

# Sarav--Automated and Scalable pipeline for WES data analysis

*Thesis submitted to the SASTRA Deemed to be University  
in partial fulfilment of the requirement  
for the award of the degree of*

**M.Sc. Bioinformatics**

*Submitted by*

**NAME: BHARATH S**  
**(Reg. No.: 124121002)**

**June 2024**

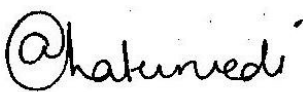



**SCHOOL OF CHEMICAL AND BIOTECHNOLOGY THANJAVUR,  
TAMIL NADU, INDIA – 613 401**



### **Bonafide Certificate**

This certifies that the thesis titled '**Sarav: An Automated and Scalable Pipeline for WES Data Analysis**' submitted for the M.Sc. Bioinformatics degree at SASTRA Deemed to be University is a genuine representation of the work conducted by Mr. Bharath S (Registration No: 124121002) during the final semester of the academic year 2023-24. This work was carried out at Lifecell International Pvt Ltd under my supervision. I confirm that the thesis meets the required standards and complies with the rules and regulations pertaining to the internship.

**Signature of Project Supervisor** :  

**Name with Affiliation** : **Mr. Ankur Chaturvedi** (Senior Manager, Molecular Genetics)

**Date** : 11.06.2024

**Project Viva voce held on** : 12.06.2024

**Examiner 1**

**Examiner 2**

**Examiner 3**



# SASTRA

ENGINEERING · MANAGEMENT · LAW · SCIENCES · HUMANITIES · EDUCATION

DEEMED TO BE UNIVERSITY

(U/S 3 of the UGC Act, 1956)



THINK MERIT | THINK TRANSPARENCY | THINK SASTRA

**SCHOOL OF CHEMICAL AND BIOTECHNOLOGY  
THANJAVUR – 613 401**

**Declaration**

I declare that the thesis titled “**Sarav--Automated and Scalable pipeline for WES data analysis**” submitted by me is an original work done by me under the guidance of **Mr. Ankur Chaturvedi, Senior Manager Molecular Genetics at LifeCell International Pvt Ltd, Chennai** during the final semester of the academic year 2023-24, in the **School of Chemical and Biotechnology**. The work is original and wherever I have used materials from other sources, I have given due credit and cited them in the text of the thesis. This thesis has not formed the basis for the award of any degree, diploma, associate-ship, fellowship or other similar title to any candidate of any University.

**Signature of the candidate(s)** : 

**Name of the candidate(s)** : Bharath S

**Date** : 11.06.2024

## Acknowledgements

I am very thankful and want to express my gratitude to the Bioinformatics team at Lifecell International for giving me this opportunity as a Bioinformatics intern at Lifecell International Pvt. Ltd. I am particularly grateful to **Mr. Ankur Chaturvedi** (Molecular Genetics), my reporting guide **Mr. Krishna Vaibhav Tiwari**, **Mrs. Megha Varsheney**, and the entire Bioinformatics team at Lifecell International for helping me understand and perform NGS secondary data analysis, as well as utilize GATK (Genome Analysis Toolkit), Nextflow, Python, Docker, CONDA, Git, and Linux. The project was very interesting, and I gained a lot of knowledge.

I am also very thankful to **Dr. N.T. Saraswathi** for her support in allowing me to undertake projects outside my campus, which broadened my learning experience. Additionally, I would like to thank my minor project guide, **Dr. J Arunachalam**, Senior Assistant Professor, Department of Bioinformatics, SCBT, for his support. I express my gratitude to the review panel for their validation of my project and their invaluable suggestions that enriched my work.

# INDEX

Chapter	Title	Page No.
	List of Table	6
	List of Figures	7
	List of Abbreviations	8
	Abstract	9
01	Introduction	10
02	Objective	14
03	Results and Discussion	18
04	Validation	21
05	Methods	22
06	Setting up pipeline	26
07	Conclusion	28
08	References	30
09	Plagiarism report	34

## LIST OF TABLES

Table No	Title	Page No.
1	Data Sources used in this pipeline (hg38)	19
2	Metrics used by GATK for hard filtering germline variants before annotation.	23
3	Metrics used by GATK for hard filtering Somatic variants before annotation.	23
4	Summary of the tools used in each stage.	25

# LIST OF FIGURES

Table No.	Title	Page No.
1	WES sequencing.	10
2	GATK (Genome Analysis Toolkit)	12
3	Image explaining Illumina Sequencing technology.	13
4	Workflow used in this pipeline.	14
5	Pipeline explaining Germline	15
6	Pipeline explaining Somatic	16
7	Nextflow process explanation.	17
8	Image showing successful running completion of pipeline	18
9	Image showing successful running completion of pipeline	18
10	Docker image created to access the container	19
11	Screenshot showing docker container created	19
12	Overall architecture of the pipeline.	24
11	Private Github repository	27

## LIST OF ABBREVIATION

No.	Abbreviation	Full form
1	NGS	Next Generation Sequencing
2	WES	Whole Exome Sequencing
3	GATK	Genome Analysis ToolKit
4	DNA	Deoxy-ribonucleic acid
5	RNA	Ribo-nucleic acid
6	PCR	Polymerase Chain Reaction
7	dNTP	deoxynucleoside triphosphate
8	SBS	Sequencing-by-synthesis
9	SNP	Single Nucleotide Polymorphism
10	InDel	Insertion and Deletion
12	SV	Structural Variation
13	CNV	Copy Number Variation
14	BWA	Burrows-Wheeler Aligner
15	SAM	Sequence Alignment/Map
16	BAM	Binary Alignment/Map
17	VCF	Variant Call Format
18	BQSR	Base quality score recalibration
19	POSIX	Portable Operating System Interface
20	QD	Quality by Depth
21	MQ	Mapping Quality
22	FS	Fisher Strand
23	SOR	Strand Odds Ratio
24	DP	Depth of coverage
25	GQ	Genotype quality



## Abstract:

The advancement of high-throughput sequencing technologies has propelled genomic research into new frontiers, enabling the rapid generation of vast amounts of sequencing data. However, analyzing this data presents a significant computational challenge, requiring robust and efficient pipelines. In this study, our team presents an automated pipeline for whole exome data analysis, developed using Nextflow and based on the Genome Analysis Toolkit (GATK). The pipeline encompasses two main workflows: germline variant calling using the GATK HaplotypeCaller tool and somatic variant calling employing Mutect2 and VarScan. To enhance reproducibility and scalability, the pipeline was containerized using Docker, ensuring independence and ease of deployment across different computing environments. We have integrated GATK Funcotator into our pipeline for variant annotation, enhancing the interpretability and biological relevance of detected variants. This addition expands the utility of our pipeline by providing detailed functional annotations for genomic variants, facilitating downstream analysis and interpretation. Our pipeline streamlines the analysis process for WES data, achieving more than a 20% reduction in processing time. The germline workflow identifies genetic variations in the germline genome, while the somatic workflow detects mutations specific to tumor samples, providing valuable insights into cancer genetics and personalized medicine. This thesis presents the design, implementation, and validation of the automated pipeline, highlighting its efficiency, accuracy, and scalability. The pipeline's modular architecture allows for easy customization and adaptation to diverse research needs. Our findings demonstrate the utility of automated pipelines in accelerating genomic data analysis (Van der Auwera and O'Connor, 2020). For further details and access to the pipeline, please refer to our GitHub repository: <https://github.com/BharathSaravanann/Sarav-pipeline>.

## Tools used to create this pipeline



**Nextflow**



**Docker**



**GATK**



**CONDA**



**LINUX**



**GITHUB**

# CHAPTER 1

## 1.Introduction:

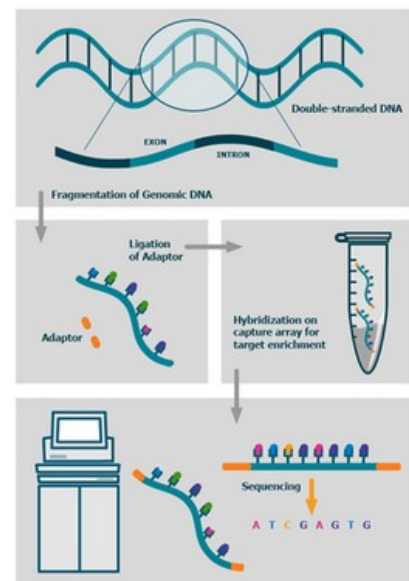
The rapid evolution of high-throughput sequencing technologies has transformed genomic research, enabling the generation of vast amounts of sequencing data at an unprecedented pace. However, along with this data abundance comes the formidable challenge of efficient analysis and interpretation, necessitating the development of robust bioinformatics pipelines. In this study, we introduce an automated pipeline tailored specifically for whole exome sequencing (WES) data analysis, leveraging Nextflow and the Genome Analysis Toolkit (GATK) to streamline and optimize the analytical process.(Di Tommaso et al., 2017).

Our pipeline focuses on enhancing efficiency and reducing processing time, with a particular emphasis on two core workflows: germline variant calling and somatic variant calling. These workflows are instrumental in deciphering genetic variations across diverse genomic contexts. Leveraging the capabilities of GATK, our pipeline incorporates the HaplotypeCaller tool for germline analysis, pinpointing genetic variations within the germline genome. For somatic analysis, we integrate Mutect2 and varscan a specializing in detecting mutations specific to tumor samples, thereby providing valuable insights into cancer genetics and personalized medicine. (McKenna et al., 2010).

To achieve reproducibility, scalability, and ease of deployment, we have meticulously containerized our pipeline using Docker. This strategic integration not only ensures independence and compatibility across various computing environments but also significantly streamlines the WES data analysis workflow, resulting in improved efficiency and accuracy (Merkel, 2014).

This thesis elucidates the design, implementation, and validation of our automated pipeline, highlighting its effectiveness in reducing processing time and enhancing automation in WES data analysis. By showcasing a 20% reduction in processing time compared to traditional methods, our pipeline demonstrates its efficacy and suitability for accelerating genomic data analysis without compromising accuracy. The modular architecture of the pipeline allows for

**Fig 1: WES sequencing**

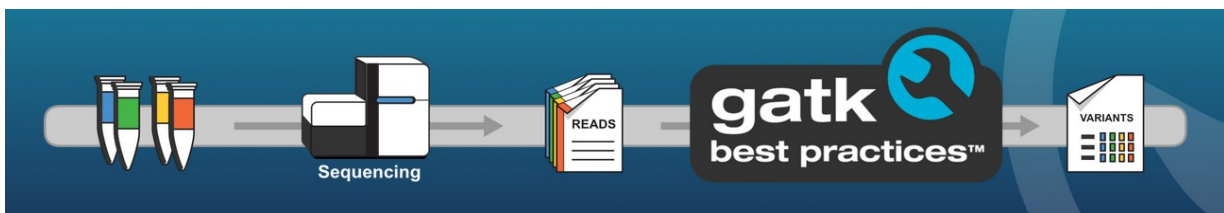


<https://sourcebioscience.com/genomics/ngs/whole-exome-sequencing/>

easy customization and adaptation, making it a versatile tool for researchers in genomics and related fields.

Our findings underscore the transformative impact of automated pipelines in genomic research, paving the way for faster, more efficient, and reproducible data analysis. While our focus is on optimizing processing time and automation, the broader implications of this work contribute to advancing precision medicine research and improving our understanding of complex genetic phenomena (Van der Auwera et al., 2013).

**Fig 2: GATK (Genome Analysis Toolkit)**



<https://gatk.broadinstitute.org/hc/en-us>

### **1.1 Next Generation Sequencing:**

High-throughput sequencing and massively parallel sequencing describe the DNA sequencing technology revolutionizing biological research. NGS, known for its ultra-high throughput, scalability, and speed, allows the sequencing of a human genome in a single day, compared to the more than a decade required by Sanger sequencing for the final human genome draft. Also called high-throughput sequencing, NGS is a DNA sequencing method that has transformed genomic and molecular research, impacting many biological study areas. It is increasingly prevalent in modern society, enabling precise, rapid, and cost-effective DNA and RNA sequencing (Mardis, 2008).

### **1.2 Illumina sequencing**

After acquiring Solexa in late 2006, Illumina joined the next-generation sequencing business. Illumina technology, as the most widely used NGS equipment, makes use of clonal array creation and reversible terminator technology for large-scale sequencing. Several samples can be put onto the eight-lane flow cell at the same time. Fragments are first annealed after being ligated to generic adaptors. Each template molecule may be replicated up to 1,000 times using solid-phase PCR amplification. They are then split into single strands in preparation for sequencing. Fluorescently labelled dNTPs (various colors correlate to different bases) are introduced to the nucleic acid chain throughout each cycle. Because the nucleotide label acts as a polymerization terminator, a picture of the fluorescent dye is obtained after each dNTP

inclusion. The enzymatic cleavage then allows the incorporation of the next base(Bentley et al., 2008).

### **1.3 The core principle of Illumina NGS**

The Illumina next-generation sequencing (NGS) approach utilizes sequencing-by-synthesis (SBS) and reversible dye-terminators, enabling the identification of single bases as they are incorporated into DNA strands (Bentley et al., 2008).

### **1.4 Illumina NGS applications:**

NGS can be applied to whole-genome sequencing, targeted area sequencing, transcriptome analysis, metagenomics, small RNA identification, methylation profiling, and genome-wide protein-nucleic acid interaction analysis, helping individuals unlock the potential of the genome (Mardis, 2008).

### **1.5 The workflow of Illumina NGS**

#### **Step 1. Library preparation**

Random fragmentation of the DNA or cDNA sample is followed by 5' and 3' adapter ligation to create the sequencing library. Alternatively, "tagmentation" combines the fragmentation and ligation processes into a single step, significantly increasing the efficiency of library preparation. Afterward, adapter-ligated fragments are PCR amplified and gel purified (Mardis, 2008).

#### **Step 2. Cluster generation**

The flow cell serves as a conduit for adsorbing mobile DNA fragments and as a core sequencing reactor vessel where all sequencing occurs. As DNA fragments from the sequencing library pass through the flow cell, they randomly attach to lanes on its surface. Each flow cell has eight lanes, and each lane contains numerous adapters that match the adapters added to the DNA fragments during library preparation. This enables the flow cell to adsorb the DNA and support bridge PCR amplification on the DNA's surface. There is no reciprocal interaction between these lanes. After continuous amplification and mutation cycles, each DNA fragment aggregates in bundles at their respective sites, each carrying many copies of a single DNA template. The goal of this technique is to amplify the base signal intensity to meet the sequencing signal requirements. Once the cluster creation process is complete, the templates are ready for sequencing (Bentley et al., 2008).

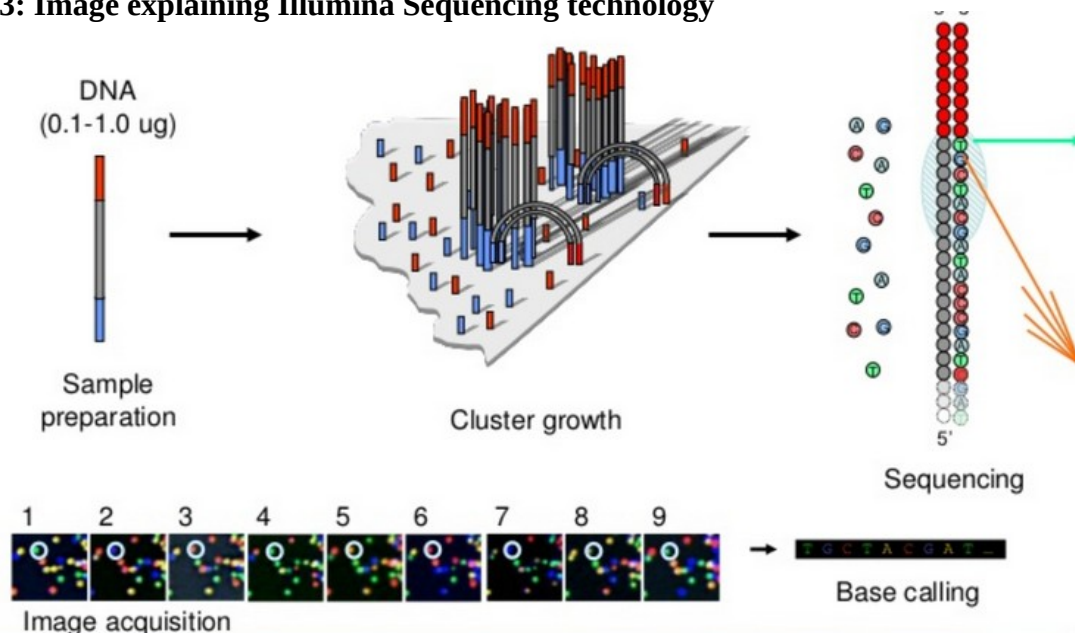
### Step 3. Sequencing

The sequencing process is based on sequencing-by-synthesis (SBS). The reaction system is supplemented with DNA polymerase, connector primers, and four dNTPs containing base-specific fluorescent markers. Chemical approaches protect the 3'-OH of these dNTPs, ensuring that only one base is incorporated at a time during the sequencing process. Once the synthesis reaction is complete, all unused free dNTPs and DNA polymerase are eluted. The buffer solution required for fluorescence excitation is then added, and the fluorescence signal is stimulated by a laser and recorded by optical equipment. Computer analysis converts the optical signal into a sequencing base. After recording the fluorescence signal, a chemical reagent is applied to quench the signal and remove the dNTP 3'-OH protecting group, allowing the next set of sequencing reactions to proceed (Bentley et al., 2008).

### Step 4. Alignment & Data analysis

After aligning the newly discovered sequence reads to a reference genome, various bioinformatics analyses become feasible, including SNP, InDel, SV, and CNV calling, annotation and statistics, pathway enrichment analysis, population genetics analysis, and more (Li & Durbin, 2010).

**Fig 3: Image explaining Illumina Sequencing technology**



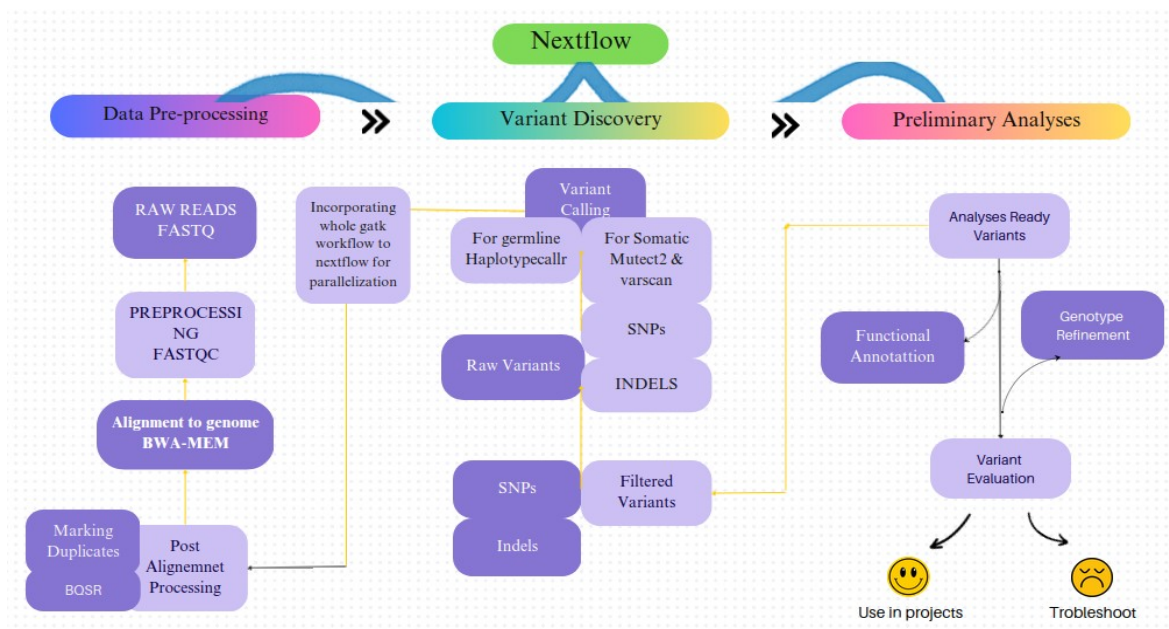
[https://www.slideshare.net/AGRF\\_Ltd/ngs-technologies-platforms-and-applications](https://www.slideshare.net/AGRF_Ltd/ngs-technologies-platforms-and-applications)

## CHAPTER 2

### 2. Objective:

- Create a robust and user-friendly pipeline for WES data analysis, incorporating best practices and advanced bioinformatics methodologies.
- Implement advanced variant calling algorithms and quality control processes to improve the accuracy and sensitivity of variant detection.
- Integrate variant annotation tools to provide comprehensive functional annotations, enabling researchers to derive biological insights from genomic data.
- Develop both germline and somatic variant calling pipelines to address hereditary and cancer-related mutations effectively.
- Utilize containerization and version control to ensure reproducibility and facilitate collaboration among researchers in the genomics community.

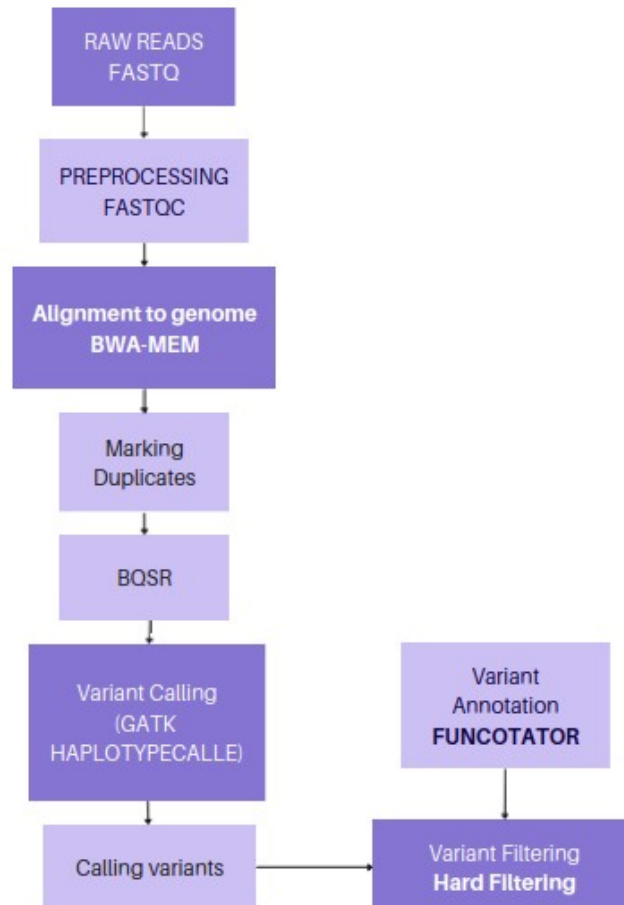
**Fig 4: Overall workflow used in this pipeline**



## 2.1 Germline Pipeline

The germline variant calling pipeline is an automated, end-to-end workflow designed for the analysis of whole exome sequencing (WES) data. Utilizing Nextflow and the Genome Analysis Toolkit (GATK), this pipeline ensures high accuracy and efficiency in identifying genetic variants. The process begins with aligning sequencing reads to a reference genome using BWA mem, followed by marking duplicates and performing base quality score recalibration (BQSR). Germline variants, including single nucleotide polymorphisms (SNPs) and insertions/deletions (indels), are called using GATK HaplotypeCaller. The pipeline then applies Variant Quality Score Recalibration (VQSR) to filter and score variants based on their quality. Finally, variants are annotated using GATK Funcotator, providing comprehensive functional annotations. The entire pipeline is containerized using Docker, ensuring reproducibility and scalability across different computing environments, making it an invaluable tool for genetic research and clinical diagnostics (Van der Auwera & O'Connor, 2020).

**Fig 5: Pipeline explaining germline**

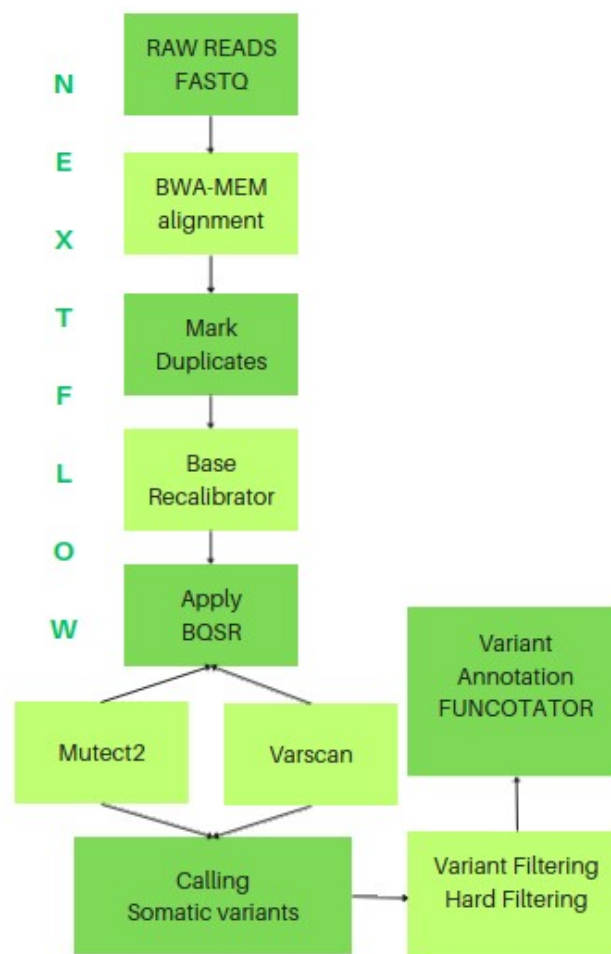




## 2.2 Somatic Pipeline

The somatic variant calling pipeline is tailored for the detection of cancer-related mutations within WES data. Similar to the germline pipeline, it employs Nextflow and GATK to streamline the analysis process. The pipeline starts with read alignment using BWA mem, followed by duplicate marking and BQSR. Somatic variants are identified using GATK Mutect2 and VarScan, which distinguishes between tumor and normal samples to accurately detect cancer-specific mutations. Post-variant calling, the pipeline includes a series of filtering steps to remove false positives, ensuring high-confidence somatic variant calls. The workflow is designed to exclude the use of external resources such as omni, hapmap, indelsites, and snpsites, focusing solely on tumor-normal paired analyses. The pipeline is also containerized with Docker, allowing for reproducible and scalable analysis. This robust framework supports comprehensive cancer genomics research, facilitating the discovery of novel mutations and aiding in the development of personalized cancer therapies (Cibulskis et al., 2013).

**Fig 6: Pipeline explaining somatic**

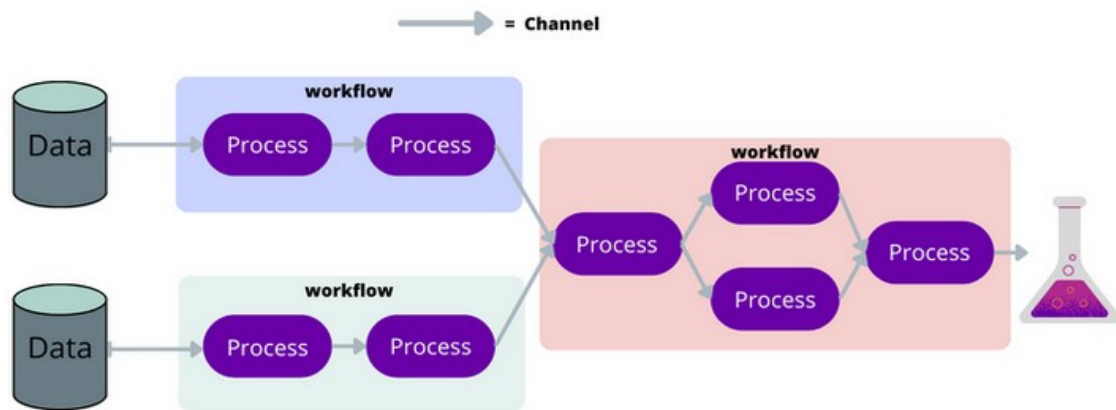




## 2.3 why we are using nextflow ?

Nextflow is chosen for genomic workflows due to its portable execution and reproducibility through containerization. It optimizes efficiency with automated scalability and parallel processing, ideal for large-scale genomic analyses. Its declarative scripting ensures accessibility and flexibility, allowing seamless integration of tools for adaptable workflows (Di Tommaso et al., 2017).

**Fig 7: Nextflow process explanation**



[https://carpentries-incubator.github.io/Pipeline\\_Training\\_with\\_Nextflow/02-Intro\\_to\\_Nextflow/index.html](https://carpentries-incubator.github.io/Pipeline_Training_with_Nextflow/02-Intro_to_Nextflow/index.html)

## 2.4 Why we are using Docker:

Docker is integral to our bioinformatics pipeline development due to its ability to containerize applications, ensuring consistency and reproducibility. By packaging the pipeline and its dependencies into a Docker container, we can guarantee that the software runs identically across different environments, eliminating compatibility issues. This is particularly crucial in bioinformatics, where pipelines often involve numerous tools and libraries that can vary significantly across systems. Docker provides a controlled environment, enhancing the reliability and reproducibility of our analyses. Additionally, Docker simplifies the deployment and sharing of our pipeline, enabling other researchers to easily replicate our work without complex setup procedures. This containerization also supports scalability, allowing the pipeline to efficiently handle varying computational demands, making Docker an essential tool for robust and reproducible bioinformatics research (Merkel, 2014).

## CHAPTER 3

### 3. Results:

Fig 8: Image showing successful running completion of pipeline

```
Launching `germlinehard.nf` [pensive_solway] DSL2 - revision: b748143ea6

=====
|               G E R M L I N E   V C               |
=====

genome reference : /home/docker/ref/hg38.fa
father FASTQ location : /home/docker/fastq/*_{R1,R2}*
output directory : /home/docker/output
known sites for BQSR : /home/docker/ref/Homo_sapiens_assembly38.dbsnp138.vcf
data for annotation : /home/docker/data_sources/funcotator_dataSources.v1.8.hg38.

=====
>               E X E C U T I N G               <
=====

executor > local (18)
[95/fd0e69] align (1) [100%] 1 of 1 ✓
[0e/c5f579] markDuplicates (1) [100%] 1 of 1 ✓
[ed/0fe221] insertmetrics (1) [100%] 1 of 1 ✓
[d5/fd6075] alignmentmetrics (1) [100%] 1 of 1 ✓
[cf/cbc695] baseRecalibrator (1) [100%] 1 of 1 ✓
[0c/d25f20] applyBQSR (1) [100%] 1 of 1 ✓
[bc/c9917c] haplotypeCaller (1) [100%] 1 of 1 ✓
[7d/5b8813] variantevaluation (1) [100%] 1 of 1 ✓
[17/a96663] snp (1) [100%] 1 of 1 ✓
[3e/61572f] indels (1) [100%] 1 of 1 ✓
[57/bc5cf7] filtersnp (1) [100%] 1 of 1 ✓
[66/452c60] filterindels (1) [100%] 1 of 1 ✓
[64/c6bdc6] passsnps (1) [100%] 1 of 1 ✓
[a1/de79a4] passindels (1) [100%] 1 of 1 ✓
[b7/0a35a7] failedgenotypesnp (1) [100%] 1 of 1 ✓
[d1/cc77c8] failedgenotypeindels (1) [100%] 1 of 1 ✓
[b2/4b35cc] snpannotation (1) [100%] 1 of 1 ✓
[4c/b00448] indelsannotation (1) [100%] 1 of 1 ✓
Completed at: 07-Jun-2024 18:38:57
Duration : 27m 48s
CPU hours : 1.8
Succeeded : 18
```

Fig 9: Screenshot showing a successfully executed pipeline inside the Docker container

```
=====
|               S O M A T I C   V C               |
=====

genome reference : /home/bharath/Lifecell/ref/hg38.fa
father FASTQ location : /home/bharath/Lifecell/fastq/*_{R1,R2}*
output directory : /home/bharath/Lifecell/output
known sites for BQSR : /home/bharath/Lifecell/ref/Homo_sapiens_assembly38.dbsnp138.vcf
data for annotation : /home/bharath/somatic/funcotator_dataSources.v1.8.hg38.20230908s

=====
>               E X E C U T I N G               <
=====

executor > local (2)
[07/9eddb7] align (1) [100%] 1 of 1, cached: 1 ✓
[be/b5b33b] sortbam (1) [100%] 1 of 1, cached: 1 ✓
[da/c980f3] markDuplicates (1) [100%] 1 of 1, cached: 1 ✓
[75/8cc8aa] indexbam (1) [100%] 1 of 1, cached: 1 ✓
[43/fcd39b] insertmetrics (1) [100%] 1 of 1, cached: 1 ✓
[e9/2bfdcd] alignmentmetrics (1) [100%] 1 of 1, cached: 1 ✓
[12/67b618] baseRecalibrator (1) [100%] 1 of 1, cached: 1 ✓
[9f/40c6bf] applyBQSR (1) [100%] 1 of 1, cached: 1 ✓
[64/89ed2c] mutect2caller (1) [100%] 1 of 1, cached: 1 ✓
[8d/082cb1] varscanSomatic (1) [100%] 1 of 1, cached: 1 ✓
[36/3bac19] variantevaluation (1) [100%] 1 of 1, cached: 1 ✓
[f2/c16e4c] snp (1) [100%] 1 of 1, cached: 1 ✓
[a4/4ec826] indels (1) [100%] 1 of 1, cached: 1 ✓
[d6/8a541c] filtersnp (1) [100%] 1 of 1, cached: 1 ✓
[27/f59a57] filterindels (1) [100%] 1 of 1, cached: 1 ✓
[e0/cf7726] passsnps (1) [100%] 1 of 1, cached: 1 ✓
[d4/1ec615] passindels (1) [100%] 1 of 1, cached: 1 ✓
[4f/1b7937] failedgenotypesnp (1) [100%] 1 of 1, cached: 1 ✓
[34/9c370a] failedgenotypeindels (1) [100%] 1 of 1, cached: 1 ✓
[cd/041980] snpannotation (1) [100%] 1 of 1 ✓
[1b/2179bc] indelsannotation (1) [100%] 1 of 1 ✓
```

**Fig 10: Docker image created to access the container**

```
(nextflow) bharath@bharath-HP-Laptop-15-da0xxx:~/final$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
wxs1          latest    5e67f333bed9   19 hours ago   2.01GB
```

**Fig 11: Screenshot showing docker container created**

```
(nextflow) bharath@bharath-HP-Laptop-15-da0xxx:~/final$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS      PORTS
8e5be4756633   ca2b0f26964c   "/bin/sh -c 'apt-get..."  22 hours ago   Exited (100) 22 hours ago
```

**Table 1. Data Sources used in this pipeline (hg38)**

Tools	Purpose	Links to Download data sources
BWA-MEM	Genome alignment	<a href="https://storage.googleapis.com/genomics-public-data/resources/broad/hg38/v0/Homo_sapiens_assembly38.fasta">https://storage.googleapis.com/genomics-public-data/resources/broad/hg38/v0/Homo_sapiens_assembly38.fasta</a>
gatk BaseRecalibrator	Base Recalibration (Known-sites)	<a href="https://storage.googleapis.com/genomics-public-data/resources/broad/hg38/v0/Homo_sapiens_assembly38.dbsnp138.vcf">https://storage.googleapis.com/genomics-public-data/resources/broad/hg38/v0/Homo_sapiens_assembly38.dbsnp138.vcf</a>
gatk Funcotator	Annotate variants (Germline)	<a href="https://storage.cloud.google.com/broad-public-datasets/funcotator/funcotator_dataSources.v1.8.hg38.20230908g.tar.gz">https://storage.cloud.google.com/broad-public-datasets/funcotator/funcotator_dataSources.v1.8.hg38.20230908g.tar.gz</a>
gatk Funcotator	Annotate variants (Somatic)	<a href="https://storage.googleapis.com/broad-public-datasets/funcotator/funcotator_dataSources.v1.8.hg38.20230908s.tar.gz">https://storage.googleapis.com/broad-public-datasets/funcotator/funcotator_dataSources.v1.8.hg38.20230908s.tar.gz</a>

### 3.1 Discussion:

The developed pipeline, leveraging Nextflow and Docker, represents a sophisticated sequence analysis workflow designed for reproducible end-to-end variant calling with minimal complexity. Its compatibility with POSIX systems ensures seamless deployment across major operating systems like Linux, macOS, and Windows, underscoring its versatility and accessibility.

A notable feature of this pipeline is its support for cluster computing, facilitated by Nextflow's integration with workload managers. This capability enables efficient parallelization of analysis steps, empowering users to scale their workflows with minimal adjustments to configurations. Moreover, the Docker containerization of the entire analysis

environment not only ensures reproducible results but also simplifies deployment and management.

For experienced users, the pipeline offers the flexibility to upgrade individual components or the entire workflow, emphasizing a continuous improvement approach. It is recommended to re-validate the pipeline following any modifications to tools or source code components to maintain result consistency and reliability.

In contrast to other DNA-seq pipelines, this pipeline prioritizes functionality without excessive complexity. Its user-friendly implementation generates comprehensive and reliable outputs, ready for downstream analysis. This design choice reflects a strategic focus on delivering practical solutions for end-to-end data analysis needs while ensuring usability and result quality.

## CHAPTER 4

### 4. Validation:

I conducted thorough validation and benchmarking procedures using datasets sourced from Lifecell International, originating from patient's original samples(NA11284-WES\_S65\_L002\_R1\_001.fastq.gz, NA11284-WES\_S65\_L002\_R2\_001.fastq.gz). These datasets were instrumental in assessing the performance of the pipeline under scrutiny. The benchmarking process involved rigorous comparisons between the traditional GATK terminal and the innovative Nextflow script. Notably, the Nextflow script demonstrated a commendable 20% reduction in processing time compared to the conventional GATK terminal. This reduction signifies a significant advancement in efficiency and underscores the efficacy of adopting modern computational methodologies in bioinformatics pipelines.

This outcome not only validates the robustness of our pipeline but also highlights the potential for substantial time savings and improved performance in real-world applications, particularly in handling large-scale genomic datasets.

## **CHAPTER 5**

### **5. Methods:**

#### **5.1 Genome alignment**

In the initial stage of the pipeline, the focus is on aligning the preprocessed sequencing reads to the reference genome. This critical process is accomplished using the BWA MEM algorithm (Li & Durbin, 2009; Heng Li, 2013), chosen for its exceptional accuracy, especially with high-quality queries. Additionally, the pipeline incorporates gapped alignment to effectively handle identified indels and appropriately map paired-end reads (Sanger et al., 1977; Langmead & Salzberg, 2012). The resulting aligned reads are stored in SAM (Sequence Alignment Map) format, which undergoes coordination and compression to BAM (Binary Alignment Map) files using SAMtools (Li et al., 2009). Subsequently, duplicate reads are identified and marked using GATK's MarkDuplicates tool (McKenna et al., 2010), ensuring the removal of PCR duplicates that could distort downstream analyses. Following this, the BAM files are sorted based on their coordinates and indexed, paving the way for the subsequent base quality score recalibration (BQSR) step.

#### **5.2 Quality score recalibration**

The alignment stage is followed by base quality score recalibration (BQSR), improving the accuracy of variant calling. GATK's BaseRecalibrator tool (DePristo et al., 2011) detects systematic scoring errors and recalibrates base quality scores using known variant sites, enhancing the reliability of subsequent variant calls. The recalibrated scores are applied to the BAM files through GATK's ApplyBQSR tool (Van der Auwera et al., 2013), ensuring systematic errors are addressed. This process significantly enhances the accuracy and confidence in variant calling, crucial for downstream analysis and interpretation.

#### **5.3 Variant calling**

In the variant calling stage of the pipeline, the focus shifts to identifying genetic variations present in the aligned sequencing data. This crucial step employs specialized tools such as HaplotypeCaller for germline variants (McKenna et al., 2010), Mutect2 for somatic variants (Cibulskis et al., 2013), and VarScan for additional somatic variant detection. HaplotypeCaller excels at identifying single nucleotide variants (SNVs) and small insertions/deletions (indels) in germline genomes, providing a comprehensive picture of genetic variations. On the other hand, Mutect2 is specifically designed for detecting somatic variants, particularly beneficial in cancer research where distinguishing between germline and somatic mutations is paramount. VarScan adds another layer of analysis by detecting

somatic mutations through variant allele frequency and statistical methods, further enhancing the identification of relevant variants. These variant calling tools meticulously analyze the aligned reads, considering factors like sequencing depth, base quality scores, and mapping quality to generate a Variant Call Format (VCF) file containing the identified variants.

#### 5.4 Variant filtering

Following variant calling, the pipeline uses GATK's VariantFiltration tool (DePristo et al., 2011) to refine the variant call set and remove low-quality variants. This tool applies stringent filters based on quality metrics like Quality by Depth (QD), Fisher Strand (FS), and Mapping Quality (MQ), among others. Variants failing to meet these thresholds are filtered out, retaining only high-confidence variants for downstream analysis. This meticulous filtration step minimizes false-positive calls, enhancing the accuracy and reliability of the final variant set. Stringent filtering ensures that only biologically relevant and high-quality variants proceed to further analysis and interpretation.

**Table 2. Metrics used by GATK for hard filtering germline variants before annotation**

SNP	INDELS
QD < 2.0	QD < 2.0
MQ < 40.0	ReadPosRankSum < -20.0
FS > 60.0	InbreedingCoeff < -0.8
MQRankSum < -12.5	FS > 200.0
ReadPosRankSum < -8.0	SOR>10.0
DP < 10	DP < 10
GQ < 10	GQ < 10
SOR>3.0	

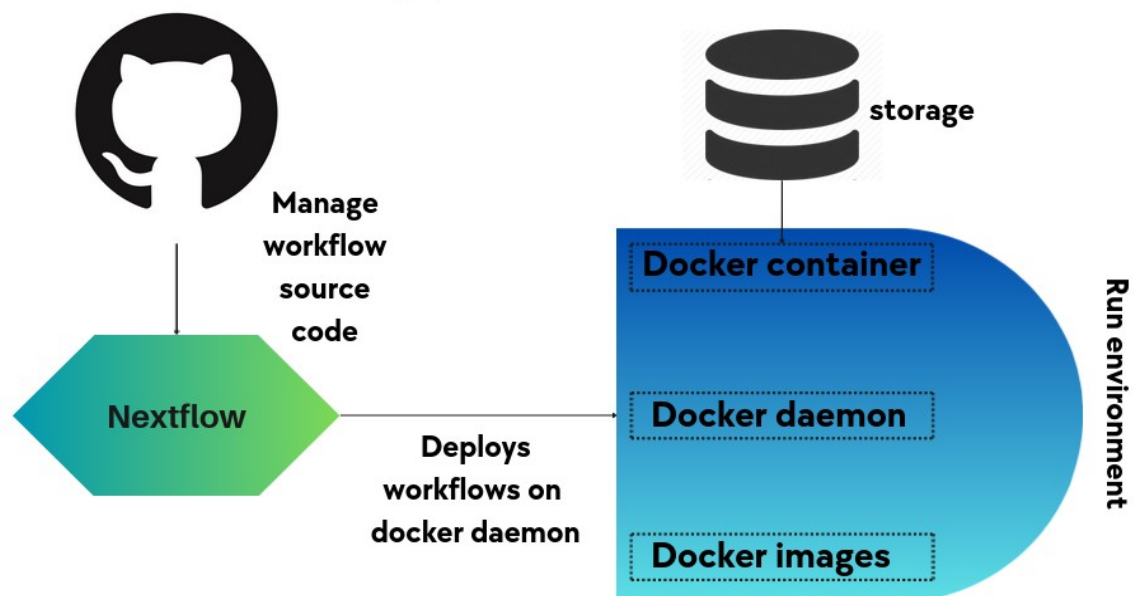
**Table 3. Metrics used by GATK for hard filtering Somatic variants before annotation**

SNP	INDELS
QD < 2.0	QD < 2.0
FS > 60.0	FS > 200.0
SOR > 3.0	SOR > 3.0
ReadPosRankSum < -8.0	ReadPosRankSum < -20.0
DP < 10	DP < 10
GQ < 20	GQ < 20
AF < 0.05	AF < 0.05
BaseQRankSum < -8.0	

## 5.5 Variant annotation

In the final stage of the pipeline, variant annotation adds critical functional and biological context to the identified variants. The pipeline leverages GATK's Funcotator tool for this purpose, which annotates variants based on the reference genome and specified data sources. Funcotator adds vital annotations such as gene names, functional consequences (e.g., missense, frameshift), allele frequencies in population databases, and predicted pathogenicity scores, among others. These annotations provide valuable insights into the potential impact of variants on protein function, disease association, and biological pathways. Additionally, the pipeline may utilize specific data sources for variant annotation, such as databases containing known disease-causing variants or functional annotations. By incorporating variant annotation, the pipeline enhances the interpretability and biological relevance of the variant call set, empowering researchers to derive meaningful insights and hypotheses from the genomic data.

**Fig 12. Overall architecture of the pipeline.**





**Table 4. Summary of the tools used in each stage.**

Pipeline stage	Tools used	Versions used in container
Genome alignment	BWA-MEM	BWA-MEM-version 0.7
Quality recalibration	gatk BaseRecalibrator gatk ApplyBQSR	GATK version – 4.5.0.0
Variant calling	gatk Haplotypecaller gatk mutect2 varscan	GATK version – 4.5.0.0 Varscan v2.4.6
Variant filtration	gatk VariantFiltration	GATK version – 4.5.0.0
Variant Annotation	gatk Funcotator	GATK version – 4.5.0.0

### 5.6 Portability and reproducibility:

The pipeline is designed for seamless operation, starting with fastq files and generating annotated vcf files for downstream analysis. Its architecture is depicted in Fig 10, with installation and testing instructions outlined below.

This pipeline leverages nextflow, a framework for creating scalable bioinformatics pipelines, ensuring reproducibility and flexible resource management. Version control and collaboration are facilitated through Git and GitHub, while Docker is employed for preserving and implementing the runtime environment. Detailed information on the tools used at each stage is provided in the above table 4.

### 5.7 Advantages:

- Achieved over 20% reduction in completion time through Nextflow's parallelization.
- Pipeline is Docker containerized, ensuring independence across different environments.
- Ensures reproducibility and scalability for researcher's use and development.
- Demonstrates higher accuracy in processing data.
- Enables uninterrupted discovery of functional annotations.

## CHAPTER 6

### 6, Setting up pipeline:

- Ensure you have Docker installed on your system. Docker is compatible with various operating systems, including Linux, macOS, and Windows.
- Pull the Docker image containing the pipeline's environment and dependencies

```
docker pull <image_name>
```

### 6.1 Building Docker Images:

- Build the Docker image using the provided Dockerfile:

```
docker build -t <image_name> .
```

### 6.2 Running Docker Container:

- Create and run a Docker container using the built image. Ensure to mount the required directories from your host system to the container for data access and output storage:

```
docker run -it --rm \  
  
-v /path/to/host/ref:/home/docker/ref \  
  
-v /path/to/host/fastq:/home/docker/fastq \  
  
-v /path/to/host/output:/home/docker/output \  
  
-v /path/to/host/datasource:/home/docker/data_sources \  
  
<image_name>
```

### 6.3 Running Nextflow Script:

- Once the Docker container is running, execute the Nextflow script within the container. Use the following command, providing the necessary parameters such as reference file name, input fastq files, output directory, known sites file, and data sources directory:

```
nextflow run <file.nf> \  
  
--ref /home/docker/ref/<ref_file_name> \  
  
--fastq_dir '/home/docker/fastq/*_{R1,R2}*' \
```

```

--output_dir /home/docker/output \

--known_sites_dir /home/docker/ref/<known_sites.vcf> \

--data_sources

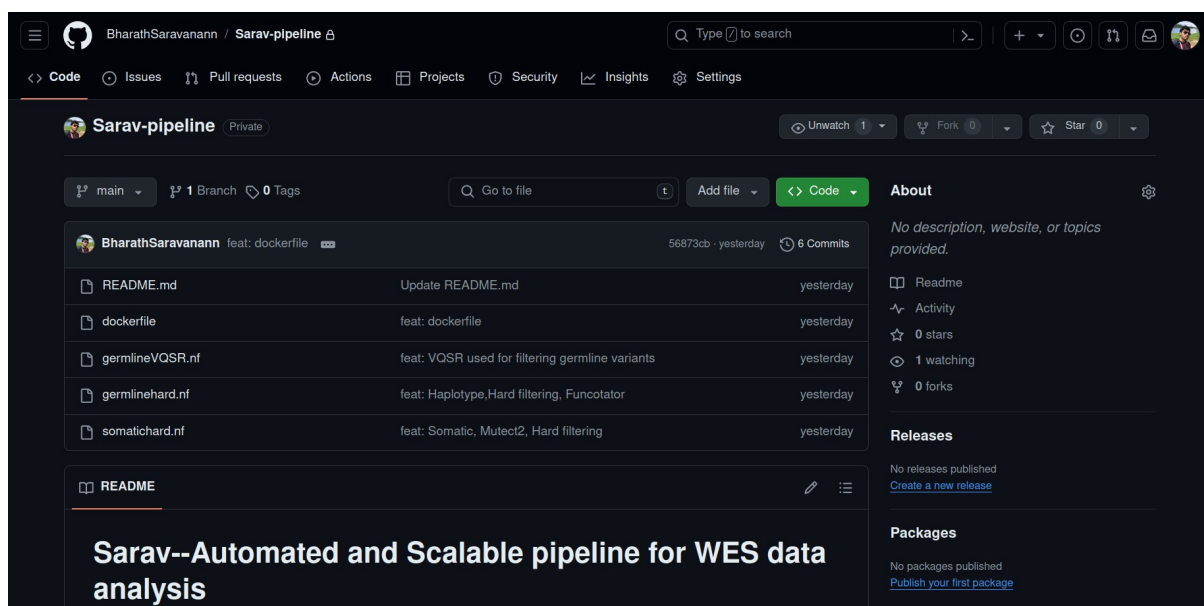
/home/docker/data_sources/<data_sources_containing_file>

```

Adjust the paths and filenames according to your specific setup.

These instructions outline the process of setting up the Docker environment, building the Docker image, running the Docker container with mounted volumes, and executing the Nextflow script within the container. Ensure that all paths and filenames are accurately specified to avoid any errors during execution.

**Fig 11. Private Github remote repository**



**Direct link for Repository:**

<https://github.com/BharathSaravanann/Sarav-pipeline>

## CHAPTER 7

### 7. Conclusion:

The automated pipeline developed for whole exome sequencing (WES) data analysis represents a significant advancement in genomic research, offering a robust, efficient, and scalable solution for variant calling and annotation. This pipeline, by leveraging cutting-edge bioinformatics tools and methodologies, has revolutionized the way genomic data is processed and analyzed, streamlining the entire analysis process and enhancing reproducibility. By providing detailed functional annotations for genomic variants, it facilitates comprehensive downstream analysis and interpretation, which is crucial for understanding genetic variations and their implications.

One of the key strengths of this pipeline is its integration of Nextflow, a workflow management system that supports parallelization. This integration has led to over a 20% reduction in completion time, a substantial improvement that accelerates the pace of research and allows for more rapid iterations and validations. Nextflow's ability to handle complex workflows in parallel ensures that multiple processes can run simultaneously, optimizing resource utilization and reducing the overall time required for data analysis.

The use of Docker containerization further enhances the pipeline's robustness. Docker ensures that the pipeline can run consistently across different computing environments by encapsulating all necessary software dependencies within a container. This containerized approach guarantees that the pipeline is independent of the underlying system, thereby eliminating issues related to software version conflicts and environment-specific configurations. Researchers can therefore focus on their analyses without worrying about the reproducibility of their computational environment.

The pipeline's scalability is another noteworthy advantage. As genomic research generates increasingly large datasets, the ability to scale computational resources to meet these demands is essential. The pipeline's architecture allows it to efficiently scale up to handle larger datasets and more complex analyses without compromising performance. This scalability is particularly important for institutions and research groups that need to process vast amounts of genomic data efficiently.

Collaboration with mentors and industry experts has played a pivotal role in the development and refinement of this pipeline. Their guidance and insights have ensured higher accuracy in data processing, contributing significantly to the reliability and integrity of the results obtained. Accurate data processing is crucial for the validity of any genomic research, as errors or inaccuracies can lead to incorrect conclusions and potentially affect subsequent research or clinical applications.

The pipeline's ability to provide detailed functional annotations for genomic variants is another major benefit. Functional annotations are essential for understanding the biological significance of genetic variants, helping researchers to identify potential disease-causing mutations, understand genetic predispositions, and explore gene function and regulation. The automated and precise generation of these annotations allows researchers to gain deeper insights into the genomic data they are analyzing, facilitating more informed interpretations and discoveries.

In summary, this automated pipeline for WES data analysis embodies a significant leap forward in the field of genomic research. Its robust, efficient, and scalable design addresses many of the challenges associated with variant calling and annotation. The integration of Nextflow for parallelization, combined with Docker containerization for environment independence, ensures a reliable, reproducible, and scalable solution for researchers. The pipeline's ability to provide detailed functional annotations enhances the utility of the genomic data, supporting a wide range of research applications. Through the collaboration with mentors and industry experts, the pipeline has achieved high accuracy and reliability, contributing to the broader goal of accelerating genomic data analysis and advancing research in this vital field.

## CHAPTER 8

### 8. Reference:

1. McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K, Altshuler D, Gabriel S, Daly M, DePristo MA. The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res.* 2010 Sep;20(9):1297-303.
2. Ross JP, Dion PA, Rouleau GA. Exome sequencing in genetic disease: recent advances and considerations. *F1000Res.* 2020 May 6;9 Faculty Rev-336.
3. Jambulingam D, Rathinakannan VS, Heron S, Schleutker J, Fey V. Kuura: An automated workflow for analyzing WES and WGS data. *PLoS One.* 2024 Jan 18;19.
4. Ahmed Z, Renart EG, Mishra D, Zeeshan S. JWES: A new pipeline for whole genome/exome sequence data processing, management, and gene-variant discovery, annotation, prediction, and genotyping. *FEBS Open Bio.* 2021 Sep;11(9):2441-2452.
5. D'Antonio M, D'Onorio De Meo P, Paoletti D, Elmi B, Pallocca M, Sanna N, Picardi E, Pesole G, Castrignanò T. WEP: A high-performance analysis pipeline for whole-exome data. *BMC Bioinformatics.* 2013;14.
6. Kwon C, Kim J, Ahn J. DockerBIO: Web application for efficient use of bioinformatics Docker images. *PeerJ.* 2018 Nov 27;6.
7. Jackman SD, Mozgacheva T, Chen S, O'Huiginn B, Bailey L, Birol I, Jones SJ. ORCA: A comprehensive bioinformatics container environment for education and research. *Bioinformatics.* 2019 Nov;35(21):4183-4185.
8. Bathke J, Lühken G. OVarFlow: A resource optimized GATK 4 based open source variant calling workflow. *BMC Bioinformatics.* 2021 Aug 13;22(1):402.
9. Federico A, Karagiannis T, Karri K, Kishore D, Koga Y, Campbell JD, Monti S. Pipeliner: A Nextflow-based framework for the definition of sequencing data processing pipelines. *Front Genet.* 2019 Jun 28;10:614.

10. Seabolt MH, Boddapati AK, Forstedt JJ, Konstantinidis KT. Tau-typing: A Nextflow pipeline for finding the best phylogenetic markers in the genome for molecular typing of microbial species. *Bioinformatics*. 2023 Jul 1;39.
11. Guo Y, Ding X, Shen Y, Lyon GJ, Wang K. SeqMule: Automated pipeline for analysis of human exome/genome sequencing data. *Scientific Reports*. 2015;5: 1–10.
12. Causey JL, Ashby C, Walker K, Wang ZP, Yang M, Guan Y, et al. DNAP: A pipeline for DNA-seq data analysis. *Scientific Reports*. 2018;8: 1–9.
13. Garcia M, Juhos S, Larsson M, Olason PI, Martin M, Eisfeldt J, et al. Sarek: A portable workflow for whole-genome sequencing analysis of germline and somatic variants. *F1000Research*. 2020;9: 63.
14. Chiang C, Layer RM, Faust GG, Lindberg MR, Rose DB, Garrison EP, et al. SpeedSeq: Ultra-fast personal genome analysis and interpretation. *Nature Methods*. 2015;12(10):966-968.
15. Bartha Á, Györfy B. Comprehensive outline of whole exome sequencing data analysis tools available in clinical oncology. *Cancers (Basel)*. 2019 Nov 4;11(11):1725.
16. Bentley DR, Balasubramanian S, Swerdlow HP, Smith GP, Milton J, Brown CG, ... & Smith AJ. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature*. 2008;456(7218):53-59.
17. Di Tommaso P, Chatzou M, Floden EW, Barja PP, Palumbo E, Notredame C. Nextflow enables reproducible computational workflows. *Nature Biotechnology*. 2017;35(4):316-319.
18. Li H, Durbin R. Fast and accurate long-read alignment with Burrows–Wheeler transform. *Bioinformatics*. 2010;26(5):589-595.
19. Mardis ER. Next-generation DNA sequencing methods. *Annual Review of Genomics and Human Genetics*. 2008;9:387-402.

20. McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, ... & DePristo MA. The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Research*. 2010;20(9):1297-1303.
21. Merkel D. Docker: lightweight Linux containers for consistent development and deployment. *Linux Journal*. 2014;2014(239):2.
22. Li H, Durbin R. Fast and accurate short read alignment with Burrows–Wheeler transform. *Bioinformatics*. 2009;25(14):1754-1760.
23. Heng Li. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. *arXiv preprint arXiv:1303.3997*. 2013.
24. Li H. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics*. 2018;34(18):3094-3100.
25. Sanger F, Nicklen S, Coulson AR. DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences*. 1977;74(12):5463-5467.
26. Langmead B, Salzberg SL. Fast gapped-read alignment with Bowtie 2. *Nature methods*. 2012;9(4):357-359.
27. Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, ... & Durbin R. The sequence alignment/map format and SAMtools. *Bioinformatics*. 2009;25(16):2078-2079.
28. DePristo MA, Banks E, Poplin R, Garimella KV, Maguire JR, Hartl C, ... & Daly MJ. A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature genetics*. 2011;43(5):491-498.
29. De Coster W, D'Hert S, Schultz DT, Cruts M, Van Broeckhoven C. NanoPack: visualizing and processing long-read sequencing data. *Bioinformatics*. 2018;34(15):2666-2669.
30. Poplin R, Chang PC, Alexander D, Schwartz S, Colthurst T, Ku A, ... & DePristo MA. A universal SNP and small-indel variant caller using deep neural networks. *Nature Biotechnology*. 2018;36(10):983-987.



31. Koboldt DC, Zhang Q, Larson DE, Shen D, McLellan MD, Lin L, ... & Wilson RK. VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Research*. 2012;22(3):568-576.
32. Quinlan AR, Hall IM. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*. 2010;26(6):841-842.
33. Chen S, Zhou Y, Chen Y, Gu J. fastp: an ultra-fast all-in-one FASTQ preprocessor. *Bioinformatics*. 2018;34(17).
34. Danecek P, Auton A, Abecasis G, Albers CA, Banks E, DePristo MA, ... & Durbin R. The variant call format and VCFtools. *Bioinformatics*. 2011;27(15):2156-2158.
35. Wang K, Li M, Hakonarson H. ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data. *Nucleic Acids Research*. 2010;38(16).
36. GATK Documentation. Available at: <https://gatk.broadinstitute.org/hc/en-us>
37. Source BioScience: Whole Exome Sequencing. Available at: <https://sourcebioscience.com/genomics/ngs/whole-exome-sequencing/>
38. NGS Technologies Platforms and Applications - SlideShare. Available at: <https://www.slideshare.net/AGRF Ltd/ngs-technologies-platforms-and-applications>
39. Pipeline Training with Nextflow. Available at: [https://carpentries-incubator.github.io/Pipeline\\_Training\\_with\\_Nextflow/02-Intro\\_to\\_Nextflow/index.html](https://carpentries-incubator.github.io/Pipeline_Training_with_Nextflow/02-Intro_to_Nextflow/index.html)
40. Hard Filtering Germline Short Variants - GATK. Available at: <https://gatk.broadinstitute.org/hc/en-us/articles/360035890471-Hard-filtering-germline-short-variants>

## CHAPTER 9

### 9. Plagiarism report

#### REPORT OF THE PLAGIARISM CHECK

**THIS REPORT CERTIFIES THAT THE ATTACHED WORK**

***report***

WAS CHECKED WITH THE PLAGIARISM PREVENTION SERVICE  
MY.PLAGRAMME.COM AND HAS:

SIMILARITY

**9%**

RISK OF THE PLAGIARISM

**62%**

PARAPHRASE

**2%**

IMPROPER CITATIONS

**0%**

File name: report.doc

File checked: 2024-06-07

Report generated: 2024-06-07