

UNIVERSITY OF MILANO-BICOCCA
Department of Informatics, Systems and Communication
Ph.D. Program in Computer Science
XXIX Cycle



Anti-Cancer Drug Resistance Causal Modeling from Lentiviral-Vector Integration Site Studies

Ph.D. Candidate

Dr. Giulio Spinozzi, 787765

Tutor

Prof. Giancarlo Mauri

Supervisors

Dr. Eugenio Montini

Prof. Marco Antoniotti

Coordinator: *Prof. Stefania Bandini*

Academic Year: 2015/2016

*To my beloved Laura,
my parents
and to my cat, Vincent*

Acknowledgments

This work has been possible because of so many great people I worked with in the last years. I want to thank them here.

After graduating in engineering at Perugia I replied to a very interesting proposal of San Raffaele Telethon Institute for Gene Therapy (SR-Tiget), who was looking for a young candidate, graduated, to work alongside a postdoc in bioinformatics analysis that the group of Eugenio Montini was facing. The first thanksgiving can only be at that postdoc, Andrea Calabria, which from day one has believed in me and still beside each Bioinformatics analysis of SR-Tiget. Right after I thank the head of my unit (Safety of Gene Therapy and Insertional Mutagenesis Research Unit) Eugenio Montini, who with Andrea, made possible my job to SR-Tiget and my PhD in Milano-Bicocca. In this faculty I had the pleasure of working with fantastic special people and especially scientists, always available and present, as my tutor Prof. Giancarlo Mauri, Prof. Marco Antoniotti who followed me and led immediately, Giulio Caravagna, Alex Graudenzi and Daniele Ramazzotti, they have always led in the scientific part and with whom I shared hours of hard work. The last collaboration and thanks are for National Research Council, Institute for Biomedical Technologies (CNR-ITB), and in particular to Stefano Beretta, Ivan Merelli and Luciano Milanese, for the help on the computational part of this project. I finally want to thank Davide Rambaldi for the invaluable assistance to the develop of ISAnalytics, the starting point for each future analysis after VISPA2. Thank you.

Acronyms

ACDR, Anti-Cancer Drug Resistance
adLIMS, ADempiere Laboratory Information Management System
AF, Association File
AML, Acute Myeloid Leukemia
AS, Alignment Score
BAM, Binary-sequence Alignment Format
BED, Browser Extensible Data
BP, Base Pair
CAPRESE, CAncer PRogression Extraction with Single Edges
CAPRI, CAncer PRogression Inference
CBN, Conjunctive Bayesian Network
CIGAR, CIGAR String from BWA
CIS, Common Insertion (Integration) Site
CSV, Comma-Separated Values
DAG, Direct Acyclic Graph
DB, DataBase
DBMS, DataBase Management System
DNA, Deoxyribonucleic Acid
FPR, False Positive Rate
GO, Gene Ontology
GTF, Gene Transfer Format
GUI, Graphical User Interface
HER2 Human Epidermal Growth Factor Receptor 2
HIV, Human Immunodeficiency Virus
HSC, Hematopoietic Stem Cell
HSPC, Hematopoietic Stem-Progenitor Cell
IP, Internet Protocol
IS, Integration Site
LAM-PCR, Linear Amplification Mediated PCR
LC, Linker Cassette
LTR, Long Terminal Repeat
LV, Lentiviral Vector
MEM, Maximum Exact Matches
MD, Mismatching Positions/Bases
miRNA (microRNA), Small non-Coding RNA Molecule
MLD, Metachromatic Leukodystrophy
Myr, Million Years
NGS, Next Generation Sequencing

PCR, Polymerase Chain Reaction

PE, Paired-End

R1, Illumina Read in 5'-3'

R2, Illumina Read in 3'-5'

ROC, Receiver Operating Characteristic

RNA, Ribonucleic Acid

RV, Retroviral Vector

SAM, Sequence Alignment Format

SC, Sequence Count

SE, Single End

SLiM-PCR, Sonicated-Linker-Mediated-PCR

SR-Tiget, San Raffaele Telethon Institute for Gene Therapy

TAG, Sequence of Fixed Nucleotides

TPR, True Positive Rate

TRONCO, TRanslational ONCOlogy

TSS, Transcription Start Site

TSV, Tab-Separated Values

VCN, Vector Copy Number

VISPA, Vector Integration Site Parallel Analysis

WAS, Wiskott Aldrich Syndrome

XS, Secondary Alignment Score

Abstract

Evolution plays a key role in *cancer* as the result of the accumulation of genetic alterations, which provide selective advantages to a tumor cell, allowing resistance to anti-cancer drugs. Unfortunately, however, the identification of the driver mutations and thus the mechanisms underlying *anti-cancer drug resistance* (ACDR) still remains a challenge. We previously demonstrated that *lentiviral vectors* (LVs), when properly modified, might integrate near specific genes, alter their expression and induce cancer or ACDR in vivo and in vitro [1, 2, 3]. The analysis of vector-cellular genomic junctions in tumor or ACDR cells allowed identifying causative genes of HER2+ breast cancer cell line using a statistical approach defined *Common Insertion Sites* (CISs) that highlight genomic regions targeted at significantly higher frequency than expected by a random distribution [4, 5, 6]. The reconstruction of cumulative cancer progression from CIS genes has not been yet addressed and may produce causative gene networks. The aim of this project is studying anti-cancer drug resistance from exclusive and co-occurring genes using cumulative cancer progression from cell line CIS genes and investigating the relation between them.

Bioinformatics tools aimed at inferring cancer progression models, in terms of selective advantage relation among relevant genomic alteration from cross-sectional data (Next Generation Sequencing platforms), would allow identifying specific combinations of targeted drugs to overcome the occurrence of resistance. In a new context of vector *Integration Sites* (ISs), I developed an integrated bioinformatics workflow composed of: (i) an updated and more accurate version of *VISPA* (Vector Integration Site Parallel Analysis) [7], a pipeline for automated IS identification and annotation based on a distributed environment with a simple web based interface; (ii) identification of the CISs with a sliding window approach developed in [8, 9, 10]; (iii) a new statistical tool, *CAnceR PRogression Inference* (CAPRI) - [11, 12], to infer selective advantage relations among various mutational events in cancer cell genomes, mostly in relation with drug-resistance. The model is based on probabilistic causation and is able to reconstruct cancer progression *Direct Acyclic Graphs* (DAGs), involving the CIS genes. With the use of *GeneMANIA*¹ [13] and *Enrichr*² [14, 15], I studied the protein-protein interaction, Gene Ontology and Pathway relations between selected genes, collecting and visualizing results in gene networks.

By applying my new method to the published IS dataset from the two cell lines, I was able to generate progression models involving relevant genes (confirming that these are not mutually exclusive genes, by *Mutex* [16]), which are consistent with previously validated results, confirming the role of PIK3CA-ERBB2 genes in ACDR. Unfortunately, one of the two cell line has a low quality samples. For this reason,

¹<http://www.genemania.org>

²<http://amp.pharm.mssm.edu/Enrichr>

CAPRI was not able to generate the progression DAG. I generated the progression DAG for the other cell line, BT474, pre-treatment and post-treatment with Lapatinib respectively. The following step is to investigate the relations between genes, produced by the model, trying to find some useful new interactions and confirmations for ACDR studies (i.e. SUMO1-ERBB2-PIK3CA-CSMD3). New insertional mutagenesis data from lung cancer cell lines aimed to induce ACDR in vivo and in vitro are ongoing and will allow to validate and/or identify novel cancer progression models, as well as possible combinatorial therapies.

Preface

In this work I will try to give my contribution to science that I love, to which I have devoted the last years of my life and that surely will continue to do in the near future. When I think to this I always remind of the beauty and depth of what Newton said, talking about his life in relation to nature.

"I don't know what I may appear to the world, but to myself I seem to have been only a boy playing on the sea-shore, and diverting myself in now and then finding a smoother pebble or a prettier shell than ordinary, whilst the great ocean of truth lay all undiscovered before me." Isaac Newton

In the era of *clinical genomics* and *personalized medicine*, Genetics and Molecular Biology are becoming the keys for understanding the mechanisms of many of the human diseases, like cancer and genetic diseases.

Genetics... *Wikipedia*

"Genetics is the study of genes, genetic variation and heredity in living organisms. It is generally considered a field of biology, but it intersects frequently with many of the life sciences and is strongly linked with the study of information systems."

Molecular Biology... *William Astbury* in Nature 1961

"...not so much a technique as an approach, an approach from the viewpoint of the so-called basic sciences with the leading idea of searching below the large-scale manifestations of classical biology for the corresponding molecular plan. It is concerned particularly with the forms of biological molecules and [...] is predominantly three-dimensional and structural which does not mean, however, that it is merely a refinement of morphology. It must at the same time inquire into genesis and function."

I am an engineer, a computer scientist, but the last years I have learned also genetics and molecular biology, useful to produce a comprehensive framework, described

in this thesis, to investigate some aspects related to breast cancer. Given my background, I think that my support to genetic and clinical research will be useful to build new solutions in Bioinformatics, with new methods, algorithms and tools.

All this work was only possible thanks to all the collaborations and people, that are:

- **San Raffaele Telethon Institute for Gene Therapy:** Andrea Calabria, Eugenio Montini, Stefano Brasca, Daniela Cesana, Fabrizio Benedicenti, Valentina Pirazzoli, Riccardo Biavasco, Monica Volpin, Yasmin Natalia Serina Secanechia, Pierangela Gallina, Laura Rudilosso, Erika Tenderini, Marco Ranzani, Stefano Annunziato and Luigi Naldini
- **San Raffaele Hospital:** Davide Rambaldi
- **University of Milano-Bicocca:** Marco Antoniotti, Giancarlo Mauri, Giulio Caravagna, Alex Graudenzi, Daniele Ramazzotti and Stefano Beretta
- **National Research Council, Institute for Biomedical Technologies:** Ivan Merelli, Marco Moscatelli and Luciano Milanese

My work has been supported by many projects, both international and national funding:

- Association for International Cancer Research (AICR 09-0784 to E.M.)
- Telethon Foundation (TGT11D1 to E.M.)
- Ph.D. Program in Computer Science, University of Milano-Bicocca (to G.S.)
- Italian SuperComputing Resource Allocation sponsored by CINECA for VISPA pipeline (to A.C.)

For this work I have won the *Travel Grant* and the *Oral Presentation* at Bioinformatics Italian Society (BITS) 2016 at Salerno, this year.

Contents

1	Introduction	25
1.1	Gene Therapy: Insertional Mutagenesis	26
1.2	Retrieving Vector Integration Sites	28
1.3	Cancer Progression Models	28
1.4	Computational Requirements	28
1.5	Aim of this Work	29
2	State-of-the-Art	31
2.1	Integration Site Retrieval Tools	31
2.1.1	A New Improved Version of VISPA	38
2.2	Cancer Progression Modeling	39
2.2.1	Linear Path Models	40
2.2.2	Oncogenetic Tree Models	40
2.2.3	Distance-Based Tree Models	41
2.2.4	Conjunctive Bayesian Networks and Directed Acyclic Graphs	42
3	VISPA2: Faster and Extended Version of Vector Integration Site Parallel Analysis Tool	45
3.1	Bioinformatics Pipeline	46
3.1.1	Quality Controls and Filters	46
3.1.2	Adapter Removal and Trimming	50
3.1.3	Demultiplexing and Association File	50
3.1.4	Association File for Breast Cancer Experiment	51
3.1.5	LTR/LC Trimming and Internal Control Band Removal	52
3.1.6	Alignment to the Reference Genome	52
3.1.7	Filtering	54
3.1.8	IS Import in MySQL Database and Stats Summary	58
3.1.9	IS Merging and Collisions	59
3.1.10	IS Annotation	61
3.2	Web Interface	61
3.3	Performances	65

3.3.1	Precision and Recall	65
3.3.2	Space Required and Time Consuming	66
3.4	SR-Tiget IS Pipelines, Final Overview	68
4	VISPA2 Post Analysis: ISAnalytics and Common Insertion Site Identification	71
4.1	Integration Site Analytics (ISAnalytics)	72
4.1.1	Motivation	72
4.1.2	Data Loader	72
4.1.3	Collision Detection and Removal	76
4.2	Frequency and Filtering of a IS	79
4.3	Clonal Abundance	80
4.3.1	IS Sharing	81
4.4	Common Insertion Sites	85
5	Cancer Progression Modeling	89
5.1	Introduction	89
5.1.1	New Usage of Causation	89
5.1.2	Shrinkage Estimator and Progression Tree Extraction	91
5.2	CAPRI	92
5.3	Integration Site Data and Causal Modeling	95
6	Case Study: Breast Cancer	99
6.1	Experimental Strategy Description	99
6.2	CAPRI on Breast Cancer Data	100
6.3	Enrichment Analysis	105
6.3.1	Enrichr	106
6.3.2	GeneMANIA	108
6.4	CAPRI Adaptation	112
7	Conclusions	113
7.1	Future Works	114
	Bibliography	128
	Appendices	131
A	Wet-Lab Procedures	131
A.1	LAM-PCR	131
A.1.1	Fusion Primers for LAM-PCR	132
A.2	Sonicated Linker-Mediated (SLiM)-PCR	133
A.2.1	Fusion Primers for SLiM-PCR	133

B	Information System	135
B.1	Computational Resources	135
B.1.1	@SR-Tiget: Gemini and Oracle	135
B.1.2	@CINECA: PICO	136
B.1.3	@CNR	136
B.2	Storage	136
B.2.1	NAS Server: QNAP TS-412 4-BAY	136
B.2.2	LaCie 4Big Quadra	137
B.3	Configurations	137
B.4	MySQL Tables	139
B.4.1	Import ISs from BED	139
B.4.2	Import ISs from BAM	141
B.4.3	Stats Summary	144
C	Some Useful Programs	147
C.1	VISPA2	147
C.1.1	Example of VISPA2 bash launch	147
C.1.2	FASTQ Quality Filter	148
C.1.3	Alignment to the Reference Genome with BWA-MEM	149
C.1.4	Repetitive Element Analysis with RepeatMasker	149
C.1.5	Filtering	150
C.1.6	IS Merging and Collisions	151
C.1.7	IS Annotation	153
C.2	ISAnalytics	154
C.2.1	Example of Usage	154
C.2.2	Example of Abel R Package Usage	154
C.3	Causal Modeling	156
C.3.1	TRONCO Script for Breast Cancer Progression with CAPRI	156
D	Anticancer Drug Resistance Supplementary Information	157
D.1	Test of Mutual Exclusivity with Mutex	157
D.1.1	BT474: Pre-Treatment	157
D.1.2	BT474: Post-Treatment	160
D.2	CAPRI on Merged (Pre and Post) Dataset	166
D.3	CAPRI before VISPA2	168

List of Figures

1.1	The founding clone in the primary tumor in AML1 contained somatic mutations that are all recurrent in AML and probably relevant for pathogenesis; one subclone within the founding clone evolved to become the dominant clone at relapse by acquiring additional mutations, including recurrent mutations [17]. This plot can be created with Fish-Plot R package [18].	26
1.2	How gene therapy works: the therapeutic lentiviral vector inserts the new gene into a cell. If the treatment is successful, the new gene will make a functional protein to treat a disease.	27
1.3	The Big Data era, WordCloud.	29
2.1	General IS retrieval pipeline.	31
2.2	Evaluation of VISPA related to the other IS analysis tools. (A) Overall strategy for reliability assessment, from the generation of the simulated dataset to the final results.	32
2.3	Genomic distribution of the 283 randomly selected loci. Each point represents a locus in the genome and its color refers one of the three categories (“RMSK” for repeat marker group, random or observed).	33
2.4	Density plot of the sequence lengths for the test dataset.	34
2.5	ROC curve of VISPA, Mavric, SeqMap and QuickMap.	37
2.6	Mismatch Analysis: (A) Box-plot of the distances in terms of genomic position (bp) between each mismatched IS and the reference IS, for each tool (Mavric, SeqMap and QuickMap). (B) Binning the genomic distances of the mismatched IS in intervals, each point accounts for the sum of IS for each bin, comparing the three tools.	38
2.7	Bar-plot of the relative percentages of aligned reads for each patient in [19]: the input sequences are the trimmed reads, whereas the resulting subsets from the alignment step are LV, U, A, and N.	39
2.8	Taken from <i>Howard Hughes Medical Institute</i> . Model of Colorectal Cancer, described in [20]	40

2.9	Taken from [21], Example of an oncogenetic tree model with $n = 6$ events	41
2.10	Taken from [21], Example of a distance based tree model with $n = 3$ events (left: conditional probabilities, right: distances)	42
2.11	Taken from [21], example of a conjunctive bayesian network with $n = 4$ events	43
3.1	VISPA2, workflow. In bold custom programs.	46
3.2	Quality filter for LAM-PCR read	47
3.3	Quality filter for SLiM-PCR read	47
3.4	Bad run: FastQC on R1 for MLD6 pool ET06	48
3.5	Good run: FastQC on R1 for MLD6 pool ET06v2	48
3.6	Venn diagram of the three runs to see the IS in common.	49
3.7	Repetitive families in human genome, [22].	53
3.8	MLD Statistics: each dot is a sample; (a) barcode demultiplexing; (b) sequence count; (c) unique ISs per replicate (3 replicates for each sample are not merged); (d) unique ISs merged in replicates.	58
3.9	Example matrix file of breast cancer Project, Chapter 6.	60
3.10	VISPA2 GUI: main page.	62
3.11	VISPA2 GUI: results, IS matrix, from breast cancer post-treatment, Chapter 6.	63
3.12	VISPA2 GUI: results, circos plot. The inner circle represents the IS density and the outer the IS read count, from breast cancer post-treatment, Chapter 6.	64
3.13	VISPA2 GUI: results, word cloud. The most targeted genes (not CISs necessarily), from breast cancer post-treatment, Chapter 6.	65
3.14	Precision and Recall definition, <i>Wikipedia</i> .	66
3.15	Precision and Recall of VISPA2 versus all other pipelines.	67
3.16	Precision and Recall of VISPA2 versus all other pipelines, a zoom.	67
3.17	Performances compared to VISPA (green) and VISPA2 (blue), time and space. Shorter bar is better.	68
3.18	History of SR-Tiget IS Pipelines. The old VISPA [7] supports 454 and Illumina single-end reads, has the Galaxy/Bash interface but lacks of paired-end support, is very slow because BLAST [23] and does not support repetitive elements. γ -TRIS has only the Bash version (developed in C++), is graph-based, supports repetitive elements, is genome free (because of the clustering algorithms in consensus sequences), supports the Illumina paired-end technology, but is terribly slow and CPU-intensive. The new VISPA2 combine the easiness of the old VISPA and the power of γ -TRIS. It has the Bash/GUI interface, supports the paired-end sequencing, is very fast and supports repetitive elements.	69

4.1	Steps after VISPA2.	71
4.2	ISAnalytics Data Loader: SummarizedExperiment integration.	73
4.3	ISAnalytics Data Loader: tab-separated and excel import into dataframes.	74
4.4	GRanges of an isset.	74
4.5	Conditions between samples.	75
4.6	ISAnalytics Structure: SummarizedExperiment and GRanges.	76
4.7	Density plot of inter-patients collisions MLD-WAS.	77
4.8	Density plot of inter-patients collisions MLD-WAS. Decision Plot.	78
4.9	SCFilter on MLD patient 1, distribution of IS.	79
4.10	Clonal Dominance in β -Thalassaemia [24].	80
4.11	MLD, Abundance Box-Plot of CD34, BM. ISAnalytics automatically writes gene label if abundance is $> 5\%$	82
4.12	MLD, Clonality Heatmap between shared ISs. For each IS, colored cells indicate retrieval at $> 5\%$, with higher color intensity indicating higher percentage, whereas gray cells indicate retrieval at low percentage (from 0.006% to $< 5\%$). Lack of color indicates that the integration was not retrieved at the indicated time point and source. The targeted genes are indicated on right, samples on bottom.	83
4.13	Tracking of ISs shared between multiple lineages with time in patient MLD01. Each row represents a specific IS, with colored bars indicating retrieval from the indicated cell lineage and time point after gene therapy (columns). The line color varies with the degree of sharing among lineages (red, high level of sharing; blue, low level of sharing; white, no integration retrieved). Only samples are visualized, in bottom.	84
4.14	Common Insertion Sites: CISs are defined as regions of the genome that are targeted by vector integrations in independent tumors with a frequency higher than the one that is expected to occur by chance [25]. Also a new model, graph-based, has been proposed in [26].	86
5.1	CAPRI Overview [11]. The algorithm examines cancer cross-sectional data to determine relationships related to genomic alterations (e.g., somatic mutations, copy-number variations, etc.) that modulate the somatic evolution of a tumor. When CAPRI concludes that mutation (EGFR) “selects for” aberration b (CDK mutation), such relations can be rigorously expressed using Suppes’ conditions, which postulates that if a selects b, then a occurs before b (<i>temporal priority</i>) and occurrences of a raises the probability of emergence of b (<i>probability raising</i>).	92

5.2	CAPRI Pipeline [11]. The first step is to collect experimental data and perform genomic analyses to derive profiles of, e.g., somatic mutations or copy-number variations for each patient or sample. Then, statistical analysis and biological priors are used to select events relevant to the progression and imputable by CAPRI - e.g., <i>driver mutations</i> . CAPRI can extract a progression model from these data and to assess various confidence measures on its constituting relations - e.g., (non-)parametric bootstrap and hypergeometric testing. Experimental validation concludes the pipeline.	94
5.3	CAPRI Pipeline for AML [11]. (left) Mutational profiles of $n = 64$ aCML patients - exome sequencing - with alterations in $ G = 9$ genes with either mutation frequency $> 5\%$ or belonging to an hypothesis imputed to CAPRI. Mutation types are classified as nonsense point, missense point and insertion/deletions, yielding $m = 16$ input events. Purple annotations report the frequency of mutations per sample. (right) Progression model inferred by CAPRI in supervised mode. Node size is proportional to the marginal probability of each event, edge thickness to the confidence estimated with 1000 non-parametric bootstrap iterations. The p-value of the hypergeometric test is displayed too. . .	95
5.4	Data Generation, step 1: after IS matrix creation, annotation and collision removal with ISAnalytics, I perform a count of ISs for each single CIS with $SC \geq 3$	96
5.5	Data Generation, step 2: The count matrix of CISs is now binarized and transposed for CAPRI input.	97
6.1	Anti-Cancer Drug Resistance Forward Screening Strategy at SR-Tiget.	100
6.2	Anti-Cancer Drug Resistance progression for BT474 cell line in pre-treatment condition. CISs are highlighted in blue, in green the CISs that CAPRI cannot distinguish (CISs in same samples), in this case the two CISs have the same progression. The edges in bold have a confidence $> 50\%$. I used <i>Cytoscape</i> to refine the figure [27].	103
6.3	Anti-Cancer Drug Resistance progression for BT474 cell line in post-treatment condition. CISs are highlighted in blue, in green the CISs that CAPRI cannot distinguish (CISs in same samples), in this case the two CISs have the same progression. The edges in bold have a confidence $> 50\%$. I used <i>Cytoscape</i> to refine the figure [27].	104
6.4	Selection of the most relevant relations in pre and post treatment in BT474 breast cancer cell line, from CAPRI. I used <i>Cytoscape</i> to refine the figure [27].	105

6.5	Enrichr: Cancer Cell Line Encyclopedia. Correctly shows a relation with BT474 of breast cancer, $p - value = 0.000002032$	107
6.6	Enrichr: Reactome. No significant pathways related with the gene set.	107
6.7	Enrichr: Cancer Cell Line Encyclopedia. Correctly shows a relation with BT474 of breast cancer, $p - value = 0.000005972$. Similar to pre-treatment, indeed this is a intrinsic characteristic of the cell line. .	107
6.8	Enrichr: Reactome. Significant relation with “Signaling of ERBB4”. Well known signaling pathway in breast cancer, as showed in [28, 29]. Moreover the “EGFR Signaling Pathway” results involved, as confirmed in [30, 31].	108
6.9	GeneMANIA for SUMO1 and pre-treatment. The pathway on which it maps the genes (blue), co-expression (purple) and physical interactions (protein-protein interactions, pink). For details please download, http://giuliospinozzi.altervista.org/docs/GMSUMO1.pdf	109
6.10	GeneMANIA for UBC and post-treatment. The pathway on which it maps the genes (blue), co-expression (purple) and physical interactions (protein-protein interactions, pink). For details please download, http://giuliospinozzi.altervista.org/docs/GMUBC.pdf	110
6.11	GeneMANIA for CSMD3 and post-treatment. The pathway on which it maps the genes (blue), co-expression (purple) and physical interactions (protein-protein interactions, pink). For details please download, http://giuliospinozzi.altervista.org/docs/GMCSMD3.pdf	111
6.12	Enrichr: post-treatment enrichment with KEGG [32] pathway engine. Significant relation with Ubiquitin related pathways, $p - value = 0.0002006$	111
6.13	CAPRI Pipeline [11], edited from Figure 5.2. The framework to study anticancer drug resistance from IS data. CAPRI is integrated with Mutex for checking the mutual exclusivity between CISs, the cross sectional usage of CISs as input, the <code>scFilter</code> of ISAnalytics to set the gene threshold and the final results interpretation with enrichment analysis with Enrichr and GeneMANIA.	112
A.1	LAM-PCR, sequential steps.	131
A.2	SLiM-PCR, fragments (P5 and P7 are the Illumina adapters)	133
B.1	MySQL Table Structure for IS imported from BED file.	139
B.2	MySQL Table Structure for IS imported from BAM file.	141
B.3	MySQL Table Structure for Stats Summary.	144
D.1	CAPRI on merged Dataset (no distinction of pre and post treatment).	166

D.2 CAPRI with the old version of VISPA, some CISs were false positives, LOC1001312234, ACACA and ARHGAP39 for the pre-treatment; SLITRK6, PKIA and KIFAP3 for the post-treatment.	168
--	-----

List of Tables

2.1	Specificity data for all IS tested tools.	36
2.2	Sensitivity data for all IS tested tools.	36
2.3	False Positive Rate (FPR) or 1-SPC data for all IS tested tools.	36
2.4	General Overview of the tested pipelines: test done on simulated reads described before.	37
3.1	MLD06-ET6 runs comparison. Bad pool (MLD06ET6), correct pool (MLD06ET6v2) and filtered correct pool with the custom quality filter program.	49
3.2	Association file example, pool fb360, Chapter 6	51
3.3	VISPA2 Performances compared to VISPA. I considered two types of Illumina sequencing: <i>**MiSeq</i> and <i>*HiSeq</i>	68
4.1	Theoretical use case scenarios for collisions between two patients.	77
4.2	Theoretical use case scenarios for collisions between two patients.	78
6.1	VISPA2 output of breast cancer experiment. For each cell line there are 2 conditions, before (Pre-Treatment) and after (Post-Treatment) the drug treatment. The number of samples is the same for the 2 conditions in the 2 cell lines, but the number of ISs retrieved is relatively different.	101
6.2	CISs obtained with VISPA (<i>CIS Order</i> > 10), identified with [9, 33]. <i>*FBXL20</i> is identified as <i>ERBB2</i> gene.	101
B.1	Gemini , HP Z820 Workstation. Intel(R) Xeon(R) CPU E5-2690, 2.90GHz. The storage is composed of 4 HD with 500GB and 10k rpm. <i>*Memory</i> per node. <i>**SR-Tiget</i> people in charge of computational resources are Giulio Spinozzi and Andrea Calabria.	135
B.2	Oracle , HP Z840 Workstation. Intel(R) Xeon(R) CPU E5-2690 v3, 2.60GHz. The storage is composed of 5 HD with 4TB and 7k rpm and 1 primary disk of 500GB SSD. <i>*Memory</i> per node. <i>**SR-Tiget</i> people in charge of computational resources are Giulio Spinozzi and Andrea Calabria.	135

-
- B.3 PICO is a BigData infrastructure that has been acquired (Nov 2014) devoted to "Big Analytics". It is named after the Italian Renaissance philosopher famous for his amazing memory. *Memory per node. Intel(R) Xeon(R) CPU 2670 v2, 2.5GHz. 136
- B.4 CNR-ITB bioinformatics computational resources present at ITB-Milano consist of more than 700 CPU-cores (Intel(R) Xeon(R) L5640 Westmere, Intel(R) Xeon(R) E5420 Harpertown based on Penryn microarchitecture and Intel(R) Xeon(R) E5420 Harpertown), more than 270 TB of disk space and more than 1700GBs of total memory, in an dual and quadri infiniband interconnected computational clusters. The architecture provides both an advanced HPC computational infrastructure and a distributed cloud-like virtualization facility for aggregating virtual servers, in turn providing the CNR-ITB bioinformatics services exposed to the Internet. *Memory per node. Intel(R) Xeon(R) CPU 2670 v2, 2.5GHz. 136
- B.5 TS-412 is a powerful yet easy to use networked storage center for backup, synchronization and remote access. It supports comprehensive RAID configuration and hot-swapping to allow hard drive replacement without system interruption. 136
- B.6 LaCie 4Big Quadra: Noctua magnetic levitation cooling fan: high-performance, quiet, zero-vibration. 32MB cache (or greater) hard disks. 137

Chapter 1

Introduction

Cancer is a group of diseases involving abnormal cell growth with the potential to invade or spread to other parts of the body. Many treatment options for cancer exist. The primary ones include *surgery*, *chemotherapy* and *radiation therapy*. Which treatments are used depends on the type, location and grade of the cancer as well as the patient's health and preferences. *Surgery* is the primary method of treatment for most cancers and may play a role in prolongation of survival. It is typically an important part of definitive diagnosis and staging of tumors, as biopsies are usually required. In localized cancer, surgery typically attempts to remove the entire mass along with, in certain cases, the lymph nodes in the area. For some types of cancer this is sufficient to eliminate the cancer. *Radiation Therapy* involves the use of ionizing radiation in an attempt to either cure or improve symptoms. It works by damaging the DNA of cancerous tissue, killing it. *Chemotherapy* is the treatment of cancer with one or more cytotoxic anti-neoplastic drugs (chemotherapeutic agents) as part of a standardized regimen. While surgery and radiotherapy are the primary treatment used for local and non-metastatic cancers, anti-cancer drugs (chemotherapy, hormone and biological therapies) are the choice currently used in metastatic cancers.

Cancer is a disease of evolution [34, 35, 36, 37], see Figure 1.1. Its initiation and progression is caused by dynamic somatic alterations to the genome manifested as point mutations, structural alterations, DNA methylation and histone modification changes [38, 39]. A cell, through mutations, acquires the ability to ignore anti-growth signals from the body, this cell may thrive and divide, and its progeny may eventually dominate part of the tumor. This clonal expansion can be seen as a discrete state of the cancer's progression, marked by the acquisition of a set of genetic events. Cancer progression can then be thought of as a sequence of these discrete steps, where the tumor acquires certain distinct properties at each state. Resistance to chemotherapy and molecularly targeted therapies is one of the major issues now in cancer research. The mechanisms of resistance share many features, such as alterations in the drug target, activation of survival pathways and ineffective induction of cell death [40].

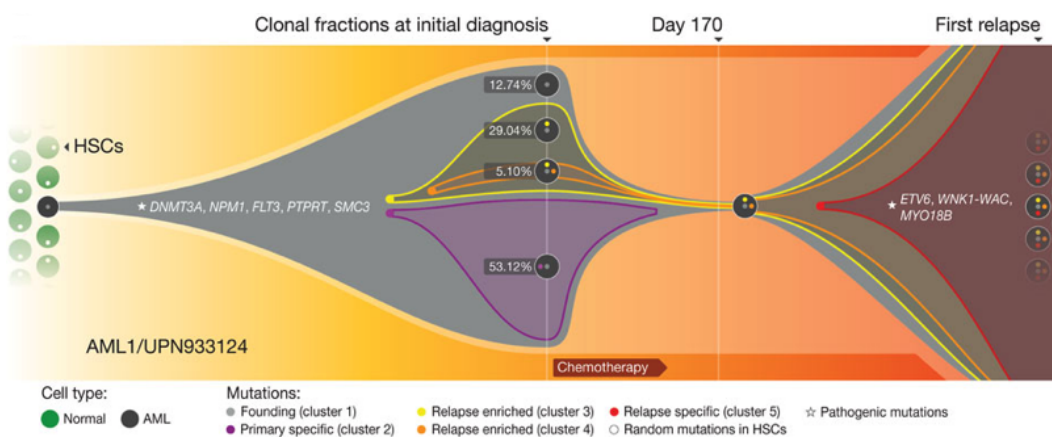


Figure 1.1: The founding clone in the primary tumor in AML1 contained somatic mutations that are all recurrent in AML and probably relevant for pathogenesis; one subclone within the founding clone evolved to become the dominant clone at relapse by acquiring additional mutations, including recurrent mutations [17]. This plot can be created with FishPlot R package [18].

1.1 Gene Therapy: Insertional Mutagenesis

A lot of strategies have been devised in order to identify the culprits of drug resistance [41, 42] including association between the genomic mutation landscape and the sensitivity/resistance profiles of clinical cases, in vivo and in vitro induction of spontaneous resistance upon chronic drug administration/exposure. The main disadvantages of these strategies are the difficulty to distinguish between driver and passenger mutations on a subset of known genes [43].

Gene therapy [44, 45] is the therapeutic delivery of genetic material into patient's cells (Figure 1.2¹) to treat disease. The first attempt, an unsuccessful one, at gene therapy (as well as the first case of medical transfer of foreign genes into humans not counting organ transplantation) was performed by Martin Cline on 10 July 1980 [46]. Cline claimed that one of the genes in his patients was active six months later, though he never published this data or had it verified and even if he is correct, it's unlikely it produced any significant beneficial effects treating β -Thalassemia. After extensive research on animals throughout the 1980s and a 1989 bacterial gene tagging trial on humans, the first gene therapy widely accepted as a success was demonstrated in a trial that started on 14 September 1990, when Ashanthi DeSilva was treated for ADA-SCID [47]. Here at SR-Tiget, we treat several pathologies, such as Metachromatic Leukodystrophy (MLD) [19, 48], Wiskott Aldrich Syndrome (WAS) [49], Mucopolysaccharidosis type 1 (MPS1) [50], β -Thalassemia [51, 52]. To delivery

¹<http://www.yourgenome.org/facts/what-is-gene-therapy>

the therapeutic material we use generally *Lentiviral Vectors* (LV) that are based on HIV virus, but rendered harmless. Integration into host genome, the distinctive feature of retroviral vectors, should be considered as a *double-edged sword* when it comes to gene therapy [53]. Genomic integration ensures the stability of transgene (material that has been transferred naturally, or by any of a number of genetic engineering techniques from one organism to another) and persistent transgene expression in daughter cells following genome replication and cell division, but its randomness results in the risk of *insertional mutagenesis* by potentially disrupting tumor suppressor genes or activating oncogenes [54].

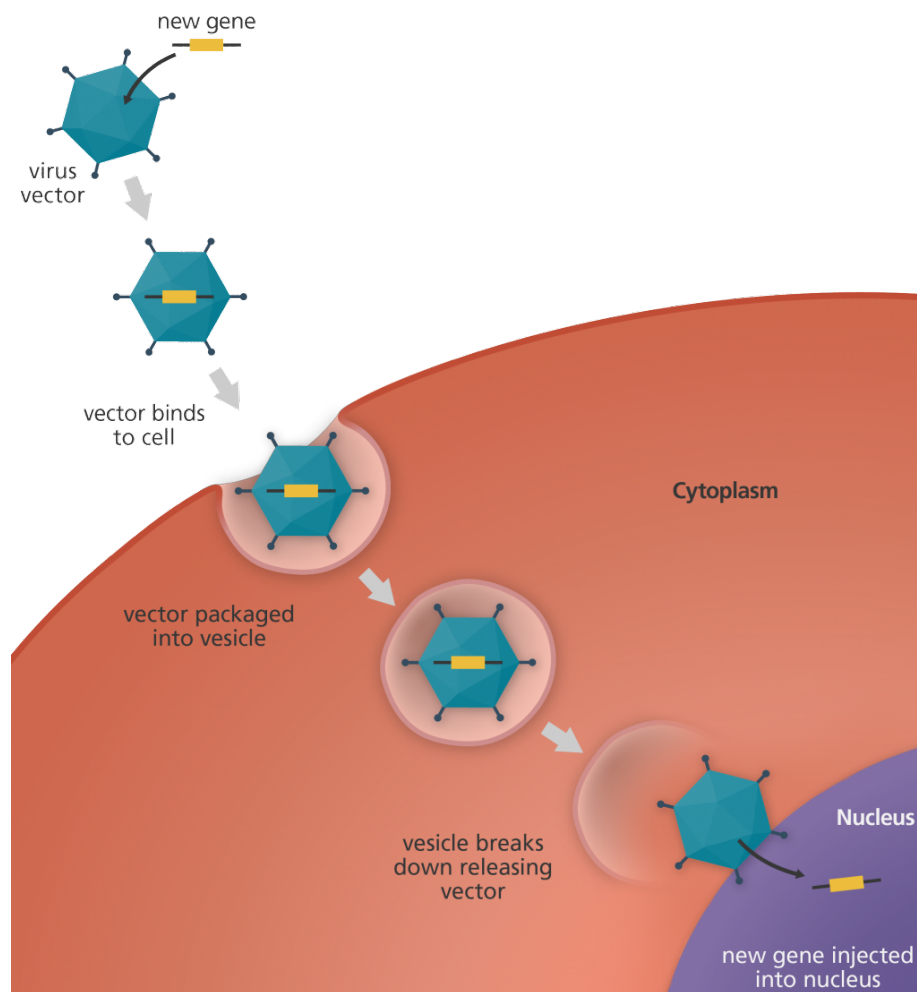


Figure 1.2: How gene therapy works: the therapeutic lentiviral vector inserts the new gene into a cell. If the treatment is successful, the new gene will make a functional protein to treat a disease.

Insertional mutagenesis refers to mutation of an organism produced by the insertion of additional DNA material into the organism's preexisting DNA [55]. This

process leads to the deregulation of genes in the neighborhood of the insertion sites and causes a perturbation of the cell phenotype, that can result into the rise of cancer. On the other hand, insertional mutagenesis is a forward genetic approach that has been used for the functional identification of novel genes involved in the pathogenesis of human cancers [1]. The use of LVs (contrary to *Retroviral Vectors*, γ -RV) and toxicity studies have allowed to obtain vectors now much safer and therefore able to be engineered perfectly for both behaviors (to treat a single-gene disease or to induce cancer). Thus is now possible to induce insertional mutagenesis in vitro, to study some interesting paths of cancer related diseases, like anti-cancer drug resistance.

1.2 Retrieving Vector Integration Sites

Analyzing the integration profile [6] of lentiviral vectors (Integration Sites, IS) is crucial in determining their potential genotoxic effects and also important for mutagenesis screenings. These IS act as molecular tags, enabling the detection of lentiviral vectors insertion through methods such as linear amplification-mediated (LAM)-PCR [56]. Sequence reads that contain the IS must be mapped to the human genome. For this reason we should have a custom pipeline, compliant with the new big data standard and simple to use, also for non computer scientists.

1.3 Cancer Progression Models

The identification of highly targeted genome areas allow us to identify the driver resistance genes. With these, to investigate in more detail the biology and function, we want to try to reconstruct the clonal progression over time. This technique can be implemented with many algorithms, which allow us to create relationships between genes and display the network obtained in terms of graphs, but will be discussed later in the thesis. The generation of the progression graph will be very helpful to begin a study of enrichment on the relationships between genes, to confirm the results obtained and to open new areas for consideration [57].

1.4 Computational Requirements

Right now (at SR-Tiget) we have a lot of data available, that it is not only related to cancer, notable, both the management that the analysis of this huge amount of FASTQ generated by Illumina HiSeq sequencing platforms, are becoming increasingly complex and costly. We are now in the big data era, datasets are growing rapidly, for this reason we are now scaling to Cluster architectures and design or optimize each software/tool to have the best performances in terms of time and space.



Figure 1.3: The Big Data era, WordCloud.

1.5 Aim of this Work

Unfortunately, the identification of driver mutations and thus the mechanisms underlying anti-cancer drug resistance (ACDR) still remains a challenge. My laboratory previously demonstrated that lentiviral vectors (LVs), when properly modified, might integrate near specific genes, alter their expression and induce cancer or ACDR in vivo and in vitro [1, 3]. The analysis of vector-cellular genomic junctions in tumor or ACDR cells allowed identifying causative genes of HER2+ breast cancer cell line using a statistical approach defined Common Insertion Sites (CISs) that highlight genomic regions targeted at significantly higher frequency than expected by a random distribution. The reconstruction of cumulative cancer progression from CIS genes has not been yet addressed and may produce causative gene networks. *The aim of this project is studying anti-cancer drug resistance from exclusive and co-occurring genes using cumulative cancer progression from our cell line CIS genes and investigating the relation between them.*

Chapter 2

State-of-the-Art

2.1 Integration Site Retrieval Tools

Integration site retrieval is the analytical process to identify unambiguous genomic loci in the reference genome where the virus is integrated. In literature there are not so many integration site retrieval pipelines. The major used are: Mavric [58], SeqMap [59], QuickMap [60] and VISPA [7]. All available pipelines have been specifically designed to analyze DNA fragments generated by linear-amplification (LAM) mediated PCR [56], a technique used to retrieve and amplify DNA fragments containing the junctions between the integrated proviral and the cellular genome, see Appendix A. The DNA fragments generated with this method have length up to 3,000bp (base pair) and contain the proviral long terminal repeat (LTR), the flanking genomic DNA and a linker cassette (LC). LAM-PCR products are then reamplified by PCR with fusion primers containing a specific 8-nucleotides sequence (barcode) that acts as a tag to allow sample recognition after multiplexing. Barcoded fragments are then purified, quantified, grouped into pools and sequenced with either Illumina MiSeq or HiSeq platforms. As a result of this procedure, the sequencing reads contain not only the genomic fragment needed for IS identification, but also viral and artificial sequences that must be trimmed out before alignment to the reference genome. Finally, sequencing reads must be processed by a bioinformatics pipeline that yields the final list of annotated ISs.

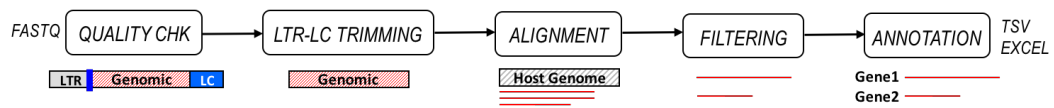


Figure 2.1: General IS retrieval pipeline.

Usually, a generic IS pipeline is represented in Figure 2.1 and it is composed of

few essential steps. First of all a good practice is to check the quality of Illumina MiSeq or HiSeq FASTQ. Immediately after checking the quality of the run, if it is good, the next step is to process data starting from the cut of the adapters, which are for sequencing or amplification (i.e. LAM-PCR) processes. Awarded the cleaned genomic sequence, it is necessary align it to the reference genome using an aligner (such as BLAST, Bowtie, BWA, GEM), then to perform some filters to remove bad quality mapped reads and the final annotation for gene names of integration sites. As reported before, in [7] with the first version of our pipeline at SR-Tiget (VISPA), we obtained good performances in terms of true positive rate (TPR) against the false positive rate (FPR) at various threshold settings, compared to the other three tools. We used a test set of sequences generated *in silico* to (1) reduce the computational time, since most of the published IS tools are available with limit in file transfer and with a priori unknown resource allocation, and (2) to compute statistical assessment, since assessing false positive values and thus compute statistical measures requires to have reference IS known a priori (same dataset of [7] for input test sequences in FASTA file format).

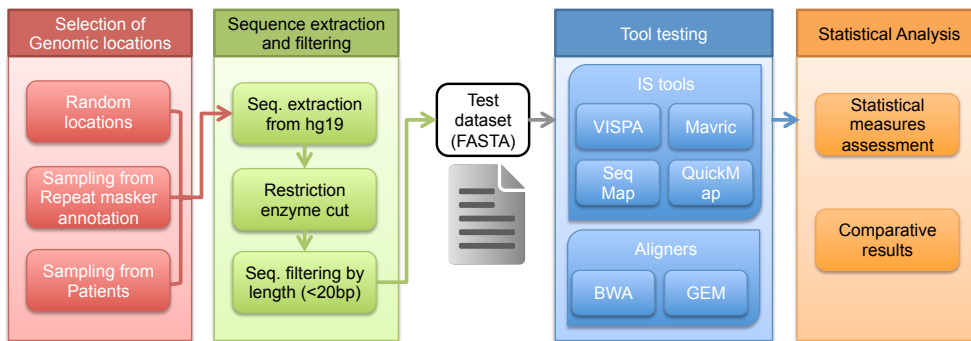


Figure 2.2: Evaluation of VISPA related to the other IS analysis tools. (A) Overall strategy for reliability assessment, from the generation of the simulated dataset to the final results.

We exploited the two most used NGS aligners, BWA [61] and GEM [62], as reference to verify the genomic mappability of the test sequences in the target genome, thus allowing the classification of each sequence as a repetitive element or not. The evidence of multiple best matches in the alignment is provided in the output SAM file [63] by the tags AS (alignment score) and XS (suboptimal alignment score). The classification of each sequence as repeat or unique position in the genome can be identified with two tags (XS/AS) that highlights the homology ratio between the best alignments. By applying a threshold to the homology ratio, is possible to separate the test sequences in two groups: IS to accept and IS to reject (because identified as repeats).

Genomic Distribution of Simulated Loci

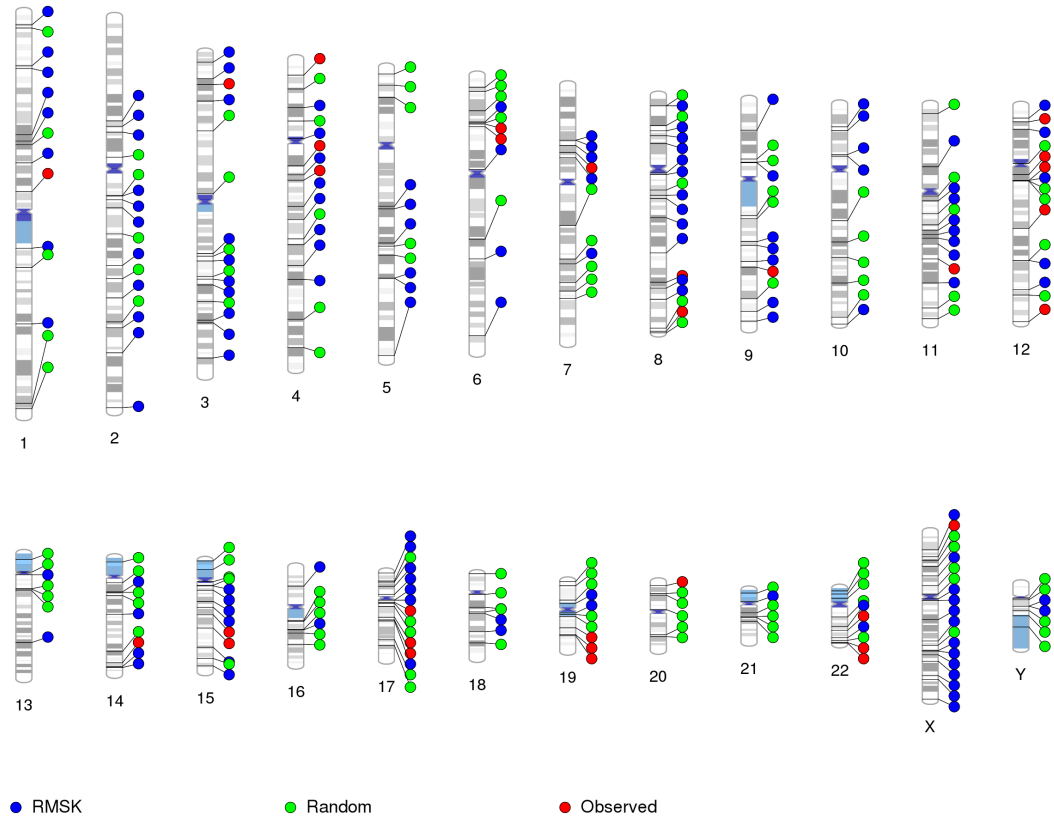


Figure 2.3: Genomic distribution of the 283 randomly selected loci. Each point represents a locus in the genome and its color refers one of the three categories (“RMSK” for repeat marker group, random or observed).

To create a simulated dataset of IS we first downloaded the human reference genome (hg19 GRCh37¹, from chromosome 1 to chromosome Y) from UCSC, and the annotated database of repeat masker regions. Then we selected 283 genomic locations (Figure 2.3) as surrogate of IS following three categories:

1. 120 IS randomly selected in the genome, exploiting MS excel in the range of each chromosome length obtaining a comparable number of loci for each chromosome (between 4 and 5 loci for each chromosome)
2. 132 IS randomly selected within the repeat masker annotation dataset: we decided to include IS derived by an annotation database of low complexity and repeats regions to add regions with potentially multiple matches in the genome.
3. 31 IS randomly selected from a patient of our MLD clinical study [19].

¹UCSC, <http://hgdownload.soe.ucsc.edu/goldenPath/hg19/bigZips>

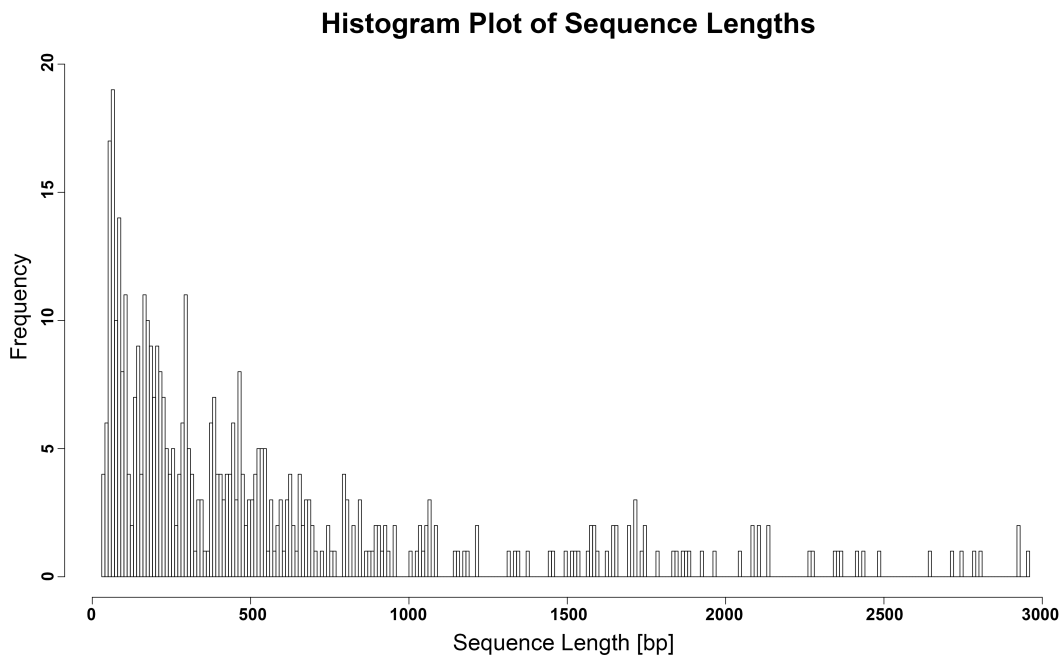


Figure 2.4: Density plot of the sequence lengths for the test dataset.

For each IS, we extracted from the reference genome the first 3,000bp starting from the locus to both orientations (forward and reverse) obtaining two reference sequences that share the same IS, thus retrieving a set of 566 sequences. We then cut each of the 566 sequences with the enzyme Tsp, that is one of the enzymes used in MLD and WAS clinical trials and that recognizes the genomic sequence AATT, simulating what we can observe in a real case IS dataset. We only discarded cut sequences shorter than 20 bp (Figure 2.4 for sequence length distribution). We finally obtained a test set composed by 253 IS with relative remaining sequences for a total amount of 455 sequences.

For example, in our dataset of 455 sequence, using a threshold for the homology ratio of 90% (that means that two alignments are considered repeats if and only if the ratio between the alternative alignment and the best alignment is ≥ 0.9 or 90%), we retrieved a number of 449 IS to accept and 6 IS to rejects because repeats. Thinking on a real dataset, and on the human genome (that is composed for an half of repetitive elements [22]) we must discard, up to now, a relative huge part of the total IS. This is one of the major issue of integration site analysis. A sequence is labeled as discarded if it is not returned as mapped IS (or directly discarded); on the other hand, each sequence, returned as IS, is tagged as matched if the chromosome and the genomic position are correct (with a tolerance in the position of ± 2 bp), otherwise it is classified mismatched/wrong. To evaluate the performances of all the four tools,

labeling:

- **True positive values (TP)** are all sequences returned by a tool as IS that we labeled as IS
- **False positive values (FP)** are all returned IS that we labeled as repeats and all mismatched IS
- **False negative values (FN)** are all returned repeats that we labeled as IS
- **True negative values (TN)** are all returned repeats (discarded sequences) that we classified as repeats

Given the previous definitions and P as the group of positives and N the group of negatives, it is possible to define:

- **True Positive Rate (TPR) or Sensitivity or Recall:**

$$TPR = \frac{TP}{P} = \frac{TP}{(TP + FN)} \quad (2.1)$$

- **Specificity (SPC) or True Negative Rate:**

$$SPC = \frac{TN}{N} = \frac{TN}{(FP + TN)} \quad (2.2)$$

- **False Positive Rate (FPR):**

$$FPR = \frac{FP}{N} = \frac{FP}{(FP + TN)} = 1 - SPC \quad (2.3)$$

- **Positive Predictive Value (PPV) or Precision:**

$$PPV = \frac{TP}{(TP + FP)} \quad (2.4)$$

The Receiver Operating Characteristic (ROC), or ROC curve, is a graphical plot that illustrates the performance of a binary classifier system as its discrimination threshold (homology) is varied.

Since TPR is equivalent to sensitivity and FPR is equal to $1 - specificity$, the ROC graph is sometimes called the sensitivity vs $(1 - specificity)$ plot. Each prediction result or instance of a confusion matrix represents one point in the ROC space.

Specificity (SPC)				
Homology (%)	VISPA	Mavric	SeqMap	QuickMap
20	0.37	0.222	0.889	0.148
30	0.37	0.222	0.889	0.148
40	0.37	0.222	0.889	0.148
50	0.37	0.222	0.889	0.148
60	0.625	0.313	0.875	0.25
70	0.667	0.417	0.833	0.25
80	0.889	0.556	0.778	0
90	1	0.833	0.667	0
100	1	1	0.6	0

Table 2.1: Specificity data for all IS tested tools.

Sensitivity (TPR)				
Homology (%)	VISPA	Mavric	SeqMap	QuickMap
20	0.946	0.785	0.682	0.956
30	0.946	0.785	0.682	0.956
40	0.946	0.785	0.682	0.956
50	0.946	0.785	0.682	0.956
60	0.948	0.788	0.667	0.957
70	0.944	0.79	0.661	0.955
80	0.944	0.791	0.657	0.948
90	0.94	0.793	0.653	0.949
100	0.938	0.793	0.651	0.949

Table 2.2: Sensitivity data for all IS tested tools.

False Positive Rate (FPR) or 1-SPC				
Homology (%)	VISPA	Mavric	SeqMap	QuickMap
20	0.63	0.778	0.111	0.852
30	0.63	0.778	0.111	0.852
40	0.63	0.778	0.111	0.852
50	0.63	0.778	0.111	0.852
60	0.375	0.678	0.125	0.75
70	0.333	0.583	0.167	0.75
80	0.111	0.444	0.222	1
90	0	0.167	0.333	1
100	0	0	0.4	1

Table 2.3: False Positive Rate (FPR) or 1-SPC data for all IS tested tools.

ROC analysis provides tools to select possibly optimal models and to discard sub-optimal ones independently from (and prior to specifying) the cost context or the class distribution. ROC analysis is related in a direct and natural way to cost/benefit analysis of diagnostic decision making. In this case, to select the best pipeline, using the Equations 2.1, 2.2, 2.3 and 2.4, we obtain the Tables 2.1, 2.2 and 2.3 useful to plot the ROC curve.

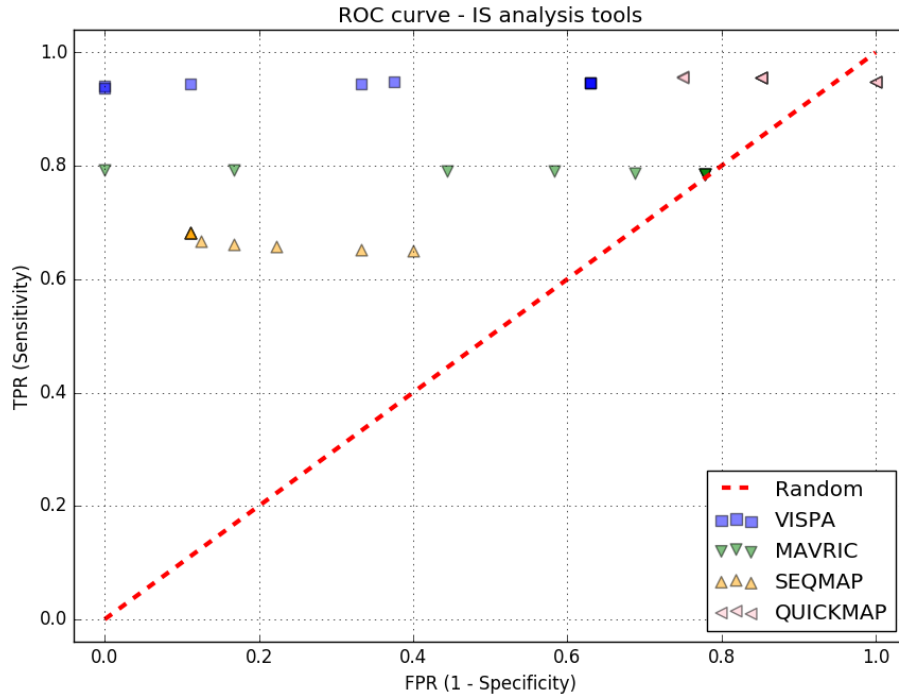


Figure 2.5: ROC curve of VISPA, Mavric, SeqMap and QuickMap.

Pipeline	CPU time	User Interface	Paired-End	Repeats
Mavric	30m	web	no	no
SeqMap	30m	web	no	no
QuickMap	1h	web	no	no
VISPA	30m	web/bash	no	no

Table 2.4: General Overview of the tested pipelines: test done on simulated reads described before.

We also analyzed all mismatches in relation to the reference genomic position for a distance $>2\text{bp}$ (Figure 2.6). Since VISPA did not report any mismatches, we only reported here results from MAVRIC, SeqMap and QuickMap. All tests demonstrated that VISPA is highly reliable and that the internal parameters chosen are well balanced to obtain precise results with high levels of sensitivity and accuracy. Moreover in Table

2.4 we summarized also the performance results in time and the possibility to handle paired-end² and repeats data.

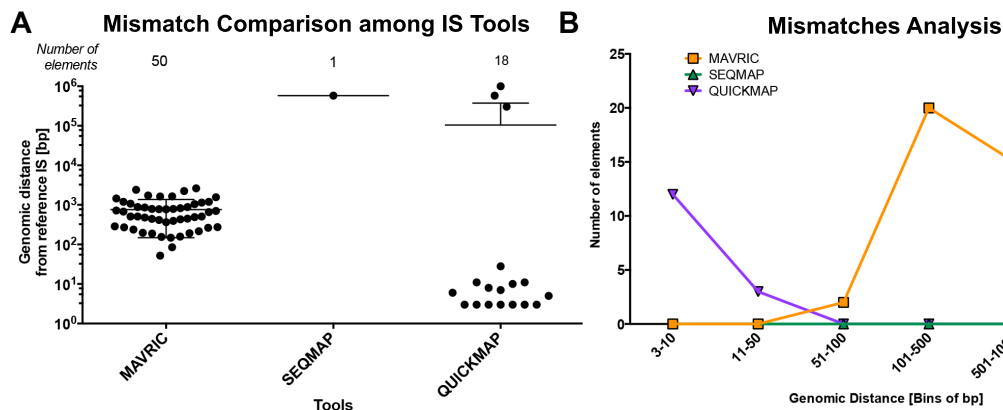


Figure 2.6: Mismatch Analysis: (A) Box-plot of the distances in terms of genomic position (bp) between each mismatched IS and the reference IS, for each tool (Mavric, SeqMap and QuickMap). (B) Binning the genomic distances of the mismatched IS in intervals, each point accounts for the sum of IS for each bin, comparing the three tools.

2.1.1 A New Improved Version of VISPA

Compared with the other pipelines, VISPA has good performance in terms of precision (correctly identified integrations) and recall (number of integrations obtained), but at present lacks of important features.

The standard of Illumina sequencing platforms (MiSeq and HiSeq) now includes the use of paired-end mode. This technique requires that the DNA strand is sequenced from both directions, generating two FASTQ, one called R1 (5'-3') and the other R2 (3'-5'). Aligners, such as BWA, already provide the paired-end alignment mode, which uses both sequences to greatly improve the final alignment. This feature is therefore essential, to refine the accuracy and to be especially compatible with the current standard.

In terms of performance, however, taking more and more present that we are going through the era of the *Big Data*, VISPA, like other tools, needs a lot of improvements to make it much faster. Especially the management of HiSeq runs (which on average generate compressed FASTQ of ~60GB, and about 180M of reads) is virtually impossible, both for the space occupied by temporary files, both from the time of calculation.

²Paired-end sequencing allows users to sequence both ends of a fragment and generate high-quality, alignable sequence data. Paired-end sequencing facilitates detection of genomic rearrangements and repetitive sequence elements, as well as gene fusions and novel transcripts. Taken from *Illumina*.

The *Graphical User Interface* (GUI) in Galaxy, although it allows interfacing VISPA with many NGS tool, is very slow and not very intuitive. It would be appropriate to make the GUI much more intuitive and easy to use, especially by non-IT users.

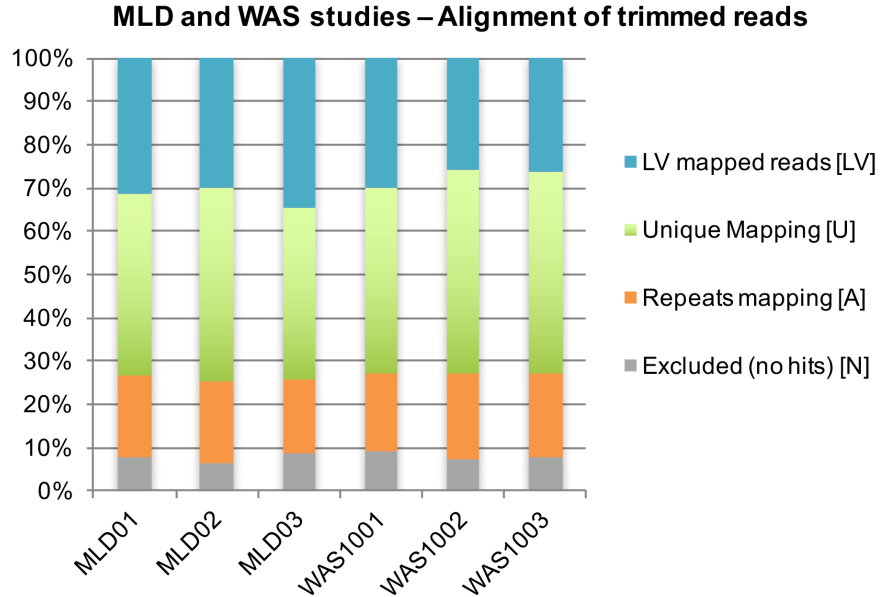


Figure 2.7: Bar-plot of the relative percentages of aligned reads for each patient in [19]: the input sequences are the trimmed reads, whereas the resulting subsets from the alignment step are LV, U, A, and N.

As already highlighted in many studies and common problem of all alignment bioinformatics pipelines, the management of repetitive elements in the genome still has not been resolved, often not addressed. Virtually every tool, at the time, does nothing and discard reads that map in these areas, since no aligner can handle them. As it is possible to see in Figure 2.7 (created using VISPA [7]), about 20% of total reads map on repeated areas. If from this percentage are removed those that map into the genome of the vector, the percentage refers to the human genome is about 30%, a very high percentage.

2.2 Cancer Progression Modeling

The inference of progression graphs from cross-sectional data is a difficult task. Disease progression models have been mainly applied to HIV and cancer data, with the possibility to extend the methodology to other evolutionary diseases. In a comprehensive review [21], the methods are grouped in: (1) *Linear Path Models*, (2) *Oncogenetic*

Tree Models, (3) *Distance-based Trees*, (4) *Conjunctive Bayesian Networks and Directed Acyclic Graphs*. Apart from the performance and accuracy of the different approaches, the essential point in this case is whether the method actually accurately describes the progression of event. Also the other essential aspect is the data type of input that the method can receive at its input, which should be as general as possible, adaptable to the greatest number of possible studies. On this aspect now I want to compare, very briefly, these different algorithms. There are also some theories and models on the formation of cancer (*carcinogenesis*) [64, 65, 66, 67, 68] which is not subject of this work. In this field, taking account on mutation only data, there is a model [69], derived by [70], that is able to couple between evolution and ecology to study tumour dynamics.

2.2.1 Linear Path Models

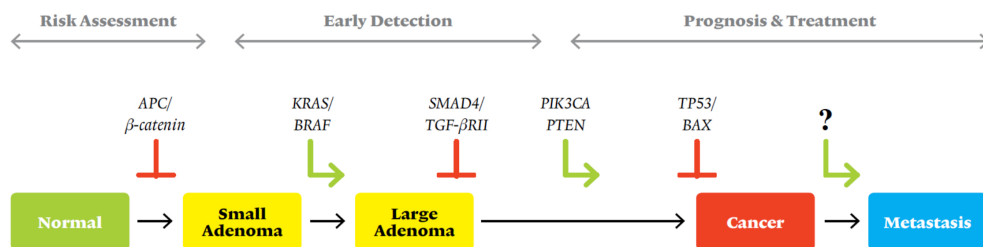


Figure 2.8: Taken from *Howard Hughes Medical Institute*. Model of Colorectal Cancer, described in [20]

By analyzing the occurrence of genetic alterations in colorectal cancer, Vogelstein [20] found a sequence of alterations that paralleled the clinical progression of colorectal adenomas and carcinomas, Figure 2.8. The same algorithm is too reductive to the great heterogeneity of human tumors, both for the difficulty in finding individual subsequent mutations, both for the direct causality between the one and the other, without taking into account the non-sequential nature of most of the human mutations. For this reason, attempts to find similar progression models in other types of cancer have not been successful yet, indicating that such *linear path models* are exceptions rather than the rule in cancer biology.

2.2.2 Oncogenetic Tree Models

Based on the idea of linear path models, the *oncogenetic tree models* [71] do no longer use only one path of genetic events to characterize disease progression but several ones, Figure 2.9. Multiple pathways can be represented in a tree, that means pathways can be parallel to each other or branch out and therefore capture dependent and

independent events. A rooted tree is defined by $T = (V, E, r)$, where V is the set of nodes (genetic events), $E \subseteq V \times V$ is the set of directed edges (relationships between events) and $r \in V$ is the root of the tree (starting point of the disease). Directed edges are represented as the tuple (a, b) , with starting point a and end b . In a rooted tree, there is a directed path from r to every node $v \in V$. The conditional probability of observing the child event ($S \subseteq V$) given that the parent event has already occurred is given as an edge weight which is associated with the corresponding directed edge, which represents the dependency between parent and child.

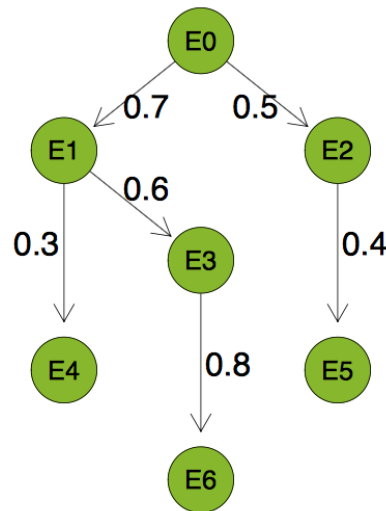


Figure 2.9: Taken from [21], Example of an oncogenetic tree model with $n = 6$ events

Oncogenetic tree models are not as restrictive as path models are. They allow for multiple pathways and can therefore model disease progression in a more flexible way. But some patterns of events are still missing. If such an observation is made nevertheless, the tree model is useless and wrong. Furthermore, the probability of occurrence only depends on the direct predecessor, no other events have influence and only one parent is allowed.

2.2.3 Distance-Based Tree Models

Besides oncogenetic trees, [71] also suggests a tree model based on distances between events to model disease progression, Figure 2.10. The underlying structure is still a tree, but genetic events are only represented by leaf-nodes or leaves, which are nodes without children. The nodes in between root and leaves are inner nodes and represent arbitrary unknown events which cannot be observed. A distanced based tree is specified by $T = (V, E, r, \alpha, L)$ where in addition to an oncogenetic tree $L \subseteq V$ is a

nonempty set of leaves, i.e. the set of genetic events. The advantage of this approach is that every combination of events has a positive probability. One does not have sets of events with probability 0. This model can therefore give information about the relationship between every two arbitrary events and not only for certain ones. Having events only in leaf-nodes does not give an order of occurrence for events as oncogenetic trees do. But one can convert the conditional probabilities to distances by calculating their negative logarithm. Summing up these values alongside the path to a certain leaf results in the distance between root and event. By comparing these distances one can distinguish between early and late events, because small distances refer to early occurrence.

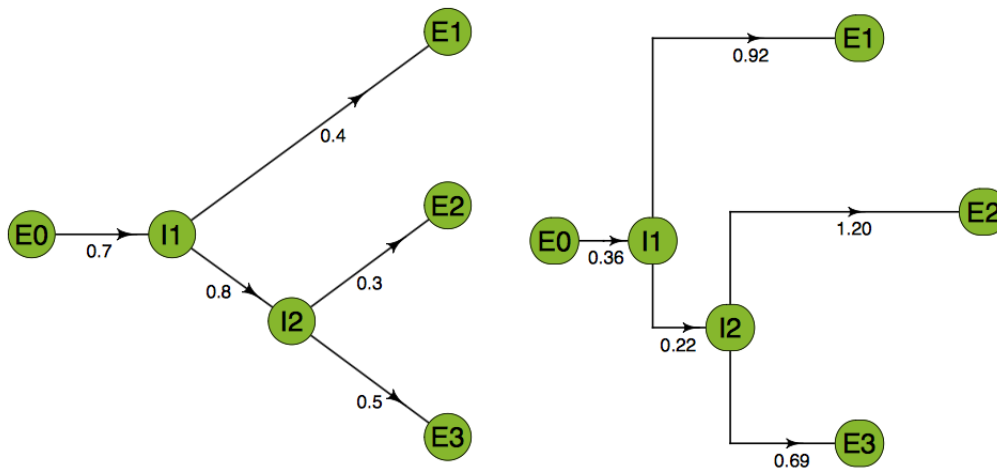


Figure 2.10: Taken from [21], Example of a distance based tree model with $n = 3$ events (left: conditional probabilities, right: distances)

In contrast to oncogenetic tree models, distance based trees do not require the occurrence of an event to be able to observe another one. That means, every combination of events has a positive probability and one does not need to cope with observations that do not fit the model. Instead there are inner nodes which represent unknown and non-observable events. One might question whether such events really exist.

2.2.4 Conjunctive Bayesian Networks and Directed Acyclic Graphs

In comparison to path models, tree models are able to model parallel or branching pathways. However, one restriction is that they do not allow for multiple parents. To model not only branchings but also conjunctions of edges respectively pathways one needs the broader class of *Directed Acyclic Graphs* (DAGs). Here, a node can have more than one parent-node. In [72] the authors use *Conjunctive Bayesian Networks*

(CBNs) that are a generalization of oncogenetic tree models, Figure 2.11. in which branchings of edges conjunctions are allowed. Thus, there is one node per event, no cycles and again the occurrence of the child-event depends on the occurrence of the parents. In this case multiple parents are allowed, that means every parent-event has to be observed before the child-event can occur. A CBN consists of a finite set of binary random variables, which stand for the genetic events, and a partial order, which gives their dependencies. Instead in [73] the authors also use directed acyclic graphs to model disease progression.

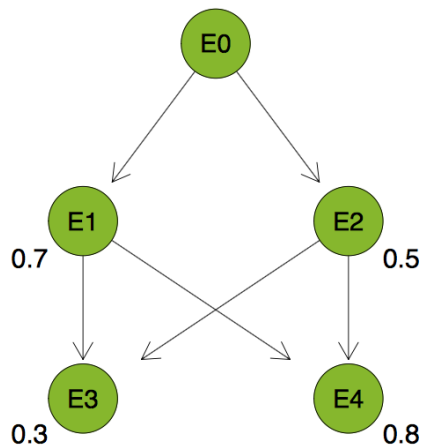


Figure 2.11: Taken from [21], example of a conjunctive bayesian network with $n = 4$ events

Chapter 3

VISPA2: Faster and Extended Version of Vector Integration Site Parallel Analysis Tool

Source: <https://bitbucket.org/andreacalabria/isatk>

Mercurial Repository:

```
hg clone -b 'v3' https://bitbucket.org/andreacalabria/isatk
```

As described in Chapter 2.1 mapping the cellular genomic portion to the reference genome allows to accurately locate IS on the genome, that is a crucial point for each analysis in gene therapy. For this reason I developed VISPA2, improving our old version of the pipeline and introducing a huge list of rising:

- Illumina paired-end (PE) reads support.
- Compliance with the new standard of Sonicated Linker-Mediated-PCR (SLiM), Appendix A.
- Quality filter on each pair of paired-end reads.
- Parallel computation of each Single VISPA2 Step (GNU Parallel):

```
cat file | parallel --pipe 'cat >{#}; my_program {#}; rm {#}'.
```
- Automated generation of final annotated IS matrix.
- Command line version and GUI version (in collaboration with CNR-ITB).

Thanks to all the improvements listed above, VISPA2 was able to obtain excellent performance both in terms of disk space used and memory. Even the usability of the tool has been cured, because in addition to the classic bash version, a Graphical User Interface (GUI) has been developed, allowing non bioinformatics users to use VISPA2 without a hitch. At the end of the chapter also will be presented the precision and recall tests, on the same simulated data sets used to test the previous version, [7].

3.1 Bioinformatics Pipeline

The bioinformatics pipeline, Figure 3.1, consists of several sequential steps that lead from raw sequencing reads to the annotated ISs.

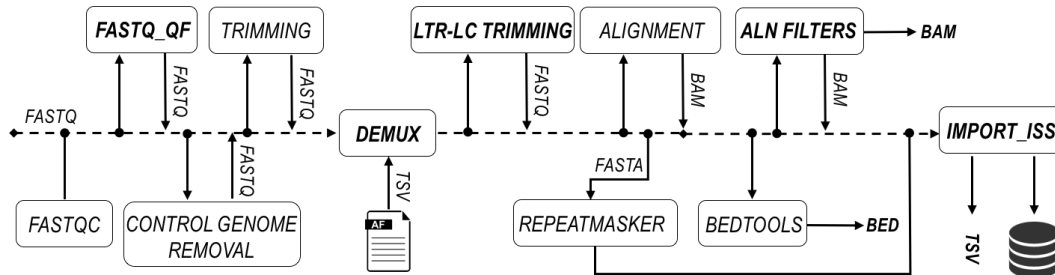


Figure 3.1: VISPA2, workflow. In bold custom programs.

The first step checks (*FASTQC*) and filters out the bad quality reads (*FASTQ_QF*); adapters and control sequences are removed (*CONTROL GENOME REMOVAL*, *TRIMMING*); sequencing data are then parsed (*DEMUX*) to identify barcodes and perform demultiplexing (that is, write a separate FASTQ file for each barcode); the LTR and LC sequences are subsequently removed from each read to isolate genomic fragments (*LTR-LC TRIMMING*); in the next step, reads are mapped to the reference genome (*ALIGNMENT*) and several filters are applied to avoid unambiguous alignment (*ALN FILTERS*); after that, all ISs are imported in a database structure for easy access and storage (*IMPORT_ISS*). In a subsequent post-processing step, each IS is associated to the LAM-PCR sample from which it was originally derived, allowing its assignment to a source (for example, peripheral blood, bone marrow and so on), cell type (for example, CD34, T cell, B cell, and so on), and time point after treatment. An example of launch is in Appendix C.1.1, the main program is `isatk/pipeline/illumina/VISPA2.IlluminaMiSeq.pipeline.sh` that is linked as `vispa2` command. In the following sections all main steps will be explored in detail.

3.1.1 Quality Controls and Filters

First of all VISPA2 checks quality of raw sequence data coming from high throughput sequencing with FastQC [74], to provide graphical data reports. To filter out reads with low quality there are a huge list of third-party software but I decided to create a custom quality filter specific for our kind of fragments (from LAM or SLiM -PCR), the bash script (`fastq_qf`, `isatk/script/fastq_qf.sh`, see Appendix C.1.2). It is interesting to see how the two possible filters can be applied to the pair of FASTQs. For the classical LAM-PCR (Appendix A) fragments, the filter consists in a window of 80bp

(it contains the 12bp random barcodes, 8bp of barcodes for demultiplexing of samples, 32bp of LTR and a small portion of the genomic content), Figure 3.2. This windows is created with a parallel trimming tool (to improve performances), `trimmomatic` [75] and `fastq_quality_filter` a part of FASTX-toolkit [76], used with a set of parameters (`-q 28 -p 95 -Q 33`) then the program extracts only the high quality reads with `fqextract_pureheader` (`isatk/script/fqextract.pureheader.py`).

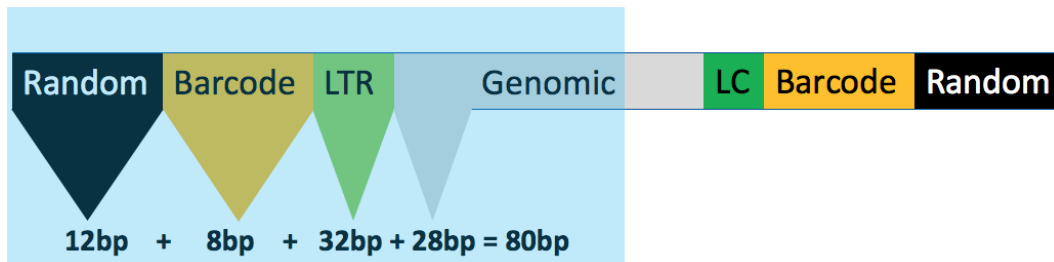


Figure 3.2: Quality filter for LAM-PCR read

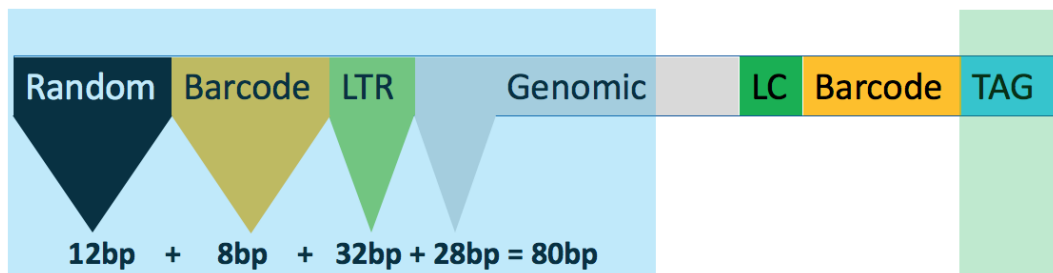


Figure 3.3: Quality filter for SLiM-PCR read

Instead, for the new SLiM-PCR (Appendix A), the filter is also extended for the random barcode (TAG sequence, 12bp), but, for better quality purposes it is checked on R2 pair, Figure 3.3. The process is the same described before to create the 80bp window with a window of 12bp for the TAGs (random barcodes at the beginning of R2, or at the end of R1) filtered with `fastq_quality_filter` (`-q 28 -p 100 -Q 33`). Indeed a list for high quality reads for 80bp of R1 and 12 bp of R2 is created but then the two lists are merged in one (with `comm` command in bash) and at the end only the good quality reads are maintained in the final FASTQs.

To test the goodness of the quality filter I run the new custom quality filter program on two different NGS sequencing runs (one with bad quality and the other with good quality, checked with FastQC), Figures 3.4, 3.5.

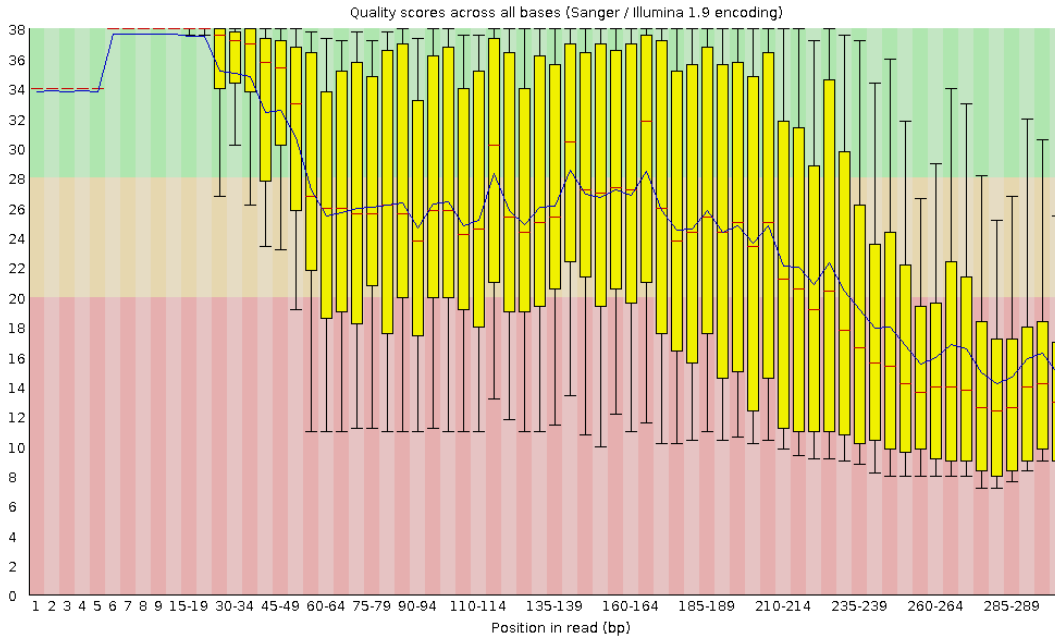


Figure 3.4: Bad run: FastQC on R1 for MLD6 pool ET06

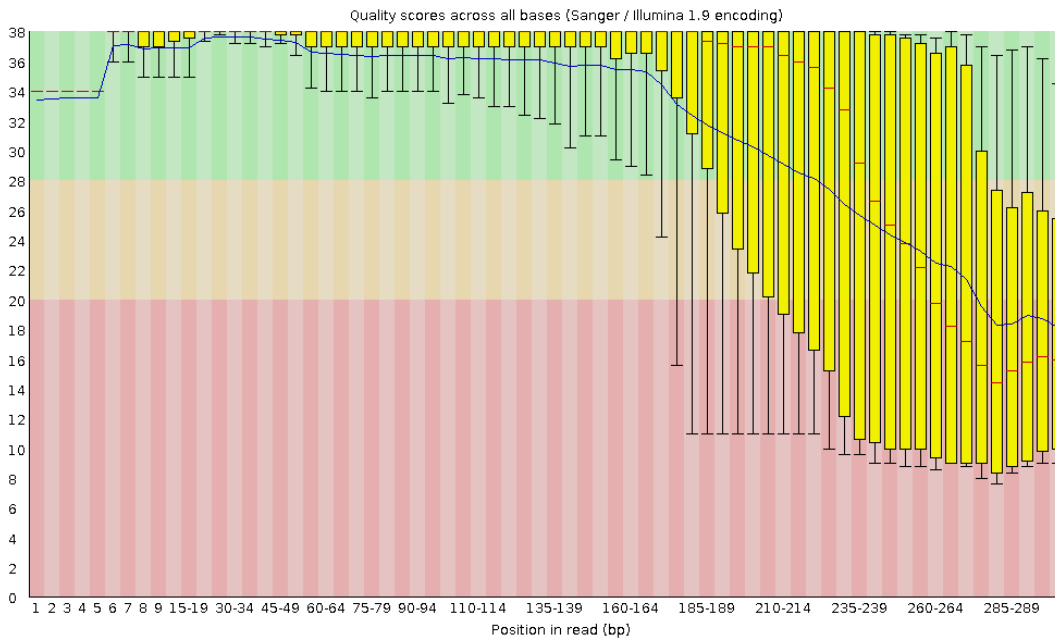


Figure 3.5: Good run: FastQC on R1 for MLD6 pool ET06v2

Comparison of MLD-ET6 runs		
Pool	Unique ISs	SeqCount
MLD06-ET6	33,203	4,941,050
MLD06-ET6v2	5,704	12,221,508
MLD06-ET6v2-HQ	4,774	10,479,042

Table 3.1: MLD06-ET6 runs comparison. Bad pool (MLD06ET6), correct pool (MLD06ET6v2) and filtered correct pool with the custom quality filter program.

In Table 3.1 I reported the results of three cases. The bad pool (MLD06ET6) has an huge number of unique integration sites and low sequence count (SC, number of reads in the same locus), a typical issue for a contaminated (i.e. from LAM, Sequencing, between samples) run. A second pool (MLD06ET6v2), was rebuilt and sequenced. In this scenario I used the quality filter program to test the parameters.

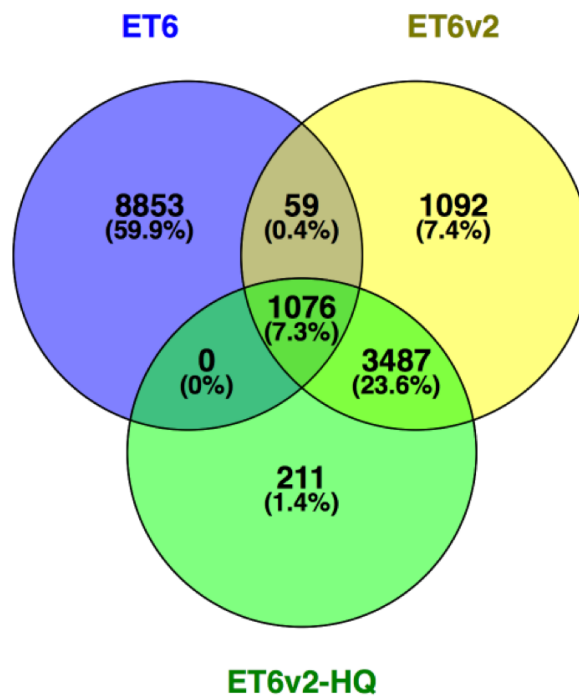


Figure 3.6: Venn diagram of the three runs to see the IS in common.

The ET6 pool, resulted with poor sequencing quality, presents a large number of additional ISs that have not been found then in the resequenced pool (ET6v2). This clearly suggests that the majority of these ISs are false positives (only the 7.7% of ISs are in common with the ET6v2). Instead, comparing ET6v2 and ET6v-HQ (with

high sequencing quality, and the subset of passing quality filter reads respectively) the improvement in applying the filter is evident (ET6v2: for the 1092 not in common with ET6v2-HQ 95% of these are with $SC < 3$ and 711 are in repeats, ET6v2-HQ: for the 211 not in common with ET6v2 80% of these are with $SC < 3$ and 147 are in repeats). The vast majority of the total ISs of ET6v2 and ET6v2-HQ are in common, correctly.

3.1.2 Adapter Removal and Trimming

Once the reads are filtered by quality the next step is to remove the reads mapping in PhiX and the first 12bp of random barcodes (not the TAGs).

PhiX is a reliable, adapter-ligated library used as a control for Illumina sequencing runs. It is also a quality control for cluster generation, sequencing, and alignment, and a calibration control for cross-talk matrix generation, phasing, and pre-phasing. Roughly for each run the amount of PhiX on the total is 30%. VISPA2 removes it aligning all the reads on PhiX genome with BWA-MEM (BWA with maximum exact match), producing a list of the reads that map on that genome and discard them using `fqextract_pureheader`. Generally the 30% of the total reads are removed for PhiX.

The first 12bp of R1 must be removed from the read (instead the first 12bp of R2 are the TAG and must be removed from the read but conserved in a FASTA file for the quantification, [77] and Appendix A). To improve the performances, in term of time, I select `trimmomatic` (as described before) to use parallelism in our computers (Appendix B.1).

3.1.3 Demultiplexing and Association File

Several samples are often sequenced at the same time, this technique is called multiplexing. To enable the redistribution of output reads into separate groups (demultiplexing), samples are tagged with individual barcode sequences.

I demultiplex out samples with `fastq-multx`, a part of the EA-Utils suite [78]. It identifies barcodes and uses them to demultiplex sequence data, producing a separate FASTQ file for each barcode. To demultiplex sequencing data, I developed a simple exact string pattern matching: the input is a list of barcode sequences (taken from the association file) that will be searched for at the beginning of each read (reads that do not contain any known tag are discarded). To avoid biases due to the possible misclassification of similar sequences, no mismatches are tolerated in this phase. All information regarding samples are written in the association file (AF, a tab-separated file). This file is created automatically from adLIMS [79], that I previously developed for our laboratory. The AF contains all the following fields, reported in Table 3.2.

3.1.4 Association File for Breast Cancer Experiment

Barcode	Barcode	Tissue	Sample	Treatment	LAM-ID	Complete Name	Cell Marker	Enzyme	Vector
---------	---------	--------	--------	-----------	--------	---------------	-------------	--------	--------

Table 3.2: Association file example, pool fb360, Chapter 6

LTR22.LC32	LTR22.LC32	S3L1	784H	1	#784-H	Exp_S3L1_SKBR3_#784_H_MOI_1_1	OuM_Lapatinib MOI_1	tsp	1, OuM_Lapatinib
LTR22.LC64	LTR22.LC64	S3L1	784I	1	#784-I	Exp_S3L1_SKBR3_#784_I_MOI_1_1	OuM_Lapatinib MOI_1	tsp	1, OuM_Lapatinib
LTR22.LC74	LTR22.LC74	S3L1	784F	10	#784-F	Exp_S3L1_SKBR3_#784_F_MOI_10_1	OuM_Lapatinib MOI_10	tsp	1, OuM_Lapatinib
LTR22.LC90	LTR22.LC90	S3L1	784B	100	#784-B	Exp_S3L1_SKBR3_#784_B_MOI_100_1	OuM_Lapatinib MOI_100	tsp	1, OuM_Lapatinib
LTR60.LC32	LTR60.LC32	B2L1	784A	0	#784-A	Exp_B2L1_BT474_#784_A_MOI_0_5_1	OuM_Lapatinib MOI_0,5	tsp	1, OuM_Lapatinib
LTR60.LC64	LTR60.LC64	B2L1	784B	0	#784-B	Exp_B2L1_BT474_#784_B_MOI_0_5_1	OuM_Lapatinib MOI_0,5	tsp	1, OuM_Lapatinib
LTR60.LC74	LTR60.LC74	B2L1	784A	5	#784-A	Exp_B2L1_BT474_#784_A_MOI_5_1	OuM_Lapatinib MOI_5	tsp	1, OuM_Lapatinib
LTR60.LC90	LTR60.LC90	B2L1	784A	50	#784-A	Exp_B2L1_BT474_#784_A_MOI_50_1	OuM_Lapatinib MOI_50	tsp	1, OuM_Lapatinib
LTR80.LC32	LTR80.LC32	S3L1	784H	1	#784-H	Exp_S3L1_SKBR3_#784_H_MOI_1_1	OuM_Lapatinib MOI_1	hpy	1, OuM_Lapatinib
LTR80.LC64	LTR80.LC64	S3L1	784I	1	#784-I	Exp_S3L1_SKBR3_#784_I_MOI_1_1	OuM_Lapatinib MOI_1	hpy	1, OuM_Lapatinib
LTR80.LC74	LTR80.LC74	S3L1	784F	10	#784-F	Exp_S3L1_SKBR3_#784_F_MOI_10_1	OuM_Lapatinib MOI_10	hpy	1, OuM_Lapatinib
LTR80.LC90	LTR80.LC90	S3L1	784B	100	#784-B	Exp_S3L1_SKBR3_#784_B_MOI_100_1	OuM_Lapatinib MOI_100	hpy	1, OuM_Lapatinib
LTR88.LC32	LTR88.LC32	B2L1	784A	0	#784-A	Exp_B2L1_BT474_#784_A_MOI_0_5_1	OuM_Lapatinib MOI_0,5	hpy	1, OuM_Lapatinib
LTR88.LC64	LTR88.LC64	B2L1	784B	0	#784-B	Exp_B2L1_BT474_#784_B_MOI_0_5_1	OuM_Lapatinib MOI_0,5	hpy	1, OuM_Lapatinib
LTR88.LC74	LTR88.LC74	B2L1	784A	5	#784-A	Exp_B2L1_BT474_#784_A_MOI_5_1	OuM_Lapatinib MOI_5	hpy	1, OuM_Lapatinib
LTR88.LC90	LTR88.LC90	B2L1	784A	50	#784-A	Exp_B2L1_BT474_#784_A_MOI_50_1	OuM_Lapatinib MOI_50	hpy	1, OuM_Lapatinib

The first two columns contain the barcodes list, for demultiplexing, the other fields all the metadata for IS analysis. Indeed is reported the tissue of the sample, a sample ID, the time point of the harvest, the LAM-ID for backlinks identification of the sample, a unique complete containing all required fields for MySQL non-blind identification of a specific sample, the cell marker, the enzyme used for the LAM reaction (if SLiM, this field is unused) and finally the type of vector used.

3.1.5 LTR/LC Trimming and Internal Control Band Removal

After the demultiplexing procedure and before the alignment to the reference genome it is necessary to remove the LTR and the LC from each sample, in R1 and in R2. The recognition and removal of the LTR and LC is done with `flexbar` [80], that enables accurate recognition, sorting and trimming of sequence tags with maximal flexibility, based on exact overlap sequence alignment. The software supports data formats from all current sequencing platforms, including paired-end reads. It maintains read pairings and processes separate barcode reads with multi-threading support. After that each pair of read is aligned with BWA-MEM [61] to the lentiviral vector genome. The reads that perfectly mapping on the vector genome are discarded from the FASTQ. Generally the 30% of the total reads are removed from internal control band.

3.1.6 Alignment to the Reference Genome

To find the exact location where the vector is integrated into the genome, sequencing reads must be mapped to a reference genome. I chose BWA-MEM because its good performances compared to BWA-ALN and Bowtie2 [81, 82].

VISPA2 gives to BWA-MEM the two pairs (R1 and R2) and then samtools [63], processed in the following way (for details see Appendix C.1.3):

1. BWA-MEM align R1 and R2 to the reference genome fixing the minimum seed length to 18 and an alignment score filter to 15^1 .
2. Samtools then filters out the reads that are unmapped, not primary alignment and marked as supplementary alignment.

Repetitive Element Annotation

Transposable elements (TEs; or “jumping genes”) are discrete pieces of DNA that can move within (and sometimes between) genomes during the evolution [22, 83, 84]. Approximately 45% of the human genome can currently be recognized as being derived from transposable elements, the majority of which are non-long terminal repeat (LTR) retrotransposons, such as LINE-1 (L1), Alu and SVA elements [22].

¹**Phred Quality Score:** Q is defined as a property which is logarithmically related to the base-calling error probabilities P . $Q = -10 \log_{10} P$.

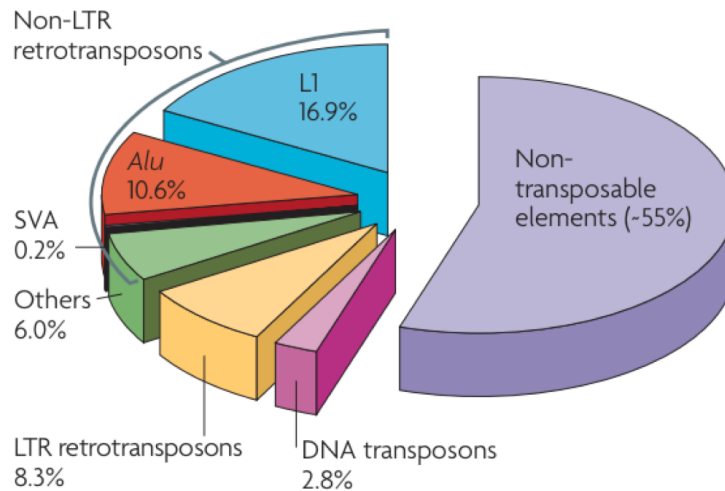


Figure 3.7: Repetitive families in human genome, [22].

L1 elements are $> 500,000$ different copies in the human genome as a result of their continued mobilization activity over the past 150 Million Years (Myr). L1 elements constitute $\sim 17\%$ of the human genome, which makes them the most successful TEs in our genome.

Alu elements are > 1 million different copies in the human genome as a result of their continued mobilization activity over the past ~ 65 Myr. This makes Alu elements the most successful TEs in our genome in terms of copy number.

SVA elements have been active throughout the ~ 25 Myr of hominoid evolution, and there are now $\sim 3,000$ copies in the human genome. A typical full-length SVA element is ~ 2 kb long and is composed of a hexamer repeat region, an Alu-like region, a region consisting of a variable number of tandem repeats.

Other non-LTR Retrotransposons are in addition to the L1, Alu and SVA elements, which are currently active, there are families of old, inactive non-LTR retrotransposons that comprise $\sim 6\%$ of the human genome. Although they are far less numerous than L1 and Alu elements, these elements provide a rich molecular 'fossil record' that testifies to the long relationship between TEs and the our genome.

Current bioinformatics pipelines for IS detection (Chapter 2.1) can efficiently analyze hundreds of millions of reads containing junctions between the proviral and the host genome, but considering only sequences that align to unique genomic positions, which can be then easily merged to a single IS. For those IS landing in repetitive genomic elements, which cannot be precisely mapped, it is difficult to understand if they represent a single or multiple IS. For this reason, sequencing reads mapped to multiple genomic regions are commonly discarded from the analyses. However, in human Hematopoietic Stem Cells (HSCs) Lentiviral Vector IS within repeats amount

to about 35% of the entire IS dataset. Therefore, the loss of such significant portion of the dataset, and thus the corresponding clones, has a major impact on the clonal abundance estimations, reducing the power of clonal tracking analyses and limiting the ability to detect potentially malignant clones caused by IS landing in repeats within or near oncogenes and reducing the reliability of IS studies. As represented in Figure 3.1, the repeats analysis takes place after the reference genome alignment, taking account of the discarded reads from BWA-MEM (generally the 30% of the total reads). The reads that after the alignment to the reference genome have mapping quality (MQ) less or equal to 5 (68.377%) pass to RepeatMasker [85], a program that screens DNA sequences for interspersed repeats and low complexity DNA sequences. This value is calculated as optimal value from [86] as inflection point (second derivative equal to 0, because of twice differentiable function) of the accuracy curve in worst case (short reads):

$$f''(x) = \frac{d^2 f}{dx^2} = 0 \quad (3.1)$$

The process is the following (using bamtools [87]; see Appendix C.1.4 for details):

1. Identification of low mapping quality reads after alignment ($MQ \leq 5$).
2. Filtering out R2 from BAM.
3. FASTA conversion from BAM file.
4. Execution of RepeatMasker on FASTA (engine=rmbblast), skipping bacterial insertion element check, multiprocessing option and quick search.
5. Filtering ISs (to remove false positives) with Smith-Waterman Algorithm (threshold of 250, see RepeatMasker manual² [88]).
6. Creation of a BED file from .out file, generated by RepeatMasker.
7. Importing of ISs into the relational MySQL database from BED file.

3.1.7 Filtering

The filtering part is composed of three different steps: (1) *filtering by mate* sequences, (2) *filtering by CIGAR* string and (3) *filtering by alignment quality*.

Filter aligned reads by mate pair properties

Paired-end reads alignment requires that mate reads are properly paired, means that R1 and R2 aligned in opposite orientation and with the last portion of the reads close to each other. In case of short sequenced DNA fragments, paired-end reads may also

²<http://www.animalgenome.org/bioinfo/resources/manuals/RepeatMasker.html>

overlap of a sequence portion or fully overlapping, otherwise, when sequenced longer DNA fragments, is present a the distance between the two reads called insert size.

In integration site studies, where the LTR is placed in one of the pair (such as R1 for our experimental design) and the LC is in the other pair, once aligned both pairs I analyzed paired results to grant:

1. Reads are properly paired.
2. If the DNA fragment is short enough to be sequenced identically from both pairs, the alignment of the genomic portion must be proper, following these rules:
 - (a) R2 alignment must not start within the alignment of R1.
 - (b) R2 must not end over R1 alignment start.
 - (c) If R1 alignment ends at R2 alignment start, then R2 end must be in R1 start.
 - (d) If R2 and R1 are fully overlapping then I also process the alignment score to identify potential issues in the sequence quality (suspicious divergence between the alignment scores of the pairs, that is: a full overlap of the aligned reads means that also the alignment quality should be similar, given the high quality of the reads).

If a read is not satisfying one or more of the rules, then the read is discarded. Since no existing tools are able to analyze mate properties with our custom details, I designed a new program to implement the rules. The software, `filter_by_mate` (`isatk/script/filter_by_mate.py`), has been developed in Python and leverages on PySAM [63] package to process BAM files. To speed-up performances, I developed the parallelization step using the chromosome selection such that each chromosome (or region) can be processed as an independent process.

Filter aligned reads by CIGAR and MD flags

In many genomics projects, aligned reads could be inspected by their properties embedded by the aligner in the optional flags (using the SAM/BAM file format, <https://github.com/samtools/hts-specs>). BWA [61], in the latest used algorithm, maximum exact match (MEM), fills standard mandatory flag fields, such as the CIGAR, and extra fields useful to better understand the alignment quality, such as the fields MD (mismatching positions/bases), AS (alignment score) and XS (secondary alignment score). The MD field is a detailed description of the mismatches reported in the CIGAR flag such that is possible to combine and use both flags to better characterize the mismatches and base changes (also insertions and deletions).

In vector integration site projects, the identification of an IS is critical and requires important rules to include only good alignments and avoid potential false positive IS. To achieve this goal, I envisage to divide in the following steps the analysis:

1. Analyze the beginning of the alignment and remove reads with mismatches (insertions, deletions or mismatches, or soft clipped alignments) within the first 3 bp. IS with any mismatches in the first 3 bp may arise from PCR artifacts or wrong trimming of the LTR portion. For this reason, given that the resolution of our IS identification is within a range of ± 3 bp, I defined a maximum accepted threshold of 3 bp at the beginning of the read as perfect alignment. Over this interval, I let the read alignment be processed by the subsequent filter by alignment scores.
2. Remove aligned reads if the alignment score of the best hit is highly similar the secondary alignment score. If a read returns two alignment matches with high scores, it means that the read cannot univocally be placed in the genome and the IS could be landed within a low complexity region or a repetitive element. In this case, a potential mis-assignment of the read to the correct position may result in a false positive and thus bias the biological analyses. To overcome this issue, I decided to apply a filter based on the alignment scores by comparing the best hit score with the secondary alignment score. The comparison produces a ratio that is evaluated by a parameter, here set up at 0.4: if the best alignment score is not better than the secondary score by 0.4, then the read is discarded. The value 0.4 comes from specific tests manually curated on experimental reads from murine samples landing in repetitive regions in proximity of the *Lrrc4c* gene.

Since no existing tools are able to process CIGAR flag nor comparing two or more tags from BAM files, I designed a new software and implemented the rules in Python. The program, called `filter_by_cigar_bam` (`isatk/script/filter_by_cigar_bam.py`), exploits the PySAM [63] library to read input BAM files (creating the index if missing), split in processes based on chromosomes (I decided to parallelize the code using chromosomes or input regions), and process reads by flags.

In the implementation of the first rule, the tool reads the CIGAR and the orientation (that drives the CIGAR reading orientation), and acquires the MD flag to best identify mismatches of indels. The tool discards the read if:

- At the beginning of the alignment, in the CIGAR flag, is found the presence of a soft clip (marked with an S).
- At the first bases of the alignment directly attached to the IS I identify by the MD flag any mismatch or indel. I developed this option as parameter (`--minStartingMatches`), and set up in default at 3.

I developed the tool also to process this only first step and left optional the second filter by flagging the option `--compareSubOptimal`. I also designed the program with the flexibility to process both single end or paired end aligned reads, and with custom experimental design such that each laboratory may choose to sequence the LTR to R1 or R2 or both pairs. Once enabled the *suboptimal* filter, the program processes the alignments by reading the AS and XS flags. In case of using a different aligner than BWA, users may change the name of the flags with the proper option: `--ASlikeTag` and `--XSlikeTag`. For each alignment read, I compute the formula $delta = (1 - XS/AS) * 100$ and compare this value with the input parameter `--suboptimalThreshold` such that if delta is higher than the threshold (that is the best hit score is higher enough than the secondary alignment), then the read is kept, otherwise removed. By default, the threshold is set up to 40.

As an example: given $AS = 100$, $XS = 80$, $delta = (1 - XS/AS) * 100 = 20$; then $delta \geq suboptimalThreshold$? In this case no, thus remove this read.

Filter by Alignment Quality

After MATE and CIGAR filtering the last step is to filter out reads with low mapping quality. This filter must be put after the two custom filters, otherwise the reads in pair should be removed before. The following properties must be satisfied for the output reads (Appendix C.1.5 for details):

- Alignments mapped.
- Alignments with mates mapped.
- Alignments sequenced as paired.
- Alignments that passed PE resolution.
- Alignments marked as primary.
- Keep reads with mapping quality ≥ 12 .

Then a BED file with only R2 reads is created (for shear site quantification is fundamental to retrieve the length of the fragment, Chapter A, [89]), thus this output file will be used to extract the product end (that will be the R2 start). For the filtering on R2 the requirements are (Appendix C.1.5 for details):

- Alignments marked as second mate.
- Not alignments marked as first mate.
- Alignments mapped.
- Alignments with mates mapped.

- Alignments sequenced as paired.
- Alignments that passed PE resolution.
- Alignments marked as primary.

Then the BAM file is converted in BED file with `bedtools bamtobed`.

3.1.8 IS Import in MySQL Database and Stats Summary

After filtering and final BAM/BED generation, all reads containing ISs are imported into MySQL databases, in two different tables (Appendixes B.4.1, B.4.2). Using MySQL tables permits an easy link to IS data for other custom analyses and softwares, compared to the use of simple plain text files only.

Stats Summary

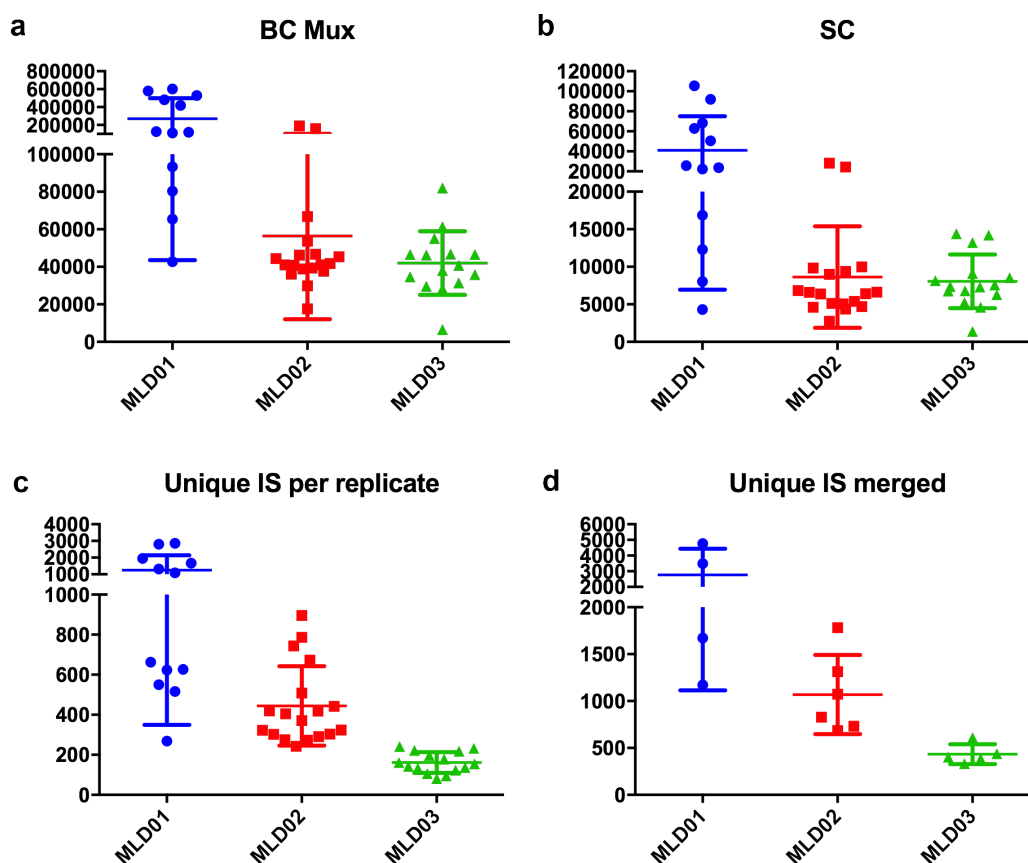


Figure 3.8: MLD Statistics: each dot is a sample; (a) barcode demultiplexing; (b) sequence count; (c) unique ISs per replicate (3 replicates for each sample are not merged); (d) unique ISs merged in replicates.

All statistics of the sequencing run are in `stats_summary_vispa2` table (Appendix B.4.3). These statistics are useful for sequencing run diagnostics, like sequencing depth quantification and counts like sequence count per sample and the number of unique ISs. For example, Figure 3.8, shows balanced sequencing depth for MLD02 and MLD03 samples, while MLD01 received an higher sequencing depth (more demultiplexed reads, SC and unique ISs). Moreover, statistics are useful to highlight some bad samples or outliers.

3.1.9 IS Merging and Collisions

State-of-the-art techniques for integration sites retrieval, both wet and bioinformatics, may introduce some artifacts that bias actual methods for integration sites identification in terms of precision in genomic location and of potential overestimation of integration sites. For this reason, I applied a static tolerance window, [90, 91], on the genomic position of the IS (that is, the starting point of the alignment): all reads that are in the same window are merged into a single locus, represented by the mode in the window. We developed a tool (Appendix C.1.6) in Python downloadable here: https://bitbucket.org/tigetbioinformatics/integration_analysis

The tool also has a more accurate algorithm to merge ISs, that is called *density*. The density-based approach we designed starts acquiring all mapped reads, accounting starting bases with their reads pileup count, then splits targeted regions into sub-regions (ensembles) of neighboring reads, defining for such ensembles an histogram of covered bases with heights corresponding to the count of piled-up reads. Once all ensembles have been detected, our procedure identifies ISs using a 3 steps model upon each one: (1) Exploration, detecting all peaks of the ensemble, (2) Evaluation, scoring all bases surrounding each peak, and (3) Decision, identifying ISs among local peaks and their surrounding bases.

1. The *Exploration* step incrementally detects local peaks and, for each one, it considers the first n nearest-neighbors bases as related to the same biological process underlying the peak.
2. The *Evaluation* process then assigns a score to such bases with respect of the peak; n parameter is given as input, typically is 8 (up to 4 base far from each side of the peak as noted in literature [90, 91]). The score is derived from a comparison between a theoretical statistical curve (e.g. Gaussian process) and the curve derived from the histogram of the peak and its surroundings, thus

A	B	C	D	E	F	G	H	I	J	K	L	M	AN	AO	AP	AQ	AR	AS	
1	chr	integration_locus	strand	DMSO_T64KDMSO_B2L1_090	DMSO_T64KDMSO_HPAC_075	DMSO_T64KDMSO_S3L1_100	DMSO_T64KDMSO_B2L1_000	DMSO_T64KDMSO_B2L1_005	DMSO_T64KDMSO_B2L1_090	DMSO_T64KDMSO_HPAC_075	DMSO_T64KDMSO_S3L1_100	DMSO_T64KDMSO_B2L1_005	all	Collision matrix_midwas_mid_was_allpatients @ T18170028	Collision sequence_diam.com_reference 9				
2	1	26451026 -	+										876	80					
3	1	26461858 +	+										62	1					
4	1	26476232 +	+										1						
5	1	26566312 -	+										58						
6	1	26571445 +	+										68						
7	1	26804423 -	+									9							
8	1	26815604 +	+										546						
9	1	26879474 -	-										8						
10	1	26974344 +	+										1						
11	1	27002717 -	-										9						
12	1	27013146 +	+										1						
13	1	27014531 -	-								1								
14	1	27057283 -	-										129						
15	1	27099177 +	+										49						
16	1	27128279 -	-										269						
17	1	27173372 -	-										2						
18	1	27215541 -	-										1343						
19	1	27309522 +	+										69						
20	1	27458906 -	-										9						
21	1	27687505 +	+										374						
22	1	27737818 +	+										73						
23	1	27746096 +	+										360						
24	1	27772795 +	+										7						
25	1	27786107 +	+										109						
26	1	27786421 -	-										148						
27	1	27792832 +	+										386						
28	1	27813038 -	-										1						
29	1	28089315 +	+										31						
30	1	28131908 -	-										3						
31	1	28132863 +	+										92						

Figure 3.9: Example matrix file of breast cancer Project, Chapter 6.

generating during the incremental process consecutive empirical distributions with potential multiple overlapping (“conflict area” whose covered bases are ranked with respect to different peaks).

3. The *Decision* step will determine the most likely attribution, assigning each covered bases of the ensemble univocally to a specific peak, using score comparisons; this procedure starts from the lowest peak till the highest: this way, if a lower peak was ranked as belonging to an higher one, the algorithm is able to assign the former to the latter, along with the bases just assigned to it; at the end, all the final groups of covered bases will collapse into unique ISs, located according to peak placement and characterized by an overall read count.

Whichever method is used the program also can annotate the contamination between independent projects (always IS based). The final IS matrix will contain, indeed, in addition to the number of ISs per sample, at each genomic location, the number of ISs in common with other datasets. This step is very useful in post-analysis phase for contamination removal. An example of a matrix file is reported in Figure 3.9.

3.1.10 IS Annotation

The final step is the annotation of ISs (the matrix), where each site is associated to nearby genomic features such as genes, miRNAs, and so on. I developed an annotation tool, `annotate_matrix`, Appendix C.1.7. This software, developed in Bash, takes in input only two files: the IS matrix and the GTF file containing all annotation data (also chrM coordinates and chrR for repeat annotation).



For each IS, the program finds the closest feature(s) among those listed in the annotation file and, for each feature, outputs the following information: the (chromosome, position) tuple that identifies the IS; the name and strand of the feature as they appear in the IS matrix file; the feature’s starting and ending position; the distance of the IS from the feature’s transcription start site (TSS); the relative position of the IS with respect to the feature (upstream, downstream or in-gene); the integration percentage for in-gene integrations (from 0% when the IS coincides with the TSS to 100% when the IS lies at the opposite end of the feature). At the end a new tab-separated file equal to the IS matrix plus the columns of the ‘GeneName’ and ‘GeneStrand’ is written.

3.2 Web Interface

Source: <http://155.253.6.236/vispa2.0P>

In collaboration with the University of Milano-Bicocca and the CNR-ITB, we developed a web interface to make easier the usage of the VISPA2 pipeline.

3. VISPA2: Faster and Extended Version of Vector Integration Site Parallel Analysis Tool



Vispa2.0 Pipeline - Paired-Ends

Input R1 Sequence File - FASTQ (GZIP)

```
@STVWD15035877:1101:1214:2000
GCAAAGGTGGGGACAGCAGAACCCCTTTAGTCAGTGTGGAGATCGGAAG...
+
DDDDDDGHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH...
@STVWD15035877:1101:1278:1998
TAGCTAGCTTTGCTCGGTAACCCCTTTAGTCAGTGTGGAAAATCTCTA...
+
DDDDDDGHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH...
...
```

AssayValidation_R1.fastq.gz (Example) Remove Browse ...

Input R2 Sequence File - FASTQ (GZIP)

```
@STVWD15035877:1101:1214:2000
GCAAAGGTGGGGACAGCAGAACCCCTTTAGTCAGTGTGGAGATCGGAAG...
+
DDDDDDGHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH...
@STVWD15035877:1101:1278:1998
TAGCTAGCTTTGCTCGGTAACCCCTTTAGTCAGTGTGGAAAATCTCTA...
+
DDDDDDGHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH...
...
```

AssayValidation_R2.fastq.gz (Example) Remove Browse ...

Reference Genome hg19 - GRCh37

Input Association File

```
LTR51.LC26 LTR51.LC26 LMv2-II Block L LMv2-II_Block_L_1 ...
LTR69.LC66 LTR69.LC66 LMv2-II Block L LMv2-II_Block_L_11 ...
LTR89.LC10 LTR89.LC10 LMv2-II Block L LMv2-II_Block_L_21 ...
LTR43.LC26 LTR43.LC26 LMv2-II Block L LMv2-II_Block_L_33 ...
LTR37.LC66 LTR37.LC66 LMv2-II Block L LMv2-II_Block_L_43 ...
LTR57.LC10 LTR57.LC10 LMv2-II Block L LMv2-II_Block_L_53 ...
LTR35.LC26 LTR35.LC26 LMv2-II Block L LMv2-II_Block_L_65 ...
```

AssayValidation.tsv (Example) Remove Browse ...

Input Disease Name PureCEM6

Input Patient Name byLAM

Input Barcode LTR File

```
>LTR
ACCCTTTTAGTCAGTGTGGAAAATCTCTAGCA
```

LTR.32bp.fa (Example) Remove Browse ...

Input Barcode LC File

```
>LC_rc
CCTAACTGCTGTGCCACT
```

LC.rc.fa (Example) Remove Browse ...

Input Suboptimal Threshold 40

Input Vector Name Iv

Submit






Figure 3.10: VISPA2 GUI: main page.

This web interface is open to all users and there is no login requirement. It has been developed using Java and Javascript technologies. In the first page an “how to” is present and the user can upload the FASTQ sequences (gzip or not) and an association file, beside an Input Barcode LTR File and a Input Barcode LC File, Figure 3.10. Other options are available in order to customize the computation. A full working example is uploaded by default. Then, after a page in which the user can confirm all the parameters the computation starts. Since the computation can take several minutes, a result page is displayed, showing the progresses of the pipeline, until results are available. The link can be bookmarked, to access it in a second time. The results page is composed of several *panels*, according to the datasets described in the association file, plus a *page* representing the IS matrix. This matrix has a column for each dataset and a row for each insertion. Each cell is filled according to the abundance of that IS in the specific dataset (a cell with zero means that the IS is not present in that dataset).

Result Table [Circos](#) [Gene Cloud](#) [GO Enrichment Analysis](#) [Samstats Tool](#)

Search:

Showing page 1 of 94

Chr	Locus	Gene	ReadCount	Strand
1	163673332	LOC100422212	1	-
1	150078756	VPS45	1	+
1	174394337	GPR52	7	+
1	3126410	PRDM16	1	+
1	1258585	CPSF3L	5	-
1	46058527	NASP	2	+
1	237049631	MTR	18	+
1	1817665	GNB1	2	-
1	191043160	LOC440704	1	+
1	33014444	ZBTB8A	4	+

Display records per page

Previous **1** 2 3 4 5 ... 94 Next

Figure 3.11: VISPA2 GUI: results, IS matrix, from breast cancer post-treatment, Chapter 6.

Concerning the *pages* presenting the results for each dataset, many different statistics are reported. In the upper part of the *page*, an histogram of the IS distribution in the genome is reported. In the bottom part of the *page* some tabs are present, showing different graphs and representation which describe the IS distribution in a more detailed way. The first tab represents a table showing the specific chromosome locus and strand of each ISs and reporting the nearest gene, Figure 3.11. The second tab shows a circos plot of the IS density in the genome, Figure 3.12. The third tab shows a gene cloud of the gene more targeted by insertions, Figure 3.13. The fourth tab shows the Gene Ontology (GO) enrichment of the target genes, considering the three branches of GO (Molecular Function, Biological Process and Cellular Components). Beside the p-values achieved in the enrichment analysis, a diagram is reported of the most representative GO, bi-clustered according to their semantic similarity. The last tab present the statistics concerning the dataset, as computed by *samstats* [92].

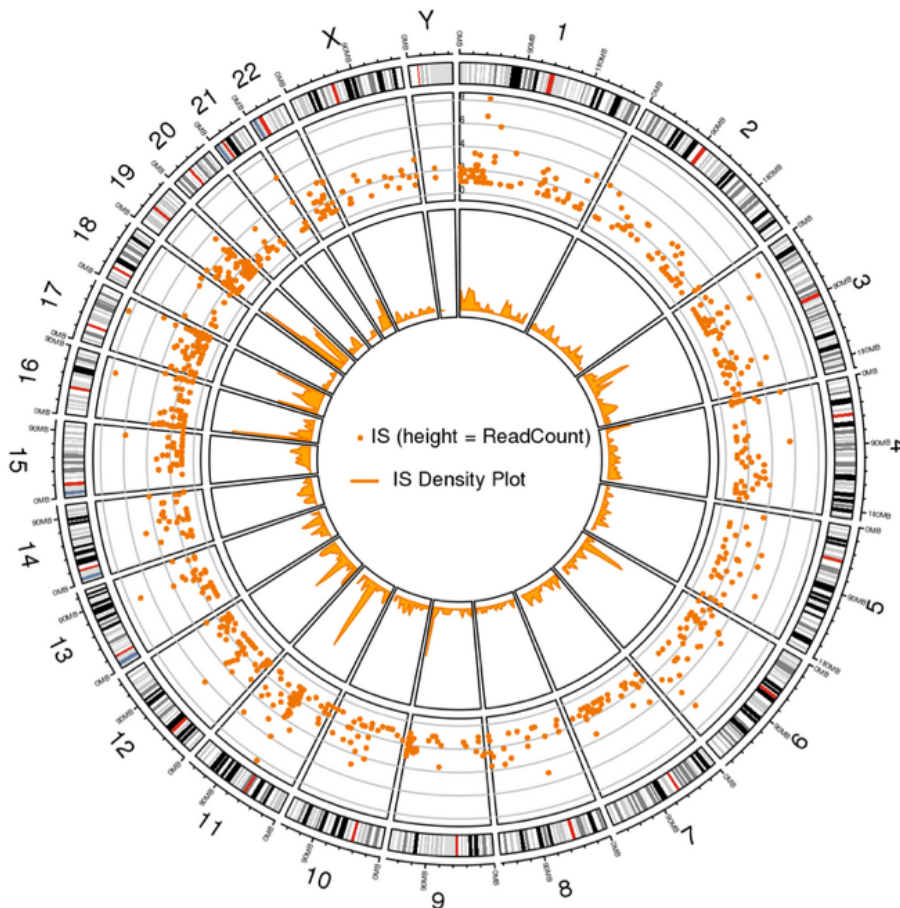


Figure 3.12: VISPA2 GUI: results, circos plot. The inner circle represents the IS density and the outer the IS read count, from breast cancer post-treatment, Chapter 6.

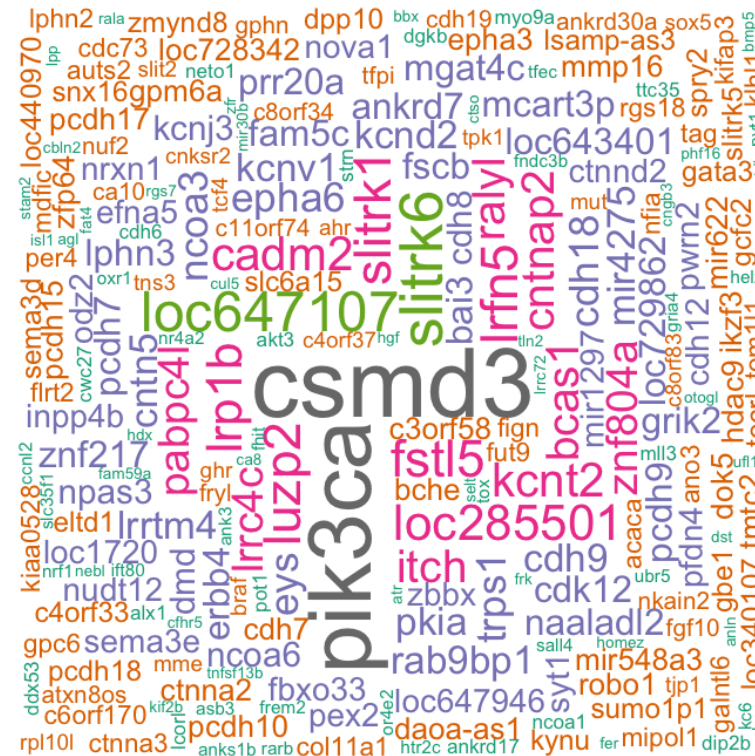


Figure 3.13: VISPA2 GUI: results, word cloud. The most targeted genes (not CISs necessarily), from breast cancer post-treatment, Chapter 6.

3.3 Performances

3.3.1 Precision and Recall

Precision and recall³ assessments are proposed here in accordance with the information retrieval context. Given that the items are the simulated ISs (the same dataset tested for VISPA and the other tools in Chapter 2.1), I accounted as relevant items all the ones that the approach under evaluation was able to collapse in ISs showing a perfect match with simulation input, in terms of location and total sequence count.

Thus I defined precision as the ratio between the number of ISs in perfect match with input and the total number of ISs detected, measure that provides scores as closer to 1 as the outcomes become more precise; please note that by definition is precision $\epsilon[0, 1]$. Conversely, I defined recall as the ratio between the same numerator of the precision and the total number of generated ISs: here to note that recall $\epsilon[0, 1]$ and a score closer to 1 highlights a higher retrieval power (number of ISs retrieved).

Precision and recall assessments in single IS simulations (Figure 3.15) show a neat

³https://en.wikipedia.org/wiki/Precision_and_recall

superiority of VISPA2 compared to VISPA, MAVRIC and SEQMAP.

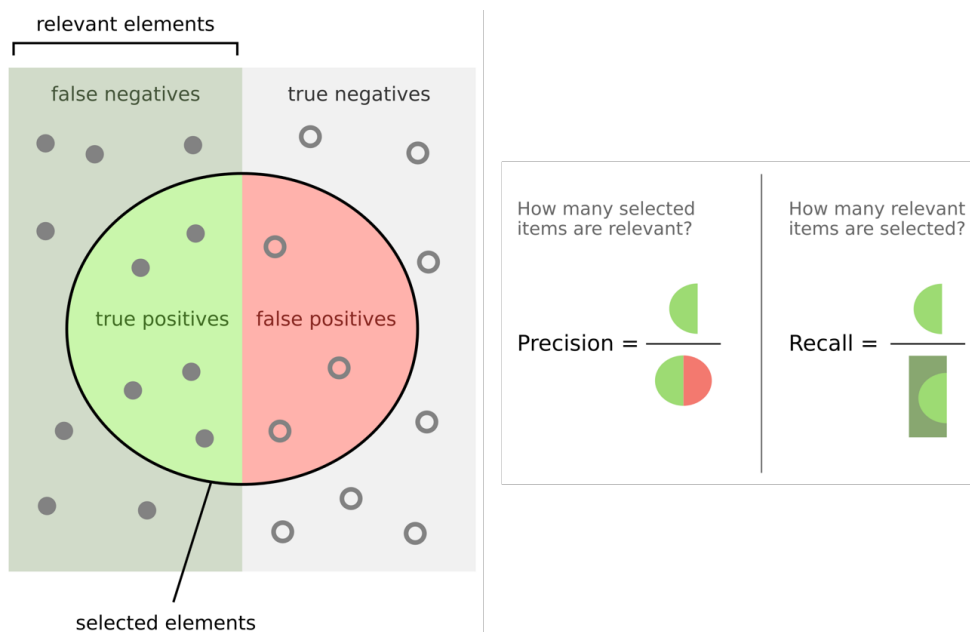


Figure 3.14: Precision and Recall definition, *Wikipedia*.

Only QuickMap is comparable, indeed it is the only tool that supports a simple repeats analysis, based on statistical analysis and not with RepeatMasker, that is more robust and reliable. The problem of QuickMap, as I said before, is that only support FASTA, single pair reads and uses BLAST (that is so slow) to map the reads. In conclusion the repetitive element support is the key feature to reach optimal results.

If I zoom the section of plot regarding VISPA2 in Figure 3.16 it is possible to see in detail the difference between the two best tools. VISPA2 has 1.0/0.97 and QuickMap 0.97/0.98 for precision/recall, at the end VISPA2 can be considered as best bioinformatics pipeline for IS detection.

3.3.2 Space Required and Time Consuming

Taking in consideration the following setup on Gemini workstation (Appendix B.1), I compared VISPA (comparable to the other tools) versus VISPA2 in Table 3.3.

Considering the space and the time required for the two type of runs, VISPA2 obtains a 6X and 7X improvements respectively, as in Figure 3.17.

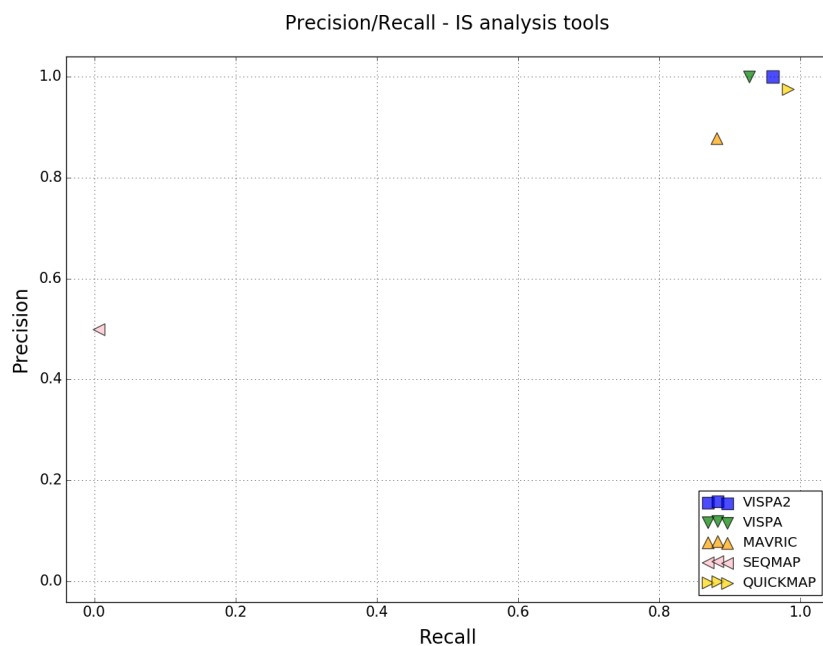


Figure 3.15: Precision and Recall of VISPA2 versus all other pipelines.

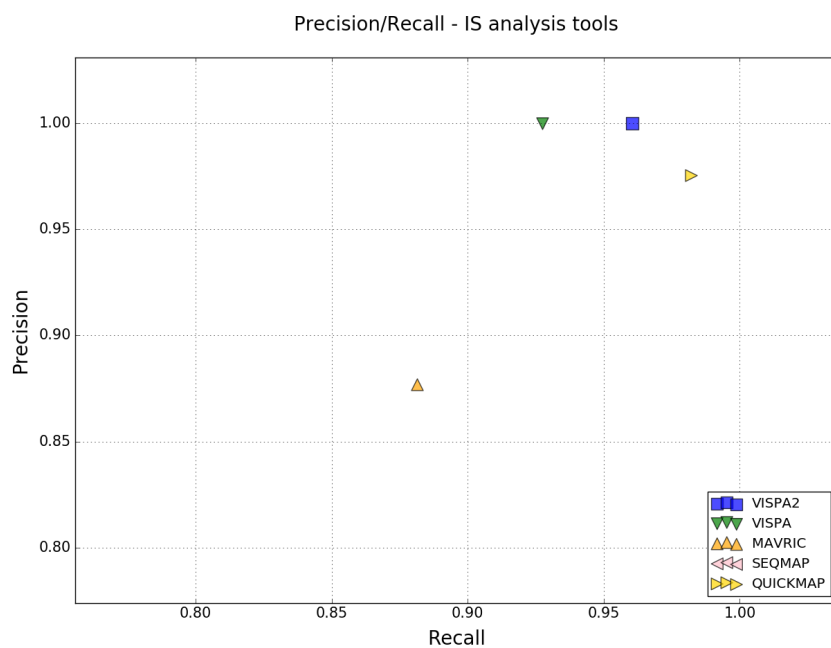


Figure 3.16: Precision and Recall of VISPA2 versus all other pipelines, a zoom.

VISPA2 Performances - Setup		
Pipeline	NGS Technology	Total Number of Reads
VISPA	MiSeq	14,583,450**
VISPA	HiSeq	186,300,301*
VISPA2	MiSeq	14,583,450**
VISPA2	HiSeq	186,300,301*

Table 3.3: VISPA2 Performances compared to VISPA. I considered two types of Illumina sequencing: **MiSeq and *HiSeq.

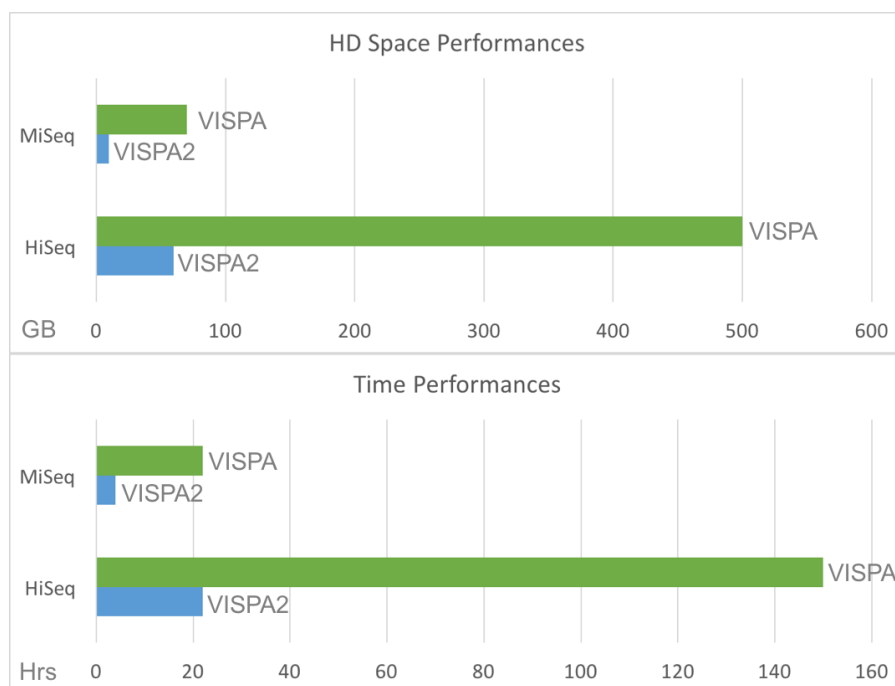


Figure 3.17: Performances compared to VISPA (green) and VISPA2 (blue), time and space. Shorter bar is better.

VISPA2 takes less than one day to process an Illumina HiSeq and only 3 hours for a Illumina MiSeq run. It has also great results in terms of number of IS retrieved and precision of mapping also in repetitive elements. All these achievements bring VISPA2 as top IS retrieval pipeline for gene therapy studies.

3.4 SR-Tiget IS Pipelines, Final Overview

As it is possible to see in Figure 3.18, VISPA [7] is now obsolete, because the lack of paired-end and repeats support. The standard now is becoming VISPA2 and γ -TRIS [93]. VISPA2 is installed in all our servers and is freely available also for non-bioinformatician people because is very fast and simple. γ -TRIS, with its genome-free

and graph-based approach, is more robust but the slowness (because of the clustering/graph algorithms) and difficulty to use (no GUI) are crucial for its future use.

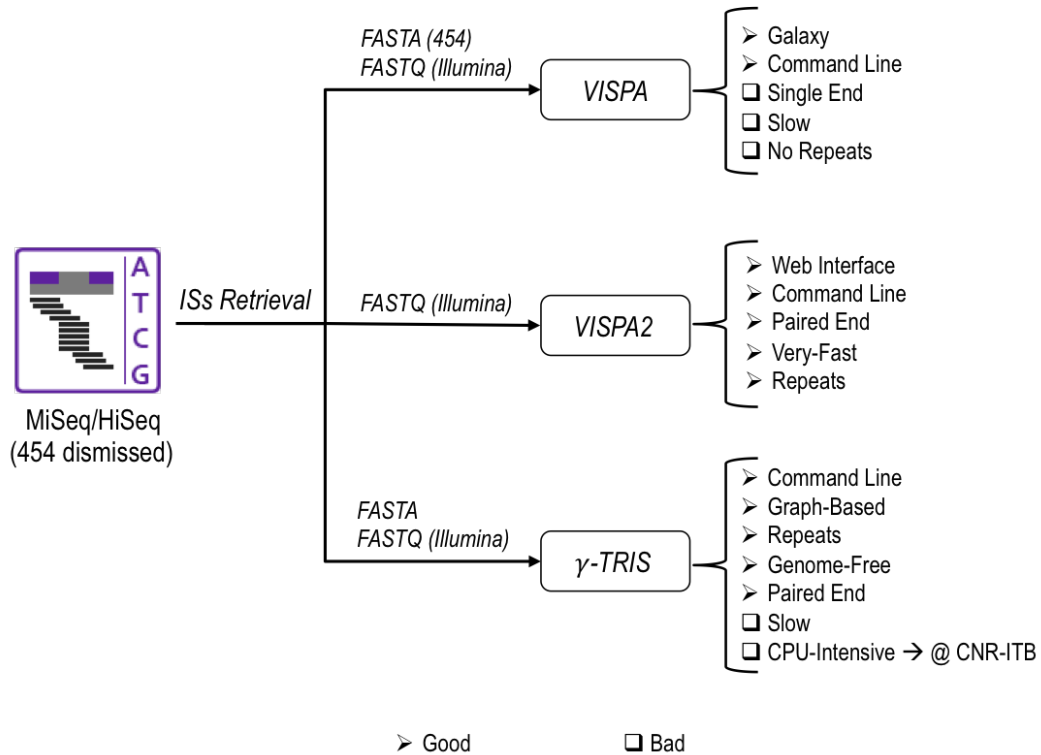


Figure 3.18: History of SR-Tiget IS Pipelines. The old VISPA [7] supports 454 and Illumina single-end reads, has the Galaxy/Bash interface but lacks of paired-end support, is very slow because BLAST [23] and does not support repetitive elements. γ -TRIS has only the Bash version (developed in C++), is graph-based, supports repetitive elements, is genome free (because of the clustering algorithms in consensus sequences), supports the Illumina paired-end technology, but is terribly slow and CPU-intensive. The new VISPA2 combine the easiness of the old VISPA and the power of γ -TRIS. It has the Bash/GUI interface, supports the paired-end sequencing, is very fast and supports repetitive elements.

Chapter 4

VISPA2 Post Analysis: ISAnalytics and Common Insertion Site Identification

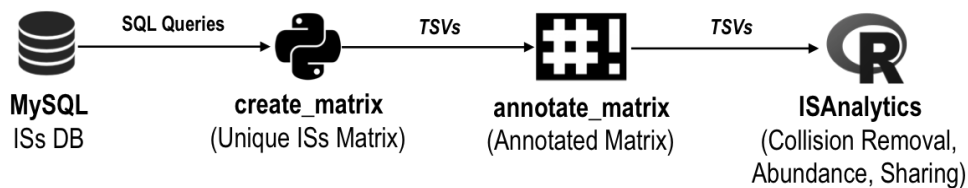


Figure 4.1: Steps after VISPA2.

After the genomic annotation of the matrix file, we usually process matrixes to target and answer specific biological questions, from vector toxicity to clonal dynamics over time. Given the broader scenario of different analyses that can be done and implemented or extended, the software requirements need a tool with high flexibility for data manipulation and processing, and with flexible data structures to easily add and extend analyses. This software needs to take in input the data matrix of ISs and identifies from the column labels the semantic content of the columns. Then, it requires to have implemented a set of independent analyses (as activities and data manipulation functions) that can be composed as a workflow and finally to generate the resulting output (plots or data). No available tools satisfy these requirements and thus we need to design and develop a new analytical framework. I designed a new tool for data exploration, mining and analysis, enabling not only the analyses of IS datasets but also generalized to other potential applications. My approach is realized by importing the input dataset as matrix/object and designing all the different analyses as independent functions that acquire the same input and produce the output with the same

input data structure so that a workflow can be easily customized as set of consecutive operations.

Once generate the matrix file, the first main steps are summarized in Figure 4.1.

As described before, the `create_matrix` and `annotate_matrix` programs create the final IS matrix (tab-separated matrix with ISs with their sequence count between samples) with annotation informations (nearest gene and gene strand for each IS) ready to be processed for molecular analysis. For this last step I decided to create an R [94] package, compliant with Bioconductor 3.1 [95], that can be the starting point for all common analysis in gene therapy studies.

4.1 Integration Site Analytics (ISAnalytics)

ISAnalytics can do the following:

- Collision detection (versus other gene therapy studies collected before)
- Collision removal and assignment (if it is possible)
- Clonal abundance by sample or by group of samples
- IS sharing between different samples
- IS binning
- Output generation for detection of Common Insertion Site

4.1.1 Motivation

Although I have always adopted the direct programming and often the choice of Python and Bash for data generation, I have decided (with Davide Rambaldi, the co-author of this side project), at this stage of post-analysis, to use R to analyze and, in particular Bioconductor (allowing publication of the package as a methodology paper), to ensure that other groups were using this package and can combine this step with the final one that I will describe in the following chapters. In addition will be very easy and useful to use two R suitable structure to handle genomic coordinates and samples/projects, which are *GenomicRanges* (GRanges) [96] and *SummarizedExperiment* [97].

4.1.2 Data Loader

To correctly install and import ISAnalytics package some other packages are needed: *S4Vectors*, *stats4*, *BiocGenerics*, *parallel*, *BiocGenerics*, *IRanges*, *GenomicRanges*, *GenomeInfoDb*, *reshape2*, *ggplot2*, *plyr*, *scales*, *gridExtra*. As Reported in Figure

4.2 an input matrix (IS matrix, annotated with or without collision columns, as in Figure 3.9) needs to be converted in a dataframe before to use it. For the conversion step, for simplicity, each empty entry is converted in '*', genomic coordinates (chr, integration locus, strand, gene name and gene strand) are also merged creating a label like GENE_NAME(STRAND)_CHR:INTEGRATION_LOCUS(STRAND). Sequence counts 0 are also converted to 'NA' and the sample IDs are split in three parts (if the sample ID is BM_CD34_01, sample is BM, condition is CD34 and time is 01). If there are some summary columns they will be treated apart.

Once the IS matrix is produced by VISPA2 it can be imported in R and therefore in ISAnalytics in Excel format using *ISDataSetFromXlxs* (but JAVA libraries for this function are relatively slow) or with *ISDataSetFromISA* (that imports the classic tab-separated IS matrix file, as described until now). In every case the matrix file is converted in a R data-frame and then to an ISDataSet, Figure 4.3.

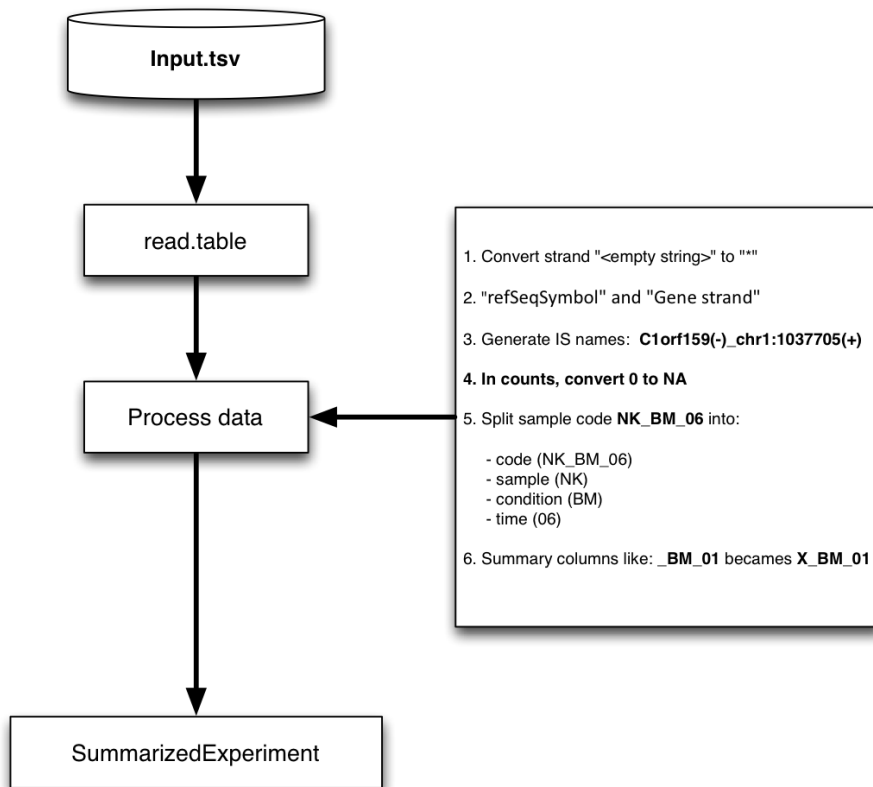


Figure 4.2: ISAnalytics Data Loader: SummarizedExperiment integration.

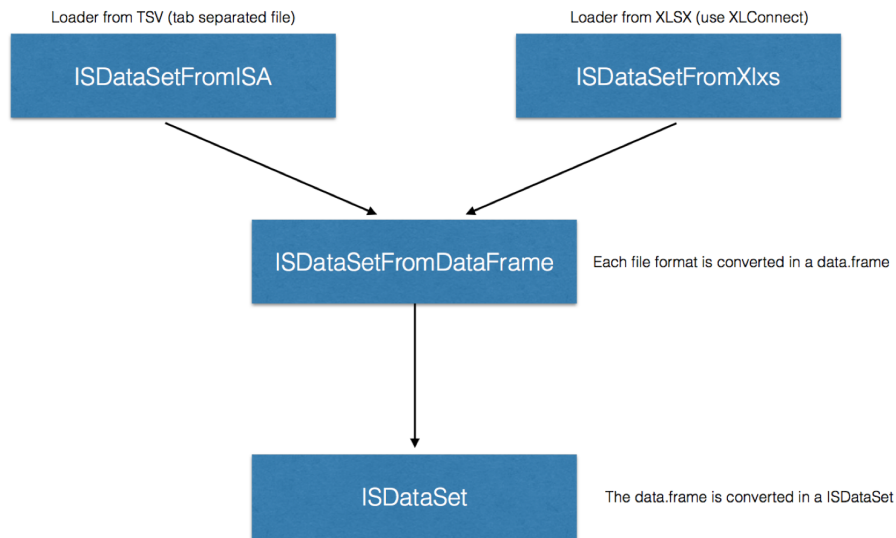


Figure 4.3: ISAnalytics Data Loader: tab-separated and excel import into dataframes.

```

granges (isset)
GRanges object with 19295 ranges and 2 metadata columns:
      seqnames      ranges strand | closest_gene strand_gene
      <Rle>         <IRanges> <Rle> | <character> <character>
  LOC729737 (-)_chr1:121249 (+) chr1 [121249, 121249] + | LOC729737 -
  LOC729737 (-)_chr1:133364 (-) chr1 [133364, 133364] - | LOC729737 -
  LOC100288069 (-)_chr1:710530 (+) chr1 [710530, 710530] + | LOC100288069 -
  LOC643837 (+)_chr1:775536 (+) chr1 [775536, 775536] + | LOC643837 +
  FAM41C (-)_chr1:804555 (-) chr1 [804555, 804555] - | FAM41C -
  ...
  RBMY2EP (-)_chrY:23301216 (+) chrY [23301216, 23301216] + | RBMY2EP -
  RBMY2EP (-)_chrY:23452748 (+) chrY [23452748, 23452748] + | RBMY2EP -
  RBMY2EP (-)_chrY:23533941 (-) chrY [23533941, 23533941] - | RBMY2EP -
  TTTY13 (-)_chrY:23839821 (+) chrY [23839821, 23839821] + | TTTY13 -
  TTTY3B (+)_chrY:28493922 (+) chrY [28493922, 28493922] + | TTTY3B +
  -----
  seqinfo: 24 sequences from an unspecified genome; no seqlengths
  
```

Figure 4.4: GRanges of an isset.

GRanges objects are very useful, indeed they permit nice operations between genomic ranges:

- Retrieve all ISs on chromosome one:

```
isset <- ISDataSetFromISA(file);
region <- GRanges(
  seqnames = Rle("chr1"),
  ranges = IRanges(start=1, end=15784278),
  strandRle = ("+")
)
p = subsetByOverlaps(isset,region)
```

- Retrieve all ISs around one gene:

```
y <- subset(isset, mcols(isset)$closest\_gene == "OR11L1")
```

In addition metadata can be used to retrieve all samples in a particular condition, for example all samples at condition “BM”, output in Figure 4.5:

```
isset[, colData(isset)$condition == "BM"]
```

	code	sample	condition	time
	<character>	<character>	<factor>	<factor>
BM_BM_01	BM_BM_01	BM	BM	01
BM_BM_03	BM_BM_03	BM	BM	03
BM_BM_12	BM_BM_12	BM	BM	12
BM_BM_18	BM_BM_18	BM	BM	18
BM_BM_24	BM_BM_24	BM	BM	24
BM_BM_30	BM_BM_30	BM	BM	30
BM_BM_36	BM_BM_36	BM	BM	36

Figure 4.5: Conditions between samples.

Bioconductor defines the *SummarizedExperiment* class. The computed summaries for the ranges are compiled into a rectangular array whose rows correspond to the ranges and whose columns correspond to the different samples (Figure 4.6). For a typical experiment (or project), there can be millions of ranges and from a handful to hundreds of samples [98]. Like, for example, Excel, *SummarizedExperiment* (`se`) offers the possible switch between different experiments with `exptData(se)` and, very interesting, a switch between sheets (assays, for instance different type of analysis in the same experiment) with `assays(se)`. With `colData(se)` I can access the samples

and with `rowData(se)` access to the `GRanges` (because the use of them instead `rowID`). At the end the combination of `GRanges` and `SummarizedExperiment` is very powerful and well-fitting for our studies.

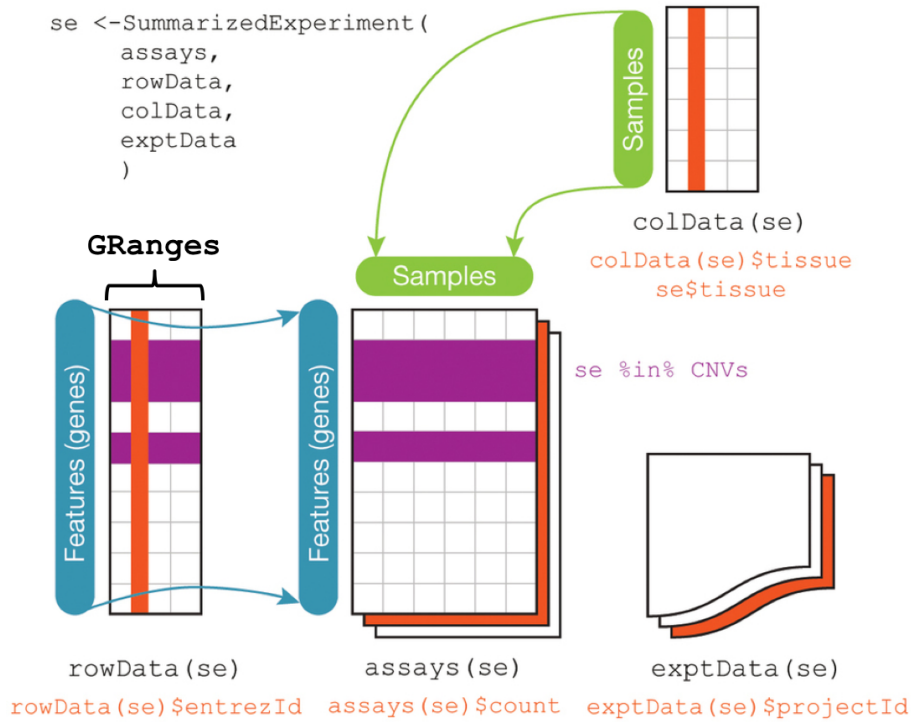


Figure 4.6: ISAnalytics Structure: SummarizedExperiment and GRanges.

4.1.3 Collision Detection and Removal

The term 'collision' is used to identify the presence of identical ISs in independent samples. In our experimental settings, the integration of vector in the very same genomic position in different cells is a very low probability event. Thus, the detection of identical ISs in independent samples likely derives from contamination, which may occur at different stages of wet laboratory procedures (sample purification, DNA extraction, LAM-PCRs and NGS). Although our working pipeline is designed to minimize the occurrence of inter-samples contacts, the high-throughput analysis of ISs intrinsically carries a certain degree of background contamination. Identification of the extent of contamination between samples is crucial also because the retrieval of the same IS in different samples obtained from the same patient is used in subsequent steps to make inference on biological properties of the vector-marked hematopoietic cells (i.e. multi-lineage potential and sustained clonogenic activity). Thus, we must be able to distinguish the actual occurrence of the same IS in different samples (from the same

patient) from a contamination/collision.

We created an advanced collision detection process in [19, 49] with MLD-WAS patients' samples. Given \mathbf{C} the set of collisions, each \mathbf{c} in \mathbf{C} has a sequence count \mathbf{s} . For each patient, we independently analyzed all collisions: given \mathbf{P} the set of patients, \mathbf{p} the current patient and $\mathbf{p}(\mathbf{n})$ all other patients ($\mathbf{P}-\mathbf{p}$), for each patient \mathbf{p} , for collision \mathbf{c} in $\mathbf{C}_\mathbf{p}$ we computed the ratio $s|_{cp}/s|_{cp(n)}$, called collision relative frequency (*ColRF*). We then analyzed the distribution of all *ColRF* in \mathbf{C} to look for flexes and peaks.

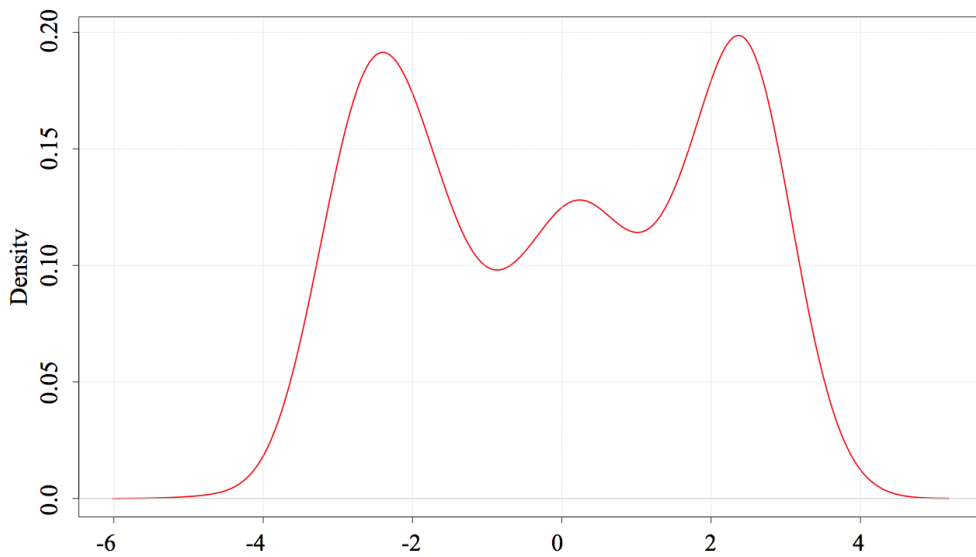


Figure 4.7: Density plot of inter-patients collisions MLD-WAS.

Collisions scenarios between 2 patients				
IS	Pt A	Pt B	A/B	Log10 A/B
1	1	100	0.01	-2.0
2	20	100	0.20	-0.7
3	100	100	1.00	0.00
4	150	100	1.50	0.18
5	250	100	2.10	0.40
6	1010	100	10.01	1.00

Table 4.1: Theoretical use case scenarios for collisions between two patients.

Positive peak at +2 means that all collisions of this area carry sequence counts 20 times higher in the analyzed patient compared to the others. On the other hand, the collisions under peak at -2 have 20 times lower sequence counts in the same patient as compared to the others. The peak at 0 indicates that all these collisions have identical

sequence counts among patients. The table below shows theoretical use case scenarios for two patients.

Applying these theoretical scenarios to our empirical ColRF curve, we interpreted data as decision plot. Thus the chosen threshold to set for contamination identification patient-based is 1, corresponding to 10 fold difference in linear scale.

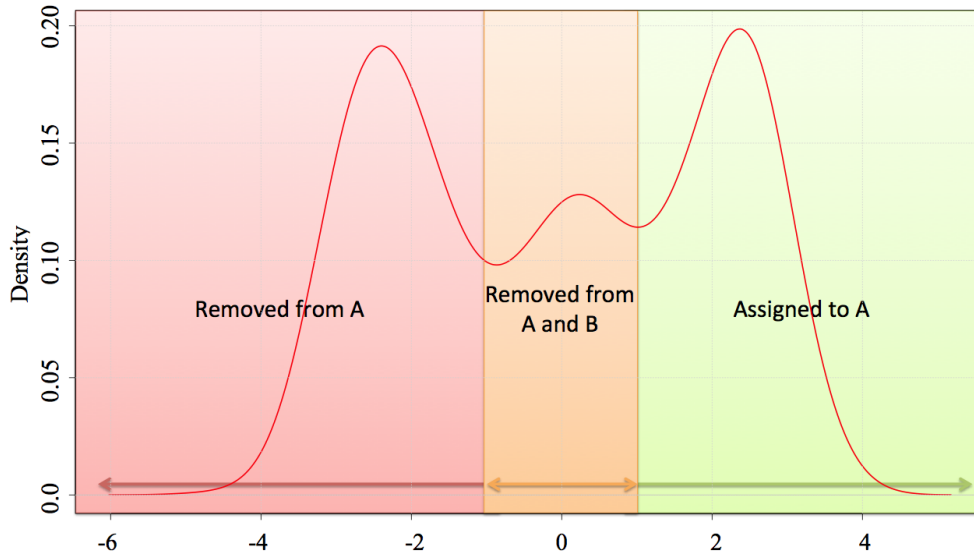


Figure 4.8: Density plot of inter-patients collisions MLD-WAS. Decision Plot.

Collision scenarios between 2 patients								
IS	Pt A	Pt B	Ratio A/B	Log10 A/B	Removed from A	Removed from B	Assigned to A	Assigned to B
1	1	100	0.01	-2.0	X			X
2	20	100	0.20	-0.7	X	X		
3	100	100	1.00	0.00	X	X		
4	150	100	1.50	0.18	X	X		
5	250	100	2.10	0.40	X	X		
6	1010	100	10.01	1.00		X	X	

Table 4.2: Theoretical use case scenarios for collisions between two patients.

We obtained the results in Table 4.2, from Figure 4.8. In ISAnalytics I implemented this strategy to remove the collisions between datasets, comparing indeed the fold increase of an IS in all conditions/cell population against the maximum frequency observed in other samples, describing the three possible outcomes previously said:

$$mFold = \frac{\sum_{i=1}^j C}{\max_{m=1}^n (C)} \quad (4.1)$$

- $mFold > 10x$: Assigned to the current patient
- $1x < mFold < 10x$: Removed from the current patient and NOT assigned
- $mFold > -10x$: Assigned to other patient

4.2 Frequency and Filtering of a IS

It is possible define the relative frequency of an IS, as follow:

$$F = \frac{c_i}{\sum_{i=1}^j c_i} \quad (4.2)$$

where c_i is the sequence count of the IS in that sample. Basically it is a normalization of the sequence count of an IS in the relative sample (dividing it by the total sequence count of the sample). In the collision column (Figure 3.9) we have the total sequence count of the colliding dataset (or patients, in purple). Given that not all ISs are represented (only intersection), the sum of the reads is taken from the column label (T):

$$F = \frac{c_i}{T} \quad (4.3)$$

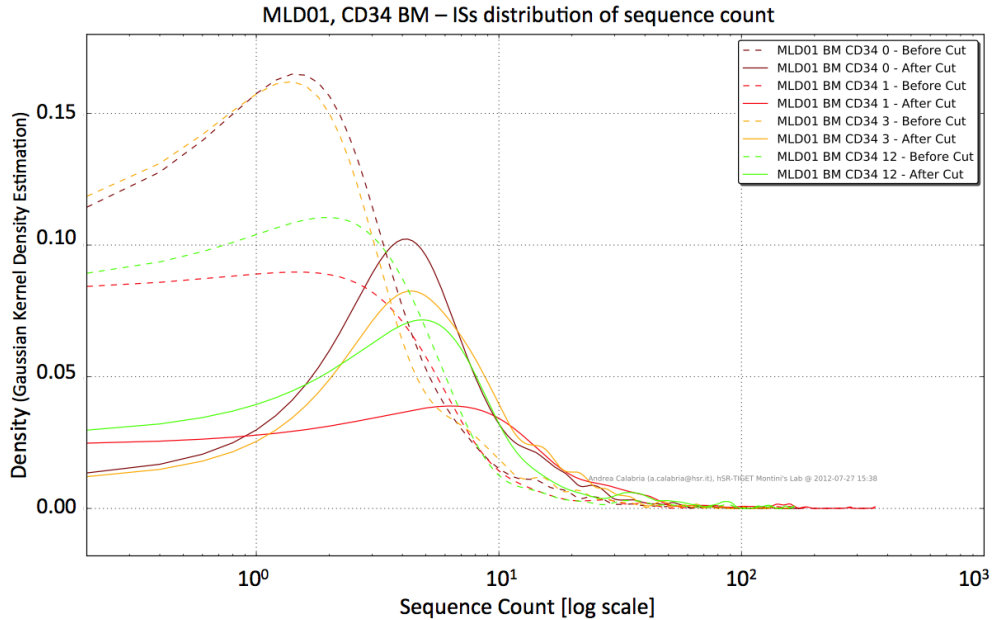


Figure 4.9: SCFilter on MLD patient 1, distribution of IS.

In the case of Figure 3.9, the two frequency are (versus MLD/WAS patients, first row):

$$F = \frac{c_i}{T} = \frac{80}{118170028} = 0.00000067699062$$

Sometime can be useful to remove all the IS with sequence count below a certain threshold, for contamination removal purposes or to remove possible false positives. In ISAnalytics the function `scFilter` on a certain `isset` does exactly this:

```
isset <- scFilter(isset, threshold=3)
assays(isset)
```

```
> List of length 2
> names(2): counts scfilter
```

producing another assay (sheet) with the IS matrix in which the sequence count 1 or 2 of the ISs is put to “NA”. In Figure 4.9 the vast majority of the ISs are with SC 1 or 2.

4.3 Clonal Abundance

The relative abundance of vector insertions in a target genome is important to understand the safety and the efficacy treatment of the gene therapy in patients. To avoid *Insertional Mutagenesis* (mutagenesis of DNA by the insertion of one or more bases) it is important to monitor the absence of clonal dominance in time, mostly in malignant genes. IS abundance could be estimated from a sample of cells if only the host genomic sites of retroviral insertions could be directly counted (Vector Copy Number, $VCN \sim 1$, number of vector copies infused in patient).

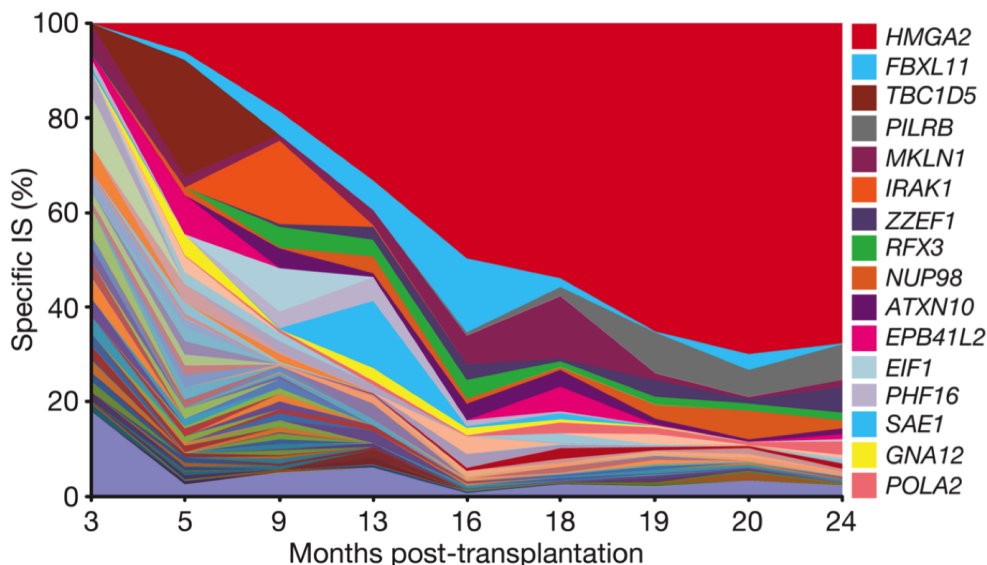


Figure 4.10: Clonal Dominance in β -Thalassaemia [24].

Now the possible questions are:

- What is the relative abundance of a given IS compared with the total number of reads observed in a set of conditions?

```
isset <- calculateColAbundance(isset, tot.in.label = FALSE,  
normalize.by.source=TRUE, output.assay = "abundance")  
writeISA(isset,"abundance.bySample.tsv",assay = "abundance")
```

- What is the relative abundance of a given IS compared with the total number of reads observed in a single sample? (IMPORTANT: to observe a clonal expansion of a *clone* (cells that harbor the same IS, it is a cell population) in time, only this calculation can be done correctly). This strategy is necessary to avoid biases due to sub-sampling issues or sequencing depth of different samples.

```
isset <- calculateColAbundance(isset, tot.in.label = FALSE,  
normalize.by.source=FALSE, output.assay = "abundance")  
writeISA(isset,"abundance.bySample.tsv",assay = "abundance")
```

The function `calculateColAbundance` performs abundance computation in two ways: answering to the first question with `normalize.by.source=TRUE`, to the second with `normalize.by.source=FALSE`. The parameter `tot.in.label = FALSE` set to calculate the total to divide each IS sequence count at each sample, instead to read it to the label. `writeISA` gives in input the experiment and the assay to process then write the output as a CSV file (tab-separated or comma-separated).

Once the abundance is calculated a way to represent it is with box-plots¹, Figure 4.11.

Here the code:

```
pdf("abundanceBySample.bplot.ISA.pdf")  
boxplotOutliers(mld01, assay="abundance", samples="^CD34")  
dev.off()
```

4.3.1 IS Sharing

Another important analysis that can be done is the IS sharing between samples. Once the abundance is calculated, it is possible to sort the IS by the most abundant and samples, in a specific order, like this:

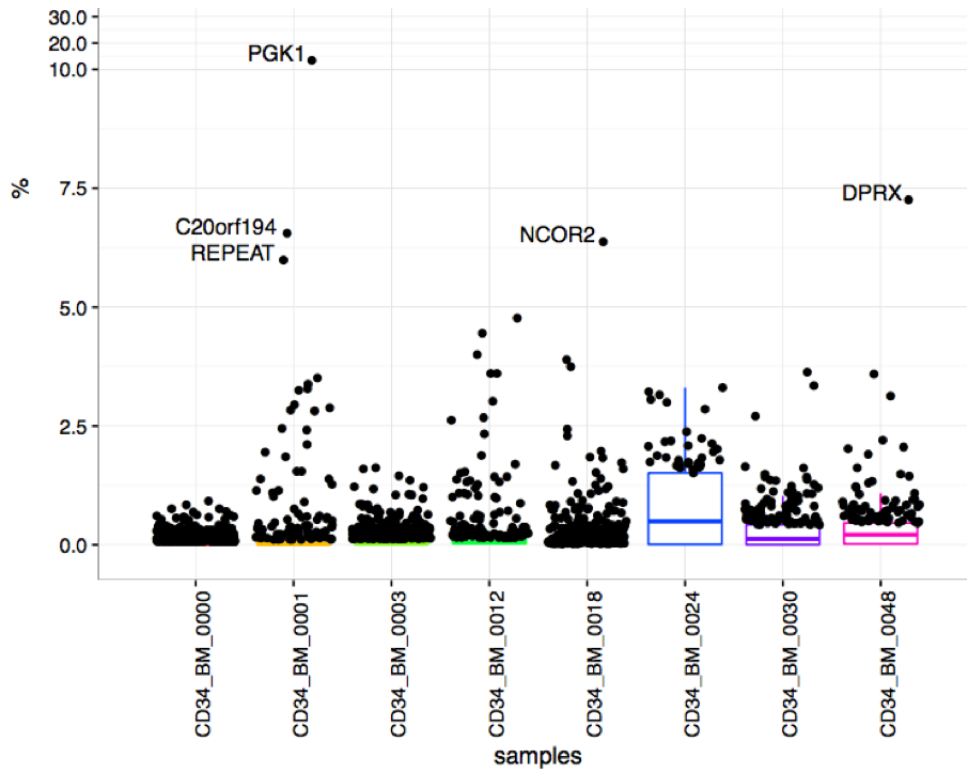


Figure 4.11: MLD, Abundance Box-Plot of CD34, BM. ISAnalytics automatically writes gene label if abundance is $> 5\%$

```
# Collect cells
cd34_bm <- findLabels(isset, "^CD34_BM")
cd14_pb <- findLabels(isset, "^CD14_PB")
cd15_pb <- findLabels(isset, "^CD15_PB")
cd19_pb <- findLabels(isset, "^CD19_PB")
cd3_pb <- findLabels(isset, "^CD3_PB")
samples <- c(cd34_bm, cd14_pb, cd15_pb, cd19_pb, cd3_pb)

# HeatMap Clonality
isset.heatmap <- mld01[,samples]
isset.heatmap <- sortByColData(isset.heatmap)
isset.heatmap <- driverSelectionFilter(misset.heatmap, assay="purity",
threshold=5)
```

¹It is in developing a new way to represent the abundance based on movie-chart plot (we call it gene-chart), created by Zach Beane, <http://xach.com/moviecharts/>

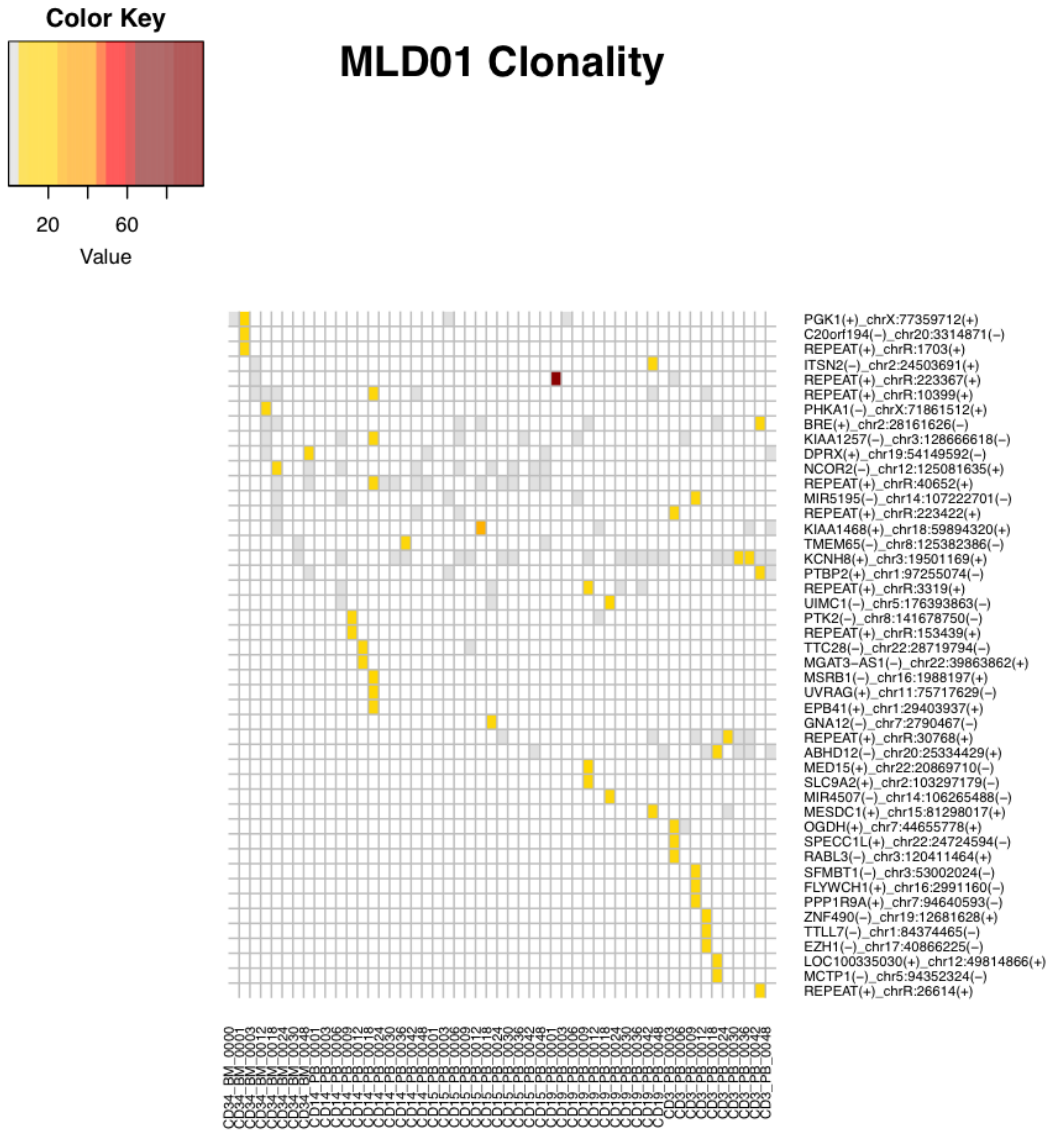


Figure 4.12: MLD, Clonality Heatmap between shared ISs. For each IS, colored cells indicate retrieval at > 5%, with higher color intensity indicating higher percentage, whereas gray cells indicate retrieval at low percentage (from 0.006% to < 5%). Lack of color indicates that the integration was not retrieved at the indicated time point and source. The targeted genes are indicated on right, samples on bottom.

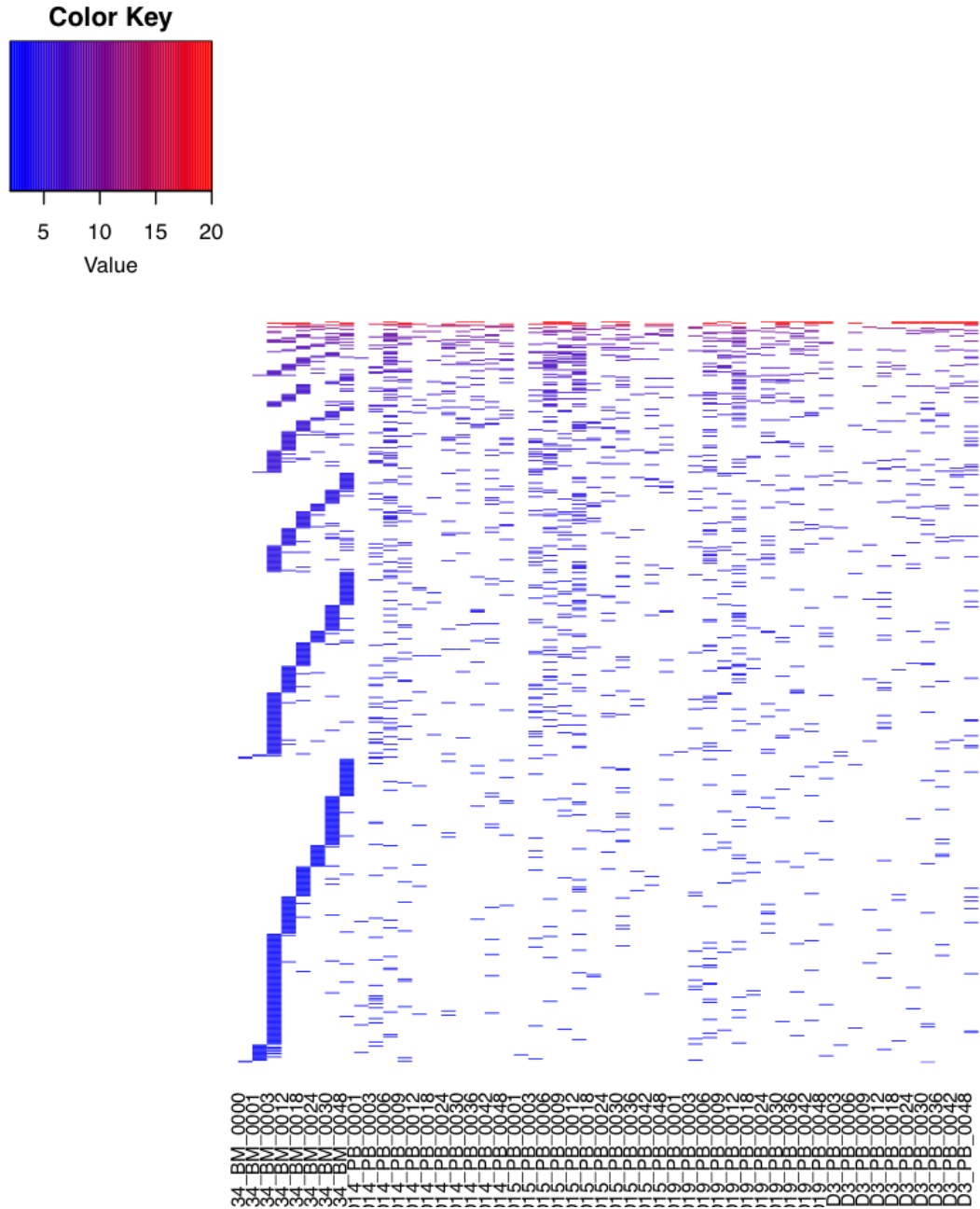


Figure 4.13: Tracking of ISs shared between multiple lineages with time in patient MLD01. Each row represents a specific IS, with colored bars indicating retrieval from the indicated cell lineage and time point after gene therapy (columns). The line color varies with the degree of sharing among lineages (red, high level of sharing; blue, low level of sharing; white, no integration retrieved). Only samples are visualized, in bottom.


```
pdf("clonality.heatmap.ISA.pdf")
heatmapDriver(isset.heatmap, assay="purity", title = "MLD01 Clonality",
force = TRUE)
dev.off()
```

In Figure 4.12 the results. We use a lentiviral vector to introduce a functional ARSA gene into HSCs ex vivo and shows that reinfusion of the engineered HSCs prevents and corrects disease manifestations (in MLD clinical trial, taken as one of the example datasets [19]). To assess HSC gene marking in vivo, it is necessary to analyze the subset of IS shared among three datasets representing progenitors and mature myeloid and mature lymphoid cells in each patient. After stringent filtering to reduce the false discovery rate due to the impurity of each cell fraction analyzed (using `scFilter`) and the occurrence of collisions during IS processing, a fraction of ISs are consistently shared among the three datasets of each patient, here the code and the heatmap, Figure 4.13.

```
sharing_progenitor <- findLabels(isset, "^CD34_BM")
sharing_against <- c(cd14_pb, cd15_pb, cd19_pb, cd3_pb)

isset <- sharing(isset, with = sharing_progenitor, against = sharing_against,
assay="purity", output.assay = "sharing")

pdf("sharing.heatmap.ISA.pdf")
heatmapSharing(isset, cell.types = c("CD34", "CD14", "CD15", "CD19", "CD3"),
title = "MLD01 Sharing", force = TRUE)
dev.off()
```

To finalize this part, an example of ISAnalytics' usage is reported in Appendix C.2.1.

4.4 Common Insertion Sites

Vector integration frequency along the genome is not homogeneous. Dense clusters of integrations contained in a relatively narrow genomic interval, known as Common Insertion Sites (CIS) have been used as an indicator of ongoing genetic selection and enrichment of cell clones harboring integrations that, by targeting specific genes, have acquired a selective advantage in vivo. In hematopoietic cells from patients from the γ -Retroviral Vector (γ -RV) based clinical trials for X-SCID, CGD and WAS, CIS were identified. Among all CIS identified, some, targeting cancer genes such as LMO2, MECOM, PRDM16, CCND2 and SETBP1, were found in

leukemic/dysplastic/dominant cell clones from patients' blood. To investigate the presence of CISs in our study we used:

1. A region-based approach based on *sliding windows* [8, 9, 10].
2. A method for CIS identification based on a new genome-wide *Grubbs test* for outliers' analysis [33].
3. A *scan statistics* approach to validate hotspot regions [99, 100].

The method 1 is implemented in R and compatible with ISAnalytics. The second method, the Grubbs test, is manually curated and it is useful to remove the false positives and outliers used in combination with the first. The last is the new one and the developing is ongoing, up to now is not usable with our tools.

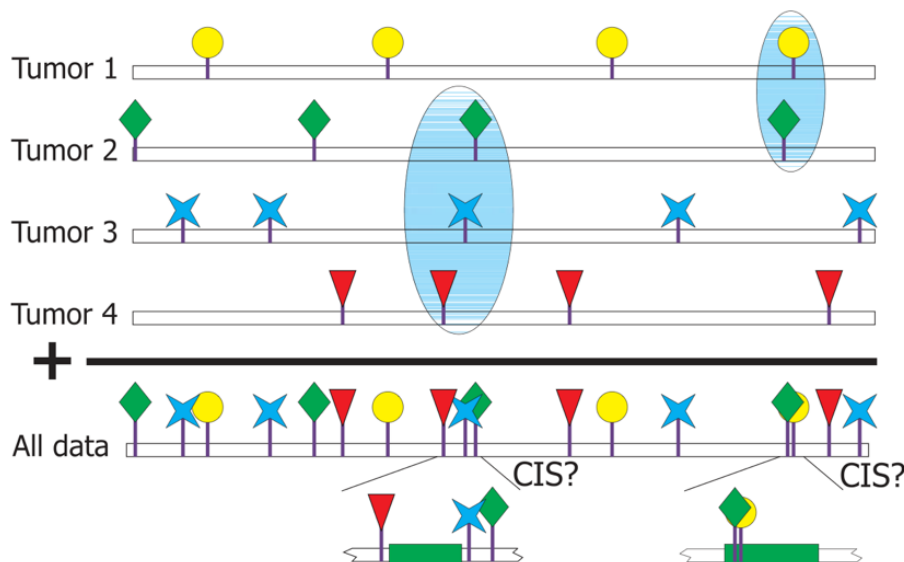


Figure 4.14: Common Insertion Sites: CISs are defined as regions of the genome that are targeted by vector integrations in independent tumors with a frequency higher than the one that is expected to occur by chance [25]. Also a new model, graph-based, has been proposed in [26].

The algorithm developed in [9] is the starting point for CIS analysis. The quantification measure of a CIS is expressed as *CIS of Order* that is defined as an n -tuple of ISs such that the maximum distance between the elements is no greater than a fixed bound d_n , the size of the window used. The interesting part of this approach is the possibility of modeling various vector distribution² of the IS, like γ -RV (R function

²The term “distribution” is used to designate a distribution of the insertions which assumes that insertions occur preferentially in the vicinity of the TSS (transcription start site, the location where transcription starts at the 5'-end of a gene sequence), but are uniformly distributed in the remainder of the genome

`Cisretro`) or, like in our scenario, a lenti distribution (R function `Cislenti`). The Monte Carlo simulation works with the functions `Cislenti`, `Cisretro` to give the expected numbers and p-values of CISs and ISs involved in CIS based on a γ -RV or lenti IS distribution:

1. Calculates p-values based on the simulated distribution of results.
2. Generates uniformly distributed IS locations.
3. Generates randomly distributed IS near the TSS.
4. Carries out the statistical analysis, compresses highly disconnected genomic regions produced when discarding the interval of the TSS.
5. Counts the CISs
6. Carry out the simulations and count the CIS for each simulation run

Unfortunately, the comparison and the simulations of the IS profiles with known distributions is computational intensive and can require a lot of time, in the case of a high IS number. For this reason it is possible to avoid this step (`Cislenti`, `Cisretro`) and calculate directly the CISs with the fixed sliding window, the function `Cluster`. The input file (in each case) is minimal, a tab-separated file, with header of “chromosome” and “position”, then for each IS the relative chromosome (remember 23 instead X and 24 instead 24 for human genome). Every R function takes in input the data, the interval d for the window and also the RefSeq genomic coordinates³. The output is the same tab-separated file with the two columns of results appended (“CIS Order” and “Cluster”). The column “Cluster” indicates the window to which the IS belongs, useful to remove false positives CISs, like integrations in the proximity of the end or start of a new gene (if the CIS is in a narrow gene for example).

If the control distribution is unavailable, correction may be performed using the integration frequency of genomic region surrounding the CIS interval. However, the different insertional platforms target the genome with different specificities. The distribution of integrations along the different chromosomal regions is very heterogeneous, with scattered clusters of integrations concentrated within the entire length of specific genes, or narrow regions outside or inside transcription units, or specific chromatin marks. To correctly compare the integration frequency at a CIS interval in relation to the surrounding intervals all these factors should be taken in account. To determine whether a CIS gene within the region was a significant outlier compared to other genes contained in the same region, it is possible to use the *Grubbs test*[33], with an available online tool⁴. In order to provide reliable results the Grubbs test requires that at least

³<http://hgdownload.cse.ucsc.edu/downloads.html>

⁴<http://www.graphpad.com/quickcalcs/Grubbs1.cfm>

7 values must be compared and the integration frequencies for all targeted genes in an interval of 10Mb or more, depending on the number of targeted genes contained in the interval. The Grubbs test for outliers requires that the analyzed values are normally distributed. However frequency values may vary from 0 to infinite and thus it is not normally distributed. Therefore, the negative logarithm base 2 transformation of the values was required to pass the D'Agostino and Pearson normality test. This method calculates the ratio Z of the normalized integration frequency for each targeted gene. The ratio Z is obtained as the absolute value of the difference between the normalized integration frequency of a given gene and the average of the frequencies of all genes contained in the interval divided by the standard deviation:

$$Z = \frac{|mean - value|}{standard - dev} \quad (4.4)$$

The ratio Z is then t-studentized (normalized to a t distribution) by the following formula. N is the number of values (genes) analyzed:

$$T = \sqrt{\frac{N(N-2)Z^2}{(N-1)^2 - NZ^2}} \quad (4.5)$$

The approximate p value is calculated for the t-student distribution. Genes targeted at significantly higher or lower integration frequency would both be considered outliers. It will be necessary to verify if the integration frequency is higher or lower than the average for the correct interpretation of the Z ratios and p values. The assumption underlying these forward genetic screens is that insertion is essentially a random process and that the occurrence of multiple integrations within a narrow genomic region in independently derived tumors provides evidence of a selective advantage and positive selection of cells bearing these events. Indeed, *driver mutations*, particularly those which happen earlier, are selected and clonally propagated. Conversely, passenger mutations should ideally be present only in a small number of cells. Moreover, even if some passenger integrations may be over-represented in a specific tumoral mass since they co-occurred early in the tumor progression, it is extremely unlikely that the same will occur in independent tumors. Therefore, clustering of insertions into CIS scores for a role in carcinogenesis of the genes encoded within that region. An example of CIS retrieval and identification with R is in Appendix C.2.2.

Chapter 5

Cancer Progression Modeling

There are several competing approaches to modeling cancer progression [101], some of which incorporate some observed effects as *cancer hallmarks*, *heterogeneity in cell-types*, *drug responses and resistance development* and can be described as in Chapter 2.1. The approach described here try to understand initiation and progression of cancer in terms of “chronological” and “causal” relations among somatic alterations as they occur in the genomes and manifest as point mutations, structural alterations, DNA methylation and histone modification changes. A cell, through mutations, acquires the ability to ignore anti-growth signals from the body, this cell may thrive and divide, and its progeny may eventually dominate part of the tumor. This *clonal expansion* can be seen as a *discrete state* of the cancer progression, marked by the acquisition of a set of genetic events. Cancer progression can then be thought of as a sequence of these discrete steps, where the tumor acquires certain distinct properties at each state.

5.1 Introduction

5.1.1 New Usage of Causation

In [102] the authors introduced a novel theoretical framework for the reconstruction of the causal topologies underlying cumulative progressive phenomena, based on the *probability raising* (PR) notion of causation.

Definition 1 (Probabilistic Causation, [103]) *For any two events c and e , occurring respectively at times t_c and t_e , under the mild assumptions that $0 < \mathcal{P}(c), \mathcal{P}(e) < 1$, the event c causes the event e if it occurs before the effect and the cause raises the probability of the effect, i.e.*

$$t_c < t_e \quad \text{and} \quad \mathcal{P}(e | c) > \mathcal{P}(e | \bar{c}). \quad (5.1)$$

The authors remark that they consider cross-sectional data where no information about t_c and t_e is available, so they are restricted to consider solely the *probability raising* (PR) property, i.e. $\mathcal{P}(e | c) > \mathcal{P}(e | \bar{c})$. Now some properties.

Proposition 1 (Dependency) *Whenever the PR holds between two events a and b , then the events are statistically dependent in a positive sense, i.e.*

$$\mathcal{P}(b | a) > \mathcal{P}(b | \bar{a}) \iff \mathcal{P}(a, b) > \mathcal{P}(a)\mathcal{P}(b). \quad (5.2)$$

Notice that the opposite implication holds as well: when the events a and b are still dependent but in a negative sense, i.e. $\mathcal{P}(a, b) < \mathcal{P}(a)\mathcal{P}(b)$, the PR does not hold, i.e. $\mathcal{P}(b | a) < \mathcal{P}(b | \bar{a})$.

Should be good to use the asymmetry of the PR to determine whether a pair of events a and b satisfy a causation relation so to place a before b in the progression tree but, unfortunately, the PR satisfies the following property.

Proposition 2 (Mutual PR) $\mathcal{P}(b | a) > \mathcal{P}(b | \bar{a}) \iff \mathcal{P}(a | b) > \mathcal{P}(a | \bar{b})$.

That is, if a raises the probability of observing b , then b raises the probability of observing a too. The PR is not symmetric, and the *direction* of probability raising depends on the relative frequencies of the events. Authors make this asymmetry precise in the following proposition.

Proposition 3 (Probability Raising and Temporal Priority) *For any two events a and b such that the probability raising $\mathcal{P}(a | b) > \mathcal{P}(a | \bar{b})$ holds:*

$$\mathcal{P}(a) > \mathcal{P}(b) \iff \frac{\mathcal{P}(b | a)}{\mathcal{P}(b | \bar{a})} > \frac{\mathcal{P}(a | b)}{\mathcal{P}(a | \bar{b})}. \quad (5.3)$$

That is, given that the PR holds between two events, a raises the probability of b more than b raises the probability of a , if and only if a is observed more frequently than b . Notice that they use the ratio to assess the PR inequality. Given these results, it is possible to define the following notion of causation.

Definition 2 *They state that a causes b if a is a probability raiser of b , and it occurs more frequently: $\mathcal{P}(b | a) > \mathcal{P}(b | \bar{a})$ and $\mathcal{P}(a) > \mathcal{P}(b)$.*

Finally, they recall the conditions for the PR to be computable: every mutation a should be observed with probability strictly $0 < \mathcal{P}(a) < 1$. Moreover, they need each pair of mutations (a, b) to be *distinguishable* in terms of PR, that is $\mathcal{P}(a | \bar{b}) < 1$ or $\mathcal{P}(b | \bar{a}) < 1$ similarly to the above condition. Any non-distinguishable pair of events can be merged as a single composite event.

Algorithm 1 Tree-alike reconstruction with shrinkage estimator

- 1: consider a set of genetic events $G = \{g_1, \dots, g_n\}$ plus a special event \diamond , added to each sample of the dataset;
- 2: define a $n \times n$ matrix M where each entry contains the shrinkage estimator

$$m_{i \rightarrow j} = (1 - \lambda) \cdot \frac{\mathcal{P}(j | i) - \mathcal{P}(j | \bar{i})}{\mathcal{P}(j | i) + \mathcal{P}(j | \bar{i})} + \lambda \cdot \frac{\mathcal{P}(i, j) - \mathcal{P}(i)\mathcal{P}(j)}{\mathcal{P}(i, j) + \mathcal{P}(i)\mathcal{P}(j)}$$

according to the observed probability of the events i and j ;

- 3: [PR causation] define a tree $\mathcal{T} = (G \cup \{\diamond\}, E, \diamond)$ where $(i \rightarrow j) \in E$ for $i, j \in G$ if and only if:

$$m_{i \rightarrow j} > 0 \quad \text{and} \quad m_{i \rightarrow j} > m_{j \rightarrow i} \quad \text{and} \quad \forall i' \in G, m_{i, j} > m_{i', j}.$$

- 4: [Correlation filter] define $G_j = \{g_i \in G \mid \mathcal{P}(i) > \mathcal{P}(j)\}$, replace edge $(i \rightarrow j) \in E$ with edge $(\diamond \rightarrow j)$ if, for all $g_w \in G_j$, it holds

$$\frac{1}{1 + \mathcal{P}(j)} > \frac{\mathcal{P}(w)}{\mathcal{P}(w) + \mathcal{P}(j)} \frac{\mathcal{P}(w, j)}{\mathcal{P}(w)\mathcal{P}(j)}.$$

5.1.2 Shrinkage Estimator and Progression Tree Extraction

Besides such a probabilistic notion, they should introduce the use of a shrinkage estimator to efficiently remove noisy. The reconstruction method is described in Algorithm 1. The algorithm is very similar in spirit to [71] algorithm, with the main difference being an alternative weight function based on this shrinkage estimator.

Definition 3 (Shrinkage Estimator) *They define the shrinkage estimator $m_{a \rightarrow b}$ of the confidence in the causation relationship from a to b as*

$$m_{a \rightarrow b} = (1 - \lambda)\alpha_{a \rightarrow b} + \lambda\beta_{a \rightarrow b}, \quad (5.4)$$

where $0 \leq \lambda \leq 1$ and

$$\alpha_{a \rightarrow b} = \frac{\mathcal{P}(b | a) - \mathcal{P}(b | \bar{a})}{\mathcal{P}(b | a) + \mathcal{P}(b | \bar{a})} \quad \beta_{a \rightarrow b} = \frac{\mathcal{P}(a, b) - \mathcal{P}(a)\mathcal{P}(b)}{\mathcal{P}(a, b) + \mathcal{P}(a)\mathcal{P}(b)}. \quad (5.5)$$

This estimator combines a normalized version of the PR, the raw model estimate α , with the correction factor β . The shrinkage aims at improving the performance of the *overall* reconstruction process, not limited to the performance of the estimator itself. In other words, m induces an ordering to the events reflecting our confidence for their causation. However, this framework does not imply any performance bound for the, e.g., mean squared error of m .

CAPRESE (CAnCER PRogression Extraction with Single Edges) is based on two

main steps:

1. Instead using correlation to infer progression structures, authors base their technique on a notion of probabilistic causation.
2. To increase robustness against noise, they adopt a shrinkage-like estimator to measure causation among any pair of events.

5.2 CAPRI

Results that extend tree representations of cancer evolution exploit mixture tree models, i.e. multiple oncogenic trees, each of which can independently result in cancer development [104]. All these methods are capable of modeling diverging temporal orderings of events in terms of branches, although the possibility of converging evolutionary paths is precluded. To overcome this limitation, the most recent approaches tend to adopt *Bayesian Graphical Models*, i.e. Conjunctive Bayesian Networks (CBN), Chapter 2.1.

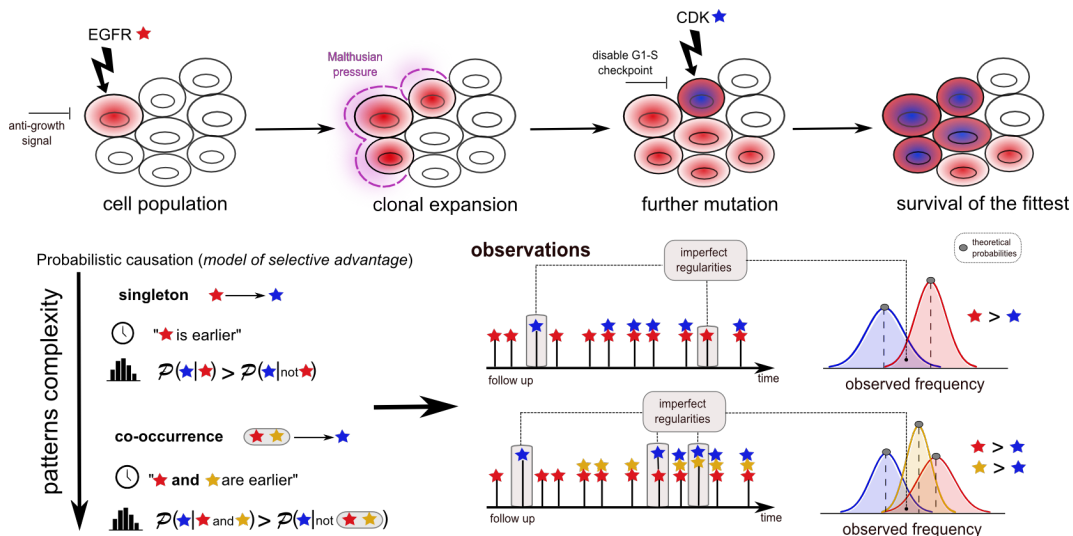


Figure 5.1: CAPRI Overview [11]. The algorithm examines cancer cross-sectional data to determine relationships related to genomic alterations (e.g., somatic mutations, copy-number variations, etc.) that modulate the somatic evolution of a tumor. When CAPRI concludes that mutation (EGFR) “selects for” aberration b (CDK mutation), such relations can be rigorously expressed using Suppes’ conditions, which postulates that if a selects b, then a occurs before b (*temporal priority*) and occurrences of a raises the probability of emergence of b (*probability raising*).

Algorithm 2 *C*Ancer *P*Rogression *I*nference (CAPRI)

- 1: **Input:** A set of events $G = \{g_1, \dots, g_n\}$, an $m \times n$ matrix $D \in \{0, 1\}^{m \times n}$ and k CNF causal claims $\Phi = \{\varphi_1 \triangleright e_1, \dots, \varphi_k \triangleright e_k\}$ where, for any i , $e_i \not\sqsubseteq \varphi_i$ and $e_i \in G$;
- 2: [*Lifting*] Define the *lifting of D to $D(\Phi)$* as the augmented matrix

$$D(\Phi) = \begin{pmatrix} D_{1,1} & \dots & D_{1,n} & \varphi_1(D_{1,\cdot}) & \dots & \varphi_k(D_{1,\cdot}) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ D_{m,1} & \dots & D_{m,n} & \varphi_1(D_{m,\cdot}) & \dots & \varphi_k(D_{m,\cdot}) \end{pmatrix}. \quad (5.6)$$

by adding a column for each $\varphi_i \triangleright c_i \in \Phi$, with φ_i evaluated row-by-row, define the coefficients

$$\Gamma_{i,j} = \mathcal{P}(i) - \mathcal{P}(j), \quad \text{and} \quad \Lambda_{i,j} = \mathcal{P}(j | i) - \mathcal{P}(j | \bar{i}), \quad (5.7)$$

pair-wise over $D(\Phi)$;

- 3: [*DAG structure*] Define a DAG $\mathcal{D} = (N, \pi)$ where

$$N = G \cup \left(\bigcup_{\varphi_i} \text{chunks}(\varphi_i) \right), \quad \pi(j \notin G) = \emptyset;$$

$$\pi(j \in G) = \left\{ i \in G \mid \Gamma_{i,j} \wedge \Lambda_{i,j} > 0 \right\} \cup \left\{ \text{chunks}(\varphi) \mid \Gamma_{\varphi,j} \wedge \Lambda_{\varphi,j} > 0, \varphi \triangleright j \in \Phi \right\}. \quad (5.8)$$

- 4: [*DAG labeling*] Define the labeling α as follows

$$\alpha(j) = \begin{cases} \mathcal{P}(j), & \text{if } \pi(j) = \emptyset \text{ and } j \in G; \\ \mathcal{P}(j | i_1 \wedge \dots \wedge i_n), & \text{if } \pi(j) = \{i_1, \dots, i_n\}. \end{cases}$$

- 5: [*Likelihood fit*] Filter out all spurious causes from \mathcal{D} by likelihood fit with the regularization BIC score and set $\alpha(j) = 0$ for each removed connection.
- 6: **Output:** the DAG \mathcal{D} and α ;

A new algorithm proposed in [11] is called CAnceR PRogression Inference (CAPRI) and is part of the TRanslational ONCOlogy (TRONCO) R package [105]. From cross-sectional genomic data, CAPRI reconstructs a probabilistic progression model by inferring *selectivity advantage relations*, where a mutation in a gene A “selects” for a later mutation in a gene B. These relations are depicted in a combinatorial graph and resemble the way a mutation exploits its “selective advantage” to allow its host cells to expand clonally. These relations are expected to also imply *probability raising* for a pair of events. A selectivity relation between a pair of events signifies that the presence of the earlier genomic alteration (i.e., the upstream event) that is advantageous in a Darwinian competition scenario increases the probability with which a subsequent advantageous genomic alteration (i.e., the downstream event) appears in the clonal evolution of the tumor.

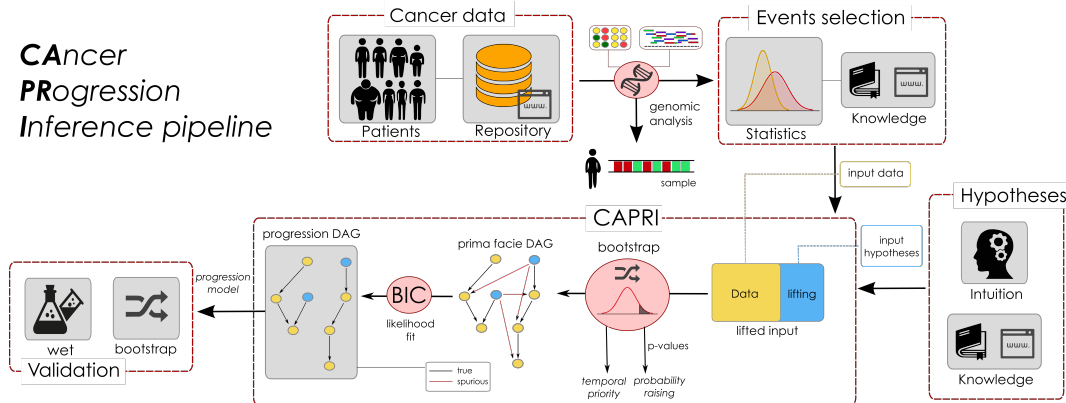


Figure 5.2: CAPRI Pipeline [11]. The first step is to collect experimental data and perform genomic analyses to derive profiles of, e.g., somatic mutations or copy-number variations for each patient or sample. Then, statistical analysis and biological priors are used to select events relevant to the progression and imputable by CAPRI - e.g., *driver mutations*. CAPRI can extract a progression model from these data and to assess various confidence measures on its constituting relations - e.g., (non-)parametric bootstrap and hypergeometric testing. Experimental validation concludes the pipeline.

This approach has been tested by authors in [11] for Atypical Chronic Myeloid Leukemia (AML), data taken from [106].

In Figure 5.3 CAPRI predicts a progression involving mutations in SETBP1, ASXL1 and CBL, consistently with the recent study by [107], in which these genes were shown to be highly correlated and possibly functioning in a synergistic manner for aCML progression. Specifically, CAPRI predicts a selective advantage relation between missense point mutations in SETBP1 and nonsense point mutations in ASXL1.

Among the hypotheses given as input to CAPRI, the algorithm seems to suggest that the exclusivity pattern among ASXL1 and SF3B1 mutations selects for CBL missense point mutations. The role of the ASXL1/SF3B1 exclusivity pattern is consistent with the study of [108] which shows that, on 479 MDS patients, mutations in SF3B1 ARE inversely related to ASXL1 mutations.

Finally, CAPRI predicts selective advantage relations among TET2 and EZH2 missense point and indel mutations. Even though the limited sample size does not allow to draw definitive conclusions on the ordering of such alterations, we can hypothesize that they may play a synergistic role in aCML progression.

Encouraged by CAPRI's ability to infer interesting relationships in a complex disease such as aCML and the ability to use cross-sectional data, I expect that CAPRI will help uncover relationships to aid our understanding of cancer and eventually improve targeted therapy design, in particular to *anticancer-drug-resistance studies*.

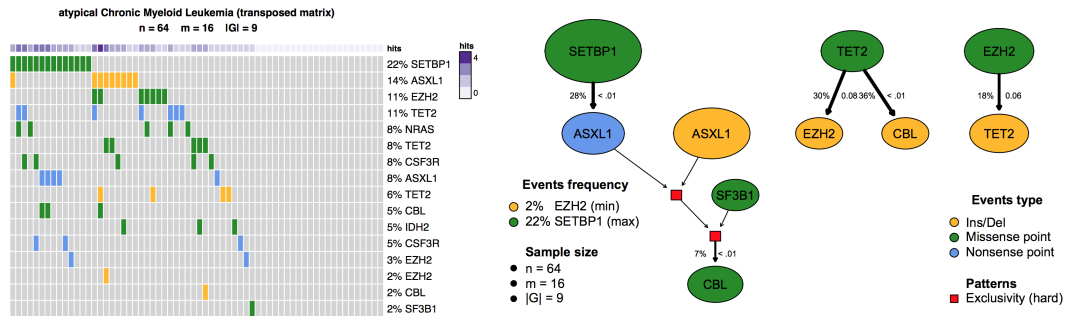


Figure 5.3: CAPRI Pipeline for AML [11]. (left) Mutational profiles of $n = 64$ aCML patients - exome sequencing - with alterations in $|G| = 9$ genes with either mutation frequency $> 5\%$ or belonging to an hypothesis inputted to CAPRI. Mutation types are classified as nonsense point, missense point and insertion/deletions, yielding $m = 16$ input events. Purple annotations report the frequency of mutations per sample. (right) Progression model inferred by CAPRI in supervised mode. Node size is proportional to the marginal probability of each event, edge thickness to the confidence estimated with 1000 non-parametric bootstrap iterations. The p-value of the hypergeometric test is displayed too.

Some limitations of this approach (general, not necessary linked to CAPRI) are:

1. The selection of the nodes (genes). If you not have a selection criteria put in input a non-curated list of genes can be misleading and at the end the result interpretation can be impossible or not confident.
2. To improve the confidence and to reduce the bias generated by the use a non-curated list of genes is necessary to have a large number of samples, not always possible.
3. For drug resistance studies in cancer, unless you have the pre and the post treatment samples, is impossible to reconstruct the progression DAG due to a drug treatment.
4. The crucial aspect, for all these models, is the final interpretation of the results (DAGs, progression, links...).

5.3 Integration Site Data and Causal Modeling

The identification of genomic regions recurrently targeted by ISs (CIS) in tumors induced by insertional mutagens has allowed the discovery of new cancer driver genes, as in [1, 109]. As I will discuss in the next chapter my institute has developed a new strategy to study some of these aspects.

In our scenarios the CIS quantification (*CIS Order*) is an important improvement in every causal model reconstruction, independently of the method used to build the progression. Indeed the CISs with high order are the most confident candidates to be the driver genes of a specific phenotype¹, [2, 3, 11, 12].

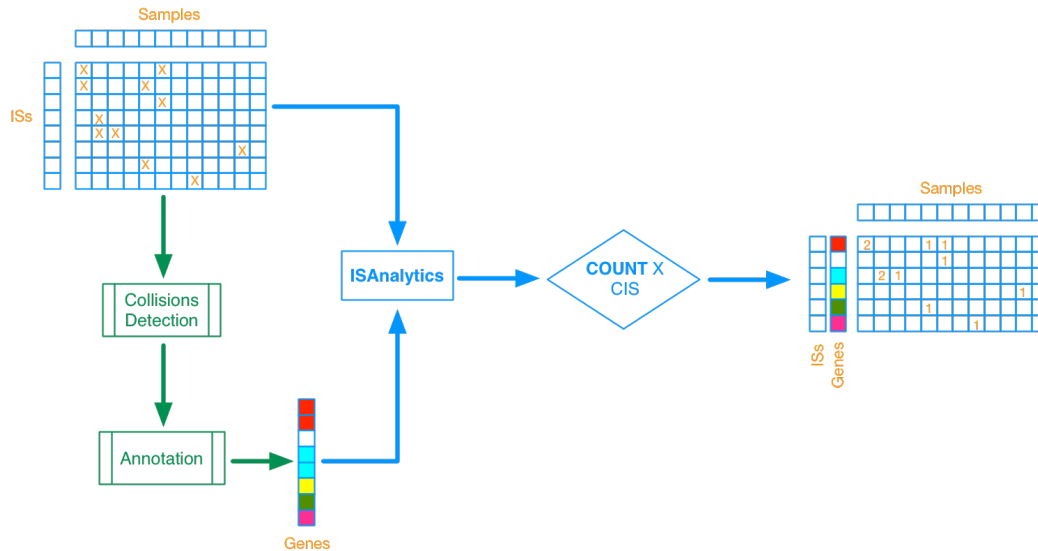


Figure 5.4: Data Generation, step 1: after IS matrix creation, annotation and collision removal with ISAnalytics, I perform a count of ISs for each single CIS with $SC \geq 3$.

After the VISPA2's run, the IS matrix generation with `create_matrix` program (described in Chapter 3), the annotation to the reference genome with `annotate_matrix` (described in Chapter 3) and ISAnalytics to remove the collisions against other different projects (contaminations, described in Chapter 4), it is necessary to define a correct input for CAPRI, Figure 5.4. To count the number of different ISs per CIS (gene) should be useful to insert a threshold on sequence count (≥ 3 , as used in [19, 49]), to reduce false positives. For this scope a function in ISAnalytics can be useful (described in Chapter 4), `scFilter`. After the selection of the corrected CISs the count of ISs per gene can be completed for the next step.

¹A phenotype is the composite of an organism's observable characteristics or traits, such as its morphology, development, biochemical or physiological properties, behavior and products of behavior (such as a bird's nest). A phenotype results from the expression of an organism's genetic code, its genotype, as well as the influence of environmental factors and the interactions between the two. When two or more clearly different phenotypes exist in the same population of a species, the species is called polymorphic. *Wikipedia*

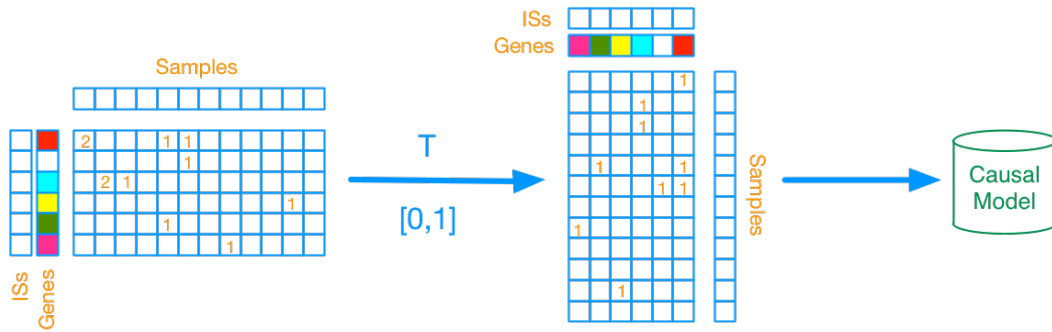


Figure 5.5: Data Generation, step 2: The count matrix of CISs is now binarized and transposed for CAPRI input.

The IS matrix per CIS now must be transposed and binarized (1 if there is at least an IS in a CIS, 0 otherwise) to lift the input for CAPRI.

For the limitations described in the section before it is very important to input in CAPRI only the highly targeted genes, changing the CAPRI's classic input, thus taking advantage of the possibility of using cross-sectional data, instead of mutation type with CISs that cut off the first two previous described limitations:

1. The selection of the nodes (genes). *Using CISs allows to take in consideration for the progression DAG only the driver genes and not the passengers.*
2. *With CISs as input the number of necessary samples can be reduced a lot, more or less 1 sample for 1 CIS.*

Chapter 6

Case Study: Breast Cancer

My laboratory developed a new lentiviral vector-based insertional mutagenesis screening to identify genes that confer resistance to clinically relevant targeted anti-cancer therapies [3]. By applying this approach to cell lines representing two subtypes of HER2+ breast cancer [110], we identified 62 candidate Lapatinib resistance genes. We validated in vitro the top ranking genes by showing that their forced expression confer resistance to Lapatinib and found that their mutation or over-expression is associated to poor prognosis in human breast tumors [111, 112, 113, 114]. As a proof of concept of the flexibility of our platform, we then successfully applied it to the identification of erlotinib resistance genes in pancreatic cancer. This experimental platform can be easily applied to different types of cancer and drugs. The acquired knowledge can help identifying combinations of targeted drugs to overcome the occurrence of resistance, thus opening new horizons for more effective treatment of tumors. We decided to apply LV-based insertional mutagenesis screening to identify genes whose deregulation is involved in resistance to Lapatinib, a HER2-inhibitor recently approved for the treatment of metastatic HER2+ breast cancer; as targets for insertional mutagenesis we used the BT474 and SKBR3 cell lines¹.

6.1 Experimental Strategy Description

The experimental outline of the in vitro insertional mutagenesis screen for the identification of drug resistance genes developed in my institute (SR-Tiget) is represented in Figure 6.1. Initially drug sensitive cells² (a) are infected with a mutagenic vector which induce random genome-wide mutations. When insertional mutagenesis deregulates genes with roles in drug response, drug-resistant clones (cell clones in which the drug has no effects anymore) are generated (b). Drug treatment promotes the positive

¹List of breast cancer cell lines, https://en.wikipedia.org/wiki/List_of_breast_cancer_cell_lines

²Several effective drugs having differing mechanisms of action are available. In this case cells sensitive to Lapatinib.

selection of resistant clones (c). DNA is then extracted from resistant colonies (d) and vector integration sites are retrieved using LAM-PCR (e). Vector-genome junctions are sequenced (f) and aligned to the reference genome, and statistical analysis is performed to identify the genes which are recurrently targeted (CISs) and which therefore may have a role in inducing drug resistance.

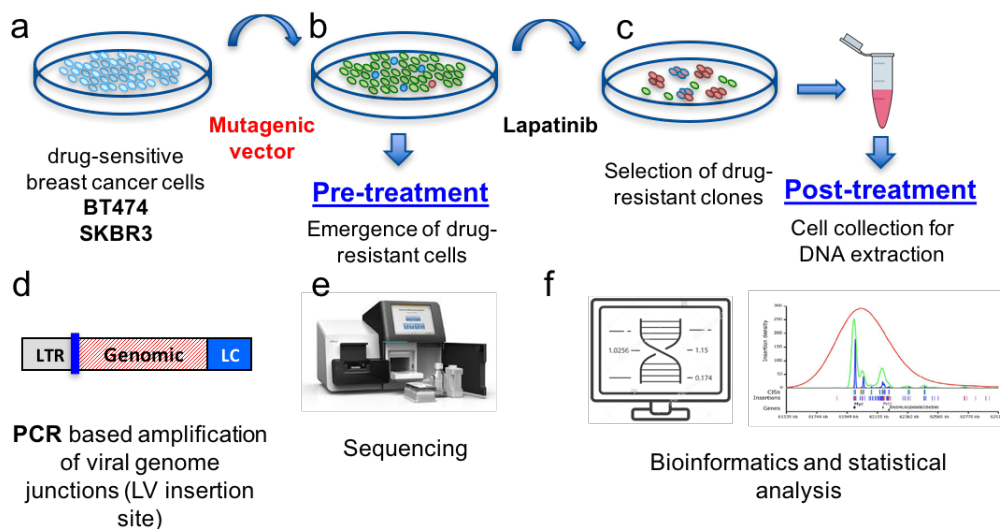


Figure 6.1: Anti-Cancer Drug Resistance Forward Screening Strategy at SR-Tiget.

Obviously my activity regards the point (f) of Figure 6.1. In the following sections two different results will be reported, based on two different pipelines, VISPA (at the time of our strategy paper [3]) and VISPA2 (results, CISs, after the development of the new version of the pipeline). The comparison only in terms of CISs will be described (Appendix D.3), all the other data are referred to VISPA2's output.

6.2 CAPRI on Breast Cancer Data

This project has produced 4 Illumina MiSeq runs with, at the end, more than 8 millions of LAM-PCR products. The experimental strategy of anti-cancer drug resistance screening has been applied on two different cell lines (BT474 and SKBR3), Table 6.1. In this work only data of BT474 will be processed for simplicity.

In [3], with VISPA, the authors obtained the CISs in Table 6.2, for BT474 cell line. Before applying the strategy described at the end of Chapter 5, it is necessary to test the mutual exclusivity of the CISs (genes). I used Mutex to verify the the invalidity of this hypothesis [16], for all CISs with *CIS Order* > 5. Verified the goodness of the input (Appendix D.1, score > 0.05), the R script is in Appendix C.3.1 and necessary

ISs Output on VISPA2			
Cell Line	Condition	N of Samples	N of ISs
BT474	Control (Pre-Treatment)	16	6,031
BT474	Drug (Post-Treatment)	12	9,202
SKBR3	Control (Pre-Treatment)	16	11,892
SKBR3	Drug (Post-Treatment)	12	12,438

Table 6.1: VISPA2 output of breast cancer experiment. For each cell line there are 2 conditions, before (Pre-Treatment) and after (Post-Treatment) the drug treatment. The number of samples is the same for the 2 conditions in the 2 cell lines, but the number of ISs retrieved is relatively different.

CISs for BT474 Cell Line	
Pre-Treatment	Post-Treatment
FBXL20*	PIK3CA
VMP1	CDK12
SUMO1P1	BCAS1
ITCH	ITCH
DENND1B	NCOA3
MIPOL1	VMP1
ARHGAP39	KIFAP3
ACACA	ARHGAP39
STXBP4	PKIA
C8orf83	CADM2
LOC100131234	SLITRK6
KCNV1	CSMD3

Table 6.2: CISs obtained with VISPA (*CIS Order* > 10), identified with [9, 33].

*FBXL20 is identified as ERBB2 gene.

files for CAPRI's pipeline are the following:

`bt474-types.txt` (comma-separated) file identify the content type and the color of the nodes (CIS), in this case a light blue.

CIS , cornflowerblue

For the pre-treatment: `bt474-events-pre-treatment.txt` (comma-separated) file identify the events (CISs, genes) with their relative order.

FBXL20 , CIS , 1

VMP1 , CIS , 2

SUMO1P1 , CIS , 3

ITCH , CIS , 4

DENND1B , CIS , 5

MIPOL1 , CIS , 6

ARHGAP39 , CIS , 7

ACACA , CIS , 8
 STXBP4 , CIS , 9
 C8orf83 , CIS , 10
 LOC100131234 , CIS , 11
 KCNV1 , CIS , 12

bt474-data-pre-treatment.txt (tab-separated) file contains the binary matrix, as described in Chapter 5. Each row is a CIS (gene) and each column is a sample.

```

1 1 1 1 1 1 0 0 1 1 1 1
0 0 1 0 0 0 0 0 0 0 0 1
0 0 1 0 0 0 0 0 0 1 0 0
0 0 1 1 0 0 0 0 0 1 0 0
0 0 1 0 0 0 0 0 0 1 0 0
1 0 0 0 0 0 0 0 0 0 1 1
1 1 1 0 0 1 0 0 1 1 0 0
0 1 0 1 1 0 0 0 1 0 0 0
1 1 1 1 0 1 1 1 1 1 1 1
1 1 1 1 1 1 0 1 1 1 1 1
1 1 1 1 1 1 0 0 1 1 1 1
0 0 1 0 0 0 0 0 0 0 0 1
0 0 1 1 0 0 0 0 0 1 0 0
1 1 1 0 0 1 0 0 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 0 1 1 1 1 1

```

For the post-treatment: bt474-events-post-treatment.txt (comma-separated) file identify the events (CISs, genes) with their relative order.

PIK3CA , CIS , 1
 CDK12 , CIS , 2
 BCAS1 , CIS , 3
 ITCH , CIS , 4
 NCOA3 , CIS , 5
 VMP1 , CIS , 6
 KIFAP3 , CIS , 7
 ARHGAP39 , CIS , 8
 PKIA , CIS , 9
 CADM2 , CIS , 10
 SLITRK6 , CIS , 11
 CSMD3 , CIS , 12

`bt474-data-post-treatment.txt` (tab-separated) file contains the binary matrix, as described in Chapter 5. Each row is a CIS (gene) and each column is a sample.

```

1 0 1 1 0 1 0 0 0 1 1 1
1 1 1 1 0 0 0 0 0 0 1 1
1 1 1 1 1 1 1 1 1 1 1 1
0 1 1 0 0 1 1 0 1 1 1 1
1 1 1 1 1 1 0 1 0 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1
1 0 0 1 1 0 0 0 0 0 1 1
1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1
0 1 1 0 0 1 1 0 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1
0 0 0 0 0 0 0 0 0 0 0 0
    
```

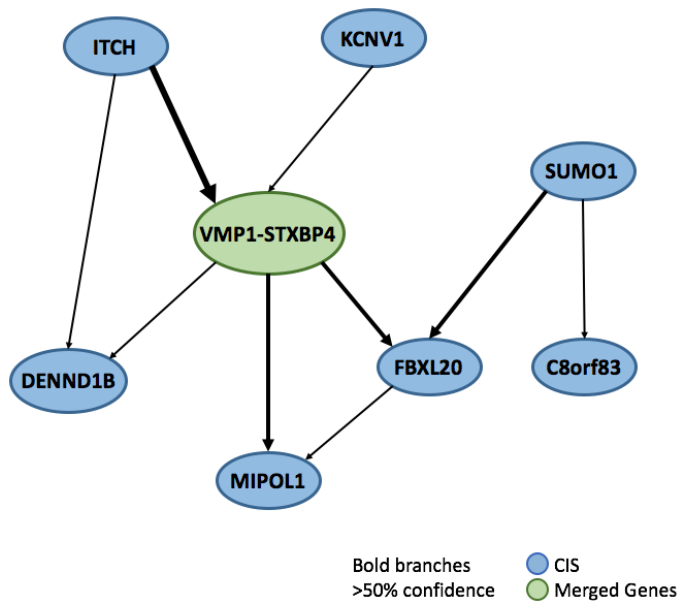


Figure 6.2: Anti-Cancer Drug Resistance progression for BT474 cell line in pre-treatment condition. CISs are highlighted in blue, in green the CISs that CAPRI cannot distinguish (CISs in same samples), in this case the two CISs have the same progression. The edges in bold have a confidence > 50%. I used *Cytoscape* to refine the figure [27].

The output of the algorithm is composed of two DAGs, in which each node is a CIS (gene), the edges the relations between them in terms of causality, selective advantage

relations among driver genes. The thickness of the edges represents the confidence of the relation. Any inequality (i.e., checking *temporal priority* and *probability raising*) is estimated using the non-parametric Mann-Whitney U [115] test with p-values set to 0.05. Authors of [11] compute confidence p-values for both *temporal priority* and *probability raising* using this test, which need not assume Gaussian distributions for the populations. In each graph the bold edges have a confidence $> 50\%$.

The progression, the topology of the DAG and the relations between CISs should be investigated in detail, but immediately one aspect emerges, the most confident relations regard the CISs with the highest *CIS Order*. This confirms that the selection of the CIS was well done.

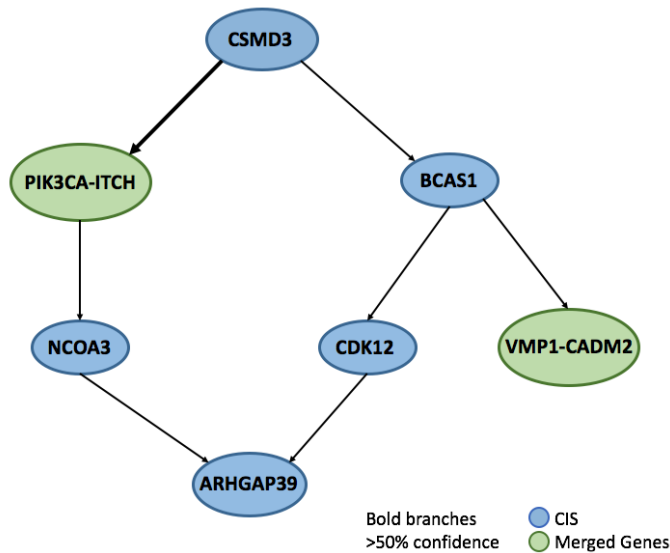


Figure 6.3: Anti-Cancer Drug Resistance progression for BT474 cell line in post-treatment condition. CISs are highlighted in blue, in green the CISs that CAPRI cannot distinguish (CISs in same samples), in this case the two CISs have the same progression. The edges in bold have a confidence $> 50\%$. I used *Cytoscape* to refine the figure [27].

Because the limited number of samples, the progression is less detailed, two of the most targeted genes seems to be related to the activity of CSMD3. In this case the *CIS Order* is very unbalanced between PIK3CA and the others, for this reason the confident relations are less in terms of numbers. Should be interesting investigate the relation between PIK3CA and CSMD3.

I have tried, at the first time, the CAPRI pipeline without the test of mutual exclusivity of the CISs and with no distinction of pre and post treatment, and I have produced the progression represented in Appendix D.2. This approach is terribly wrong, for the following reasons:

1. Without any distinction between pre and post treatment the causal model used in CAPRI cannot correctly identify the real progression generated by the drug treatment. It can only produce the DAG of the total CISs that is meaningless.
2. The *temporal priority* property is misleading without the distinction of pre and post treatment samples.
3. Mutual exclusivity test for genes is necessary also to avoid problems of redundant pathways [116, 117], redundant information. Not only for CAPRI.
4. The CISs are calculated separately in the two different conditions (pre and post Lapatinib) thus merging CISs is a wrong simplification.

The limitation 3 in Chapter 5 is outdated because in the experiment there are the pre and post treatment conditions. For the limitation 4 I decided to do some enrichment analyses due to improve the DAGs interpretation and investigate in deeply some CISs.

6.3 Enrichment Analysis

In [3], where these data have been generated and analyzed for the first time, GREAT [118], a tool that enables the analysis integrated with gene ontology and genomic coordinates, was used for gene ontology (GO)³ assessments, without any relevant results (because false positives and difficulties of interpretation).

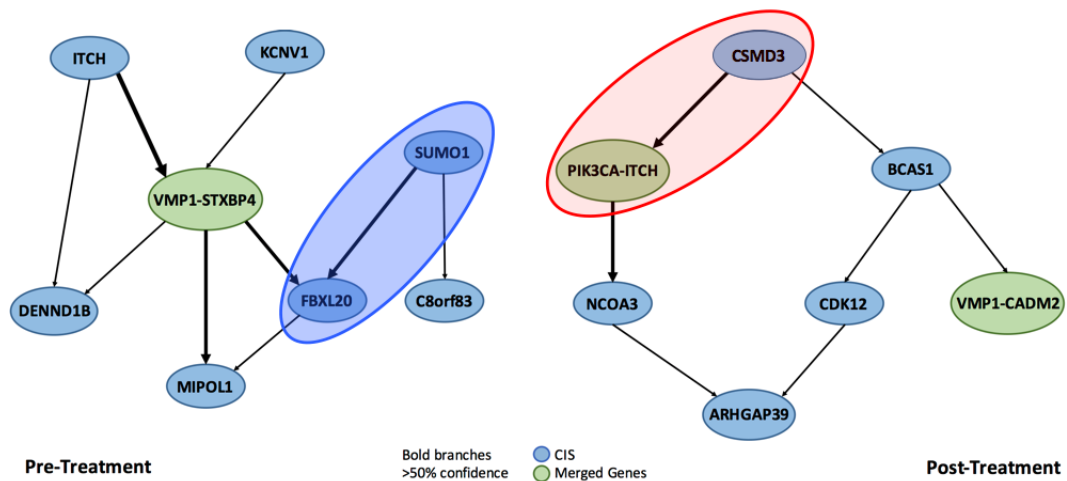


Figure 6.4: Selection of the most relevant relations in pre and post treatment in BT474 breast cancer cell line, from CAPRI. I used *Cytoscape* to refine the figure [27].

³Gene ontology (GO) is a major bioinformatics initiative to unify the representation of gene and gene product attributes across all species.

Exploiting instead the CISs on the DAGs, selecting branches based on *CIS Order* and confidence score and causal relations given by CAPRI, Figure 6.4, I found two analytical tools more appropriate for enrichment analysis: Enrichr [14, 15] and GeneMANIA [13].

6.3.1 Enrichr

Enrichr uses a list of gene symbols as input (RefSeq gene symbols as nomenclature). Each symbol in the input must be on its own line optionally followed by a comma. It is possible to upload the list by either selecting the text file that contains the list or just simply pasting in the list into the text box. On the results page, the analysis is divided into different categories of enrichment, the most important:

- *Pathways*: most common database resources for understanding high-level functions (pathways), like KEGG [32], WikiPathways [119], Reactome [120]...
- *Ontologies*⁴: its built-in tools to produce GO Biological Process, GO Cellular Component, GO Molecular Function...
- *Disease/Drug*: OMIM Disease, OMIM Expanded [121], MSigDB Computational, MSigDB Oncogenic Signatures [122], LINCS L1000 Chem Pert up, LINCS L1000 Chem Pert down, LINCS L1000 Ligand Perturbations up, LINCS L1000 Ligand Perturbations down [123], DrugMatrix [124]...
- *Cell Types*: Cancer Cell Line Encyclopedia [125], Human Gene Atlas, Mouse Gene Atlas [126]...

Statistics is powered by Z-Score of the deviation from the expected rank by the Fisher's exact test.

As input, in this particular case, I split it in two gene lists, for pre and post treatment, as discussed before, with the same list of CISs (genes). Interestingly, in the pre-treatment scenario Enrichr correctly confirmed by blind analysis that the cell line used was the BT474 of breast cancer (the input is only 12 CISs, confirming the role of driver genes), Figure 6.5, and no significant information about pathways related, Figure 6.6.

For the post-treatment CISs, using in the same way Enrichr:

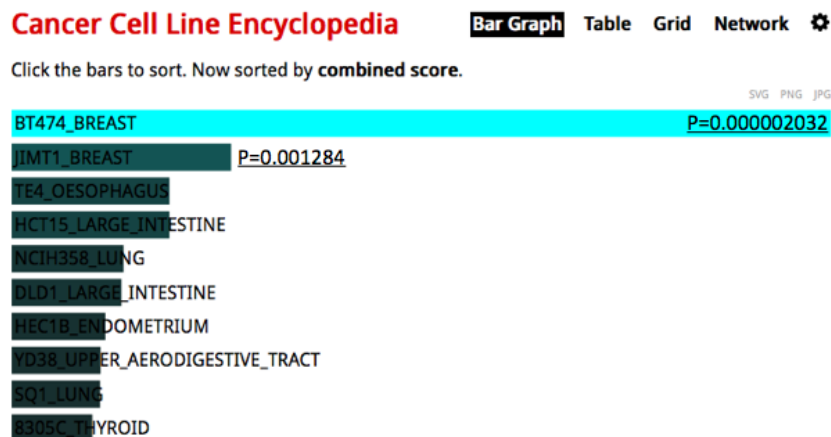


Figure 6.5: Enrichr: Cancer Cell Line Encyclopedia. Correctly shows a relation with BT474 of breast cancer, p -value = 0.000002032

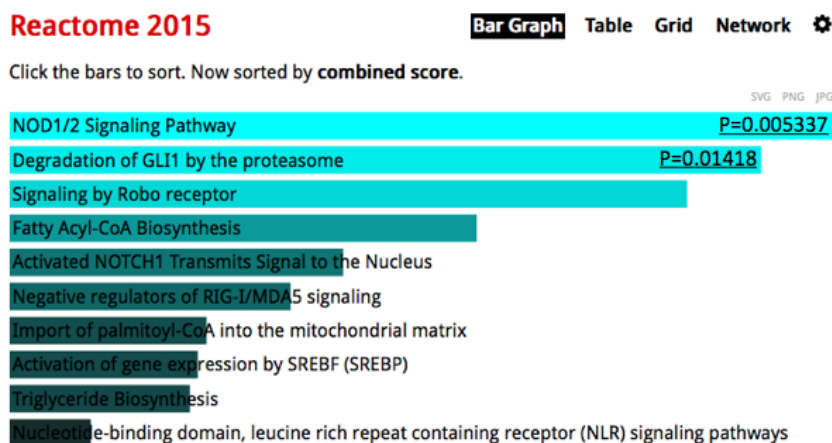


Figure 6.6: Enrichr: Reactome. No significant pathways related with the gene set.

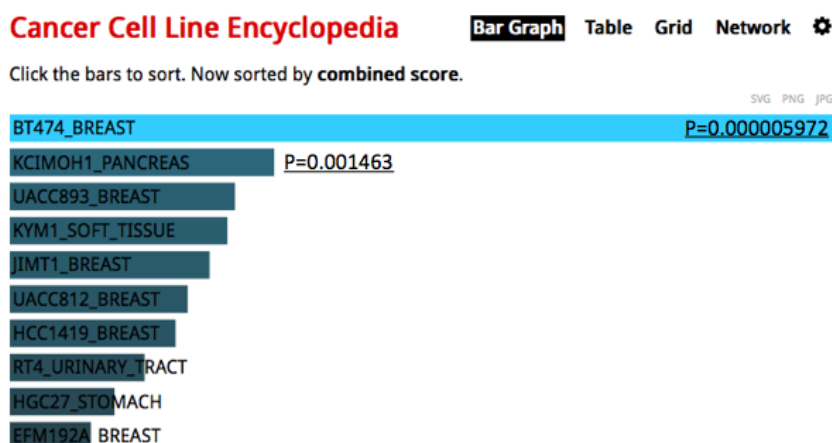


Figure 6.7: Enrichr: Cancer Cell Line Encyclopedia. Correctly shows a relation with BT474 of breast cancer, p -value = 0.000005972. Similar to pre-treatment, indeed this is an intrinsic characteristic of the cell line.

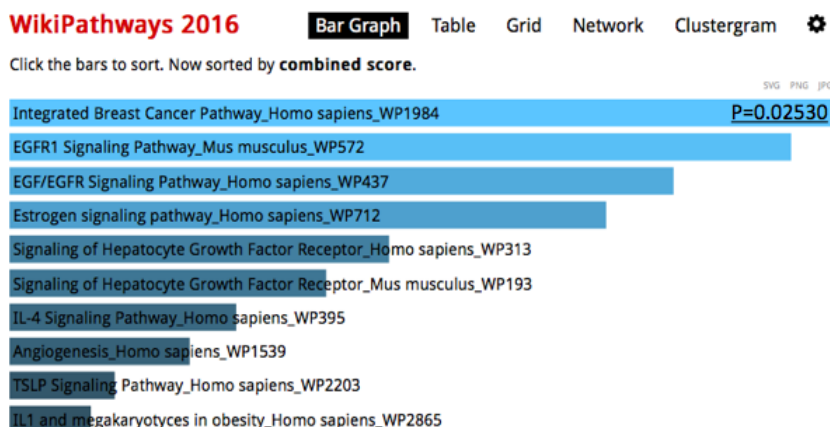


Figure 6.8: Enrichr: Reactome. Significant relation with “Signaling of ERBB4”. Well known signaling pathway in breast cancer, as showed in [28, 29]. Moreover the “EGFR Signaling Pathway” results involved, as confirmed in [30, 31].

Enrichr provided excellent verification of the IS data used (by CISs), confirming strong relationships among CISs and notes related to the cancer cells of breast cancer and the use of Lapatinib as a chemotherapy drug, Figure 6.8.

6.3.2 GeneMANIA

After the confirmations obtained with Enrichr, I wanted to investigate in detail the two highlighted branches: ERBB2-(FBXL20)-SUMO1 for pre-treatment and PIK3CA-CSMD3 for post-treatment.

The progression in Figure 6.4 (blue selection) predicts that SUMO1 is an “early event” for the BT474 cell line and that this event “selects” for FBXL20 (ie, ERBB2, intrinsic mutation of the cell line). According to the theory used to develop CAPRI this is interpreted as follows: *the emergence of a mutation in SUMO1 allows clonal expansion through SUMO1, which, subsequently, is replaced by a clone SUMO1 and FBXL20 (the selected clone contains both mutations)*.

Thus the first step is to verify the relation SUMO1-FBXL20 for the pre-treatment condition with GeneMANIA. It extracts from a curated database relationships between genes; the pathway on which it maps the genes (blue), co-expression (purple) and physical interactions (protein-protein interactions, pink).

The output reports of GeneMANIA listed the literature at the basis of the results. Relations extracted from GeneMANIA for SUMO1 - mainly protein-protein interactions and some pathway-level - involving various genes linked to the Ubiquitin (UBE2I, UB2, UBE2G1...), Figure 6.9.

⁴To GO analysis I recommend the use of GREAT that takes in consideration also different ISs, non only the different genes.

From these results, I can speculate that SUMO1 clones modify the function of the mechanisms linked to Ubiquitin that, in cascade, alter the functioning of the pathway of ERBB2. Interestingly, Ubiquitin is known in the literature to be often linked to breast cancer and other tumors.

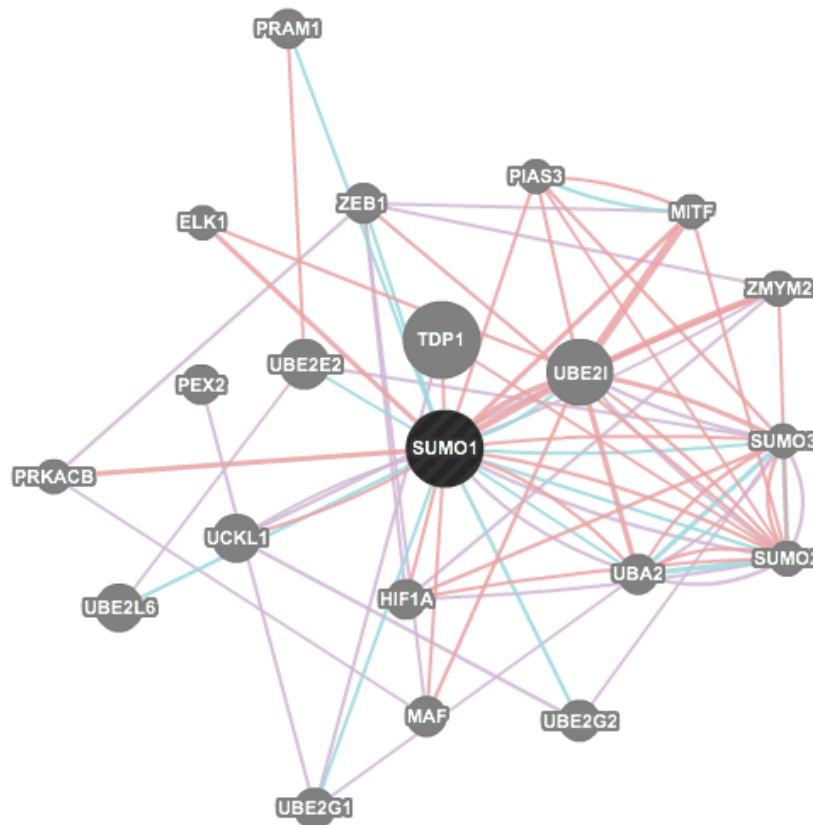


Figure 6.9: GeneMANIA for SUMO1 and pre-treatment. The pathway on which it maps the genes (blue), co-expression (purple) and physical interactions (protein-protein interactions, pink). For details please download, <http://giuliospinozzi.altervista.org/docs/GMSUMO1.pdf>

The second step (selective pressure induced by Lapatinib) is the verification of the similarity between the relationship CSMD3-PIK3CA and SUMO1-FBXL20. To justify resistance to the drug, the functional “relation” SUMO1-FBXL20 should be equivalently in the post-treatment. If so, the progression of the breast may start involving the Ubiquitin mechanisms, and continues through the pathway of ERBB2. Following this interpretation, CSMD3 should be verified that is relevant to the mechanisms linked to Ubiquitin, Figure 6.11.

PIK3CA (Figure 6.10) is also directly involved in the pathway of ERBB2 signaling. In general it appears that: the Ubiquitin is linked (through both pathways, both at

the protein level) to ERBB2 and PIK3CA, the first is precisely the predominant gene in the cell line prior to treatment, the second in the post-treatment with Lapatinib.



Figure 6.10: GeneMANIA for UBC and post-treatment. The pathway on which it maps the genes (blue), co-expression (purple) and physical interactions (protein-protein interactions, pink). For details please download, <http://giuliospinozzi.altervista.org/docs/GMUBC.pdf>

All these findings allowed to draft the hypothesis that exactly the Lapatinib intervenes on SUMO1 clones, along with CSMD3. Note that it is recently discovered that CSMD3 is a potential tumor suppressor gene, and it is believed that PIK3CA is responsible for the response to Lapatinib in HER2+ breast cancer [127, 128, 129].

The selective pressure due to Lapatinib - in this cell line - induces a “change of evolutionary trajectory” (resistance) of breast cancer, which goes from evolve through SUMO1 and ERBB2 (FBXL20) to CSMD3 trajectory and PIK3CA, both paths may be explained mechanistically by Ubiquitin, as confirmed by several papers [127, 128, 129] and pathways, Figure 6.12 (information produced by Enrichr, with only PIK3CA, SUMO1, ERBB2 and CSMD3 as input genes).

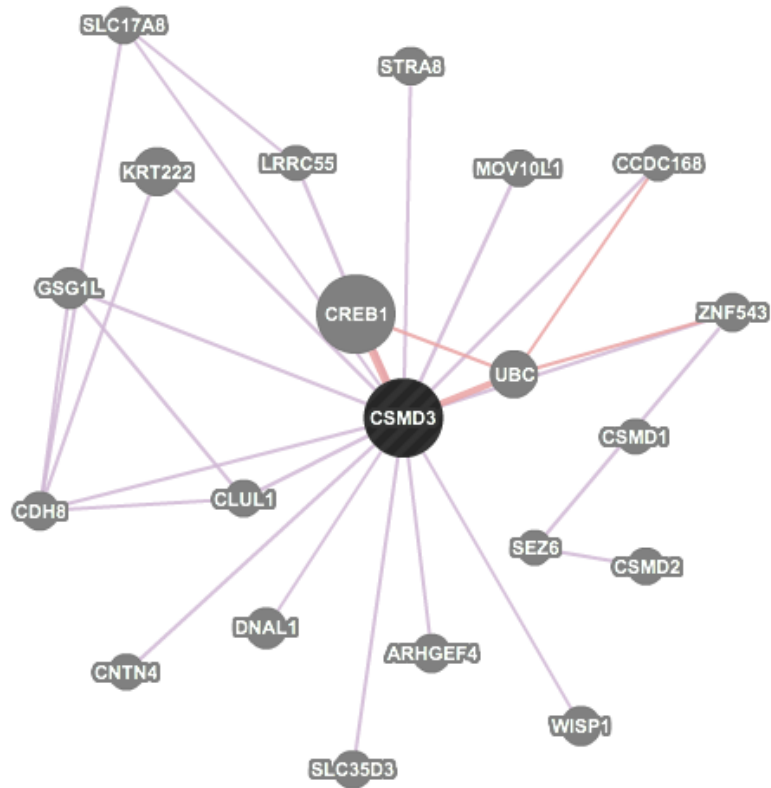


Figure 6.11: GeneMANIA for CSMD3 and post-treatment. The pathway on which it maps the genes (blue), co-expression (purple) and physical interactions (protein-protein interactions, pink). For details please download, <http://giuliospinozzi.altervista.org/docs/GMCSMD3.pdf>

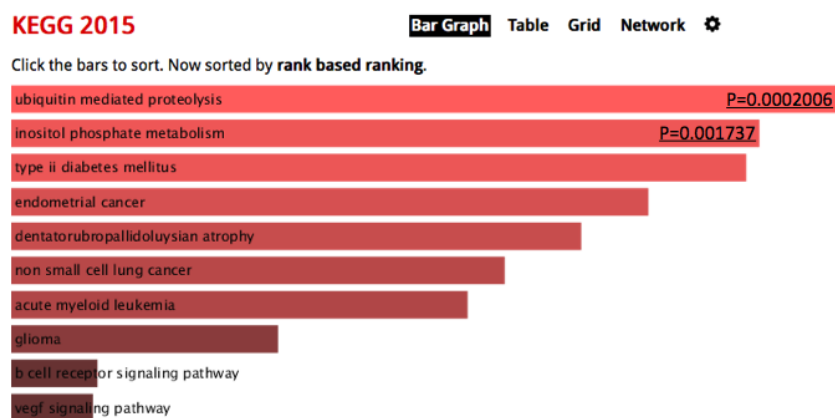


Figure 6.12: Enrichr: post-treatment enrichment with KEGG [32] pathway engine. Significant relation with Ubiquitin related pathways, p - value = 0.0002006

6.4 CAPRI Adaptation

Considering all the adaptations, filters and enrichments done for anticancer drug resistance analysis, the final workflow for this last part (modeling) is represented in Figure 6.13. It encloses the Mutex's test for exclude mutual exclusive genes, the use of CISs for driver genes selection, sequence count filter to select only the true positives events between different samples and the enrichment analysis with Enrichr and GeneMANIA for final speculations.

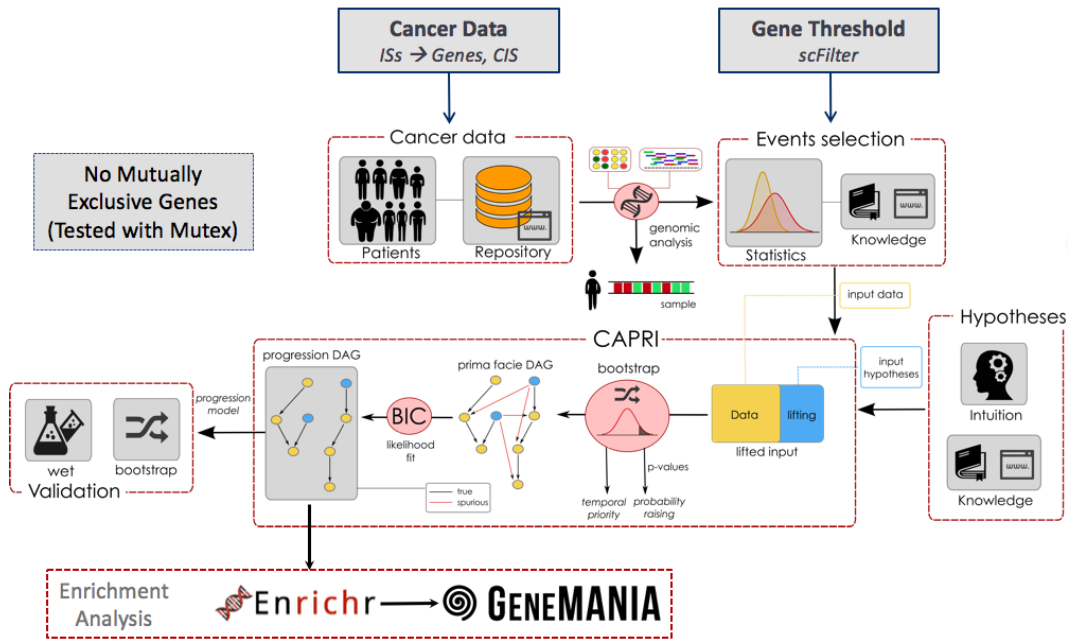


Figure 6.13: CAPRI Pipeline [11], edited from Figure 5.2. The framework to study anticancer drug resistance from IS data. CAPRI is integrated with Mutex for checking the mutual exclusivity between CISs, the cross sectional usage of CISs as input, the **scFilter** of ISAnalytics to set the gene threshold and the final results interpretation with enrichment analysis with Enrichr and GeneMANIA.

Chapter 7

Conclusions

This work provides the first example of a framework of insertional mutagenesis screen for drug resistance gene identification employing LVs as mutagens, with IS data (*CISs* in particular) and *causal modeling reconstruction*. Recently my laboratory and SR-Tiget has performed for the first time a LV-based insertional mutagenesis screen for cancer gene discovery [1, 2]. The developed framework in synergy with this biotechnology technique provides a powerful tool also for pre-clinical screening and not only for cancer therapies. The presented workflow includes the three powerful tools presented in Chapters 3 (**VISPA2**), 4 (**ISAnalytics**) and 5 (**CAPRI**). The first two developed completely by me and the last one is in collaboration (I adapted the model and generated tools for data conversion) with the University of Milano-Bicocca.

VISPA2, Chapter 3, is an essential step for assessing the safety and efficacy of molecular therapies that use genetically-modified hematopoietic stem cells through integrating viral vectors. VISPA2 was extensively used to monitor lentiviral integrations in several clinical trials (such as MLD, WAS, β -Thalassemia, MPS1) and internal/external projects. Moreover, compared with the previous version of VISPA, is more precise (reducing the false positives), supports repetitive element analysis and is very fast.

ISAnalytics, Chapter 4, is a new R package enabling IS analysis from the matrix file, output of VISPA2, compliant with Bioconductor 3.1. I developed many functions in ISAnalytics, such as collision detection and CIS analysis enabling the identification of over-targeted genomic regions. Furthermore, it uses very efficient data structures such as GRanges and SummarizedExperiment, the standard *de facto* for genomic operation and experiment managing in R. It is also possible to extend ISAnalytics with new functions and structures, within the constraints of Bioconductor.

CAPRI, Chapter 5, is a new algorithm to causal modeling reconstruction. It was able to create progression DAGs for the breast cancer case study of my laboratory. Because the cross-sectional data capabilities of the model, I created some tools to lift and to adapt the input (ISs, CISs) that worked very well. The post analysis with

enrichment tools (Enrichr and GeneMANIA) was fundamental to interpret the data (the graphs), to validate past results [3] and for new speculations on Ubiquitin.

Concluding, the biological use case has been confirmed, as published [3], using these precise and powerful new tools. Moreover, this study allowed:

1. Extend the previously developed causal model [11, 12, 102] with more precise ISs (with the usage of VISPA2), mutual exclusivity check and enrichment analysis through CISs and DAGs.
2. Find new biological results, some confirmed by scientific publications, other can be validated in the next future.

7.1 Future Works

VISPA2 has been largely extended and optimized. An innovative extension will be the inclusion of IS landing in repetitive elements, that can be improved, in terms of filtering and precision. At the moment a research article for VISPA2 is in preparation, at the beginning of the 2017 should be sent to a research journal.

I planned to extend ISAnalytics by adding/implementing new quantification methods (thinking on the new SLiM-PCR and shear site analysis [77, 89, 130]), ecology studies [131, 132] or new CIS identification methods [10, 26, 99, 100]. A separated methodological paper can be written (also for the Bioconductor compliance).

Regarding the framework should be useful to analyze also other datasets, in particular *in vivo*, to test the accuracy and the robustness of the tools. In my laboratory a new anticancer drug resistance experiment is ongoing and in the next months for sure I will analyze this dataset (8 MiSeq and 2 HiSeq Illumina runs, ~ 400 samples). A scientific paper for the whole framework, the anticancer drug resistance screening tools and methods proposed here, can be written in the near future.

An interesting open problem is the *big data* management. The bioinformatics tools, as I previously mentioned, are just ready to the big data. VISPA2 can process 1 Illumina HiSeq in less than 1 day and R is always improving. New infrastructural and information system management tools need to be applied to best handle the increasing number of input data. A possible solution is to migrate to a cloud system (for computing probably only Illumina HiSeq runs, 40-50 per year), like Amazon S3¹, and Spark², to run programs faster, reduce memory usage and accesses to the disks.

¹<https://aws.amazon.com/s3>

²<http://spark.apache.org>

Bibliography

- [1] M. Ranzani, S. Annunziato, D. J. Adams, and E. Montini. Cancer Gene Discovery: Exploiting Insertional Mutagenesis. *Molecular Cancer Research*, 11(10):1141–1158, 2013.
- [2] M. Ranzani, D. Cesana, C. C. Bartholomae, F. Sanvito, M. Pala, F. Benedicenti, P. Gallina, L. S. Sergi, S. Merella, A. Bulfone, C. Doglioni, C. von Kalle, Y. J. Kim, M. Schmidt, G. Tonon, L. Naldini, and E. Montini. Lentiviral vector-based insertional mutagenesis identifies genes associated with liver cancer. *Nat Methods*, 10(2):155–61, 2013.
- [3] M. Ranzani, S. Annunziato, A. Calabria, S. Brasca, F. Benedicenti, P. Gallina, L. Naldini, and E. Montini. Lentiviral Vector-based Insertional Mutagenesis Identifies Genes Involved in the Resistance to Targeted Anticancer Therapies. *Molecular Therapy*, 2014.
- [4] A. Ambrosi, C. Cattoglio, and C. Di Serio. Retroviral integration process in the human genome: is it really non-random? a new statistical approach. *PLoS Comput Biol*, 4(8):e1000144, 2008.
- [5] A. Ambrosi, I. K. Glad, D. Pellin, C. Cattoglio, F. Mavilio, C. Di Serio, and A. Frigessi. Estimated comparative integration hotspots identify different behaviors of retroviral gene transfer vectors. *PLoS Comput Biol*, 7(12):e1002292, 2011.
- [6] L. Biasco, A. Ambrosi, D. Pellin, C. Bartholomae, I. Brigida, M. G. Roncarolo, C. Di Serio, C. von Kalle, M. Schmidt, and A. Aiuti. Integration profile of retroviral vector in gene therapy treated patients is cell-specific according to gene expression and chromatin conformation of target cell. *EMBO Mol Med*, 3(2):89–101, 2011.
- [7] A. Calabria, S. Leo, F. Benedicenti, D. Cesana, G. Spinozzi, M. Orsini, S. Merella, E. Stupka, G. Zanetti, and E. Montini. VISPA: a computational pipeline for the identification and analysis of genomic vector integration sites. *Genome Medicine*, 6(9):1–12, 2014.

- [8] U. Abel, A. Deichmann, C. Bartholomae, K. Schwarzwaelder, H. Glimm, S. Howe, A. Thrasher, A. Garrigue, S. Hacein-Bey-Abina, M. Cavazzana-Calvo, A. Fischer, D. Jaeger, C. von Kalle, and M. Schmidt. Real-time definition of non-randomness in the distribution of genomic events. *PLoS ONE*, 2(6):e570, 2007.
- [9] U. Abel, A. Deichmann, A. Nowrouzi, R. Gabriel, C. C. Bartholomae, H. Glimm, C. von Kalle, and M. Schmidt. Analyzing the number of common integration sites of viral vectors – new methods and computer programs. *PLoS ONE*, 6(10):1–8, 10 2011.
- [10] A. P. Presson, N. Kim, Y. Xiaofei, I. S. Chen, and S. Kim. Methodology and software to detect viral integration site hot-spots. *BMC Bioinformatics*, 12(1):1–13, 2011.
- [11] D. Ramazzotti, G. Caravagna, L. O. Loohuis, A. Graudenzi, I. Korsunsky, G. Mauri, M. Antoniotti, and B. Mishra. CAPRI: efficient inference of cancer progression models from cross-sectional data. *Bioinformatics*, 31(18):3016–3026, 2015.
- [12] G. Caravagna, A. Graudenzi, D. Ramazzotti, R. Sanz-Pamplona, L. De Sano, G. Mauri, V. Moreno, M. Antoniotti, and B. Mishra. Algorithmic methods to infer the evolutionary trajectories in cancer progression. *Proceedings of the National Academy of Sciences*, 113(28):E4025–E4034, 2016.
- [13] D. Warde-Farley, S. L. Donaldson, O. Comes, K. Zuberi, R. Badrawi, P. Chao, M. Franz, C. Grouios, F. Kazi, C. T. Lopes, A. Maitland, S. Mostafavi, J. Montojo, Q. Shao, G. Wright, G. D. Bader, and Q. Morris. The genemania prediction server: biological network integration for gene prioritization and predicting gene function. *Nucleic Acids Research*, 38(suppl 2):W214–W220, 2010.
- [14] E. Y. Chen, C. M. Tan, Y. Kou, Q. Duan, Z. Wang, G. V. Meirelles, N. R. Clark, and A. Ma’ayan. Enrichr: interactive and collaborative html5 gene list enrichment analysis tool. *BMC Bioinformatics*, 14(1):1–14, 2013.
- [15] M. V. Kuleshov, M. R. Jones, A. D. Rouillard, N. F. Fernandez, Q. Duan, Z. Wang, S. Koplev, S. L. Jenkins, K. M. Jagodnik, A. Lachmann, M. G. McDermott, C. D. Monteiro, G. W. Gundersen, and A. Ma’ayan. Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. *Nucleic Acids Research*, 44(W1):W90–W97, 2016.
- [16] Ö. Babur, M. Gönen, B. A. Aksoy, N. Schultz, G. Ciriello, C. Sander, and E. Demir. Systematic identification of cancer driving signaling pathways based on mutual exclusivity of genomic alterations. *Genome Biology*, 16(1):1–10, 2015.

- [17] L. Ding, T. J. Ley, D. E. Larson, C. A. Miller, D. C. Koboldt, J. S. Welch, J. K. Ritchey, M. A. Young, T. Lamprecht, M. D. McLellan, J. F. McMichael, J. W. Wallis, C. Lu, D. Shen, C. C. Harris, D. J. Dooling, R. S. Fulton, L. L. Fulton, K. Chen, H. Schmidt, J. Kalicki-Veizer, V. J. Magrini, L. Cook, S. D. McGrath, T. L. Vickery, M. C. Wendl, S. Heath, M. A. Watson, D. C. Link, M. H. Tomasson, W. D. Shannon, J. E. Payton, S. Kulkarni, P. Westervelt, M. J. Walter, T. A. Graubert, E. R. Mardis, R. K. Wilson, and J. F. DiPersio. Clonal evolution in relapsed acute myeloid leukaemia revealed by whole-genome sequencing. *Nature*, 481(7382):506–10, 2012.
- [18] C. A. Miller, J. McMichael, H. X. Dang, C. A. Maher, L. Ding, T. J. Ley, E. R. Mardis, and R. K. Wilson. Visualizing tumor evolution with the fishplot package for r. *bioRxiv*, 2016.
- [19] A. Biffi, E. Montini, L. Lorioli, M. Cesani, F. Fumagalli, T. Plati, C. Baldoli, S. Martino, A. Calabria, S. Canale, F. Benedicenti, G. Vallanti, L. Biasco, S. Leo, N. Kabbara, G. Zanetti, W. B. Rizzo, N. A. L. Mehta, M. P. Cicalese, M. Casiraghi, J. J. Boelens, U. Del Carro, D. J. Dow, M. Schmidt, A. Assanelli, V. Neduva, C. Di Serio, E. Stupka, J. Gardner, C. von Kalle, C. Bordignon, F. Ciceri, A. Rovelli, M. G. Roncarolo, A. Aiuti, M. Sessa, and L. Naldini. Lentiviral hematopoietic stem cell gene therapy benefits metachromatic leukodystrophy. *Science*, 341(6148), 2013.
- [20] B. Vogelstein, E. R. Fearon, S. R. Hamilton, S. E. Kern, A. C. Preisinger, M. Leppert, Y. Nakamura, R. White, A. M. Smits, and J. L. Bos. Genetic alterations during colorectal-tumor development. *New England Journal of Medicine*, 319(9):525–32, Sep 1988.
- [21] K. Hainke, J. Rahnenführer, and R. Fried. Cumulative disease progression models for cross-sectional data: a review and comparison. *Biom J*, 54(5):617–40, Sep 2012.
- [22] R. Cordaux and M. A. Batzer. The impact of retrotransposons on human genome evolution. *Nat Rev Genet*, 10(10):691–703, Oct 2009.
- [23] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *J Mol Biol*, 215(3):403–10, 1990.
- [24] M. Cavazzana-Calvo, E. Payen, O. Negre, G. Wang, K. Hehir, F. Fusil, J. Down, M. Denaro, T. Brady, K. Westerman, R. Cavalleco, B. Gillet-Legrand, L. Caccavelli, R. Sgarra, L. Maouche-Chretien, F. Bernaudin, R. Girot, R. Dorazio, G.-J. Mulder, A. Polack, A. Bank, J. Soulier, J. Larghero, N. Kabbara, B. Dalle,

- B. Gourmel, G. Socie, S. Chretien, N. Cartier, P. Aubourg, A. Fischer, K. Cornetta, F. Galacteros, Y. Beuzard, E. Gluckman, F. Bushman, S. Hacein-Bey-Abina, and P. Leboulch. Transfusion independence and hmga2 activation after gene therapy of human b-thalassaemia. *Nature*, 467(7313):318–322, Sep 2010.
- [25] J. de Ridder, A. Uren, J. Kool, M. Reinders, and L. Wessels. Detecting statistically significant common insertion sites in retroviral insertional mutagenesis screens. *PLOS Computational Biology*, 2(12):1–13, 12 2006.
- [26] R. Fronza, A. Vasciaveo, A. Benso, and M. Schmidt. A graph based framework to model virus integration sites. *Computational and Structural Biotechnology Journal*, 14:69–77, 2016.
- [27] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker. Cytoscape: A software environment for integrated models of biomolecular interaction networks. *Genome Research*, 13(11):2498–2504, 2003.
- [28] C. P. Chuu, R. Y. Chen, J. L. Barking, M. F. Ciaccio, and R. B. Jones. Systems-level analysis of erbb4 signaling in breast cancer: A laboratory to clinical perspective. *Molecular Cancer Research*, 6(6):885–891, 2008.
- [29] M. Sundvall, K. Iljin, S. Kilpinen, H. Sara, O. P. Kallioniemi, and K. Elenius. Role of erbb4 in breast cancer. *Journal of Mammary Gland Biology and Neoplasia*, 13(2):259–268, 2008.
- [30] H. Masuda, D. W. Zhang, C. Bartholomeusz, H. Doihara, G. N. Hortobagyi, and N. T. Ueno. Role of epidermal growth factor receptor in breast cancer. *Breast Cancer Research and Treatment*, 136(2):331–345, 2012.
- [31] F. Milanezi, S. Carvalho, and F. C. Schmitt. Egfr/her2 in breast cancer: a biological approach for molecular diagnosis and therapy. *Expert Review of Molecular Diagnostics*, 8(4):417–434, 2008.
- [32] M. Kanehisa and S. Goto. Kegg: kyoto encyclopedia of genes and genomes. *Nucleic Acids Res*, 28(1):27–30, 2000.
- [33] A. Biffi, C. C. Bartolomae, D. Cesana, N. Cartier, P. Aubourg, M. Ranzani, M. Cesani, F. Benedicenti, T. Plati, E. Rubagotti, S. Merella, A. Capotondo, J. Sgualdino, G. Zanetti, C. von Kalle, M. Schmidt, L. Naldini, and E. Montini. Lentiviral vector common integration sites in preclinical models and a clinical trial reflect a benign integration bias and not oncogenic selection. *Blood*, 117(20):5332–5339, 2011.

- [34] M. R. Stratton, P. J. Campbell, and P. A. Futreal. The cancer genome. *Nature*, 458(7239):719–724, 2009.
- [35] M. Greaves and C. C. Maley. Clonal evolution in cancer. *Nature*, 481(7381):306–13, 2012.
- [36] M. Janiszewska and K. Polyak. Clonal evolution in cancer: A tale of twisted twins. *Stem Cell*, 16(1):11–12, 2015.
- [37] L. M. F. Merlo, J. W. Pepper, B. J. Reid, and C. C. Maley. Cancer as an evolutionary and ecological process. *Nature Reviews Cancer*, 6(12):924–935, 2006.
- [38] D. Hanahan and R. A. Weinberg. The hallmarks of cancer. *Cell*, 100(1):57–70, 2000.
- [39] D. Hanahan and R. A. Weinberg. Hallmarks of cancer: the next generation. *Cell*, 144(5):646–674, Mar. 2011.
- [40] M. Martini, L. Vecchione, S. Siena, S. Tejpar, and A. Bardelli. Targeted therapies: how personal should we go? *Nature Reviews Clinical Oncology*, 9(2):87–97, 2012.
- [41] L. A. Garraway and P. A. Jänne. Circumventing Cancer Drug Resistance in the Era of Personalized Medicine. *Cancer Discovery*, 2(3):214–226, Mar. 2012.
- [42] G. Giaccone and H. M. Pinedo. Drug resistance. *Oncologist*, 1(1-2):82–87, 1996.
- [43] D. Cross and J. K. Burmester. Gene therapy for cancer treatment: past, present and future. *Clin Med Res*, 4(3):218–27, 2006.
- [44] L. Naldini. Ex vivo gene transfer and correction for cell-based therapies. *Nat Rev Genet*, 12(5):301–15, 2011.
- [45] L. Naldini. Gene therapy returns to centre stage. *Nature*, 526(7573):351–60, 2015.
- [46] M. Sun. Martin cline loses appeal on nih grant. *Science*, 218(4567):37, 1982.
- [47] C. Sheridan. Gene therapy finds its niche. *Nat Biotechnol*, 29(2):121–8, 2011.
- [48] M. Sessa, L. Lorioli, F. Fumagalli, S. Acquati, D. Redaelli, C. Baldoli, S. Canale, I. D. Lopez, F. Morena, A. Calabria, R. Fiori, P. Silvani, P. M. Rancoita, M. Galbaldo, F. Benedicenti, G. Antonioli, A. Assanelli, M. P. Cicalese, U. Del Carro, M. G. Sora, S. Martino, A. Quattrini, E. Montini, C. Di Serio, F. Ciceri, M. G. Roncarolo, A. Aiuti, L. Naldini, and A. Biffi. Lentiviral haemopoietic stem-cell

- gene therapy in early-onset metachromatic leukodystrophy: an ad-hoc analysis of a non-randomised, open-label, phase 1/2 trial. *Lancet*, 388(10043):476–87, 2016.
- [49] A. Aiuti, L. Biasco, S. Scaramuzza, F. Ferrua, M. P. Cicalese, C. Baricordi, F. Dionisio, A. Calabria, S. Giannelli, M. C. Castiello, M. Bosticardo, C. Evangelio, A. Assanelli, M. Casiraghi, S. Di Nunzio, L. Callegaro, C. Benati, P. Rizzardi, D. Pellin, C. Di Serio, M. Schmidt, C. Von Kalle, J. Gardner, N. Mehta, V. Neduva, D. J. Dow, A. Galy, R. Miniero, A. Finocchi, A. Metin, P. P. Banerjee, J. S. Orange, S. Galimberti, M. G. Valsecchi, A. Biffi, E. Montini, A. Villa, F. Ciceri, M. G. Roncarolo, and L. Naldini. Lentiviral hematopoietic stem cell gene therapy in patients with wiskott-aldrich syndrome. *Science*, 341(6148), 2013.
- [50] I. Visigalli, S. Delai, F. Ferro, F. Cecere, M. Vezzoli, F. Sanvito, F. Chanut, F. Benedicenti, G. Spinozzi, R. Wynn, A. Calabria, L. Naldini, E. Montini, P. Cristofori, and A. Biffi. Preclinical testing of the safety and tolerability of lv-mediated above normal alpha-l-iduronidase expression in murine and human hematopoietic cells using toxicology and biodistribution glp studies. *Hum Gene Ther*, 2016.
- [51] E. A. Roselli, R. Mezzadra, M. C. Frittoli, G. Maruggi, E. Biral, F. Mavilio, F. Mastropietro, A. Amato, G. Tonon, C. Refaldi, M. D. Cappellini, M. Andreani, G. Lucarelli, M. G. Roncarolo, S. Markt, and G. Ferrari. Correction of beta-thalassemia major by gene transfer in haematopoietic progenitors of pediatric patients. *EMBO Mol Med*, 2(8):315–28, 2010.
- [52] A. Miccio, R. Cesari, F. Lotti, C. Rossi, F. Sanvito, M. Ponzoni, S. J. Routledge, C. M. Chow, M. N. Antoniou, and G. Ferrari. In vivo selection of genetically modified erythroblastic progenitors leads to long-term correction of beta-thalassemia. *Proc Natl Acad Sci U S A*, 105(30):10547–52, 2008.
- [53] J. Kaiser. American society of gene therapy meeting. retroviral vectors: a double-edged sword. *Science*, 308(5729):1735–6, 2005.
- [54] S. Hacein-Bey-Abina, J. Hauer, A. Lim, C. Picard, G. P. Wang, C. C. Berry, C. Martinache, F. Rieux-Laucat, S. Latour, B. H. Belohradsky, L. Leiva, R. Sorensen, M. Debre, J. L. Casanova, S. Blanche, A. Durandy, F. D. Bushman, A. Fischer, and M. Cavazzana-Calvo. Efficacy of gene therapy for x-linked severe combined immunodeficiency. *N Engl J Med*, 363(4):355–64, 2010.
- [55] R. Craigie and F. D. Bushman. Hiv dna integration. *Cold Spring Harbor Perspectives in Medicine*, 2(7):a006890–a006890, 2012.

- [56] M. Schmidt, K. Schwarzwaelder, C. Bartholomae, K. Zaoui, C. Ball, I. Pilz, S. Braun, H. Glimm, and C. von Kalle. High-resolution insertion-site analysis by linear amplification-mediated pcr (LAM-PCR). *Nat Meth*, 4(12):1051–1057, Dec 2007.
- [57] N. Beerenwinkel, M. Dumer, T. Sing, J. Rahnenfhrer, T. Lengauer, J. Selbig, D. Hoffmann, and R. Kaiser. Estimating hiv evolutionary pathways and the genetic barrier to drug resistance. *The Journal of infectious diseases*, 191(11):1953–1960, 2005.
- [58] M. W. Huston, M. H. Brugman, S. Horsman, A. Stubbs, P. van der Spek, and G. Wagemaker. Comprehensive investigation of parameter choice in viral integration site analysis and its effects on the gene annotations produced. *Human Gene Therapy*, 23(11):1209–1219, Aug 2012.
- [59] T. Hawkins, J. Dantzer, B. Peters, M. Dinauer, K. Mockaitis, S. Mooney, and K. Cornetta. Identifying viral integration sites using seqmap 2.0. *Bioinformatics*, 27(5):720–722, 3 2011.
- [60] J. Appelt, F. Giordano, M. Ecker, I. Roeder, N. Grund, A. Hotz-Wagenblatt, G. Opelz, W. Zeller, H. Allgayer, S. Fruehauf, and S. Laufs. Quickmap: A public tool for large-scale gene therapy vector insertion site mapping and analysis. *Gene Therapy*, 16(7):885–893, 2009.
- [61] H. Li and R. Durbin. Fast and accurate short read alignment with burrows-wheeler transform. *Bioinformatics*, 25(14):1754–1760, 2009.
- [62] S. Marco-Sola, M. Sammeth, R. Guigo, and P. Ribeca. The gem mapper: fast, accurate and versatile alignment by filtration. *Nat Meth*, 9(12):1185–1188, Dec 2012.
- [63] H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, R. Durbin, and . G. P. D. P. Subgroup. The sequence alignment/map format and samtools. *Bioinformatics*, 25(16):2078–2079, 2009.
- [64] S. A. Frank. *Dynamics of Cancer: Incidence, Inheritance, and Evolution*. Princeton University Press, 2007.
- [65] M. A. Nowak. *Evolutionary dynamics: exploring the equations of life*. Harvard University Press, 2006.
- [66] J. C. Fisher and J. H. Hollomon. A hypothesis for the origin of cancer foci. *Cancer*, 4(5):916–918, 1951.

- [67] P. Armitage and R. Doll. The age distribution of cancer and a multi-stage theory of carcinogenesis. *Br J Cancer*, 8(1):1–12, Mar 1954. 13172380[pmid].
- [68] E. R. Fearon and B. Vogelstein. A genetic model for colorectal tumorigenesis. *Cell*, 61(5):759 – 767, 1990.
- [69] K. S. Korolev, J. B. Xavier, and J. Gore. Turning ecology and evolution against cancer. *Nat Rev Cancer*, 14(5):371–380, May 2014. Perspectives.
- [70] P. Nowell. The clonal evolution of tumor cell populations. *Science*, 194(4260):23–28, 1976.
- [71] R. Desper, F. Jiang, O. P. Kallioniemi, H. Moch, C. H. Papadimitriou, and A. A. Schäffer. Inferring tree models for oncogenesis from comparative genome hybridization data. *J Comput Biol*, 6(1):37–51, 1999.
- [72] M. Gerstung, N. Eriksson, J. Lin, B. Vogelstein, and N. Beerenwinkel. The temporal order of genetic and pathway alterations in tumorigenesis. *PLoS One*, 6(11):e27136, 2011.
- [73] N. Beerenwinkel, N. Eriksson, and B. Sturmfels. Conjunctive Bayesian networks. *Bernoulli*, 13(4):893–909, 2007.
- [74] S. Andrews. Fastqc a quality control tool for high throughput sequence data.
- [75] A. M. Bolger, M. Lohse, and B. Usadel. Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics*, page btu170, 2014.
- [76] H. Lab. FASTX Toolkit.
- [77] S. Firouzi, Y. López, Y. Suzuki, K. Nakai, S. Sugano, T. Yamochi, and T. Watanabe. Development and validation of a new high-throughput method to investigate the clonality of htlv-1-infected cells based on provirus integration sites. *Genome Medicine*, 6(6):46, 2014.
- [78] E. Aronesty. ea-utils : Command-line tools for processing biological sequencing data.
- [79] A. Calabria, G. Spinozzi, F. Benedicenti, E. Tenderini, and E. Montini. adLIMS: a customized open source software that allows bridging clinical and basic molecular research studies. *BMC Bioinformatics*, 16(9):S5, 2015.
- [80] M. Dodt, J. T. Roehr, R. Ahmed, and C. Dieterich. Flexbar: Flexible barcode and adapter processing for next-generation sequencing platforms. *Biology*, 1(3):895, 2012.

- [81] H. Li. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. *ArXiv e-prints*, Mar. 2013.
- [82] B. Langmead and S. L. Salzberg. Fast gapped-read alignment with bowtie 2. *Nat Meth*, 9(4):357–359, Apr 2012. Brief Communication.
- [83] C. Ade, A. M. Roy-Engel, and P. L. Deininger. Alu elements: an intrinsic source of human genome instability. *Curr Opin Virol*, 3(6):639–45, 2013.
- [84] L. B. Cohn, I. T. Silva, T. Y. Oliveira, R. A. Rosales, E. H. Parrish, G. H. Learn, B. H. Hahn, J. L. Czartoski, M. J. McElrath, C. Lehmann, F. Klein, M. Caskey, B. D. Walker, J. D. Siliciano, R. F. Siliciano, M. Jankovic, and M. C. Nussenzweig. Hiv-1 integration landscape during latent and active infection. *Cell*, 160(3):420–32, 2015.
- [85] A. Smit, R. Hubley, and P. Green. RepeatMasker Open-3.0, 1996-2004. <http://www.repeatmasker.org>.
- [86] M. Ruffalo, M. Koyutürk, S. Ray, and T. LaFramboise. Accurate estimation of short read mapping quality for next-generation genome sequencing. *Bioinformatics*, 28(18):i349–i355, Sep 2012. 22962451[pmid].
- [87] D. W. Barnett, E. K. Garrison, A. R. Quinlan, M. P. Strömberg, and G. T. Marth. BamTools: a C++ API and toolkit for analyzing and managing BAM files. *Bioinformatics (Oxford, England)*, 27(12):1691–1692, 2011.
- [88] T. Smith and M. Waterman”. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147(1):195 – 197, 1981.
- [89] C. C. Berry, N. A. Gillet, A. Melamed, N. Gormley, C. R. M. Bangham, and F. D. Bushman. Estimating abundances of retroviral insertion sites from dna fragment length data. *Bioinformatics*, 28(6):755–762, 2012.
- [90] F. Bushman, M. Lewinski, A. Ciuffi, S. Barr, J. Leipzig, S. Hannenhalli, and C. Hoffmann. Genome-wide analysis of retroviral dna integration. *Nat Rev Micro*, 3(11):848–858, Nov 2005.
- [91] S. D. Barr, A. Ciuffi, J. Leipzig, P. Shinn, J. R. Ecker, and F. D. Bushman. Hiv integration site selection: Targeting in macrophages and the effects of different routes of viral entry. *Mol Ther*, 14(2):218–225, Aug 2006.
- [92] T. Lassmann, Y. Hayashizaki, and C. O. Daub. Samstat: monitoring biases in next generation sequencing data. *Bioinformatics*, 27(1):130–1, 2011.

- [93] A. Calabria, S. Beretta, I. Merelli, G. Spinozzi, S. Brasca, Y. Pirola, F. Benedicenti, E. Tenderini, L. Milanese, and E. Montini. Lentiviral Vector-based Insertional Mutagenesis Identifies Genes Involved in the Resistance to Targeted Anticancer Therapies. *IN PREPARATION*, 2016.
- [94] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2016.
- [95] Huber, W., Carey, V. J., Gentleman, R., Anders, S., Carlson, M., Carvalho, B. S., Bravo, H. C., Davis, S., Gatto, L., Girke, T., Gottardo, R., Hahne, F., Hansen, K. D., Irizarry, R. A., Lawrence, M., Love, M. I., MacDonald, J., Obenchain, V., Ole's, A. K., Pag'es, H., Reyes, A., Shannon, P., Smyth, G. K., Tenenbaum, D., Waldron, L., Morgan, and M. Orchestrating high-throughput genomic analysis with Bioconductor. *Nature Methods*, 12(2):115–121, 2015.
- [96] M. Lawrence, W. Huber, H. Pagès, P. Aboyoun, M. Carlson, R. Gentleman, M. Morgan, and V. Carey. Software for computing and annotating genomic ranges. *PLoS Computational Biology*, 9, 2013.
- [97] M. Morgan, V. Obenchain, J. Hester, and H. Pags. SummarizedExperiment: SummarizedExperiment container, 2016. R package version 1.3.1.
- [98] W. Huber, V. J. Carey, R. Gentleman, S. Anders, M. Carlson, B. S. Carvalho, H. C. Bravo, S. Davis, L. Gatto, T. Girke, R. Gottardo, F. Hahne, K. D. Hansen, R. A. Irizarry, M. Lawrence, M. I. Love, J. MacDonald, V. Obenchain, A. K. Oles, H. Pages, A. Reyes, P. Shannon, G. K. Smyth, D. Tenenbaum, L. Waldron, and M. Morgan. Orchestrating high-throughput genomic analysis with bioconductor. *Nat Meth*, 12(2):115–121, Feb 2015.
- [99] C. C. Berry, K. E. Ocwieja, N. Malani, and F. D. Bushman. Comparing dna integration site clusters with scan statistics. *Bioinformatics*, 30(11):1493–1500, Jun 2014.
- [100] D. Pellin and C. Di Serio. A novel scan statistics approach for clustering identification and comparison in binary genomic data. *BMC Bioinformatics*, 17(11):320, 2016.
- [101] N. Beerenwinkel, R. F. Schwarz, M. Gerstung, and F. Markowetz. Cancer evolution: mathematical models and computational inference. *Systematic biology*, page syu081, 2014.
- [102] L. O. Loohuis, G. Caravagna, A. Graudenzi, D. Ramazzotti, G. Mauri, M. Antoniotti, and B. Mishra. Inferring tree causal models of cancer progression with probability raising. *PLOS ONE*, 9(10):1–14, 10 2014.

- [103] P. Suppes. *A Probabilistic Theory of Causality*. Amsterdam, North-Holland Pub. Co., 1970.
- [104] B. N. et al. Learning multiple evolutionary pathways from cross-sectional data. *J. Computational Biology*, 12(6):584–98, 2003.
- [105] L. De Sano, G. Caravagna, D. Ramazzotti, A. Graudenzi, G. Mauri, B. Mishra, and M. Antoniotti. TRONCO: an R package for the inference of cancer progression models from heterogeneous genomic data. *Bioinformatics*, 2016.
- [106] R. Piazza, S. Valletta, N. Winkelmann, S. Redaelli, R. Spinelli, A. Pirola, L. Antolini, L. Mologni, C. Donadoni, E. Papaemmanuil, S. Schnittger, D.-W. Kim, J. Boultonwood, F. Rossi, G. Gaipa, G. P. De Martini, P. F. di Celle, H. G. Jang, V. Fantin, G. R. Bignell, V. Magistroni, T. Haferlach, E. M. Pogliani, P. J. Campbell, A. J. Chase, W. J. Tapper, N. C. P. Cross, and C. Gambacorti-Passerini. Recurrent setbp1 mutations in atypical chronic myeloid leukemia. *Nat Genet*, 45(1):18–24, Jan 2013.
- [107] M. Meggendorfer, U. Bacher, T. Alpermann, C. Haferlach, W. Kern, C. Gambacorti-Passerini, T. Haferlach, and S. Schnittger. Setbp1 mutations occur in 9% of mds/mpn and in 4% of mpn cases and are strongly associated with atypical cml, monosomy 7, isochromosome i(17)(q10), asxl1 and cbl mutations. *Leukemia*, 27(9):1852–1860, Sep 2013.
- [108] C.-C. Lin, H.-A. Hou, W.-C. Chou, Y.-Y. Kuo, S.-J. Wu, C.-Y. Liu, C.-Y. Chen, M.-H. Tseng, C.-F. Huang, F.-Y. Lee, M.-C. Liu, C.-W. Liu, J.-L. Tang, M. Yao, S.-Y. Huang, S.-C. Hsu, B.-S. Ko, W. Tsay, Y.-C. Chen, and H.-F. Tien. Sf3b1 mutations in patients with myelodysplastic syndromes: The mutation is stable during disease evolution. *American Journal of Hematology*, 89(8):E109–E115, 2014.
- [109] J. Kool and A. Berns. High-throughput insertional mutagenesis screens in mice to identify oncogenic networks. *Nat Rev Cancer*, 9(6):389–399, Jun 2009.
- [110] D. J. Slamon, G. M. Clark, S. G. Wong, W. J. Levin, A. Ullrich, and W. L. McGuire. Human breast cancer: correlation of relapse and survival with amplification of the her-2/neu oncogene. *Science*, 235(4785):177–82, 1987.
- [111] L. D. Wood, D. W. Parsons, S. Jones, J. Lin, T. Sjoblom, R. J. Leary, D. Shen, S. M. Boca, T. Barber, J. Ptak, N. Silliman, S. Szabo, Z. Dezso, V. Ustyanksky, T. Nikolskaya, Y. Nikolsky, R. Karchin, P. A. Wilson, J. S. Kaminker, Z. Zhang, R. Croshaw, J. Willis, D. Dawson, M. Shipitsin, J. K. Willson, S. Sukumar, K. Polyak, B. H. Park, C. L. Pethiyagoda, P. V. Pant, D. G. Ballinger, A. B.

- Sparks, J. Hartigan, D. R. Smith, E. Suh, N. Papadopoulos, P. Buckhaults, S. D. Markowitz, G. Parmigiani, K. W. Kinzler, V. E. Velculescu, and B. Vogelstein. The genomic landscapes of human breast and colorectal cancers. *Science*, 318(5853):1108–13, 2007.
- [112] T. Sjoblom, S. Jones, L. D. Wood, D. W. Parsons, J. Lin, T. D. Barber, D. Mandelker, R. J. Leary, J. Ptak, N. Silliman, S. Szabo, P. Buckhaults, C. Farrell, P. Meeh, S. D. Markowitz, J. Willis, D. Dawson, J. K. Willson, A. F. Gazdar, J. Hartigan, L. Wu, C. Liu, G. Parmigiani, B. H. Park, K. E. Bachman, N. Papadopoulos, B. Vogelstein, K. W. Kinzler, and V. E. Velculescu. The consensus coding sequences of human breast and colorectal cancers. *Science*, 314(5797):268–74, 2006.
- [113] P. Eroles, A. Bosch, J. A. Perez-Fidalgo, and A. Lluch. Molecular biology in breast cancer: Intrinsic subtypes and signaling pathways. *Cancer Treatment Reviews*, 38(6):698–707, 2012.
- [114] S. X. Lin, J. Chen, M. Mazumdar, D. Poirier, C. Wang, A. Azzi, and M. Zhou. Molecular therapy of breast cancer: progress and future directions. *Nature Reviews Endocrinology*, 6(9):485–493, 2010.
- [115] H. Mann and D. Whitney. On a test of whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statistics*, 18:50–60, 1947.
- [116] R. Nussinov, H. Jang, and C. J. Tsai. The structural basis for cancer treatment decisions. *Oncotarget*, 5(17):7285–302, 2014.
- [117] M. van Dijk, A. Halpin-McCormick, T. Sessler, A. Samali, and E. Szegezdi. Resistance to trail in non-transformed cells is due to multiple redundant pathways. *Cell Death and Disease*, 4, 2013.
- [118] C. Y. McLean, D. Bristor, M. Hiller, S. L. Clarke, B. T. Schaar, C. B. Lowe, A. M. Wenger, and G. Bejerano. Great improves functional interpretation of cis-regulatory regions. *Nat Biotechnol*, 28(5):495–501, 2010.
- [119] A. R. Pico, T. Kelder, M. P. van Iersel, K. Hanspers, B. R. Conklin, and C. Evelo. Wikipathways: pathway editing for the people. *PLoS Biol*, 6(7):e184, 2008.
- [120] D. Croft, A. F. Mundo, R. Haw, M. Milacic, J. Weiser, G. M. Wu, M. Caudy, P. Garapati, M. Gillespie, M. R. Kamdar, B. Jassal, S. Jupe, L. Matthews, B. May, S. Palatnik, K. Rothfels, V. Shamovsky, H. Song, M. Williams, E. Birney, H. Hermjakob, L. Stein, and P. D’Eustachio. The reactome pathway knowledgebase. *Nucleic Acids Research*, 42(D1):D472–D477, 2014.

- [121] J. Amberger, C. Bocchini, and A. Hamosh. A new face and new challenges for online mendelian inheritance in man (omim (r)). *Human Mutation*, 32(5):564–567, 2011.
- [122] A. Subramanian, P. Tamayo, V. K. Mootha, S. Mukherjee, B. L. Ebert, M. A. Gillette, A. Paulovich, S. L. Pomeroy, T. R. Golub, E. S. Lander, and J. P. Mesirov. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43):15545–15550, 2005.
- [123] U. D. Vempati, C. Chung, C. Mader, A. Koleti, N. Datar, D. Vidovic, D. Wrobel, S. Erickson, J. L. Muhlich, G. Berriz, C. H. Benes, A. Subramanian, A. Pillai, C. E. Shamu, and S. C. Schurer. Metadata standard and data exchange specifications to describe, model, and integrate complex and diverse high-throughput screening data from the library of integrated network-based cellular signatures (lincs). *Journal of Biomolecular Screening*, 19(5):803–816, 2014.
- [124] B. Ganter, R. D. Snyder, D. N. Halbert, and M. D. Lee. Toxicogenomics in drug discovery and development: mechanistic analysis of compound/class-dependent effects usign the drugmatrix (r) database. *Pharmacogenomics*, 7(7):1025–1044, 2006.
- [125] J. Barretina, G. Caponigro, N. Stransky, K. Venkatesan, A. A. Margolin, S. Kim, C. J. Wilson, J. Lehar, G. V. Kryukov, D. Sonkin, A. Reddy, M. W. Liu, L. Murray, M. F. Berger, J. E. Monahan, P. Morais, J. Meltzer, A. Korejwa, J. Janevalbuena, F. A. Mapa, J. Thibault, E. Bric-Furlong, P. Raman, A. Shipway, I. H. Engels, J. Cheng, G. Y. K. Yu, J. J. Yu, P. Aspesi, M. de Silva, K. Jagtap, M. D. Jones, L. Wang, C. Hatton, E. Palesscandolo, S. Gupta, S. Mahan, C. Sougnez, R. C. Onofrio, T. Liefeld, L. MacConaill, W. Winckler, M. Reich, N. X. Li, J. P. Mesirov, S. B. Gabriel, G. Getz, K. Ardlie, V. Chan, V. E. Myer, B. L. Weber, J. Porter, M. Warmuth, P. Finan, J. L. Harris, M. Meyerson, T. R. Golub, M. P. Morrissey, W. R. Sellers, R. Schlegel, and L. A. Garraway. The cancer cell line encyclopedia enables predictive modelling of anticancer drug sensitivity. *Nature*, 483(7391):603–607, 2012.
- [126] A. I. Su, T. Wiltshire, S. Batalov, H. Lapp, K. A. Ching, D. Block, J. Zhang, R. Soden, M. Hayakawa, G. Kreiman, M. P. Cooke, J. R. Walker, and J. B. Hogenesch. A gene atlas of the mouse and human protein-encoding transcriptomes. *Proceedings of the National Academy of Sciences of the United States of America*, 101(16):6062–6067, 2004.
- [127] I. J. Majewski, P. Nuciforo, L. Mittempergher, A. J. Bosma, H. Eidtmann, E. Holmes, C. Sotiriou, D. Fumagalli, J. Jimenez, C. Aura, L. Prudkin, M. C.

- Diaz-Delgado, L. de la Pena, S. Loi, C. Ellis, N. Schultz, E. de Azambuja, N. Harbeck, M. Piccart-Gebhart, R. Bernards, and J. Baselga. Pik3ca mutations are associated with decreased benefit to neoadjuvant human epidermal growth factor receptor 2-targeted therapies in breast cancer. *Journal of Clinical Oncology*, 33(12):1334, 2015.
- [128] S. Thaler, G. Thiede, J. G. Hengstler, A. Schad, M. Schmidt, and J. P. Sleeman. The proteasome inhibitor bortezomib (velcade) as potential inhibitor of estrogen receptor-positive breast cancer. *International Journal of Cancer*, 137(3):686–697, 2015.
- [129] T. Ohta and M. Fukuda. Ubiquitin and breast cancer. *Oncogene*, 23(11):2079–2088, 2004.
- [130] N. A. Gillet, N. Malani, A. Melamed, N. Gormley, R. Carter, D. Bentley, C. Berry, F. D. Bushman, G. P. Taylor, and C. R. Bangham. The host genomic environment of the provirus determines the abundance of htlv-1-infected t-cell clones. *Blood*, 117(11):3113–22, 2011.
- [131] A. Chao. Nonparametric estimation of the number of classes in a population. *Scandinavian Journal of Statistics*, 11(4):1–6, 1984.
- [132] A. Chao, R. K. Colwell, C.-W. Lin, and N. J. Gotelli. Sufficient sampling for asymptotic minimum species richness estimators. *Ecology*, 90(4):1125–1133, 2009.

Appendices

Appendix A

Wet-Lab Procedures

A.1 LAM-PCR

DNA fragments containing the vector genome junctions are retrieved and amplified from vector marked genomic DNA by *Linear Amplification Mediated Polymerase Chain Reaction* (LAM-PCR).

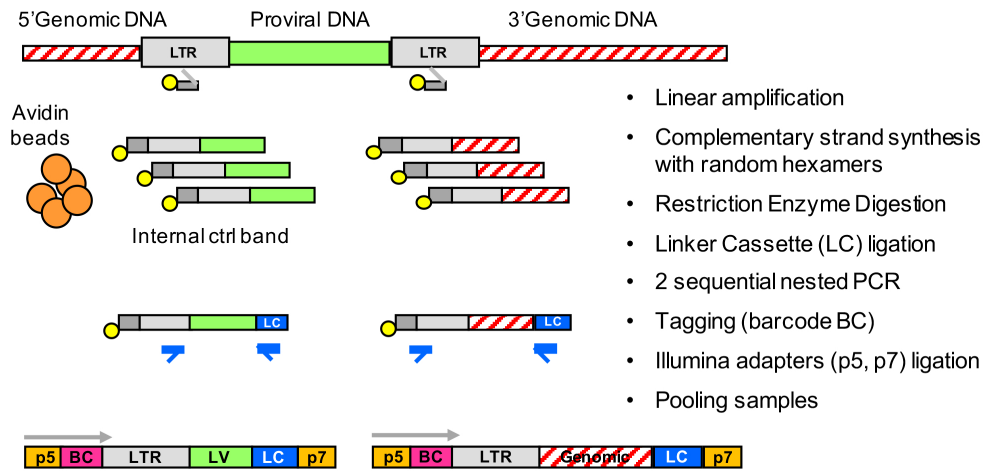


Figure A.1: LAM-PCR, sequential steps.

The LAM-PCR was performed on ~300ng of DNA extracted from cells. Following the Figure A.1 our technicians made 100 cycles of linear PCR pre-amplification of vector-genome junctions, followed by magnetic capture of the biotinylated target DNA by streptavidin-coupled magnetic beads, hexanucleotide priming, restriction digestion using MluCI, HpyCHIV4 and AciI enzymes, and ligation to a restriction site complementary linker cassette (LC, AGTGGCACAGCAGTTAGG). The ligation product was then amplified by two nested PCR with primers specific for the vector Long Terminal Repeat (LTR, ACCCTTTTAGTCAGTGTGGAAAATCTCTAGCA) and the LC

sequences. LAM-PCR amplicons were separated on Shimadzu MultiNA Microchip Electrophoresis System to evaluate PCR efficiency and the bands pattern for each sample. Primers and PCR thermal protocols used were previously described in [56]. LAM-PCR products were then purified by AmpureXP beads and quantified by Qubit Fluorimeter. 40ng of PCR product were re-amplified with Fusion-LTR and Fusion-LC primers (Section A.1.1, containing 8 nucleotides (X) tags allowing samples barcoding on both on the LTR and the Linker cassette side of the amplicons, specific sequences that allow paired end sequencing with the Illumina MiSeq System, and a random 12 nucleotides (N) sequence to increase cluster separation. The PCR was performed using the Qiagen TAQ DNA Polymerase at the following conditions: 95 °C for 2 min, and 95 °C for 45 sec, 58 °C for 45 sec, and 72 °C for 1 min, for 12 cycles, followed by a further 5 min incubation at 72 °C.

A.1.1 Fusion Primers for LAM-PCR

>Fusion-LTR

```
AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGCTCTTCCGATCT
NNNNNNNNNNNNNNNNNNNNXXXXXXXXXACCCTTTTAGTCAGTGTGGA
```

>Fusion-LC

```
CAAGCAGAAGACGGCATAACGAGATGTGACTGGAGTTCAGACGTGTGCTCTTCCGATCT
NNNNNNNNNNNNNNNNNNNNXXXXXXXXXGATCTGAATTCAGTGGCACAG
```


A.2 Sonicated Linker-Mediated (SLiM)-PCR

The old LAM-PCR, although sensitive, allows only approximate clonal abundance estimations since before PCR amplification the genomic DNA is fragmented with restriction enzymes that, depending on the distance between the integrated vector and the site recognized by restriction enzyme will produce DNA fragments of different sizes upon amplification. Therefore, IS in long DNA fragments could be lost or amplified at low efficiency while short fragments would be favored and still produce sequences too short to be univocally mapped on the target cell genome. Moreover, the nucleotide composition at the vector/cell genome junction may impact on the PCR amplification efficiency impacting on the reliability of clonal quantifications based on sequence count statistics.

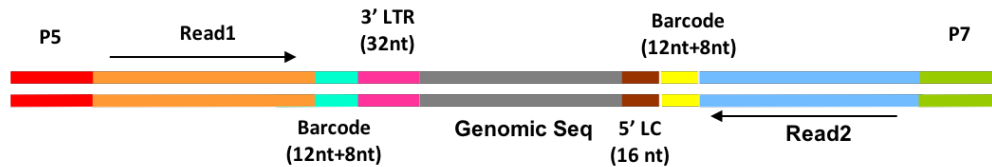


Figure A.2: SLiM-PCR, fragments (P5 and P7 are the Illumina adapters)

To avoid the biases produced by the exponential amplification and the use of restriction enzymes, my laboratory developed the new Sonicated Linker-Mediated (SLiM)-PCR. In this method, similar to [77], the genomic DNA of vector marked cells is sonicated to obtain randomly sheared fragments, ligated to a synthetic DNA linker cassette (LC, GTCACCGTGTCTCAATCCT) needed as template for the successive PCR amplification. The tagged DNA is then used as template for PCR using oligonucleotides complementary to vector sequences and the linker cassette in order to specifically amplify the vector/cell genome junctions contained in between. Given that the random DNA fragmentation, achieved by sonication, occurs prior PCR amplification, a clonal population harboring the same IS will produce a number of DNA fragments containing the vector/cell genome junctions of different sizes that will be proportional to the initial number of contributing cells. Therefore counting the number of shear sites of the same IS allows to estimate the clonal abundance avoiding the PCR biases. Moreover, in the linker cassette ligated after DNA shearing we included a 12 nucleotide sequence, a random barcode that is tagged to the sheared DNA fragments prior PCR amplification.

A.2.1 Fusion Primers for SLiM-PCR

>Fusion-LTR

Appendix B

Information System

B.1 Computational Resources

I configured VISPA2 in various infrastructures, also different, in which the command line version is present everywhere, only in the Institute of Biomedical Technologies of the CNR at Segrate (MI) is implemented the GUI version. This section explained the technologies used and the resources used. A clarification is in order: all this has been done, not only to differentiate certain aspects, but also to make available VISPA2 with a larger scale.

B.1.1 @SR-Tiget: Gemini and Oracle

Gemini				
Nodes	Cores	RAM	Operating System	Research Group
2	16	128GB*	Ubuntu Server 12.04 LTS	SR-Tiget**

Table B.1: **Gemini**, HP Z820 Workstation. Intel(R) Xeon(R) CPU E5-2690, 2.90GHz. The storage is composed of 4 HD with 500GB and 10k rpm. *Memory per node. **SR-Tiget people in charge of computational resources are Giulio Spinozzi and Andrea Calabria.

Oracle				
Nodes	Cores	RAM	Operating System	Research Group
2	24	128GB*	Ubuntu Server 16.04 LTS	SR-Tiget**

Table B.2: **Oracle**, HP Z840 Workstation. Intel(R) Xeon(R) CPU E5-2690 v3, 2.60GHz. The storage is composed of 5 HD with 4TB and 7k rpm and 1 primary disk of 500GB SSD. *Memory per node. **SR-Tiget people in charge of computational resources are Giulio Spinozzi and Andrea Calabria.

B.1.2 @CINECA: PICO

PICO				
Nodes	Cores	RAM	Operating System	Description
74	1480	126GB*	Red Hat Enterprise	VM with Ubuntu Server 12.04

Table B.3: PICO is a BigData infrastructure that has been acquired (Nov 2014) devoted to "Big Analytics". It is named after the Italian Renaissance philosopher famous for his amazing memory. *Memory per node. Intel(R) Xeon(R) CPU 2670 v2, 2.5GHz.

B.1.3 @CNR

Cluster @ CNR-ITB Milano				
Nodes	Cores	RAM	Operating System	Description
40+	700+	24GB*	CentOS	VM with CentOS 6.5

Table B.4: CNR-ITB bioinformatics computational resources present at ITB-Milano consist of more than 700 CPU-cores (Intel(R) Xeon(R) L5640 Westmere, Intel(R) Xeon(R) E5420 Harpertown based on Penryn microarchitecture and Intel(R) Xeon(R) E5420 Harpertown), more than 270 TB of disk space and more than 1700GBs of total memory, in an dual and quadri infiniband interconnected computational clusters. The architecture provides both an advanced HPC computational infrastructure and a distributed cloud-like virtualization facility for aggregating virtual servers, in turn providing the CNR-ITB bioinformatics services exposed to the Internet. *Memory per node. Intel(R) Xeon(R) CPU 2670 v2, 2.5GHz.

B.2 Storage

B.2.1 NAS Server: QNAP TS-412 4-BAY

QNAP TS-412 4-BAY				
CPU	DRAM	HDs	Operating System	File Sharing
Marvell 6281 1.2GHz	256MB DDRII	4x 3.5" SATAII 4x 2.5" SATAII	Embedded Linux	CIFS/SMB, AFP, NFS

Table B.5: TS-412 is a powerful yet easy to use networked storage center for backup, synchronization and remote access. It supports comprehensive RAID configuration and hot-swapping to allow hard drive replacement without system interruption.

This NAS is available online with CIFS/SMB or AFP protocols for UNIX systems.

B.2.2 LaCie 4Big Quadra

LaCie 4big Quadra USB 3		
HDs	RAID Types	Interfaces
4x 7.200 rpm	0/10/5/5+replacement	Thunderbolt, USB2-3, FireWire

Table B.6: LaCie 4Big Quadra: Noctua magnetic levitation cooling fan: high-performance, quiet, zero-vibration. 32MB cache (or greater) hard disks.

The LaCie 4Big Quadra is attached, through thunderbolt, to Gemini workstation, because the lack of HD space. Is also possible to put it into LAN network.

All final files (BAM, BED..) and MySQL databases are backed up into LaCie 4Big Quadra with RAID5, weekly.

B.3 Configurations

VISPA2 SERVER CONFIGURATION (Rev 1.0 13/04/2016)

 Author: Giulio Spinozzi, PhD Student
 San Raffaele Telethon Institute for Gene Therapy (SR-Tiget)
 Ospedale San Raffaele, Basilica, 5A3, Room 55
 E-Mail: spinozzi.giulio@hsr.it

Required software (pre-installed):

MySQL server (<https://dev.mysql.com/downloads/mysql/>)
 Fastqc (<http://www.bioinformatics.babraham.ac.uk/projects/fastqc/>)
 Bwa (<http://bio-bwa.sourceforge.net/bwa.shtml>)
 Samtools (<http://www.htslib.org/doc/samtools.html>)
 Trimmomatic (www.usadellab.org/cms/?page=trimmomatic)
 fastq-multx (<https://expressionanalysis.github.io/ea-utils/>)
 flexbar (<https://github.com/seqan/flexbar>)
 bamtools (<https://github.com/pezmaster31/bamtools>)
 RepeatMasker (<http://www.repeatmasker.org/RMDownload.html>)
 FilterSamReads/MergeSamFiles (<https://broadinstitute.github.io/picard/>)
 Bedtools (<https://github.com/arq5x/bedtools2/releases>)

Configuration:

Directories
 sudo mkdir /opt/applications
 sudo chmod -R 777 /opt/applications
 mkdir /opt/applications/bin
 sudo mkdir /opt/genome
 sudo chmod -R 777 /opt/genome
 sudo mkdir /opt/NGS
 sudo chmod -R 777 /opt/NGS
 mkdir /opt/NGS/results

In /opt/applications/bin you should install all third-party software like Picard, EA-Utils...
 while in /opt/NGS/results the final results folder (BAM, BED...)

Genomes
 mkdir /opt/genome/human
 mkdir /opt/genome/human/hg19/
 mkdir /opt/genome/human/hg19/annotation
 mkdir /opt/genome/human/hg19/index
 mkdir /opt/genome/human/hg19/index/bwa_7
 mkdir /opt/genome/vector

In /opt/genome/human/hg19/index/bwa_7 must be inserted the genome (FASTA) and its indexes. The indexes can be built in this way:
 bwa index -a bwtsv REF.fa

```
samtools faidx REF.fa
java -jar /opt/applications/bin/picard/picard-tools-1.79/CreateSequenceDictionary.jar R=CE.cns.fa O=CE.cns.dict
The same for vector genomes.
```

```
R
Install R (3.2 or 3.3) system with Bioconductor 3.1 (exactly this version)
```

```
MySQL
Install the dev lib and create two users (all privileges and readonly)
sudo apt-get install libmysqlclient-dev
mysql -uroot -p
mysql> GRANT ALL PRIVILEGES ON *.* TO 'andrea'@'localhost' IDENTIFIED BY 'andrea';
mysql> GRANT SELECT ON *.* TO 'readonly'@'localhost' IDENTIFIED BY 'readonlypswd';
```

```
BWA
sudo ln -s /opt/applications/bin/bwa/bwa-0.7.15/bwa /usr/bin/bwa-stable
```

```
Flexbar - Version 2.5
export LD_LIBRARY_PATH=/path/FlexbarDir:$LD_LIBRARY_PATH
sudo ln -s /opt/applications/bin/flexbar/flexbar_v2.5/flexbar /usr/bin/flexbar2.5
```

```
Trimmomatic
sudo ln -s /opt/applications/scripts/isatk/utils/trimmomatic.sh /usr/bin/trimmomatic
the trimmomatic.sh bash contains:
--
#!/bin/bash
java -jar /opt/applications/bin/trimmomatic/trimmomatic-0.36.jar $@
```

```
Picard
sudo ln -s /opt/applications/scripts/isatk/utils/FilterSamReads.sh /usr/bin/FilterSamReads
--
#!/bin/bash
picard FilterSamReads $@
sudo ln -s /opt/applications/scripts/isatk/utils/MergeSamReads.sh /usr/bin/MergeSamReads
--
#!/bin/bash
picard FilterSamReads $@
```

```
Python 2.7
sudo -H pip install MySQL-python
sudo -H pip install pysam==0.7.7
sudo -H pip install biopython
sudo -H pip install HTSeq
sudo -H pip install rpy2
sudo -H pip install scipy
sudo -H pip install numpy
sudo -H pip install matplotlib
sudo -H pip install xlswriter
sudo -H pip install pandas
```

```
ISATK Configuration (VISPA2 repository and others)
cd /opt/applications
hg clone -b 'v3' https://bitbucket.org/andreacalabria/isatk
sudo ln -s isatk/script/import_iss.py /usr/bin/import_iss
sudo ln -s isatk/script/fqreverseextract.pureheader.py /usr/bin/fqreverseextract.pureheader
sudo ln -s isatk/script/fqextract.pureheader.py /usr/bin/fqextract.pureheader
sudo ln -s isatk/script/rev_extract_header.py /usr/bin/rev_extract_header
sudo ln -s isatk/script/extract_header.py /usr/bin/extract_header
sudo ln -s isatk/script/filter_by_cigar_bam.py /usr/bin/filter_by_cigar_bam
sudo ln -s isatk/script/filter_by_mate.py /usr/bin/filter_by_mate
sudo ln -s isatk/script/dbimport_redundantiss_from_bed.v2.py /usr/bin/isa_importrediss_frombed
sudo ln -s /opt/applications/scripts/isatk/script/annotate_matrix_v2.sh /usr/bin/annotate_matrix
sudo ln -s /opt/applications/scripts/isatk/script/fastq_qf.sh /usr/bin/fastq_qf
sudo ln -s /opt/applications/scripts/isatk/script/fasta_to_csv.rb /usr/bin/fasta_to_csv
hg clone -b '2.1-seqTracker' https://bitbucket.org/tigetbioinformatics/integration_analysis
sudo ln -s integration_analysis/src/Integration_Analysis.py /usr/bin/create_matrix
```

```
RepeatMasker
http://www.repeatmasker.org/RMBlast.html
./configure --with-mt --prefix=/opt/applications/bin/rmbblast/ncbi-rmbblastn-2.2.28 --without-debug
http://www.repeatmasker.org/RMDownload.html
```

B.4 MySQL Tables

B.4.1 Import ISs from BED

Column	Type	Null	Default	Links to	Comments	MIME
group_name	varchar(200)	Yes	NULL			
n_LAM	varchar(200)	Yes	NULL			
pool	varchar(200)	Yes	NULL			
tag	varchar(20)	Yes	NULL			
sample	varchar(200)	Yes	NULL			
vector	varchar(50)	Yes	NULL			
tissue	varchar(50)	Yes	NULL			
treatment	varchar(20)	Yes	NULL			
enzyme	varchar(50)	Yes	NULL			
complete_name	varchar(200)	Yes	NULL			
header (<i>Primary</i>)	varchar(200)	No				
chr	varchar(100)	Yes	NULL			
integration_locus	int(11)	Yes	NULL			
sequence_count	int(3)	No	1			
score	double	Yes	NULL			
strand	varchar(2)	Yes	NULL			
label	varchar(200)	Yes	NULL			
sequence_raw	varchar(500)	Yes	NULL			
sequence_trimmed	varchar(1000)	Yes	NULL			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	header	30578	A	No	
integration_locus	BTREE	No	No	integration_locus	1911	A	Yes	
header	BTREE	No	No	header	30578	A	No	

Figure B.1: MySQL Table Structure for IS imported from BED file.

isa_importrediss_frombed (isatk/script/dbimport_redundantiss_from_bed.v2.py)

1. group_name: project name
2. n_LAM: LAM ID
3. pool: Pool ID
4. tag: TAG ID (for sample recognition)
5. sample: Sample ID
6. vector: Vector ID

7. `tissue`: Biological tissue of the harvest
8. `treatment`: time point
9. `enzyme`: Enzyme used (for SLiM-PCR is NONE)
10. `complete_name`: Complete name of the sample (link to the association file)
11. `header` (Primary): Header of the read (from FASTQ)
12. `chr`: Chromosome
13. `integration_locus`: Locus of the integration site
14. `sequence_count`: Count of how many reads are collapsing in the same locus
15. `score`: Mapping quality
16. `strand`: Strand of the read ('+' for 5'-3', '-' for 3'-5')
17. `label`: String
18. `sequence_raw`: Raw sequence
19. `sequence_trimmed`: Trimmed sequence

B.4.2 Import ISs from BAM

Column	Type	Null	Default	Links to	Comments	MIME
prod_header (<i>Primary</i>)	varchar(255)	No				
prod_chr	varchar(100)	Yes	NULL			
prod_locus	int(30)	No				
prod_end	int(30)	No				
prod_strand	varchar(50)	Yes	NULL			
ref_associationid	varchar(255)	Yes	NULL			
ref_matrixid	varchar(255)	Yes	NULL			
ref_poolid	varchar(255)	Yes	NULL			
isread_chr	varchar(255)	Yes	NULL			
isread_start	int(30)	No				
isread_end	int(30)	No				
isread_strand	varchar(255)	Yes	NULL			
isread_RG	varchar(250)	Yes	NULL			
isread_quality	int(10)	No				
isread_NM	int(30)	Yes	NULL			
isread_flag	varchar(255)	Yes	NULL			
isread_cigar	varchar(255)	Yes	NULL			
isread_MD	varchar(255)	Yes	NULL			
isread_insert_size	int(30)	Yes	NULL			
isread_AS	int(30)	Yes	NULL			
isread_XS	int(30)	Yes	NULL			
isread_SA	varchar(255)	Yes	NULL			
isread_nasequence	varchar(1000)	Yes	NULL			
mate_chr	varchar(255)	Yes	NULL			
mate_start	int(30)	Yes	NULL			
mate_end	int(30)	Yes	NULL			
mate_strand	varchar(255)	Yes	NULL			
mate_RG	varchar(250)	Yes	NULL			
mate_quality	int(10)	Yes	NULL			
mate_NM	int(30)	Yes	NULL			
mate_flag	varchar(255)	Yes	NULL			
mate_cigar	varchar(255)	Yes	NULL			
mate_MD	varchar(255)	Yes	NULL			
mate_insert_size	int(30)	Yes	NULL			
mate_AS	int(30)	Yes	NULL			
mate_XS	int(30)	Yes	NULL			
mate_SA	varchar(255)	Yes	NULL			
mate_nasequence	varchar(1000)	Yes	NULL			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	prod_header	424	A	No	
prod_locus	BTREE	No	No	prod_locus	212	A	No	

Figure B.2: MySQL Table Structure for IS imported from BAM file.

import_iss (isatk/script/import_iss.py) produces _refactored tables

1. prod_header (Primary): Header of the read (from FASTQ)
2. prod_chr*: Chromosome

3. `prod_locus*`: Starting locus of the integration site
4. `prod_end*`: Ending locus of the integration site
5. `prod_strand*`: Strand of the read ('+' for 5'-3', '-' for 3'-5')
6. `ref_associationid`: Complete name of the sample (link to the association file)
7. `ref_matrixid`: String
8. `ref_poolid`: Pool ID
9. `isread_chr`: Chromosome
10. `isread_start`: Starting locus of the integration site
11. `isread_end`: Ending locus of the integration site
12. `isread_strand`: Strand of the read ('+' for 5'-3', '-' for 3'-5')
13. `isread_RG`: Read group
14. `isread_quality`: Mapping quality
15. `isread_NM`: Number of mismatches
16. `isread_flag`: BAM alignment flag
17. `isread_cigar`: CIGAR string
18. `isread_MD`: MD score from string
19. `isread_insert_size`: Insert Size
20. `isread_AS`: Alignment score from BAM
21. `isread_XS`: Suboptimal alignment score
22. `isread_SA`: Secondary alignment
23. `isread_nasequence`: Genomic part of the read (trimmed)
24. `mate_chr`: Pair chromosome
25. `mate_start`: Pair starting locus of the integration site
26. `mate_end`: Pair ending locus of the integration site
27. `mate_strand`: Pair strand of the read ('+' for 5'-3', '-' for 3'-5')
28. `mate_RG`: Pair read group

-
29. `mate_quality`: Pair mapping quality
 30. `mate_NM`: Pair number of mismatches
 31. `mate_flag`: Pair BAM alignment flag
 32. `mate_cigar`: Pair CIGAR string
 33. `mate_MD`: Pair MD score from string
 34. `mate_insert_size`: Pair insert Size
 35. `mate_AS`: Pair alignment score from BAM
 36. `mate_XS`: Pair suboptimal alignment score
 37. `mate_SA`: Pair secondary alignment
 38. `mate_nasequence`: Pair genomic part of the read (trimmed)
- * If the mate (pair) is present, these fields are considered for the paired read.

B.4.3 Stats Summary

Column	Type	Null	Default	Links to	Comments	MIME
RUN_ID	varchar(255)	Yes	NULL			
RUN_NAME	varchar(1000)	Yes	NULL			
DISEASE	varchar(255)	Yes	NULL			
PATIENT	varchar(255)	Yes	NULL			
POOL	varchar(255)	Yes	NULL			
TAG	varchar(255)	Yes	NULL			
LTR_ID	varchar(255)	Yes	NULL			
LC_ID	varchar(255)	Yes	NULL			
PHIX_MAPPING	int(30)	Yes	NULL			
PLASMID_MAPPED_BYPOOL	int(30)	Yes	NULL			
RAW_NO_PLASMID	int(30)	Yes	NULL			
BARCODE_MUX	int(30)	Yes	NULL			
LTR_IDENTIFIED	int(30)	Yes	NULL			
TRIMMING_LTRR1	int(30)	Yes	NULL			
TRIMMING_LTRR1R2	int(30)	Yes	NULL			
TRIMMING_LTRR1R2_LCR1	int(30)	Yes	NULL			
TRIMMING_FINAL_RESCUED	int(30)	Yes	NULL			
TRIMMING_FINAL_LTRLC	int(30)	Yes	NULL			
LV_MAPPED	int(30)	Yes	NULL			
BWA_INPUT	int(30)	Yes	NULL			
BWA_MAPPED	int(30)	Yes	NULL			
BWA_MAPPED_PP	int(30)	Yes	NULL			
BWA_MAPPED_ST	int(30)	Yes	NULL			
BWA_MAPPED_OVERALL	int(30)	Yes	NULL			
BWA_ALIGNED_R1	int(30)	Yes	NULL			
RECALIB_MAPPED	int(30)	Yes	NULL			
RECALIB_MAPPED_PP	int(30)	Yes	NULL			
RECALIB_MAPPED_ST	int(30)	Yes	NULL			
RECALIB_MAPPED_OVERALL	int(30)	Yes	NULL			
RECALIB_ALIGNED_R1	int(30)	Yes	NULL			
RECALIB_SOFCLIPPED_READS	int(30)	Yes	NULL			
FILTER_MATE_TO_REMOVE	int(30)	Yes	NULL			
FILTER_CIGARMD_TO_REMOVE	int(30)	Yes	NULL			
FILTER_JOINT_MC_TO_REMOVE	int(30)	Yes	NULL			
FILTER_JOINT_MC_PP	int(30)	Yes	NULL			
FILTER_JOINT_MC_ST	int(30)	Yes	NULL			
FILTER_JOINT_MC_OVERALL	int(30)	Yes	NULL			
FILTER_JOINT_ALIGNED_R1	int(30)	Yes	NULL			
FILTER_ALMQUAL_PP	int(30)	Yes	NULL			
FILTER_ALMQUAL_ST	int(30)	Yes	NULL			
FILTER_ALMQUAL_OVERALL	int(30)	Yes	NULL			
FILTER_ALMQUAL_ALIGNED_R1	int(30)	Yes	NULL			
ISS_FINAL	int(30)	Yes	NULL			
ISS_MAPPED	int(30)	Yes	NULL			
ISS_MAPPED_PP	int(30)	Yes	NULL			
ISS_MAPPED_ST	int(30)	Yes	NULL			
ISS_MAPPED_OVERALL	int(30)	Yes	NULL			
ISS_ALIGNED_R1	int(30)	Yes	NULL			

Figure B.3: MySQL Table Structure for Stats Summary.

1. RUN_ID: RUN (VISPA2) ID
2. RUN_NAME: Concatenation of \$DISEASE\$PATIENT\$POOL variables
3. DISEASE: Disease ID

4. PATIENT: Patient ID
5. POOL: Pool ID
6. TAG: TAG ID (for sample recognition)
7. LTR_ID: LTR ID
8. LC_ID: LC ID
9. PHIX_MAPPING: Number of reads mapping on PhiX genome (overall)
10. PLASMID_MAPPED_BYPOOL: Number of reads mapping on plasmid genome by pool
11. RAW_NO_PLASMID: Number of reads not mapping on plasmid genome by pool
12. BARCODE_MUX: Number of reads Demultiplexed reads by sample
13. LTR_IDENTIFIED: Number of reads with LTR identified
14. TRIMMING_LTRR1: Number of reads with LTR identified on R1
15. TRIMMING_LTRR1R2: Number of reads with LTR identified on R1 and R2
16. TRIMMING_LTRR1R2_LCR1: Number of reads with LTR identified and LC on R1
17. TRIMMING_FINAL_RESCUED: Number of reads rescued
18. TRIMMING_FINAL_LTRLC: Number of trimmed reads overall
19. LV_MAPPED: Number of reads mapping LV genome (internal control band)
20. BWA_INPUT: Number of reads in input on BWA-MEM
21. BWA_MAPPED: Number of reads mapped with BWA-MEM
22. BWA_MAPPED_PP: Number of reads mapped with BWA-MEM properly pair
23. BWA_MAPPED_ST: Number of reads mapped with BWA-MEM as singletons
24. BWA_MAPPED_OVERALL: Number of reads mapped with BWA-MEM overall
25. BWA_ALIGNED_R1: Number of reads mapped with BWA-MEM with only R1
26. RECALIB_MAPPED: NONE
27. RECALIB_MAPPED_PP: NONE
28. RECALIB_MAPPED_ST: NONE
29. RECALIB_MAPPED_OVERALL: NONE

- 30. RECALIB_ALIGNED_R1: NONE
- 31. RECALIB_SOFCLIPPED_READS: NONE
- 32. FILTER_MATE_TO_REMOVE: Number of reads filtered by mate program
- 33. FILTER_CIGARM_D_TO_REMOVE: Number of reads filtered by CIGAR program
- 34. FILTER_JOINT_MC_TO_REMOVE: Number of joint reads filtered by MD
- 35. FILTER_JOINT_MC_PP: Number of joint properly pair reads filtered
- 36. FILTER_JOINT_MC_ST: Number of joint singleton reads filtered
- 37. FILTER_JOINT_MC_OVERALL: Number of joint reads filtered
- 38. FILTER_JOINT_ALIGNED_R1: Number of joint reads filtered by R1
- 39. FILTER_ALMQUAL_PP: Number of properly pair reads filtered by alignment quality
- 40. FILTER_ALMQUAL_ST: Number of singleton reads filtered by alignment quality
- 41. FILTER_ALMQUAL_OVERALL: Number of reads filtered by alignment quality overall
- 42. FILTER_ALMQUAL_ALIGNED_R1: Number of reads filtered by alignment quality on R1
- 43. ISS_FINAL: Number of IS reads (non unique, redundant)
- 44. ISS_MAPPED: Number of IS reads mapped (non unique, redundant)
- 45. ISS_MAPPED_PP: Number of properly pair IS reads (non unique, redundant)
- 46. ISS_MAPPED_ST: Number of singleton IS reads mapped (non unique, redundant)
- 47. ISS_MAPPED_OVERALL: Number of IS reads mapped (non unique, redundant)
- 48. ISS_ALIGNED_R1: Number of IS reads mapped (non unique, redundant) on R1

Appendix C

Some Useful Programs

C.1 VISPA2

C.1.1 Example of VISPA2 bash launch

vispa2 link: `isatk/pipeline/illumina/VISPA2.IlluminaMiSeq.pipeline.sh`

```
#!/bin/bash
source /etc/environment
source /etc/profile

#####

#####
#
#           Breast Cancer - POOL1
#
#
#####

#####

TODAY=$(date +"%Y/%m/%d/%H/%M/%S");

##### ----- start editing from here ----- #####
R1="/storage/dx/backup/nas/LabDevelopment/ftp.gatc-biotech.com/2015-12-28/NG-8959_VP1_lib102988_4287_1_1.fastq.gz";
R2="/storage/dx/backup/nas/LabDevelopment/ftp.gatc-biotech.com/2015-12-28/NG-8959_VP1_lib102988_4287_1_2.fastq.gz";

DISEASE="BreastCancer"; # project main name
PATIENT="InsertionalMutagenesis";
POOLNAME="POOL1";

GENOME="/opt/genome/human/hg19/index/bwa_7/hg19.fa"; ## hg19: /opt/genome/human/hg19/index/bwa_7/hg19.fa ;
mm9: /opt/genome/mouse/mm9/index/bwa_7/mm9.fa ; mfa5: /opt/genome/monkey/mfa5/index/bwa_7/mfa5.fa
BARCODE_LTR="/opt/applications/scripts/isatk/elements/barcode/barcode.LTR.48.list";
BARCODE_LC="/opt/applications/scripts/isatk/elements/barcode/barcode.LC.48.list";

ASSOCIATIONFILE="/opt/applications/scripts/isatk/elements/association/asso.breastcancer.pool1.tsv";
LTR="/opt/applications/scripts/isatk/elements/sequences/LTR.32bp.fa"; # LTR in forward
LTR_rc="/opt/applications/scripts/isatk/elements/sequences/LTR.32bp.rev.fa"; # LTR in reverse complement
LC_fwd="/opt/applications/scripts/isatk/elements/sequences/LC.assayvalidation.fwd.fa"; # Linker Cassette in forward
LC_rev="/opt/applications/scripts/isatk/elements/sequences/LC.assayvalidation.rc.fa"; # Linker Cassette in reverse

DBHOSTID="local";
DBTARGETSCHEMA="sequence_breastcancer";
DBTARGETTABLE="allPools";

GATKREFGENOME="/opt/genome/human/hg19/index/bwa_7/hg19.fa";
CIGARGENOMEID="hg19" ; # Reference genome ID: choose among {hg19 | mm9 | mfa5}
VECTORCIGARGENOMEID="lv"; ## This is the vector reference name (id) used to remove vector sequences.
Choose among: {lv, lvarsa, lvwas, lvkana, lvamp, transposon, giada, hiv}
```

```

CONTAMINANTDB="/opt/applications/scripts/isatk/elements/sequences/UniVec_Tiget.fa";
REMOVE_TMP_DIR="remove_tmp_yes"; # remove tmp dirs? remove_tmp_yes
##### ----- end editing here ----- #####

TMPDIR="/opt/NGS/pipetmpdir/${TODAY}" ;
LOGF="/opt/NGS/log/${TODAY}.${DISEASE}.${PATIENT}.${POOLNAME}.log";
mkdir ${TMPDIR};
PHIXGENOME="/opt/genome/control/phix174/bwa_7/phix174.fa";
LVGENOME="/opt/genome/vector/lv/bwa_7/lv.backbone.fa"; # Change it ONLY if you want to quantify and remove other vectors
or inserted sequences. Alternatives in the GEMINI folder /opt/genome/vector/lv/bwa_7/: {lv.backbone.fa,
lv.backbone.hpgk.arsa.wprem.fa, lv.backbone.wasp.was.wprem.fa, lv.plasmid.amp.fa, lv.plasmid.kana.fa}.
HIV: /opt/genome/hiv/hiv_hxb2cg/bwa_7/hiv.fa ;
# available CPUs
CPUN=$(cat /proc/cpuinfo | grep "model name" | wc -l);
MAXTHREADS=16;
FASTQ_QF="slim"; # FASTQ Quality Filter Methods: slim (QF on R1 80bp and R2 TAGs) or lam (QF only on R1 80bp)
SUBOPTIMALTHRESHOLD='40';
REPEATS="repeats_yes";

#####
if [ ! -r "$R1" ]; then
echo "Error: can't open input file for R1."
exit 1
fi
if [ ! -r "$R2" ]; then
echo "Error: can't open input file for R2."
exit 1
fi
if [ ! -r "$ASSOCIATIONFILE" ]; then
echo "Error: can't open input ASSOCIATION FILE."
exit 1
fi
echo "
[ VISPA2 - PE ] -> STARTING PROCESSING AT:
"
date
vispa2 ${DISEASE} ${PATIENT} ${NGSWORKINGPATH} ${R1} ${R2} ${POOLNAME} ${BARCODE_LTR} ${BARCODE_LC} ${GENOME} ${TMPDIR} ${ASSOCIATIONFILE}
${DBHOSTID} ${DBTARGETSCHEMA} ${DBTARGETTABLE} ${PHIXGENOME} ${LVGENOME} ${CONTAMINANTDB} ${MAXTHREADS} ${GATKREFGENOME} ${CIGARGENOMEID}
${VECTORCIGARGENOMEID} ${SUBOPTIMALTHRESHOLD} ${REMOVE_TMP_DIR} ${LTR} ${LTR_rc} ${LC_fwd} ${LC_rev} ${FASTQ_QF} ${REPEATS}
echo "
[ VISPA2 - PE ] -> FINISHED PROCESSING AT:
"
date;

#####
#-----***-----#

```

C.1.2 FASTQ Quality Filter

Source: `isatk/script/fastq_qf.sh`

Python program to filter out reads with low quality. This software works per FASTQ.
`fastq_qf -a $R1.FASTQ.GZ -b $R2.FASTQ.GZ -o $OUTDIR -t $MAXTHREADS -m $METHOD`

1. `-a [R1.FASTQ.GZ]` - Input File: Illumina R1 FASTQ zipped
2. `-b [R2.FASTQ.GZ]` - Input File: Illumina R2 FASTQ zipped
3. `-o [OUTDIR]` - Output directory
4. `-t [MAXTHREADS]` - Maximum number of parallel threads
5. `-m [METHOD]` - Quality filter on R1 only (LAM) or R1 and R2 (SLiM)

C.1.3 Alignment to the Reference Genome with BWA-MEM

1. `bwa-mem -k 18 -r 1 -M -T 15 -c 1 -R -t $THREADS $R1.FASTQ $R2.FASTQ > $TMPDIR/sam/SAMPLE_1.sam`

- `-k [18]`: Minimum seed length. Matches shorter than k will be missed.
- `-r [1]`: Trigger re-seeding for a MEM longer than `minSeedLen*r`. This is a key heuristic parameter for tuning the performance. Larger value yields fewer seeds, which leads to faster alignment speed but lower accuracy.
- `-M`: Mark shorter split hits as secondary (for Picard compatibility).
- `-T [15]`: Do not output alignment with score lower than 15. This option only affects output.
- `-c [1]`: Discard a MEM if it has more than c occurrence in the genome. This is an insensitive parameter.
- `-R [COMPLETE_NAME]`: Complete read group header line. `'\t'` can be used in STR and will be converted to a TAB in the output SAM. The read group ID will be attached to every read in the output. An example is `'@RG\tID:foo\tSM:bar'`.

2. `samtools view -F 2308 -uS $TMPDIR/sam/SAMPLE_1.sam`

- `-F [FLAG]`: Filter the alignments that will be included in the output to only those alignments that match certain criteria¹.
- `-uS`: Output uncompressed BAM. This option saves time spent on compression/decompression and is thus preferred when the output is piped to another `samtools` command. Ignored for compatibility with previous `samtools` versions. Previously this option was required if input was in SAM format, but now the correct format is automatically detected by examining the first few characters of input.

C.1.4 Repetitive Element Analysis with RepeatMasker

1. `bamtools filter -in $TMPDIR/bam/SAMPLE_1.bam -mapQuality "<10" > $TMPDIR/bam/SAMPLE_1.lowMQ.bam`; (Identifying low mapping quality reads after alignment).

2. `bamtools filter -in $TMPDIR/bam/SAMPLE_1.lowMQ.bam -isFirstMate true > $TMPDIR/bam/SAMPLE_1.lowMQ.R1.bam`; (Filter out the R2 from BAM).

3. `bamtools convert -format fasta -in $TMPDIR/bam/SAMPLE_1.lowMQ.R1.bam -out $TMPDIR/SAMPLE_1.lowMQ.R1.fa`; (FASTA conversion from BAM file).

¹<https://broadinstitute.github.io/picard/explain-flags.html>

4. `RepeatMasker -no_is -species $SPECIE -pa $MAXTHREADS -dir $OUTDIR -q $TMPDIR/SAMPLE_1.lowMQ.R1.fa;`

- `-no_is`: Skips bacterial insertion element check.
- `-species [SPECIE]`: Species of the input sequence. The species name must be a valid NCBI Taxonomy Database² species name and be contained in the RepeatMasker repeat database. Some examples are: *human* and *mouse*.
- `-pa [MAXTHREADS]`: The number of threads to use in parallel (only works for batch files or sequences over 50 kb).
- `-dir [OUTDIR]`: Writes output to this directory (default is query file directory, '-dir .' will write to current directory).
- `-q`: Quick search; 5-10% less sensitive, 2-5 times faster than default.

5. `tail -n+4 ${OUTDIR}/SAMPLE_1.lowMQ.R1.fa.out | awk '
 {if ($1 >= 250) print $1,$2,$3,$4,$5,$6,$7,$8,$9,$10,$11}' |
 sort -uk5,5 | awk 'BEGIN{OFS="\t"}{if($9=="C"){strand="-"}
 else{strand="+"};print"R_"$11,$6-1,$7,$11_"$5",".",strand,$10}'
 > ${OUTDIR}/SAMPLE_1.RM.R1.bed` (this command creates a bed file with all required informations, like an unique ID, the start and the end position in the genome, the family and the sub-family of the repetitive elements).

6. `isa_importrediss_frombed -b $OUTDIR/SAMPLE_1.RM.R1.bed -a $ASSOCIATIONFILE -patient $PATIENT -pool $POOL -tag $TAG -d $DBHOSTID -dbschema $DBSCHEMA -dbtable $DBTABLE_repeats`

- `-b`: BED file to import into MySQL database.
- `-a`: Association File.
- `-patient`: Patient ID.
- `-pool`: Sequencing Pool
- `-tag`: Sample Barcode TAG.
- `-d`: IP of MySQL DBMS.
- `-dbschema`: MySQL target database name.
- `-dbtable`: MySQL target table name to import.

C.1.5 Filtering

`bamtools filter -in $TMPDIR/bam/SAMPLE_1.sorted.md.filter.bam -isMapped true -isMateMapped true -isPaired true -isProperPair true -isPrimaryAlignment true -mapQuality ">=12" -out $TMPDIR/bam/SAMPLE_1.sorted.md.filter_pp.bam`

²<https://www.ncbi.nlm.nih.gov/taxonomy>

- *-in* [BAM] Input BAM to filter.
- *-isMapped* [TRUE] Keep only alignments that were mapped.
- *-isMateMapped* [TRUE] Keep only alignments with mates that mapped.
- *-isPaired* [TRUE] Keep only alignments that were sequenced as paired.
- *-isProperPair* [TRUE] Keep only alignments that passed PE resolution.
- *-isPrimaryAlignment* [TRUE] Keep only alignments marked as primary.
- *-mapQuality* [≥ 12] Keep reads with map quality that matches pattern.
- *-out* [BAM] Output BAM filtered.

Then a BED file with only R2 reads is created (for shear site quantification):

```
bamtools filter -in $TMPDIR/bam/SAMPLE_1.sorted.md.rel.bam -isSecondMate
true -isFirstMate false -isMapped true -isMateMapped true -isPaired true -isProperPair
true -isPrimaryAlignment true | bedtools bamtobed >
$TMPDIR/bed/SAMPLE_1.sorted.allr2reads.bed
```

- *-in* [BAM] Input BAM to filter.
- *-isSecondMate* [TRUE] Keep only alignments marked as second mate.
- *-isFirstMate* [FALSE] Keep only alignments marked as first mate.
- *-isMapped* [TRUE] Keep only alignments that were mapped.
- *-isMateMapped* [TRUE] Keep only alignments with mates that mapped.
- *-isPaired* [TRUE] Keep only alignments that were sequenced as paired.
- *-isProperPair* [TRUE] Keep only alignments that passed PE resolution.
- *-isPrimaryAlignment* [TRUE] Keep only alignments marked as primary.

C.1.6 IS Merging and Collisions

```
create_matrix -dbDataset "matrix_mldwas.mld_was_patients, sequence_breastcancer.bc_merge,
sequence_qlam.cem_reference" -columns sample,tissue,treatment,vector,enzyme -IS_method
classic -bp_rule 7 -collision -tsv -no_xlsx
```

1. *-dbDataset*: Datasets to analyze (tables must be present on MySQL Database).
2. *-columns*: Indicates the columns for the final matrix output. Available fields (from the association file): n_LAM, tag, pool, tissue, sample, treatment, group_name and enzyme.

3. *-IS_method*: Specifies which method run to retrieve Integration Sites: 'classic' or 'gauss' (strand_specific only). You'll be able to tune 'classic' through *-bp_rule* (default provided); 'gauss' method has to be set-up through *-interaction_limit* and *-alpha* (no defaults provided for it).
 - *Classic*: All reads that are in the same window are merged into a single locus, represented by the mode in the window.
 - *Gauss*: *interaction_limit* states, de facto, the number of bin of the histogram you get. Alpha states how many sigmas are equal to half-basepair. Some examples: $\alpha = 1$ means that sigma is half-bp long; then 3bp are long 6sigma, $\alpha = 0.5$ means that sigma is 1-bp long.
4. *-bp_rule*: Minimum number of empty base-pairs between reads belonging to different cluster (also called Covered Bases Ensembles). If you chose 'classic' method to retrieve IS, this number also sets the maximum dimension allowed for a Covered Bases Ensemble ($n+1$ bases). Default option is '3', i.e. 'minimum 3 empty-bp between independent ensembles, an ensemble can span at most 4bp'. Conversely, if you chose 'gauss' method, it will be automatically set equal to *interaction_limit* (overriding your setting) and no limit of dimension will be set for ensembles construction.
5. *-collision*: Produces "collisions" between one dataset and a list of some others. All datasets versus each other. For each dataset, over current, is hung at the end of a column matrix containing for each integration, how many have been found in comparison datasets. Each IS is compared between datasets with a radius equal to *bp_rule+1*.
6. *-tsv*: Produces output matrixes in tab-separated format (UTF-8 encoded).
7. *-no_xlsx*: With this option no excel files are produced.

The program has also some default parameters like **host** (IP address to establish a connection with the server that hosts DB, localhost), **user** (username to log into the server you just chosen through, readonly), **pw** (password for the user you choose to log through, readonlypswd), **dbport** (database port, 3306), **query_steps** (number of row simultaneously retrieved by a single query, 50000000), **reference_genome** (specify reference genome, hg19), **strand_aspecific** (if called, strands will be merged together instead of be treated separately), **set_radius** (along with *-collision* option, here you can set the maximum distance (i.e. loci difference) between two covered bases regarded as 'colliding', None), **diagnostic** (Excel output will be created without any frills but equipped with specific formulas to perform output control, self-coherence and DB coherence), **statistics** (statistical report will be created, equipped with graphs and

many more features constantly developing (bioinfo-oriented). By default, this report is an Excel Workbook file (*.xlsx) but a *.tsv version (less featured) is also available, using `-tsv` option).

C.1.7 IS Annotation

`annotate_matrix [-m $IS_MATRIX.TSV] [-t $TYPE] [-g $GTF_FILE.GTF] [-o $OUTPUT_DIR]`

1. `-m [IS_MATRIX.TSV]`: IS matrix file (tab-separated).
2. `-t [TYPE]`: vispa or γ -TRIS [93] Matrix file.
3. `-g [GTF_FILE.GTF]`: Positions of all data items in a standard gene prediction format (similar to a BED file)³.
4. `-o [OUTPUT_DIR]`: Output directory.

The lines added to the GTF file for compatibility with repeats and mitochondrial chromosome annotation:

```
chrM hg19_knownGene exon 1 368 0.000000 - . gene_id "uc004coq.4"; transcript_id "uc004coq.4";
chrM hg19_knownGene exon 651 674 0.000000 + . gene_id "uc022bqo.2"; transcript_id "uc022bqo.2";
chrM hg19_knownGene exon 1604 1634 0.000000 + . gene_id "uc004cor.1"; transcript_id "uc004cor.1";
chrM hg19_knownGene exon 1844 4264 0.000000 + . gene_id "uc004cos.5"; transcript_id "uc004cos.5";
chrM hg19_knownGene exon 5544 5566 0.000000 - . gene_id "uc022bqp.1"; transcript_id "uc022bqp.1";
chrM hg19_knownGene exon 5586 5606 0.000000 - . gene_id "uc022bqq.1"; transcript_id "uc022bqq.1";
chrM hg19_knownGene exon 5691 5714 0.000000 - . gene_id "uc022bqr.1"; transcript_id "uc022bqr.1";
chrM hg19_knownGene exon 5905 7439 0.000000 + . gene_id "uc031tga.1"; transcript_id "uc031tga.1";
chrM hg19_knownGene exon 7587 7982 0.000000 - . gene_id "uc022bqs.1"; transcript_id "uc022bqs.1";
chrM hg19_knownGene exon 15504 15888 0.000000 - . gene_id "uc022bqs.1"; transcript_id "uc022bqs.1";
chrM hg19_knownGene exon 7587 9208 0.000000 + . gene_id "uc011mfi.2"; transcript_id "uc011mfi.2";
chrM hg19_knownGene exon 8367 8472 0.000000 - . gene_id "uc022bqt.1"; transcript_id "uc022bqt.1";
chrM hg19_knownGene exon 13450 14149 0.000000 - . gene_id "uc022bqt.1"; transcript_id "uc022bqt.1";
chrM hg19_knownGene exon 10060 10404 0.000000 + . gene_id "uc022bqu.2"; transcript_id "uc022bqu.2";
chrM hg19_knownGene exon 10471 12138 0.000000 + . gene_id "uc004cov.5"; transcript_id "uc004cov.5";
chrM hg19_knownGene exon 10761 11231 0.000000 - . gene_id "uc031tgb.1"; transcript_id "uc031tgb.1";
chrM hg19_knownGene exon 13979 14149 0.000000 - . gene_id "uc031tgb.1"; transcript_id "uc031tgb.1";
chrM hg19_knownGene exon 12208 12264 0.000000 + . gene_id "uc004cow.2"; transcript_id "uc004cow.2";
chrM hg19_knownGene exon 12908 14149 0.000000 + . gene_id "uc004cox.4"; transcript_id "uc004cox.4";
chrM hg19_knownGene exon 14675 14698 0.000000 - . gene_id "uc022bqv.1"; transcript_id "uc022bqv.1";
chrM hg19_knownGene exon 14857 15888 0.000000 + . gene_id "uc022bqw.1"; transcript_id "uc022bqw.1";
chrM hg19_knownGene exon 15960 16024 0.000000 - . gene_id "uc022bqx.1"; transcript_id "uc022bqx.1";
chrM hg19_knownGene exon 15999 16571 0.000000 + . gene_id "uc004coz.1"; transcript_id "uc004coz.1";
chrR hg19_knownGene exon 1 99999999 . + . gene_id "REPEAT"; transcript_id "uc000000.0";
```

³Downloadable from: <https://genome.ucsc.edu/cgi-bin/hgTables> but with added lines with chrM and repeats description to annotation.

C.2 ISAnalytics

C.2.1 Example of Usage

```
#####
##      MLD Molecular Follow Up, sorted cells          ##
##      R Script based on ISAnalytics (v0.40)         ##
##      Date:      Rev1: 18 May 2016                 ##
##      Author:    Giulio Spinozzi                  ##
#####

library(XLConnect)
library(ISAnalytics)

##### ===== MLD01, pipe 3.0.8d latest FU ===== #####
isset <- ISDataSetFromXlsx("IA_mldso40cellm_mld01.xlsx")
nm <- colnames(isset)
# identify collisions
patient <- colnames(isset)[colnames(isset) %in% grep("all", colnames(isset),
value=TRUE)]
collisions <- colnames(isset)[colnames(isset) %in% grep("Collision",
colnames(isset), value=TRUE)]
isset <- calculateColAbundance(isset, tot.in.label = TRUE)
isset <- annotateCollisions(isset, with=patient, against=collisions,
assay="abundance")
# remove collisions
isset <- removeCollisions(isset, with=patient, against=collisions,
remove.extra.columns = TRUE)
writeISA(isset, "mld01.countsNoCollisions.tsv", assay= "counts")
# compute abundance by source and time point (and write both the
percentage and the count resulting files)
isset <- calculateColAbundance(isset, normalize.by.source=TRUE)
writeISA(isset, "mld01.abundanceBySource.tsv", assay= "abundance")
# ISAnalytics BoxPlot
pdf("mld01.abundanceBySource.bplot.ISA.pdf", height=8, width=18)
boxplotOutliers(isset, assay="abundance", samples="~CD34")
boxplotOutliers(isset, assay="abundance", samples="~CD14_PB")
boxplotOutliers(isset, assay="abundance", samples="~CD15_PB")
boxplotOutliers(isset, assay="abundance", samples="~CD19_PB")
boxplotOutliers(isset, assay="abundance", samples="~CD3_PB")
dev.off()
# compute sample abundance (column percentage) and write results
isset <- calculateColAbundance(isset, tot.in.label = FALSE)
writeISA(isset, "mld01.abundanceBySampleTP.tsv", assay= "abundance")
# compute sample abundance (column percentage) and write results with
sequence count filter
isset <- scFilter(isset, threshold = 2)
isset <- calculateColAbundance(isset, tot.in.label = FALSE,
assay = "scfilter2",output.assay="abundanceSCfilter2")
writeISA(isset, "mld01.abundanceBySampleTP.sc3filter.tsv",
assay= "abundanceSCfilter2")
```

C.2.2 Example of Abel R Package Usage

```
#####
##      MLD Molecular Follow Up, sorted cells          ##
##      CIS analysis with Abel method V2 (Lenti distr.) ##
##      Date:      Rev1: 120 May 2016                 ##
##      Author:    Giulio Spinozzi                  ##
#####

load("/AbelV2/CISALL1_3-0.RData")

mld01.iss <- "mld01.list"

geneshg19 = "/AbelV2/gene_ucsc_hg19.txt"
addrgene <- "/AbelV2/gene_ucsc_hg19.txt"

##### MLD01 #####
CisLenti(dat.is=mld01.iss, dat.gene=geneshg19, CHR.hum_hg19, vd30, 150,
```

```
10, "c")  
# do classic CIS analysis, with clusters  
Cluster(dat.is=mld01.iss, dat.res="cis.abel.out.mld01.csv", vd30, 150)
```

C.3 Causal Modeling

C.3.1 TRONCO Script for Breast Cancer Progression with CAPRI

R script to generate the progression DAGs for pre/post -treatment conditions with breast cancer cell line BT474 for drug resistance studies.

```
#####
###
###   GIULIO SPINOZZI
###
###   © Last update:   4 February 2016
###   © Revision:     1
###   © Contact:      spinozzi.giulio@hsr.it
###   © Institute:    HSR TIGET
###                   DISCo
###
#####

#set the working directory
work.dir = '~/Desktop/reconstruction/tool_stable';
setwd(work.dir);

#####
#datasets E2L
file.dataset.breast = "data/bt474-data-pre-treatment.txt";
file.dataset.breast = "data/bt474-data-post-treatment.txt";
file.types.breast = "data/bt474-types.txt";
file.events.breast = "data/bt474-events-pre-treatment.txt";
file.events.breast = "data/bt474-events-post-treatment.txt";
#####

#load TRONCO (with CAPRI) package
invisible(sapply(list.files(pattern=".[R$]",path="R",full.names=TRUE),source));

#STARTING EXAMPLE OVARIAN CANCER
#reset all the settings
reset();

#load the types
types.load(file.types.breast);

#load the events
events.load(file.events.breast);

#load the dataset and print it
data.load(file.dataset.breast);
str(data.values);

#load required library
library(Rgraphviz);

#reconstruct the topology with CAPRI and print it
set.seed(12345);
topology = tronco.capri(data.values,nboot=1000);
print(topology);

#plot the resulting topology
tronco.plot(topology,node.th.on=TRUE);
#tronco.plot(topology,primafacie=TRUE,node.th.on=TRUE);

#perform non-parametric and parametric bootstrap
set.seed(12345);
topology <- tronco.bootstrap(topology,type="non-parametric",nboot=100);
#plot the topology
tronco.plot(topology,confidence=TRUE,node.th.on=TRUE);
```


Appendix D

Anticancer Drug Resistance Supplementary Information

D.1 Test of Mutual Exclusivity with Mutex

D.1.1 BT474: Pre-Treatment

Score q-val Members

0.05	0.6	KIF5B	CTNND1	EFNA5
0.05194805194805195	0.315	PBX1	PSMD3	CCNG2
0.09491884262526465	0.6966666666666667	MYO9B	EFNA5	CTNND1
0.10079575596816977	0.565	GRIK1	EFNA5	CTNND1
0.10771302173494902	0.5	PKN2	CTNND1	MAP4K3
0.1093	0.425	FOXA1	CTNND1	CUL3
0.1093	0.36428571428571427	FER	CTNND1	CUL3
0.1189	0.38125	WDR75	EFNA5	CTNND1
0.12413394919168591	0.3611111111111111	MAPK1	ANGPT1	
0.12413394919168591	0.325	ANGPT1	FN1	
0.12413394919168591	0.2954545454545454	GSK3B	ANGPT1	
0.12413394919168591	0.2708333333333333	FN1	ANGPT1	
0.12413394919168591	0.25	NF1	ANGPT1	
0.15476190476190477	0.3364285714285714	EFNA5	NBEA	
0.15476190476190477	0.314	NBEA	EFNA5	
0.17988525143435707	0.375625	CUL3	GNA13	
0.17988525143435707	0.35352941176470587	GNA13	CUL3	
0.2702702702702703	0.7027777777777778	CSNK1D	ANGPT1	
0.2777777777777778	0.7042105263157895	TRAF2	DCC	
0.2777777777777778	0.669	DCC	TRAF2	
0.2815926139642239	0.6476190476190476	RIPK2	CTNND1	MAP2K4
0.2815926139642239	0.6181818181818182	VAV3	CTNND1	MAP4K3
0.2815926139642239	0.591304347826087	CDH2	CTNND1	MAP2K4
0.2885670088595134	0.5983333333333333	BCL2	PPP2CB	PSMD3
0.2925	0.5824	CEP290	CTNND1	
0.2925	0.56	SMYD3	CTNND1	
0.2925	0.5392592592592593	SOS2	CTNND1	
0.2925	0.52	PRKACB	CTNND1	PSMD3
0.2925	0.5020689655172415	AHR	CTNND1	
0.2925	0.4853333333333334	CNTN1	CTNND1	
0.2925	0.4696774193548387	MIR1297	CTNND1	
0.2925	0.455	CTNND1	CEP290	
0.3035594358629953	0.4703030303030303	RPS6KB1	ANGPT1	MAP2K4
0.3083612040133779	0.4697058823529412	PVRL3	CTNND1	MAP2K4
0.3083612040133779	0.4562857142857143	CBLE	CTNND1	MAP2K4
0.3083612040133779	0.4436111111111111	CDK17	CTNND1	MAP2K4
0.3179791976225854	0.4508108108108108	MAP2K4	ANGPT1	TAOK3
0.3179791976225854	0.43894736842105264	TAOK3	ANGPT1	MAP2K4
0.3225806451612903	0.4415384615384615	NDST4	FN1	
0.3333333333333333	0.4575	MIR622	CUL3	
0.3333333333333333	0.44634146341463415	PPP2CB	CUL1	PSMD3
0.3333333333333333	0.4357142857142857	SUPT3H	CUL1	PSMD3

0.35219399538106233 0.4679069767441861 PSMD3 ANGPT1
0.36905582356995176 0.49500000000000005 NUP160 PPP2CB PSMD3
0.37037037037037035 0.49444444444444444 ITCH PSMD3
0.38095238095238093 0.5006521739130435 NME2 ESR1 CTNND1
0.38095238095238093 0.49000000000000005 WWC2 ESR1
0.38095238095238093 0.47979166666666667 ESR1 WWC2
0.4153 0.5491836734693878 MIR30B MIR1297
0.4284559417946645 0.5672 TADA2A EHMT1 MEN1
0.4284559417946645 0.556078431372549 EHMT1 TADA2A MEN1
0.4284559417946645 0.5453846153846154 MEN1 EHMT1 TADA2A
0.4324712643678161 0.5443396226415095 LAMA2 EFNA5
0.4324712643678161 0.5342592592592593 ZMYM2 EFNA5
0.4324712643678161 0.5245454545454545 PTPRA EFNA5
0.4345 0.5208928571428572 GRIN1 EFNA5
0.4546 0.5631578947368421 COL12A1 LAMA2
0.5263157894736842 0.7153448275862069 TRPC6 WASL
0.5263157894736842 0.7032203389830509 WASL TRPC6
0.5270903010033445 0.6925 TRPC4AP TRPC6 MAP2K4
0.5277777777777778 0.6827868852459016 LPAR4 DCC
0.5277777777777778 0.6717741935483871 MARK3 DCC
0.5401 0.6796825396825397 BRAF CTNND1 LRRK2
0.5401 0.6690625 SPOP CTNND1
0.5401 0.6587692307692308 CEACAM6 CTNND1 FN1
0.5401 0.6487878787878788 MED1 CTNND1
0.5401 0.6391044776119403 CAPN1 CTNND1
0.5401 0.6297058823529412 FGF12 CTNND1
0.5401 0.6205797101449275 IQGAP1 CTNND1 MAP2K4
0.5401 0.6117142857142858 TRIM37 CTNND1 MAP2K4
0.5401 0.6030985915492958 LRPPRC GSK3B CTNND1
0.5401 0.5947222222222223 HSPD1 GSK3B CTNND1
0.5510204081632653 0.6031506849315069 RICTOR TRDN
0.5510204081632653 0.595 TSC2 RICTOR
0.5510204081632653 0.5870666666666667 TRDN RICTOR
0.5714285714285714 0.6251315789473684 TLK1 CTNND1 CBLB
0.5882352941176471 0.6535064935064935 EPHA5 EFNA5
0.5888290713324361 0.645897435897436 B3GALT6 MAPK1 CTNND1
0.5888290713324361 0.6377215189873418 MIR181A1 MAPK1 CTNND1
0.6009209516500383 0.6515 PIAS1 NLK
0.6009209516500383 0.6434567901234568 NLK PIAS1
0.6135394456289979 0.6587804878048781 MAP4K5 TLK1
0.627020702702071 0.6860240963855422 WDR3 NBEA
0.632183908045977 0.6898809523809524 KIF21A EFNA5
0.6414 0.7003529411764706 MIR30D MIR30B
0.6585365853658537 0.7309302325581395 HSF1 CUL3 MTOR
0.6758 0.768735632183908 ACAP2 CTNND1
0.6758 0.7599999999999999 CDC42BPA CTNND1
0.6875 0.7811235955056179 DSCAM TRAF2 MAP2K4
0.6875418620227729 0.7724444444444444 ALMS1 GSK3B
0.6875418620227729 0.7639560439560439 RAB6A GSK3B
0.7 0.7860869565217391 CUL1 NLK CCNG2
0.7142857142857143 0.8186021505376344 ACTR3 ARHGAP39
0.7142857142857143 0.8098936170212766 NR2F1 WWC2
0.7142857142857143 0.8013684210526315 PTP4A1 ARHGAP6
0.7142857142857143 0.7930208333333333 ARHGAP6 PTP4A1
0.7142857142857143 0.7848453608247422 ARHGAP39 ACTR3
0.7251908396946565 0.7911224489795918 RPS6KA6 AHR
0.7589491916859122 0.842929292929293 GRB7 ANGPT1
0.7647058823529411 0.8454 EPHA3 MIR30B EFNA5
0.7684021543985637 0.8428712871287128 HTR2B LPAR4
0.7692307692307693 0.8451960784313725 ZNFX1 UBE3A
0.7692307692307693 0.8369902912621359 KDM2A EHMT1 MEN1
0.7692307692307693 0.8289423076923076 UBE3A ZNFX1
0.7712 0.8229523809523809 PGR CTNND1
0.8125 0.8735849056603773 CCNI ARMC8 CCNG2
0.8125 0.8654205607476635 MAP4K3 ARMC8
0.8125 0.8574074074074074 ARMC8 MAP4K3
0.8125 0.8495412844036697 NDUFV2 ARMC8
0.8333333333333334 0.8905454545454545 NUP35 KPNA4
0.8333333333333334 0.8825225225225225 NUP205 P4HB
0.8333333333333334 0.874642857142857 MYO3A MAP4K3 CTNND1
0.8333333333333334 0.8669026548672566 P4HB NUP205
0.8333333333333334 0.859298245614035 KPNA4 NUP35
0.8368 0.8543478260869565 MIR548A3 PSMD3
0.8368 0.8469827586206896 PDCD6IP PSMD3
0.8368 0.8397435897435898 UBE2E1 PSMD3 TADA2A
0.8581560283687943 0.8641525423728813 TJP1 TRDN

0.864406779661017 0.8659663865546218 ARHGAP32 ARHGAP39
0.8666666666666667 0.86475 CCNG2 CTNND2
0.8666666666666667 0.8576033057851239 UIMC1 CTNND2
0.8666666666666667 0.8505737704918033 CTNND2 CCNG2
0.8818681318681318 0.8625203252032521 VAPA VAPB
0.8818681318681318 0.8555645161290323 VAPB VAPA
0.8821603927986906 0.84904 ITGAV COL12A1
0.8928571428571429 0.8597619047619047 MIR612 PGR
0.8982 0.860236220472441 SGK3 MIR30B
0.8982 0.853515625 SH3GLB1 MIR30B
0.906801007556675 0.8604651162790697 SDCCAG8 ALMS1
0.9090909090909091 0.8746923076923077 PIK3C3 CTNND1 MAP4K3
0.9090909090909091 0.8680152671755724 PRIM2 MCM8
0.9090909090909091 0.8614393939393938 EPHA6 MIR30B
0.9090909090909091 0.8549624060150376 PHF16 MEN1
0.9090909090909091 0.8485820895522388 MCM8 PRIM2
0.9090909090909091 0.8422962962962962 TBL1XR1 MEN1
0.9090909090909091 0.8361029411764705 CAMK2D ANGPT1
0.9090909090909091 0.83 CENPQ NUP160
0.9130434782608695 0.8298550724637681 FOXH1 PIAS1
0.92 0.8410071942446044 MTOR RPA3
0.92 0.8350000000000001 RPA3 MTOR
0.9230769230769231 0.8370212765957447 ERBB2 MIR30B
0.9230769230769231 0.8311267605633803 HLF MIR30B
0.9268 0.8307692307692307 RAB14 CTNND1
0.9268 0.825 MYO5A CTNND1
0.9268 0.8193103448275861 SOCS6 CTNND1
0.9268 0.8136986301369863 FOXF1 CTNND1
0.9268 0.8081632653061225 NBN CTNND1
0.9268 0.8027027027027027 HHAT CTNND1
0.9268 0.7973154362416107 ARID2 CTNND1
0.9268 0.7919999999999999 MAPK8IP3 CTNND1 LRRK2
0.9337027914614121 0.8071523178807947 SEMA3A EFNA5
0.9411764705882353 0.8196052631578947 MIR618 PIAS1
0.9722222222222222 0.8909803921568628 LPHN3 DCC
0.9722222222222222 0.8851948051948052 FGF10 DCC EFNA5
0.9722222222222222 0.8794838709677419 GPR37 DCC
0.975609756097561 0.8801923076923077 LRRK2 RICTOR
0.9789260969976905 0.8822929936305733 MMP16 ANGPT1
0.98328025477707 0.884493670886076 TNFSF13B TRAF2
0.9858718125430738 0.8825157232704403 CPSF1 NUP160
0.9906 0.8885 PCDH20 CTNND1
0.9959218112783013 0.9045962732919254 STXBP4 PPP2CB
0.9977744807121661 0.9115432098765431 ADCY9 FOXA1
0.9979 0.9068098159509203 OSMR CTNND1
0.9979 0.901280487804878 PHLPP1 CTNND1
0.99822695035461 0.8986060606060606 SYT1 TRDN
1.0 1.3884939759036146 MED24 TBL1XR1
1.0 1.380179640718563 TPD52 CAMK2D
1.0 1.3719642857142857 ERO1LB CTNND2
1.0 1.363846153846154 KIFAP3 PBX1
1.0 1.3558235294117646 STAG1 CENPQ
1.0 1.3478947368421053 PTPRK CTNND1
1.0 1.3400581395348838 UBR4 MTOR
1.0 1.3323121387283237 MED13 MAPK1
1.0 1.3246551724137932 SCYL3 FOXA1
1.0 1.3170857142857144 UBE2J2 CTNND1
1.0 1.3096022727272727 OR4E2 DCC
1.0 1.3022033898305085 MED17 MAPK1
1.0 1.2948876404494383 NEK7 CTNND1
1.0 1.2876536312849163 ARHGEF26 GRIK1
1.0 1.2805 AHCTF1 CENPQ
1.0 1.2734254143646409 RIMS1 TRPC6
1.0 1.2664285714285715 SKA2 CENPQ
1.0 1.2595081967213115 PLA2G4A TRDN
1.0 1.2526630434782609 GBE1 GSK3B
1.0 1.245891891891892 PDE11A XRN1
1.0 1.2391935483870968 GUCY1A2 ESR1
1.0 1.2325668449197862 FANCB PDCD6IP
1.0 1.2260106382978724 PGAP1 MAPK1
1.0 1.2195238095238097 SPEN CAMK2D
1.0 1.2131052631578947 ARHGEF12 GRIK1
1.0 1.2067539267015708 ARL4A PBX1
1.0 1.20046875 GULP1 ACAP2
1.0 1.1942487046632124 EXOC3 CTNND1
1.0 1.1880927835051547 PDS5A SKA2

1.0 1.182 OR2F1 DCC
 1.0 1.175969387755102 CDH18 CTNND2
 1.0 1.1700000000000000 THRB MIR612
 1.0 1.1640909090909090 GRIK2 EFNA5
 1.0 1.1582412060301508 KNTC1 NUP160
 1.0 1.15245 XRN1 PDE11A
 1.0 1.1467164179104479 CSE1L BCL2
 1.0 1.141039603960396 HCN1 ADCY9
 1.0 1.1354187192118228 SLC6A15 MIR612
 1.0 1.1298529411764706 ATG3 SH3GLB1
 1.0 1.1243414634146343 NDST3 NDST4
 1.0 1.118883495145631 APLF CAMK2D
 1.0 1.1134782608695653 ARID4B MIR622
 1.0 1.108125 STXBP5L CTNND1
 1.0 1.1028229665071771 ANO3 TBL1XR1
 1.0 1.0975714285714286 FIGU MAPK1
 1.0 1.0923696682464454 RFWD2 WDR3
 1.0 1.0872169811320755 NCOA3 EHMT1
 1.0 1.082112676056338 NCOA6 MEN1
 1.0 1.0770560747663551 ARHGAP42 VAV3
 1.0 1.072046511627907 KDM3A MEN1
 1.0 1.0670833333333334 PDC MIR612

D.1.2 BT474: Post-Treatment

Score q-val Members

0.054945054945054944 1.0 AHCTF1 NSL1
 0.054945054945054944 0.5 NSL1 AHCTF1
 0.17144060657118787 2.54 PVRL3 FOXA1
 0.17144060657118787 1.905 FOXA1 PVRL3
 0.23809523809523808 3.19 TCF3 CAV2
 0.23809523809523808 2.6583333333333333 CAV2 TCF3
 0.3074291300097752 3.775714285714286 GNB1 COL12A1
 0.3074291300097752 3.30375 COL12A1 GNB1
 0.3125 3.058888888888889 PX1 CAV2
 0.3225806451612903 3.005 GALNT13 GALNT3
 0.3225806451612903 2.731818181818182 GALNT3 GALNT13
 0.3333333333333333 2.7175 CAB39L PRKAA1
 0.3333333333333333 2.5084615384615385 PRKAA1 CAB39L
 0.38461538461538464 2.942142857142857 SLC8A1 STX19
 0.38461538461538464 2.746 STX19 SLC8A1
 0.416666666666667 2.92375 CUL3 FOXA1 HSF1
 0.4242424242424242 2.7852941176470587 C2 MBL2
 0.4242424242424242 2.6305555555555555 MBL2 C2
 0.43478260869565216 2.6310526315789473 GLI3 SLC8A1
 0.45743329097839897 2.663 NRK CLK4
 0.46627131208302447 2.5828571428571427 CORO1C EML4
 0.46627131208302447 2.4654545454545453 EML4 CORO1C
 0.4749455337690632 2.3982608695652172 MAP3K7 CLK4
 0.4749455337690632 2.2983333333333333 CLK4 MAP3K7
 0.4749455337690632 2.2064 TAB2 MAP3K7
 0.47619047619047616 2.187307692307692 RFWD2 EML4
 0.5 2.278888888888889 PGR YES1
 0.5 2.1975000000000002 YES1 PGR
 0.5166177908113392 2.176896551724138 WDR26 GNB1
 0.5166177908113392 2.1043333333333334 VCP EML4 GNB1
 0.5166177908113392 2.036451612903226 PRICKLE2 GNB1
 0.5263157894736842 2.06 CTCF PGR
 0.541666666666667 2.053030303030303 ARMC1 PVRL3
 0.5714285714285714 2.1644117647058825 KITLG IL6ST
 0.5769230769230769 2.1228571428571428 WDR64 EML4
 0.5853658536585366 2.1005555555555555 P2RY1 OR2A5
 0.5853658536585366 2.043783783783784 OR2A5 P2RY1
 0.6 2.108421052631579 RPS6KA3 PVRL3
 0.6025641025641025 2.058205128205128 HUWE1 PRKAA1
 0.6190476190476191 2.0695 MED13 MED1
 0.6190476190476191 2.0190243902439025 TP63 MED1
 0.6190476190476191 1.970952380952381 MED1 TP63
 0.6190476190476191 1.9251162790697676 MED13L MED1
 0.625 1.9506818181818182 LPHN3 P2RY1 GRM3
 0.625 1.9073333333333333 CUL5 MIB2
 0.625 1.8658695652173913 PIK3CB GNB1
 0.625 1.8261702127659574 IRS4 PRKAA1
 0.625 1.788125 MIB2 CUL5

0.625 1.7516326530612245 RPL8 EIF1AX
0.625 1.7166 EIF1AX RPL8
0.625 1.6829411764705882 NCOR1 MED1
0.625 1.650576923076923 PRKCA GNB1
0.6412157153446998 1.677358490566038 RAE1 EML4
0.6486486486486487 1.6725925925925924 COL4A5 COL12A1
0.6486486486486487 1.642181818181818 LEPREL1 COL12A1
0.6494178525226391 1.6155357142857143 PIK3C2A IL6ST
0.6494178525226391 1.5871929824561404 IL6ST PIK3C2A
0.6494178525226391 1.5598275862068964 YWHAE IL6ST
0.65 1.5359322033898306 RICTOR PIP4K2A
0.65 1.5103333333333333 PIP4K2A RICTOR
0.65625 1.5034426229508195 ZZZ3 SETDB2 CHD3
0.6666666666666666 1.5651612903225807 ZWINT ESCO2
0.6666666666666666 1.5403174603174603 CHD3 ARID2 SETDB2
0.6666666666666666 1.51625 ESCO2 ZWINT
0.6666666666666666 1.492923076923077 SETDB2 ARID2 CHD3
0.6756756756756757 1.4903030303030302 STAM2 HUWE1
0.6799601196410767 1.4804477611940299 NEK10 PVRL3
0.6799601196410767 1.4586764705882354 PTPN2 PVRL3
0.6799601196410767 1.4375362318840579 GRB7 PVRL3
0.6842105263157895 1.4345714285714286 TADA1 GATAD2B
0.6842105263157895 1.4143661971830985 GATAD2B TADA1
0.6875 1.4068055555555556 SMAD2 IL6ST
0.7073170731707317 1.4426027397260275 GRM3 OR2A5 P2RY1
0.7142857142857143 1.4925675675675676 XRCC5 PAWR
0.7142857142857143 1.4726666666666668 DCAF6 EML4
0.7142857142857143 1.4532894736842106 RPS6KB1 IL7
0.7142857142857143 1.4344155844155844 PAWR XRCC5
0.7291666666666666 1.458974358974359 TSC2 ARMC1
0.7291666666666666 1.440506329113924 RAPGEF1 ARMC1
0.7333333333333333 1.442125 ARHGEF10L ARHGEF38
0.7333333333333333 1.424320987654321 ARHGEF38 ARHGEF10L
0.7407407407407407 1.435609756097561 TRAF7 WDR26
0.7407407407407407 1.4183132530120481 KPNA1 WDR26
0.7619047619047619 1.4714285714285713 SIAH2 TCF3
0.7676214546475709 1.4728235294117646 WDR90 FOXA1 EML4
0.7676214546475709 1.4556976744186045 HLF FOXA1 IL6ST
0.7676214546475709 1.4389655172413793 WDR24 FOXA1 EML4
0.7676214546475709 1.4226136363636364 TRAF2 FOXA1 IL7
0.7676214546475709 1.406629213483146 EHM2 FOXA1 MED1
0.7676214546475709 1.391 ARID2 FOXA1
0.7676214546475709 1.3757142857142857 KAT7 FOXA1 MED1
0.7692307692307693 1.4051086956521741 PIK3CA IL6ST
0.7692307692307693 1.3900000000000001 STAG1 ESCO2 NSL1
0.7692307692307693 1.375212765957447 IL7 PIK3CA
0.7692307692307693 1.3607368421052632 RPL28 EIF1AX
0.7692307692307693 1.3465625 TBL1XR1 MME
0.7692307692307693 1.3326804123711342 MME TBL1XR1
0.7692307692307693 1.3190816326530612 CDK11B PAWR
0.7692307692307693 1.305757575757576 SDCS7 IL7
0.7777777777777778 1.3214 ANO6 MME
0.782608695652174 1.3233663366336634 YWHAQ IL6ST
0.7904656319290465 1.333235294117647 YWHAB WDR26
0.8 1.358155339805825 BAZ2A ARID2
0.8096304591265397 1.3692307692307693 MIR16-2 MAP3K7 TAB2
0.8125 1.369142857142857 ITSN2 PIP5K1B ARHGEF38
0.8125 1.3562264150943395 PIP5K1B FOXA1
0.8125 1.3435514018691588 FARP2 PIP5K1B ARHGEF38
0.8125 1.3311111111111111 GULP1 PIP5K1B
0.8125 1.3188990825688072 TLN2 PIK3CE
0.8125 1.3069090909090909 ARHGAP15 PIP5K1B
0.8125 1.295135135135135 DCAF8L1 PIP5K1B
0.8163265306122449 1.2949107142857144 SH2D4B KITLG
0.8163265306122449 1.2834513274336283 EPHA7 KITLG ARHGEF38
0.8163265306122449 1.2721929824561404 EPHA6 KITLG ARHGEF38
0.8181818181818182 1.267304347826087 ETF1 RPL8
0.8214285714285714 1.2650862068965518 TAOK1 TAB2
0.8214285714285714 1.2542735042735043 BMP5 TAB2
0.8214285714285714 1.24364406779661 ITCH TAB2
0.8235294117647058 1.2408403361344538 ARHGAP6 ARHGEF38
0.8292682926829268 1.2485833333333334 OR4E2 GRM3 P2RY1
0.830423940149626 1.2400826446280993 FOSL1 SLC8A1
0.8333333333333334 1.2850000000000001 HTR1B GRM3 P2RY1
0.8333333333333334 1.2745528455284554 BRAF TAB2 RIPK2
0.8333333333333334 1.2642741935483872 SRGAP1 ARHGEF38

0.8333333333333334 1.2541600000000002 LRP5 CUL3
0.8333333333333334 1.2442063492063493 IFT80 EML4
0.8333333333333334 1.2344094488188977 WASL ARHGEF38
0.8333333333333334 1.224765625 BAI3 GRM3 P2RY1
0.8333333333333334 1.2152713178294574 GRM8 GRM3 P2RY1
0.8406346531873069 1.2218461538461538 ROCK1 FOXA1
0.8406346531873069 1.2125190839694657 UBE2J2 FOXA1
0.8406346531873069 1.2033333333333334 ARHGEF4 FOXA1
0.8406346531873069 1.1942857142857144 TBC1D4 FOXA1
0.8406346531873069 1.1853731343283582 CAPN1 FOXA1
0.8406346531873069 1.1765925925925926 NBN FOXA1
0.8406346531873069 1.1679411764705883 RRAS2 FOXA1
0.8406346531873069 1.1594160583941606 HS3ST6 FOXA1
0.8406346531873069 1.1510144927536232 MIR1206 FOXA1
0.8406346531873069 1.1427338129496403 RIT2 FOXA1
0.8406346531873069 1.1345714285714286 MCF2L FOXA1
0.8406346531873069 1.1265248226950355 LPXN FOXA1
0.8406346531873069 1.1185915492957748 DMPK FOXA1
0.8406346531873069 1.1107692307692307 AXIN1 FOXA1
0.8406346531873069 1.1030555555555557 P4HB FOXA1
0.8406346531873069 1.095448275862069 UBE2W FOXA1
0.8406346531873069 1.087945205479452 MOB1A FOXA1
0.8421052631578947 1.0861904761904762 CCNL2 CTCF
0.84375 1.0836486486486485 PRPF4B CLK4
0.8461538461538461 1.0904026845637584 SCYL2 NRK
0.8461538461538461 1.0831333333333333 CNTRL RIPK2
0.8461538461538461 1.0759602649006623 LEMD3 VRK2
0.8461538461538461 1.0688815789473685 VRK2 LEMD3
0.8461538461538461 1.0618954248366013 NFKB1 IL6ST
0.8461538461538461 1.055 RIPK2 CNTRL
0.8565493646138808 1.0735483870967741 ERBB4 PIP5K1B GNB1
0.8565493646138808 1.0666666666666667 ERBB2 PIP5K1B GNB1
0.8565493646138808 1.0598726114649681 PKP4 PIP5K1B GNB1 ARHGEF38
0.8565493646138808 1.0531645569620254 EFNAS CNTRL GNB1
0.8565493646138808 1.0465408805031446 SLIT2 PIP5K1B GNB1 ARHGEF38
0.8565493646138808 1.04 FGF10 PIP5K1B GNB1 ITSN2
0.8565493646138808 1.0335403726708075 ARHGAP39 PIP5K1B ARHGEF38 GNB1
0.8571428571428571 1.038641975308642 NCOA3 MED1
0.8571428571428571 1.0322699386503067 TRPC3 IL6ST
0.8648648648648649 1.045731707317073 MYBL1 HUWE1
0.8666666666666667 1.0484848484848486 AURKA TNS3
0.8666666666666667 1.0421686746987953 TNS3 AURKA
0.8666666666666667 1.035928143712575 PUF60 STAM2 NOTCH1
0.8695652173913043 1.0360119047619047 PTPRK NOTCH1
0.8695652173913043 1.0298816568047338 NOTCH1 PTPRK
0.8813278008298755 1.0562352941176472 NOX4 CAPN1
0.8823529411764706 1.0553801169590644 PSMD1 PAWR
0.8947368421052632 1.0825581395348836 JMJ1D1 TADA1
0.8947368421052632 1.076300578034682 CEP192 CDK11B
0.8947368421052632 1.0701149425287355 CEP63 CEP192
0.90625 1.0946285714285715 XBP1 PIP5K1B
0.9090909090909091 1.1427272727272728 TGFBR1 AXIN1
0.9090909090909091 1.136271186440678 HGF IL6ST
0.9090909090909091 1.1298876404494382 GRIK2 KITLG ARHGEF38
0.9090909090909091 1.1235754189944134 NCOA1 SLC8A1 CDK11B
0.9090909090909091 1.1173333333333333 GALNT7 GALNT13
0.9090909090909091 1.111160220994475 LHCGR OR2A5
0.9090909090909091 1.1050549450549452 PRDM9 SETDB2 CHD3
0.9090909090909091 1.099016393442623 SKA2 ESCO2 NSL1
0.9090909090909091 1.0930434782608696 WHD1 EML4
0.9090909090909091 1.0871351351351353 TRPC6 SLC8A1
0.9090909090909091 1.0812903225806452 EPHA5 PIP5K1B ARHGEF38
0.9090909090909091 1.0755080213903743 RDB01 PIP5K1B PTPRK
0.9090909090909091 1.0697872340425532 PSMA1 CUL3 PSMD1
0.9090909090909091 1.0641269841269843 UBE2H UBE2J2
0.9090909090909091 1.0585263157894738 CTNND2 ARHGEF38 GNB1
0.9102564102564102 1.053717277486911 CUL4A PRKAA1
0.9102564102564102 1.0482291666666665 WWP1 PRKAA1
0.9166666666666666 1.066943005181347 STRN FOXA1 GNB1
0.9166666666666666 1.061443298969072 ABI1 WASL
0.9166666666666666 1.0559999999999998 CSTF1 CORO1C
0.9166666666666666 1.0506122448979591 PLCB4 GNB1 SLC8A1
0.9166666666666666 1.045279187817259 ECT2 MCF2L
0.9230769230769231 1.0596464646464647 MIR622 CUL3
0.9230769230769231 1.054321608040201 CSNK1D CUL3
0.9230769230769231 1.04905 BMP2K PRKAA1

0.9230769230769231 1.0438308457711443 YAP1 PRKAA1
0.9230769230769231 1.0386633663366336 SCNN1D STX19
0.9230769230769231 1.0335467980295567 PDGFRA PRKAA1
0.9230769230769231 1.0284803921568628 RAB5A AXIN1
0.9230769230769231 1.0234634146341464 ANAPC11 CSNK1D
0.9230769230769231 1.018495145631068 MSL2 CSNK1D
0.9230769230769231 1.0135748792270531 AKT3 PRKAA1
0.9230769230769231 1.0087019230769232 NFIA PRKAA1
0.9259259259259259 1.0093301435406699 EML5 WDR26
0.9285714285714286 1.0146190476190475 TRIM37 GNB1 GATA3
0.9333333333333333 1.022132701421801 OR2F1 OR2A5 CALCRL
0.9375 1.0266509433962265 HSF1 XRCC5
0.9375 1.021830985915493 MIR181A1 SMAD2
0.9375 1.017056074766355 HSPD1 LAMC1
0.9375 1.012325581395349 TANK XRCC5
0.9375 1.007638888888889 LAMC1 COL4A5
0.9375 1.0029953917050691 VPS37A EIF1AX
0.9411764705882353 1.0085321100917433 SYMPK CSTF1
0.9473684210526315 1.0208675799086757 RARB CTCF
0.9512195121951219 1.0248181818181818 VCAM1 PIP4K2A
0.9565217391304348 1.0365610859728507 RAB13 GLI3
0.9599217986314761 1.04009009009009 RYR3 GNB1
0.9599217986314761 1.0354260089686098 ARHGFE26 GNB1
0.9599217986314761 1.0308035714285715 DYNLL2 GNB1
0.9599217986314761 1.0262222222222221 NUP54 GNB1
0.9599217986314761 1.0216814159292036 RASAL2 GNB1
0.9612403100775194 1.0206607929515419 MARCKS ROCK1
0.9655172413793104 1.0319298245614035 CRK TLN2
0.9666666666666667 1.030524017467249 DNMT3 RPS6KA3
0.96875 1.0325652173913045 CTNNA3 PIP5K1B
0.9696969696969697 1.0309090909090908 NSD1 SETDB2
0.9696969696969697 1.0264655172413792 PPM1L VAPB
0.9696969696969697 1.0220600858369098 VAPB PPM1L
0.9705093833780161 1.0190598290598292 KPNA4 KPNA1
0.9791666666666666 1.0400851063829786 RRM2B ARMC1
0.9868035190615836 1.0533050847457628 HNF4G PIP5K1B GNB1
0.9868035190615836 1.048860759493671 SPRY2 PIP5K1B GNB1
0.9890560875512996 1.0490336134453782 CDC42SE2 HLF
0.9953664700926707 1.0595397489539748 TJP1 FOXA1
0.9953664700926707 1.0551249999999999 NRIP1 MED1 FOXA1
1.0 1.8209958506224067 ERO1LB FOXA1
1.0 1.8134710743801654 PIK3C3 ARHGFE38
1.0 1.806008230452675 STAG2 NSL1
1.0 1.7986065573770493 TADA2A SETDB2
1.0 1.791265306122449 PTPRG NFKB1
1.0 1.7839837398373983 MECP2 CSNK1D
1.0 1.776761133603239 F7 SLC8A1
1.0 1.7695967741935485 SCYL3 FOXA1
1.0 1.7624899598393575 UNC13B PRKCA
1.0 1.7554400000000001 KCNJ3 GNB1
1.0 1.7484462151394422 SCRIB NFKB1
1.0 1.7415079365079367 PABPC1 RNPS1
1.0 1.7346245059288539 RCHY1 NBN
1.0 1.7277952755905512 SPOP FOXA1
1.0 1.7210196078431372 GUCY2F FOXA1
1.0 1.714296875 PTH2R CALCRL
1.0 1.707626459143969 PCDH10 CTNND2
1.0 1.7010077519379845 ACACA PRKAA1
1.0 1.6944401544401544 NUF2 ESCO2
1.0 1.687923076923077 PCDH15 CTNND2
1.0 1.681455938697318 MIR548A3 IL6ST
1.0 1.6750381679389315 PCDH18 CTNND2
1.0 1.6686692015209126 OR5AK2 CALCRL
1.0 1.6623484848484849 CDC42BPA PRKCA
1.0 1.6560754716981132 RGS7 PRKCA
1.0 1.6498496240601503 BACH1 CTCF
1.0 1.6436704119850187 SH3RF1 FOXA1
1.0 1.637537313432836 ELF1 FOSL1
1.0 1.631449814126394 UBE2G1 FOXA1
1.0 1.6254074074074074 GHR GNB1
1.0 1.619409594095941 RAP2B FOXA1
1.0 1.6134558823529412 TPX2 AURKA
1.0 1.6075457875457877 MMP16 PIP5K1B
1.0 1.6016788321167883 MMP13 FOXA1
1.0 1.5958545454545454 BRWD3 FOXA1
1.0 1.590072463768116 EIF4A3 RNPS1

1.0 1.584332129963899 CHSY3 CTCF
1.0 1.57863309352518 DOK5 FOXA1
1.0 1.5729749103942652 NAB1 NFKB1
1.0 1.567357142857143 RUVBL2 SMARCE1
1.0 1.561779359430605 ZNF652 TCF12
1.0 1.556241134751773 USP34 FOXA1
1.0 1.5507420494699646 PHLDA1 AURKA
1.0 1.545281690140845 CDH12 ERBB4
1.0 1.5398596491228072 PLSCR1 CTCF
1.0 1.5344755244755246 RPS6KA6 YES1
1.0 1.5291289198606273 P2RY10 GRM3
1.0 1.5238194444444444 CDH18 ERBB4
1.0 1.5185467128027683 PHF16 KAT7
1.0 1.5133103448275862 CDH19 ERBB4
1.0 1.5081099656357388 NARFL BRIP1
1.0 1.502945205479452 SELE NFKB1
1.0 1.497815699658703 MPZL1 FOXA1
1.0 1.492721088435374 STK38 CRK
1.0 1.4876610169491526 KNTC1 NSL1
1.0 1.4826351351351352 DAB2 PIP5K1B
1.0 1.4776430976430976 RALA AURKA
1.0 1.4726845637583894 PNPLA8 NFKB1
1.0 1.4677591973244148 SMARCE1 NBN
1.0 1.4628666666666666 KDM2A AURKA
1.0 1.4580066445182724 PDCD6IP CUL5
1.0 1.4531788079470198 TFAP2C PRKAA1
1.0 1.4483828382838284 PAM PBX1
1.0 1.4436184210526317 MIR30B PRKAA1
1.0 1.4388852459016395 E4GALT5 GALNT13
1.0 1.4341830065359478 SMEK2 BAZ2A
1.0 1.4295114006514658 BRIP1 FOXA1
1.0 1.4248701298701298 SLC6A15 SLC8A1
1.0 1.4202588996763754 NRF1 RPS6KA3
1.0 1.4156774193548387 MYH14 CTCF
1.0 1.4111254019292605 IER3 NFKB1
1.0 1.4066025641025641 NDST4 PRKCA
1.0 1.402108626198083 WWC2 TCF3
1.0 1.397643312101911 ANO3 MME
1.0 1.3932063492063493 RTEL1 NARFL
1.0 1.3887974683544304 EBAG9 NCOA3
1.0 1.384416403785489 CNTN6 FOXA1
1.0 1.380062893081761 ARID5B SETDB2
1.0 1.3757366771159876 CCNL1 FOXA1
1.0 1.3714375 PCDH9 CTNND2
1.0 1.367165109034268 PCDH7 CTNND2
1.0 1.362919254658385 STXBP4 PRKAA1
1.0 1.3586996904024768 HLTf NFKB1
1.0 1.3545061728395063 MIR578 TCF3
1.0 1.3503384615384615 PRPF6 GLI3
1.0 1.346196319018405 TNFSF13B FOXA1
1.0 1.342079510703364 NCOA6 SETDB2
1.0 1.337987804878049 CPSF3 RAE1
1.0 1.333920972644377 CPSF1 NUP188
1.0 1.3298787878787879 PPP2R2A KITLG
1.0 1.3258610271903324 PHLPP2 AURKA
1.0 1.321867469879518 MIR1297 FOXA1
1.0 1.317897897897898 HS3ST1 PIP5K1B
1.0 1.3139520958083832 CDH6 CTNND2
1.0 1.3100298507462687 CDH8 ERBB4
1.0 1.3061309523809523 CDH7 ERBB4
1.0 1.3022551928783384 CDH9 ERBB4
1.0 1.2984023668639053 APPBP2 CTCF
1.0 1.2945722713864307 NEDD4L FOXA1
1.0 1.290764705882353 COL8A1 TCF3
1.0 1.2869794721407626 NF1 CTCF
1.0 1.283216374269006 COL5A2 YES1
1.0 1.2794752186588922 NDOR1 NARFL
1.0 1.2757558139534884 INVS CUL3
1.0 1.2720579710144928 RAD51C AURKA
1.0 1.2683815028901735 SYT1 PPF1A2
1.0 1.2647262247838618 DZIP3 TBL1XR1
1.0 1.2610919540229886 TPD52 CLTC
1.0 1.2574785100286534 PLOD2 COL23A1
1.0 1.2538857142857143 KIFAP3 FOXA1
1.0 1.2503133903133903 RAPGEF2 PIK3CA
1.0 1.2467613636363637 UBR5 FOXA1

1.0 1.243229461756374 NEK7 FOXA1
1.0 1.2397175141242938 ADD1 PRKCA
1.0 1.236225352112676 SUV420H1 ARID2
1.0 1.232752808988764 SRSF11 RAE1
1.0 1.2292997198879552 ASAP1 FOXA1
1.0 1.2258659217877095 TIPRL TCF3
1.0 1.222451253481894 MACF1 NFKB1
1.0 1.219055555555556 STX16 FOXA1
1.0 1.2156786703601108 GYS1 AGL
1.0 1.2123204419889504 AEBP2 KAT7
1.0 1.2089807162534436 IKZF3 PRKCA
1.0 1.2056593406593408 TPD52L2 GLI3
1.0 1.2023561643835616 PTPN12 AURKA
1.0 1.199071038251366 EPHA3 PIK3CA
1.0 1.1958038147138965 CBLB PRKAA1
1.0 1.192554347826087 BMP7 PIP5K1B
1.0 1.1893224932249322 TCF12 ZNF652
1.0 1.1861081081081082 KAT6A AURKA
1.0 1.182911051212938 CCNT2 NOTCH1
1.0 1.1797311827956989 PHKB RYR3
1.0 1.1765683646112601 SEMA3E PIK3CB
1.0 1.1734224598930483 RAB6A ADAM17
1.0 1.1702933333333334 NR2F2 MED1
1.0 1.1671808510638297 ADAM10 PIP5K1B
1.0 1.1640848806366049 ADAM17 MALT1
1.0 1.161005291005291 PPFIA2 SYT1
1.0 1.1579419525065964 YWHAZ IL6ST
1.0 1.1548947368421052 SDCCAG8 ANAPC11
1.0 1.1518635170603675 ZBTB16 TP63
1.0 1.148848167539267 RB1CC1 PRKAA1
1.0 1.1458485639686684 MLLT4 PIP5K1B
1.0 1.1428645833333333 AGL GBE1
1.0 1.139896103896104 SI BCHE
1.0 1.1369430051813472 DKK1 NOTCH1
1.0 1.1340051679586565 CNGB3 SLC8A1
1.0 1.1310824742268042 GATA3 CAB39L
1.0 1.1281748071979434 DTL CORO1C
1.0 1.1252820512820514 STK4 PRKAA1
1.0 1.1224040920716112 AHR KITLG
1.0 1.1195408163265306 MAP4K3 PRKAA1
1.0 1.1166921119592876 LYST NFKB1
1.0 1.1138578680203046 SDCBP PIP5K1B
1.0 1.1110379746835444 GIPC1 CTCF
1.0 1.1082323232323232 ZMYND8 PLCB4
1.0 1.10544080604534 BCHE SI
1.0 1.1026633165829145 EWSR1 PRKCA
1.0 1.0998997493734337 TTC21B FOXA1
1.0 1.09715 RNPS1 NUP188
1.0 1.0944139650872817 MALT1 RIPK2
1.0 1.0916915422885571 WRN PRKCA
1.0 1.0889826302729528 CPNE3 OR2A5
1.0 1.0862871287128713 NUP188 GNB1
1.0 1.0836049382716049 CLTC PIP5K1B
1.0 1.080935960591133 LGR4 OR2A5
1.0 1.0782800982800984 PIGK PIGU
1.0 1.0756372549019608 ARHGAP5 PIP5K1B
1.0 1.0730073349633251 GALNTL6 GALNT13
1.0 1.070390243902439 CDK12 FOXA1
1.0 1.067785888077859 CALCRL PTH2R
1.0 1.0651941747572815 COL23A1 LEPREL1
1.0 1.0626150121065376 ARHGAP32 ITSN2
1.0 1.060048309178744 SUPT3H HSPD1
1.0 1.0574939759036144 TNFRSF21 TBL1XR1
1.0 1.054951923076923 ING3 SMARCE1
1.0 1.05242206235012 PPP1R12B ANAPC11
1.0 1.0499043062200957 PPP1R12A ANAPC11
1.0 1.0473985680190931 RGS18 FOXA1
1.0 1.044904761904762 RNF43 PRKCA
1.0 1.0424228028503564 JHDM1D TADA1
1.0 1.039952606635071 L2HGDH GPD2
1.0 1.0374940898345153 TRPC4AP SLC8A1
1.0 1.0350471698113208 BMPR1B PIP5K1B
1.0 1.0326117647058823 CSN2 ERBB4

D.2 CAPRI on Merged (Pre and Post) Dataset

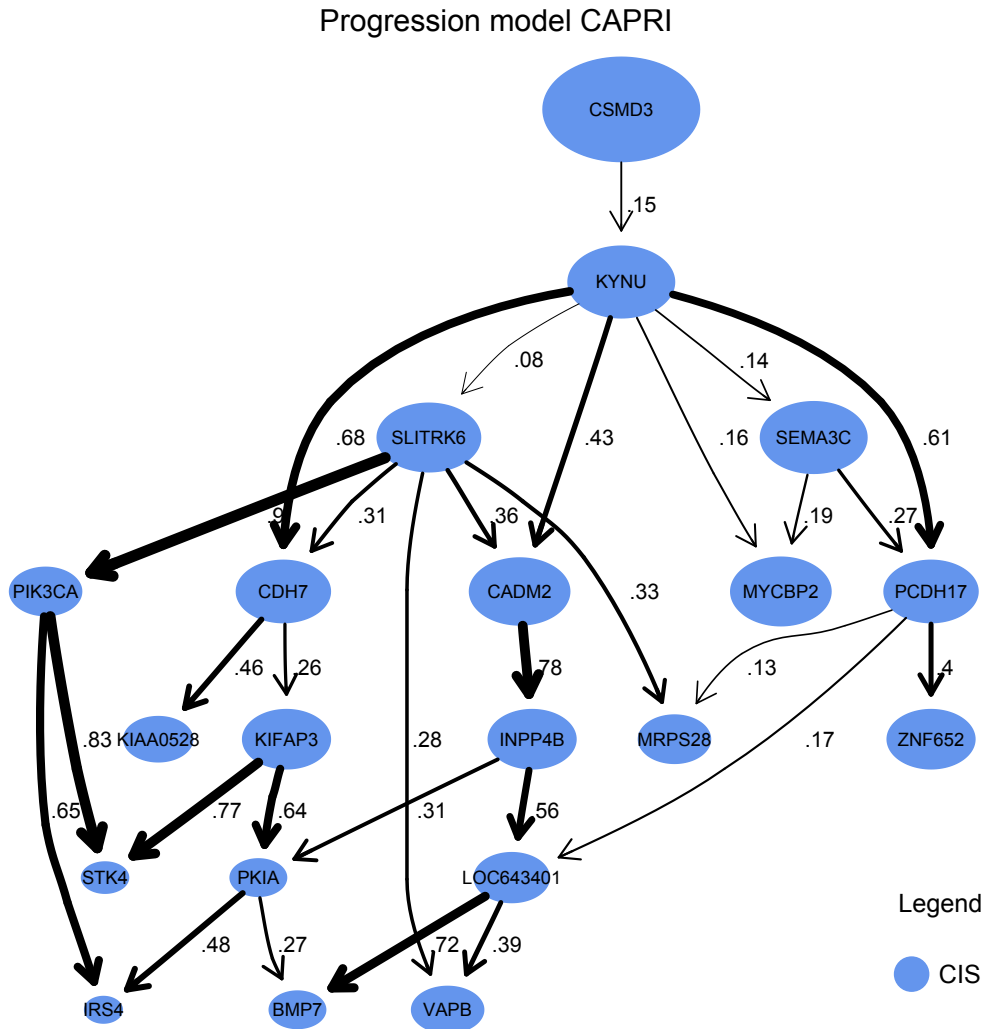


Figure D.1: CAPRI on merged Dataset (no distinction of pre and post treatment).

bt474-types.txt (comma-separated) file identify the content type and the color of the nodes (CIS), in this case a light blue.

CIS , cornflowerblue

For the complete model: bt474-events-complete.txt (comma-separated) file identify the events (CISs, genes) with their relative order.

PIK3CA , CIS , 1

CDK12 , CIS , 2

BCAS1 , CIS , 3

ITCH , CIS , 4
 NCOA3 , CIS , 5
 VMP1 , CIS , 6
 KIFAP3 , CIS , 7
 ARHGAP39 , CIS , 8
 PKIA , CIS , 9
 CADM2 , CIS , 10
 SLITRK6 , CIS , 11
 CSMD3 , CIS , 12
 PWRN2 , CIS , 13
 LOC643401 , CIS , 14
 DENND1B , CIS , 15
 LOC100131234 , CIS , 16
 ACACA , CIS , 17
 KIAA0528 , CIS , 18
 ZNF652 , CIS , 19
 MIPOL1 , CIS , 20

bt474-data-complete.txt (tab-separated) file contains the binary matrix, as described in Chapter 5. Each row is a CIS (gene) and each column is a sample.

```

1 0 1 1 0 1 0 0 0 1 1 1 0 0 0 0 1 1 0 0
1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 1 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
0 0 1 1 0 1 0 0 0 1 0 1 1 1 1 1 1 0 0 1 1
0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
0 1 1 0 0 1 1 0 1 1 1 1 1 0 0 0 0 1 0 1
0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
1 1 1 1 1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1
0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0
0 1 1 0 1 1 1 0 0 0 0 1 1 0 0 0 0 1 1 1
0 0 1 1 0 1 1 0 0 1 1 1 0 1 1 0 0 0 0 0
1 1 1 1 1 1 1 1 0 1 1 1 0 0 0 1 1 0 1 1
1 0 0 1 1 0 0 0 0 0 1 1 0 0 1 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
0 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 0 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
0 0 1 1 0 1 0 0 0 1 0 1 1 1 1 1 0 0 1 1

```

```

0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
0 1 1 0 0 1 1 0 1 1 1 1 1 0 0 0 0 1 0 1
0 1 0 1 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
0 1 1 0 1 1 1 0 0 0 0 1 1 0 0 1 0 1 1 1
1 1 1 1 1 1 1 1 0 1 1 1 0 1 1 1 1 0 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
0 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 0 1 1

```

The DAG obtained in output is in Figure D.1.

D.3 CAPRI before VISPA2

With the old version of VISPA, the CISs are slightly different. This means that the CISs retrieved are the same, but the binary matrix given as input to CAPRI is different, because some CISs are not present in the same sample and with the same *CIS Order*. This is due to the false positives bias of VISPA versus VISPA2, see precision and recall results in Chapter 3. The DAGs with the old version of the pipeline are in Figure D.2.

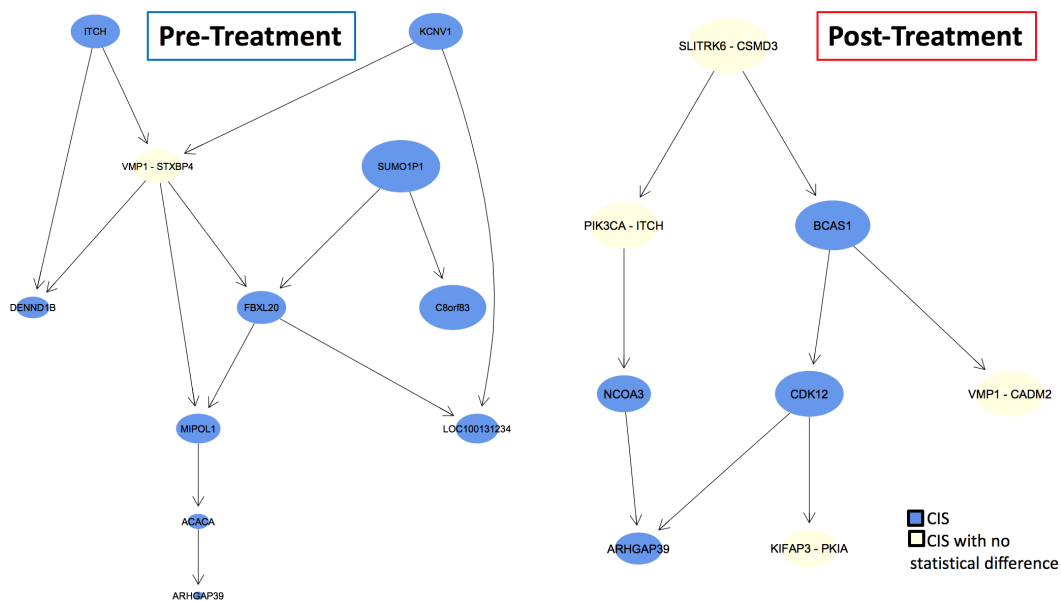


Figure D.2: CAPRI with the old version of VISPA, some CISs were false positives, LOC1001312234, ACACA and ARHGAP39 for the pre-treatment; SLITRK6, PKIA and KIFAP3 for the post-treatment.

” Bisogna fare della propria vita come si fa un’opera d’arte. Bisogna che la vita d’un uomo d’intelletto sia opera di lui. La superiorità vera è tutta qui.”

Gabriele D’Annunzio