Roma Tre University
Ph.D. in Computer Science and Engineering

Big biomedical data modeling for knowledge extraction with machine learning techniques

Eleonora Cappelli

Big biomedical data modeling for knowledge extraction with machine learning techniques

A thesis presented by
Eleonora Cappelli
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Computer Science and Engineering

Roma Tre University
Dept. of Informatics and Automation

April 2020

Advisors:
Prof. Riccardo Torlone
Prof. Emanuel Weitschek

Reviewers:
Prof. Mourad Elloumi
Prof. Aleksandra Swiercz

Contents

Abstract

Background. Over the last ten years biomedical data daily produced by Next Generation DNA Sequencing (NGS) techniques has doubled every seven months. Nowadays genomics plays a relevant role in the field of Big Data, because of the large amount of biomedical data being produced, analyzed, and stored in many public databases. Currently, the storage of this data is performed by many different organizations and their acquisition methods are highly distributed and involve heterogeneous formats.
Methods. In this dissertation the problem of biomedical data heterogeneity is addressed by proposing new standardization methods and pipelines, which permit to easily integrate genomic and clinical data of cancer related to different NGS experiments. Moreover, novel methods for querying them are defined: (i) use cases of the GenoMetric Query Language, a high-level domain-specific query language, are presented to demonstrate the efficiency of the data standardization in terms of information retrieval; (ii) a new data model that minimizes the amount of redundant information is defined, allowing the creation of an Application Programming Interfaces (API) for data retrieval; (iii) methods for discovering and querying large datasets through taxonomy-based methodologies are proposed. Finally, thanks to biomedical data standardization, it is possible to easily apply machine learning techniques for the analysis of genomic data and their interpretation. In particular, knowledge extraction experiments are shown on big biomedical datasets of cancer with promising performance and models.
Results. The main results of the dissertation are new software tools and methods: i) OpenGDC, which allows to automatically standardize and extend

genomic and clinical data of cancer; OpenGDC software is freely available at http://geco.deib.polimi.it/opengdc/, and additionally, a publicly accessible repository, containing homogenized and enhanced data (resulting in more than 1.5 TB) is released;
ii) OpenOmics, which provides a flexible collection of Application Programming Interfaces (APIs), in particular a set of implemented endpoints are available at http://bioinformatics.iasi.cnr.it/api/routes; An ontological software layer that allows users to interact with experimental data and metadata without knowledge about their representation schema;
iii) new software pipelines for gene-oriented data preprocessing are implemented, and a large knowledge base of classification results (datasets, logic formulas, performance, and statistics) obtained by the application of different machine learnings algorithms on a big repository of public available RNA sequencing and DNA methylation of Cancer. iv) CamurWeb, a web service that aims to make the CAMUR machine learning software easily accessible and usable.
Conclusions. The aim of the dissertation is to provide tools for the management and analysis of Big Biomedical Data and to allow the definition of a framework for standardization, querying, and knowledge extraction from clinical and genomic data. The obtained experimental results confirm the soundness of the proposed approaches.

Introduction

In the last decades computer science has given important contributions to healthcare. This fact is also confirmed by the increasing investments for realizing health information systems. The need to improve health services and the diagnosis of many diseases fosters the design and development of new systems able to record clinical history of patients.

Bioinformatics is the subject at the crossroads between biology and computer science; it aims to collect, consult and analyze biological data in order to understand the biological insights of diseases and to identify more accurate diagnoses. The main subject of study of bioinformatics is DNA. In particular the analysis of DNA mapping is the most faced theme, because the study, the therapy and the prevention of many diseases starts from the analysis of the genetic makeup of an individual. The DNA, deoxyribonucleic acid, is the main molecule of genetic information, in all cells. It is composed of four chemical bases: adenine (A), thymine (T), guanine (G) and cytosine (C). Particular DNA sequences determine the genes, which constitute the whole genetic heritage, i.e. the human genome. Genome sequencing is a very important field whose study has started since 70s with the Sanger's method and evolved in recent years with the development of Next Generation Sequencing (NGS), an "high throughput" technique able to extract DNA sequences with high speed and low cost. The NGS techniques are nowadays applied to many research fields, e.g., cancer research, non invasive prenatal diagnosis. Next generation sequencing is the set of sequencing technologies of the nucleic acids which are able to sequence, in parallel, thousands of DNA fragments. These technologies marked a revolutionary turning point in terms of genome characterization with

respect to the first generation DNA sequencing technique (Sanger sequencing), thanks to the ability to generate genetic information millions of times faster and at lower cost. Nowadays NGS techniques allow to sequence the human genome in one day with a 800 dollars cost outputting more than 100 GB of DNA sequences that have to be assembled with bioiformatics algorithms. For these reasons, genome sequencing and the resulting quantity of data produced constitute the main source of "Big Biological Data" processed in bioinformatics.

Bioinformatics provides the tools to deal with large amounts of NGS data produced, with the objective to manage data effectively and to extract knowledge about biological processes. Accessing and sharing genomic data, experiments and results obtained allows to broaden knowledge about diseases caused by genetic alterations or mutations. Therefore, in order to infer reliable results from the data, it is necessary to access many complex datasets and then to extract and integrate large amounts of heterogeneous data. Indeed, NGS data produced by organizations and research groups are often available with different formats and semantics; heterogeneous data integration through complex workflows and pipelines becomes crucial.

Big Biological Data production, management and access are the first steps for the achievement of the final objective, i.e., meaningful interpretation of them. Instruments for data knowledge extraction are provided in this dissertation with computer science, mathematics, statistics, and other quantitative techniques. The "Biological Data Science" paradigm, which aims to apply advanced algorithms and techniques toward understanding biological data [Sch15] is exploited.

In this dissertation three main aspects of the "big genomic data" management are deepened: genomic and clinical data standardization, data accessibility and querying, and finally biological knowledge extraction.

In Chapter 1 NGS technologies are deeply described, and the challenges that the amount of NGS data brings with it are introduced. Additionally the main genomic databases, repositories and data portals are described. In particular, the Genomic Data Commons (GDC), whose data are considered for the following chapters, is deeply investigated.

Chapter 2 deals with automated extraction, integration, extension, and standardization of genomic and clinical data of The Cancer Genome Atlas (TCGA) program from the GDC portal. A standardization pipeline to model heterogeneous genomic data is proposed. Expertimental data and their biospecimen and clinical data are modelled by applying a state of the art data model, the Genomic Data Model (GDM) [CKM+16]. According with the GDM rep-

resentation, NGS experiments are represented by their genomic regions, and the related biological and clinical features, which are described with a set of attribute-value pairs, called metadata. The mapping of these data into a standardized common schema allows supporting the integration and analysis of different types of NGS experiments.

In Chapter 3 the issues about accessability and querying biomedical data is considered, which turns out to be an important step before the analysis process, i.e., knowledge extraction. Different tools for accessing these standardized data are shown. The strong usability of the data model, proposed in Chapter 2, is demonstrated by applying the Genometric Query Language (GMQL) [MPV+15]. Afterward a framework for accessing genomic and clinical data is proposed, which provides a more efficient model of data in a document-oriented no-SQL database and the definition of an Application Programming Interface (API) for fast and effective access to data. Additionally, domain-specific ontologies are exploited and a new ontological software layer is implemented, which allows users to interact with experimental data and metadata without knowledge about their representation schema.

Finally, Chapter 4 deals with the topic of knowledge extraction from NGS data. Here techniques of data preparation are shown and different machine learning algorithms are applied. The goal is to provide a gene-oriented representation of datasets and to extract classification models able to distinguish between tumoral and normal samples.

Section "Conclusions" describes the obtained results from the application of proposed methods, concludes the dissertation and mentions future developments.

In Appendix A bioinformatics, genomics, and the fundamentals of biological sciences are defined. In particular, the components of the human genome are described and how they interact with each other in the process of gene expression.

Chapter 1

Big data in bioinformatics

1.1 Introduction

The attention to Big Data in bioinformatics is steadily increasing, proportionally to the growth of the amount of biological data obtained through sequencing or "omics" techniques. With the advent of Big Science (i.e., the study of storage of huge amounts of data and the knowledge discovery process), Big Data technologies and approaches have been applied to several scientific domains. In particular, with respect to large collections of genomic data, the scientific field of bioinformatics (Appendix A.8) stands out. In the last few years, bioinformaticians are faced with large amounts of biomedical data, and need tools and techniques in order to handle them. We consider Big Biological Data, because of their volume (amount of data generated), variety (type of data generated), speed (speed of data generation), variability (data inconsistency) and truthfulness (quality of the data acquired) [LC14]. Sequencing data is the most spread example of Big Data in the field of bioinformatics, because of the advancement in next generation sequencing (NGS) technology. The concept of Big Data associated with biological and medical databases becomes even more realistic from a personalized medicine perspective, where each one can request an analysis of his DNA (Appendix A.3) to learn either about his genetic predispositions to diseases, or his personal attitudes, or his susceptibility to drugs. Genomics, the science that studies the genome of living beings, is based on bioinformatics for processing and displaying the enormous amount of data it produces. In fact, the sequencing of the human genome (Appendix A.1) has produced large amounts of information and bioinformatics aims to manage

them in two ways. On one hand, it must provide automated and cost-effective methodologies for data analysis, and on the other it must optimize the search algorithms to improve data management and accessibility.

In this chapter, we describe the NGS technologies and the data produced, the databases that collect these data, and the challenges that bioinformatics have to face to manage and analyze them.

1.2 Next Generation Sequencing data analysis

In the Big Data era the human life sciences are playing a leading role, thanks to the recent technological and experimental advances that have led to the increase of biological data. The definition "Big Biological Data" is mainly linked to "omics" fields, such as genomics, epigenomics, proteomics or metabolomics, sciences that study molecular biology. Today, the high-throughput technologies, like NGS, have revolutionized the sequencing of genomes, producing large quantities of DNA and RNA data [Hay14, She14, WSF+14, SJ08]. These data represent the main source of Big Biological Data, providing the development of omics fields and bioinformatics methodologies.

An important task of bioinformatics is to support the analysis of the complex biological data with effective and efficient computer science techniques.The main aim is to support the genomic analysis by means of latest and innovative information technologies. In particular DNA sequencing is one of the most relevant topic in the field of genomic and it leads to other correlated topics: (i) genome's assembly starting from million of DNA fragments [CPT04, CP08]; (ii) DNA fragments' alignment along a reference genome [KTN04, LHW+09, HME12]; (iii) analysis of variants between genomes (Variant calling) [PKP+14, KLW13]; (vi) annotation of genomes [HFG+12]; (v) analysis of gene expression (Appendix A.2) by the sequencing of transcripts (RNASeq) [OM11]. Once the sequencing has been completed, the data is analyzed with a bioinformaticsanalysis pipeline that is completely automized and made up of three steps: alignment, variant calling, filtering and annotation.

The outcome of the sequencing is composed of little fragments of DNA called reads, that must be assembled to obtain gene sequences. *Assembling* the reads is a computationally difficult task since the part of the gene where the fragments come from is unknown. Thus the most used technique consists in *aligning* the reads along a reference sequence. Once the sequences of the sample's gene (Appendix A.6) are reconstructed by aligning the reads, the subsequent step is to detect the points in which the sample's gene differ from the reference

sequences of the human genome stored in the databases. This phase is called *variant calling* and it is carried out through specific softwares. The precision of the outcomes obtained in this phase is a very important parameter to consider. In particular the probability to identify a variant that can have a pathogenic effect depends on the quality of the outcome of the variant calling procedure. The variant with pathogenic effect is also known as disease mutation. The variants obtained are thousands thus a further phase is needed. In this phase the variants are filtered and *annotated*, that is they are stored in the databases with all the information needed to describe them.

Thanks to next generation sequencing techniques, the number of sequenced genomes has considerably grown up and this has lead to the need of further bioinformatics analyses of these sequences that are suitable to the amount of data involved. The genome sequencing consists in detecting the location, the structure and the functionalities of the composing element. The goal is to determine the order of the nucleotides and thus of the four nitrogenous bases A-C-T-G that made the DNA up. Since the 70s and 80s different sequencing techniques have been developed. One of the most common is the Frederick Sanger's strategy [SNC77] also called chain terminator method or Sanger's method. It is still largely in use but it is very challenging, even if very effective. It is related to the use of radioactive enzymatic substances since it requires the use of an enzyme. This technique is grounded on the use of modified nucleotides (dideoxynucleotide, ddNTPs) in order to interrupt the synthesis reaction in particular position and thus obtain a fragment.

The Sanger method had been considered as a reference standard in diagnostic molecular genetics for many years, but the uprising demand of low sequencing leads to the development of new technologies highthroughput sequencing (also known as Next-generation sequencing). Next generation sequencing represents a paradigm shift in the clinical diagnostic field [Sch07, Met10]. It allows to parallelize the sequencing procedure and make thousands or even million of sequences simultaneously. The Sanger's technique is still in use nowadays, however the new generation sequencing technologies are able to analyze big amounts of sequences and have the great advantage of being cost-effective, lowering the costs of DNA sequencing with respect to the standard methods. Differently from the traditional Sanger's method, the NGS is also called high-throughput sequencing since it allows to sequence lots of fragments in parallel. Comparing traditional and new generation techniques the amount of bases that comes out from the analysis is considerably higher with the NGS. In fact the daily outcome of a capillary sequencer with the Sanger's method is in the order of thousands of bases, the output of high-throughput sequencing machines is in

the gigabase order. However it has to be considered that these new techniques have some drawbacks with respect to the traditional methods. In particular they reduce the reads's length (50-400 nucleotide compared to 1000 nucleotide of Sanger's read) and they also are less precise in reading the bases. The NGS can be done by different sequencing systems and there are different kind of NGS sequencing platforms commercially available, which use different techniques. All the different technologies included in the NGS techniques have three common steps (Figure 1.1):

1. *The preparation and immobilization of the DNA, the so-called sequencing library.* In this step the DNA sample undergoes a fragmentation procedure. Then some adaptor are added to the fragments. The adaptors are particular sequences that allow the fragment's immobilization on the support where the sequencing reaction will take place. The length of the fragment tied to the adaptor depends on the kind of the analysis and the platform or NGS technology used.

2. *Amplification reaction.* In this phase a fragment from the sequencing library is incorporated in a microscopic water bubble together with some tiny spheres, the so called enrichment beads, to which the adaptors can bind. The amplification reaction (PCR) [IGSW12] takes place in this water bubble, in which the DNA fragment is amplified many times.

3. *Sequencing reaction.* The sequencing consists in adding a solution with a specific nucleotide to the immobilized DNA. If the nucleotide is complementary to the sequence, it is incorporated. Then the molecular event is recorded with an imaging system that depends on the used technology.

The sequencing procedure provides a lot of complementary copies of each fragment, that are the reads. The NGS sequencing is considered to be effective if the obtained reads for each fragments are 30 at least. Gathering the reads is needed to achieve a signal that is strong enough to be detected and to cover the reads signal containing mistakes. In fact the new generation sequencing can be inaccurate but making an high number of reads can soften the error signals introducing clean ones. The output of a sequencing machine is therefore composed of millions of reads, i.e. short character sequences, for a human genome 100 GB for plants 10000 GB (10 TB).

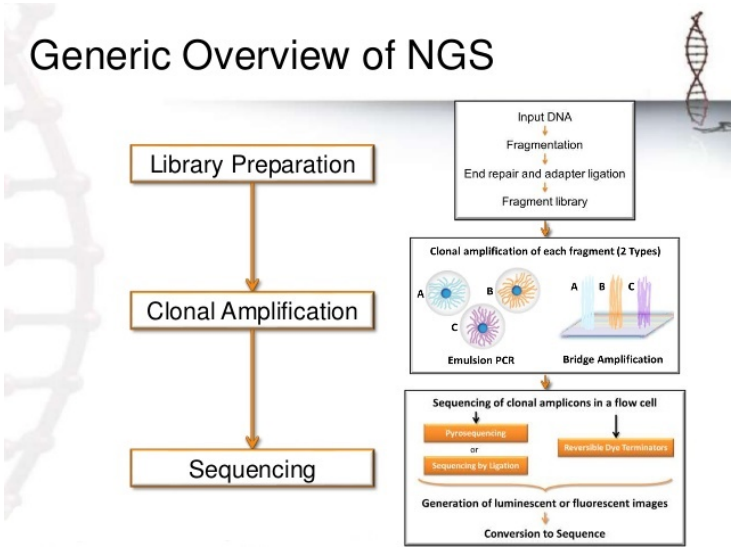

Figure 1.1: Next Generation Sequencing steps.

1.3 NGS experiments and applications

This abundance of data allow to perform analyses on the genetic makeup of many human subjects, studying the predisposition to diseases like cancer [Mar08, MDT11, KSL+13, ALK17]. NGS techniques are not only applied to DNA sequencing [MHB+10], but also to other types of experiments, e.g.: transcriptome profiling (RNA sequencing) [MWM+08, LD11], microRNA sequencing (miRNA-seq) [ZC03], Copy Number Variation (CNV) [CPR+10b], and characterization of the epigenome or chemical changes in the DNA (DNA methylation) [Bir85, Bir02] (Figure 1.2). The DNA methylation and RNA sequencing experiments have been proven to play an important role in knowledge discovery of cancer [Jon86, Ehr02, Bay05, LLL+15, PMP+10, EHG+13, YTS+12, DCHW16, XWOZ11, CFW18]. Indeed NGS data are actually use in early disease diagnosis especially in cancer [WV18, CL13, WCCF16] because with this sequencing appoach it is possibile to identify mutations as these technologies are able to sequence the whole genome, exome or transcriptome.

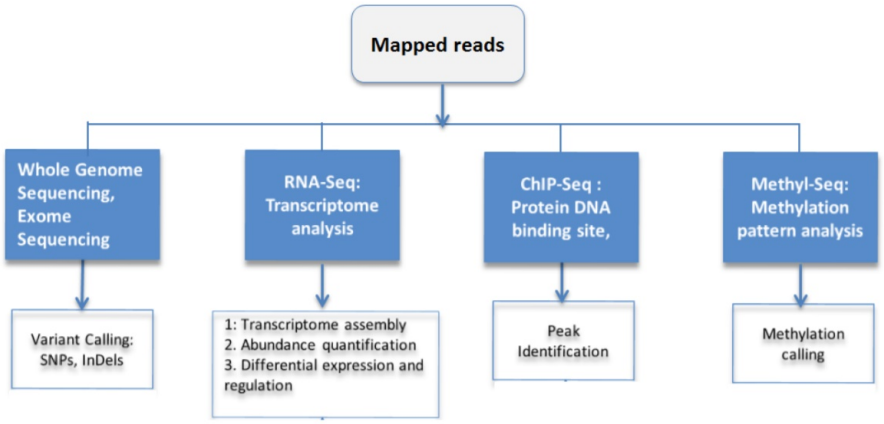

Figure 1.2: NGS Applications.

RNA sequencing is a transcriptome (see Appendix A.2) analysis and quantification technique. It is used for detecting the presence and the quantity of RNA (see Appendix A.4) in a biological sample at a given moment in time During a RNA-seq experiment the gene expression is measured in terms of counts that is the number of reads mapped on the genes of a genome or transcriptome reference for quantifying the transcriptome abundance. In fact the reads alignment phase over the reference is a very critical aspect to consider since the position of the genome where the reference is identical to the read will never be univocal. This is due to the fact that the reference is never perfectly equal to the biological sample. Thus in this phase the reads' positions over the reference are defined and it is possible to count the number of aligned reads for each gene. This measured amount is representative of the gene expression's level. Two main methods for measuring gene expression are used in practice: the first approach is based on the Reads Per Kilobase per Million mapped reads (RPKM) [MWM+08] method for quantifying gene expression by RNA sequencing data, normalizing for the total length of the reads and for the number of

reads sequenced. Another method (also known as RNASeqV2), uses a combination of two algorithms, MapSplice [WSZ+10] to make the alignments and RNA-Seq by Expectation-Maximization (RSEM) [LD11] to perform the quantification. Newer versions of RSEM call the *scaled estimate* value Transcripts Per Million (TPM) [WKL12], independent from the transcript length. Furthermore, several other normalization techniques are available. The *Fragments Per Kilobase per Million mapped* (FPKM) [TWP+10] computes the expected fragments (pairs of reads) per kilobase of the transcript per million sequenced fragments. The RPKM value is really close to the FPKM value, indeed, if all mapped reads are paired, the two values will be coincident. However, the latter is able to handle a higher number of reads from single molecules.

The RNA sequencing techniques are mainly made of the three steps already reported in the previous section, and they characterize these new technologies. It has to be noticed that the available sequencers are not able to sequence directly the RNA, but only the DNA. In this vein, a further phase has to be added after the the fragmentation. This supplementary phase is the reverse transcription and consists in synthesizing a complementary DNA filament (cDNA) starting from an RNA template. For further details about RNA sequencing we point the reader to [OM11], where the authors perform a comprehensive overview of this NGS technique.

miRNA sequencing. MicroRNAs (miRNAs) are small single-stranded noncoding RNA molecules found in the transcriptome of plants, animals and some DNA viruses. They are mainly active in the regulation of gene expression at the transcriptional and post-transcriptional level. Dysregulated miRNAs play a role in diseases such as cancer, through the regulation of onco- and tumor suppressor gene expression. The development of high-throughput profiling methods, led the identification of miRNAs as biomarkers for cancer classification, response to therapy, and prognosis [BGT+07, IFL+05, CPW11].

Copy Number Variation is a variation in the number of copies of a given genomic segment (DNA fragment) per cell, with respect to a reference sequence. They can be classified into: Copy Number Loss or Deletion, the number of copies of a given region is less than the reference sequence, and Copy Number Gain or Duplication, the number of copies of a given region is greater than the reference sequence. Thanks to technological progress in sequencing, the number of CNVs is increased exponentially [DZDW13, WNY14]. The Database of Genomic Variants (DGV) [MZY+13] has collected and edited more than

2,000,000 of CNVs that globally map in about 200,000 chromosomal regions (Copy Number Variation Regions). The 70% of CNVRs are affected by Copy Number Loss, the 13% from Copy Number Gain.

CNVs cover a large and important slice of human genetic variability. The main challenge linked to CNVs is the evaluation of which CNVs are neutral and which instead affect vital biological functions and therefore result in a pathology [SMLS+14, MST+10].

Somatic Mutation. A mutation is an accidental, random and heritable modification which involves a change in the nucleotide sequence in the DNA, caused by an error in replication process, occurring often due to environmental conditions. Somatic mutations are changes in genes that affect cells in tissues such as skin or blood, and are not transmitted to descendants. In tumor genetics, somatic mutations specifically refer to mutations that originate in tumors (which are not present in healthy tissues). Such mutations are often responsible for leading to tumor growth. Next-generation sequencing is becoming more widely adopted as a valuable method for somatic mutation analysis in cancer, in order to characterize the mutations and identify biomarkers that are prognostic or predictive [BDF+04].

DNA Methylation. In the next-generation sequencing technologies era, techniques for epigenetics (Appendix A.7) are growing. In cancer research, sequencing evolve with the aim to find novel biomarkers, factors of prognosis and prediction, and targets for achieving personalized treatments [SRAV+16]. DNA Methylation is one of the most studied epigenetic modification in human cells. The changes in DNA methylation patterns are crucial in the development of diseases such as multiple sclerosis, diabetes, schizophrenia, and many forms of cancer [LMT+10, TAK+12, MTK+08, YGZ16, ZLW+17, LTN+17].

Methylation is a biochemical modification that involves the addition of a methyl group in correspondence of carbon-5 in cytosine. This happen almost only in the dinucleotide CpG, that is the cytosine followed by a guanine [Bir02]. This phenomenon is so common in DNA that it can be assessed that among all the CpG islands, the 80% of all them is methylated in mammals [JB04]. CpG islands, are preferentially localized to the promoter of the many genes, particular DNA regions where gene transcription begins. This epigenetic modification is associated with the transcriptional repression of a gene, therefore, when present, it is an epigenetic mark and it prevents the expression of the gene (inactive gene), i.e. the inactivity of the transcripts of the promoter gene.

We can speak of hypo- or hyper-methylation, which regulate the turning on or off of genes that, for example, act as oncosuppressors.

In the latest years the rising interest in DNA methylation leads to strong changes in the analysis method and consequently in deep modification in the DNA sequencing technologies. Many methods used in the methylation analysis take advantage of the high quality and sensitivity of the NGS. The DNA methylation experiment consists in deep sequencing of bisulfite-treated DNA. It can be obtained as the covalent modification of cytosine bases at the C-5 position, generally within a CpG sequence context. The Bisulfite Sequencing technique (BS) turns the non methylated cytosine into uracil, during the preparation of the sequencing library. After the amplification reaction, the converted bases are detected as thymine in the sequenced data, then the reads count is used to determine the percentage of methylated cytosine in a CpG island. This measure is called *beta value* [DZH+10]. The beta value is defined as the ratio of the methylated allele intensity and the overall intensity (i.e. the sum of methylated and unmethylated allele intensities). It is worth to note that beta value is a measure in the range of 0 – 1, where 1 represents full methylation and 0 no methylation at all. For more details about the DNA methylation experimental techniques the reader may refer to [PE10, HER10].

1.4 Big biological databases

One of the most important task of bioinformatics is to define and maintain a variety of databases, where the biological information is collected and annotated with all the additional data needed to understand its functionalities [HCF+08]. The databases allow to better understand the bioinformatics field and to manage and to store different kind of information such as bibliography, nucleotide and amino acid sequences and protein structures.

The use of bioinformatics portals is the easiest way to get acquainted with this field. In fact they allow to consult many different databases. The NCBI (National Center for Biotechnology Information), created in 1988 in the United States, is the biggest bioinformaticsportal and it contains 35 databases that can be consulted simultaneously in an integrated way, using Entrez [TKMO99], a text search engine. Some of the databases containing nucleotide sequences are:

– GenBank, created in 1983 and managed by NCBI. It has a extremely fast growth doubling its size every 18 months. It contains up to 116,5 million sequences and 112,3 billion bases [BKML+08];

- EMBL Nucleotide Sequence Database, also known as EMBL-Bank [SBvdB+02];
- DDBJ (DNA Data Bank of Japan) whose databases contain more than 110 million sequences each [MKK+15].

These are the largest databases available. GenBank contains sequences obtained from 250.000 different organisms. However the annotations are scarce, so the sequences description is inaccurate and there can be multiple voices regarding the same genes. Other databases available, such as RefSeq (Reference Sequence) [OWB+15], managed from NCBI, contain a smaller number of sequences, but they are better annotated and the choice of the included informations is based on quality rather than quantity.

The amount of information contained in the database and the variety of terminologies used to describe genes and proteins, make the research of these data very difficult. Thus a uniform terminology is needed to make the interrogation of the different database easier. In this context the project Gene Ontology (GO) [Con14] has been created. It is a recognised and shared ontology, that can be consulted from a database. GO assigns to each gene attributes that determine its functions, the biological processes in which they participate and the cellular components.

A database to refer for the study of cancer is The Cancer Genome Atlas (TCGA) [WCM+13]. This project aims to catalog the genetic mutations responsible for cancer, applying the NGS to improve the ability to diagnose, treat and prevent cancer through a better understanding of the genetic basis of this disease. This database contains the genomic characterization and analysis of sequences of 33 types of tumors, including 10 rare tumors. Patient samples are processed through different types of techniques such as gene expression, methylated DNA and microRNA profiling, and exon sequencing of at least 1,200 genes. Additionally TCGA collects and analyzes high quality cancer samples and makes the following data available to the research community:

- clinical information on the patients participating in the program;
- samples metadata (for example the weight of the sample, etc.);
- histopathological images of portions of the sample.

On July 15th 2016, the TCGA data portal was officially closed, which since 2006 made available to researchers numerous genomic and clinical data of affected patients. With the NGS technologies and the increase of data, the

TCGA turned out to be inefficient in storing methods and in data extraction methods by researchers.

Thus the GDC (Genomic Data Commons) [JFGS17] project was born to collect and standardize all data produced not only by the TCGA project, but also by other research institutes. The GDC is born from an initiative of the National Cancer Institute (NCI) with the aim of creating a unified data system that can promote the sharing of genomic and clinical data among researchers. The GDC allows access to a high quality set of data derived from programs supported by NCI, and recommends guidelines for organizations providing personal data sets. High quality data is guaranteed by a list of procedure that the GDC strictly observes:

- **Maintenance of high quality samples of tissues.** The GDC gets most of the data from the previously listed NCI programs; these guarantee high quality because the only accepted tissues that have annotated sources and that have been subjected to rigorous quality controls throughout the entire course of processing. For organizations not supported by the NCI, on the other hand, the GDC provides recommended collection strategies, and before accepting the data, submits the samples to examination to make sure they adhere to the high quality standards used by the BCR (Biospecimen Core Resource) .
- **Implementation of data validation procedures.** Data validation is performed both on data imported from NCI programs and on data sent by external organizations; The data is made available by the GDC Portal only if they pass the validation.
- **Ensure the production of reliable and harmonized derivative data.** The GDC uses the genomic sequence data available to create derived data such as somatic DNA mutations, tumor gene expression, and copy number variation. The bioinformatics pipelines described in the GDC Data Harmonization are developed with the continuous contribution of experts from the cancer genomics community. Pipelines are implemented using techniques that make them reproducible, interoperable on multiple platforms, and shareable with all interested members of the community. GDC receives all the pipeline suggestions, and keeps them constantly updated by replacing old tools and technologies to keep up with new discoveries.

The GDC distinguishes between open access data and controlled access data; open access data do not require authorization and are generally high-

level genomic data that cannot be individually identified, therefore aggregated data both clinical and biological samples; data with controlled access require authorization to access it and are generally individually identifiable data such as genomic sequencing data, and some clinical data.

1.5 NGS data Integration

In Section 1.3 and 1.4 we described the different NGS experiment data types and diverse biological databases for storing them. This assumption allow us to introduce the concept of NGS data integration, which represents one of the main challenges of bioinformatics. We define *NGS data integration* the procedure of joining different experiments (possibly extracted from heterogeneous databases) sharing common features (e.g., same disease / patient under study) in order to extract knowledge. The aim of integration is to aggregate genomic data in an unique schema that provides querying capabilities for retrieving data from a multitude of heterogeneous experiments and databases.
Heterogeneous data are the first problem of NGS, because the structure of data is different in diverse experiments and can be different in diverse databases. Therefore, the term integration in NGS data can have different meanings [GCAM+14]. On one hand, we consider integration for a need to have a uniform language that facilitates the access to different genomic databases. On the other hand data heterogeneity is caused by the experiment types and by the information that they bring. It is worth noting that dis-uniformity of the data schema is present not only when considering different databases, but also when dealing with a single one. We distinguish four conditions, where NGS data integration can be performed: (i) different databases represent the same NGS experiment (e.g. RNA-Seq) with different data schemas; (ii) different experiments (e.g., DNA methylation and RNA-Seq) in distinct databases; in this case there are two different data schemas, because the experiments need a different representation, but no standardization of the schemas is defined that allows the access to these experiments; (iii) the same problem exists even in the same databases, which contains different experiments and different data representation schemas. Finally, we consider an ideal case (iv) where a previously defined schema standardization allows to integrate different experiments that come from different databases or from the same database, and it allows also to provide interoperability between the same experiments but with different schemas. An example of this type of standardization is provided by [MKPC16] with the Genomic Data Model (GDM) that supports many NGS formats

Data integration (i.e., providing a unified access to heterogeneous and independent data sources as a single source) is the key problem to allow everyone to store, organize, access, and analyze the information available on the web. The Heterogeneous Database Systems (HDBS) try to unify these databases providing conceptual schemes that solve the heterogeneity of representations and providing querying capabilities that aggregate and integrate distributed data, in order to guarantee a complete transparency. The process of heterogeneous databases integration can be defined as the creation of a single querying interface for the data collected and stored in a multitude of heterogeneous databases [DGLLR07, LMMS+07]. The features of an HDBS are summarized in [Has00]. These systems are very similar to the Distributed Database Systems (DDBS) as described in [SL90], i.e., a set of multiple logically interconnected databases distributed through a computer network. The idea of both approaches is to provide a common interface to the data stored in different physical locations. The DDBS implement the same language and querying data model of the HDBS and these use the same software for data management of distributed databases. Moreover, in DDBS, the fragmentation of data is designed to achieve advantages in terms of efficiency and autonomy of distributed computing.

Other important elements in the integration of heterogeneous data are expressed by the variety with which similar data are represented in different databases. This multitude of schemes is called representational heterogeneity (RH). The most general type of heterogeneity stems from the data: to aggregate data from relational, hierarchical, object-oriented, and flat-file databases into a single representation is the first activity in the integration of schemes. However, although different systems use the same model, such as the relational one, significant differences remain in terms of the representation of heterogeneity with regard to the structures, content, and semantics. Several efforts have been made on NGS data formats and standards. The authors of [EGFF16] provide the reader with an overview of the most widespread data formats for NGS and describe a set of standardization approaches for them. In [TKMO99] the NCBI Entrez search and retrieval system used at the National Center for Biotechnology Information to access distributed heterogeneous data is described. Also the authors of [SPB+15] present a text search engine to access data resources in the European Bioinformatics Institute (EMBL-EBI) and to help understand the relationship between different data types. Other implementations for bioinformatics data integration include retrieval systems like SRS [EUA96], integration tools for information fusion such as BioData Server [FHL+02], federated databases (BioKleisli [DOTW97]), multi-databases (TAMBIS [SBB+00]) and data warehousing systems (BioWarehouse [LPW+06]). Despite of several ef-

forts made in this direction [MPGC14, MMBR+14], many problems remain unsolved. The integration of genomic data involves multiple fields, i.e., bioinformatics, statistics, data mining, and classification. The integration of different types of NGS experiments may offer additional knowledge about a disease like cancer [ZSX+15].

1.6 Big data challenges in bioinformatics

The growth of biological data has led biologists and bioinformatics to face new challenges, and to approach them served by methodologies, tools and technologies typically used to handle Big Data.
Genomics and bioinformatics technologies have allowed the development of knowledge in the biomolecular field, expecially thanks to the enormous amounts of data analyzed. The analysis of these data is possible when they are cataloged, stored and processed in large collections, databases that make them available. As described in Section 1.4, there are many biological databases that provide public access to genomic experiments. TCGA is an important example considering the study of cancer in the genomic field. Through its data portal it offers access to clinical information, to the genomic characterization of tumor genomes of over 11,000 cases involving about 30 different types of cancer.

Data modeling and querying Data management problems are inevitable because of the big amount of data, different platforms, different formats, modeling and storing of genetic data, data accessibility. Therefore the critical points in this field are not only characterized from the massive amount of data, but also from the heterogeneity of both the NGS machines that generate them, the repositories that manage them, and the type of generated data.

In this scenario, a new generation of computer systems, data formats and languages for querying heterogeneous data, which ensure scalability and performance even in a distributed environment, are required. A data model must be defined, supported by querying, research and analysis systems of genomic data.

Biological data come from many heterogeneous sources (Figure 1.3), thus in order to make them reliable and understandable, they must be integrated with clinical data that represents the phenotype of each individual, their origin must be monitored, and their privacy must be protected.

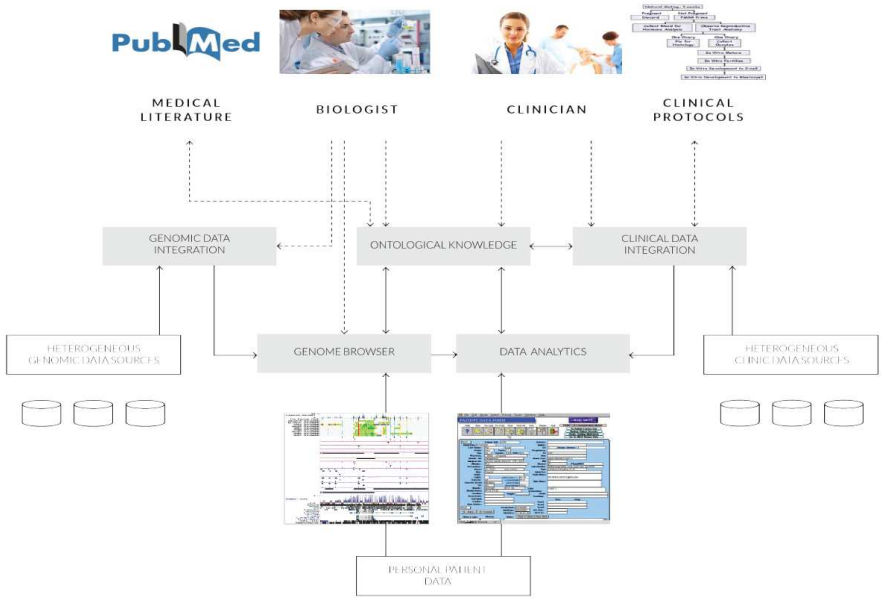

Figure 1.3: Distributed Heterogeneous data.

Additionally it is necessary to define a unique and global platform for the effective storage, search and retrieval of genomic data, with the aim of linking and organizing genomic data spread around the world. The data must be automatically standardized in order to be usable through a unified platform.

The creation of a reference standard with the aim of storing all these data, makes querying easier and allows a more effective analysis. In particular the data can be considered to be heterogeneous due both to the sources they come from and the kind of experiment they represent.

A considerable contribution is given by the **GeCo** [CBC+17, KGM+17] (**Data-Driven Genomic Computing**) project, which has defined a single and global platform for the effective storage, the research and recovering of the genomic data by means of distributed computing (we refer to http://bioinformatics.deib.polimi.it/geco for further details). The GeCo project has changed the paradigm of genomic informatics with the goal of connecting and organizing genomic data that are spread all over the world. The main components of this platform are the data modeling, *Genomic Data Model (GDM)* [CKM+16], and the query language,

GenoMetric Query Language (GMQL) [MPV+15]. The GDM defines a gold standard for storing and dealing with Big Biomedical Data, i.e., genomic and clinical data. It provides abstractions for the samples DNA regions and for the metadata that describe its characteristics, in order to simplify the data format. It is based on the idea of genomic region, that can be compared to million of other regions, and it describes the samples through both the genomic regions and the metadata linked to them. The GMQL uses this data model (GDM) and it support an efficient query elaboration level on thousand of data experimental samples. GMQL is structurally consistent with the traditional database management techniques, but it also aims improving the interaction between biologists and biomedical data. GMQL algebraic operations are designed for the management of genomic data, specifically for the bioinformatics domain. It is a declarative language and its operators admit parameters based on metadata or on the attributes of the schema of the genomic regions. In particular this last operations are related to the distances between the genomic regions.

The GDM allows the integration of multiple heterogeneous data sets from different sources and GMQL computes massive operations on genomic data, which take into account regions, relative positions, and distances.

Data interpretation. In the interpretation of genomic data coming from new generation sequencers, machine learning techniques stand out. The machine learning algorithms have deeply transformed the diagnosis processes and the therapies of many diseases, especially cancer [Saj06, CHH+17]. The complexity of the biologic phenomena leads to a variety of the data types produced by the new generation sequencers. However thanks to this high availability of different types of NGS data, coming from numerous genomes of different individuals, it is possible to examine many different genomic features simultaneously, in order to characterize their functional role and to clarify genetic and epigenetic phenomena. The analysis of these data is effective only if different kind of data are considered together at the same time, in other words the study of the data is useful if it considers the whole heterogeneity of the genomic experiments [SBT+19]. In this context bioinformatics has important tasks to understand the complex biological mechanisms: the integration and the analysis of genomic data.

Eventually, although storing, querying and management are very important steps in the analysis process, in the latest years the interpretation of genomic data has turned out to be one of the most challenging topic within bioinfor-

matics.

1.7 Conclusions

The DNA molecule that contains precious information to create and model a living organism is considered one of the main emerging characters of the Big Data world. Genomics and new generation sequencing technologies have deeply transformed scientific research and the amount of data generated with NGS is constantly growing at an incredible rate, also as a result of the lower costs. On the grounds of these consideration, the Big Genomics Data age has been defined and within this context the management, the storage and the sharing of this huge amount of data have become a very important chokepoint. Acquiring and sharing the genetic data from million people has the important goal to realize new reliable diagnosis models for specific diseases as well as develop new therapies suitable for each patients elaborated on the grounds of genetic heritage of the patient himself. Maintaining such a big amount of data clearly needs the use of proper systems. Thus the scientific world takes advantage of the Big Data field, with the goal to manage biomedical data in an effective way, using technologies and algorithms suitable in this context. The genomic analysis requires suitable algorithms that can sustain the huge amount of data generated by NGS. These algorithms come from the combination of both big data technologies, and the analysis of sequencing data techniques.

But, before developing new sophisticated computational methods for data analysis and applying machine learning techniques for data interpretation, bioinformatics has to deal with the integration of complex heterogeneous genomic data.

In the field of cancer, the Genomic Data Commons (GDC) addressed the issue of integrating NGS data provided by different sources, defining restrictive procedures and schemas for representation of each NGS experiments. GDC partially solves the NGS data integration, aggregating data from different databases and representing the same NGS experiment with unique data schema. The GDC experiments are standardized depending on NGS data type, but NGS data from different experiments are provided in different data formats and schemas. A new level of standardization is necessary to allow the integration of several NGS experiments and to connect more biological information before applying a learning algorithm.

Chapter 2

Automated biomedical data standardization

2.1 Introduction

Thanks to NGS techniques, different types of experimental data are produced, whose storage and analysis can be very demanding [OM11, ZJ10, AKMB+09]. More and more often researchers have to face with Big Biological Data [Bel14, LWGZ16], frequently lacking of integrated data models and accessible schema representations. Thus, storing, retrieving, integrating, comparing, and analyzing heterogeneous biomedical data becomes a major challenge. Therefore new models and methods to easily access, integrate, and search them effectively are needed. In cancer research, several organizations are involved in the collection, management, and publication of genomic and clinical data. In particular, the Genomic Data Commons (GDC) supports several programs and defines bioinformatics pipelines: it provides clinical / biospecimen supplements and genomic data harmonization procedures related to DNA-sequencing [TKR+10], RNA-sequencing [MWM+08, TWP+10], miRNA-sequencing [ZC03], Copy Number Variation [CPR+10a], and DNA-methylation [BBT+11]. The processed data is publicly available through the GDC portal, which deals with different cancer programs, and also through its application programming interface (API), (as described in Section 1.4).

In this chapter, we enhance GDC harmonization by applying a state of the art data model, the Genomic Data Model (GDM) [CKM+16] composed of two components: the genomic data, in Browser Extensible Data (BED) format, and

the related metadata, in a tab-delimited key-value format. We extend the GDC genomic data with information extracted from other public genomic databases (i.e., GENCODE, HGNC, miRBase, and NCBI). For creating metadata, we implement automatic procedures to recognize redundant clinical / biospecimen supplements that are present on the two different sources of GDC (i.e., data portal and API). Moreover, we developed and release the OpenGDC software [CCB+18], which is able to extract, integrate, extend, and automatically standardize genomic and clinical data of The Cancer Genome Atlas (TCGA) from GDC.

2.2 The Genomic Data Model

The Genomic Data Model (GDM) represents each genomic sample through two fundamental concepts, the genomic regions and their metadata. The genomic regions are described by coordinates and some high-level properties. For example to represent DNA-sequencing are not included DNA sequences, but rather they are storing some properties, such as the sequences involved in mutations and their categorization. Conversely, the metadata describes the biological and the clinical properties associated with each sample, not specifically related to the genomic regions. The metadata associated with the sample are extremely heterogeneous and they are represented as attribute-value pairs. They include the experiment type, the sequencing and analyzing method that have been used, and other patient-related information. Each pair of files is part of a dataset that has the same features: dataset is a collection of samples with the same region scheme. Therefore datasets can be considered as homogeneous collections of samples, generally produced within the same project with the same technology and the same tools [CKM+16].

A sample is defined as $< id, \{ < r_i, v_i > \}, \{ M_j \} >$:

- id, is the sample identifier;
- r_i, is the $i-th$ genomic region, or portion of the genome defined by four coordinates $< chr, left, right, strand >$. chr is typed string and represents the chromosome, $left$ and $right$ are the extremes of the region along the DNA coordinates and are typed integer, $strand$ is typed string and symbolizes the reading direction of the DNA, and can therefore assume $+$ or $-$ values, and $*$ if the direction is not specified;
- v_i, is the $i-th$ value associated with the $i-th$ region of the sample. Indeed each region is a pair of coordinates r_i and values v_i, which are

attributes that describe the properties of the regions and can be of any types: boolean, string, integer, long, etc.;

– m_j, is the $j-th$ attribute-typed value pair, typed string, and is part of a collection of pairs that compose the sample's metadata.

The model allows the integration of heterogeneous data coming from different sources, standardizing them with a single data representation.

2.3 The transition from TCGA to GDC data portal

The Cancer Genome Atlas (TCGA) is the most relevant project within GDC, collecting genomic and clinical data of 33 different tumor types of over 11000 patients [LLH$^+$18].

TCGA data were available at its own portal until late 2016, but since early 2017 they were migrated to the new GDC portal, resulting in a major change of genomic and clinical / biospecimen formats and schema. In the GDC portal, experimental data (i.e., DNA-sequencing, RNA-sequencing, miRNA-sequencing, Copy Number Variation, and DNA-methylation) are produced from harmonization procedures applied on different analysis strategies, improving the quality of data available at the old TCGA portal. Indeed, GDC provides a programmatic access to interact with these harmonized data through APIs, e.g., to obtain aliquot UUIDs that identify uniquely GDC experiments. The harmonization procedures provide standardized and comparable data, depending on the type of NGS experiment, regardless of the program which was used in the generation.

For what concerns metadata, clinical / biospecimen supplements were represented in an unstructured format in the TCGA portal; conversely, GDC introduced a new structured data model (i.e., the GDC Data Model). The transition is however still incomplete: GDC provides some relevant clinical / biospecimen information in the old unstructured format and some other in the new format. Correspondingly, GDC exposes two different methods for retrieving clinical and biospecimen information. The first one is the direct download of supplements from the portal in XML format, which is semi-structured and does not adhere to a specific data model. The second one is through the GDC APIs, which allow to download structured information according to the GDC Data Model and which provide output in the JSON format. These methods allow to reach two different materializations of the metadata, partly overlapping with each other.

GDC is proceeding with the migration from the first representation to the second one, importing and inserting the data contained in the first within the second. However, in this transitory phase (that has lasted for several months and that will probably last for a long time), much of the information in the first model is not yet replicated to the second, and there is no single source that provides information from both models. In order to obtain a comprehensive representation of such information, it is therefore necessary to extract data using a pipeline that deals with model differences and identifies/manages the overlapping information. The first contribution of our work is the design and development of such a pipeline, such that the clinical and biospecimen data (referred as metadata) are represented with a common format.

We solve the issues arisen in the transition from the TCGA data portal to the GDC one, providing genomic data and their associated clinical / biospecimen data in a standardized format, making both of them seamless, straightforward, and easy to be used. We enhance GDC harmonized data by defining a new data model, in order to uniform genomic and clinical / biospecimen data. We automatically standardize data by mapping to such unique common schema, thereby supporting scientists in integration and analyses [CFW18, CCW18, WCCF16]. We widely exploit the GDC API to interact with GDC data and extract them; for experimental data we apply the extension and standardization procedures defined in Section 2.5. We also integrate information extracted from external public databases, i.e., GENCODE [HFG+12], HGNC [EDS+06], miRBase [GJSvDE07], and Genome annotation of NCBI [BWL+12], enriching the content of the experiments.

We consider this work as an evolution of another project, TCGA2BED [CFC+17], where we faced similar issues, but focusing on the old TCGA portal; unlike TCGA2BED, we widely exploit the GDC API to interact with GDC data and extract them. Our main contribution is the representation of experimental and clinical / biospecimen data by applying the Genomic Data Model (GDM).

GDM consists of two parts, one describing processed datasets with a region-based format, and one describing the metadata. For the former, we map the content of GDC data to GDM, thereby transforming the experimental data of GDC into a new data collection, that we denote as OpenGDC, which are harmonized and extended by linking with other public databases. For the latter, the clinical and biospecimen supplements (which are semi-structured, not part of a data model) are extracted and merged with all the information on clinical and biospecimen data available through the GDC API (which is structured and adheres to the GDC Data Model) and finally converted to the metadata format of GDM, used by OpenGDC.

Other works have dealt with the problem of storing, retrieving, and enhancing data of GDC, almost all of them are focused on the TCGA program. Among them, we mention: i) TCGA Assembler 2 [WJY+17], a software pipeline, which allows to download TCGA data from GDC defining filtering criteria to merge the extracted data files of samples into a single data table, and finally to process them; ii) The International Cancer Genome Consortium (ICGC, [ZBC+11]), which provides a data portal to characterize genomic abnormalities in different cancer types including data from TCGA; iii) The Seven Bridges Cancer Genomics Cloud (CGC, [LLS+17]), which allows to access data from public cancer genomic datasets (e.g., TCGA) and to analyze them in the cloud by using bioinformatics tools and workflows. All these tools are of great interest and improve the access to GDC data, in particular they aggregate them, they identify important genomic features, and they analyze them with cloud computating resources. Our solution is different, as it aims at facilitating the use of TCGA data of GDC by providing it in a standardized and extended format. For a more detailed overview of these tools we refer to the work [SC18], where the authors identify two main categories of TCGA tools, for *Extraction* and for *Integrative data analysis*. We can use this distinction and classify our system in the first category.

2.4 Genomic and clinical data modeling

The main aim of our work is the standardization of GDC experimental data and clinical / biospecimen supplements through the application of the Genomic Data Model (GDM), which provides a representation of the genomic experiments (i.e., data) in the Browser Extensible Data (BED) format [QH10] and its biological / clinical properties (i.e., metadata) in a key-value format. Moreover, we extend genomic data with additional information extracted from external public databases, i.e., GENCODE, HGNC, miRBase, and genome annotations of NCBI. Thanks to GDM, experimental data are unified to a single format and thus become homogeneous, coherent, and comparable. The GDM metadata format is also unified, as the original TCGA metadata formats become all associated to a single format of key-value pairs, although with different choices of keys and with a variable number of pairs. Because of the heterogeneous nature of the data, it is not possible to know a priori all the clinical, biological, and experimental properties of the experimental samples; these are produced as result of metadata mapping. Futhermore, to generate the metadata we develop intelligent procedures for identifying redundant information in clinical /

biospecimen supplements that are present on the two different sources of GDC (i.e., data portal and API).

In Sections 2.5 and 2.6 we describe the genomic data and metadata format that we obtain applying the Genomic Data Model to eight different GDC data types.

2.5 Genomic data format

For genomic data, we use a free-BED data representation, in which we have fixed coordinate fields (chromosome, start position, end position, strand) and then included additional fields according to the specific type of experiment; for every data type we provide a specific ready-to-use schema in XML format. We implemented automatic procedures for converting the original GDC genomic data into such free-BED format; to index our BED output files, we introduce *opengdc_id*, an extension of the aliquot Universal Unique Identifier (UUID) (i.e., the unit of analysis for GDC genomic data identifying a sample analyzed portion). Since in GDC an aliquot corresponds to different data types, *opengdc_id* concatenates the *aliquot uuid* with the specific *data type*. In the following, we provide an overview of the input and outputs data of our standardization procedures. For a detailed description of all input and output fields of each data type, the reader may refer to the OpenGDC format definition (http://geco.deib.polimi.it/opengdc/data/OpenGDC$_{f}ormat_{d}efinition.pdf$).

Gene expression quantification data (Paragraph *RNA sequencing* in Section 1.3) are provided in GDC for each aliquot in three tab-delimited files, each of which presents the Ensembl ID [FAB+11] of the gene and one of the following values: i) *FPKM*, the number of Fragments Per Kilobase of transcript per Million mapped reads; ii) *FPKM-UQ*, the Upper Quartile normalized FPKM value; iii) *counts*, the number of reads aligned to each gene, calculated by HT-Seq. We merge the content of these files using the common *Gene_Ensembl* field. Then, we extract additional information to describe the genomic regions. In the final free-BED structure we include the genomic coordinates (i.e., *chromosome*, *start position*, *end position*, and *strand*), the *gene_symbol* from GENCODE (human genome version GRCh38 annotation), and the corresponding *entrez_gene_id* from the Genome annotation of NCBI.

Isoform Expression Quantification data contain expression profiles calculated for each isoform of the miRNA sequence (Paragraph miRNA sequencing in Section 1.3). GDC provides one file for each aliquot, where each row refers to a single isoform. For the free-BED structure, all input fields are left unchanged with the exception of the *isoform_coords* field, which is parsed to obtain genomic coordinates. As an addition, we retrieve the *entrez_gene_id* and the *gene_symbol* from HGNC.

MiRNA Expression Quantification data (Paragraph *miRNA sequencing* in Section 1.3) are derived from the sequencing of micro RNAs (i.e., miRNA). They contain information about the nucleotide sequence and the miRNA expression. One file per aliquot is provided by GDC, where each row refers to a single isoform, containing the expression computed on all reads aligning to a particular miRNA. In the free-BED output we consider the fields provided in input with the addition of genomic coordinates extracted from miRBase, the *entrez_gene_id*, and the *gene_symbol* extracted from HGNC.

Copy Number Segment and Masked Copy Number Segment (Paragraph *Copy Number Variation* in Section 1.3). GDC provides two data types related to CNVs: Copy Number Segment (including both germline and somatic CNVs) and Masked Copy Number Segment (including only somatic CNVs). The internal representation is the same for both data types. A single experiment is represented by a tab-delimited file, where each row refers to a single CNV. For the free-BED representation we reuse all fields except for the sample id; we add the *strand* — required from the BED standard — which is always set to 'unknown', using the wildcard character '*'.

Masked Somatic Mutation (Paragraph *Somatic Mutation* in Section 1.3) experiments discover mutations by aligning DNA sequences derived from tumor samples to sequences derived from normal samples and a reference sequence. A Mutation Annotation Format (MAF) file is used to specify, for each sample, the discovered putative or validated mutations and to categorize those mutations (SNP, deletion, or insertion) as 'somatic' (i.e., originating in the tissue) or 'germline' (i.e., originating from the germline), as well as to specify additional information about the mutations. Four MAF files for each tumor are provided by GDC, each representing DNA-sequencing data. Each file is generated by a specific analysis pipeline [FXH$^+$16, LHC$^+$11, CLC$^+$13, KZL$^+$12] and includes 125 attributes. By merging the four input fils, we defined a free-BED structure

with 18 fields, including genomic coordinates, *gene_symbol* and *entrez_gene_id*, the type of mutation, the tumor and matched normal sequencing alleles 1 and 2, and the aliquot barcode/UUID for tumor and matched normal samples.

Methylation Beta Value (Paragraph *DNA methylation* in Section 1.3). We consider both Illumina Infinium HumanMethylation27 (HM27) and HumanMethylation450 (HM450) DNA methylation platforms, used for measuring the level of methylation at 27,578 and 485,577 known CpG sites as beta values (respectively for HM27 and HM450). Using probe sequence information provided in the manufacturer manifest, HM27 and HM450 probes are remapped to the GRCh38 reference genome. These probes coordinates are then used to identify the associated transcripts from GENCODE, the associated CpG island (CGI), and the distance of the CpG sites from each of these features. For each methylated site GDC reports a list of gene symbols. The genes that fall within 1500 bp (base pairs) from the methylated site are used, considering the gene as starting from the transcription start site (TSS) to the end of the gene body. For each Methylation Beta Value data aliquot GDC provides a tab-delimited file with 11 fields. We define a free-BED structure composed of 18 fields, which includes all original fields with the addition of the strand, the *entrez_gene_id* retrieved from GENCODE and HGNC, *the ensembl_ transcript_ id*, the *position_to_tss* (distances in base pairs of the CpG site from each associated transcript start site; negative values indicate that the CpG site is located downstream with respect to the TSS), and the *cgi_coordinate* (i.e., the start and end coordinates of the CpG island associated with the CpG site). Moreover, we filtered out the methylation sites with missing beta values (i.e., not measured or with unreliable measurement) and reported the gene symbol that is at minimum bp distance from the CpG dinucleotide, in case this is outside a gene region.

2.6 Metadata format

Each experimental BED file is associated to a metadata file containing a list of key-value pairs. Also metadata files are indexed with the *opengdc_id*, which identifies the BED-META files pair. To populate the OpenGDC metadata files, we retrieve clinical / biospecimen information from the GDC data type called Clinical and Biospecimen Supplements. In addition, we consider other properties retrieved using the GDC APIs (specifying *aliquot uuid* and *data type* as parameters).

Clinical and Biospecimen Supplements are a special data type which contains data documentation; their information is stored in two different XML format files, originally provided by Biospecimen Core Repositories (BCRs) under contract of the National Cancer Institute (NCI). A *clinical supplement*[1] is a collection of information about demographics, medical history (i.e., diagnosis, treatments, follow ups, and molecular tests), and family relationships (i.e., exposure and history) of a particular patient. A *biospecimen supplement*[2] includes information associated with the physical samples taken from a patient and its processing.

2.7 Metadata analysis

The content of an OpenGDC metadata file is obtained by taking into account: i) the GDC *clinical* and *biospecimen* supplements, ii) the information retrieved through the GDC APIs, iii) additional manually curated attributes computed within our standardization pipelines. Given a converted experimental data file in free-BED format, identified by the *opengdc_id*, the corresponding metadata file is generated as shown in Figure 2.1.

On the top left corner, we consider **Biospecimen and Clinical supplements**. They are organized by *patient* (identified by the *bcr_patient_uuid* attribute)— one patient is typically related to many aliquots. Multiple OpenGDC metadata files are created, one for each aliquot reported in the biospecimen file. We replicate the full content of the Clinical supplement over all metadata files regarding the aliquots included in the patient. The resulting attribute keys start with the *clinical__* prefix. A Biospecimen supplement, instead, contains a unique section on the patient, but also distinct sections on multiple samples, their portions, and the resulting aliquots. In each aliquot metadata file we replicate the common parts about the patient (and possibly sample/portion), while the remaining content of the Biospecimen file is divided among the different files according to the specific aliquot. The resulting keys start with the *biospecimen__* prefix.

On the bottom left corner of Figure 2.1, we query **GDC Data Model elements** using their RESTful APIs. We call the services once for each aliquot listed in the Biospecimen supplement, by specifying the *aliquot uuid* and the

[1] https://gdc.cancer.gov/about-data/data-harmonization-and-generation/clinical-data-harmonization

[2] https://gdc.cancer.gov/about-data/data-harmonization-and-generation/biospecimen-data-harmonization

data type, and then associate to each OpenGDC data file all information retrieved in the obtained response. The extracted attributes describe a data file along different GDC Data Model conceptual areas (i.e., analysis, administrative, biological, and clinical). Relevant administrative entities include the Program (i.e., the broad framework of goals to be achieved by multiple experiments, such as TCGA), the Project (i.e., the specifically defined piece of work that is undertaken or attempted to meet a single requirement, such as TCGA-LAML), the Case (i.e., the collection of all data related to a specific subject in the context of a specific project, such as a patient). Among Biological entities there are Sample (i.e., any material sample taken from a biological entity for testing, diagnostic, propagation, treatment or research purposes) and Aliquot (i.e., pertaining to a portion of the whole; any one of two or more samples of something, of the same volume or weight). Clinical entities include Treatment (i.e., therapeutic agents provided—or to be provided—to a patient to alter the course of a pathologic process) and Diagnosis (i.e., data from the investigation, analysis and recognition of the presence and nature of disease, condition, or injury from expressed signs and symptoms). Analysis entities include harmonization pipelines such as "Copy Number Variation" and "Methylation Liftover", each related to one data type.

In case the OpenGDC data file corresponds to n original GDC files, the API JSON response is divided in n partitions, containing information both on the single GDC original file and on the related aliquot (these are replicated in each partition). In one OpenGDC final metadata file, we group the information from the original files (by concatenating multiple values in a single pair) while we consider the aliquot information only once. Attribute names are prefixed with *gdc__* and obtained by flattening the hierarchical structure of the JSON responses, i.e., through concatenation of json keys at each traversed level.

As an addition to GDC inputs, we generate a set of extra **manually curated key-value pairs** (gathered in the group of keys prefixed with *manually_curated__*). These contain information that is missing in GDC and derived from other sources or specified by our system. We add the data format (e.g., BED file textual format), URLs of the data and metadata files on the FTP server offered by OpenGDC (see Section `Results and Discussion` for details about OpenGDC software and the FTP Repository), the genome built (i.e., reference assembly), the *id, checksum, size* and *download date* of the data file, and the status of the tissue, which indicates if it is of normal or control sample.

Combining clinical / biospecimen information with GDC API information has led to value redundancy, which is due to the fact that there does not exist a specific data model for the Supplements data and it is impossible to determine

a priori which information are non-overlapping. We ascertained the presence of attributes holding different names but same semantics and associated values. We profiled all input data, obtaining sets of different keys that typically present same values within a same metadata file. Example groups of pairs with the same value and the corresponding chosen candidate is shown in Table 2.1. We defined a list of heuristics to remove the redundant attributes from metadata, which is applied by the *Data Redundancy Solver* (at the center of Figure 2.1):

1) verify mappings on the official GitHub repository[3] specifying which BCR fields correspond to the API fields: when redundant, keep the second;

2) when BCR biospecimen fields are redundant w.r.t. clinical, keep the first;

3) when fields belonging to the *case* group are redundant w.r.t. *case.project* fields, keep the first;

4) when fields belonging to the *analytes* group are redundant w.r.t. *analytes.aliquots* fields, keep the second.

To facilitate key-value pairs use, in case keys are very long and cumbersome, we simplify them with the *Data Renaming Module*, which applies renaming rules based on a regular expressions match-and-replace strategy. With respect to the original keys retrieved from the APIs, we usually leave unchanged the rightmost part (i.e., last subgroup and name of the attribute), thus it is ensured that the attributes remain uniquely identified. As an example, `gdc__cases__samples__portions__analytes__aliquots__aliquot_id` becomes `gdc__aliquots__aliquot_id`. The three levels of the resulting attribute, separated by double underscore, identify an attribute retrieved through the APIs ("gdc"), belonging to the "aliquots" Data Model entity, and indicating specifically the identifier of the represented aliquot (i.e., "aliquot_id"). Examples of renaming rules and their results are shown in Table 2.2.

2.8 OpenGDC software solution

Part of this section was published in [CCB+18]

In this Section we present the OpenGDC software, which implements the data models and the retrieval, extension, standardization procedures described in previous sections.

[3] `https://github.com/NCI-GDC/gdcdatamodel/tree/release/horton/gdcdatamodel/xml_mappings`

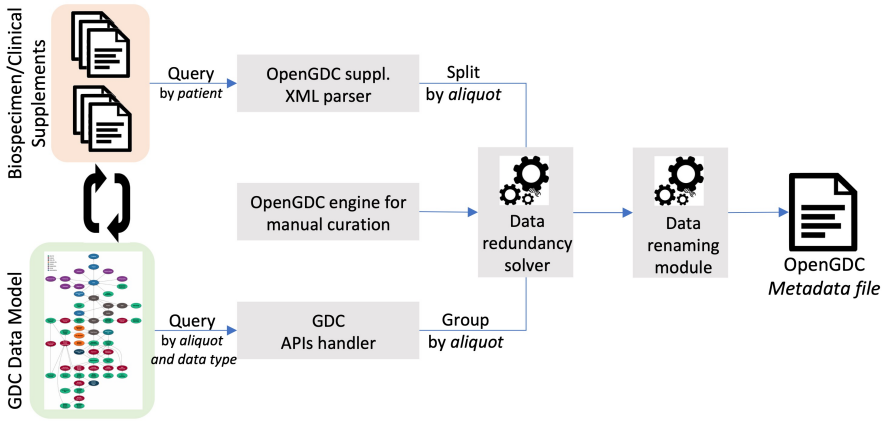

Figure 2.1: Metadata pipeline overview. The procedure starts with the download of the Biospecimen and Clinical Supplement files by exploiting the GDC API according to a *patient uuid*. Aliquot uuids are extracted from the Biospecimen file, whose content is split by the aliquot. Based on these *aliquot uuids* and their associated *data types*, the GDC Data Model is queried through the GDC API in order to obtain additional metadata information. Finally, inside a unique metadata file, we merge together: Clinical data, a portion of Biospecimen data, GDC data model metadata, and manually curated attributes (automatically generated by the pipeline). The obtained attributes, which compose the final metadata file, are previously processed by two additional components: the Data redundancy solver that deals with removing redundant attributes, and the Data renaming module that applies rules for renaming attributes.

Preserved	Different attributes	Example values
	biospecimen__bio__analyte_type	RNA
×	gdc__cases__samples__portions__analytes__analyte_type	RNA
×	biospecimen__admin__day_of_dcc_upload	31
	clinical__admin__day_of_dcc_upload	31
×	gdc__cases__primary_site	Ovary
	gdc__cases__project__primary_site	Ovary
×	gdc__cases__samples__portions__analytes__aliquots__concentration	0.17
	gdc__cases__samples__portions__analytes__concentration	0.17

Table 2.1: Example of choices produced by the Data Redundancy Solver.

GDC naming	OpenGDC flattened	OpenGDC renamed
cases.diagnoses.age_at_diagnosis	gdc__cases__diagnoses__age_at_diagnosis	gdc__diagnoses__age_at_diagnosis
analysis.input_files.data_category	gdc__analysis__input_files__data_category	gdc__input_files__data_category
cases.project.program.name	gdc__cases__project__program__name	gdc__program__name

Table 2.2: Column 1 shows the attribute names as they are specified in GDC APIs parameters; Column 2 shows the OpenGDC naming convention; Column 3 shows the results of the renaming phase applied to the attribute in Column 2.

OpenGDC is an open-source and cross-platform software written in Java, which allows the extraction, extension, and standardization of public available data from GDC. The software is available as a standalone desktop application with a friendly user interface and supports the BED, GTF, CSV, JSON, and XML standard formats as output. Its architecture has been implemented following the Model-View-Controller (MVC) design pattern as shown by the flowchart in Figure 2.2.

The software is composed of two main pipelines: (i) the GDC data download procedure and (ii) the data conversion one.

The whole system can be summarized by three software components:

- *Controller*: it redirects the user instructions to the correct module and initializes an instance of the software able to download and/or convert the GDC data;

- *Data Download System*: it manages the process of search and retrieval of the public GDC data exploiting the GDC APIs;

- *Data Standardization Module*: it allows to easily convert and standardize data according to a specific data type. The process is facilitated by an ad-hoc class *BioParser*, which provides an abstract representation for all GDC data types; this class can be extended to support new data types in case of future extensions of the GDC repository.

OpenGDC exploits the public GDC APIs during the data download procedure, to retrieve the original genomic, clinical, and biospecimen data. It also makes use of the GDC APIs during the conversion procedure of the Clinical and Biospecimen supplements to extract additional information such as the size of the downloaded files, their MD5 checksum, as well as the last creation and update timestamps that will be added in the metadata files.

Data conversion uses a different parser depending on the type of converted data. Additionally, the process retrieves complementary information from a

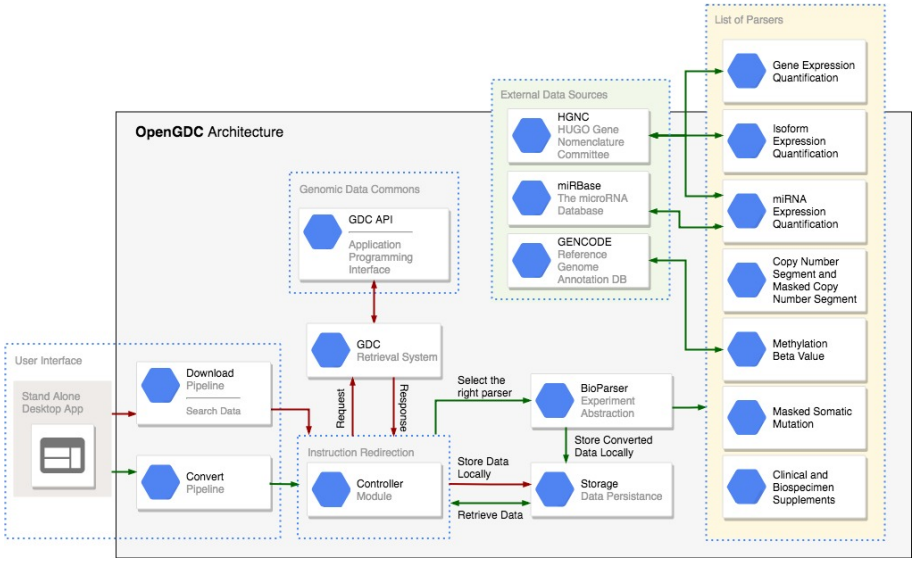

Figure 2.2: OpenGDC architecture. This is a graphical organization of the flowchart representing the OpenGDC software architecture. Every feature has been differentiated in two pipelines, i.e., Download and Convert, highlighted by red and green arrows, respectively. Every software module has been additionally surrounded to delineate its function (i.e.: User Interface, Instruction Redirection, Genomic Data Commons APIs, External Data Sources, and the List of Parsers).

set of external data sources like NCBI, GENCODE, HGNC, and miRBase, to extract the genomic coordinates, the entrez gene ID, and the gene symbols starting from information are already existing in the original data.

Interacting with the GDC public APIs

We search and extract data and other information from GDC through their public APIs. In particular we exploit three main endpoints:

- *cases*: to find all files related to a specific case (i.e., sample donor);
- *files*: to find all files with specific characteristics such as the file name, MD5 checksum, and data format;
- *data*: to download GDC data files.

To understand the interplay of the three endpoints, consider a scenario where we want to download all public available *Gene Expression Quantification* data for the tumor *Breast Invasive Carcinoma* in the context of the *TCGA* program. First we need to query GDC for all the file unique identifiers (UUIDs) related to this particular case. To this end, we make an HTTP POST request to the *files* endpoint with a payload in Listing 2.1. As a result, GDC returns a list of file UUIDs (*file_id* fields) as shown in Listing 2.2. Starting from this list, we then download the associated files; this is done by querying the *data* endpoint (i.e, one HTTP GET request for each result in the previous query response) specifying a single file UUID, e.g., http://api.gdc.cancer.gov/data/1837ad2a-4edf-4d80-9050-f78115e54454. For a detailed description about the syntax of the payload and the other ways to query GDC, the reader may refer to the GDC APIs documentation available at https://docs.gdc.cancer.gov/. For additional details about the OpenGDC software tools and its usage we point the reader to our user guide available at http://geco.deib.polimi.it/opengdc/versions/OpenGDC-v1.0.zip. We applied the OpenGDC to TCGA program and created the OpenGDC repository described in Section 3.3.

```
{
   "op":"and",
   "content":[
      {
         "op":"=",
         "content":{
            "field":"cases.project.project_id",
            "value":[
               "TCGA-BRCA"
            ]
         }
      },
      {
         "op":"=",
         "content":{
            "field":"files.data_type",
            "value":[
               "Gene Expression Quantification"
            ]
         }
      },
      {
         "op":"=",
         "content":{
            "field":"access",
```

```
            "value":[
               "open"
            ]
         }
      }
   ]
}
```

Listing 2.1: JSON representation of the payload required to query GDC for all public available Gene Expression Quantification data about the Breast Invasive Carcinoma in the context of the TCGA project.


```
{
   "hits":[
      {
         "data_type":"Gene Expression Quantification",
         "updated_datetime":"2018-08-07T15:05:9:14.863537+00:00",
         "file_name":"bb12bb45-2c4a-46dc-98dd-9acbe2a0c4ee.FPKM.txt.gz",
         "submitter_id":"bb12bb45-2c4a-46dc-98dd-9acbe2a0c4ee_fpkm",
         "file_id":"1837ad2a-4edf-4d80-9050-f78115e54454",
         "file_size":490684,
         "id":"1837ad2a-4edf-4d80-9050-f78115e54454",
         "created_datetime":"2016-05-29T10:02:5:44.747424-05:00",
         "md5sum":"236e7f21947de1053d99c0c16c8f494c",
         "data_format":"TXT",
         "access":"open",
         "data_category":"Transcriptome Profiling",
         "type":"gene_expression",
         "experimental_strategy":"RNA-Seq"
      },
      {
         "data_type":"Gene Expression Quantification",
         "updated_datetime":"2018-08-07T15:05:9:14.863537+00:00",
         "file_name":"4dcd18d6-b7b2-4a92-bcbe-361c7278e84e.FPKM.txt.gz",
         "submitter_id":"4dcd18d6-b7b2-4a92-bcbe-361c7278e84e_fpkm",
         "file_id":"b9610459-bd3e-4d65-92cd-8eb34541f259",
         "file_size":500108,
         "id":"b9610459-bd3e-4d65-92cd-8eb34541f259",
         "created_datetime":"2016-05-30T18:02:9:36.681823-05:00",
         "md5sum":"a4cef437e5523efb4f60f50386065534",
         "data_format":"TXT",
         "access":"open",
         "data_category":"Transcriptome Profiling",
         "type":"gene_expression",
         "experimental_strategy":"RNA-Seq"
      }
   ],
   "pagination":{
      "count":2,
      "sort":"",
```

```
    "from":1,
    "page":1,
    "total":3669,
    "pages":734,
    "size":2
  }
}
```


Listing 2.2: GDC JSON response corresponding to the query in Listing 2.1. The number of hits is limited to 2 results for brevity, however the software automatically scrolls over all the files listed in the result.

2.9 Conclusions

In this chapter, we presented a new data model for biomedical data and defined automatic procedures able to extract, integrate, extend and standardize genomic and clinical data of The Cancer Genome Atlas (TCGA) as included in the Genomic Data Commons portal. Our model and software was applied to data types that are obtained from different NGS experiments (i.e., Gene-, miRNA-, Isoform Expression Quantification, Masked Somatic Mutation, Copy Number Segment, Masked Copy Number Segment, Methylation Beta Value). Additionally, we considered clinical and biospecimen information about the experimental data.

To reach our objective, we exploited the Genomic Data Model (GDM), that allowed us to represent an experimental sample by its genomic regions and its related metadata. The genomic regions are defined by genomic coordinates (chr, left, right, strand) and genomic features which are produced by the specific NGS experiment. Conversely, metadata report clinical and biological properties in attribute-value pairs format.

Based on GDM representation, we implemented OpenGDC, a software for retrieving TCGA experimental data, which is then processed with ad-hoc procedures for each data type. Our standardization procedure provides all the data in free-BED format, which contains a set of experimental-specific fields in addition to the genomic coordinates. In order to obtain this standardized format, the software is able to extract additional features from external data sources (i.e., GENCODE, HGNC, and miRBase), which are not provided in the original GDC files. The software is also integrating experimental data with clinical and biospecimen information derived from different GDC sources.

Our pipeline extracts metadata attributes from the original Clinical and Biospecimen Supplements and from the GDC RESTful APIs. The obtained

attributes are merged in a single metadata file, using a tab-delimited key-value format. We then used two software components in the metadata pipeline: (i) the Data Redundancy Solver, to detect and remove redundant metadata attributes, and (ii) the Data Renaming Module to redefine the attribute names. In particular, we performed data profiling activity to identify redundant attributes, i.e., with the same values and different attribute names.
All these procedures and the input/output data types are thoroughly described in the OpenGDC Format Definition document available at
http://geco.deib.polimi.it/opengdc/data/OpenGDC_format_definition.pdf.

Future work concerns the application of our data representation and software pipeline to other projects integrated in the GDC portal and to other cancer-related repositories in order to facilitate knowledge discovery over cancer data. Additionally, we plan to use our model and software in order to further enhance the data integration among the different biomedical public repositories. Finally, we are going to exploit the standardized data, which is easily processable by several state of the art bioinformatics tools, in order to perform new knowledge extraction analyses about cancer.

Chapter 3

Biomedical data accessibility and querying

3.1 Introduction

Public and private repositories of genomic data have been created with the aim of spreading NGS data [SMB+17, BWL+12, BKML+08]. Unfortunately, these databases often lack of standardization and of efficient storage models, resulting in waste of storage space, usually related with data redundancy. Nowadays, browsing all these data and retrieving significant insights from them is a big challenge that has been already faced with different techniques in order to facilitate their accessibility and interoperability. Repositories and databases have to provide *discoverability*, *availability* and *accessibility* of data resources in order to promote and facilitate data sharing for the genomics research community [LDC+19, vSKP+14]. These represent significant steps to improve data sharing, which ensure continued progress in understanding diseases.

In Chapter 2 we proposed an automatic standardization pipeline [CCB+18] to model heterogeneous genomic data, applying novel methodologies that involve the most modern technological innovations in data management [KGM+17]. This model can be the base for the development of advanced software solutions for efficiently *querying* these data [FLM+15] in order to retrieve potentially relevant insights. Unfortunately, for querying these data, is often required a deep knowledge about (i) the strategies with which the experiments are produced and (ii) the technical terminology adopted by domain-experts used to define the metadata associated to the experimental data.

In this Chapter, we face with the problem of modeling genomic and clinical data in order to minimize the amount of redundant information [CZV17] (therefore also storage space), describing a possible solution [CWC19].
We address followed biomedical data management issues: (i) how to reduce the redundancy of genomic and clinical data, (ii) how to make this big amount of data easily accessible, and finally (iii) which tools to use to extract and query these data.

We introduce data source which we consider as the beginning starting point to treat this topic, i.e., the repository that contains the standardized genomic data and metadata obtained by applying OpenGDC to The TCGA program as reported in Chapter 2. Then we describe the GenoMetric Query Language (GMQL, [MPV+15]), a high-level domain-specific query language, and its application on the standardized data in order to highlight the advantages of our representation in terms of information retrieval. Therefore, we show how this data can be accessed, and a language to be able to query them.

From Section 3.5 we move a step forward. We propose an approach to organize the standardized genomic and clinical data by taking into account data redundancy and a method able to save space by exploiting the no-SQL technologies. We suggest principles for organizing biomedical data and make them easily accessible.

Finally we face with the issue of querying these data by external users who do not know the specific nature of these records, the granularity of the query may not always correspond to what is present in the database. This last issue occurs when the structure of data is not formally defined usually due to the heterogeneous nature of the this kind of data. The metadata represent the most significant case of data variability on structure and content. Thanks to the application of taxonomies, we can figure out what kind of relationship exists between the information sought by a given user through a query and the data that are actually present in the records, which may not be exactly the requested ones. In Section 3.10 we exploit domain-specific ontologies in order to allow executing taxonomy-based relaxed queries. We apply the upward and downward query extension methods, to obtain a finer or coarser granularity of the requested information.

3.2 Biomedical Data management

Recently, many efforts have been made towards a better management of genomic and clinical data through harmonization procedures for standardiza-

tion and improved accessibility. Concrete examples of these methods are TCGA2BED and OpenGDC, two software tools for the automatic extraction, extension, and standardization of public available genomic and clinical data from the The Cancer Genome Atlas (TCGA) portal, and the Genomic Data Commons (GDC) portal, respectively. With these tools we provide also open-access FTP repositories with standardized data in free-BED, which is widely used format in the bioinformatics community, available at ftp://bioinf.iasi.cnr.it/tcga2bed/ and at ftp://geco.deib.polimi.it/opengdc/.
In the following section, we describe the OpenGDC repository that is constantly syncronyzed and updated with more recent GDC data.

3.3 Standardized cancer genomic data repository

The OpenGDC repository contains all the public available data of the TCGA program of GDC in the standardized and extended version (BED format and key-value format) described in Chapter 2, thanks to the application of the OpenGDC software.. The data are firstly divided in two branches, original GDC data / extended BED ones (*original* and *bed* folders, respectively). The structure of the FTP space is then organized within the two branches using the following structure: *program* (e.g., TCGA), *tumor* (e.g., TCGA-BRCA, TCGA-KIRP, TCGA-OV, etc.), and finally *data type* (e.g., gene-expression-quantification, methylation-beta-value, clinical-and-biospecimen-supplements, etc.). For each data type the genomic and meta data are provided for each aliquot. Currently, a total volume of 2.7 TB of data (1.2 TB of original GDC data and 1.5 TB of converted one) of 33 different tumors is maintened.
Table 3.1 shows the details about the number of aliquots, patients, and samples the reader may refer to

The FTP repository also includes an automatic data update procedure to maintain the original and converted data always up to date with the latest data version available at GDC. It consists in a subroutine that exploits the GDC APIs to search for updates or new data availability once a month, and eventually synchronizes them with our repository.

3.4 Querying OpenGDC Data with GMQL

In this Section, we show the application of the GenoMetric Query Language (GMQL, [MPV+15]) on the standardized data in order to highlight the advantages of our representation in terms of information retrieval. GMQL is a high-

Tumor	Aliquots	Samples	Patients
Adrenocortical Carcinoma	770	770	595
Bladder Urothelial Carcinoma	3787	3763	2874
Breast Invasive Carcinoma	10301	10276	7522
Cervical Squamous Cell Carcinoma and Endocervical Adenocarcinoma	2706	2706	2118
Cholangiocarcinoma	401	401	267
Colon Adenocarcinoma	4350	4236	3117
Lymphoid Neoplasm Diffuse Large B-cell Lymphoma	415	415	323
Esophageal Carcinoma	1705	1701	1271
Glioblastoma Multiforme	3325	3260	2178
Head and Neck Squamous Cell Carcinoma	4938	4934	3627
Kidnay Chromophobe	615	615	429
Kidnay Renal Clear Cell Carcinoma	5315	5147	3495
Kidnay Renal Papillary Cell Carcinoma	2812	2784	2023
Acute Myeloid Leukemia	1215	1215	927
Brain Lower Grade Glioma	4670	4670	3588
Liver Hepatocellular Carcinoma	3604	3602	2610
Lung Adenocarcinoma	5246	5147	3723
Lung Squamous Cell Carcinoma	4780	4736	3460
Mesothelioma	775	775	603
Ovarian Serous Cystadenocarcinoma	4727	4699	3538
Pancreatic Adenocarcinoma	1659	1659	1267
Pheochromocytoma and Paraganglioma	1650	1650	1251
Prostate Adenocarcinoma	4777	4777	3472
Rectum Adenocarcinoma	1456	1449	1122
Sarcoma	2241	2335	1797
Skin Cutaneous Melanoma	4197	4197	3242
Stomach Adenocarcinoma	4107	4079	3018
Testicular Germ Cell Tumors	1311	1311	1013
Thyroid Carcinoma	4827	4827	3523
Thymoma	1120	1120	862
Uterine Corpus Endometrial Carcinoma	5066	5036	3838
Uveal Melanoma	720	720	560

Table 3.1: List of processed tumors with the related number of involved aliquots, samples, and patients.

level domain-specific query language. It can be executed in the architecture described in [MCP+18], which is specific for genomic data processing. Every operation in GMQL is strictly connected to the data structure. The operations are based either on the genomic regions or the metadata. A GMQL query has the following structure: $< variable >= operator(< parameters >) < variables >$. The variables represent the GDM dataset and they are used as input for the transformations and as output in order to maintain the resulting dataset and in order to use the dataset itself in the consequent transformations. The operators determine what kind of transformation is to be obtained and some of them can be applied to one or more variables. The parameters are peculiar for every

operator and may include predicates, based on boolean expressions which have the task to select or merge samples. Moreover, depending on the kind of operator, the predicates can be applied to every attribute of the metadata or to the genomic region of the samples thus applying the data scheme attributes. The attributes mainly describe the operation on the distances between the genomic regions, that represent the fundamental point of the whole language.

The current available version of the GMQL system uses Apache Spark[1] as its backbone; along with other design choices, this provides high scalability in cloud computing. GMQL system contains a multiplicity of public genomic datasets ready to be used within tertiary analysis pipelines (as shown in [MCP+18]); among other sources, it features all the datasets available in the OpenGDC FTP service, providing an interface for browsing and processing curated data in OpenGDC.

In this Section, we propose three GMQL use cases along with their queries; we focus on query aspects, acting on both region data and metadata, which highlight the strengths of the datasets produced by OpenGDC: 1) enabling the combined use of metadata derived from GDC Data Model, from the submitters clinical / biospecimen supplements, and from manually curated additions;
2) providing positional information (i.e., genomic coordinates) in a standardized structure, which encourages data inter- and intra-source interoperability;
3) allowing joined use of different data types (e.g., gene expression and methylation) based on common gene identifiers (e.g., the HUGO gene_symbol).

Use case 1. *For kidney cancer, find the frequency of mutations in the exons of genes.*

For this example, we consider public somatic mutation data samples of Kidney Adenomas and Adenocarcinomas patients; such partition contains three projects, i.e., Kidney Chromophobe (KICH), Kidney Renal Clear Cell Carcinoma (KIRK), and Kidney Renal Papillary Cell Carcinoma (KIRP); and select novel mutations (i.e., not listed in dbSNP [SWK+01]). For each sample, we count the mutations occurring in each exon and filter-out the exons without any mutation. We then return such samples together with the number of exons remaining in each sample and the maximum number of mutations in a single exon.

In this example: 1) we use GDC mutation data in combination with a GENCODE annotation dataset—demonstrating the interoperability of OpenGDC curated data with other sources; 2) we use seamlessly metadata from GDC

[1] http://spark.apache.org/

API (i.e., first and second conditions in the selection) and clinical supplements (third and fourth conditions)—this is not possible on GDC portal, where only the former are supported; 3) we select three TCGA projects together by using the characterization of the tissue and the classification of diseases (note that the attribute *gdc__disease_type* represents the type of malignant disease[2], while *gdc__project__disease_type* is contains the full name for the project. The output dataset contains 230 samples with 19,270 regions.

```
#Select mutation data based on both region and metadata attributes
MUT = SELECT(gdc__primary_site=="Kidney" AND gdc__disease_type=="Adenomas and
      Adenocarcinomas" AND clinical__shared__history_of_neoadjuvant_treatment == "No" AND
      clinical__clin_shared__followup_treatment_success == "Complete Remission/Response";
      region: dbsnp_rs=="novel") GRCh38_TCGA_somatic_mutation_masked_2018_12;
#Select known human protein−coding and non−protein−coding exon regions of the GENCODE
      annotation release 22
EXON = SELECT(annotation_type=="exon" AND release_version=="27")
      GRCh38_ANNOTATION_GENCODE;
#Map the mutations to the exon regions and counts how many they are in each sample
EXON_MUT = MAP(count_name: MUT_count) EXON MUT;
#Remove exons that do not contain mutations
EXON_MUT_SELECT = SELECT(region:MUT_count>0) EXON_MUT;
#In the metadata of each sample add the count how many exons remain and the maximum number of
      mutations in an exon
EXON_RES = EXTEND(exon_count AS COUNT(), max_mut AS MAX(MUT_count)) EXON_MUT_SELECT;
MATERIALIZE EXON_RES INTO result1_exons_mutations;
```

Listing 3.1: Example of GMQL query to find exons with somatic mutations.

Use case 2. *In Breast Invasive Carcinoma, find the regions that are present in at least 10% miRNA expression tumoral samples whose miRNA counts result above average.*

We translate these specifications into selecting samples corresponding to patients who are affected by primary tumors and exhibit a value for *reads_per_million_mirna_mapped*[3] above the average of the dataset. We first use a simple query to evaluate the average of miRNA normalized reads. In order to obtain the lightest query possible in terms of computational time, we PROJECT only the required field, MERGE all samples into one, compute the average as a metadata and MATERIALIZE a tiny dataset in order to get

[2]The disease is categorized by the World Health Organization's (WHO) International Classification of Diseases for Oncology (ICD-O).
[3]In miRNA Expression Quantification data type, it is the read normalized count in reads-per-million-miRNA-mapped associated to each miRNA ID.

the required value (in this case 531.6). We then perform a query to filter out regions that present a *reads_per_million_mirna_mapped* value equal or below the average of the dataset. Then, we use COVER to combine in one sample only regions that are present in at least 10% of dataset samples and equip each region with: 1) the number of miRNA regions that contribute to create the result region; 2) the list of co-located genes, using specifically the *entrez_gene_id* region attribute, which is an additional attribute w.r.t. original GDC data. The output dataset contains 1 samples with 102 regions.

```
#This query materializes a set from whose metadata we read the average reads value to be used as
      threshold
S0 = SELECT(gdc__project__disease_type=="Breast Invasive Carcinoma" AND
      gdc__samples__sample_type=="Primary Tumor") GRCh38_TCGA_miRNA_expression_2018_12;
P = PROJECT(reads_per_million_mirna_mapped; metadata:none) S0;
M = MERGE() P;
E = EXTEND(avg_reads AS AVG(reads_per_million_mirna_mapped)) M;
MATERIALIZE E INTO result2_reads_threshold;

#Find regions with reads above threshold
S = SELECT(gdc__project__disease_type=="Breast Invasive Carcinoma" AND
      gdc__samples__sample_type=="Primary Tumor"; region: reads_per_million_mirna_mapped
      > 531.6) GRCh38_TCGA_miRNA_expression_2018_12;
#Find regions present in more than 10% samples; for each region report a list of overlapping genes and
      number of samples
C = COVER(ALL/10,ANY;aggregate: all_genes AS BAGD(entrez_gene_id), num_samples AS
      COUNT()) S;
MATERIALIZE C INTO result2_cover;
```

Listing 3.2: Example of GMQL query that finds regions with miRNA expression levels above average that are present in more than 10% samples with associated genes.

Use case 3. *For follow-up comparative analysis, extract for all genes the expression and methylation levels for both control and tumoral cases for each patient affected by "Cholangiocarcinoma".*

Using the *manually_curated__tissue_status* attribute with value "normal" we can select samples with different sample types at once (i.e., Blood Derived Normal, Solid Tissue Normal, Buccal Cell Normal, EBV Immortalized Normal, Bone Marrow Normal—corresponding to codes 10-14 in http://gdc.cancer.gov/resources-tcga-users/tcga-code-tables/sample-type-codes).
Similarly, "tumor" includes 10 different types (codes 01-09 and 40). To combine meaningfully the gene expression regions with methylation ones, we expand the former's coordinates from 4000 bases upstream to 1000 bases downstream

around the gene TSSs, since the methylation sites of interest may be located in the surroundings of the gene (Lines 3 and 12). For expression data, we only keep the *fpkm* values and the *gene_symbol*. For methylation data we keep the *beta_value*.

Note that the code described in Lines 1-8 for normal samples is repeated in Lines 10-17 for tumoral samples. With the MAP at Line 8 we associate to each gene expression region (reference) the average of *beta_values* of their overlapping methylated regions (experiment). Reference and experiment are matched only if belonging to the same patient (uniquely identified by the *gdc__case_id*). At Line 20 the datasets resulting from line 8 and 17 are combined using a JOIN operation, which allows to associate to each region, the *gene_symbol* that includes its coordinates, and the *fpkm* and *avg_beta_value* from both normal and tumoral cases. Note that the equi predicate `on_attribute` can only be applied thanks to the addition of *gene_symbol* attribute in the gene expression dataset (original GDC data did not provide it). Lines 23-24 are only needed for shaping results into a convenient format, as it can be observed in Table 3.2, which contains an excerpt from the result dataset.

```
1  #Select Cholangiocarcinoma gene expression normal samples
2  N0_EXPR = SELECT(gdc__project__disease_type == "Cholangiocarcinoma" AND
       manually_curated__tissue_status == "normal") GRCh38_TCGA_gene_expression_2018_12;
3  N_EXPR = PROJECT(fpkm,gene_symbol; metadata:gdc__case_id; region_update: start AS start
       - 4000, stop AS start + 1000) N0_EXPR;
4  #Select Cholangiocarcinoma methylation normal samples
5  N0_METH = SELECT(gdc__project__disease_type == "Cholangiocarcinoma" AND
       manually_curated__tissue_status == "normal") GRCh38_TCGA_methylation_2018_12;
6  N_METH = PROJECT(beta_value; metadata:gdc__case_id) N0_METH;
7  #For each patient, for each reference gene region (in normal data) add the average of beta values of the
       regions that overlap it
8  N_EXPR_METH = MAP(avg_beta_value AS AVG(beta_value); joinby: gdc__case_id) N_EXPR N_METH;

10 #Select Cholangiocarcinoma gene expression tumor samples
11 T0_EXPR = SELECT(gdc__project__disease_type == "Cholangiocarcinoma" AND
       manually_curated__tissue_status == "tumoral") GRCh38_TCGA_gene_expression_2018_12;
12 T_EXPR = PROJECT(fpkm,gene_symbol; metadata:gdc__case_id; region_update: start AS start
       - 4000, stop AS start + 1000) T0_EXPR;
13 #Select Cholangiocarcinoma methylation tumor samples
14 T0_METH = SELECT(gdc__project__disease_type == "Cholangiocarcinoma" AND
       manually_curated__tissue_status == "tumoral") GRCh38_TCGA_methylation_2018_12;
15 T_METH = PROJECT(beta_value; metadata:gdc__case_id) T0_METH;
16 #For each patient, for each reference gene region (in tumor data) add the average of beta values of the
       regions that overlap it
17 T_EXPR_METH = MAP(avg_beta_value AS AVG(beta_value); joinby: gdc__case_id) T_EXPR T_METH;

19 #For each region see corresponding gene and normal/tumor expression values/average beta values
20 J = JOIN(DLE(0); on_attributes: gene_symbol; joinby: gdc__case_id) N_EXPR_METH
```

```
     T_EXPR_METH;

22 #Format results as a matrix with only significant columns
23 J1 = PROJECT(region_update: gene_symbol AS N_EXPR_METH.gene_symbol, normal_fpkm AS
     N_EXPR_METH.fpkm, normal_avg_beta_value AS N_EXPR_METH.avg_beta_value, tumor_fpkm
     AS T_EXPR_METH.fpkm, tumor_avg_beta_value AS T_EXPR_METH.avg_beta_value;
     metadata_update: patient AS N_EXPR_METH.N_EXPR.gdc__case_id) J;
24 J2 = PROJECT(gene_symbol, normal_fpkm, normal_avg_beta_value, tumor_fpkm,
     tumor_avg_beta_value; metadata: patient) J1;
25 MATERIALIZE J2 INTO result3_matrix_normal_tumor;
```

Listing 3.3: Example of GMQL query that builds a matrix comparing, for each gene symbol, the average fpkm expression values and methylation meta values for normal and tumoral tissues.

In the column names of Table 3.2 we use the subscripts n and t for *normal* and *tumoral*, respectively. Occurrences of `null` in the average beta values correspond to cases where no methylation points are located in the specified gene. Overall, the output dataset contains 9 samples with in total more than 500k regions.

chr	left	right	strand	gene_symbol	fpkm_n	avg_beta_value_n	fpkm_t	avg_beta_value_t
chr1	166971581	166976581	+	MAEL	0.27401479	0.07428182	0.19981536	0.06583118
chr1	166974482	166979482	-	ILDR2	0.13031929	0.11815327	0.06208503	0.13756338
chr3	38949561	38954561	-	SCN11A	0.04643162	0.88310268	0.01814642	0.73347131
chr6	152746797	152751797	+	VIP	0.50472323	0.13604175	0.11766157	0.35010738
chr11	114558895	114563895	-	NXPE1	0	0.80843122	0.0161897	0.82677058
chr4	8955627	8960627	+	UNC93B8	0	null	0	null
chr12	126615554	126620554	-	RP11-407A16.8	0	0.96168949	0	0.97617533
chr1	154205333	154210333	-	C1orf189	0.16309294	0.8960085	0	0.9050279
chr10	88786061	88791061	-	RCBTB2P1	0	null	0	null
...	...	...	...	...	...	...	...	...

Table 3.2: Excerpt from Example 3 output matrix.

3.5 Genomic and clinical data redundancy

Part of this section was published in [CWC19]

We took into account the free-BED representation of data provided by the OpenGDC public FTP repository. In particular, although these data are well organized following a clear conceptual data division schema — according to the *tumor*, *experimental data type*, and finally the most atomic data represented by the single experiment denoted by its *universal unique identifier* — they suffer

of data redundancy issue caused by the requirements imposed by the adopted standard.

Starting from the data representation described above, we extract all the free-BED data and we model them to minimize their content redundancy. In particular, we focus on *Gene Expression Quantification* (GEQ) and *Methylation Beta Value* (MBV) data, which are the main experimental data types affected by the problem of redundant information (see the Paragraphs about these experiments in Section 2.5).
Every GEQ experiment contains indeed the same information about the genes — genomic coordinates of the involved genes (*chromosome*, *start position*, *end position*, *strand*), the *ensembl gene id*, the *entrez gene id*, and *type* related to the corresponding ensembl id of the gene, according to the previously cited OpenGDC format definition — except for the those values that characterize an experiment from all the others (*htseq_count*, *fpkm*, and *fpkm_uq*, that correspond to the number of reads aligned to each gene (calculated by HT-Seq), the number of Fragments Per Kilobase of transcript per Million mapped reads (FPKM), and the upper quartile normalized FPKM value respectively). Also MBV data contains redundant information consisting the genomic coordinates of the involved methylated sites (also called CpG islands) that correspond to a single nucleotide position in this case (*chromosome*, *start position*, *end position*, and*strand*), the *composite_element_ref* (the CpG site identifier), *gene_symbol*, *entrez_gene_id*, *gene_type*, *ensembl_transcript_id*, *position_to_tss*, *all_gene_symbols*, *all_entrez_gene_ids*, *all_gene_types*, *all_ensembl_transcript_ids*, *all_positions_to_tss*, *cgi_coordinate*, and *feature_type*. We may suggest the reader to have a look at the previously cited format definition document for a detailed explanation of the information listed above. As in GEQ experiments, also for MBV experiments the *beta_value* identifies the information that characterized every single experiment. It is worth noting that redundant fields are repeated for each experiment, because of the sequencing technology and the related chips adopted to generate these data. This allows us to consider two distinct annotations, which describe GEQ and MBV experiments and that we use in order to reduce the size of the data involved in these two types of experiments. Other experimental data types are available at the OpenGDC public FTP repository, i.e., *Isoform Expression Quantification* [TWP+10], *miRNA Expression Quantification* [ZC03], *Masked Somatic Mutation* [TKR+10], *Copy Number Segment*, and *Masked Copy Number Segment* [CPR+10a], but, conversely to GEQ and MBV, every experiment contains different characterizing information. Thus, we could not consider annotations for these cases.

3.6 Genomic and clinical data persistence

Part of this section was published in [CWC19]

For dealing with data persistency, we adopt a no-SQL document-based Database Management System (DBMS), i.e., MongoDB, to represent all the previously described data types. We use a document-based DBMS both to avoid the problem of being tied to a fixed structure of the data, to vertically and horizontally scale, and to obtain rapid access capabilities. Through this system we are able to represent both semi-structured data as those related to genomic regions, since each type of experiment is defined by specific features, and unstructured data such as clinical data, since the attributes do not follow a well defined schema. Additionally, MongoDB allows us to organize data in different collections, one for each experimental data type. Every collection is defined as a set of documents that are represented as JavaScript Object Notation (JSON) standardized objects. We structured a document to contain one row of the original free-BED files only, which corresponds to an experimental feature on the considered sample. For instance, when taking into account a Copy Number Segment experiment (represented by a specific *.bed* file in the OpenGDC repository) with 355 rows, we added 355 JSON documents to the *Copy Number Segment* collection. In this particular case, every row (i.e., every JSON document) contains a set of information like the genomic coordinates (i.e.: *chromosome*, *start position*, *end position*, and *strand*), the number of probes (*num_probes*), and the segment mean (*segment_mean*), as in the example shown in Listing 3.4.
It is worth noting that we added three more fields to every document: the experiment identifier *aliquot*, the tumor tag *tumor* (the name of the specific tumor related to the considered experiment), and the *source* field that denotes the data source where data have been extracted. These fields have been added to guarantee the correct reconstruction of the original experimental data.

```
{
   "chrom": "chr1",
   "start": 62920,
   "end": 15827002,
   "strand": "*",
   "num_probes": 8317,
   "segment_mean": 0.0031,
   "aliquot": "01175aae-ce8c-4b95-9293-f73329673009",
   "tumor": "tcga-acc",
   "source": "gdc"
}
```

Listing 3.4: JSON representation of a document containing information related to a single row of a Copy Number Segment experiment standardized in free-BED and retrieved from the OpenGDC public FTP repository.

Conversely to all other experimental data types, GEQ and MBV are managed differently. In order to minimize the amount of redundant information we create two additional *annotations* collections. We split every row in the GEQ and MBV free-BED experiments between the *annotation* collection and that one with the information that characterizes the experiment, like in the examples shown in Listing 3.5 and Listing 3.6 respectively. It is worth noting that a field in common between both documents is required to allow the reconstruction of the original row. To make an analogy with a classical relational model, the collections represent distinct tables and this shared field is a *foreign key*.

```
{
  "chrom": "chr1",
  "start": 14404,
  "end": 29570,
  "strand": "-",
  "ensembl_gene_id": "ENSG00000227232.5",
  "entrez_gene_id": "653635",
  "gene_symbol": "WASH7P",
  "type": "gene"
}
```

Listing 3.5: Example of JSON document in the collection representing the *annotation* of the Gene Expression Quantification experiments. It containing the identified redundant information. The same structure is applied to every documents of the same collection.

```
{
  "ensembl_gene_id": "ENSG00000227232.5",
  "htseq_count": 76,
  "fpkm_uq": 28588.2994297,
  "fpkm": 1.21544115087,
  "aliquot": "0ffc0b01-bc1c-4277-8fc4-865590dcc461",
  "tumor": "tcga-acc",
  "source": "gdc"
}
```

Listing 3.6: Structure of a JSON document in the collection containing the Gene Expression Quantification experiments rows. The same structure is applied to every documents of the same collection.

We also include clinical and biospecimen information in a separate collection called *metadata*. In this case, every *meta* file in the OpenGDC repository is represented as a document in the *metadata* collection (Figure 3.7. Here, the procedure of representing these kind of data as documents is pretty simple because of their original key-value structure. The data representation described above allowed us to reduce the size of the whole *bed/tcga* branch of the OpenGDC public FTP repository from $\sim 1.3TB$ to $\sim 0.3TB$, producing a gain of $\sim 77\%$ of storage space also thanks to the built-in data compression features natively provided by the adopted DBMS.

```
{
    ...,
    "biospecimen__bio__year_of_creation": "2013",
    "biospecimen__bio__year_of_shipment": "2013",
    "biospecimen__shared__bcr_patient_barcode": "TCGA-PK-A5HC",
    "biospecimen__shared__patient_id": "A5HC",
    "clinical__acc_shared__mitoses_count":"19",
    "clinical__acc_shared__mitotane_therapy":"YES",
    "clinical__acc_shared__mitotane_therapy_adjuvant_setting":"NO".
    "clinical__acc_shared__mitotic_rate": "Mitotic Rate > 5/50 HPF Present",
    "gdc__access": "open",
    "gdc__aliquots__aliquot_id": "01175aae-ce8c-4b95-9293-f73329673009",
    "gdc__aliquots__concentration": "0.15",
    "gdc__aliquots__source_center": "23",
    "gdc__aliquots__submitter_id": "TCGA-PK-A5HC-11A-11D-A309-01",
    ...
}
```

Listing 3.7: Structure of a JSON document in the collection containing the metadata.

3.7 Data Accessibility

Part of this section was published in [CWC19]

To guarantee an easy, fast, and programmatic access to all the information stored and organized in MongoDB, we designed and developed the framework OpenOmics in order to provide a flexible collection of Application Programming Interfaces (APIs). The complete set of implemented endpoints is available at http://bioinformatics.iasi.cnr.it/api/routes. Our APIs are differentiated in three main groups, reflecting the same collections organization of MongoDB. We indeed release (i) a set of endpoints responsible for the interaction with the collections related to the experimental data types, (ii) another group of

endpoints able to operate on the annotations collections, and finally (iii) one additional set of endpoints for the interaction with the *metadata* collection. In the following, we describe the main implemented endpoints for each of these groups.

Experiment Endpoints

This set of endpoints is able to interact with the collection containing the experimental data. In particular, it allows to:

i. retrieve the complete list of aliquot ids that represent the whole set of processed experiments:
`/experiment/source/<source>/program/<program>/tumor/<tumor>/datatype/<datatype>/aliquots`

ii. extract (a) one single row of the original processed free-BED files or (b) the complete experiment according to the specified data *source*, *program*, *tumor*, *datatype*, *aliquot*, and a particular entity id (e.g.: a methylated site id, a specific ensembl gene id, etc.):

   (a) `/experiment/source/<source>/program/<program>/tumor/<tumor>/datatype/<datatype>/aliquot/<aliquot>/id/<elem\_id>`

   (b) `/experiment/source/<source>/program/<program>/tumor/<tumor>/datatype/<datatype>/aliquot/<aliquot>/all`

iii. extract a list of overlapping genomics coordinates in a specific experiment according to the specified *chromosome*, *start position*, *end position*, and *strand*:
`/experiment/source/<source>/program/<program>/tumor/<tumor>/datatype/<datatype>/aliquot/<aliquot>/overlap/chrom/<chrom>/start/<start>/end/<end>/strand/<strand>`

Annotation Endpoints

This group of endpoints is related to the GEQ and MBV experimental data only. It allows to interact with the redundant information identified in Section 2. In particular, according to a specified*annotation_name* (i.e.: *geneexpression* or *humanmethylation*), it allows to:

i. extract one single document representing the annotation of a specific entity id (i.e.: the methylated site id and the ensembl gene id for the MBV and GEQ data respectively):
`/annotation/<annotation_name>/id/<elem_id>`

ii. retrieve the complete annotation:
`/annotation/<annotation_name>/all`

Metadata Endpoints
By exploiting this set of endpoints, we are able to extract clinical and biospecimen information related to the experimental data stored in set of *experiments* and *annotations* collections. In particular, these endpoint are able to:

i. retrieve the list of all possible values associate to a specific attribute:
`/metadata/attribute/<attribute>/all`

ii. extract the list of all aliquots related to a specific attribute-value couple:
`/metadata/attribute/<attribute>/value/<value>/aliquots`

iii. extract a list of all metadata related to a specific *attribute-value* couple:
`/metadata/attribute/<attribute>/value/<value>/list`

By releasing these APIs, we aim to extend the features that characterize similar software tools like IRIS-TCGA [CWBF16] and GDCWebApp [CF17]. This can represent a preliminary step towards analyses of integrated genomic data as descibed in [WCC+18, CFW18, WDLC+18].

3.8 APIs use cases

Exploiting our data organization and the implemented sets of endpoints presented in Section 3.7 and deeply described in the official APIs documentation available at http://openomics.docs.apiary.io/, we are able to build smart queries to MongoDB and retrieve potentially relevant insights from the stored data. In this Section we illustrate how we made this process quite easy for a researcher with minimal computer science background. In particular, Listing 3.8, Listing 3.9, and Listing 3.10 show three simple use cases in which we adopt the Python 3 programming language to interact with our APIs.

The first one shows how to count the *distinct DNA somatic mutations* in each group of *ethnicity* independently from the *programs* and related *tumors*. It exploits only three endpoints for (i) the identification of the ethnicities of the patients related to experiments stored in our database, (ii) the retrieval of the aliquots related to the experiments conducted on patients with the previously extracted ethnicities, and (iii) the extraction of the DNA somatic mutation experiments related to the extracted aliquots, and finally count the distinct number of DNA somatic mutations.

The second use case shows instead how to easily find the methylated sites (*targets*) overlapped to a specified genomic coordinate (*source*), exploiting only

one endpoint. It is worth noting that both the *source* and *targets* coordinates can refer to particular gene, or isoform, or somatic mutation regions.

The last use case illustrates a scenario in which, starting from a given experimental aliquot id, all the experiments conducted on the same patient related to the specified aliquot are retrieved.

```
import json, urllib.request
apis_base_url = 'http://bioinformatics.iasi.cnr.it/'
source = 'gdc'
datatype = 'maskedsomaticmutation'

# retrieve all the ethnicities in the 'metadata' collection
ethnicity_attribute = 'gdc__demographic__ethnicity'
ethnicities = json.loads( urllib.request.urlopen(
                    apis_base_url+
                    '/metadata/source/{}/attribute/{}/all'
                    .format(source, ethnicity_attribute)
            ).read() )

for ethnicity_value in ethnicities['values']:
    distinct_somatic_mutations = list()
    # retrieve aliquots related to the current ethnicity
    aliquots = json.loads( urllib.request.urlopen(
                    apis_base_url+
                    '/metadata/source/{}/attribute/{}/'+
                    'value/{}/aliquots'
                    .format(source, ethnicity_attribute,
                            ethnicity_value)
             ).read() )

    for aliquot_url in aliquots['hits']:
        if '/datatype/'+datatype in aliquot_url:
            coords_position = aliquot_url.rfind('all')
            coords_url = aliquot_url[:coords_position]+
                         'coordinates'
            # extract the somatic mutation positions available
            # in the experiment related to the current aliquot
            coords_list = json.loads( urllib.request.urlopen(
                                apis_base_url+
                                coords_url
                          ).read() )
            for coordinates in coordinates_list['coordinates']:
                coords_arr = [
                    coordinates['chrom'], coordinates['start'],
                    coordinates['end'], coordinates['strand']
                ]
                if coords_arr not in distinct_somatic_mutations:
                    distinct_somatic_mutations.append(coords_arr)

    print( 'Number of distinct somatic mutation for the '+
```

```
        'ethnicity {} is {}'.format( ethnicity_value,
        str( len(distinct_somatic_mutations) ) ) )
```

Listing 3.8: APIs use case able to count the number of distinct DNA somatic mutations available on all the experimental data in our database. This count is grouped by the ethnicity of the patients on which a this kind of experiment has been performed.

```
import json, urllib.request
apis_base_url = 'http://bioinformatics.iasi.cnr.it/'
annotation = 'humanmethylation'
# genomic coordinates of the WASH7P gene
wash7p = [ 'chr1', 14404, 29570, '-' ]

# extract methylated site coordinates overlapped to WASH7P
for overlap in = json.loads( urllib.request.urlopen(
                  apis_base_url+
                  '/annotation/{}/overlap/chrom/{}/'+
                  'start/{}/end/{}/end/{}/strand/{}'
                  .format(annotation, wash7p[0], wash7p[1],
                      wash7p[2], wash7p[3])
              ).read() )[ 'hits' ]:
   print( str( hit ) )
```

Listing 3.9: This use case show how to exploit our APIs to identify the methylated sites overlapped to a specific genomic region. This is performed by providing a chromosome, start position, end position, and strand of the such region.

```
import json, urllib.request
apis_base_url = 'http://bioinformatics.iasi.cnr.it/'
aliquot_attribute = 'gdc__aliquots__aliquot_id'
aliquot_value = '00168e86-d23a-48ae-8c60-36d970051907'
patient_attribute = 'biospecimen__shared__bcr_patient_barcode'

# retrieve the patient id related to the specified aliquot
patient_barcode = json.loads( urllib.request.urlopen(
                      apis_base_url+
                      '/metadata/attribute/{}/value/{}/list'
                      .format(aliquot_attribute,
                              aliquot_value)
                ).read() )[ 'hits' ][ 0 ][ patient_attribute ]

# extract all the experiments related to the patient barcode
for aliquot in json.loads( urllib.request.urlopen(
                  apis_base_url+
                  '/metadata/attribute/{}/value/{}/aliquots'
```

```
                .format(patient_attribute, patient_barcode)
            ).read() )[ 'hits' ]:
    print( aliquot )
```

Listing 3.10: Use case scenario in which the patient id related to a provided experimental aliquot id is extracted. This patient id is used to retrieve all the experiments conducted on the same patient.

3.9 Taxonomy-based relaxed queries: upward and downward extension

In this Section we consider an extension to query functionalities presented above. We are able to facilitate the accessibility of experimental data and metadata even with a small knowledge about how these data have been modelled. This method is based on the use of domain-specific ontologies (i.e. focusing on the biological domain) able to describe the data provided by GDC and modelled by OpenOmics framework. We implemented this feature by revising the concept of taxonomy-based relaxed query applied to domain-specific ontologies.

Taxonomy-based relaxed queries Following this vision, we based our method accordingly to the *upward* and *downward* theoretical concepts of taxonomy-based relaxed queries [MT14, CW17]. Formally, starting with a taxonomy T, a set of levels $L = l_1, ..., l_n$ of T, a dataset S, and an attribute a of S, if a is provided at a level l_i, T can be used to extend the dataset to a level l_j with:

- $l_j > l_i$: moving to a high level results in a coarser data granularity;
- $l_j < l_i$: traversing the taxonomy from the top to the bottom to achieve a lower level results in a finer granularity of the data.

In particular, we refer to the concept of *upward extension* with the first case in which $l_j > l_i$. The main goal is to store new information in the original dataset providing a higher level in the taxonomy, i.e., at a less fine granularity than the one that is available. On the other hand, the second case in which $l_j < l_i$ is called *downward extension*. The goal here is the same of the *upward extension* (i.e., extend the original dataset with new information) but by providing a more detailed level, i.e., at a finer granularity than that available before.

3.10 SPARQL queries on domain-specific ontologies

Starting from the analysis of metadata treated by the OpenOmics framework, we selected a set of attributes that can represent a specific biological topic. In particular, we exploited the *metadata* set of endpoints to investigate the metadata attributes together with all their possible associated values (i.e., data dictionary), and identify a list of metadata that can be described through domain-specific ontologies. Here we reported some of the selected metadata attributes:

- *clinical__acc_shared__metastatic_neoplasm_initial_diagnosis_anatomic_site*: it describes the anatomical site in which a neoplasm is identified for the first time;
- *clinical__rx__drug_name*: it is the name of the pharmaceutical drug dispensed to a patient affected by a specific type of cancer;
- *gdc__center__name*: it represents the name of the clinic or hospital in which a particular sample extracted by tumoral-affected patients has been sequenced and analyzed;
- *gdc__demographic__ethnicity*: it describes the ethnicity of the patients;
- *gdc__demographic__race*: it is the race of the involved patients (e.g., asian, white, black or african american);
- *gdc__diagnoses__primary_diagnosis*: it is the first diagnosis, including the type of cancer.

We repeated the same process for the experimental data, where we identified the *gene* names and related synonyms as the most atomic descriptive concept that characterize these kind of data.

Each of them represent a concept that can be modelled through a specific ontology. We focused on The Open Biological and Biomedical Ontology (OBO) Foundry [SAR+07], which is a repository of ontologies that explain different sides of the biological world domain. In particular, we extracted from this repository five distinct Web Onotlogy Language (OWL) standardized ontologies:

- the *Geographical Entity Ontology* (GEO), able to describe geographical relations between regions and countries (for the identification of a particular sample sequencing site, or the original country of the patients involved in the experiments);

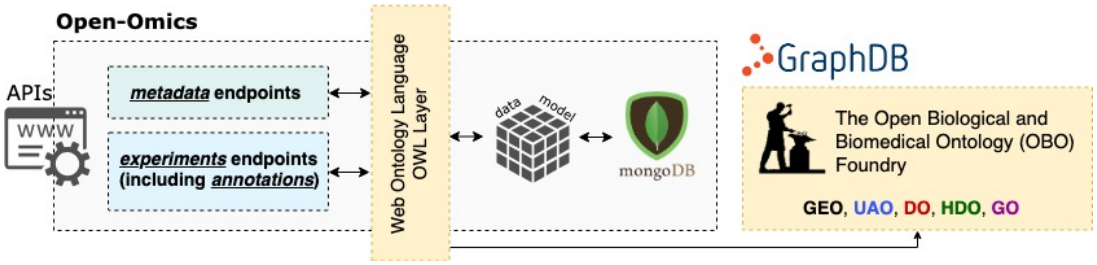

Figure 3.1: The Ontological (OWL) Layer in the OpenOmics Framwork is marked with the orange area. We adopted GraphDB for the management and querying of the OWL-defined ontologies retrieved from The OBO Foundry (i.e., the Geographical Entity, Uberon multi-species Anatomy, Drug, Human Disease, and Gene ontologies). The new application layer is a bridge between the the endpoints and the data model with which data are organized in MongoDB.

- the *Uberon multi-species Anatomy Ontology* (UAO), essential to identify the anatomical regions of some pathologies;
- the *Drug Ontology* (DO), able to identify the chemical components of pharmaceutical drugs dispensed to the patients followed during the whole process of disease development;
- the *Human Disease Ontology* (HDO), for the definition of causes and effects of a series of diseases that affect humans (included different cancer types);
- the *Gene Ontology* (GO), able to describe the function of genes and gene products.

We adopted GraphDB, a no-SQL graph-oriented database management system, to organize the selected ontologies in the same environment, merging them together in a single graph ontological representation. Through GraphDB, we were able to query the produced graph exploiting the integrated SPARQL query engine. SPARQL is for *Simple Protocol and Resource Description Framework (RDF) Query Language*, which is a SQL-like query language able to efficiently search this kind of relational data. This produced a new application layer in the OpenOmics framework, as shown in Figure 3.1.

According to the previously described concept of Taxonomy-based relaxation queries, here we report some promising results based on a couple of char-

acterizing use cases, which make use of two of the selected ontologies (i.e.: the *Human Disease Ontology* and the *Gene Ontology*). We exploit the SPARQL query language to interact with both of them. We also consider UniProt knowledgebase [uni16], a large resource of protein sequences and associated detailed annotation. In particular we expolit the SPARQL endpoint, that allows complex queries of the more than 22 billion triples of data in UniProt (http://sparql.uniprot.org). It is worth noting that no SPARQL queries have to be written by the users in order to interact with the selected ontologies. A set of pre-implemented flexible queries are defined in a new software layer operating after the user request through URLs and directly before the interaction with the OpenOmics endpoints. Here we report a couple of these SPARQL query schemas in order to show a practice use case of both the *upward* and *downward* extension concepts.

Upward extension Here we interact with the Human Disease Ontology and the Uberon multi-species Anatomy Ontology entities through a upward-extended SPARQL query that is shown in Listing 3.11. In this particular case, we are searching for the anatomical region related to a specific type of cancer (i.e. the Breast Invasive Carcinoma (BRCA), called *breast cancer* in the query). We used on the result of this query to retrieve all tumors located in a particular anatomical region (the same of BRCA) and focus on tumor-related experiments exploiting the set of endpoints provided by the OpenOmics APIs. In particular, we extended the *metadata* endpoints with this feature that is automatically used everytime the attributes specified into the URL request are not known metadata attributes.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
SELECT DISTINCT ?obouberon WHERE {
    ?obodoid rdfs:subClassOf ?relation;
           rdfs:label 'breast carcinoma';
           rdfs:subClassOf ?cancer.
    ?cancer rdfs:label 'cancer'.
    ?relation owl:onProperty ?property;
             owl:someValuesFrom ?obouberon.
    ?property rdfs:label 'located_in'.
    ?obouberon rdfs:label ?property_value.
}
```

Listing 3.11: Upward-extended SPARQL query on the Human Disease Ontology and the Uberon multi-species Anatomy Ontology to retrieve the anatomical region (i.e. *located_in*) in which a specific type of tumor (i.e. *breast cancer*) occurs on.

Downward extension This scenario shows an example of downward-extended SPARQL query. Here we interacted with the Gene Ontology (GO) to retrieve the properties of a specific gene in the ontological graph. In particular, as shown in Listing 3.12, we extracted the GO IDs of the gene functions, and their related domains, called namespaces (i.e., biological process, molecular function and cellular component) starting from the gene-product (i.e., protein name) retrieved through querying the UniProt SPARQL endpoint with a specific gene name. This kind of query has the scope to retrieve the biological function related to a gene of interest. Thus, the same query can be used reversing the arguments order, i.e. retrieving a set of genes involved in a particular biological function. These information is powerful to deeply analyze the extracted group of genes and investigate their properties exploiting the set of *experiments* endpoints of the OpenOmics framework (e.g., retrieving their expression values, or obtaining information about the methylated sites that occur in their genomic regions).

```
PREFIX up: <http://purl.uniprot.org/core/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX oboInOwl: <http://www.geneontology.org/formats/oboInOwl#>
SELECT DISTINCT ?goid ?namespace WHERE {
    SERVICE <https://sparql.uniprot.org/> {
        ?protein a up:Protein .
        ?protein up:encodedBy ?gene ;
                 up:classifiedWith ?go.
        ?gene skos:prefLabel 'EAF3' .
        ?go rdfs:label ?golabel.
    }
    ?go oboInOwl:id ?goid;
        oboInOwl:hasOBONamespace ?namespace.
}
```

Listing 3.12: Downward-extended SPARQL query on the Gene Ontology and the UniProt SPARQL endpoint to retrieve GO functions of a specific gene name (i.e., *EAF3* in this case).

3.11 Upward and downward extension of the GDM for taxonomy-based relaxed query with the GMQL

Part of this section was published in [CW17]

Here we apply the Upward and Downward extensions operators, to the GDM, basing on the integration of external taxonomies to provide an extended data schema and to allow taxonomy-based relaxed GMQL queries. As shown in Section 3.4, GMQL operators could be apply to metadata attributes, or to genomic region features. This property allows us to relax GMQL queries extending the GDM, traversing taxonomies upward or downward. In this Section we describe a use case for the upward extension, which involve extension of metadata and two use cases for downward extension, which involve extension of genomic region features.

We explain the intentional meaning of the upward extension application based on metadata, as which defined by the Genomic Data Model. In GDM, for each sample, an attribute-value list is associated. To upward extend this data structure, consider the $i-th$ pair $< a_i,\ v_i >$ for the sample s, and a taxonomy for the attribute a_i, T_a. We query the taxonomy to identify the value v, and starting from that level, $l(v)$ which corresponds to a, we cross it upward, up to the level $l(v)-n$. The reached level represents the new attribute to be included in the metadata, and the associated value is that obtained going through the taxonomy values starting from v.

For the extensional meaning we consider NGS metadata extracted from the OpenGDC FTP repository (3.3), and the following use case:

Prerequisites. The metadata of the samples for TCGA program have an attribute that represents the processing center, which has executed the sequencing of a given tissue. In particular, there are at most two attributes of this type:"center", which as the name of the processing center;"center_id", with the identifier value of the center.

Use Case. Add, to each metadata of each experiment and tumor, the *state* where the sequencing center is located.

Solution. Consider the geographycal taxonomy in Figure 3.2 of the sequencing centers which provides information about name, city and state of the centers. We will traverse the levels of the taxonomy starting from bottom (center name) to top (state of the center) performing an upward extension, for upgrading the information about the sequencing centers to a coarser grain.

Implementation. Retrieve and store the cities of each center, and then

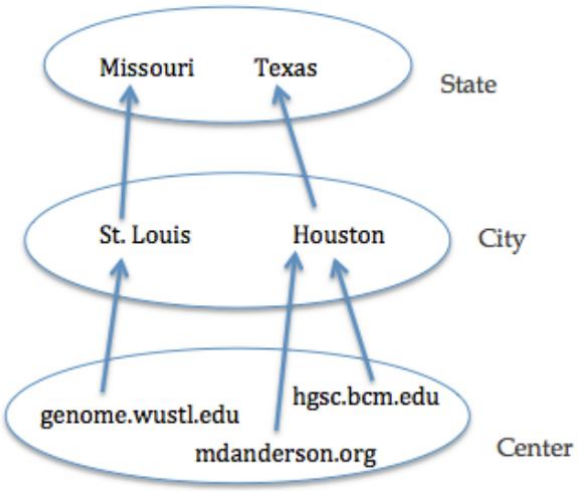

Figure 3.2: Sequencing center taxonomy.

query geographical ontologies such as GeoNames [WIC12], to get the state from the city. GeoNames provides access to geographic information (features), represented by the Resource Description Framework (RDF) data model which can be queried using SPARQL language. As example we report in Listing 3.13 a RDF/XML serialization of GeoNames *feature*.

```
<rdf:RDF
...
    xmlns:gn="http://www.geonames.org/ontology#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    ...
    <gn:Feature rdf:about="http://sws.geonames.org/6955119/">
    <gn:name>St. Louis</gn:name>
    ...
    <gn:parentADM1 rdf:resource="http://sws.geonames.org/4398678/"/>
    ...
</rdf:RDF>
```

Listing 3.13: RDF/XML serialization of GeoNames feature

This feature represent the city and its properties like the name ("name") and the state ("parentADM1"). In Listing 3.14 with the SPARQL query we request the state name of a city whose we specify the name. For the execution of this

```
.....
center hgsc.bcm.edu
center_id       08
.....
center_state Texas
```

Figure 3.3: Example of a metadata Upward-extended file, with the state where the sequencing center is located.

query we used a SPARQL Endpoint, http://factforge.net/sparql [BKO+11], which integrates some linked RDF datasets, including GeoNames.

```
PREFIX geo: <http://www.geonames.org/ontology#>
SELECT DISTINCT ?namestate
WHERE {
 ?o geo:name 'St. Louis';
    geo:parentADM1 ?state.
 ?state geo:name ?namestate.
}
```

Listing 3.14: SPARQL query on the GeoNames feature

Result. The metadata of each sample is extended, inserting the new attribute, as show in Figure 3.3.

We explain the intentional meaning of the downward extension application based on genomic region features, as which defined by the Genomic Data Model. In GDM, for each sample, a genomic regions list is associated. A schema defines specific features for each genomic region. To downward extend this data structure, consider the $i-th$ pair feature-value $< r_i$,$v_i >$ for the sample s, and a taxonomy for the feature r_i, T_r. We query the taxonomy to identify the value v, and starting from that level, $l(v)$ which corresponds to r, we cross it downward, down to the level $l(v) + n$. The reached level represents the new attribute to be included in the genomic region schema, and the associated value will be added in all the genomic region of the sample.

For the extensional meaning we consider NGS experiment extracted from the OpenGDC FTP repository (Section 3.3), and two following use cases.

Prerequisites. Consider the gene identifiers of all genomic regions of the samples of Gene Expression Quntification experiment of TCGA program. The schema defines for each genomic region the gene symbol (*gene_symbol*) and the gene identifier (*entrez_gene_id*):

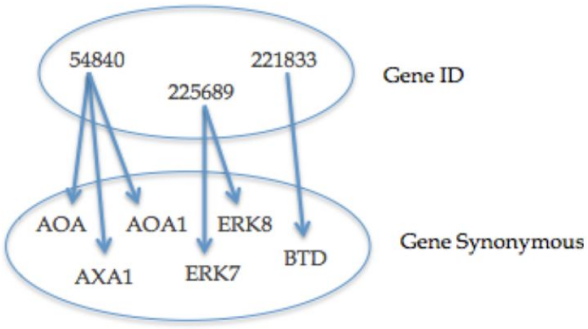

Figure 3.4: Gene taxonomy (1).

$< field\ type = \text{“}STRING'' > gene\_symbol < /field >$
$< field\ type = \text{“}STRING'' > entrez\_gene\_id < /field >$.

Use Case 1. Extend the schema of the genomic regions with a field that corresponds to a possible synonym for the gene related to a given genomic region.

Solution. Consider the gene taxonomy in Figure 3.4 and starting from the *entrez_gene_id* of a gene, retrieve synonyms corresponding to it. The taxonomy is traversed from top to bottom in order to increase the level of detail and to consider the characteristics of the gene to a finer grain.

Implementation. The schema of each experiment is subjected to downward extensions by addition of a new field corresponding to the synonym for the gene: $< field\ type = \text{“}STRING'' >$ **gene_symbol_synonyms** $< /field >$. For each line of each sample the gene id has been selected in order to obtain all the possible synonyms. The data source used for the recovery of synonyms is NCBI, and through the Entrez search engine, it is possible to access the Gene database in NCBI, which collects information on individual genes; this database has been used to obtain synonyms of genes available in the various samples. Entrez uses the API Entrez uses Entrez Programming Utilities (E-utilities), to access different databases. The E-utilities use a fixed URL-based syntax to recover the requested data. In this case the Java language has been used to send the URL to NCBI with following parameters: the Gene database, to which we request the information, the id of the gene, and the desired output format (xml). From the received xml file it has been possible to get all the synonyms of the input gene, thus obtaining for each sample, a corresponding

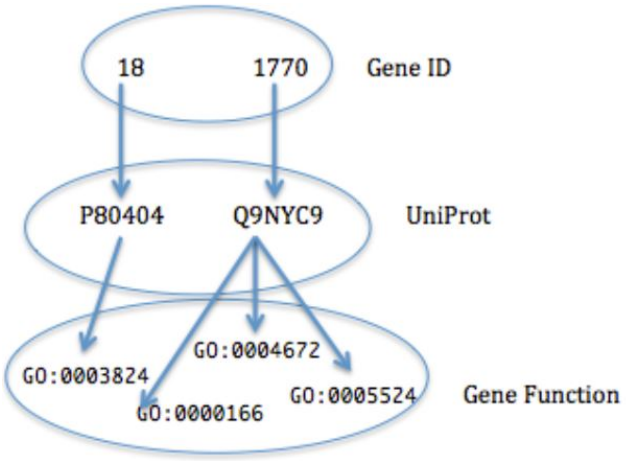

Figure 3.5: Gene taxonomy (2).

list with all the gene ids and its synonyms. Then these synonyms are integrated in each sample file of genomic regions (Table 3.3).

Chrom	Start	End	Strand	Symbol	Symbol_syn	Gene_id
chr8	144801161	144801161	+	MAPK15	ERK7	225689
chr8	144801161	144801161	+	MAPK15	ERK8	225689
chr7	20824445	20824445	+	SP8	BTD	221833
chr9	32988092	32988092	+	APTX	AOA	54840
chr9	32988092	32988092	+	APTX	AOA1	54840
chr9	32988092	32988092	+	APTX	AXA1	54840

Table 3.3: The original genomic regions are duplicated n times, how many synonyms are associated with the gene related to that region.

Use Case 2. Extend the schema of the genomic regions with a field that corresponds to a possible funtion of the gene related to a given genomic region.

Solution. Consider a taxonomy of genes (Figure 3.5) that provides the information about the functions of a particular gene, having his *entrez_gene_id* as input. The goal is to be able to extend the genomic regions with the information on elementary activities of the gene product at molecular level.

Implementation. The schema of each experiment is subjected to downward

extensions by addition of a new field corresponding to the synonym for the gene: $< field\ type = \text{“}STRING'' >$ **gene_function_id** $< /field >$
$< field\ type = \text{“}STRING'' >$ **gene_function_name** $< /field >$.
We exploit the Gene Ontology, a bioinformatics project that unifies all the descriptions of the characteristics of the products of the genes in all species. The databases that mainly populate the information exposed by Gene Ontology are the ones of the UniProt project, the largest bioinformatics database for protein sequences of all living organisms. It is necessary to access to UniProt identifiers of each gene and then take advantage of the Gene Ontology web services to get names and identifiers of the molecular functions of the gene. Then these identifiers and names of the functions are integrated in each sample file of genomic regions (Table 3.4).

Chrom	Start	End	Strand	Symbol	Gene_id	Func_id	Func_name
chr1	94548962	94548962	+	ABCA4	24	GO:0005215	transporter_activity
chr1	94548962	94548962	+	ABCA4	24	GO:0000166	nucleotide_binding
chr1	94548962	94548962	+	ABCA4	24	GO:0016887	ATPase_activity
chr11	14510132	14510132	+	COPB1	1315	GO:0005515	protein_binding
chr11	14510132	14510132	+	COPB1	1315	GO:0005198	structural_mol_activity
chr17	11772554	11772554	+	DNAH9	1770	GO:0003774	motor_activity
chr17	11772554	11772554	+	DNAH9	1770	GO:0016887	ATPase_activity

Table 3.4: The original genomic regions are duplicated n times, how many functions are associated with the gene related to that region.

3.12 Conclusions

In this chapter we have addressed two aspects of biological Big Data management: accessibility and querying. These features are not always carefully considered in the data management cycle, but are required for information sharing. Data sharing is the most effective means for the evolution in the study and treatment of diseases. Furthermore, to make data accessible and easily queried, also allows non-computer scientists to be able to get data from different sources. A critical point in the field of genomics is the imprecise and redundant annotation of clinical and genomic information that is often not connected, and that blocks the advancement of precision cancer care [SLH+16]. Here we have shown two simple genomic data access points: (i)The OpenGDC open access **FTP repository**, containing all the public accessible genomic and clinical data of the TCGA program of GDC, both as originally provided by GDC and converted into the BED format; the latter ones, resulting in more

than 1.5 TB of data. This data are organized by tumor type, and for each tumor we provide genomic data of 7 different experiments, and the respective clinical / biospecimen data (metadata). We also have shown a usage example of these data through the application of GMQL queries to highlight the validity of our approach. These queries demonstrate that our data representation facilitates the analyses, especially thanks to the combination of the filtering on specific clinical / biospecimen attributes and the extraction of genomic features. (ii) we also presented an efficient data organization method applied on genomic and clinical data extracted from the OpenGDC FTP repository exploiting the most recent no-SQL technologies. This method allowed us to detect redundant information and to reduce the size of all the considered data by more than 70%. Additionally, we presented a set of **open-source APIs** able to facilitate genomic data access and extract potentially significant insights from them. We plan to maintain our system constantly synchronyzed with the OpenGDC repository, and to extend our database and APIs by modeling new data from other sources. This is straightforward because of the document oriented and non structured format of our data model. This system represent a step towards the realization of a unified access point to multi-diseases omics data belonging with their clinical and biospecimen information. We rely on this solution to finally converge on a landmark system able to highlight complex biological systems through the integration of different kind of experimental data. Considering this streamlined data representation, we presented different application of the upward and downward extension methodologies, to query a set of data and metadata with no a priori knowledge about how they are modelled and organized. We focused on the biological ontologies domain, in particular on data derived by experiments aimed at investigating the cancer development in humans, obtaining promising results. Exploiting these ontologies leads to a lot of advantages. One of these is the ability to unlock the potential of easily grouping data according to attributes and criteria that are not originally available in the dataset. Additionally, extending the dataset by adding new information produces positive side-effects (e.g., the creation of a new more informative dataset that can be analyzed through machine learning algorithms to better classify data according to new potentially relevant features). We will further investigate the adoption of other domain-specific ontologies to improve The Ontological (OWL) Layer that is in the middle between the user interface and the conceptual schema with which data are modelled.

Additionally we established how it is possible to apply the upward and downward methodologies also to the data previously considered. We extended data in BED format, exploiting the GDM model, obtaining information that

are not previously found in the original dataset (e.g., synonyms of a gene, gene functions, the state in which a sample is sequenced, etc.). The final aim was to enable a user to perform a GMQL taxonomy-based relaxed query to a database that can automatically activate this extension process. A future improvement is to apply the defined extensions to GMQL operations such as selection or join as explained in [MT14]. The relaxation of the query can be provided by introducing the derived operators in taxonomy-based relaxation modalities, i.e., by considering the upward and downward extension of the data.
The extension is based on the field requested by the user, using a specific operator in the relaxed query: the user is guided in the selection of the field that is most suitable to his needs, then an appropriate taxonomy is queried to proceed with the data extension.

Chapter 4

Biological Knowledge Extraction

4.1 Introduction

Interpreting and extracting knowledge from data is one of the primary objectives in bioinformatics. The current applications of machine learning in the field of genomics, combined with NGS technologies, are fostering research of personalized medicine and patient care. In cancer research many computational methods deal with classification problems, e.g., disease characterization, prognosis, treatment response of patients, mutation pathogenicity, biomarker prediction, and sample malignancy. Indeed machine learning has been used for cancer diagnosis and detection and also applied towards cancer prediction and prognosis [CW06]. A recent effort has achieved good performance using state of the art machine learning methods [CWT+14], including Adaboost [PYOA16] and decision trees [MGR+98, LTBD16].

NGS can be applied for case control studies, i.e., specific studies that aim to identify subjects by outcome status at the outset of the investigation, e.g., whether the subject is diagnosed with a disease. Subjects with such an outcome are categorized as cases. Once outcome status is identified, controls (i.e., subjects without the outcome but from the same source population) are selected. In this chapter, we propose to analyze NGS data with supervised machine learning methods [WFB13, PWD+14, WVFB13, WFB12, CFF+16]. The aims are to distinguish the case and the control samples in an effective way.
Classification problems are intended to identify the characteristics that indicate the group to which each sample belongs [TSK05]. A classification model can be used to understand the existing data and to predict to which class a

new sample belongs. The performance of a classifier is measured on the generalization ability, i.e., the ability to give to each new experimental observation the correct class.

Here we consider rule-based classifiers, where the pattern to be found is a set of conditions for which a certain class can be assigned to a sample. The rules are therefore logic formulas that bind a subset of features of the samples to their class label. Example of a logic formula or ("if then" rule) on gene expression data is the following "*if ENSG00000167676.3 < 16.15 OR ENSG00000166819.10 < 15.28 then the sample can be classified as tumoral*". Among rule-based machine learning methods, we focus on a new supervised learning method that is able to extract more knowledge in terms of classification models than state of the art ones, called Classifier with Alternative and MUltiple Rule-based models (CAMUR) [CFF+16]. CAMUR is designed to find alternative and equivalent solutions for a classification problem building multiple rule-based classification models. In particular we propose CamurWeb, a web implementation of CAMUR that is able to extract multiple rule-based classification models from RNA sequencing experiments and to create a large knowledge base of these rules. Moreover, we apply CamurWeb to all public RNA sequencing datasets extracted from The Cancer Genome Atlas database, obtaining a large open access knowledge base of classification rules related to several cancer types. Thanks to its user friendly interface, the tool allows to execute the software CAMUR, to query the results, and to manage the analyzed experiments.

Additionally, we address preprocessing, a required step of data mining, and a prerequisite for accessing to the machine learning step. The elaboration and organization of a biological dataset is defined by several procedures, all grouped together in a step called data pre-processing [Chi17], which allows to prepare and structure the data with a well-defined logic.

In the next sections we describe two different methods for data pre-processing, and the application of different machine learning algorithms. We are going to focus on DNA methylation and RNA sequencing, as these two NGS experiments have been proven to play an important role in knowledge discovery of cancer. We extracted all the samples of these experiments from TCGA, considering different tumor types. For data extraction we consider the TCGA2BED repository that contains genomic, clinical, and biospecimen data in BED format, Every BED file is related to an experiment on a given sample identified

by its TCGA barcode [1], which contains several information about the sample including the type [WJZ16]. The sample type permits to distinguish between normal and tumoral samples, which are the two classes used for classifying the experiments.

For the first method [WCC+18] we extract DNA methylation data, focusing on three types of tumors, i.e., Breast Invasive Carcinoma (BRCA), Prostate Adenocarcinoma (PRAD), and the Thyroid Carcinoma (THCA). We select the samples for which we have the experiments conducted on both case and control tissues, and we perform a processing of data in order to map the methylated sites to the genes where they are located. Additionally, we analyze the processed data with supervised machine learning (i.e., classification) algorithms for identifying the case and the control samples. We select the best performing genes for studying the three types of cancer and therefore we identify many potential oncogenes.

In the second method [CFW18] we address the issue of combining RNA sequencing and DNA methylation experiments, which have different data schemas containing heterogeneous information. Our aim is to obtain a gene oriented organization of both experiments, and therefore we define a new measure on DNA methylation data called *gene methylation quantity*. We consider experiments of three tumors: the Breast Invasive Carcinoma (BRCA), the Kidney Renal Papillary Cell Carcinoma (KIRP), and the Thyroid Carcinoma (THCA). Furthermore we apply machine learning algorithms, and we show the advantage of combining DNA methylation and RNA sequencing data, i.e., the increase of extracted knowledge resulting in combinations of genes from both experimental strategies. Finally, we study the three types of cancer and identify sets of relevant genes. The intersection of them results in a smaller set of genes that should be considered for further investigation.

4.2 Machine learning algorithms for bioinformatics

In bioinformatics, machine learning can be used to analyze multi-omics data sets, through the use of specific algorithms. Machine learning techniques can be divided into two main categories: supervised and unsupervised learning. In supervised learning a model is defined starting from labeled training data, with which we try to make predictions about unavailable or future data. Supervision therefore means that in our set of samples or datasets (e.g. omics-data), the desired output signals (e.g. presence of a disease or not) are already

[1] https://docs.gdc.cancer.gov/Encyclopedia/pages/TCGA_Barcode/

known as previously labeled. In this type of learning, based on labels of discrete classes, we will therefore have a task based on classification techniques [WCC+18, CFW18, WCCF16, PWD+14]. In unsupervised learning, unlike the supervised one the data are not labeled or are not structured, and the algorithm has to develop all the possible pathway that link inputs and output in order to find the general rule that link them. In bioinformatics clustering, i.e. unsupervised learning technique, is widely used. It is an exploratory technique that allows to aggregate data within groups (called clusters) on which we have no previous knowledge of belonging to groups. Large datasets or clusters are defined, each with data that has many similar features [WFB12, ABN+99, OIO+16]. Between the supervised and the unsupervised methods there are several intermediate techinques, such as the semi-supervised learning. In this kind of machine learning categories the initial information are limited, e.g few data in the training set have been labeled [PYKM18].

In the next sections we consider supervised learning, where the class labels of samples (e.g. disease or normal) are given and are used for training the machine learning classification algorithms.

4.3 Supervised data analysis

The aims of our works are to assign an unknown instance to a given class by analyzing its features and to compute a compact and clear classification model. For instance the "if-then" rules (e.g., if $featureM > 2.3$ and $featureX < 0.65$ then the sample is tumoral). The supervised learning approach is adopted: unknown objects are automatically assigned to a class by analyzing their attributes (features) by using a classification model computed from objects with a known class (training set). The classification model can then be applied on a test set for verifying the soundness of it or for classifying new instances whose class is unknown. For further details about supervised classification the reader may refer to [WFF14]. In this section we describe, function-, tree- and rule-based classifiers, which we use for our experiments because they provide the investigator with a compact, clear, and human readable classification model, which permits to identify the features (genes) that are related to the particular cancer under study.

Decision trees: C4.5. Decision Trees are supervised classifiers, which are composed of nodes and edges: internal nodes in the tree are associated to the predicate of the objects of the data set, whereas each edge represents a splitting

rules over one attributes (typically, binary splitting rules). Indeed, every node has two (or more) outgoing branches: one is associated with objects whose attributes satisfy the predicate, whereas the other to the ones which do not. The attribute classes are represented in the tree by leaf nodes. The classification is given by a model that predicts the class of the object by learning simple decision rules inferred from the data features. The class attribute is then assigned to the object by means of a path from the root to the output leaf node, where the predicates are applied to the object attributes and each node defines the path split. The widespread tree decision classifiers, such as C4.5 [Qui14], rely on entropy rule or information gain.

Rule-based classifiers: RIPPER. Rule-based classifiers [WFF14] assign a given class to each object according to a specific function r : *condition* $\rightarrow$ c (called *classification rule*), such that the rule r covers an object x if the attributes of x satisfy the *condition* of r. Therefore, in this type of classification the classifier uses logic propositional formulas in disjunctive or conjunctive normal form ("if then rules") for classifying the given samples.
A rule-based classifier classifies on the basis of the formula triggered by the sample. For extracting a set of classification rules there are two main classes of methods: direct extraction from data and indirect extraction, which extract the rules from other classification models, like Decision Trees. As example for indirect method we can derive from a decision tree the logic formulas whose clauses are represented by the paths from the root to the leaves. RIPPER [Coh95] is a direct rule extraction method based on a pruning procedure, whose aim is to minimize the error on the training set; it performs the following steps: i) growth of the rules; ii) pruning of the rules; iii) optimization of the model; iv) selection of the model. In the first step, thanks to a greedy procedure, RIPPER extracts many classification rules. Then, the rules are simplified and optimized in step two and three, respectively. Finally, the best model (i.e., set of rules) is selected.

Random Forest [Bre01] is an ensemble machine learning method that uses decision trees as basic classifiers. Each tree refers to a class and relies on a random and independent vector, which is generated with the same distribution of the others. The decision trees are trained thanks to the contribution of these random vectors. Random Forest generates distinct decision trees, because it varies the training sets selection and the selected features for each model. The trees are grown to the maximum depth on different training data using a

combination of features, and these trees are not pruned unlike other decision tree methods. For missing or not valid data, the prediction is based on the last preceding node (not a leaf) in the tree. The classification results of a Random Forest execution are computed by counting the votes for the most popular class predicted by the different trees and by assigning that class to the considered instance.

Support Vector Machines [CST00] are a set of supervised learning methods that can be used for both classification and regression. This algorithm, given labeled training data (training set), labeled with the class to which they belong, builds a model (optimal hyperplane) which categorizes new examples into one of the two classes. The algorithm goal is to find an hyperplane with the maximum margin, which is the maximum distance between the data points of both classes.

CAMUR (Classifier with Alternative and MUltiple Rule-based models) is a new supervised method and software package able to extract multiple, alternative, and equivalent classification models [CFF+16]. CAMUR iteratively computes a rule-based classification model, calculates the power set (or a partial combination) of the features present in the rules, iteratively eliminates those combinations from the data set, and performs again the classification procedure until a stopping criterion is verified. CAMUR includes an ad-hoc knowledge repository (database) and a complete querying tool. CAMUR can be successfully applied on genomic and clinical data for the classification of patients (samples).

4.4 Combining DNA methylation and RNA sequencing data

Part of this section was published in [CFW18]

Data processing and combination

We create data matrices of RNA sequencing and DNA methylation experiments in the following way. Consider n samples (tissues) each one with m features (genes) and a class label (condition), which indicates whether the sample is normal or tumoral.

A data matrix is composed by n vectors as $F_i = (f_{i,1}, f_{i,2}, ..., f_{i,m}, f_{i,c})$, which represent sample i, where
$f_{i,j} \in \mathbb{R}$; $i = 1, ..., n$; $j = 1, ..., m$; $f_{i,c} \in \{normal, tumoral\}$.
When considering RNA sequencing, the rows represent the samples, the columns the genes (except the last that represents the class labels) and the items of the matrix contain the RSEM gene expression values for each gene. The structure of this matrix is shown in Table 4.1. When considering DNA methylation,

Sample_ID	$Gene1_{rnaSeq}$	$Gene2_{rnaSeq}$	..	$GeneM_{rnaSeq}$	class
S1	$val_{1,1}$	$val_{1,2}$	..	$val_{1,m}$	normal
..	..	..	..	..	..
Si	$val_{i,1}$	..	..	$val_{i,m}$	..
..	..	..	..	..	..
Sn	$val_{n,1}$	..	..	$val_{n,m}$	tumoral

Table 4.1: Structure of the RNA sequencing matrix.

the corresponding matrix is composed by the rows that represent the samples, the columns that represent the genes, while the items contain a new measure that represent the quantity of methylation associates to each gene and that is explained in the following. Indeed, for DNA methylation TCGA encloses the beta values for each methylated site, so each sample has s methylated sites, l of them belonging to a given gene. For aggregating the methylation quantity at gene level, we consider the sum of the beta values as a measure of the overall intensity of the methylation on a gene. Let a_{ijh} be the methylation quantity associated to the sample i with $i = 1, .., n$, to the gene j with $j = 1, ...m$, and to the methylated site h with $h = 1, .., l$. Then we have $b_{i,j} = \sum_{h=1}^{l} a_{ijh}, \forall i, j$. In the following, we refer to this new measure as *gene methylation quantity*. It is worth noting that we consider the beta values of CpG sites with a related gene symbol, i.e., the symbol of the gene where the methylation occurs. If a methylation occurs on other genomic regions it is not considered in our data processing procedure, whose aim is to provide a gene oriented data organization. In Table 4.2 we show the structure of the DNA methylation matrix. A software tool, which performs the data extraction and the creation of the matrices, is freely available at http://bioinf.iasi.cnr.it/genint. The flowchart that reports the computational steps of the software is depicted in Figure 4.1. In order to perform our analysis on both gene oriented measures (RSEM for RNA sequencing and gene methylation quantity for DNA methylation) at the same time, we propose a combination of these two experiments by applying

Sample_ID	$Gene1_{dnaMeth}$	$Gene2_{dnaMeth}$	..	$GeneM_{dnaMeth}$	class
S1	$b_{1,1}$	$b_{1,2}$	..	$b_{1,m}$	normal
..	..	..	..	..	..
Si	$b_{i,1}$	..	..	$b_{i,m}$	..
..	..	..	..	..	..
Sn	$b_{n,1}$	..	..	$b_{n,m}$	tumoral

Table 4.2: Structure of the DNA methylation matrix.

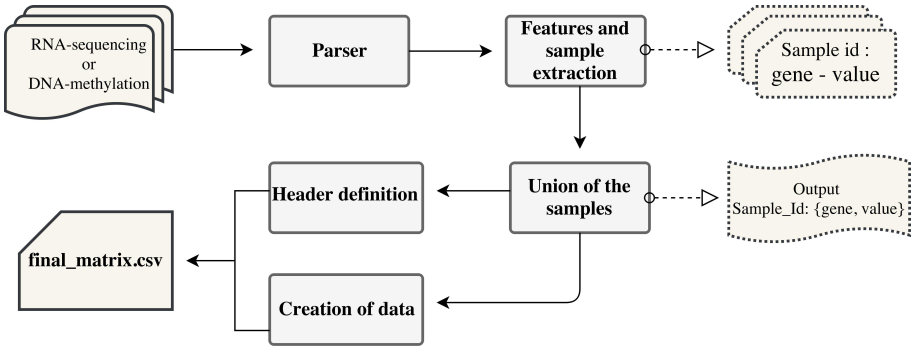

Figure 4.1: The flowchart of the computational steps for creating the RNA sequencing and DNA methylation matrices. The first step represents the parsing of the input dataset of TCGA. The samples are read for the extraction of the features (genes) and their related values, which are the gene expression measures in case of RNA sequencing, or the methylation quantities for each gene in case of DNA methylation. Subsequently, the samples and the related gene-value pairs are unified in a single file. From this file the header (columns) and the values (rows) of the matrix are created. In the final matrix (comma separated values format), the header reports all the genes, while the rows are identified by the sample id and report the related values.

an intersection of the matrices on common sample IDs and a union of those not in common (this result trace over the full outer join in SQL language), keeping both experimental data and performing a union of the genes that are present in RNA sequencing and DNA methylation as features. The resulting matrix is shown in Table 4.3. Let i be the *i-th* sample, j the *j-th* gene, with $i = 1, .., n$ and $j = 1, .., m$ in case of a gene of the DNA methylation experiment, $j = m + 1, ..., z$ in case of a gene of the RNA sequencing experiment. Furthermore, we have $b_{i,j}$ and $val_{i,j} \in \{\mathbb{R},?\}$, $\forall j, i$, where the "?" symbol means

Sample_ID	$G1_{dnaMeth}$	$G2_{dnaMeth}$	..	$G1_{rnaSeq}$	$G2_{rnaSeq}$	..	class
S1	$b_{1,1}$	$b_{1,2}$	..	$val_{1,m+1}$	$val_{1,m+2}$	..	normal
..	..	..	..	..	..	..	..
Si	$b_{i,1}$	..	..	$val_{i,m+1}$	..	$val_{i,z}$	..
..	..	..	..	..	..	..	..
Sn	$b_{n,1}$	..	..	$val_{n,m+1}$	..	..	tumoral

Table 4.3: Structure of the combined matrix.

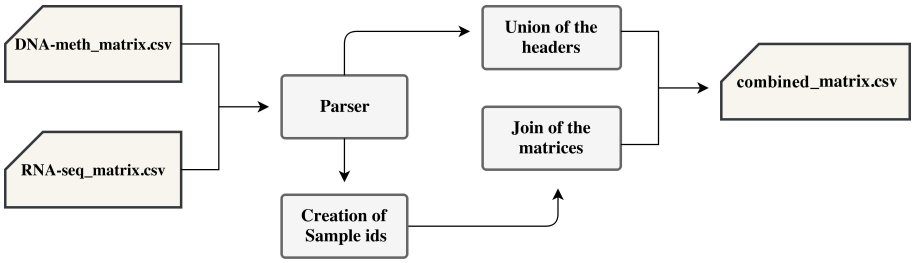

Figure 4.2: Flowchart for creating the combined matrices. Firstly, a parser reads the DNA methylation and RNA sequencing matrices in input (computed as described in Figure 4.1), and sends the next elaborations to two distinct processes. A step is responsible of the creation of the full header of the combined matrix with all the genes, of both the DNA methylation and RNA sequencing. The other step takes the parsed sample IDs to modify the identification of the sample (TCGA barcode) deleting the details of the performed experiments. After the creation of the sample IDs, the join step follows: the initials matrices are joined on the modified IDs and the new rows of the matrix are created, including both gene expression and gene methylation quantity. The join defines the rows with the values of the two experiments (on which the join is made because the sample id is present in both input matrices), and also the rows with values of only one experiment (if the sample is not available in both input matrices of DNA methylation and RNA sequencing).

that there is no value associated to the gene j for the sample i. We release also a software that is able to perform the combination of the experiments, available at http://bioinf.iasi.cnr.it/genint. Finally, we report the main steps of the procedure in Figure 4.2.

Analysis method

The classification that we perform has the objective of being able to determine a set of rules for each type of tumor (composed of the genes and their related values), which can define if a tissue is in tumoral or in normal condition.

It is worth noting that also in previous works [MAOP01, BLZ+06, MWZG13, STSC14, WCCF16, PRP+17] DNA methylation has been used to classify data samples (and patients) of cancer, but only a subset of single methylated sites have been used as features. Recently, the authors of [CCW18] perform the classification task by considering all the single methylated sites in the genome with Big Data techniques. Conversely, we use the previously defined gene methylation quantity and not the single methylated sites. Also gene expression data of RNA sequencing has been widely used for cancer classification [GST+99, KWR+01, SNM+03, WFB12, MdRD+13, NRE14, WFFB15, CFF+16] and proven to be effective in distinguishing normal from tumoral samples. Our aim is to combine both information in order to extract a wider knowledge and to better focus on those genes that are related to the disease.

For performing the task of knowledge discovery, we used four different classification algorithms (C4.5, Ripper, Random Forest, CAMUR).

Performed tests

We describe the performed experiments to test our method and the results of the classification algorithms applied to the RNA sequencing and DNA methylation data of three cancer types (BRCA, THCA, KIRP).
For each type of tumor, we created three data matrices, the first containing only the gene expression values (RNA sequencing), the second containing only the gene methylation quantities, and the third combining both experiments according to the procedure described in Methods. Table 4.4 and Table 4.5 show an example of the RNA sequencing and of the DNA methylation data matrices of BRCA. The numeric values shown in Table 4.5 are obtained as sums of the beta values associated with the same gene. For example for the TCGA-A7-A4SD-01A-11D-A268-05 sample, the 1.6 value, associated with the GDA gene, is the result of the sum of all the beta values of the methylation sites associated with this gene. Table 4.6 shows an example of the BRCA combined matrix, where each column represents a gene of the DNA methylation and then a gene of the RNA sequencing; the first row represents a sample only with data about the RNA sequencing experiment, and the second row has only data for the DNA methylation experiment, whereas the last row is an example

RNA sequencing	GDA_rna	SCN3A_rna	SCN3B_rna	class
TCGA-A1-A0SD-11A-11R-A115-07	0.6	28.4	43.6	normal
TCGA-A7-A4SD-01A-11R-A266-07	0.0	6.4	1.9	tumoral
..	..	..	..	..
TCGA-3C-AALK-01A-11R-A41B-07	0.0	5.3786	18.2044	tumoral

Table 4.4: Example of RNA sequencing matrix on breast cancer data. The columns represent genes, and the last shows the class. For each row, we consider the full TCGA identifier of the sample. The full identifiers is called TCGA aliquot and reports the type of the performed NGS experiment (RNA sequencing). The gene expression values are reported for all samples.

DNA methylation	GDA_dMeth	SCN3A_dMeth	SCN3B_dMeth	class
TCGA-A7-A4SD-01A-11D-A268-05	1.6	2.3	2.0	tumoral
TCGA-GI-A2C9-01A-11D-A21R-05	1.9	2.7	2.3	tumoral
..	..	..	..	..
TCGA-3C-AALK-01A-11D-A41Q-05	3.8	2.1	3.8	tumoral

Table 4.5: Example of DNA methylation matrix on breast cancer data. Also in this case, rows are represented by the TCGA aliquot of samples, reporting the type of the performed NGS experiment (DNA methylation). The gene methylation quantity values are reported for all samples.

of a sample with both experiments. Details about the combined matrices of

Combined	GDA_dMeth	SCN3A_dMeth	...	GDA_rna	SCN3A_rna	class
TCGA-A1-A0SD-11A	?	?	...	0.6	28.4	normal
TCGA-GI-A2C9-01A	1.9	2.7	...	?	?	tumoral
..	..	..	..	..	..	..
TCGA-A7-A4SD-01A	1.6	2.3	...	0.0	6.4	tumoral

Table 4.6: Example of combined matrix on the breast cancer data. In the combined matrix, rows are identified by the TCGA Barcode (excluding the part that identifies the type of experiment carried out on a sample). In this way it is possible to recognize the sequenced sample with both NGS techniques (RNA sequencing and DNA methylation). In this case the matrix has as many rows as the total samples (union of RNA sequencing samples and DNA methylation samples), counting only one time those samples in common, on which both experiments were preformed.

BRCA, THCA, and KIRP tumors are summarized in Table 4.7, while details about the datasets are depicted in Table 4.8: with the column 'Experiment' we specify the sequencing experiment (RNA sequencing or DNA methylation), followed by the 'Cancer' column where we indicate with a code the considered

types of cancer for the two experiments. The last four columns represent the number of tumor samples, the number of normal samples, the total number of genes and the size of the matrices in MB, respectively. We performed binary classifications (two classes, normal and tumoral), and we considered three cancers with both normal and tumoral samples.

	KIRP	THCA	BRCA
# RNA sequencing samples	28	9	346
# DNA methylation samples	22	8	23
# DNA methylation and RNA sequencing samples	295	563	872

Table 4.7: Details of the number of samples in the combined matrices. # RNA sequencing samples represent the number of samples having only RNA sequencing data, # DNA methylation samples represent the number of samples having only DNA methylation data, # DNA methylation and RNA sequencing samples represent the number of samples having both information.

Experiment	Cancer	tumoral	normal	features	MB
RNA sequencing	BRCA	1104	114	20485	198,5
	THCA	513	59	20489	93
	KIRP	291	32	20489	52,6
DNA methylation	BRCA	799	98	20045	330
	THCA	515	56	20045	210,2
	KIRP	274	43	20045	116,9
Combined	BRCA	1114	127	40530	542,5
	THCA	515	65	40534	303,9
	KIRP	292	53	40534	171,4

Table 4.8: Overview of the datasets.

The data matrices of the different experiments and tumors have been analyzed with the above-mentioned classification algorithms (C4.5, Random Forest and RIPPER) through the use of the Weka software package [HFH+09]. For the application of these algorithms, we adopted a parameter tuning process to prevent overfitting and to optimize the classification results in term of accuracy. We used the Cross-Validated Parameter selection (CVParameterSelection) [Koh95], that can optimize an arbitrary number of parameters according to input data and number of cross validation folds. We have chosen this meta-classifier for performing parameter selection by cross-validation for all our classifiers. For example, if we consider the RNA-sequencing matrix for

KIRP tumor, and the different classifiers (RIPPER, C4.5 and Random Forest), we obtain the following results:

- J48 (C4.5), -C (confidence threshold for pruning.) 0.1, -M (minimum number of instances per leaf) 1, -U (use unpruned tree) false;
- JRip (RIPPER), -F (the number of folds for Reduced Error Pruning) 5, -N (the minimal weights of instances within a split) 1, -O (the number of runs of optimizations) 2 -S (the seed of randomization) 1;
- RandomForest, -I (number of iterations) 30, -K (number of attributes to randomly investigate) 0, -S (seed for random number generator) 1, -num-slots (number of execution slots) 1.

In addition, we performed the classifications with multiple rule-based models obtained by CAMUR. Therefore four different classification algorithms were applied on three data matrices (RNA sequencing, DNA methylation, and their combination) of each considered cancer, resulting in 36 different knowledge discovery analyses. For evaluating the classifiers we take into consideration the F-measure, which is defined as $F\text{-}measure = \frac{2P \cdot R}{P+R}$, where R stands for $Recall$ and P is for $Precision$. Considering True Positives (TP) objects of a given class recognized in this class; False Positives (FP) objects recognized in a class but not belonging in this class; True Negatives (TN) objects not belonging and not recognized in a given class; False Negatives (FN) objects in a given class but not recognized in that, we can then define $Recall = \frac{TP}{TP+FN}$ and $Precision = \frac{TP}{TP+FP}$. We performed the tuning of parameters also for CAMUR adopting the Cross-Validated Parameter selection described above [Koh95] for its internal RIPPER algorithm, and we finally set the execution mode to loose, the maximum number of iterations to 100, the minimum F-measure value to 0.8, and the maximum time to 30 days.

In Table 4.9 we show the average of the resulting F-measures for the performed classifications of each algorithm in 10-fold cross validation scheme. It is worth noting that all values are greater than 95%. Proper parameter tuning was performed with a large set of tests in order to prevent potential overfitting of the classification models. The results obtained on the combined datasets are slightly lower due to the increase in features and missing values that make the job of the classification algorithms harder. In order to clarify this point we also applied the classification algorithms on the combined matrices, deleting the samples for which only one NGS experiments is available. In this way

Experiment	Cancer	C4.5	RIPPER	CAMUR	Random Forest
RNA sequencing	BRCA	98.5	98.1	98.2	97.3
	THCA	97.7	97.2	97.6	98.4
	KIRP	98.8	98.8	95.2	99.4
DNA methylation	BRCA	97.2	97.5	97.4	98.3
	THCA	96.1	96.3	95.1	97.0
	KIRP	97.8	96.5	98.0	99.0
Combined	BRCA	97.2	97.5	97.8	98.9
	THCA	96.4	95.2	97.2	97.3
	KIRP	98.0	96.8	98.4	98.2

Table 4.9: Average performance (F-measure) of the classification algorithms.

we reduced the missing values and the resulting classification performance (F-measure) improved with all the classifiers (i.e., on BRCA +0.2% with C4.5, +0.1% with RF, +1% with RIPPER; on KIRP +0.7% with C4.5, +0.8% with RF, +2,1% with RIPPER; on THCA +0.1% with C4.5, +0.5% with RF, +0.2% with RIPPER).

The performance of the algorithms are important in order to validate the classification, but the main purpose of the work is to extract more and different genes from diverse experiments. The improvement given by the classification of the combined data is that the resulting classification models do not only consider the genes and their associated values for a single experiment, but both from gene expression and DNA methylation data in a single model, providing multiple related genes. In Table 4.10 we show the number of genes obtained with the execution of RIPPER, C4.5 and Random Forest classification algorithms on all tumors for DNA methylation, RNA sequencing and their combination. It is worth noting that one extracted gene is in common among the three algorithms. The reason is that the algorithms operate differently and use diverse extraction functions of the models, so the extracted features are disjoint. It is important to distinguish between Random Forest that extracts multiple classification models, while RIPPER and C4.5 extract a single classification model. We obtain almost 5000 genes with Random Forest, 38 genes with RIPPER, and 26 with C4.5; we also report that 17 genes are in common between RIPPER and Random Forest, 9 between C4.5 and Random Forest, and 4 between RIPPER and C4.5. Further details and the complete list of extracted genes are at http://bioinf.iasi.cnr.it/genint. We also investigated if the algorithms misclassify the same samples by comparing the predictions of each one. We found out that only some instances are misclassified by all the

Algorithm	RNA-Seq and DNA methylation	Combination
RIPPER	26	12
C4.5	22	4
Random Forest	2098	2471

Table 4.10: Number of genes obtained with the different classification algorithm . We show the number of genes obtained with RNA sequencing and DNA methylation data matrices, and the number of genes obtained thanks to the combination of the two experiments.

three algorithms.

Finally, in order to prove the validity of the extracted models we performed random permutations of class membership for each classification problem and each combination. We tested if our procedure is able to extract meaningful classification models regardless of the class partition imposed on the training set. This would be verified only in the presence of a marked overfitting behavior. For validating our results and the extracted classification models, we applied the procedure to data with random permutations of class labels. This validation test was performed on 100 different random permutations for each classification problem. In particular, we obtain low values of F-measure and we report the resulting averages in Table 4.11. We obtain a low overall average classification accuracy on permutated data, whose values are halved when compared to the ones obtained on original data. This confirms the reliability of our classification models and suggesting the absence of overfitting when considering the correct classes. We ran more than 2000 classification procedures with

Experiment	Cancer	C4.5	RIPPER	CAMUR	Random Forest
RNA sequencing	BRCA	51.1	51.6	50.2	50.9
	THCA	49.5	50.8	49.6	50.7
	KIRP	55.4	48.8	50.3	50.1
DNA methylation	BRCA	50.0	49.7	51.1	49.1
	THCA	51.1	50.2	53.2	47.9
	KIRP	50.2	49.6	52.0	50.8
Combined	BRCA	51.9	49.4	52.6	49.9
	THCA	52.4	50.7	50.1	51.3
	KIRP	50.1	50.4	50.3	50.2

Table 4.11: Average performance (F-measure) of the classification algorithms on random permutated class labels.

CAMUR, obtaining rules, literal and conjuncion lists, feature pairs and literals statistics for each tumor and each considered dataset. Detailed results are described at http://bioinf.iasi.cnr.it/genint. In Table 4.12 we summarize the results obtained with CAMUR, in particular the table shows the total number of extracted rules and all the features (i.e., genes) that appear. In Table 4.13 we

Experiment	Cancer	rules_list	genes
RNA sequencing	BRCA	1866	920
	THCA	1880	695
	KIRP	3	2
DNA methylation	BRCA	2658	1543
	THCA	3778	1918
	KIRP	159	53
Combined	BRCA	895	1045
	THCA	3703	1450
	KIRP	310	88

Table 4.12: Rules and genes obtained with CAMUR. This results summarize the obtained output for each considered tumor and experiment.

report the execution times, the number of iterations and the execution mode of CAMUR. The execution of the classifications procedures were run on a 4-Core 3 giga hertz Intel-7 processor with 24 gigabytes RAM and Linux Debian Kernel Version 2.6.26-2-amd64. The classifications obtained with the implementations of C4.5, Random Forest, and RIPPER algorithms, are executed with two software tools available at http://bioinf.iasi.cnr.it/genint. In Table 4.14 we report the execution times of the classification procedure for each tumor in 10-fold cross-validation sampling scheme [TSK05]. Conversely to CAMUR, the execution times are in the order of minutes, because those algorithm extract just a single classification model. We also compared the execution times of Random Forest to those of CAMUR, which extract both multiple solutions. We note that CAMUR has higher running times than Random Forest, which are in the order of hours for CAMUR and in the order of minutes for Random Forest. We can justify this differences by considering the amount of logic formulas extracted from both classifiers, indeed CAMUR extracts many more rule-based models w.r.t. Random Forest tree-based ones.

Experiment	Cancer	CAMUR_time	iterations	mode
RNA sequencing	BRCA	14d:20h:59m:20s	60	loose
	THCA	05d:04h:00m:51s	100	loose
	KIRP	00d:00h:01m:22s	100	loose
DNA methylation	BRCA	29d:00h:21m:19s	44	loose
	THCA	29d:00h:19m:52s	39	loose
	KIRP	00d:00h:25m:51s	100	loose
Combined	BRCA	29d:20h:21m:25s	7	loose
	THCA	07d:20h:53m:16s	100	loose
	KIRP	00d:01h:34m:08s	100	loose

Table 4.13: Timing of the CAMUR executions, number of iterations and execution mode. We specified different maximum number of iterations according to the computation time, 80% as minimum threshold value for the classification reliability, and loose as execution mode. It is worth noting that only 7 iterations in 29 days have been performed for the combined matrix of BRCA, because the extracted classification models are composed of a high number of genes.

Experiment	Cancer	C4.5_time	RIPPER_time	RandomForest_time
RNA sequencing	BRCA	04m:07s	09m:09s	00m:48s
	THCA	01m:28s	02m:30s	00m:30s
	KIRP	00m:27s	00m:46s	00m:16s
DNA methylation	BRCA	02m:53s	06m:10s	00m:37s
	THCA	01m:34s	03m:12s	00m:34s
	KIRP	00m:45s	01m:02s	00m:22s
Combined	BRCA	06m:31s	10m:20s	6m:38s
	THCA	01m:58s	3m:35s	00m:28s
	KIRP	01m:10s	01m:45s	02m:55s

Table 4.14: Execution time of C4.5, RIPPER and Random Forest algorithms.

Gene methylation quantity

In previous studies, efforts have been made for aggregating DNA methylation at gene level. In [SRD+15] a methylation index is defined as the mean percent methylation across all CpG sites in the gene. In [VHSE+13] another methylation index is defined as the ratio of methylated and unmethylated copy numbers measured by absolute quantitative assessment of methylated alleles. Our measure differs from previous attempts to represent DNA methylation at gene level, because it takes into account both the number and the values of methy-

lated sites for each gene. In order to validate the gene methylation quantity we provide a qualitative and a quantitative explanation: (i) the defined gene methylation quantity index represents how much a gene is methylated, because it is defined as the sum of the methylation values of the sites that are within the genomic coordinates of the gene, therefore if the gene methylation quantity is low/high, than the gene will be low/high methylated; (ii) we have shown that four classification algorithms are able to successfully distinguish tumoral from non tumoral samples by considering the gene methylation quantities as features. In addition, the index provides a gene oriented data representation of the DNA methylation experiment.

Correlation between DNA methylation and RNA sequencing

An interesting problem is to investigate if there is correlation between gene expression and DNA methylation. The authors of [MNB+10, AGBK+12, KHB+12, CZC+14, KME+15, LLDS18]) address the question if there is correlation between the expression values and the methylated sites of a gene in cancer data and prove that a correlation exists only for a few set of genes. Specifically for the Breast Invasive Carcinoma, in [FFJ+14] the correlation between DNA methylation and gene expression of almost 3,000 genes is discussed, and in [SBJ+17] it is shown how the CpG-SNP (partnership between DNA methylation and Single Nucleotide Polymorphism) pairs are strongly associated with differential expression of genes. Indeed, DNA methylation has been related also to mutations, and it has been proven that Single Nucleotide Polymorphism at specific loci can result in different patterns of DNA methylation [SKK+14].

Tree-based classification models of C4.5

We extracted a classification model for each experiment and each cancer with C4.5, resulting in 9 decision trees composed of 26 genes (16 for DNA methylation and 10 for RNA sequencing). We show some examples on the Kidney Renal Papillary Cell Carcinoma (KIRP) data, in Figure 4.15 we report the RNA sequencing decision tree, and Figure 4.16 shows it for the combined data. The classification models on the other tumors and experiments are available at http://bioinf.iasi.cnr.it/genint.
In the leaves of the trees the total weight of instances reaching that leaf, and the total weight of misclassified instances are specified. In each leaf a fractional weight representing the instances with a missing value is considered. We can then see how missing values are handled by comparing the Figures 4.15 and

4.16. In Figure 4.15, the weights of instances are integers, whereas in Table 4.16 weights are all fractional values, due to the fact that in the combined matrix most of the instances contain missing values. As we can see the model obtained from the combined data provides additional knowledge in the resulting classification rules, compared to that obtained from RNA sequencing data. In particular, the first rule of the model in Table 4.15 is enriched with additional rule conditions on the genes of the DNA methylation data, as shown in Figure 4.16.

UMOD_rnaSeq $\leq$ 2370.6675: tumoral (291.0)
UMOD_rnaSeq $>$ 2370.6675: normal (32.0)
Number of leaves: 2
Size of the tree: 3

Table 4.15: The decision tree for full training set, obtained from the RNA sequencing KIRP data matrix, with 319 correctly classified instances and 4 incorrectly classified instances.

UMOD_rnaSeq $\leq$ 2370.6675
‖ VMP1_dnaMeth $\leq$ 5.468451: tumoral (291.59/1.8)
‖ VMP1_dnaMeth $>$ 5.468451: normal (19.23/2.11)
UMOD_rnaSeq $>$ 2370.6675: normal (34.18/0.1)
Number of Leaves: 3
Size of the tree: 5

Table 4.16: The decision tree for full training set, obtained from the combined KIRP data matrix, with 338 correctly classified instances and 7 incorrectly classified instances. The two experiments are considered, then tumor and normal tissues are defined, both by RNA sequencing RSEM measures and DNA methylation beta values.

Finally, we validated the tree-based RNA-sequencing classification models on two external datasets extracted from Gene Expression Omnibus [EDL02] (GSE56022 and GSM1308330), obtaining 90% correct classification.

Tree-based classification models of Random Forest

We applied Random Forest to all matrices, extracting 9 classification models, each one composed of 30 trees. The total number of genes obtained is 2301 for RNA sequencing and 2574 for DNA methylation of which 306 are in common between two experiments. Thanks to their combination, we extracted 2471

genes with the execution of this algorithm. As example, we show in Table 4.17 a random tree obtained with the application of Random Forest on the DNA methylation matrix of Breast Invasive Carcinoma (BRCA) data. The Random

```
DNM2_dnaMeth < 3.68
|   CRYAB_dnaMeth < 2.01
|   |   AUNIP_dnaMeth < 1.51 : tumoral (27/0)
|   |   AUNIP_dnaMeth ≥ 1.51
|   |   |   NPY_dnaMeth < 4.85
|   |   |   |   SACM1L_dnaMeth < 2.65
|   |   |   |   |   LINC00336_dnaMeth < 4.05 : tumoral (2/0)
|   |   |   |   |   LINC00336_dnaMeth ≥ 4.05 : normal (8/0)
|   |   |   |   SACM1L_dnaMeth ≥ 2.65 : normal (76/0)
|   |   |   NPY_dnaMeth ≥ 4.85
|   |   |   |   PLEKHM2_dnaMeth < 20.49 : tumoral (9/0)
|   |   |   |   PLEKHM2_dnaMeth ≥ 20.49 : normal (5/0)
|   CRYAB_dnaMeth ≥ 2.01 : tumoral (114/0)
DNM2_dnaMeth ≥ 3.68
|   SPRYD4_dnaMeth < 1.3 ||   ||   DAP3_dnaMeth < 2.21
|   |   |   MYOG_dnaMeth < 11.05 : tumoral (646/0)
|   |   |   MYOG_dnaMeth ≥ 11.05
|   |   |   |   GMEB2_dnaMeth < 9.57 : tumoral (1/0)
|   |   |   |   GMEB2_dnaMeth ≥ 9.57 : normal (1/0)
|   |   DAP3_dnaMeth ≥ 2.21 : normal (1/0)
|   SPRYD4_dnaMeth ≥ 1.3
|   |   LBX1-AS1_dnaMeth < 10.63 : normal (5/0)
|   |   LBX1-AS1_dnaMeth ≥ 10.63 : tumoral (2/0)
Size of the tree: 25
```

Table 4.17: Model A tree of the classification model for full training set, obtained by the execution of Random Forest on DNA methylation data of Breast Invasive Carcinoma. The full output is composed by 30 trees with different sizes with multiple leaves containing also the total weight of instances.

Forest algorithm is particularly suited for knowledge extraction on combined data (which presents a high number of features), because of its randomized and multiple model extraction.

For the validation of the classification models on the external datasets GSE56022 and GSM1308330, we consider all trees generated for the RNA-sequencing experiment, and the samples are classified with an average accuracy of 80%.

Rule-based classification models of RIPPER

The RIPPER algorithm provides 9 rule-based classification models composed of 38 genes, 22 for DNA methylation and 16 for RNA sequencing. Below we show some examples of the rule-based classification models obtained with the RIPPER algorithm on the Kidney Renal Papillary Cell Carcinoma (KIRP). We show some rules for the DNA methylation dataset in Table 4.18, and for the combined dataset in Table 4.19. For example the rule depicted in Figure 4.19 can be interpreted as: classify the considered sample into normal, if the gene methylation quantity of MAP3K11 is lower-equal then 12.3 and the one of PIP5KLI is greater-equal then 2.1 or the RSEM RNA-Seq value of NELL1 is greater-equal then 437.3. Conversely, assign the sample to the tumoral class. The reader may find all the classification models in http://bioinf.iasi.cnr.it/genint.

(MAP3K11_dnaMeth $\leq$ 12.3) and (PIP5KL1_dnaMeth $\geq$ 2.1) $\rightarrow$ class=normal
$\rightarrow$ class=tumoral
Number of Rules: 2

Table 4.18: Rule-based model for full training set, obtained from the DNA methylation KIRP data matrix, with 306 correctly classified instances and 11 incorrectly classified instances.

(MAP3K11_dnaMeth $\leq$ 12.3) and (PIP5KL1_dnaMeth $\geq$ 2.1) $\rightarrow$ class=normal
(NELL1_rnaSeq $\geq$ 437.3)$\rightarrow$ class=normal
$\rightarrow$ class=tumoral
Number of Rules: 3

Table 4.19: Rule-based model for full training set, obtained from the combined KIRP data matrix, with 334 correctly classified instances and 11 incorrectly classified instances. Also in this case, features of both experiments appear in the extracted rule.

The rules in the resulting model obtained from the combined matrix, confirm the added value. In this model we can find the same rule obtained from the single DNA methylation data, enriched with a new rule-based on a feature derived from RNA sequencing data.

We also applied the rule-based RNA-sequencing model extracted on the Gene Expression Omnibus datasets, obtaining a correct classification rate of 90% on average.

Rule-based classification models of CAMUR

By running more than 2000 classification procedures, we extracted 15.252 rules composed of 1758 genes from RNA sequencing and 3655 genes from DNA methylation. From those genes 509 are in common in both experiments. The reader may find the gene lists at http://bioinf.iasi.cnr.it/genint. In this subsection, we show some example of the rules obtained through the execution of CAMUR on the Thyroid Carcinoma (THCA) data. CAMUR extracts many multiple classification models (available at http://bioinf.iasi.cnr.it/genint), as example we report only those with the highest level of accuracy in Table 4.20. The rules for the combined data classification model confirm what is derived

(TMEM127_dnaMeth $\geq$ 1.99) and (IRGM_dnaMeth $\geq$ 1.79) and (SCN3A_dnaMeth $\geq$ 2.88) OR (TMEM2_dnaMeth $\geq$ 1.206751) and (IL2RA_dnaMeth $\leq$ 6.32) and (NENF_dnaMeth $\geq$ 1.88) OR (AWAT2_dnaMeth $\geq$ 3.62) and (SNORA69_dnaMeth $\leq$ 1.89) $\rightarrow$ class=normal
(TNFRSF12A_dnaMeth $\geq$ 0.53) and (SGK2_dnaMeth $\leq$ 11.15) OR (TMEM127_dnaMeth $\geq$ 2.07) and (OR10J1_dnaMeth $\geq$ 2.96) and (SCN3A_dnaMeth $\geq$ 2.92) $\rightarrow$ class=normal
(TNFRSF12A_dnaMeth $\geq$ 0.53) and (CDKN1C_dnaMeth $\leq$ 12.92) OR (TMEM127_dnaMeth $\geq$ 2.04) and (ADH4_dnaMeth $\leq$ 0.79) and (GFER_dnaMeth $\geq$ 2.07) $\rightarrow$ class=normal

Table 4.20: Three classification models shown for full training set, of DNA methylation experiment for Thyroid Carcinoma, with about 100% level of accuracy.

from the model for DNA methylation data, and also provide a further classification rule-based on RNA sequencing genes that is not obtained by performing classification on data of single experiment of RNA sequencing. It is worth noting that CAMUR can be successfully adopted for knowledge extraction on combined data (which presents a high number of features), because it is able to extract multiple rule-based models.

For the validation of CAMUR we selected ten BRCA rules extracted by CAMUR and used them on the Gene Expression Omnibus datasets, noting that with these rules 80% of the samples are correctly classified, confirming the validity of the extracted models.

Genes extracted by CAMUR

In this subsection we analyze the large quantity of classification models extracted by CAMUR, focusing on the sets of genes that occur in the rules. We summarize the common genes that appear in the different tumors and experiments with six Venn diagrams, which report the intersections of the genes among the considered tumors and the intersections among each considered experiment.

In Figure 4.3 we show the intersections between the tumors sets of genes for the DNA methylation experiment, the RNA sequencing experiment and their combination.
The classification models obtained for the DNA methylation experiment of Breast Cancer (BRCA) and of Thyroid Carcinoma (THCA) result in 324 common genes, while the intersections with the sets of genes for the Kidney Renal Carcinoma (KIRP) result in fewer genes (5 for BRCA and KIRP intersection and 6 for THCA and KIRP intersection). Furthermore, 20 genes are in common between all tumors. For RNA sequencing we have 150 genes in common for the BRCA and the THCA tumors. Only 2 genes are extracted for the KIRP tumor that are not in common with the other tumors. In the combined Venn diagram 19 genes are in common between all the tumors, 146 between BRCA and THCA, and less than 10 genes for the intersection of THCA and BRCA with the KIRP set. In Figure 4.4 we consider a Venn diagram for each tumor, where the set on the left reports the number of genes calculated from the union of genes of RNA sequencing and DNA methylation experiments, and the set on the right stands for the number of genes extracted from the combined matrix. In each tumor different and common genes are extracted from the combined matrices. For example, in BRCA 733 genes are in common, and 312 are the new genes obtained thanks to the combination of RNA sequencing and DNA methylation. We report also 1730 genes that do not appear in the combined dataset of genes. For the THCA tumor 861 gene are in common, 589 belong to the combined dataset of genes, and 1752 genes are extracted from DNA methylation and RNA sequencing matrices. Finally, the KIRP diagram reports 53 new genes for the combined matrix, 35 in common, and 20 present in the union of the two experiments. In this case, the combination produced more genes than those extracted from the single experiments. The combination leads to the extraction of new genes, which are not computed when analyzing single experiments.

The goal of this work is also to study the different types of cancer and identify many genes related to the disease. By performing the intersection it is

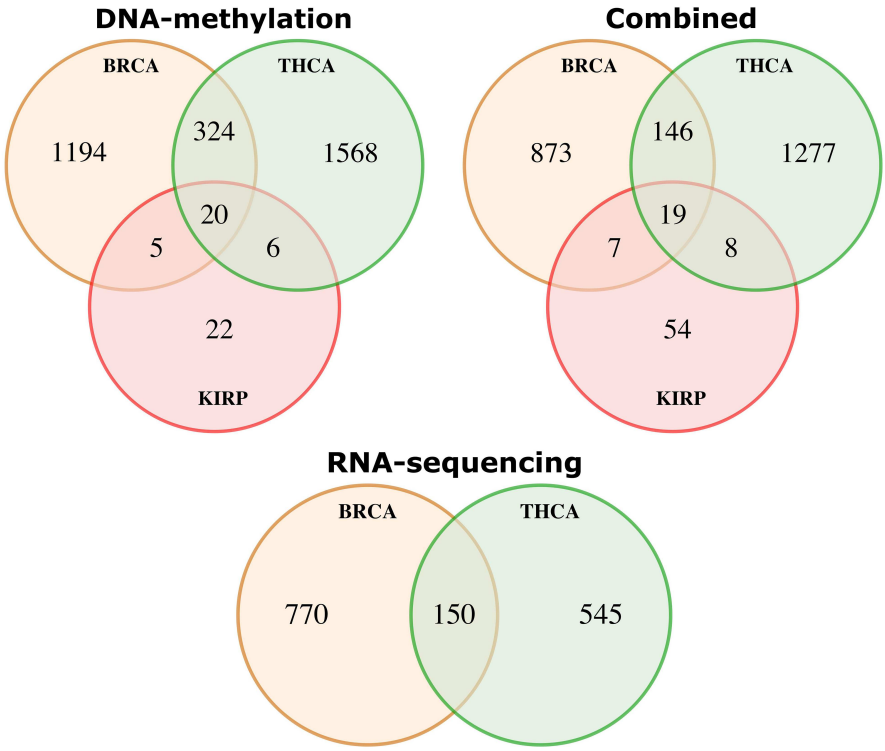

Figure 4.3: Venn diagrams representing the number of genes and their intersection that appear in the DNA methylation experiments, RNA sequencing experiments and in the combined experiments. In the DNA methylation Venn diagram, THCA, BRCA and KIRP sets have 20 common genes, 6 genes are in common between THCA and KIRP, 5 genes between BRCA and KIRP, and 324 between THCA and BRCA. In the RNA sequencing Venn diagram, BRCA and THCA have 150 genes in common, whereas the intersections with the KIRP set of genes are empty, therefore they are not represented.In the combined Venn diagram THCA, BRCA and KIRP sets have 19 common genes, 8 genes are in common between THCA and KIRP, 7 genes between BRCA and KIRP, and 146 between THCA and BRCA.

possible to reduce the number of them and to focus on a number of potential oncogenes. At http://bioinf.iasi.cnr.it/genint we provide all the gene lists that are in common and not in common for each tumor and experiment. Thanks

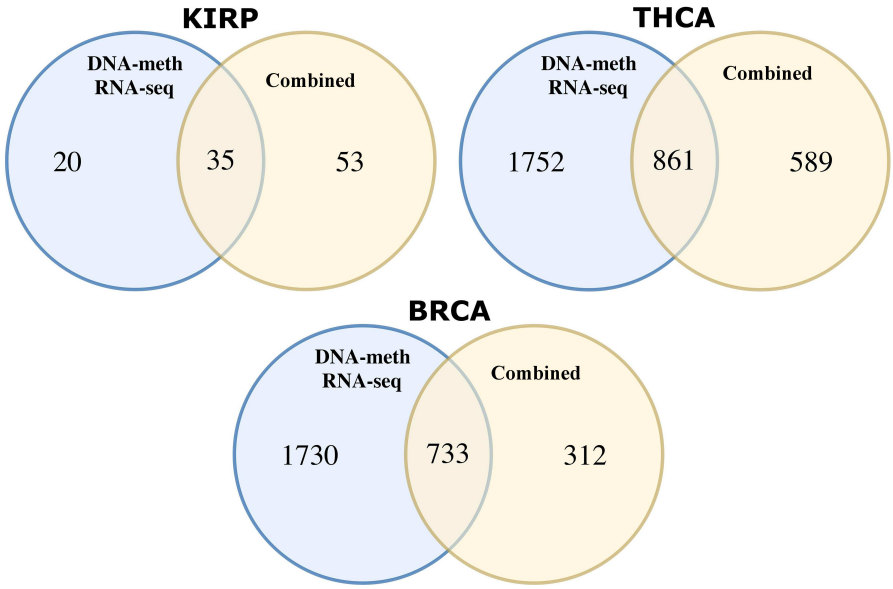

Figure 4.4: The Venn diagrams representing the number of genes and their intersection in each tumor. In KIRP 35 genes are in common between the set of genes extracted from the combined matrix and the union of the genes sets extracted from the DNA methylation and RNA sequencing matrices. We obtain 861 common genes for THCA and 733 for BRCA matrices.

to the intersections, we are able to detect 509 genes in common among DNA methylation and RNA sequencing experiments. From these genes we extract a subset of 13 genes, which are in common among the different tumors. In order to check biological relevance of the obtained subset of genes, we compared our result with the Entrez Gene database of NCBI [MOPT10], which provides information about oncogenes, tumor suppressor genes, the over-expression or lower-expression of gene regulation of cancer cells growth, and also hyper-methylation and hypo-methylation closely associated with the progression of cancer. This analysis led to the detection of 5 and 279 cancer related genes, from the subsets of 13 and 509 genes discussed above, respectively.

4.5 Gene-Oriented Approach for big DNA Methylation Data

Part of this section was published in [WCC+18]

Another knowledge extraction approach is applied on NGS data, to which an efficient data processing procedure is applied that permits to obtain a gene-oriented organization. This approach is described in the following subsection.

Data extraction and preparation

DNA methylation data of TCGA can be organized as follows. We collect n samples each one with its m features and their class labels (conditions), e.g., normal and tumoral. We represent every sample i by the vector $bv_i = (bv_{i1}, bv_{i2}, \cdots, bv_{im}, bv_{ic})$, where $bv_{ij} \in \mathbb{R}$, $i = 1, \cdots, n$, $j = 1, \cdots, m$ and $bv_{ic} \in \{normal, tumoral\}$.
We build the data matrix with the vectors $bv_1, f_2, \cdots, bv_n$, where the rows represent the samples and the columns the features. In DNA methylation experiments the features are the methylated sites and their values represent the percentages of methylated cytosines in a CpG island. This percentage is called *beta value* (bv). The DNA methylation matrix is represented in Table 4.21. A DNA methylation experiment extracts more than 450 thousand sites

Sample	Site_1	Site_2	$\cdots$	Site_m	Class
Sa_1	bv_{11}	bv_{12}	$\cdots$	bv_{1m}	normal
Sa_2	bv_{21}	bv_{22}	$\cdots$	bv_{2m}	tumoral
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
Sa_n	bv_{n1}	bv_{n2}	$\cdots$	bv_{nm}	normal

Table 4.21: DNA-methylation data matrix.

on hundreds of samples. Therefore the DNA methylation matrix is composed of a large number of features ($>$ 450 thousand) and is not easily tractable with state of the art data analysis methods, which are not able to handle such big datasets. We propose to analyze these large data matrices by dividing them into S sub-matrices, with S representing the number of genes, where the methylated sites can be mapped. Indeed, each methylated site can be assigned to a specific gene region, where the site is located. The processing procedure is composed of following steps:

1. find the number S of distinct genes where the methylated sites can be mapped;
2. order the methylated sites according to their associated gene symbol;
3. extract the S sub-matrices with n samples and h features ($h <<< n$), whose site are located within the same gene region, i.e. with the same gene symbol.

So we obtain S sub-matrices, one for each gene, with the same format of Table 4.21, but with only h features (with $2 \leq h \leq 20$). We propose to analyze each sub-matrix with supervised machine learning methods [WFB13, PWD$^+$14, WVFB13, WFB12, CFF$^+$16], This procedure permits to reduce substantially the number of features in each matrix and to perform a gene-oriented analysis, because each matrix is associated to a gene. We report in section 4.5 the list of the genes, whose matrices obtain the best performances when analyzed with classification algorithms.

Performed test

In this work, we adopt Support Vector Machines (SVM) [CST00], C4.5, RIPPER, and CAMUR, in order to evaluate the best performing method and to identify the genes whose DNA methylation data has the most discriminating power.

The application of the classification algorithms on the S sub-matrices (each one associated to a gene) allows us to select for each tumor the best performing genes according to the obtained accuracy.

A synthetic representation of the results is shown in Table 4.22, where we report for each tumor (i) the number of genes that perform with an accuracy $\geq 90\%$ for all the three considered classifiers, and the number of analyzed experiments (one half tumoral samples, one half on normal samples). It is worth noting that the total number of considered genes per tumor is always in the range of ~ 25 thousand and the number of considered methylated sites is in the range of > 450 thousand, because of the used Illumina DNA methylation technique.

For BRCA the number of extracted genes is still high, see 4.5 for details regarding the number of genes extracted by each algorithm.
This means that the beta value is a good metric to establish the discriminant power of a gene for this particular tumor. Therefore, we further investigate the

Cancer	Samples	Selected Genes
BRCA	192	2392
PRAD	100	103
THCA	112	42

Table 4.22: Number of samples and of genes obtained from the classification procedures for each tumor.

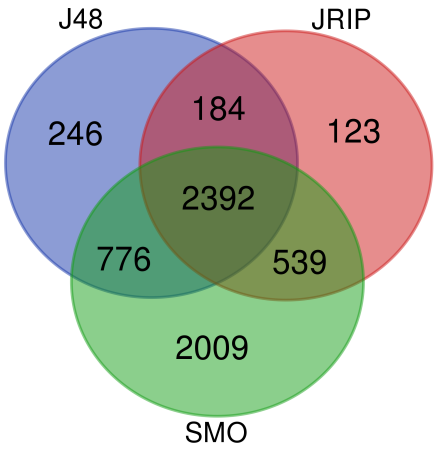

Figure 4.5: Number of genes in BRCA where the classifiers (SMO, J48, and Jrip) reach more than 90% of accuracy. All classifiers obtain more than 90% of accuracy in 2392 genes.

Breast Invasive Carcinoma (BRCA).

In particular, for further reducing the selected genes we apply CAMUR on gene expression data of the same samples and extract a list of logic formulas highlighting the genes that appear in the rules. We subsequently intersect this set of genes with the previously generated set obtained from the other classifiers (SVM, C4.5, and RIPPER) on DNA methylation data generating a new set of 72 common genes. Finally, we highlight a list of logic formulas (shown in Table 4.23) with a classification rate $\geq 90\%$ that contain the genes in this new set.

$(SDPR\|8436 \geq 10.53)$ *and* $(SLC44A4\|80736 \leq 47.42)$
$(SDPR\|8436 \geq 10.53)$ *and* $(SLC44A4\|80736 \leq 51.96)$
$(SDPR\|8436 \geq 11.64)$ *and* $(SLC44A4\|80736 \leq 47.69)$
$(TMEM220\|388335 \geq 2.51)$ *and* $(ITIH5\|80760 \geq 9.74)$
or $(MYOM2\|9172 \geq 29.70)$
$(TMEM220\|388335 \geq 2.52)$ *and* $(LIMS2\|55679 \geq 10.72)$
$(TMEM220\|388335 \geq 2.52)$ *and* $(LIMS2\|55679 \geq 10.92)$
$(TMEM220\|388335 \geq 2.62)$ *and* $(LIMS2\|55679 \geq 10.72)$
or $(SDPR\|8436 \geq 17.55)$ *and* $(A2BP1\|54715 \geq 0.019)$
$(TMEM220\|388335 \geq 2.62)$ *and* $(LIMS2\|55679 \geq 10.72)$
or $(TRIM59\|286827 \leq 1.23)$ *and* $(A2BP1\|54715 \geq 0.019)$

Table 4.23: Examples of classification rules extracted by CAMUR for BRCA with a classification rate $\geq$ 90%, each rule is able to distinguish tumoral from normal samples.

Gene enrichment analysis

In order to validate the classification results, we perform an enrichment analysis on the previous selected genes. We exploit the NCBI [MOPT10] Entrez Gene database to find a relationship between the extracted genes and the analyzed tumors. In particular, we highlight all the genes that could potentially cause the development of a neoplastic phenotype in the cell, as shown in Table 4.24.

In the case of the BRCA, 23 genes are known in literature to be related to the Breast Invasive Carcinoma, 16 genes in the PRAD gene set are related to the Prostate Adenocarcinoma, and only one gene in the THCA gene set, is related to the Thyroid Cancer. The final result of this enrichment analysis is reported in Table 4.25.

The remaining genes in the three extracted gene sets could be of course considered in future experiments as new targets for those specific diseases. It is worth noting that the highlighted genes are also involved in other diseases (e.g., Obesity, Spinal Muscolar Atrophy, Hypotension, etc).

Tumor Abbreviation	Cancer related genes
BRCA	A2M, ABCB1, ABCC1, ABCG1, ACOT7, ACOX2, ADAM19, ADAM33, ADAMTS16, ADAMTS17, ADCYAP1R1, ADORA2A, AKAP2, ALDH1A2, AQP1, ATXN1, B4GALNT3, CA12, CD300LG, CDCP1, CPA1, CREB3L1, CRYAB, DENND2D, DST, FAM92A1, FGF1, FHL1, HEPACAM, HOXA7, IGFBP6, IL11RA, INHBA, ITIH5, KIF26B, LGI4, LIMS2, LRRC3B, MEG3, MUC1, PPP1R14A, PRKD1, SDPR, SPRY2, SSTR1, TMEM220, TNXB, TRIM59
PRAD	ADORA3, AKAP2, ARHGEF2, CA9, CASC2, CAV2, CCDC8, CCK, CD8B, CDH23, CHST11, CLIC3, CNTN1, COL3A1, COL4A5, COL4A6, CYBA, CYP2A13, DAB1, DOCK2, EFEMP2, FEV, FZD7, GALNT6, GATA3, GGT5, GJA1, HAPLN3, HIF3A, HVCN1, IL1B, IL2RB, INCA1, KCNJ3, KCTD8, LRRC4, LTC4S, MASP1, MCAM, MIR1258, MIR130B, MIR301B, MIR575, NBR1, NISCH, PDZD2, PFKP, PRR5, PYCARD, RASL10B, RBM38, SALL2, SEPT4, SIX2, SLC2A5, SLC6A2, SND1, TLX1, TMEM106A, TOM1L2, UBE4B, ZNF154, ZNF385B, ZNF577
THCA	AFAP1, BMPR1B, CAMP, CD96, CDH23, CHRNB4, CMIP, COX5B, DDAH2, ELOVL5, FCGR3B, IL23R, ITIH2, KIFC3, KLHDC8A, LOH12CR1, MAP3K6, MGAT5, NCOR2, PLA2G3, RARA, SCTR, STRA6, TCL1B, TMEM127, TNFRSF12A, ZBTB20

Table 4.24: Subset of extracted genes, which are generally related to cancer, for each investigated tumor dataset.

Tumor Abbreviation	Cancer related genes
BRCA	A2M, ABCB1, ABCC1, ACOT7, ACOX2, ADAMTS16, ADAMTS17, ADORA2A, AQP1, CA12, CDCP1, CREB3L1, CRYAB, FGF1, IL11RA, INHBA, ITIH5, KIF26B, LRRC3B, MEG3, MUC1, PRKD1, SDPR
PRAD	ADORA3, AKAP2, CA9, CNTN1, COL4A6, DOCK2, FZD7, GATA3, GJA1, IL1B, MCAM, MIR130B, MIR301B, PDZD2, PYCARD, SND1
THCA	NCOR2

Table 4.25: Subset of extracted genes, which are specifically related to the cancer under study, for each investigated tumor dataset.

4.6 CamurWeb

Part of this section was published in [WDLC+18]

Implementation

This section introduces CamurWeb, the application designed and developed in this work. CamurWeb is a web service that aims to make the CAMUR software easily accessible and usable. CAMUR was developed in 2015 for the analysis and classification of genomic data, in particular to classify RNA-seq experiments and to extract an interesting body of rule-based classification models. For the software and its algorithm the reader may refer to Section 4.3. CAMUR has two main innovative aspects with respect to many machine learning algorithms: i) it derives many possible classification models and ii) it stores them to allow further and deeper analyses.

CamurWeb is designed to support these two aspects, making easy to exploit these two powerful functionalities even for a non specialized user. Before the release of CamurWeb, in order to run CAMUR the following tasks had to be performed by the user:

- install and configure a valid Java Virtual Machine [java];
- install and configure a MySQL database management system [mys];

- download the CAMUR software package composed of the Multiple Solutions Extractor (MSE) and the Multiple Solutions Analyzer (MSA);
- start the MSE via the command line with its parameters;
- wait for the execution to complete;
- start MSA via the command line, and save the results of CAMUR by querying the interface.

These steps require time and effort and a good knowledge of computer usage. Conversely, CamurWeb allows using CAMUR in a fast and an intuitive way with a simple interface, directly through the browser without the need to install software or dealing with configurations. In the next paragraphs, we will describe the application requirements, and then deepen the architecture and its development.

CamurWeb portal. The CamurWeb portal supports three main tasks:

- it permits to freely access, query, and visualize the large knowledge base of classification results (datasets, logic formulas, performance, and statistics) obtained running CAMUR on all public available RNA sequencing datasets of TCGA extracted from GDC;
- it enables the users to run the software online and to view the results of their classification analyses;
- it allows to download the CAMUR software package.

Therefore, CamurWeb home page is composed of three main sections, as depicted in Figure 4.6: in the first one the users can perform the classification analyses, in the second one they can view the public analyses performed on the cancer datasets extracted from TCGA, and in the third one they can download the CAMUR software package. The main users of CamurWeb can be of two types: the unregistered user, who can mainly access to the public results and repository about cancer; the registered one, who can run the classification software, save the performed analyses, and view her private as well as the public results.
In particular, the unregistered user can (i) learn and deepen the CAMUR classification tool: a section of the website is dedicated to briefly present the software and the web platform, and redirects the user to the bibliographic and

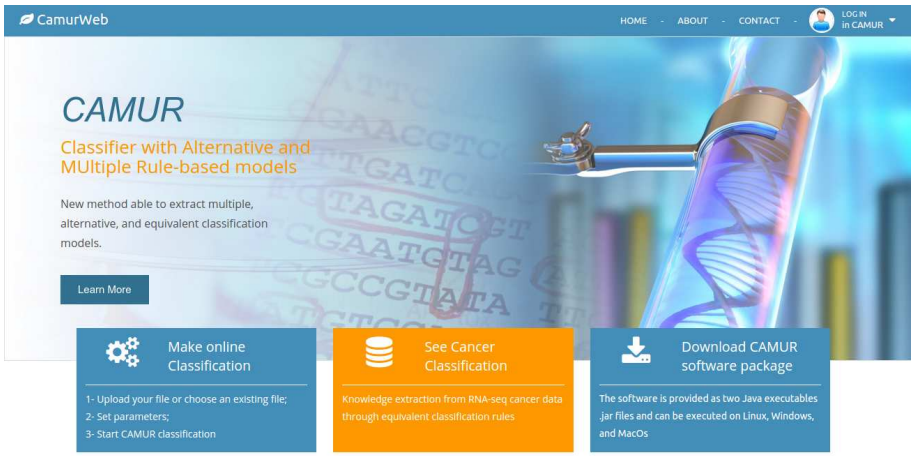

Figure 4.6: The homepage of CamurWeb.

web resources that deepen CAMUR; (ii) view the results of the classification analyses performed on 21 cancer datasets extracted from the Genomic Data Common (GDC) portal (detailed results of these analyses will be presented in section Results and Discussion); (iii) ask for additional information or custom solutions through a simple form; (iv) sign up to the system simply by specifying an email and a password.
The registered user can perform all the previous operations and additionally has the possibility to: (i) perform a classification analysis with CAMUR by using a wizard, which allows to upload a dataset or choose from a set of existing ones containing data extracted from the GDC portal, set the parameters, and run the classification; (ii) view the classification results, i.e., the rule-based classification formulas, charts, and tables; then the user can query the database to see the results and export them; (iii) see a personal section with a report of the analyses started on the system and with her profile information. In order to run a classification task on a private dataset (see Figure 4.7) the user must be registered. The system alerts the user with an e-mail at the end of the execution. This is another strength of CamurWeb, because processing a dataset with CAMUR can take few minutes to hours; so the user does not have to wait for the end of the execution connected to the system or with her computer turned on. The input file format of the CamurWeb classification online procedure is a standard comma separated values (csv) text file containing

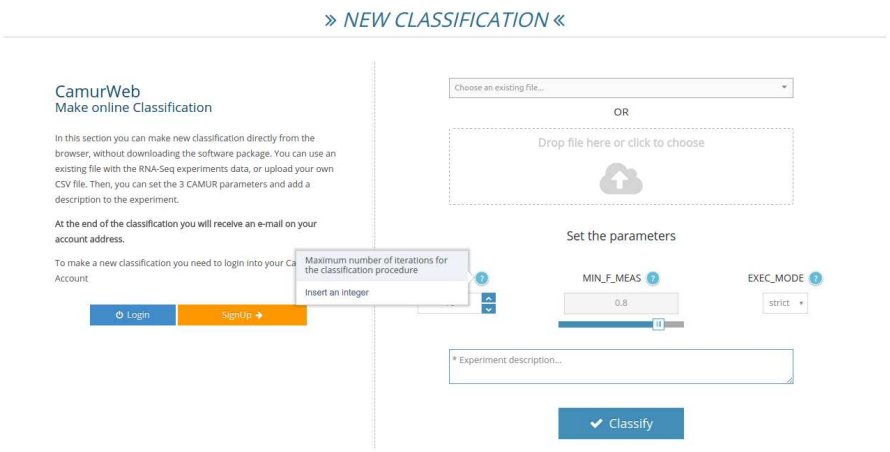

Figure 4.7: The classification section of CamurWeb.

the data matrix of the RNA-seq experiments. For further details about the input format, we point the reader to the user guide of CAMUR available at http://dmb.iasi.cnr.it/camur.php. The results of the access to the knowledge base, either the public or the private ones produced by running CAMUR, are reported on a *results page* (see Figure 4.8). In this page CamurWeb shows: (i) a table with information about the uploaded file and the experiment, in particular the number of rows, which corresponds to the number of samples; the number of columns, which corresponds to the number of features; the size of the file; the time it took for the classification; the number of iterations chosen for the classification, and the number of extracted rules; (ii) a pie chart with the classes in the dataset with the percentage and number of samples; (iii) another table with the list of features extracted by the classifier and their number of occurrences; if the features are genes contained in the Ensembl database [FAB+11], the link leads to the page at www.ensembl.org with a description of the genes. Additionally, in the same page the user can perform the following knowledge extraction queries:

1. Features List: extracts the list of genes and their occurrences in all the classification models obtained in the considered analysis;

2. Literals and conjunctions list: identifies the conjunctions of the literals present in the rules; for each one the number of correct and incorrect

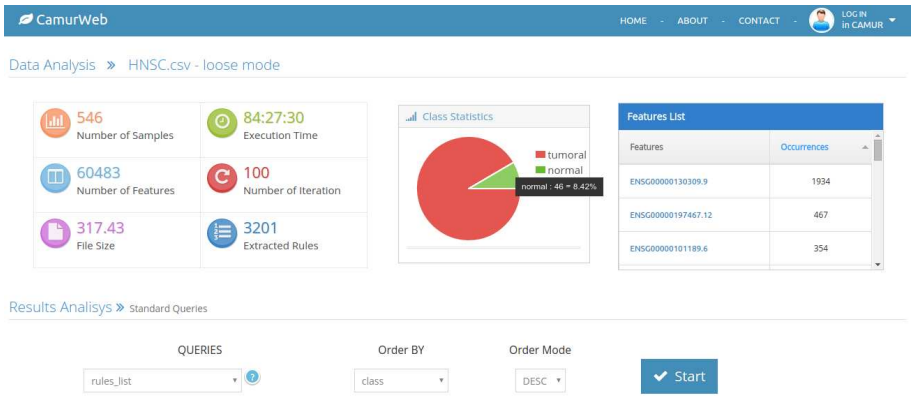

Figure 4.8: The results page of CamurWeb.

instances and their percentages are returned;

3. Rules list: extracts the literal disjunctions with their precision and accuracy;

4. Literals statistics: returns more detailed statistics on the extracted genes and their thresholds;

5. Feature pairs: extracts the pairs of genes present in the same rule and counts how many times they appear together.

The results of such queries can be visualized or downloaded.

Tools and technologies

This section briefly presents the technologies and tools used for the CamurWeb application development.

CamurWeb is written in the Javascript programming language [javb], which is suited not only for client-side applications but also for server-side ones. The Node.js framework [nod] is adopted in this project. Node.js is a platform created on the Javascript engine, which allows to create fast and scalable web applications. The main features of Node.js are (i) an orientation towards the

development of asynchronous code; (ii) a modular architecture; (iii) an optimized transmission of information through the HTTP connection. In addition to the APIs provided by Node.js, we use the Express.js library [exp], a Node.js based framework that offers a robust set of functionalities to easily build single-page, multi-page, and hybrid web applications. It is a mature framework that offers several features including middleware, routing, the ability to manage application configurations in an easy way, and a template engine. Moreover, JQuery [jqu], a JavaScript library for web applications, is adopted in CamurWeb. It is born with the goal of simplifying selection, manipulation, event management, and animation in HTML pages. The jQuery library allows us to simplify JavaScript by writing complex instructions in one line. Additionally, the Bootstrap JavaScript library [boo] is used for the development of the web interface. For managing the different executions of CAMUR, we adopt the REmote DIctionary Server (REDIs) [red], which is one of the most popular key-value databases. In CamurWeb, REDIs is used in Node.js for supporting the development of execution queues. It is used to handle a queue for CAMUR executions requested by the users. The maximum number of parallel executions of CAMUR is set in the application configuration file: a job being in the queue only starts if the number of active runs of CAMUR is less than the maximum number, otherwise the job is entered in the queue. Finally, CamurWeb uses MySql [mys] as database management system in order to store the users identification data and the results of their analyses. In particular, the structure designed and used by CAMUR has been extended with new tables for the purposes. The MySql library is integrated in Node.js.

Software architecture

CamurWeb follows the standard client-server model, i.e., the reference architecture for web applications [Han00]. In particular, CamurWeb uses is the *Model-View-Controller* (MVC) architectural pattern that allows to decouple the different components of the application to gain benefits in terms of reusability and maintenance [BHS07]: *Model* contains data access methods; *View* takes care of displaying data to the user and manages the interaction between the user and the underlying infrastructure;

Controller receives user commands across *View* and reacts by performing operations that may affect the *Model* and which generally lead to a View state change. The software architecture of CamurWeb is shown in Figure 4.9 and described in the following. The software is composed of four main components and six other stand alone software modules.

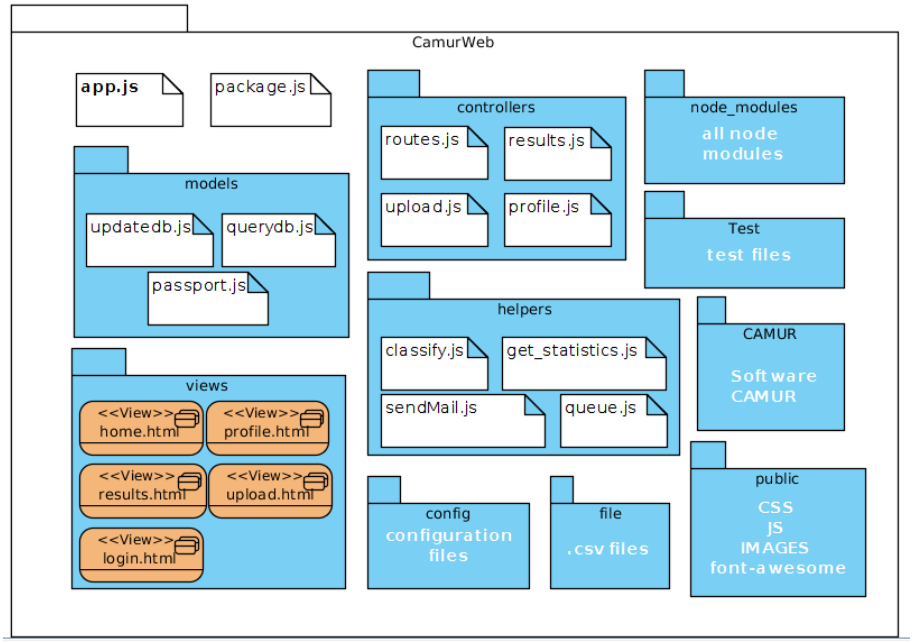

Figure 4.9: The software architecture of CamurWeb.

The *Controllers* component contains the routes of the application. Routes play a primary role: their job is to translate the different request urls by addressing the call to the correct function on the server.

The *Views* component contains the software modules that constitute the web application interface described more in detail in subsection CamurWeb portal.

The *Models* component contains the software modules that interact with the database. All operations that need to retrieve data from the database, insert, or update it, are handled by these modules.

The *Helpers* component contains support software modules for the web application, e.g., the statistics functions, the send email facility, and the CAMUR executor. Finally, six additional stand alone modules are part of the software architecture: the node modules, which group the system libraries of Node.js; the config module, which contains the configuration files of the software; the CAMUR module, which contains the CAMUR software package; the public

module, which contains useful files for the GUI; the file module, which manages the storage of the users' file and of the public datasets; and lastly the test module, which manages the public analyses and the private ones performed by the different users.

Analyzed data

In order to prove the validity of CamurWeb, we performed a classification analysis on all public available RNA sequencing datasets of The Cancer Genome Atlas database extracted from the Genomic Data Commons portal. For each dataset we obtain a large body of accurate classification models, which are composed of rule-based classification formulas containing many genes and their association to a particular cancer type. With these models we build a large knowledge base about cancer focusing on the extracted genes. Interested researchers and medical doctors can access these knowledge on our public section "See cancer classification" available at http://bioinformatics.iasi.cnr.it/camurweb. In the following, we describe the analyzed data and some of the obtained results.

The data selected for the analyses are extracted from the Genomic Data Commons (GDC) portal through its APIs [gdc] (scripts to download ad process data are available upon request).

In this study we focus on data of RNA-seq, which provides a comprehensive view of the transcripts of a cell, can identify new transcripts, is able to monitor splicing events, and permits to quantify gene expression. For this reason RNA-seq is considered a valid tool for a deep understanding of tumor processes. Therefore we select from the GDC portal all publicly available RNA-seq TCGA data, which are composed of gene expression measures on 9030 diseased and healthy tissues (92.6% and 7.4%, respectively). These data are obtained by adopting the Illumina HiSeq 2000 RNA Sequencing Version 2 (RNA-seq V2) platform [ill] and are collected in GDC by the Cancer Genomic Characterization Center (CGCC) University of North Carolina. The public available tumors are 30, each one consisting of a set of samples taken from healthy tissues or diseased ones: healthy tissues are labeled in GDC with the term "normal" and diseased ones with the term "tumoral". For each tissue GDC provides 60,483 gene expression values expressed with the *Fragments Per Kilobase per Million mapped* (FPKM) measure..

In order to be classified, the downloaded data are processed and transformed into a matrix format. We build a matrix for each tumor containing the FPKM

gene expression values: the rows correspond to the samples, which range from 45 for the CHOL tumor to 1222 for BRCA; the first column represents the sample identifier; the central columns correspond to the 60,483 genes, whose expression is measured and which are identified by their Ensembl ID [FAB+11]; the last column represents the class of the sample (normal or tumoral); the element c_{ij} contains the FPKM value of the sample i measured on the gene j. An example of data matrix is shown in Table 4.26. Scripts for the conversion and assembly of the GDC data to a matrix format are available upon request. The input of CamurWeb is am RNA-seq matrix encoded in a comma separated values (csv) text file. In Table 4.27 we show the main characteristics of the

Aliquot	**ENSG00000 130309.9**	**ENSG00000 101189.6**	**...**	**ENSG00000 260597.1**	**Class**
TCGA-4G..	0	9,7872338	....	0,141	tumoral
TCGA-W5..	0,0323	1,4725	...	0,62107	normal
.....	.....	.....	.....	.....	....
TCGA-ZH..	0,06223	8,7757	.....	0,4818	tumoral

Table 4.26: An example of RNA-seq data matrix. Rows are indexed by the tissues, columns by the genes (except the last one containing the class). Each element of the matrix represents the FPKM gene expression value associated to the respective gene and tissue.

obtained matrices. As the reader can see, RNA-seq experiments of cancers ACC, DLBC, LAML, LGG, MESO, OV, TGCT, UCS, and UVM only include samples of tumoral tissues. Therefore it is not possible to perform a supervised classification analysis of such cancer datasets.

Classification analyses and creation of the knowledge base

We performed the classification analyses through the CamurWeb platform on all datasets containing normal and tumoral tissues. The parameters of CAMUR have been set as follows: the execution mode to *loose*, the maximum number of iterations to 100 and the minimum F-measure value to 0.8. The execution mode indicates how CAMUR runs, the loose mode is slower than the strict one, because computational complexity grows exponential to the number of features. On the other hand the loose mode permits to extract more knowledge with greater accuracy (F-measure). The maximum number of desired iterations of CAMUR is set to 100; this means that CAMUR is going to perform 100 runs each one with several classification procedures. The minimum

Cancer	# of tissues	# of tumoral	# of normal	% of tumoral	File size (MB)
ACC	79	79	0	100	45,08
BLCA	433	414	19	95,61	250,69
BRCA	1222	1102	120	90,18	592,77
CESC	309	304	5	98,38	180,67
CHOL	45	36	9	80,00	26,49
COAD	521	478	43	91,75	293,15
DLBC	48	48	0	100	28,62
ESCA	173	161	12	93,06	117,00
GBM	174	156	18	89,66	107,08
HNSC	546	500	46	91,58	317,43
KICH	89	65	24	73,03	52,83
KIRC	611	538	73	88,05	372,75
KIRP	321	288	33	89,72	187,99
LAML	173	173	0	100	98,28
LGG	534	534	0	100	319,55
LIHC	424	371	53	87,50	233,13
LUAD	594	533	61	89,73	353,07
LUSC	551	502	49	91,11	333,09
MESO	86	86	0	100	50,96
OV	309	309	0	100	238,69
PAAD	182	177	5	97,25	108,34
PCPG	186	178	8	95,70	107,82
READ	177	166	11	93,79	100,34
SARC	265	259	6	97,74	152,34
STAD	407	375	32	92,14	268,86
TGCT	156	156	0	100	95,25
THYM	121	119	2	98,35	72,01
UCEC	587	551	36	93,87	336,61
UCS	56	56	0	100	34,28
UVM	80	80	0	100	43,96

Table 4.27: The considered data of The Cancer Genome Atlas extracted from the Genomic Data Commons portal. The number of tissues, the ratio of tumoral and normal ones, and the file size in MB is reported for each cancer dataset.

F-measure is the value below which the classification results are not considered. CAMUR will stop after the maximum number of iterations has been reached or if the F-measure of all current runs is below the given threshold. For further details about the parameters setting, the reader may refer to [CFF+16]. The classification analyses have been performed on an Intel i7 workstation with 24 GB of RAM and by using the CentOs 7 64bit linux operating system with kernel 3.10.0-514.26.2.el7.x86_64. We executed 3 analyses concurrently. A total of 21 analysis tasks have been accomplished, resulting in more than 10,000 classification procedures.

Table 4.28 shows the results in terms of running time, number of inferred rules, and number of extracted genes (features). By comparing the results reported in Table 4.28 with the characteristics of the datasets shown in Table 4.27, we can draw some considerations regarding the link between the number of samples of the dataset and the execution time. The running time of CAMUR is not directly proportional to the number of samples (the number of rows) of the considered dataset. The number of samples actually affects only execution time of a single iteration of the CAMUR classifier; what determines the total time of the execution is the number of iterations. CAMUR continues its iterations since one of the stopping criteria is verified: (i) the maximum number of iterations imposed by the user is reached; (ii) the F-measure values are smaller than the threshold set by the user; (iii) all possible combinations are eliminated from the set of features.

The fastest analyses, where not all 100 iterations are executed, are CESC, CHOL, KICH, KIRP, LUSC, READ, and THYM. In fact, in these analyses a small number of rules are extracted and consequently a small set of relevant genes is obtained. The cause can be a combination of the stopping criteria (ii) and (iii): it is possible that the rules extracted after the first iterations do not exceed the minimum value of F-Measure (0.8), and hence all their genes are not considered. The consequence is that the set of genes does not increase and the combinations to be eliminated from the original dataset quickly becomes empty.

It is worth to note that for the BLCA, BRCA, GBM, HNSC, KIRK, LIHC, LUAD, PCPG, SARC, STAD, UCEC tumors CAMUR extracted a high number of rules and many features (genes) that are potentially involved in the tumoral processes. For the other tumors CAMUR extracted a smaller set of genes that are related to the cancer under study.

As an example Figure 4.10 shows the results page of the classification analysis on the LUSC tumor. The reader can see that among the extracted features the ADGRF5 gene with Ensembl ID ENSG00000069122.17 is the one that occurs most in the classification rules. Previous studies have already shown that mutations within this gene are possible causes of lung cancer (LUSC) [LSB16]. Similarly, many other genes extracted from the classification rules of LUSC are listed in several publications that concern this tumor [UFH+15]. The CHOL and KICH tumors are characterized by a small set of tissues (45 and 89) though with a percentage of normal ones greater than others. The classification analyses on these two tumors did not produce many rules, but for all the extracted ones the F-Measure and the accuracy was 1, i.e., no classification errors occurred.

Cancer	Execution time	# of iterations	# of rules	# of genes
BLCA	4:36:52	100	334	164
BRCA	190:29:57	30	3015	1847
CESC	0:01:50	20	5	3
CHOL	0:00:13	47	3	2
COAD	1:48:12	100	90	32
ESCA	0:56:09	100	229	122
GBM	14:21:12	100	1487	832
HNSC	84:27:30	100	3201	1363
KICH	0:00:52	26	8	5
KIRC	6:36:45	100	470	183
KIRP	0:01:17	9	3	2
LIHC	24:08:10	100	1890	854
LUAD	12:06:36	100	775	298
LUSC	0:06:23	32	8	5
PAAD	0:29:37	100	132	71
PCPG	6:35:40	100	348	173
READ	0:01:11	23	6	5
SARC	7:42:24	100	358	164
STAD	2:04:16	100	416	243
THYM	0:00:19	14	3	3
UCEC	3:52:26	100	496	209

Table 4.28: Results of the classification analyses with CamurWeb. We report for each considered cancer the execution time, the number of performed iterations, the number of extracted rules and genes by CAMUR.

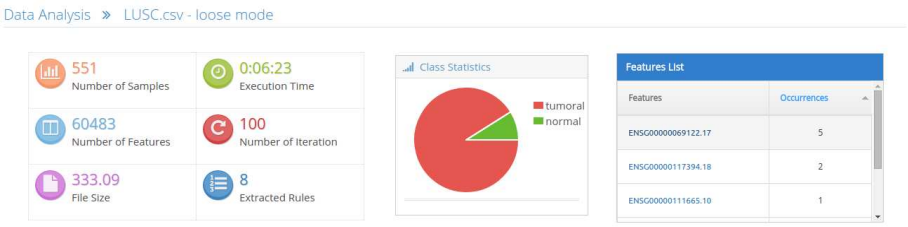

Figure 4.10: The results page of the classification analyses on the LUCS tumor.

Other examples and some considerations are reported in the following.

Head and Neck squamous cell carcinoma (HNSC) HNSC is one of the analyses with higher execution time, because the CAMUR software was able to run 3201 classification procedures producing rules with accuracy values ranging from 0.95 to 1 and extracting 1363 genes. In Table 4.29 we report the genes that are most represented in the rules. We can see that the COLGALT1 gene with Ensembl ID ENSG00000130309.9 is the one that appears in the largest number of rules (1934 rules out of 3201). By examining more deeply the rules, this gene has an FPKM value above 18.16 in all tumoral tissues. Similar observations can be made for the genes COL13A1 (ENSG00000197467.12), MRGBP (ENSG00000101189.6), and following. Such examinations can be at a basis for

Gene	Occurrences
ENSG00000130309.9	1934
ENSG00000197467.12	467
ENSG00000101189.6	354
ENSG00000260597.1	250
ENSG00000197766.6	218
...	...

Table 4.29: Most represented genes in the rules extracted from the HNSC tumor.

targeted research and studies about cancer. Another investigation can be made with CamurWeb by studying pairs of genes that appear often together in the classification rules. This information can be obtained from the CamurWeb database with a simple query called "feature pairs". We report part of the results for the HNSC tumor in Table 4.30. As the reader can see, the genes COLGALT1 (ENSG00000130309.9) and AC012531.25 (ENSG00000260597.1) is the most frequent couple that appears in the rules occurring 250 times. In particular, AC012531.25 is always extracted together with COLGALT1, because its number of occurrences as single gene is exactly 250. Even this investigation generates important results in helping to understand the genetics of cancer.

Liver hepatocellular carcinoma (LIHC) For this tumor CAMUR has identified 854 genes by running 1890 classification procedures. In this dataset the percentage of normal tissues (12.5%) is higher than in other tumors. In Table 4.31 we show the most represented genes that occur in the rules. It is worth noting that the GABRD (ENSG00000187730.7) gene is the most represented one, followed by the TOMM40L (ENSG00000158882.11) gene. Existing studies

Gene 1	Gene 2	occurrences
ENSG00000260597.1	ENSG00000130309.9	250
ENSG00000130309.9	ENSG00000197766.6	203
ENSG00000256229.6	ENSG00000130309.9	167
ENSG00000164114.17	ENSG00000130309.9	165
...	...	...

Table 4.30: Pairs of genes that occur most in the classification rules related to the HNSC tumor.

on the GABRD gene confirm that alterations in its expression can play a key role in differentiating tumor cells. In particular, an abnormal regulation leads to its overexpression that can cause the proliferation of tumor cells [GKI15]. Regarding the second gene, a study has been published that relates the alteration of TOMM40L expression to the excess of smoke in humans [LVRF13]. In this study, the authors relate the effect of smoke and the elevated expression of TOMM40L by concentrating on neurodegenerative diseases such as Alzheimer's and Parkinson's. The findings of CamurWeb can be objective of future studies on this gene (and on other ones) that focus on cancer.

Gene	occurrences
ENSG00000187730.7	413
ENSG00000158882.11	376
ENSG00000231856.2	295
ENSG00000164283.11	229
...	...

Table 4.31: Most represented genes in the rules extracted from the LIHC tumor.

Breast Invasive Carcinoma (BRCA) Analyses on the BRCA dataset are particularly interesting for the large number of available tissues (1,222, 1,102 tumoral, and 120 normal). Breast cancer is the most common tumor in the female population and represents 29% of all tumors affecting women. For this reason it is deeply studied, and we can find in literature a lot of findings about it. CAMUR executed 30 iterations on the BRCA dataset producing 3,015 rules and extracting 1,847 genes with a running time of 190 hours and 29 minutes. In Table 4.32 we report the most frequent genes that are present in the obtained classification rules. We highlight that previous research

confirms the relationship between the alteration of the expression of the first three most occurring genes - SPRY2 (ENSG00000136158.9) [SIH+13], VEGFD (ENSG00000165197.4) [NYT+03], and MMP11 (ENSG00000099953.8) [RCDV+14] - and the predisposition to Breast Cancer.

Gene	occurrences
ENSG00000136158.9	1078
ENSG00000165197.4	993
ENSG00000099953.8	725
ENSG00000157766.14	515
...	...

Table 4.32: Most frequent genes in the rules extracted from the BRCA tumor.

4.7 Conclusions

In this chapter we presented a methodology to efficiently extract a list of relevant genes in NGS data of cancer exploiting consolidated machine learning algorithms. We analyze these data by means of supervised classification algorithms, extracting classification models, which are able to distinguish the samples in two classes (tumoral and normal) and which are composed of features that represent the genes related to the disease and the different NGS experiment. In particular we proposed two methods for data pre-processing, in order to extract different insights.

We proposed a method for the combination of two distinct NGS experiments with different data schemas. The NGS experiments considered in this study were DNA methylation and RNA sequencing and were extracted from TCGA. We focused on three forms of tumors, i.e. BRCA, KIRP, and THCA. We defined the data matrices, one for each NGS experiment, with samples in the rows and genes in the columns, and a third matrix for representing the combination of DNA methylation and RNA sequencing samples. In particular the objective of the combination was the creation of data matrices indexed on the genes that are related to both NGS experiments. In the RNA sequencing matrices the items are the RNA-Seq by Expectation-Maximization (RSEM) values that quantify the gene expression, whereas in DNA methylation matrices we defined a new gene-wide measure based on the beta value, that we call *gene methylation quantity* for denoting the quantity of methylation associates to each gene. After the combination of RNA sequencing and DNA methylation,

we proposed the application of supervised analyses. We were able to extract many classification models containing the genes and their quantification values by applying different supervised algorithms (decision trees, rule-based classifiers, and multiple rule-based ones). The classifications were performed on all matrices for each tumor, with the objective to obtain models to separate the normal from the tumoral samples. All the classification models have an accuracy greater than 95%. In particular we obtained 9 decision trees with C4.5, 9 rule-based classification models with RIPPER, and 9 classification models (each one composed of 30 decision trees) with Random Forest. Moreover, thanks to the execution of more than 2000 classification procedures with CAMUR, we extracted 15.252 classification models, from which we derived 5413 genes related to DNA methylation and RNA sequencing. 509 genes are in common among the different experiments and 13 genes among the different tumors. Through the NCBI Entrez gene database we performed functional analysis of those genes. We found 279 out of 509 and 5 out of 13 of them already marked as oncogenes. CAMUR was applied in order to extract possible oncogenes and to find new ones. Many of the extracted genes have been already classified as oncogenes,and this confirms that our method is able to identify relevant genes and justifies further analyses. Indeed, we suggest as future direction a further biological investigation of the classifications models and the extracted sets of genes to confirm their relation with the considered tumors. As other future work, we suggest the application of our method to other forms of tumors and, we plan to define new gene wide measures on different NGS experiments (e.g., mutations, copy number variations, chip-sequencing) in order to consider the combination of them for a comprehensive knowledge extraction.

We presented a second methodology which consider for each gene the beta values of all its methylated sites aggregated in a matrix. This data representation allowed us to create good performing classification models able to discriminate tumoral and non tumoral samples and to select the best accurate genes. To confirm our methodology, we applied our procedure to three different types of cancers (breast, prostate, and thyroid carcinomas) obtaining promising results. Finally, we executed an enrichment analysis in order to highlight the genes related to the development of a particular tumor as a final validation of our procedure. In this simple approach, each gene is considered independently of the others and some genes that act simultaneously in the tumoral process may be ignored by this approach. Therefore, in future we are going to improve the method for considering many methylated sites of different genes simultaneously as in [CCW18]. As additional future directions, we suggest to extend the

number of applied machine learning algorithms for retrieving a stricter list of relevant genes thanks to the combination of their results. This will allow one to drastically reduce the number of genes to guide a smarter design of future experiments. To conclude, we propose to apply our procedure to many other DNA methylation datasets related to different cancer types in order to extend our analysis, further validate the methodology, and discover novel biological insights in tumor studies.

Finally we proposed CamurWeb, a new web portal for classifying NGS data of RNA sequencing and for sharing the obtained results. CamurWeb is a web application based on NodeJs, ExpressJs, and MySQL, which makes use of the CAMUR classification software. CAMUR is able to compute a large body of knowledge by finding a high number of genes that are likely to be involved in the processes that cause the formation of tumors. Conversely, state of the art rule-based classifiers extract from a dataset a set of two or three rules that describe it. However, this small set of rules may be insufficient to describe the data in a comprehensive way and to extract sufficient knowledge from it.

In order to prove the validity of CamurWeb and to release a large knowledge base of classification rules about cancer, we performed a wide supervised analysis on gene expression data belonging to more than 9,000 patients and 21 different tumor types of The Cancer Genome Atlas extracted from the Genomic Data Commons portal. The obtained results were evaluated in terms of performance, execution times, and extracted features (genes related to a particular type of tumor). Among those genes, we identified a part of them already linked to the literature about cancer, confirming our classification procedure, and another part that still has to be investigated; this could be the starting point for new research studies. The identified genes can act as possible diagnostic and prognostic markers or therapeutic targets. All the extracted knowledge, the classification results, and the selected genes have been made public on the CamurWeb platform and can be consulted or queried for further investigation by biologists, medical doctors, and bioinformaticians in order to prove their association to a particular cancer.

Topics of future work may concern both the extension of the performed analyses and the development of new features for the CamurWeb application. Regarding the analyses, we plan to (i) investigate the role of the extracted genes for the different analyzed tumors and to compare them with existing studies; (ii) perform a set theoretic analysis of the extracted logic formulas in order to find common biomarkers among the studied cancers; (iii) repeat the classification analyses with the same data, but using different parameters, and

then compare the results both in terms of extracted features, execution time, and accuracy of the rules; (vi) perform other classification analyses with new data extracted from other gene expression databases (e.g., GEO [EDL02]) or projects (e.g., TARGET); (v) increase the number of public analyses, using other input or other classification parameters.

Regarding the CamurWeb platform we plan to: (i) design and develop automatic procedures able to integrate, compare, and analyze the logic classification formulas stored in the database; (ii) add a feature that allows users to share their own analyses; (iii) expand the user profile page by entering a field that allows the user to add observations or personal considerations about the analyses; (iv) increase the number of queries that can be made on the results database produced by CAMUR.

To conclude, we wish to highlight that the CamurWeb software and the published knowledge base are promising research tools for performing analyses on new released data and for discovering novel insights about cancer.

Conclusions

In this dissertation big biomedical data modeling, accessing and querying for knowledge extraction with machine learning techniques have been investigated. In particular, GDC, a large repository of genomic and clinical data about cancer containing different formats and schemes has been standardized. Therefore a framework able to automatically extract, extend, and convert all public clinical and genomic data of the Genomic Data Commons, with the aim to standardize data of different Next Generation Sequencing experiments (i.e. Somatic Mutations, DNA methylation, Copy Number Variations, Gene-, Isoform-, and miRNA- Expression Quantification) to the Browser Extensible Data (BED) format. The framework is called OpenGDC, an open-source Java software able to standardize public accessible GDC genomic and clinical data allowing researchers to easily perform ad-hoc integrated genomic analyses. The OpenGDC converted data are fully supported by bioinformatics frameworks like the GenoMetric Query Language (GMQL) system that exploits a SQL-like declarative language to make integrative queries on heterogeneous genomic data; a valid example about how our data standardization approach makes integrative analyses easy to be performed by ad-hoc bioinformatics frameworks.

Additionally, in order to make easily accessible these data, a collection of Application Programming Interfaces (APIs) have been provided and integrated in a flexible framework that we called OpenOmics. In this framework a new ontological software layer has been defined. It allows users to interact with experimental data and metadata without knowledge about their representation schema. Domain-specific ontologies have been exploited in order to allow executing taxonomy-based relaxed queries. In particular, the upward and down-

ward query extension methods have been applied to obtain a finer or coarser granularity of the requested information. Different use cases proved that a user can perform a query specifying particular attributes related to metadata or genomic data, even if they are not available in the considered repository. Thus, the requested data have been extracted through the use of domain-specific ontologies of The Open Biological and Biomedical Ontology (OBO) Foundry [SAR+07].

Finally knowledge extraction methods and different methodologies for the integration and manipulation of NGS data of cancer in order to obtain significant insights from them have been proposed. A first method provides the combination of two different information at gene level: the RNA sequencing gene expression values and the gene methylation quantity, a new measure defined for representation tof he methylation quantity associated to a gene. The integrated data have been analyzed through tree- and rule-based classification algorithms (C4.5, Random Forest, RIPPER, and CAMUR) and 5000 classification models (composed of gene sets) from the integrated experiments have been obtained. These models consider both the gene measures related to RNA sequencing and DNA methylation experiments, and allow to distinguish the tumoral samples from the normal ones, with an average accuracy of 95%. Another method, is focused on the NGS experiment of DNA methylation, whose data matrices are composed of hundred thousands of features (i.e., methylated sites). A gene-oriented organization of data has been proposed, in particular this procedure divides the original data matrices into several sub-matrices, each one containing the sites located within the same gene. Tumoral and normal samples have been successfully discriminated using function-, tree-, and rule-based classifiers. From the obtained models the best performing genes have been selected ranking them according to the accuracy of the classifiers. Those genes can be further investigated by domain experts for proving their relation to the cancers under study. In order to facilitate knowledge extraction a classification software (CamurWeb) has been implemented. It provide a user-friendly interface for the execution of the CAMUR algorithm on a chosen genomic data matrix, the resulting data models visualization, and additionally a large knowledge base for gene expression data of cancer from GDC repository is provided.

Future work concerns the definition of a framework that, starting from standardized data in OpenOmics, allows (i) the extraction of datasets of interest by querying the data through the different access endpoints of the Openomics APIs collection, (ii) the creation of integrated data structures derived from the combination of different biological experiments, and finally (iii) the execution

of machine learning algorithms for the interpretation of the integrated data. Another future perspective of the dissertation is the application of the presented methodologies and software tools to new and different datasets in order to foster additional knowledge about biological processes and about important diseases for aiding medical diagnosis.

Appendix A

Bioinformatics, genomics and fundamentals of biological sciences

A.1 Human Genome

The human genome is in part composed of all the DNA contained in the cells' nucleus and folded on itself to form the chromosomes. Each DNA molecule is packaged in a separate chromosome and the total genetic information of an organism is stored in the chromosomes and it constitutes its genome. The DNA, deoxyribonucleic acid, makes up the main part of the genomes, which contain the biologic information needed to build and maintain a living organism. The DNA is made up of a nucleotide chain and every nucleotide is composed of a sugar, a phosphate group and a nitrogenous base (A adenine, T thymine, G guanine, C cytosine). The DNA has a is double helix shaped structure since it is made up of two strand which weave together. Each single strand is the lateral filament of the DNA molecule and it is put together by sugar-phosphate bonds, while the double helix structure is stabilized by hydrogen bond between the nitrogenous base (A, T, G, C) on the two strands, in particular the hydrogen bonds are A-T, G-C. The genome of an organism characterize the species and it includes both the genes, which are the coding part, and the non-coding material. The human genome is made up of these elements:

– nuclear genome, a set of nucleotides grouped in 23 pairs of molecules called chromosomes. In particular 22 autosomes, that are chromosomes which do not contain genetic information related to the individual sexual

characterization, plus two chromosomes that determine sex, the X and Y chromosomes;

– mitochondrial genome, circular double helix shaped molecule. The mitochondrial DNA is inherited only by the mother while the nuclear DNA comes from both parents.

The main goal of the genome is to express the biological information by means of a process called genes expression. The gene expression allows the transformation of these biological information into proteins. Within the cellular genes expression system, three main aspect can be detected and they represent the fundamental direction of the information flow:

– the genetic information is stored in the DNA, which can be duplicated to transmit the information;

– the DNA is transcribed into RNA in order to be expressed in the cell;

– the RNA is translated into proteins, which are the operative and final form of the information contained in the genome.

A.2 Central Dogma

In the gene expression, the central dogma ([Sha01]) is a particular process which determines that the information flow is unidirectional and it starts from the DNA up to the proteins, without considering any inverse path. This means that the protein does not contain any information to produce other proteins. The main steps of the central dogma are:

1. Transcription: the first step of the process takes place in the nucleus and produces the transcriptome, which represents the gene expression in the messenger RNA (mRNA) of a whole organism or of a particular organ, tissue or cell. Thus the transcriptome is the set of mRNAs produced by a specific cell population. For each different cell type, around 10,000 different genes are expressed.

2. Translation: the second phase of the gene expression process takes plane in the cytoplasm where the information contained in the genes is converted into proteins. The final result is then the proteome, that is the set al all the proteins expressed by the genome of a cell or tissue. Hereinafter the main aspects of these phases are reported together with the

description of some functions and the fundamentals characteristics of the protagonists involved in the central dogma: the DNA, the RNA and the proteins, which constitute essential elements of the living organisms.

A.3 DNA

During the transcription process only one strand of the double helix of the DNA is transcripted. The strand which manage the synthesis thanks to the complementary coupling of the bases is called template strand, while the other strand, which has the same sequence of the mRNA with the substitution of the thymine (T) with the uracil (U), is called coding strand. The nucleotides sequencing of RNA is determined by the sequence of the nucleotides in the template DNA, according to the following pairs, which will form a polynucleotide strand complementary to the DNA: C-G, A-U, T-A.

A.4 RNA

RNA, ribonucleic acid, is composed of a nucleotide chain and it is common to find it in nature as a single strand folded on itself. There are different kind of RNA, each of them has a different task within the process of gene expression.

The *messenger RNA* (mRNA), it is the first product of the gene expression process. It codes and transfer the informations from DNA in the nucleus to the cytoplasm, where the protein synthesis takes place and where it undergoes to the translation process. The short life of a mRNA molecule starts with the transcription and ends with the degradation.

The *functional RNA*, transport RNA (tRNA) and ribosomal RNA (rRNA). The tRNA is a molecule with a clover-shaped structure that allows to activate the information translation mechanism, in the form of transcriptome, into proteins.It acts as an adapter between the sequence of mRNA and the amino acid. The tRNA has a bond site for the amino acid and a region with three bases (nucleotides), called anticodon, which recognizes the corresponding three-base codon mRNA through pairing complementary bases. Each specific tRNA anticodon contains a (triplet) sequence which can match one or more codons for the same amino acid. However the genetic code is redundant, and some amino acids are correspondent to more codons of mRNA (hence more anticodons of tRNA) This means that different codons can code for the same amino acid, or that the pairing is accurate for the first two bases while for the third there can be a tolerance to the “wrong” matches In fact, some anticodons can pair with

more than one codon, thanks to the wavering of the bond between the anti-codon and the codon; this phenomenon is called wobble pairing. The rRNA is an essential component of ribosomes and it manages the translation that occurs between mRNA and tRNA in order to create the amino acid chain. The ribosome put itself together on the mRNA and it moves on it until it is all read, recruiting the tRNA in the correct positions to create the different peptide bonds between the amino acids; the ribosome catalyzes the binding reaction between the amino acids and gradually releases the tRNAs until the chain is created and released.

A.5 Protein

The proteins synthesis end with the formation of a polypeptide chain, which is the primary structure of a protein. Proteins are made up of one or more long amino acid chains of 20 different structures and they differ one from another mainly because of their amino acid sequence. The amino acid sequence is determined by the nucleotides sequence stored in the gene and it is commonly translated into a protein folding in a tridimensional structure which determines the protein task.

A.6 Genome sequence

The genome sequence is the sorted succession of the nitrogenous bases included in the genome. In the genome of every organism the four nitrogenous bases (Adenine, Thymine, Cytosine, Guanine) are organized in a a highly precise and sorted scheme and are grouped in different combination to form the genes, the basic units of genetic information. Every gene does not work independently, but it depends on other elements for the replication and expression. Within the genome sequence all the instructions needed to the development and the operation of the organism are included. For this reason the knowledge of the complete sequence is very important. Due to these consideration it is clear the the sequencing of the human genome represent a fundamental issue (more than three billion of nucleotide bases). It has been carried out by two independent groups between the end of 80s and the beginning of 90s in the previous century. However, even if almost 98% of the genomic sequence has been sequenced thanks to the Human Genome Project, the global genome structure is not yet completely clear. The genomic sequence, in fact, is not only made up of coding

sequences, but also of regulatory sequences, repetitive sequences (called junk DNA) and of introns, which often have unknown functions.

A.7 Epigenetics

The word "epigenetics" was introduced by the biologist Conrad Waddington for the first time in 1942 in order to describe the inheritance of a particular characteristic acquired within a population in response to an environmental stimulus. The epigenetics is the study of the changes in gene expression (active genes versus inactive ones). It studies the changes in the phenotype that are not caused by change in the genotype. In fact the individuale genotype is given by his genetic makeup, that is written in the DNA contained in the nucleus of every cell and thus it is unchangeable. The phenotype instead is given by the set of characters that the individual shows: it depends on his genotype, on the genes interaction but also on external causes and thus it may vary. The so called epigenetic changes can be inherited, then can be stable and can be transmitted to the future generations, however the most relevant aspect is given by the dynamisms of these changes that quickly vary in response to the environmental stimulus. Almost every aspect of the cellular life is influenced by epigenetics and this is why it is one of the most important field of modern biology. The epigenetic changes of a cell tells the genes whether to be active or not and their outline is given by the set of all the signals that the cell has received during all its life and these signals act such as a kind of cellular memory. The epigenetic changes record the cell experiences on the DNA, contributing to regulate the genes expression. Generally the variations of the epigenetic state of a cell (epigenome) allow us to adapt ourself to the changing of the world that surround us. However in some case the epigenetic change may have damaging effects on the cells and can cause serious diseases, like cancer.

A.8 Bioinformatics

Bioinformatics is a scientific branch which tries to face up to biological problem related to a molecular level, in particular it elaborates the biological information by means of computer science tools. Some kind of information which are of interests in this field are reported in the following:

– the analysis of -ome data, such as genome, transcriptome, proteome, metabolome, epigenome;

- molecular interaction (systems biology);
- structural biology, which studies protein folding;
- chromatin conformation, molecular docking;
- imaging, which includes cells tracking under a microscope, automatic diagnosis from histological images.

The genomic, which studies the genome of living beings, is the branch that mainly bases its considerations on bioinformatics to elaboration and visualization of the huge amount of data produced. In fact the the birth of bioinformaticsis related to the high amounts of significant data produced by biotechnology. These data were no more analyzable only by hand and some automatic tools were needed. At the beginning of the millennium, within the Human Genome Project ([CMP03]), it was possible to sequence the human genome for the first time and then to determine the sequence of pairs of nitrogenous bases, which make up the DNA and also to identify and map genes of the human genome (the expected data was about one hundred, and about 20-25 thousand have been found). The sequencing of human genome has produced big amount of biological data and the goal of bioinformatics is to manage, classify and analyze them. As a matter of fact such a huge amount of information give rise to many problems like storing, creating complex querying systems and analysis.

Publications

Journal publications

Combining DNA methylation and RNA sequencing data of cancer for supervised knowledge extraction
E. Cappelli, G. Felici, E. Weitschek.
BioData mining 2018 (doi: 10.1186/s13040-018-0184-6)

CamurWeb: a classification software and a large knowledge base for gene expression data of cancer
E. Weitschek, S. Di Lauro, E. Cappelli, P. Bertolazzi, G. Felici.
BMC bioinformatics 2018 (doi: 10.1186/s12859-018-2299-7)

Conference publications

Extending knowledge on genomic data and metadata of cancer by exploiting taxonomy-based relaxed queries on domain-specific ontologies.
E. Cappelli, E. Weitschek, F. Cumbo.
International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics (CIBB). In press

Smart Persistence and Accessibility of Genomic and Clinical Data
E. Cappelli, E. Weitschek, F. Cumbo.
International Conference on Database and Expert Systems Applica-

tions (DEXA). 2019 (Communications in Computer and Information Science, volume 1062), (doi: 10.1007/978-3-030-27684-3_2)

An ontology-based approach to improve data querying and organization of Alzheimer's Disease data
I. Arisi, P. Bertolazzi, E. Cappelli, F. Conte, F. Cumbo, G. Fiscon, M Sonnessa, F. Taglino.
IEEE International Conference on Bioinformatics and Biomedicine (BIBM). 2018 (doi: 10.1109/BIBM.2018.8621524)

Classifying big DNA methylation data: a gene-oriented approach
E. Weitschek, F. Cumbo, E. Cappelli, G. Felici, P. Bertolazzi.
International Conference on Database and Expert Systems Applications (DEXA). 2018 (Communications in Computer and Information Science, volume 903), (doi: 10.1007/978-3-319-99133-7_11)

Extending the Genomic Data Model and the Genometric Query Language with Domain Taxonomies
E. Cappelli, E. Weitschek.
International Conference on Web Engineering (ICWE). 2017 (Lecture Notes in Computer Science, volume 10360), (doi: 10.1007/978-3-319-60131-1_44)

Genomic data integration: A case study on next generation sequencing of cancer
E. Weitschek, F. Cumbo, E. Cappelli, G. Felici.
International Workshop on Database and Expert Systems Applications (DEXA). 2016 (doi: 10.1109/DEXA.2016.025)

Conference abstracts publications

OpenGDC: standardizing, extending, and integrating genomics data of cancer
E. Cappelli, F. Cumbo, A. Bernasconi, M. Masseroli, E. Weitschek.
ESCS 2018: 8th European Student Council Symposium, International Society for Computational Biology (ISCB) 2018

Posters

Combining knowledge-based approach with logic data mining techniques to improve data querying and analysis on Alzheimer's Disease Data
G. Antognoli, I. Arisi, P. Bertolazzi, E. Cappelli, F. Conte, F. Cumbo, G. Fiscon, G. Mavelli, F. Perazzoni, M. Sonnessa, F. Taglino, R. Voyat.
Intelligent Systems for Molecular Biology & European Conference on Computational Biology (ISMB/ECCB). 2019

An ontology-based approach to improve data querying and organization of Alzheimer's Disease data
I. Arisi, P. Bertolazzi, E. Cappelli, F. Conte, F. Cumbo, G. Fiscon, M. Sonnessa, and F. Taglino.
Bioinformatics and Biomedicine (BIBM), IEEE International Conference. 2018

Bibliography

[ABN+99] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc Natl Acad Sci U S A*, 96(12):6745–6750, Jun 1999.

[AGBK+12] Altuna Akalin, Francine E Garrett-Bakelman, Matthias Kormaksson, Jennifer Busuttil, Lu Zhang, Irina Khrebtukova, Thomas A Milne, Yongsheng Huang, Debabrata Biswas, Jay L Hess, C David Allis, Robert G Roeder, Peter J M Valk, Bob Löwenberg, Ruud Delwel, Hugo F Fernandez, Elisabeth Paietta, Martin S Tallman, Gary P Schroth, Christopher E Mason, Ari Melnick, and Maria E Figueroa. Base-pair resolution dna methylation sequencing reveals profoundly divergent epigenetic landscapes in acute myeloid leukemia. *PLoS Genet*, 8(6):e1002781, 2012.

[AKMB+09] Can Alkan, Jeffrey M Kidd, Tomas Marques-Bonet, Gozde Aksay, Francesca Antonacci, Fereydoun Hormozdiari, Jacob O Kitzman, Carl Baker, Maika Malig, Onur Mutlu, et al. Personalized copy number and segmental duplication maps using next-generation sequencing. *Nature genetics*, 41(10):1061, 2009.

[ALK17] Alexander M Aravanis, Mark Lee, and Richard D Klausner. Next-generation sequencing of circulating tumor dna for early cancer detection. *Cell*, 168(4):571–574, 2017.

[Bay05] Stephen B Baylin. Dna methylation and gene silencing in cancer. *Nature Reviews. Clinical Oncology*, 2(S1):S4, 2005.

[BBT+11] Marina Bibikova, Bret Barnes, Chan Tsan, Vincent Ho, Brandy Klotzle, Jennie M Le, David Delano, Lu Zhang, Gary P Schroth, Kevin L Gunderson, et al. High density dna methylation array with single cpg site resolution. *Genomics*, 98(4):288–295, 2011.

[BDF+04] Sally Bamford, Emily Dawson, Simon Forbes, Jody Clements, Roger Pettett, Ahmet Dogan, A Flanagan, Jon Teague, P Andrew Futreal, Michael R Stratton, et al. The cosmic (catalogue of somatic mutations in cancer) database and website. *British journal of cancer*, 91(2):355, 2004.

[Bel14] Riccardo Bellazzi. Big data and biomedical informatics: a challenging opportunity. *Yearbook of medical informatics*, 23(01):08–13, 2014.

[BGT+07] Cherie Blenkiron, Leonard D Goldstein, Natalie P Thorne, Inmaculada Spiteri, Suet-Feung Chin, Mark J Dunning, Nuno L Barbosa-Morais, Andrew E Teschendorff, Andrew R Green, Ian O Ellis, et al. Microrna expression profiling of human breast cancer identifies new markers of tumor subtype. *Genome biology*, 8(10):R214, 2007.

[BHS07] Frank Buschmann, Kelvin Henney, and Douglas Schimdt. *Pattern-oriented Software Architecture: on patterns and pattern language*, volume 5. John Wiley & Sons, Hoboken, New Jersey, USA, 2007.

[Bir85] Adrian P Bird. CpG-rich islands and the function of DNA methylation. *Nature*, 321(6067):209–213, 1985.

[Bir02] Adrian Bird. DNA methylation patterns and epigenetic memory. *Genes & Development*, 16(1):6–21, 2002.

[BKML+08] Dennis A Benson, Ilene Karsch-Mizrachi, David J Lipman, James Ostell, and Eric W Sayers. Genbank. *Nucleic acids research*, 37(suppl_1):D26–D31, 2008.

[BKO+11] Barry Bishop, Atanas Kiryakov, Damyan Ognyanov, Ivan Peikov, Zdravko Tashev, and Ruslan Velkov. Factforge: A fast track to the web of data. *Semantic Web*, 2(2):157–166, 2011.

[BLZ+06] Marina Bibikova, Zhenwu Lin, Lixin Zhou, Eugene Chudin, Eliza Wickham Garcia, Bonnie Wu, Dennis Doucet, Neal J Thomas, Yunhua Wang, Ekkehard Vollmer, Torsten Goldmann, Carola Seifart, Wei Jiang, David L Barker, Mark S Chee, Joanna Floros, and Jian-Bing Fan. High-throughput dna methylation profiling using universal bead arrays. *Genome research*, 16(3):383–393, 2006.

[boo] Bootstrap html, css, and js library. `http://getbootstrap.com`. Accessed 6 June 2018.

[Bre01] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[BWL+12] Tanya Barrett, Stephen E Wilhite, Pierre Ledoux, Carlos Evangelista, Irene F Kim, Maxim Tomashevsky, Kimberly A Marshall, Katherine H Phillippy, Patti M Sherman, Michelle Holko, et al. Ncbi geo: archive for functional genomics data sets—update. *Nucleic acids research*, 41(D1):D991–D995, 2012.

[CBC+17] Stefano Ceri, Anna Bernasconi, Arif Canakoglu, Andrea Gulino, Abdulrahman Kaitoua, Marco Masseroli, Luca Nanni, and Pietro Pinoli. Overview of geco: A project for exploring and integrating signals from the genome. In *International Conference on Data Analytics and Management in Data Intensive Domains*, pages 46–57. Springer, 2017.

[CCB+18] E Cappelli, F Cumbo, A Bernasconi, M Masseroli, and E Weitschek. Opengdc: standardizing, extending, and integrating genomics data of cancer. In *ESCS 2018: 8th European Student Council Symposium, International Society for Computational Biology (ISCB)*, pages 1–1, 2018.

[CCW18] Fabrizio Celli, Fabio Cumbo, and Emanuel Weitschek. Classification of large dna methylation datasets for identifying cancer drivers. *Big data research*, 13:21–28, 2018.

[CF17] Fabio Cumbo and Giovanni Felici. Gdcwebapp: filtering, extracting, and converting genomic and clinical data from the genomic data commons portal. In *Cold Spring Harbor meeting: Genome Informatic*, 2017.

[CFC+17] Fabio Cumbo, Giulia Fiscon, Stefano Ceri, Marco Masseroli, and Emanuel Weitschek. Tcga2bed: extracting, extending, integrating, and querying the cancer genome atlas. *BMC bioinformatics*, 18(1):6, 2017.

[CFF+16] Valerio Cestarelli, Giulia Fiscon, Giovanni Felici, Paola Bertolazzi, and Emanuel Weitschek. Camur: Knowledge extraction from rna-seq cancer data through equivalent classification rules. *Bioinformatics*, 32(5):697–704, 2016.

[CFW18] Eleonora Cappelli, Giovanni Felici, and Emanuel Weitschek. Combining dna methylation and rna sequencing data of cancer for supervised knowledge extraction. *BioData mining*, 11(1):22, 2018.

[CHH+17] Min Chen, Yixue Hao, Kai Hwang, Lu Wang, and Lin Wang. Disease prediction by machine learning over big data from healthcare communities. *Ieee Access*, 5:8869–8879, 2017.

[Chi17] Davide Chicco. Ten quick tips for machine learning in computational biology. *BioData mining*, 10(1):35, 2017.

[CKM+16] Stefano Ceri, Abdulrahman Kaitoua, Marco Masseroli, Pietro Pinoli, and Francesco Venco. Data management for heterogeneous genomic datasets. *IEEE/ACM transactions on computational biology and bioinformatics*, 14(6):1251–1264, 2016.

[CL13] Fengqi Chang and Marilyn M Li. Clinical application of amplicon-based next-generation sequencing in cancer. *Cancer genetics*, 206(12):413–419, 2013.

[CLC+13] Kristian Cibulskis, Michael S Lawrence, Scott L Carter, Andrey Sivachenko, David Jaffe, Carrie Sougnez, Stacey Gabriel, Matthew Meyerson, Eric S Lander, and Gad Getz. Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nature biotechnology*, 31(3):213, 2013.

[CMP03] Francis S Collins, Michael Morgan, and Aristides Patrinos. The human genome project: lessons from large-scale biology. *Science*, 300(5617):286–290, 2003.

[Coh95] William W. Cohen. Fast effective rule induction. In *In Proceedings of the Twelfth International Conference on Machine Learning*, pages 115–123. Morgan Kaufmann, 1995.

[Con14] Gene Ontology Consortium. Gene ontology consortium: going forward. *Nucleic acids research*, 43(D1):D1049–D1056, 2014.

[CP08] Mark J Chaisson and Pavel A Pevzner. Short read fragment assembly of bacterial genomes. *Genome research*, 18(2):324–330, 2008.

[CPR+10a] Donald F Conrad, Dalila Pinto, Richard Redon, Lars Feuk, Omer Gokcumen, Yujun Zhang, Jan Aerts, T Daniel Andrews, Chris Barnes, Peter Campbell, et al. Origins and functional impact of copy number variation in the human genome. *Nature*, 464(7289):704, 2010.

[CPR+10b] Donald F Conrad, Dalila Pinto, Richard Redon, Lars Feuk, Omer Gokcumen, Yujun Zhang, Jan Aerts, T Daniel Andrews, Chris Barnes, Peter Campbell, Tomas Fitzgerald, Min Hu, Chun Ihm Hwa, Kati Kristiansson, Daniel G MacArthur, Jeffrey R MacDonald, Ifejinelo Onyiah, Andy Wing Chun Pang, Sam Robson, Kathy Stirrups, Armand Valsesia, Klaudia Walter, John Wei, Chris Tyler-Smith, Nigel P Carter, Charles Lee, Stephen W Scherer, and Matthew E Hurles. Origins and functional impact of copy number variation in the human genome. *Nature*, 464(7289):704–712, 2010.

[CPT04] Mark Chaisson, Pavel Pevzner, and Haixu Tang. Fragment assembly with short reads. *Bioinformatics*, 20(13):2067–2074, 2004.

[CPW11] Elcie Chan, Daniel Estévez Prado, and Joanne Barnes Weidhaas. Cancer microrna: from subtype profiling to predictors of response to therapy. *Trends in molecular medicine*, 17(5):235–243, 2011.

[CST00] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, 2000.

[CW06] Joseph A Cruz and David S Wishart. Applications of machine learning in cancer prediction and prognosis. *Cancer informatics*, 2:117693510600200030, 2006.

[CW17] Eleonora Cappelli and Emanuel Weitschek. Extending the genomic data model and the genometric query language with domain taxonomies. In *International Conference on Web Engineering*, pages 567–574. Springer, 2017.

[CWBF16] Fabio Cumbo, Emanuel Weitschek, Paola Bertolazzi, and Giovanni Felici. Iris-tcga: An information retrieval and integration system for genomic data of cancer. In *International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics*, pages 160–171. Springer, 2016.

[CWC19] Eleonora Cappelli, Emanuel Weitschek, and Fabio Cumbo. Smart persistence and accessibility of genomic and clinical data. In *International Conference on Database and Expert Systems Applications*, pages 8–14. Springer, 2019.

[CWT+14] Kun-Huang Chen, Kung-Jeng Wang, Min-Lung Tsai, Kung-Min Wang, Angelia Melani Adrian, Wei-Chung Cheng, Tzu-Sen Yang, Nai-Chia Teng, Kuo-Pin Tan, and Ku-Shang Chang. Gene selection for cancer identification: a decision tree model empowered by particle swarm optimization algorithm. *BMC bioinformatics*, 15(1):49, 2014.

[CZC+14] Chao Chen, Chunling Zhang, Lijun Cheng, James L Reilly, Jeffrey R Bishop, John A Sweeney, Hua-Yun Chen, Elliot S Gershon, and Chunyu Liu. Correlation between dna methylation and gene expression in the brains of patients with bipolar disorder and schizophrenia. *Bipolar disorders*, 16(8):790–799, 2014.

[CZV17] Qingyu Chen, Justin Zobel, and Karin Verspoor. Duplicates, redundancies and inconsistencies in the primary nucleotide databases: a descriptive study. *Database*, 2017, 2017.

[DCHW16] Su-Ping Deng, Shaolong Cao, De-Shuang Huang, and Yu-Ping Wang. Identifying stages of kidney renal cell carcinoma by combining gene expression and dna methylation data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2016.

[DGLLR07] Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. On reconciling data exchange, data integra-

tion, and peer data management. In *Proceedings of the twenty-sixth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 133–142. ACM, 2007.

[DOTW97] Susan B. Davidson, Christian Overton, Val Tannen, and Limsoon Wong. Biokleisli: A digital library for biomedical researchers. *International Journal on Digital Libraries*, 1(1):36–53, 1997.

[DZDW13] Junbo Duan, Ji-Gang Zhang, Hong-Wen Deng, and Yu-Ping Wang. Comparative studies of copy number variation detection methods for next-generation sequencing technologies. *PloS one*, 8(3):e59128, 2013.

[DZH+10] Pan Du, Xiao Zhang, Chiang-Ching Huang, Nadereh Jafari, Warren A Kibbe, Lifang Hou, and Simon M Lin. Comparison of beta-value and m-value methods for quantifying methylation levels by microarray analysis. *BMC bioinformatics*, 11(1):587, 2010.

[EDL02] Ron Edgar, Michael Domrachev, and Alex E Lash. Gene expression omnibus: Ncbi gene expression and hybridization array data repository. *Nucleic acids research*, 30(1):207–210, 2002.

[EDS+06] Tina A Eyre, Fabrice Ducluzeau, Tam P Sneddon, Sue Povey, Elspeth A Bruford, and Michael J Lush. The hugo gene nomenclature database, 2006 updates. *Nucleic acids research*, 34(suppl_1):D319–D321, 2006.

[EGFF16] Christoph Endrullat, Jörn Glökler, Philipp Franke, and Marcus Frohme. Standardization and quality management in next-generation sequencing. *Applied & Translational Genomics*, 10:2–9, 2016.

[EHG+13] Jeyanthy Eswaran, Anelia Horvath, Sucheta Godbole, Sirigiri Divijendra Reddy, Prakriti Mudvari, Kazufumi Ohshiro, Dinesh Cyanam, Sujit Nair, Suzanne AW Fuqua, Kornelia Polyak, Liliana D Florea, and Rakesh Kumar. Rna sequencing of cancer reveals novel splicing alterations. *Scientific reports*, 3:1689, 2013.

[Ehr02] Melanie Ehrlich. Dna methylation in cancer: too much, but also too little. *Oncogene*, 21(35):5400, 2002.

[EUA96] Thure Etzold, Anatoly Ulyanov, and Patrick Argos. Srs: Information retrieval system for molecular biology data banks. *Methods in enzymology*, 266:114–128, 1996.

[exp] The express.js framework. `http://expressjs.com/it`. Accessed 6 June 2018.

[FAB+11] Paul Flicek, M Ridwan Amode, Daniel Barrell, Kathryn Beal, Simon Brent, Denise Carvalho-Silva, Peter Clapham, Guy Coates, Susan Fairley, Stephen Fitzgerald, et al. Ensembl 2012. *Nucleic acids research*, 40(D1):D84–D90, 2011.

[FFJ+14] Thomas Fleischer, Arnoldo Frigessi, Kevin C Johnson, Hege Edvardsen, Nizar Touleimat, Jovana Klajic, Margit LH Riis, Vilde D Haakensen, Fredrik Wärnberg, Bjørn Naume, et al. Genome-wide dna methylation profiles in progression to in situ and invasive carcinoma of the breast with impact on gene transcription and prognosis. *Genome biology*, 15(8):435, 2014.

[FHL+02] Andreas Freier, Ralf Hofestädt, Matthias Lange, Uwe Scholz, and Andreas Stephanik. Biodataserver: a sql-based service for the online integration of life science data. *In silico biology*, 2(2):37–57, 2002.

[FLM+15] Javier D Fernandez, Maurizio Lenzerini, Marco Masseroli, Francesco Venco, and Stefano Ceri. Ontology-based search of genomic metadata. *IEEE/ACM transactions on computational biology and bioinformatics*, 13(2):233–247, 2015.

[FXH+16] Yu Fan, Liu Xi, Daniel ST Hughes, Jianjun Zhang, Jianhua Zhang, P Andrew Futreal, David A Wheeler, and Wenyi Wang. Muse: accounting for tumor heterogeneity using a sample-specific error model improves sensitivity and specificity in mutation calling from sequencing data. *Genome biology*, 17(1):178, 2016.

[GCAM+14] David Gomez-Cabrero, Imad Abugessaisa, Dieter Maier, Andrew Teschendorff, Matthias Merkenschlager, Andreas Gisel, Esteban Ballestar, Erik Bongcam-Rudloff, Ana Conesa, and Jesper Tegnér. Data integration in the era of omics: current and future challenges. *BMC systems biology*, 8(2):I1, 2014.

[gdc] Gdc application programming interface (api). https://gdc.cancer.gov/developers/gdc-application-programming-interface-api. Accessed 6 June 2018.

[GJSvDE07] Sam Griffiths-Jones, Harpreet Kaur Saini, Stijn van Dongen, and Anton J Enright. mirbase: tools for microrna genomics. *Nucleic acids research*, 36(suppl_1):D154–D158, 2007.

[GKI15] Andrew M Gross, Jason F Kreisberg, and Trey Ideker. Analysis of matched tumor and normal profiles reveals common transcriptional and epigenetic signals shared across cancer types. *PloS one*, 10(11):e0142618, 2015.

[GST+99] Todd R Golub, Donna K Slonim, Pablo Tamayo, Christine Huard, Michelle Gaasenbeek, Jill P Mesirov, Hilary Coller, Mignon L Loh, James R Downing, Mark A Caligiuri, Clara D Bloomfield, and Eric S Lander. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *science*, 286(5439):531–537, 1999.

[Han00] M David Hanson. The client/server architecture. *Server Management*, page 3, 2000.

[Has00] Wilhelm Hasselbring. Information system integration. *Communications of the ACM*, 43(6):32–38, 2000.

[Hay14] Erika Check Hayden. Technology: the $1,000 genome. *Nature*, 507(7492):294–5, 2014.

[HCF+08] Doug Howe, Maria Costanzo, Petra Fey, Takashi Gojobori, Linda Hannick, Winston Hide, David P Hill, Renate Kania, Mary Schaeffer, Susan St Pierre, et al. Big data: The future of biocuration. *Nature*, 455(7209):47, 2008.

[HER10] Adam E Handel, George C Ebers, and Sreeram V Ramagopalan. Epigenetics: molecular mechanisms and implications for disease. *Trends in molecular medicine*, 16(1):7–16, 2010.

[HFG+12] Jennifer Harrow, Adam Frankish, Jose M Gonzalez, Electra Tapanari, Mark Diekhans, Felix Kokocinski, Bronwen L Aken,

Daniel Barrell, Amonida Zadissa, Stephen Searle, et al. Gencode: the reference human genome annotation for the encode project. *Genome research*, 22(9):1760–1774, 2012.

[HFH+09] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11(1):10–18, November 2009.

[HME12] Soumaya Hosni, Ahmed Mokaddem, and Mourad Elloumi. A new progressive multiple sequence alignment algorithm. In *2012 23rd International Workshop on Database and Expert Systems Applications*, pages 195–198. IEEE, 2012.

[IFL+05] Marilena V Iorio, Manuela Ferracin, Chang-Gong Liu, Angelo Veronese, Riccardo Spizzo, Silvia Sabbioni, Eros Magri, Massimo Pedriali, Muller Fabbri, Manuela Campiglio, et al. Microrna gene expression deregulation in human breast cancer. *Cancer research*, 65(16):7065–7070, 2005.

[IGSW12] Michael A Innis, David H Gelfand, John J Sninsky, and Thomas J White. *PCR protocols: a guide to methods and applications*. Academic press, 2012.

[ill] Illumina rna sequencing v2. `www.illumina.com/products/by-type/sequencing-kits/library-prep-kits/truseq-rna-v2.html`. Accessed 6 June 2018.

[java] Java virtual machine. `https://www.java.com`. Accessed 6 June 2018.

[javb] The javascript programming language. `https://www.javascript.com`. Accessed 6 June 2018.

[JB04] Kamel Jabbari and Giorgio Bernardi. Cytosine methylation and cpg, tpg (cpa) and tpa frequencies. *Gene*, 333:143–149, 2004.

[JFGS17] Mark A Jensen, Vincent Ferretti, Robert L Grossman, and Louis M Staudt. The nci genomic data commons as an engine for precision medicine. *Blood*, 130(4):453–459, 2017.

[Jon86] Peter A Jones. Dna methylation and cancer. *Cancer research*, 46(2):461–466, 1986.

[jqu] Jquery javascript library. `https://jquery.com`. Accessed 6 June 2018.

[KGM+17] Abdulrahman Kaitoua, Andrea Gulino, Marco Masseroli, Pietro Pinoli, and Stefano Ceri. Scalable genomic data management system on the cloud. In *2017 International Conference on High Performance Computing & Simulation (HPCS)*, pages 58–63. IEEE, 2017.

[KHB+12] Marta Kulis, Simon Heath, Marina Bibikova, Ana C Queirós, Alba Navarro, Guillem Clot, Alejandra Martínez-Trillos, Giancarlo Castellano, Isabelle Brun-Heath, Magda Pinyol, Sergio Barberán-Soler, Panagiotis Papasaikas, Pedro Jares, Sílvia Beà, Daniel Rico, Simone Ecker, Miriam Rubio, Romina Royo, Vincent Ho, Brandy Klotzle, Lluis Hernández, Laura Conde, Mónica López-Guerra, Dolors Colomer, Neus Villamor, Marta Aymerich, María Rozman, Mónica Bayes, Marta Gut, Josep L Gelpí, Modesto Orozco, Jian-Bing Fan, Víctor Quesada, Xose S Puente, David G Pisano, Alfonso Valencia, Armando López-Guillermo, Ivo Gut, Carlos López-Otín, Elías Campo, and José I Martín-Subero. Epigenomic analysis detects widespread gene-body dna hypomethylation in chronic lymphocytic leukemia. *Nature genetics*, 44(11):1236–1242, 2012.

[KLW13] Daniel C Koboldt, David E Larson, and Richard K Wilson. Using varscan 2 for germline variant calling and somatic mutation detection. *Current protocols in bioinformatics*, 44(1):15–4, 2013.

[KME+15] Anshul Kundaje, Wouter Meuleman, Jason Ernst, Misha Bilenky, Angela Yen, Alireza Heravi-Moussavi, Pouya Kheradpour, Zhizhuo Zhang, Jianrong Wang, Michael J Ziller, et al. Integrative analysis of 111 reference human epigenomes. *Nature*, 518(7539):317, 2015.

[Koh95] Ron Kohavi. Wrappers for performance enhancement and oblivious decision graphs. Technical report, Carneige-Mellon University Pittsburhg PA, Department of Computer Science, 1995.

[KSL+13] Daniel C Koboldt, Karyn Meltz Steinberg, David E Larson, Richard K Wilson, and Elaine R Mardis. The next-generation sequencing revolution and its impact on genomics. *Cell*, 155(1):27–38, 2013.

[KTN04] Sudhir Kumar, Koichiro Tamura, and Masatoshi Nei. Mega3: integrated software for molecular evolutionary genetics analysis and sequence alignment. *Briefings in bioinformatics*, 5(2):150–163, 2004.

[KWR+01] Javed Khan, Jun S Wei, Markus Ringner, Lao H Saal, Marc Ladanyi, Frank Westermann, Frank Berthold, Manfred Schwab, Cristina R Antonescu, Carsten Peterson, and Paul S. Meltzer. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature medicine*, 7(6):673, 2001.

[KZL+12] Daniel C Koboldt, Qunyuan Zhang, David E Larson, Dong Shen, Michael D McLellan, Ling Lin, Christopher A Miller, Elaine R Mardis, Li Ding, and Richard K Wilson. Varscan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome research*, 22(3):568–576, 2012.

[LC14] Yixue Li and Luonan Chen. Big biological data: challenges and opportunities. *Genomics, proteomics & bioinformatics*, 12(5):187, 2014.

[LD11] Bo Li and Colin N Dewey. RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC Bioinformatics*, 12(1):323, 2011.

[LDC+19] Katrina Learned, Ann Durbin, Robert Currie, Ellen Towle Kephart, Holly C Beale, Lauren M Sanders, Jacob Pfeil, Theodore C Goldstein, Sofie R Salama, David Haussler, et al. Barriers to accessing public cancer genomic data. *Scientific Data*, 6(1):98, 2019.

[LHC+11] David E Larson, Christopher C Harris, Ken Chen, Daniel C Koboldt, Travis E Abbott, David J Dooling, Timothy J Ley, Elaine R Mardis, Richard K Wilson, and Li Ding. Somaticsniper: identification of somatic point mutations in whole genome sequencing data. *Bioinformatics*, 28(3):311–317, 2011.

[LHW+09] Heng Li, Bob Handsaker, Alec Wysoker, Tim Fennell, Jue Ruan, Nils Homer, Gabor Marth, Goncalo Abecasis, and Richard Durbin. The sequence alignment/map format and samtools. *Bioinformatics*, 25(16):2078–2079, 2009.

[LLDS18] Catherine Li, Juyon Lee, Jessica Ding, and Shuying Sun. Integrative analysis of gene expression and methylation data for breast cancer cell lines. *BioData mining*, 11(1):13, 2018.

[LLH+18] Jianfang Liu, Tara Lichtenberg, Katherine A Hoadley, Laila M Poisson, Alexander J Lazar, Andrew D Cherniack, Albert J Kovatich, Christopher C Benz, Douglas A Levine, Adrian V Lee, et al. An integrated tcga pan-cancer clinical data resource to drive high-quality survival outcome analytics. *Cell*, 173(2):400–416, 2018.

[LLL+15] Zheng Li, Huizi Lei, Min Luo, Yi Wang, Lei Dong, Yanni Ma, Changzheng Liu, Wei Song, Fang Wang, Junwu Zhang, Jianxiong Shen, and Jia Yu. Dna methylation downregulated mir-10b acts as a tumor suppressor in gastric cancer. *Gastric Cancer*, 18(1):43–54, 2015.

[LLS+17] Jessica W Lau, Erik Lehnert, Anurag Sethi, Raunaq Malhotra, Gaurav Kaushik, Zeynep Onder, Nick Groves-Kirkby, Aleksandar Mihajlovic, Jack DiGiovanna, Mladen Srdic, et al. The cancer genomics cloud: collaborative, reproducible, and democratized—a new paradigm in large-scale computational research. *Cancer research*, 77(21):e3–e6, 2017.

[LMMS+07] Brenton Louie, Peter Mork, Fernando Martin-Sanchez, Alon Halevy, and Peter Tarczy-Hornoch. Data integration and genomic medicine. *Journal of biomedical informatics*, 40(1):5–16, 2007.

[LMT+10] Thomas Liggett, Anatoliy Melnikov, Shilpa Tilwalli, Qilong Yi, Haiyan Chen, Charles Replogle, Xuan Feng, Anthony Reder, Dusan Stefoski, Roumen Balabanov, and Victor Levenson. Methylation patterns of cell-free plasma dna in relapsing–remitting multiple sclerosis. *Journal of the neurological sciences*, 290(1):16–21, 2010.

[LPW+06] Thomas J Lee, Yannick Pouliot, Valerie Wagner, Priyanka Gupta, David WJ Stringer-Calvert, Jessica D Tenenbaum, and Peter D Karp. Biowarehouse: a bioinformatics database warehouse toolkit. *BMC bioinformatics*, 7(1):1, 2006.

[LSB16] Marie-Gabrielle Ludwig, Klaus Seuwen, and James P Bridges. Adhesion gpcr function in pulmonary development and disease. In *Adhesion G Protein-coupled Receptors*, pages 309–327. Springer, New York, NY, USA, 2016.

[LTBD16] Yang Li, Xu-Qing Tang, Zhonghu Bai, and Xiaofeng Dai. Exploring the intrinsic differences among breast tumor subtypes defined using immunohistochemistry markers based on the decision tree. *Scientific reports*, 6:35773, 2016.

[LTN+17] Steve Tin-Chi Luk, Man Tong, Kai Yu Ng, Kevin Yuk-Lap Yip, Xin Yuan Guan, and Stephanie Ma. Identification of zfp42/rex1 as a regulator of cancer stemness in cd133+ liver cancer stem cells by genome-wide dna methylation analysis. *Nature genetics*, 77(13):4352, 2017.

[LVRF13] Ruolan Liu, Radhika A Vaishnav, Andrew M Roberts, and Robert P Friedland. Humans have antibodies against a plant virus: evidence from tobacco mosaic virus. *PloS one*, 8(4):e60621, 2013.

[LWGZ16] Jake Luo, Min Wu, Deepika Gopukumar, and Yiqing Zhao. Big data application in biomedical research and health care: a literature review. *Biomedical informatics insights*, 8:BII–S31559, 2016.

[MAOP01] Fabian Model, Peter Adorjan, Alexander Olek, and Christian Piepenbrock. Feature selection for dna methylation based cancer classification. *Bioinformatics*, 17(suppl 1):S157–S164, 2001.

[Mar08] Elaine R Mardis. The impact of next-generation sequencing technology on genetics. *Trends in genetics*, 24(3):133–141, 2008.

[MCP+18] Marco Masseroli, Arif Canakoglu, Pietro Pinoli, Abdulrahman Kaitoua, Andrea Gulino, Olha Horlova, Luca Nanni, Anna Bernasconi, Stefano Perna, Eirini Stamoulakatou, et al. Processing of big heterogeneous genomic datasets for tertiary analysis of next generation sequencing data. *Bioinformatics*, 35(5):729–736, 2018.

[MdRD+13] Laetitia Marisa, Aurélien de Reyniès, Alex Duval, Janick Selves, Marie Pierre Gaub, Laure Vescovo, Marie-Christine

Etienne-Grimaldi, Renaud Schiappa, Dominique Guenot, Mira Ayadi, Sylvain Kirzin, Maurice Chazal, Jean-François Fléjou, Daniel Benchimol, Anne Berger, Arnaud Lagarde, Erwan Pencreach, Françoise Piard, Dominique Elias, Yann Parc, Sylviane Olschwang, Gérard Milano, Pierre Laurent-Puig, and Valérie Boige. Gene expression classification of colon cancer into molecular subtypes: characterization, validation, and prognostic value. *PLoS medicine*, 10(5):e1001453, 2013.

[MDT11] Cliff Meldrum, Maria A Doyle, and Richard W Tothill. Next-generation sequencing for cancer diagnostics: a practical perspective. *Clin Biochem Rev*, 32(4):177–195, 2011.

[Met10] Michael L Metzker. Sequencing technologies—the next generation. *Nature reviews genetics*, 11(1):31, 2010.

[MGR+98] Nikola Mašić, Alenka Gagro, Sabina Rabatić, Ante Sabioncello, Gorana Dašić, Branimir Jakšić, and Branko Vitale. Decision-tree approach to the immunophenotype-based prognosis of the b-cell chronic lymphocytic leukemia. *American journal of hematology*, 59(2):143–148, 1998.

[MHB+10] Aaron McKenna, Matthew Hanna, Eric Banks, Andrey Sivachenko, Kristian Cibulskis, Andrew Kernytsky, Kiran Garimella, David Altshuler, Stacey Gabriel, Mark Daly, and Mark A DePristo. The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Research*, 20(9):1297–1303, 2010.

[MKK+15] Jun Mashima, Yuichi Kodama, Takehide Kosuge, Takatomo Fujisawa, Toshiaki Katayama, Hideki Nagasaki, Yoshihiro Okuda, Eli Kaminuma, Osamu Ogasawara, Kousaku Okubo, et al. Dna data bank of japan (ddbj) progress report. *Nucleic acids research*, 44(D1):D51–D57, 2015.

[MKPC16] Marco Masseroli, Abdulrahman Kaitoua, Pietro Pinoli, and Stefano Ceri. Modeling and interoperability of heterogeneous genomic big data for integrative processing and querying. *Methods*, 111:3–11, 2016.

[MMBR+14] Marco Masseroli, Barend Mons, Erik Bongcam-Rudloff, Stefano Ceri, Alexander Kel, François Rechenmann, Frederique Lisacek,

and Paolo Romano. Integrated bio-search: challenges and trends for the integration, search and comprehensive processing of biological information. *BMC bioinformatics*, 15(1):1, 2014.

[MNB+10] Alika K Maunakea, Raman P Nagarajan, Mikhail Bilenky, Tracy J Ballinger, Cletus D'Souza, Shaun D Fouse, Brett E Johnson, Chibo Hong, Cydney Nielsen, Yongjun Zhao, Gustavo Turecki, Allen Delaney, Richard Varhol, Nina Thiessen, Vivi M Shchors, Ksenya Heine, David H Rowitch, Xiaoyun Xing, Chris Fiore, Maximiliaan Schillebeeckx, Steven J M Jones, David Haussler, Marco A Marra, Martin Hirst, Ting Wang, and Joseph F Costello. Conserved role of intragenic dna methylation in regulating alternative promoters. *Nature*, 466(7303):253–257, 2010.

[MOPT10] Donna Maglott, Jim Ostell, Kim D Pruitt, and Tatiana Tatusova. Entrez gene: gene-centered information at ncbi. *Nucleic acids research*, 39(suppl_1):D52–D57, 2010.

[MPGC14] Marco Masseroli, Matteo Picozzi, Giorgio Ghisalberti, and Stefano Ceri. Explorative search of distributed bio-data to answer complex biomedical questions. *BMC bioinformatics*, 15(1):1, 2014.

[MPV+15] M. Masseroli, P. Pinoli, F. Venco, A. Kaitoua, V. Jalili, F. Palluzzi, H. Muller, and S. Ceri. GenoMetric Query Language: a novel approach to large-scale genomic data management. *Bioinformatics*, 31(12):1881–1888, Jun 2015.

[MST+10] Chiara Magri, Emilio Sacchetti, Michele Traversa, Paolo Valsecchi, Rita Gardella, Cristian Bonvicini, Alessandra Minelli, Massimo Gennarelli, and Sergio Barlati. New copy number variations in schizophrenia. *PloS one*, 5(10):e13422, 2010.

[MT14] Davide Martinenghi and Riccardo Torlone. Taxonomy-based relaxation of query answering in relational databases. *The VLDB Journal—The International Journal on Very Large Data Bases*, 23(5):747–769, 2014.

[MTK+08] Jonathan Mill, Thomas Tang, Zachary Kaminsky, Tarang Khare, Simin Yazdanpanah, Luigi Bouchard, Peixin Jia, Abbas Assadzadeh, James Flanagan, Axel Schumacher, Sun-Chong

Wang, and Arturas Petronis. Epigenomic profiling reveals dna-methylation changes associated with major psychosis. *The American Journal of Human Genetics*, 82(3):696–711, 2008.

[MWM+08] Ali Mortazavi, Brian A Williams, Kenneth McCue, Lorian Schaeffer, and Barbara Wold. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nature Methods*, 5(7):621–628, 2008.

[MWZG13] Xiaotu Ma, Yi-Wei Wang, Michael Q Zhang, and Adi F Gazdar. Dna methylation data analysis and its application to cancer research. *Epigenomics*, 5(3):10.2217/epi.13.26, 2013.

[mys] Mysql database management system. `https://www.mysql.com`. Accessed 6 June 2018.

[MZY+13] Jeffrey R MacDonald, Robert Ziman, Ryan KC Yuen, Lars Feuk, and Stephen W Scherer. The database of genomic variants: a curated collection of structural variation in the human genome. *Nucleic acids research*, 42(D1):D986–D992, 2013.

[nod] The nodejs technology. `https://nodejs.org`. Accessed 6 June 2018.

[NRE14] Putri W Novianti, Kit CB Roes, and Marinus JC Eijkemans. Evaluation of gene expression classification studies: factors associated with classification performance. *PloS one*, 9(4):e96063, 2014.

[NYT+03] Yasushi Nakamura, Hironao Yasuoka, Masahiko Tsujimoto, Qifeng Yang, Shigeru Imabun, Masaaki Nakahara, Kazuyasu Nakao, Misa Nakamura, Ichiro Mori, and Kennichi Kakudo. Prognostic significance of vascular endothelial growth factor d in breast carcinoma with long-term follow-up. *Clinical Cancer Research*, 9(2):716–721, 2003.

[OIO+16] Jelili Oyelade, Itunuoluwa Isewon, Funke Oladipupo, Olufemi Aromolaran, Efosa Uwoghiren, Faridah Ameh, Moses Achas, and Ezekiel Adebiyi. Clustering algorithms: Their application to gene expression data. *Bioinformatics and Biology insights*, 10:BBI–S38316, 2016.

[OM11] Fatih Ozsolak and Patrice M Milos. Rna sequencing: advances, challenges and opportunities. *Nature reviews genetics*, 12(2):87, 2011.

[OWB+15] Nuala A O'Leary, Mathew W Wright, J Rodney Brister, Stacy Ciufo, Diana Haddad, Rich McVeigh, Bhanu Rajput, Barbara Robbertse, Brian Smith-White, Danso Ako-Adjei, et al. Reference sequence (refseq) database at ncbi: current status, taxonomic expansion, and functional annotation. *Nucleic acids research*, 44(D1):D733–D745, 2015.

[PE10] Anna Portela and Manel Esteller. Epigenetic modifications and human disease. *Nature biotechnology*, 28(10):1057–1068, 2010.

[PKP+14] Mehdi Pirooznia, Melissa Kramer, Jennifer Parla, Fernando S Goes, James B Potash, W Richard McCombie, and Peter P Zandi. Validation and assessment of variant calling pipelines for next-generation sequencing. *Human genomics*, 8(1):14, 2014.

[PMP+10] Joseph K Pickrell, John C Marioni, Athma A Pai, Jacob F Degner, Barbara E Engelhardt, Everlyne Nkadori, Jean-Baptiste Veyrieras, Matthew Stephens, Yoav Gilad, and Jonathan K Pritchard. Understanding mechanisms underlying human gene expression variation with rna sequencing. *Nature*, 464(7289):768–772, 2010.

[PRP+17] Milos Pavlovic, Pradipta Ray, Kristina Pavlovic, Aaron Kotamarti, Min Chen, and Michael Q Zhang. Direction: A machine learning framework for predicting and characterizing dna methylation and hydroxymethylation in mammalian genomes. *Bioinformatics*, btx316(btx316):10.1093/bioinformatics/btx316, 2017.

[PWD+14] Dimitris Polychronopoulos, Emanuel Weitschek, Slavica Dimitrieva, Philipp Bucher, Giovanni Felici, and Yannis Almirantis. Classification of selectively constrained dna elements using feature vectors and rule-based classifiers. *Genomics*, 104(2):79–86, 2014.

[PYKM18] Nikolaos Perakakis, Alireza Yazdani, George E Karniadakis, and Christos Mantzoros. Omics, big data and machine learning as tools to propel understanding of biological mechanisms and to

discover novel diagnostics and therapeutics. *Metabolism-Clinical and Experimental*, 87:A1–A9, 2018.

[PYOA16] Elham Pashaei, Alper Yilmaz, Mustafa Ozen, and Nizamettin Aydin. A novel method for splice sites prediction using sequence component and hidden markov model. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3076–3079. IEEE, 2016.

[QH10] Aaron R Quinlan and Ira M Hall. Bedtools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, 26(6):841–842, 2010.

[Qui14] J Ross Quinlan. *C4. 5: programs for machine learning*. Elsevier, 2014.

[RCDV+14] Giuseppe Roscilli, Manuela Cappelletti, Claudia De Vitis, Gennaro Ciliberto, Arianna Di Napoli, Luigi Ruco, Rita Mancini, and Luigi Aurisicchio. Circulating mmp11 and specific antibody immune response in breast and prostate cancer patients. *Journal of translational medicine*, 12(1):54, 2014.

[red] Remote dictionary server (redis). `https://redis.io`. Accessed 6 June 2018.

[Saj06] Paul Sajda. Machine learning for detection and diagnosis of disease. *Annu. Rev. Biomed. Eng.*, 8:537–565, 2006.

[SAR+07] Barry Smith, Michael Ashburner, Cornelius Rosse, Jonathan Bard, William Bug, Werner Ceusters, Louis J Goldberg, Karen Eilbeck, Amelia Ireland, Christopher J Mungall, et al. The obo foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology*, 25(11):1251, 2007.

[SBB+00] Robert Stevens, Patricia Baker, Sean Bechhofer, Gary Ng, Alex Jacoby, Norman W Paton, Carole A Goble, and Andy Brass. Tambis: transparent access to multiple bioinformatics information sources. *Bioinformatics*, 16(2):184–186, 2000.

[SBJ+17] Arunima Shilpi, Yingtao Bi, Segun Jung, Samir K Patra, and Ramana V Davuluri. Identification of genetic and epigenetic variants associated with breast cancer prognosis by integrative bioinformatics analysis. *Cancer informatics*, 16:CIN–S39783, 2017.

[SBT+19] Nikola Simidjievski, Cristian Bodnar, Ifrah Tariq, Paul Scherer, Helena Andres-Terre, Zohreh Shams, Mateja Jamnik, et al. Variational autoencoders for cancer data integration: Design principles and computational practice. *BioRxiv*, page 719542, 2019.

[SBvdB+02] Guenter Stoesser, Wendy Baker, Alexandra van den Broek, Evelyn Camon, Maria Garcia-Pastor, Carola Kanz, Tamara Kulikova, Rasko Leinonen, Quan Lin, Vincent Lombard, et al. The embl nucleotide sequence database. *Nucleic acids research*, 30(1):21–26, 2002.

[SC18] Marzia Settino and Mario Cannataro. Survey of main tools for querying and analyzing tcga data. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1711–1718. IEEE, 2018.

[Sch07] Stephan C Schuster. Next-generation sequencing transforms today's biology. *Nature methods*, 5(1):16, 2007.

[Sch15] Michael C Schatz. Biological data sciences in genome research. *Genome research*, 25(10):1417–1422, 2015.

[Sha01] Ron Shamir. Algorithms for molecular biology. *Fall Semester, Lectures*, pages 1–3, 2001.

[She14] Cormac Sheridan. Illumina claims $1,000 genome win. *Nature Biotechnology*, 32(2):115, 2014.

[SIH+13] Valgardur Sigurdsson, Saevar Ingthorsson, Bylgja Hilmarsdottir, Sigrun M Gustafsdottir, Sigridur Rut Franzdottir, Ari Jon Arason, Eirikur Steingrimsson, Magnus K Magnusson, and Thorarinn Gudjonsson. Expression and functional role of sprouty-2 in breast morphogenesis. *PloS one*, 8(4):e60798, 2013.

[SJ08] Jay Shendure and Hanlee Ji. Next-generation DNA sequencing. *Nature Biotechnology*, 26(10):1135–1145, 2008.

[SKK+14] Alicia K Smith, Varun Kilaru, Mehmet Kocak, Lynn M Almli, Kristina B Mercer, Kerry J Ressler, Frances A Tylavsky, and Karen N Conneely. Methylation quantitative trait loci (meqtls) are consistently detected across ancestry, developmental stage, and tissue type. *BMC genomics*, 15(1):145, 2014.

[SL90] Amit P Sheth and James A Larson. Federated database systems for managing distributed, heterogeneous, and autonomous databases. *ACM Computing Surveys (CSUR)*, 22(3):183–236, 1990.

[SLH+16] Lillian L Siu, Mark Lawler, David Haussler, Bartha Maria Knoppers, Jeremy Lewin, Daniel J Vis, Rachel G Liao, Fabrice Andre, Ian Banks, J Carl Barrett, et al. Facilitating a culture of responsible and effective sharing of cancer genome data. *Nature medicine*, 22(5):464, 2016.

[SMB+17] Peter D Stenson, Matthew Mort, Edward V Ball, Katy Evans, Matthew Hayden, Sally Heywood, Michelle Hussain, Andrew D Phillips, and David N Cooper. The human gene mutation database: towards a comprehensive repository of inherited mutation data for medical research, genetic diagnosis and next-generation sequencing studies. *Human genetics*, 136(6):665–677, 2017.

[SMLS+14] Hreinn Stefansson, Andreas Meyer-Lindenberg, Stacy Steinberg, Brynja Magnusdottir, Katrin Morgen, Sunna Arnarsdottir, Gyda Bjornsdottir, G Bragi Walters, Gudrun A Jonsdottir, Orla M Doyle, et al. Cnvs conferring risk of autism or schizophrenia affect cognition in controls. *Nature*, 505(7483):361, 2014.

[SNC77] Frederick Sanger, Steven Nicklen, and Alan R Coulson. Dna sequencing with chain-terminating inhibitors. *Proceedings of the national academy of sciences*, 74(12):5463–5467, 1977.

[SNM+03] Christos Sotiriou, Soek-Ying Neo, Lisa M McShane, Edward L Korn, Philip M Long, Amir Jazaeri, Philippe Martiat, Steve B Fox, Adrian L Harris, and Edison T Liu. Breast cancer classification and prognosis based on gene expression profiles from a population-based study. *Proceedings of the National Academy of Sciences*, 100(18):10393–10398, 2003.

[SPB+15] Silvano Squizzato, Young Mi Park, Nicola Buso, Tamer Gur, Andrew Cowley, Weizhong Li, Mahmut Uludag, Sangya Pundir, Jennifer A Cham, Hamish McWilliam, and Rodrigo Lopez. The ebi search engine: providing search and retrieval functionality

for biological data from embl-ebi. *Nucleic acids research*, page gkv316, 2015.

[SRAV+16] Javier Soto, Carlos Rodriguez-Antolin, Elena Vallespin, Javier De Castro Carpeno, and Inmaculada Ibanez De Caceres. The impact of next-generation sequencing on the dna methylation–based translational cancer research. *Translational Research*, 169:1–18, 2016.

[SRD+15] Erin M Siegel, Bridget M Riggs, Amber L Delmas, Abby Koch, Ardeshir Hakam, and Kevin D Brown. Quantitative dna methylation analysis of candidate genes in cervical cancer. *PLoS One*, 10(3):e0122495, 2015.

[STSC14] Clare Stirzaker, Phillippa C Taberlay, Aaron L Statham, and Susan J Clark. Mining cancer methylomes: prospects and challenges. *Trends in Genetics*, 30(2):75–84, 2014.

[SWK+01] S. T. Sherry, M.-H. Ward, M. Kholodov, J. Baker, L. Phan, E. M. Smigielski, and K. Sirotkin. dbSNP: the NCBI database of genetic variation. *Nucleic Acids Research*, 29(1):308–311, 01 2001.

[TAK+12] Gidon Toperoff, Dvir Aran, Jeremy D Kark, Michael Rosenberg, Tatyana Dubnikov, Batel Nissan, Julio Wainstein, Yechiel Friedlander, Ephrat Levy-Lahad, Benjamin Glaser, and Asaf Hellman. Genome-wide survey reveals predisposing diabetes type 2-related dna methylation variations in human peripheral blood. *Human molecular genetics*, 21(2):371–383, 2012.

[TKMO99] Tatiana A Tatusova, Ilene Karsch-Mizrachi, and James A Ostell. Complete genomes in www entrez: data representation and analysis. *Bioinformatics*, 15(7):536–543, 1999.

[TKR+10] Bernd Timmermann, Martin Kerick, Christina Roehr, Axel Fischer, Melanie Isau, Stefan T Boerno, Andrea Wunderlich, Christian Barmeyer, Petra Seemann, Jana Koenig, et al. Somatic mutation profiles of msi and mss colorectal cancer identified by whole exome next generation sequencing and bioinformatics analysis. *PloS one*, 5(12):e15661, 2010.

[TSK05] P. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining*. Addison Wesley, 2005.

[TWP+10] Cole Trapnell, Brian A Williams, Geo Pertea, Ali Mortazavi, Gordon Kwan, Marijke J Van Baren, Steven L Salzberg, Barbara J Wold, and Lior Pachter. Transcript assembly and quantification by rna-seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nature biotechnology*, 28(5):511, 2010.

[UFH+15] Mathias Uhlén, Linn Fagerberg, Björn M Hallström, Cecilia Lindskog, Per Oksvold, Adil Mardinoglu, Åsa Sivertsson, Caroline Kampf, Evelina Sjöstedt, Anna Asplund, et al. Tissue-based map of the human proteome. *Science*, 347(6220):1260419, 2015.

[uni16] Uniprot: the universal protein knowledgebase. *Nucleic acids research*, 45(D1):D158–D169, 2016.

[VHSE+13] AQ Van Hoesel, Y Sato, DA Elashoff, RR Turner, AE Giuliano, JM Shamonki, PJK Kuppen, CJH Van De Velde, and DSB Hoon. Assessment of dna methylation status in early stages of breast cancer development. *British journal of cancer*, 108(10):2033, 2013.

[vSKP+14] Tempest A van Schaik, Nadezda V Kovalevskaya, Elena Protopapas, Hamza Wahid, and Fiona GG Nielsen. The need to redefine genomic data sharing: A focus on data accessibility. *Applied & translational genomics*, 3(4):100–104, 2014.

[WCC+18] Emanuel Weitschek, Fabio Cumbo, Eleonora Cappelli, Giovanni Felici, and Paola Bertolazzi. Classifying big dna methylation data: a gene-oriented approach. In *International Conference on Database and Expert Systems Applications*, pages 138–149. Springer, 2018.

[WCCF16] Emanuel Weitschek, Fabio Cumbo, Eleonora Cappelli, and Giovanni Felici. Genomic data integration: A case study on next generation sequencing of cancer. In *2016 27th International Workshop on Database and Expert Systems Applications (DEXA)*, pages 49–53. IEEE, 2016.

[WCM+13] John N Weinstein, Eric A Collisson, Gordon B Mills, Kenna R Mills Shaw, Brad A Ozenberger, Kyle Ellrott, Ilya Shmulevich, Chris Sander, Joshua M Stuart, Cancer Genome Atlas Research

Network, et al. The cancer genome atlas pan-cancer analysis project. *Nature genetics*, 45(10):1113, 2013.

[WDLC+18] Emanuel Weitschek, Silvia Di Lauro, Eleonora Cappelli, Paola Bertolazzi, and Giovanni Felici. Camurweb: a classification software and a large knowledge base for gene expression data of cancer. *BMC bioinformatics*, 19(10):245, 2018.

[WFB12] Emanuel Weitschek, Giovanni Felici, and Paola Bertolazzi. Mala: a microarray clustering and classification software. In *Database and Expert Systems Applications (DEXA), 2012 23rd International Workshop on Biological Knowledge Discovery*, pages 201–205. IEEE, 2012.

[WFB13] Emanuel Weitschek, Giovanni Felici, and Paola Bertolazzi. Clinical data mining: problems, pitfalls and solutions. In *Database and Expert Systems Applications (DEXA) 2013, 24th International Workshop on Biological Knowledge Discovery and Data Mining*, pages 90–94. IEEE, 2013.

[WFF14] Emanuel Weitschek, Giulia Fiscon, and Giovanni Felici. Supervised dna barcodes species classification: analysis, comparisons and results. *BioData mining*, 7(1):1, 2014.

[WFFB15] Emanuel Weitschek, Giulia Fiscon, Giovanni Felici, and Paola Bertolazzi. Gela: A software tool for the analysis of gene expression data. In *2015 26th International Workshop on Database and Expert Systems Applications (DEXA)*, pages 31–35. IEEE, 2015.

[WIC12] Marc WICK. Geonames, the geonames geographical database, 2012.

[WJY+17] Lin Wei, Zhilin Jin, Shengjie Yang, Yanxun Xu, Yitan Zhu, and Yuan Ji. Tcga-assembler 2: software pipeline for retrieval and processing of tcga/cptac data. *Bioinformatics*, 34(9):1615–1617, 2017.

[WJZ16] Zhining Wang, Mark A Jensen, and Jean Claude Zenklusen. A practical guide to the cancer genome atlas (tcga). In *Statistical Genomics*, pages 111–141. Springer, 2016.

[WKL12] Günter P Wagner, Koryu Kin, and Vincent J Lynch. Measurement of mrna abundance using rna-seq data: Rpkm measure is inconsistent among samples. *Theory in biosciences*, 131(4):281–285, 2012.

[WNY14] Heng Wang, Dan Nettleton, and Kai Ying. Copy number variation detection using next generation sequencing read counts. *BMC bioinformatics*, 15(1):109, 2014.

[WSF+14] Emanuel Weitschek, Daniele Santoni, Giulia Fiscon, Maria C De Cola, Paola Bertolazzi, and Giovanni Felici. Next generation sequencing reads comparison with an alignment-free distance. *BMC Research Notes*, 7(1):869, 2014.

[WSZ+10] Kai Wang, Darshan Singh, Zheng Zeng, Stephen J Coleman, Yan Huang, Gleb L Savich, Xiaping He, Piotr Mieczkowski, Sara A Grimm, Charles M Perou, James N MacLeod, Derek Y Chiang, Jan F Prins, and Jinze Liu. Mapsplice: accurate mapping of rna-seq reads for splice junction discovery. *Nucleic acids research*, page gkq622, 2010.

[WV18] Rucha M Wadapurkar and Renu Vyas. Computational analysis of next generation sequencing data and its applications in clinical oncology. *Informatics in Medicine Unlocked*, 11:75–82, 2018.

[WVFB13] Emanuel Weitschek, Robin Velzen, Giovanni Felici, and Paola Bertolazzi. Blog 2.0: a software system for character-based species classification with dna barcode sequences. what it does, how to use it. *Molecular ecology resources*, 13(6):1043–1046, 2013.

[XWOZ11] Linglin Xie, Brent Weichel, Joyce Ellen Ohm, and Ke Zhang. An integrative analysis of dna methylation and rna-seq data for human heart, kidney and liver. *BMC systems biology*, 5(3):S4, 2011.

[YGZ16] Xiaofei Yang, Lin Gao, and Shihua Zhang. Comparative pan-cancer dna methylation analysis reveals cancer common and specific patterns. *Briefings in Bioinformatics*, page bbw063, 2016.

[YTS+12] Min Yu, David T Ting, Shannon L Stott, Ben S Wittner, Fatih Ozsolak, Suchismita Paul, Jordan C Ciciliano, Malgorzata E Smas, Daniel Winokur, Anna J Gilman, Matthew J Ulman,

Kristina Xega, Gianmarco Contino, Brinda Alagesan, Brian W Brannigan, Patrice M Milos, David P Ryan, Lecia V Sequist, Nabeel Bardeesy, Sridhar Ramaswamy, Mehmet Toner, Shyamala Maheswaran, and Daniel A Haber. Rna sequencing of pancreatic circulating tumour cells implicates wnt signalling in metastasis. *Nature*, 487(7408):510–513, 2012.

[ZBC+11] Junjun Zhang, Joachim Baran, Anthony Cros, Jonathan M Guberman, Syed Haider, Jack Hsu, Yong Liang, Elena Rivkin, Jianxin Wang, Brett Whitty, et al. International cancer genome consortium data portal—a one-stop shop for cancer genomics data. *Database*, 2011, 2011.

[ZC03] Yan Zeng and Bryan R Cullen. Sequence requirements for micro rna processing and function in human cells. *Rna*, 9(1):112–123, 2003.

[ZJ10] Yingying Zhang and Albert Jeltsch. The application of next generation sequencing in dna methylation analysis. *Genes*, 1(1):85–101, 2010.

[ZLW+17] Youzhi Zhu, Shuiqin Li, Qingshui Wang, Ling Chen, Kunlin Wu, Yide Huang, Xiangjin Chen, and Yao Lin. Quantitative and correlation analysis of the dna methylation and expression of dapk in breast cancer. *PeerJ*, 5:e3084, 2017.

[ZSX+15] Qing Zhao, Xingjie Shi, Yang Xie, Jian Huang, BenChang Shia, and Shuangge Ma. Combining multidimensional genomic measurements for predicting cancer prognosis: observations from tcga. *Briefings in Bioinformatics*, 16(2):291, 2015.