GL4HS BIOINFORMATICS MANUAL

© 2021 United States Government as represented by the Administrator of the National Aeronautics and Space Administration. All Rights Reserved.

© 2021 Elizabeth Blaber and Samrawit Gebre

Notice for Copyrighted Information

This manuscript is a joint work of the National Aeronautics and Space Administration and employee(s) of KBRwyle under Contract No. NNA16BD14C with the National Aeronautics and Space Administration. The United States Government may prepare derivative works, publish or reproduce this manuscript, and allow others to do so. Any publisher accepting this manuscript for publication acknowledges that the United States government retains a nonexclusive, irrevocable, worldwide license to prepare derivative works, publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government Purposes. All other rights are retained by copyright owner.

All rights reserved. No part of this publication may be reproduced or transmitted, in any form or by any means, without the prior permission in writing by copyright owners. Inquiries concerning reproduction should be sent to NASA Space Biology or GeneLab for High Schools (GL4HS) management.

Table of Contents

TABLE OF CONTENTS

Exercises 4

Lesson 1 – Omics Data and Genelab 5

What are Omics? 5

Analytical techniques for studying omics 6

Genomics and transcriptomics 7

Proteomics 7

Metabolomics 7

Data Analysis 7

How does GeneLab relate to omics? 8

Lesson 2: Using Genelab and understanding metadata 9

Genelab 9

The GeneLab Open Science Data Repository 10

Metadata – What is it and how do we use it? 11

Viewing metadata 11

Lesson 3: Analysis of omics data – RNA-Seq 15

RNA-Seq 15

How does RNA-Seq work? 16

RNA-Seq analysis workflow 20

Analysis Platform (Galaxy) 20

Key Galaxy actions 23

Analyzing RNA-Seq data 31

Checking quality and trimming data 31

Alignment 43

Count how many reads mapped to each gene 52

Normalizing and performing differential expression 61

Annotation and visualization of differentially expressed genes 71

Functional analysis - GOSEQ 76

Gene Set Enrichment Analysis 81

Molecular Signature Databases: 89

Lesson 4: WHat's next? Analysis techniques for interpreting data 92

References 94

EXERCISES

Exercise 1: GeneLab 9
Exercise 2: The GeneLab Data Repository 10
Exercise 3: Analyzing MetaData 12
Exercise 4: Understanding RNA-Seq 18
Exercise 5: Accessing the GeneLab Analysis Platform 21
Exercise 6: Getting familiar with Galaxy 23
Exercise 7: Importing data 24
Exercise 8: Renaming the history 26
Exercise 9: Checking the quality of the data 35
Exercise 10: Trimming the data 40
Exercise 11: Aligning to a reference genome 48
Exercise 12: Counting mapped reads 55
Exercise 13: Checking the quality of read assignment 59
Exercise 14: Normalization and differential gene expression 62
Exercise 15: DESeq2 65
Exercise 16: Filter and annotate differentially expressed genes 72
Exercise 17: Generate the Volcano Plot 74
Exercise 18: Generate the files for Gene Ontology testing with goseq 76
Exercise 19: Gene Ontology testing with goseq 78
Exercise 20: Generate the files for FGSEA 81
Exercise 21: Running FGSEA 83
Exercise 22: Analyzing the FGSEA results 86
Exercise 23: Using MSigDB 89

LESSON 1 – OMICS DATA AND GENELAB

Enduring Understanding: Omics is the field of studying large scale datasets and GeneLab helps to archive omics data from spaceflight experiments.

Essential Question: What is 'omics'? How can omics be used to understand changes in an organism's physiology?

Learning Objectives: By the conclusion of this module, students will be able to:

1. Explain what is meant by 'omics'
2. Identify some of the main omics fields
3. Describe the different types of analytical tools used to conduct omics analysis
4. Describe some of the statistical tools that can be used to interpret omics data
5. Explain how GeneLab uses omics datasets

WHAT ARE OMICS?

Omics are defined as analyses that end in –omics, e.g. proteomics, genomics, metabolomics, but what does that really mean? The –ome suffix is really used to address the study of the –omics fields, so genomics is the study of the genome and metabolomics is the study of the metabolome. Omic technology adopts a holistic view of the molecules that make up a cell, tissue, or organism. They are aimed at the universal detection of genes, mRNA, proteins and metabolites in a biological system in a way that is not targeted (e.g. investigation of a specific gene) and is aimed to be unbiased. The basic principle behind this approach, is that a complex system can be understood more thoroughly if it is studied as a whole. This method of studying systems is known as **systems biology** and it differs from hypothesis driven research because the experiments conducted are designed to enable the formation of new hypotheses. Omics technology can be applied to basic research to help understand the response of organisms to different external stimuli but can also be used in disease processes where they can play a role in screening, diagnosis and prognosis as well as aiding our understanding of disease progression. These types of studies also drive the development of new omics fields, e.g. pharmacogenomics is the study of how an individual's genetics will change their response to various pharmacological drugs, therefore enabling doctors and pharmacists to optimize a drug for a specific individual. The main types of omics are listed below:

Genome (Genomics) – the total DNA of a cell or organism. This includes DNA that encodes for genes, non-coding DNA and genetic material contained outside of the nucleus. Genes can be studied individually through techniques such as PCR and quantitative PCR, or they can be studied genome wide e.g. microarrays and sequencing techniques.

Transcriptome (transcriptomics) – the study of the RNA molecules within an organism, including messenger RNA (mRNA), ribosomal RNA (rRNA), transfer RNA (tRNA) and all other non-coding RNAs. Generally, transcriptomics focuses on studying the mRNA within a cell which is the template for protein

synthesis. The transcriptome is studied through expression-based techniques, such as, microarrays or through sequencing techniques, e.g. RNA-Seq

Proteome (Proteomics) – the study of the expression of proteins of an organism. Proteomics aims to study the functional relevance of proteins, and it does this by characterizing the information flow within the cell and the organism through protein pathways and networks. These proteins can be presented in any part of the organism, e.g. cell, tissue or secreted, and generally the proteome is studied in specific areas in order to refine the area of study. For example, if we wanted to study the secreted proteins of a mouse (e.g. in the blood) than we would study the secretome. The proteome is usually studied through mass spectrometry-based techniques, whereas individual proteins are studied through a wide variety of techniques including physical biochemistry, enzymological, and immunological techniques.

Metabolome (metabolomics) – the metabolome refers to all of the small molecule chemicals that are found within an organism and may be produced through metabolic activities of the organism (endogenous metabolites) or may not be naturally produced (exogenous metabolites). Metabolomics, therefore, is generally defined as the study of global metabolite profiles in a system under a given set of conditions. The metabolome is the final downstream product of gene transcription and therefore changes in the metabolome are amplified compared to those of the transcriptome and the proteome.

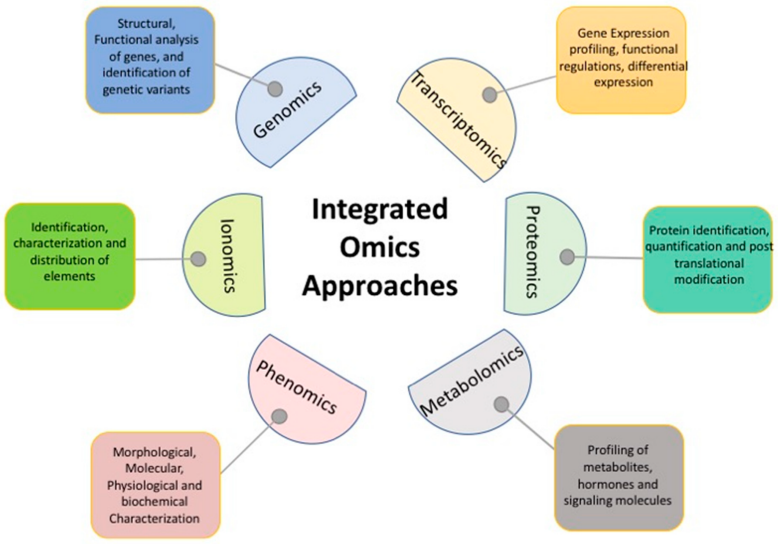


Figure 1. Omics branches used for characterization of an organisms response to a stressor (Adapted from [1]).

ANALYTICAL TECHNIQUES FOR STUDYING OMICS

Sample preparation for all omics analyses is very important and should result in reproducible results, however, sample preparation can vary from lab to lab resulting in varied results that are not reproducible. Therefore, standardizing sample preparation for omics analyses is imperative for enabling reproducible results between different laboratories and different analysis platforms.

GENOMICS AND TRANSCRIPTOMICS

RNA-seq is a powerful tool that enables us to analyze the expression profile of an entire organism's genome. RNA-seq is more accurate than the previous microarray technology which was based on the principle that two DNA strands with complimentary sequence pair with each other by forming hydrogen bonds and complimentary nucleotide base pairs (hybridization). RNA-seq, however, fragments RNA into small pieces, which are then converted to complementary DNA (cDNA) and introduced to a flow cell whereby the actual sequence is determined using the addition of fluorescent tags. This results in millions of reads that must then be aligned and assembled to determine the sequence of the transcriptome and determine expression levels for each gene.

PROTEOMICS

There are multiple methods for studying proteomics, however, the most common method for studying large scale datasets are based on mass spectrometry. Mass spectrometry uses the mass to charge ratio of the protein to identify its peptides which can be used to identify the protein. One method of studying proteins uses two-dimensional electrophoresis to separate proteins from different samples, which can then be isolated and analyzed via mass spectrometry. Alternatively, one can use isotope tags to label proteins from two different mixtures. These mixtures can then be combined and run through liquid chromatography followed by mass spectrometry.

Protein chips are also a new technology that can be used in a similar manner as DNA microarrays, however, instead of probes, the microarrays contain antibodies, or other protein detecting features to determine the expression profile.

METABOLOMICS

Metabolomics can also be studied using mass spectrometry. For analysis to occur, metabolites must be separated into smaller groups to enable detection to occur. Gas chromatography coupled with mass spectrometry may then be used to identify individual components within the metabolome.

DATA ANALYSIS

Omics experiments generate huge amounts of data that complicate the statistics and interpretation of the results. To understand what the data means; we employ complex bioinformatics tools and sophisticated statistics.

One of the biggest problems with genomics and transcriptomics data is the large number of genes that are being analyzed. The overall goal of a transcriptomic experiment is to identify genes that are differentially regulated or changed between two different treatment groups (e.g. spaceflight and the ground controls). We may also want to identify groups of genes that work together or to find molecular similarities between a subset of samples if there is more than one treatment group (e.g. microgravity and radiation exposure). We can use rigorous statistical methods to identify genes or groups of genes that

are changing in response to a specific treatment. We can then use more precise gene expression methods, such as quantitative real time PCR, to validate whether these changes are real or not.

The simplest and most common experimental set up in transcriptomic data is control versus treatment e.g. ground control versus spaceflight. The most common way for conducting statistical analysis on two groups of data is the t-statistic, in which you subtract the mean value of a gene of interest in one group from the other group and divide the result by the standard error of the gene of interest. The problem with this is that for transcriptomic data, the number of samples is generally small, and therefore the standard error for the gene is somewhat unreliable. For example, in a number of cases the standard error is underestimated and therefore a small change in expression levels results in extreme t values and 'false positives'.

So, we need to ask how often a random sampling of a single group is as far from 0 as the t values observed in the data. When we say that we accept a $t < 0.05$, then we are accepting a 5% false positive rate, i.e. for every 20 tests that are conducted one will be a false positive. The false positive rate of a procedure is the fraction of unchanged genes which appear as positives (falsely). If we want to generate a list of genes which are actually different (and not false positives) then we need to minimize the number of false positives. We therefore need to determine the false discovery rate for a list, i.e., the expected fraction of false positives in a list of genes following some statistical procedures.

HOW DOES GENELAB RELATE TO OMICS?

Modern biological sciences have become increasingly based on molecular biology and high-throughput molecular techniques, such as genomics, transcriptomics, and proteomics. Such detailed analysis of the RNA, DNA and protein expression profiles enable significant advances in the understanding of individual genes, molecules, cells and ultimately organisms. NASA scientists and the NASA Space Biology Program have aimed to examine the fundamental building blocks of life (RNA, DNA, and protein) in order to understand the response of living organisms to spaceflight and aid in fundamental research discoveries on Earth. In an effort to enable NASA funded science to be available to everyone, NASA has collected the data from 'omics' studies and curated them in a data system called GeneLab. GeneLab is an open access system that enables anyone, including students; to examine and analyze data from mostly government funded space research. Whilst most college-level interns, academics and other scientists have had some interaction with 'omics data sets and analysis tools, high school students often have not. Therefore, the aim of this program is to inspire the next generation of scientists to get involved in space research using NASA's GeneLab Data System (GLDS). Through the analysis of data sets in GLDS, high school students will learn about organismal physiology in space and will gain complex and critical bioinformatics skills, which they will then be able to utilize to propose a hypothesis driven research project.

LESSON 2: USING GENELAB AND UNDERSTANDING METADATA

Enduring Understanding: GeneLab compiles omics data from spaceflight experiments into a repository that can be used by scientists to generate new hypotheses.

Essential Question: How can we find omics data on GeneLab? How do we use this data?

Learning Objectives: By the end of this module, students will be able to:

1. Explain the importance of GeneLab
2. Navigate the GeneLab website and data repository
3. Explain the significance of Metadata
4. Analyze Metadata for a dataset

GENELAB

GeneLab can be found at: **https://genelab.nasa.gov**. The website contains extensive information about omics data related to space biology and how this data can be used to advance our understanding of life in space and on the Earth.

Exercise 1: GeneLab

Explore the 'About' page on the GeneLab website. Click on the other menu links to learn more about the various activities, publications, and resources GeneLab provides. Read some of the 'Latest News' to determine recent advances in space biology omics data.

Notes:

_____

_____

_____

_____

_____

_____

_____

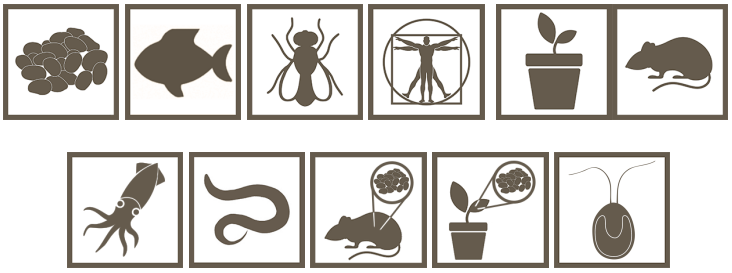
_____

_____

THE GENELAB OPEN SCIENCE DATA REPOSITORY

GeneLab data is compiled from '-omics' experiments performed by scientists around the world that have conducted research on board the Space Shuttle, the International Space Station and/or in ground-based simulation models. The data from these experiments is compiled into the GeneLab Open Science Data Repository which can be found by clicking on the 'Data Repository' link on the top menu under Data or using the navigation button on the GeneLab website. Here you can learn about the GeneLab Data Repository.

All the data housed in the 'Data Repository' is organized by study or experiment. Each dataset has a symbol that represents which organism the data came from as shown below:



Each dataset in the repository contains specific information about the organism, the 'spaceflight factors' e.g. microgravity, radiation, simulated spaceflight etc., the type of analysis that was conducted on the tissue, and a brief description of the experiment. More information can be obtained by clicking on a specific dataset.

Once on the Data Repository page, there are several ways to search or browse for datasets:

1. Use keywords in the search bar to narrow down your search.
2. Use the Search Filters to filter on pre-populated terms (GeneLab database only).
3. Use the scroll bar(s) to browse through the datasets.

<u>Exercise 2: The GeneLab Open Science Data Repository</u>

Read about the GeneLab Data Repository on the homepage and by selecting 'About' at the top menu and selecting 'About GeneLab Data Systems'.

Read about various features and functions of the GeneLab repository by selecting 'Help' on the top menu and then selecting 'Tutorials'. Under tutorials, you will find several guides and videos on how to use GeneLab and the Data Repository.

Notes:

_____

_____

_____

_____

_____

_____

METADATA – WHAT IS IT AND HOW DO WE USE IT?

Metadata is information that provides information about other data; essentially, it is data about data. In this case, metadata provides all of the experimental details that we need to know in order to analyze a dataset. The metadata will provide an overview of the experiment, how the samples were collected, the sample types that were collected, how the 'omics' experiment was performed or the assay type, and finally, how the data was processed. The metadata in GeneLab can be viewed by selecting a dataset and scrolling through the different panels or navigating to the different sections.

VIEWING METADATA

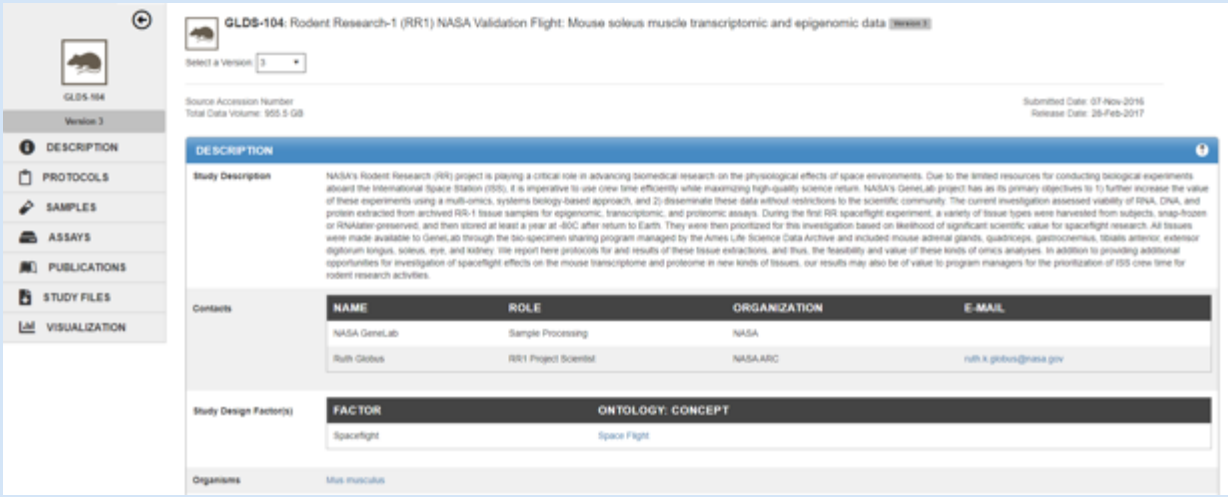
There are several important factors that we will consider from the metadata to help understand the data we are analyzing. GeneLab has input a variety of information into the metadata tab organized under the following headings:

- ***Description*** provides an overview of the study including a text description, contact information, experimental factors, organism(s), type of assay(s), project and/or mission details, and funding information.
- ***Protocols*** provides the names and detailed descriptions of the sample collection, treatment, assay, data processing, and any other study protocols.
- ***Samples and Assay(s) table*** provide sample and assay level details formatted in a navigable table. Information found in these tabs include specific organism characteristics, study factors and treatments, radiation metadata, sample and sample processing metadata, assay execution parameters, and data processing metadata.
- ***Publications*** contains metadata for relevant publication(s) including title, authors and link(s).
- ***Study Files*** provides metadata regarding raw and/or processed study data files. Each row includes information about the size, type, and description of the file set.
- ***Visualizations*** will navigate you to the visualization portal if processed data is available for that particular dataset. The portal includes an interactive view of how the genes are expressed.

Exercise 3: Analyzing MetaData

For this exercise we will use the search bar to find dataset GLDS-104. This will navigate us to a page where we can explore and download both the experiment files as well as the metadata.

1. In the search bar, type in 'GLDS-104' and click enter.
2. Select the first study Rodent Research-1 (RR1) NASA Validation Flight: Mouse soleus muscle transcriptomic and epigenomic data
3. Navigate through the various sections to find information about the data (metadata) and the data files



Investigate the metadata and see what you learn about the study. Then try to answer the following questions:

1. What organism was being studied?

   _____

2. What was the hardware used for this experiment (Experiment Platform)?

   _____

3. How long were mice in space?

   _____

4. What kind of tissue(s) were collected?

   _____

Exercise 3 (cont'd): Analyzing MetaData

5. Learn more about this tissue by conducting a google search. What type of tissue is this? How is this tissue affected by spaceflight?

_____

_____

_____

_____

_____

6. How many samples were there in each condition?

_____

7. What type of '-omics' was conducted on the samples?

_____

8. Which scientist(s) ran this experiment?

_____

Specifically for the rest of this tutorial we will need the following information. Therefore, ensure that you have the details completed:

Organism: _____

Sex: _____

Age at launch: _____

Factor(s): _____

Experiment duration: _____

Tissue (material type): _____

Exercise 3 (cont'd): Analyzing MetaData

Some studies have several assays, used the drop-down menu under Assays/Measurements and select Transcription profiling.

Assay name: _____

Stranded: _____

Read length: _____

More about RR-1: https://www.nasa.gov/ames/research/space-biosciences/rodent-research-1

Notes:

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

LESSON 3: ANALYSIS OF OMICS DATA – RNA-SEQ

Enduring Understanding: RNA-seq is a powerful technique that we can use to analyze genome wide expression data and understand complex biology using a variety of tools

Essential Question: How do we analyze RNA-seq data? What can they tell us about an organism's response to a specific treatment, e.g. spaceflight?

Learning Objectives: By the conclusion of this module, students will be able to:

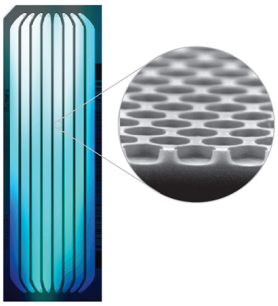
1. Explain the usefulness of RNA-seq
2. Understand the components of RNA-seq analysis
3. Describe the different types of analytical tools for RNA-seq analysis
4. Interpret RNA-seq data and associate biological relevance with it

RNA-SEQ

RNA Sequencing is a powerful tool to study the transcriptome in depth. Remember that the transcriptome includes all of the messenger RNA (mRNA) molecules (and other RNAs such as lncRNA, miRNA, etc.) that are expressed from the genes of an organism, therefore, transcriptomics is the study of all messenger RNA molecules. Studying mRNA enables us to understand how an organism's gene expression is altered when that organism is exposed to a specific condition, e.g. spaceflight or radiation, or a specific disease, e.g. cancer or diabetes. Previously, microarrays were considered to be a powerful tool to study the transcriptome. Microarrays are designed on the principle that when two DNA strands with complementary sequences pair with each other, they form hydrogen bonds and complimentary nucleotide base pairs. The more complementary base pairs that form, the stronger the bonds will be between the two strands of DNA.

However, recent advances in technology and our understanding of the transcriptome have enabled the development of RNA-sequencing technologies which are superior to microarrays in a number of ways. More specifically, RNA-sequencing enables the detection of low abundance transcripts and can more readily detect subtle changes in gene expression under different conditions. Furthermore, microarrays are limited to known sequences as they contain probes (short sequences) to which the cDNA within the sample must hybridize. Furthermore, microarrays present a number of technical limitations including varying probe performance, cross-hybridization, and non-specific hybridization resulting in false positives. Therefore, RNA-seq has a wider applicability and is not limited to known transcripts therefore enabling us to gather unbiased and unprecedented information about the transcriptome and gene expression levels. Due to these reasons, we will be analyzing RNA-seq data in this course.

HOW DOES RNA-SEQ WORK?



SAMPLE PREPARATION

Most RNA-Seq experiments follow a similar protocol. First, total RNA (including ribosomal RNA (rRNA), transfer RNA (tRNA), messenger RNA (mRNA), and other types of RNA (lncRNA etc.) is extracted from a sample of interest. There are several ways to extract RNA, but most procedures rely on disrupting the tissues or cells using an organic solution with or without mechanical agitation. Once the RNA is isolated, the purity and yield (quantity) should be measured using a Bioanalyzer or similar device. Then, depending on the type of analyses the scientist would like to conduct, it may be necessary to enrich for a specific class of RNA depending on the scientific objectives of the study. Total RNA usually contains approximately 80% ribosomal RNA (rRNA), and therefore, it is often necessary to remove these rRNA sequences. There are several methods to accomplish this, and the method used should be contained within the study metadata. Some of the most common methods include selection of the target RNAs via hybridization, removal of non-target RNAs by hybridization or target enrichment by size. For most of the samples we will be working with, we are interested in mRNA as this will enable us to understand what genes are being expressed in response to a specific stimulus e.g. spaceflight or radiation. mRNA contains a specific feature (poly-A tail) that can be used to select for mRNAs via hybridization. Alternatively, we can remove ribosomal sequences by hybridization using oligos (short DNA molecules with lengths of approximately 25 bp) that are specific to highly conserved ribosomal sequences. As mentioned, rRNA is by far the most abundant type of RNA in a sample. Removal of rRNA thus enables us to determine changes in mRNA with more accuracy. Additionally, this method enables us to preserve non-polyadenylated RNAs (lncRNAs, miRNAs etc) and therefore broadens the classes of RNA that may be investigated. Enrichment that is targeted to specific size-selection is generally conducted to enrich for small RNA molecules including miRNA and siRNA. As they are generally much smaller than mRNA or rRNA, these RNAs can be separated by size.

LIBRARY PREPARATION

After RNA is extracted and purified, we then prepare the library. The first step in library preparation is fragmentation and conversion to cDNA. The RNA must be fragmented because most sequencing platforms are only able to accurately provide relatively short sequencing reads (~40-400 bp) whereas most mRNA transcripts are thousands of nucleotides in length. Therefore, fragmentation enables the RNA to be read by the sequencing instruments and improves the sequence coverage over the transcriptome. There are several methods that can be used to fragment RNA including enzymatic reactions, heat and sonication. The fragmentation effectiveness is evaluated using gel electrophoresis or a Bioanalyzer prior to proceeding with sequencing. RNA is converted to DNA using reverse transcriptase and the second cDNA strand is synthesized using DNA polymerase.

Before sequencing, platform-specific adaptors are added to the ends of the cDNA. These adaptors are generally complimentary to sequences contained within the platform that is being used to conduct the

sequencing. The composition of these adaptors depends on the sequencing platform that is used. Several platforms also enable multiplexing, or the sequencing of several samples at once. In order to determine which reads are derived from each sample, adaptors that contain specific tags or barcodes are used. The sequence of the insert and tag can then be used to determine identify the sample or experiment for that read.

STRANDED LIBRARIES

Early RNA-Seq protocols did not retain information on the strand of origin for each transcript. This is because synthesis of randomly primed double-stranded cDNA followed by the addition of adaptors does not retain information on which strand (sense or anti-sense) the original mRNA template is coming from. Without that information, it is difficult to accurately determine gene expression from overlapping genes. i.e. genes that have partially overlapping genomic coordinates but are transcribed from opposite strands. Therefore, knowing the strand information is very important for determining whether a given RNA transcript sequence derives from one of two overlapping genes.

It is now possible to retain this information by modifying the original RNA-seq protocol. There are several ways to do this, but the most popular is to use E. coli DNA polymerase which can synthesize DNA using either dUTP or dTTP and therefore dUTP can be used to synthesize the second strand. The dUTP strand is then degraded by Uracil-DNA-glycosylase (UDG) and the remaining strand is amplified to create a cDNA library suitable for sequencing. A second approach is to use a DNA polymerase that is capable of extending past dUTP bases.

VALIDATION OF THE LIBRARY PREPARATION

Once the library is prepared, it's extremely important to validate the quality of the library before proceeding to sequencing. As sequencing is quite expensive, it is extremely important to ensure that the library is of high quality and contains sufficient cDNA to enable effective sequencing. Additionally, it is important to assess the fragment size distribution of the library.

SEQUENCING

There are many different sequencing platforms (machines with accompanying chemistry and sequence detection technology) available including Illumina, PacBio, IonTorrent and others. As mentioned, we will mainly be conducting data analysis on experiments conducted using the Illumina platform and therefore will focus on this platform in this manual. These sequencing platforms perform "next generation sequencing" (NGS) also termed "massively parallel sequencing" which are high throughput techniques.

Once the libraries have been prepared and validated, they are hybridized to a flow cell. There are several different types of flow cells, but they generally consist of fluidic channels that contain covalently bound oligonucleotides complimentary to the sequencing adapters that were introduced during the library preparation. Once the fragments are hybridized, DNA polymerase extends the oligonucleotide resulting in a covalently bound full-length complimentary copy of the cDNA fragment. All four dNTPs are fluorescently tagged but once the nucleotide incorporation is detected by fluorescence, the fluorophore

is removed therefore allowing the incorporation of the next dNTP. During base calling, the fluorescence intensity values for each nucleotide are converted to nucleotide identity, enabling the sequence to be determined.

Exercise 4: Understanding RNA-Seq

Conduct the following short course on RNA-seq with Illumina:

https://support.illumina.com/content/dam/illumina-support/courses/Sequencing_Illumina_Technology/story.html?_gl=1*o1kwv5*_ga*MzcxMTQ3ODcoLjE3MTY1MTUwOTg.*_ga_VVVPY8BDYL*MTcxNjUxNTA5Ny4xLjEuMTcxNjUxNTEzMS4yNi4wLjA.

Conduct the following tutorial on RNA-Seq: https://discoveringthegenome.org/discovering-genome/course-modules/rna-sequencing

Answer the following questions to check your understanding:

1. Which technique would you use if you are trying to discover new genes: microarray or RNA-seq? Why?

_____

_____

_____

2. Why is RNA fragmented during sequencing?

_____

_____

3. What is the purpose of adapters? Where are they added?

_____

_____

Notes:

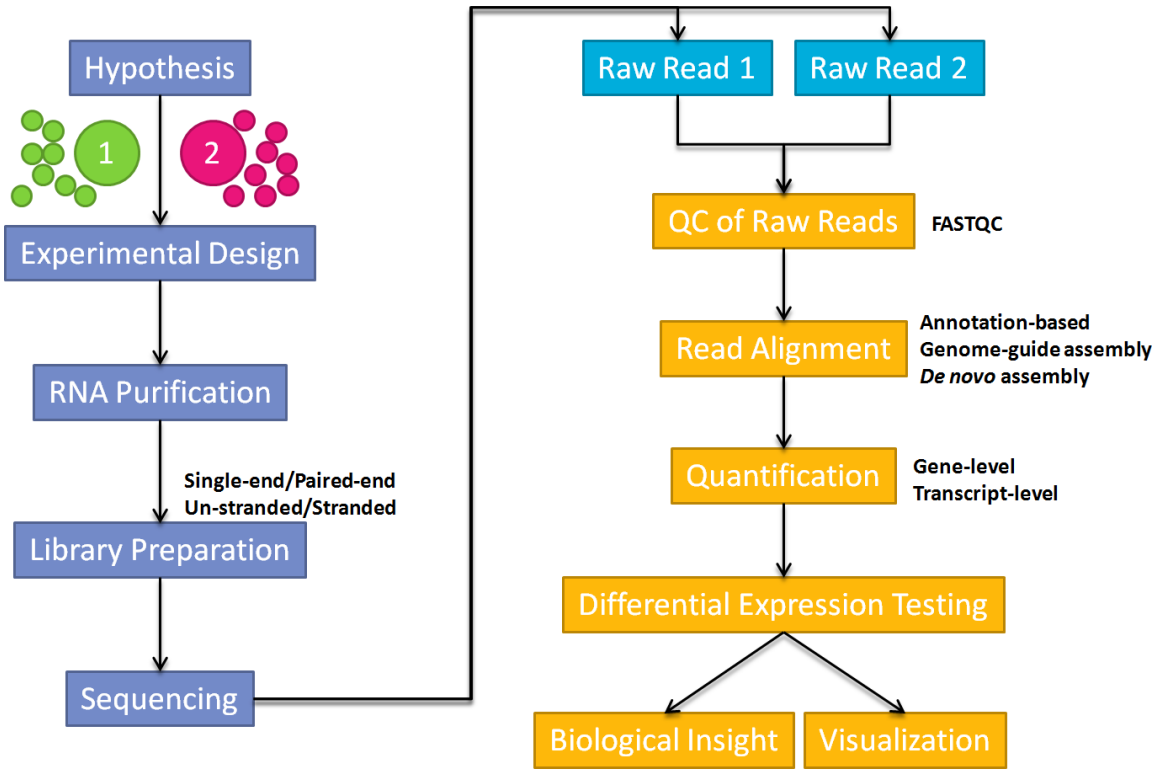
_____

_____

_____

RNA-SEQ ANALYSIS WORKFLOW

Once you have conducted the sequencing experiment there are a number of steps that need to be conducted in order to take your data from thousands of 'reads' or sequences to real genes that have been up or down-regulated in one treatment versus another. The diagram below shows the general RNA-seq analysis workflow. In this manual, we will focus on normalization and analysis, and clustering class discovery. We will also begin to look at mechanistic analysis by conducting analysis of biological pathways in the data sets. In the sections below, we will explain each analysis step in more detail.



ANALYSIS PLATFORM (GALAXY)

Galaxy is a powerful open-source computational analysis program that was developed by the Nekrutenko lab in the Center for Comparative Genomics and Bioinformatics at Penn State, the Taylor lab at Johns Hopkins University, and the Goecks Lab at Oregon Health & Science University, along with contributions from the community. The program provides access to hundreds of -omics analysis tools that can be used to analyze microarrays, next generation sequencing, proteomics, metabolomics, imaging, and many more assays. Furthermore, users can generate their own 'workflows' that allow specific modules to be used together to create an analysis workflow.

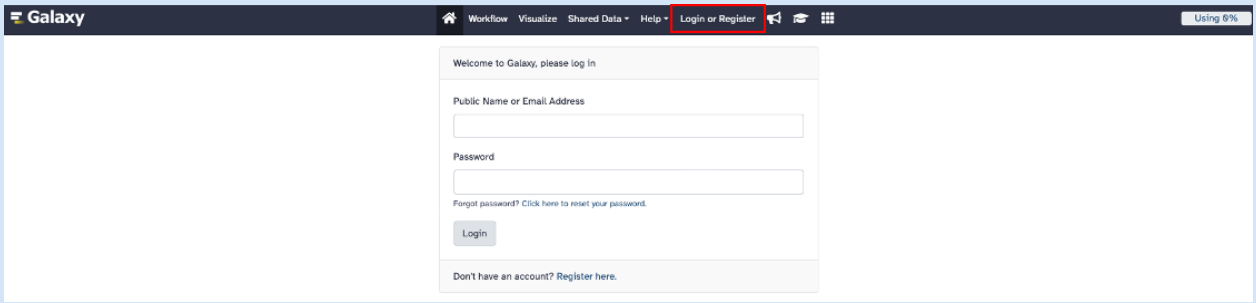
Exercise 5: Accessing the Analysis Platform

For this analysis, we will use the public version of Galaxy (www.usegalaxy.org) The Galaxy Platform provides tools for transcriptomics analysis. The tools and workflows are part of the standardized pipelines used by GeneLab and the Analysis Working Groups.

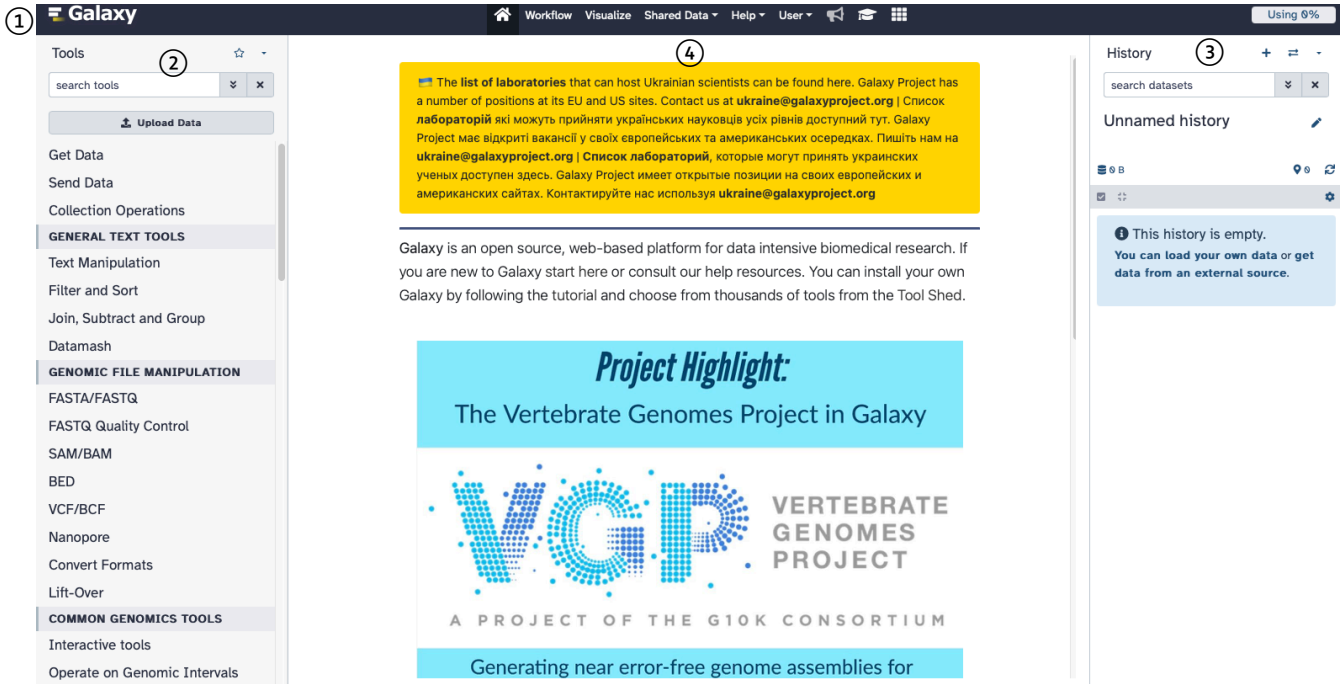
To access the Analysis Platform, navigate to the Galaxy homepage (www.usegalaxy.org) and click on the 'Login or Register' button.

1. To register click on the 'Login or Register' button and then click 'Register here'.
2. Create a Galaxy account using your email address.
3. Navigate back to: www.usegalaxy.org and sign in.

You have now successfully created and logged into Galaxy account. From now on, you will just need to navigate to www.usegalaxy.org and log in to use the analysis platform.



When first opened in your browser, Galaxy displays the home page. To return to this page at any time, click the 'Galaxy' logo in the top right corner or the home icon. See below for a broad description of the Analysis Platform.



The **menu bar** ① includes:

- ***Workflow***: Click this link to navigate to the list of public workflows
- ***Visualize:*** Click this link to see the available visualization tools
- ***Shared Data:*** Click here for menu of links to privately or publicly shared data (Data Libraries, Histories, Workflows, Visualizations, Pages)
- ***Help***: Click here for tutorials and support
- ***User***: Profile and account information

- Galaxy release notes
- Galaxy training materials
- Enable/disable window manager

The **Tools panel** ② provides a list of all the available tools

The **History panel** ③ provides access to your running or completed jobs

The **center panel** ④ is the main display panel. Galaxy uses this pane to display information and to prompt you for input. Initially, the center panel contains a welcome page, which provides instructions for running common analysis protocols. To redisplay the welcome page, click on 'Analyze Data' icon in the menu bar.

KEY GALAXY ACTIONS

Before we begin analysis, let's start by getting a little more familiar with the functions and layout of the Galaxy Platform.

Exercise 6: Getting familiar with Galaxy

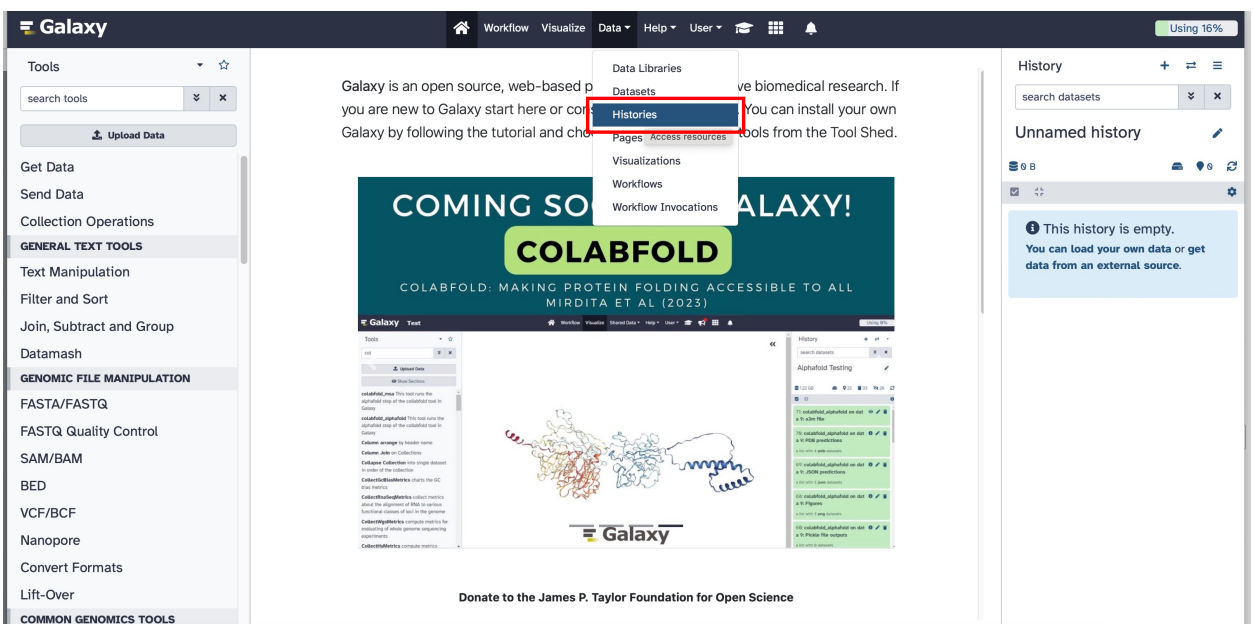
Go to Galaxy and log in. Then, start by taking a short introduction tutorial of Galaxy.

1. Go to Galaxy training materials on the top menu bar.
2. In the pop-out window select along the top menu bar 'Introduction to Galaxy Analyses'.
3. Under Lesson, select the 'play' icon next to 'A short introduction to Galaxy'.
4. Watch the video.
5. By selecting 'A short Introduction to Galaxy', you can also read though the tutorial.

Below you will find key actions that we will use throughout the analysis workflow.

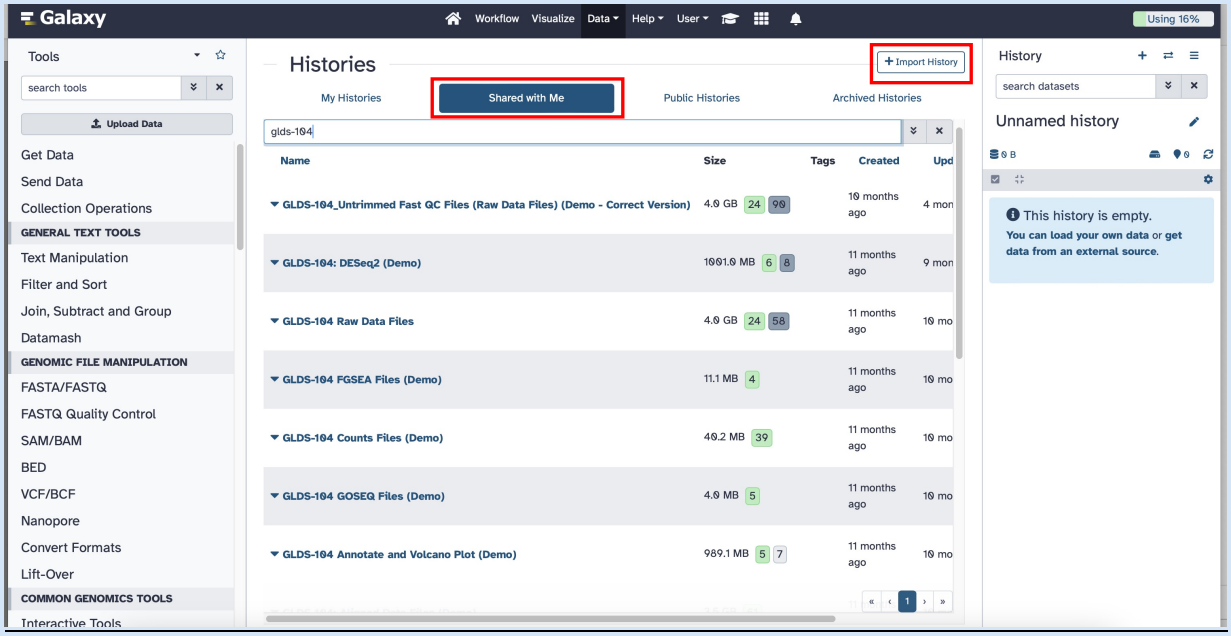
Getting Data

For this manual, data will be accessed using **Histories** under **Data** on the top panel. To access the shared data, navigate to Data -> Histories. We will use this to import some of the data for specific analysis steps.

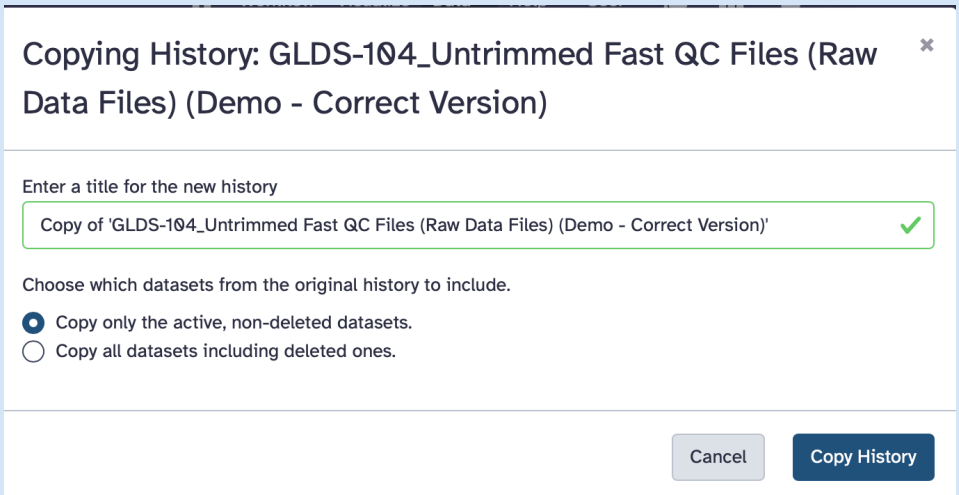


Exercise 7: Importing Data

In Galaxy go to the top menu and select **Data -> Histories.** Then navigate to the **Shared with Me** tab under **Histories**. Search for and select 'GLDS-104 *Raw Data* Files'. This history will have the raw data files. For each of the analysis steps the 'Raw Data' will be replaced with the name of the specific analysis that was run e.g. 'GLDS-104: *Trimmed* Data Files' contains the results of trimming.



To import the history, first select the dataset you wish to import and then select 'view'. Click on **Import this history** on the left side. In the pop-up window – Importing history "GLDS-104 *Raw Data* Files", enter a new history name or leave as is and select **Copy History**.

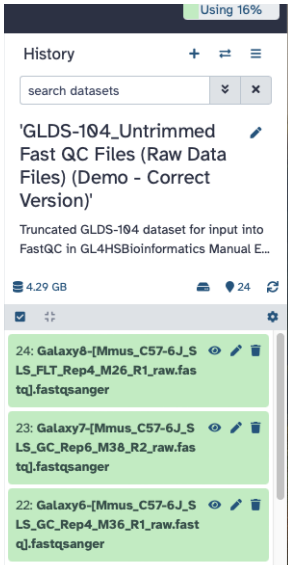


Once you select 'Copy History', navigate you back to the home page and the History panel on the right side of the screen will be populated with the imported data files.

Features of the History Panel

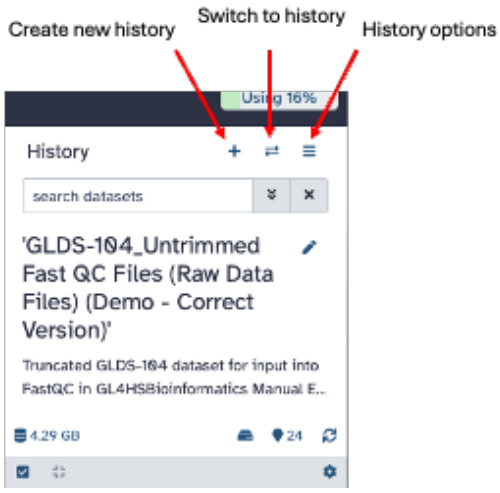
The Current History

All users have one 'current' history, which can be thought of as a workspace or a current working directory in bioinformatics terms. Your current history is displayed in the right-hand side of the main 'Analyze Data' Galaxy interface in what is called the history panel shown below.



The history panel displays output datasets in the order in which they were created with the oldest/first shown at the bottom. As new analyses are done and new output datasets are generated, the newest datasets are added to the top of the history panel. In this way, the history panel displays the history of your analysis over time.

Users can have many histories and the history panel allows switching between them and creating new ones. This can be very useful to keep your analyses organized. Above the current history panel are three buttons: create new history, switch to history and history options. Under history options, you can also choose to view all histories.



Histories also store information in addition to the datasets they contain. They can be named/re-named, tagged, and annotated.

Renaming a history

All histories begin with the name 'Unnamed history' or the name of the imported history. As more analyses are conducted on different datasets, the name of the history becomes very important to remember what was done and to which dataset. Therefore, it is very useful to rename the history at the start of your analysis.

Exercise 8: Renaming the history

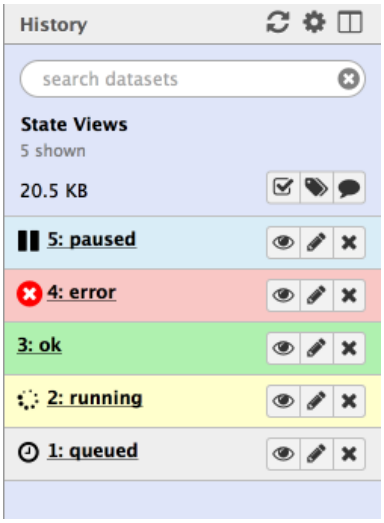
To rename the history follow these steps:

1. Click on the pencil icon next to 'Unnamed history'. A text input field will appear with the current name.
2. Enter a new name or edit the existing name in the way you'd like.
3. Press 'Enter' or save to save the new name. The input field will disappear, and the new name will display immediately.
4. To cancel renaming, press 'ESC' or 'cancel'.

Dataset states

There are several states that a dataset can display in the history panel.

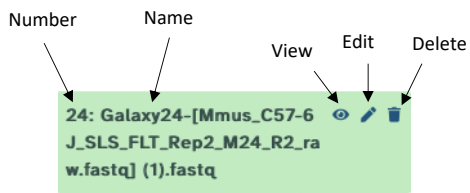
- ***Queued***: When a file is first uploaded or a tool is first selected for running, but the job has not commenced yet, then the dataset will be in the **queued** state. This indicates that the job that will create this dataset has not yet started and is in line to begin.
- ***Running***: When the job starts the dataset will be in the **running** state. The job that will create these datasets is now running on Galaxy's cluster.
- ***Ok***: When the job has completed successfully, the datasets will be in the **ok** state.
- ***Error***: If an error occurred while running the tool, the datasets will be in the **error** state.
- ***Paused***: If a previously running or queued job has been paused by Galaxy, the dataset will be in the **paused** state. You can re-start/resume paused jobs using the options menu above the history panel and selecting 'Resume Paused Jobs'.



Summary view

Datasets in the panel are initially shown in a 'summary' view that only displays:

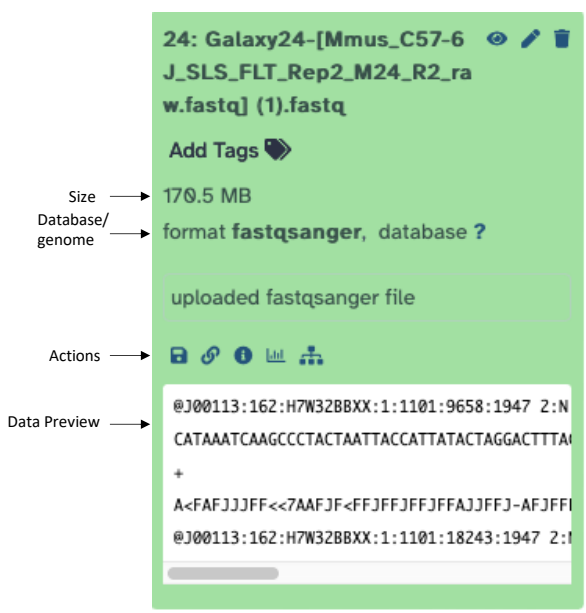
- ***Number*** indicating in what order (or what step) this dataset was created .
- The ***dataset name.***
- ***View*** button: click this to view the dataset contents in raw format in the browser.
- ***Edit*** button: click this to edit dataset properties.
- ***Delete*** button: click this to delete the dataset from the history (don't worry, you can undo this action).



Expanded view

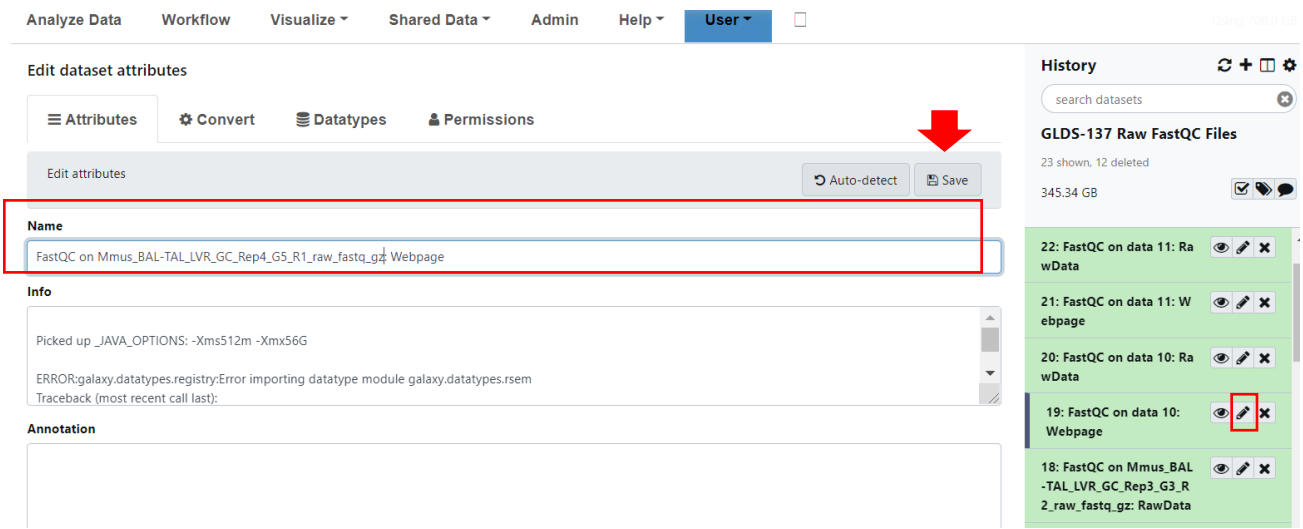
If you click on the dataset name, the view will expand to show more details:

- A short ***description*** of the data.
- The ***file format.***
- The ***reference sequence*** (or database) for the data.
- Some ***information***/output from the job that produced this dataset.
- Row of buttons that allow further ***actions*** on the dataset.
- ***Data Peek***: a couple of rows of data with the column headers (if available).



Renaming outputs in your History.

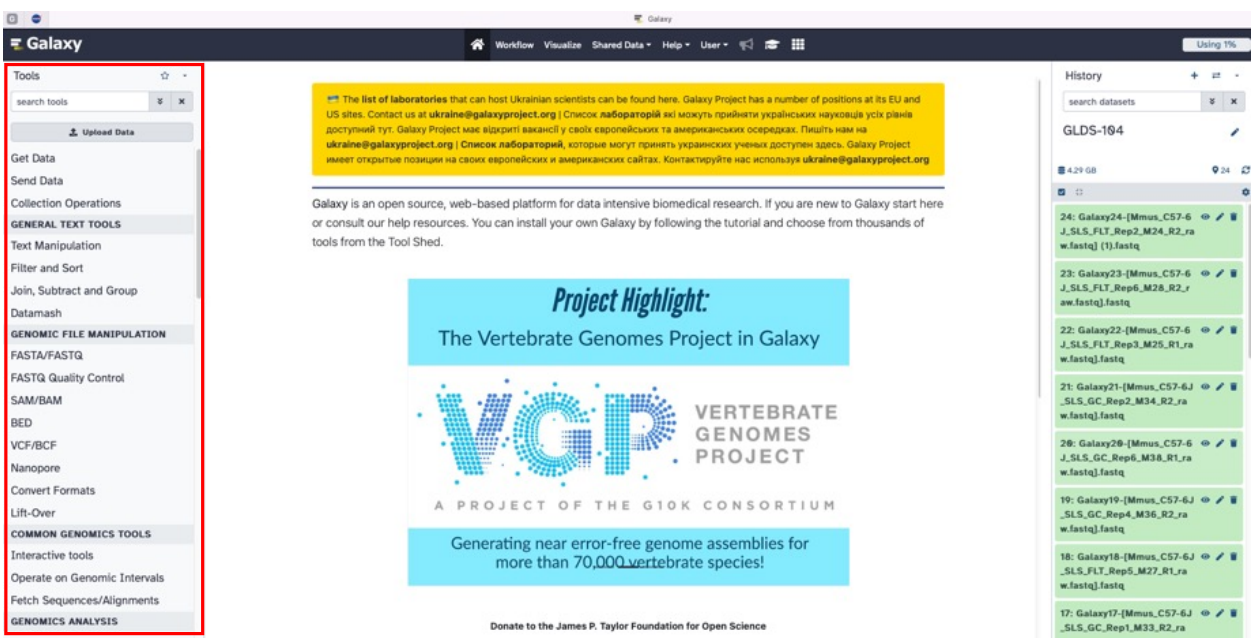
It is important to rename the outputs of your jobs so that you can keep track of the samples through the analysis. To rename the new files, click on the pencil . Here you can edit the sample name but keep the other attributes like the step (i.e. FastQC) and type of file (i.e. Webpage) the same. Once the samples have been renamed, click **Save**.



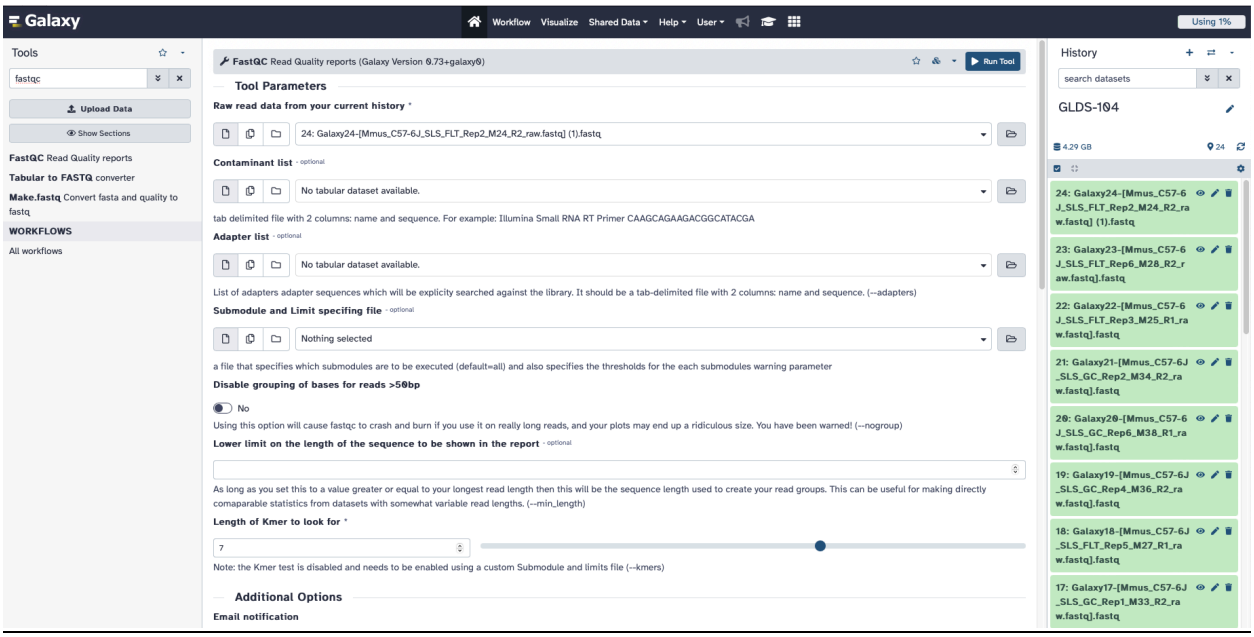
Remember to rename your outputs after each step.

Selecting a Tool

To select a tool to use for analysis of your dataset, navigate to the tools column and then click on the category. Within the categories, you will find a full list of tools that can be used.



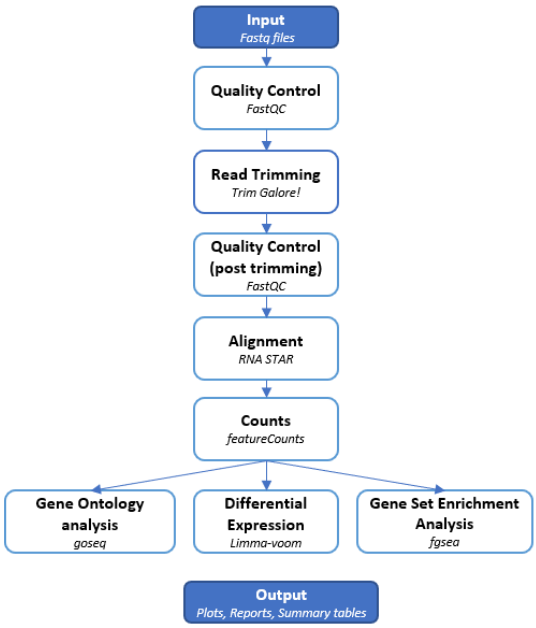
The search bar can also be used to find a tool if you know the name. To select a tool, click on the name of the tool and the tool parameters will populate the center panel. For example, if we search for 'FastQC' in the Tools box and then select the **FastQC** tool, the parameters and options for the **FastQC** tool will populate the center panel. Each tool also contains additional information including description of the tool, parameters, and author(s). We will learn how to use RNASeq tools, including selection of specific analysis parameters in the next section.



Notes:

ANALYZING RNA-SEQ DATA

In this tutorial, we will use the workflow below to go from raw fastq data to differential expression data and gene set enrichment analysis [2]. You will learn about the functions of each tool, input and outputs, and how to interpret the results.



To get started, let's take a look at the quality of our input files – Fastq Raw files. For the exercises in this manual, we will use a subset of reads from the original data files, so the run times per step are shorter.

CHECKING QUALITY AND TRIMMING DATA

During sample preparation and sequencing a variety of errors can be introduced leading to poor data quality. In order to prevent conducting time-consuming analysis on low quality data, or to enable low quality reads to be pre-processed to enable more accurate analysis, we need to pre-process the data and determine its quality. The main sources of errors are listed below.

1. Sample/library Preparation
    a. Sample quality
    b. RNA isolation
    c. PCR artifacts
    d. Sequence contamination
2. Sequencing errors
    a. Low quality reads
    b. Low-confidence bases
    c. Sequence specific bias

d. 3'/5' positional bias
e. Untrimmed adaptors

These errors can affect mapping to the reference genome as well as assembly and expression estimates, however, many of these errors can be corrected for by trimming, filtering or bias correction. Some problems cannot be corrected for and therefore, the researcher needs to determine whether to proceed with analysis, and if so, they need to be aware of how these errors may affect the data.

HOW DOES FASTQC WORK?

FastQC is one tool that has been developed to assess read quality. It is available as a standalone Java program and has also been integrated into Galaxy and several other platforms. The input files for **FastQC** can be fastq files or SAM/BAM files. Fastq files contain 4 lines per read or sequence:

Line 1: begins with @ and is followed by the sequence identifier/read name

Line 2: raw sequence (bases). The number of bases is equal to the cycles in the sequencing run

Line 3: begins with + and read name

Line 4: characters which represent the quality of the bases (ASCII characters).

Every base within the read has a confidence value assigned to it which tells you how likely it is that the base call is wrong. The is on the Phred quality scale ($Q= -10*\log_{10}$). For example, if there is a 1 in 100 chance of the base is wrong, the quality value is $Q = -10*\log_{10}(0.01) = 20$. Quality values typically range from 0 to 40 and aren't shown as numbers due to the complexity of assigning double digits and spaces to millions of sequences. Therefore, the values are converted to ASCII characters (usually done using Sanger encoding where the 33rd ASCII character is used as 0). It is important to know what quality coding is used because several of the analysis methods require this information to be inputted. If you use the wrong encoding method, then the software will assign the wrong quality score. **FastQC** uses these scores to assign the quality of each base within the read with every read. It uses several different plots to enable us to determine the overall quality of our data. Generally, the base quality decreases in the later cycles of sequencing and this is easy to detect in the **FastQC** results files. We can also inspect the distribution of base qualities per base position and check how the reads mean quality is distributed. This allows us to determine if a subset of reads have an overall bad quality. Reads containing low-quality bases can then be filtered or trimmed.

The output data from **FastQC** contain the following plots/data:

1. **Sequencing Statistics**
   Describes basic statistics about the sequencing run including what type of base calls, the number of sequences processed, their size and GC content.
2. **Per base sequencing quality**
   The per base sequencing quality score tells us the range of quality values across all bases at each position in the file. In this plot, the x-axis represents the individual bases for the read, whilst the

y-axis plots the quality score. For each base call i.e. base position, the software plots the distribution of quality values. These are displayed as box plots where the yellow box represents the inter-quartile range (25-75%), the black whiskers represent the 10-90th percentile range, the blue line shows the mean and the red line plots the median. The higher the score the better the base call. The background of the graph shows where good quality data should reside (green), average quality (yellow) and poor quality (red). For good quality data, the quality scores should remain consistently high across all reads and throughout the reads. Ideally, we want all quality scores to be >20-25. **Warnings** are issued if the lower quartile for any base is <10 or if the median for any base is <25. **Failures** are shown if the lower quartile for any base is <5 or if the median for any base is <20.

3. **Per sequencing quality score**
   This report allows you to determine the quality score per sequence. It takes every sequence and works out the mean score across every base in the sequence and then plots the distribution of the means. In a good quality sequence, you see tight distribution with uniformly high quality (x-axis). Bad sequences may have high average quality score but may also have a second population with lower quality scores. This second subset can be filtered out if desired. A warning is raised if the most frequently observed mean quality is <27 (0.2% error rate). A failure error is shown when the most frequently observed mean quality is <20 (1% error rate)
4. **Per base sequencing content**
   This report plots the proportion of each base position in the file (i.e. A, T, G, C). In a diverse library, you would expect there to be an even distribution of the 4 bases that does not change with the base position. If this is the case, the lines in the plot would run parallel to each other. In a biased library, there may be spikes present in the plot indicating higher proportion of one or more bases over the others. However, there are some types of libraries that will always produce a biased sequence composition at the state of the read. This generally can't be corrected by trimming but does not usually affect downstream analysis. A **warning** will be issued in this module if the difference between A and T or G and C is greater than 10% in any position. A **failure** is issued if the difference between A and T or G and C is greater than 20% in any position.
5. **Per sequence GC content**
   This plot measures the distribution of GC content across the whole length of each sequence and models it against a normal GC content distribution. The red line is the data of interest and the blue line is the theoretical normal distribution with same mean and standard deviation as the library of interest. The central peak should correspond to the overall GC content of the underlying genome. An unusual distribution, such as secondary peaks, could indicate contaminants in the library (e.g. adaptor dimers). A **warning** is raised if the sum of the deviations from the normal distribution represents more than 15% of the reads. A **failure** is indicated if the sum of the deviations from the normal distribution represents >30% of the reads.
6. **Per base N content**
   This plot tells you how many uncalled bases are present in the library. Generally, there should not be any, but it is not unusual to see a low proportion of Ns occurring towards the end of the sequence. **Warning** occurs if any position shows N content >5%. **Failure** occurs if any position shows N content >20%.

7. **Sequence length distribution**
   This plot tells you the distribution of fragment sizes i.e. are the sequences all the same length or not. Generally, sequencers generate fragments of the same length, but others can generate sequences of different lengths. It is also useful to perform this analysis after trimming to ensure accurate fragment size.
8. **Sequence duplication levels**
   This graph tells you how unique your sequences are in the library. In a diverse library, most sequences will occur only once. Overall duplication level is shown at the top. A high level of duplication is likely to indicate some sort of enrichment bias e.g. PCR over amplification. A warning Is issued if non-unique sequences make up >20% of total. A failure is issued if non-unique sequences make up >50% of total.
9. **Overrepresented sequences**
   This module lists all of the sequences that are overrepresented within the set i.e. they represent >0.1% of total sequences in library. For each sequence, it lists the sequence, number of times observed and % of library it represents. These sequences are also screened against potential contaminants e.g. primers and adaptors. **Warning** will be issues if any sequence is found to represent >0.1% of total. **Failure** will be issued if any sequence is found to represent >1% of total.

The **FastQC** tool contains multiple parameters:

- ***Short read data from your current history***: This will pull files from your history for use in the tool. There are three methods to do this – single dataset, multiple datasets and dataset collection.
- ***Contaminant list:*** Here you can upload a tab delimited file containing sequences of potential contaminants that may be present as a result of library preparation.
- ***Adapter list:*** Here you can upload a tab delimited file containing the adapter sequences used during library preparation.
- ***Submodule and Limit specifying file.***
- ***Disable groupings of bases for reads >50 bp***: Yes or no – usually left as default.
- ***Lower limit on the length of the sequences shown in the report:*** This should remain longer than the longest read in order to be used for generation of read groups
- ***Length of kmer to look for***: The kmer test can only be used if a custom submodule and limits file is added.
- ***Email notification***: This will send an email notification once the job is complete.

Exercise 9: Checking the quality of the data

Go to Galaxy and log in. Navigate to **Data** -> **Histories -> Shared with me** and select **GLDS-104_Raw Data Files**. Here you find all the datasets we will use through this course. To learn how to analyze RNA-seq data we will use the GLDS-104 dataset, which is a truncated dataset enabling the data processing to run more rapidly than it would on a full dataset.

This dataset contains two sets of samples (as we learned when looking at the metadata).

GLDS-104 Spaceflight (FLT) Samples:

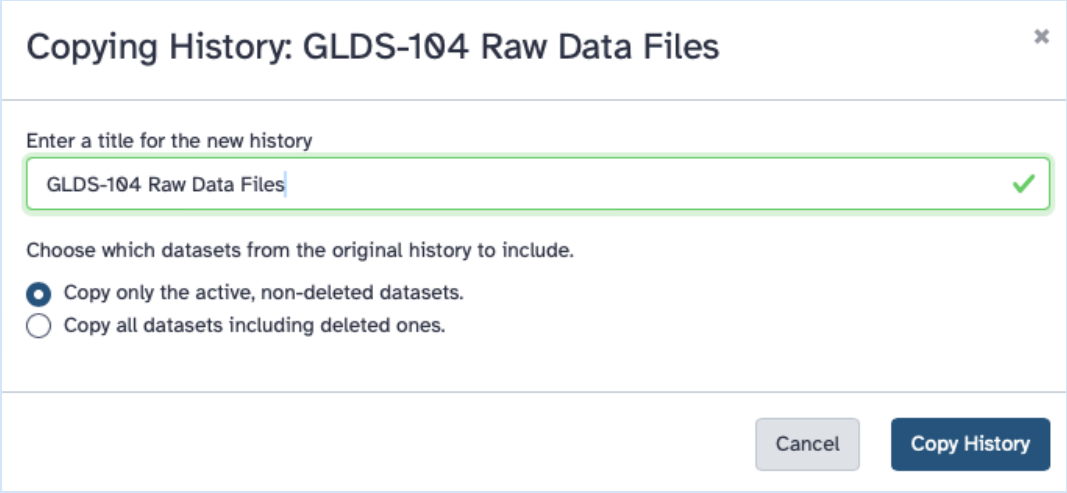
Mmus_C57-6J_SLS_FLT_Rep1_M23_R1_raw.fastq.gz, Mmus_C57-6J_SLS_FLT_Rep1_M23_R2_raw.fastq.gz
Mmus_C57-6J_SLS_FLT_Rep2_M24_R1_raw.fastq.gz, Mmus_C57-6J_SLS_FLT_Rep2_M24_R2_raw.fastq.gz
Mmus_C57-6J_SLS_FLT_Rep3_M25_R1_raw.fastq.gz, Mmus_C57-6J_SLS_FLT_Rep3_M25_R2_raw.fastq.gz
Mmus_C57-6J_SLS_FLT_Rep4_M26_R1_raw.fastq.gz, Mmus_C57-6J_SLS_FLT_Rep4_M26_R2_raw.fastq.gz
Mmus_C57-6J_SLS_FLT_Rep5_M27_R1_raw.fastq.gz, Mmus_C57-6J_SLS_FLT_Rep5_M27_R2_raw.fastq.gz
Mmus_C57-6J_SLS_FLT_Rep6_M28_R1_raw.fastq.gz, Mmus_C57-6J_SLS_FLT_Rep6_M28_R2_raw.fastq.gz

GLDS-104 Ground Control (GC) Samples:

Mmus_C57-6J_SLS_GC_Rep1_M33_R1_raw.fastq.gz, Mmus_C57-6J_SLS_GC_Rep1_M33_R2_raw.fastq.gz
Mmus_C57-6J_SLS_GC_Rep2_M34_R1_raw.fastq.gz, Mmus_C57-6J_SLS_GC_Rep2_M34_R2_raw.fastq.gz
Mmus_C57-6J_SLS_GC_Rep3_M35_R1_raw.fastq.gz, Mmus_C57-6J_SLS_GC_Rep3_M35_R2_raw.fastq.gz
Mmus_C57-6J_SLS_GC_Rep4_M36_R1_raw.fastq.gz, Mmus_C57-6J_SLS_GC_Rep4_M36_R2_raw.fastq.gz
Mmus_C57-6J_SLS_GC_Rep5_M37_R1_raw.fastq.gz, Mmus_C57-6J_SLS_GC_Rep5_M37_R2_raw.fastq.gz
Mmus_C57-6J_SLS_GC_Rep6_M38_R1_raw.fastq.gz, Mmus_C57-6J_SLS_GC_Rep6_M38_R2_raw.fastq.gz

In **Histories** under **Shared with me**, click on the file labeled **GLDS-104: Raw Data Files** . Click '**view**' and then on the top left click '**Import this history**'.

1. In the pop-out window, rename the history to 'GLDS-104: Untrimmed FastQC files.
2. Select '**Copy only the active, non-deleted datasets'.**
   Then click on 'Copy History'.



Exercise 9 cont'd: Checking the quality of the data

3. Once complete, navigate back to the 'Home' page to begin analysis of the dataset.
4. In the history bar on the right side of your screen, all of the data files should be there and highlighted as green (**Ok status**). All your files should now be in your history and all green.

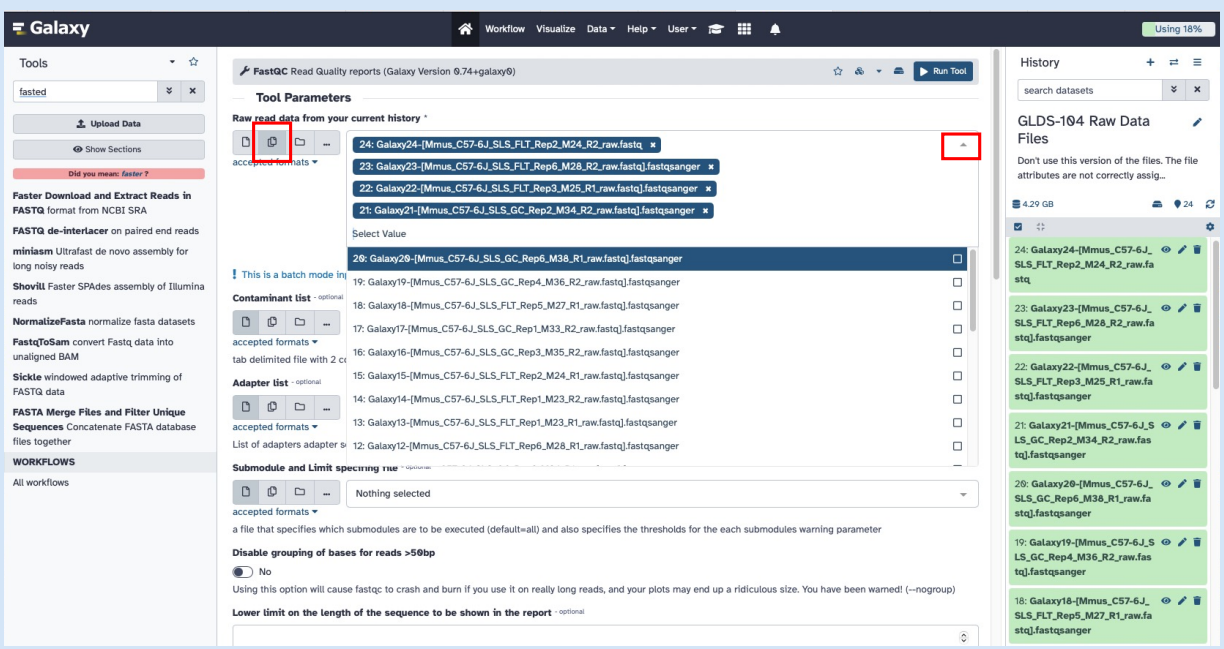
Now that the data has been imported into our history, we need to check the quality of the data using the **FastQC** tool.

In the Tools panel, type **FastQC** in the Tools search bar. This will populate the center panel with the tool parameters.

For this analysis, we will use the following parameters:

- **Short read data from your current history –** this will pull in the files in your history.
- All other parameters to be left in the default setting.

1. Short read data from your current history – click on the second icon from left – **multiple datasets** and you will see all your files from your history populated here.
2. Click on the down arrow on the right to expand the drop-down menu and see all your files. Then click on each file in your dataset to add them to the analysis.
3. All other parameters should be left at the default settings.
4. Click **Run Tool.**



If green, the job was executed correctly and will start queuing the jobs in your history panel.

Exercise 9 cont'd: Checking the quality of the data

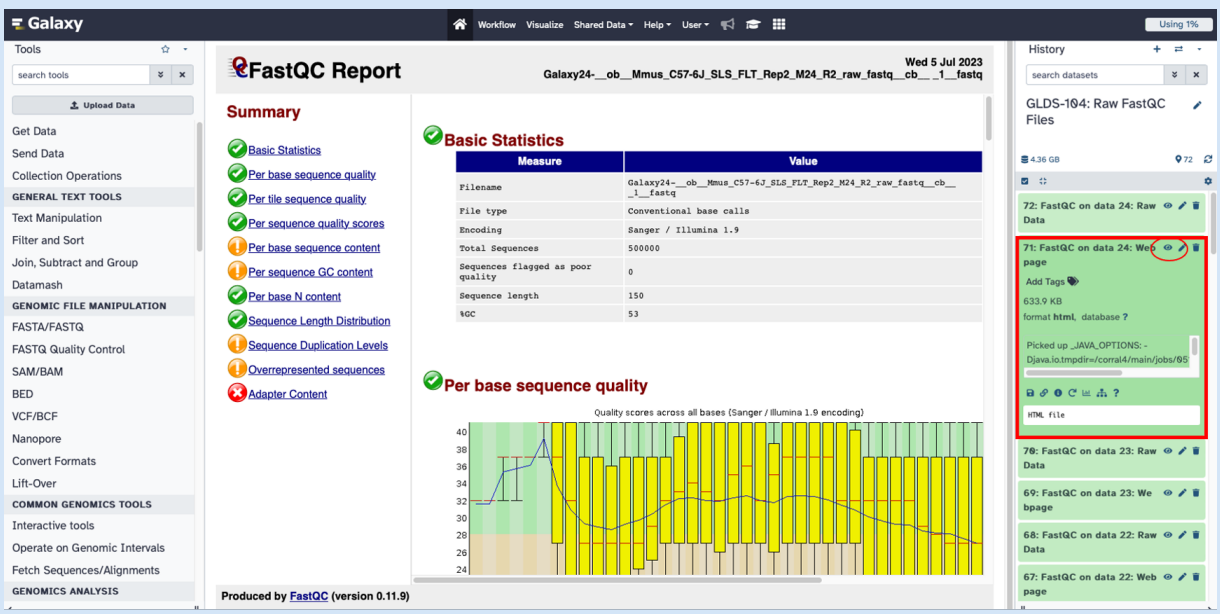
To continue with this tutorial, we will use Published Histories to go to the next step in the analysis. Your jobs will continue running.

Importing Histories

Go to the top menu and select **Data -> Histories -> Shared with me.** Click on the History labeled 'GLDS-104: Raw FastQC Files'. This history will have the raw data files and the **FastQC** files generated. To import the history, click on the link and select 'view'. In tope right side of the import window select 'Import this history' as we did previously. Rename the history.

We are now ready to review the quality of the data.

FastQC will provide two outputs per file (remember you have a R1 and R2 per sample). To view the report, click 👁 of an output that has 'FastQC on *sample name:* Webpage'. This will display the html report of **FastQC**. Note that it doesn't matter which file you select to view.



- To learn about the details of each section, watch How to Check the Quality of Illumina Sequencing Reads with FastQC (Part 2) - https://www.youtube.com/watch?v=GnWSXwQeJ_U
- A detailed description of each section is described by Michigan State University https://rtsf.natsci.msu.edu/sites/_rtsf/assets/File/FastQC_TutorialAndFAQ_080717.pdf

Exercise 9 cont'd: Checking the quality of the data

Once you have learned about the **FastQC** report, select a file to view and navigate through each section:

1. What is the filename?

_____

2. What are the total sequences?

_____

3. Check the Per base sequencing quality. Are there any quality scores below 20? What position in the read (bp) are the majority of the lower scores?

_____

_____

_____

4. Check the Adapter Content. Are there adapters present? Based on the chart, which adapter is most abundant?

_____

_____

_____

Check a few more reports by clicking the on the of an output that has 'FastQC on *sample name*: Webpage'. This will give you a general idea of the quality of all your samples.

Before we get started with trimming, what are paired-end versus single-end libraries?

_____

_____

_____

_____

Notes:

HOW DOES TRIM GALORE! WORK?

Quality control and removal of problematic sequences is vital for the analysis of almost all sequencing platforms. Once we have checked the quality of our data, we can determine what steps need to be taken to ensure data analysis is conducted on only high-quality sequences. If low quality bases are detected at the ends of reads, the simplest way to remove them is to trim reads at a given length or to trim a given number of bases from either end. We can also trim left-over adaptor sequences to ensure that they are not included in our analysis. To do this we can use **TrimGalore! TrimGalore!** uses several steps:

1. **Quality Trimming**
   In this step, low-quality base calls are trimmed off from the 3' ends of the reads before adaptor removal. This removes poor quality portions of the reads.
2. **Adaptor trimming**
   In this step, adaptors are removed using Cutadapt. If the adaptor sequence is not supplied, **TrimGalore!** will auto-detect the adaptors which have been used using the standard adaptors. It does this by scanning the first 1 million sequences of the file and attempting to find the first 12-13 bp of the adaptors:
   Illumina: AGATCGGAAGAGC
   Small RNA: TGGAATTCTCGG
   Nextera: CTGTCTCTTATA
   If the adaptors are not recognized by the system, the default is Illumina for this tool. We can also specify which adaptor sequence was used.
3. **Removing short sequences**
   As quality and adaptor trimming can result in very short sequences, Trim Galore can filter reads based on sequence length (default 20).

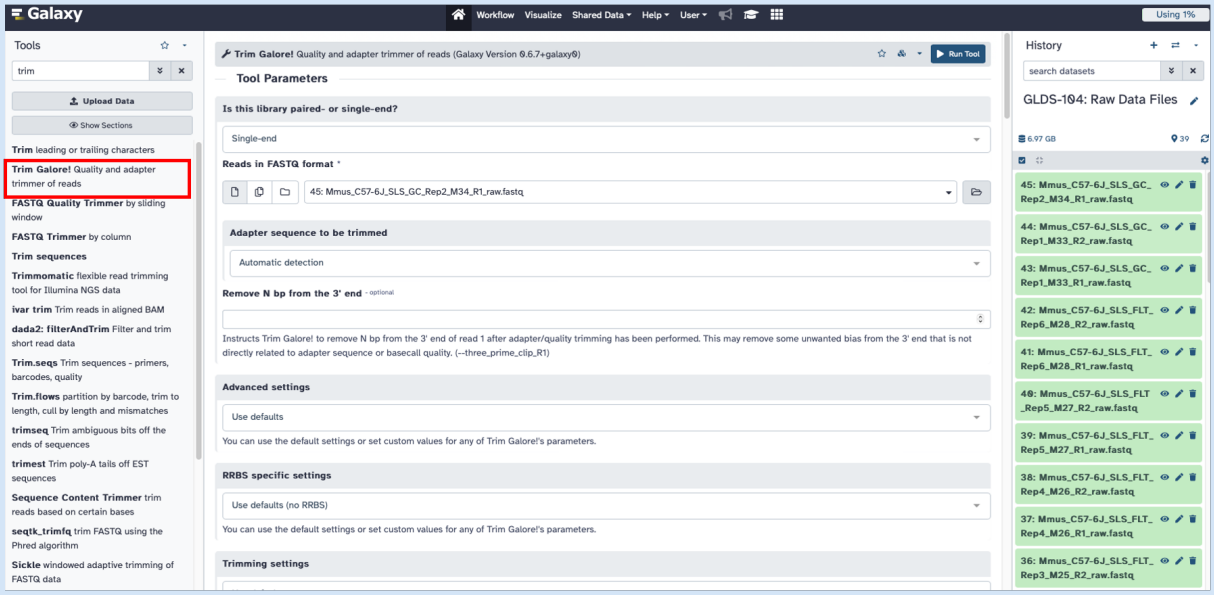
The **TrimGalore!** tool contains multiple parameters:

- ***Is this library paired- or single-end?*** This information is found in the GLDS metadata on the GeneLab website under library layout and is based on your files per sample. If paired, you will have _R1 and _R2 files per sample. For example: in the history you just imported you should have two files with almost identical names – differing only by_R1 and _R2
    - **FastQC on Mmus_C57-6J_SLS_GC_Rep6_M38_R1_Webpage.html**
    - **FastQC on Mmus_C57-6J_SLS_GC_Rep6_M38_R2_Webpage.html**
- ***Reads in FASTQ format*:** This will pull in your FASTQ files from your history. Three methods can be selected to do this – single dataset, multiple datasets, or dataset collection.
- ***Adapter sequence to be trimmed:*** This allows you to select from a list of known adapters or the user can add a custom sequence of adapter to be trimmed off
- ***Remove N bp from the 3' end:*** This option allows you to specify a specific number of base pairs to be removed after both adapter and quality trimming has been conducted.
- ***Trim Galore! advanced settings*:** Additional settings to specify cut offs and to alter the generation of reports.

Exercise 10: Trimming the data

Now that we have checked the quality, we will need to trim low-quality reads and the adapters.

TrimGalore! tool is used to trim, and filter reads. First, we will need to import the raw files again as we did in **Exercise 9**. Then, in the tool panel, search for and select **TrimGalore!**



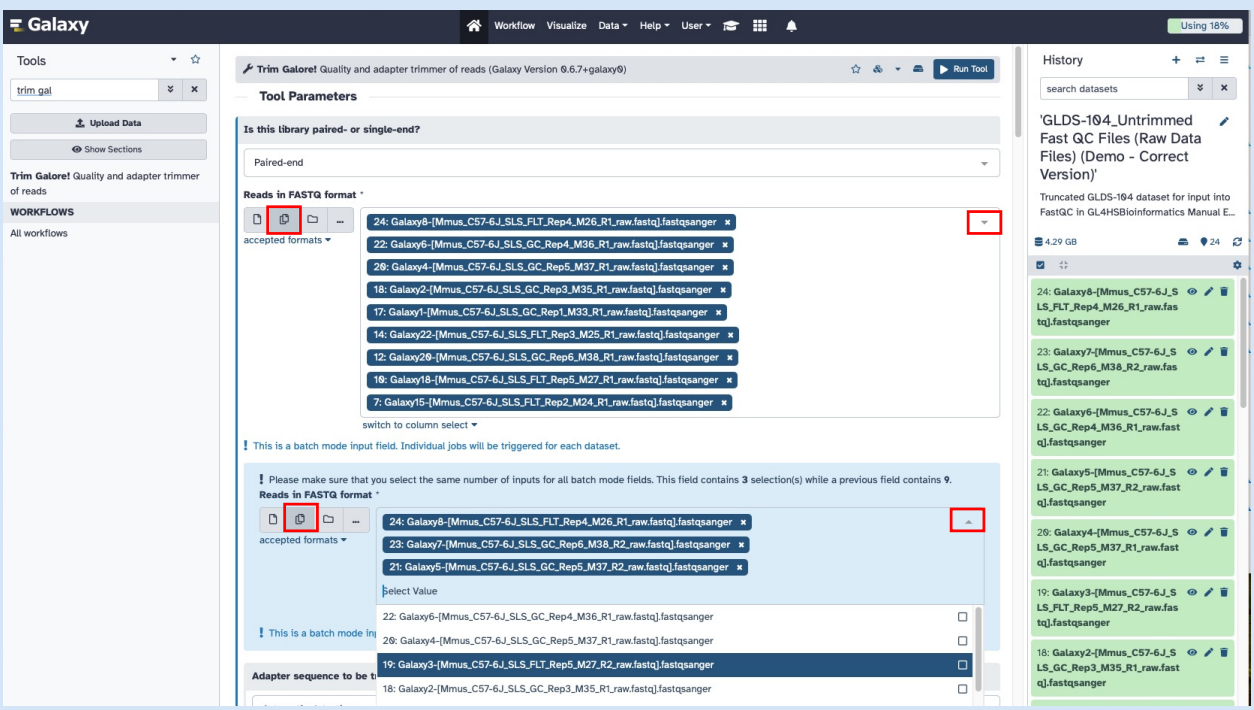
Exercise 10 cont'd: Trimming the data

Once the tool has been loaded, you will see all the options in the center panel. Now we can select parameters and run the job.

1. Is this library paired- or single-end? For this study, it was paired-end. Select **Paired-end** from the drop down.
2. Reads in FASTQ format click on the middle icon – **multiple datasets** and you will see all the files from your history populated here.
    a. For the first Reads in FASTQ format, select all the _R1 files.
    b. For the second Reads in FASTQ format, select all the _R2 files.

To select multiple files, click on the down arrow on the right side to expand the drop-down menu, Then select the files you would like to add by clicking on each file individually. If you select the wrong file, you can remove it by clicking on the 'x' next to the file.

3. Adapter sequence to be trimmed – select **Illumina universal** from the drop-down menu.
4. Trim Galore **advanced settings** – select **Full parameter list** from drop down menu.
5. Scroll down to Generate a report file and select **Yes.**
6. All other parameters to left at the default settings.
7. Click **Run Tool.**



If green, the job was executed correctly and will start queuing your jobs in your history panel.

Exercise 10 cont'd: Trimming the data

To continue with this tutorial, we will use Shared Histories to go to the next step of the analysis. Your jobs will continue running.

Once we have trimmed the data, we need to re-check the quality using **FastQC**.

1. Navigate to **Data -> Histories -> Shared with me.** Select GLDS-104: Trimmed Data Files. Click on the **file name and then 'view'. Then click on 'import this history'** on the far-left corner to add this history to your history.
2. Now the trimmed files, will appear in your current history.
3. To check the quality, we will use the **FastQC** tool again. In the tools panel, search for **FastQC** and select it.
4. Parameters:
    a. Short read data from your current history – click on the middle icon – **multiple datasets** and you will see all your files from your history populated here.
    b. Click on the first file in the listing, hold down shift on your keyboard and click on the last file. This will select all the files listed.
    c. All other parameters to left at the default settings.
5. Click **Run Tool.**

If green, the job was executed correctly and will start queuing you jobs in your history panel. To continue with this tutorial, we will use Published Histories to go to the next step in the analysis. Your jobs will continue running.

Navigate to **Data -> Histories -> Shared with me.** Select GLDS-104: Trimmed FastQC file and import as we have done previously. Rename the history**.** Now the trimmed files, will appear in your current history.

Let's check a few samples by clicking the on the of an output that has '**FastQC** on *sample name*: Webpage'. This will give you a general idea of the quality of all your trimmed samples.

1. Do you see a difference in the quality?

_____

    a. What is the sequence length?

_____

    b. What are the total number of sequences?

_____

_____

Exercise 10 cont'd: Trimming the data

c. Are there any flagged sequences?

_____

2. Per base sequence quality – are there any reads under 20? *Note: the default quality score for trimming reads in Trim Galore! is <20.*

_____

3. Adapter content – do you have any more adapter contamination?

_____

Notes:

_____

_____

_____

_____

_____

_____

_____

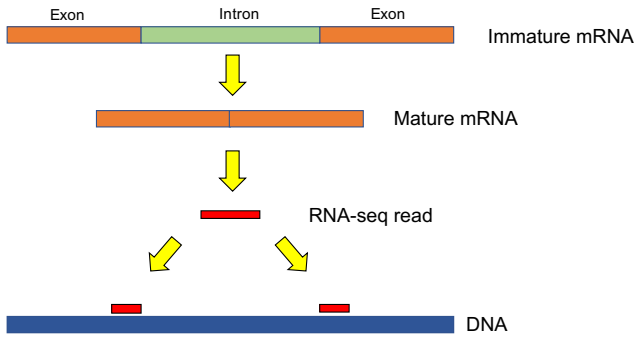
ALIGNMENT

The overall goal of RNA-seq is to reconstruct full-length transcripts based on the sequence reads. Due to current sequencing methods, we can only sequence short fragments in a single unit. Therefore, in order to get a full-length transcriptome, it must be built from short over lapping segments. If there is a reference genome available for the sample, we can use this to guide the assembly. RNA-seq reads are first mapped to the genome and then we must determine which mapped reads correspond to which transcripts. Therefore, once we have conducted QC and trimming of our data, we are now ready to align the sequences to a reference genome. This step enables us to determine where in the genome the reads came from. To do this, we will use **RNA STAR** (Spliced Transcripts Alignment to a Reference).

Alignment is incredibly challenging for a number of reasons – reads are short and there are millions of them, and genomes can be large and contain non-unique sequences such as repeats and pseudogenes, which lowers the map-ability of these areas. Aligners also have to cope with mismatches and indels (insertions and deletions) caused by genomic variation and sequencing errors. Also, many organisms have introns in their genes and therefore, reads align non-contiguously (not touching). Placing spliced reads across introns and determining exon-intron boundaries is difficult, especially considering that introns can be thousands of base pairs long.

There are many different alignment tools available, and these aligners generally use different indexing methods to speed up the process. Many tools consider base quality values when scoring mismatches and they can also make use of the expected distance and relative orientation of paired end reads. The confidence in mapping location is reported as mapping quality ($Q = -10 \log_{10} P$, where P is the probability that the read originated somewhere else.

Spliced aligners for RNA-seq reads use different approaches for aligning spliced reads. What is a spliced read? This is when a sequencing read comes from a splice site where two exons are connected, resulting in the two portions of the read aligning thousands of base pairs away on the genome as shown below:



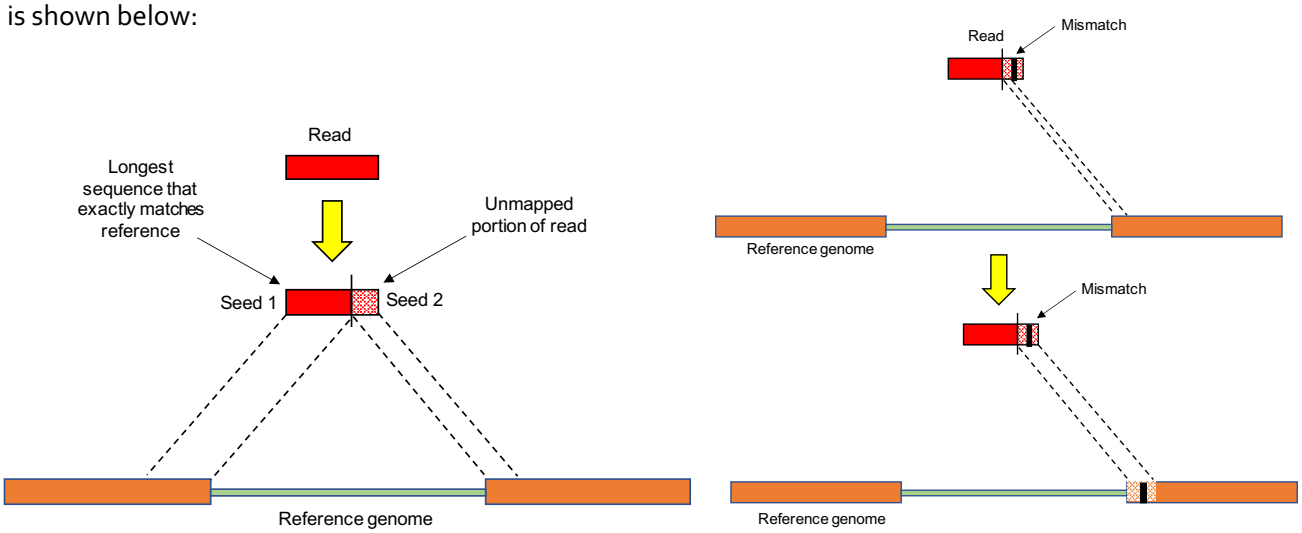
As you can see, spliced alignment is quite difficult because the sequences at the splice sites are limited, but the introns between them can be thousands of bp long. Furthermore, if a read is primarily in one exon with just a small overhang in a second exon, this can pose additional difficulties when attempting to align the read. Spliced aligners use different methods to align these difficult reads. This might include performing an initial alignment to discover exon junctions that can then be used to guide the final alignment. If genomic annotation is available, the aligner can use that to place spliced reads. The main differences between spliced aligners is their alignment yield, base-wise accuracy, splice detection performance, tolerance for mismatches and indel detection.

When deciding which aligner to use, the main factor that should be considered is whether we need to do spliced alignment. If the organism of interest does not have introns, or other types of RNA are of interest e.g. miRNAs, then we can use a contiguous aligner (e.g. bowtie, BWA), but if we need to do spliced alignment then we should use non-contiguous aligners (e.g. STAR, TopHat).

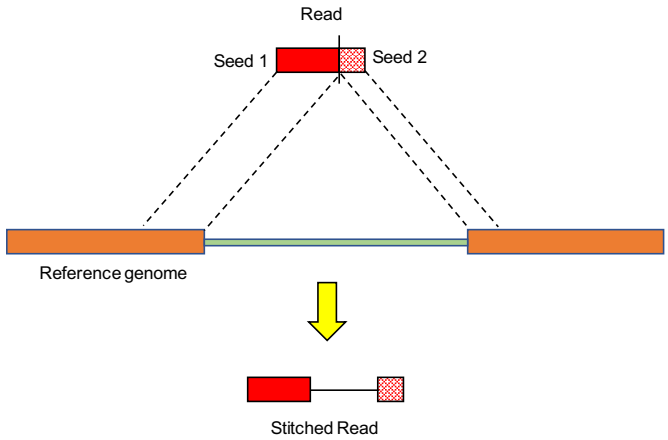
HOW DOES STAR WORK?

STAR (Spliced Transcript Alignment to a Reference) is a super-fast spliced alignment program. It also has several other advantages, including being able to perform an unbiased search for splice junctions without needing any prior information on location, sequence signals, or intron length. It is also capable of aligning a read with multiple splice junctions, indels and mismatches and those with low-quality ends. **STAR** can find the spliced junctions *de novo*, but they can also be supplied to the program when building the reference index

STAR finds splice junctions by using an approach based on 'maximum mappable length'. **STAR** splits a read into pieces (50 bp by default) and finds the best portion that can be mapped per piece. The longest matching sequences are called the maximum mappable prefixes (MMPs). The different parts of the read that are mapped separately are called seeds. The first part of the read that was mapped is called seed 1. It then searches again to find the next longest sequence to that exactly matches the reference genome (MMP) and this will be seed 2. By sequentially searching for the unmapped portions of reads, this makes the algorithm very efficient. It does this by using an uncompressed suffix array (SA) to search for the MMPs which allows for quick searching against large genomes. If there is not an exact matching sequence for each part due to mismatches or indels, then **STAR** will extend the previous MMP. If this still does not give good alignment, then the poor-quality end sequence or adapter sequence will be soft clipped. This is shown below:



Once this is done, **STAR** stitches the seeds together to allow for a complete read. This is done by first clustering the seeds together based on proximity to a set of anchor seeds or seeds that aren't multi-mapping. The seeds are stitched together based on the best alignment for the read, with the score being based on mismatches, indels and splice junctions. This is shown below:



Now that you know how this tool works, we will use **RNA STAR** to align our sequences to the reference genome. For this study, we will use the built in M_mus_GRCm38 reference genome. This step is the most computationally intensive and longest step in analyzing RNA-Seq data.

The **RNA STAR** tool contains multiple parameters:

- ***Single-end or pair-end reads*:** This information is found in the metadata under library layout and based on your files per sample. If paired, you will have _R1 and _R2 files per sample.
- ***RNA-Seq FASTQ/FASTA file, forward reads*:** For paired-end reads, this is your R1.
- ***RNA-Seq FASTQ/FASTA file, reverse reads*:** For paired-end reads, this is your R2.
- ***Custom or built-in reference genome*:** A custom genome can be uploaded, or you can use a built-in index.
- ***Reference genome with or without an annotation*:** This enables you to select from the list of available indexes that were built with splice junction information or without.
- ***Select reference genome:*** This contains a list of available reference genomes.
- ***Gene model file for splice junctions:*** This enables a file to be uploaded with exon junction information.
- ***Length of the genomic sequence around annotated junctions:*** Length of reads i.e., number of base pairs.
- ***Use 2-pass mapping for more sensitive novel splice junction discovery:*** This can be used for studies with multiple samples to increase sensitivity.
- ***Per gene/transcript output:*** This provides extra analysis results related to the genes and transcripts of a gene model.
- ***Report chimeric alignments?*** This enables chimeric alignments to be reported.
- ***BAM output format specification:*** This determines what is contained within the BAM output file.
- ***Output filter criteria:*** This enables specific things to be excluded from the output file.

RNA STAR produces five outputs per sample:

- **.bam file** – sequence alignment data in a binary format that contains information such as read name, read sequence, read quality, alignment information, and custom tags. This will be used to generate count data.

- **log.txt** – contains information about the alignment such as uniquely mapped reads, read length, splice isoform info, unmapped and chimeric read info.
- **transcriptome-mapped.bam –** sequence alignments translated into transcript coordinates.
- **chimeric junctions** – chimeric junction data points – position, score, type.
- **splice junctions.bed**- provides the data that can be visualized using the integrative genomics viewer (IGV).

Notes:

Exercise 11: Aligning to a reference genome.

To get started we will need to switch histories to trimmed reads.

1. Navigate to **Data -> Histories -> Shared with me.** Select GLDS-104: Trimmed Data Files. Click on the **file name and then 'view'. Then click on 'import this history'** on the far-left corner to add this history to your history.
2. Rename the history. Now the trimmed files, will appear in your current history.

Alternatively, you can switch histories to where the trimmed files are:

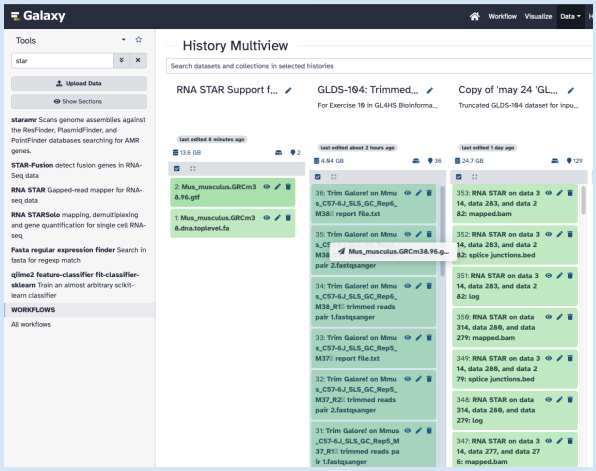
1. In the history panel. Click on 'switch to history ' (middle icon).
2. In the pop-out window, find the history that contains **GLDS-104: Trimmed data files.**
3. Click on that history and it will be loaded into your current history.

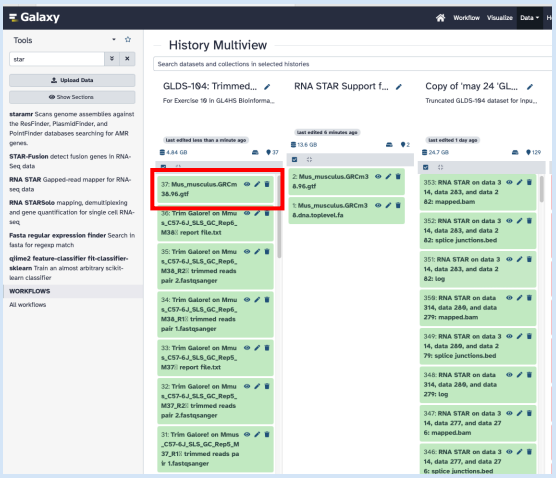
Navigate back to the main page by clicking 'home' at the top of the page.

Adding Genome and Annotation files between histories.

We first need to import the associated genome and annotation files to perform alignment. For this exercise, we will use genome and annotation files from ensembl - http://useast.ensembl.org/Mus_musculus/Info/Index. These files are available in shared histories. **Data -> Histories -> Shared with me**. Click on the **'RNA STAR Support Files'** folder and select view. Then click 'import this history' on the top left.

Next, click on the three horizontal bars (right icon) in the history panel and select '**show histories side-by-side**'. This will display all your histories next to each other and allow you to drag files between your histories. For this exercise you will need to have the RNA support files that you just imported and the GLDS-104 trimmed data files in the same history. Locate these two histories. Two add a file from one history to another, locate the file you require, left click and hold on the file and drag it into the destination history. For this exercise, locate the file in '**RNA Support Files**' history named '**Mus_musculus.GRCm38.96.gtf**', left click and hold and then drag it into the history with your trimmed data files '**GLDS-104: Trimmed data files**' as shown below.

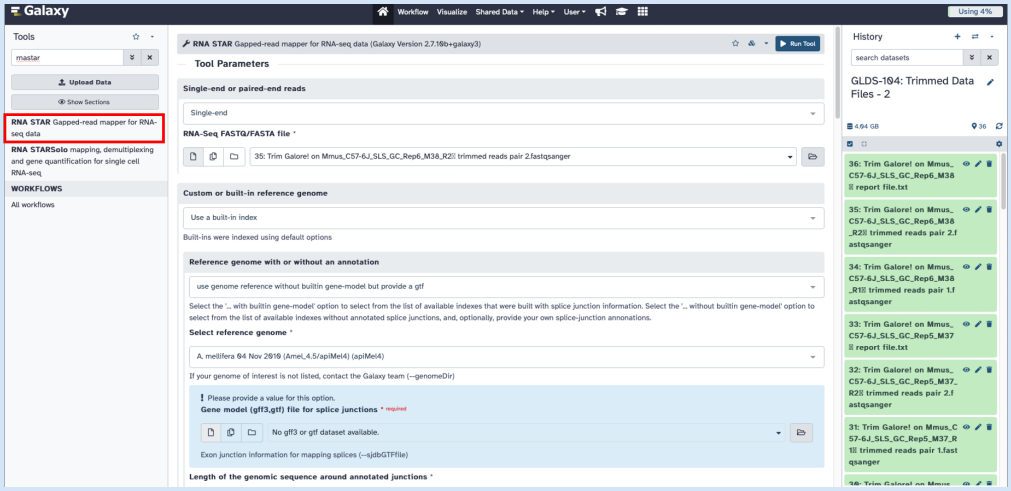




Exercise 11 cont'd: Aligning to a reference genome.

If your file was added correctly, it will be added to the top of your trimmed data files history as shown in the image on the right above. Repeat this for the second file contained within the '**RNA Star Support Files**' History name '**Mus_musculus.GRCm38.dna.toplevel.fa'**. Once your history contains both the trimmed data files and the RNA Support files, we are ready to align our data using **RNA STAR.**

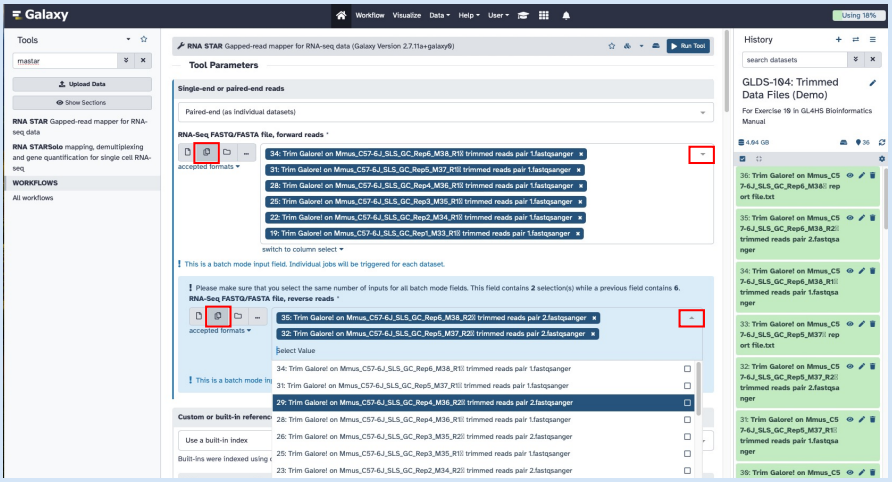
From the tools panel, search for **RNA STAR** or click on **RNAseq_tools** category and find **RNA STAR**.



Once the tool parameters are displayed in the center panel, we can select parameters and begin alignment.

1. ***Single end or paired-end reads***: Select Paired-end (as individual datasets).
2. ***RNA-Seq FASTQ/FASTA file, forward reads:*** For the forward reads, make sure to select only the **_R1** files.
3. ***RNA-Seq FASTQ/FASTA file, reverse reads:*** For reverse reads, make sure to select only the **_R2** files.

You can select the files in batch by using the multiple datasets (second from left icon), clicking on the down arrow on the right and selecting the corresponding files for each section.



Exercise 11 cont'd: Aligning to a reference genome

4. ***Custom or built-in reference genome:*** Select **Use a built-in index** from the drop down
5. ***Reference genome with or without an annotation***: Select **use reference genome from history and create temporary index** from the drop down.
6. ***Select reference genome***: Select **Mus_musculus.GRCm38.dna.toplevel.fa**
7. ***Build index with or without known splice junctions annotation*:** Select **build index with gene model**
8. ***Gene model (gff3,gtf), file for splice junctions*:*** To use the .gtf file (Mus_musculus.GRCm38.96.gtf) from shared histories, follow the guidelines above to add the .gtf file to your current history if you have not already done so.
9. ***Length of the genomic sequence around annotated junctions:*** Enter **149** (150 bp -1)
10. All other parameters should be left as the default options.
11. Since RNA STAR is computationally intensive, we will **NOT** click Execute. We will import the files from **Data -> Histories -> Shared with me**. This will prevent the system from being overwhelmed with 15+ alignment jobs being loaded simultaneously and will reduce your usage of the cloud computing.

Annotated Genomes & Reference Gene Sets

For other model organisms such as Drosophila, Arabidopsis, C. elegans, import the correct annotation from *GL4HS Google Drive > Data Libraries > Annotated Genomes & Reference Gene sets > GTF, GFF.* Select the file associated with the step and import to your history.

To continue with this tutorial, we will use **Data -> Histories -> Shared with me** to go to the next step in the analysis. Your jobs will continue running.

Let's check the results of your alignment:

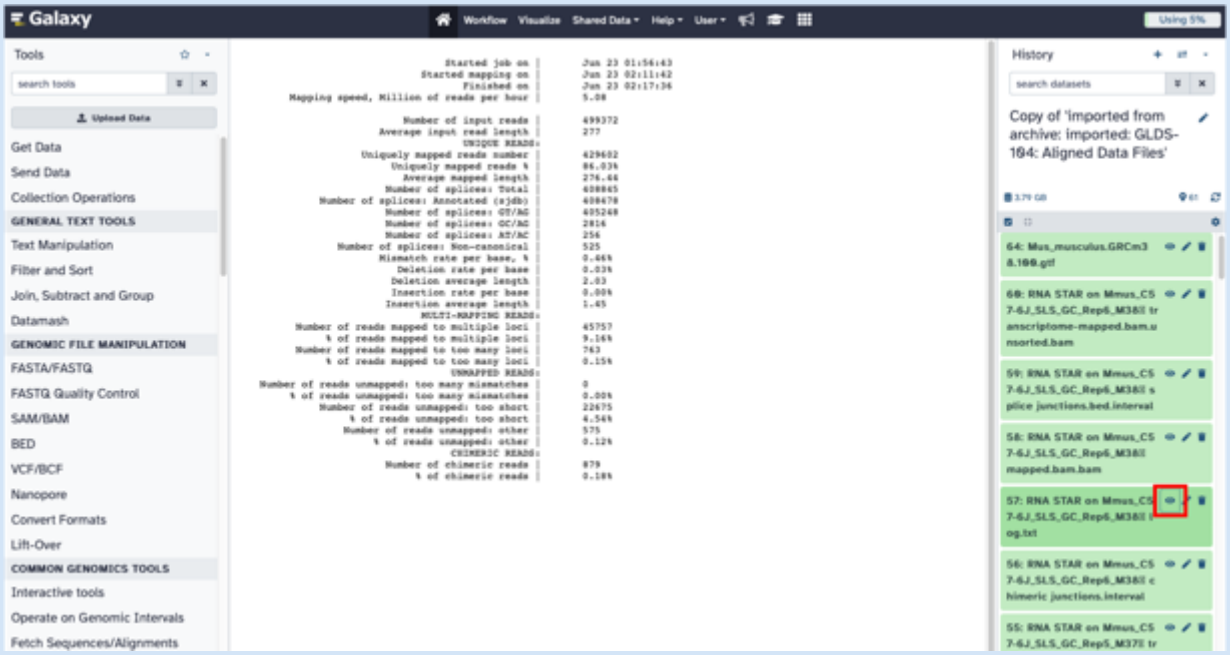
1. Navigate to **Data -> Histories -> Shared with me.** Select GLDS-104: Aligned Data Files. Click on the **file name and then 'view'. Then click on 'import this history'** on the far-left corner to add this history to your history.

Remember that there are 5 output files for each sample:

- Bam file
- Log.txt
- Transcriptome-mapped.bam
- Chimeric junctions
- Splice junctions.bed

Exercise 11 cont'd: Aligning to a reference genome

Let's review the log.txt file which can be viewed by clicking on the eye icon:



This file provides information regarding the mapping of your dataset to the reference genome. Look at the log file and answer the following questions:

1. What is the percentage of uniquely mapped reads?

_____

2. What does that mean?

_____

3. What are the remaining mapped reads associated to? Include the percentages.

_____

_____

4. How many reads were unmapped due to read length?

_____

5. What is the average mapped length per read?

_____

Exercise 11 cont'd: Aligning to a reference genome

Depending on the organism and quality of the library, the uniquely mapped reads will vary. We would usually like to see the uniquely mapped reads value be ≥ 75% - 80%. If the values drop below 60%, we should troubleshoot and/or look at the genome assembly for this organism. For badly assembled genomes we may not observe a high mapping rate, even if the actual sequence sample is good.

Review the log.txt files for all FLT and GC samples.

1. Are there any samples with lower than 80% for uniquely mapped reads? If so, which samples.

_____

2. What does that mean?

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

COUNT HOW MANY READS MAPPED TO EACH GENE

Now that we have aligned the data to the genome, it is time to count the number of reads assigned to a feature (genes, exon, chromosomal locations). To do this, we will use the tool **featureCounts**. The

alignment tool generates a BAM file which contains the chromosomal location for every read mapped. **featureCounts** takes the BAM files from the alignment step and an annotation file as input to generate a counts table and summary statistics. The counts output table includes raw counts per gene annotated via the **gtf** file provided. Summarized results include the number of successfully assigned reads and number of reads that failed to be assigned due to various reasons.

This tool contains the following parameters:

- ***Alignment file***: This where the alignment files are input either in SAM or BAM format.
- ***Specify strand information*:** Indicates if the data is stranded and whether strand-specific read counting should be performed.
- ***Gene annotation file:*** This provides an option to add annotation file from either your history, locally cached, or a built-in file.
- ***Output format:*** Provides the option to alter output format structure depending on the tool used for downstream analysis.
- ***Create gene-length file***: Creates a tabular file that contains the effective (nucleotides used for counting reads) length of the feature.
- ***Options for paired-end reads:*** Provides options specific for paired-end reads.
    - ***Count fragments instead of reads*:** Will instruct the tool to count fragments.
    - ***Only allow fragments with both reads aligned***: Will only include fragments that have both ends successfully aligned.
    - ***Exclude chimeric fragments***: Will NOT include chimeric fragments (i.e. fragments that have their two ends aligned to different chromosomes).
- ***Advanced options***:
    - ***GFF feature type filter:*** Selects the feature type.
    - ***GFF gene identifier:*** Selects the attribute type used to group features.
    - ***On feature level:*** Allows for read summarization to be performed at the feature level.
    - ***Allow reads to map to multiple features:*** This enables the user to choose whether reads map to multiple features.
    - ***Minimum mapping quality per read:*** This sets the minimum mapping quality score a read must satisfy in order to be counted.
    - ***Exon-Exon junctions***: Enables specification of whether reads spanning exon-exon junctions will be counted.
    - ***Long reads***: Enables long reads to be counted (e.g. those generated using Nanopore technologies).
    - ***Count reads by group***: Enables reads to be counted for each group separately.
    - ***Largest overlap***: Enables reads to be assigned to the target that has the largest number of overlapping bases.
    - ***Minimum bases of overlap***: Specifies the number of bases overlapping between a read and a feature.
    - ***Minimum fraction overlapping a read***: Specifies the minimum required fraction of bases included in a feature overlapping bases between a read.

  - ***Read 5' extension***: Number of bases by which a read is extended upstream from the 5' end.
  - ***Read 3' extension***: Number of bases by which a read is extended upstream from the 3' end.
  - ***Reduce read to a single position***: Read is reduced to its 5' or 3' most base.
  - ***Only count primary alignments***: Tells the tool to only count primary alignments.
  - ***Ignore reads marked as duplicate***: Reads that are marked as duplicates will be ignored.
  - ***Annotates the alignment file with 'XS:Z:'-tags to described per read or read-pair the corresponding assigned feature(s):*** Used for R code.
  - ***Ignore unspliced alignments:*** Will tell the tool to only count split alignments (e.g. exon spanning reads).
  - ***Email notification***: Sends an email notification upon completion.
- All other parameters to be left in the default setting.

featureCounts produces three files:

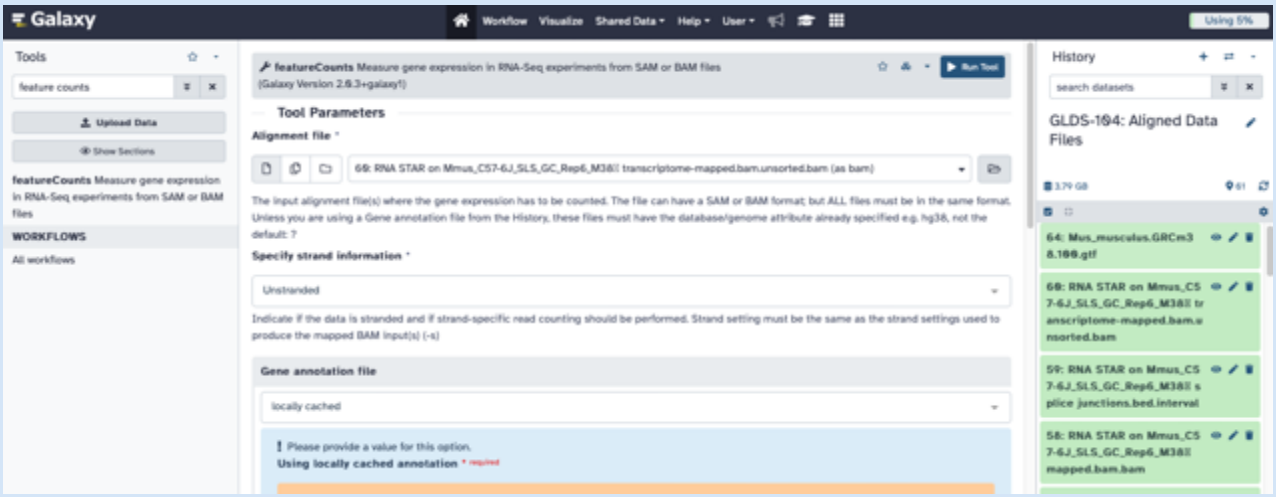
- **Counts** – table of how many reads were assigned to the gene (gene_id).
- **Summary** – statistics of how many reads were assigned and reason why some remained unassigned.
- **Feature length** - a tabular file that contains the effective (nucleotides used for counting reads) length of the feature.

Notes:

Exercise 12: Counting mapped reads

To get started, ensure that the aligned files are in your current history. If not, import them by navigating to **Data -> Histories -> Shared with me.** Select GLDS-104: Aligned Data Files.

From the tools panel, search for **featureCounts** and select it.



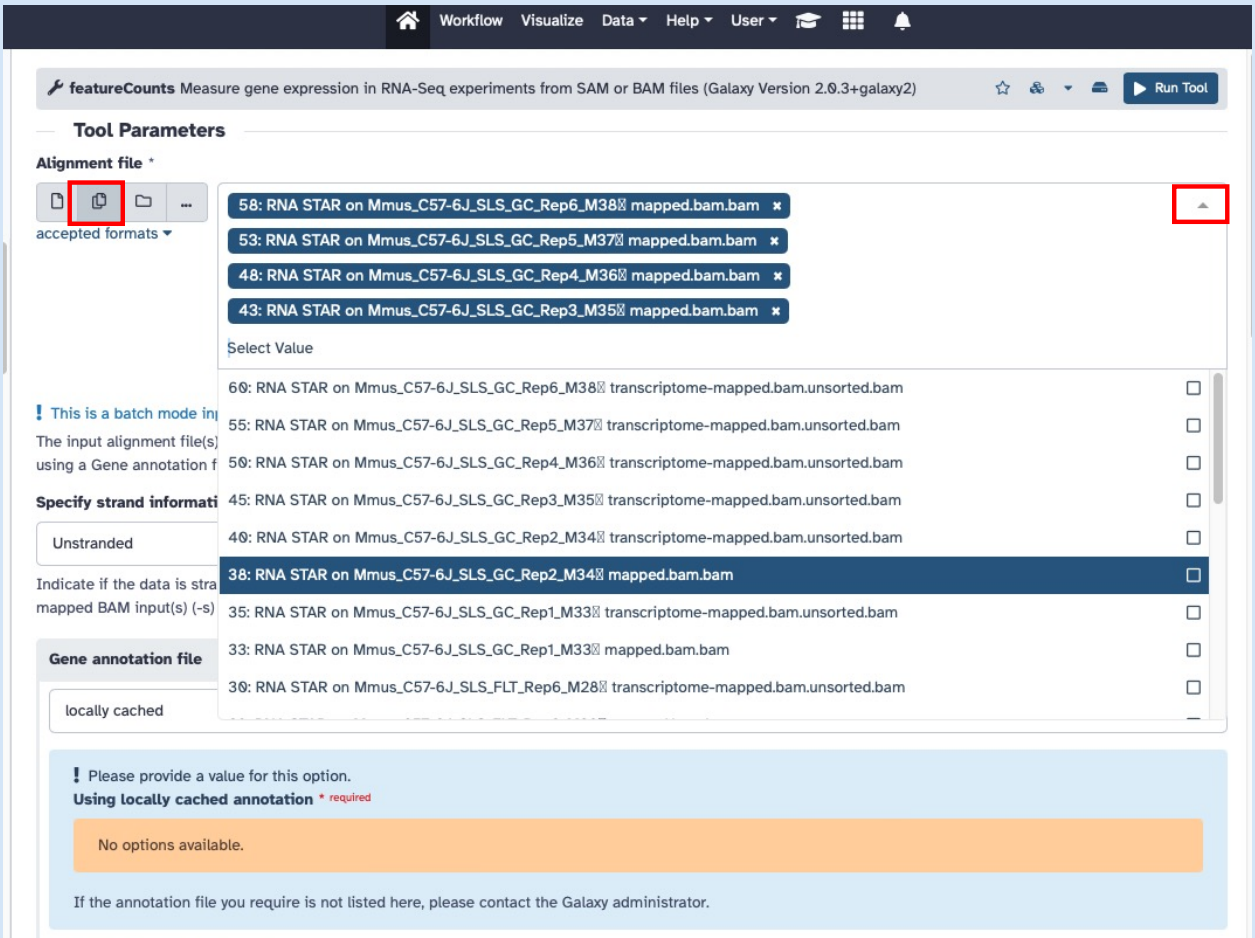
Once the tool parameters are displayed in the center panel, we can select parameters and begin counting mapped reads.

1. ***Alignment file***: select the **mapped.bam** for each sample (*not transcriptome-mapped.bam*).
2. ***Specify strand information:*** Select **Stranded (Reverse)**
3. ***Gene annotation file***: Select **a GFF/GTF file in your history** and select **Mus_musculus.GRCm38.96.gtf** file*
4. ***GFF feature type filter***: Delete exon and type in **transcript** (case sensitive – all lower case)
5. ***GFF gene identifier:*** Keep as default **gene_id.**
6. ***Output format***: make you keep the default – **Gene-ID "\t" read-count (MultiQC/DESeq2/edgeR/limma-voom compatible)**
7. ***Create gene-length file:*** Click **Yes**
8. ***Does the input have read pairs?*** In the dropdown menu change to '**Yes, paired-end but still count them as if individuals reads**'.
9. Click on **Read filtering options** to expand the options.
    a. ***Minimum mapping quality per read:*** Change to **12**
10. All other parameters to be left in the default setting.
11. Click **Run Tool.**

Annotated Genomes & Reference Gene Sets

For other model organisms e.g. Drosophila, Arabidopsis, C. elegans, import the correct annotation from *Data -> Histories -> Shared with me.* Select the file associated with the step and add to your history.

Exercise 12 cont'd: Counting mapped reads



To continue with this tutorial, we will use Published Histories to go to the next step in the analysis. Your jobs will continue running.

Let's check the results of assigning genes & counting.

Navigate to **Data -> Histories -> Shared with me.** Select GLDS-104: Count Files. Import the history as we have done previously. Now the counts files, will appear in your current history.

Remember that **featurecounts** generates three output files

- Counts
- Summary
- Feature Length

Let's review the Summary file. Click on the eye to see the contents of the file.

Exercise 12 cont'd: Counting mapped reads

Status	RNA STAR on Mmus_C57-6J_SLS_FLT_Rep3_M25: mapped.bam
Status	RNA STAR on Mmus_C57-6J_SLS_FLT_Rep3_M25: mapped.bam
Assigned	467511
Unassigned_Unmapped	41967
Unassigned_MappingQuality	167106
Unassigned_Chimera	0
Unassigned_FragmentLength	0
Unassigned_Duplicate	0
Unassigned_MultiMapping	0
Unassigned_Secondary	0
Unassigned_NonSplit	0
Unassigned_NoFeatures	404833
Unassigned_Overlapping_Length	0
Unassigned_Ambiguity	9027

Review the data displayed and answer the following questions:

1. How many reads were assigned?

_____

2. What are the reasons why some reads where not assigned?

_____

3. Look through several samples. Are there large differences in the number of reads assigned between the different samples?

_____

Notes:

_____

_____

_____

_____

_____

_____

QC: ASSIGNMENT OF READS TO GENES

Using the **featureCounts** summary output we can view and assess the number of reads assigned to genes across all our samples. We will do this using the **MultiQC** tool. **MultiQC** aggregates results from several bioinformatics tools into a single interactive report.

The parameters for this tool are shown below:

- ***Which tool was used generate logs***: Tells the tool which tool was used to generate the input data.
- ***Output of tool name:*** Select the input files
- ***Report title***: Free text to title your report
- ***Custom comment***: Enables the user to add a comment to the top of the report
- ***Use only flat-plots (non-interactive images):*** Uses less disk space and resources for display purposes
- ***Output the multiQC log file?:*** Generates a log file of the tool
- ***Email notification***: Sends an email notification when the job is complete

MultiQC provides multiple outputs such as plots and stats.

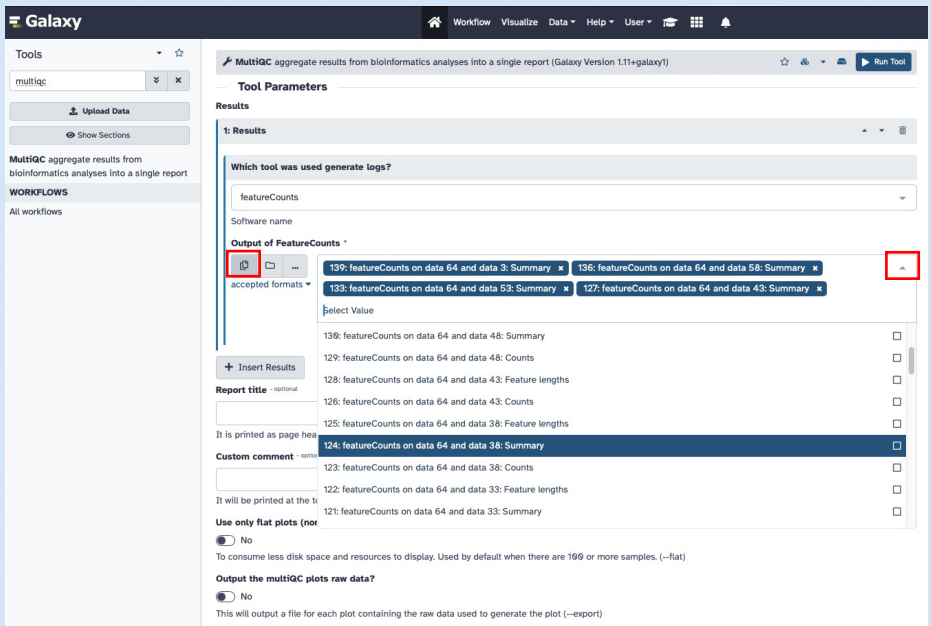
New to MultiQC? watch this tutorial video: https://www.youtube.com/watch?v=qPbIlO_KWNo

Notes:

Exercise 13: Checking the quality of read assignment.

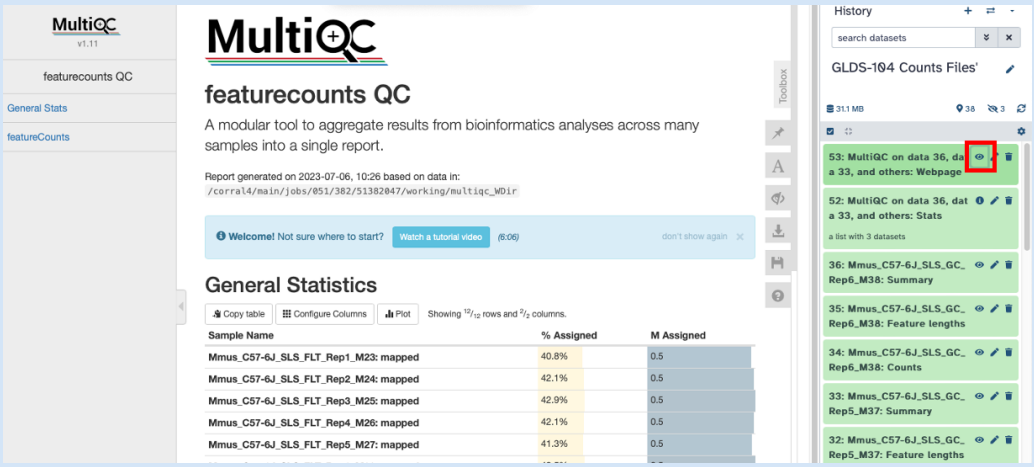
In the tools panel, search for **MultiQC** and select it. For this tool, you will use the feature counts files that were just generated or import them from **Histories -> Shared with me**.

1. Select **featureCounts** for Which tool was used generate logs?
2. Select all your **Summary** output files from **featureCounts**. *Delete any file linked that is not from **featureCounts** Summary. Click on the box to see list of files and select all that apply.*
3. Enter Report title – **featureCounts QC.**
4. All other parameters to be left in the default setting.
5. Click **Run Tool.**



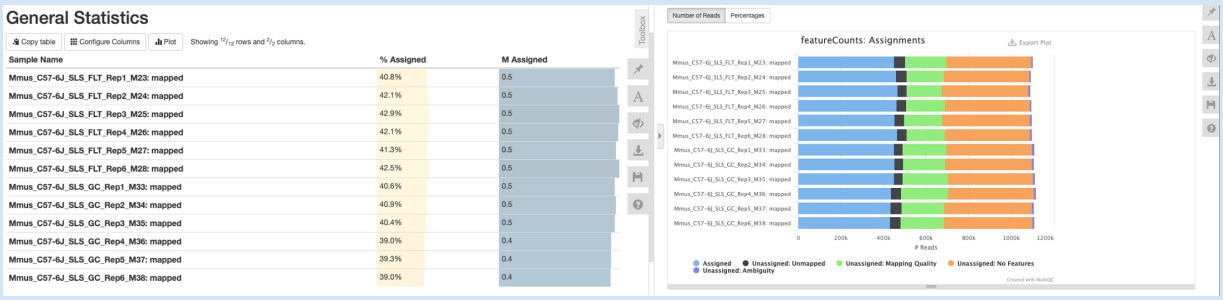
Let's review the MultiQC: Webpage.

MultiQC provides multiple outputs such as plots and stats. We will focus on reviewing the MultiQC Webpage output. Click on the eye to view it.



Exercise 13 cont'd: Checking the quality of read assignment

Review the General Statistics and **featureCounts** sections:



General Statistics

Copy table | Configure Columns | Plot | Showing 12/12 rows and 7/9 columns.

Sample Name	% Assigned	M Assigned
Mmus_C57-6J_SLS_FLT_Rep1_M23: mapped	40.8%	0.5
Mmus_C57-6J_SLS_FLT_Rep2_M24: mapped	42.1%	0.5
Mmus_C57-6J_SLS_FLT_Rep3_M25: mapped	42.9%	0.5
Mmus_C57-6J_SLS_FLT_Rep4_M26: mapped	42.1%	0.5
Mmus_C57-6J_SLS_FLT_Rep5_M27: mapped	41.3%	0.5
Mmus_C57-6J_SLS_FLT_Rep6_M28: mapped	42.5%	0.5
Mmus_C57-6J_SLS_GC_Rep1_M33: mapped	40.6%	0.5
Mmus_C57-6J_SLS_GC_Rep2_M34: mapped	40.9%	0.5
Mmus_C57-6J_SLS_GC_Rep3_M35: mapped	40.4%	0.5
Mmus_C57-6J_SLS_GC_Rep4_M36: mapped	39.0%	0.4
Mmus_C57-6J_SLS_GC_Rep5_M37: mapped	39.3%	0.4
Mmus_C57-6J_SLS_GC_Rep6_M38: mapped	39.0%	0.4

1. What % of reads are assigned (on average)?

_____

2. What does this mean?

_____

These plots can also be exported. Remember to view the tutorial video to aid your understanding of these plots. Also remember that this is a truncated dataset which means the size of the dataset was reduced for this tutorial.

Notes:

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

NORMALIZING AND PERFORMING DIFFERENTIAL EXPRESSION

During sample processing, library preparation and sequencing, external factors that are not of biological interest can affect the RNA-seq results of individual samples and introduce confounding variables. Therefore, normalization is a critical step to accurately interpret the results of genomic and transcriptomic studies. In other words, the RNA-seq raw data needs to be adjusted so that comparisons are based on biological effects rather than on technical variation/bias. Additionally, the proportion of reads attributed to a given gene in a library depends on the expression properties of the whole sample rather than just the expression level of that gene. For example, if an RNA sample is contaminated, the reads that represent the contamination will lower the percentage of reads from the true sample, thus dropping the number of reads of interest and offsetting the proportion for every gene. However, as we demonstrate, true biological differences in RNA composition between samples will be the main reason for normalization. The problem with RNA-seq analysis is that while we know the total number of reads, the total RNA output for a sample is not known and can vary drastically between samples, depending on composition of the RNA within that sample. An example of normalization is explained in the video in Exercise 14.

There are multiple methods for normalization, and several are based on standardizing data between samples by scaling the number of reads in a given lane/library to a common value across all sequenced libraries in the experiment. This makes sense as it is expected that sequencing to half the depth, will on average produce half the number of reads mapping to each gene. However, this may be over-simplified. This is because the number of tags expected to map to a gene is based on the expression level, length of the gene, and the composition of the RNA population being sampled. Therefore, if a large number of genes are highly expressed in one experimental condition the sequencing capacity for the rest of the genes in the sample is decreased. If the analysis isn't adjusted with this consideration, then differential expression analysis will be skewed towards one condition.

There are several different normalization methods that exist. We will use DESeq2 for both normalization and differential expression analysis. It takes read count files from different samples, combines them into a big table (with genes in the rows and samples in the columns) and applies normalization for sequencing depth and library composition. Gene length normalization does not need to be accounted for because we are comparing the counts between sample groups for the same gene.

This expression analysis is estimated from read counts and attempts are made to correct for variability in measurements using replicates, that are essential for accurate results. For your own analysis, we advise you to use at least 3, but preferably 5 biological replicates per condition. It is possible to have different numbers of replicates per condition.

Multiple factors with several levels can then be incorporated in the analysis describing known sources of variation (e.g. treatment, tissue type, gender, batches), with two or more levels representing the conditions for each factor. After normalization we can compare the response of the expression of any gene to the presence of different levels of a factor in a statistically reliable way.

Differential Gene Expression (DGE) analysis will provide the following:

- An estimate of the biological variance using the replicates for each condition
- An estimate of the significance of expression differences between any two conditions

Exercise 14: Normalization and differential gene expression

Watch this video StatQuest: DESeq2, part 1, Library Normalization to understand how the counts are normalized - https://www.youtube.com/watch?v=UFB993xufUU&t=35s

Answer the following questions:

1. Why is normalization important?

_____

_____

2. Why is it important to adjust for differences in library composition?

_____

_____

The expression analysis is estimated from read counts and corrections for variability in measurements are made using replicates.

3. Why is it important to have biological replicates?

_____

_____

4. For DGE, what is the minimum recommended number of replicates? How many replicates per condition are there for GLDS-104?

_____

_____

For DGE, we are looking to see what factors contribute to differences in gene expression. You can compare multiple factors to one another.

5. For this exercise, what are the factors that we are comparing?

_____

We will use **DeSeq2** for both normalization and gene expression analysis.

This tool has the following parameters:

- ***How****:* Select the format the input data is in, datasets per level or group tags corresponding to levels.
- ***Factor*:** Name of experiment factor. *Only use numbers, letters, or underscores (case sensitive). No spaces.*
- ***Factor level*:** Free text for group name. *only use numbers, letters, or underscores (case sensitive). No spaces.*
- ***Count files*:** select the input files per factor level.
- ***+ Insert Factor Level or +Insert Factor:*** Used to insert more factor levels and to add more factors.
- ***Provide a tabular file with additional batch factors to include in the model (Optional):*** Provides a file with additional factors.
- ***File have header?*** If this option is set to Yes, the tool will assume that the count files have column headers in the first row.
- ***Choice of Input data***: Choose the input format from which tool was used to generate the count files.
- ***Visualizing the analysis results*:** Yes or No option to generate visualizations.
- ***Output normalized counts table:*** Yes or No option to generate additional output files.
- ***Output rLog normalized table*:** Yes or No option to generate additional output files.
- ***Output VST normalized table:*** – Yes or No option to generate additional output files.
- ***Method for estimateSizeFactors (Optional):*** Determines the method used for estimation.
- ***Fit type*:** Method to estimate dispersion
- ***Turn off outliers replacement:*** Used for studies with more than 6 replicates per sample
- ***Turn off outliers filtering:*** Automatic filtering conducted when >2 replicates are detected for a given sample.
- ***Turn off independent filtering:*** Turns off standard DESeq2 filtering.
- ***Email notification:*** Sends an email notification when the job is complete.

DESeq2 generates 3 outputs:

- A **table with the normalized counts** for each gene (rows) in the samples (columns).
- A **summary table** with statistics and DGE results per gene.
- A **graphical summary** of the results, useful to evaluate the quality of the experiment.

The summary table contains the following values for each gene:

1. Gene identifiers.
2. Mean normalized counts, averaged over all samples from both conditions.
3. Fold change in log2 (logarithm base 2).
    - The log2 fold changes are based on the primary factor level 1 vs factor level 2, hence the input order of factor levels is important. Here, DESeq2 computes fold changes of

'treated' samples against 'untreated' from the first factor 'Treatment', i.e. the values correspond to up- or downregulation of genes in treated samples.

4. Standard error estimate for the log2 fold change estimate.
5. Wald statistic (https://en.wikipedia.org/wiki/Wald_test).
6. p-value for the statistical significance of this change.
7. p-value adjusted for multiple testing with the Benjamini-Hochberg procedure, which controls false discovery rate (FDR - https://en.wikipedia.org/wiki/False_discovery_rate).

We will focus on the following visualization plots for our data analysis:

Principal Component Analysis

Principal Component analysis is a dimensionality-reduction method that transforms a large set of data into smaller ones without reducing the amount of overall data present. The overall idea is to reduce the number of variables in the data set whilst preserving as much of the data as possible. The plot produced by DESeq2 demonstrates the first two dimensions from the PCA run on the normalized counts of the samples. It shows the samples in the 2D plane spanned by their first two principal components. Each replicate is plotted as an individual data point. This type of plot is useful for visualizing the overall effect of experimental covariates and batch effects.

Heatmaps

Heatmaps give an overview of similarities and dissimilarities between samples: the color represents the distance between the samples. For the plots produced by DESeq2, dark blue means shorter distance, i.e. closer samples given the normalized counts.

Dispersion estimates

Dispersion is the measure of the variability or spread of the data. DESeq2 uses a specific measure of dispersion related to the mean and variance of the data. The DESeq2 dispersion estimates are inversely related to the mean and directly related to variance. This means that dispersion is higher for small count means and lower for large count means. To model the dispersion based on mean counts of replicates, or expression level, the dispersion for each gene is estimated using maximum likelihood estimation. Then a curve is fit the dispersion estimates for each gene resulting in the gene-wise dispersion estimates. This plot therefore gives dispersion estimates for the gene-wise estimates (black), the fitted values (red), and the final maximum a posteriori estimates used in testing (blue). This dispersion plot is typical, with the final estimates shrunk from the gene-wise estimates towards the fitted estimates. Some gene-wise estimates are flagged as outliers and not shrunk towards the fitted value. The amount of shrinkage can be more or less, depending on the sample size, the number of coefficients, the row mean and the variability of the gene-wise estimates.

Histograms

A histogram is a way to graphically depict data using bars that represent ranges of data. Taller bars indicate that more data falls into that range than shorter bars. Therefore, the histogram displays the

spread of the data. **DESeq2** generates a histogram of p-values versus frequency that gives us information regarding the statistical significance of the data.

MA plot

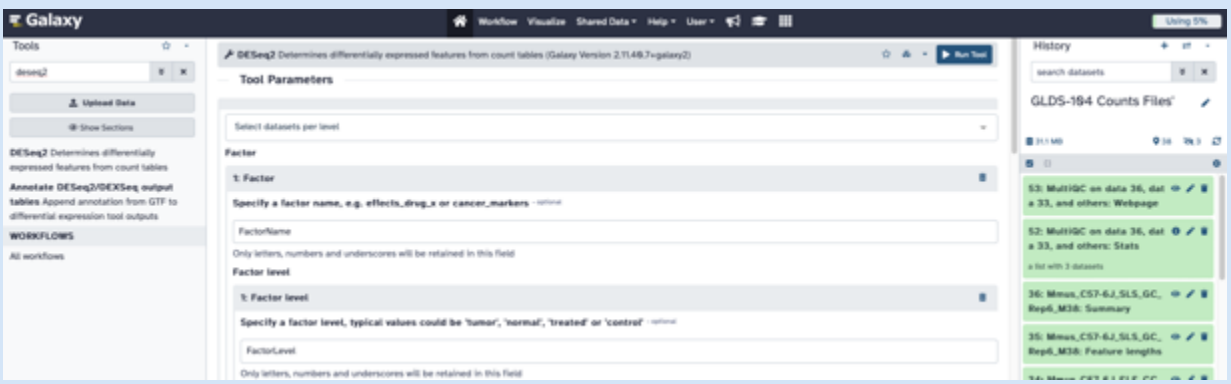
MA plots enable visual representation of genomic data and displays the global view of the relationship between the expression change of conditions (log ratios, M), the average expression strength of the genes (average mean, A), and the ability of the algorithm to detect differential gene expression. The genes that passed the significance threshold (adjusted p-value < 0.1) are colored in red.

Now that we understand DESeq2 and the results of this analysis, let's run the tool.

Exercise 15: DESeq2

Let's navigate to the history with our count data. In the History panel, click on the middle icon on the top to expand the history. Find your history with **GLDS 104: Count Files** and click on that history to move it to your current history.

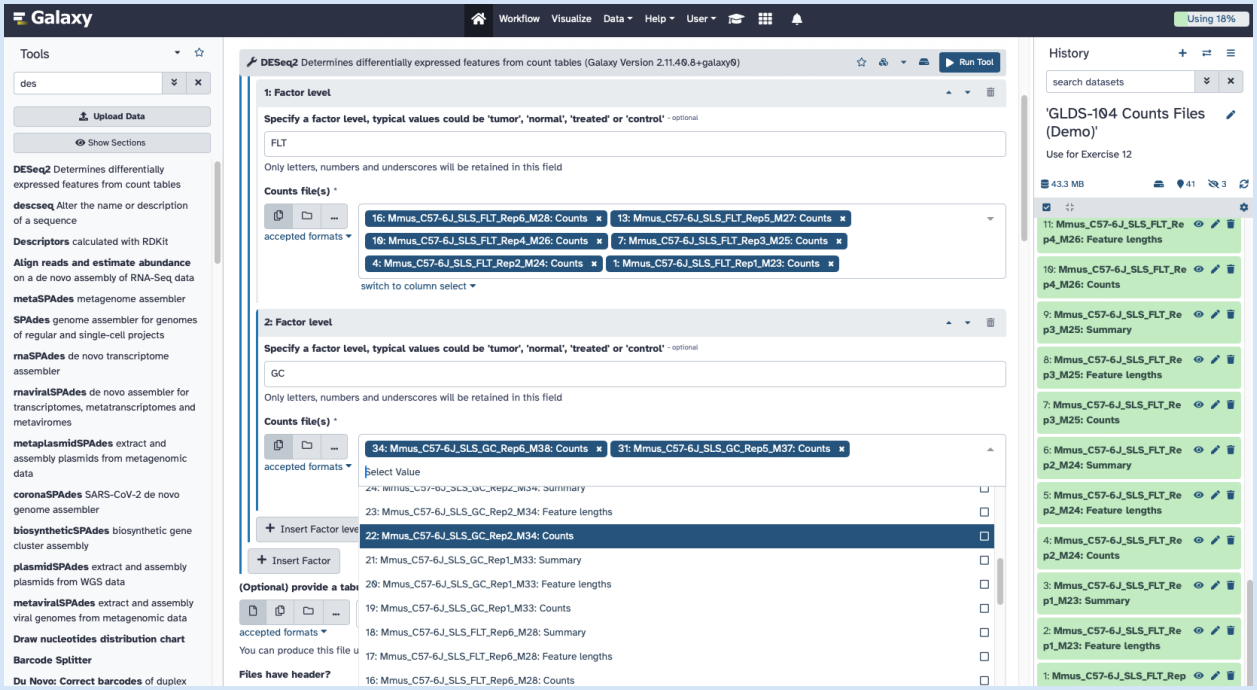
From the tools panel, search and select **DESeq2**.



Once the tool parameters have populated the center panel, we can select options and run the tool.

1. Under tool parameters: **Select datasets per level.**
2. For factor: enter **Spaceflight** under name.
3. For first Factor level, **enter FLT** (no spaces).
4. Select the **Counts** files for all your **FLT** samples (6 files).
5. For second Factor level, **enter GC** (no spaces).
6. Select the Counts files for all you **GC** samples (6 files).

Exercise 15 cont'd: DESeq2



7. (Optional) provide a tabular file with additional batch factors to include in model remains at default – **Nothing selected**.
8. Select **Yes** for Files have header.
9. Under Advanced Options:
    a. Select **ratio** for Method of estimateSizeFactors.
    b. Select **parametric** for Fit type.
10. Under Output options:
    a. Check the box next to Generate plots for visualizing the analysis results.
    b. Check the box next to Output normalized counts.
11. Do not check output sample size factors, rLog normalized table, VST normalized table, all levels vs all levels of primary factors.
12. All other parameters to be left in the default setting.
13. Click **Run Tool.**

To continue with this tutorial, we will use Histories shared with me to go to the next step in the analysis. Your jobs will continue running.

Let's check the results of assigning genes & counting.

Navigate to **Data -> Histories -> Shared with me.** Select GLDS-104: DESeq2 DGE Files and import to your current history.

Exercise 15 cont'd: DESeq2

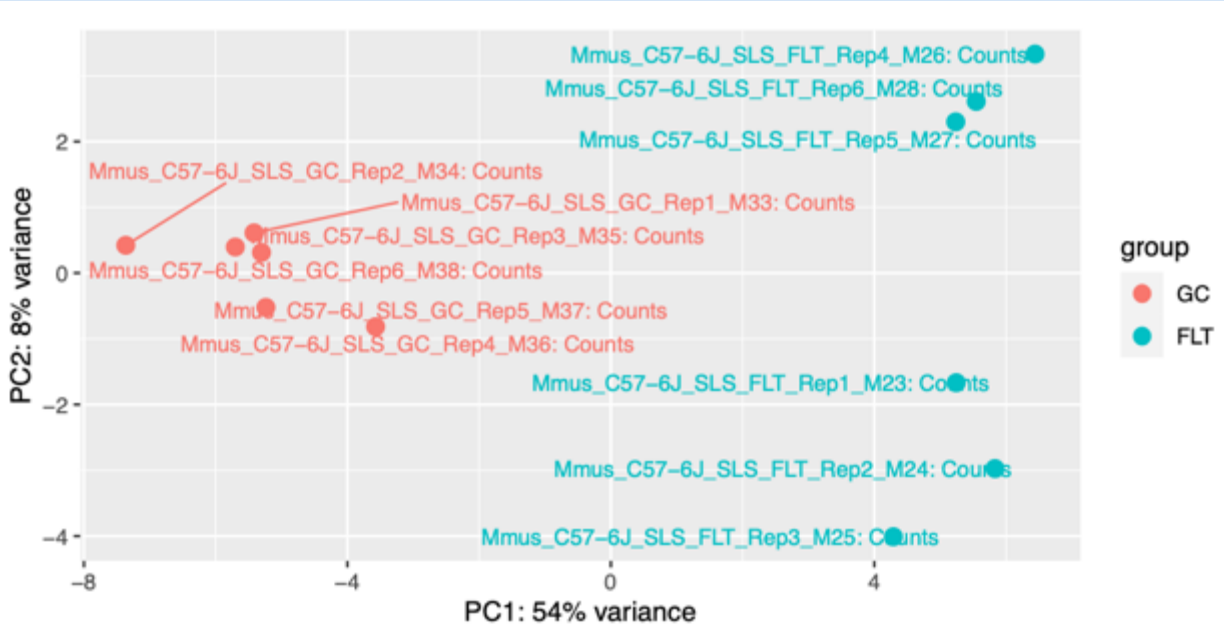
Remember that DESeq2 generates several output files including:

- A table with normalized counts.
- A summary table with statistics and DGE results per gene.

A graphical summary of the results.

Select the file labeled 'DESeq2 plots on GLDS-104'. This file will enable us to visualize the results. Remember to select the eye icon to view the results.

Let's look at the PCA plot first. This is a plot with the first two dimensions from the PCA run on the normalized counts of the samples. It shows the samples in the 2D plane spanned by their first two principal components. Each replicate is plotted as an individual data point. This type of plot is useful for visualizing the overall effect of experimental covariates and batch effects.



Answer the following questions:

1. What is the first dimension (PC1) separating?

_____

2. And the second dimension (PC2)?

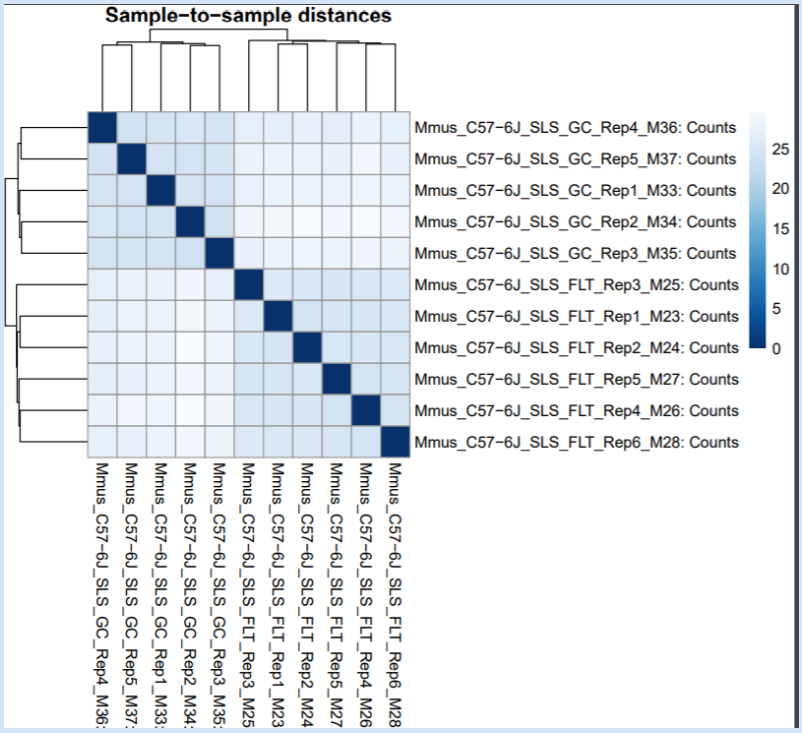
_____

3. What can we conclude about the DESeq2 design (factors, levels) we choose?

_____

Exercise 15 cont'd: DESeq2

Now let's scroll down to look at the heat map.



Answer the following questions:

1. How are the samples grouped?

_____

2. What does this tell us about the samples?

_____

_____

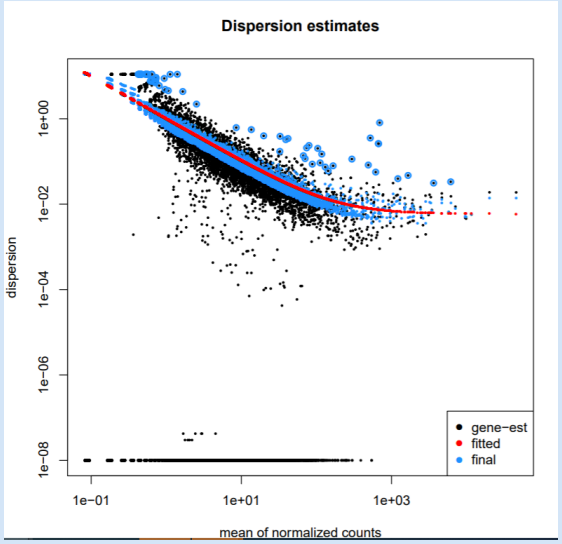
_____

_____

_____

Exercise 15 cont'd: DESeq2

Now let's look at plot of dispersion estimates.



Answer the following questions:

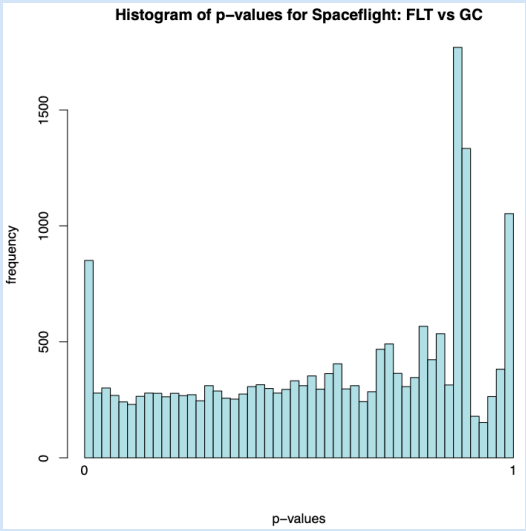
1. What do you notice about the dispersion of this data?

_____

2. What does this tell us about the samples?

_____

Now let's look at the histogram:



Exercise 15 cont'd: DESeq2

Answer the following questions:

1. What is the standard significance cut-off?

_____

2. Why is this value chosen?

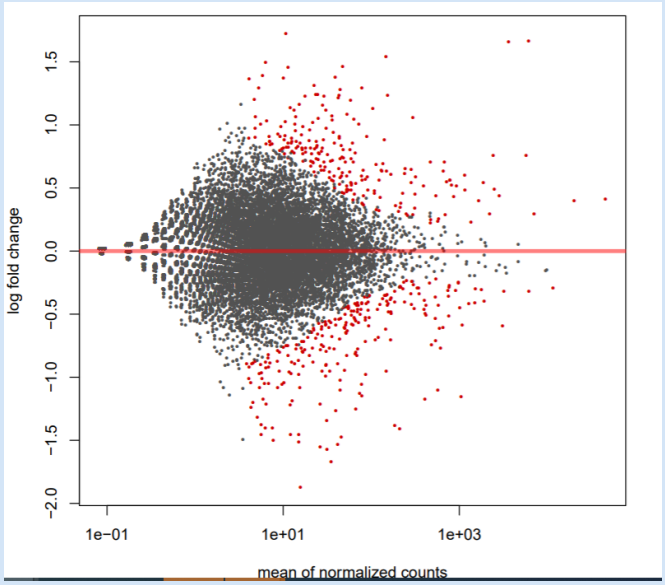
_____

3. What are some causes for decreased significance in samples?

_____

_____

Finally, let's look at the MA plot:



Answer the following questions:

2. What do each of the dots represent?

_____

3. What is the significance of the red dots?

_____

For more information about DESeq2 and its outputs, you can have a look at https://www.bioconductor.org/packages/release/bioc/manuals/DESeq2/man/DESeq2.pdf.

ANNOTATION AND VISUALIZATION OF DIFFERENTIALLY EXPRESSED GENES

Now that we have our differentially expressed genes, let's select the *most* differentially expressed genes and annotate the gene ID's to get the gene names. Once, the top genes have been selected, we will use two tools to visualize the data – **Volcano plot** and **Heatmap**.

To visualize the data, we will first need to use some additional tools to **Filter and Sort** the data. These tools are found in the tool panel and can be used to add specific parameters to the data to make it easier to analyze/understand. For example, we may only want to look at genes that demonstrate statistical and biological significance. Therefore, we would want to sort the data based on adjusted p-values and fold-change.

Remember that the summary table is not assigned headers, so we need to make a note of what is contained within each column:

1. Gene identifiers.
2. Mean normalized counts, averaged over all samples from both conditions.
3. Fold change in log2 (logarithm base 2)
   - The log2 fold changes are based on the primary factor level 1 vs factor level 2, hence the input order of factor levels is important. Here, DESeq2 computes fold changes of 'treated' samples against 'untreated' from the first factor 'Treatment', i.e. the values correspond to up- or downregulation of genes in treated samples.
4. Standard error estimate for the log2 fold change estimate.
5. Wald statistic.
6. p-value for the statistical significance of this change.
7. p-value adjusted for multiple testing with the Benjamini-Hochberg procedure, which controls false discovery rate (FDR).

We will also use **Annotate DESeq2/DEXSeq output tables** to add specific details to the data. This tool will specifically add gene symbols, chromosome positions etc. to the DESeq-2 output table and therefore enable us to add details to the table to facilitate analysis. Specifically, the following columns will be added:

8. ***Chromosome***
9. ***Start***
10. ***End***
11. ***Strand***
12. ***Feature***
13. ***Gene name***

The parameters that can be added are modifiable to enable customization.

Exercise 16: Filter and annotate differentially expressed genes.

To get started, make sure the DGE Files are in your current history. If not, navigate to User -> Histories shared with me and import GLDS-104: DESeq-2 DGE Files.

To annotate our files, we need also include the .gtf file. This should already be in your history, but if it isn't import the **Mus_musculus.GRCm38.96.gtf** to your current History as we did in **Exercise 11** .

First, we will filter the genes using statistical significance, using an adjusted p-value of p<0.05. To do this we will use the GLDS-104 DGE tables.

1. Click on the eye next to '**DESeq2 result file'** to view the DESeq2 results table. Find the column number for Adjusted p value (P-adj), and log2 fold change. Make a note of these columns (it should be column 7 and column 3) as we will use them to identify significantly expressed genes.
2. Search and select **Annotate DESeq2/DEXSeq output tables** from the tools panel under RNA-Seq.
    a. **Tabular output of DESeq2/edgeR/limma/DEXSeq** – select the file your filtered – *Genes with significant adj p-value and FC.*
    b. **Input file type** – select DESeq2/edgeR/limma.
    c. **Reference annotation in GFF/GTF format** – select the imported GTF file. *If there is no available file here, check to make sure you imported the GTF from above in your current history.*
    d. All other parameters to be left on default settings.
    e. Click **Run Tool.**

Note: You can also perform this function on filtered datasets using the 'filter and sort' function described below to see how many genes are significant.

3. In the **Tools** panel on the right-hand side of the screen, select **Filter and Sort**. Then select **Filter** *data on any column using simple expressions.*
    a. **Filter.** Select the table from DESeq2 output i.e. '**DESeq2 result file on GLDS-104'.**
    b. **With following conditions**: Enter c7<0.05.
        i. Note column 7 is where the adjusted p-value was found from above.
    c. **Number of header lines to skip**: leave as 0.
    d. Click **Run Tool**.
    e. Once your file has successfully completed, rename your output to **Genes with significant adj p-value.**
4. Now we will use Filter again to select only the genes with an absolute fold change (FC) >1.
    a. **Filter.** Select the output *Genes with significant adj p-value* you generated in the previous step.
    b. **Number of header lines to skip**: leave as 0.

Exercise 16 cont'd: Filter and annotate differentially expressed genes

c. **With following conditions**: Enter abs(c3)>1.
   i. Note log2 fold change is found in column 3 i.e. c3.
d. Click **Run Tool.**
e. Once your file has successfully completed, rename your output to **Genes with significant adj p-value and FC.**

Compare the filtered data file to the original data file. From DESeq2. **Make sure that you rename the files** to ensure you know which file is based on each filtering parameter.

Annotated Genomes & Reference Gene Sets

For other model organisms e.g. Drosophila, Arabidopsis, C. elegans, import the correct annotation from *Data -> Histories -> Shared with me.* Select the file associated with the step and add to your history.

As mentioned, the generated output is an extension of the previous DESeq2 output file that will add additional columns to your data. Note that this tool removes the header so you can use the input files to remember the column headers. The columns that were added are:

- ***Chromosome (column 8)***
- ***Start position (column 9)***
- ***End position (column 10)***
- ***Strand (column 11)***
- ***Feature (column 12)***
- ***Gene name (column 13)***

Answer the following questions:

1. What is the most over-expressed gene?

_____

2. After filtering, how many genes are significantly differentially expressed between FLT vs GC? (hint: look at the information associated with the filtered list)

_____

3. Are there any other notifiable features?

_____

_____

_____

Now we can generate the volcano plot. Volcano plots are scatter plots that show statistical significance compared to the magnitude of fold change. Therefore, this plot enables us to quickly and easily identify genes that have changed significantly in response to the treatment and may warrant further investigation. The parameters for this tool are related to visualization preferences:

- ***Specify an input file***: This file will provide the data for generation of the plot.
- ***FDR (adjusted P value)***: Specify which column this data is in.
- ***P value (raw):*** Specify which column this data is in.
- ***Log Fold Change:*** Specify which column this data is in.
- ***Labels****:* Specify which column this data is in.
- ***Significance threshold:*** Specify the threshold to be used for statistical significance.
- ***LogFC threshold to color*:** Specify the threshold for biological significance. Genes that are above this threshold will be highlighted in a different color.
- ***Points to label*:** Select to label significant points.
- ***Plot options:*** Contains a variety of labeling options.

Exercise 17: Generate the Volcano Plot

From the tools panel, search for **Volcano Plot.** Refer back to the names of the columns above to map the parameters for the Volcano plot.

1. Select **Volcano Plot**
    - ***Specify an input file***: select the output annotated file for original dataset file from DeSeq2.
    - ***FDR (adjusted P value)****:* Column 7.
    - ***P value (raw):*** Column 6.
    - ***Log Fold Change*:** Column 3.
    - ***Labels****:* Column 13.
    - ***Significance threshold*:** Enter 0.05.
    - ***LogFC threshold to color*:** Enter 1.
    - ***Points to label*:** Significant.
        i. ***Only label top-most significant*:** Enter 20.
    - All other parameters to be left on default settings.
2. Click Run Tool.

Note: You can also conduct this on the filtered DeSeq2 data.

Click on the eye icon to view the results then answer the following questions:

1. What does the volcano plot tell us?

_____

Exercise 17 cont'd: Generate the Volcano Plot

2. What is the meaning of the colors? Red vs blue?

_____

_____

3. Select one upregulated gene. Search the function of this gene by going to **http://www.genecards.org.** Why do you think this gene is upregulated in space?

_____

_____

_____

_____

_____

_____

4. Select one downregulated gene? Search the function of this gene by going to **http://www.genecards.org.** Why do you think this gene is downregulated in space?

_____

_____

Notes:

_____

_____

_____

_____

_____

_____

_____

_____

FUNCTIONAL ANALYSIS - GOSEQ

Now that we know which genes are differentially expressed, lets analyze which biological processes and pathways are affected.

One way to do this is Gene Ontology analysis which highlights over- or under-represented biological processes based on your gene list. Gene Ontology (http://geneontology.org/) maintains a controlled hierarchical vocabulary of terms and definitions to describe molecular functions, biological processes, and cellular components.

To perform this analysis, we will use **goseq** to test for over-represented gene categories. In additional to GO analysis, this tool also corrects potential length bias in differentially expressed genes.

Goseq requires two input files:

1. Differentially expressed genes file (from DESeq2)
2. A gene lengths file (from featureCounts)

The differential expressed file should only contain two columns:

1. A tabular file with the differentially expressed genes from all genes assayed in the RNA-Seq experiment with 2 columns:
    a. the Gene IDs (unique within the file), in uppercase letters
    b. a boolean indicating whether the gene is differentially expressed or not (True if differentially expressed or False if not).
2. A file with information about the length of a gene to correct for potential length bias in differentially expressed genes.

Exercise 18: Generate the files for Gene Ontology testing with goseq

To get started, make sure you have your DESeq2 results file (DESeq2 result file on GLDS-104) in your current history. Add one of the gene length files (**Mmus_C57-6J_SLS_GC_Rep6_M38: Feature lengths**) generated by **featureCounts** tool to your current history as we did before in Exercise 11. Switch back to the history with your DESeq2 results.

In the tool panel, we will be selecting several tools: **Compute** and **Cut** that are found under the Text Manipulation section. Follow the steps below:

1. Look at the DESeq2 results file 'DESeq2 result file on GLDS-104' (not the annotated file). To view the file, click on the eye icon to view the table. Find the column number for Adj. p value, we will use this column identifying our significantly expressed genes – True or False.

Exercise 18 cont'd: Generate the files for Gene Ontology testing with goseq

2. Select **Text Manipulation** from the tool panel and then select the **Compute** tool
   - ***Input file:*** Select DESeq2 Result File (not the annotated file)
   - ***Expressions → Add expression:*** bool(float(c7)<0.05) Note: *C7 is our Adj. p value column number.*
   - ***Mode of the operation***: Append.
   - ***Error Handling***:
     - ***Autodetect column types:*** No
     - ***Fail on references to non-existent columns***: No
     - ***If an expression cannot be computed for a row***: Fill in a replacement value
   - All other parameters to be left on default settings.
     a. Click **Run Tool**
3. Rename the output file to GLDS-104_DESeq2 Results_significant p. If you click the eye icon to see the file, you will notice that there is now an 8th column added (true or false) indicating significant adj p-value <0.05
4. Select **Cut** tool from **Text Manipulation** tool panel
   - ***Cut columns:*** c1,c8
   - ***Delimited by:*** Tab
   - ***From:*** the output of the **Compute** tool (GLDS-104_DESeq2 Results_significant p)
   - All other parameters to be left on default settings.
   - Click **Run Tool**
5. **Rename** the output to GLDS-104 Gene IDs and DGE. Click on the eye icon to view the file. Notice that you now only have two columns – the first column contains gene IDs and the second includes the Boolean for adj p-value <0.05
6. Next, we need to remove the uncomputed expressions. Download the results of Step 5 and rename as .txt to enable the file to be opened in excel.
7. Open the file in excel and find rows that have a '.' in the second column. Remove these rows from the spreadsheet.
8. Upload the cut file as a .tabular back into Galaxy (Tools -> Upload Data). The uploaded file will appear in your current history.

We just generated the first input for **goseq**. As second input for **goseq** we need the gene lengths. We can use here the gene lengths generated by **featureCounts** tool and format the gene IDs. Now that the inputs are in the correct format, lets run **goseq**.

After the files for **goseq** are generated, we can run the tool. Gene Ontology is a knowledgebase that contains information on gene function. This database is used as the foundation for the **goseq** tool and therefore helps to reduce the complexity of the data and highlight the biological processes that are altered in response to the treatment. Some of the processing methods used for RNA-seq can result in biases due to over-detection of transcripts that are either long or highly expressed. This tool takes this bias into account when processing the data and therefore can detect lowly expressed biological processes as well as highly expressed ones.

The parameters for this tool are as follows:

- ***Differentially expressed genes file***: Tabular file with Gene IDs in the first column, and True/False in the second column.
- ***Gene lengths file:*** Tabular file with Gene IDs in the first column, and gene length in the second column.
- ***Gene categories*:** Mapped genes categories, selected for local genomes or a file from your history.
    - ***Select a genome to use:*** Select an organism.
    - ***Select Gene ID format*:** Select the Gene ID format used in previous steps.
    - ***Select one or more categories:*** Select the databases to use for your output.
- ***Method options*:** Enables choice of analysis methods**.**
- ***Advanced options:*** Options to select for multiple hypothesis testing and to count genes that do not have a category.
- ***Output Options*:** Select Yes or No
    - ***Output Top GO terms plot?***
    - ***Produce diagnostic plots?***
    - ***Extract the DE genes for the categories (GO/KEGG terms)?***
    - ***Output RData file?***

Exercise 19: Gene Ontology testing with goseq

From the tools panel, search and select **goseq**. Once the tool options load, select the following parameters:

1. ***Differentially expressed gene files***: Select upload and browse to the local location of the file generated using excel in the previous exercise. Select upload and ensure to choose txt as the file type.
2. ***Gene lengths file***: Select Mmus_C57-6J…**Feature lengths** file from history. Remember this is the file you uploaded previously. See above if you did not do this.
3. ***Gene categories***:*
    a. ***Gene categories:*** Get categories.
    b. ***Select a genome to use:*** Mouse (mm10)
    c. ***Select Gene ID format***: Ensembl Gene ID
    d. ***Select one or more categories:*** GO: Cellular Component, GO: Biological Process, GO: Molecular Function
4. ***Output Options***:
    a. ***Output Top GO terms plot?*** Yes
    b. Extract ***the DE genes for the categories (GO/KEGG terms)?*** Yes*
5. Click **Run Tool.**

Exercise 19 cont'd: Gene Ontology testing with goseq

Annotated Genomes & Reference Gene Sets

For other model organisms such as Drosophila, and Arabidopsis import the correct annotation from Data > Histories -> Shared with me -> Annotated Genomes & Reference Gene sets > GoSeq. Select the file associated with the step and import to your history.

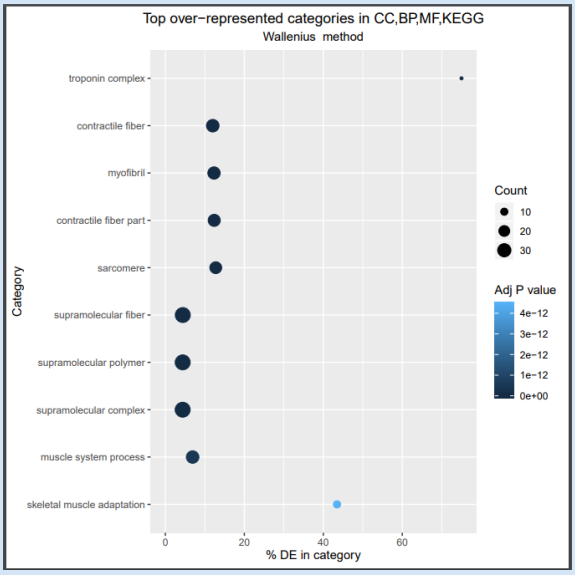
Step 3 Select Use a Category file from history and select the GO list you imported.

Step 4 Output Options:

a. Output Top GO terms plot? Yes

b. Extract the DE genes for the categories (GO/KEGG terms)? No (Select Yes, only for mouse/human)

Goseq outputs several tables and plots. Click on the eye to review the **Top over-represented GO terms plot.**



1. Select 3 processes and search in http://geneontology.org/ to learn more. Briefly describe each below.

_____

_____

_____

_____

_____

_____

Exercise 19 cont'd: Gene Ontology testing with goseq

2. Pick one of the terms. How and why do you think this GO term/biological process is affected by spaceflight?

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

Notes:

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

GENE SET ENRICHMENT ANALYSIS

Gene Set Enrichment Analysis, or GSEA, is a program that enables the detection of functional enrichment from RNA-Seq data. It basically compares two groups using the signal to noise ratio or the t-statistic, so that each probe within the sequencing dataset is assigned a score. This score is then interrogated for enrichment. The gene sets used in GSEA are defined based on prior biological knowledge e.g. published information about biological or biochemical pathways.

To perform gene set enrichment analysis, we will use the **fgsea** tool. **fgsea** requires a ranked list of genes and some gene sets to test.

fgsea outputs a table of results containing a list of pathways with P values and enrichment scores. It can also output a summary table plot of the top pathways. Click on the eye to view the outputs.

- Ranked category list.
- Plots

Exercise 20: Generate the files for FGSEA.

Before we run FGSEA, we need to format the input format using the Cut and Sort tools.

1. Switch History back to GLDS-104 DESeq2 DGE Files
2. We will use the annotated file from **Exercise 16** for this. If you have not completed this – follow detailed instructions in **Exercise 16**.

Annotated Genomes & Reference Gene Sets

For other model organisms e.g. Drosophila, Arabidopsis, C. elegans, import the correct annotation from *Data -> Histories -> Shared with me.* Select the file associated with the step and add to your history.

3. Select **Cut** under **Text Manipulations** in the tools panel
   - **Cut Columns**: c13, c5 (Gene Symbol and Wald statistics)
   - **Delimited by**: Tab
   - **From**: the output from **Annotate** step
   - All other parameters to be left on default settings.
   - Click **Run Tool.**

Exercise 20 cont'd: Generate the files for FGSEA

4. Select **Sort** tool under Filter and Sort in the tools panel.
   - **Sort Dataset:** the output from **Cut.**
   - **Number of header lines to skip:** 0.
   - **Column selections:**
     - **On column:** Column 2.
     - **In:** Descending order.
     - **Flavor:** General numeric sort.
   - All other parameters to be left on default settings.
   - Click **Run Tool.**
   - Rename your file – **GLDS-104 Gene Symbol and Wald's Statistic**
5. Select **Filter** tool in the tools panel.
   - Input: select file from **Sort** – **GLDS-104 Gene Symbol and Wald's Statistic**
   - **With the following condition**: c2<100
     - All other parameters to be left on default settings.
     - Click **Run Tool**
   - Rename your file – GLDS 104_Gene Symbol and Walds statistic-Filtered
6. Finally, we need to import the gene set from Data Libraries. Go to GL4HS Google Drive -> Annotated Genomes and Reference Gene sets > FGSEA . Select the **m_list.RData** file and import to History. Be sure tp set the file type to .rdata for mouse sets and tabular for Drosophila and Arabidopsis.

M_list.RData is Gene sets derived from the GO Biological Process Ontology (format was downloaded using https://cran.r-project.org/web/packages/msigdbr/vignettes/msigdbr-intro.html)

Now that the input files are in the correct format, we can run **fgsea** from the tools panel.

The parameters for this tool are:

- ***Ranked Genes***: Tabular file of a subset of significant genes
- ***File has header?*** Yes or no
- ***Gene Sets***: Select a gene set from the local cache or your history (this should be in rdata format)
- ***Minimum Size of Gene Set****:* Minimal size of a gene set to test. All pathways below the threshold are excluded. Default: 1
- ***Maximal size of Gene Set:*** Maximum size of the gene set to test. Default: 500
- ***Number of permutations:*** number of permutations to run
- ***Output plots***: Yes or no
- ***Plot most significant pathways:*** Specify the number of pathways to plot

fgsea outputs a table of results containing a list of pathways with P values and enrichment scores. It can also output a summary table plot of the top pathways. Click on the eye to view the outputs.

- Ranked category list.
- Plots

Exercise 21: Running fgsea

From the tools panel, search for **fgsea**. Once the tool options load, select the following parameters:

1. ***Ranked Genes***: Select the file from Sort in history i.e. GLDS 104_ Gene Symbol and Walds statistic-Filtered.
2. ***Files has header?*** Yes.
3. ***Gene Sets***: Select **m_list.RData** from history*.
4. ***Minimum Size of Gene Set***: 15.
5. ***Output plot:*** Yes.
6. ***Plot topmost significant pathways:*** 20.
7. All other parameters to be left in default settings.
8. Click **Run Tool.**

Annotated Genomes & Reference Gene Sets

For other model organisms e.g. Drosophila, Arabidopsis, C. elegans, import the correct annotation from *Data -> Histories -> Shared with me.* Select the file associated with the step and add to your history.

To continue with this tutorial, we will use Histories shared with me to go to the next step in the analysis. Your jobs will continue running. Navigate to **Data -> Histories -> Shared with me.** Select GLDS-104 FGSEA files and import them into your current history.

ANALYZING GSEA RESULTS (FROM THE GSEA TUTORIAL):

GSEA computes four key statistics for the gene set enrichment analysis report:

1. Enrichment Score (ES)
2. Normalized Enrichment Score (NES)
3. Nominal P Value
4. Adjusted P Value

Enrichment Score (ES)

The primary result of the gene set enrichment analysis is the enrichment score (ES), which reflects the degree to which a gene set is overrepresented at the top or bottom of a ranked list of genes. GSEA calculates the ES by walking down the ranked list of genes, increasing a running-sum statistic when a gene is in the gene set and decreasing it when it is not. The magnitude of the increment depends on the correlation of the gene with the phenotype. The ES is the maximum deviation from zero encountered in walking the list. A positive ES indicates gene set enrichment at the top of the ranked list; a negative ES indicates gene set enrichment at the bottom of the ranked list.
In the analysis results, the enrichment plot provides a graphical view of the enrichment score for a gene set:

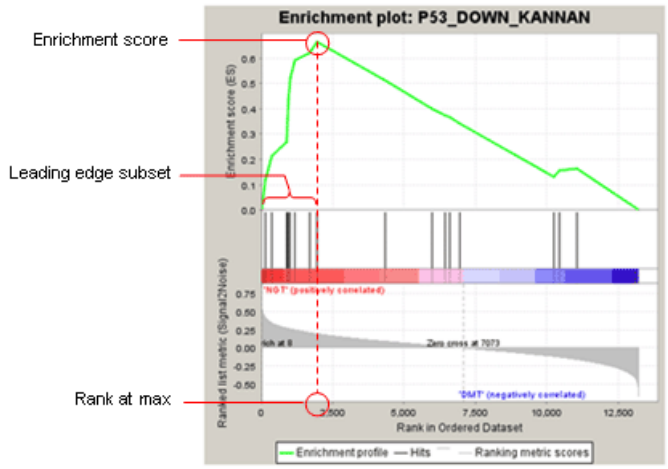


Fig 1: Enrichment plot: P53_DOWN_KANNAN
Profile of the Running ES Score & Positions of GeneSet Members on the Rank Ordered List

The top portion of the plot shows the running ES for the gene set as the analysis walks down the ranked list. The score at the peak of the plot (the score furthest from 0.0) is the ES for the gene set. Gene sets with a distinct peak at the beginning (such as the one shown here) or end of the ranked list are generally the most interesting. The middle portion of the plot shows where the members of the gene set appear in the ranked list of genes. The leading-edge subset of a gene set is the subset of members that contribute most to the ES. For a positive ES (such as the one shown here), the leading-edge subset is the set of members that appear in the ranked list prior to the peak score. For a negative ES, it is the set of members that appear subsequent to the peak score. The bottom portion of the plot shows the value of the ranking metric as you move down the list of ranked genes. The ranking metric measures a gene's correlation with a phenotype. The value of the ranking metric goes from positive to negative as you move down the ranked list. A positive value indicates correlation with the first phenotype and a negative value indicates correlation with the second phenotype. For continuous phenotypes (time series or gene of interest), a positive value indicates correlation with the phenotype profile and a negative value indicates no correlation or inverse correlation with the profile. **Note:** By default, the ranking metric is the signal-to-noise ratio.

Normalized Enrichment Score (NES)

The normalized enrichment score (NES) is the primary statistic for examining gene set enrichment results. By normalizing the enrichment score, GSEA accounts for differences in gene set size and in correlations between gene sets and the expression dataset; therefore, the normalized enrichment scores (NES) can be used to compare analysis results across gene sets. GSEA determines NES as follows:

$$\mathrm{NES} = \frac{\text{actual ES}}{\textit{mean}(\text{ESs against all permutations of the dataset})}$$

NES is based on the gene set enrichment scores for all dataset permutations; therefore, changing the permutation method, the number of permutations, or the size of the expression dataset affects the NES. As an example, consider two analyses: (1) you analyze an expression dataset, GSEA generates a ranked list and analyzes that ranked list; (2) you use a ranked list generated by the first analysis. If you use the

same parameter settings, your enrichment scores are identical; however, the normalized enrichment scores reflect the very different datasets (the expression dataset versus the ranked list of genes) used for the permutations:

	Expression Dataset		Ranked List	
Gene Set Name	**ES**	**NES**	**ES**	**NES**
BRENTANI_DNA_MET_AND_MOD	0.1233649	0.37071982	0.1233649	0.42405358
BRCA_BRCA1_NEG	0.13040805	0.6497973	0.13040808	0.6497975
PENG_RAPAMYCIN_DOWN	0.14286387	0.84542555	0.14286387	0.76681024
BASSO_REGULATORY_HUBS_SET	0.14299561	0.6870111	0.14299563	0.69177157
VENTRICLES_UP	0.14565612	0.7033464	0.14565612	0.6915998
ALANINE_AND_ASPARTATE_METABOLISM	0.14693332	0.422703	0.14693332	0.36949828
BRCA1_OVEREXP_DN	0.15077576	0.7929205	0.15077576	0.68026066

P Value

The nominal p value estimates the statistical significance of the enrichment score for a single gene set. However, when you are evaluating multiple gene sets, you must correct for gene set size and multiple hypothesis testing. Because the p value is not adjusted for either, it is of limited value when comparing gene sets.

Notes:

Exercise 22: Analyzing fgsea results

Remember that **fgsea** outputs both the ranked category list and the plots.

First, let's look at the ranked categories list by clicking on the 'eye' icon. This will bring up a table showing the GSEA results for this gene set.

1	2	3	4	5	6	7	8
pathway	pval	padj	ES	NES	nMoreExtreme	size	leadingEdge
GO_RIBOSE_PHOSPHATE_METABOLIC_PROCESS	0.00163398692810458	0.00881817787982498	0.395844491932956	2.2367160245601	0	411	Gapdh, Aldoa, Pfkfb3, Eno3, Rora, Ak1, Prkag3, Gpd1, Actn3, Pdp1, Ddit4, Prkaa2, Ampd1, Ldh
GO_REGULATION_OF_LYMPHOCYTE_ACTIVATION	0.00163934426229508	0.00881817787982498	0.318679905830753	1.79674735339643	0	408	Stat5b, Zbtb16, Zmiz1, Nck2, Dusp10, Zeb1, Dusp22, Pag1, Il7, Cdkn1a, Bcl6, Pde5a, Pik3r1, Gp
GO_SYNAPSE_ORGANIZATION	0.00163934426229508	0.00881817787982498	0.362858948487462	2.04583296098157	0	408	Six1, Lrrn1, Asic2, Lrp4, Gpc4, Frmpd4, Nrp1, Eef2k, Pafah1b1, Nfia, Spock2, Sema4d, Afg3l2, I
GO_REGULATION_OF_DNA_BINDING_TRANSCRIPTION_FACTOR_ACTIVITY	0.00164203612479475	0.00881817787982498	0.313038342739841	1.76618175466336	0	410	Smad3, Jak2, Eda2r, Ar, Camk2a, Tcf3, Ptch1, Fzd4, Clock, Med13, Psmd10, Lrp6, Trappc9, Rps6
GO_EPIDERMIS_DEVELOPMENT	0.00164473684210526	0.00881817787982498	0.399585707829464	2.25419818053906	0	405	Igfbp5, Paild, Eda2r, Lrp4, Dll1, Egfr, Wdpcp, Ptch1, Trp63, Pafah1b1, Acvr1b, Fgfr2, Sharpin, Za
GO_POSITIVE_REGULATION_OF_ESTABLISHMENT_OF_PROTEIN_LOCALIZATION	0.00164744645799012	0.00881817787982498	0.330669448489171	1.87893389993706	0	449	Smad3, Gapdh, Jak2, Ablim3, Glul, Sec24a, Egfr, Ttn, Trp63, Tpr, Tardbp, Sar1b, Oaz2, Gpl1, Cle
GO_REGULATION_OF_GTPASE_ACTIVITY	0.00164744645799012	0.00881817787982498	0.380377240741217	2.16619269200827	0	456	Rabep2, Aldh1a1, Plcb1, Prex2, Rasa2, Ralgapa1, Arhgap26, Nrp1, Dock9, Dennd1b, Pafah1b1,
GO_NEGATIVE_REGULATION_OF_IMMUNE_SYSTEM_PROCESS	0.00165562913907285	0.00881817787982498	0.30837196561381	1.75024894094151	0	436	Stat5b, Zbtb16, Dll1, Lrrc17, Tsc22d3, Plcb1, Dusp10, Adora1, Dusp22, Mafb, Dpf2, Pag1, Kif13
GO_REGULATION_OF_HEMOPOIESIS	0.00165562913907285	0.00881817787982498	0.337662172589196	1.916493474995	0	436	Stat5b, Crebbp, Zbtb16, Dll1, Lrrc17, Zmiz1, Prkca, Dusp10, Kmt2c, Tcf3, Stat1, Zeb1, Acvr1b, F
GO_REGULATION_OF_LIPID_METABOLIC_PROCESS	0.00165562913907285	0.00881817787982498	0.336017483641148	1.88956694812393	0	398	Stat5b, Rora, Pdp1, Crebbp, Prkaa2, Thra, Ncor2, Idh1, Abcd2, Srebf1, Enho, Ncoa1, Adora1, W
GO_COVALENT_CHROMATIN_MODIFICATION	0.00165837479270315	0.00881817787982498	0.355744389778228	2.02471238590893	0	463	Jak2, Mecom, Crebbp, Eya4, Dpf3, Prkaa2, Prkca, Kmt2c, Srebf1, Meaf6, Prdm9, Prdm9, Auts2,
GO_PURINE_CONTAINING_COMPOUND_METABOLIC_PROCESS	0.00165837479270315	0.00881817787982498	0.409642344821873	2.32552889382554	0	447	Gapdh, Aldoa, Pfkfb3, Eno3, Rora, Ak1, Prkag3, Gpd1, Actn3, Pdp1, Ddit4, Prkaa2, Ampd1, Ldh
GO_REGULATION_OF_ION_TRANSMEMBRANE_TRANSPORT	0.00165837479270315	0.00881817787982498	0.353500425349469	2.01194090530939	0	463	Cacna1s, Atp2a1, Kcnq5, Cicn1, Casq1, Asic2, Ahnak, Tmem109, Dhrs7c, Neto2, Kcnd2, Tpcn2,
GO_LEUKOCYTE_DIFFERENTIATION	0.00166112956810631	0.00881817787982498	0.342670725132358	1.96072423692366	0	498	Stat5b, Rora, Zbtb16, Dll1, Lrrc17, Zmiz1, Prkca, Dusp10, Pou2f2, Tcf3, Pafah1b1, Zeb1, Wnt4, I
GO_POSITIVE_REGULATION_OF_CYTOKINE_PRODUCTION	0.00166389351081531	0.00881817787982498	0.3122464987243	1.77222389920663	0	446	Stat5b, Smad3, Gapdh, Jak2, Rora, Crebbp, Plcb1, Pde4b, Stat1, Clec9a, Agt, Il7, Casp1, Nod1,
GO_REGULATION_OF_PROTEIN_SERINE_THREONINE_KINASE_ACTIVITY	0.00166389351081531	0.00881817787982498	0.331228660505318	1.89139066372459	0	496	Tnik, Jak2, Prkag3, Sbk2, Raf1, Egfr, Ksr1, Dusp10, Map2k3, Fzd4, Cdkn1c, Ccng1, Mnat1, Map3
GO_SMALL_MOLECULE_CATABOLIC_PROCESS	0.00166389351081531	0.00881817787982498	0.376975171614125	2.13230739596103	0	430	Gapdh, Aldoa, Lpin1, Eno3, Actn3, Aldh1a1, Glul, Gpd2, Gpt2, Slc25a12, Tpi1, Abcd2, Pfkm, Gp
GO_GLAND_DEVELOPMENT	0.00166666666666667	0.00881817787982498	0.410243676354361	2.32284268491236	0	434	Stat5b, Smad3, Igfbp5, Jak2, Six1, Stat5a, Raf1, Thra, Egfr, Ar, Ncor2, Rps6ka1, Thrb, Ptch1, Trp

1. Look through the GSEA results and see if you recognize any of the names. Pick one that looks interesting to you and investigate it further by copying the name of the pathway and investigating it further here: http://www.gsea-msigdb.org/gsea/msigdb/genesets.jsp Read about the GeneSet and what processes it is mapped to. on the name and reading about what the gene set is mapped to.

GeneSet name: _____

Details about the GeneSet: _____

_____

_____

_____

_____

_____

Research the dataset more and determine what biological processes are actually being investigated in the GeneSet: _____

_____

_____

_____

_____

_____

Exercise 22 cont'd: Analyzing fgsea results

How might this biological process affect muscle tissue during spaceflight? _____

_____

_____

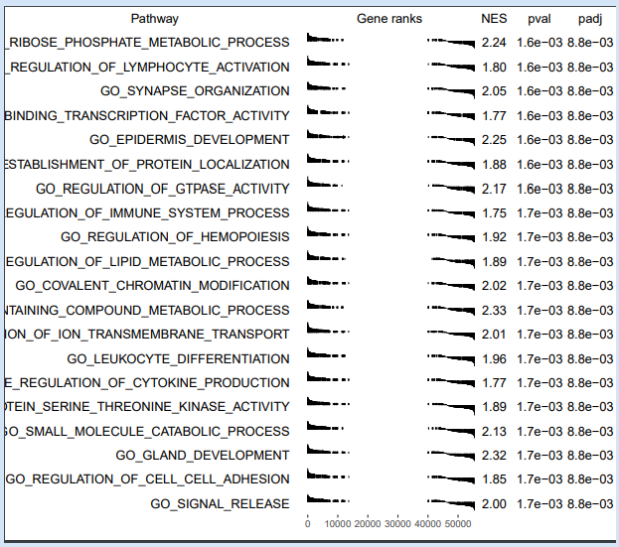
_____

_____

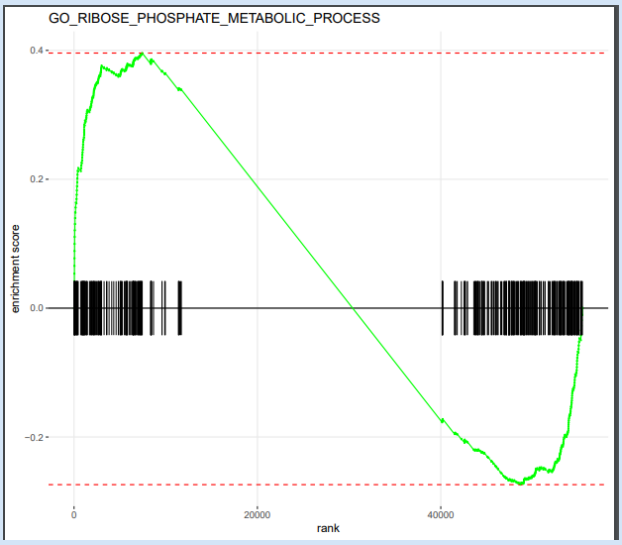
_____

_____

Now let's look at the enrichment plots. Click on the 'eye' icon next to **fgsea** on GLDS-104: Plots



Pathway	NES	pval	padj
_RIBOSE_PHOSPHATE_METABOLIC_PROCESS	2.24	1.6e−03	8.8e−03
_REGULATION_OF_LYMPHOCYTE_ACTIVATION	1.80	1.6e−03	8.8e−03
GO_SYNAPSE_ORGANIZATION	2.05	1.6e−03	8.8e−03
BINDING_TRANSCRIPTION_FACTOR_ACTIVITY	1.77	1.6e−03	8.8e−03
GO_EPIDERMIS_DEVELOPMENT	2.25	1.6e−03	8.8e−03
ESTABLISHMENT_OF_PROTEIN_LOCALIZATION	1.88	1.6e−03	8.8e−03
GO_REGULATION_OF_GTPASE_ACTIVITY	2.17	1.6e−03	8.8e−03
EGULATION_OF_IMMUNE_SYSTEM_PROCESS	1.75	1.7e−03	8.8e−03
GO_REGULATION_OF_HEMOPOIESIS	1.92	1.7e−03	8.8e−03
EGULATION_OF_LIPID_METABOLIC_PROCESS	1.89	1.7e−03	8.8e−03
GO_COVALENT_CHROMATIN_MODIFICATION	2.02	1.7e−03	8.8e−03
NTAINING_COMPOUND_METABOLIC_PROCESS	2.33	1.7e−03	8.8e−03
ION_OF_ION_TRANSMEMBRANE_TRANSPORT	2.01	1.7e−03	8.8e−03
GO_LEUKOCYTE_DIFFERENTIATION	1.96	1.7e−03	8.8e−03
E_REGULATION_OF_CYTOKINE_PRODUCTION	1.77	1.7e−03	8.8e−03
OTEIN_SERINE_THREONINE_KINASE_ACTIVITY	1.89	1.7e−03	8.8e−03
GO_SMALL_MOLECULE_CATABOLIC_PROCESS	2.13	1.7e−03	8.8e−03
GO_GLAND_DEVELOPMENT	2.32	1.7e−03	8.8e−03
GO_REGULATION_OF_CELL_CELL_ADHESION	1.85	1.7e−03	8.8e−03
GO_SIGNAL_RELEASE	2.00	1.7e−03	8.8e−03



As mentioned above, the gene sets with a distinct peak at the beginning (such as the one shown here) or end of the ranked list are generally the most interesting. The middle portion of the plot shows where the members of the gene set appear in the ranked list of genes. The genes in the ranked list ago from positive to negative as you move down the list. A positive value indicates correlation with the first phenotype i.e. spaceflight and a negative value indicates correlation with the second phenotype i.e. ground control.

Investigate the other pathways that are altered in the Enrichment Analysis – see if you can start to build a bigger picture about what changes are happening because of exposure to spaceflight. Think about a pathway or biological process that you are interested in and start to investigate it more using the tools you have learnt so far.

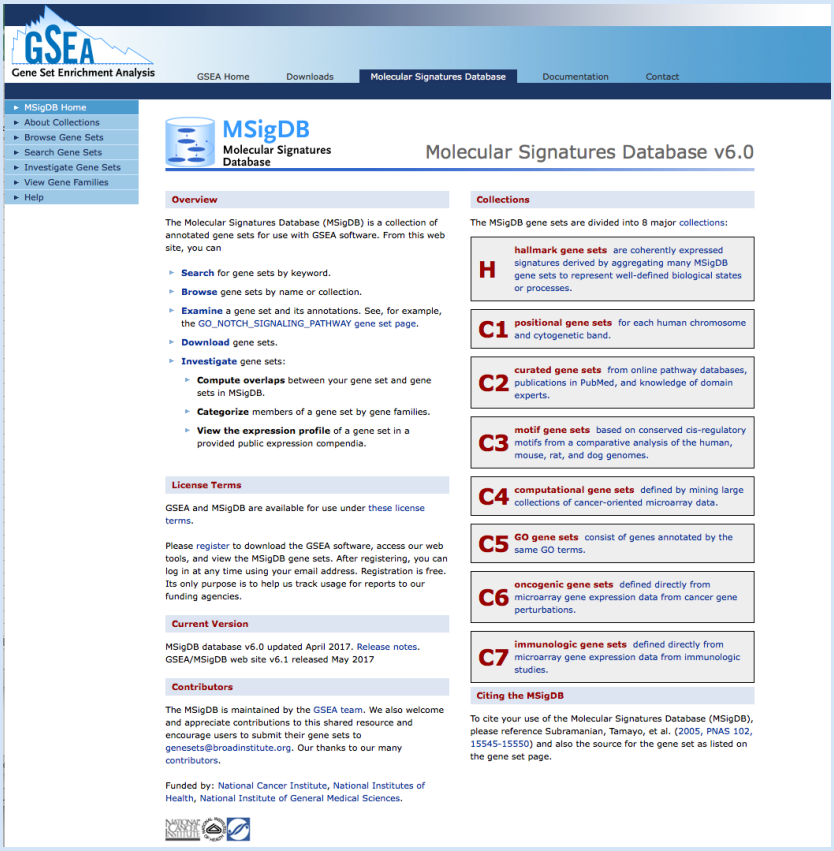
Notes:

MOLECULAR SIGNATURE DATABASES:

Another way that data can be analyzed is by investigating how a specific subset of genes are collectively related to a variety of different pathways, molecular mechanisms, cellular locations and others by using a powerful online database called the Molecular Signature Database or **MSigDB**. This database is a collection of gene sets that are also used in the GeneSet Enrichment Analysis we just conducted. The database has a number of different gene sets e.g. hallmark gene sets which are based around biological processes or pathways, positional gene sets, based on their position on a chromosome, and curated gene sets which focus on online pathway databases such as KEGG.

Exercise 23: Using MSigDB

1. Go to http://software.broadinstitute.org/gsea/msigdb.
2. Select 'investigate gene sets'.
3. Register as a new user.



4. Now you will have to use your gene identifiers as input. To do this:
   - Download by clicking the floppy disk file generated in the **DESeq2** step and filtered - **Genes with significant adj p-value.**
5. *(Use the View all Histories button on the top right and find the history for DGE files, then click Switch to button. Click Analyze Data from the top menu and download **DESeq2 DGE table** file.)*

Exercise 23 cont'd: Using MSigDB

6. Import the table file into Excel. Copy those corresponding gene names (furthest column on the right) and paste into **MSigDB.**
7. Select Species: Mouse.
8. Select the gene sets you want to look into e.g. C5 GO > biological function, Hallmark gene sets.
9. Select compute overlap.

Now we can see which biological processes our data maps to, including how many genes in our gene set map to the curated gene set and the significance of this mapping. To investigate what each parameter does, click on the process, and look at the brief description. You can also look at the genes contained within this GeneSet and download them.

Answer the following questions:

1. What processes do you see changed in your GeneSet in response to spaceflight?

_____

_____

_____

_____

2. Investigate your data with other gene sets e.g. KEGG, Reactome. What does this tell you about your data?

_____

_____

_____

_____

3. Research what these databases are and where their data comes from.

_____

_____

_____

_____

You can also follow the above steps to look at bigger sets of data e.g. all genes with $p<0.05$. This can be full gene list before we filtered for the "True" ($p>0.05$ from DGE table) from the earlier step.

Other databases (various organisms):

Intermine: Integrates many biological databases and allows you to run analysis or query keywords/ontologies for each type organism group. http://intermine.org/

- Flymine: http://www.flymine.org/flymine/begin.do
- Yeastmine https://yeastmine.yeastgenome.org/yeastmine/begin.do
- Wormmine: http://intermine.wormbase.org/tools/wormmine/begin.do;jsessionid=98761DC4BD69B4AAC4EC57265ED5ED15
- Mousemine: http://www.mousemine.org/mousemine/begin.do

TAIR - GO Enrichment for plants: https://www.arabidopsis.org/tools/go_term_enrichment.jsp

TOAST: Test of Arabidopsis Space Transcriptome - https://astrobiology.botany.wisc.edu/astrobotany-toast

WEB-based GEne SeT AnaLysis Toolkit: http://www.webgestalt.org/option.php#

Reactome (human related organisms): https://reactome.org/

DAVID Bioinformatics database: https://david.ncifcrf.gov/

Cytoscape: https://cytoscape.org/

Use for generating GO clusters

String: https://string-db.org/

Use for clustering results of protein-protein interactions.

TopFunn: https://toppgene.cchmc.org/enrichment.jsp

Use for defining enriched gene ontologies and KEGG pathways for organisms with human orthologs, such as zebrafish studies.

LESSON 4: WHAT'S NEXT? ANALYSIS TECHNIQUES FOR INTERPRETING DATA

So far in this manual you have learnt about large scale datasets i.e. omics analysis, with a specific focus on RNA-Seq. You have also learnt about GeneLab and the importance of GeneLab for curating data that was obtained from experiments conducted in space or using models of spaceflight and you have worked on analyzing a sample dataset using parameters and methods that we have provided.

Now that you have learnt all of these things and have an understanding of the tools that can be used to analyze RNA-Seq data, it is time to analyze some of the data on your own. We have shown you a straightforward method, using Galaxy that will give you data that is readily understandable and interpretable. You can also run these modules in different ways with different 'slices' of data and that will allow you to dig deeper into the biological meaning behind the data.

Galaxy also contains a vast number of modules that can be used in various combinations to delve deeper into the large amount of data that RNA sequencing and other types of omics analyses produces. The documentation that accompanies each module provided by Galaxy will guide you through most of the parameters and give you a starting point for these modules.

Some examples of other modules that you can use include:

- **Multiqc** - aggregates results from bioinformatics analyses across many samples into a single report. It takes results of multiple analyses and creates a report that can be viewed as a single beautiful web-page.
- **Cutadapt** - finds and removes adapter sequences, primers, poly-A tails and other types of unwanted sequence from your high-throughput sequencing reads.
- **Bowtie2** – another alignment tool that can be used. It is particularly good at aligning reads of about 50 up to 100s or 1,000s of characters to relatively long (e.g. mammalian) genomes. Bowtie 2 supports gapped, local, and paired-end alignment modes.
- **Htseq-count** - can be used to count aligned reads in a BAM file that overlap features in a GFF file.
- **edgeR -** quasi-likelihood pipeline (edgeR-quasi) for differential expression analysis. This statistical methodology uses negative binomial generalized linear models, but with F-tests instead of likelihood ratio tests. This method provides stricter error rate control than other negative binomial based pipelines, including the traditional edgeR pipelines or DESeq2.

A couple of final points to consider:

1. Always look at the Metadata and study the experimental design. All data interpretation should be done thinking about the original experimental conditions including organism, tissue type, duration of treatment, sample collection, sample preparation and processing methods.
2. Always rename your output files with the sample name.
3. Always check the quality of data before starting analysis and trim low-quality reads.
4. Always normalize your data – data that is not normalized will give misleading results.

5. Always track your analysis and the parameters you use for these analyses. Also, if you want to publish your findings you will need to describe the details of your analysis in depth. Good record keeping is a key part of science.
6. Remember that your hypothesis may not always be right – it is part of scientific thinking to be able to rework your hypothesis to test the data, do NOT rework the data to fit your hypothesis!
7. Most of all – have fun and explore the mysteries that organisms have hidden in their genomes!

Happy Analyzing from the

GeneLab for High Schools Team!

REFERENCES

Galaxy Training Network: https://training.galaxyproject.org/training-material/topics/transcriptomics/

RNA-Seqlopedia: https://rnaseq.uoregon.edu

1. Chaudhary, J., et al., *Advances in Omics Approaches for Abiotic Stress Tolerance in Tomato.* Biology (Basel), 2019. **8**(4).

2. Overbey, E.G., et al., *NASA GeneLab RNA-seq consensus pipeline: standardized processing of short-read RNA-seq data.* iScience, 2021. **24**(4): p. 102361.