Practical section I:
Introduction to R and RNAseq data analysis

Dr. Enrique Audain Martinez
September 2024

Outline

Part I: Introduction to R

- S1: Data structures and basic operations.
- S2: Data import and export.
- S3: Summary statistics and data visualization.

Part II: Introduction to RNAseq data analysis

- S4: RNAseq data analysis with DESeq2.
- S5: Exploring and visualizing RNAseq data.

Course materials on GitHub

What is GitHub?

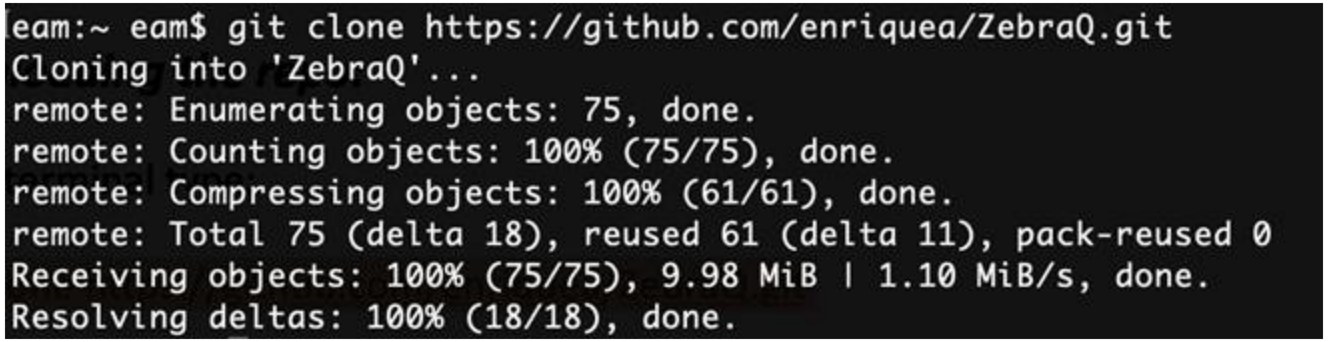
GitHub is like a digital library where people store and share their writing projects, allowing others to view, discuss, or contribute to them. Think of it as a collaborative workspace for writers, but instead of stories or essays, people work on computer programs.

GitHub repository:

https://github.com/enriquea/ZebraQ

Downloading the repo:

$ git clone https://github.com/enriquea/ZebraQ.git



```
eam:~ eam$ git clone https://github.com/enriquea/ZebraQ.git
Cloning into 'ZebraQ'...
remote: Enumerating objects: 75, done.
remote: Counting objects: 100% (75/75), done.
remote: Compressing objects: 100% (61/61), done.
remote: Total 75 (delta 18), reused 61 (delta 11), pack-reused 0
Receiving objects: 100% (75/75), 9.98 MiB | 1.10 MiB/s, done.
Resolving deltas: 100% (18/18), done.
```

Course materials (Dataset)

Disclaimer: The data used in this course is intended to be used for educational purposes only.

/data folder contains the following files:

- Genes associated with CHD with functional annotations:

  /data/chd_genes.annotations.tsv

- Gene counts from RNA-seq data of wild type and mutant zebrafish (*Danio Rerio*) hearts:

  /data/salmon.merged.gene_counts.tsv

Outline

Part I: Introduction to R (Supp. slides)

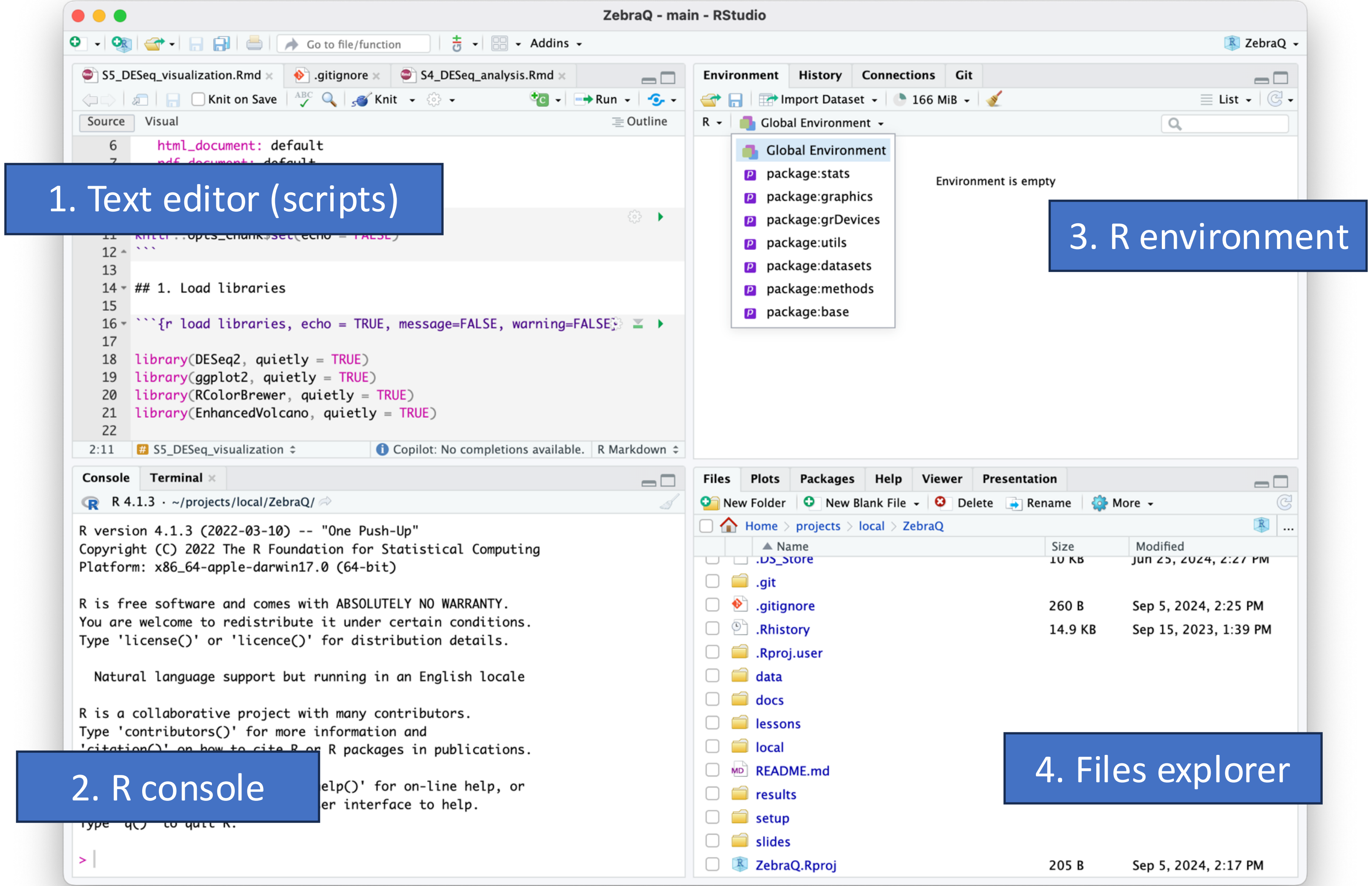
- S1: Data structures and basic operations.
- S2: Data import and export.
- S3: Summary statistics and data visualization.

Part II: Introduction to RNAseq data analysis

- S4: RNAseq data analysis with DESeq2.
- S5: Exploring and visualizing RNAseq data.

Part I: Introduction to R

RStudio



1. Text editor (scripts)

```
  html_document: default
  pdf_document: default
knitr::opts_chunk$set(echo = FALSE)
```

1. Load libraries

```{r load libraries, echo = TRUE, message=FALSE, warning=FALSE}

library(DESeq2, quietly = TRUE)
library(ggplot2, quietly = TRUE)
library(RColorBrewer, quietly = TRUE)
library(EnhancedVolcano, quietly = TRUE)

```

2. R console

```
R version 4.1.3 (2022-03-10) -- "One Push-Up"
Copyright (C) 2022 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin17.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

                                 elp()' for on-line help, or
                                 er interface to help.
Type 'q()' to quit R.

> 
```

3. R environment

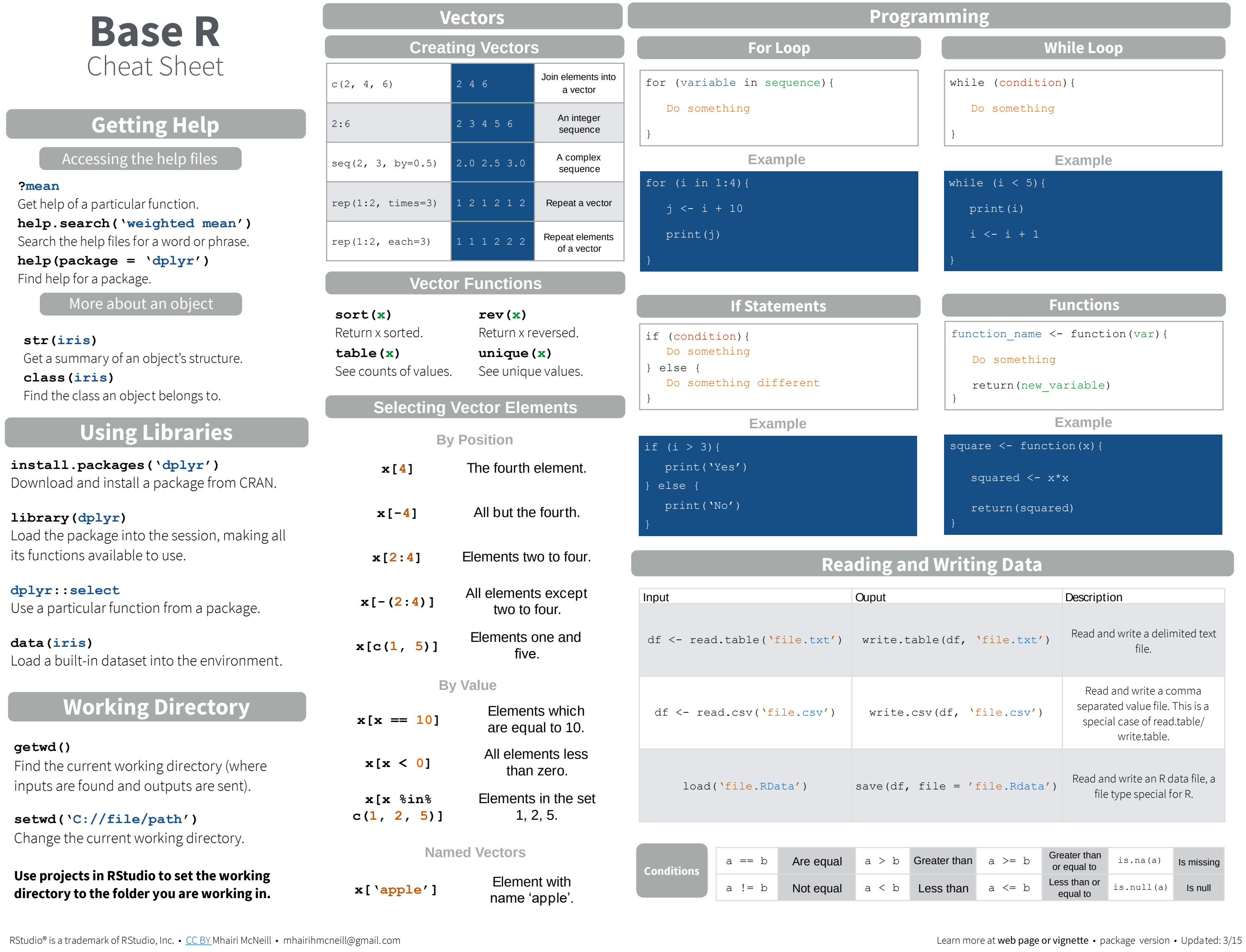
4. Files explorer

Name	Size	Modified
.DS_Store	10 KB	Jun 25, 2024, 2:27 PM
.git		
.gitignore	260 B	Sep 5, 2024, 2:25 PM
.Rhistory	14.9 KB	Sep 15, 2023, 1:39 PM
.Rproj.user		
data		
docs		
lessons		
local		
README.md		
results		
setup		
slides		
ZebraQ.Rproj	205 B	Sep 5, 2024, 2:17 PM

IDE: *Integrated Development Environment*

Image credit: https://education.rstudio.com/teach/tools

Part I: Introduction to R



Base R
Cheat Sheet

Getting Help

Accessing the help files

?mean
Get help of a particular function.

help.search(‘weighted mean’)
Search the help files for a word or phrase.

help(package = ‘dplyr’)
Find help for a package.

More about an object

str(iris)
Get a summary of an object’s structure.

class(iris)
Find the class an object belongs to.

Using Libraries

install.packages(‘dplyr’)
Download and install a package from CRAN.

library(dplyr)
Load the package into the session, making all its functions available to use.

dplyr::select
Use a particular function from a package.

data(iris)
Load a built-in dataset into the environment.

Working Directory

getwd()
Find the current working directory (where inputs are found and outputs are sent).

setwd(‘C://file/path’)
Change the current working directory.

Use projects in RStudio to set the working directory to the folder you are working in.

Vectors

Creating Vectors

`c(2, 4, 6)`	`2 4 6`	Join elements into a vector
`2:6`	`2 3 4 5 6`	An integer sequence
`seq(2, 3, by=0.5)`	`2.0 2.5 3.0`	A complex sequence
`rep(1:2, times=3)`	`1 2 1 2 1 2`	Repeat a vector
`rep(1:2, each=3)`	`1 1 1 2 2 2`	Repeat elements of a vector

Vector Functions

sort(x)
Return x sorted.

table(x)
See counts of values.

rev(x)
Return x reversed.

unique(x)
See unique values.

Selecting Vector Elements

By Position

`x[4]`	The fourth element.
`x[-4]`	All but the fourth.
`x[2:4]`	Elements two to four.
`x[-(2:4)]`	All elements except two to four.
`x[c(1, 5)]`	Elements one and five.

By Value

`x[x == 10]`	Elements which are equal to 10.
`x[x < 0]`	All elements less than zero.
`x[x %in% c(1, 2, 5)]`	Elements in the set 1, 2, 5.

Named Vectors

`x[‘apple’]`	Element with name ‘apple’.

Programming

For Loop

```
for (variable in sequence){

   Do something

}
```

Example

```
for (i in 1:4){

   j <- i + 10

   print(j)

}
```

While Loop

```
while (condition){

   Do something

}
```

Example

```
while (i < 5){

   print(i)

   i <- i + 1

}
```

If Statements

```
if (condition){
   Do something
} else {
   Do something different
}
```

Example

```
if (i > 3){
   print(‘Yes’)
} else {
   print(‘No’)
}
```

Functions

```
function_name <- function(var){

   Do something

   return(new_variable)
}
```

Example

```
square <- function(x){

   squared <- x*x

   return(squared)
}
```

Reading and Writing Data

Input	Ouput	Description
`df <- read.table(‘file.txt’)`	`write.table(df, ‘file.txt’)`	Read and write a delimited text file.
`df <- read.csv(‘file.csv’)`	`write.csv(df, ‘file.csv’)`	Read and write a comma separated value file. This is a special case of read.table/ write.table.
`load(‘file.RData’)`	`save(df, file = ’file.Rdata’)`	Read and write an R data file, a file type special for R.

Conditions

`a == b`	Are equal	`a > b`	Greater than	`a >= b`	Greater than or equal to	`is.na(a)`	Is missing
`a != b`	Not equal	`a < b`	Less than	`a <= b`	Less than or equal to	`is.null(a)`	Is null

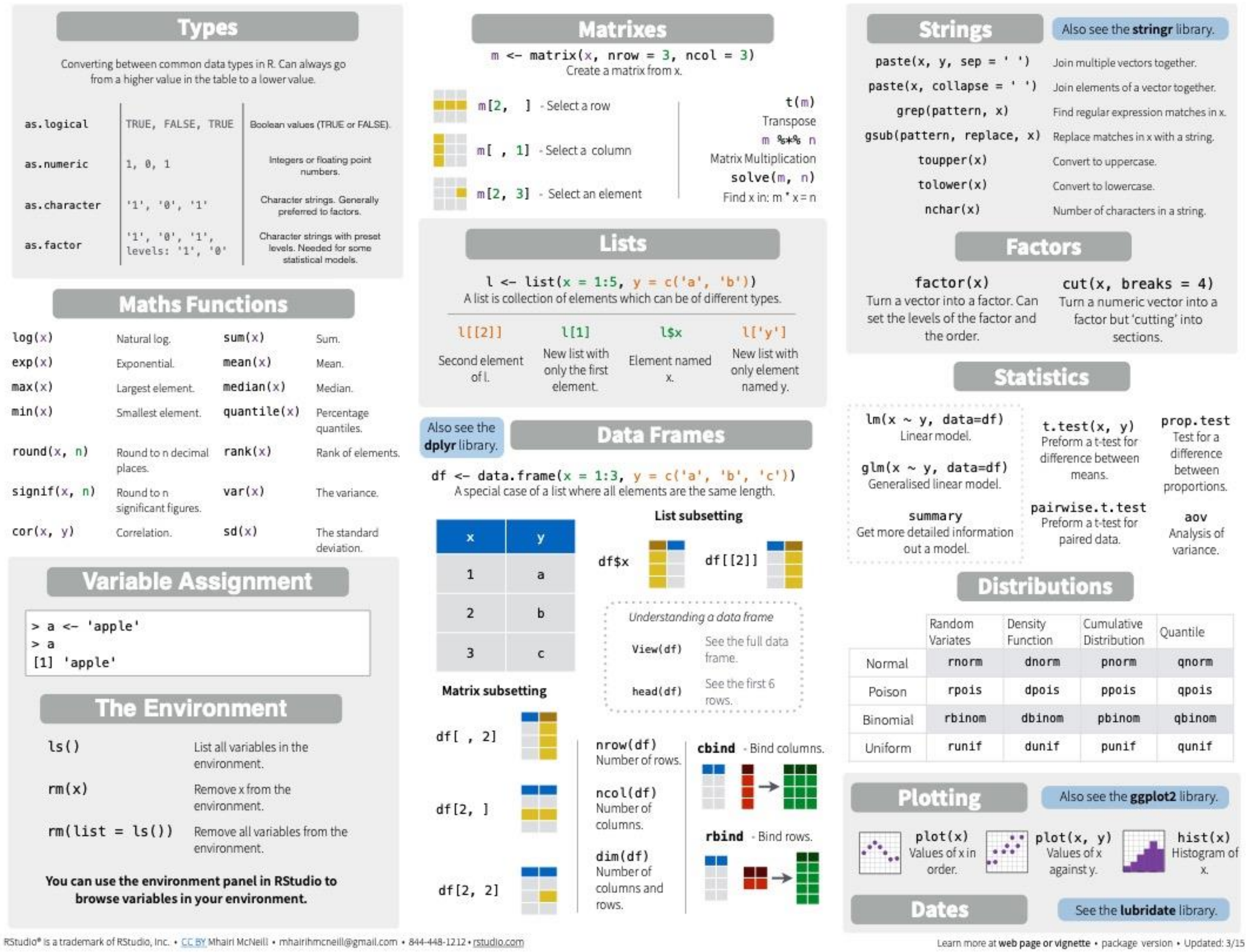
RStudio® is a trademark of RStudio, Inc. • CC BY Mhairi McNeill • mhairihmcneill@gmail.com

Learn more at **web page or vignette** • package version • Updated: 3/15

https://iqss.github.io/dss-workshops

Carl von Ossietzky
Universität
Oldenburg

Part I: Introduction to R



Types

Converting between common data types in R. Can always go from a higher value in the table to a lower value.

`as.logical`	`TRUE, FALSE, TRUE`	Boolean values (TRUE or FALSE).
`as.numeric`	`1, 0, 1`	Integers or floating point numbers.
`as.character`	`'1', '0', '1'`	Character strings. Generally preferred to factors.

`as.factor`	`'1', '0', '1',`
	`levels: '1', '0'`
		Character strings with preset levels. Needed for some statistical models.

Maths Functions

`log(x)`	Natural log.	`sum(x)`	Sum.
`exp(x)`	Exponential.	`mean(x)`	Mean.
`max(x)`	Largest element.	`median(x)`	Median.
`min(x)`	Smallest element.	`quantile(x)`	Percentage quantiles.
`round(x, n)`	Round to n decimal places.	`rank(x)`	Rank of elements.
`signif(x, n)`	Round to n significant figures.	`var(x)`	The variance.
`cor(x, y)`	Correlation.	`sd(x)`	The standard deviation.

Variable Assignment

```
> a <- 'apple'
> a
[1] 'apple'
```

The Environment

`ls()`	List all variables in the environment.
`rm(x)`	Remove x from the environment.
`rm(list = ls())`	Remove all variables from the environment.

You can use the environment panel in RStudio to browse variables in your environment.

Matrixes

`m <- matrix(x, nrow = 3, ncol = 3)`
Create a matrix from x.

`m[2, ]` - Select a row

`m[ , 1]` - Select a column

`m[2, 3]` - Select an element

`t(m)`
Transpose

`m %*% n`
Matrix Multiplication

`solve(m, n)`
Find x in: m * x = n

Lists

`l <- list(x = 1:5, y = c('a', 'b'))`
A list is collection of elements which can be of different types.

`l[[2]]`	`l[1]`	`l$x`	`l['y']`
Second element of l.	New list with only the first element.	Element named x.	New list with only element named y.

Data Frames

Also see the **dplyr** library.

`df <- data.frame(x = 1:3, y = c('a', 'b', 'c'))`
A special case of a list where all elements are the same length.

x	y
1	a
2	b
3	c

List subsetting

`df$x` `df[[2]]`

Understanding a data frame

`View(df)`	See the full data frame.
`head(df)`	See the first 6 rows.

Matrix subsetting

`df[ , 2]`

`df[2, ]`

`df[2, 2]`

`nrow(df)`
Number of rows.

`ncol(df)`
Number of columns.

`dim(df)`
Number of columns and rows.

cbind - Bind columns.

rbind - Bind rows.

Strings

Also see the **stringr** library.

`paste(x, y, sep = ' ')`	Join multiple vectors together.
`paste(x, collapse = ' ')`	Join elements of a vector together.
`grep(pattern, x)`	Find regular expression matches in x.
`gsub(pattern, replace, x)`	Replace matches in x with a string.
`toupper(x)`	Convert to uppercase.
`tolower(x)`	Convert to lowercase.
`nchar(x)`	Number of characters in a string.

Factors

`factor(x)`
Turn a vector into a factor. Can set the levels of the factor and the order.

`cut(x, breaks = 4)`
Turn a numeric vector into a factor but 'cutting' into sections.

Statistics

`lm(x ~ y, data=df)`
Linear model.

`glm(x ~ y, data=df)`
Generalised linear model.

`summary`
Get more detailed information out a model.

`t.test(x, y)`
Preform a t-test for difference between means.

`pairwise.t.test`
Preform a t-test for paired data.

`prop.test`
Test for a difference between proportions.

`aov`
Analysis of variance.

Distributions

	Random Variates	Density Function	Cumulative Distribution	Quantile
Normal	`rnorm`	`dnorm`	`pnorm`	`qnorm`
Poison	`rpois`	`dpois`	`ppois`	`qpois`
Binomial	`rbinom`	`dbinom`	`pbinom`	`qbinom`
Uniform	`runif`	`dunif`	`punif`	`qunif`

Plotting

Also see the **ggplot2** library.

`plot(x)`
Values of x in order.

`plot(x, y)`
Values of x against y.

`hist(x)`
Histogram of x.

Dates

See the **lubridate** library.

Learn more at **web page or vignette** • package version • Updated: 3/15

https://iqss.github.io/dss-workshops

Outline

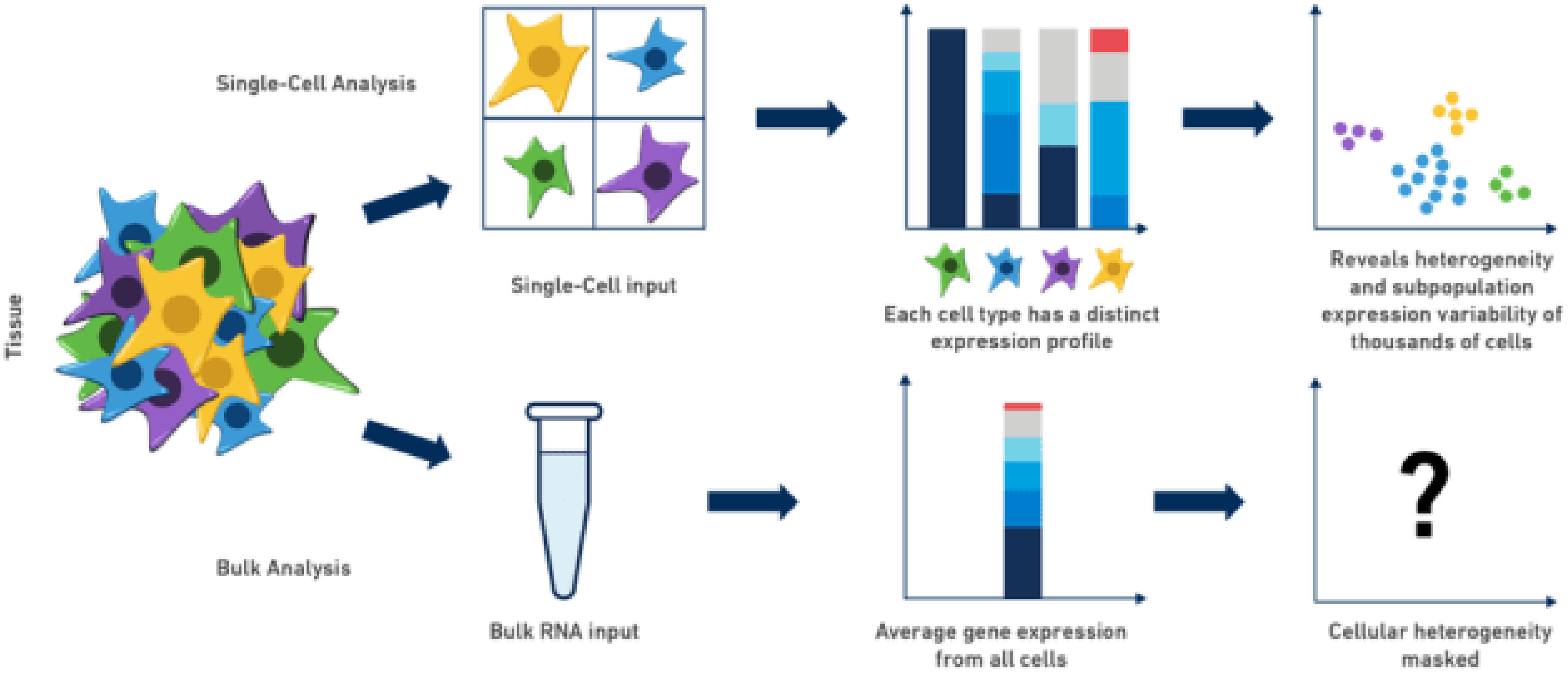
Part I: Introduction to R

- S1: Data structures and basic operations.
- S2: Data import and export.
- S3: Summary statistics and data visualization.

Part II: Introduction to RNAseq data analysis (Supp. slides)

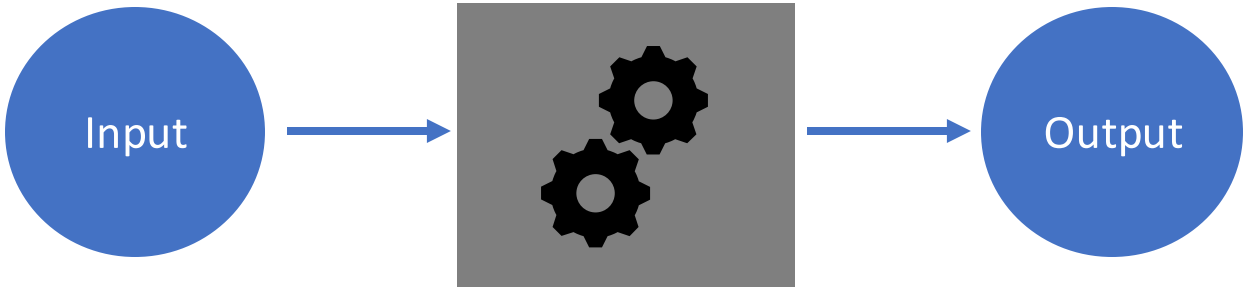
- S4: RNAseq data analysis with DESeq2.
- S5: Exploring and visualizing RNAseq data.

Bulk vs scRNA-seq


Single-Cell Analysis
Tissue
Single-Cell input
Each cell type has a distinct expression profile
Reveals heterogeneity and subpopulation expression variability of thousands of cells
Bulk Analysis
Bulk RNA input
Average gene expression from all cells
?
Cellular heterogeneity masked

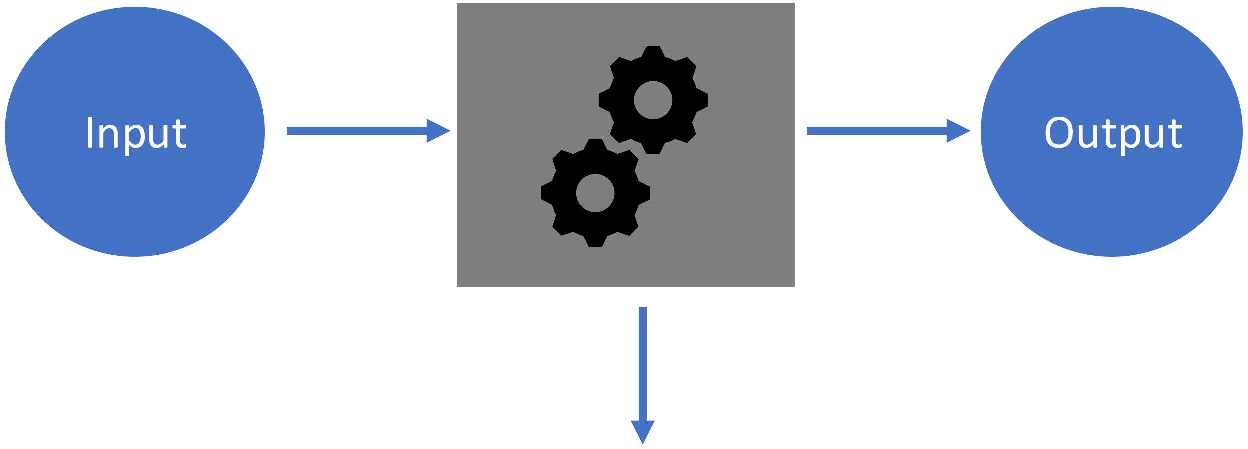
https://www.10xgenomics.com/blog/single-cell-rna-seq-an-introductory-overview-and-tools-for-getting-started

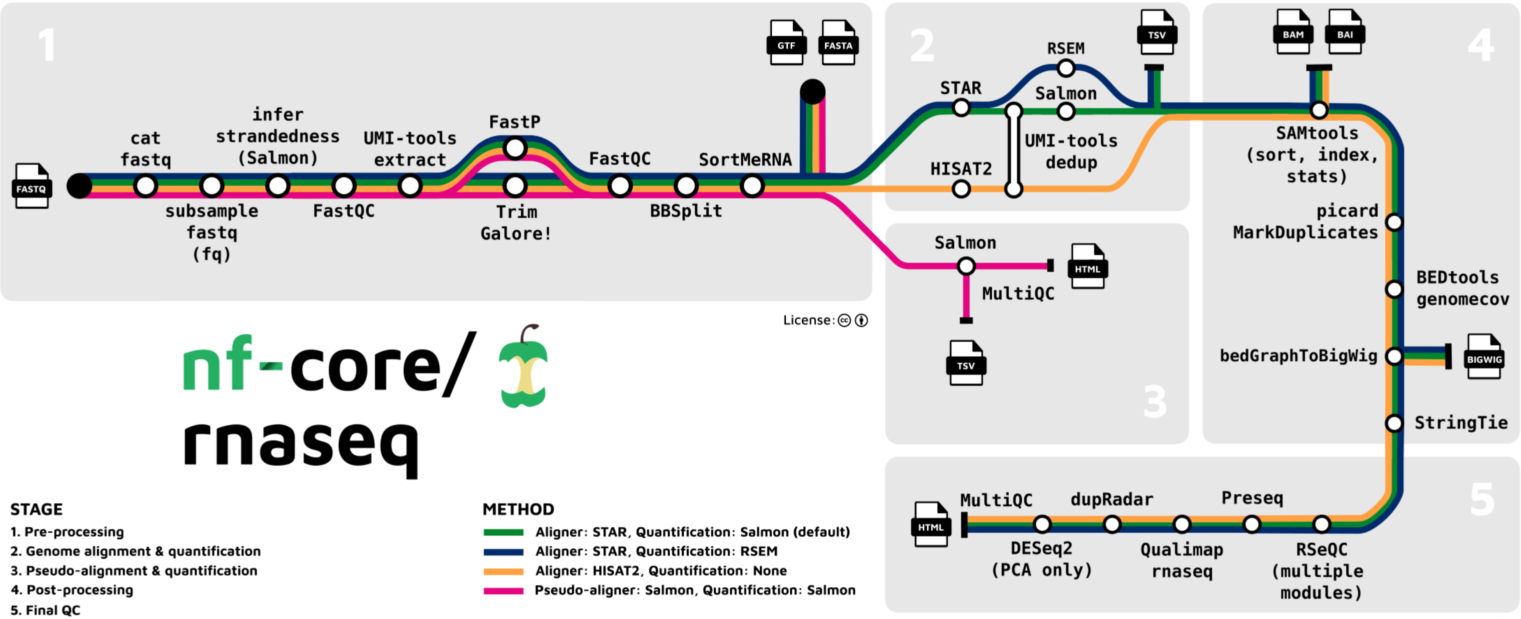
RNAseq pipeline



Black box?

RNAseq pipeline





nf-core/
rnaseq

STAGE

1. Pre-processing
2. Genome alignment & quantification
3. Pseudo-alignment & quantification
4. Post-processing
5. Final QC

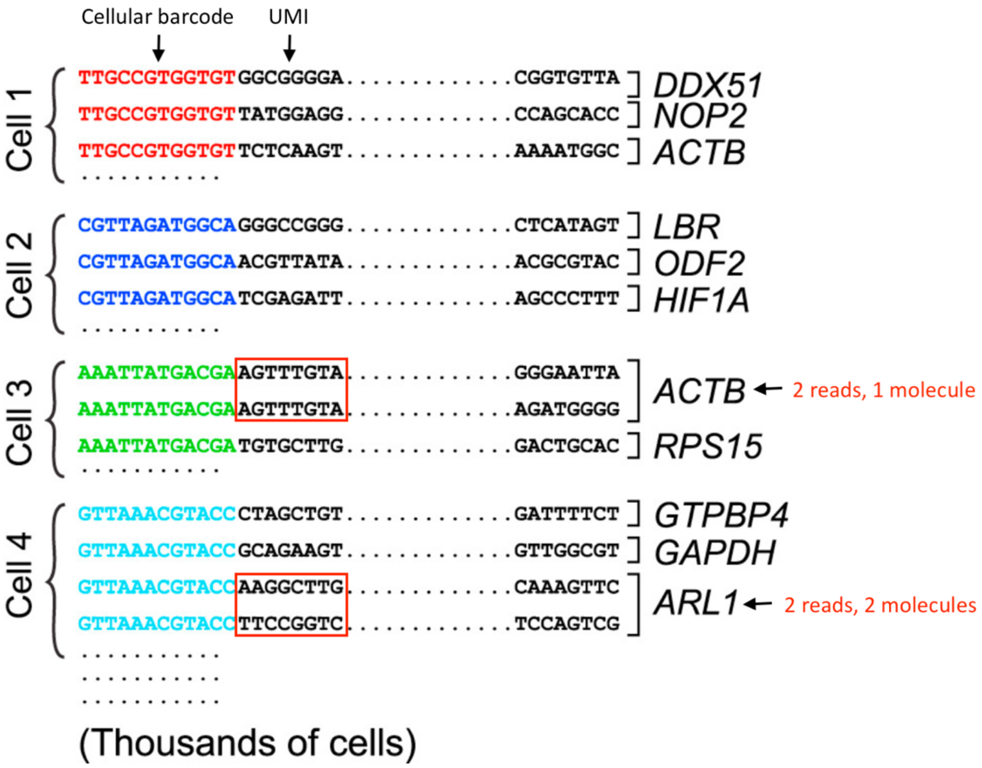
METHOD

- 🟩 Aligner: STAR, Quantification: Salmon (default)
- 🟦 Aligner: STAR, Quantification: RSEM
- 🟧 Aligner: HISAT2, Quantification: None
- 🟪 Pseudo-aligner: Salmon, Quantification: Salmon

https://nf-co.re/rnaseq

“Matrix count”

“The outcome of this procedure is a gene/cell count matrix, which is used as an estimate of the number of RNA molecules in each cell for each gene”



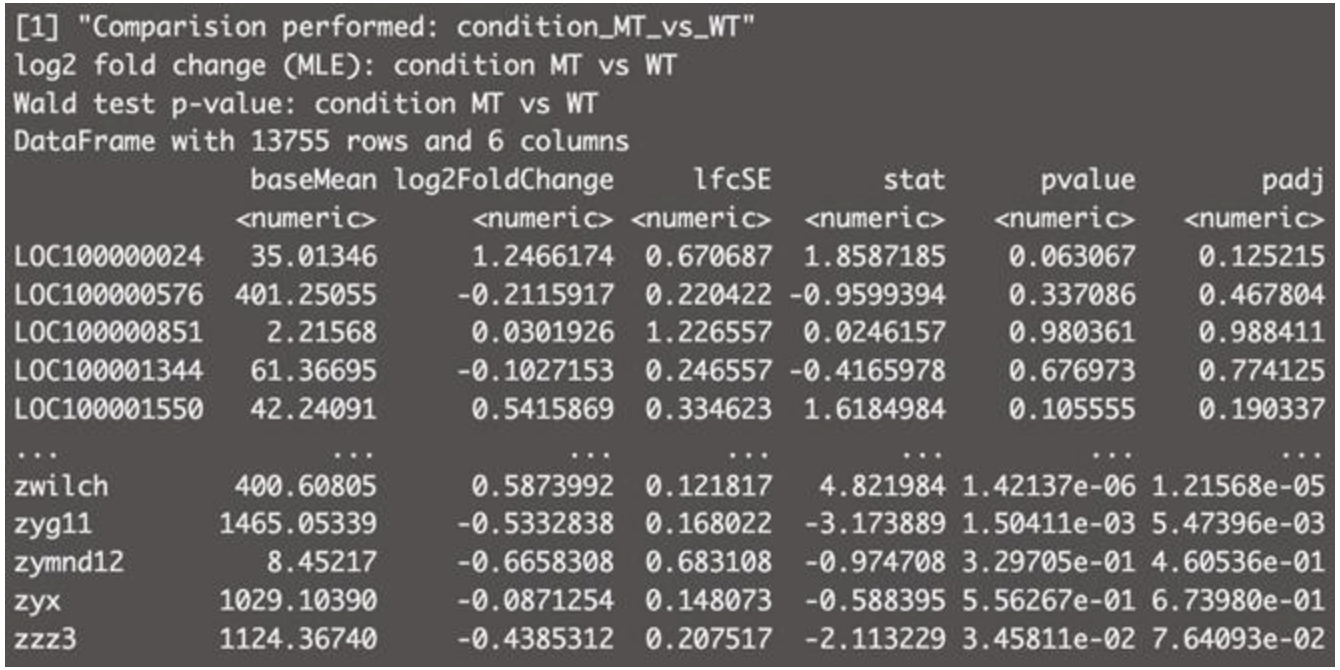
Cellular barcode
UMI
Cell 1
TTGCCGTGGTGT GGCGGGGA..............CGGTGTTA DDX51
TTGCCGTGGTGT TATGGAGG..............CCAGCACC NOP2
TTGCCGTGGTGT TCTCAAGT..............AAAATGGC ACTB
Cell 2
CGTTAGATGGCA GGGCCGGG..............CTCATAGT LBR
CGTTAGATGGCA ACGTTATA..............ACGCGTAC ODF2
CGTTAGATGGCA TCGAGATT..............AGCCCTTT HIF1A
Cell 3
AAATTATGACGA AGTTTGTA..............GGGAATTA
AAATTATGACGA AGTTTGTA..............AGATGGGG
ACTB
2 reads, 1 molecule
AAATTATGACGA TGTGCTTG..............GACTGCAC RPS15
Cell 4
GTTAAACGTACC CTAGCTGT..............GATTTTCT GTPBP4
GTTAAACGTACC GCAGAAGT..............GTTGGCGT GAPDH
GTTAAACGTACC AAGGCTTG..............CAAAGTTC
GTTAAACGTACC TTCCGGTC..............TCCAGTCG
ARL1
2 reads, 2 molecules
(Thousands of cells)

	Cell1	Cell2	...	CellN
Gene1	3	2	.	13
Gene2	2	3	.	1
Gene3	1	14	.	18
...	.	.	.	.
...	.	.	.	.
...	.	.	.	.
GeneM	25	0	.	0

Source: https://github.com/hbctraining/scRNA-seq

Results from DESeq experiment

The results table when printed will provide the information about the comparison, e.g. "log2 fold change (MAP): condition treated vs untreated", meaning that the estimates are of log2(treated / untreated).



```
[1] "Comparision performed: condition_MT_vs_WT"
log2 fold change (MLE): condition MT vs WT
Wald test p-value: condition MT vs WT
DataFrame with 13755 rows and 6 columns
               baseMean log2FoldChange     lfcSE       stat      pvalue        padj
              <numeric>      <numeric> <numeric>  <numeric>   <numeric>   <numeric>
LOC100000024   35.01346      1.2466174  0.670687  1.8587185    0.063067    0.125215
LOC100000576  401.25055     -0.2115917  0.220422 -0.9599394    0.337086    0.467804
LOC100000851    2.21568      0.0301926  1.226557  0.0246157    0.980361    0.988411
LOC100001344   61.36695     -0.1027153  0.246557 -0.4165978    0.676973    0.774125
LOC100001550   42.24091      0.5415869  0.334623  1.6184984    0.105555    0.190337
...                 ...            ...       ...        ...         ...         ...
zwilch        400.60805      0.5873992  0.121817   4.821984 1.42137e-06 1.21568e-05
zyg11        1465.05339     -0.5332838  0.168022  -3.173889 1.50411e-03 5.47396e-03
zymnd12         8.45217     -0.6658308  0.683108  -0.974708 3.29705e-01 4.60536e-01
zyx          1029.10390     -0.0871254  0.148073  -0.588395 5.56267e-01 6.73980e-01
zzz3         1124.36740     -0.4385312  0.207517  -2.113229 3.45811e-02 7.64093e-02
```

Log2FoldChange: For a given comparison, a positive fold change value indicates an increase of expression, while a negative fold change indicates a decrease in expression.

P-value: Indicates whether the gene analysed is likely to be differentially expressed in that comparison.

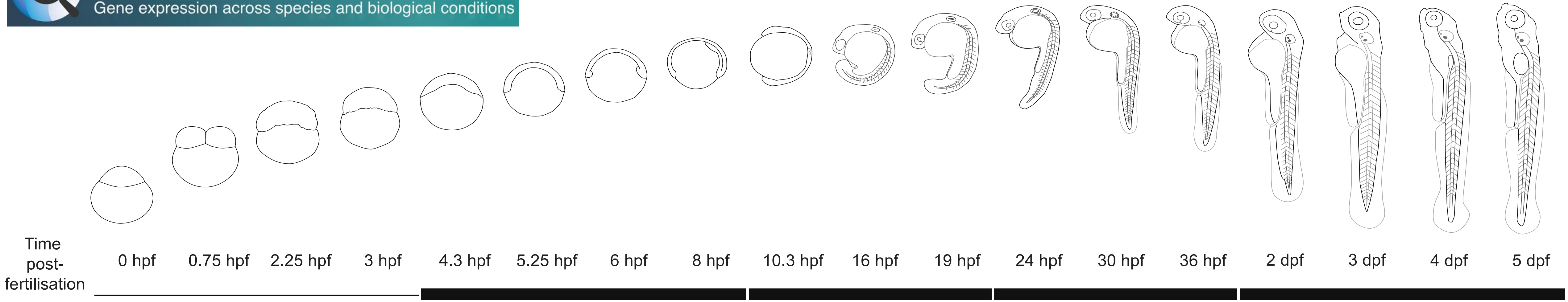
Adj. p-value: The p-value obtained for each gene above is re-calculated to correct for multiple testing (n genes).

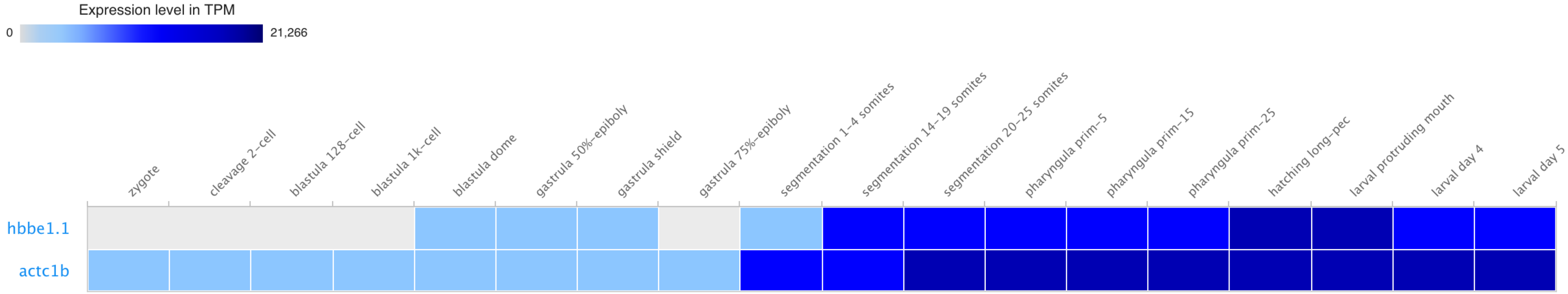
https://biocorecrg.github.io/CRG_Bioinformatics_for_Biologists/differential_gene_expression.html

Comparing results with external resources/databases

Expression Atlas

Gene expression across species and biological conditions

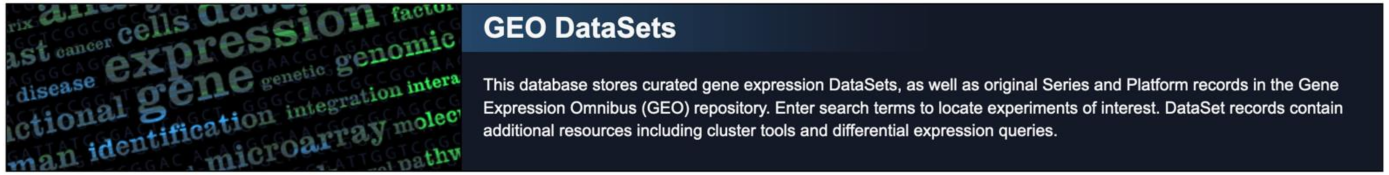




https://www.ebi.ac.uk/gxa/experiments/E-ERAD-475/Results; https://doi.org/10.7554/eLife.30860

Other resources & databases

- Co-expression analysis



GEO DataSets

This database stores curated gene expression DataSets, as well as original Series and Platform records in the Gene Expression Omnibus (GEO) repository. Enter search terms to locate experiments of interest. DataSet records contain additional resources including cluster tools and differential expression queries.

https://www.ncbi.nlm.nih.gov/geo/

- Protein-protein interaction analysis

STRING https://string-db.org/

- Pathway enrichment analysis

reactome https://reactome.org/