

# R: Introduction to Basic Features

Helen Lockstone  
hel23@well.ox.ac.uk



**iT Centre**  
**Learning**

**iT**  
services



# Aims of today's workshop



- A introduction to how you interact with and use R
- A sense of R's capabilities and how it works
- Explain some programming jargon and concepts, and the R language
- Awareness of the vital importance of good programming practice
- Use R to run some typical tasks

# R - Strengths and Weaknesses



- Incredibly powerful and versatile statistical programming software....but where do I start?
- Features that make life easier in many ways....but potential pitfalls as well
- Open-source, free software with a strong support and development community
- Extensive additional functionality for Genomics data through the Bioconductor project (<https://www.bioconductor.org/>)

# Course Overview



To harness its capabilities, an R user needs to be able to do all of the following:

1. Program instructions in the R language
2. Understand R's data structures, functions and behaviour
3. Use appropriate statistical tests and models for their data
4. Interpret the output correctly

On this course we focus on the first 2 items, introducing the R programming language and starting to handle data in the R environment. The course is intended to help those new to programming get acquainted with using R and help overcome some of the barriers created by technical jargon and notation.

# About you



1. I am using R for the first time today.
2. I tried using R but didn't get very far before I ran into something I didn't understand or a problem I couldn't resolve.
3. I've used R quite a bit, but sometimes it does unexpected things and I am not sure why.
4. I've started using R and thought 'Wow, this is wonderful - how easy and intuitive it is to use!' Anyone??

# About me



- Lead the Bioinformatics Core at Wellcome Centre for Human Genetics and have used R for a number of years for data analysis, particularly gene expression data.
- We have developed and taught a range of R and Bioinformatics Data Analysis courses to DPhil students and post-doctoral researchers over the past few years.

# Related R Courses



- R: Kick-off
- R: Introduction to Basic Features
- R: Data Handling
- R: Visualisation

[`https://skills.it.ox.ac.uk/whats-on#/`](https://skills.it.ox.ac.uk/whats-on#/)



We are grateful to Dave Baker, Gabriele Pani and the administrative staff at IT Learning Centre for their support and assistance.

A series of related R courses are currently offered through the IT Learning Centre, developed and taught by:

Punam Amratia (Big Data Institute)

Rohan Arambepola (Big Data Institute)

Helen Lockstone (Wellcome Centre for Human Genetics)

Andre Python (Big Data Institute)

Ben Wright (Wellcome Centre for Human Genetics)

Similar courses were previously taught by Samantha Curle, who recently moved on from Oxford. We are grateful to all of them for discussions, contributions and ideas.

# Schedule



09:30 - 10:00	Introducing the R environment, basic commands and data types
10:00 - 10:30	Understanding variables, functions and arguments
10:30 - 10:40	Tea/coffee break
10.40 - 11:30	Setting the working directory, reading in and accessing data
11.30 - 12:30	Performing simple data analysis and plotting

Course format: informal workshop-style - please feel free to ask questions at any time.

## Post-course information:

You can attend one of our related courses or use the Additional Material links to further develop your R skills.

One-hour of follow-up time per course is available with the respective tutor if you would like any further guidance or discussion after a course. Please contact IT Learning Centre to arrange.

# Housekeeping



- Exit the building if the fire alarm sounds
- Water cooler in the room and hot drinks available in registration area
- Toilets are located along the corridor
- The seats/monitors are adjustable
- If you have any concern or problem during the course please let me know

# Getting Started with R



**iT Centre**  
**Learning**

**iT**  
services



# Introductory Remarks



Understanding how to interact properly and carefully with R is vital to analyse your data correctly - mistakes can all too easily arise from oversights such as extracting the wrong portion of data or failing to spot data entry errors or inconsistencies before performing the analysis.

These can also be hard to detect in the output so a key habit to adopt is frequent checks of data objects and their contents, and making plots that will help spot issues or confirm if an output makes sense.

We will spend today's session introducing basic features of R and gaining some familiarity with using it as a programming language and to perform typical tasks associated with analysing data

Learning R is a long process but we hope this course will help you get started.

# The RStudio interface



An interactive and easy-to-use interface with many features that make working with R easier:

(<https://www.rstudio.com>)

The screenshot displays the RStudio interface with the following components:

- Script Editor:** Contains R code for generating random data and plotting it.

```
1 # Write your script here
2
3 x = rnorm(100, 1, 1)
4 y = rnorm(100, 1, 1)
5
6 p = plot(x, y)
7
8 print(p)
```
- Environment Pane:** Shows the current environment with variables `x` and `y` of type `num` (numeric), each with 100 elements.
- Console:** Shows the output of the R session, including a workspace load message and the execution of the script code, resulting in `NULL` for the `print(p)` command.
- Viewer Pane:** Displays a scatter plot of the generated data, with the x-axis labeled `x` and the y-axis labeled `y`.

# The RStudio interface



Console for entering  
R commands

A screenshot of the RStudio interface. The main window is titled 'my\_script.R' and contains the following R code:

```
1 # Write your script here
2
3 x = rnorm(100, 1, 1)
4 y = rnorm(100, 1, 1)
5
6 p = plot(x, y)
7
8 print(p)
```

The right-hand side of the interface shows the 'Environment' pane with the following data:

Variable	Value
p	NULL (empty)
x	num [1:100] 2.46 0.287 -0.193 1.568 0.914 ...
y	num [1:100] 0.543 0.696 2.439 2.238 3 ...

The bottom-left pane is the 'Console', which is highlighted with a green border. It displays the R startup message and the execution of the script:

```
Console ~/
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.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Workspace loaded from ~/.RData]
>
>
> source('~/.active-rstudio-document')
NULL
> source('~/.active-rstudio-document')
NULL
>
```

The bottom-right pane shows a scatter plot of the generated data, with 'x' on the horizontal axis and 'y' on the vertical axis. The plot displays a cloud of approximately 100 data points.

# The RStudio interface



Script, data  
tables etc



The screenshot displays the RStudio interface with the following components:

- Script Editor:** Contains the following R code:

```
1 # Write your script here
2
3 x = rnorm(100, 1, 1)
4 y = rnorm(100, 1, 1)
5
6 p = plot(x, y)
7
8 print(p)
```
- Environment Pane:** Shows the Global Environment with the following values:

Variable	Value
p	NULL (empty)
x	num [1:100] 2.46 0.287 -0.193 1.568 0.914 ...
y	num [1:100] 0.543 0.696 2.439 2.238 3 ...
- Console:** Shows the R startup message and the execution of the script:

```
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.
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
[Workspace loaded from ~/.RData]
>
>
> source('~/.active-rstudio-document')
NULL
> source('~/.active-rstudio-document')
NULL
>
```
- Plots Pane:** Displays a scatter plot of the generated data with x and y axes ranging from -1 to 4.

# The RStudio interface



The screenshot displays the RStudio interface with the following components:

- Script Editor:** Contains R code for generating random data and plotting it.
- Console:** Shows the execution output, including R version information and the results of the `source()` function.
- Environment Pane:** A yellow-bordered window showing the current workspace with variables `x` and `y` as numeric vectors.
- Viewer Pane:** Displays a scatter plot of the generated data.

Variable	Value
<code>p</code>	NULL (empty)
<code>x</code>	num [1:100] 2.46 0.287 -0.193 1.568 0.914 ...
<code>y</code>	num [1:100] 0.543 0.696 2.439 2.238 3 ...

Clickable list of objects in memory



# The RStudio interface

A screenshot of the RStudio software interface. The main window is titled 'RStudio' and shows a project named '(None)'. The interface is divided into several panes:

- Source Editor:** Contains an R script named 'my\_script.R' with the following code:

```
1 # Write your script here
2
3 x = rnorm(100, 1, 1)
4 y = rnorm(100, 1, 1)
5
6 p = plot(x, y)
7
8 print(p)
```
- Environment Pane:** Shows the 'Global Environment' with variables 'p', 'x', and 'y'.

Variable	Value
p	NULL (empty)
x	num [1:100] 2.46 0.287 -0.193 1.568 0.914 ...
y	num [1:100] 0.543 0.696 2.439 2.238 3 ...
- Console:** Shows the R startup message and the execution of the script. The output is 'NULL' for the 'print(p)' command.

```
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.
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
[Workspace loaded from ~/.RData]
>
>
> source('~/.active-rstudio-document')
NULL
> source('~/.active-rstudio-document')
NULL
>
```
- Plots Pane:** A window titled 'Plots' showing a scatter plot of 'y' versus 'x'. The plot contains approximately 100 data points, mostly clustered between x=0 and x=2, and y=-1 and y=3. The axes are labeled 'x' and 'y'.

←  
Plots, files,  
packages,  
help etc...