A first look at an automated pipeline for NGS-based breast-cancer diagnosis: the CArDIGAN approach

Giuseppe Aceto[1,2], Antonio Montieri[2],
Valerio Persico[1], Antonio Pescapé[1,2]

[1]Dipartimento di Ingegneria Elettrica e Tecnologie
dell'Informazione, Università di Napoli "Federico II"
[2]NM2 s.r.l. (Italy)
{giuseppe.aceto, valerio.persico, pescape}@unina.it
montieri@nm-2.com

Valeria D'Argenio[3,4], Francesco Salvatore[3,4]
Lucio Pastore[3,4]

[3]Dipartimento di Biochimica e Biotecnologie Mediche,
Università di Napoli "Federico II"
[4]CEINGE-Biotecnologie Avanzate and SEMM (Italy)
{valeria.dargenio, salvator, lucio.pastore}@unina.it

***Abstract*—Continuous improvements on Next-Generation Sequencing approaches are providing a wealth of data for life sciences, and massive application of ICT (Information and Communication Technologies) to molecular research has become essential for the progress of medical research. While several software tools have been developed to assist in analysis, there is still a lack of a focused and integrated solution for several of specific research goals. One such goal is molecular diagnostics of Breast Cancer (BC), the most common malignancy in females. In this paper we present, describe, and evaluate experimentally—on real data—part of a software pipeline we have designed, and are implementing, specifically aimed at assisting in BC diagnostics. The results show that the pipeline is effective in assisting and enhancing BC diagnostics, and encourage towards further automation.**

Keywords: NGS, Breast Cancer, Software Pipeline, Automated Analysis, Feature Extraction.

I. Introduction

The last ten years have been featured by the rapid development and diffusion of novel disruptive technologies that have completely changed the way researchers think to molecular research . These **novel technologies, collectively called "Next-Generation Sequencing" (NGS)**, have dramatically increased the throughput of DNA sequencing, simultaneously reducing its costs (see [13] for an introduction to the basic concepts and the techniques, both traditional and NGS-based, involved in molecular cancer diagnostics, and [20] for details on the NGS techniques). In fact, while the first human genome sequence took more than 10 years to be completed and cost USD 3 billion, now, using NGS techniques, the whole genome sequence (WGS) from a tissue sample and even from few cells can be obtained in weeks or months at a much lower cost [32]. In addition, whole exome sequencing (WES), consisting in the DNA sequences that are actually transcribed and translated, is becoming a clinically relevant procedure that allows the identification of causative mutations in patients with complex clinical presentations. To date, several NGS-based strategies have been used to improve molecular diagnostics of human diseases [10], [28]. In particular, gene panels or WES analysis have shown their potential since they allow the detection of disease-related sequence variants through the analysis of large genomic regions in a single run [28], [7].

The **massive application of ICT to molecular research** has played a major role in this evolution, being eventually included in the best practices for the sector. In fact, according to [23] in their *Guidelines for reproducibility* (of research on NGS data interpretation): "Accept that the computational component is becoming an integral component of biomedical research. As the life sciences are becoming increasingly data-driven, there will be no escape from computation and data handling." Indeed several biomedical areas are developing their specific pipelines to process NGS data (e.g. [14] in the field of Metagenomics).

This scenario has led to the creation of CArDIGAN (Cloud-bAsed Data mIning of clinically- and phenotypically-relevant Genetic variants in breAst caNcer samples), a reseach project of the University of Napoli Federico II in which biotech and computer engineer researchers closely work together for improving the molecular diagnostics of Breast Cancer (BC). **BC is the most common malignancy in females** [22]. About 10% of all BCs fall within the so-called hereditary breast and ovarian cancers (HBOCs), and can be related to a germline predisposing-mutation in the high penetrance-genes BRCA1 [31] and BRCA2 [33]. BRCA1/BRCA2 germline mutations escalate the risk of developing HBOCs by up to 20 fold [22]. Therefore, testing for BRCA gene mutations is important to improve the clinical management of the high-risk patients and of their mutation carriers family members. However, only a small fraction of at-risk patients carry mutations in these genes. Consequently, familial BC may be caused by germline mutations in other high-, moderate-, and low-penetrance cancer genes [21]. NGS-based strategies could be useful to analyze a large panel of candidate genes in order to identify BC-predisposing mutations in high-risk families.

In our recent and current work, such procedures imply manual operation in several of the phases, that require highly specific expertise in the domain of the analysis, thus are not parallelizable without increasing personnel units and adopting a strict labour partitioning schema. Further issues are related to the heterogeneous nature of the datasets deriving from the integration of genomic data and clinical information, and the volume of data and the complexity of processing to be performed on it. As such, the analysis procedure is not scalable, and there are several potential bottlenecks for an intensive and massively parallel application of the procedure, limiting the research possibilities and barring the way to more ambitious results. As a consequence, the search for genetic causes of human diseases is now aiming at developing novel, sensitive, accurate, and cost- and time-effective pipelines for molecular diagnostics, and at elucidating mechanisms involved in disease development in order to identify novel diagnostic, prognostic, and therapeutic markers [29]. Advanced bioinformatic techniques and tools are essential for the successful application of NGS technology, and despite recent efforts have been made in the attempt to answer this need [18], it remains an open issue. Moreover, while several tool sets exist to perform biomedical research (we refer to Sect.II for a list of major ones), they are not integrated and engineered to provide a complete solution (automated analysis pipeline) to get from NGS data to information directly useful for diagnostics of BC. The most advanced pipelines, such as the one presented in [8], are often at the stage of conceptual frameworks, including many non-automated steps with significant expert judgment implied. This is detrimental both to the repeatability of the experiments and to their reproducibility.

In this paper, we present part of the fully-automated pipeline we have designed, and the experimental evaluation of its resource usage. Also, we present an use case in which the features obtained thanks to the proposed pipeline are used to improve the molecular diagnosis of hereditary BC through the **automated** analysis, visualization, and storage of NGS-derived genomic data from HBOC patients and their correlation with clinical phenotype and other diagnostic parameters. This will allow to identify the clinically-relevant information that can impact patients risk of developing cancers and that require specific preventive interventions.

II. Background

A. Molecular research in BC

The effectiveness and the importance of NGS-based approach for BC diagnostics is testified by a plethora of publications on this specific topic. We refer to [30] for a review on how the analysis of genome sequencing data contributes to BC classification (and thus definition of treatment) by revealing insight into tumor heterogeneity. Other works (such as [34], focused on microRNAs sequencing) have applied automated annotation and statistical analysis to the sequence and microRNAs quantification, aiming to the search of noninvasive markers for BC detection. Several statistical techniques have been proposed to support automatic BC diagnostics: a recent work [16] addresses the impact of three different methods (Fisher's Discriminant Ratio, two-tailed T-Test, and vector norm) on the identification of genes potentially responsible for BC, based on expression data.

B. Bioinformatic tools

The available bioinformatic tools for genomic research are currently highly fragmented in toolsets, with many alternative software components to perform the same type of processing (or sequences of processing), each with specific pro and cons. Integrated analysis environments have been proposed aiming at easily accessing cloud computing services, such as [4], specialized for metagenomic. Even in the case of integrated High-Performance-Computing suites, designed for easy-of-use on Cloud deployments, such as *Cloud BioLinux* [15] and Galaxy [12], they do not provide single pipelines, but just present a basic configuration of a wide set of tools and an interface to deploy and execute them on cloud IaaS, some by means of a command line, some by a web interface. We refer to [9], [5] for a review of such suites.

In order to build an analysis pipeline several tools are needed, each devoted to a very specific step or group of steps. For each of these steps, there are several software tools equivalent or overlapping in functionality. In the following we describe briefly the most known ones, among which we selected the components of the pipeline presented in this paper, grouping them according to the processing step they are devoted to.

1) Alignment: This step is the first in the analysis pipeline. It takes as input a file containing strings of bases, resulting from the sequencing process, **often represented in the FASTQ** file format. The Alignment process associates each string to a position in the reference genome. The output of this process is a mapping, represented in SAM (sequence alignment/map) or BAM (binary alignment/map)format. A survey of tools to perform this process is compared in [19], including *Maq, Bowtie, SSAHA2, BWA, SOAP2*, characterized by their compatibility with the sequencing methods (due to supported readings length), and their computation speed (ranging from ≈ 0.2 to ≈ 7 Gbp per CPU day). We have chosen BWA [17] as alignment tool because it offers both the highest speed and the widest support of sequencing platforms (Illumina, SOLiD, and 454-Roche).

2) Variant Calling: Variant calling is the identification, at single base level, of each nucleotide present in the sequenced reads respect to the genome sequence used as reference. It allows the identification of all the positions in the tested DNA sample that differ from the reference. Therefore, a high accuracy is mandatory to avoid uncurrect variants calling due to tecnical limitations. This step takes as input the set of the aligned sequences and compares them with known sequences (from the same reference genome used for alignmment) to detect differences (variants). The output of this process is a list of calls, whose format is standardized as VCF (Variant Call File).

Several tools are available to perform this process, the most popular and publicly available ones (namely: GATK Unified Genotyper, VarScan, Pindel, SAMtools, Dindel, GATK HaplotypeCaller, and Platypus) have been recently analyzed and compared in [11]. The comparison criteria include computing time, number of indels called, and classification accuracy metrics (against "gold standard" data). Platypus and HaplotypeCaller outperform other tools in most of the aspects, and we have adopted HaplotypeCaller (included in the GATK set of tools).

3) Annotation: The last step allows **both variants filtering and annotation**. Variants filtering means the ability to obtain a small set of variants starting from thousands of variants. This procedure could involve, for example in our case study, the comparison between normal and cancer data taking into account the pattern of inheritance of the disease. Annotation is the ability to associate to each variant specific information (features) that characterize it and help in the assignment of a biological significance (if present). This step requires a VCF file from step 2 as input and produce a novel VCF file including a new section with the annotation features. An analysis and comparison of tools for variants annotation has been presented in [24], including ANNOVAR, AnnTools, NGS-SNP, SwattleSeq, snpEff, SVA, VARIANT, and VEP). Common capabilities are: (i) reporting of a set of attributes for the identified mutations, that help assessing the potential impact of the mutation; (ii) linking to one or more public databases of known mutations, referencing known studies on the specific variant. Of the available tools we have chosen for our pipeline snpEff, that supports annotations for INDELs (INsertions/DELetions) and multiple-nucleotide polymorphisms in addition to single-nucleotide polymorphisms (supported by all tools). Moreover, snpEff classifies the effect of variant according to the functional impact (high, moderate, low, and modifier).

III. Methodology

A. Materials and methods

1) Patients enrollment and data gathering: Blood samples and clinical data have been obtained from women attending the Breast Unit of the "Istituto Nazionale dei Tumori - Fondazione G. Pascale" of Naples. All participants have been fully informed about the study and provided written informed consent prior to samples collection. All patients have been clinically approached: for everyone extensive family information has been collected in order to verify the familial risk. In particular, the patients enrolled for this gene-panel screening must have, in addition to a strong familial history of cancers, at least one of the following specific selection criteria: (i) BRCA1/2 negative mutation status (a part of a small subset of BRCA mutation carriers will be specified in the following); (ii) young age of BC onset (< 40 years); (iii) invasive and/or bilateral BC (any age) and/or multiple organ cancers; (iv) invasive ovarian cancer (any age). All the selected patients have been previously screened for BRCA1/2 mutations in our lab during routine diagnostic flow [6]. A subset of BRCA mutation carriers have been selected as positive controls for methodology feasibility assessment. In addition to personal and familial cancer history, to evaluate factors that could influence BC risk, the following information have been also collected for each patient: the geographic area of birth and residence; the geographic area of work and the kind of work; the kind of delivery; the kind of feeding; the diet habits (i.e. vegan or vegetarian); presence of food allergies; practice of physical activity; assumption of drugs or probiotics; obesity and/or familiarity for obesity, smoke, oral contraceptive use, pregnancies and abortions.

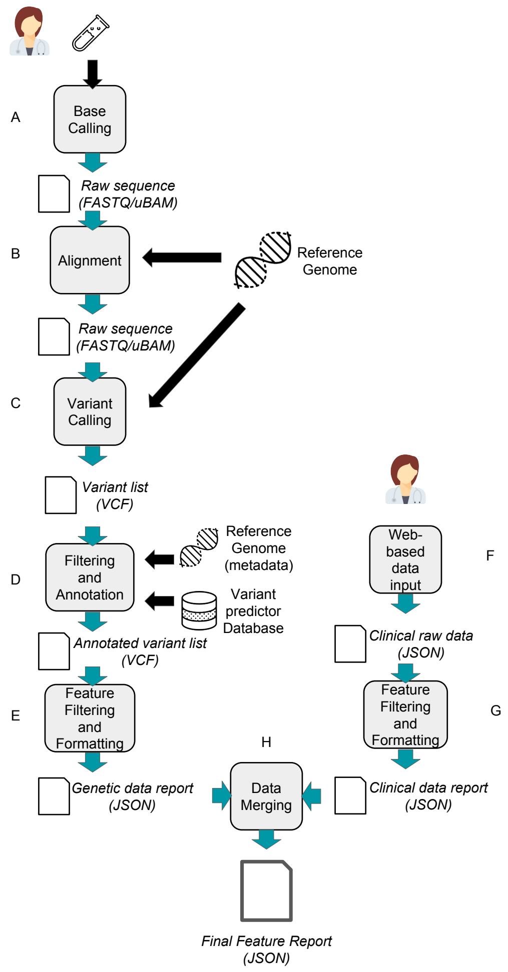

A
Base Calling
Raw sequence (FASTQ/uBAM)
B
Alignment
Reference Genome
Raw sequence (FASTQ/uBAM)
C
Variant Calling
Variant list (VCF)
D
Filtering and Annotation
Reference Genome (metadata)
Variant predictor Database
Annotated variant list (VCF)
E
Feature Filtering and Formatting
Genetic data report (JSON)
F
Web-based data input
Clinical raw data (JSON)
G
Feature Filtering and Formatting
Clinical data report (JSON)
H
Data Merging
Final Feature Report (JSON)

Fig. 1: Pipeline designed for the extraction of the features for the diagnosis of inherited BC. Two main branches can be identified that implement (i) extraction of the genetic features (left branch) and (ii) extraction of the clinical features (right branch).

2) Cancer-related gene panel screening: A custom panel of 84 cancer-related genes, including the entire coding regions, 100 bp in the intronic boundaries, the promoters and the 3' UTRs of each selected gene, has been used to analyze the BC patients enrolled for this study. This cancer-related gene panel, already present in the host laboratory, is "in-house" designed and has been validated for its analytic proficiency by analyzing 24 samples. It contains about 2,300 primer pairs that allow the simultaneous enrichment of 1,032,813 (the total target) for each patient. The genes enclosed in this panel are principally tumor suppressor genes, oncogenes, cell-cycle regulators, DNA repair sensors and effectors. BRCA1/2 have been also included. A library has been obtained for each DNA sample. Briefly, each genomic DNA has been sheared into small fragments (average size=500 bp) and specific adaptors, required for the following amplification and sequencing reactions, have been ligated to the end of each fragment. During this step we also added a specific barcode sequence, univocally assigned to each DNA sample, to allow downstream samples multiplexing. Then, the adapted fragments have been hybridized to the capture probes. At the end of the hybridization reaction, the enriched DNA fragments have been recovered and amplified to obtain an enriched library/sample. Libraries quality (Agilent 2100 BioAnalyzer) and quantity (picogreen assay) have been carefully evaluated before to proceed to the next steps. Equimolar amounts of several libraries have been pooled before sequencing. Sequencing reactions have been carried out using the MiSeq system (Illumina).

B. Pipeline design and implementation

In this section we detail the overall pipeline implemented for the diagnosis of inherited BC. For each step in the pipeline, we describe the inputs and the outputs together with their formats. As shown in Fig. 1, two main branches can be easily identified, namely the *genetic feature extraction* and the *clinical feature extraction* procedures. These two branches merge together at the step H, where their partial results are combined.

Regarding the branch related to *genetic feature extraction* we refer to section II-B for a description of the different phases and alternate tools that implement them. The genetic feature extraction procedure starts from the **DNA sequencing** (step A) that produces the raw sequence data (either in FASTQ or uBAM format). This data is then processed for the **alignment** to a reference genome (step B). This step also involves some data cleanup operations needed to make the data suitable for analysis and implemented for correcting potential biases due to technical issues. This step produces analysis-ready SAM/BAM (hereafter simply BAM) files. To implement this step, the pipeline adopts the `BWA` tool. The result of the reference genome alignment feeds the **variant calling** procedure (step C). This procedure is aimed at identifying the variants by comparing the results of the previous step to the reference genome. In more details the variant calling procedure is composed of a number of sub-steps described in the following. First, a sequence dictionary (step C.i) and an index are created from the reference genome (step C.ii). These steps provide optimized data structures required to the following analyses and allow to improve their performance by enabling efficient random access to arbitrary regions within the reference sequence. The BAM file is then reordered (step C.iii) and information on Read Groups (i.e. a set of reads that were generated from a single run of a sequencing instrument [1]) is added or replaced if needed (step C.iv). In detail, the implemented pipeline leverages `ReorderSam` and `AddOrReplaceReadGroups` facilities made available by `picard` [3] for this last procedure. Finally, the `HaplotypeCaller` tool, belonging to the `GATK` suite [11], is run to extract the variants—Single Nucleotide Polymorphisms (SNPs)—from the BAM file (step C.v). `HaplotypeCaller` is a cutting edge solution for SNPs and indels calling and comprises several advanced functionalities such as the reference confidence model (which enables efficient and incremental variant discovery on large cohorts) and special settings for SNPs and indel calling on RNAseq data. At the end of the variant calling step, a file containing all the identified variants is produced. It is formatted according the Variant Calling Format (VCF) standard. This file reporting the detected variants is then passed to the **filtering and annotation** block (step D). This block is in charge of annotating the variants, also predicting their effects on genes (such as amino acid changes). To reach this goal, at this step the interaction with a number of databases is required. These databases are conveniently queried depending on considered species (homo sapiens in this study) and reference genome releases (e.g. GRCh38/hg38, GRCh37/hg19, etc.) [2]. The output of this process is an annotated VCF file that matches each variant to its effect (e.g., variation of the chromosome number, exon loss, etc.) and putative impact (such as low, moderate, or high). The variants are ordered by deleteriousness. This step is primarily implemented through the `SnpEff` suite, whose output is also integrated with the information available in other databases (e.g., dbSNP, ClinVar, etc.) to enhance the available information base and add clinical information. At step E a number of **filtering operations** can be applied, such to properly reduce the information extracted and provide the right level of detail needed for the analysis, also according to the type of the successive procedure to be enforced (e.g., manual or automated). The final result of this branch is a JSON-encoded report of extracted genetic information.

For what concern the branch related to the extraction of the clinical features, the pipeline leads the human operator to obtain the vector of clinical features from the information associated to each of the patients. To this aim, the pipeline makes available a **web-based interactive form** that eases the work of the user supporting her during the filling of the expected data fields (step F). This web-based form is though to support the operator in filling the data fields both when gathering information during patient surveys and when updating it offline. In addition, this module also checks and validates the information inserted by the operator and encodes the possible values. The output of this step is a JSON-encoded file containing the raw clinical data. This clinical information is then **filtered and formatted** in order to extract the clinical features (step G).

At step H these clinical and the genetic features are merged to generate the **final feature report**, containing the overall view required that can be either used by the human operator to formulate the diagnosis or processed by artificial intelligence algorithms to automatically extract new knowledge from it. An example for the final report is shown in Table I.

C. Post-pipeline procedure

When the tasks of the pipeline have been processed, molecular biologists can carefully evaluate the content of the final report. This allows to identify one or a small set of variants probably related to the disease of interest and responsible for, or contributing to patients clinical phenotype. This applies also to the case study discussed herein. Usually, to identify a BC predisposing mutation starting from a list of hundreds of high quality annotated variants, some filters can be applied to the final report table to prioritize and highlight those variants most likely to be pathogenetic. First, it is possible to look for coding or splicing affecting variants. Among these, the variants with high impact and low frequency in the general population can be filtered. If a pathogenetic effect has been already reported (also with corresponding matches in the literature), the variant with these features could be responsible for the clinical phenotype (e.g., variant #3 listed in Tab. I). The subsequent correlation both with clinical data and personal and familial history is crucial to relate the genomic data to patients phenotype.

Based on the considerations reported above, we are considering to extend the pipeline integrating a Decision Support System (DSS) such to also automate the diagnostic phase.

IV. Discussion

Implementing the CArDIGAN approach, a number of advantages are achieved. The automation of the pipeline provides benefits that are both functional and related to the performance. In this section, we first describe the functional benefits the

TABLE I: A snippet of the final data report.

Variant	ID	chr	pos	Reference Allele	Variant Allele	Impact	Coding Effect	Frequency	Patogenicity	Already seen in
#1	BRCA2	chr13	32915005	G	C	LOW	synonymous_variant	0.2564/1834	NO	-
#2	BRIP1	chr17	59763465	T	C	LOW	synonymous_variant	0.4831/1834	NO	-
#3	BARD1	chr2	215632255	C	T	HIGH	missense_variant	0.0356/1563	YES	Vahteristo P et al.
#4	MSH6	chr2	48018081	A	G	LOW	synonymous_variant	0.3662/1834	NO	-
#5	PMS2	chr7	6026988	G	A	MODERATE	missense_variant	0.467/1380	NO	-

pipeline guarantees (Sec. IV-A); we then profile its performance, also discussing the related implications (Sec. IV-B).

A. Functional assessment

First, the complete automation allows to deal with input data of larger orders of magnitude: thousands of patients can be analyzed considering hundreds of heterogeneous features extracted from either the sequencing of tens of genes or rich clinical information. This approach is therefore able to guarantee a broader picture over the BC, easing the the diagnosis and potentially providing additional useful hints.

Secondly, the aspects the approach focuses on are related to different facets, thus leading to features with different semantics and allowing to disclose relations among factors that would not stem out when common procedures are adopted.

Finally, the automation of the process guarantees the complete repeatability of the overall process—such to foster further experimental replications—and sensitively reduces the probability of introducing human errors into the processing chain. Moreover, where a set of choices is available (e.g., specific parameters needed to customize a step of the analysis) the proposed approach allows to follow multiple paths in parallel, thus relieving the operator from the burden of prematurely taking any decision.

B. Performance assessment

All the above functional advantages being given, in the following we provide an assessment of the performance of the pipeline we have implemented. This operation is useful for profiling the automated process from the computational angle and possibly identifying any performance bottleneck to the elaboration.

To this aim, an experimental campaign has been performed as detailed in the following. We are interested in analyzing the performance of the pipeline from step C.i to step D that are supposed to be the most critical. The evaluation was run leveraging a machine with the characteristics reported in Tab. IIa. Three different BAM files were randomly selected based on the their size and were considered as inputs, as reported in Tab. IIb. These input files having different sizes allowed us to also evaluate the impact of the dimension of the input onto the performance.

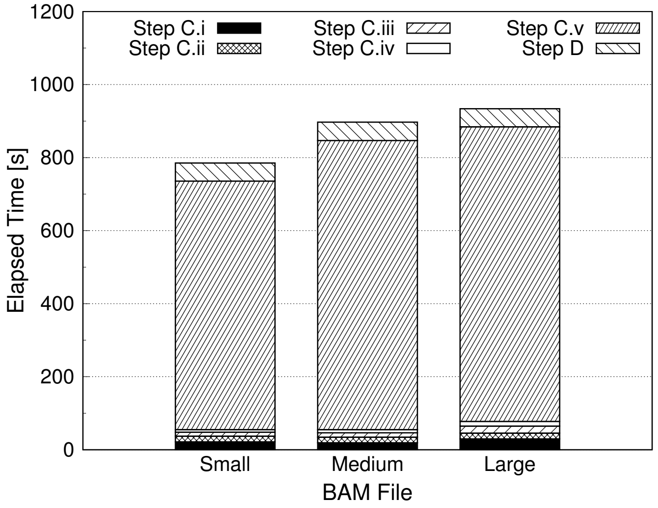


Fig. 2: Execution time for each step on varying BAM files. The execution time required by steps C.iii, C.iv, and C.v is impacted by the size of the BAM file provided as input.

Fig. 2 reports the execution time in seconds for each step taken into account on varying inputs. The results show how the greater the size of the input is, the longer the pipeline takes for accomplishing all the tasks. In particular a $+115\%$ variation of the sizes of the inputs in terms of reads (from Small to Large) leads to an increase in the processing time of 20% (from 785 s to 935 s). For some of the steps (e.g., C.i and C.ii), no major discrepancy in terms of elaboration time was observed. Otherwise, the execution time for steps C.iii, C.iv, and C.v is dramatically impacted by the size of the input.

Fig. 3 shows how computing resource overhead varies at different steps. We considered both CPU and memory usage as parameters of interest. No more than computational capability equivalent to 2.5 CPUs is required, on average. In Fig. 3a is shown how CPU usage varies across different steps. As reported, step C.iii (245%) is the one generating the major overhead for the CPU. No major discrepancy is generated when analyzing BAM files of different sizes in most of the cases. Results for steps C.i and C.iv changed for different

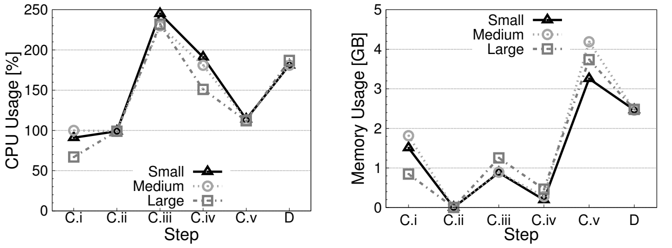
TABLE II: Experimental evaluation details.

(a) Hardware characteristics.

CPU	Intel(R) Core(TM) i7-4710MQ CPU @ 2.50GHz
Memory	12GB
Disk	Buffered disk reads: 107.01 MB/sec; cached reads:11183.27 MB/sec

(b) BAM files.

id	#Reads	#Alignements	Dimension (MB)
Small	577034	553823	33.8
Medium	752880	721918	40.8
Large	1240777	1191643	64.9



(a) CPU

(b) Memory

Fig. 3: Resource overhead in terms of CPU and memory usage at different steps.

inputs, although no dependence from the size is observed. Notably, in these cases Large BAM files show the lowest CPU overhead. For what concerns the memory usage, the most burdensome step is C.v where more than 4GB are required for the variant extraction of the Large BAM file.

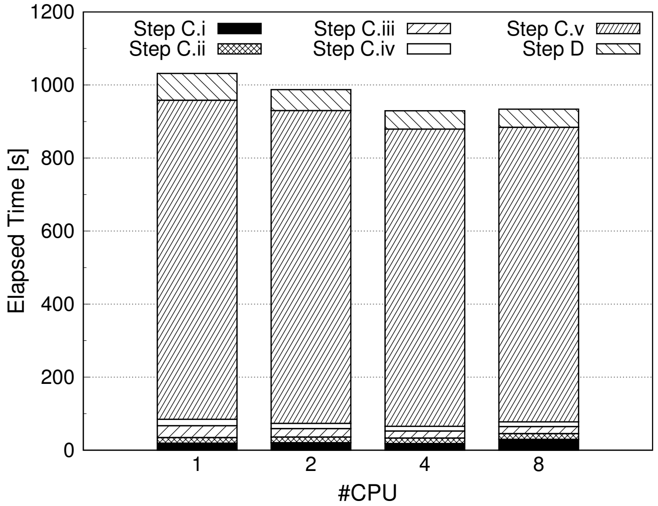


Fig. 4: Execution time for each step when varying the number of cores. No performance enhancement has been observed when running the pipeline on more than 4 cores.

Finally, we investigated how the current implementation of the pipeline is able to benefit from multi-core (or multi-processor) architectures. To this aim, when executing the pipeline we forced the tools at each step to run over a limited number of cores (i.e. 1, 2, 4, and 8). As shown in Fig. 4, better performance in terms of execution time is achieved when executing the pipeline on CPU architectures leveraging multiple CPUs. When moving from 1 to 8 cores the time needed to execute the tasks composing the pipeline is reduced by 10%. However, there is no evidence of performance enhancement when running the pipeline on more than 4 cores. In conclusion, leveraging multi-core architectures with up to 4 cores leads to a clear performance improvement. This result is due to the limited number of steps, whose performance is impacted by the number of cores available.

The above analysis is also dictated by the need to migrate some portions of the proposed architecture onto the cloud according to the CArDIGAN view. Indeed, the results of the analysis provide an overall characterization of the computing effort required by the pipeline, that is of the utmost importance to properly configure the cloud environment in terms of leased resources (i.e. CPU, memory, disk, etc.) such to obtain the desired performance level and reduce leasing costs [25], [26], [27].

V. Conclusion

Motivated by the wide availability of NGS-based methods and tools, we have investigated their application to improve Breast Cancer molecular diagnostics, and found that there is a lack of a solution that is fully automated. This has impact on the extensiveness of analysis that can be performed in a given time, and on the reproducibility and repeatability of the process, that in our previous and current research activities still involves manual intervention and expert judgment. In this paper we have presented part of a software pipeline we have designed and are implementing, specifically aimed at assisting in BC diagnostics. After an overview of the molecular diagnostic process and the involved phases and tools, we have described our pipeline implementation, and we have validated it using real data (both genetic and clinical). The functional assessment of the pipeline has shown its usefulness in speeding up the work of the researchers while providing at the same time more structured information, compared with the previous (operator-intensive and partially subjective) procedure. Moreover we have conducted a performance assessment of the pipeline, detailing the contribution of each component while varying key operational parameters, namely: the dimension of the genetic dataset that is fed to the pipeline, and the number of CPUs dedicated to the processing. The resource usage has been assessed in terms of CPU percentage, RAM occupation, and time to complete the task. These assessment parameters inform on the hardware and service requirements to run the pipeline, either on-premises or as Cloud instances. The results show that the pipeline is effective in assisting and enhancing BC diagnostic and encourage towards further automation.

Ongoing and future work is aimed at further expanding the pipeline adding automation of the current post-pipeline analysis, and expanded feature extraction and analysis tasks. Moreover the deployment on cloud services will be assessed and automated for the parallelizable and computing-intensive phases of the pipeline.

References

[1] GATK - Genome Analysis Toolkit: Variant Discovery in High-Throughput Sequencing Data. https://software.broadinstitute.org/gatk/, Sept. 2016.

[2] NCBI - National Center for Biotechnology Information. http://www.ncbi.nlm.nih.gov/, Sept. 2016.

[3] Picard: A set of command line tools (in Java) for manipulating high-throughput sequencing (HTS) data and formats such as SAM/BAM/CRAM and VCF. https://broadinstitute.github.io/picard/, Sept. 2016.

[4] S. V. Angiuoli, M. Matalka, A. Gussman, K. Galens, M. Vangala, D. R. Riley, C. Arze, J. R. White, O. White, and W. F. Fricke. CloVR: A virtual machine for automated and portable sequence analysis from the desktop using cloud computing. *BMC Bioinformatics*, 12(1):356, 2011.

[5] P. C. Church and A. Goscinski. A survey of approaches and frameworks to carry out genomic data analysis on the cloud. In *Cluster, Cloud and Grid Computing (CCGrid), 2014 14th IEEE/ACM International Symposium on*, pages 701–710. IEEE, 2014.

[6] V. D'Argenio, M. V. Esposito, A. Telese, V. Precone, F. Starnone, M. Nunziato, P. Cantiello, M. Iorio, E. Evangelista, M. D'Aiuto, et al. The molecular analysis of brca1 and brca2: Next-generation sequencing supersedes conventional approaches. *Clinica Chimica Acta*, 446:221–225, 2015.

[7] V. D'Argenio, G. Frisso, V. Precone, A. Boccia, A. Fienga, G. Pacileo, G. Limongelli, G. Paolella, R. Calabrò, and F. Salvatore. Dna sequence capture and next-generation sequencing for the molecular diagnosis of genetic cardiomyopathies. *The Journal of Molecular Diagnostics*, 16(1):32–44, 2014.

[8] M. A. DePristo, E. Banks, R. Poplin, K. V. Garimella, J. R. Maguire, C. Hartl, A. A. Philippakis, G. del Angel, M. A. Rivas, M. Hanna, A. McKenna, T. J. Fennell, A. M. Kernytsky, A. Y. Sivachenko, K. Cibulskis, S. B. Gabriel, D. Altshuler, and M. J. Daly. A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics*, 43(5):491–498, apr 2011.

[9] M. P. Dolled-Filhart, M. Lee, C. wen Ou-yang, R. R. Haraksingh, and J. C.-H. Lin. Computational and bioinformatics frameworks for next-generation whole exome and genome sequencing. *The Scientific World Journal*, 2013:1–10, 2013.

[10] R. R. Gullapalli, K. V. Desai, L. Santana-Santos, J. A. Kant, M. J. Becich, et al. Next generation sequencing in clinical medicine: Challenges and lessons for pathology and biomedical informatics. *Journal of pathology informatics*, 3(1):40, 2012.

[11] M. S. Hasan, X. Wu, and L. Zhang. Performance evaluation of indel calling tools using real short-read data. *Human genomics*, 9(1):1, 2015.

[12] J. Hillman-Jackson, D. Clements, D. Blankenberg, J. Taylor, A. Nekrutenko, and G. Team. Using galaxy to perform large-scale interactive data analyses. *Current protocols in bioinformatics*, pages 10–5, 2012.

[13] S. Jauhari and S. A. M. Rizvi. Mining gene expression data focusing cancer therapeutics: A digest. *IEEE/ACM Trans. Comput. Biol. and Bioinf.*, 11(3):533–547, may 2014.

[14] M. Kim, K.-H. Lee, S.-W. Yoon, B.-S. Kim, J. Chun, and H. Yi. Analytical tools and databases for metagenomics in the next-generation sequencing era. *Genomics Inform*, 11(3):102, 2013.

[15] K. Krampis, T. Booth, B. Chapman, B. Tiwari, M. Bicak, D. Field, and K. E. Nelson. Cloud BioLinux: pre-configured and on-demand bioinformatics computing for the genomics community. *BMC Bioinformatics*, 13(1):42, 2012.

[16] G. Kumar, T. Lahiri, and R. Kumar. Statistical discrimination of breast cancer microarray data. In *2016 International Conference on Bioinformatics and Systems Biology (BSB)*. Institute of Electrical and Electronics Engineers (IEEE), mar 2016.

[17] H. Li and R. Durbin. Fast and accurate long-read alignment with burrows–wheeler transform. *Bioinformatics*, 26(5):589–595, 2010.

[18] J. Luo, M. Wu, D. Gopukumar, and Y. Zhao. Big data application in biomedical research and health care: A literature review. *Biomedical informatics insights*, 8:1, 2016.

[19] A. Magi, M. Benelli, A. Gozzini, F. Girolami, F. Torricelli, and M. L. Brandi. Bioinformatics for next generation sequencing data. *Genes*, 1(2):294–307, sep 2010.

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

[21] L. Melchor and J. Benítez. The complex genetic landscape of familial breast cancer. *Human genetics*, 132(8):845–863, 2013.

[22] S. A. Narod. Breast cancer in young women. *Nature reviews Clinical oncology*, 9(8):460–470, 2012.

[23] A. Nekrutenko and J. Taylor. Next-generation sequencing data interpretation: enhancing reproducibility and accessibility. *Nature Reviews Genetics*, 13(9):667–672, 2012.

[24] S. Pabinger, A. Dander, M. Fischer, R. Snajder, M. Sperk, M. Efremova, B. Krabichler, M. R. Speicher, J. Zschocke, and Z. Trajanoski. A survey of tools for variant analysis of next-generation genome sequencing data. *Briefings in bioinformatics*, 15(2):256–278, 2014.

[25] V. Persico, A. Botta, A. Montieri, and A. Pescapé. A first look at public-cloud inter-datacenter network performance. In *2016 IEEE Global Communications Conference: Communication QoS, Reliability and Modeling (Globecom2016 CQRM)*, Washington, USA, Dec. 2016.

[26] V. Persico, P. Marchetta, A. Botta, and A. Pescapé. On network throughput variability in microsoft azure cloud. In *2015 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6. IEEE, 2015.

[27] V. Persico, A. Montieri, and A. Pescapé. CloudSurf: a platform for monitoring public-cloud networks. In *2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI) (IEEE RTSI 2016)*, Bologna, Italy, Sept. 2016.

[28] V. Precone, V. Del Monaco, M. V. Esposito, F. D. E. De Palma, A. Ruocco, F. Salvatore, and V. D'Argenio. Cracking the code of human diseases using next-generation sequencing: Applications, challenges, and perspectives. *BioMed research international*, 2015, 2015.

[29] J. A. Reuter, D. V. Spacek, and M. P. Snyder. High-throughput sequencing technologies. *Molecular cell*, 58(4):586–597, 2015.

[30] H. G. Russnes, N. Navin, J. Hicks, and A.-L. Borresen-Dale. Insight into the heterogeneity of breast cancer through next-generation sequencing. *The Journal of clinical investigation*, 121(10):3810–3818, 2011.

[31] M. H. Skolnick et al. A strong candidate for the breast and ovarian cancer susceptibility gene brca1. *Science*, 266(5172):66–71, 1994.

[32] A. von Bubnoff. Next-generation sequencing: the race is on. *Cell*, 132(5):721–723, 2008.

[33] R. Wooster, G. Bignell, J. Lancaster, S. Swift, S. Seal, J. Mangion, N. Collins, S. Gregory, C. Gumbs, G. Micklem, et al. Identification of the breast cancer susceptibility gene brca2. *Nature*, 378(6559):789–792, 1995.

[34] Q. Wu, Z. Lu, H. Li, J. Lu, L. Guo, and Q. Ge. Next-generation sequencing of micrornas for breast cancer detection. *BioMed Research International*, 2011, 2011.