

# Practical - Read classification of our sequenced data

One of the things to help us understand what's in our data is to classify the reads using Kraken2. We can use Kraken2 to classify reads against a database of known sequences. This is a quick way to get an idea of what is in our data. We can then visualise the results in another tools like Krona or Pavian.

Kraken 2 is a bioinformatics tool and software platform designed for the taxonomic classification of DNA sequences in metagenomic data. Metagenomics involves the study of genetic material collected from environmental samples, such as soil, water, or clinical specimens, to understand the microbial diversity present in these samples. Kraken 2 is a popular tool in this field, as it allows researchers to assign taxonomic labels to the sequences, helping them identify the microorganisms present in the samples.

For this exercise, we have three of the same samples that were processed in three different labs, for a total of nine samples. Usually, you know which organisms have been sent to you, but in this case I will let you figure that out from the data provided.



## Thanks

Many thanks to Andrea Telatin and Thanh Le Viet, who provided these sequence data.

## Your tasks are:

- Download/Upload the sequenced read data
- Process them with Kraken2
- View the Kraken2 report
- Visualise the result in Pavian and/or Krona



## Tip

Remember that there are three original isolates (Sample-1, Sample-3, Sample-8), that have been processed by three different groups (Lab-1, Lab-2, Lab-3); This means that we expect "Lab-1-Sample-1", "Lab-2-Sample-1", "Lab-3-Sample-1" to be the same.

Then use this information to answer the following questions:

- Which species were each sample supposed to be?
- Are there indications of contamination?
- If there is contamination, what are the top three (in terms of abundance) other species identified?
- For each sample, how many reads were unclassified?
- Consider the typical genome size for each species, and calculate whether the samples have enough coverage for genome assembly.
- What are some possible sources of contamination (if any)? You can simply speculate.

The rest of this page gives information on how to answer these questions. The [answers to these questions is here](#).

**Tip**

We previously discussed the requirements regarding yield in "[A framework for QC](#)". In this case, we would like at least 20X coverage.

## Help! I'm stuck

If you are having problems getting Kraken2 to run, [here are the report output files](#). These files have enough information to answer the questions above. You use these files in Pavian as well.

There are some [krona plots available here](#).

## Running Kraken2 on these samples

I will use <https://usegalaxy.eu/> as an easy way to run Kraken2, and it will allow you to follow along. You may be able to do this on the command-line later using [some instructions here](#)

**Note**

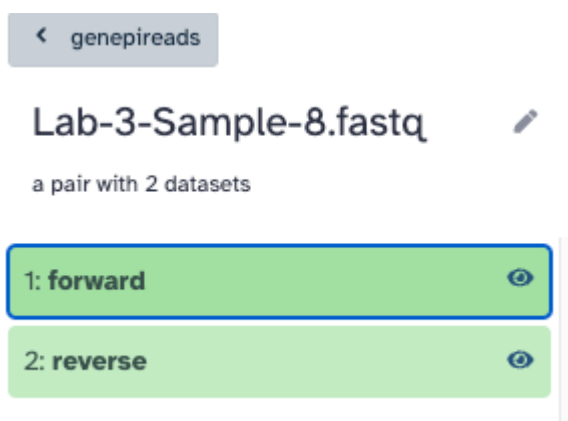
It's more important to understand the output, so if you are short on time; please skip to the following exercises exploring the results.

## Log in to Galaxy and upload the data

Using the data linked above, upload the sequenced reads to Galaxy - be sure to create these in a List of Pairs collection.




The collection should look like this, a list of nine pairs, and each pair has a forward and reverse.



## Running Kraken2

You should be able to find Kraken2 with the search bar on the left. The input should be Paired Collection and should be the collection of data you uploaded.

 **Kraken2** assign taxonomic labels to sequencing reads (Galaxy Version 2.1.1+galaxy1)

Tool Parameters

Single or paired reads

Paired Collection

--paired

Collection of paired reads \*

accepted formats ▾

55: genepireads

Print scientific names instead of just taxids

☐ No

(--use-names)

Confidence \*

0.1

Confidence score threshold. Must be in [0, 1] (--confidence)

Minimum Base Quality \*

0

Minimum base quality used in classification (only effective with FASTQ input) (--minimum-base-quality)

Minimum hit groups \*

2

Number of overlapping k-mers sharing the same minimizer needed to make a call (--minimum-hit-groups)

Enable quick operation

☐ No


Quick operation (use first hit) (--quick)

Split classified and unclassified outputs?

☐ No

Sets --unclassified-out and --classified-out

The database I selected was "Preprint refseq indexes PlusPF". If you use a different database, you will get slightly different results.

 **Warning**

Remember to "Print a report" under the Create a report dropdown

Create Report

Print a report with aggregate counts/clade to file

☒ Yes

--report

Format report output like Kraken 1's kraken-mpa-report

☐ No

(--use-mpa-style)

Report counts for ALL taxa, even if counts are zero

☐ No

(--report-zero-counts)

Report minimizer data

☐ No

Report minimizer and distinct minimizer count information in addition to normal Kraken report (--report-minimizer-data)

Select a Kraken2 database \*

<

Prebuilt Refseq indexes: PlusPF (Standard plus protozoa and fungi) (Version: 2022-06-07 - Downloaded: 2022-09-04T165121Z)

Additional Options

Email notification

☐ No

Send an email notification when the job completes.

Run Tool

Help

If this is all in order, click Run tool. It may take some time to run, so [here are the report output files](#) I prepared earlier that you can use for the next step.

## Exploring the results in the Kraken2 report

Open the Kraken reports you created in Galaxy, or [use the prepared reports here](#). The report files are text files and should open in any text editor. It will look something like this,

| Column 1 | Column 2 | Column 3 | Column 4 | Column 5 | Column 6            |
|----------|----------|----------|----------|----------|---------------------|
| 76.86    | 105399   | 105399   | U        | 0        | unclassified        |
| 23.14    | 31740    | 1197     | R        | 1        | root                |
| 22.20    | 30448    | 312      | R1       | 131567   | cellular organisms  |
| 12.58    | 17254    | 3767     | D        | 2        | Bacteria            |
| 8.77     | 12027    | 2867     | P        | 1224     | Proteobacteria      |
| 4.94     | 6779     | 3494     | C        | 28211    | Alphaproteobacteria |
| 1.30     | 1782     | 1085     | O        | 204455   | Rhodobacterales     |
| 0.43     | 593      | 461      | F        | 31989    | Rhodobacteraceae    |
| 0.05     | 74       | 53       | G        | 265      | Paracoccus          |

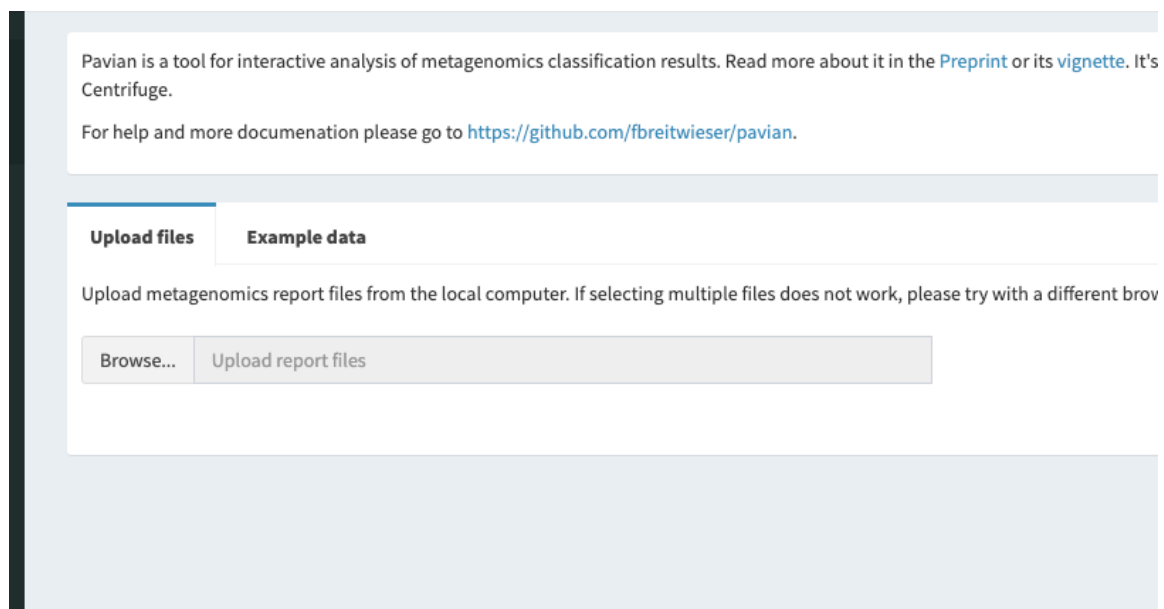
There are six columns in each report file:

- Percentage of fragments covered by the clade rooted at this taxon
- Number of fragments covered by the clade rooted at this taxon
- Number of fragments assigned directly to this taxon

- A rank code, indicating (U)nclassified, (R)oot, (D)omain, (K)ingdom, (P)hylum, (C)lass, (O)rder, (F)amily, (G)enus, or (S)pecies. Taxa that are not at any of these 10 ranks have a rank code that is formed by using the rank code of the closest ancestor rank with a number indicating the distance from that rank. E.g., "G2" is a rank code indicating a taxon is between genus and species and the grandparent taxon is at the genus rank.
- NCBI taxonomic ID number
- Indented scientific name

## Exploring the results with Pavian


Pavian is available on a separate website: <https://fbreitwieser.shinyapps.io/pavian/>. To use it, download the Kraken reports you created in Galaxy, or [use the prepared reports here](#). Extract the report files from the zip file, and upload them into Pavian.



The screenshot shows the Pavian web interface. At the top, there is a text box stating: "Pavian is a tool for interactive analysis of metagenomics classification results. Read more about it in the [Preprint](#) or its [vignette](#). It's Centrifuge." Below this, another text box says: "For help and more documentation please go to <https://github.com/fbreitwieser/pavian>." The main interface has two tabs: "Upload files" (selected) and "Example data". Under the "Upload files" tab, there is a text instruction: "Upload metagenomics report files from the local computer. If selecting multiple files does not work, please try with a different browser." Below this instruction are two buttons: "Browse..." and "Upload report files".

## Exploring the results with Krona

There are two steps that take the Kraken2 report and create the visualisation with Krona. You must convert reports with the "Krakentools Convert kraken report file" as shown below.





 **KrakenTools: Convert kraken report file** to krona text file (Galaxy Version 1.2+galaxy1)

---

**Tool Parameters**

---

**Kraken report file \***



56: Kraken2 on collection 55: Report

accepted formats ▼

**!** This is a batch mode input field. Individual jobs will be triggered for each dataset.

(--report)

**Include intermediate (i.e. non-standard) ranks**

☒ No

(--intermediate-ranks)

---


**Additional Options**

---

**Email notification**

☒ No

Send an email notification when the job completes.


 **Run Tool**

---

**Help**

---

You must then use the output of this step in Krona, and set the input type to be Tabular.

 **Krona pie chart** from taxonomic profile (Galaxy Version 2.7.1+galaxy0)



**Tool Parameters**

**What is the type of your input data**

Tabular

Choose between Galaxy Taxonomy and generic table format (e.g. from MetaPhlAn or mothur)

**Input file \***

  ...

76: Krakentools: Convert kraken report file on collection 56 x

accepted formats ▼ switch to column select ▼

Select a MetaPhlAn dataset

**Provide a name for the basal rank** - optional

Root

-n; Otherwise it will simply be called "Root"

**Combine data from multiple datasets?**

☒ No


-c; Combine data from each dataset, rather than creating separate datasets within the chart

**Additional Options**

**Email notification**

☒ No

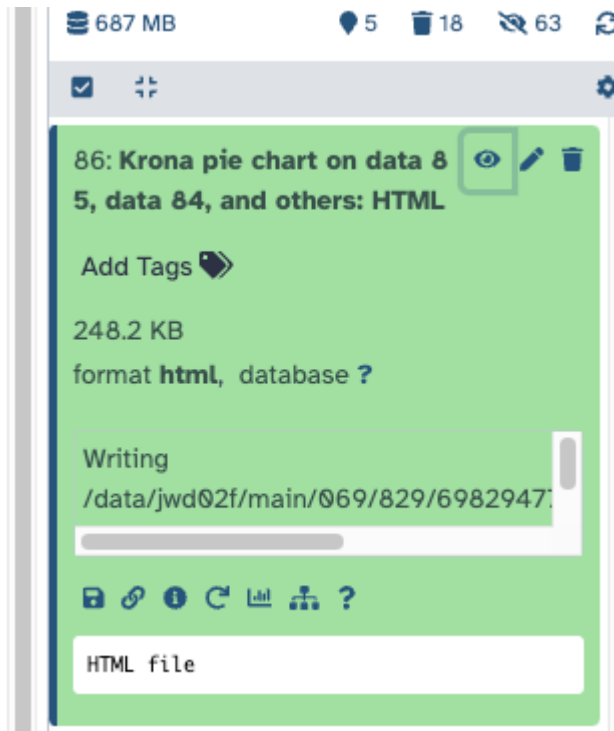
Send an email notification when the job completes.



Help

The output will be an HTML file, with the results of all the samples; You can open this directly in Galaxy using the "eye".





If you have difficulty running Krona, [here are the precalculated results](#)