Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Preprint

OXFORD

Genome Analysis

BiGPICC: a graph-based approach to identifying carcinogenic gene combinations from mutation data

Vladyslav Oles [1,*], Sajal Dash [1,*] and Ramu Anandakrishnan [2]

[1]National Center for Computational Sciences, Oak Ridge National Laboratory and
[2]Edward Via College of Osteopathic Medicine, Virginia Tech.

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Genome data from cancer patients encapsulates explicit and implicit relationships between the presence of a gene mutation and cancer occurrence in a patient. Different types of cancer in human are thought to be caused by combinations of two to nine gene mutations. Identifying these combinations through traditional exhaustive search requires the amount of computation that scales exponentially with the combination size and in most cases is intractable for even the cutting-edge supercomputers.
Results: We propose a parameter-free heuristic approach that leverages the intrinsic topology of gene-patient mutations to identify carcinogenic combinations. The biological relevance of the identified combinations is measured by using them to predict the presence of tumor in previously unseen samples. The resulting classifiers for 16 cancer types perform on par with exhaustive search results, and score the average of 80.1% sensitivity and 91.6% specificity for the best choice of hit range per cancer type. Our approach is able to find higher-hit carcinogenic combinations targeting which would take years of computations using exhaustive search.
Availability: Our code and the datasets for two cancer types are available at `https://code.ornl.gov/vo0/bigpicc`. The datasets for other cancer types are available upon demand.

1 Introduction

Multi-hit theory of carcinogenesis states that it takes combinations of gene mutations to initiate carcinogenesis in humans (Sutherl and Bailar III, 1984). Clinical studies and a body of mathematical models established the length of the combinations (hits) to two-nine depending on the cancer types (Tomasetti *et al.*, 2015; Zhang and Simon, 2005; Little and Wright, 2003; Ashley, 1969; Anandakrishnan *et al.*, 2019). However, most computational efforts to find carcinogenic mutations focus on finding individual "driver mutations" responsible for carcinogenesis (Tian *et al.*, 2014; Tamborero *et al.*, 2013; Dees *et al.*, 2012; Kumar *et al.*, 2016). While these mostly mutational frequency and signature based methods identify driver mutations associated with increased risk of cancer, these mutations by themselves cannot cause cancer.

Cancer screening methods based on genetic predisposition rely heavily on identifying driver mutations. But a subset of people with a certain genetic predisposition never get the cancer, and the others get the cancer as they grow older and accumulated more mutations. For example, women under the age of 20 with the BRCA1 mutation are almost unlikely to get breast cancer; and, 28% of the women with BRCA1 mutation never get the cancer (Kuchenbaecker *et al.*, 2017). Similar statistics can be observed for Li Fraumeni syndrome (Mai *et al.*, 2012; Pantziarka, 2015; Guha and Malkin, 2017; Amadou *et al.*, 2018). These observations provide strong evidence that cancer in a patient is caused not by an individual driver gene mutation but rather by a combination of gradually accumulated gene mutations. Though there are several other factors responsible for cancer such as epigenetic modifications, tumor environment, and adaptive evolution, carcinogenesis is primarily a result of genetic mutations.

Different combinations of gene mutations can cause cancer in patients with same cancer type, but with different etiologies and pathologies representing different subtypes. To design and develop individualized and precision drugs for treating an individual with cancer, we need to identifying the combinations of gene mutations in that patient. Since

current computational approaches mostly look for driver gene mutations, they come short in the context of precision drug discovery. However, human genome has ∼20,000 genes and each gene can host tens of mutations. Finding the carcinogenic combinations of length h requires examining $\binom{|\mathcal{G}|}{h}$ combinations even with the simplistic assumption of one mutation per gene (here, $\mathcal{G}$ is the set of genes in human genome). For $2 \leq h \leq 9$, computationally evaluating these large number of combinations can take years. In this work, we propose an efficient computational method for identifying carcinogenic gene mutations.

1.1 Exhaustive search methods

A recent body of work (Dash *et al.* (2019); Al Hajri *et al.* (2020); Dash *et al.* (2021)) explored all possible gene combinations to identify all possible combinations responsible for cancer by comparing tumor and normal samples from cancer patients. They mapped the problem to a weighted set cover problem, where each combination is associated with a set of tumor samples where that combination of genes is mutated simultaneously, and each set is assigned a weight based on their classification accuracy when used for differentiating between tumor and normal samples. They then developed an approximate algorithm to identify minimum weight set cover that requires $O\left(\binom{|\mathcal{G}|}{h}\right)$ computations for identifying a single combination of length h. For $|\mathcal{G}| \approx$ 20,000, for human genome, and $h \geq 5$ it can take days to years on one of the fastest supercomputers in the world, Summit. While investigating every possible combination is desirable, the sheer cost of doing so is prohibitive.

Cancer Rule Set Optimization (CRSO) by Klein et al. Klein *et al.* (2021) developed a method for inferring the combinations of gene mutations cooperating to drive tumor formation in individual patients. Their approach relies on a combination of exhaustive search and heuristic when navigating the combinatorial space for the human genome, and can identify carcinogenic combinations of length h for $h \leq 6$.

1.2 Graph algorithms in cancer genomics

Graphs are combinatorial structures that model pairwise relationships between objects, e.g. protein-protein interactions (PPI) or mutations of genes in humans. In recent years, a number of studies have relied on graph-based analyses to identify combinations of cancer drivers Cho *et al.* (2016); Leiserson *et al.* (2015); Leung *et al.* (2014); Ruffalo *et al.* (2015); Drake *et al.* (2016); Dimitrakopoulos *et al.* (2018). A typical approach is to enrich a molecular network of choice using gene-sample mutation data, then analyze the resulting graph, usually by means of network propagation, to identify clusters of carcinogenic genes. For example, Leiserson *et al.* (2015) uses a heat diffusion algorithm on PPI networks to find their mutated subnetworks, based on the gene mutation frequencies extracted from data. The identified communities (or modules) are hypothesized to represent carcinogenic gene combinations, and many of the corresponding proteins have a documented role in development of cancer. In a similar vein, Leung *et al.* (2014) employs a greedy search to identify co-mutated communities in gene networks. Cho *et al.* (2016) relies on gene neighborhoods in functional networks in combination with mutation information.

Unlike the above studies, Iranzo *et al.* (2018) directly uses the graphical structure of mutation data to examine community structure in cancer drivers. The approach of this work is similar to ours in the idea of leveraging the topology of gene-sample mutations to learn about group interactions of carcinogenic genes. Beside using vastly different methodologies for it, a key distinction is that Iranzo et al. studies the ∼1% of the human genome that has known associations with cancer, while our work aims to discover novel carcinogenic combinations by analyzing the full genome.

1.3 Our contribution

Here we present BiGPICC (Bipartite Graph Partitioning for Identifying Carcinogenic Combinations) — a parameter-free approach to finding multi-hit combinations based on the topology of mutation data. Analyzing mutation data is potentially more insightful for cancer genomics than using molecular networks, as the latter are simplified abstractions of gene or protein interactions. To the best of our knowledge, this is one of the first network-based algorithms working directly with gene-sample mutations.

We formulate the search in terms of a community detection problem on a bipartite graph representation of the data, and design and implement an algorithm for solving it. Our numerical experiments on Summit supercomputer for 16 cancer types demonstrate that it identifies combinations of comparable biological relevance to state-of-the-art results, while requiring only a fraction of their compute resource. An additional advantage of our approach is its capacity to find (5+)-hit combinations, unavailable to the existing methods due to high computational cost. We use the available data to produce plausible carcinogenic combinations of genes that can be used in biomedical applications.

2 Methods

Our approach for identifying carcinogenic combinations from data on gene mutations in tumor and normal tissues is outlined in Figure 1. Viewing the data as a bipartite graph, we iteratively partition it using community detection to find candidate gene combinations whose mutations tend to occur in tumor tissues, then filter out those with frequent mutations in normal samples. Subsequently, a set of carcinogenic combinations is identified by selecting a minimum set cover of the tumor samples from the filtered candidate pool. Finally, we assess the relevance of the identified combinations from the minimum set cover by using them to differentiate between tumor and normal samples on previously unseen data. A detailed description of these steps is provided below.

2.1 Graph formulation

Let $\mathcal{G}$ and $\mathcal{S}$ denote the sets of genes and samples in the data, respectively. The input data contains information about whether a mutation of gene g was observed in sample s for every gene-sample pair $(g, s) \in \mathcal{G} \times \mathcal{S}$. Specifically, the data is stored in a binary $|\mathcal{G}| \times |\mathcal{S}|$ matrix $\mathbf{D}$ whose entries are 1 if mutation of the corresponding gene was observed in the corresponding sample, and 0 otherwise. This information can be equivalently represented as an unweighted bipartite graph on the vertices of two distinct classes $\mathcal{G}$ and $\mathcal{S}$, where an edge connects some $g \in \mathcal{G}$ and $s \in \mathcal{S}$ if and only if the corresponding mutation has been observed. We denote such a graph G, and notice that the adjacency matrix of G is the symmetric block matrix $\begin{bmatrix} \mathbf{0} & \mathbf{D} \\ \mathbf{D}^T & \mathbf{0} \end{bmatrix}$. The presence of both normal and tumor samples in the data means that the vertex class $\mathcal{S}$ is itself partitioned into two subclasses, denoted $\mathcal{S}^{\text{norm}}$ and $\mathcal{S}^{\text{tum}}$ respectively, with opposing significance to the problem. To isolate the information about gene mutations in tumor tissues, we consider the graph $G^{\text{tum}} := G[\mathcal{G} \sqcup \mathcal{S}^{\text{tum}}]$, the induced subgraph of G on the vertices $\mathcal{G} \sqcup \mathcal{S}^{\text{tum}}$.

Recall that a *community* is a subset of graph vertices that are more densely connected with one another than with the rest of the graph, according to some metric. Any community C in G (or G^{tum}) consists of the gene component $C_{\mathcal{G}} := C \cap \mathcal{G}$ — a combination of genes whose mutations tend to occur together, and the sample component $C_{\mathcal{S}} := C \cap \mathcal{S}$ (or $C \cap \mathcal{S}^{\text{tum}}$, respectively) — the samples in which these genes co-mutate the most. Because bipartite communities of multiple vertices must contain vertices from both classes to allow for internal connections, both $C_{\mathcal{G}}$ and $C_{\mathcal{S}}$ are non-empty whenever $|C| > 1$.

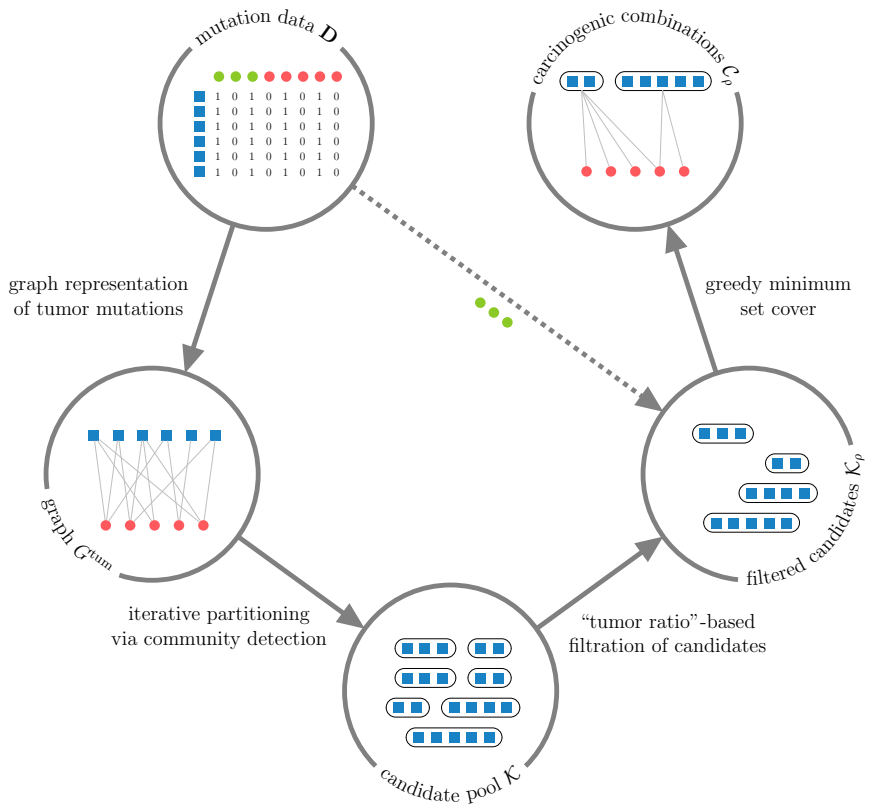

Fig. 1. Outline of BiGPICC approach.

For some gene $g \in \mathcal{G}$, let $M(g) \subset \mathcal{S}$ denote the samples in which a mutation of g has been observed. Furthermore, for a gene combination $C_\mathcal{G}$, denote the samples in which these genes are jointly mutated as $M(C_\mathcal{G}) \coloneqq \bigcap_{g \in C_\mathcal{G}} M(g)$. Analogously, we define the tumor samples with a mutation of g as $M^{\text{tum}}(g) \coloneqq M(g) \cap \mathcal{S}^{\text{tum}}$, and with a joint mutation of $C_\mathcal{G}$ — as $M^{\text{tum}}(C_\mathcal{G}) \coloneqq M(C_\mathcal{G}) \cap \mathcal{S}^{\text{tum}}$. If $C_\mathcal{G}$ is carcinogenic, its joint mutation is thought to explain some number of tumor samples from the data, so $C_\mathcal{G}$ must be fully connected to the non-empty set $M^{\text{tum}}(C_\mathcal{G})$ in the graph representation of data. For any unrelated gene $g' \in \mathcal{G} \setminus C_\mathcal{G}$, the connectivity between genes $C_\mathcal{G} \cup \{g'\}$ and samples $M^{\text{tum}}(C_\mathcal{G})$ is expected to be less dense, because a mutation of g' is unlikely to appear in every sample from $M^{\text{tum}}(C_\mathcal{G})$ by chance unless it is mutated in most of $\mathcal{S}^{\text{tum}}$. For example, if $C_\mathcal{G}$ is jointly mutated in $|M^{\text{tum}}(C_\mathcal{G})| = 10$ tumor samples and g' is mutated in half of all tumor samples (i.e. $\left|M^{\text{tum}}(g')\right| = 0.5|\mathcal{S}^{\text{tum}}|$), the probability of its mutation to appear in all of $M^{\text{tum}}(C_\mathcal{G})$ is

$$\mathbb{P}_{g'}\left[M^{\text{tum}}(C_\mathcal{G}) \subseteq M^{\text{tum}}(g')\right] = \left(\frac{\left|M^{\text{tum}}(g')\right|}{|\mathcal{S}^{\text{tum}}|}\right)^{|M^{\text{tum}}(C_\mathcal{G})|} \approx 0.001.$$

It follows that any carcinogenic gene combination $C_\mathcal{G}$ is expected to correspond to a community $C_\mathcal{G} \sqcup M^{\text{tum}}(C_\mathcal{G})$ in G^{tum}. At the same time, the converse is not necessarily true — because carcinogenic combinations are assumed to jointly mutate in predominantely tumor samples, not every community in G^{tum} has its gene component as a carcinogenic combination. For example, if tumor samples are but a small fraction of all the samples with a joint mutation of $C_\mathcal{G}$ (i.e. $|M^{\text{tum}}(C_\mathcal{G})| \ll |M(C_\mathcal{G})|$), then $C_\mathcal{G}$ is unlikely a carcinogenic combination even despite $C_\mathcal{G} \sqcup M^{\text{tum}}(C_\mathcal{G})$ being a community in G^{tum}. Therefore, the task of finding carcinogenic combinations of gene mutations can be cast as a problem of identifying communities with some desired degree of tumor prevalence in their sample components.

2.2 Community detection

We use the *Constant Potts Model* (CPM) to formally define the notion of community structure in a graph. CPM was proposed in Traag *et al.* (2011) as an alternative to the commonly-chosen *modularity* approach that does not suffer from the issue of inconsistent communities across different scales. For the case of G^{tum}, an unweighted bipartite graph on the vertex classes $\mathcal{G}$ and $\mathcal{S}^{\text{tum}}$, CPM formalizes its partition into disjoint communities $\mathcal{P}$ (i.e. $\bigsqcup_{C \in \mathcal{P}} C = \mathcal{G} \sqcup \mathcal{S}^{\text{tum}}$) as desirable if it maximizes the partition quality

$$\sum_{C \in \mathcal{P}} (e(C) - \gamma |C_\mathcal{G}||C_\mathcal{S}|), \qquad (*)$$

where $e(C)$ is the total number of internal edges in community C (that is, between $C_\mathcal{G} = C \cap \mathcal{G}$ and $C_\mathcal{S} = C \cap \mathcal{S}^{\text{tum}}$), and $\gamma \in [0, 1]$ is the so-called resolution parameter. This parameter can be viewed as the density of internal connectivity required from a set of vertices C for it to qualify as a community — or, more specifically, for C to positively contribute to the partition quality $(*)$. Notice that whenever gene component or sample component of C are empty, its contribution to $(*)$ is zero.

To identify communities within a graph, we rely on the Leiden algorithm Traag *et al.* (2019), chosen for its guarantees on the community connectivity and speed. The Leiden algorithm starts by viewing every vertex as a separate community, and then alternates between moving nodes between communities, refining the partition, and aggregating communities

into single vertices to reduce the graph size, until no further improvement to the chosen partition quality (in our case, $(*)$) can be made.

Because the possible number of carcinogenesis drivers in a combination is known or hypothesized for many cancer types, we are interested in finding communities whose gene component size $h := |C_{\mathcal{G}}|$ is within a certain range $l \leq h \leq u$ obtained from the literature. To enforce this size constraint, we iteratively refine the identified communities using the Leiden algorithm until the size of their gene component does not exceed u, the maximum possible number of carcinogenesis drivers. Specifically, if a community C in G^{tum} has gene component $C_{\mathcal{G}}$ that is too large, we refine it by first extending its sample component to be $\bigcup_{g \in C_{\mathcal{G}}} M^{\text{tum}}(g)$ to include all relevant tumor samples, then applying the Leiden algorithm to the resulting subgraph $G^{\text{tum}}[C_{\mathcal{G}} \sqcup \bigcup_{g \in C_{\mathcal{G}}} M^{\text{tum}}(g)]$ to partition $C_{\mathcal{G}}$ based on their mutations. Every time the Leiden algorithm is run on a (sub)graph induced by genes $C_{\mathcal{G}}$ and samples $C_{\mathcal{S}}$, the CPM resolution parameter is set to the connectivity density of this (sub)graph,

$$\gamma = \frac{e(C_{\mathcal{G}} \sqcup C_{\mathcal{S}})}{|C_{\mathcal{G}}||C_{\mathcal{S}}|},$$

requiring the communities to be internally connected at least as densely as the (sub)graph itself. After filtering out the the communities whose gene component is too small from the results, the gene components of the remaining communities are considered candidates for carcinogenic combinations.

Procedure 1 Iterative Partitioning

Input: graph G^{tum} on vertices $\mathcal{G} \sqcup \mathcal{S}^{\text{tum}}$, carcinogenic combination size bounds l and u
Output: candidates for carcinogenic combinations $\mathcal{K}^{\bullet}$

$\mathcal{K}^{\bullet} \leftarrow \emptyset$
$\mathcal{Q} \leftarrow \{\mathcal{G} \sqcup \mathcal{S}^{\text{tum}}\}$ ▷ *initialize queue of communities to partition*
for $C \in \mathcal{Q}$ **do**
  $\mathcal{Q} \leftarrow \mathcal{Q} \setminus \{C\}$
  $\mathcal{P} \leftarrow \texttt{Leiden}(G^{\text{tum}}[C])$ ▷ *partition C into smaller communities*
  for $C' \in \mathcal{P}$ **do**
    if $C'_{\mathcal{G}} \subsetneq C_{\mathcal{G}}$ **then** ▷ *ensure gene component of C was reduced by the partitioning*
      if $|C'_{\mathcal{G}}| > u$ **then**
        $\mathcal{Q} \leftarrow \mathcal{Q} \cup \{C'_{\mathcal{G}} \sqcup \bigcup_{g \in C'_{\mathcal{G}}} M^{\text{tum}}(g)\}$ ▷ *set extended C' for partitioning*
      else if $l \leq |C'_{\mathcal{G}}|$ **then**
        $\mathcal{K}^{\bullet} \leftarrow \mathcal{K}^{\bullet} \cup \{C'_{\mathcal{G}}\}$ ▷ *consider gene component of C' a candidate combination*

The above approach to partitioning G^{tum} to identify candidate gene combinations is described formally in Procedure 1. Because the Leiden algorithm has a randomized component to it, every call of Procedure 1 may yield a different set of candidate gene combinations. To increase the likelihood of carcinogenic combinations to appear among the results, we perform multiple iterative partitioning passes via Procedure 1 and combine the obtained sets of candidate gene combinations $\mathcal{K}^{\bullet}$ as a joint candidate pool $\mathcal{K}$.

The iterative partitioning passes are easily parallelizable as they are conducted independently of one another. We perform them in parallel on the Summit supercomputer using MPI standard and utilizing all 42 CPU cores per Summit node. We rely on the implementation of Leiden algorithm from the Python library `leidenalg`.

2.3 Filtration of candidates

For a candidate combination $C_{\mathcal{G}} \in \mathcal{K}$, define its *tumor ratio* as the share of tumor samples among all those in which $C_{\mathcal{G}}$ is jointly mutated,

$$r(C_{\mathcal{G}}) := \frac{|M^{\text{tum}}(C_{\mathcal{G}})|}{|M(C_{\mathcal{G}})|} \in [0, 1].$$

We impose a threshold $\rho \in [0, 1]$ on the tumor ratio of candidate combinations, so that only "sufficiently carcinogenic" candidates $\mathcal{K}_{\rho} := \{C_{\mathcal{G}} \in \mathcal{K} : r(C_{\mathcal{G}}) \geq \rho\}$ (notice that $\mathcal{K} = \mathcal{K}_0$) are considered when selecting the final set of carcinogenic combinations from the pool. For example, the choice of $\rho = 1$ implies considering only candidate combinations with no joint mutations in normal samples. This particular choice however is likely to result in implausible carcinogenic combinations due to the possibility of non-mutagenic drivers behind carcinogenesis and potential inaccuracies of genomic data.

2.4 Minimum set cover

For a reasonable choice of ρ, the set of tumor samples explained by the "sufficiently carcinogenic" candidates,

$$\mathcal{S}^{\text{tum}}_{\rho} := \bigcup_{C_{\mathcal{G}} \in \mathcal{K}_{\rho}} M^{\text{tum}}(C_{\mathcal{G}}),$$

is expected to coincide with $\mathcal{S}^{\text{tum}}$ assuming that every tumor there is caused by a joint mutation of some gene combination from $\mathcal{G}$ whose size is in the hypothesized range.

Given the threshold ρ, we select carcinogenic combinations as a subset of $\mathcal{K}_{\rho}$ whose joint mutations explain $\mathcal{S}^{\text{tum}}_{\rho}$ most concisely, by constructing a *minimum set cover* of $\mathcal{S}^{\text{tum}}_{\rho}$ using sets of tumor samples from $\{M^{\text{tum}}(C_{\mathcal{G}}) : C_{\mathcal{G}} \in \mathcal{K}_{\rho}\}$. The minimum set cover problem is a fundamental optimization aiming to find the smallest subcollection of sets (from some given collection) whose union fully covers some set in question. While finding the exact solution to the problem is known to be NP-complete, tractable approximation algorithms exist. We employ the standard greedy algorithm that iteratively chooses a candidate combination from $\mathcal{K}_{\rho}$ to cover the most of yet unexplained samples in $\mathcal{S}^{\text{tum}}_{\rho}$ on each step (see Procedure 2). The algorithm takes linear time in $|\mathcal{S}^{\text{tum}}_{\rho}|$, and the size of its resulting set of carcinogenic combinations $\mathcal{C}_{\rho}$ is guaranteed to approximate the size of the true solution within the factor of $\ln m$, where $m := \max\{|M^{\text{tum}}(C_{\mathcal{G}})| : C_{\mathcal{G}} \in \mathcal{K}_{\rho}\}$ is the biggest number of tumor samples explained by a "sufficiently carcinogenic" candidate.

2.5 Learning threshold for the tumor ratio

To choose the best value of ρ, we frame the BiGPICC pipeline as a classification problem by using the selected carcinogenic combinations $\mathcal{C}_{\rho}$ to predict whether previously unseen samples are normal or tumor. Namely, a sample is classified as tumor if it has a joint mutation of any

Procedure 2 Greedy Minimum Cover

Input: samples $\mathcal{S}^{\text{tum}}_{\rho}$ to cover, "sufficiently carcinogenic" candidate combinations $\mathcal{K}_{\rho}$
Output: carcinogenic combinations $\mathcal{C}_{\rho}$

$\mathcal{C}_{\rho} \leftarrow \emptyset$
while $\mathcal{S}^{\text{tum}}_{\rho} \neq \emptyset$ **do**
  $C_{\mathcal{G}} \leftarrow \arg\max_{C'_{\mathcal{G}} \in \mathcal{K}_{\rho}} |M^{\text{tum}}(C'_{\mathcal{G}}) \cap \mathcal{S}^{\text{tum}}_{\rho}|$ ▷ *find $C_{\mathcal{G}}$ covering most unexplained samples*
  $\mathcal{C}_{\rho} \leftarrow \mathcal{C}_{\rho} \cup \{C_{\mathcal{G}}\}$
  $\mathcal{S}^{\text{tum}}_{\rho} \leftarrow \mathcal{S}^{\text{tum}}_{\rho} \setminus M^{\text{tum}}(C_{\mathcal{G}})$

Abbreviation	Cancer type	h, # hits	$\lvert\mathcal{G}\rvert$, # genes	$\lvert\mathcal{S}\rvert$, # samples	# mutations
BLCA	Bladder urothelial carcinoma	4, 6-7, 7	18,511	699	519,225
BRCA	Breast Invasive arcinoma	3, 2-3, 6-7	19,411	1,242	837,207
CESC	Cervical squamous cell carcinoma and endocervical adenocarcinoma	5, 6-7	18,813	605	459,351
COAD	Colon adenocarcinoma	3, 4, 5, 6	19,487	716	665,932
GBM	Glioblastoma multiforme	2, 3	18,920	662	852,723
HNSC	Head and neck squamous cell carcinoma	4, 5	18,707	801	635,410
KIRP	Kidney renal papillary cell carcinoma	8	18,117	559	429,005
LGG	Brain lower grade glioma	3, 6	17,770	810	495,887
LIHC	Liver hepatocellular carcinoma	8	17,592	643	438,879
LUAD	Lung adenocarcinoma	3, 6-7	18,783	740	629,351
LUSC	Lung squamous cell carcinoma	5	18,454	636	488,441
OV	Ovarian serous cystadenocarcinoma	2, 6-7	18,814	658	671,888
PRAD	Prostate adenocarcinoma	6-7, 7	17,274	752	476,115
SARC	Sarcoma	5, 8	17,687	550	379,824
THCA	Thyroid carcinoma	5, 6	18,264	752	524,992
UCEC	Uterine corpus endometrial carcinoma	2, 6-7	19,889	826	911,903

Table 1. Hit ranges required for carcinogenesis and data parameters for the 16 cancer types.

$C_{\mathcal{G}} \in \mathcal{C}_\rho$, and as normal otherwise. Notice that an increase in the value of ρ improves the precision score of the classifier while also growing the size of $\mathcal{C}_\rho$ (as the choice of $C_{\mathcal{G}} \in \mathcal{K}_\rho$ at each step of Procedure 2 is narrowed). Because the number of gene combinations in $\mathcal{C}_\rho$ corresponds to classifier's complexity and therefore its propensity to overfit, $1 - \rho$ can be viewed as the amount of regularization in the training, a trade-off between the "carcenogenicity" and the generalizability of learned combinations.

After designating 25% of all the samples as validation set $\mathcal{S}_{\text{valid}}$, we employ 4-fold cross-validation on the remaining samples to find the optimal value of hyperparameter ρ: samples $\mathcal{S} \setminus \mathcal{S}_{\text{valid}}$ are split into 4 equal parts and, in 4 separate scenarios, the pipeline is run on the dataset $\mathcal{G} \sqcup (\mathcal{S} \setminus \mathcal{S}_{\text{valid}})$ with one of them held-out to produce $\mathcal{C}_\rho$ for each value of ρ ranging from 0 to 1 with the increment of 0.01. The values of ρ are then assessed by averaging the performance of $\mathcal{C}_\rho$-based binary classifier on the held-out part across the 4 scenarios. We use the *Matthews correlation coefficient* (MCC, also called the *phi coefficient*) — which takes values between -1 and 1 and is considered more informative than the commonly chosen F1 and accuracy scores Chicco and Jurman (2020) — as a single performance metric. Because constructing $\mathcal{C}_\rho$ and evaluating its performance for individual values of ρ can be done in parallel, the search for best ρ does not introduce a significant computational overhead. The value of ρ delivering highest mean MCC is then used to train the final classifier (by re-running BiGPICC pipeline) on the dataset $\mathcal{G} \sqcup (\mathcal{S} \setminus \mathcal{S}_{\text{valid}})$ and report its performance on $\mathcal{S}_{\text{valid}}$.

3 Results

3.1 Experimental setup

Our somatic mutation data was collected in mutation annotation format (MAF) from The Cancer Genome Atlas (TCGA) for 16 cancer types using Mutect2 software. We identify a set of 331 matched blood-derived normal samples from all cancer types. We use the Variant Effect Predictor (VEP) to determine the location (intron, exon, UTR) and effect of these variants (synonymous, non-synonymous, missense, nonsense), and only consider protein-altering variants (non-synonymous, nonsense, and insertion/deletions in exons).

We apply BiGPICC to 16 datasets for different cancer types in 33 numerical experiments, each exploring a combination size h or its range for a cancer type according to existing literature. Each of the 16 datasets contains the same set of 331 normal tissue samples. Table 1 shows the hypothesized ranges of h and the dataset parameters for each cancer type. For each cancer type and multi-hit range, we run the community detection step of BiGPICC (Procedure 1) 10,000 times to increase the probability of including all carcinogenic combinations to the candidate pool.

3.2 Classification performance

In each numerical experiment we build a classifier based on the identified gene combinations of desired size to differentiate between tumor and normal samples. We use the MCC, specificity, sensitivity, and F1 metrics to highlight various aspects of our classifiers. Each of the numerical experiments performs the 4-fold cross-validation runs and chooses the best value of ρ based on the highest mean MCC across the 4 folds. We demonstrate the effect of the value of ρ on the MCC in individual numerical experiments in Figure 2. After using the chosen value of ρ to train the final classifier on $\mathcal{G} \sqcup (\mathcal{S} \setminus \mathcal{S}_{\text{valid}})$, we report its combination count and performance metrics on the previously unseen validation samples $\mathcal{S}_{\text{valid}}$ in Table 2. The MCC ranges from 0.336 (BLCA, 7-hit combinations) to 0.911 (COAD, 3), specificity — from 0.759 (GBM, 2) to 1 (SARC, 5), sensitivity — from 0.37 (BLCA, 7) to 0.987 (BRCA, 2-3), and F1 score — from 0.511 (BLCA, 7) to 0.974 (BRCA, 2-3). The spread of classification performance within the same cancer type is likely attributed to the biological relevancy of the chosen hit range in individual experiments. The variation in performance across different cancer types can be explained by the data availability in each case and the representativity of available dataset for the high-dimensional search space. In particular, poor performance of PRAD and BLCA classifiers is consistent with the results from Klein *et al.* (2021) and Dash *et al.* (2019), which were based on the same data.

3.3 Comparison against exhaustive search results

Weighted set cover-based method identified carcinogenic combinations of length 2, 3, and 4 for various cancer types through an exhaustive search (ES) of all possible gene combinations of given length Al Hajri *et al.* (2020). We compare the classification performance of carcinogenic combinations identified by BiGPICC for $h < 5$ against their reported metrics. As can be seen from Figure 3, the two methods have comparable

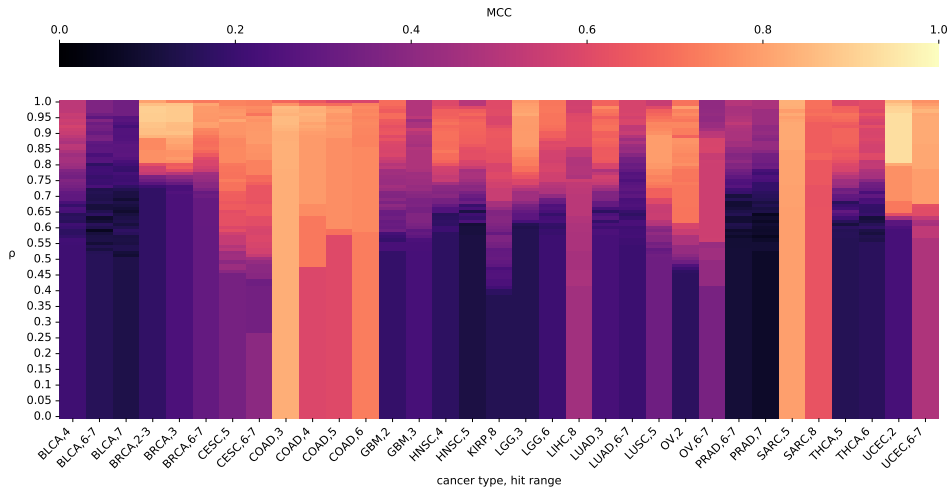

Fig. 2. MCC fold-average of the MCC in cross-validation runs for the full range of ρ values. The range of MCC is truncated to $[0, 1]$ due to the absence of negative values in our results.

Cancer type	Hit range	$\lvert\mathcal{C}_\rho\rvert$, # combinations	MCC	Specificity	Sensitivity	F1
BLCA	4	24	**0.555**	0.855	0.696	0.762
	6-7	43	0.550	0.976	0.522	0.676
	7	71	0.336	0.916	0.37	0.511
BRCA	2-3	5	**0.900**	0.892	0.987	0.974
	3	9	**0.900**	0.892	0.987	0.974
	6-7	34	0.749	0.916	0.882	0.922
CESC	5	12	0.826	0.916	0.912	0.905
	6-7	13	**0.868**	0.976	0.882	0.923
COAD	3	14	**0.911**	0.976	0.938	0.957
	4	18	0.89	0.964	0.938	0.952
	5	24	0.827	0.964	0.865	0.912
	6	33	0.747	0.916	0.833	0.874
GBM	2	22	0.447	0.759	0.687	0.712
	3	26	**0.462**	0.795	0.663	0.710
HNSC	4	12	0.734	0.855	0.881	0.889
	5	18	**0.736**	0.867	0.873	0.888
KIRP	8	30	**0.663**	0.952	0.667	0.768
LGG	3	11	**0.847**	0.916	0.933	0.937
	6	27	0.687	0.916	0.783	0.851
LIHC	8	59	**0.542**	0.952	0.538	0.677
LUAD	3	18	**0.777**	0.928	0.853	0.892
	6-7	31	0.525	0.904	0.608	0.721
LUSC	5	10	**0.723**	0.867	0.855	0.855
OV	2	15	**0.790**	0.916	0.873	0.890
	6-7	47	0.404	0.928	0.418	0.559
PRAD	6-7	45	0.436	0.831	0.6	0.692
	7	62	**0.487**	0.928	0.533	0.671
SARC	5	10	**0.840**	1	0.800	0.889
	8	19	0.626	0.988	0.545	0.698
THCA	5	26	**0.671**	0.904	0.771	0.835
	6	39	0.499	0.904	0.581	0.701
UCEC	2	3	**0.889**	0.928	0.960	0.956
	6-7	15	0.818	0.880	0.935	0.928

Table 2. Count and classification performance of identified combinations in the 33 numerical experiments. Best MCC values per cancer type are in bold.

classification performance except for GBM cancer where BiGPICC performs significantly worse than ES. In seven out of ten cases, BiGPICC outperforms ES in either specificity or sensitivity, and in two cases — in both the metrics.

One explanation for the heuristics-based BiGPICC outperforming an exhaustive search method may be suboptimality of the latter's choice of parameters for some of the datasets. In addition, the train-validation splits used in our runs are not the same as in Al Hajri *et al.* (2020). If the amount of samples in the dataset is insufficient to capture all relevant gene interactions (see Section 4), a particular choice of the split may significantly impact the resulting performance.

For example, the GBM, 2 cross-validation run achieves the fold-average MCC of 0.695, but the final GBM, 2 classifier trained on the same samples $\mathcal{S} \setminus \mathcal{S}_{\text{valid}}$ and for the same value of ρ yields the MCC of only 0.447 on the validation samples $\mathcal{S}_{\text{valid}}$. This suggests that the gene interactions behind the tumor samples in $\mathcal{S}_{\text{valid}}$ are significantly distinct from those reflected in $\mathcal{S} \setminus \mathcal{S}_{\text{valid}}$. Furthermore, we achieve the MCC of 0.791 for GBM, 2 and 0.714 for GBM, 3 on $\mathcal{S}_{\text{valid}}$ from the second attempt of a random train-validation split of the GBM data, with the new specificity for GBM, 2 outperforming its ES counterpart.

3.4 Biological relevance

There are 316 genes from Cosmic gene database among the participating genes in the combinations identified by the full-dataset runs of BiGPICC. The number of Cosmic genes identified for different cancer types and hit ranges varies (Figure 4) but is generally proportional to the total count of identified genes in an experiment. For example, the number of Cosmic genes among the 7-hit combinations for BLCA, the worst performing classifier in our experiments, is 25 out of 303. At the same time, the BRCA, 2-3 run has no Cosmic genes among the identified 10, despite delivering one of the highest classification performances. Because the Cosmic database contains the genes that are associated with cancer individually, the dissonance between the Cosmic gene counts and classification performance can be partially attributed to the emergent carcinogenic properties of combined gene mutations. In addition, unlike many other studies BiGPICC does not set a minimum requirement for the number of explained tumor samples per combination which allows uncovering rare carcinogenic combinations. The potential presence of passenger genes in the combinations identified by BiGPICC (see Section 5) would also lower the respective proportion of Cosmic genes.

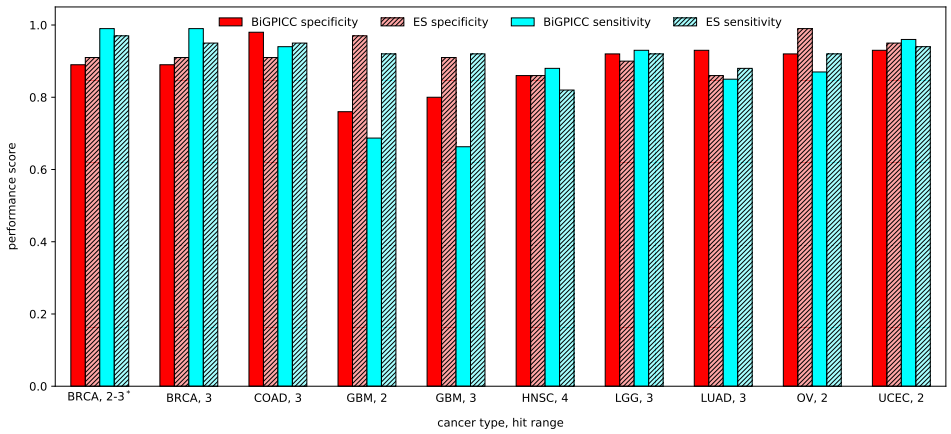

Fig. 3. Comparison between classification performances of BiGPICC and Exhaustive Search (ES), rounded to 2 decimal places. * BiGPICC runs for BRCA, 2-3 result in at most one 3-hit combination per fold and are compared against the ES results for BRCA, 2 as the best matching option.

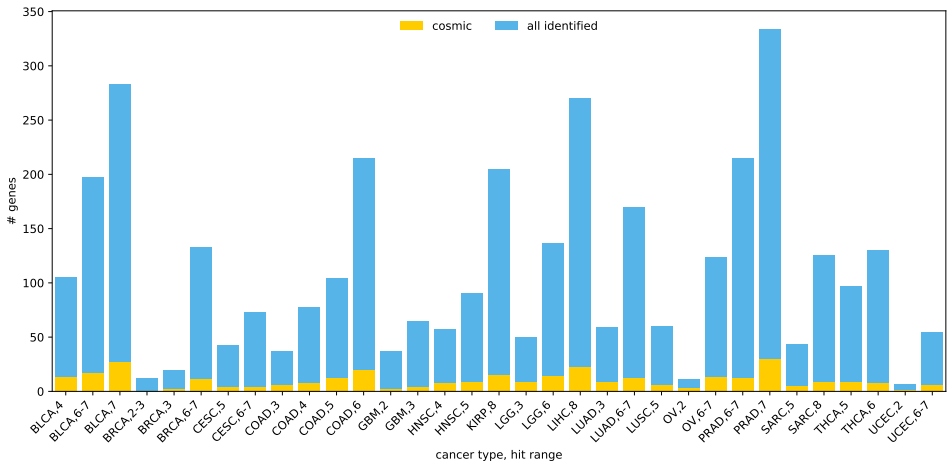

Fig. 4. The proportion of Cosmic genes in all the carcinogenic genes identified by BiGPICC in each experiment.

3.5 Runtime performance

The 4-fold cross-validation BiGPICC runs on 75% of all samples for tuning the parameter ρ were conducted using 280 Summit nodes each and took on average 26 minutes across all experiments, ranging from 11 minutes (SARC, 8) to 1 hour 29 minutes (UCEC, 2). Running the pipeline on the full dataset (without the validation samples) to measure the resulting classification performance was done using 70 nodes and took 23 minutes on average, ranging from 8 minutes (SARC, 8) to 1 hour 33 minutes (UCEC, 6-7). Adding the two mean runtimes gives that identifying carcinogenic combinations on the full dataset from scratch takes BiGPICC on average 146 node-hours on Summit.

For comparison, the ES method from Dash *et al.* (2019) requires 124 Summit node-hours to identify 4-hit combinations for BRCA cancer type based on 75% of the available samples. Assuming ideal scaling of the method, the time required to identify 5-, 6-, and 7-hit combinations using all 4,600 nodes of Summit supercomputer would be 4.4 days, 38.6 years, and 107,100 years, respectively. In contrast, the BiGPICC runtimes are not noticeably affected by the combination size h and instead mainly depend on the size and topology of the input graph, determined by the cancer type. It follows that BiGPICC is orders of magnitude faster than ES for any $h > 4$.

While ES needs to use GPUs, BiGPICC only uses CPU cores. Per Summit node, BiGPICC uses 42 CPU cores, while ES uses $6 \times 5120 = 30{,}720$ GPU cores. Though CPU and GPU cores are not directly comparable, BiGPICC is essentially using only 2% of a given compute node. So, BiGPICC can be moved to a less expensive CPU-only cluster to achieve the same runtime.

We note that the above runtimes could have been considerably shorter if not for the two issues associated with `leidenalg` library. To prevent deadlocking caused by thousands of MPI processes attempting to import the library simultaneously, we force every process to sleep for up to 8 minutes, which is counted towards the total runtime. In addition, we found that while a call to the Leiden community detection algorithm typically takes only a few seconds to finish, in $\sim$0.1% of cases it fails to terminate after over an hour, possibly due to a bug causing insignificant improvements to the partition quality to occur indefinitely. This made us allocate "spare" processes so that the desired number of iterative partitioning passes can be achieved even if some processes do not finish theirs. Due to the lack of compute resources for experimenting with the optimal percentage of spare processes, it was set to the conservative 17.6% in both cross-validation and final runs. It follows that the resulting runtime could be 1.176 times shorter if the issue with `leidenalg` library is fixed.

4 Conclusions and discussion

BiGPICC enables discovery of higher-hit combinations unavailable to exhaustive search methods even on most modern supercomputers. In all relevant cancer types considered, biological relevance of the identified combinations, measured as the classification performance, tends to drop

as the number of hits increases. An identical trend in the exhaustive search results from Al Hajri *et al.* (2020) suggests that, aside from biological reasons, the issue may lie with the so-called curse of dimensionality in both the machine learning and combinatorial contexts. An increase in the number of hits means fewer samples with joint mutation of a gene combination exist in the data, and exponential growth of the search space of all possible combinations. Therefore, the number of samples required for finding carcinogenic combinations grows with the number of hits, while our runs use the same dataset for every multi-hit range.

The classification performance of BiGPICC exhibits a significantly higher variability in sensitivity, the percentage of correctly classified tumor samples, than in specificity, the analogous percentage for normal samples. The sensitivity and specificity scores across the experiments are distributed as 0.763 ± 0.175 (mean $\pm$ 1 standard deviation) and 0.911 ± 0.054, respectively. We attribute this trend to the implicit control exhibited over the specificity score by the parameter ρ, whose learned value is consistently high in the experiments (0.938 ± 0.051). Let n_{TP}, n_{FN}, n_{TN}, and n_{FP} denote respectively the number of correctly predicted tumor samples, incorrectly predicted tumor samples, correctly predicted normal samples, and incorrectly predicted normal samples in the training dataset with the total number of samples $n = n_{\text{TP}} + n_{\text{FN}} + n_{\text{TN}} + n_{\text{FP}}$. Parameter ρ ensures that the combinations selected for the classifier have sufficient tumor ratio, which translates into controlling its precision $\frac{n_{\text{TP}}}{n_{\text{TP}}+n_{\text{FP}}}$. Assuming for simplicity no overlap between the samples in which the selected combinations are jointly mutated, the relationship is given by $\frac{n_{\text{TP}}}{n_{\text{TP}}+n_{\text{FP}}} \geq \rho$ (in general, the right hand side can be both larger or smaller depending on the tumor ratio in the overlapping samples). It follows that the number of false positives is limited by the number of tumor samples in the dataset,

$$n_{\text{FP}} \leq \frac{1-\rho}{\rho} n_{\text{TP}} \leq \frac{1-\rho}{\rho}(n_{\text{TP}} + n_{\text{FN}}) = \frac{1-\rho}{\rho} rn,$$

where $r = \frac{n_{\text{TP}}+n_{\text{FN}}}{n}$ is the ratio of tumor samples in the data. Because n_{FP} is the only factor negatively affecting the specificity $\frac{n_{\text{TN}}}{n_{\text{TN}}+n_{\text{FP}}}$, the latter is bounded from below by

$$\frac{n_{\text{TN}}}{n_{\text{TN}} + n_{\text{FP}}} = 1 - \frac{n_{\text{FP}}}{(1-r)n} \geq 1 - \frac{(1-\rho)r}{(1-r)\rho} \xrightarrow[\rho \to 1]{} 1.$$

In particular, the average value of $1 - \frac{(1-\rho)r}{(1-r)\rho}$ across the experiments is 0.865 (if r is calculated using the full datasets; it is expected to be the same along the training-validation split). At the same time, the pipeline does not exhibit control over n_{FN}, the adverse factor for the sensitivity $\frac{n_{\text{TP}}}{n_{\text{TP}}+n_{\text{FN}}}$, which is comprised by the tumor samples left unexplained by "sufficiently carcingonenic" combinations.

Unlike most state-of-the-art approaches, BiGPICC does not require manually balancing the importance of tumor and normal samples in the data. Instead, its parameter ρ which implicitly takes on this role is learned from the data using cross-validation. While the number of cross-validation folds and the increment used in the search of the optimal ρ can too be considered parameters of the method, using state-of-the-practice values for both renders our approach parameter-free for practical purposes.

5 Limitations and future work

BiGPICC does not offer a dedicated mechanism to differentiate between the driver mutations causing carcinogenesis and the passenger mutations that do not contribute to cancer formation. It means that some of the highly mutable genes among the identified ones can be mutated in tumor samples by chance and therefore lack biological relevance. In particular, each of our datasets contains between 41 and 67 genes with mutations in over 50% of the samples. This potentially leads to a big number of candidate combinations with near-identical explanatory power, which can be behind the observed phenomenon of the similar classification performance exhibited by significantly different sets of carcinogenic combinations. Under the assumption that highly mutable genes are unlikely to drive carcinogenesis, one approach to address the issue of driver-passenger distinction would be to remove genes with mutation frequency above a certain threshold from the analysis. In general, however, a sufficient number of samples in the data should amplify the signal of carcinogenic pathways on the backdrop of noise from chance mutations.

Another limitation of the method is that it is run on the gene-patient data encoding all possible mutations of a gene as a binary variable, thus ignoring the variability in mutations of individual genes. However, the BiGPICC pipeline could similarly be applied to mutation-level data for the identified carcinogenic gene combinations. Because the identified genes typically constitute less than 1% of the original gene pool with dozens of potential mutations per one gene, the mutation-level data would be no less tractable for the BiGPICC approach than the datasets used in this paper.

Assuming that the sufficient amount of data is available, an additional use of our pipeline is testing out the competing ranges of multi-hit theories by comparing the performance of learned carcinogenic combinations of the corresponding size. The ICGC database Zhang *et al.* (2019) contains roughly twice as many samples as our datasets for individual cancer types, and may be a prospective dataset to try this approach on.

A potential improvement of the pipeline would be to allow the community detection step to leverage topology of the full dataset by accounting for the mutations in normal samples. One way of doing so is to augment the graph G with negative edges between the tumor and normal samples, which would promote communities in which samples of different types do not mix. The Leiden algorithm can be used for community detection on signed graphs by finding communities separately on its positive and negative edges and subsequently combining the results. However, estimating the effect on the connectivity of every gene combination from introducing a negative edge between a pair of samples is a non-trivial combinatorial task. Therefore, any guarantees of uniformity on this combined effect would require a careful choice of the added negative edges and their weights.

6 Acknowledgements

This work was supported by the resources of the Oak Ridge Leadership Computing Facility, located in the National Center for Computational Sciences at ORNL, which is managed by UT Battelle, LLC for the U.S. DOE (under the contract No. DE-AC05-00OR22725).

References

Al Hajri, Q. *et al.* (2020). Identifying multi-hit carcinogenic gene combinations: Scaling up a weighted set cover algorithm using compressed binary matrix representation on a gpu. *Scientific reports*, **10**(1), 1–18.

Amadou, A. *et al.* (2018). Revisiting tumor patterns and penetrance in germline TP53 mutation carriers: temporal phases of Li-Fraumeni syndrome. *Curr Opin Oncol*, **30**(1), 23–29.

Anandakrishnan, R. *et al.* (2019). Estimating the number of genetic mutations (hits) required for carcinogenesis based on the distribution of somatic mutations. *PLoS computational biology*, **15**(3), e1006881.

Ashley, D. (1969). The two" hit" and multiple" hit" theories of carcinogenesis. *British journal of cancer*, **23**(2), 313.

Chicco, D. and Jurman, G. (2020). The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC genomics*, **21**(1), 1–13.

Cho, A. *et al.* (2016). Muffinn: cancer gene discovery via network analysis of somatic mutation data. *Genome biology*, **17**(1), 1–16.

Dash, S. *et al.* (2019). Differentiating between cancer and normal tissue samples using multi-hit combinations of genetic mutations. *Scientific reports*, **9**(1), 1–13.

Dash, S. *et al.* (2021). Scaling out a combinatorial algorithm for discovering carcinogenic gene combinations to thousands of gpus. In *2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 837–846. IEEE.

Dees, N. D. *et al.* (2012). Music: identifying mutational significance in cancer genomes. *Genome Res*, **22**(8), 1589–1598.

Dimitrakopoulos, C. *et al.* (2018). Network-based integration of multi-omics data for prioritizing cancer genes. *Bioinformatics*, **34**(14), 2441–2448.

Drake, J. M. *et al.* (2016). Phosphoproteome integration reveals patient-specific networks in prostate cancer. *Cell*, **166**(4), 1041–1054.

Guha, T. and Malkin, D. (2017). Inherited TP53 mutations and the Li-Fraumeni syndrome. *Cold Spring Harb Perspect Med*, **7**(4), a026187.

Iranzo, J. *et al.* (2018). Cancer-mutation network and the number and specificity of driver mutations. *Proceedings of the National Academy of Sciences*, **115**(26), E6010–E6019.

Klein, M. I. *et al.* (2021). Identifying modules of cooperating cancer drivers. *Molecular systems biology*, **17**(3), e9810.

Kuchenbaecker, K. B. *et al.* (2017). Risks of breast, ovarian, and contralateral breast cancer for BRCA1 and BRCA2 mutation carriers. *JAMA*, **317**(23), 2402–2416.

Kumar, R. D. *et al.* (2016). Unsupervised detection of cancer driver mutations with parsimony-guided learning. *Nat Genet*, **48**(10), 1288–1294.

Leiserson, M. D. *et al.* (2015). Pan-cancer network analysis identifies combinations of rare somatic mutations across pathways and protein complexes. *Nature genetics*, **47**(2), 106–114.

Leung, A. *et al.* (2014). Hypermodules: identifying clinically and phenotypically significant network modules with disease mutations for biomarker discovery. *Bioinformatics*, **30**(15), 2230–2232.

Little, M. and Wright, E. (2003). A stochastic carcinogenesis model incorporating genomic instability fitted to colon cancer data. *Mathematical biosciences*, **183**(2), 111–134.

Mai, P. *et al.* (2012). Li-Fraumeni syndrome: report of a clinical research workshop and creation of a research consortium. *Cancer Genet*, **205**(10), 479–87.

Pantziarka, P. (2015). Primed for cancer: Li Fraumeni Syndrome and the pre-cancerous niche. *Ecancermedicalscience*, **9**, 541.

Ruffalo, M. *et al.* (2015). Network-based integration of disparate omic data to identify" silent players" in cancer. *PLoS computational biology*, **11**(12), e1004595.

Sutherl, J. V. and Bailar III, J. C. (1984). The multihit model of carcinogenesis: etiologic implications for colon cancer. *Journal of Chronic Diseases*, **37**(6), 465–480.

Tamborero, D. *et al.* (2013). Oncodriveclust: exploiting the positional clustering of somatic mutations to identify cancer genes. *Bioinformatics*, **29**(18), 2238–2242.

Tian, R. *et al.* (2014). Contrastrank: a new method for ranking putative cancer driver genes and classification of tumor samples. *Bioinformatics*, **30**(17), 572–578.

Tomasetti, C. *et al.* (2015). Only three driver gene mutations are required for the development of lung and colorectal cancers. *Proceedings of the National Academy of Sciences*, **112**(1), 118–123.

Traag, V. A. *et al.* (2011). Narrow scope for resolution-limit-free community detection. *Physical Review E*, **84**(1), 016114.

Traag, V. A. *et al.* (2019). From louvain to leiden: guaranteeing well-connected communities. *Scientific reports*, **9**(1), 1–12.

Zhang, J. *et al.* (2019). The international cancer genome consortium data portal. *Nature biotechnology*, **37**(4), 367–369.

Zhang, X. and Simon, R. (2005). Estimating the number of rate limiting genomic changes for human breast cancer. *Breast cancer research and treatment*, **91**(2), 121–124.