coverage comes at a cost of a less significant mutual exclusivity score (fractional value 0.4288) and a lower proportion of detected drivers (fractional value 0.1267). As may be seen from the above example, C^3 can be adapted to the user's specification to best reflect the scope and preferences of the analysis.

Another setting in which we analyzed C^3 and CoMEt involves pairwise distances of drivers in the network (see Fig. 3D). Here, we calculated the average pairwise distance between all pairs of genes clustered together. We then used Student's *t*-test to determine the statistical significance of this value. We also compared the values for both algorithms based on 1000 randomly selected genes by using a permutation test. For BRCA, we found no significant performance difference between the two methods in terms of the average pairwise distance: 3.110 for C^3 and 3.070 for CoMEt, with a *P*-value of 0.9330. In GBM, C^3 showed a smaller average pairwise distance of 2.908 compared to CoMEt's 3.097. This difference is statistically significant, with a *P*-value of 0.0379. The small average network distance results of C^3 for GBM, coupled with the low coverage, leads to the conclusion that C^3 favors niche, exclusive clusters in biologically relevant cancer pathways. Hence, the method may be useful for discovering specific molecular cancer subtypes. Both methods had an average pairwise distance well below the permutation benchmark of 3.903: the *P*-values of both C^3 and CoMEt were less than 2×10^{-16} for both cancers.

In conclusion, from our detailed evaluation we conclude that although C^3 does not simultaneously outperform CoMEt with respect to all four evaluation criteria, but only three of them (which already represents a significant advantage), the C^3 performance indicates a strong overall propensity to select biologically more relevant and more mutually exclusive clusters, with a higher degree of flexibility compared to CoMEt.

4.2 Discovering potential driver pathways

We examine next the potential of the C^3 algorithm to detect clusters whose genes may be new candidate cancer drivers. We focus our search on clusters that contain biologically significant driver genes and known biological network interactions, and exhibit high mutual exclusivity and coverage. At the same time, we only consider the large cluster size regime, as results in this domain have not been previously reported in the literature and as they offer many new interesting insights. Two examples of our analysis are shown in Figures 4 and 5.

In BRCA, one candidate cluster with several potential novel driver genes is the cluster containing *PTEN, HUWE1, CNTNAP2, GRID2, CACNA1B, CYSLTR2, MYH1* depicted in Figure 4. The genes in the candidate cluster are mutually exclusive ($P - \text{value} = 0.0084$). The genome landscape of this cluster is dominated primarily by mutations in *PTEN* and *HUWE1*, and secondarily by homozygous deletions in *PTEN* and *CYSLTR2*. The most frequently altered gene in this set is a common driver gene *PTEN*, a tumor suppressor gene that negatively regulates the AKT/PKB apoptosis pathway (Stambolic *et al.*, 1998). The remaining six

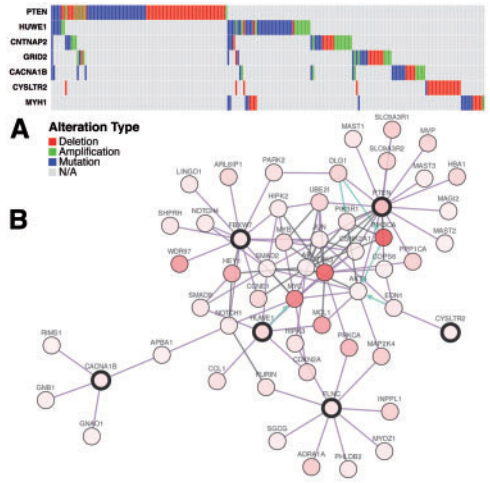

Fig. 4. A cluster of potential driver genes inferred from BRCA. (**A**) The alteration landscape of the cluster, with blue representing mutation events, red representing copy number deletions and green representing copy number amplifications. (**B**) A *known* subnetwork which contains 6 genes (out of 7) in (A). The more intense the red, the higher the alteration frequency of the gene. Nodes highlighted in black represent driver candidates identified by C^3 within a small subnetwork. Edges are depicted in black if there exists a direct interaction between two genes. Green edges represent an interaction that undergoes a protein state change. Purple edges are other interactions

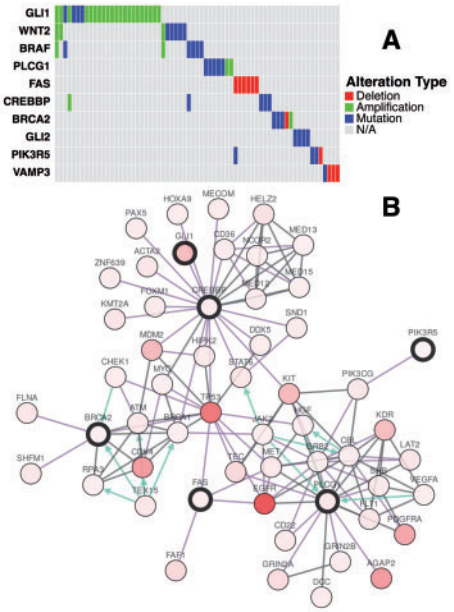

Fig. 5. A cluster of potential driver genes inferred from GBM. (**A**) The alteration landscape of the cluster, with blue representing mutation events, red representing copy number deletions and green representing copy number amplifications. (**B**) A *known* subnetwork which contains 6 genes (out of 10) in (A). The more intense the red, the higher the alteration frequency of the gene. Nodes highlighted in black represent driver candidates identified by C^3 within a small subnetwork. Edges are depicted in black if there exists a direct interaction between two genes. Green edges represent an interaction that undergoes a protein state change. Purple edges are other interactions

genes in the cluster are potential driver candidates. *HUWE1* is a part of the Mule multidomain complex of the HECT domain family of E3 ubiquitin ligases responsible for apoptosis suppression, DNA damage repair, and transcriptional regulation (Inoue *et al.*, 2013). *CNTNAP2* is a neurexin protein with functions in cell-to-cell adhesion and an epidermal growth factor and was found to be hypomethylated in breast cancer cell lines (Shann *et al.*, 2008). Hypomethylation and the association with epidermal growth factors, coupled with a large number of amplifications in the alteration landscape of *CNTNAP2* suggest potential oncogenic functions of the gene. *GRID2* is an ionotropic glutamate receptor that is frequently deleted in lymphomas (Roy *et al.*, 2011). *CACNA1B* codes for a N-type calcium channel which is responsible for calcium influx. Defects in the calcium influx channel can lead to alteration in the apoptosis, proliferation, migration and invasion pathways of breast cancer (Azimi *et al.*, 2014). *CYSLTR2* is a proinflammatory cysteinyl leukotriene receptor that plays a role in cancer cell differentiation and is associated with breast cancer survival rates (Magnusson *et al.*, 2011). MYH1 is a myosin heavy chain protein that plays a role in cell signaling and pro-apotosis pathways.

Perhaps more important than the propensity of each individual gene to be a driver is the collective interaction pattern of the seven genes in the cluster in a cancer pathway. From Figure 4, it is clear that the each gene in the cluster interacts with each other in a tightly-connected community with no gene more than three nodes away when plotted in the network, using the cBioPortal visualization tool. The seven genes in the cluster *PTEN, HUWE1, CNTNAP2, GRID2, CACNA1B, CYSLTR2, MYH1* are strong candidates to define a novel driver pathway. This conclusion is reinforced by the presence of high impact common drivers (*TP53, MYC, AKT and PIK3R1*) which define several important cancer pathways such as apoptosis, DNA repair and cell cycle arrest (Stemke-Hale *et al.*, 2008; Vazquez *et al.*, 2008).

We also examined a cluster containing potential cancer drivers relevant for GBM. In GBM, we found a cluster of size 10 with four known drivers and many potential drivers. The cluster includes *GLI1, WNT2, BRAF, PLCG1, FAS, CREBBP, BRCA2, GLI2, PIK3R5, VAMP3* (see Fig. 5). This large cluster has a *P*-value of 0.0901 in terms of mutual exclusivity, which is actually low as compared to other GBM clusters. The cluster also contains several important driver genes such as *WNT2, BRAF, BRCA2* and *CREBBP* which encompass pathways such as sonic hedgehog signaling, cell fate determination, cell growth and apoptosis, checkpoint activation and DNA repair.

Additionally, six out of the ten members are within the same compact network community (*GLI1, PLCG1, FAS, CREBBP, BRCA2, PIK3R5*). Of these six genes, GLI1 and GLI2 are hedgehog signaling genes that are common and first isolated in glioblastoma. These genes are responsible for cell differentiation and stem cell self-renewal (Clement *et al.*, 2007). *PLCG1* is involved in intracellular transduction of receptor-mediated tyrosine kinase activators, and it has been classified as a biomarker in GBM (Serão *et al.*, 2011). *FAS* is a cell surface receptor that mediates apoptosis. *FAS* is known as a histological hallmark of GBM, affecting both apoptosis and necrosis factors (Gratas *et al.*, 1997). Finally, *PIK3R5* is a subunit of phosphatidylinositol 3-kinases who together have important effects on cell growth, proliferation, differentiation, motility, survival and intracellular trafficking.

Additional cluster analysis examples are relegated to the Supplementary materials, focussing on clusters that contain lesser known and documented driver genes.

5 Discussion and conclusion

We described a novel method, termed C^3, which has the potential to precisely and efficiently identify clusters of gene modules with mutually exclusive mutation patterns. The C^3 algorithm uses large-scale cancer genomics datasets which are pre-processed to yield parameters governing novel constrained correlation clustering techniques. The optimization criteria used in clustering include patterns of mutual exclusivity of mutations, patient sample coverage and network driver concentration.

There are several major advancements of our method when compared to previously known approaches. Unlike methods that use randomized approaches without the guarantee that multiple runs of the methods on the same data will produce compatible results (such as CoMEt), C^3 is 'consistent' in so far that by running the same LP solver, the same results will be generated. Also, C^3 has computational complexity that does not depend on the chosen cluster sizes, and is hence much more appropriate for large cluster problems than other methods. Furthermore, it partitions the gene set and hence creates clusters covering all genes used in the analysis, although it may also be adapted to accommodate overlapping clusters. This is in contrast with the results produced by other methods that tend to identify only a small number of modules with limited number of genes.

None of the previous methods were able to identify clusters utilizing different sources of information via a weighting mechanism. This is important because it gives us flexibility to focus more on certain aspects based on the analysis. For example, we can focus more on mutual exclusivity instead of coverage to identify clusters specific to a group of samples which may facilitate the discovery of subtype-specific modules.

By addressing the above challenges, we believe our new method C^3 represents a unique tool to efficiently and reliably identify mutation patterns and driver pathways in large-scale cancer genomics studies.

Acknowledgements

We thank Mark Leiserson (Raphael lab, Brown University) for assistance with CoMEt.

Funding

The study was supported in part by National Science Foundation Grants CCF 0939370, CCF 111798, IOS 1339388 and National Institutes of Health grant U01 CA198943-02 to OM, National Institutes of Health grants HG007352, CA182360 and DK107965 to JM, and National Science Foundation grants 1054309 and 1262575 to JM.

Conflict of Interest: none declared.

References

Azimi, I. *et al.* (2014) Calcium influx pathways in breast cancer: opportunities for pharmacological intervention. *Br. J. Pharmacol.*, **171**, 945–960.

Babur,O. *et al.* (2015) Systematic identification of cancer driving signaling pathways based on mutual exclusivity of genomic alterations. *Genome Biol.*, **16**, 45.

Bansal,N. *et al.* (2004) Correlation clustering. *Mach. Learn.*, **56**, 89–113.

Bashashati,A. *et al.* (2012) Drivernet: uncovering the impact of somatic driver mutations on transcriptional networks in cancer. *Genome Biol.*, **13**, R124.

Brennan,C.W. *et al.* (2013) The somatic genomic landscape of glioblastoma. *Cell*, **155**, 462–477.

Carter,H. *et al.* (2009) Cancer-specific high-throughput annotation of somatic mutations: computational prediction of driver missense mutations. *Cancer Res.*, **69**, 6660–6667.

Charikar,M. *et al*. (2003) Clustering with qualitative information. In: *Proceedings. 44th Annual IEEE Symposium on Foundations of Computer Science, 2003*. IEEE, pp. 524–533.

Ciriello,G. *et al*. (2012) Mutual exclusivity analysis identifies oncogenic network modules. *Genome Res*., **22**, 398–406.

Clement,V. *et al*. (2007) HEDGEHOG-GLI1 signaling regulates human glioma growth, cancer stem cell self-renewal, and tumorigenicity. *Curr. Biol*., **17**, 165–172.

Dees,N.D. *et al*. (2012) Music: identifying mutational significance in cancer genomes. *Genome Res*., **22**, 1589–1598.

Fisher,R.A. (1922) On the mathematical foundations of theoretical statistics. Philosophical Transactions of the Royal Society of London. *Series A, Containing Papers of a Mathematical or Physical Character*, **222**, 309–368.

Futreal,P.A. *et al*. (2004) A census of human cancer genes. *Nat. Rev. Cancer*, **4**, 177–183.

Gao,J. *et al*. (2013) Integrative analysis of complex cancer genomics and clinical profiles using the cBioPortal. *Sci. Signal*, **6**, pl1.

Garcia-Alonso,L. *et al*. (2014) The role of the interactome in the maintenance of deleterious variability in human populations. *Mol. Syst. Biol*., **10**, 752.

Gonzalez-Perez,A. and Lopez-Bigas,N. (2012) Functional impact bias reveals cancer drivers. *Nucleic Acids Res*., **40**, e169.

Gratas,C. *et al*. (1997) Fas ligand expression in glioblastoma cell lines and primary astrocytic brain tumors. *Brain Pathol*., **7**, 863–869.

Hanahan,D. and Weinberg,R.A. (2011) Hallmarks of cancer: the next generation. *Cell*, **144**, 646–674.

Hartigan,J.A. and Wong,M.A. (1979) Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, **28**, 100–108.

Hou,J.P. and Ma,J. (2014) Dawnrank: discovering personalized driver genes in cancer. *Genome Med*., **6**, 56.

Inoue,S. *et al*. (2013) Mule/Huwe1/Arf-BP1 suppresses Ras-driven tumorigenesis by preventing c-Myc/Miz1-mediated down-regulation of p21 and p15. *Genes Dev*., **27**, 1101–1114.

Lawrence,M.S. *et al*. (2013) Mutational heterogeneity in cancer and the search for new cancer-associated genes. *Nature*, **499**, 214–218.

Leiserson,M.D. *et al*. (2013) Simultaneous identification of multiple driver pathways in cancer. *PLoS Comput. Biol*., **9**, e1003054.

Leiserson,M.D. *et al*. (2015a) CoMEt: a statistical approach to identify combinations of mutually exclusive alterations in cancer. *Genome Biol*., **16**, 160.

Leiserson,M.D. *et al*. (2015b) Pan-cancer network analysis identifies combinations of rare somatic mutations across pathways and protein complexes. *Nat. Genet*., **47**, 106–114.

Magnusson,C. *et al*. (2011) Cysteinyl leukotriene receptor expression pattern affects migration of breast cancer cells and survival of breast cancer patients. *Int. J. Cancer*, **129**, 9–22.

Manolakos,A. *et al*. (2014) Camodi: a new method for cancer module discovery. *BMC Genomics*, **15**, S8.

Mermel,C.H. *et al*. (2011) GISTIC2.0 facilitates sensitive and confident localization of the targets of focal somatic copy-number alteration in human cancers. *Genome Biol*., **12**, R41.

Network,C.G.A. *et al*. (2012) Comprehensive molecular portraits of human breast tumours. *Nature*, **490**, 61–70.

Ng,S. *et al*. (2012) Paradigm-shift predicts the function of mutations in multiple cancers using pathway impact analysis. *Bioinformatics*, **28**, i640–i646.

Paull,E.O. *et al*. (2013) Discovering causal pathways linking genomic events to transcriptional states using tied diffusion through interacting events (TieDIE). *Bioinformatics*, **29**, 2757–2764.

Pe'er,D. and Hacohen,N. (2011) Principles and strategies for developing network models in cancer. *Cell*, **144**, 864–873.

Porta-Pardo,E. *et al*. (2015) A pan-cancer catalogue of cancer driver protein interaction interfaces. *PLoS Comput. Biol*., **11**, e1004518.

Puleo,G.J. and Milenkovic,O. (2015) Correlation clustering with constrained cluster sizes and extended weights bounds. *SIAM J. Optim*., **25**, 1857–1872.

Rosner,B. and Grove,D. (1999) Use of the Mann–Whitney U-test for clustered data. *Stat. Med*., **18**, 1387–1400.

Roy,D. *et al*. (2011) Tumor suppressor genes FHIT and WWOX are deleted in primary effusion lymphoma (PEL) cell lines. *Blood*, **118**, e32–e39.

Serão,N.V. *et al*. (2011) Cell cycle and aging, morphogenesis, and response to stimuli genes are individualized biomarkers of glioblastoma progression and survival. *BMC Med. Genomics*, **4**, 49.

Shann,Y.J. *et al*. (2008) Genome-wide mapping and characterization of hypomethylated sites in human tissues and breast cancer cell lines. *Genome Res*., **18**, 791–801.

Skiena,S. (1990) Dijkstra's algorithm. *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica*. Addison-Wesley, Reading, MA, pp. 225–227.

Sridhar,S. *et al*. (2013). An approximate, efficient LP solver for LP rounding. In: *Advances in Neural Information Processing Systems*, pp. 2895–2903.

Stambolic,V. *et al*. (1998) Negative regulation of PKB/Akt-dependent cell survival by the tumor suppressor PTEN. *Cell*, **95**, 29–39.

Stemke-Hale,K. *et al*. (2008) An integrative genomic and proteomic analysis of PIK3CA, PTEN, and AKT mutations in breast cancer. *Cancer Res*., **68**, 6084–6091.

Tomasetti,C. *et al*. (2015) Only three driver gene mutations are required for the development of lung and colorectal cancers. *Proc. Natl. Acad. Sci. U. S. A*., **112**, 118–123.

Torkamani,A. and Schork,N.J. (2009) Identification of rare cancer driver mutations by network reconstruction. *Genome Res*., **19**, 1570–1578.

Vandin,F. *et al*. (2012) De novo discovery of mutated driver pathways in cancer. *Genome Res*., **22**, 375–385.

Vazquez,A. *et al*. (2008) The genetics of the p53 pathway, apoptosis and cancer therapy. *Nat. Rev. Drug Discov*., **7**, 979–987.

Vogelstein,B. *et al*. (2013) Cancer genome landscapes. *Science*, **339**, 1546–1558.

Zhang,J. *et al*. (2013) Identification of mutated core cancer modules by integrating somatic mutation, copy number variation, and gene expression data. *BMC Syst. Biol*., **7**, S4.

Zhang,J. *et al*. (2014) Discovery of co-occurring driver pathways in cancer. *BMC Bioinformatics*, **15**, 271.

Zhao,J. *et al*. (2012) Efficient methods for identifying mutated driver pathways in cancer. *Bioinformatics (Oxford, England)*, **28**, 2940–2947.

Zimmerman,D.W. (1987) Comparative power of Student t test and Mann–Whitney U test for unequal sample sizes and variances. *J. Exp. Educ*., **55**, 171–174.