Bioinformatics, 2022, 1–7
https://doi.org/10.1093/bioinformatics/btac312
Advance Access Publication Date: 13 May 2022
Original Paper

Systems biology

OMEN: network-based driver gene identification using mutual exclusivity

Dries Van Daele [1,*], Bram Weytjens[2], Luc De Raedt[1] and Kathleen Marchal [2,*]

[1]Department of Computer Science, KU Leuven, Leuven 3001, Belgium and [2]Department of Plant Biotechnology and Bioinformatics, Department of Information Technology, IDLab, IMEC, Gent 9000, Belgium

*To whom correspondence should be addressed.

Associate Editor: Anthony Mathelier

Received on September 28, 2021; revised on April 28, 2022; editorial decision on May 2, 2022; accepted on May 9, 2022

Abstract

Motivation: Network-based driver identification methods that can exploit mutual exclusivity typically fail to detect rare drivers because of their statistical rigor. Propagation-based methods in contrast allow recovering rare driver genes, but the interplay between network topology and high-scoring nodes often results in spurious predictions. The specificity of driver gene detection can be improved by taking into account both gene-specific and gene-set properties. Combining these requires a formalism that can adjust gene-set properties depending on the exact network context within which a gene is analyzed.

Results: We developed OMEN: a logic programming framework based on random walk semantics. OMEN presents a number of novel concepts. In particular, its design is unique in that it presents an effective approach to combine both gene-specific driver properties and gene-set properties, and includes a novel method to avoid restrictive, a priori filtering of genes by exploiting the gene-set property of mutual exclusivity, expressed in terms of the functional impact scores of mutations, rather than in terms of simple binary mutation calls. Applying OMEN to a benchmark dataset derived from TCGA illustrates how OMEN is able to robustly identify driver genes and modules of driver genes as proxies of driver pathways.

Availability and implementation: The source code is freely available for download at www.github.com/DriesVanDaele/OMEN. The dataset is archived at https://doi.org/10.5281/zenodo.6419097 and the code at https://doi.org/10.5281/zenodo.6419764.

Contact: dries.vandaele@cs.kuleuven.be or kathleen.marchal@ugent.be

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

1 Introduction

Mining cohorts of cancer genomes (International Cancer Genome Consortium *et al.*, 2010; Weinstein *et al.*, 2013) offers the potential to identify genes and/or pathways driving carcinogenesis. Gene-centric driver identification methods assess to what extent aberrant genes carrying somatic variants display properties of driver genes, such as the extent to which genes are more frequently mutated than expected by chance in a cohort, either throughout the gene (Dees *et al.*, 2012; Lawrence *et al.*, 2013) or at specific functional or clustered sites (Reimand and Bader, 2013; Tamborero *et al.*, 2013; Van den Eynden *et al.*, 2015), or the extent to which genes are affected in a cohort by aberrations with high functional impact (Gonzalez-Perez and Lopez-Bigas, 2012; Gonzalez-Perez *et al.*, 2013; Mularoni *et al.*, 2016). By relying on mutational recurrence, gene-centric methods are ideal to identify drivers that are frequently mutated in a cohort (Abeshouse *et al.*, 2015; Bailey *et al.*, 2018; Banerji *et al.*, 2012; Ciriello *et al.*, 2013). However, since in cancer the same causal pathway can become disturbed in many different ways, tumors of patients with the same disease are likely to carry different, possibly rare mutations that hit the same driver pathway. Current datasets are often underpowered to identify, based on their individual gene-centric mutational profile such rarely mutated genes as drivers. By searching for subgraphs enriched in driver properties, network-based methods in contrast can dig in the long tails of these rarely mutated genes (Dimitrakopoulos *et al.*, 2018). This network model is based on a prior interaction network in which genes are typically represented using nodes and the interactions between genes by edges.

When seeking for such subgraphs, a popular strategy involves propagating a gene-centric signal over a network model where the network model is based on a prior gene-gene interaction network. This propagation allows prioritizing drivers [MUNDIS, NetICS (Dimitrakopoulos *et al.*, 2018)], find subgraphs enriched for gene specific properties as proxies of driver pathways (HotNet2, MUNDIS) or for sub-typing [NBS (Hofree *et al.*, 2013), MUNDIS, SRF (Le Van *et al.*, 2016)].

Besides propagating gene-centric properties for prioritizing drivers (Horn *et al.*, 2018), network-based driver identification methods

can also use properties of gene-sets. Methods that exploit such gene-set properties search for connected subgraphs in the interaction network that display a high score for the relevant gene set property. A popular property is mutual exclusivity (Gao *et al.*, 2017; Kim *et al.*, 2015; Pulido-Tamayo *et al.*, 2016). Mutual exclusivity can arise because of positive epistatic effects where a single mutation in a driver pathway often yields the complete fitness advantage associated with a perturbation in that pathway and therefore additional mutations in the same pathway become redundant (Yeang *et al.*, 2008). Alternatively, mutual exclusivity can be due to negative epistatic effects where a mutation can only have a driving effect if it occurs in the absence of another mutation (van de Haar *et al.*, 2019). If genes contain mutations that are mutually exclusive they have a higher change of being true drivers. True driver mutations should therefore exhibit a pattern of mutual exclusivity. The definition of mutual exclusivity often relies on binary information: the presence or absence of mutations in the cohort. To make the search for patterns of mutual exclusivity tractable and to exclude false positives, most methods that search for patterns of mutual exclusivity filter mutations up-front—based at least partly on the mutation frequency of the genes (Babur *et al.*, 2015; Ciriello *et al.*, 2012)—and/or use scoring functions that find a good trade-off between mutational frequency and mutual exclusivity (Gao *et al.*, 2017; Kim *et al.*, 2015). Rarely mutated genes often display mutual exclusivity with more frequently mutated genes by chance because of the infrequent number of mutations in a cohort. As a result, rare driver mutations are either not considered in mutually exclusive gene sets, or deteriorate the predictions of mutual exclusivity by introducing spurious genes to the set (Pulido-Tamayo *et al.*, 2016).

Hence, network-based driver identification methods that can handle gene-set properties are ideally suited to exploit mutual exclusivity, but are not suited to identify statistically significant rare driver genes. Propagation-based methods in contrast allow for recovering driver genes with low gene specific driver scores (rare drivers). However, here spurious predictions can result from the interplay between network topology and high scoring nodes (Horn *et al.*, 2018). The specificity of driver gene detection could be improved by taking into account criteria that assess a gene set property of the immediate neighborhood of a gene of interest (such as mutual exclusivity) or an aggregate score of its gene set specific properties in addition to propagated gene-centric properties. To our knowledge, there does not yet exist a formalism that can combine the advantages of both gene specific driver properties and properties of gene sets to identify drivers.

Therefore, we propose OMEN: a flexible network-based driver identification method that captures both gene-centric properties such as the frequency with which a gene is mutated in a cohort as well as properties derived from its local context (gene-set specific properties, in case mutual exclusivity) over an interaction network. Since gene-set properties are only meaningful when considered with respect to a particular selection of genes, the influence of gene-set specific properties has to be dynamically adjusted depending on the exact network context within which a gene is analyzed. OMEN manages this using a logic programming framework based on random walk semantics. It does not explicitly enumerate all mutually exclusive patterns, but rather like SSA-ME (Pulido-Tamayo *et al.*, 2016), prioritizes genes as drivers based on their probability of belonging to a network neighborhood that displays mutual exclusivity. Not explicitly enumerating all mutually exclusive patterns reduces computational complexity and avoids heavy prefiltering.

A gene is prioritized as a driver if it has a high probability of belonging to a high-scoring subgraph of the interaction network, where subgraph scores account for the average mutational frequency of its genes across the cohort, the functional impact of the mutations, and the extent to which the mutations in that gene set display mutual exclusivity. Importantly, by reformulating mutual exclusivity in terms of functional impact scores of the contributing mutations, OMEN prevents that rarely occurring mutations with low functional impact (likely passengers) deteriorate the identified mutual exclusive gene sets. Applying OMEN to a benchmark dataset derived from TCGA illustrates how OMEN is able to robustly identify driver genes and modules of driver genes as proxies of driver pathways.

2 System and methods

2.1 Method overview

Genes that are frequently mutated in a cohort are more likely to be drivers. However, rarely mutated genes also have an increased chance of being true drivers when they are located in a subnetwork of genes that on average is highly mutated in a cohort and displays mutually exclusive mutations. OMEN is a network-based driver identification method that takes this into account. It prioritizes genes by accounting for both gene-centric and gene set properties while also considering the context of these genes in the interaction network.

Figure 1 shows a high-level overview of the method. OMEN takes an interaction network as a scaffold to drive its analysis. It starts by searching this network for a pattern. Here, the patterns of interest are fixed-length paths (sets of three nodes connected on the network). OMEN weights these paths by traversing the interaction network using a random walk semantics informed by an objective function. This objective function combines the mutational frequency of the genes in the pattern with the degree to which the gene sets are mutually exclusive. Mutual exclusivity is expressed in terms of the functional impact score of the occurring mutations. The resulting weighted patterns reflect the mutational burden and the mutual exclusivity of their involved genes while also capturing their local network context. The random walk semantics used to determine the weights implicitly accounts for the network topology and is unique among random walk approaches in its inclusion of a sink node. This node allows OMEN to avoid giving high weights to sparsely connected nodes that display a low driver potential.

The resulting pattern set is aggregated into a probabilistic network where both the edges and nodes are weighted. These weights reflect the contribution of each respective element in the aggregated pattern. Node and edge probabilities in this aggregated network are subsequently used to prioritize driver genes and infer modules of mutually exclusive driver genes, referred to as driver pathways.

Driver pathways are driver genes connected with high probability edges and for which the mutations show high intra-set mutual exclusivity, and a lesser inter-set mutual exclusivity. An in-depth description of the methodology can be found in Section 3.

2.2 Pan-cancer performance against benchmark

To test and assess our method with respect to state-of-the-art driver identification methods, we used the evaluation framework from (Tokheim *et al.*, 2016) based on a comprehensive pan-cancer dataset derived from TCGA. We adopted the performance and consistency criteria based on the definition of a *true positive* driver mutation and the concept of *TopDrop consistency* as described in the original article. Using the same standardized assessment criteria and data allowed us to compare our method to the driver identification methods already assessed by Tokheim et al. which included ActiveDriver (Reimand and Bader, 2013), OncodriveFM (Gonzalez-Perez and

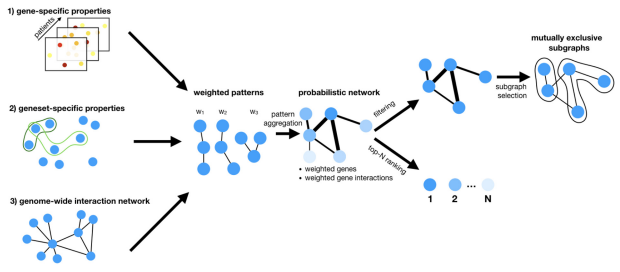

Fig. 1. High-level overview of OMEN. Inputs can be gene-specific properties and gene-set properties such as mutual exclusivity. The genome-wide interaction network acts as a scaffold to drive the analysis

Lopez-Bigas, 2012), OncodriveFML (Mularoni *et al.*, 2016), OncodriveClust (Tamborero *et al.*, 2013), MuSiC (Dees *et al.*, 2012), TUSON (Davoli *et al.*, 2013), MutSigCV (Lawrence *et al.*, 2013) and 20/20+ (Tokheim *et al.*, 2016). Since few of these methods rely on mutual exclusivity or network knowledge as a driving force to prioritize cancer driver mutations, we complemented the assessment of Tokheim et al. with Mutex (Babur *et al.*, 2015), SSA-ME (Pulido-Tamayo *et al.*, 2016) and MEMo (Ciriello *et al.*, 2012).

These methods are highly heterogeneous. They can be contrasted in their approach to learning from data (supervised versus unsupervised), in the type of information they use to prioritize drivers (using only gene-centric properties such as mutational burden or functional impact, exclusively using gene set properties such as mutual exclusivity or a combination of both), and in whether or not they use an interaction network to drive the analysis. In the latter case, we distinguish between methods that employ a pre-computed property of the network as a feature in their analysis (static network methods) and those that rely on a graph search to integrate gene-centric or gene-set driver properties on an interaction network. These latter methods are classified as network-based methods and rely on propagation or pattern finding.

2.2.1 Precision–recall against cancer gene census

As a first measure of performance we assessed to what extent the different methods could recover known driver mutations documented in the Cancer Gene Census. Results of this analysis are summarized in the precision–recall plot shown in Figure 2.

We find that methods that employ more information generally yield a better score. Methods that exclusively rely on one specific data property such as gene-centric driver score or mutual exclusivity perform lower in general, and when their results are used as input for more integrative methods, either during data prefiltering or directly as features in a predictive model the performance increases. 2020+, TUSON, OMEN and MEMo stand out as performing particularly well.

Since OMEN combines both gene-centric and gene-set properties on an interaction network, we contrast its performance with its conceptually most related methods. These are SSA-ME and MEMo. Where SSA-ME also combines gene-centric with gene-set properties, MEMo indirectly accounts for mutational burden and functional impact by prefiltering the genes with high driver potential based on their gene-centric properties. OMEN outperforms both. Its performance in comparison to MEMo suggests merit in the combined analysis of network, gene-centric and gene-set specific properties as opposed to relying on a sequential approach that prefilters before identifying mutually exclusive subnetworks.

The moderate performance displayed by SSA-ME can be explained by a significant bias toward hubs in its identification of mutually exclusive subnetworks. Spurious genes are pulled in by hubs such as TP53 and can receive very high rankings, causing poor precision even at low recall.

The supervised methods 2020+ and TUSON clearly outperform unsupervised methods, but as they rely on known drivers to train their model they are biased toward existing driver knowledge and the possibility of overfitting cannot be excluded as the benchmark set is not completely independent from the training set.

Interestingly, when considering the top performing methods, it seems that prior information in the form of connectedness on an interaction network as used by OMEN and MEMo can compensate for the lack of prior information in the form of a training set. TUSON does not use any network properties whereas 2020+ relies on precalculated network features but does not use connectivity on an interaction network to drive the analysis.

2.2.2 Overlap in predicted driver genes

Figure 3 shows a pairwise comparison of the number of overlapping genes across all investigated methods. Methods with significant conceptual similarity such as OMEN, SSA-ME and MEMo have a significant overlap in the identified driver genes. ActiveDriver and MuSiC however, stand out for sharing few of their top-ranked genes with other methods. This overlap is visualized in the Supplementary Materials Illustration S21 and Supplementary Table S1.

2.2.3 Topdrop consistency

A robust method prioritizes the same genes, irrespective of which fraction of the dataset it is applied to, provided each fraction contains the same signal. Observing highly inconsistent outcomes serves as a strong indicator that the prioritized drivers are spurious. This robustness can be assessed using TopDrop consistency. Tokheim *et al.* (2016) defines it as $|A^{1:d} \cap B^{1:2d}|/d$, where d is the number of ranked genes of interest, and A and B are lists of ranked genes extracted from different splits of the dataset. This corresponds to the percentage of top d ranked genes in A that are also in the top $2d$ of B. Our TopDrop consistency results were assessed by running each method on 10 random splits of the patient data, maintaining the original tumor type distribution within each split. Figure 4 shows the TopDrop consistency as a function of the number of selected genes d.

TopDrop consistency is expected to decrease as the number of selected genes is increased. Genes with a low signal in the dataset are expected to be ranked lower and their detection would be more prone to the exact subset of data that was selected. For most methods this expected decrease in consistency was observed except for Mutex, MEMo and ActiveDriver. This can be explained by the fact that these methods only focus on prioritizing genes that have an extremely high driver potential (strict prefiltering based on functional impact and/or mutational frequency) prior to the analysis. This prefiltering prevents finding rare mutations of which the detection is more sensitive to the selected subset of the data. SSA-ME appears very robust, with near-perfect consistency, but digging into results showed that irrespective of the subset of data used, the same drivers

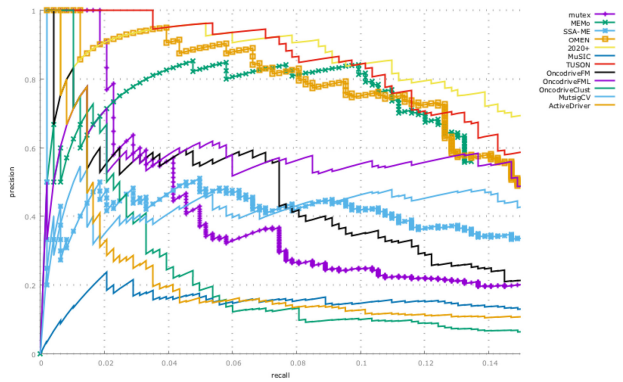

Fig. 2. Precision–recall plot of the top-80 ranked genes returned by OMEN and competing driver prioritization algorithms, applied on the benchmark pan-cancer dataset described in Tokheim *et al.* (2016)

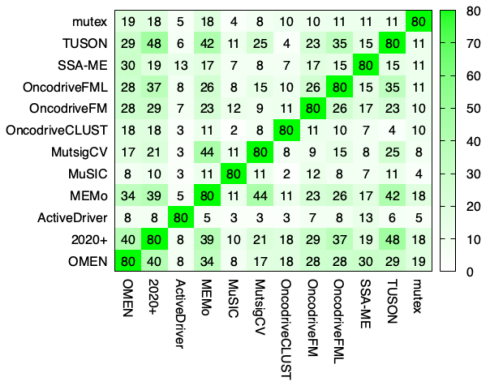

Fig. 3. Heat map showing the number of overlapping genes within the 80 most likely ranked driver genes on the benchmark pan-cancer dataset described in Tokheim *et al.* (2016)

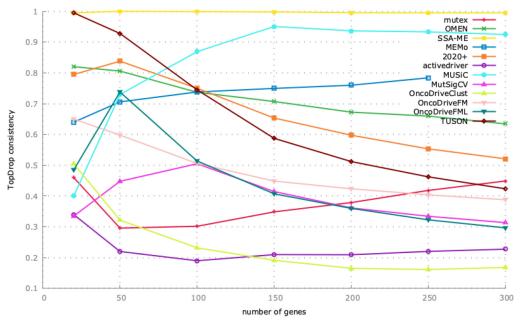

Fig. 4. TopDrop consistency in function of the number of prioritized genes considered for multiple driver gene/patterns of mutual exclusivity prioritization algorithms applied to the pan-cancer dataset presented in Tokheim *et al.* (2016) and Dees *et al.* (2012)

were always prioritized, many of which were neighbors of hubs, the principal one being TP53. This shows that despite its focus on mutual exclusivity, SSA-ME does not compensate sufficiently for the trade-off between exclusivity and mutational coverage and hence is at risk of identifying spurious patterns of mutual exclusivity that include rarely mutated genes, especially if these are located in the neighborhood of a hub such as TP53. For larger selections of genes, MuSiC is the most consistent of the remaining methods with MEMo and OMEN as runner-ups.

OMEN delivers highly consistent results competitive with MEMo and 2020+, and it also displays the expected decrease in TopDrop consistency as the number of genes are increased.

2.3 Biological relevance of pan-cancer results

Only 6 of the top-40 genes prioritized by OMEN as shown in Table 1 were not present in CGC. This suggests that OMEN has an acceptable precision. Of those six novel predictions TAF1, TCF4, CSNK2A1 and YWHAZ were also prioritized as top-ranked driver candidates by other methods.

Since OMEN and SSA-ME are the only network-based methods that focus on mutual exclusivity, the shared genes in their top 80 that are not prioritized by other methods might be characteristic of these features. Of the 30 genes they share in their top 80, 8 do not appear in the top 80 of the other methods: GATA2, GRB2, JUN, MYC, MYH9, NEB, RAD21 and TAF1. Five of these are in CGC tier 1: GATA2, JUN, MYC, MYH9 and RAD21.

An asset of OMEN is its ability to comprehensively summarize the prioritized driver genes in mutually exclusive driver subnetworks. Driver subnetworks are identified by searching the interaction network for connected sets of prioritized driver genes that display stronger patterns of mutual exclusivity amongst each other than with genes belonging to other subnetworks. The 80 top genes prioritized by OMEN on the Tokheim pan-cancer dataset could be subdivided in 10 different driver subnetworks (Supplementary Note S2). Overall the modules recapitulate important cancer hallmarks: DNA damage response and apoptosis, small GTPase mediated signaling, Wnt/β-catenin pathway, oxidative stress, EGF signaling, MAPK pathways and chromatin remodeling.

Genes can belong to the same mutually exclusive driver subnetwork because they belong to the same signaling pathway and hence a mutation in an additional component of the same signaling pathway does not confer any addition fitness to the cancer cells (van de Haar *et al.*, 2019). This is probably the case of genes that belong to the same protein complex (e.g. the Polycomb complex, SUZ12 and EZH2 in Supplementary Illustration S2 and S12). However, it might also be that the added value of a mutation in one gene depends on the absence of another mutation. The mutual exclusivity between mutations in KRAS and EEF1A1 could exemplify such a case (Supplementary Illustrations S4 and S14). KRAS-Driven cancers are dependent on METTL13-Mediated EEF1A Methylation. Hence in tumor cells that carry a mutation in EF1A further acquisition of KRAS mutations might not confer any additional benefit to the cell (Liu *et al.*, 2019). Lastly, Supplementary Illustration S1 and S11 illustrate how genes giving rise to pleiotropic effects are mutually exclusive. Aberrations in genes of this subnetwork all interfere with epigenetic regulation and hence gene expression. Mutations in these genes tend to be mutually exclusive, indicating that their net contribution to driving cancer is similar and independent. Affecting multiple mechanisms to interfere with chromatin remodeling seems to also not confer any benefit to the cells or is, given their pleiotropic effects even detrimental. The fact that these genes of the same pathway tend to be mutated in different cancer types further supports their negative epistatic interaction (van de Haar *et al.*, 2019).

3 Algorithm

3.1 Overview

OMEN performs three key steps: (i) collecting patterns, (ii) generating a probabilistic network from those patterns and (iii) prioritizing driver genes and/or pathways.

During the pattern collection phase, OMEN explores an interaction network $G_i = (V_i, E_i)$ where V_i are a set of genes, and E_i the undirected interactions between pairs of genes. OMEN retrieves patterns that satisfy a given pattern definition and provides each with a score. The pattern definition in OMEN is expressed in a (logic) programming language and specifies the criteria an individual pattern should meet to be accepted. It also encodes the objective function that scores each accepted pattern. Here, the patterns of interest are paths through the interaction network that connect three consecutive genes. The objective function assesses the average mutational burden and mutual exclusivity of the genes in the pattern. Its assigned score proves useful to identify parts of driver pathways. The minimal score required for a pattern to be considered interesting is determined through a permutation test. The patterns that meet the threshold are considered valid, and are used in the construction of a probabilistic network.

Figure 5 shows an overview of probabilistic network generation. OMEN aggregates all valid patterns and uses them to construct a probabilistic network. That is, a probabilistic variant of the interaction network. The probability of a node or edge in this aggregated network is the sum of all the path weights in which the respective node or edge occurs. Those probabilities capture the importance of each node and edge in the network.

Once a probabilistic network has been generated, the availability of node and edge probabilities puts the final goal of driver prioritization and pathway identification within reach.

Section 3.2 describes the objective function, Section 3.3 describes how OMEN generates a probabilistic network and Section 3.4 covers how the probabilistic network is used to prioritize driver genes and retrieve mutually exclusive modules.

3.2 Objective function

The objective function assigns a score to a gene set G that quantifies its driver potential. The score is a value in the [0,1] range and can thus also be interpreted as a probability. It combines two contributions (i) the frequency with which a gene in G is mutated across the cohort and (ii) the mutual exclusively of G. Each contribution is modeled by a specific term, and the objective function f is their convex combination:

$$f(G) = \alpha \cdot \text{mut\_ex}(G) + (1-\alpha) \cdot \text{gene\_freq}(G) \text{ where } 0 \leq \alpha \leq 1 \quad (1)$$

The trade-off between these terms is explicitly captured in the parameter α. This offers the user control when searching for relevant, mutually exclusive gene sets. A higher value for α results in a higher weighting of the mutual exclusivity term. In our experiments we employed $\alpha = 0.8$. We place an emphasis on the mutual exclusivity term, since it is a very rich term that integrates functional impact data. In addition, a too strong focus on coverage would hamper our search for rare driver genes. The terms of the objective function:

Table 1. Top 40 genes prioritized by OMEN on the Tokheim pan-cancer dataset annotated with respective Cosmic CGC tier

Rank	Gene	CGC	Rank	Gene	CGC	Rank	Gene	CGC	Rank	Gene	CGC
1	CTCF	T1	11	CTNNB1	T1	21	YY1	No	31	PIK3CA	T1
2	TP53	T1	12	SUZ12	T1	22	KRAS	T1	32	CSNK2A1	No
3	EP300	T1	13	SMAD4	T1	23	GATA3	T1	33	FBXW7	T1
4	PIK3R1	T1	14	HSP90AB1	T1	24	TCF4	No	34	MYH9	T1
5	VHL	T1	15	RAD21	T1	25	MAP3K1	T1	35	BRAF	T1
6	EGFR	T1	16	RB1	T1	26	AKT1	T1	36	ERBB2	T1
7	TAF1	No	17	STK11	T1	27	MTOR	T1	37	LRP1	No
8	APC	T1	18	SMAD2	T1	28	IDH1	T1	38	DDX3X	T1
9	SMARCA4	T1	19	ATM	T1	29	PTEN	T1	39	SMAD3	T1
10	TCF12	T1	20	CDKN2A	T1	30	KEAP1	T1	40	YWHAZ	No

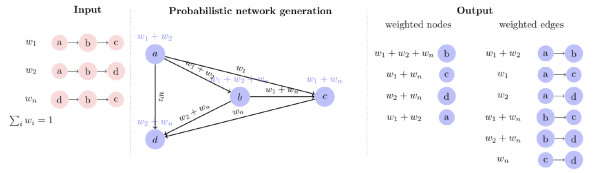

Fig. 5. The genes and interactions in the probabilistic network correspond to the set of genes and interactions used by the input paths. Gene a has probability $w_1 + w_2$ since a occurs in two paths with respectively probability w_1 and w_2. The interaction between gene a and c has probability w_1 because it is only present in the path with probability w_1

mutual exclusivity and *gene specific mutational burden* are defined as follows.

3.2.1 Mutual exclusivity term

$$\text{mut\_ex}(G) = \frac{1}{\#\text{Pts}_G} \cdot \sum_{p \in \text{Pts}_G} \sum_{g \in G} m_h(g,p) \cdot \prod_{h \in G/g} (1 - m_h(h,p)) \quad (2)$$
$$\text{where } Pts_G = \{p | \exists g \in G, \exists p \in Patients : m_h(g,p) > 0\}$$

The CADD probability m_h captures the likelihood that mutations associated with a particular gene and patient display a functional impact. It is computed by first generating PHRED-like CADD scores using the CADD algorithm (Kircher *et al.*, 2014) and subsequently mapping these to the (0, 1)-range [Given a PHRED-like CADD score x, we applied the function $(1 + e^{\frac{15-x}{2}})^{-1}$, where 15 is an inflation value, and 2 is a damping scalar]. If no CADD score is associated with a particular gene and patient, it defaults to 0. The mutual exclusivity score for a patient p and gene set G is the probability that exactly one gene in the gene set displays a functional impact. The possible world semantics (Kimmig and De Raedt, 2017) is used to compute the likelihood that the functionally impactful mutations are mutually exclusive. Mutual exclusivity with respect to an entire cohort is determined by taking the average.

A set of genes achieves a perfect mutual exclusivity score of 1 only when all the patients in the cohort who experience at least one mutated gene within the given selection G have exactly one gene mutated in that set with maximal deleteriousness. That is, a CADD probability m_h of 1. An example is shown in Supplementary Table S2.

3.2.2 Gene specific mutational burden term

$$\text{gene\_freq}(G) = \frac{1}{\#G} \cdot \sum_{g \in G} m_a(g) \quad (3)$$

Mutational recurrence across a cohort is an indication of positive selection. Genes that are recurrently mutated and hence have a high mutational burden tend to be likely drivers. This property is expressed by the *gene specific mutational burden* term which gives a higher score to gene sets consisting of recurrently mutated genes. It is the average of the mutational burden of the genes in the gene set. The mutational burden of a gene can be assessed by the frequency with which the mutation occurs in the cohort or by a mutational burden score derived from e.g. MutSigCV. In our study we considered both approaches and found no notable differences. In this study m_a stands for the percentile rank of mutational frequency mapped to the range [0,1]. Thus a gene having an m_a of 0.8 implies that 80% of the genes have fewer observed mutations.

3.3 Probabilistic network generation

In order to identify the driver pathways contained in the interaction network, all possible subgraphs would ideally be scored by the objective function. Since this task is computationally infeasible, an approximation is made. First, the interaction network is traversed to retrieve and score a limited set of paths. Next, these paths are aggregated into a probabilistic network capturing the expected importance a node or edge plays in a driver pathway. Finally, either gene driver prioritization is performed by retrieving the highest-scoring nodes, or driver pathways are identified by applying a constrained graph clustering method.

3.3.1 Collecting and weighting patterns

OMEN first collects patterns. Here, these are fixed-length, cycle-free paths over the interaction network. A path is a sequence of nodes $\mathbf{x}$ such that $(x_i, x_{i+1}) \in E_i$. Each path is required to have a length of three nodes, providing a decent trade-off between computational load and biological relevance, since a path consisting of three nodes is sufficient to allow the inclusion of linker genes, while remaining short enough to be collected in reasonable time. In addition, the pattern also requires that the objective function score of its constituent genes exceeds a given threshold. This threshold is determined through a permutation test, where pattern collection without a threshold is performed on data where patient-mutation associations have been permuted. Due to permuting the data, the patterns are no longer expected to carry a signal. As such, the threshold on the objective function is chosen to reject up to 95% of these patterns. Here, this corresponds to an objective function score > 0.786.

Upon collecting a pattern, OMEN assigns it a weight. The weight captures the quality of the pattern as well as its network context. Naively imposing a uniform weight across all patterns, would fail to do either, while choosing a weight proportional to the objective function score of the pattern would not account for the network context. Instead, OMEN uses a weighted random walk semantics informed by the objective function f to define a distribution over all possible fixed-length acyclic paths in the network. The weight of a pattern then corresponds to its probability under this semantics. A key component here is the transition probability P_t which specifies the probability of transitioning from a given node x_i to one of its neighbors x_j. Here nb is a function that given a node returns the set of its neighbors: $nb(a) = \{b | (a,b) \in E_i \vee (b,a) \in E_i\}$.

$$P_t(\mathbf{x}, i, j) = \underbrace{\frac{f(x_1, \ldots, x_i, x_j)}{\sum_{k \in nb(x_i)} f(x_1, \ldots, x_i, k)}}_{\text{(a)}} \cdot \underbrace{(1 - \prod_{k \in nb(x_i)} (1 - f(x_1, \ldots, x_i, k)))}_{\text{(b)}} \tag{4}$$

Notably, this transition probability cannot be captured using Markov chains, as it is dependent on the complete history of the walk. $(x_1, \ldots, x_j)$. Furthermore, the random walk semantics helps to take into account the local network context. Term (a) in Equation (4) captures the relative attractiveness of moving from x_i to x_j contrasted against all neighbors of x_i. Finally, OMEN introduces a probability sink. It is a single node that all nodes have some probability of transitioning into (see Fig. 6). The probability that a node x_i in pattern $\mathbf{x}$ transitions to a sink is:

$$P_t(\mathbf{x}, i, sink) = \prod_{k \in nb(x_i)} (1 - f(x_1, \ldots, x_i, k)) \tag{5}$$

It is because of this additional flow of probability that term (b) was added to Equation (4). It discounts the other transition probabilities, ensuring that all probabilities leaving any node add up to 1. The purpose of this term is to compensate for a key pitfall when considering the probabilities of all unmodified (weighted) random walks. Term (a) in Equation (4) is only sensitive to the local context. When presented with e.g. a number of equally attractive alternatives, it will return the same probability, regardless of whether those alternatives have a high or low objective function score. Term (b) in Equation (4) behaves as a noisy-OR construct. Its effect is that nodes with less attractive neighbors will transition toward the sink node with a higher probability than nodes whose neighbors hold more promise, thus reducing the total amount of probability with which such a node would be able to transition to one of its neighbors. Consider for example how in Figure 6 node *a* can only distribute a probability mass of 0.72 to its neighbors, while node *b* can distribute 0.97 as a direct result of the difference in aggregate objective function score between their neighbors.

Having defined P_t, we can now define the probability of a pattern. Given $\mathbf{x}$ the pattern along the path $(x_1, \ldots, x_n)$ and m the cardinality of the set of genes in which mutations have been observed, the probability of $\mathbf{x}$ is:

$$P(\mathbf{x}) = \begin{cases} 0 & \neg\exists p : m_b(x_1, p) > 0, \\ \frac{1}{m}\left(\prod P_t(\mathbf{x}, i, i+1), \forall i : (x_i, x_{i+1}) \in E_i\right) & otherwise \end{cases} \tag{6}$$

If no mutations were observed in the first gene in the path, the probability of the pattern is 0. Otherwise, the probability is $1/m$ (the uniform distribution over all genes for which at least 1 mutation was recorded) multiplied by the product of all its transition probabilities.

OMEN encodes this distribution as a stochastic logic program (Cussens, 2013; Van Daele *et al.*, 2015). Each gene is represented as a single stochastic predicate, and each transition is represented as a clause whose probability is computed dynamically as the pattern is constructed.

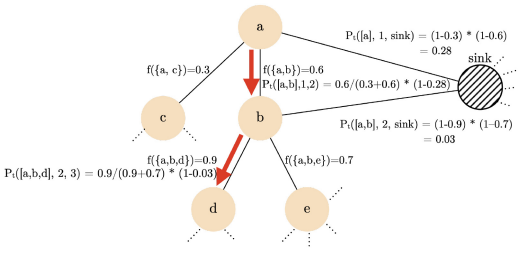

Fig. 6. A toy example of the transition probability P_t for the path (a, b, d) (indicated with arrows) on an interaction network extended with a sink node

The weighted random walk semantics mitigates the effect of hubs in the network. Since a distribution is imposed over all the neighbors of a node, the probability of a path transitioning out of a hub node is likely to be relatively low. Thus a pattern that goes through a hub is not likely to be explored because it is so improbable, and if it does get collected as a valid pattern, its weight is likely to be low due to the effect of the low probability transition out of the hub node. The effect of the sink node on the final probability of the pattern is also notable as it penalizes paths that visit regions of the network that display limited potential.

Upon having collected and weighted these patterns, it is not yet possible to straightforwardly apply them to prioritize driver genes and identify relevant modules. These tasks are facilitated by aggregating the patterns in a probabilistic network.

3.3.2 Deriving a probabilistic network through pattern aggregation

P defines a deficient probability distribution over the set of collected patterns *xs*. That is, a distribution that sums to N where $N < 1$. It fails to sum up to 1 due to the pattern definition rejecting paths that end in a sink node. In addition, some legal paths are also rejected for failing to meet the objective function threshold.

The set of patterns is used to construct the probabilistic network $G_n = (V_n, E_n)$. Given that $\mathbf{x}$ is a pattern drawn from the set of patterns *xs* and given that the $\in$-operator on $\mathbf{x}$ tests membership of the nodes that are contained in $\mathbf{x}$, we have $V_n = \{v | \mathbf{x} \in xs, v \in \mathbf{x}\}$ and $V_n = \{v | \mathbf{x} \in xs, v \in \mathbf{x}\}$, and each node v and edge (m, n) has an associated probability $p(v) = \frac{1}{N}\sum_{\mathbf{x} \in xs, v \in \mathbf{x}} P(\mathbf{x})$ and $P((m,n)) = \frac{1}{N}\sum_{\mathbf{x} \in xs, m \in \mathbf{x}, n \in p} P(\mathbf{x})$. Here, a proper probability distribution is achieved by multiplying each probability with the normalizing constant $1/N$. The node probabilities of the probabilistic network then capture the relative importance of their associated genes, while the edge probabilities also capture how frequently pairs of genes share a pattern.

3.4 Gene ranking and driver pathway identification using the probabilistic network

3.4.1 Ranking genes

The probabilistic network is used to prioritize driver genes and select driver pathways. To prioritize drivers, nodes from the probabilistic network are ranked according to their probabilities in the probabilistic network. A higher probability being an indicator that the node is of higher interest. The top ranked genes are the most likely drivers.

3.4.2 Identifying driver pathways

Driver pathways are non-overlapping groups of interacting genes that display mutual exclusivity. Identifying the driver pathways within the probabilistic network can be regarded as a constrained graph clustering task, where each driver pathway corresponds to a single cluster. In this setting, each gene in the probabilistic network is assigned to a particular driver pathway while optimizing a given clustering criterion.

Tran *et al.* (2021) presents a method for solving this problem. Supplementary Note S1 covers details on its usage within OMEN. Since the proposed algorithm does not automatically infer the most appropriate number of clusters to extract, we qualitatively analyzed the results for 6 to 12 clusters and found that 10 clusters proved most suitable. The result of this clustering is shown in Supplementary Note S2.

3.5 Materials

The genome-wide interaction network used consists of: high-quality metabolic interactions from Recon X and literature curated interactions from Intact. Recon X and Intact data was acquired from Pathway Commons version 8. These interactomics data were combined into a genome-wide interaction network in which the nodes represent genes/gene products and the edges represent possible physical interactions. Duplicate edges were

discarded. The resulting network covers 7901 samples, and contains 15.694 nodes and 178.051 edges.

Computing the probabilistic network used in this article took approximately 5 h using an Intel Xeon Processor E5-2698 v3 CPU and 264 GB RAM (2 GB RAM should be sufficient to replicate the experiment performed in this article).

4 Discussion

Our method is conceptually innovating in two respects: First, the use of random walk semantics informed by an objective function to dynamically weight patterns. Transition probabilities depend on the network topology and are informed by an objective function that combines the gene-centric property *mutational frequency*, and the gene-set derived property *mutual exclusivity*. In addition, unlike existing propagation methods, our random walk semantics introduces sink nodes to prevent weakly connected nodes with low driver potential from dominating the relevant patterns. Second, the objective function employed by OMEN is unique in defining the mutual exclusivity of a set of genes in terms of their mutations' functional impact scores, rather than in terms of binary mutation calls. As a result, OMEN does not require imposing a restrictive, a priori filtering of genes based on the likely functional impact of their mutations, nor does it involve a local search that only considers a restricted set of genes or interactions. Our results show how OMEN is highly competitive with existing methods on a benchmark dataset. The framework is generic, allowing the objective function to be extended to account for the average expression aberrations of the genes in a gene set. Features that resist quantification using a probability are more challenging to integrate and are the subject of possible future work. This includes integrating copy number variation data despite the difficulty in expressing their degree of deleteriousness relative to point mutations. Our network-based method will also not be able to identify mutual exclusivity between gene sets that are unconnected on the graph and therefore might miss negative epistasis between pathways that are completely disconnected or very distant from each other on the graph (van de Haar *et al.*, 2019). By having used a pan cancer setting we obviously found the drivers and driver pathways common to all cancer types (cancer hall marks). To detect more cancer-specific pathways and more rare mutations one could additionally drill down on particular cancer types.

Funding

This work was supported by grants of the Fonds Wetenschappelijk Onderzoek-Vlaanderen (FWO) [G046318, G.0371.06 and 3G045620], project No. HBC.2019.2528, funded by VLAIO (Flanders Innovation & Entrepreneurship), the UGent Bijzonder Onderzoeksfonds, the KU Leuven Bijzonder Onderzoeksfonds and the Flemish Government (AI Research Program).

Conflict of Interest: none declared.

References

Abeshouse,A. *et al.* (2015) The molecular taxonomy of primary prostate cancer. *Cell*, **163**, 1011–1025.

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

Bailey,M.H. *et al.*; Cancer Genome Atlas Research Network. (2018) Comprehensive characterization of cancer driver genes and mutations. *Cell*, **173**, 371–385.

Banerji,S. *et al.* (2012) Sequence analysis of mutations and translocations across breast cancer subtypes. *Nature*, **486**, 405–409.

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

Ciriello,G. *et al.* (2013) Emerging landscape of oncogenic signatures across human cancers. *Nat. Genet.*, **45**, 1127–1133.

Cussens,J. (2013) Stochastic logic programs: Sampling, inference and applications. *arXiv preprint arXiv:1301.3846*. https://doi.org/10.48550/arXiv.1301.3846.

Davoli,T. *et al.* (2013) Cumulative haploinsufficiency and triplosensitivity drive aneuploidy patterns and shape the cancer genome. *Cell*, **155**, 948–962.

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

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

Gao,B. *et al.* (2017) Identification of driver modules in pan-cancer via coordinating coverage and exclusivity. *Oncotarget*, **8**, 36115–36126.

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

Gonzalez-Perez,A. *et al.*; International Cancer Genome Consortium Mutation Pathways and Consequences Subgroup of the Bioinformatics Analyses Working Group. (2013) Computational approaches to identify functional genetic variants in cancer genomes. *Nat. Methods*, **10**, 723–729.

Hofree,M. *et al.* (2013) Network-based stratification of tumor mutations. *Nat. Methods*, **10**, 1108–1115.

Horn,H. *et al.* (2018) NetSig: network-based discovery from cancer genomes. *Nat. Methods*, **15**, 61–66.

International Cancer Genome Consortium *et al.* (2010) International network of cancer genome projects. *Nature*, **464**, 993.

Kim,Y.-A. *et al.* (2015) MEMCover: integrated analysis of mutual exclusivity and functional network reveals dysregulated pathways across multiple cancer types. *Bioinformatics*, **31**, i284–i292.

Kimmig,A. and De Raedt,L. (2017). Probabilistic logic programs: unifying program trace and possible world semantics. In: *Workshop on Probabilistic Programming Semantics*, *2017/01/17–2017/01/17*, Paris, France.

Kircher,M. *et al.* (2014) A general framework for estimating the relative pathogenicity of human genetic variants. *Nat. Genet.*, **46**, 310–315.

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

Le Van,T. *et al.* (2016) Simultaneous discovery of cancer subtypes and subtype features by molecular data integration. *Bioinformatics*, **32**, i445–i454.

Liu,S. *et al.* (2019) METTL13 methylation of eEF1A increases translational output to promote tumorigenesis. *Cell*, **176**, 491–504.

Mularoni,L. *et al.* (2016) OncodriveFML: a general framework to identify coding and non-coding regions with cancer driver mutations. *Genome Biol.*, **17**, 128.

Pulido-Tamayo,S. *et al.* (2016) SSA-ME detection of cancer driver genes using mutual exclusivity by small subnetwork analysis. *Sci. Rep.*, **6**, 36257.

Reimand,J. and Bader,G.D. (2013) Systematic analysis of somatic mutations in phosphorylation signaling predicts novel cancer drivers. *Mol. Syst. Biol.*, **9**, 637.

Tamborero,D. *et al.* (2013) OncodriveCLUST: exploiting the positional clustering of somatic mutations to identify cancer genes. *Bioinformatics*, **29**, 2238–2244.

Tokheim,C.J. *et al.* (2016) Evaluating the evaluation of cancer driver genes. *Proc. Natl. Acad. Sci. USA*, **113**, 14330–14335.

Tran,D.H. *et al.* (2021). Local search for constrained graph clustering in biological networks. *Comput. Operat. Res.*, **132**, 105299.

Van Daele,D. *et al.* (2015) PageRank, ProPPR, and stochastic logic programs. In: Davis,J. and Ramon,J. (eds) *Inductive Logic Programming*. Springer, Cham, Switzerland, pp. 168–180.

van de Haar,J. *et al.* (2019) Identifying epistasis in cancer genomes: a delicate affair. *Cell*, **177**, 1375–1383.

Van den Eynden,J. *et al.* (2015) SomInaClust: detection of cancer genes based on somatic mutation patterns of inactivation and clustering. *BMC Bioinformatics*, **16**, 125.

Weinstein,J.N. *et al.*; Cancer Genome Atlas Research Network. (2013) The cancer genome atlas pan-cancer analysis project. *Nat. Genet.*, **45**, 1113–1120.

Yeang,C.-H. *et al.* (2008) Combinatorial patterns of somatic gene mutations in cancer. *FASEB J.*, **22**, 2605–2622.