Computational approaches for integrative cancer genomics.

Christian Pérez Llamas

TESI DOCTORAL UPF / 2015

DIRECTOR DE LA TESI

Dra. Núria López Bigas

DEPARTMENT OF EXPERIMENTAL AND HEALTH SCIENCES

Universitat Pompeu Fabra Barcelona

My wife had a severe complication while she was pregnant of our daughter, and they were on death's doorstep for quite some time. Modern medicine, and a lot of love, were the main reasons they survived.

I want to dedicate this work to all the people who contributed in some way:

- Cristina and Ariadna, for their strength, resilience and willing for live.

- Our family and friends, whose love and support were decisive for our entirety.

- To the staff of "Hospital General de Catalunya", who not only applied modern medicine successfully, but also for their emotional support.

- To those scientists who, beside the pressure for having to publish in prestigious journals, believe and work hard for really contributing to the advance of our society.

Acknowledgements

My wife, Cristina, who was always ready for listening to me in my worst moments, and whose support and encouragement helped me not to give up.

My mother Sari and my sister Noelia, whose devotion to help me is almost infinite.

My father, Jose Baltasar, who always dreamed to become an inventor and inspired me to join science.

My supervisor, Núria Lopez-Bigas, who always believed in me more than I did.

All the great people I had the privilege to work with, with special mention to Abel Gonzalez, Alba Jene, Carlota Rubio, David Tamborero, Gunes Gundem, Janet Pinero, Jordi Deu, Juanra Gonzalez, Khademul Islam, Loris Mularoni, Michael Schroeder, Sophia Derdak, and Xavier Rafael.

My friend, Roman Valls, whose conversations about science and technology would never end, and always show me something new to learn.

Stephen Breslin, who really cares about engineers' growth, and offered me some time in working hours to write the dissertation.

Abstract

Given the complexity and heterogeneity of cancer, the development of new high-throughput wide-genome technologies has open new possibilities for its study. Several projects around the globe are exploiting these technologies for generating unprecedented amount of data for cancer genomes. Its analysis, integration and exploration are still a key challenge in the field. In this dissertation, we first present Gitools, a tool for accessing databases in biology, analysing high-throughput data, and visualising multi-dimensional results with interactive heatmaps. Then, we show IntOGen, the methodology employed for collection and organization of the data, the methods used for its analysis, and how the results and analysis were made available to other researchers. Finally, we compare several methods for impact prediction of non-synonymous mutations, showing that new tools specifically designed for cancer outperform those traditionally used for general diseases, and also the need for using other sources of information for better prediction of cancer mutations.

Resum

Davant de la complexitat i heterogeneitat del cancer, el desenvolupament de noves tecnologies per l'estudi de genomes, ha obert noves posibilitats. Diversos projectes al voltant del mon les fan servir per generar quantitats de dades de genomes de cancer mai vistes abans. En aquest treball, primer presentem Gitools, una eina que permet obtenir dades de bases de dades en biologia, analitzar dades genomiques, i visualitzar els resultats multidimensionals mitjançant mapes de calor interactius. Després mostrem IntOGen, les metodologies per obtenir i organitzar les dades, els metodes per el seu analisi, i com es van possar a disposició d'altres investigadors. Finalment, comparem diversos metods de predicció de l'impacte de les mutacions no sinonimes, que ens mostra com nou metods desenvolupats per cancer funcionen millor que els utilitzats tradicionalment per enfermetats generals, aixis com la necesitat de recorrer a altres fonts d'informació per tenir millor prediccions per mutacions de cancer.

Table of contents

Table of figures

Index of Tables

PART I: INTRODUCTION AND OBJECTIVES

INTRODUCTION

Cancer is a very common and complex disease that affects both sexes worldwide. According to WHO's cancer fact sheet, 8.2 million people died in 2012, which comprises around one in eight deaths worldwide. Its understanding is of vital importance to be able to develop new and improved treatments that target its origins as specifically as possible.

There are several ways to address this complex topic, in this dissertation I will focus on the disease at the molecular level and the use of computational methodologies to enhance our understanding of this complex disease.

The genome

The genome of an organism represents the genetic material that determines its observable characteristics or traits (known as phenotype). Examples of such traits are the organism's morphology, development, biochemical or physiological properties, or behaviour. The molecule that carries this information is know as DNA and it is located in the nucleus of every cell. It encodes its information by chaining simpler units called nucleotides to form a sequence. A nucleotide is a molecule composed of a phosphate, a sugar, and a base, and depending on the base, there are four nucleotides available: Adenine (A), Guanine (G), Cytosine (C), or Thymine (T) (see Figure 1).

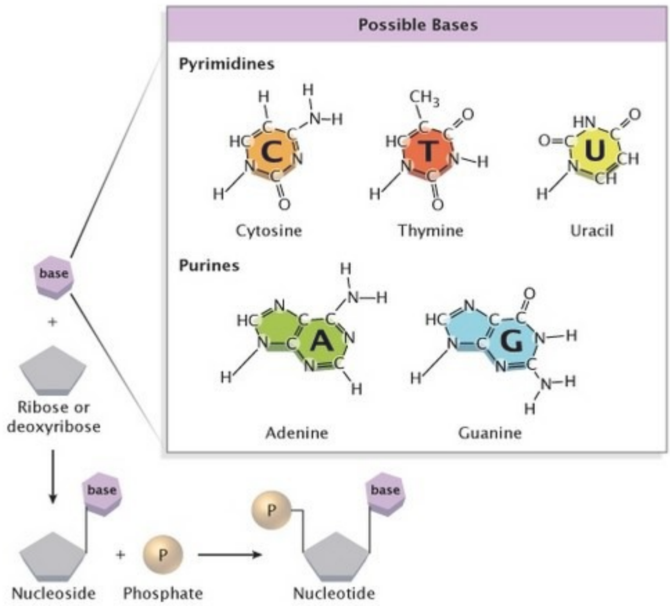

Figure 1: The structure of a nucleotide.

The DNA is composed of two of such sequences of nucleotides (strands) connected by hydrogen bonds in a conformation known as the double-helix (see Figure 2). 'A' bases, pair only with 'T' bases, and 'C' bases, pair only with 'G' bases (see Figure 3).

Some parts of the DNA sequence, known as genes, hold the information necessary to produce proteins, some other parts contain regulation information, and some others just structure. The parts of the DNA that encode for proteins are also known as coding regions of the genome, and the rest it is known as non-coding regions.

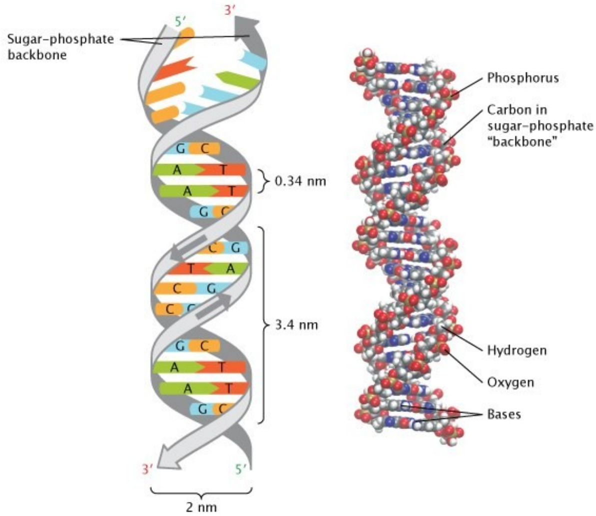

Figure 2: The double-helix of the DNA

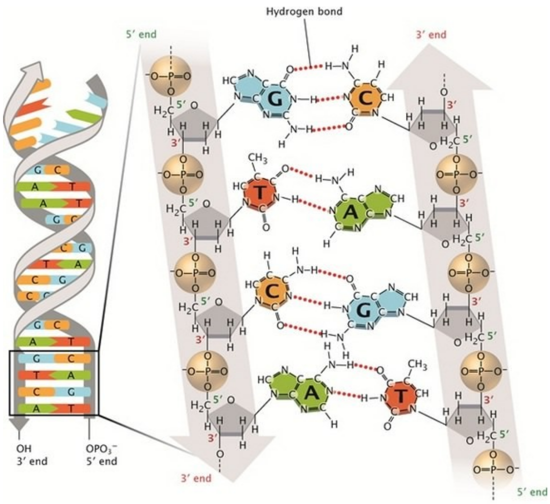

Figure 3: The base pairing of the DNA

The Central Dogma of Biology deals with how the information is transferred between different molecules and in what possible directions. In a simplified way, it distinguishes between DNA replication, when the information on one DNA molecule is replicated into a new DNA molecule, DNA transcription, when the information in a sequence of the DNA is transcribed into another molecule known as RNA, and translation, where the information in a specific type of RNA known as messenger RNA (mRNA) is translated into a sequence of amino-acids forming a protein.

DNA replication

As commented, it is the process by which a new copy of the DNA is generated, and it usually takes place during the cells division. The complex group of proteins involved in this process (i.e. the helicase, DNA polymerases or ligases) is known as replisome and perform the replica from one parent DNA strand into a complementary daughter strand in 3' to 5' direction.

This process is really complex, and accurate, to the point of including mechanisms to repair possible introduced mistakes, but as with many other processes in life, it can fail and introduce changes to the replica. Sometimes the changes are neutral and doesn't change the information encoded in the sequence, but when it does, and affects the encoding of proteins, the regulation of transcription, the mechanisms for repairing mistakes, or affects the structure, it can lead to diseases. Cancer is a particular case of diseases in which the accumulation of such changes lead to uncontrolled grow of the cells containing the mutated DNA. But this small margin for failures, can also be seen as the door for the evolution of organisms.

DNA transcription into RNA

Like DNA, RNA is a molecule that chains together a sequence of ribonucleotides (like a nucleotide but with a different kind of sugar). Each ribonucleotide can have one of the mentioned bases for the DNA with one exception, instead of Thymine (T) it uses a Uracil (U), which can also pair with the Adenine (A). Unlike DNA, RNA is single-stranded, but it can form many secondary structures by folding over itself and forming loops stabilized by hydrogen bonds between complementary bases (see Figure 4). Such secondary structure is critical for many of its functions. There are several types of RNA with different functions, transfer RNA (tRNA), ribosomal RNA (rRNA), or messenger RNA (mRNA) are some of them.

DNA transcription is the process of transferring genetic information from a portion of DNA into a new assembled single-stranded RNA molecule. In some cases, the final product is the RNA molecule itself, however, in many other cases, when the final product is a protein, the RNA, known as messenger (mRNA) is an intermediary that will be translated into a protein later.

The transcription can be divided in initiation, elongation, and termination. During the initiation an enzyme called RNA polymerase attaches to the DNA template at a specialized sequence called a promoter, usually with the intervention of other co-factors related to the regulation of the gene expression. During the elongation, the RNA polymerase keep reading from the DNA template and adding new complementary ribonucleotides to the growing RNA. Finally, the transcription terminates when the RNA polymerase reaches a terminal sequence and the transcript is released.

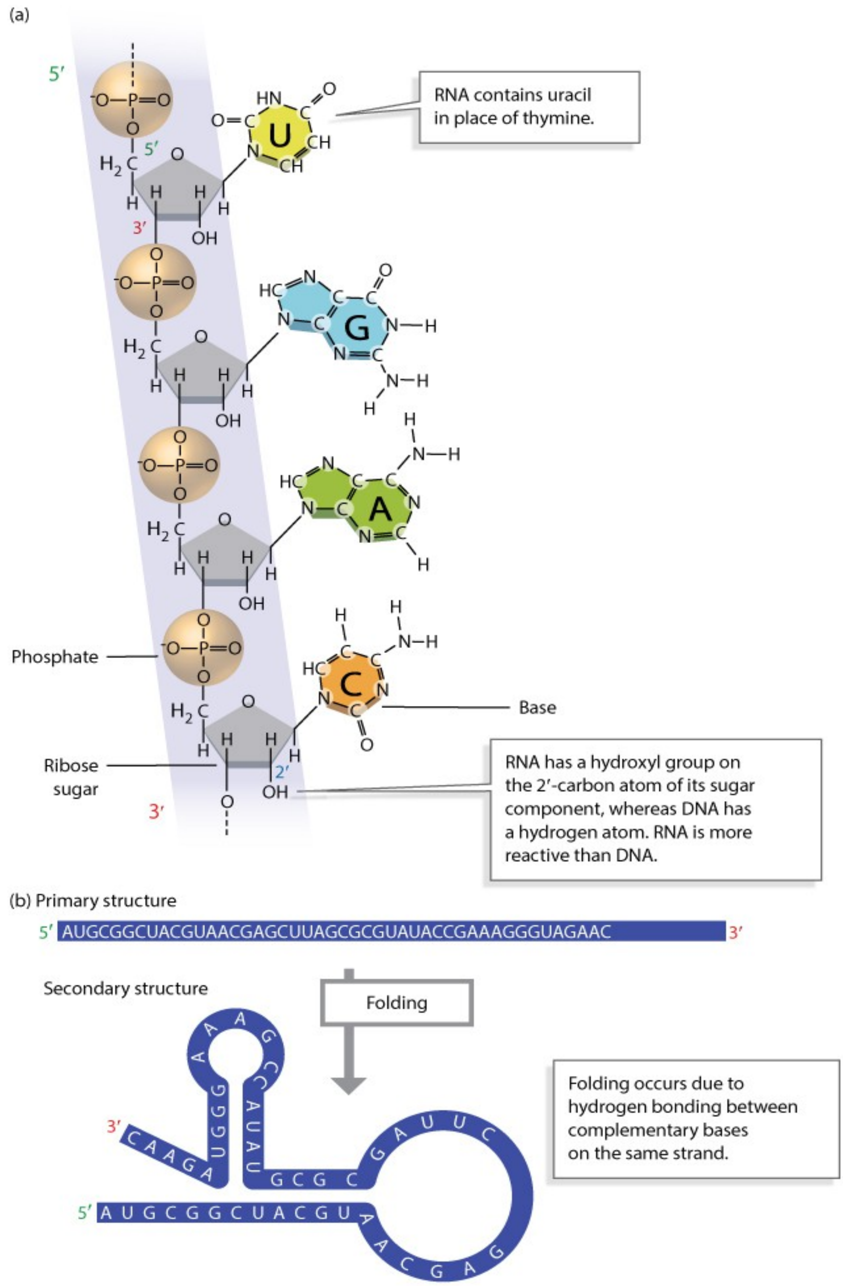

Figure 4: The primary and secondary structure of RNA.

mRNA translation into a protein

Proteins are composed of amino-acids in a sequence determined by such of the mRNA nucleotides. How the sequence of nucleotides is translated into a sequence of amino-acids is determined by the Genetic Code (see Figure 5). A Codon is a sequence of three nucleotides that encode for an amino-acid following the Genetic Code. With a sequence of three nucleotides with four possible bases more than 20 combinations exist, which allows for higher redundancy (several combinations map to the same amino-acid) and special instructions (such as the stop codon that marks the end of the translation).

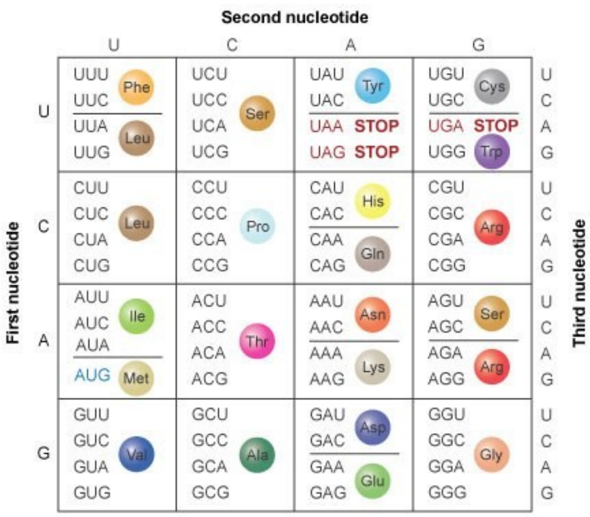

Figure 5: The genetic code that determines which amino-acid correspond to each RNA codon.

After the mRNA is transcribed it follows a series of maturation steps before the translation can start, such as the removal of the introns (interleaved fragments that doesn't encode for the sequence of amino-acids), or the transport of the matured mRNA to the cytoplasm in the case of eukaryotes where the protein assembly takes place. The translation starts by the binding of a ribosome (usually to the initial AUG codon), that will keep reading mRNA codons and binding tRNAs porting the specific amino-acid. Finally the translation terminates when a stop codon is reached. The polypeptide chain will keep folding while the translation takes place, and will follow a series of processing steps before the final protein product is ready to start its function.

The cancer genome

Cancer can also be seen as a group of diseases which common trait is the abnormal proliferation of cells that can even invade other tissues out of the originating one and metastasise other organs.

Nowadays cancer is considered to be an evolutionary process where cell populations suffering from several alterations in their genetic material over time, are naturally selected according to their capability to proliferate in their micro-environment.

We need to distinguish between somatic mutations, which represent alterations acquired through the lineage of cell divisions from the progenitor fertilized egg (see Figure 6), from germline mutations, which are acquired from parents.

The catalogue of somatic mutations is diverse, including substitutions of single bases, insertions or deletions of small or large fragments of DNA, rearrangements where parts of the DNA are moved across the genome, and copy number variations where several copies of a fragment or gene can appear, or the other way completely disappear (see Figure 7).

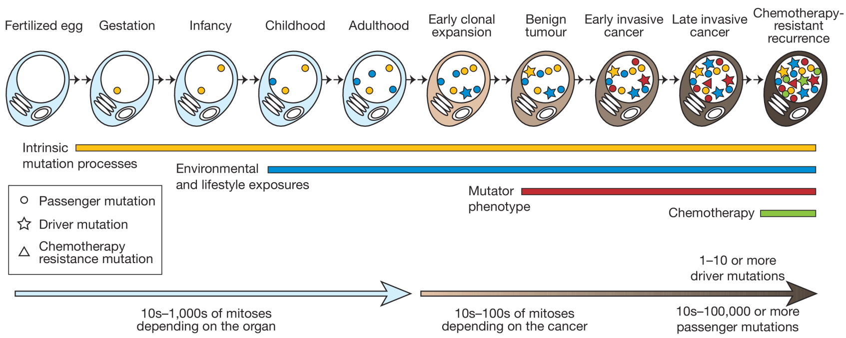

Figure 6: Somatic mutations acquired and selected over time and the processes that contribute.

Reproduced from (Stratton et al., 2009).

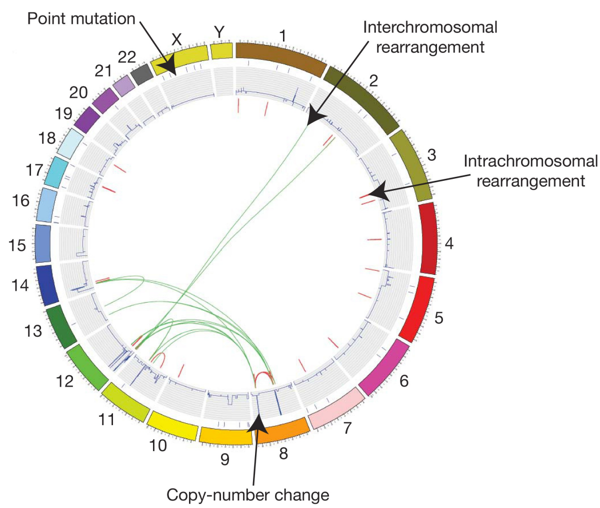

Figure 7: Representation of different types of somatic mutations in a cancer genome.

Reproduced from (Stratton, Campbell, & Futreal, 2009).

Another source of alterations in the DNA are virus such as human papilloma virus or hepatitis B, which insert their DNA into the human one. Somatic mutations applies also, not to the cell DNA, but to the thousands of small mitochondrial genomes present in the cell.

When talking about alterations, not only the DNA can be affected, nor it is the only responsible for the origination of cancer, but we also need to talk about the epigenome, which modifications through changes in the methylation status of the histones that sustain the DNA, lead to changes in chromatin structure and gene expression.

Somatic mutations can be classified, according to the consequences for the development of cancer, in *driver* or *passenger*. Driver mutations confer growth advantage and have been positively selected during the evolution of the cancer, and passenger mutations doesn't confer growth advantage but were present in the cancer cell when it acquired one of its drivers. In order to identify which are the genes involved in cancer, it is key to be able to distinguish between driver and passenger mutations.

Systematic study of cancer genomes

Since the completion of the human genome sequence around 2001 (Lander et al., 2001; Venter et al., 2001), several technologies emerged for the exploration of the genome at a large scale. Starting by micro-arrays, referred to as high-throughput technology because of the ability to explore different levels of genetic information with higher resolution, accuracy and reduced cost. And later the development of second generation of massive parallel sequencing technologies which during the recent years have been replacing micro-arrays in several cases given its fast development, progressive reduction in cost and increase in resolution.

DNA micro-arrays

DNA micro-arrays consists on a surface containing many short complementary DNA fragments attached to it. The different small DNA sequences conform the probes of the micro-array, and usually, but not necessarily, correspond with small sections of genes. The core principle of micro-arrays is the hybridization between two DNA strands by forming hydrogen bonds between complementary nucleotide base pairs. The target DNA fragments under study can be labeled with fluorescent molecules that can be later detected by a scanner. The intensity of the signal in each spot is related to the amount of hybridized DNA.

Micro-arrays can be used to measure changes in gene expression, to detect single nucleotide polymorphisms (SNPs), or to genotype different regions of the genome. There are multiple applications such as gene expression profiling, comparative genomic hybridization, chromatin immunoprecipitation (ChIP), or SNP detection. Of relevance for this work are comparative genome hybridization and expression profiling.

Comparative Genome Hybridization (CGH) is a technique that allows to find unbalanced chromosomal abnormalities such as gain and loss of genetic material in the whole genome by comparing a normal sample with a tumour one which are differentially labeled with fluorescence and competitively hybridized to metaphase chromosomes. The intensity of the fluorescence signal can be plotted across each chromosome and show the Copy Number Variations (CNV). Figure 8 shows a diagram of how CGH is done using micro-array technology. Compared to CGH, which is limited to alterations of approximately 5 to 10 Megabases, array based CGH (aCGH) increase resolution up to 100 kilobases (de Ravel, Devriendt, Fryns, & Vermeesch, 2007).

Based on the same principles, micro-arrays can also be used for gene expression profiling. The mRNA is converted into cDNA and labeled with different fluorescent colour for the normal and cancer samples. Then, the two cDNAs are hybridized into the same micro-array. As a result of the scanning, probes will show different colours proportional to the amount of mRNA with complementary sequence present in the samples analysed (see Figure 9).

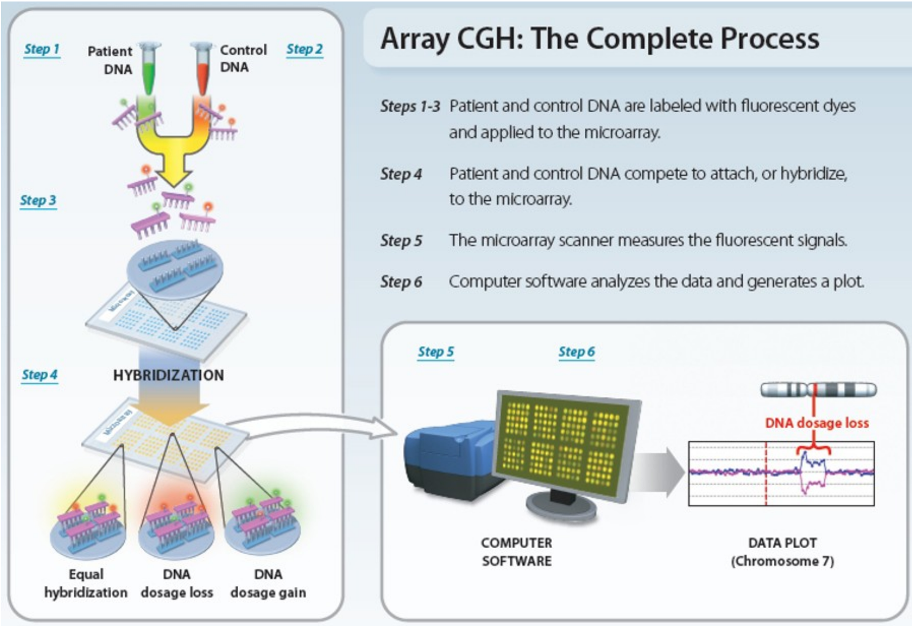

Figure 8: Diagram of the microarray-based comparative genomic hybridization (aCGH) process.

Reproduced from (Theisen, 2008)

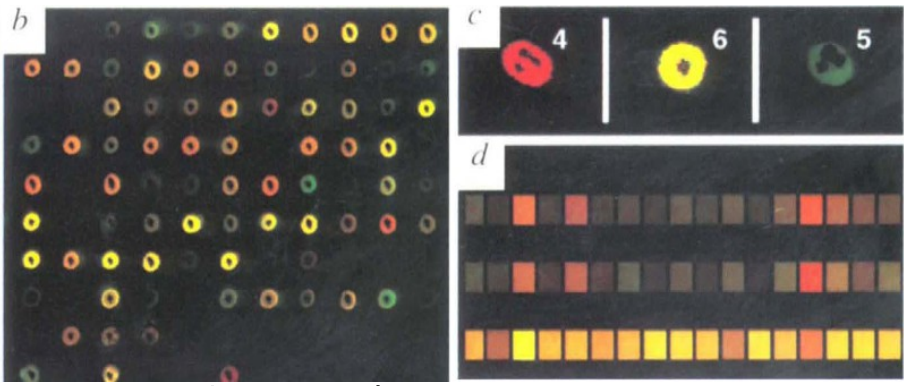

Figure 9: Scanned micro-array for gene expression, where normal sample was labeled with orange-red (Cy5) and tumour sample with green (Cy3).

Modified from (DeRisi et al., 1996)

Raw results from the device needs to be preprocessed to deal with systematic differences between genes or arrays (normalization), for example modifying the raw intensity values in order to compensate for the different dye efficiency in two channel microarray experiments using Cy3 (green) and Cy5 (red), and background corrected to adjust for non-specific hybridization, for example hybridization of fragments that doesn't match perfectly with the probe. After preprocessing, relative gene expression between the normal and tumour samples is quantified by calculating the ratio between the intensities emitted by each label in the spots corresponding to each gene. Usually the ratios are transformed by calculating the base 2 logarithm of the quotient between the tumour intensity respect to the normal one, as it is easier to map $\log_2$ ratios to fold changes than with raw ratios, and improves the characteristics of the data distribution and allows the use of classical parametric statistics for analysis (Tarca, Romero, & Draghici, 2006). The resulting datasets can be used for further analysis, such as the hierarchical clustering depicted in Figure 10.

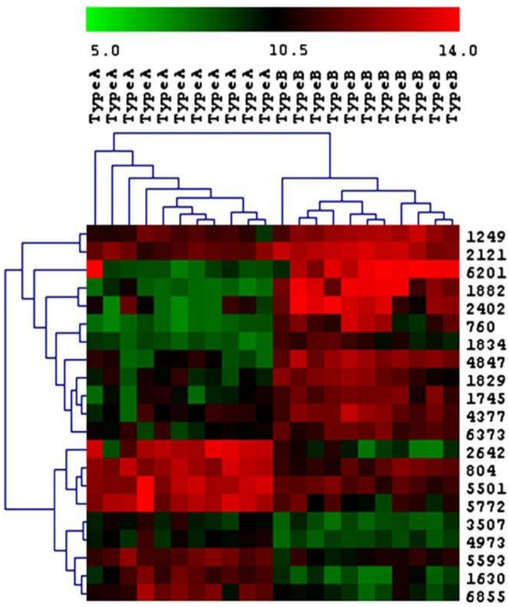

Figure 10: Hierarchical clustering of a gene expression dataset.

Next generation DNA sequencing

DNA sequencing consists on determining the sequence of nucleotides in a DNA strand, and the first technology to became mainstream was known as Sanger sequencing, as it was initially developed by Frederick Sanger in 1977, and later refined with the use of capillary electrophoresis by Applied Biosystems. Those first generation sequencing technologies were key for the completion of the Human Genome Project in 2001, which stimulated the development of next generation sequencing technologies (NGS) (L. Liu et al., 2012). The main aspects of NGS are their ability to sequence a massive amount of fragments in parallel, high-throughput, and reduced cost (see Figure 11), to the point of being really close to the $1000 cost objective per whole genome sequencing stablished by the US government programme (Check Hayden, 2014; Hayden, 2014).

Inside NGS we need to distinguish between second and third generation technologies. The second generation started in 2005 when the 454 sequencer was developed by Life Sciences (Roche), based on detecting the release of a pyrophosphate each time a new nucleotide is attached to the sequence, this principle was referred as sequencing by synthesis (SBS). Other technologies appeared soon from other companies, such as SOLID, depending on the sequential ligation of oligonucleotide probes, and Solexa (purchased by Illumina), adopting the SBS principle but based on bridge enzymatic amplification instead of requiring PCR amplification as is the case with the other technologies.

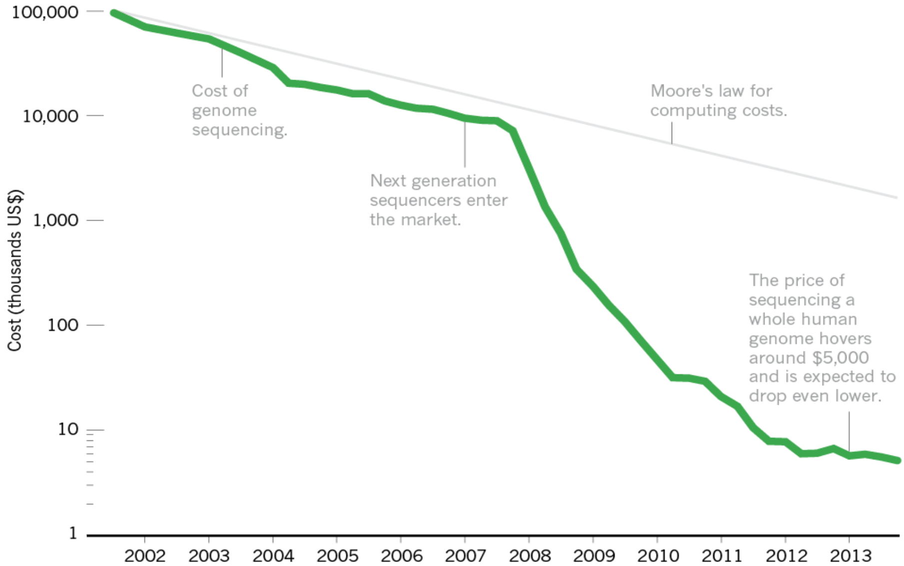

Figure 11: Evolution of the cost for sequencing a human genome.

In the first few years after the end of the Human Genome Project, the cost of Genome Sequencing roughly followed Moor's law, which predicts exponential declines in computing costs. After 2007, sequencing costs dropped precipitously.

Second generation technologies continue improving their read lengths and accuracy over time, but a third generation of technologies has been in development during the recent years. Most of them are based on the principle of single DNA molecule sequencing. Some examples are SMBT from Pacific Biosciences, which implements a fluorescence detection system that directly detects each nucleotide previously phosphor-linked with distinct colors as they are synthesized without the need of amplification, or Oxford nanopore sequencing, that relies on the conversion of the electrical signal of the nucleotides as they pass through a nanopore. A distinct approach is the used by Ion Torrent semiconductor sequencing, based on the release of hydrogen ions as a byproduct of nucleotide chain elongation and the detection of pH changes by an ion sensor during DNA synthesis.

Several applications for NGS exist depending on the input material from cancer samples (see Figure 12): whole-genome (WGS), whole-exome (WES) or whole-transcriptome sequencing (Tuna & Amos, 2013). WGS allows to identify the full range of somatic genome alterations, providing information on all 6 billion bases compared to the 5 million variants available with micro-arrays. To distinguish somatic mutations from inherited changes in cancer, matched normal samples have to be used, and has to be compared to the reference genome. WES is a cost-effective, high coverage approach to detecting mutations in known coding genes across the entire genome, and a very good diagnostic tool to detect known mutations or discover new ones in large sample cohorts in the cancer genome. The shortcoming is that, unlike WGS, it can not detect structural and non-coding variants, which have high susceptibility to be associated to cancer according to genome-wide studies (Manolio et al., 2009). Finally, transcriptome sequencing can be used for different kind of studies involving RNA material (requiring cDNA synthesis), some examples are characterization of transcripts in a given tissue and/or condition, and study of gene transcription and RNA processing during tumorigenesis. It is not limited to known genes and can be used to detect novel transcripts, alternative splice forms, and non human transcripts (microbiomes).

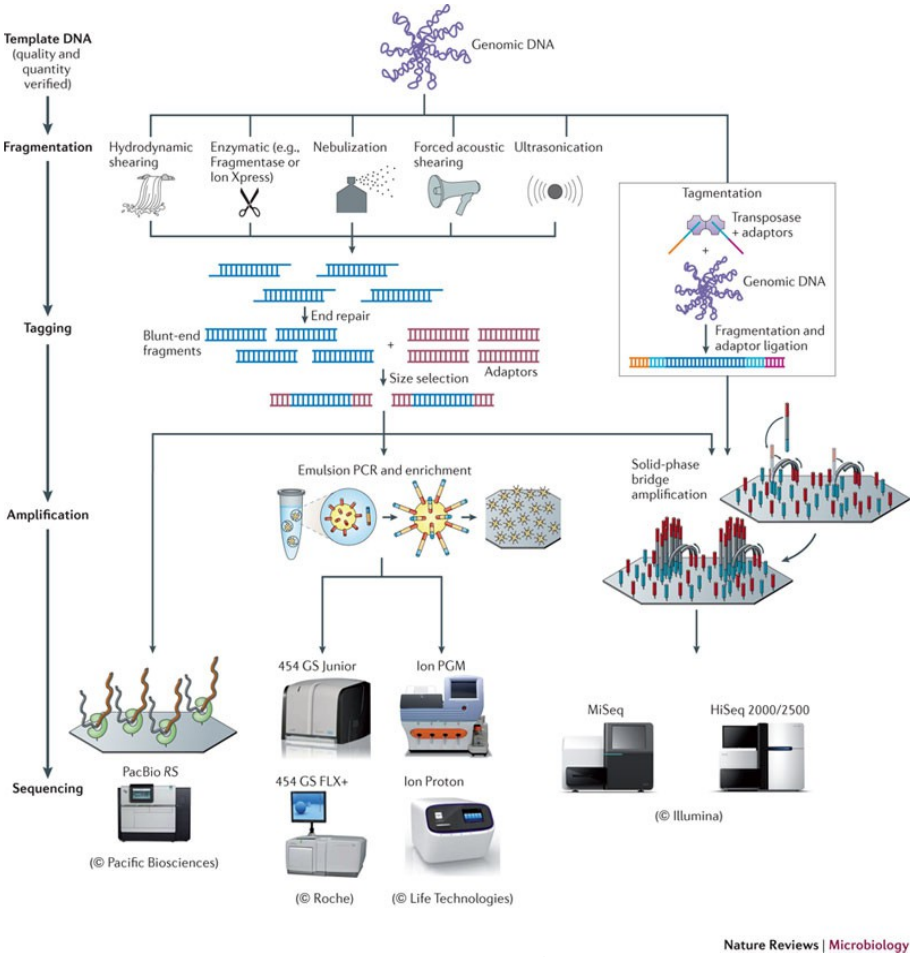

Figure 12: Next generation sequencing platforms and procedures.
Reproduced from (Loman et al., 2012).

Integration of experimental data

Integration of experimental data involves collecting and analysing data from several sources and/or platforms. Such external data may be generated by different researchers, using different technological platforms, at different points in time, which makes the data very heterogenous. To be able to perform integrative analysis over the whole set of data, it has to be normalized, annotated, and organized conveniently. But Biology knowledge is really complex, and to make easier for researchers to find and integrate information, we need of formal ways to name, define and interrelate the entities that exist for the domains of study. This is where ontologies play an important role in representing this knowledge by defining concepts and their relationships (Bard & Rhee, 2004). Currently there are several ontologies widely used, one of them is the Gene Ontology (GO) (Ashburner, Ball, Blake, Botstein, Butler, Cherry, Davis, Dolinski, Dwight, Eppig, Harris, et al., 2000), which aims at formalizing the knowledge about biological processes, molecular functions, and cell components. One important feature of this project is that GO terms are linked with gene products from many experimental organisms, so the users can explore the available knowledge for a given protein, or the other way around, from a given concept (i.e the cell cycle) it is possible to derive the proteins that are involved. Another project of relevance for this field is the Sequence Ontology (SO) (Eilbeck et al., 2005), which goal is to standardise the terms and relations to describe genomic annotations (see Figure 13 for an example), facilitating data exchange and comparative analysis of sequence annotations. Both, GO and SO, are part of the Open Biomedical Ontologies (OBO) project (Smith et al., 2007).

Finishing with examples of important projects that aim at formalizing the knowledge in the context of genomics and cancer, we need to mention the International Classification of Diseases for Oncology (ICD-O) (“WHO | International Classification of Diseases for Oncology, 3rd Edition (ICD-O-3),” n.d.), which classifies cancer diseases according to two axes that describe the tumour: the topography, which describes the anatomical site of origin (or organ system) of the tumour, and the morphology, which describes the cell type (or histology) of the tumour, together with the behaviour (malignant or benign). Ontologies also exist to model the design and organization of experiments and their results, some examples are the MAGE-OM (Spellman et al., 2002) focused on micro-array based experiments, and FUGE (Jones et al., 2007) focused on functional genomics.

Nowadays there are many biological databases with genomic data, one of the major initiatives is the NCBI's GenBank (Benson et al., 2013), that contains raw genomic sequences submitted by parties around the world to the extent of containing hundred of billions of nucleotides and keep growing exponentially. Similar initiatives exist from the EMBL in Europe (http://www.ebi.ac.uk/) and the DDBJ in Japan (http://www.ddbj.nig.ac.jp/), all three coordinated under the International Nucleotide Sequence Database Collaboration (INSDC) (http://www.insdc.org/). But because of the grow of those databases, it is very easy to find incomplete or inaccurate records as well as redundant information (Lathe, W., Williams, J., Mangan, M.

& Karolchik, 2008). RefSeq (Pruitt et al., 2014) was born to provide a scientist-curated non-redundant set of biological sequences. Given the open structure and wide scope of those databases, more specific, richer in contents, and more structured databases exist, being the UCSC Genome Browser (Sanborn et al., 2011) and the EBI's Ensembl (Flicek et al., 2014) database some of the most successful ones. They also offer advanced search interfaces, the Table Browser (Karolchik et al., 2004) for the UCSC Genome browser and Biomart (Smedley et al., 2015) for Ensembl, that can be used for performing complex queries and download information for further downstream analysis and predictions. Moreover, the Biomart portal, is not only used for Ensembl data, but for many other biological databases federated by different institutions across the globe.

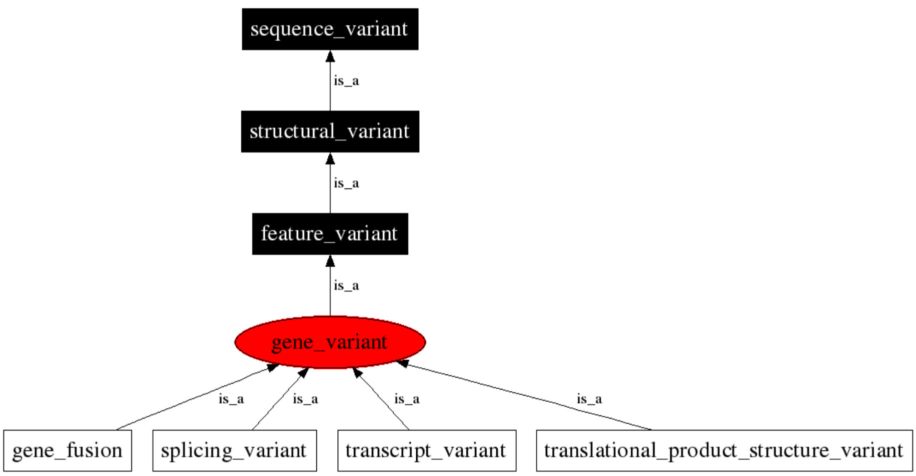

Figure 13: Example of the Sequence Ontology for the gene_variant term.

The Sequence Ontology term representing a gene variant (in red) and its relations to other terms (black for more generic terms, and white for the more specific ones). Extracted from MISO, the Sequencing Ontology web browser.

There are also specific databases or repositories for experimental data generated with high-throughput technologies. Two well known sources are ArrayExpress (Parkinson et al., 2007) and Gene Expression Omnibus (GEO) (Edgar, Domrachev, & Lash, 2002). ArrayExpress is a database for functional genomics data that consists of two parts, the Repository, supporting MIAME compliant (Brazma et al., 2001) micro-array data or MINSEQE compliant sequencing data (raw sequencing data is submitted to the European Nucleotide Archive (Cochrane et al., 2009)), and the Data Warehouse, with expression profiles selected from the repository and consistently re-annotated. GEO is a repository for high-throughput gene expression data and genomic hybridization experiments, not intended at replacing in-house databases but to complement them by acting as a central distribution hub.

Thanks to the recent advances in sequencing and computer technologies it is possible to sequence and analyse whole genomes and explore their alterations comprehensively. Two large scale projects, the International Cancer Genome Consortium (ICGC) (ICGC, 2010) as an international consortium, and The Cancer Genome Atlas (TCGA) (Omberg et al., 2013) supported by US National Institutes of Health, have been analysing cancer genomes for some years and making the data available through their respective data portals. Both include protein expression, copy number variation, somatic mutations, mRNA expression, DNA methylation, miRNA, and clinical data for several types of cancer. TCGA began in 2006 focused in three projects, glioblastoma multiforme, serous cystadenocarcinoma of the ovary, and lung squamous carcinoma, but has expanded during the later years covering other types of cancer. ICGC was born in 2007 with the following goals: to coordinate the generation of comprehensive catalogues of genomic abnormalities in tumours in 50 different cancer types and/or subtypes that are of clinical and societal importance across the globe, to ensure high quality, to make the data available to the entire research community as rapidly as possible, to coordinate research efforts among participants, and to support the dissemination of knowledge and standards to facilitate data sharing and integration.

Before those large initiatives existed, The Catalog Of Somatic Mutations In Cancer (COSMIC) (S. A. Forbes et al., 2009) was the largest public resource for information on somatically acquired mutations in human cancer, with data gathered from two sources: publications in the scientific literature for genes present in the Cancer Gene Census (CGC) (Futreal et al., 2004), and the genome-wide screens from the Cancer Genome Project (CGP) at the Sanger Institute in UK.

Computational cancer genomics

The biology of the genome is very complex, and trying to address this complexity directly would be unfeasible. Science use models to narrow the scope of study, focus on a specific part of the reality, so we can quantify and understand better our observations. Statistics provide us with tools for collecting these observations appropriately, and analysing them so we can explain and interpret the modelled reality quantitatively. But even when simplifying the reality, often we need to handle massive amounts of observations or data, and perform millions of computations in a reasonable period of time to build those models, which wouldn't be possible without the existence of computers. The main focus of Computer Science is the study of these computers (hardware and infrastructures) and their use for complex processes (algorithms and data structures). Software encodes those methodologies and processes in a way that computers can understand and execute, and Software Engineering is the discipline that cares about the systematic application of scientific and technological knowledge, methods, and experience to its design, implementation, testing, and documentation (“ISO/IEC/IEEE 24765:2010(E),” 2010).

Computational genomics focuses on understanding the human genome, and the principles of how DNA controls the biology of species at the molecular level by merging the knowledge and methodologies from several disciplines such as Biology, Statistics, Computer Science, and Software Engineering, and has become one of the most important means to biological discovery, specially given the increase in availability of massive biological datasets.

Identification of cancer drivers

One of the main challenges for computational cancer genomics is the identification of the genes that drive the tumorigenesis, as it would help to understand the mechanisms for the tumour formation and evolution, as well as open new paths for novel therapeutical solutions. Thanks to the advances in high-throughput technologies we have the tools to view which are the mutations present in the tumour cells, which can range from tens to thousands. But the difficulty comes from the need to distinguish the few of these mutations that are drivers, and thus conferring the selective grow advantage, from the ones that are sporadic passengers, most likely because of the genomic instability.

The first approach towards that aim, is to identify the functional impact of the cancer mutations. First, for each of the mutations, we need to identify the protein that overlaps with the mutated genomic region, and how affects to its sequence of amino-acids, so we can assess its consequences. Two well known tools for this identification are the Variant Effect Predictor (VEP) (McLaren et al., 2010), and snpEff (Cingolani et al., 2012). Given the redundancy of the genetic code, some mutations doesn't affect to the translation of the affected codon, and thus the sequence of amino-acids remains the same as for the reference protein. This is a synonymous variant, and represents the milder of the possible impacts to a protein's function. The opposite side is represented by the mutations that either truncate the translation of the protein by introducing a stop codon, or introduces changes to the frame that determines the triplets that will form the codons to be translated. These variants will most probably result in an inactivation of the protein and/or its function. Non-synonymous variants, represent those mutations that affect the translated amino-acid by just changing the corresponding codon, and are the main subject of a series of tools designed to predict their impact to the protein's function (see Table 1).

More advanced approaches are based on the identification of genes that exhibit signals of positive selection across a cohort of tumour samples (see Figure 14).

Some methods such as MuSic (Dees et al., 2012) and MutSigCV (Lawrence et al., 2013), are based on the recurrence of the mutations and identify genes that are mutated more frequently than the expected from the background mutation rate. Those methods have the shortcoming of having difficulties to identify driver genes mutated at a very low frequency. A method called OncodriveFM (Gonzalez-Perez & Lopez-Bigas, 2012) is based on identifying the bias towards the accumulation of functional mutations by testing the significance of the high rate non-silent mutations compared to the silent ones. Its advantage is the independence from the background mutation rate, but the selection of the method to assess if a mutation is functional or not can affect completely its performance. And OncodriveCLUST (Tamborero, Gonzalez-Perez, & Lopez-Bigas, 2013) is a method that exploits the fact that, whereas inactivating mutations are usually distributed along the sequence of the protein, mutations leading to gain of function tend to accumulate at particular residues or domains. Finally, ActiveDriver (Reimand, Wagih, & Bader, 2013) is a method based on the over-representation of mutations in specific functional residues, such as phosphorylation sites.

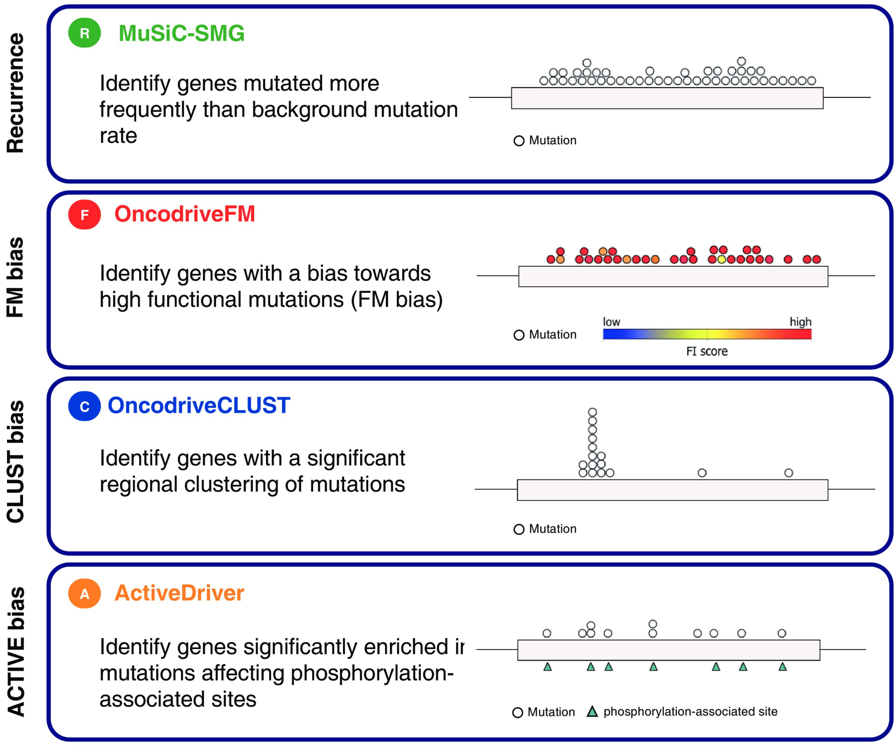

Figure 14: Signals of positive selection used to identify driver genes.

Method	Description
SIFT	Builds a MSA of similar proteins according to a database defined by the user and calculates normalized probabilities for all possible substitutions at all positions of the alignment. Based on these probabilities, SIFT classifies observed substitutions as likely neutral or deleterious.
PolyPhen 2	Naïve Bayes classifier trained from two data sets that contain both deleterious and neutral amino acid changes. Eight sequence-based and three structure-based predictive features, most of them involving comparison of a given property of the wild-type amino acid and its mutated counterpart are the properties used to build the classifier.
Mutation Assessor	A prediction of the functional impact of nsSNVs is based on the assessment of evolutionary conservation of amino acid residues. It exploits the evolutionary conservation in protein subfamilies, which are determined by clustering MSAs of homologous sequences on the background of conservation of overall function.
Condel	Condel (Consensus deleteriousness score) is an approach to combine the functional impact scores of nsSNVs. It uses values extracted from the complementary cumulative distributions of the scores produced by individual tools on a dataset of deleterious and neutral nsSNVs as weights to combine them.

Method	Description
FATHMM	Predicts the functional effects of cancer somatic mutations combining sequence conservation with hidden Markov models representing the alignment of homologous sequences and conserved protein domains with cancer "pathogenicity weights" representing the tolerance of the corresponding model to cancer mutations.
CHASM	A random forest classifier is trained on a curated set of driver mutations derived from COSMIC and randomly simulated passenger mutations. It uses eighty-six diverse features (available at SNVBox database), including physio-chemical properties of amino acid residues, scores derived from MSAs of protein or DNA, region-based amino acid sequence composition, predicted properties of local protein structure and annotations from the UniProtKB feature tables.
transFIC	transFIC (for transformed functional impact scores for cancer) takes the Functional Impact Score produced by any method aimed at evaluating the impact of a mutation on the functionality of a protein and transforms it, taking into account the baseline tolerance of similar proteins to functional impacting variants. The transformation can be interpreted as an adjustment for the impact of the somatic variant on cell operation.

Table 1: Functional impact predictors for non-synonymous variants.

OBJECTIVES

1. Let biologists without advanced knowledge in bioinformatics, access to specialized databases in biology, to analyse data generated by high-throughput technologies and to visualise the results according to the nature and dimensions of this kind of data.

2. Integrate and analyse genomic data to improve the understanding of the processes and alterations that make cells to become cancerous, opening new fronts for possible cancer treatments in the future.

    - Develop a series/system of analytical processes for integrating high-throughput oncogenomics data for the identification of genes or groups of genes involved in cancer

    - Apply these techniques to oncogenic data available in the literature and from international consortia, and make the results available for browsing by the wider scientific community

    - Make these processes available to the wider scientific community so that other researchers can use them for their own analyses and compare their results to those described above

3. Develop a benchmarking system and propose a series of datasets for comparing the performance of functional impact assessment algorithms, driver mutations identification tools and general purpose substitution scores in the context of cancer variation data.

PART II: RESULTS

1. GITOOLS

Perez-Llamas C, Lopez-Bigas N (2011) Gitools: analysis and visualisation of genomic data using interactive heat-maps. PLoS One 6(5): e19541. DOI: 10.1371/journal.pone.0019541

2. INTOGEN

There are two IntOGen projects I contributed to, IntOGen Arrays (Gundem et al., 2010) and IntOGen Mutations (Gonzalez-Perez et al., 2013). My main contribution to the IntOGen projects and papers is related to the management of genomic data, its analysis, and making the results available for internal and external researchers. The management of data involves assistance for the curation of the data gathered from several heterogeneous sources, and putting the tools and processes in place so it can be annotated and properly organized in a homogeneous way for the analysis phase. The analysis involves many interrelated steps that require coordination and flexible configuration to adapt the execution to different needs while performing the research. Furthermore, given the amount of data that has to be analysed, it is very important to allow the distribution of the work across several processors or even machines. Once the results are ready from the analysis, they have to be available to the researches, both to internal and external organizations, which requires extra care on how it is organized and exposed.

In the case of the IntOGen Mutations project there was also the requirement to allow external researchers to analyse their own data using the same methodology we were using, which required to set up an interface to submit data, monitor the progress, organize the results by user and facilitate its visualization.

Following there are the details about the analysis workflows that I developed for each of the IntOGen projects.

2.1. IntOGen Arrays

a) Organization of the data

The data required for this analysis was generated by several independent researchers around the world, and collected from several sources, so it required some pre-processing and homogenization. I developed a simple data model to organize and annotate it based on the core concepts of other existing models such as MAGE-OM (Spellman et al., 2002) and FUGE (Jones et al., 2007).

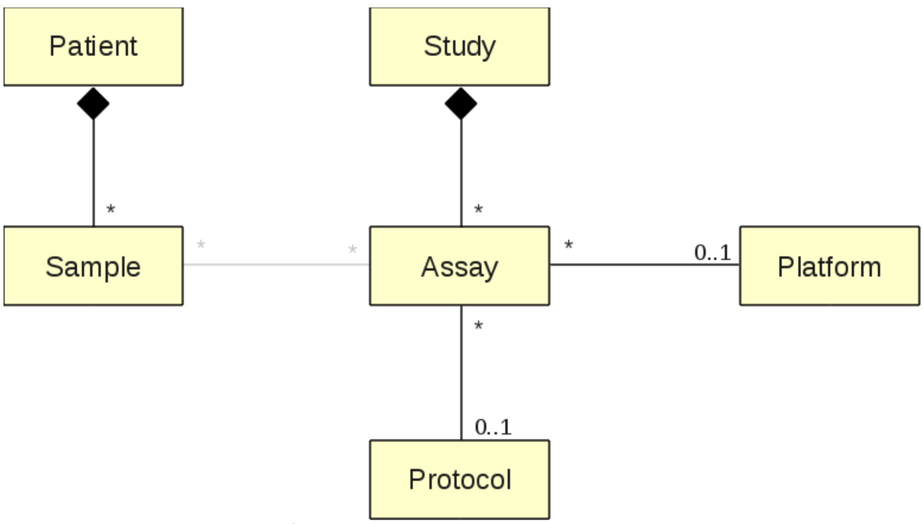

Figure 15: Data model for IntOGen Arrays source data

The main sources for the data are:

- GEO (Edgar et al., 2002): For transcriptomic alterations.
- ArrayExpress (Parkinson et al., 2007): For transcriptomic alterations.
- Progenetix (Baudis & Cleary, 2001): For copy number alterations.
- The Cancer Genome Atlas (Omberg et al., 2013): For transcriptomic and copy number alterations.
- COSMIC (Simon A Forbes et al., 2010): For mutations

The selection of experiments was based on the data to be public, the experiments to compare normal and cancer tissues, and having at least 20 samples. The collected data was manually curated from the publications or the descriptions available in the source and annotated using internally developed ontologies, the Gene Ontology (GO) (Ashburner, Ball, Blake, Botstein, Butler, Cherry, Davis, Dolinski, Dwight, Eppig, & others, 2000) and the International Classification of Diseases for Oncology (ICD-O) ("WHO | International Classification of Diseases for Oncology, 3rd Edition (ICD-O-3)," n.d.). In total there were more than 800 studies, 25000 samples and 150 different tumour types.

b) Workflow organization

The workflow can be divide into these main parts or sub-workflows:

- Retrieval of external data
- Transcriptomic alterations
- Copy number alterations

- Biomart database generation
- Web browser data generation

The source code can be found here https://github.com/chris-zen/phd-thesis/blob/master/chapter2/intogen-arrays.

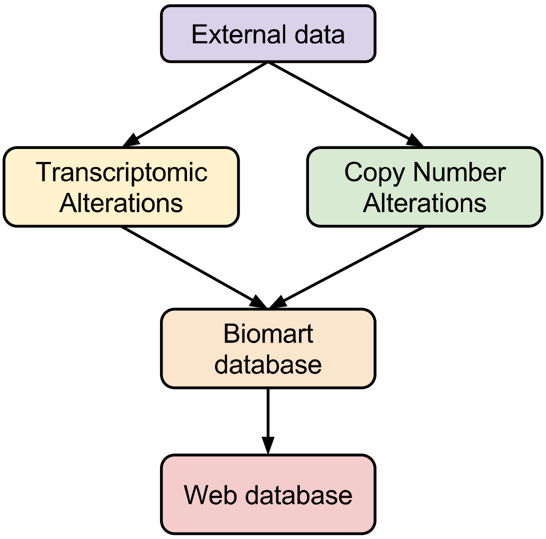

Figure 16: Workflow for IntOGen Arrays

The first step consists in retrieving data annotations from external databases required for the analysis - mainly from Biomart (Smedley et al., 2009). Then the transcriptomic and the copy number alterations processes can start. Finally the results are saved into a database required to expose the data through the Biomart portal, and the database for the web portal of IntOGen.

Following there are more details about the transcriptomic and copy number alterations workflows.

c) Transcriptomic alterations

The first step consists on loading the studies that will be analysed, and then classify each of the samples expression data into normal or cancer, annotated by study and tumour type. The normal samples are pooled by study and tumour type, and then the expression data for the cancer samples with absolute intensities are converted into log_2 ratios by comparing them to the corresponding normal pool. The samples data already in log_2 ratio are just passed through to the next step.

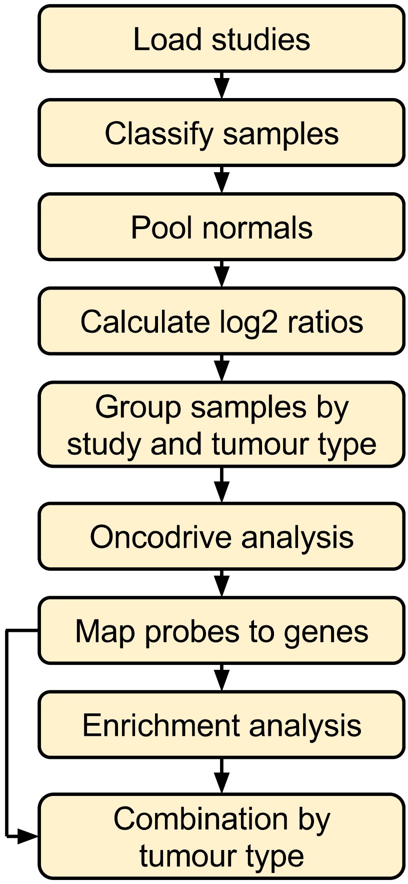

Figure 17: Transcriptomic alterations workflow for IntOGen Arrays

Next, the log_2 ratio samples datasets are grouped by study and tumour type and joined into matrices. Before determining whether a probe is over or under-expressed we need to determine a cutoff threshold for the log_2 ratios, but instead of using a constant value for all the matrices we calculate them dynamically for each matrix.

The next step converts the expression matrices into binary matrices for under and over-expression and the Oncodrive method is applied in order to see which probes are expressed less or more than expected by chance. Until this point all the matrices contained expression data for micro-array probes using GeneBank identifiers, but the following steps will require to have information at the level of genes, so the data is mapped into genes.

The enrichment analysis is also performed to determine biological modules (GO terms and pathways) that are expressed less and more than expected by chance.

The last steps consist in classifying and combining by tumour type the results of the analysis for genes and modules.

d) Copy Number Alterations

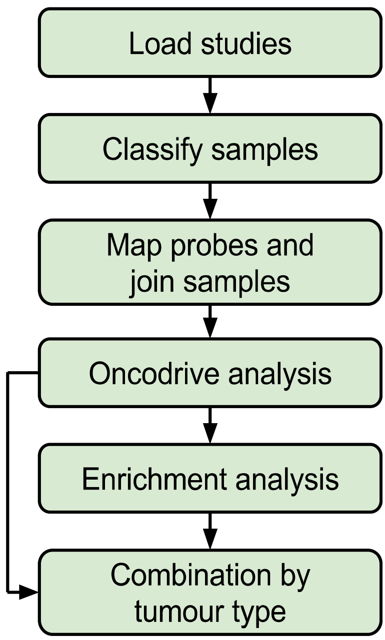

Figure 18: Copy number alterations workflow for IntOGen Arrays

The first step consists on loading the studies that will be analysed, and then classify each of the samples copy number data by study and tumour type. Then the probe identifiers are mapped into gene identifiers and the samples data joined into binary matrices by study and tumour type. There will be independent matrices for gain and loss. Next, the Oncodrive analysis is performed. The enrichment analysis is also performed to determine biological modules (GO terms and pathways) that are loss or gain more than expected by chance.

The last steps consist in classifying and combining by tumour type the results of the analysis for genes and modules.

e) Biomart

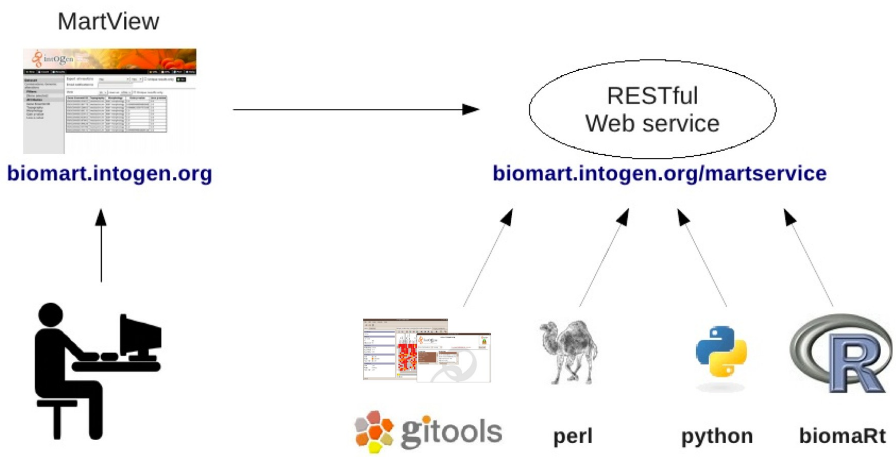

Figure 19: BioMart interface for IntOGen Arrays

With the aim of making the analysis results easily available to as much researchers as possible I developed a Biomart portal and a web service. The Biomart portal is suitable for researchers to query for data very easily through a web user interface, while the web service is suitable for accessing the data programmatically from several programming languages and tools. Gitools is an example of a tool that can access this data for further analysis using the web service.

2.2. IntOGen Mutations

a) Workflow organization

Before starting to describe the workflow in detail it is necessary to explain some general concepts. The workflow is designed to process a set of data projects in the same execution. Each data project represents a set of variants found in a cohort of tumours of the same cancer type with the same experimental conditions. Each data project can have annotations (the source of the data, the authors or institution, the publication, and the cancer type are some examples), and can be configured for some parameters independently of the other projects (for example to define different thresholds or expression filters depending on the characteristics of the tumour tissue).

The workflow can be divided into these main parts or sub-workflows:

- Variants processing
- Functional impact assessment
- Variant recurrences
- Identification of drivers
- Combination of project results
- Generation of results

The source code is open source and can be found at https://bitbucket.org/intogen/mutations-analysis

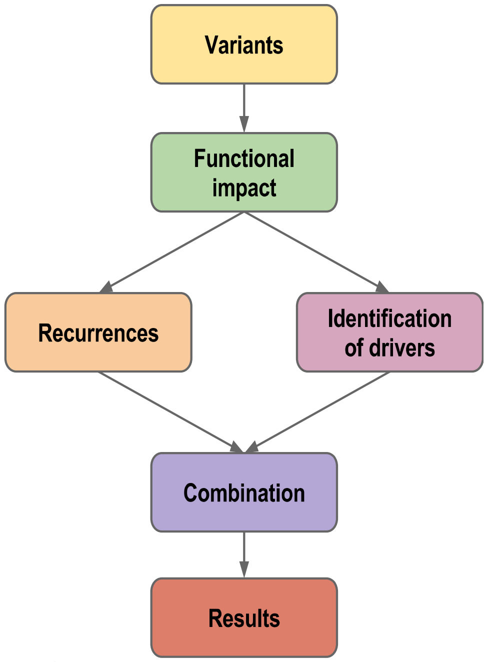

Figure 20: Workflow for IntOGen Mutations

b) Variants processing

The first part, which can be executed concurrently for each project, involves reading the information available for each data project (annotations and configuration), parsing the variant files provided by each project and mapping the coordinates that are based on the NCBI36 (hg18) genome assembly into the GRCh37 (hg19). The variant files can be in any of the following formats: simple tabulated text, MAF or VCF.

c) Functional impact assessment

To assess the impact of variants on the function of the proteins we use several methods. First we need to know which effect is having this variant on the gene products. Whether it is synonymous, missense, a frameshift or a stop codon to name some examples, is assesed by the tool Variant Effect Predictor (VEP) (McLaren et al., 2010). The consequence terms are given using the Sequence Ontology (SO) (Eilbeck et al., 2005).

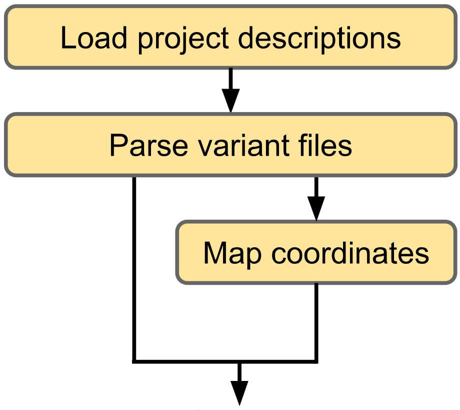

Figure 21: Variants processing workflow

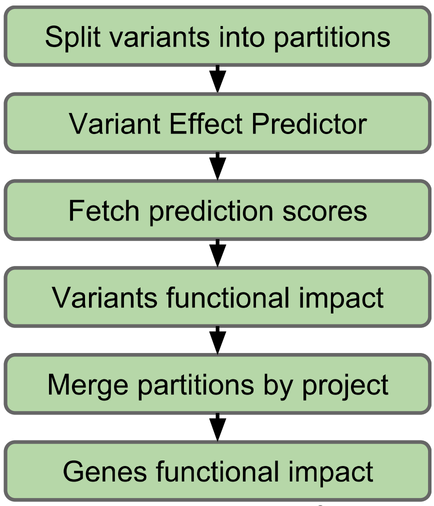

Figure 22: Functional impact assessment workflow

In the case of missense variants we collect the functional impact scores given by SIFT, Polyphen2 and MutationAssessor predicting methods and transform them using the TransFIC method which compares the original score to the distribution of scores of SNVs observed in the germline of human populations in genes that are similar to the one bearing the mutation. It allows to classify variants into one of four categories: *None* (it does not affect protein sequence), *Low*, *Medium* or *High*. For other effects we predefine the impact and the scores accordingly using a simple decision tree. For example, stop codons and frameshifts are considered *High* impacting while synonymous variants are *None*. The impact is assessed for each of the individual transcripts affected by the variant, as well as for the gene. To increase the parallelization of the execution the variants are split into partitions of fixed size (determined by the configuration parameter vep_partition_size), and finally everything is merged again by project before calculating the impact per gene.

d) Variant recurrences

This sub-flow is responsible for counting how many samples are affected by each variant, each gene and each pathway for each project. It also calculates the proportion that these frequencies represent respect to the total number of samples analysed in the project. The recurrences are implemented as a single task that can be calculated in parallel for each project.

e) Identification of drivers

Identification of drivers relies on two methods developed in our group: OncodriveFM and OncodriveCLUST.

OncodriveFM

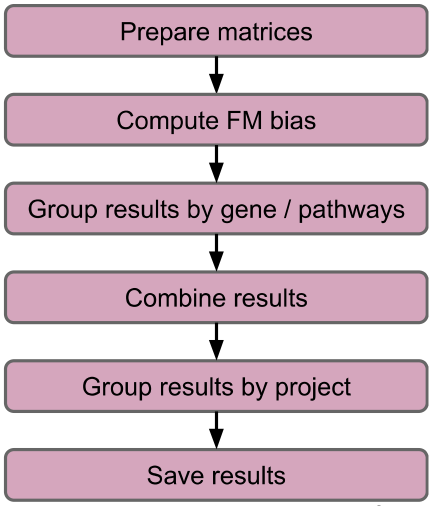

Figure 23: OncodriveFM workflow

It is based on the assumption that cancer driver genes accumulate highly functional mutations, as those are the ones that alter the function of the encoded protein, conferring a selective advantage to the cell. OncodriveFM computes a metric called FM-bias, which measures the bias towards the accumulation of functional mutations. Genes with a high FM-bias are candidate drivers. The same approach is also applied for pathways.

This method requires to perform many randomizations and so is computationally expensive. I have implemented it in a way that allows to split the whole analysis in smaller parts that can be executed independently in different computers (coarse grain) using multiple processors for each part (fine grain).

For each project three matrices are generated with genes in rows and samples in columns. The cells of each matrix will contain one of the three TransFIC scores obtained in previous steps for each of the prediction methods (SIFT, PolyPhen2 and MutationAssessor). Each matrix is computed and the results combined to get a unique result per project and gene. The input matrices will only contain genes which are affected by synonymous, missense, stop or frameshift variants and will constitute the background of the null distribution. But only the genes that pass a predefined filter of allowed genes based on the expression patterns of this gene in a given tissue and the genes being affected in at least 20 samples will be analysed and will get results from the method.

The results consists on a *p-value* and a *q-value* (the *p-value* corrected for multiple testing using Benjamini & Hochberg FDR) per gene and project.

OncodriveCLUST

This method is based on the assumption that if mutations appear clustered in a certain location of the gene is because it may confer a selective advantage to the cell. The input consists on two lists, one with the genes and samples affected by synonymous variants (used to compute the background distribution) and one with the genes and samples affected by non synonymous, stop, codon or splice junction variants. The analysis gives as a result per gene and project a *z-score*, a *p-value* and a *q-value* (the *p-value* corrected for multiple testing using Benjamini & Hochberg FDR).

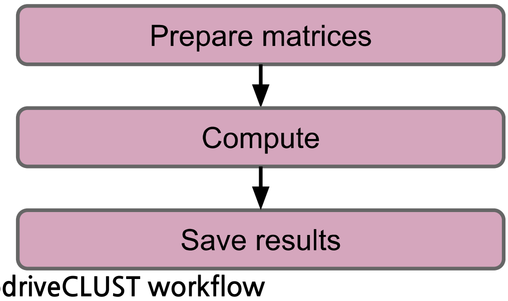

Figure 24: OncodriveCLUST workflow

f) Combination of project results

Up to this point all the calculations have been done for each project independently. This sub-workflow is responsible for combining the results generated for all the projects according to some criteria of combination. Basically it groups the projects by a set of annotations and then combines the results of recurrences and driver identification. Currently two criteria are used, one simply combines all the projects and the other combines the projects per cancer site. In the case of the recurrences the frequencies are just aggregated and the proportions recalculated for variants, genes and pathways. In the case of driver identification the gene's p-values are combined with Fisher's method and the pathways' *z-scores* with Stouffer method from which we get a *p-value*. The combined *p-values* are also corrected for multiple testing with Benjamini & Hochberg FDR.

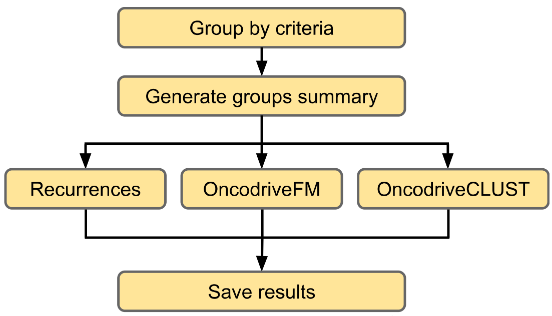

Figure 25: Combination of projects results workflow

g) Generation of results

The following files are generated for the results of the analysis:

project.tsv

This file contains information about the project analysed. Basically contains the following fixed fields:

- **PROJECT_ID**: The project identifier.
- **ASSEMBLY**: The genome assembly.
- **SAMPLES_TOTAL**: The total number of samples analysed.

There will be also columns representing project annotations in case they were specified.

consequences.tsv

This file contains information about the transcripts affected by the input mutations. It contains mainly the Variant Effect Predictor results and TransFIC calculations.

- **PROJECT_ID**: The project identifier.
- **CHR**: The mutation's chromosome.
- **STRAND**: The mutations's strand.
- **START**: The mutation's start position.
- **ALLELE**: The mutation's affected nucleotides. The reference and changed sequences are separated by a slash '/'.
- **TRANSCRIPT_ID**: The Ensembl identifier of the transcript affected by the mutation.
- **CT**: The list of Sequence Ontology terms describing the consequence of the mutation.
- **GENE_ID**: The Ensembl identifier of the gene coded by the transcript.

- **SYMBOL**: The HUGO symbol of the gene.
- **UNIPROT_ID**: The Uniprot identifier of the protein.
- **PROTEIN_ID**: The Ensembl identifier of the protein.
- **PROTEIN_POS**: The position of the mutation in protein coordinates.
- **AA_CHANGE**: The aminoacids change separated by a slash '/'.
- **SIFT_SCORE**: SIFT score of the mutation as obtained from VEP (mutations whose consequence types are not prone to affect the sequence of the protein product have empty values).
- **SIFT_TRANSFIC**: The transformed score of SIFT calculated with TransFIC.
- **SIFT_TRANSFIC_CLASS**: Classification of this mutation based on the SIFT TransFIC and its separation of highly-recurrent and non-recurrent somatic mutations in COSMIC.
- **PPH2_SCORE**: Polyphen2 score of the mutation as obtained from VEP (mutations whose consequence types are not prone to affect the sequence of the protein product have empty values).
- **PPH2_TRANSFIC**: The transformed score of Polyphen2 calculated with TransFIC.
- **PPH2_TRANSFIC_CLASS**: Classification of this mutation based on the Polyphen2 TransFIC and its separation of highly-recurrent and non-recurrent somatic mutations in COSMIC.
- **MA_SCORE**: Mutation assessor score of the mutation as obtained from the Mutation assessor database (mutations

whose consequence types are not prone to affect the sequence of the protein product have empty values).

- **MA_TRANSFIC**: The transformed score of MA calculated with TransFIC.
- **MA_TRANSFIC_CLASS**: Classification of this mutation based on the MA TransFIC and its separation of highly-recurrent and non-recurrent somatic mutations in COSMIC.
- **IMPACT**: assessment of the functional impact of the mutation on the transcript. The possible values that can take are: (4) mutation that doesn't affect the protein sequence, (3) non-synonymous mutation with low MA TransFIC, (2) non-synonymous mutation with medium MA TransFIC, (1) non-synonymous mutations with high MA TransFIC, stop mutation or frameshift causing indel.
- **IMPACT_CLASS**: Classification label for the impact: (4) none, (3) low, (2) medium, (1) high.

variant_genes.tsv

This file contains information about the mutations affecting genes.

- **PROJECT_ID**: The project identifier.
- **CHR**: The mutation's chromosome.
- **STRAND**: The mutations's strand.
- **START**: The mutation's start position.
- **ALLELE**: The mutation's affected nucleotides. The reference and changed sequences are separated by a slash '/'.
- **GENE_ID**: The Ensembl identifier of the gene coded by the transcript.

- **SYMBOL**: The HUGO symbol of the gene.
- **VAR_IMPACT**: assessment of the functional impact of the mutation on the on the gene. The possible values that can take are: (4) mutation that doesn't affect the protein sequence, (3) non-synonymous mutation with low MA TransFIC, (2) non-synonymous mutation with medium MA TransFIC, (1) non-synonymous mutations with high MA TransFIC, stop mutation or frameshift causing indel.
- **VAR_IMPACT_CLASS**: Classification label for the impact: (4) none, (3) low, (2) medium, (1) high.
- **SAMPLE_FREQ**: Number of samples where this mutation has been found.
- **SAMPLE_PROP**: Proportion of samples presenting this mutation among the total number of samples.
- **SAMPLE_TOTAL**: The total number of samples analysed.
- **CODING_REGION**: Whether the mutation affects to the coding region of the gene. It will take value 1 when at least one transcript have any of the following consequence terms: missense_variant, stop_gained, stop_lost, frameshift_variant, synonymous_variant, splice_donor_variant, splice_acceptor_variant, splice_region_variant. And 0 otherwise.
- **XREFS**: The comma separated list of external references. When the mutation is known and have an identifier in an external source (such as dbSNP or COSMIC).

variant_samples.tsv

This file contains information about the sample identifiers associated with each mutation.

- **PROJECT_ID**: The project identifier.
- **CHR**: The mutation's chromosome.
- **STRAND**: The mutations's strand.
- **START**: The mutation's start position.
- **ALLELE**: The mutation's affected nucleotides. The reference and changed sequences are separated by a slash '/'.
- **SAMPLES**: The comma separated list of samples where this mutation has been found.

genes.tsv

This file contains information for genes.

- **PROJECT_ID**: The project identifier.
- **GENE_ID**: The Ensembl identifier of the gene coded by the transcript.
- **SYMBOL**: The HUGO symbol of the gene.
- **FM_PVALUE**: P-value obtained from the OncodriveFM analysis. Genes with small P-values have a greater likelihood of being drivers.
- **FM_QVALUE**: The OncodriveFM P-value corrected by FDR.
- **SAMPLE_FREQ**: Number of samples where this gene has been found mutated.
- **SAMPLE_PROP**: Proportion of samples having this gene mutated among the total number of samples.
- **SAMPLE_TOTAL**: The total number of samples analysed.
- **CLUST_ZSCORE**: Z-score obtained from OncodriveCLUST analysis.

- **CLUST_PVALUE**: P-value obtained from OncodriveCLUST analysis.
- **CLUST_QVALUE**: The OncodriveCLUST P-value corrected by FDR.
- **CLUST_COORDS**: The coordinates obtained from OncodriveCLUST analysis.
- **XREFS**: The comma separated list of external references for the overlapping mutations. When the mutation is known and have an identifier in an external source (such as dbSNP or COSMIC).

pathways.tsv

This file contains information for pathways.

- **PROJECT_ID**: The project identifier.
- **PATHWAY_ID**: The pathway identifier.
- **GENE_COUNT**: Number of genes known to be associated with this pathway.
- **FM_ZSCORE**: Z-score obtained from the OncodriveFM analysis.
- **FM_PVALUE**: P-value obtained from the OncodriveFM analysis. Pathways with small P-values have a greater likelihood of being drivers.
- **FM_QVALUE**: The OncodriveFM P-value corrected by FDR.
- **SAMPLE_FREQ**: Number of samples where this gene has been found mutated.
- **SAMPLE_PROP**: Proportion of samples having this gene mutated among the total number of samples.

- **SAMPLE_TOTAL**: The total number of samples analysed.

fimpact.gitools.tdm

This file contains the functional impact matrix for samples and genes. It is in a format that can be opened with Gitools.

h) Quality control

The workflow measures several metrics related to the quality of the results:

Variants processing

Number of mutations by stage: Shows the number of mutations that pass or are discarded through the different stages in which are involved (reading from the source, parsing, translation from hg18 to hg19 if it is necessary, and variant effect predictor). It is represented with a bar plot and a table for each of the sources involved in the analysis.

Drivers identification with OncodriveFM

Number of genes by stage: How many genes pass or are discarded in each of the stages. There are a number of genes that has been affected by the variants (source) that are selected depending on the consequence terms obtained from the Variant Effect Predictor (selected), then the selected genes are filtered using some predefined filters depending on the tissue (filter) and finally only those affecting to a minimum of samples will be analysed with OncodriveFM (threshold).

Number of samples by stage: How many samples are represented by the genes that pass or are discarded through the previously commented stages. This and the previous indicators are represented by both a bar plot and a table.

Number of samples per gene: This is represented with a plot where each element in the x-axis is a gene and the y-axis represents the number of samples having a variant that affects this genes. The genes are sorted from higher to lower number of samples.

Number of significant p-values for different thresholds: This is a plot showing how many genes would have a significant p-value for different *p-value* thresholds.

Drivers identification with OncodriveCLUST

Number of genes by stage: How many genes pass or are discarded in each of the stages. There are a number of genes that has been affected by the variants (source) that are selected depending on the consequence terms obtained from the Variant Effect Predictor into synonymous (syn) and non synonymous (selected), then the selected genes are filtered using some predefined filters depending on the tissue (filter) and finally only those affecting to a minimum of samples will be analysed with OncodriveCLUST (threshold).

Number of samples by stage: How many samples are represented by the genes that pass or are discarded through the previously commented stages. This and the previous indicators are represented by both a bar plot and a table.

Number of samples per gene: This is represented with a plot where each element in the x-axis is a gene and the y-axis represents the number of samples having a variant that affects this genes. The genes are sorted from higher to lower number of samples.

Number of significant p-values for different thresholds: This is a plot showing how many genes would have a significant *p-value* for different p*-value* thresholds.

	Passed	Discarded
Source	932	
Parser	912	20
LiftOver	912	0
VEP	909	3
Total		23

	Genes		Samples	
	Passed	Discarded	Passed	Discarded
Source	1296		37	
Selected	505	791	37	0
Filter	276	229	36	1
Threshold*	17	259	23	13
Total		1279		

	Genes		Samples	
	Passed	Discarded	Passed	Discarded
Source	1296		37	
Selected	463	708	26	11
Filter	231	232	22	4
Threshold*	0	231	0	22
Total		1171		

Figure 26: Quality control metrics visualization

i) Web site

The workflow can be run as a standalone command line application, or using the web site at http://www.intogen.org/analysis. Using the web site is the recommended procedure for people not familiarized with unix and terminals. A part from the resulting files, it generates a web portal to browse the results in the context of the IntOGen site data.

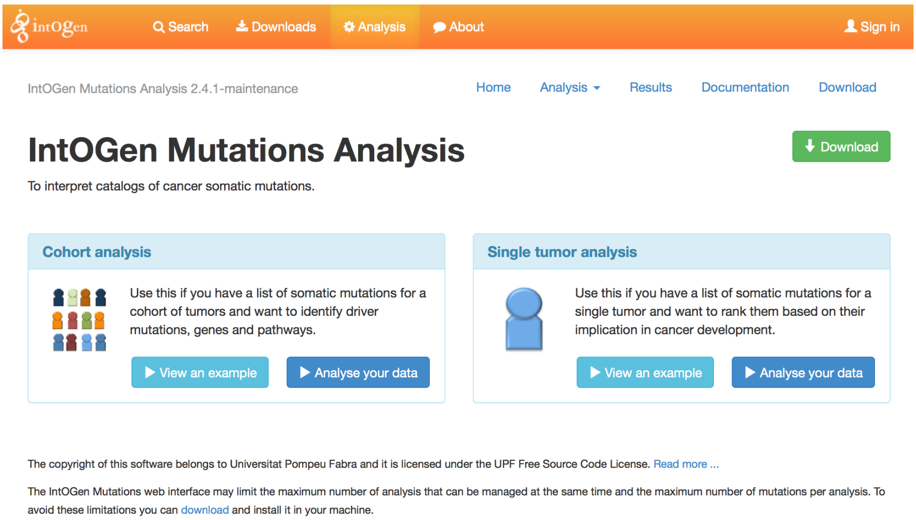

Figure 27: Screenshot of the IntOGen Mutations Analysis web site

Gundem G, Perez-Llamas C et al. (2010) IntOGen: integration and data mining of multidimensional oncogenomic data. Nature methods 7(2): 92-93. doi:10.1038/nmeth0210-92

This paper was used for Gunes' dissertation. I contributed to the collection and organization of the data, developed and performed its analysis, and reviewed the paper methods.

Perez-Llamas C, Gundem G, and Lopez-Bigas N (2011) Integrative cancer genomics (IntOGen) in Biomart. Database: The Journal of Biological Databases & Curation 2011.39. doi: 10.1093/database/bar039

This paper was used for Gunes' dissertation. I designed and developed the Biomart interface and its integration with IntOGen data, and contributed to, and reviewed the paper.

Gonzalez-Perez A, Perez-Llamas C et al. (2013) IntOGen-mutations identifies cancer drivers across tumor types. Nature methods 10(11): 1081-2. doi:10.1038/nmeth.2642

I organized the data, developed the analysis pipelines, the web portal for the analysis, and contributed for analysing the mutations data.

3. BENCHMARK OF IMPACT PREDICTION TOOLS

3.1. Introduction

Tumorigenesis is often described as a darwinian evolutionary process, where cells with genetic or epigenetic somatic alterations conferring favourable capabilities proliferate faster than their neighbours (Bignell et al., 2010; Greenman et al., 2007). Nevertheless, as an added consequence of malignization, chromosomal instability favours the acquisition of somatic alterations that are neutral to cancer cells. At the time of tumour sequencing, dozens to thousands of somatic alterations are frequently uncovered, thus posing the problem to distinguish between the drivers that contribute to the cancer phenotype and the passengers. Identifying driver alterations in a patient's tumour is essential to understand tumorigenesis and to devise therapeutic strategies. Driver mutations may be relevant both as targets, like in the case of mutant oncoproteins which may be inhibited by small molecules, and as potential determinants of resistance to therapy (Rubio-Perez et al., 2015).

Several bioinformatics tools have been developed in recent years that aid to identify potentially driver missense variants. Some of them are designed to assess the functional impact of non-synonymous variants, others, specifically aim to recognize potential driver mutations and could then, in principle, be used to detect potential driver non-synonymous variants ab initio (Carter et al., 2009).

Benchmarking the performance of these tools is a difficult task, due to the lack of precise datasets of driver and passenger mutations. Gonzalez-Perez et al (Gonzalez-Perez, Deu-Pons, & Lopez-Bigas, 2012) introduced the concept of proxy datasets enriched for driver and passenger mutations to assess the performance of tools aimed at identifying missense cancer mutations. The usage of multiple proxy datasets is intended to detect trends of the performance of methods in the task of identifying likely driver mutations, rather than a meticulous assessment of their precision.

Here, following that rationale, I assembled several proxy datasets composed of somatic missense mutations obtained from both episodic gene sequencing and systematic whole exome or whole genome sequencing studies. I used them to benchmark the performance of several functional impact assessment algorithms, driver mutations identification tools and general purpose conservation scores.

As a result of this work I ended up with a database of pre-calculated scores for the whole proteome (FannsDB), an updated Condel tool (González-Pérez & López-Bigas, 2011), and a methodology framework for benchmarking.

3.2. Methodology

All the steps required to collect data, create the database, prepare the proxy datasets and evaluate the performance of the tools were implemented using Python and IPython Notebooks, and the source code versioned with git, so all this work can be reviewed and reproduced in the future.

a) Prediction tools

The selected tools can be divided by categories as:

- Functional impact assessment algorithms: SIFT (Ng & Henikoff, 2003), PolyPhen2 (Adzhubei et al., 2010), MutationTaster (Schwarz, Cooper, Schuelke, & Seelow, 2014), and FATHMM for inherited disease (Shihab et al., 2013).
- Driver mutations identification tools: Mutation Assessor (Reva, Antipin, & Sander, 2011), FATHMM for cancer (Shihab et al., 2013), CHASM (Carter, Samayoa, Hruban, & Karchin, 2010), and InCa (Supek & Vlahovicek, 2004).
- General purpose conservation scores: GERP RS (Cooper et al., 2005), PhyloP (Pollard, Hubisz, Rosenbloom, & Siepel, 2010), and C-Score (Kircher et al., 2014).

I also used tools that combine or transform some of the previous ones to improve the performance of the predictions:

- Combination scores: Condel (González-Pérez & López-

Bigas, 2011)

- Transformation scores: TransFIC (Gonzalez-Perez et al., 2012)

Most tools' scores (SIFT, PolyPhen2, MutationTaster, MutationAssessor, SIFT, PolyPhen2, MutationTaster, GERP RS, PhyloP, and FATHMM for inherited disease) were obtained from the dbNSFP database (X. Liu, Jian, & Boerwinkle, 2011), and the others (CADD, CHASM and InCa) by using the original tool.

b) Predictors' scores database

Having to calculate the scores independently for each tool whenever they are needed is not scalable at a genome wide level, so I used pre-calculated scores for most of them for all the proteome (SIFT, PolyPhen2, MutationTaster, FATHMM, Mutation Assessor, GERP RS, PhyloP, C-Score), and some others only for the SNVs in the following proxy datasets (mainly CHASM and InCa due to time constrains), and store the results in a MongoDB database with a total of 241 millions of records, where each record represents a possible proteome variant with all the scores. A record looks something like:

```
{
    "_id" : ObjectId("5306541e830434415357dcb7"),
    "g" : {
        "c" : "9",
        "s" : 32473058,
        "r" : "T",
        "a" : "A",
        "d" : "-",
        "t" : "ENST00000379868"
```

```
        },
        "p" : {
            "n" : "ENSP00000369197"
            "p" : 440,
            "r" : "L",
            "a" : "F",
        },
        "s" : {
            "CONDEL" : 0.5258461337047758,
            "FATHMM" : 0.4,
            "MA" : 3.78,
            "PPH2" : 0.907,
            "SIFT" : 0
        }
    }
```

Where:

- **g**: contains genome coordinates
  - **c**: chromosome
  - **d**: strand
  - **s**: start position
  - **r**: nucleotide in the reference genome
  - **a**: nucleotide for the alternative change
  - **t**: Ensembl transcript ID
- **p**: protein coordinates
  - **n**: Ensembl protein ID
  - **p**: protein amino-acid position
  - **r**: amino-acid in the reference proteins
  - **a**: amino-acid fro the alternative change
- **s**: predictors scores, where the key is the predictor ID and the value the score.

Given that MongoDB stores the key names together with the values and that there are millions of records I decided to use those short key names to save disk space (in the order of giga-bytes).

I created indices for querying scores by both genome and protein coordinates which, after a long process of creating the database, allowed fast queries.

The IPython Notebooks can be viewed at:

- Creation of the predictors scores database (cluster)
- Creation of the predictors scores database (workstation)

c) Condel scores

In the framework of this project I updated Condel scores using the previous database. The update consisted in using an updated Ensembl database, and changing which primary predictors were integrated, the last version of Mutation Assessor which was already used in the previous version, and FATHMM which was not used in the previous version. SIFT and PolyPhen2 which were used in the previous version were removed for the new one.

The IPython Notebook can be viewed at: Calculation of Condel scores

d) Proxy datasets

Several datasets were created, each one containing a list of protein mutations extracted from a certain project under certain criteria.

The IPython Notebook can be viewed at: Generation of proxy datasets

HumVar Polymorphisms

I basically took the HumVar training datasets used for Polyphen 2, one for neutral mutations (*humvar-n*) and other for deleterious ones (*humvar-d*).

IntOGen Mutations

After executing the IntOGen mutations pipeline for 48 projects and merging all the missense variants, a total of 489234 SNVs were identified and used for generating new datasets according to different criteria:

- *wg-<n>*, where <n> in {1, 2, 3, 4}, are datasets containing all SNVs occurring at least 1, 2, 3 or 4 times respectively.
- *wg-CGC* and *wg-noCGC* contain SNVs according to whether the genes in the Cancer Gene Census (CGC) are affected or not (respectively).
- *wg-TD* and *wg-noTD* contain SNVs according to whether the list of genes proposed in (Tamborero, Gonzalez-Perez, Perez-Llamas, et al., 2013) are affected or not (respectively).
- *wg-PD* and *wg-noPD* contain SNVs according to whether the list of genes proposed in (Rubio-Perez et al., 2015) are affected or not (respectively).
- *wg-CD* and *wg-noCD* contain SNVs according to whether the list of genes identified as drivers in Chasm are affected

or not (respectively).

- *wg-TD-noCGC*, *wg-PD-noCGC*, *wg-noTD-noPD-noCGC*, *wg-*-noCD* are variants of the previous ones excluding different sets of genes.
- *wg-ov-** are datasets specific for the TCGA ovary project.

COSMIC

I generated also some datasets from COSMIC v68, both for individual gene (prefixed *cm-*) and wide screen studies (prefixed *cw-*).

- *{cm,cw}-⟨n⟩*, where ⟨n⟩ in {1, 2, 3, 4}, are datasets containing all SNVs occurring at least 1, 2, 3 or 4 times respectively.
- *{cm,cw}-CGC* and *{cm,cw}-noCGC* contain SNVs according to whether the genes in the Cancer Gene Census (CGC) are affected or not (respectively).
- *{cm,cw}-CGC-{D,R}-⟨n⟩* contain SNVs according to whether the genes in the Cancer Gene Census (CGC) are dominant (D) or recessive (R) with at least n occurrences (for n in {1, 2})..
- *{cm,cw}-TD* and *{cm,cw}-noTD* contain SNVs according to whether the list of genes proposed in (Tamborero et al., 2013) are affected or not (respectively).
- *{cm,cw}-PD* and *{cm,cw}-noPD* contain SNVs according to whether the list of genes proposed in (Rubio-Perez et al., 2015) are affected or not (respectively).
- *{cm,cw}-CD* and *{cm,cw}-noCD* contain SNVs according to

whether the list of genes identified as drivers in Chasm are affected or not (respectively).

- {cm,cw}-TD-noCGC, {cm,cw}-PD-noCGC, {cm,cw}-noTD-noPD-noCGC, {cm,cw}-*-noCD are variants of the previous ones excluding different sets of genes.
- {cm,cw}-<n>-<gene> are datasets specific for a given gene where gene in {TP53, EGFR, CTNNB1, PTEN, PIK3CA} with at least n occurrences where n in {1, 2}.

e) Performance evaluation

Using the database for prediction scores generated previously, the scores for the SNVs in the generated proxy datasets were retrieved. Then the testing datasets comparing expected driver SNVs (positive cases) versus expected non driver SNVs (neutral cases) were generated, and used to calculate the following performance metrics (see Figure 28) using the *scikit-learn* (Pedregosa et al., 2011) Python library:

- True Positives, False Positives, True Negatives, False Negatives
- Sensitivity / Recall (TPR)
- Specificity (SPC)
- Precision (PPV)
- Accuracy (ACC)
- Matthews Correlation Coefficient (MCC)
- ROC curve and Area under the curve (AUC)

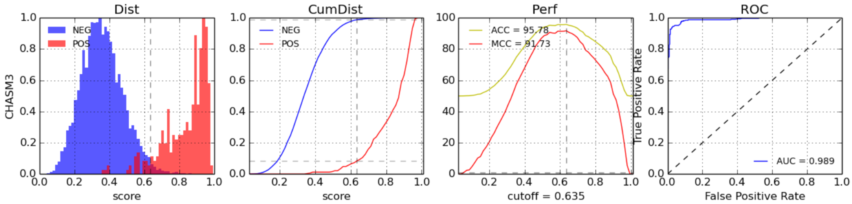

Figure 28: Example of performance metrics calculated for each predictor and its representation in plots.

One of the major problems with the testing datasets (specially with wide genome studies) is that they are quite unbalanced, having many more cases for the neutral ones. Some examples extracted from the notebook shows clearly the problem:

```
wg-2__wg-1    POS: [  18831]  NEG: [ 469692]
wg-3__wg-1    POS: [   2743]  NEG: [ 469692]
wg-4__wg-1    POS: [    967]  NEG: [ 469692]
```

The unbalancing affects many of the performance metrics and the problem needs to be addressed. I used under-sampling with randomization, where I generate several random sub-datasets of the size of the positive cases, calculate the performance metrics for each one, and aggregate the results by using the mean. Moreover, the main metric used to compare performance between different predictors was the MCC, as it is not biased by unbalanced datasets (Thusberg, Olatubosun, & Vihinen, 2011) (see Figure 29 for an example of the comparisons).

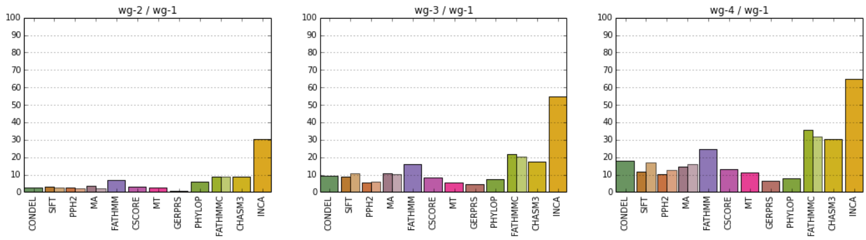

Figure 29: Example of MCC performance comparison between predictors for 3 datasets

The IPython Notebook can be viewed at: [Evaluation of performance]

3.3. Results

FannsDB

We have created a database integrating several pre-calculated predictors' scores for all possible non-synonymous SNVs in the whole proteome (called FannsDB). The database was used to update the Condel scores (see Figure 30), using different predictors than the previous version (Mutation Assessor and FATHMM) and a newer version of the Ensembl database (feb-2012). We also created a new web portal for the database, to allow other researchers to perform queries and retrieve, not only the updated Condel scores, but also SIFT, PolyPhen2, Mutation Assessor and FATHMM scores. The web can be accessed at http://bg.upf.edu/fannsdb/

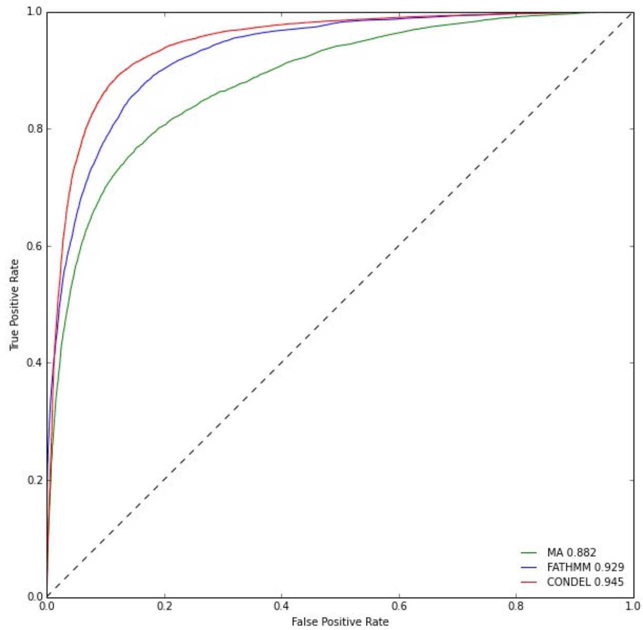

Figure 30: ROC curve for Condel compared to the other predictors.
The legend shows the Area Under the Curve (AUC) for each tool.

Benchmarking

One first pattern that emerged from the MCC assessment on all proxy datasets assembled from the collections of somatic mutations, is that tools specifically designed to detect driver mutations perform better than tools aimed at assessing the functional impact of variants and conservation scores (see Figure 31). For example, for the testing dataset *cm3-noCD/cm1*, while functional impact tools score a maximum MCC of 0.41 and the conservation tools 0.20, drivers tools score between 0.59 and 0.78. This is no surprise, because drivers tools incorporate information specific to somatic mutations involved in cancer, either in the form of weighted scores or training features.

All tools and scores perform better separating proxy datasets containing episodic somatic mutations detected in gene panels in COSMIC, as opposed to proxy datasets composed of systematically registered mutations in whole genome/exome sequencing studies in IntOGen (see Figure 32). For instance, the maximum MCC for drivers tools drops from 0.68 in the testing dataset cm-CGC-noCD/cm-noCGC, to 0.28 in dataset wg-CGC-noCD/wg-noCGC. The decline is from 0.24 to 0.05 for functional impact tools, and from 0.15 to 0.13 for score tools. This bias raises a concern, because real case studies probably resemble more the latter type of proxy dataset.

INCA produces the highest MCCs among the three tested drivers tools in a sustained manner across most proxy datasets. Nevertheless, the number of mutations it can actually score (coverage) is very low, because the tool only evaluates mutations occurring in proteins with a solved 3D structure.

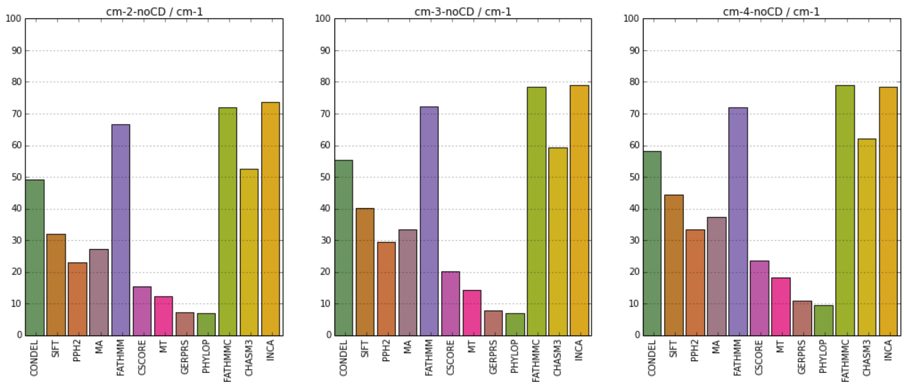

Figure 31: Comparison of MCCs for COSMIC manual curated mutations based on recurrences.

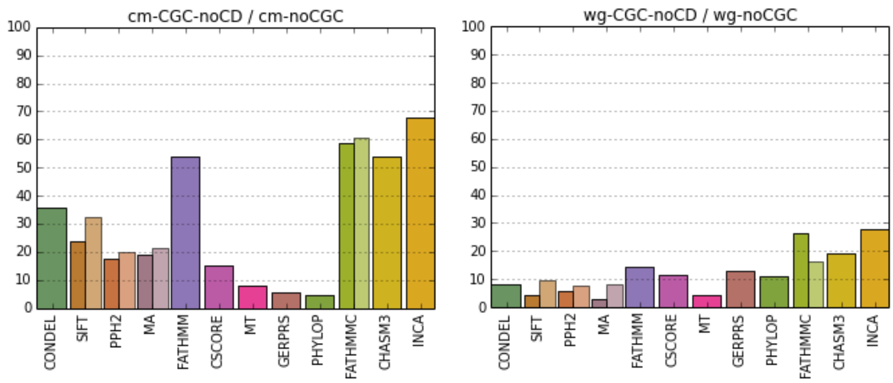

Figure 32: Comparison of MCCs between COSMIC datasets for mutations from gene panels and IntOGen mutations from whole genome/exome studies.

3.4. Discussion

Although tools originally designed to assess the functional impact of missense mutations are frequently employed by cancer genomics projects with the aim of detecting driver mutations, our results show that tools specifically designed to fulfil this purpose actually preform better. This trend is sustained across all the analysed proxy datasets, suggesting that researchers should prefer specific drivers tools in the attempt to detect driver mutations in tumour samples. Nevertheless it is important to point out that the performance of these three tools decrease in proxy datasets composed of mutations from whole genome/exome tumour sequencing compared to mutations extracted from COSMIC. Two of these tools probably exhibit some kind of overfitting towards catalogs of known driver mutations, either from the features they are trained on (CHASM) or from the weights the scores they implement contain (FATHMMC). These biases probably explain the aforementioned decrease in accuracy. The general recommendation for identifying driver mutations across cohorts of tumour samples is to employ in combination to these tools other sources of information, such as catalogs of known driver genes, or the accurate annotation of the functional consequence of mutations.

PART III: DISCUSSION AND CONCLUSION

DISCUSSION

Gitools

At the time of writing Gitools, we found ourselves with the need to represent and explore genomic datasets in a very intuitive way, being able to contextualize the data within the knowledge obtained from other biological databases such as Biomart, and needing to perform simple but powerful analysis on it. Furthermore, we envisioned that if all of these features were put in a single graphical application, not only us, but also other collaborators without advanced knowledge in bioinformatics, and without having to use more advanced tools such as R or other programming languages, could benefit from it. The relevance of this work became evident not only by the number of citations to the paper (currently 61 according to ResearchGate.com) or visits to its web page (500 sessions per month on July 2011, more than 1000 on July 2015), but specially, for its relevance for the development of other projects of vital importance for our research such as IntOGen Arrays (Gundem et al., 2010). Many projects in bioinformatics loose support just after its main contributor finishes its PhD and stops working on it, but in the case of Gitools its development continued beyond my last contributed version, thanks to the efforts of Michael Schroeder and Jordi Deu-Pons, who did a great job adding new methods for analysis, new visualization features, preparing ready-to-explore datasets from TCGA data, integrating it with the IGV (Thorvaldsdóttir, Robinson, & Mesirov, 2012), as well as optimizing its management of memory and thus its capabilities to work with bigger data.

Nowadays technologies for visualization and application development are moving from desktop platforms to web and mobile ones, and this is how I see the future development for this tool and any other one of these characteristics. As a precedent to this move, some intent to have the rich interactivity of heatmaps with web technologies has successfully been done by Michael Schroeder with jHeatmap (Deu-Pons, Schroeder, & Lopez-Bigas, 2014).

A downside of this project is that, as it keep expanding on usability and visualization features, its development goes beyond the scope of the research done in the group and require people that can focus exclusively in the software engineering side, and UX/web design and technologies. One way to overcome these limitations is to open its development to allow for external collaboration and contributions, which we already did by creating a repository in Github (https://github.com/gitools/gitools).

IntOGen

Integrate experimental data from several sources has not been an easy task, specially when it required manual curation and preprocessing of data that had to be coordinated among several interdisciplinary researchers. Even when, after several iterations, I was able to develop a simple model for managing experiments in IntOGen Arrays (based on more complete and complex ones such as MAGE and FUGE), our efforts to use and develop the appropriate ontologies, and develop and document the processes and conventions, the coordination of individuals to ensure a common criteria and standard for annotations, and the communication between the team members with diverse backgrounds and mental models, was really a challenge (definitively the way a biologist understands models and reasons about information is different from the way an engineer does). The result, a part from several months of delay for the third release of IntOGen Arrays, was a great experience for all of us who really understood how important is to keep things as simple as possible, as well as to narrow the scope for the short to medium term needs as much as possible. I wish I knew all what I know now about agile development in software engineering, which I think could also be applied, and of great value, for research projects.

The previous learning was successful in the development of the data model for IntOGen Mutations, which was really simple and easy to understand, and only with a bit of care to normalize the experimental data, it didn't become an obstacle to our research. As a result the analysis pipeline became a very valuable tool for deriving other research results (Rubio-Perez et al., 2015; Tamborero, Gonzalez-Perez, Perez-Llamas, et al., 2013), and used by other researchers around the globe through the web portal.

Aside from the challenges from managing the experimental data, the work related to the integration of several bioinformatic tools, and the analysis of the data, deserve some attention too. A computational workflow, also referred to as pipeline, defines the steps that need to be followed for a certain analysis of a minimum complexity, as well as which are the tools and parameters that have to be used. At the time of working on the first version of the IntOGen Arrays analysis workflow, I just used makefiles to glue the steps together, but the source become obscure rapidly and difficult to understand, as well as difficult to reproduce. By then, too low level distributed computing technologies such as MPI existed, as well as some advanced workflow management systems, such as Kepler (Altintas et al., n.d.), or, Taverna (Oinn et al., 2004) in its beginnings. But neither of them fitted all the requirements I was expecting, and inspired by the so called big data technologies emerging at that time, Hadoop (https://hadoop.apache.org/), I decided to start the development of my own system using the Python programming language called Wok (https://github.com/bbglab/wok). As a result, I was able to perform complex analysis for IntOGen in both multi-core computers and High Performant Computing (HPC) clusters with Sun Grid Engine (now Open Grid Scheduler at http://gridscheduler.sourceforge.net/) or SLURM (https://computing.llnl.gov/linux/slurm/). Furthermore, even some of my colleges without the knowledge to work with such HPC systems, were able to develop and use, their own workflows with Wok.

Nevertheless, I would completely discourage anyone else from doing the same I did with Wok, basically reinventing the wheel for distributed data management and analysis. Given the fast pace for new technologies, methods, and computational solutions to come and improve current solutions, even when I understand the benefits of using common workflow languages and platforms for integrating diverse scientific tools and promote reproducibility (CWL at https://github.com/common-workflow-language/common-workflow-language, WDL at https://github.com/broadinstitute/wdl), I am not sure that the amount of effort required to successfully implement them, prove its performance, and promote its adoption, will be worth enough compared to the more agile alternative of just learn and use existing tools and computational technologies, proven to be successful by companies that deal with very big amounts of data every day (i.e. Google, Yahoo, or Facebook). An example of what I see as a good move for bioinformatics nowadays is the ADAM project (Massie et al., 2013) developed by the AMPLab at Berkeley, and based on Spark, a trending technology for big data and machine learning processing, supported and contributed by hundred of engineers around the world. For an interesting view about computation technologies for science, and some of the concerns, the following paper and correspondences are worth the reading (Schadt, Linderman, Sorenson, Lee, & Nolan, 2010, 2011; Trelles, Prins, Snir, & Jansen, 2011). My view is that cloud and on-premises resources will co-exist for a while, or eventually become hybrid. I see that distributed systems, governed by cluster managers such as Mesos (Hindman et al., 2011) or Google Borg (Verma et al., 2015), and composed of hybrid CPU-GPU computers for dealing with both high IO throughput and HPC, will become the norm to work with massive data, and not that massive if their learning curve of adoption is not that hard as it is for traditional HPC computing. By the way, there will be several challenges for their adoption in science: (1) Updating the financial institutions rules to consider cloud computing as a fungible resource and encourage collaboration and sharing of resources among different researchers and institutions, (2) Consider computer science and software engineering people a first class citizen in research groups, even if they don't produce papers, to lower the barrier for the adoption of those new technologies by other researchers and promote technological innovation.

Benchmarking of functional impact predictors

One side product of this work is FannsDB, a database integrating functional impact scores for all possible punctual variants of the whole proteome for several prediction tools. Thanks to it, I was able to reduce the overhead of calculating the scores each time I generated a new dataset or performed a new test (which happened quite frequently during the development of the benchmarking framework). Furthermore, many different datasets contained common SNVs that would require re-calculating them each time. Its generation required some time, and the final size was significative, a total of 241 million records and hundred of gigabytes of disk space, but once created with the adequate indices and distributed across the nodes of a cluster, querying for prediction scores was a matter of milliseconds. Integrating several prediction scores in a single database is not a novel approach, dbNSFP (X. Liu et al., 2011; X. Liu, Jian, & Boerwinkle, 2013) is a file-based database integrating several scores together with annotations, and I was using it for getting most of the scores. The reasons I needed to derive a new database is because: (1) I needed to query the database in different ways (for example, by protein coordinates) with more flexibility and performance, thus the data and the annotations needed to be organized differently, (2) I needed to generate new scores calculated from the existing ones, as was the case for Condel and transFIC scores. Even if extracting and digesting its data was not trivial and required some care to map its annotations to Ensembl vocabularies for genes, transcripts and proteins according to my needs, it showed to be a very valuable resource and saved me a lot of time of getting each of the predictors scores individually.

FannsDB is not only a database, but also a public web application (http://bg.upf.edu/fannsdb/) that allows to make queries from lists of either genomic or protein coordinates for a subset of the available predictors. It was developed mainly to update the previous Condel version, which was outdated and having performance problems. Its initial intention was to serve updated scores for both projects Condel and transFIC, but due to lack of time, transFIC was not made public. One possible future work would be to include an update for transFIC, and expand on the available predictors for querying.

The results from the benchmarking are not revealing and show some obvious results such as it is better to use tools specifically designed for cancer instead of functional impact or conservation based tools. Some of the difficulties to interpret the results where related to the need to deal with the overfitting of tools, to either, catalogs of mutations commonly used by different tools for the purpose of comparison (such as HumVar), or to known drivers used for their training. When identifying driver mutations across cohorts of tumour samples it is recommended to combine the results of the prediction tools with other sources of information.

Reproducibility and Open Science

I started to use IPython Notebook (now Jupyter at http://ipython.org/) for the benchmarking, and rapidly became aware of how much useful is this tool for research. It allows to organize the workflow as if it was a history, where each command and its results live together in a cell, and the cells are organized like in a document, with headers and rich formatted explanations. There are other ways to organize the work, for example I was previously keeping, together with the source code, “readme” files in Markdown format (http://daringfireball.net/projects/markdown/), but using IPython Notebooks represented a step further. It allowed me to remember and understand all the work that I was doing more that one year before thanks to have it well organized in notebooks, as well as publish it.

With no doubt, reproducibility and open science are required features for research, and together with best practices for software development in science, should be taken seriously by any PhD candidate or researcher. Publishing the source code is an example, it allows for other researchers to understand and review better your work, as well as reproduce your results. Every one works under pressure and need the results as fast as possible. The consequences, specially when the best practices have not been converted and internalized as habits, derive in not having perfect code, documentation and so on and so forth (Baxter, Day, Fetrow, & Reisinger, 2006; Merali, 2010), but it is understandable, and at the end of the day, what really matters and make the difference is whether it is publicly available or not (Barnes, 2010; Nature, 2014).

I have published the source code related to my work, the notebooks for the benchmarking, and this dissertation in a public repository, so other researchers interested in my work can re-view it:

https://github.com/chris-zen/phd-thesis

CONCLUSION

- We have created Gitools, a tool that allows accessing to specialized databases in biology, to analyse data generated by high-throughput technologies, and to visualise multi-dimensional results with interactive heatmaps.

- The analysis workflows for IntOGen integrate public bioinformatic tools (such as Variant Effect Predictor), and specifically designed tools (such as Oncodrive, OncodriveFM and OncodriveCLUST), for the identification of alterations that drive tumorigenesis.

- We have integrated cancer data from several public repositories and large scale projects, and performed integrative analysis with IntOGen, revealing lists of genes that most likely drive tumorigenesis.

- IntOGen analysis results are accessible to other researchers through its web portal, a Biomart web service, and Gitools.

- We have developed FannsDB, a database integrating impact prediction scores for all possible non-synonymous SNVs of the whole proteome for several tools, and published a web portal for allowing other researchers to make queries.

- The benchmark of several prediction tools, using FannsDB and proxy datasets, shows that tools specifically designed for cancer, perform better than general ones, and does not show a clear recommended tool over the others, suggesting the need to combine them with other sources of information.

BIBLIOGRAPHY

Adzhubei, I. A., Schmidt, S., Peshkin, L., Ramensky, V. E., Gerasimova, A., Bork, P., … Sunyaev, S. R. (2010). A method and server for predicting damaging missense mutations. *Nature Methods*, *7*(4), 248-249. http://doi.org/10.1038/nmeth0410-248

Altintas, I., Berkley, C., Jaeger, E., Jones, M., Ludascher, B., & Mock, S. (n.d.). Kepler: an extensible system for design and execution of scientific workflows. In *Proceedings. 16th International Conference on Scientific and Statistical Database Management, 2004.* (pp. 423-424). IEEE. http://doi.org/10.1109/SSDM.2004.1311241

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., … others. (2000). Gene Ontology: tool for the unification of biology. *Nature Genetics*, *25*(1), 25-29.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., … Sherlock, G. (2000). Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nature Genetics*, *25*(1), 25-29. http://doi.org/10.1038/75556

Bard, J. B. L., & Rhee, S. Y. (2004). Ontologies in biology: design, applications and future challenges. *Nature Reviews Genetics*, *5*(3), 213-222. http://doi.org/10.1038/nrg1295

Barnes, N. (2010). Publish your computer code: it is good enough. *Nature*, *467*(7317), 753. http://doi.org/10.1038/467753a

Baudis, M., & Cleary, M. L. (2001). Progenetix.net: an online repository for molecular cytogenetic aberration data. *Bioinformatics (Oxford, England)*, *17*(12), 1228-1229. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/11751233

Baxter, S. M., Day, S. W., Fetrow, J. S., & Reisinger, S. J. (2006). Scientific software development is not an oxymoron. *PLoS Computational Biology*, *2*(9), e87. http://doi.org/10.1371/journal.pcbi.0020087

Benson, D. A., Cavanaugh, M., Clark, K., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., & Sayers, E. W. (2013). GenBank. *Nucleic Acids Research*, *41*(Database issue), D36-42. http://doi.org/10.1093/nar/gks1195

Bignell, G. R., Greenman, C. D., Davies, H., Butler, A. P., Edkins, S., Andrews, J. M., … Stratton, M. R. (2010). Signatures of mutation and selection in the cancer genome. *Nature*, *463*(7283), 893-8. http://doi.org/10.1038/nature08768

Brazma, A., Hingamp, P., Quackenbush, J., Sherlock, G., Spellman, P., Stoeckert, C., … Vingron, M. (2001). Minimum information about a microarray experiment (MIAME)-toward standards for microarray data. *Nature Genetics*, *29*(4), 365-71. http://doi.org/10.1038/ng1201-365

Carter, H., Chen, S., Isik, L., Tyekucheva, S., Velculescu, V. E., Kinzler, K. W., … Karchin, R. (2009). Cancer-specific high-throughput annotation of somatic mutations: computational prediction of driver missense mutations. *Cancer Research*, *69*(16), 6660-6667. http://doi.org/10.1158/0008-5472.CAN-09-1133

Carter, H., Samayoa, J., Hruban, R. H., & Karchin, R. (2010). Prioritization of driver mutations in pancreatic cancer using cancer-specific high-throughput annotation of somatic mutations (CHASM). *Cancer Biology & Therapy*, *10*(6). Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/20581473

Check Hayden, E. (2014). Is the $1,000 genome for real? *Nature*. http://doi.org/10.1038/nature.2014.14530

Cingolani, P., Platts, A., Coon, M., Nguyen, T., Wang, L., Land, S. J., … Ruden, D. M. (2012). A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of Drosophila melanogaster strain w1118; iso-2; iso-3. *Fly*, *6*(2), 80-92.

Cochrane, G., Akhtar, R., Bonfield, J., Bower, L., Demiralp, F., Faruque, N., … Birney, E. (2009). Petabyte-scale innovations at the European Nucleotide Archive. *Nucleic Acids Research*, *37*(Database issue), D19-25. http://doi.org/10.1093/nar/gkn765

Cooper, G. M., Stone, E. A., Asimenos, G., Green, E. D., Batzoglou, S., & Sidow, A. (2005). Distribution and intensity of constraint in mammalian genomic sequence. *Genome Research*, *15*(7), 901-13. http://doi.org/10.1101/gr.3577405

De Ravel, T. J. L., Devriendt, K., Fryns, J.-P., & Vermeesch, J. R. (2007). What's new in karyotyping? The move towards array comparative genomic hybridisation (CGH). *European Journal of Pediatrics*, *166*(7), 637-43. http://doi.org/10.1007/s00431-007-0463-6

Dees, N. D., Zhang, Q., Kandoth, C., Wendl, M. C., Schierding, W., Koboldt, D. C., … Ding, L. (2012). MuSiC: identifying mutational significance in cancer genomes. *Genome Research*, *22*(8), 1589-98. http://doi.org/10.1101/gr.134635.111

DeRisi, J., Penland, L., Brown, P. O., Bittner, M. L., Meltzer, P. S., Ray, M., … Trent, J. M. (1996). Use of a cDNA microarray to analyse gene expression patterns in human cancer. *Nature Genetics*, *14*(4), 457-60. http://doi.org/10.1038/ng1296-457

Deu-Pons, J., Schroeder, M. P., & Lopez-Bigas, N. (2014). jHeatmap: an interactive heatmap viewer for the web. *Bioinformatics (Oxford, England)*, *30*(12), 1757-8. http://doi.org/10.1093/bioinformatics/btu094

Edgar, R., Domrachev, M., & Lash, A. E. (2002). Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Research*, *30*(1), 207-10. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=99122&tool=pmcentrez&rendertype=abstract

Eilbeck, K., Lewis, S. E., Mungall, C. J., Yandell, M., Stein, L., Durbin, R., & Ashburner, M. (2005). The Sequence Ontology: a tool for the unification of genome annotations. *Genome Biology*, *6*(5), R44. http://doi.org/10.1186/gb-2005-6-5-r44

Flicek, P., Amode, M. R., Barrell, D., Beal, K., Billis, K., Brent, S., … Searle, S. M. J. (2014). Ensembl 2014. *Nucleic Acids Research*, *42*(Database issue), D749-55. http://doi.org/10.1093/nar/gkt1196

Forbes, S. A., Tang, G., Bindal, N., Bamford, S., Dawson, E., Cole, C., … Futreal, P. A. (2009). COSMIC (the Catalogue of Somatic Mutations in Cancer): a resource to investigate acquired mutations in human cancer. *Nucleic Acids Research*, *38*(Database), D652-D657. http://doi.org/10.1093/nar/gkp995

Forbes, S. A., Tang, G., Bindal, N., Bamford, S., Dawson, E., Cole, C., … Futreal, P. A. (2010). COSMIC (the Catalogue of Somatic Mutations in Cancer): a resource to investigate acquired mutations in human cancer. *Nucleic Acids Research*, *38*(Database issue), D652-657. http://doi.org/10.1093/nar/gkp995

Futreal, P. A., Coin, L., Marshall, M., Down, T., Hubbard, T., Wooster, R., … Stratton, M. R. (2004). A census of human cancer genes. *Nature Reviews. Cancer*, *4*(3), 177-183. http://doi.org/10.1038/nrc1299

Gonzalez-Perez, A., Deu-Pons, J., & Lopez-Bigas, N. (2012). Improving the prediction of the functional impact of cancer mutations by baseline tolerance transformation. *Genome Medicine*, *4*(11), 89. http://doi.org/10.1186/gm390

Gonzalez-Perez, A., & Lopez-Bigas, N. (2012). Functional impact bias reveals cancer drivers. *Nucleic Acids Research*, 1-10. http://doi.org/10.1093/nar/gks743

González-Pérez, A., & López-Bigas, N. (2011). Improving the Assessment of the Outcome of Nonsynonymous SNVs with a Consensus Deleteriousness Score, Condel. *The American Journal of Human Genetics*, *88*(4), 440-449. http://doi.org/10.1016/j.ajhg.2011.03.004

Gonzalez-Perez, A., Perez-Llamas, C., Deu-Pons, J., Tamborero, D., Schroeder, M. P., Jene-Sanz, A., … Lopez-Bigas, N. (2013). IntOGen-mutations identifies cancer drivers across tumor types. *Nature Methods*, *10*(11), 1081-2. http://doi.org/10.1038/nmeth.2642

Greenman, C., Stephens, P., Smith, R., Dalgliesh, G. L., Hunter, C., Bignell, G., … Stratton, M. R. (2007). Patterns of somatic mutation in human cancer genomes. *Nature*, *446*(7132), 153-8. http://doi.org/10.1038/nature05610

Gundem, G., Perez-Llamas, C., Jene-Sanz, A., Kedzierska, A., Islam, A., Deu-Pons, J., … Lopez-Bigas, N. (2010). IntOGen: integration and data mining of multidimensional oncogenomic data. *Nature Methods*. http://doi.org/10.1038/nmeth0210-92

Hayden, E. C. (2014). Technology: The $1,000 genome. *Nature*, *507*(7492), 294-5. http://doi.org/10.1038/507294a

Hindman, B., Konwinski, A., Zaharia, M., Ghodsi, A., Joseph, A. D., Katz, R., … Stoica, I. (2011). Mesos: a platform for fine-grained resource sharing in the data center, 295-308. Retrieved from http://dl.acm.org/citation.cfm?id=1972457.1972488

ICGC. (2010). International network of cancer genome projects. *Nature*, *464*(7291), 993-998. http://doi.org/10.1038/nature08987

Jones, A. R., Miller, M., Aebersold, R., Apweiler, R., Ball, C. A., Brazma, A., … Pizarro, A. (2007). The Functional Genomics Experiment model (FuGE): an extensible framework for standards in functional genomics. *Nature Biotechnology*, *25*(10), 1127-33. http://doi.org/10.1038/nbt1347

Karolchik, D., Hinrichs, A. S., Furey, T. S., Roskin, K. M., Sugnet, C. W., Haussler, D., & Kent, W. J. (2004). The UCSC Table Browser data retrieval tool. *Nucleic Acids Research*, *32*(Database issue), D493-6. http://doi.org/10.1093/nar/gkh103

Kircher, M., Witten, D. M., Jain, P., O'Roak, B. J., Cooper, G. M., & Shendure, J. (2014). A general framework for estimating the relative pathogenicity of human genetic variants. *Nature Genetics*, *46*(3), 310-5. http://doi.org/10.1038/ng.2892

Lander, E. S., Linton, L. M., Birren, B., Nusbaum, C., Zody, M. C., Baldwin, J., … Szustakowki, J. (2001). Initial sequencing and analysis of the human genome. *Nature*, *409*(6822), 860-921. http://doi.org/10.1038/35057062

Lathe, W., Williams, J., Mangan, M. & Karolchik, D. (2008). Genomic Data Resources: Curation, Databasing, and Browsers. *Nature Education*, *1*(3), 2. Retrieved from http://www.nature.com/scitable/topicpage/genomic-data-resources-challenges-and-promises-743721

Lawrence, M. S., Stojanov, P., Polak, P., Kryukov, G. V, Cibulskis, K., Sivachenko, A., … Getz, G. (2013). Mutational heterogeneity in cancer and the search for new cancer-associated genes. *Nature*, *499*(7457), 214-8. http://doi.org/10.1038/nature12213

Liu, L., Li, Y., Li, S., Hu, N., He, Y., Pong, R., … Law, M. (2012). Comparison of next-generation sequencing systems. *Journal of Biomedicine & Biotechnology*, *2012*, 251364. http://doi.org/10.1155/2012/251364

Liu, X., Jian, X., & Boerwinkle, E. (2011). dbNSFP: a lightweight database of human nonsynonymous SNPs and their functional predictions. *Human Mutation*, *32*(8), 894-9. http://doi.org/10.1002/humu.21517

Liu, X., Jian, X., & Boerwinkle, E. (2013). dbNSFP v2.0: a database of human non-synonymous SNVs and their functional predictions and annotations. *Human Mutation*, *34*(9), E2393-402. http://doi.org/10.1002/humu.22376

Loman, N. J., Constantinidou, C., Chan, J. Z. M., Halachev, M., Sergeant, M., Penn, C. W., … Pallen, M. J. (2012). High-throughput bacterial genome sequencing: an embarrassment of choice, a world of opportunity. *Nature Reviews. Microbiology*, *10*(9), 599-606. http://doi.org/10.1038/nrmicro2850

Manolio, T. A., Collins, F. S., Cox, N. J., Goldstein, D. B., Hindorff, L. A., Hunter, D. J., … Visscher, P. M. (2009). Finding the missing heritability of complex diseases. *Nature*, *461*(7265), 747-53. http://doi.org/10.1038/nature08494

Massie, M., Nothaft, F., Hartl, C., Kozanitis, C., Schumacher, A., Joseph, A. D., & Patterson, D. A. (2013). *ADAM: Genomics Formats and Processing Patterns for Cloud Scale Computing*. Retrieved from http://www.eecs.berkeley.edu/Pubs/TechRpts/2013/EECS-2013-207.html

McLaren, W., Pritchard, B., Rios, D., Chen, Y., Flicek, P., & Cunningham, F. (2010). Deriving the consequences of genomic variants with the Ensembl API and SNP Effect Predictor. *Bioinformatics (Oxford, England)*, *26*(16), 2069-70. http://doi.org/10.1093/bioinformatics/btq330

Merali, Z. (2010). Computational science: ...Error. *Nature*, *467*(7317), 775-7. http://doi.org/10.1038/467775a

Nature. (2014). Code share. *Nature*, *514*(7524), 536. http://doi.org/10.1038/514536a

Ng, P. C., & Henikoff, S. (2003). SIFT: predicting amino acid changes that affect protein function. *Nucleic Acids Research*, *31*(13), 3812-3814. Retrieved from http://www.ncbi.nlm.nih.gov/pmc/articles/PMC168916/

Oinn, T., Addis, M., Ferris, J., Marvin, D., Senger, M., Greenwood, M., … Li, P. (2004). Taverna: a tool for the composition and enactment of bioinformatics workflows. *Bioinformatics (Oxford, England)*, *20*(17), 3045-3054. http://doi.org/10.1093/bioinformatics/bth361

Omberg, L., Ellrott, K., Yuan, Y., Kandoth, C., Wong, C., Kellen, M. R., … Margolin, A. a. (2013). Enabling transparent and collaborative computational analysis of 12 tumor types within The Cancer Genome Atlas. *Nature Genetics*, *45*, 1121-6. http://doi.org/10.1038/ng.2761

Parkinson, H., Kapushesky, M., Shojatalab, M., Abeygunawardena, N., Coulson, R., Farne, A., … Brazma, A. (2007).

ArrayExpress--a public database of microarray experiments and gene expression profiles. *Nucleic Acids Research*, *35*(Database issue), D747-50. http://doi.org/10.1093/nar/gkl995

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011). Scikit-learn: Machine Learning in {P}ython. *Journal of Machine Learning Research*, *12*, 2825-2830.

Pollard, K. S., Hubisz, M. J., Rosenbloom, K. R., & Siepel, A. (2010). Detection of nonneutral substitution rates on mammalian phylogenies. *Genome Research*, *20*(1), 110-21. http://doi.org/10.1101/gr.097857.109

Pruitt, K. D., Brown, G. R., Hiatt, S. M., Thibaud-Nissen, F., Astashyn, A., Ermolaeva, O., … Ostell, J. M. (2014). RefSeq: an update on mammalian reference sequences. *Nucleic Acids Research*, *42*(Database issue), D756-63. http://doi.org/10.1093/nar/gkt1114

Reimand, J., Wagih, O., & Bader, G. D. (2013). The mutational landscape of phosphorylation signaling in cancer. *Scientific Reports*, *3*, 2651. http://doi.org/10.1038/srep02651

Reva, B., Antipin, Y., & Sander, C. (2011). Predicting the functional impact of protein mutations: application to cancer genomics. *Nucleic Acids Research*, *39*(17), e118. http://doi.org/10.1093/nar/gkr407

Rubio-Perez, C., Tamborero, D., Schroeder, M. P., Antolín, A. A., Deu-Pons, J., Perez-Llamas, C., … Lopez-Bigas, N. (2015). In Silico Prescription of Anticancer Drugs to Cohorts of 28 Tumor Types Reveals Targeting Opportunities. *Cancer Cell*, *27*(3), 382-396. http://doi.org/10.1016/j.ccell.2015.02.007

Sanborn, J. Z., Benz, S. C., Craft, B., Szeto, C., Kober, K. M., Meyer, L., … Zhu, J. (2011). The UCSC Cancer Genomics Browser: update 2011. *Nucleic Acids Research*, *39*(Database issue), D951-959. http://doi.org/10.1093/nar/gkq1113

Schadt, E. E., Linderman, M. D., Sorenson, J., Lee, L., & Nolan, G. P. (2010). Computational solutions to large-scale data management and analysis. *Nature Reviews. Genetics*, *11*(9), 647-657. http://doi.org/10.1038/nrg2857

Schadt, E. E., Linderman, M. D., Sorenson, J., Lee, L., & Nolan, G. P. (2011). Cloud and heterogeneous computing solutions exist today for the emerging big data problems in biology. *Nature Reviews. Genetics*, *12*(3), 224. http://doi.org/10.1038/nrg2857-c2

Schwarz, J. M., Cooper, D. N., Schuelke, M., & Seelow, D. (2014). MutationTaster2: mutation prediction for the deep-sequencing age. *Nature Methods*, *11*(4), 361-2. http://doi.org/10.1038/nmeth.2890

Shihab, H. A., Gough, J., Cooper, D. N., Stenson, P. D., Barker, G. L. A., Edwards, K. J., … Gaunt, T. R. (2013). Predicting the functional, molecular, and phenotypic consequences of amino acid substitutions using hidden Markov models. *Human Mutation*, *34*(1), 57-65. http://doi.org/10.1002/humu.22225

Smedley, D., Haider, S., Ballester, B., Holland, R., London, D., Thorisson, G., & Kasprzyk, A. (2009). BioMart - biological queries made easy. *BMC Genomics*, *10*(1), 22. http://doi.org/10.1186/1471-2164-10-22

Smedley, D., Haider, S., Durinck, S., Pandini, L., Provero, P., Allen, J., … Kasprzyk, A. (2015). The BioMart community portal: an innovative alternative to large, centralized data repositories. *Nucleic Acids Research*, *43*(W1), W589-598. http://doi.org/10.1093/nar/gkv350

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., … Lewis, S. (2007). The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature Biotechnology*, *25*(11), 1251-5. http://doi.org/10.1038/nbt1346

Spellman, P. T., Miller, M., Stewart, J., Troup, C., Sarkans, U., Chervitz, S., … Brazma, A. (2002). Design and implementation of microarray gene expression markup language (MAGE-ML). *Genome Biology*, *3*(9), RESEARCH0046. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=126871&tool=pmcentrez&rendertype=abstract

Stratton, M. R., Campbell, P. J., & Futreal, P. A. (2009). The cancer genome. *Nature*, *458*(7239), 719-724. http://doi.org/10.1038/nature07943

Supek, F., & Vlahovicek, K. (2004). INCA: synonymous codon usage analysis and clustering by means of self-organizing map. *Bioinformatics*, *20*(14), 2329-2330. http://doi.org/10.1093/bioinformatics/bth238

Systems and software engineering -- Vocabulary. (2010). http://doi.org/10.1109/IEEESTD.2010.5733835

Tamborero, D., Gonzalez-Perez, A., & Lopez-Bigas, N. (2013). OncodriveCLUST: exploiting the positional clustering of somatic mutations to identify cancer genes. *Bioinformatics (Oxford, England)*, *29*(18), 2238-44. http://doi.org/10.1093/bioinformatics/btt395

Tamborero, D., Gonzalez-Perez, A., Perez-Llamas, C., Deu-Pons, J., Kandoth, C., Reimand, J., … Lopez-Bigas, N. (2013). Comprehensive identification of mutational cancer driver genes across 12 tumor types. *Scientific Reports*, *3*, 2650. http://doi.org/10.1038/srep02650

Tarca, A. L., Romero, R., & Draghici, S. (2006). Analysis of microarray experiments of gene expression profiling. *American Journal of Obstetrics and Gynecology*, *195*(2), 373-88. http://doi.org/10.1016/j.ajog.2006.07.001

Theisen, A. (2008). Microarray-based Comparative Genomic

Hybridization (aCGH). Retrieved August 18, 2015, from http://www.nature.com/scitable/topicpage/microarray-based-comparative-genomic-hybridization-acgh-45432#

Thorvaldsdóttir, H., Robinson, J. T., & Mesirov, J. P. (2012). Integrative Genomics Viewer (IGV): High-Performance Genomics Data Visualization and Exploration. *Briefings in Bioinformatics*. http://doi.org/10.1093/bib/bbs017

Thusberg, J., Olatubosun, A., & Vihinen, M. (2011). Performance of mutation pathogenicity prediction methods on missense variants. *Human Mutation*, *32*(4), 358-68. http://doi.org/10.1002/humu.21445

Trelles, O., Prins, P., Snir, M., & Jansen, R. C. (2011). Big data, but are we ready? *Nature Reviews. Genetics*, *12*(3), 224. http://doi.org/10.1038/nrg2857-c1

Tuna, M., & Amos, C. I. (2013). Genomic sequencing in cancer. *Cancer Letters*, *340*(2), 161-70. http://doi.org/10.1016/j.canlet.2012.11.004

Venter, J. C., Adams, M. D., Myers, E. W., Li, P. W., Mural, R. J., Sutton, G. G., … Zhu, X. (2001). The sequence of the human genome. *Science (New York, N.Y.)*, *291*(5507), 1304-51. http://doi.org/10.1126/science.1058040

Verma, A., Pedrosa, L., Korupolu, M. R., Oppenheimer, D., Tune, E., & Wilkes, J. (2015). Large-scale cluster management at

{Google} with {Borg}. In *Proceedings of the European Conference on Computer Systems (EuroSys)*. Bordeaux, France.

WHO | International Classification of Diseases for Oncology, 3rd Edition (ICD-O-3). (n.d.). Retrieved from http://www.who.int/classifications/icd/adaptations/oncology/en/