IntOGen

Release 3.0

BBGLab

Aug 10, 2023

CONTENTS:

CHAPTER

ONE

DATA COLLECTION

1.1 TCGA

TCGA somatic mutations (mc3.v0.2.8 version) were downloaded from (https://gdc.cancer.gov/about-data/publications/pancanatlas). We then grouped mutations according to their patient's cancer type into 32 different cohorts. Additionally, we kept somatic mutations passing the somatic filtering from TCGA (i.e., column FILTER == "PASS").

1.2 PCAWG

PCAWG somatic mutations were downloaded from the International Cancer Genome Consortium (ICGC) data portal (https://dcc.icgc.org/releases/PCAWG/consensus_snv_indel/). Note that only ICGC samples can be freely downloaded from this site, the TCGA portion of the callsets is controlled data. Instructions on how to obtain them can be found in the same webpage.

1.3 cBioPortal

Somatic mutations from Whole Exome Sequencing (WXS) and Whole Genome Sequencing (WGS) cohorts uploaded in cBioPortal that were not part of any other projects included in the analysis (i.e., TCGA, PCAWG, St. Jude or HARTWIG) were downloaded on 2018/09/01 (http://www.cbioportal.org/datasets). We then created cohorts following the next criteria:

1. Cohorts with a limited number of samples (i.e., lower than 30 samples) associated to cancer types with extensive representation (such as Breast cancer, Prostate cancer or Colorectal adenocarcinoma) across the compendium of cohorts were removed.
2. Samples were uniquely mapped into one single cohort. If the same sample was originally included in two cohorts, we removed the sample from one of them.
3. Samples not sequenced from human cancer biopsies were discarded (cell lines, xenografts, normal tissue, etc.).
4. When patient information was available, only one sample from the same patient was selected. The criteria to prioritize samples from the same patient was: WXS over WGS; untreated over treated, primary over metastasis or relapse and, finally, by alphabetical order. When there is no patient information we assume that all patients have only one sample in the cohort.
5. When sequencing platform information was available, samples from the same study but with different sequencing platforms were further subclassified into WXS and WGS datasets (only if the resulting cohorts fulfilled the requirements herein described; otherwise, the samples were discarded).

6. When variant calling information was available, samples from the same cohort and sequencing type were further classified according to their calling algorithm (VarScan, MuTect, etc.). If the resulting cohorts for each subclass fulfilled the requirements herein described, the samples were included;otherwise, the samples were discarded. When variant calling information was not available we assumed that all the samples went through the same pipeline.
7. When treatment information was available, samples from the same cohort, sequencing type, calling algorithm were further classified according to their treatment status (i.e, treated versus untreated). If the resulting cohorts from the subclassification fulfilled the requirements herein described, the samples were included;otherwise, the samples were discarded. When information was not available we assumed that samples had not been treated.
8. When biopsy information was available, samples from the same cohort, sequencing type, calling algorithm, treatment status were further classified according to their biopsy type (i.e, primary, relapse or metastasis). If the resulting datasets from the subclassification fulfilled the requirements herein described, the samples were included; otherwise, the samples were discarded. When information was not available we assumed that the biopsy type of the sample was primary.

1.4 Hartwig Medical Foundation

Somatic mutations of metastatic WGS from Hartwig Medical Foundation https://www.hartwigmedicalfoundation.nl/en/database/ were downloaded on 2021/10/21 through their platform. Datasets were split according to their primary site. Samples from unknown primary sites (i.e., None, Nan, Unknown, Cup, Na), double primary or aggregating into cohorts of fewer than 5 samples (i.e., Adrenal, Myeloid, Thymus and Eye) were not considered. A total of 25 different cohorts were created.

1.5 ICGC

Somatic mutations from Whole Exome Sequencing (WXS) and Whole Genome Sequencing (WGS) studies uploaded in ICGC Data Portal (https://dcc.icgc.org/repositories) not overlapping with other projects included in the analysis (i.e., TCGA, PCAWG, CBIOP or St. Jude) were downloaded from release 2019 on on 2021/10/21. We then created cohorts following the criteria used for the cBioPortal datasets (cBioPortal).

1.6 St. Jude

Somatic mutations from Whole Exome Sequencing (WXS) and Whole Genome Sequencing (WGS) of Pediatric Cancer Genome Project uploaded in the St. Jude Cloud (https://www.stjude.cloud/data.html) were downloaded on 2018/07/16. Cohorts were created according to their primary site and their biopsy type (i.e., primary, metastasis and relapse). Resulting datasets with fewer than 5 samples were discarded.

1.7 PedcBioPortal

Somatic mutations from Whole Exome Sequencing (WXS) and Whole Genome Sequencing (WGS) studies uploaded in PedcBioPortal that were not part of any other projects included in the analysis (i.e., St. Jude or CBIOP) were downloaded on 2018/10/01 (http://www.pedcbioportal.org/datasets). We then created cohorts following the criteria described in the cBioPortal dataset (cBioPortal).

1.8 TARGET

Somatic SNVs from WXS and WGS of four TARGET studies, Neuroblastoma (NB) and Wilms Tumor (WT), from the TARGET consortium were downloaded on 2019/03/07, Osteosarcoma (OS) and Acute Myeloid Leukemia (AML) we downloaded in 2020/09/17 from the Genomic Data Commons Portal.

1.9 Beat AML

We downloaded unfiltered somatic mutations from samples included in the Beat AML study from the Genomic Data Commons Portal. We next applied the following criteria to create our Beat AML cohort:

1. We focused on somatic single nucleotide variants from VarScan2 using skin as normal control. All samples that did not belong to this class were not further analyzed.
2. Samples from relapses were filtered out.
3. Samples from bone-marrow transplants were discarded.
4. If there were several samples per patient fulfilling the points 1-3, we selected the first in chronological order.

257 independent samples of Beat AML tumors composed our Beat AML cohort.

1.10 CGCI

Somatic mutations from Whole Genome Sequencing (WGS) of the The Cancer Genome Characterization Initiative (CGCI) were downloaded from the Genomic Data Commons (GDC portal) on 2021/05/06.

1.11 CPTAC

Somatic mutations from Whole Exome Sequencing (WXS) of the Clinical Proteomic Tumor Analysis Consortium (CPTAC) were downloaded from the GDC portal on 2021/05/06.

1.12 Literature

We also manually collected publicly available cohorts from the literature. Each cohort was filtered following the same steps than mentioned above for the cBioPortal dataset (see above).

Note: For further information of all datasets used in the latest release of intOGen, please visit https://www.intogen.org/beta/download.

CHAPTER

TWO

PREPROCESSING

Given the heterogeneity of the datasets analyzed in the current release of intOGen (resulting from e.g. differences in the genome aligners, variant calling algorithms, sequencing coverage, sequencing strategy), we implemented a pre-processing strategy aiming at reducing possible biases. Specifically, we conducted the following filtering steps:

1. The pipeline is configured to run using GRCh38 as reference genome. Therefore, for each input dataset the pipeline requires that the reference genome is defined. Datasets using GRCh37 as reference genome were lifted over using PyLiftover (https://pypi.org/project/pyliftover/; version 0.4) to GRCh38. Mutations failing to liftover from GRCh37 to GRCh38 were discarded.
2. We removed mutations with equal alternate and reference alleles, duplicated mutations within the sample sample, mutations with 'N' as reference or alternative allele, mutations with a reference allele not matching its reference genome and mutations within non-canonical chromosomes (i.e., mutations outside chr1 to chr22, chrX and chrY).
3. Additionally, we removed mutations with low pileup mappability, i.e. mutations in regions that could potentially map elsewhere in the genome. For each position of the genome we computed the pileup mappability, defined as the average uniqueness of all the possible reads of 100bp overlapping a position and allowing up to 2 mismatches. This value is equal to 1 if all the reads overlapping a mutation are uniquely mappable while it is close to 0 if most mapping reads can map elsewhere in the genome. Positions with a pileup mappability lower than 0.9 were removed from further analyses.
4. We filtered out multiple samples from the same donor. The analysis of positive selection in tumors requires that each sample in a cohort is independent from the other samples. That implies that if the input dataset includes multiple samples from the same patient –resulting from different biopsy sites, time points or sequencing strategies– the pipeline automatically selects the first according to its alphabetical order. Therefore, all mutations in the discarded samples are not considered anymore.
5. We also filtered out hypermutated samples. Samples carrying more than 1000 mutations for WXS and 10000 for WGS and a mutation count greater than 1.5 times the interquartile range length above the third quartile in their respective dataset were considered hypermutated and therefore removed from further analyses.
6. Datasets with filtered synonymous variants are not runnable. Most cancer driver identification methods need synonymous variants to fit a background mutation model. Therefore, datasets with less than 5 synonymous and datasets with a missense/synonymous ratio greater than 10 were excluded .
7. When the Variant Effect Predictor[1] (VEP) mapped one mutation into multiple transcripts associated with HUGO symbols, we selected the canonical transcript of the first HUGO symbol in alphabetical order.
8. We also discarded mutations mapping into genes without canonical transcript in VEP.101.

[1] McLaren W, Gil L, Hunt SE, Riat HS, Ritchie GR, Thormann A, Flicek P, Cunningham F. The Ensembl Variant Effect Predictor. Genome Biology Jun 6;17(1):122. (2016) doi:10.1186/s13059-016-0974-4

CHAPTER

THREE

METHODS FOR CANCER DRIVER GENE IDENTIFICATION

The current version of the intOGen pipeline uses seven cancer driver identification methods (hereinafter DIMs) to identify cancer driver genes from somatic point mutations: dNdScv and cBaSE, which test for mutation count bias in genes while correcting for regional genomic covariates, mutational processes and coding consequence type; OncodriveCLUSTL, which tests for significant clustering of mutations in the protein sequence; smRegions, which tests for enrichment of mutations in protein functional domains; HotMAPS, which tests for significant clustering of mutations in the 3D protein structure; and OncodriveFML, which tests for functional impact bias of the observed mutations. Next, we briefly describe the rationale and the configuration used to run each DIM.

3.1 dNdScv

dNdScv[1] asserts gene-specific positive selection by inferring the ratio of non-synonymous to synonymous substitutions (dN/dS) in the coding region of each gene. The method resorts to a Poisson-based hierarchical count model that can correct for: i) the mutational processes operative in the cohort determined by the mutational profile of single-nucleotide substitutions with its flanking nucleotides; ii) the regional variability of the background mutation rate explained by histone modifications – it incorporates information about 10 histone marks from 69 cell lines obtained in ENCODE project[2]; iii) the abundance of sites per coding consequence type across in the coding region of each gene.

We downloaded (release date 2023/02/24) and built a new reference database based on the list of canonical transcripts defined by VEP.101 (GRCh38). We then used this reference database to run dNdScv on all datasets of somatic mutations using the default setting of the method.

3.2 OncodriveFML

OncodriveFML[3] is a tool that aims to detect genes under positive selection by analysing the functional impact bias of the observed somatic mutations. Briefly, OncodriveFML consists of three steps: in the first step, it computes the average Functional Impact (FI) score (in our pipeline we used CADD v1.6) of coding somatic mutations observed in gene of interest across a cohort of tumor samples. In the next step, sets of mutations of the same size as the number of mutations observed in the gene of interest are randomly sampled following trinucleotide context conditional probabilities consistent with the relative frequencies of the mutational profile of the cohort. This sampling is repeated N times (with N = 10^6 in our configuration) to generate expected average scores across all genes of interest. Finally, it compares the observed average FI score with the expected from the simulations in the form of an empirical p-value. The p-values are then adjusted with a multiple testing correction using the Benjamini–Hochberg (FDR).

[1] Martincorena, I. et al. Universal Patterns of Selection in Cancer and Somatic Tissues. Cell 171, 1029-1041.e21 (2017). doi: 10.1016/j.cell.2017.09.042

[2] Roadmap Epigenomics Consortium. Integrative analysis of 111 reference human epigenomes. Nature volume 518, pages 317–330 (19 February 2015). doi: 10.1038/nature14248

[3] Loris Mularoni, et al. OncodriveFML: a general framework to identify coding and non-coding regions with cancer driver mutations . Genome Biology (2016)

3.3 OncodriveCLUSTL

OncodriveCLUSTL is a sequence-based clustering algorithm to detect significant linear clustering bias of the observed somatic mutations[4]. Briefly, OncodriveCLUSTL first maps somatic single nucleotide variants (SNVs) observed in a cohort to the genomic element under study. After smoothing the mutation count per position along its genomic sequence using a Tukey kernel-based density function, clusters are identified and scored taking into account the number and distribution of mutations observed. A score for each genomic element is obtained by adding up the scores of its clusters. To estimate the significance of the observed clustering signals, mutations are locally randomized using tri- or penta-nucleotide context conditional probabilities consistent with the relative frequencies of the mutational profile of the cohort.

For this analysis, OncodriveCLUSTL version 1.1.3 was run for the set of defined canonical transcripts bearing 2 or more SNVs mapping the mutations file. Tuckey-based smoothing was conducted with 11bp windows. The different consecutive coding sequences contained on each transcript were concatenated to allow the algorithm to detect clusters of 2 or more SNVs expanding two exons in a transcript. Simulations were carried out using pre-computed mutational profiles. All cohorts were run using tri-nucleotide context SNVs profiles except for cutaneous melanomas, where penta-nucleotide profiles were calculated. Default randomization windows of 31bp were not allowed to expand beyond the coding sequence boundaries (e.g., windows overlapping part of an exon and an intron were shifted to fit inside the exon). A total number of N = 1,000 simulations per transcript were performed.

3.4 cBaSE

cBaSE[5] asserts gene-specific positive and negative selection by measuring mutation count bias with Poisson-based hierarchical models. The method allows six different models based on distinct prior alternatives for the distribution of the regional mutation rate. As in the case of dNdScv, the method allows for correction by i) the mutational processes operative in the tumor, with either tri- or penta- nucleotide context; ii) the site count per consequence type per gene; iii) regional variability of the neutral mutation rate.

We run a modified version of the cBaSE script to fit the specific needs of our pipeline. The main modification was adding a rule to automatically select a regional mutation rate prior distribution. Based on the total mutation burden in the dataset, the method runs either an inverse-gamma (mutation count < 12,000), an exponential-inverse-gamma mixture (12,000 < mutation count < 65,000) or a gamma-inverse-gamma mixture (mutation count > 65,000) as mutation rate prior distributions – after communication with Donate Weghorn, cBaSE's first author). We also skip the negative selection analysis part, as it is not needed for downstream analyses.

3.5 Mutpanning

Mutpanning[9] resorts to a mixture signal of positive selection based on two components: i) the mutational recurrence realized as a Poisson-based count model reminiscent to the models implemented at dNdScv or cBaSE; ii) a measure of deviance from the characteristic tri-nucleotide contexts observed in neutral mutagenesis; specifically, an account of the likelihood that a prescribed count of non-synonymous mutations occur in their observed given a context-dependent mutational likelihood attributable to the neutral mutagenesis.

[4] Claudia Arnedo-Pac, et al. OncodriveCLUSTL: a sequence-based clustering method to identify cancer drivers. 2019 Jun 22. Bioinformatics. pii: btz501. doi: 10.1093/bioinformatics/btz501 .

[5] Weghorn, et al. D. & Sunyaev, S. Bayesian inference of negative and positive selection in human cancers. Nature Genetics 49, 1785–1788 (2017). doi: 10.1038/ng.3987

[9] Dietlein, F., Weghorn, D., Taylor-Weiner, A. et al. Identification of cancer driver genes based on nucleotide context. Nat Genet (2020). https://doi.org/10.1038/s41588-019-0572-y

3.6 HotMaps3D

HotMAPS[6] asserts gene-specific positive selection by measuring the spatial clustering of mutations in the 3D structure of the protein. The original HotMAPS method assumes that all amino-acid substitutions in a protein structure are equally likely. We employed HotMAPS-1.1.3 and modified it to incorporate a background model that more accurately represents the mutational processes operative in the cohort.

We implemented a modified version of the method where the trinucleotide context probability of mutation is compatible with the mutational processes operative in the cohort. Briefly, for each analyzed protein structure harbouring missense mutations, the same number of simulated mutations were randomly generated within the protein structure considering the precomputed mutation frequencies per tri-nucleotide in the cohort. This randomization was performed N times (N = 10^5 in our configuration) thereby leading to a background with which to compare the observed mutational data. The rest of HotMAPS algorithm was not modified.

We downloaded the pre-computed mapping of GRCh37 coordinates into structure residues from the Protein Data Bank (PDB) (http://karchinlab.org/data/HotMAPS/mupit_modbase.sql.gz). We also downloaded (on 2019/09/20) all protein structures from the PDB alongside all human protein 3D models from Modeller (ftp://salilab.org/databases/modbase/projects/genomes/H_sapiens/2013/H_sapiens_2013.tar.xz). and (ftp://salilab.org/databases/modbase/projects/genomes/H_sapiens/2013/ModBase_H_sapiens_2013_refseq.tar.xz). We then annotated the structures following the steps described in HotMAPS tutorial (https://github.com/KarchinLab/HotMAPS/wiki/Tutorial-(Exome-scale)).

Since HotMAPS configuration files are pre-built in GRCh37 coordinates and our pipeline is designed to run using GRCh38, for each input cohort, we first converted input somatic mutations to GRCh37, executed the HotMAPS algorithm and transformed the output to coordinates to GRCh38. All conversions were done using the PyLiftover tool.

3.7 smRegions

smRegions[7] is a method developed to detect linear enrichment of somatic mutations in user-defined regions of interest. Briefly, smRegions first counts the number of non-synonymous mutations overlapping with a Pfam domain in a particular protein. Next, these non-synonymous variants are randomized N times (N = 1,000 in our configuration) along the nucleotide sequence of the gene, following the trinucleotide context probability derived from precomputed mutation frequencies per tri-nucleotide in the cohort. The observed and average number of simulated mutations in the Pfam domain and outside of it are compared using a G-test of goodness-of-fit, from which the smRegions p-value is derived. We discarded those domains with a number of observed mutations lower than the average from the randomizations. The p-values were adjusted with a multiple testing correction using the Benjamini–Hochberg procedure. Therefore, we confined the analysis to Pfam domains with a number of observed mutations higher or equal than the mean simulated number of mutations in the re-sampling.

To create the database of genomic coordinates of Pfam domains we followed the next steps: i) we gathered the first and last amino acid positions of all Pfam domains for canonical transcripts (VEP.101) from BioMart; ii) for each Pfam domain we mapped the first and last amino acid positions into genomic coordinates using TransVar –using GRCh38 as reference genome–; iii) we discarded Pfam domains failing to map either the first or last amino acid positions into genomic coordinates.

smRegions was conceptually inspired by e-driver[8], although significant enhancements were introduced. Particularly, i) our background model accounts for the observed tri-nucleotide frequencies rather than assuming that all mutations are equally likely; ii) the statistical test is more conservative; iii) Pfam domains are part of the required input and can

[6] Tokheim C, et al. Exome-scale discovery of hotspot mutation regions in human cancer using 3D protein structure. Cancer research. 2016a;76:3719–3731. doi: 10.1158/0008-5472.CAN-15-3190

[7] Francisco Martínez-Jiménez, et al. Disruption of ubiquitin mediated proteolysis is a widespread mechanism of tumorigenesis. bioRxiv 2019. doi: https://doi.org/10.1101/507764

[8] Porta-Pardo E, et al. e-Driver: a novel method to identify protein regions driving cancer. Bioinformatics. 2014;30(21):3109–3114. doi:10.1093/bioinformatics/btu499

be easily updated by downloading the last Pfam release iv) the method can be configured to any other setting that aims to detect genes possibility selected by enrichment of mutations in pre-defined gene regions.

CHAPTER

FOUR

COMBINING THE OUTPUTS OF DRIVER IDENTIFICATION METHODS

4.1 Rationale

Our goal is to provide a catalogue of driver elements which appropriately reflects the consensus from the DIMs we run.

To combine the results of individual statistical tests, p-value combination methods continue to be a standard approach in the field: e.g., Fisher[1], Brown[2,3] and Stouffer Z-score[4] methods have been used for this purpose. These methods are useful for combining probabilities in meta-analysis, hence to provide a ranking based on combined significance under statistical grounds. However, the application of these methods may bear some caveats:

1. The ranking resulting from p-value combination may lead to inconsistencies when compared to the individual rankings, i.e., they may yield a consensus ranking that does not preserve recurrent precedence relationships found in the individual rankings.
2. Some methods, like Fisher's or Brown's method, tend to bear anti-conservative performance, thus leading to many likely false discoveries.
3. Balanced (non-weighted) p-value combination methods may lead to faulty results just because of the influence of one or more DIM performing poorly for a given dataset.

Weighted methods to combine p-values, like the weighted Stouffer Z-score, provide some extra room for proper balancing, in the sense of incorporating the relative credibility of each DIM. We reasoned that any good operational criteria to allocate weights should satisfy the following requirements: i) provide weighting on a cohort-specific basis, thereby allowing the relative credibility of a DIM to depend on the cohort; ii) reflect prior knowledge about known bona-fide driver genes; iii) reflect prior knowledge about the criteria that each DIM employed to yield its output.

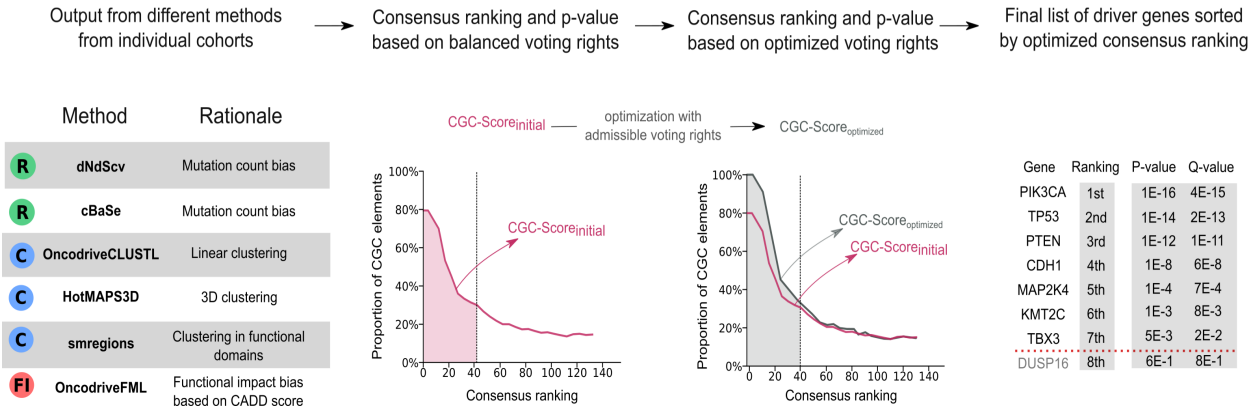
Our approach works independently for each cohort: to create a consensus list of driver genes for each cohort, we first determine how credible each DIM is when applied to this specific cohort, based on how many bona-fide cancer genes reported in the COSMIC Cancer Gene Census database (CGC v95) are highly ranked according to the DIM. Once a credibility score has been established, we use a weighted method for combining the p-values that each DIM gives for each candidate gene: this combination takes the DIMs credibility into account. Based on the combined p-values, we conduct FDR correction to conclude a ranking of candidate driver genes alongside q-values.

[1] Fisher R.A. (1948) figure to question 14 on combining independent tests of significance. Am. Statistician , 2, 30–31.

[2] Brown, M. B. 400: A Method for Combining Non-Independent, One-Sided Tests of Significance. Biometrics 31, 987 (1975). DOI: 10.2307/2529826

[3] William Poole, et al. Combining dependent P-values with an empirical adaptation of Brown's method, Bioinformatics, Volume 32, Issue 17, 1 September 2016, Pages i430–i436, https://doi.org/10.1093/bioinformatics/btw438

[4] Zaykin, D. V. Optimally weighted Z-test is a powerful method for combining probabilities in meta-analysis. Journal of Evolutionary Biology 24, 1836–1841 (2011). doi: 10.1111/j.1420-101.2011.02297.x



4.2 Weight Estimation by Voting

The relative credibility for each method is based on the ability of the method to give precedence to well-known genes already collected in the CGC catalogue of driver genes. As each method yields a ranking of driver genes, these lists can be combined using a voting system –Schulze's voting method. The method allows us to consider each method as a voter with some voting rights (weighting) which casts ballots containing a list of candidates sorted by precedence. Schulze's method takes information about precedence from each individual method and produces a new consensus ranking[5].

Instead of conducting balanced voting, we tune the voting rights of the methods so that we the enrichment of CGC genes at the top positions of the consensus list is maximized. We limit the share each method can attain in the credibility simplex –up to a uniform threshold. The resulting voting rights are deemed the relative credibility for each method.

4.3 Ranking Score

Upon selection of a catalogue of bona-fide known driver elements (CGC catalogue of driver genes) we can provide a score for each ranking R of genes as follows:

$$E(R) = \sum_{i=1}^{N} \frac{p_i}{\log(i+1)}$$

where p_i is the proportion of elements with rank higher than i which belong to CGC and N is a suitable threshold to consider only the N top ranked elements. Using E we can define a function f that maps each weighting vector w (in the 5-simplex) to a value $E(R_w)$ where R_w denotes the consensus ranking obtained by applying Schulze's voting with voting rights given by the weighting vector w.

4.4 Optimization with constraints

Finally we are bound to find a good candidate for $\widehat{w} = \text{argmax}(f)$. For each method to have chances to contribute in the consensus score, we impose the mild constraint of limiting the share of each method up to 0.3.

Optimization is then carried out in two steps: we first find a good candidate $\widehat{w_0}$ by exhaustive search in a rectangular grid satisfying the constraints defined above (with grid step=0.05); in the second step we take $\widehat{w_0}$ as the seed for a stochastic hill-climbing procedure (we resort to Python's scipy.optimize "basinhopping", method=SLSQP and stepsize=0.05).

[5] https://arxiv.org/pdf/1804.02973.pdf

4.5 Estimation of combined p-values using weighted Stouffer Z-score

Using the relative weight estimate that yields a maximum value of the objective function f we combined the p-values resorting to the weighted Stouffer Z-score method. Thereafter we performed Benjamini-Hochberg FDR correction with the resulting combined p-values, yielding one q-value for each genomic element. If the element belongs to CGC, we computed its q-value using only the collection of p-values computed for CGC genes. Otherwise, we computed the q-value using all the computed p-values.

4.6 Tiers of driver genes from sorted list of combined rankings and p-values

To finalize the analysis we considered only genes with at least two mutated samples in the cohort under analysis. These genes were classified into four groups according to the level of evidence in that cohort that the gene harbours positive selection.

1) The first group, named as TIER1, contained genes showing high confidence and agreement in their positive selection signals. Given the ranked list of genes obtained by the Schulze voting, TIER1 comprises all the ranked genes whose ranking is higher than the first gene with combined q-value lower than a specific threshold (by default threshold=0.05).

2) The second group, name as TIER2, was devised to contain known cancer driver genes, showing mild signals of positive selection, that were not included in TIER1. More in detail, we defined TIER2 genes as those CGC genes, not included in TIER1, whose CGC q-value was lower than a given threshold (default CGC q-value=0.25). CGC q-value is computed by performing multiple test correction of combined p-values restricted to CGC genes.

3) The third group, are genes not included in TIER1 or TIER2 with scattered signals of positive selection, frequently coming from one single method. Particularly, given the ranked list of genes by the Schulze voting, TIER3 was composed of all the ranked genes with q-value lower than a given threshold (by default threshold=0.05) whose ranking is higher than TIER1 last gene position and lower than the rejection ranking position. The rejection ranking position is defined as the ranking position for which all elements have a q-value lower than the input threshold (by default threshold=0.05). Finally, other genes not included in the aforementioned classes are considered non-driver genes.

4.7 Combination benchmark

Warning: This benchmark was performed on IntOGen Plus v2020

We have assessed the performance of the combination compared to i) each of the six individual methods and ii) other strategies to combine the output from cancer driver identification methods.

4.7.1 Datasets for evaluation

To ensure a reliable evaluation we decided to perform an evaluation based on the 32 Whole-Exome cohorts of the TCGA PanCanAtlas project (downloaded from *https://gdc.cancer.gov/about-data/publications/pancanatlas*). These cohorts sequence coverage, sequence alignment and somatic mutation calling were performed using the same methodology guaranteeing that biases due to technological and methodological artifacts are minimal.

The Cancer Genes Census –version v87– was downloaded from the COSMIC data portal (*https://cancer.sanger.ac.uk/census*) and used as a positive set of known cancer driver genes.

We created a catalog of genes that are known not to be involved in cancerogenesis. This set includes very long genes (HMCN1, TTN, OBSCN, GPR98, RYR2 and RYR3), and a list of olfactory receptors from Human Olfactory Receptor Data Exploratorium (HORDE) (https://genome.weizmann.ac.il/horde/; download date 14/02/2018). In addition, for all TCGA cohorts, we added non-expressed genes, defined as genes where at least 80% of the samples showed a RSEM expressed in log2 scale smaller or equal to 0. Expression data for TCGA was downloaded from *https://gdc.cancer.gov/about-data/publications/pancanatlas*.

4.7.2 Metrics for evaluation

We defined a metric, referred to as CGC-Score, that is intended to measure the quality of a ranking of genes as the enrichment of CGC elements in the top positions of the ranking; specifically given a ranking R mapping each element to a rank, we define the CGC-Score of R as

$$\text{CGC-Score}(R) = \sum_{i=1}^{N} \frac{p_i}{log(i+1)} \; / \; \sum_{i=1}^{N} \frac{1}{log(i+1)}$$

where p_i is the proportion of elements with rank $\leq i$ that belong to CGC and N is a suitable threshold to consider just the top elements in the ranking (by default N=40).

We estimated the CGC-Score across TCGA cohorts for the individual methods ranking and the combined Schulze ranking.

Similarly, we defined a metric, referred to as Negative-Score, that aims to measure the proportion non-cancer genes among the top positions in the ranking. Particularly, given a ranking R mapping each element to a rank, we define the Negative-Score of R as:

$$\text{Negative-Score}(R) = \sum_{i=1}^{N} \frac{p_i}{log(i+1)} \; / \; \sum_{i=1}^{N} \frac{1}{log(i+1)}$$

where p_i is the proportion of elements with rank $\leq i$ that belong to the negative set and N is a suitable threshold to consider just the top elements in the ranking (by default N = 40). We estimated the Negative-Score across TCGA cohorts for the individual methods ranking and the combined Schulze ranking.

4.7.3 Comparison with individual methods

We compared the CGC-Score and Negative-Score of our combinatorial selection strategy with the individual output from the six driver discovery methods integrated in the pipeline.

As a result we observed a consistent increase in CGC-Score of the combinatorial strategy compared to individual methods across TCGA cohorts (see Figure below panel A-B). Similarly, we observed a consistent decrease in Negative-Score across TCGA cohorts (see Figure below panel C). In summary, the evaluation shows that the combinatorial strategy increases the True Positive Rate (measured using the CGC-Score) preserving lower False Positive Rate (measured using the Negative-Score) than the six individual methods included in the pipeline.

4.7.4 Leave-one-out combination

We aimed to estimate the contribution of each method's ranking to the final ranking after Schulze's weighted combination. To tackle this question, we measured the effect of removing one method from the combination by, first, calculating the CGC-Score of the combination excluding the aforementioned method and, next, obtaining its ratio with the original combination (i.e., including all methods). This was iteratively calculated for all method across TCGA cohorts. Methods that positively contributed to the combined ranking quality show a ratio below one, while methods that negatively contributed to the combined ranking show a ratio greater than one.

We observed that across TCGA cohorts most of the methods contributed positively (i.e., ratio above one) to the weighted combination performance. Moreover, there is substantial variability across TCGA cohorts in the contribution

of each method to the combination performance. In summary, all methods contributed positively to the combinatorial performance across TCGA supporting our methodological choice for the individual driver discovery methods (see Figure below panel E).

4.7.5 Comparison with other combinatorial selection methods

We compared the CGC-Score and Negative-Score of our combinatorial selection strategy against other methods frequently used employed to produce ranking combinations, either based on ranking information –such as Borda Count[6] – or based on statistical information –such as Fisher[1] or Brown[2],[3] methods. Hereto, we briefly describe the rationale of the four methods we used to benchmark our ranking. For the sake of compact notation, let's denote the rank and p-value of gene g produced by method m_i as $r_{i,g}$ and $p_{i,g}$, respectively.

Borda Count: For each ranked item g and method m_i, it assigns a score $s_{i,g} = N - l_{i,g}$, where N stands for the total number of items to rank and $l_{i,g}$ is the number of items ranked below g according to method m_i. For each item g an overall score $s_g = s_{1,g} + \ldots + s_{k,g}$ can then be computed for each g, whence a ranking is obtained by descending sort.

Fisher: It is based on the p-values $p_{i,g}$. For each item g the method produces a new combined p-value by computing the statistic:

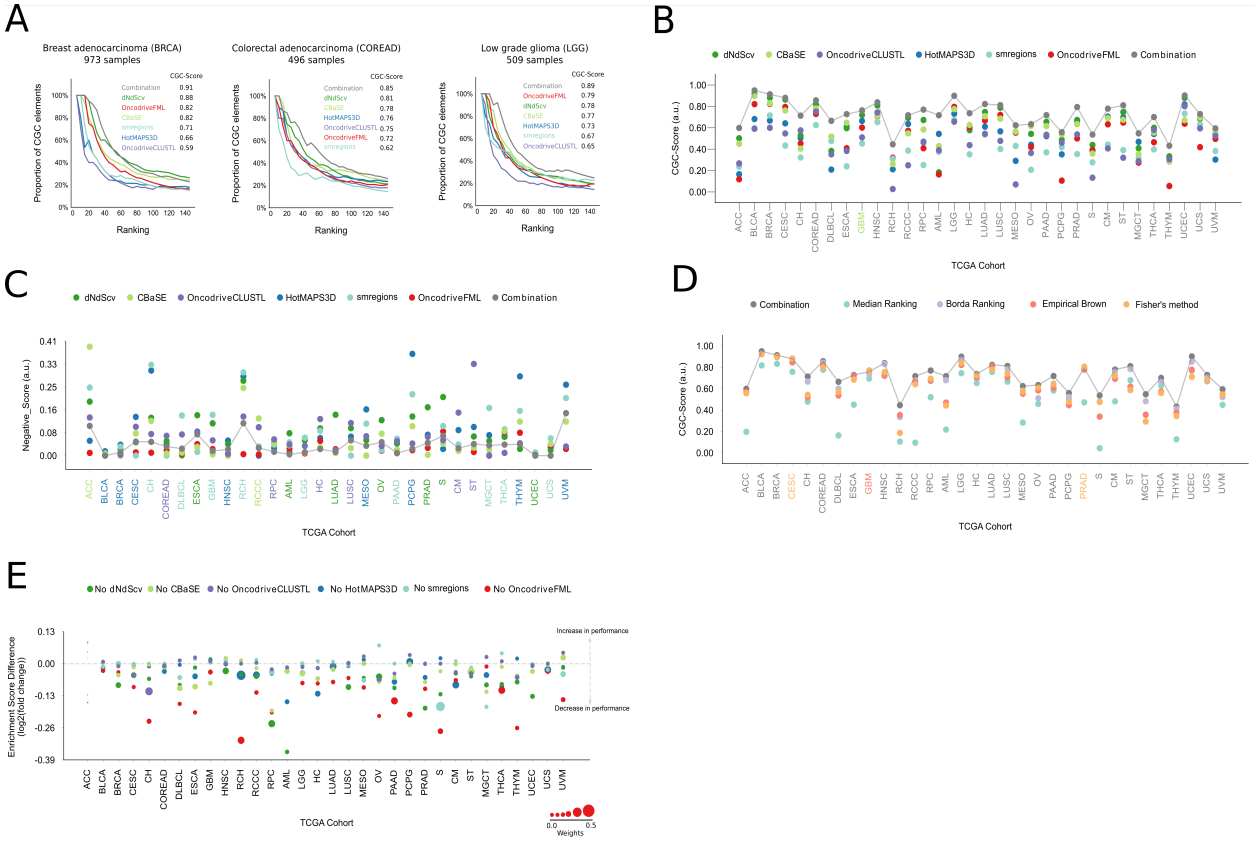
$F_g = -2 \log\ p_{i,g} \sim \chi^2_{2k}$.

Under the null hypothesis, F_g are distributed as a chi-square with $2k$ degrees of freedom, whence a p-value, which in turn yields a raking by ascending sort. Its applicability is limited by the assumption that the methods provide independent significance tests.

Brown: This method overcomes the independence requirement of Fisher's method by modeling the dependencies between the statistical tests produced by each method, specifically by estimating the covariance $\Omega_{i,j} = \text{cov}(-2 \log p_{i,g}, -2 \log p_{j,g})$. Brown's method[2] and its most recent adaptation[3] have been proposed as less biased alternatives to Fisher.

We then computed the CGC-Score and Negative-Score based on the consensus ranking from the aforementioned combinatorial methods and compared them to our Schulze's weighted combination ranking across all TCGA cohorts. Our combinatorial approach met a larger enrichment in known cancer genes for 29/32 (90%) TCGA cohorts (see Figure below panel D).

[6] Emerson P. The original Borda count and partial voting. Soc Choice Welf (2013) 40:353–358. doi 10.1007/s00355-011-0603-9


A
Breast adenocarcinoma (BRCA)
973 samples
Colorectal adenocarcinoma (COREAD)
496 samples
Low grade glioma (LGG)
509 samples
Proportion of CGC elements
Ranking
CGC-Score
Combination 0.91
dNdScv 0.88
OncodriveFML 0.82
CBaSE 0.82
smregions 0.71
HotMAPS3D 0.66
OncodriveCLUSTL 0.59
Combination 0.85
dNdScv 0.81
CBaSE 0.78
HotMAPS3D 0.76
OncodriveCLUSTL 0.75
OncodriveFML 0.72
smregions 0.62
Combination 0.89
OncodriveFML 0.79
dNdScv 0.78
CBaSE 0.77
HotMAPS3D 0.73
smregions 0.67
OncodriveCLUSTL 0.65
B
dNdScv
CBaSE
OncodriveCLUSTL
HotMAPS3D
smregions
OncodriveFML
Combination
CGC-Score (a.u.)
TCGA Cohort
C
Negative_Score (a.u.)
TCGA Cohort
D
Combination
Median Ranking
Borda Ranking
Empirical Brown
Fisher's method
CGC-Score (a.u.)
TCGA Cohort
E
No dNdScv
No CBaSE
No OncodriveCLUSTL
No HotMAPS3D
No smregions
No OncodriveFML
Enrichment Score Difference
(log2(old change))
Increase in performance
Decrease in performance
TCGA Cohort
Weights

CHAPTER

FIVE

DRIVERS POSTPROCESSING

The intOGen pipeline outputs a ranked list of driver genes for each input cohort. We aimed to create a comprehensive catalog of driver genes per tumor type from all the cohorts included in this version.

Then, we performed a filtering on automatically generated driver gene lists per cohort. This filtering is intended to reduce artifacts from the cohort-specific driver lists, due to e.g. errors in calling algorithms, local hypermutation effects, undocumented filtering of mutations.

We first created a collection of candidate driver genes by selecting either: i) significant non-CGC genes (q-value < 0.05) with at least two significant bidders (methods rendering the genes as significant); ii) significant CGC genes (either q-value < 0.05 or CGC q-value < 0.25) from individual cohorts. All genes that did not fulfill these requirements were flagged as 'No driver' in the DRIVER column at the unfiltered_drivers.tsv file.

Additionally, candidate driver genes were further filtered using the following criteria:

1. We discarded non-expressed genes using TCGA expression data. For tumor types directly mapping to cohorts from TCGA –including TCGA cohorts– we removed non-expressed genes in that tumor type. We used the following criterion for non-expressed genes: genes where at least 80% of the samples showed a negative log2 RSEM. For those tumor types which could not be mapped to TCGA cohorts this filtering step was not done.

2. We also discarded genes highly tolerant to Single Nucleotide Polymorphisms (SNP) across human populations. Such genes are more susceptible to calling errors and should be taken cautiously. More specifically, we downloaded transcript specific constraints from gnomAD (release 2.1; 2018/02/14) and used the observed-to-expected ratio score (oe) of missense (mys), synonymous (syn) and loss-of-function (lof) variants to detect genes highly tolerant to SNPs. Genes enriched in SNPs (oe_mys > 1.5 or oe_lof > 1.5 or oe_syn > 1.5) with a number of mutations per sample greater than 1 were discarded. Additionally, we discarded mutations overlapping with germline variants (germline count > 5) from a panel of normals (PON) from Hartwig Medical Foundation (https://storage.googleapis.com/hmf-public/HMFtools-Resources/dna_pipeline/v5_31/38/variants/SageGermlinePon.98x.38.tsv.gz).

3. We also discarded genes that are likely false positives according to their known function from the literature. We convened that the following genes are likely false positives: i) known long genes such as TTN, OBSCN, RYR2, etc.; ii) olfactory receptors from HORDE (http://bioportal.weizmann.ac.il/HORDE/; Build #44c - 30 July 2019); iii) genes not belonging to Tier1 CGC genes lacking literature references according to CancerMine[2] (http://bionlp.bcgsc.ca/cancermine/; As of 7 December 2021).

4. We also removed non CGC genes with more than 2 mutations in one sample. This abnormally high number of mutations in a sample may be the result of either a local hypermutation process or cross contamination from germline variants.

5. Finally we discarded genes whose mutations are likely the result of local hypermutation activity. More specifically, some coding regions might be the target of mutations associated to COSMIC Signature 9 (https:

[2] Lever J, et al. CancerMine: a literature-mined resource for drivers, oncogenes and tumor suppressors in cancer. Nat Methods. 2019 Jun;16(6):505-507. doi: 10.1038/s41592-019-0422-y. Epub 2019 May 20.

//cancer.sanger.ac.uk/cosmic/signatures) which is associated to non-canonical AID activity in lymphoid tumours. In those cancer types were Signature 9 is known to play a significant mutagenic role (i.e., AML, Non-Hodgkin Lymphomas, B-cell Lymphomas, CLL and Myelodysplastic syndromes) we discarded genes where more than 50% of mutations in a cohort of patients were associated with Signature 9.

Candidate driver genes that were not discarded composed the catalog of driver genes.

5.1 Classification according to MSKCC oncotree

We then annotated the catalog of highly confident driver genes according to their annotation level in CGC[1]. Specifically, we created a three-level annotation: i) the first level included driver genes with a reported involvement in the source tumor type according to the CGC; ii) the second group included CGC genes lacking reported association with the tumor type; iii) the third group included genes that were not present in CGC.

To match the tumor type of our analyzed cohorts and the nomenclature/acronyms of cancer types reported in the CGC we used MSKCC oncotree (as of November 2021). Resulting in 889 cancer type nodes. We customized the oncotree according to the following rules:

1. NON_SOLID node added after TISSUE and before MYELOID and LYMPHOID
2. SOLID node added after TISSUE and before the rest of tissues
3. ALL node added after LMN and before BLL and TLL

Note: The current version of the oncotree used in IntOGen 2023 is available at this GitHub repo: bbglab/oncotree .

5.2 Mode of action of driver genes

We computed the mode of action for highly confident driver genes. To do so, we first performed a pan-cancer run of dNdScv across all TCGA cohorts. We then applied the aforementioned algorithm (see Mode of action section below for more details on how the algorithm determines the role of driver genes according to their distribution of mutations in a cohort of samples) to classify driver genes into the three possible roles: Act (activating or oncogene), LoF (loss-of-function or tumor suppressor) or Amb (ambiguous or non-defined). We then combined these predictions with prior knowledge from the Cancer Genome Interpreter[3] according to the following rules: i) when the inferred mode of action matched the prior knowledge, we used the consensus mode of action; ii) when the gene was not included in the prior knowledge list, we selected the inferred mode of action; iii) when the inferred mode of action did not match the prior knowledge, we selected that of the prior knowledge list.

[1] Sondka Z, et al. The COSMIC Cancer Gene Census: describing genetic dysfunction across all human cancers. Nat Rev Cancer. 2018;18(11):696–705. doi:10.1038/s41568-018-0060-1

[3] Tamborero D, et al. Cancer Genome Interpreter annotates the biological and clinical relevance of tumor alterations. Genome Med. 2018;10(1):25. Published 2018 Mar 28. doi:10.1186/s13073-018-0531-8

CHAPTER

SIX

BOOSTDM CONNECTION

IntOGen pipeline integrates the generation of files needed to run by BoostDM[1] in order to keep a unified data and software environment and limit preprocessing of input for BoostDM as much as possible. The integrations consists in two new steps in the pipeline:

6.1 DriverSaturation

It computes all the possible mutations for a given gene mapping to the canonical transcript. It uses VEP v101. Specifically, it considers both the exons of the transcript and intronic sites within 25 bps distance from the intron-exon junctions.

6.2 Filter MNVs

Individual SNVs in adjacent positions reported in the same sample are discarded as potential multiple nucleotide variants (MNVs) that are wrongly called as separate SNVs.

[1] Ferran Muiños, et al. In silico saturation mutagenesis of cancer genes; Nature 2021. (https://doi.org/10.1038/s41586-021-03771-1)

CHAPTER

SEVEN

REPOSITORY OF MUTATIONAL FEATURES

7.1 Linear clusters

Linear clusters for each gene and cohort were identified by OncodriveCLUSTL. We defined as significant those clusters in a driver gene with a p-value lower than 0.05. The start and end of the clusters were retrieved from the first and last mutated amino acid overlapping the cluster, respectively.

7.2 3D clusters

Information about the positions involved in the 3D clusters defined by HotMAPS were retrieved from the gene specific output of each cohort. We defined as significant those amino acids in a driver gene with a q-value lower than 0.05.

7.3 Pfam Domains

Pfam domains for each driver gene and cohort were identified by smRegions. We defined as significant those domains in driver genes with a q-value lower than 0.1 and with positive log ratio of observed-to-simulated mutations (observed mutations / simulated mutations > 1). The first and last amino acid are defined from the start and end of the Pfam domain, respectively.

7.4 Excess of mutations

The so-called excess of mutations for a given coding consequence-type quantifies the proportion of observed mutations at this consequence-type that are not explained by the neutral mutation rate. The excess is inferred from the dN/dS estimate ω as $(\omega - 1) \ / \ \omega$. We computed the excess for missense, nonsense and splicing-affecting mutations.

7.5 Mode of action

We computed the gene-specific dN/dS estimates for nonsense and missense mutations, denoted ω_{non} and ω_{mis}. Then each gene induces a point in the plane with coordinates $(\omega_{\text{non}},\ \omega_{\text{mis}})$. We deemed a gene Act (resp. LoF) if its corresponding point sits above (resp. below) the diagonal ($x = y$) up to an uncertainty threshold of 0.1. Genes within the uncertainty area as well as genes with $\omega_{\text{non}} < 1$ and $\omega_{\text{mis}} < 1$ were deemed "ambiguous".

CHAPTER

EIGHT

INSTALLATION

The IntOGen pipeline requires Nextflow and Singularity in order to run.

Beside them, a number of different datasets need to be downloaded, and other pieces of software installed (as Singularity containers).

For information on how to download and build all these requirements, check the README file in the build folder.

8.1 Usage

Once all the prerequisites are available, running the pipeline only requires to execute the `intogen.nf` file with the appropiate parameters. E.g.:

```
nextflow run intogen.nf --input <input>
```

There are a number of parameters and options that can be added:

-resume	Nextflow feature to allow for resumable executions.
--input <path>	Path of the input. See below for more details.
--output <path>	Path where to store the output. Default: `intogen_analysis`.
--datasets <path>	Path to the folder containing the datasets. Default: `datasets`.
--containers <path>	Path to the folder containing the singularity images. Default: `containers`.
--annotations <file>	Path to the default annotations file. Default: `config/annotations.txt`. See the input section for more details.
--seed <int>	Seed to be used for reproducibility. This applies to 4 methods: smRegions, OncodriveCLUSTL, OncodriveFML, dNdScv.
--debug <bool>	Ask methods for a more verbose output if set to True.

8.2 Input & output

8.2.1 Input

Although the pipeline does most of its computations at the cohort level, the pipeline is prepared to work with multiple cohorts at the same time.

Each cohort must contain, at least, the chromosome, position, ref, alt and sample. Files are expected to be TSV files with a header line.

Important: All mutations should be mapped to the positive strand. The strand value is ignored.

In addition, each cohort must be associated with:

- cohort ID (`DATASET`): a unique identifier for each cohort.
- a cancer type (`CANCER`): although any acronym can be used here, we recommend to restrict to the acronyms that can be found in `extra/data/dictionary_long_name.json`.
- a sequencing platform (`PLATFORM`): `WXS` for whole exome sequencing and `WGS` for whole genome sequencing
- a reference genome (`GENOMEREF`): only `HG38` and `HG19` are supported

Cohort file names, as well as the fields mentioned above must not contain dots.

The way to provide those values is through OpenVariant , a comprehensive Python package that provides different functionalities to read, parse and operate different multiple input file formats (e. g. tsv, csv, vcf, maf, bed). Whether you are planning to run single or multiple cohorts, you would need to provide an annotation file in yaml format to specify the above mentioned structure required by IntOGen. Instructions on how to build an annotation file are documented here: OpenVariant annotation file .

8.2.2 Output

By default this pipeline outputs 4 files:

- `cohorts.tsv`: summary of the cohorts that have been analyzed
- `drivers.tsv`: summary of the results of the driver discovery by cohort
- `mutations.tsv`: summary of all the mutations analyzed by cohort
- `unique_drivers.tsv`: information on the genes reported as drivers (in any cohort)
- `unfiltered_drivers.tsv`: information on the filters applied to the post-processing step: from the output of the combination to the final set of driver genes.

Those files can be found in the path indicated with the `--output` options.

Moreover, the `--debug true` options will generate a `debug` folder under the output folder, in which all the input and output files of the different methods are linked.

CHAPTER

NINE

INDICES AND TABLES

- genindex
- modindex
- search