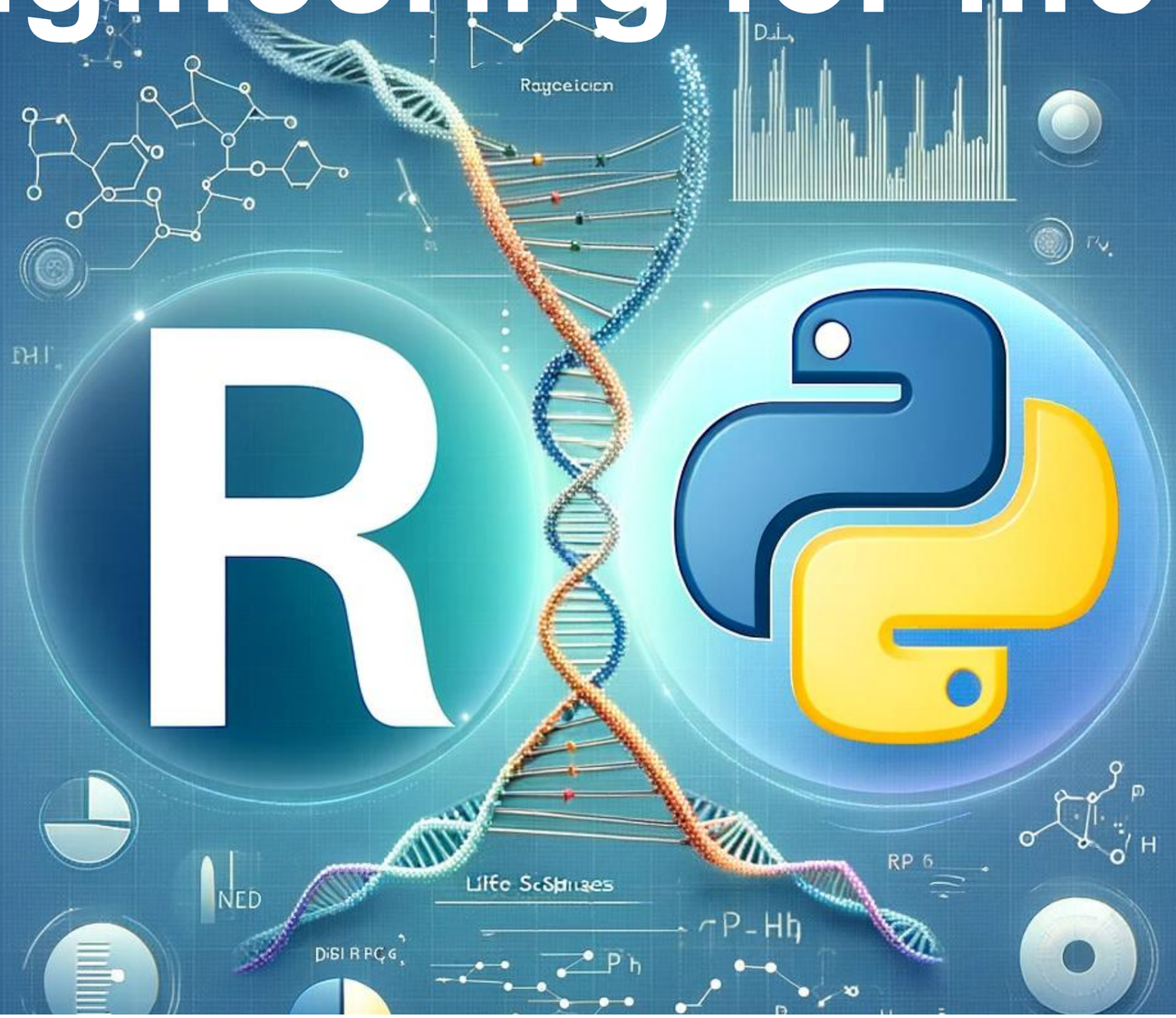
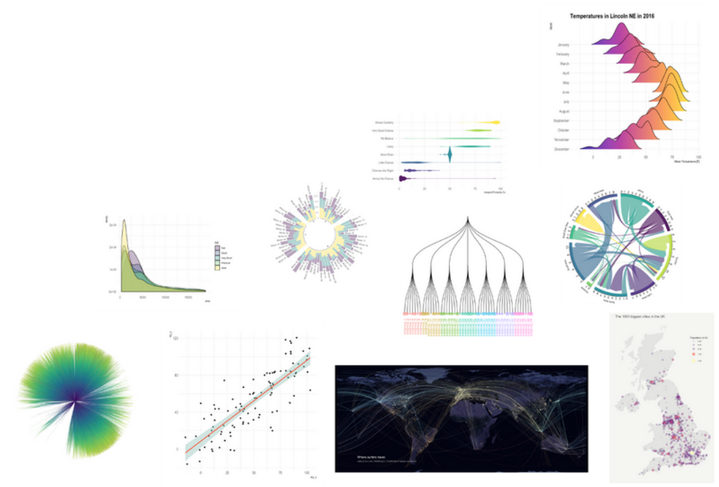
# Data Engineering for life science

**MODULE -1**




MANCHESTER 1824
The University of Manchester

pacem Technology




## INSTALLING R AND RSTUDIO

To begin using R, you must first install R and RStudio. Visit [R Project](https://www.r-project.org/) and [RStudio](https://www.rstudio.com/) to download the necessary files. Follow the installation guides for your operating system to set up your R environment.

### Introduction to R

- R is a powerful statistical programming language.
- R is widely used for data analysis and visualization.

### Why R is important for data science

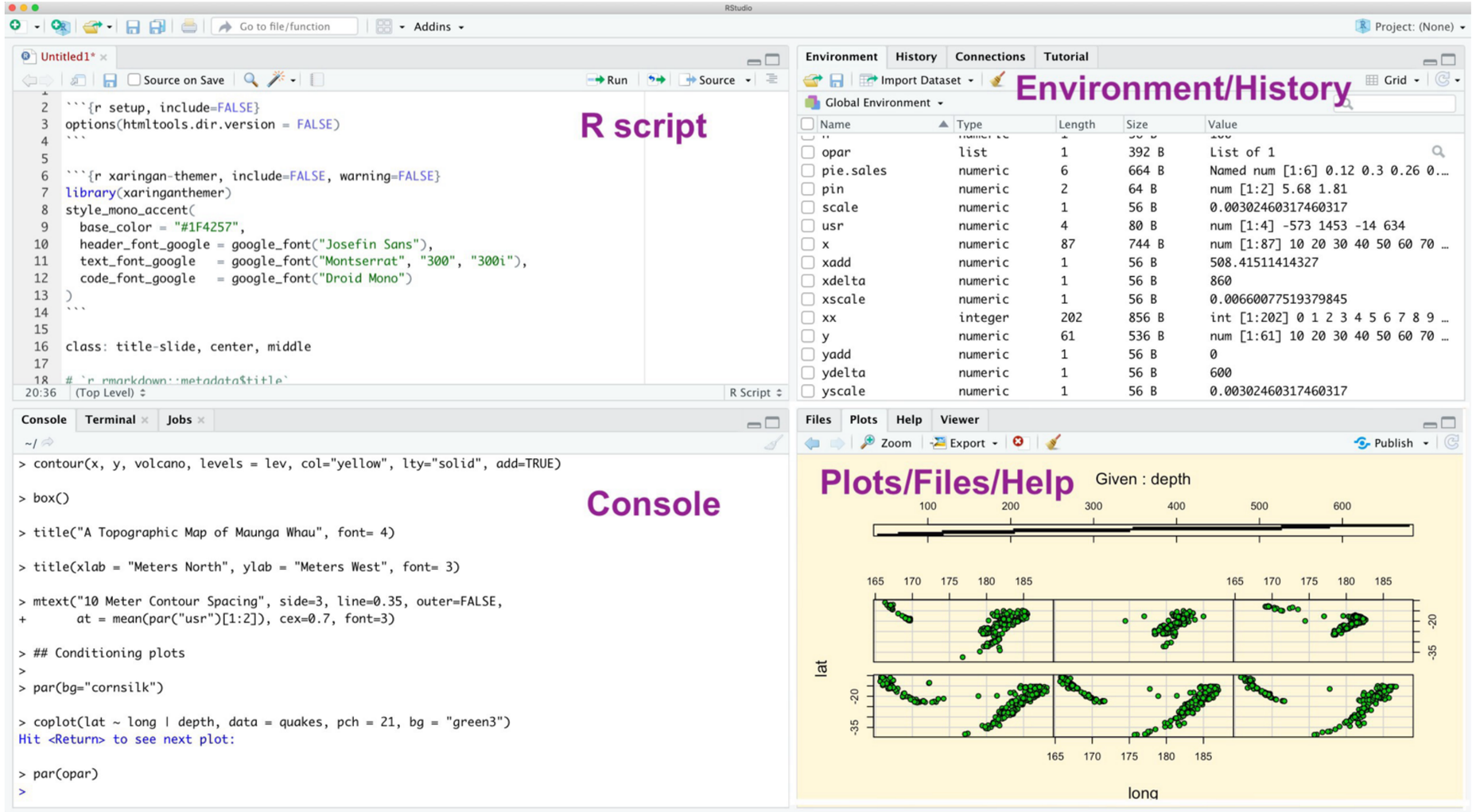
- R offers extensive statistical capabilities for analysis.
- R facilitates data manipulation and visualization tasks.

## why you should learn R

Employability

Opportunity to get involved with a fantastic and supportive community

# Rstudio


RStudio
Go to file/function
Addins
Project: (None)
Untitled1*
Source on Save
Run
Source
R script
```{r setup, include=FALSE}
options(htmltools.dir.version = FALSE)
```
```{r xaringan-themer, include=FALSE, warning=FALSE}
library(xaringanthemer)
style_mono_accent(
base_color = "#1F4257",
header_font_google = google_font("Josefin Sans"),
text_font_google = google_font("Montserrat", "300", "300i"),
code_font_google = google_font("Droid Mono")
)
```
class: title-slide, center, middle
# `r rmarkdown::metadata$title`
20:36
(Top Level)
R Script
Environment
History
Connections
Tutorial
Import Dataset
Grid
Global Environment
Environment/History
Name
Type
Length
Size
Value
opar list 1 392 B List of 1
pie.sales numeric 6 664 B Named num [1:6] 0.12 0.3 0.26 0.…
pin numeric 2 64 B num [1:2] 5.68 1.81
scale numeric 1 56 B 0.00302460317460317
usr numeric 4 80 B num [1:4] -573 1453 -14 634
x numeric 87 744 B num [1:87] 10 20 30 40 50 60 70 …
xadd numeric 1 56 B 508.41511414327
xdelta numeric 1 56 B 860
xscale numeric 1 56 B 0.00660077519379845
xx integer 202 856 B int [1:202] 0 1 2 3 4 5 6 7 8 9 …
y numeric 61 536 B num [1:61] 10 20 30 40 50 60 70 …
yadd numeric 1 56 B 0
ydelta numeric 1 56 B 600
yscale numeric 1 56 B 0.00302460317460317
Console
Terminal
Jobs
~/
> contour(x, y, volcano, levels = lev, col="yellow", lty="solid", add=TRUE)
> box()
> title("A Topographic Map of Maunga Whau", font= 4)
> title(xlab = "Meters North", ylab = "Meters West", font= 3)
> mtext("10 Meter Contour Spacing", side=3, line=0.35, outer=FALSE,
+ at = mean(par("usr")[1:2]), cex=0.7, font=3)
> ## Conditioning plots
>
> par(bg="cornsilk")
> coplot(lat ~ long | depth, data = quakes, pch = 21, bg = "green3")
Hit <Return> to see next plot:
> par(opar)
>
Console
Files
Plots
Help
Viewer
Zoom
Export
Publish
Plots/Files/Help
Given : depth
100 200 300 400 500 600
165 170 175 180 185
-20 -35
lat
long

## INSTALL PACKAGES

R enhances its capabilities through packages.
These packages contain functions, data, and compiled code in a well-defined format. The primary repository for R packages is CRAN (The Comprehensive R Archive Network).
To install a package, use the command install.packages('package_name'). For example, install.packages('rafalib'). Once installed, load a package into your session with library('package_name').

https://swirlstats.com

https://www.statmethods.net

?install.packages
help("install.packages")

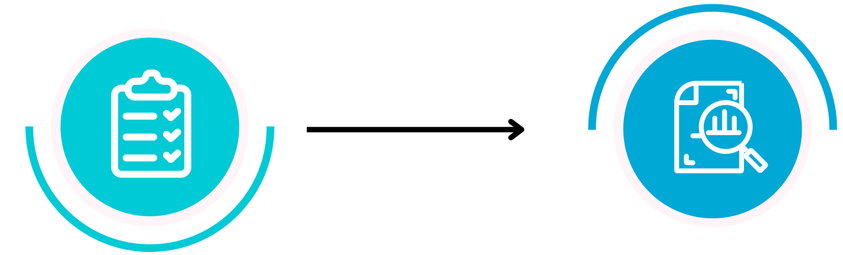
Install and upload these packages:

#rafalib
#downloader

# IMPORTING DATA INTO R

Data can be imported into R in various formats. CSV and Excel are common formats. Use read.csv('file.csv') for CSV files. For Excel files, libraries like readxl can be used. It's essential to understand file paths and the working directory in R, as they determine where R looks for files.

https://raw.githubusercontent.com/genomicsclass/dagdata/master/inst/extdata/femaleMiceWeights.csv



downloader read.csv

Many of the datasets we include in this book are available in custom-built packages from GitHub. The reason we use GitHub, rather than CRAN, is that on GitHub we do not have to vet packages, which gives us much more flexibility.

**install.packages("devtools")**

"genomicsclass/dagdata"

## HOMEWORK

download the **femaleMiceWeights.csv** using downloader package in your path.

1. Read in the file femaleMiceWeights.csv and report the body weight of the mouse in the exact name of the column containing the weights.
2. The [ and ] symbols can be used to extract specific rows and specific columns of the table.
What is the entry in the 12th row and second column?
3. You should have learned how to use the $ character to extract a column from a table and return it as a vector.
Use $ to extract the weight column and report the weight of the mouse in the 11th row.
4. The length function returns the number of elements in a vector. How many mice are included
in our dataset?
5. To create a vector with the numbers 3 to 7, we can use seq(3,7) or, because they are consecutive, 3:7. View the data and determine what rows are associated with the high fat or hf diet. Then use the mean function to compute the average weight of these mice.
6. One of the functions we will be using often is sample. Read the help file for sample using
?sample. Now take a random sample of size 1 from the numbers 13 to 24 and report back the weight of the mouse represented by that row. Make sure to type set.seed(1) to ensure that everybody gets the same answer

## Brief Introduction to dplyr

The learning curve for R syntax is slow. One of the more difficult aspects that requires some getting used to is subsetting data tables. The dplyr packages brings these tasks closer to English and we are therefore going to introduce two simple functions: one is used to subset and the other to select columns.

```
filename <- "femaleMiceWeights.csv"
dat <- read.csv(filename)
head(dat) or view(dat)
```

```
library(dplyr)
chow <- filter(dat, Diet=="chow") #keep only the ones with chow diet
head(chow)

chowVals <- select(chow,Bodyweight)
head(chowVals)
```

| ## | | **Diet** | **Bodyweight** |
|---|---|---|---|
| ## | 1 | chow | 21.51 |
| ## | 2 | chow | 28.14 |
| ## | 3 | chow | 24.04 |
| ## | 4 | chow | 23.45 |
| ## | 5 | chow | 23.68 |
| ## | 6 | chow | 19.79 |

## HOMEWORK - 2

```
library(ggplot2)
data(msleep)
```

1. Read in the msleep_ggplot2.csv file with the function read.csv and use the function class to determine what type of object is returned.
2. Now use the filter function to select only the primates. How many animals in the table are primates? Hint: the nrow function gives you the number of rows of a data frame or matrix.
3. What is the class of the object you obtain after subsetting the table to only include primates?
4. Now use the select function to extract the sleep (total) for the primates. What class is this object? Hint: use %>% to pipe the results of the filter function to select.
5. Now we want to calculate the average amount of sleep for primates (the average of the numbers computed above). One challenge is that the mean function requires a vector so, if we simply apply it to the output above, we get an error. Look at the help file for unlist and use it to compute the desired average.

https://www.youtube.com/watch?v=Q5g6lYUn6Q4: until flow control