**Transcriptomics: RNA-Seq Based Differential Gene Expression Analysis Using Linux, Galaxy, and R**

**A**

**Summer Internship Report 2025**

**presented**

**by**

**Muhammed Iyash Mp**

**to**

**Amity Institute of Biotechnology**

**in partial fulfilment of the requirements**

**for the degree of**

**BTech Biotechnology**

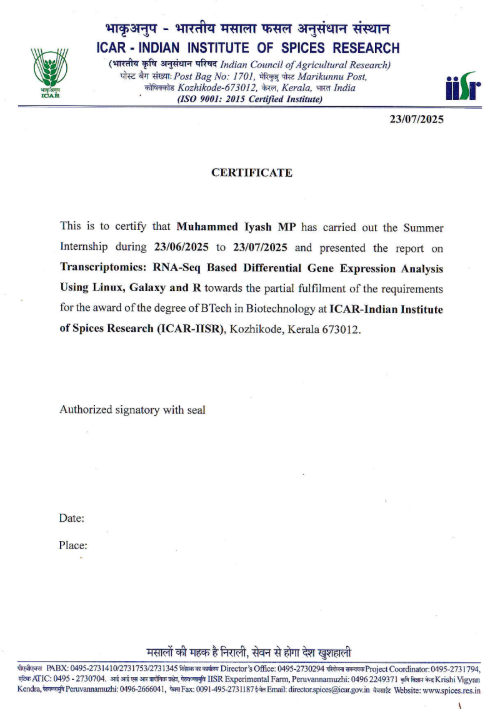
**in the subject of Biotechnology**

****

**Amity University Maharashtra Mumbai**

**Under the Guidance of**

**Dr. Sona Charles  
Scientist (Agricultural Bioinformatics)  
Bioinformatics and Integrative Genomics (BIG) Facility  
Crop Improvement and Biotechnology Division  
ICAR – Indian Institute of Spices Research, Kozhikode**

****

****

**23/07/2025**

**CERTIFICATE**

This is to certify that **Muhammed Iyash MP** has carried out the Summer Internship during **23/06/2025** to **23/07/2025** and presented the report on **Transcriptomics: RNA-Seq Based Differential Gene Expression Analysis Using Linux, Galaxy, and R** towards the partial fulfilment of the requirements for the award of the degree of BTech in Biotechnology at **ICAR IISR-Indian Institute of Spices Research**,ISR Road, Moozhikkal - Chelavoor, Road, P.O, Vellimadukunnu, Kozhikode, Kerala 673012.

Authorized signatory with seal

Date:

Place:

**ACKNOWLEDGEMENT**

I would like to express my heartfelt gratitude to the Indian Institute of Spices Research (ICAR-IISR), under the Indian Council of Agricultural Research (ICAR), for providing me with the opportunity to undertake a one-month internship program. The training offered a well-structured and insightful exposure to the interdisciplinary field of bioinformatics, combining elements of computational biology, data analysis, and molecular science.

I am especially thankful to my guide, Dr. Sona Charles, for her invaluable guidance, mentorship, and unwavering support throughout the course of the internship. Her expert advice, constructive feedback, and encouragement at every stage of my learning helped me build a strong foundation in transcriptomic analysis and its real-world applications. Her dedication to scientific rigor and clarity has deeply inspired me.

I would also like to express my appreciation to the Bioinformatics Division at ICAR-IISR for the structured learning modules and resources made available during the internship. The hands-on training provided in Linux-based command line tools, R programming for statistical computing, and Galaxy platform for high-throughput data processing greatly enhanced my understanding of modern bioinformatics workflows. Of particular importance was the practical experience gained in differential gene expression (DGE) analysis using publicly available datasets, which added a significant dimension to my academic learning.

I would also like to thank **Ms. Shahana**, Project Associate at the Bioinformatics and Integrative Genomics Facility, for her kind assistance, timely help, and valuable inputs during various stages of the project.

I am also thankful to Amity University Mumbai, especially the Department of Biotechnology, for supporting and encouraging such internship initiatives, which play a crucial role in bridging theoretical knowledge with practical research-based skills. The integration of institutional learning with external research environments has been instrumental in shaping my scientific outlook.

**ABBREVIATIONS**

|  |  |
| --- | --- |
| Abbreviation | Full Form |
| BAM | Binary Alignment Map |
| BED | Browser Extensible Data |
| CSV | Comma Separated Values |
| DGE | Differential Gene Expression |
| ENA | European Nucleotide Archive |
| ENSEMBL | European Molecular Biology Laboratory's Genome Database |
| FASTA | Text Format for Nucleotide/Protein Sequences |
| FASTQ | Text Format for Raw Sequencing Reads |
| FDR | False Discovery Rate |
| FPKM | Fragments Per Kilobase of transcript per Million mapped reads |
| FTP | File Transfer Protocol |
| GALAXY | Web-based Platform for Data-Intensive Biomedical Research |
| GFF | General Feature Format |
| GO | Gene Ontology |
| GSEA | Gene Set Enrichment Analysis |
| GTF | Gene Transfer Format |
| HISAT2 | Hierarchical Indexing for Spliced Alignment of Transcripts |
| ICAR | Indian Council of Agricultural Research |
| IISR | Indian Institute of Spices Research |
| KEGG | Kyoto Encyclopedia of Genes and Genomes |
| LFC | Log Fold Change |
| MA Plot | Minus-Average Plot |
| PCA | Principal Component Analysis |
| p-value | Probability Value |
| Q-value | Adjusted p-value for Multiple Testing |
| QC | Quality Control |
| R | Statistical Programming Language |
| RPKM | Reads Per Kilobase per Million mapped reads |
| SAM | Sequence Alignment Map |
| SRA | Sequence Read Archive |
| STAR | Spliced Transcripts Alignment to a Reference |
| TPM | Transcripts Per Million |
| UTR | Untranslated Region |
| VCF | Variant Call Format |
| WGS | Whole Genome Sequencing |
| Cutadapt | Adapter Trimming Tool for FASTQ Files |
| Trimmomatic | A Flexible Trimmer for Illumina NGS Data |
| FastQC | Quality Control Tool for High Throughput Sequence Data |
| FeatureCounts | Read Counting Tool from Subread Package |
| DESeq2 | R Package for Differential Gene Expression Analysis |
| ggplot2 | R Package for Data Visualization |
| EnhancedVolcano | R Package for Generating Volcano Plots |
| SAMtools | Suite for Handling High-Throughput Sequence Data (SAM/BAM files) |

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| Chapter No. | Title | Page No. |
| 1 | Introduction – Overview of transcriptomics, project objectives, and tools used | 8-9 |
| 2 | Theoretical Background – RNA-Seq, gene expression, splicing, DESeq2 theory | 10-13 |
| 3 | Fundamentals of Bioinformatics Tools – Linux, R programming, Galaxy, databases | 14-38 |
| 4 | Linux Techniques and Applications – Shell commands, file handling, Standalone BLAST | 39-46 |
| 5 | Galaxy Platform and Preprocessing Workflow – FASTQC, Cutadapt, Trimmomatic, HISAT2 | 47-57 |
| 6 | Differential Gene Expression Analysis using R – DESeq2, plots, thresholds, results | 58-64 |
| 7 | Results and Interpretation – Gene expression outcomes and biological relevance | 65-73 |
| 8 | Conclusion and Future Scope – Summary of learning and potential applications | 74-76 |
| 9 | References – Manuals, software, websites, tools | 77-78 |

**CHAPTER 1: INTRODUCTION**

In recent years, the field of transcriptomics has emerged as a key area of genomics, enabling scientists to explore and understand gene activity at a molecular level. Transcriptomics refers to the comprehensive study of RNA transcripts produced by the genome under specific conditions, developmental stages, or in response to genetic mutations or environmental changes. It provides a snapshot of active genes and their expression levels, making it a valuable approach for uncovering molecular mechanisms, cellular responses, and functional roles of genes.

Among the available transcriptomic approaches, RNA sequencing (RNA-Seq) is a powerful next-generation sequencing (NGS) method that allows researchers to quantitatively and qualitatively assess RNA populations in a high-throughput manner. RNA-Seq facilitates the identification of differentially expressed genes (DEGs), discovery of novel transcripts, detection of alternative splicing events, and overall transcriptome profiling. This level of insight is critical for understanding gene regulation during biological processes such as development, disease progression, and response to treatments.

**Project Overview**

This project is focused on studying the differential gene expression (DGE) in *Drosophila melanogaster* (fruit fly) embryos to understand the molecular impact of a specific mutation. The biological model involves embryos carrying the P218 mutation in the *faint sausage (fas)* gene, which plays a vital role in embryonic morphogenesis and possibly pre-mRNA splicing. By comparing transcriptomic data between mutant embryos and wild-type controls, this study aims to elucidate the role of the *fas* gene in regulating early embryonic development through gene expression changes.

**Objective of the Study**

The primary objective of this project is to:

Identify and analyze differentially expressed genes (DEGs) between wild-type and P218 *fas*-mutant *Drosophila* embryos using RNA-Seq data, to understand the potential role of *fas* in pre-mRNA splicing and developmental regulation.

To achieve this, a complete bioinformatics pipeline was established that integrates multiple tools across different platforms.

**Bioinformatics Workflow**

The analysis began with gaining familiarity with Linux, a command-line based operating system commonly used in bioinformatics. Using the Linux terminal, we performed essential operations like data management and executed basic bioinformatics tools including BLAST for sequence similarity searches and annotation.

The core preprocessing of RNA-Seq data was conducted on the UseGalaxy.org platform, a web-based GUI that facilitates reproducible and user-friendly bioinformatics workflows. The pipeline steps included:

* FASTQC: Assessed the quality of raw sequencing reads
* Cutadapt and Trimmomatic: Removed adapters and trimmed low-quality regions
* HISAT2: Aligned the high-quality reads to the *Drosophila melanogaster* reference genome
* SAMtools: Converted and sorted alignment files (SAM/BAM formats)
* featureCounts: Generated a matrix of read counts mapped to annotated genes

Following data preprocessing, the count matrix was imported into R, where the statistical package DESeq2 was used for differential gene expression analysis. DESeq2 normalizes the data, estimates dispersions, and fits models to test for significant differences in expression between sample groups. This analysis outputs key results such as:

* Normalized gene expression values
* Log2 fold changes
* Adjusted p-values (FDR)
* Statistical visualizations including volcano plots, MA plots, and heatmaps

These results allow us to interpret which genes are significantly upregulated or downregulated in the mutant embryos compared to controls, thereby linking the *fas* gene mutation to specific transcriptomic effects.

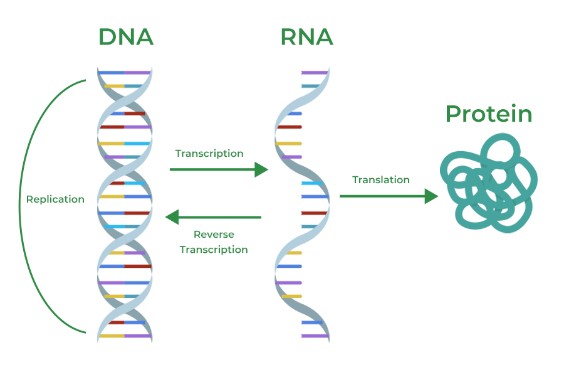
**Chapter 2: Theoretical Background**

**2.1 Introduction to Transcriptomics**

Transcriptomics is the study of the complete set of RNA transcripts produced by the genome under specific circumstances or in a specific cell. It is a key subfield of functional genomics and provides insights into the patterns of gene expression. Unlike the static genome, the transcriptome is dynamic and reflects the cellular response to internal and external stimuli. By analyzing the transcriptome, researchers can determine when and where genes are turned on or off, revealing molecular mechanisms underlying development, differentiation, and disease.

The transcriptome includes all types of RNA, such as messenger RNA (mRNA), ribosomal RNA (rRNA), transfer RNA (tRNA), and non-coding RNAs. Among these, mRNA is often the main focus in transcriptomics studies because it represents the protein-coding genes that are actively expressed.

**2.2 Central Dogma of Molecular Biology**

At the heart of transcriptomics lies the central dogma of molecular biology, which describes the flow of genetic information within a biological system:

* DNA → RNA → Protein

Genes in the DNA are transcribed into RNA, and most of these RNAs are translated into proteins. Transcription is carried out by RNA polymerase, which produces a complementary RNA copy of a gene's DNA sequence. This RNA undergoes processing to become a mature mRNA, which is then translated into protein in the ribosome.

Understanding gene expression regulation at the transcriptional level is vital to comprehend how genes control the phenotype. Aberrations in these processes can lead to diseases, developmental issues, or altered cellular states.

**2.3 RNA-Seq: Technology Overview**

RNA-Seq (RNA sequencing) is a powerful technique for transcriptome profiling. It utilizes next-generation sequencing (NGS) technologies to quantify and characterize the entire transcriptome.

Workflow Overview:

1. RNA Extraction: Total RNA is extracted from the sample.
2. Library Preparation: RNA is fragmented, converted into cDNA, and adapter-ligated.
3. Sequencing: The libraries are sequenced using platforms like Illumina.
4. Data Processing: Reads are aligned to a reference genome/transcriptome.
5. Quantification: Expression levels are quantified, followed by differential expression analysis.

RNA-Seq surpasses previous techniques like microarrays due to its high sensitivity, ability to detect novel transcripts, isoforms, and low-abundance genes.

**2.4 Gene Expression and Regulation**

Gene expression is tightly regulated and controlled by transcription factors, epigenetic marks, RNA-binding proteins, and splicing machinery. Several layers of control ensure that genes are expressed at the right time, place, and amount.

* Promoters and enhancers regulate transcription initiation.
* Alternative splicing generates multiple mRNA variants from a single gene.
* Post-transcriptional modifications (like polyadenylation, capping, and editing) influence mRNA stability and translatability.
* Non-coding RNAs, such as microRNAs, can silence gene expression post-transcriptionally.

Studying these elements through transcriptomics can reveal critical insights into gene function and disease mechanisms.

**2.5 Alternative Splicing and the Role of fas Gene**

One significant focus of the project is to understand the role of the faint sausage (fas) gene in pre-mRNA splicing during embryogenesis in *Drosophila melanogaster*. The fas gene has been implicated in the development of the nervous system and embryonic patterning. Mutations such as P218 in fas could potentially disrupt normal splicing, leading to downstream defects.

Alternative splicing allows a single gene to code for multiple proteins by rearranging exon combinations. This is especially vital during embryogenesis, where precise control of gene expression and isoform diversity is required.

Analyzing how the P218 mutation affects the transcriptome using RNA-Seq can highlight the fas gene’s contribution to developmental processes.

**2.6 Differential Gene Expression (DGE) Analysis**

Differential gene expression analysis is a critical step in transcriptomics that identifies genes with significant expression changes between two or more conditions — in this case, mutant versus wild-type embryos.

Key components of DGE analysis include:

* Read Count Generation: Using tools like featureCounts to quantify reads mapped to genes.
* Normalization: Adjusting for sequencing depth and RNA composition.
* Statistical Testing: Identifying genes with significant expression changes using tools like DESeq2 in R.

This analysis helps identify potential biomarkers, disease genes, or regulatory pathways affected by the fas mutation.

**2.7 Visualization Techniques in Transcriptomics**

Data visualization is crucial for interpreting complex RNA-Seq data. Some common plots include:

* Heatmaps: Show expression levels of genes across samples. Useful to identify clusters of co-expressed genes.
* Principal Component Analysis (PCA): A dimensionality reduction technique to explore sample similarity and variation.
* MA Plot: Displays log fold change against mean expression, useful for spotting differentially expressed genes.
* Volcano Plot: Highlights genes with both high statistical significance and large fold change.

These visualizations support biological interpretation and provide evidence of the underlying transcriptomic alterations.

**2.8 Tools and Platforms in Transcriptomic Analysis**

Numerous bioinformatics tools are involved in RNA-Seq preprocessing and analysis:

|  |  |
| --- | --- |
| Tool | Function |
| FastQC | Quality control of raw reads |
| Cutadapt | Trimming adapter sequences |
| Trimmomatic | Quality-based read trimming |
| HISAT2 | Alignment of reads to the reference genome |
| SAMtools | Handling and manipulating alignment files (SAM/BAM) |
| featureCounts | Counting mapped reads for each gene |
| DESeq2 | Differential expression analysis in R |

Many of these tools can be accessed via the Galaxy platform, which provides a user-friendly web-based interface for bioinformatics workflows.

**Chapter 3: Fundamentals of Bioinformatics Tools**

**3.1 Introduction to Bioinformatics in Modern Biology**

Bioinformatics is a multidisciplinary field that combines biology, computer science, mathematics, and statistics to process, analyze, and interpret complex biological data. In recent decades, the advent of high-throughput sequencing technologies such as RNA sequencing (RNA-Seq), whole-genome sequencing (WGS), and exome sequencing has revolutionized biological and biomedical research. As a result, massive volumes of data are being generated at an unprecedented scale, necessitating the use of computational tools and algorithms to derive meaningful insights.

In the context of transcriptomics—the study of the complete set of RNA transcripts produced by the genome—bioinformatics plays a central role. It allows for the comprehensive analysis of gene expression patterns under different biological conditions, enabling researchers to identify differentially expressed genes, biological pathways, and potential biomarkers. From raw sequence data to biological conclusions, bioinformatics forms the backbone of the entire analytical workflow.

The transcriptomic project undertaken during this internship relied heavily on several core bioinformatics tools and environments. These include:

* Linux: A powerful command-line operating system essential for handling large-scale biological data, running bioinformatics software, and automating processes via scripting.
* R programming: A statistical programming language that is widely used for data analysis and visualization, especially in gene expression studies using packages like DESeq2.
* Galaxy platform: A web-based platform that provides a user-friendly interface for bioinformatics tools, making it easier to run analyses without requiring advanced programming skills.
* DESeq2: A widely used R package for analyzing count-based RNA-Seq data and performing differential gene expression analysis.
* Other tools: Software such as FastQC, Trimmomatic, HISAT2, SAMtools, and FeatureCounts form the preprocessing and alignment backbone of RNA-Seq workflows.

**3.2 Fundamentals of Linux for Bioinformatics**

Linux is the foundation of most bioinformatics pipelines due to its open-source nature, stability, powerful command-line interface, and compatibility with virtually all major bioinformatics tools. In transcriptomics and other omics-based research, Linux provides a flexible and efficient environment for managing data, automating repetitive tasks, installing and running bioinformatics software, and handling large-scale sequence files.

**3.2.1 Importance of Linux in Bioinformatics**

Biological datasets, especially those derived from high-throughput sequencing experiments like RNA-Seq, often involve files that are several gigabytes in size. Graphical operating systems struggle with such data, while Linux excels in managing, processing, and analyzing it through command-line tools. Most bioinformatics tools, including HISAT2, SAMtools, and FeatureCounts, are designed for or run best in a Unix/Linux environment. Linux scripting also allows for process automation and reproducibility of complex pipelines.

**3.2.2 Linux Fundamentals and Applications**

**3.2.2.1 Directory Navigation and Management**

➤ View Current Path

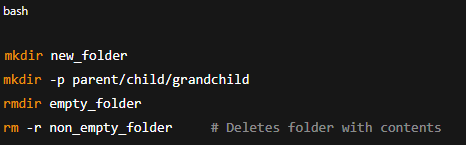


➤ List Contents in a Directory

A black screen with white text

AI-generated content may be incorrect.

➤ Create and Remove Directories



➤ Navigate the File System

A screenshot of a computer screen

AI-generated content may be incorrect.

**3.2.2.2 File Operations**

➤ Create Files

A computer screen shot of a computer screen

AI-generated content may be incorrect.

➤ Move and Rename Files

A computer screen with white text

AI-generated content may be incorrect.

➤ Copy Files and Folders

A computer screen with white text

AI-generated content may be incorrect.

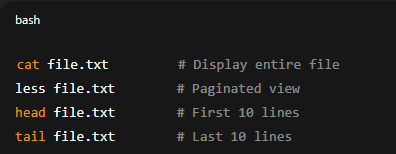
➤ Delete Files

A black screen with white text

AI-generated content may be incorrect.

**3.2.2.3 Viewing and Searching Files**

➤ View File Contents



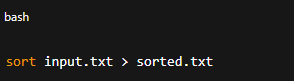
➤ Search Inside Files

A black background with white text

AI-generated content may be incorrect.

**3.2.2.4 Sorting, Counting, and Word Utilities**

➤ Sort and Save to New File



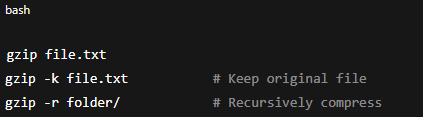
➤ Count Lines, Words, and Characters.

A black background with white text

AI-generated content may be incorrect.

**3.2.2.5 File Compression and Decompression**

➤ Compress Files Using gzip



➤ Decompress Files

A black screen with white text

AI-generated content may be incorrect.

**3.2.2.6 Shell Productivity Shortcuts**

|  |  |
| --- | --- |
| Shortcut | Description |
| Ctrl + A | Move to start of line |
| Ctrl + E | Move to end of line |
| Ctrl + K | Delete to end of line |
| Ctrl + W | Delete previous word |
| Ctrl + C | Terminate a command |
| Ctrl + L | Clear terminal |
| Ctrl + R | Reverse search through history |
| Tab | Auto-complete command or filename |
| ↑ / ↓ | Scroll through previous commands |

3.2.2.7 Bash History and Recall

A screenshot of a computer screen

AI-generated content may be incorrect.

**3.3 Fundamentals of R Programming for Bioinformatics**

R is a powerful, open-source programming language widely used in statistics, data analysis, and bioinformatics. Its extensive package ecosystem, especially those from Bioconductor, makes it the preferred tool for gene expression studies, including differential gene expression (DGE) analysis using RNA-Seq data.

During this internship, R was used for:

* Handling and cleaning count data
* Statistical modeling of differential expression using DESeq2
* Visualizing gene expression through plots like heatmaps, PCA, and volcano plots

This section introduces the fundamentals of R, the environment setup, core data structures, commonly used commands.

**3.3.1 Getting Started with R**

R programs can be written and executed using the R Console or IDEs such as RStudio, which offers a user-friendly interface for script writing, object viewing, and plotting.

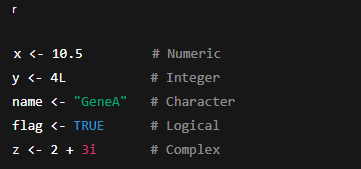
A black screen with white text

AI-generated content may be incorrect.

Comments in R are made using the # symbol.

**3.3.2 Variables and Data Types**

|  |  |  |
| --- | --- | --- |
| Data Type | Description | Example |
| Numeric | Real numbers | x <- 10.5 |
| Integer | Whole numbers with L suffix | y <- 4L |
| Character | Text | name <- "GeneA" |
| Logical | Boolean values | flag <- TRUE |
| Complex | Complex numbers | z <- 2 + 3i |



**3.3.3 Operators**

|  |  |  |
| --- | --- | --- |
| Operator Type | Operators | Example |
| Arithmetic | +, -, \*, /, ^, %% | 10 %% 3 → 1 |
| Relational | ==, !=, >, <, >= | x > y |
| Logical | &, ` | , !` |
| Special | :, %in% | 2 %in% c(1,2,3) |

A black background with white text

AI-generated content may be incorrect.

**3.3.4 Data Input and Output**

|  |  |  |
| --- | --- | --- |
| Function | Purpose | Example |
| readline() | Takes single user input | x <- readline("Enter value: ") |
| scan() | Takes multiple values | vals <- scan() |
| read.csv() | Reads dataset from CSV file | data <- read.csv("file.csv") |
| View() | Opens spreadsheet view in IDE | View(data) |

A screenshot of a computer screen

AI-generated content may be incorrect.

**3.3.5 Mathematical and Statistical Functions**

|  |  |  |  |
| --- | --- | --- | --- |
| Function | Purpose | Example | Output |
| round(x, n) | Rounds to n decimal places | round(3.14159, 2) | 3.14 |
| log10(x) | Base-10 logarithm | log10(1000) | 3 |
| mean() | Mean of numeric vector | mean(c(1,2,3)) | 2 |
| sd() | Standard deviation | sd(c(1,2,3,4,5)) | 1.58 |
| sample() | Random sample from a vector | sample(1:10, 3) | Varies |
| set.seed() | Ensures reproducible random values | set.seed(42) | – |

R offers built-in functions for mathematical and statistical calculations: A screenshot of a computer

AI-generated content may be incorrect.

Random number generation and sampling are also supported:

A black background with white text

AI-generated content may be incorrect.

**3.3.6 Data Structures in R**

a. Vectors

Vectors are 1D homogeneous data structures and are the most basic data type in R.

A screen shot of a computer

AI-generated content may be incorrect.

b. Lists

Lists can hold heterogeneous elements.

A black background with white text

AI-generated content may be incorrect.

c. Matrices and Arrays

Matrices are 2D homogeneous structures:

A black background with white text

AI-generated content may be incorrect.

Arrays can have more than 2 dimensions:

A black background with colorful numbers and symbols

AI-generated content may be incorrect.

d. Data Frames

Data frames are 2D heterogeneous tabular structures (like spreadsheets):

A black background with white text

AI-generated content may be incorrect.

e. Factors

Factors represent categorical data:

A black screen with white text

AI-generated content may be incorrect.

**3.3.7 Control Structures**

R supports conditional and iterative control structures.

A computer screen shot of a program code

AI-generated content may be incorrect.

**3.3.8 Functions in R**

Functions are used to encapsulate reusable code.

A computer screen with white text and blue text

AI-generated content may be incorrect.

**3.3.9 Statistical Analysis and Hypothesis Testing**

|  |  |  |
| --- | --- | --- |
| Function | Purpose | Example |
| mean() | Average | mean(c(1,2,3)) |
| median() | Median | median(c(1,2,3)) |
| sd() | Standard Deviation | sd(c(1,2,3)) |
| var() | Variance | var(c(1,2,3)) |
| summary() | Summary stats | summary(dataframe) |

a. Descriptive Statistics

A black background with white text

AI-generated content may be incorrect.

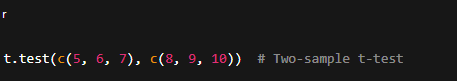
b. Probability Distributions

|  |  |  |
| --- | --- | --- |
| Function | Distribution Type | Example |
| rnorm() | Normal distribution | rnorm(5, mean=0, sd=1) |
| runif() | Uniform distribution | runif(5) |
| rbinom() | Binomial distribution | rbinom(5, 10, 0.5) |

A black background with white text

AI-generated content may be incorrect.

c. Hypothesis Testing



**3.3.9 Base R Plotting with plot()**

Base R's plot() function is a versatile tool to quickly visualize data.

|  |  |
| --- | --- |
| Parameter | Description |
| x, y | Data vectors |
| type | Plot type: "p", "l", "b", etc. |
| main | Title of the plot |
| xlab | X-axis label |
| ylab | Y-axis label |
| col | Color of points or lines |
| pch | Point shape |
| lty | Line type (solid, dashed, etc.) |

|  |  |
| --- | --- |
| Function | Purpose |
| points() | Add more points |
| lines() | Add new lines |
| abline() | Add reference lines |
| legend() | Add legends |
| text() | Add labels |

Types of Graphs in Base R

|  |  |  |
| --- | --- | --- |
| Graph Type | Function | Description |
| Scatter | plot() | Point graph of x vs y |
| Line | plot(type="l") | Connects points with lines |
| Histogram | hist() | Frequency distribution |
| Bar Plot | barplot() | Heights of categories |
| Boxplot | boxplot() | Median, IQR, and outliers |
| Pie Chart | pie() | Proportional distribution |
| Curve | curve() | Mathematical function plot |

A black screen with white text

AI-generated content may be incorrect.

a. Line Graph

A black background with white text

AI-generated content may be incorrect.

b. Histogram

A black background with colorful text

AI-generated content may be incorrect.

c. Bar Plot

A computer code on a black background

AI-generated content may be incorrect.

d. Pie Chart

A computer screen with white text

AI-generated content may be incorrect.

**3.3.10 Advanced Plotting with ggplot2**

ggplot2 follows the Grammar of Graphics and allows building plots layer by layer.

Components of ggplot2

|  |  |  |
| --- | --- | --- |
| Component | Role | Example |
| data | Dataset | mtcars |
| aes() | Aesthetic mapping | aes(x=mpg, y=wt) |
| geom\_\* | Plot type (point, line, bar, etc.) | geom\_point(), geom\_line() |
| facet\_\* | Split plots by a factor | facet\_wrap(~cyl) |
| theme\_\* | Style the plot | theme\_minimal(), theme\_bw() |

🔹 Example: Scatter Plot

A computer screen shot of a black screen with white text

AI-generated content may be incorrect.

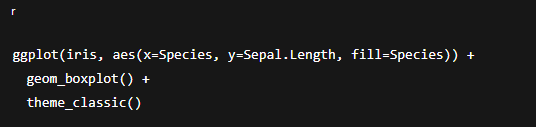
🔹 Common Geoms

|  |  |
| --- | --- |
| geom\_\* Function | Description |
| geom\_point() | Scatter plot |
| geom\_line() | Line plot |
| geom\_bar() | Bar plot |
| geom\_boxplot() | Boxplot |
| geom\_histogram() | Histogram |
| geom\_smooth() | Add regression line |

**3.3.11 Plot Customization in ggplot2**

|  |  |
| --- | --- |
| Function / Argument | Description |
| labs() | Add title and axis labels |
| scale\_color\_manual() | Custom color mapping |
| facet\_wrap(~var) | Split into panels by variable |
| theme\_minimal() | Use minimalist theme |
| aes(color=group) | Color by grouping variable |

Example: Grouped Boxplot

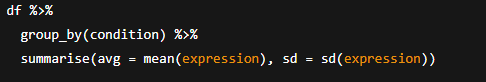


**3.3.12 Data Manipulation with dplyr**

dplyr simplifies data manipulation using intuitive functions called verbs.

|  |  |  |
| --- | --- | --- |
| Function | Purpose | Example |
| filter() | Filter rows | filter(df, value > 10) |
| select() | Select specific columns | select(df, gene, expr) |
| mutate() | Create or modify columns | mutate(df, log\_expr = log2(expr)) |
| arrange() | Sort rows | arrange(df, desc(expr)) |
| summarise() | Summarize values | summarise(df, mean = mean(expr)) |
| group\_by() | Group before summarizing | group\_by(df, condition) |

🔹 Example: Grouped Summary



**3.3.13 Reshaping Data with tidyr**

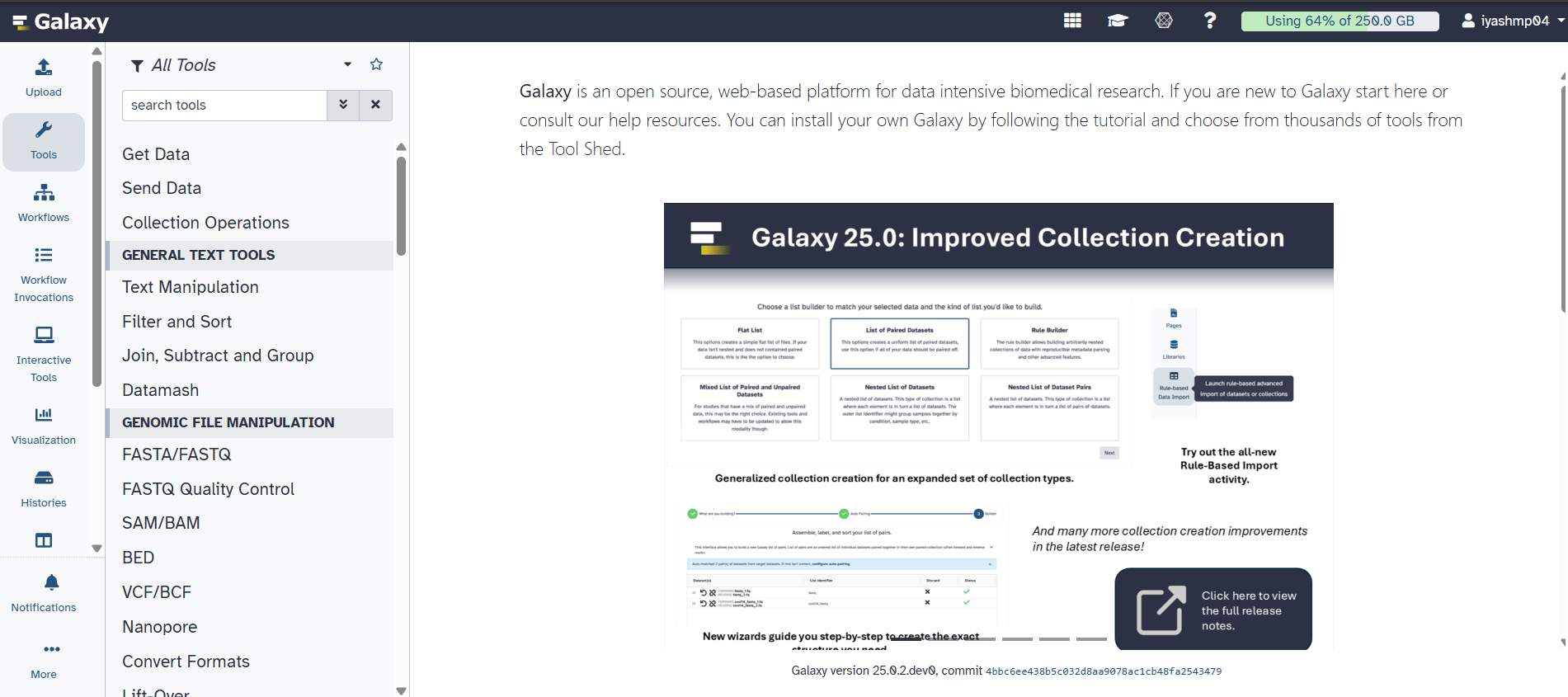
tidyr is used for tidy data formatting—reshaping, separating, combining columns.

|  |  |  |
| --- | --- | --- |
| Function | Purpose | Example |
| pivot\_longer() | Wide → Long format | pivot\_longer(df, cols=2:4) |
| pivot\_wider() | Long → Wide format | pivot\_wider(df) |
| separate() | Split a column | separate(df, col, into=c("gene", "type")) |
| unite() | Combine multiple columns | unite(df, new, c(col1, col2)) |
| drop\_na() | Remove rows with missing data | drop\_na(df) |

**3.4: Galaxy Platform – Fundamentals and Usage**

**3.4.1 Introduction to Galaxy**

Galaxy is a web-based, open-source platform designed to make computational biology accessible to researchers without requiring advanced programming skills. It provides a graphical user interface (GUI) where users can analyze data through pre-built bioinformatics tools and workflows. Galaxy is especially useful in transcriptomics studies due to its ability to perform quality control, read alignment, quantification, and more.



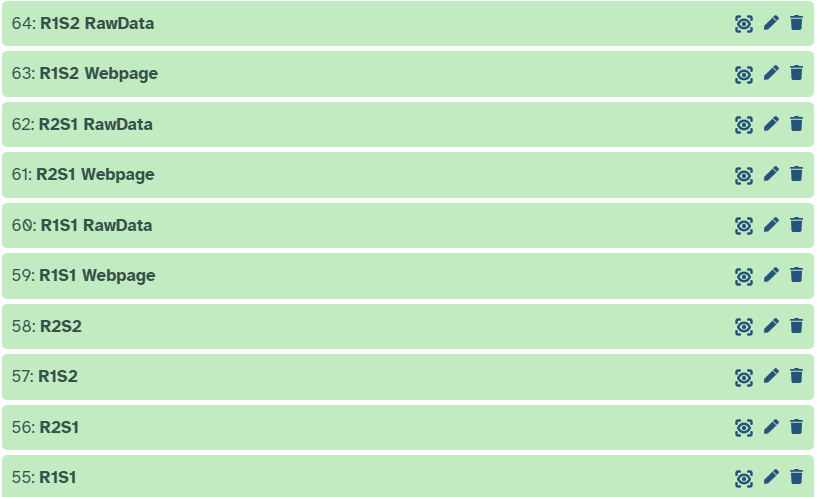
**3.4.2 Key Features of Galaxy**

|  |  |
| --- | --- |
| Feature | Description |
| Web-based Access | Requires only a browser, no local installation |
| Tool Integration | Supports hundreds of tools (FastQC, Trimmomatic, HISAT2, etc.) |
| Workflow System | Allows chaining tools into reusable analysis pipelines |
| Reproducibility | Tracks full analysis history, parameters, and versions |
| User-Friendly | No command-line required; intuitive drag-and-drop interface |
| Data Management | Organizes datasets and results in histories |

**3.4.3 Role of Galaxy in Transcriptomic Analysis**

In this internship project, Galaxy was used primarily for RNA-Seq preprocessing. The analysis steps carried out on Galaxy included:

1. Uploading raw FASTQ files
2. Performing quality control with FastQC
3. Trimming reads using Trimmomatic
4. Aligning reads to the reference genome using HISAT2
5. Generating count matrices using FeatureCounts
6. Exporting results for further analysis in R



A screenshot of a computer

AI-generated content may be incorrect.

A green and white text with black text

AI-generated content may be incorrect.

A green and white text box

AI-generated content may be incorrect.

**3.4.4 Galaxy Interface Components**

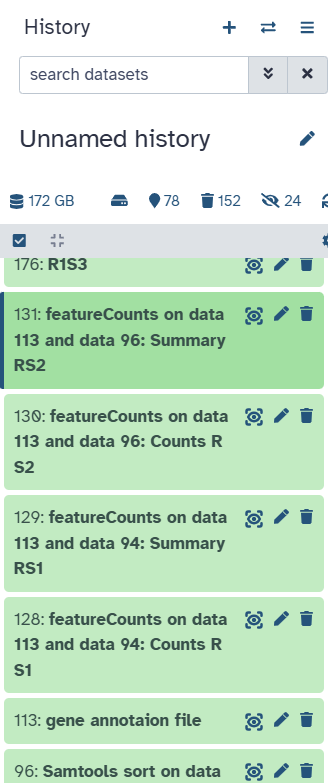
|  |  |
| --- | --- |
| Panel | Function |
| Tool Panel | Located on the left; lists available tools organized by category |
| Main Panel | Center area; displays forms, results, and tool outputs |
| History Panel | Right-hand side; shows all datasets, tool runs, and outputs for reproducibility |

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.



**3.4.5 Exporting Output for DESeq2 Analysis in R**

Once read counts were obtained using FeatureCounts, the final count matrix file (usually .txt or .tabular) was downloaded from Galaxy and used as input for downstream analysis in R, especially for differential gene expression using DESeq2.

Steps:

* Click on dataset in history → “Download”
* Optionally convert .tabular to .csv or .txt using Galaxy’s convert tools or manually

**3.4.6 Advantages of Galaxy Platform**

|  |  |
| --- | --- |
| Feature | Benefit |
| No coding needed | Great for beginners and life science researchers |
| Traceable | Maintains full record of parameters, tool versions, and datasets |
| Community support | Thousands of published workflows and public servers |
| Built-in tools | Hundreds of integrated tools for genomics and transcriptomics |

**3.4.7 Limitations Observed**

|  |  |
| --- | --- |
| Limitation | Description |
| Tool version differences | Some public Galaxy servers may use outdated tools |
| Internet dependency | Requires a stable internet connection unless Galaxy is installed locally |
| Less flexible for scripting | Custom advanced scripting is not directly supported |
| Limited resource on public server | Large datasets may take longer due to server load limitations |

**3.5 Bioinformatics Databases and Browsers**

Bioinformatics databases and genome browsers are essential digital platforms that store, organize, and provide access to biological information such as DNA/RNA sequences, gene expression profiles, protein structures, and genome annotations. These resources play a vital role in genomic and transcriptomic studies by enabling researchers to retrieve relevant datasets, explore gene features, and verify expression data.

During this internship, several publicly available bioinformatics databases and expression browsers were used to access RNA-Seq datasets, explore gene annotations, download reference genomes, and interpret gene expression patterns. The following are the primary resources utilized:

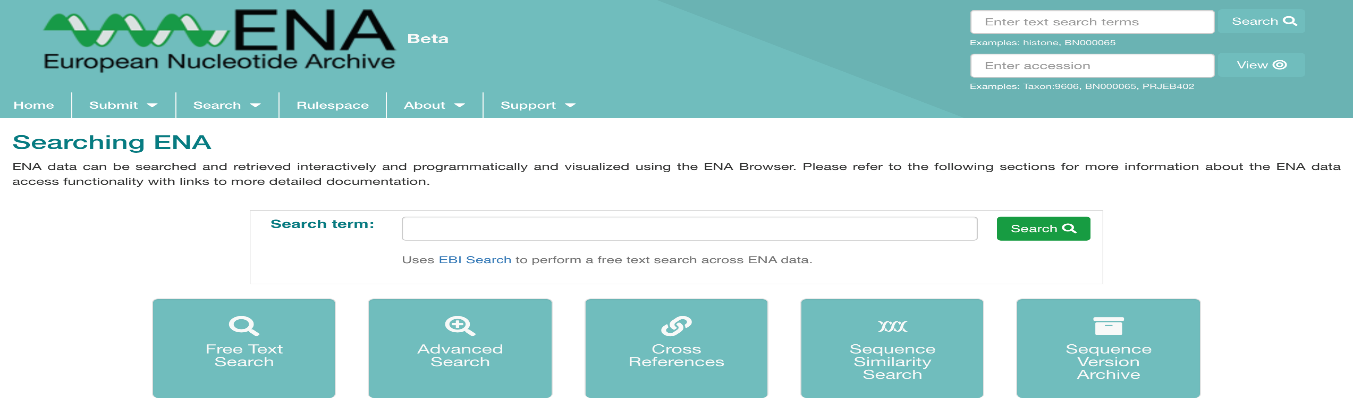
**3.5.1 ENA (European Nucleotide Archive)**

* Website: https://www.ebi.ac.uk/ena
* Managed by: EMBL-EBI (European Molecular Biology Laboratory - European Bioinformatics Institute)

The ENA is a comprehensive repository of nucleotide sequence data from all domains of life. It supports raw sequencing submissions and makes them publicly available for research. It includes both raw reads (FASTQ) and annotated sequences (FASTA, GTF).

Usage in this project:

* The raw RNA-Seq datasets (ERR1659927, ERR1659930) were downloaded from the ENA browser.
* The platform provided metadata, sequencing platform info, sample conditions, and accession IDs required for downstream analysis.
* *Example*: Navigated to ENA → Search with accession number → Downloaded FASTQ files for analysis.



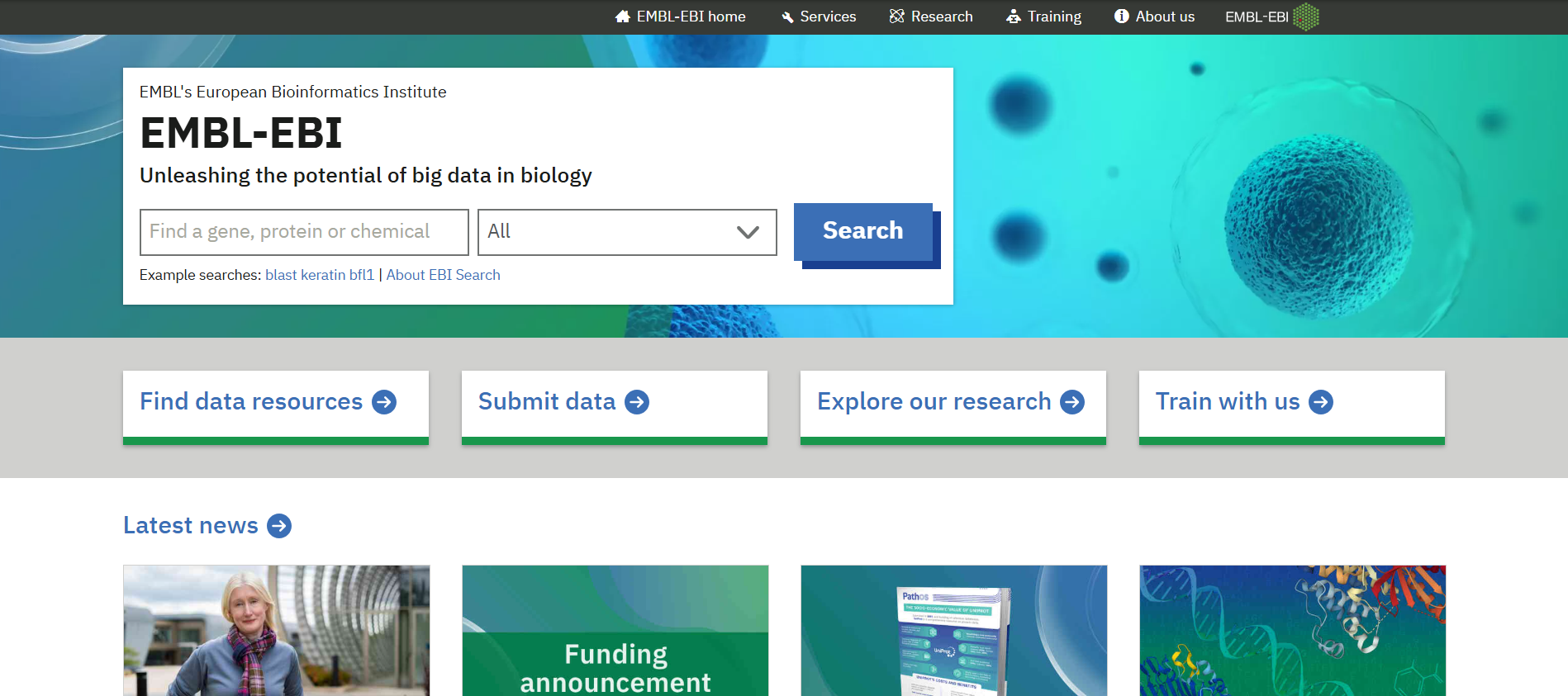
**3.5.2 EMBL-EBI (European Bioinformatics Institute)**

* Website: https://www.ebi.ac.uk
* Purpose: A central hub offering multiple tools and databases for molecular biology, including ENA, Ensembl, Expression Atlas, InterPro, and more.

EMBL-EBI serves as the parent organization for several important bioinformatics resources. It provides freely accessible data services for DNA/RNA sequencing, protein functions, gene expression, and pathway analysis.

Usage in this project:

* Used as a gateway to access Expression Atlas and ENA.
* Supported genome browsing, annotation retrieval, and visualization of transcriptomic data.
* *Used as the main source for navigating between related tools during gene expression interpretation.*



**3.5.3 NCBI (National Center for Biotechnology Information)**

* Website: https://www.ncbi.nlm.nih.gov
* Managed by: U.S. National Library of Medicine

NCBI hosts several biological databases including GenBank, GEO (Gene Expression Omnibus), RefSeq, and BLAST tools. It is widely used for sequence searches, gene annotation, reference genome downloads, and data cross-verification.

Usage in this project:

* Verified gene identifiers and annotation files for compatibility with RNA-Seq analysis.
* Explored gene features using Gene and Genome databases.
* Referred to BLAST tools when needed for sequence similarity checks.

A screenshot of a computer

AI-generated content may be incorrect.

**3.5.4 Expression Atlas**

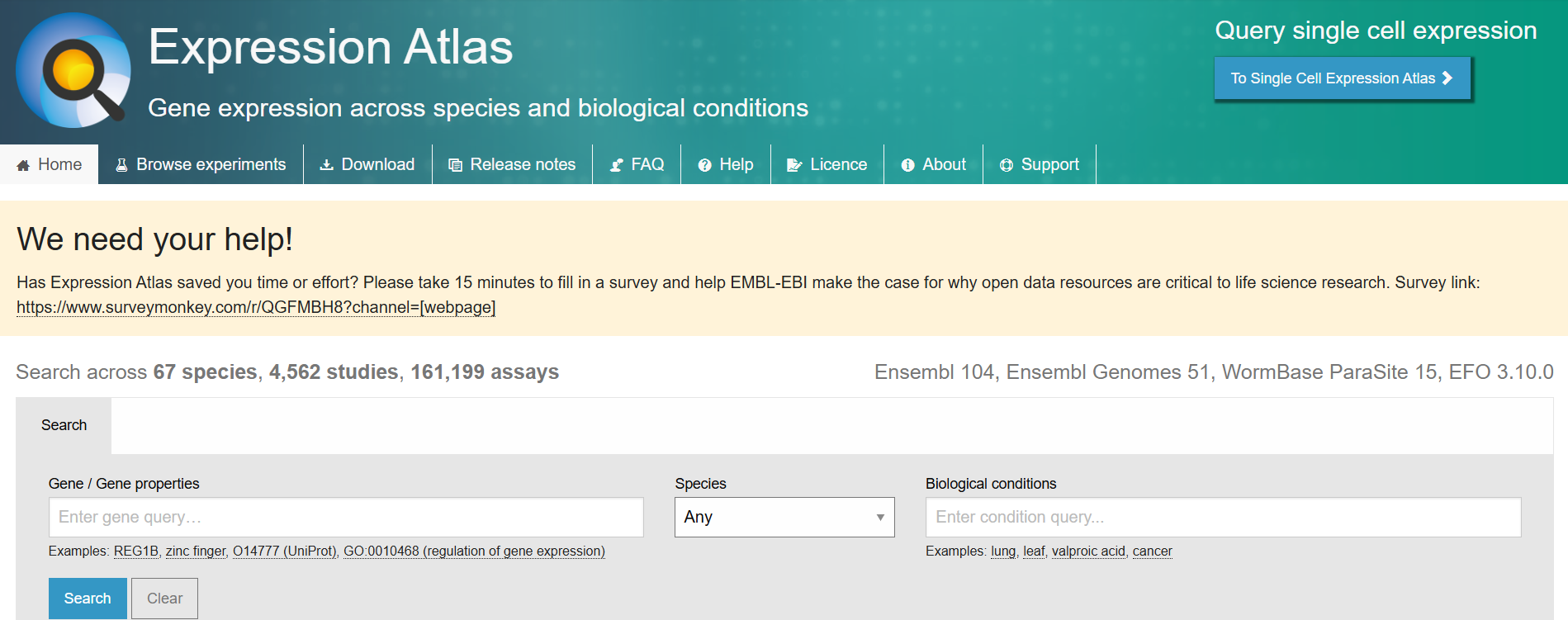
* Website: https://www.ebi.ac.uk/gxa
* Hosted by: EMBL-EBI

Expression Atlas provides information about gene and transcript expression levels across various biological conditions, tissues, and organisms. It integrates experimental data from RNA-Seq, microarrays, and proteomics studies.

Usage in this project:

* Used to explore expression levels of differentially expressed genes identified in our samples.
* Helped in confirming tissue-specific and condition-specific expression patterns.
* Provided visual plots of expression variation for selected genes.

*Example*: After DESeq2 analysis in R, genes like XYZ1 were checked in Expression Atlas for baseline expression in rice or Arabidopsis tissues.



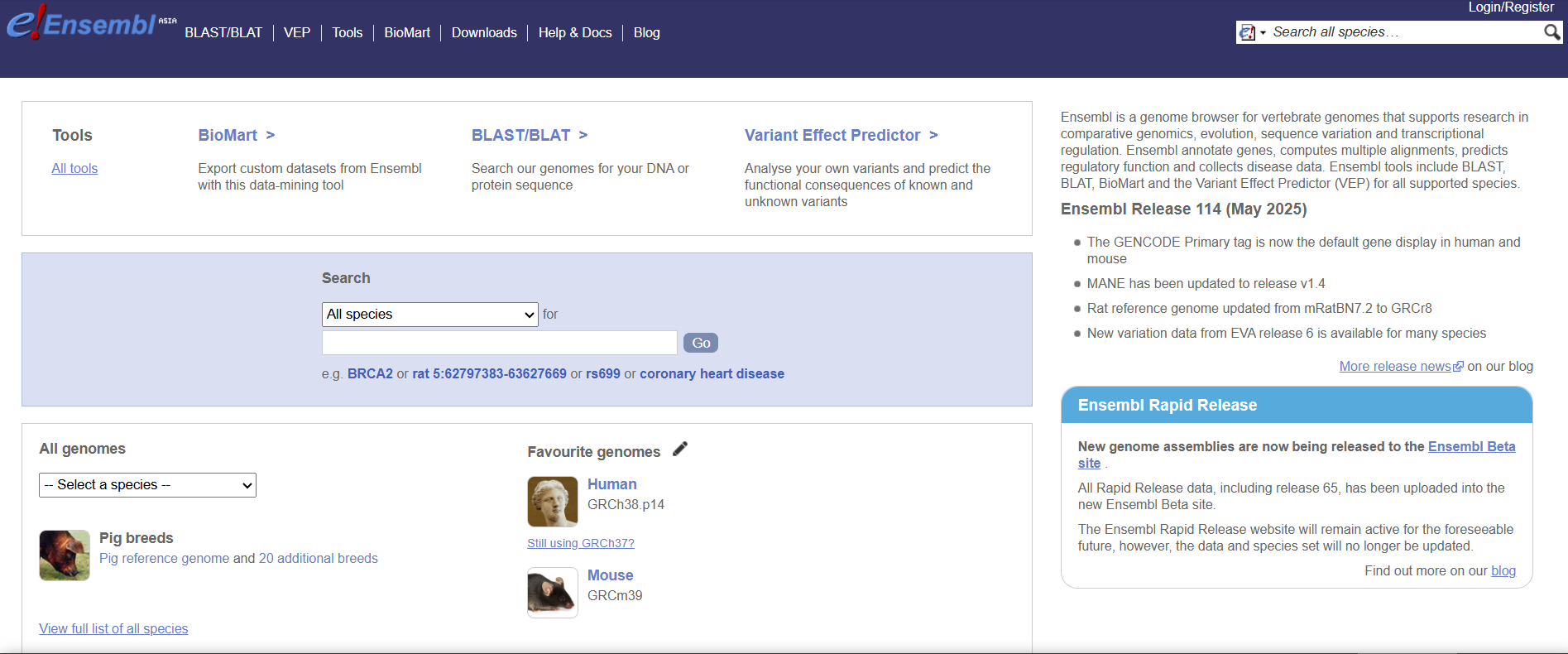
**3.5.5 Ensembl Genome Browser**

* Website: https://www.ensembl.org
* Hosted by: EMBL-EBI

The Ensembl Genome Browser allows users to view genome-wide data, including gene locations, exons, transcripts, and regulatory elements. It offers downloadable gene annotations (GTF files), protein coding data, and integrates information from multiple species.

Usage in this project:

* GTF annotation files required for FeatureCounts in Galaxy were downloaded from Ensembl.
* Gene ID mapping and chromosome locations were visualized and verified.
* Facilitated gene model understanding for accurate read counting.



**3.5.6 NCBI SRA (Sequence Read Archive)**

* Website: https://www.ncbi.nlm.nih.gov/sra
* Managed by: National Center for Biotechnology Information (NCBI)

The Sequence Read Archive (SRA) is one of the world’s largest repositories of high-throughput sequencing data. It stores raw reads generated from next-generation sequencing (NGS) experiments submitted by researchers worldwide. The data are stored in formats such as FASTQ, and often come with associated metadata including sample origin, library preparation, and sequencing platform.

Usage in this project:

* Accessed to search and retrieve raw RNA-Seq datasets based on BioProject, BioSample, or Run accessions.
* For example, datasets like ERR1659927 and ERR1659930 were accessed through SRA.
* Provided direct download links and FASTQ conversion via ENA or SRA Toolkit.

📌 *Example Process*:

* Searched for a study or run ID on SRA
* Reviewed metadata (organism, condition, sequencing type)
* Retrieved download links for use in Galaxy or Linux-based workflows

A screenshot of a computer

AI-generated content may be incorrect.

**Chapter 4: Linux Techniques and Applications**

**4.1 Remote BLAST Execution via NCBI Using Linux Terminal**

During the course of this project, remote BLAST functionality provided by NCBI was also utilized. This approach is especially useful in scenarios where users do not possess a local reference database or wish to save storage and computational resources. The remote BLAST option allows users to submit their queries directly to NCBI's BLAST servers from the Linux command line.

**4.1.1 Preparation of Query Sequence**

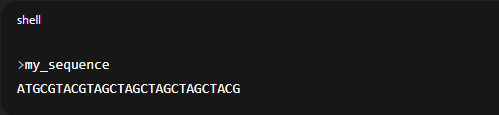
A nucleotide query sequence was created and saved in FASTA format using the terminal-based text editor nano. The steps followed were:

1. Opening a new file:

A black background with white text

AI-generated content may be incorrect.

2.Pasting the following sample sequence into the file:



3. Saving the file by pressing Ctrl+O, then Enter, and exiting with Ctrl+X.

This process created a valid FASTA file named query.fasta, ready to be submitted for BLAST analysis.

**4.1.2 Performing Remote BLAST using NCBI Servers**

To run a remote BLAST search using the created query, the following command was executed from the terminal:

A black background with white text

AI-generated content may be incorrect.

Explanation of parameters:

|  |  |
| --- | --- |
| Parameter | Description |
| -query query.fasta | Specifies the input file containing the query sequence. |
| -db nt | Uses the NCBI nucleotide database (nt) as the search database. |
| -remote | Instructs the tool to execute the BLAST search on NCBI’s remote servers. |
| -out results.txt | Saves the output in a local file named results.txt. |
| -outfmt 0 | Generates output in the standard human-readable format (default). |

This method requires an active internet connection but eliminates the need for downloading and maintaining large reference databases locally.

**4.2 RNA-Seq Quality Assessment using FASTQC (ERR1659930 Sample)**

FASTQC is a widely used quality control (QC) tool designed to evaluate high-throughput sequencing data, particularly in FASTQ format. Developed by Babraham Bioinformatics, it provides a quick overview of read quality and helps identify potential issues such as adapter contamination, sequence duplication, poor base quality, and GC bias.

RNA-Seq and other next-generation sequencing (NGS) workflows generate large volumes of data, where even subtle quality problems can affect downstream analyses like alignment, quantification, and differential expression. FASTQC addresses this by offering both summary statistics and detailed visualizations across multiple QC modules.

The tool is compatible with both single-end and paired-end sequencing data and can be executed via a graphical user interface (GUI) or from the command line, making it suitable for integration into automated pipelines. In this project, FASTQC was used in command-line mode within the Linux environment for assessing the quality of raw RNA-Seq reads.

Key features of FASTQC include:

* Rapid processing of FASTQ files
* Interactive HTML reports with modular results
* Warnings for potential quality issues
* Easy identification of contamination or technical artifacts

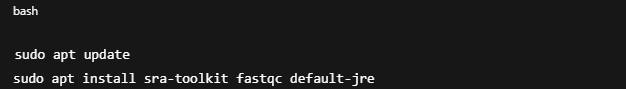
The insights provided by FASTQC are essential for making informed decisions about trimming, filtering, and overall data usability, thereby ensuring the integrity of downstream bioinformatics analyses.

**4.2.1 Tools and Environment**

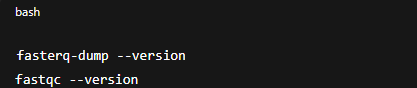
The following command-line tools were used in this QC pipeline:

|  |  |
| --- | --- |
| Tool | Purpose |
| fasterq-dump | To download .fastq files from NCBI SRA |
| fastqc | To assess sequence quality metrics |

These tools were installed using the system package manager:



Version verification commands:



**4.2.2 Directory Setup**

A dedicated working directory was created to store the project-specific files:

A black screen with white text

AI-generated content may be incorrect.

**4.2.3 Downloading RNA-Seq Reads from NCBI**

The paired-end reads for sample ERR1659930 were downloaded using the following command:

A black background with white text

AI-generated content may be incorrect.

This generated two .fastq files:

* ERR1659930\_1.fastq – forward reads
* ERR1659930\_2.fastq – reverse reads

**4.2.4 Quality Assessment using FASTQC**

To evaluate the quality of the raw reads, FASTQC was run on both FASTQ files:

A black background with white text

AI-generated content may be incorrect.

This produced the following outputs:

* ERR1659930\_1\_fastqc.html
* ERR1659930\_2\_fastqc.html

These .html files were then opened in a web browser for inspection:

A computer screen shot of a computer code

AI-generated content may be incorrect.

**4.3 Linux Shell Scripting for Genomic Data Analysis.**

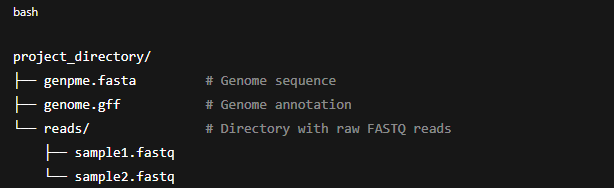
The purpose of this section is to automate the analysis of genomic and sequencing data using a custom shell script named bio\_analysis.sh. The script performs integrated operations on the following data types:

* Genomic sequences in FASTA format (genpme.fasta)
* Gene features in a GFF file (genome.gff)
* RNA-Seq data in FASTQ files located in a reads/ directory

The script also supports optional command-line flags to tailor the analysis scope and output reporting.

**4.3.1 File and Directory Structure**

Before executing the script, ensure the following directory layout is in place:



**4.3.2 Script Functionalities**

The bio\_analysis.sh script performs the following tasks:

|  |  |
| --- | --- |
| Step | Functionality |
| 1 | Parse command-line arguments (--report, --only-genome) |
| 2 | Validate presence of required files |
| 3 | Count number of sequences and calculate genome statistics |
| 4 | Extract gene and CDS counts from GFF |
| 5 | Count reads in all .fastq files (unless --only-genome is specified) |
| 6 | Compile and output a summary |
| 7 | Optionally save the summary to a report file |

**4.3.3 The Shell Script: bio\_analysis.sh**

A screenshot of a computer program

AI-generated content may be incorrect.

A computer screen shot of a black screen

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

**4.3.4 Execution Instructions**

Make the script executable:

A black background with white text

AI-generated content may be incorrect.

Example usage:

|  |  |
| --- | --- |
| Command | Description |
| ./bio\_analysis.sh | Analyze genome, GFF, and FASTQ files in the current directory |
| ./bio\_analysis.sh /path/to/data | Analyze files in a specific directory |
| ./bio\_analysis.sh --only-genome | Skip FASTQ analysis, only analyze genome and GFF |
| ./bio\_analysis.sh /path/to/data --report summary.txt | Save summary to summary.txt |

**CHAPTER 5: Galaxy Platform and Preprocessing Workflow**

**5.1 Introduction to Galaxy Platform**

Galaxy is an open-source, web-based platform designed to facilitate accessible, reproducible, and transparent computational research in life sciences. It enables researchers—especially those without advanced programming skills—to perform complex bioinformatics workflows through a graphical user interface. Galaxy provides built-in support for uploading data, selecting tools, customizing parameters, viewing intermediate outputs, and managing complete pipelines for NGS data analysis.

In the context of this RNA-Seq project, the Galaxy Europe server (*https://usegalaxy.eu*) was used to execute all preprocessing steps including quality check, adapter trimming, alignment, and read counting, preparing the data for downstream differential gene expression analysis in R.

Key Advantages of Galaxy Platform:

* No need for command-line knowledge
* Centralized tool repository
* Transparent workflows (easily documented)

**5.2 Data Upload and Organization**

* In this study, RNA-Seq data was utilized to investigate the role of the *faint sausage (fas)* gene in *Drosophila melanogaster* embryogenesis, specifically focusing on pre-mRNA splicing. The study centered on embryos harboring the P218 mutation in the *fas* gene, compared against wild-type controls. The *fas* gene is known to play a role in early neural development, and the P218 mutation is hypothesized to influence transcript splicing during embryonic development. Thus, by comparing the transcriptomic landscape between the two conditions, the study aimed to identify differentially spliced or expressed genes potentially regulated by *fas*.
* To perform this comparative analysis, raw sequencing data (FASTQ files) were downloaded from the NCBI SRA (Sequence Read Archive) and imported into the Galaxy Europe server (*https://usegalaxy.eu*) for preprocessing. Four RNA-Seq datasets were used—two replicates for each condition (mutant and wild-type)—to ensure reliability in downstream statistical analyses.
* Samples Used in This Study:

|  |  |  |
| --- | --- | --- |
| * Sample Name | * Genotype | * Description |
| * ERR1659925 | * Wild-Type | * Embryo sample without *fas* mutation |
| * ERR1659927 | * Wild-Type | * Embryo sample without *fas* mutation |
| * ERR1659929 | * P218 Mutant | * Embryo sample carrying P218 mutation in *fas* |
| * ERR1659930 | * P218 Mutant | * Embryo sample carrying P218 mutation in *fas* |

* Galaxy also allows metadata management, which was used to group and tag datasets by condition (Control vs. Mutant), simplifying downstream steps like alignment, counting, and differential expression analysis.
* Compatible with large datasets via FTP or direct import from NCBI/ENA

A screenshot of a computer

AI-generated content may be incorrect.

**5.3 Quality Control using FastQC**

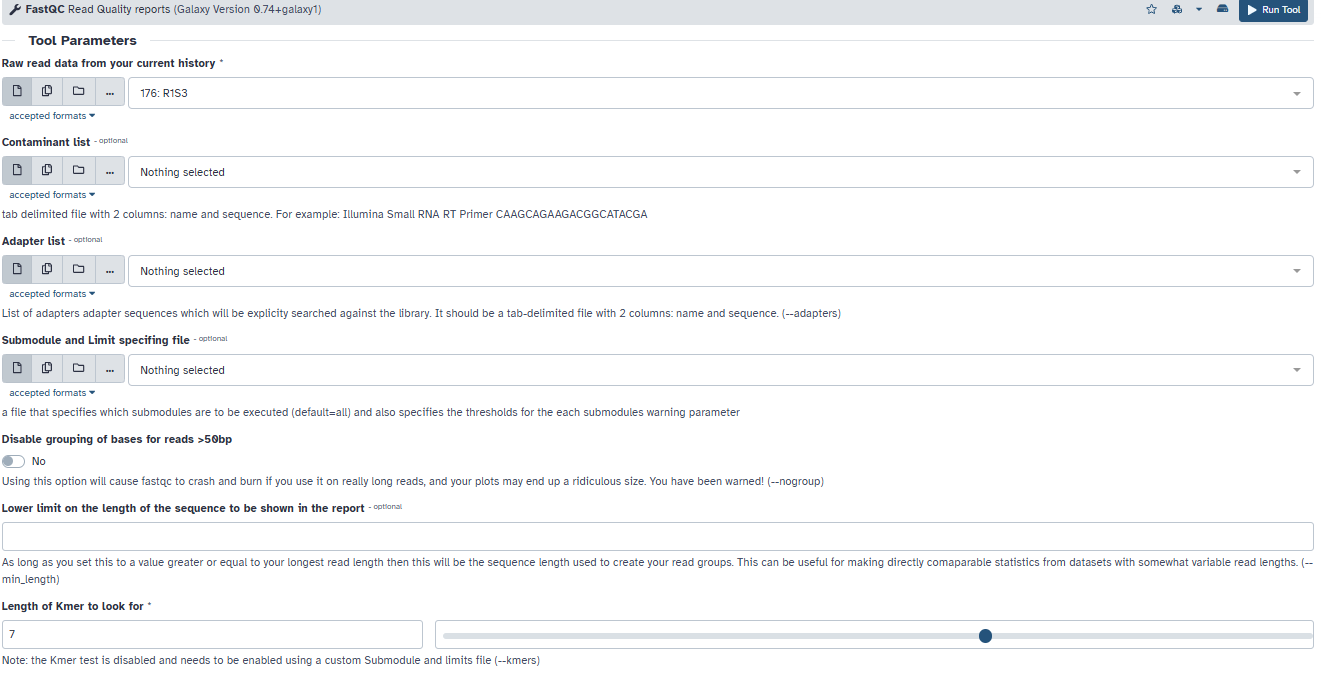
Before performing any downstream analyses, it is essential to assess the quality of raw sequencing reads. Low-quality bases, adapter contamination, and other biases can significantly affect the accuracy of read alignment and gene quantification. To address this, the tool FastQC was employed to generate detailed quality reports for each RNA-Seq dataset.

FastQC was run on all four uploaded FASTQ files using the "FastQC: Read Quality reports" tool on the Galaxy platform. This step provided an overview of the sequencing read quality and helped identify potential issues such as base-calling errors, sequencing biases, or adapter contamination.

🔍 Key Parameters Analyzed in FastQC Reports:

|  |  |
| --- | --- |
| Parameter | Description |
| Per Base Sequence Quality | Shows the quality score (Phred score) of bases across the read length. High-quality reads generally have scores above Q30. A drop at the ends may indicate sequencing degradation. |
| Per Sequence Quality Scores | Displays distribution of average quality scores per read. A normal distribution skewed toward high-quality reads is ideal. |
| Per Base Sequence Content | Measures the proportion of each base (A, T, G, C) across all sequences at each position. Large imbalances can indicate bias. |
| GC Content | Displays the GC% distribution of the reads. Deviations from the expected distribution may point to contamination or biased amplification. |
| Per Base N Content | Shows the proportion of ambiguous base calls (N) across all sequences. A high presence may indicate poor sequencing quality. |
| Sequence Length Distribution | Describes the range and uniformity of read lengths. Uniform read lengths (e.g., 100 bp) are expected in most modern platforms. |
| Sequence Duplication Levels | Indicates how many duplicate sequences exist. High duplication can suggest PCR artifacts or over-sequencing of a small library. |
| Overrepresented Sequences | Identifies sequences that occur unusually frequently. These are often adapter sequences or contaminants. |
| Adapter Content | Detects the presence of residual sequencing adapters. Adapter contamination must be removed before alignment. |

After FastQC execution, each sample produced a compressed HTML report and a raw data file. These reports were carefully examined to decide whether trimming or filtering was necessary.



**5.4 Adapter Removal using Cutadapt**

After performing initial quality control using FastQC, we identified mild adapter contamination in some of the RNA-Seq samples. Leftover adapters from library preparation can interfere with accurate read alignment and lead to biased quantification. Therefore, the first step in preprocessing was to remove adapter sequences using Cutadapt.

Tool Used:

* Cutadapt (Galaxy version 1.18)

Purpose:

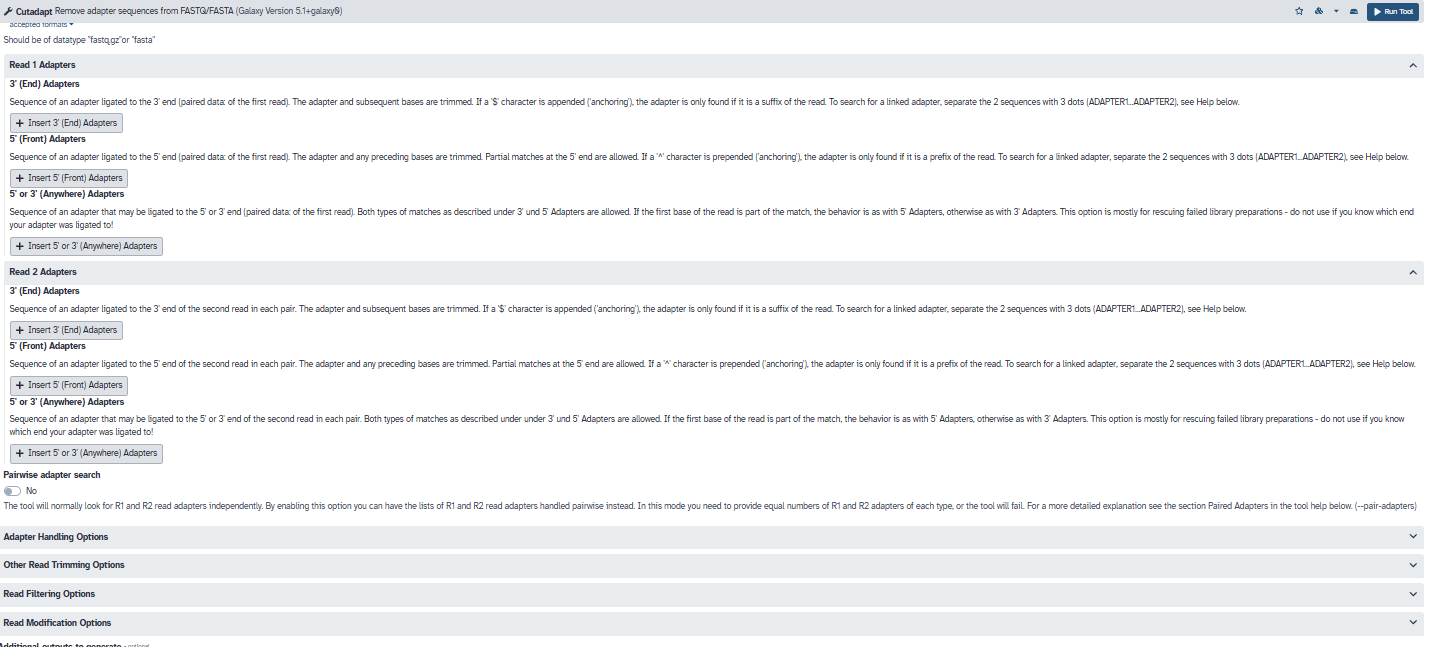
* Remove Illumina adapter sequences from the reads.
* Reduce interference during alignment and improve mapping accuracy.

⚙️ Key Settings in Galaxy:

|  |  |
| --- | --- |
| Parameter | Value |
| Adapter sequence | AGATCGGAAGAG (Illumina default) |
| Minimum read length after trimming | 20 bp |
| Discard reads with no adapter match | No |
| Quality trimming | Not applied at this stage (only adapter removal) |

Output Summary (Example from ERR1659930):

* Reads processed: 18,456,789
* Reads with adapters: 3,256,120 (17.64%)
* Reads retained: ~96%



**5.5 Quality Trimming using Trimmomatic**

Once adapter sequences were removed, the next step involved quality-based trimming using Trimmomatic, another widely used tool for preprocessing high-throughput sequencing data. Low-quality bases, especially at the ends, can lead to mismapped reads and misinterpretation during differential gene expression analysis.

Tool Used:

* Trimmomatic (Galaxy wrapper version 0.36.6)

Purpose:

* Trim low-quality bases from the ends of reads.
* Remove any reads that are too short or of poor quality.

⚙️ Key Settings in Galaxy:

|  |  |
| --- | --- |
| Parameter | Value |
| Sliding window trimming | 4:20 (4-base window; average quality ≥ 20) |
| Leading and trailing base removal | Leading: 3, Trailing: 3 |
| Minimum read length | 36 bp |
| Input | Cutadapt-cleaned reads |

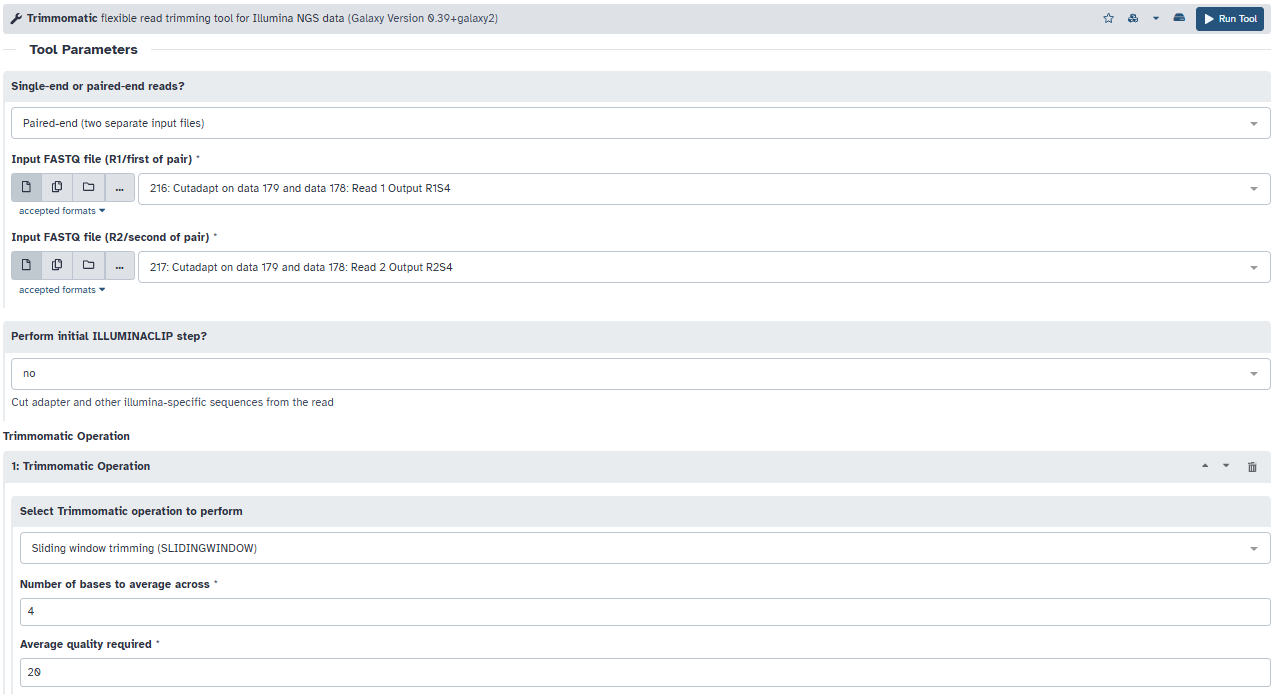
Output Summary (Example from ERR1659930):

* Input reads: 17,128,570
* Reads retained after trimming: ~95%
* Reads discarded due to short length: ~2–3%
* Improved base quality across entire read length.

FastQC Re-Evaluation

After trimming:

* FastQC was re-run to confirm improvements.
* Results showed:
  + Enhanced per-base quality scores
  + No residual adapter content
  + Balanced GC content



**5.6 Read Alignment using HISAT2**

After performing quality control and trimming, the next crucial step in the RNA-Seq data analysis pipeline is the alignment of reads to the reference genome. In this study, we used HISAT2, a widely used splice-aware aligner, to map the clean reads onto the *Drosophila melanogaster* reference genome.

Purpose of Alignment

The goal of read alignment is to determine the genomic origin of each sequencing read. In RNA-Seq, this step is critical for:

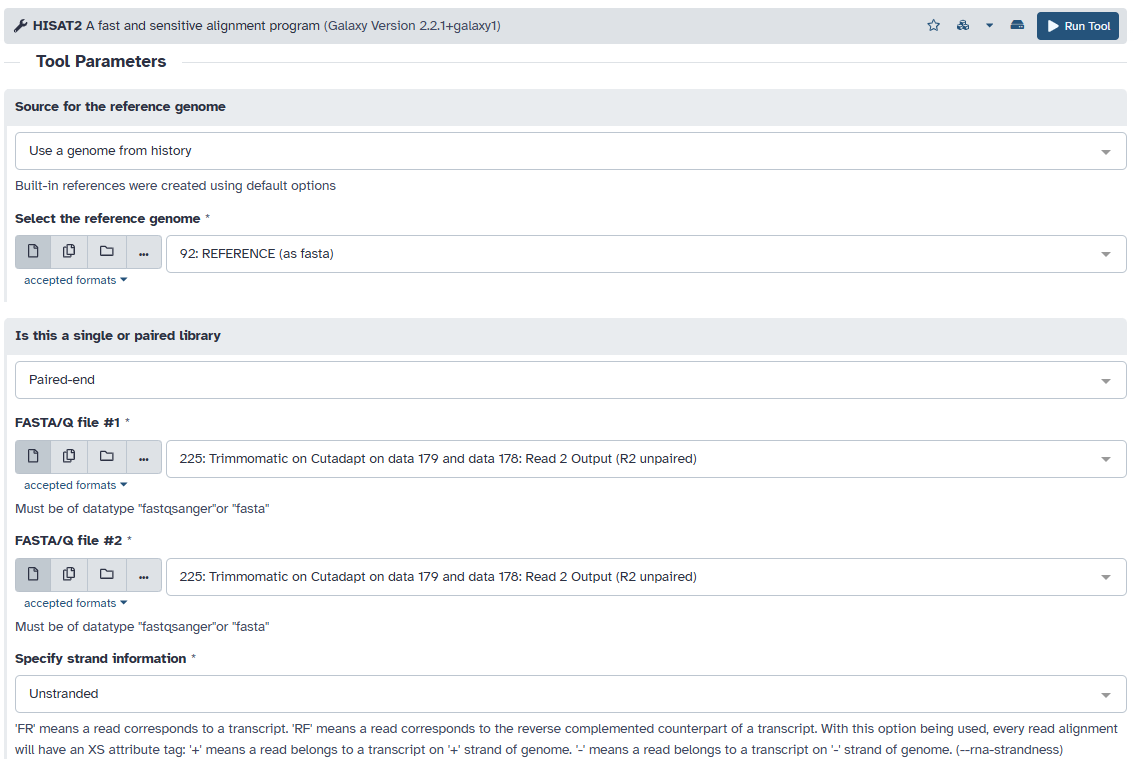
* Quantifying gene expression levels
* Detecting exon-intron boundaries
* Identifying novel splice junctions
* Enabling accurate downstream differential expression analysis

Tool Used: HISAT2

* HISAT2 (Hierarchical Indexing for Spliced Alignment of Transcripts) is optimized for speed and memory efficiency.
* It uses a graph-based genome index to support spliced alignments.
* Supports alignment of reads across exon-exon junctions, which is crucial for eukaryotic transcriptomes.

Steps Performed in Galaxy:

1. Reference Genome Indexing (if not already provided):
   * The reference genome (e.g., Drosophila melanogaster BDGP6) was either selected from pre-built indexes or uploaded and indexed using HISAT2 Build Index.
2. Run HISAT2 Alignment:
   * Tool used: HISAT2 (Galaxy Tool)
   * Inputs: Trimmed FASTQ files from Cutadapt/Trimmomatic and indexed reference genome.
   * Parameters:
     + Input Type: Single-end or paired-end depending on dataset
     + Splice-Aware Alignment: Enabled
     + Output Format: BAM file (Binary Alignment Map)
3. Outputs:
   * A .bam file for each sample containing aligned reads
   * Alignment summary showing:
     + Total reads
     + Uniquely mapped reads
     + Multiple mapped reads
     + Unmapped reads



**5.7 Sorting Aligned Reads using SAMtools Sort**

Purpose

Once reads are aligned to the reference genome using HISAT2, the resulting output is typically in SAM or BAM format. These files are unsorted, which means reads are arranged in the order they were processed, not by genomic location.

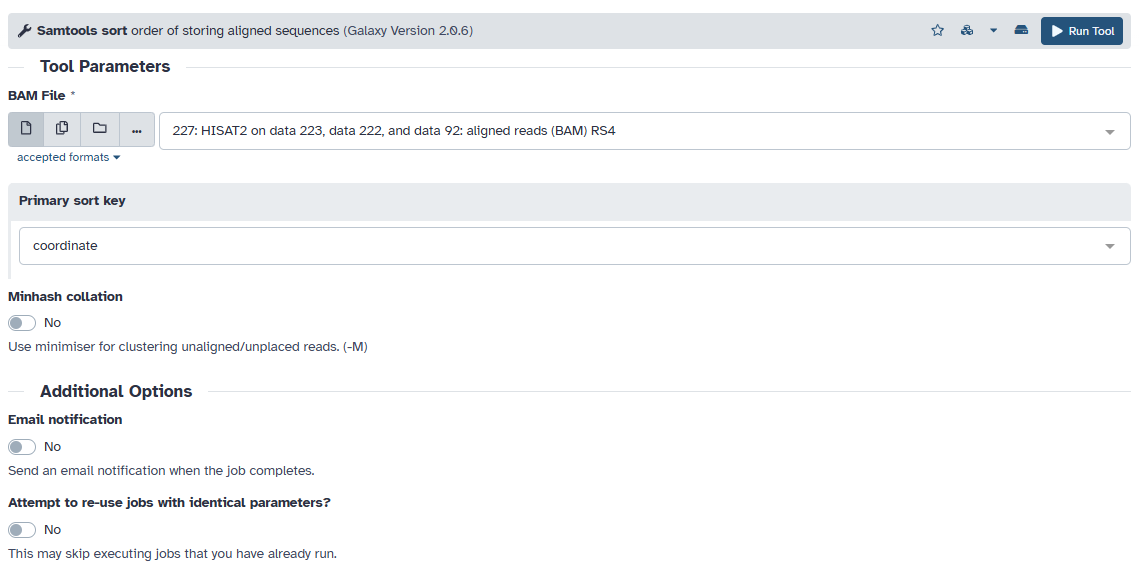
Most downstream tools, such as featureCounts, require sorted BAM files (usually sorted by coordinates) to efficiently access and process the aligned reads. Hence, sorting is a crucial intermediate step.

Tool Used: SAMtools Sort (Galaxy Wrapper)

* Tool Name: Sort SAM or BAM file by coordinates
* Galaxy Location: *NGS: SAMtools → Sort BAM*

Steps Performed

1. Input: BAM file from HISAT2 (aligned reads for each sample).
2. Sorting Order: Chose to sort by coordinates (default and recommended).
3. Output File Format: BAM (Binary Alignment Map).
4. Repeat: Step was performed for all four samples individually.



**5.8 Read Quantification using FeatureCounts**

Purpose

After aligning and sorting the RNA-Seq reads, the next critical step is quantifying how many reads map to each gene or genomic feature. This is essential for comparing gene expression levels between samples.

FeatureCounts is a fast and efficient tool used for this purpose. It counts the number of reads that align to genomic features (like genes or exons) based on a reference annotation (usually a GTF or GFF file).

Tool Used: FeatureCounts

* Tool Name: featureCounts
* Galaxy Location: *NGS: RNA Analysis → featureCounts*

Steps Performed

1. Input Files:
   * Sorted BAM files from each of the four samples (\*\_sorted.bam).
   * Corresponding gene annotation file (in GTF format) for the *Drosophila melanogaster* reference genome.
2. Count Mode:
   * Reads were counted at the gene level.
   * Only uniquely mapped reads were included.
   * Strand-specific counting was disabled (as the protocol was unstranded).
3. Count Type:
   * Reads overlapping with the exon regions of each gene were summed up per gene.

Example Settings Used (Galaxy)

|  |  |
| --- | --- |
| Parameter | Value |
| Input BAM files | ERR1659927\_sorted.bam, etc. |
| Gene annotation file (GTF) | Drosophila.gtf |
| Feature type | exon |
| Gene ID attribute | gene\_id |
| Strand-specific | Unstranded |
| Paired-end data | Yes |

Output Files

* A read count matrix was generated.
* Each row corresponds to a gene, and each column to a sample.
* Example (simplified view):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Gene ID | ERR1659927 | ERR1659930 | ERR1659925 | ERR1659929 |
| gene\_1 | 502 | 418 | 479 | 430 |
| gene\_2 | 124 | 132 | 129 | 115 |
| ... | ... | ... | ... | ... |

A screenshot of a computer

AI-generated content may be incorrect.

**Chapter 6: Differential Gene Expression Analysis using R**

**6.1 Overview of Differential Expression Analysis**

Differential gene expression (DGE) analysis is a statistical approach used to identify genes that are significantly upregulated or downregulated between experimental conditions. In this study, we compare *Drosophila melanogaster* embryos carrying the P218 mutation in the *fas* gene with wild-type controls to investigate the impact of this mutation on pre-mRNA splicing during embryogenesis.

The main steps for DGE analysis using R and DESeq2 include:

1. Preparing input files (Count Matrix and Metadata)
2. Loading data into R
3. Creating DESeq2 dataset
4. Running DESeq2 analysis
5. Extracting results and applying thresholds
6. Visualization and interpretation

**6.2 Input Files Required**

**6.2.1 Count Matrix (gene\_counts.tsv or .csv)**

* Contains raw counts of reads mapped to each gene for each sample.
* Typically generated using featureCounts or HTSeq-count after alignment.
* Rows = Genes, Columns = Samples (e.g., ERR1659927, ERR1659930, ERR1659925, ERR1659929)

**6.2.2 Metadata File (sample\_info.csv)**

* Contains experimental information about each sample.
* Must include at least two columns:
  + SampleName: matching the count matrix column names
  + Condition: indicating the group (e.g., mutant, wildtype)

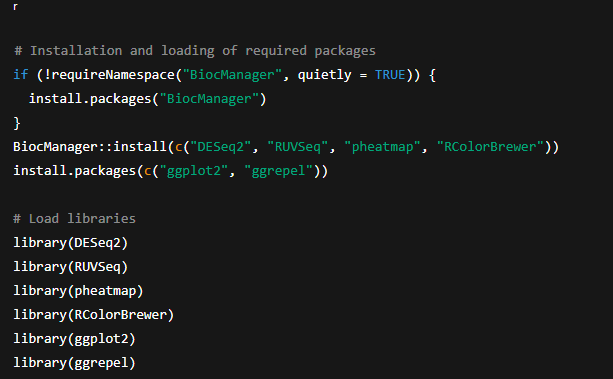
Example:

|  |  |
| --- | --- |
| SampleName | Condition |
| ERR1659927 | mutant |
| ERR1659930 | mutant |
| ERR1659925 | wildtype |
| ERR1659929 | wildtype |

**6.3 Data Preparation and Package Setup**

Before performing differential gene expression analysis, several R packages were installed and loaded. These include:

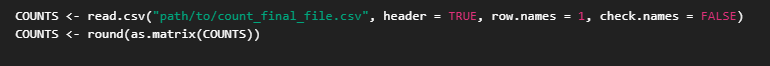
* DESeq2: Core package for normalization, statistical testing, and fold change estimation.
* RUVSeq: Used for removing unwanted variation (not directly applied here but loaded).
* pheatmap, RColorBrewer: For plotting heatmaps and applying color palettes.
* ggplot2, ggrepel: For creating volcano and PCA plots with enhanced labels.



**6.4 Importing Data**

**6.4.1 Gene Count Matrix**

Raw read counts for each gene were imported from a CSV file generated by featureCounts.



This matrix contained:

* Rows = gene IDs (e.g., *FBgn0000008*)
* Columns = sample IDs (e.g., ERR1659927)

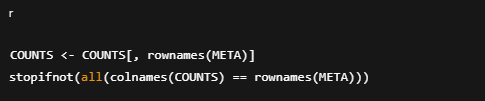
**6.4.2 Metadata File**

Sample-specific information such as genotype was loaded to match each column in the count matrix.

A black background with white text

AI-generated content may be incorrect.

Data Integrity Check: Ensured that column names in COUNTS matched row names in META.



**6.5 Creating DESeq2 Object and Filtering**

The DESeq2 dataset was constructed using the count matrix and metadata.



Low-count genes were filtered to reduce noise:

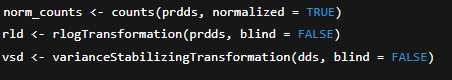


**6.6 Running DESeq2 and Normalization**

The core DESeq2 pipeline was executed:



Normalization and transformations:



**6.7 Quality Control and Visualization**

**6.7.1 Scatterplots Before and After Normalization**

A computer code with white text

AI-generated content may be incorrect.

These plots show how normalization and transformation stabilize the variance across samples.

**6.7.2 Histograms**

A black background with white text

AI-generated content may be incorrect.

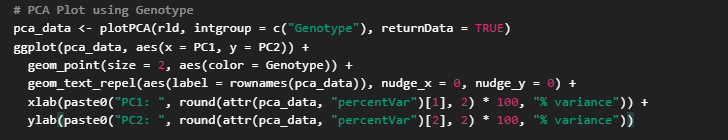
**6.7.3 Sample-to-Sample Distance Heatmap**

A screen shot of a computer program

AI-generated content may be incorrect.

Shows clustering of biological replicates based on expression similarity.

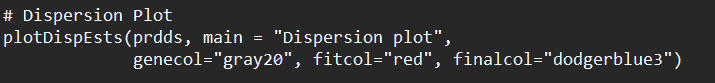
**6.7.4 PCA Plot**



What it shows:

* Reduces thousands of gene expression values into 2 principal components (PC1 and PC2).
* Each dot is a sample, colored by condition (mutant vs wild-type).

**6.7.5 Dspersion Plot**



**6.8 Differential Expression Results**

The DEGs were extracted with adjusted p-value and fold change thresholds:

A black background with white text

AI-generated content may be incorrect.

**6.8.1 MA Plot**

A screen shot of a computer

AI-generated content may be incorrect.

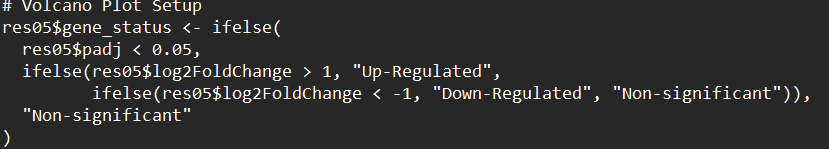
Depicts log fold change vs. mean expression for all genes.

What it shows:

* The MA plot (Mean Average plot) displays the relationship between the mean of normalized counts (average expression) on the x-axis and the log2 fold change on the y-axis.
* Each point represents a gene.
* Significant genes (adjusted p-value < 0.05) are often highlighted in red.

**6.8.2 Volcano Plot**

Classifying genes by statistical and biological significance:



A computer code on a black background

AI-generated content may be incorrect.

A black background with white text

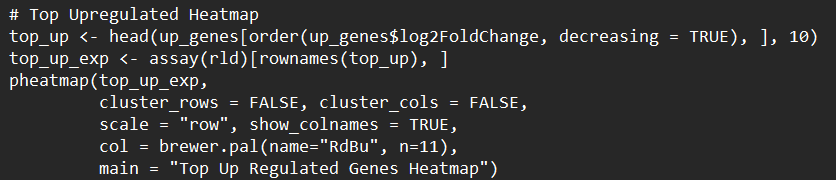
AI-generated content may be incorrect.

What it shows:

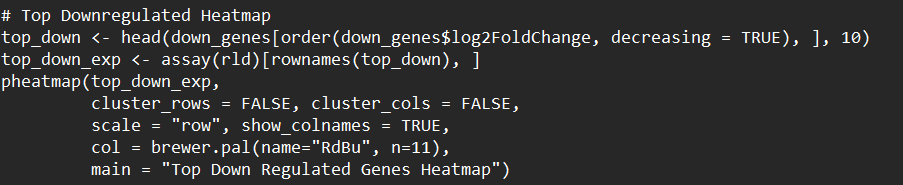
* Combines log2 fold change (x-axis) and -log10 adjusted p-value (y-axis).
* Genes with both high fold change and low p-values are seen in the top corners of the plot.

**6.9 Heatmaps of Top Differentially Expressed Genes**

Top 10 Upregulated Genes



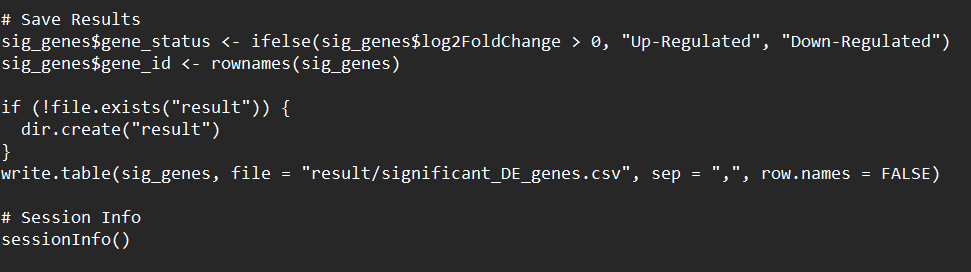
Top 10 Downregulated Genes



What it shows:

* Rows are top 30 significant genes, columns are samples.
* Colors represent expression level (after variance-stabilizing transformation).
* Clustering indicates sample similarity based on expression.

**6.10 Saving Results**



**Chapter 7: Results and Interpretation**

**Chapter 4: Linux Techniques and Applications**

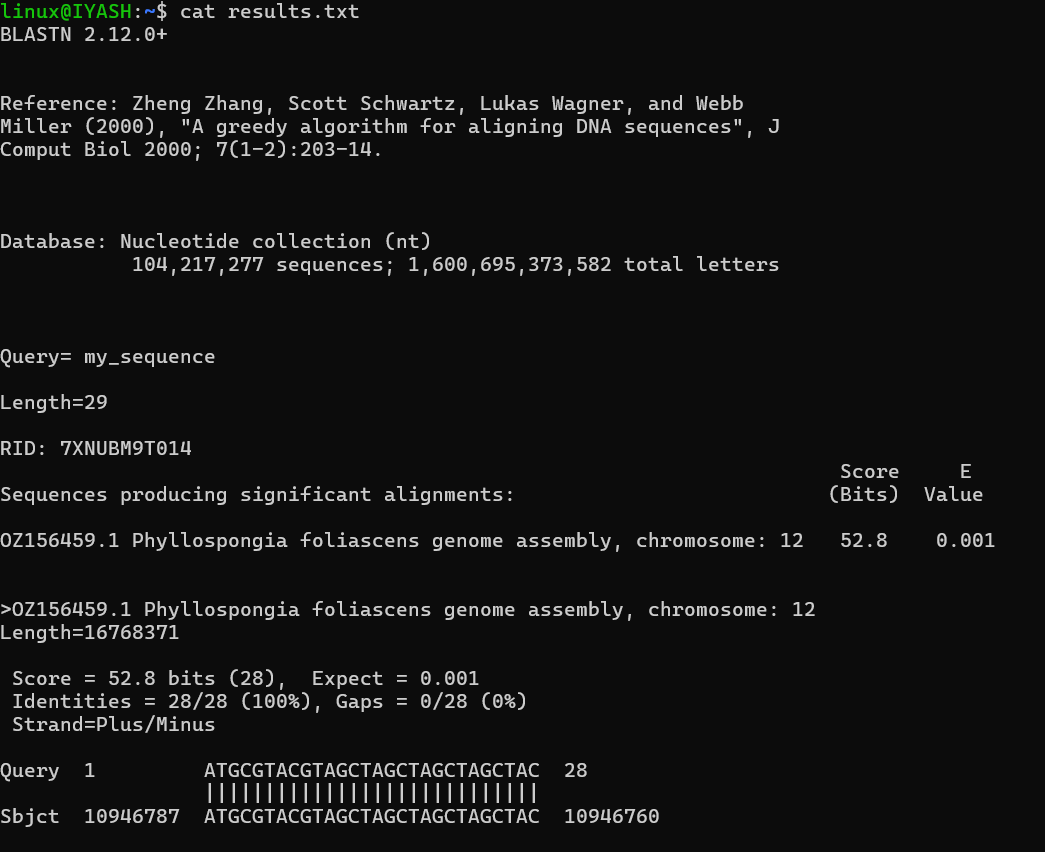
**4.1 Remote BLAST Execution via NCBI Using Linux Terminal**

Upon execution, the BLAST job was submitted remotely and completed successfully. The resulting file, results.txt, contains detailed alignments showing the query’s matches with sequences available in the NCBI nt database.

Each alignment section in the output includes:

* Query and subject sequence names
* Alignment scores and bit scores
* Percent identity
* E-values
* Matched sequence alignments

This approach provided quick and efficient confirmation of sequence similarity without the need for local infrastructure.



A computer screen with white text

AI-generated content may be incorrect.

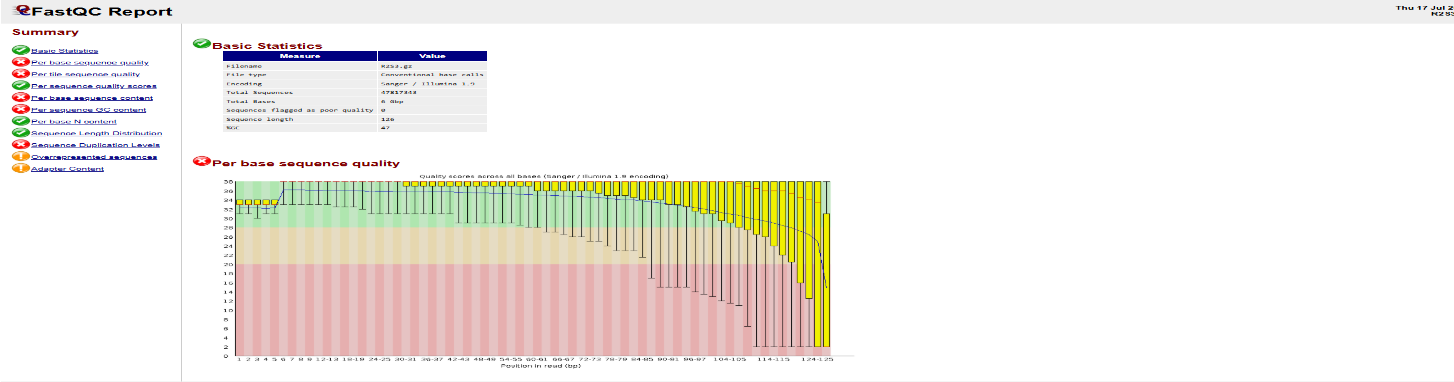
**4.3 Linux Shell Scripting for Genomic Data Analysis.**

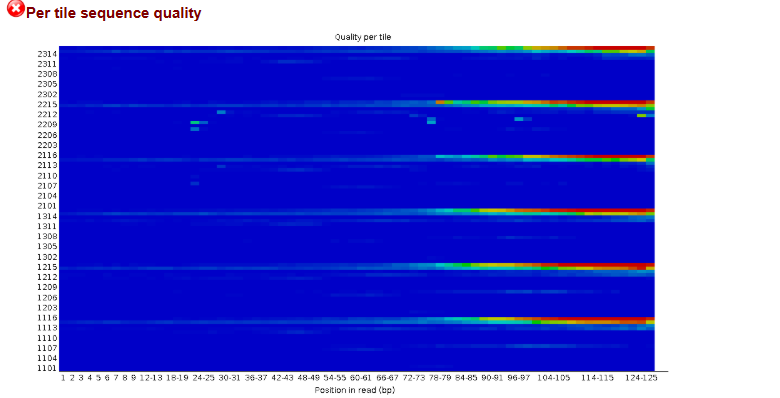
A computer screen with white text

AI-generated content may be incorrect.

**CHAPTER 5: Galaxy Platform and Preprocessing Workflow**

**5.3 Quality Control using FastQC**





A graph with a line

AI-generated content may be incorrect.

A graph showing a number of data

AI-generated content may be incorrect.

A graph of a normal distribution

AI-generated content may be incorrect.

A graph with a red line

AI-generated content may be incorrect.

A screenshot of a graph

AI-generated content may be incorrect.

A graph with a red line

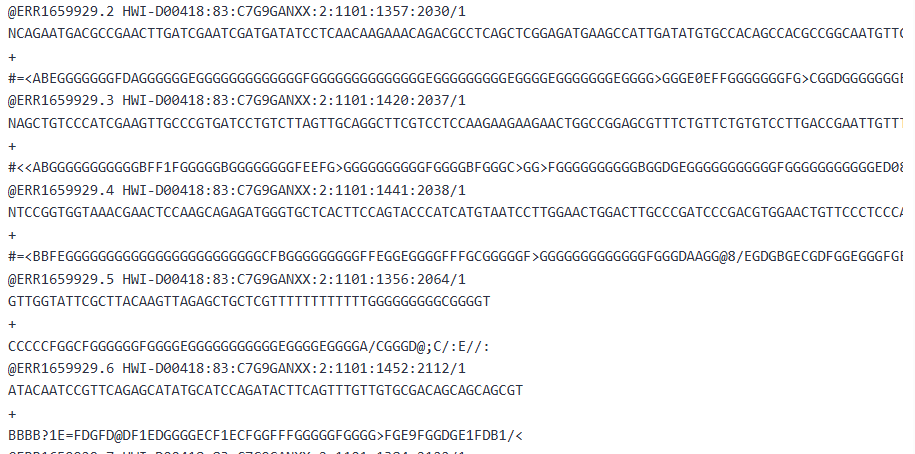
AI-generated content may be incorrect.

**5.4 Adapter Removal using Cutadapt**

A screenshot of a computer code

AI-generated content may be incorrect.

**5.5 Quality Trimming using Trimmomatic**



**5.6 Read Alignment using HISAT2**

A screenshot of a computer

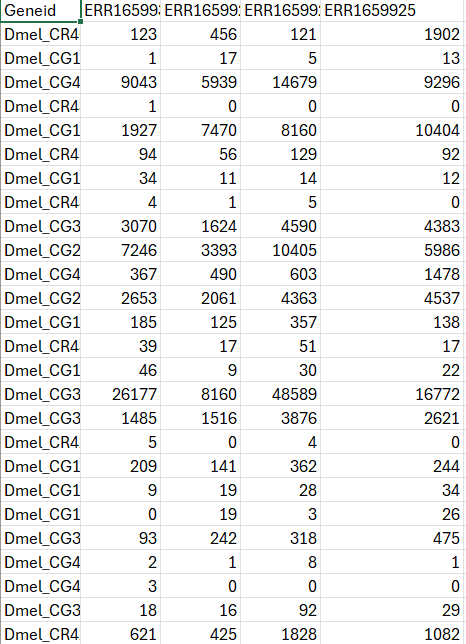
AI-generated content may be incorrect.

**5.7 Sorting Aligned Reads using SAMtools Sort**

A screenshot of a computer

AI-generated content may be incorrect.

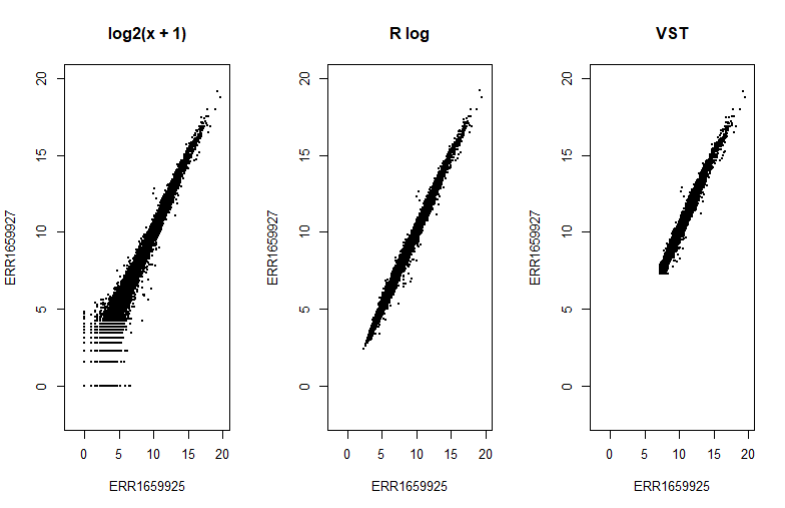
**5.9 Read Quantification using FeatureCounts**



**Chapter 6: Differential Gene Expression Analysis using R**

**6.7 Quality Control and Visualization**

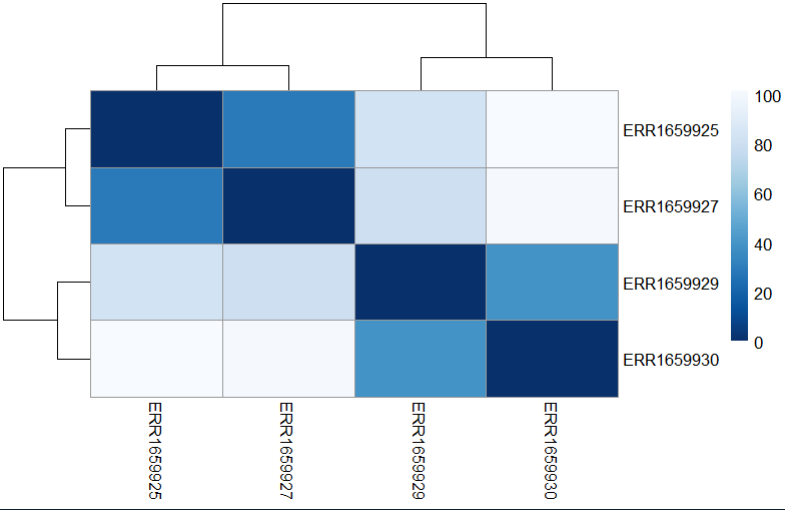
**6.7.1 Scatterplots Before and After Normalization**



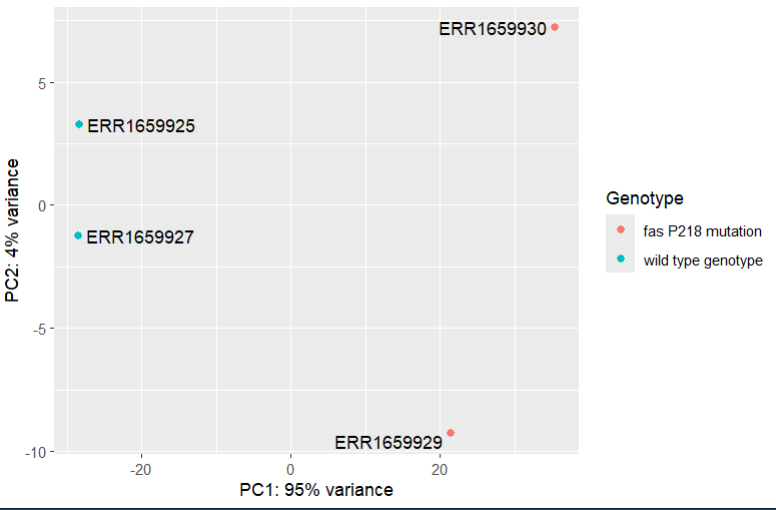
**6.7.2 Histograms**

A graph of different numbers

AI-generated content may be incorrect.

**6.7.3 Sample-to-Sample Distance Heatmap**

**6.7.4 PCA Plot**



Interpretation:

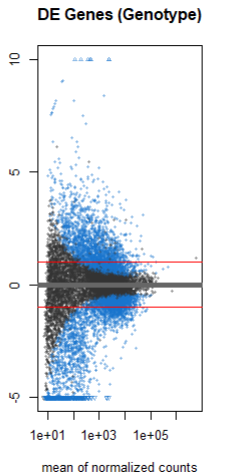
* If samples cluster clearly by condition, this indicates strong separation in gene expression.
* Helps verify data quality and batch effects.
* PCA provides evidence that the condition has a significant impact on global expression.

**6.7.5 Dspersion Plot**

A diagram of a graph

AI-generated content may be incorrect.

**6.8.1 MA Plot**



Interpretation:

* Genes near log2 fold change = 0 have similar expression in both conditions (no significant change).
* Points higher or lower on the y-axis represent upregulated or downregulated genes.
* The plot helps you visually identify which genes are most significantly differentially expressed between mutant and wild-type *Drosophila* embryos.

**6.8.2 Volcano Plot**

A graph of a function

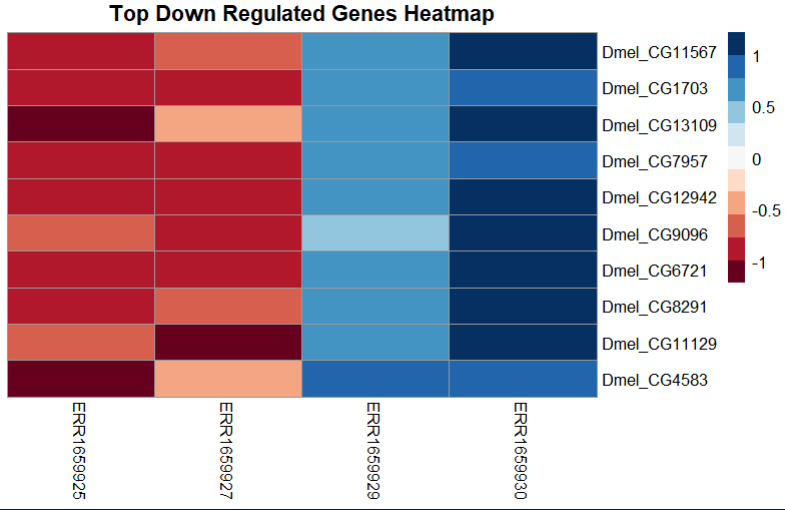
AI-generated content may be incorrect.

Interpretation:

* Genes in the top-left and top-right are significantly upregulated or downregulated.
* The volcano shape allows fast identification of high-impact DEGs.
* Good for publication figures to show significant gene expression changes.

**6.9 Heatmaps of Top Differentially Expressed Genes**





Interpretation:

* Samples from the same condition (e.g., wild-type) should cluster together if expression patterns are distinct.
* Shows clear gene expression pattern differences between P218 mutant and wild-type *Drosophila* embryos.

**Chapter 8: Results, Conclusion, and Future Scope**

**8.1 Results and Interpretation**

The analysis of transcriptomic data from Drosophila embryos with the *fas* (faint sausage) gene mutation was carried out through a complete RNA-Seq workflow, ranging from raw sequence data preprocessing to differential gene expression analysis and visualization. The samples used in this study included two mutants (ERR1659927 and ERR1659930) and two wild-type controls (ERR1659925 and ERR1659929).

After processing the raw sequencing reads using tools such as FastQC, Cutadapt, Trimmomatic, and HISAT2, the aligned reads were quantified using featureCounts. These gene count matrices, along with metadata, were then input into DESeq2 in R for statistical analysis.

The key outcomes and interpretations are as follows:

* PCA Plot revealed distinct clustering of mutant and control samples, indicating that the overall gene expression patterns are significantly altered in the *fas* mutant embryos.
* MA Plot (log ratio vs. mean average) showed a symmetrical distribution of genes, with several significantly upregulated and downregulated genes, indicating robust differential expression between groups.
* Volcano Plot provided a visual snapshot of genes with the highest statistical significance and fold change, helping to identify candidate genes affected by the *fas* mutation.
* Heatmap of the top 50 differentially expressed genes highlighted clear differences in expression patterns between mutant and wild-type samples, reinforcing the biological impact of the *fas* mutation.

The differentially expressed genes observed suggest disruptions in pathways related to RNA splicing, embryonic development, and signal transduction, consistent with the known biological role of the *fas* gene in pre-mRNA processing during early development.

**8.2 Conclusion**

This internship project provided hands-on exposure to transcriptomics and advanced bioinformatics analysis using RNA-Seq data. The investigation focused on understanding the impact of the P218 mutation in the *fas* gene of *Drosophila melanogaster* embryos.

Key achievements and learnings from this project include:

* Mastery of basic and advanced Linux commands and bioinformatics tools such as FastQC, Cutadapt, Trimmomatic, HISAT2, SAMtools, and featureCounts.
* Practical experience using the Galaxy platform for accessible, GUI-based pipeline building and processing.
* Statistical computing in R using DESeq2, including data loading, normalization, differential gene expression analysis, and result visualization.
* Interpretation of visual plots and statistical outputs to derive biological significance from computational results.

Overall, this project successfully demonstrated that the *fas* gene mutation leads to measurable changes in gene expression patterns, affirming its key role in RNA processing during embryogenesis. The workflow employed serves as a strong template for similar RNA-Seq based transcriptomic studies.

**8.3 Future Scope**

This study lays the groundwork for multiple directions in which the research can be extended or refined. The following future enhancements are recommended:

1. Functional Annotation and Pathway Enrichment  
   Tools like DAVID, ClusterProfiler, or gProfiler can be used to perform Gene Ontology (GO) enrichment and KEGG pathway analysis to further investigate the biological roles of differentially expressed genes.
2. Alternative Splicing Analysis  
   Given that *fas* is associated with pre-mRNA splicing, tools like rMATS, DEXSeq, or MAJIQ can be used to explore exon usage and splicing isoform variations between conditions.
3. Time-Series RNA-Seq Analysis  
   Performing RNA-Seq across multiple embryonic time points could capture dynamic gene regulation and reveal temporal effects of the *fas* mutation during developmental stages.
4. Transcript Isoform Reconstruction  
   Use of transcriptome assemblers like StringTie or Cufflinks could reconstruct full-length transcripts and novel isoforms that may be affected by the mutation.
5. Wet Lab Validation  
   Experimental validation using qPCR, Western blot, or in situ hybridization can confirm the differential expression of key candidate genes and strengthen the findings.
6. Cross-Species Comparison  
   Investigating whether similar gene expression disruptions occur in other organisms with homologous splicing genes may broaden the biological implications of the findings.

**8.4 Applications**

The skills and techniques gained through this internship have wide-ranging applications in modern biology and biomedical sciences:

* Genetic and Developmental Research  
  This pipeline can be used in developmental biology to understand gene regulation during critical stages such as embryogenesis.
* Disease Mechanism Studies  
  Similar RNA-Seq workflows can be employed to study gene expression in disease models, such as cancer, neurodegeneration, or genetic disorders involving splicing defects.
* Precision Medicine and Therapeutics  
  Understanding gene expression and splicing changes can help design targeted gene therapies or drug interventions for diseases caused by regulatory defects.
* Agrigenomics and Crop Improvement  
  Transcriptomics can also be applied to plants for stress response analysis, trait selection, and improving yield using genetic engineering.
* Education and Research Training  
  This project serves as a valuable learning model for students and early researchers entering the field of bioinformatics and systems biology.

**Chapter 9. References**

1. *Sona Charles, Mukesh Sankar S, TE Sheeja, Shahana Arif (2024).* Online Workshop on R for Bioinformatics *– Training Manual*. ICAR-Indian Institute of Spices Research, Kozhikode, Kerala, India. Training Manual No: 2024/04.  
   Compiled and Edited by: Dr. Sona Charles, Dr. Mukesh Sankar. S, Dr. TE Sheeja, Ms. Shahana Arif  
   Published by: Dr. R Dinesh, Director, ICAR-IISR  
   Cover Design: *Mr. Sudhakaran A*
2. Galaxy Project (UseGalaxy). (n.d.). Retrieved from https://usegalaxy.org  
   Used for RNA-Seq preprocessing including quality control, trimming, alignment, and quantification.
3. NCBI Sequence Read Archive (SRA). (n.d.). Retrieved from https://www.ncbi.nlm.nih.gov/sra  
   Source for raw sequencing data related to *Drosophila melanogaster*.
4. European Nucleotide Archive (ENA). (n.d.). Retrieved from https://www.ebi.ac.uk/ena  
   Alternative repository for retrieving public sequencing datasets.
5. Expression Atlas - EMBL-EBI. (n.d.). Retrieved from https://www.ebi.ac.uk/gxa/home  
   Used to understand gene expression across different biological conditions.
6. EMBL-EBI (European Bioinformatics Institute). (n.d.). Retrieved from https://www.ebi.ac.uk  
   Accessed for bioinformatics resources and databases.
7. DESeq2 – Bioconductor. (n.d.). Retrieved from https://bioconductor.org/packages/release/bioc/html/DESeq2.html  
   Used for differential gene expression analysis in R.
8. GeeksforGeeks – R Programming Language. (n.d.). Retrieved from https://www.geeksforgeeks.org/r-programming-language/  
   Referred for foundational and advanced concepts in R programming.
9. W3Schools – R Tutorial. (n.d.). Retrieved from https://www.w3schools.com/r/  
   Used for basic syntax, plotting functions, and data manipulation techniques in R.
10. Linux Hands-on Manual  
    ICAR-Indian Institute of Spices Research. *Basic Linux Commands and Hands-on Bioinformatics Manual*. Bioinformatics and Integrative Genomics Facility.