

# Chemometrics: Lab 1

This lab will introduce you to R and RStudio, which you will use to apply the knowledge learned in lectures to real datasets. R is the name of the programming language which is a free, open-source software environment for statistical computing and graphics. It is constantly being updated as new *packages* or libraries which perform different statistical techniques are created by users. RStudio is an *Integrated development environment* (IDE) and is an interface that makes R easier to use. To create reproducible data analyses we will also use R Markdown (which is part of RStudio). R Markdown allows us to have R code, R output and our narrative all in the same document.

## Getting started

There are two ways to use RStudio. The first is to install both R and RStudio onto your laptop. You can do this in your own time.

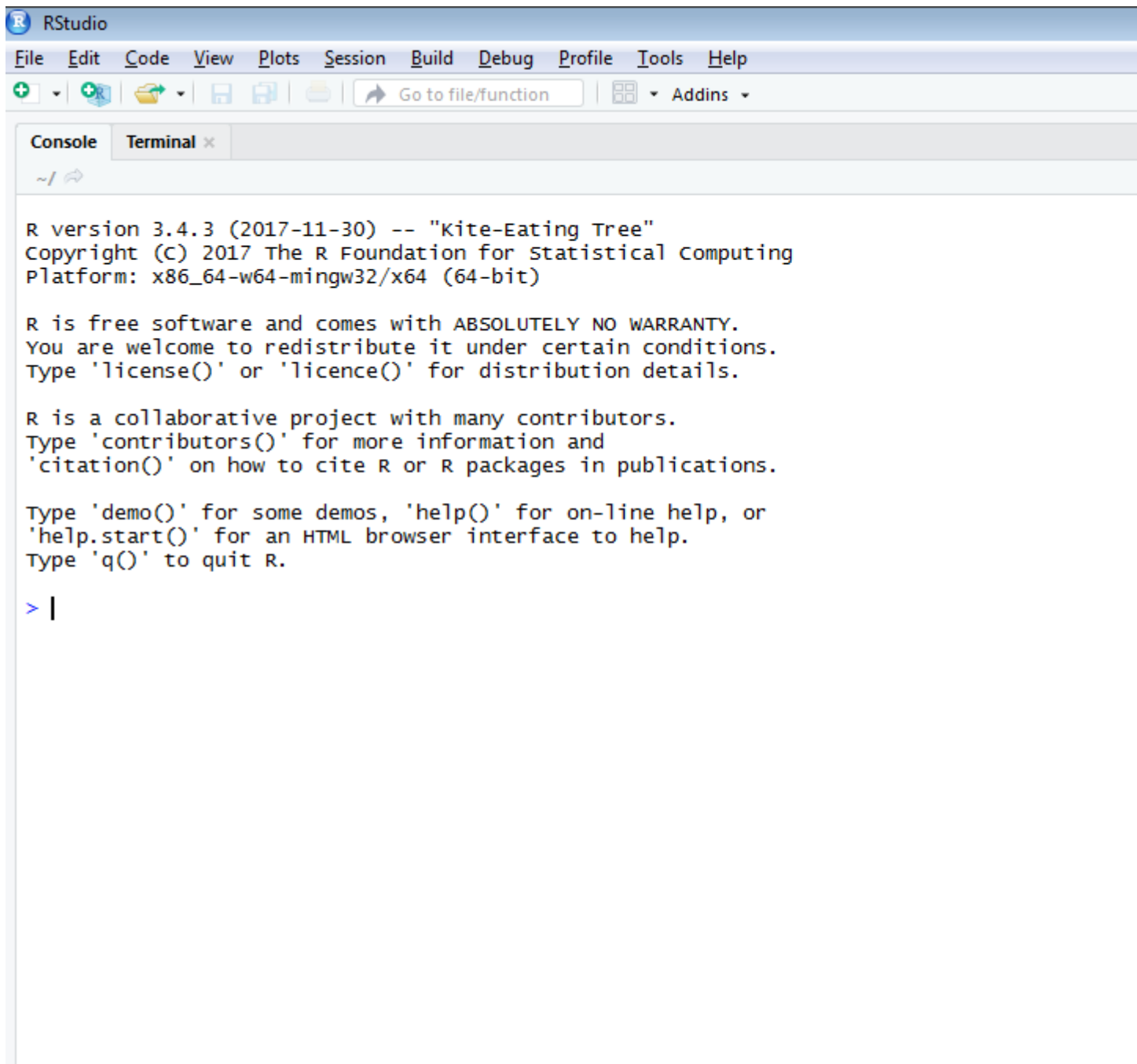
### 1. Install R and RStudio on your own computer

#### R and RStudio installation

- R: Install R from <https://cran.r-project.org>
- RStudio: Install RStudio from <https://www.rstudio.com>

#### RStudio layout

Open RStudio. You should see the following:



## 2. Use RStudio cloud

We will use RStudio cloud which does not require installation of R or RStudio. Instead it runs through your web browser. This means that you can use R and RStudio anywhere (even if they are not installed on the computer itself).

### Create an RStudio cloud account

First you need to create an account on <https://rstudio.cloud/> and log in. (**NOTE: Please ensure to use Google Chrome to run RStudio cloud.**)

## **RStudio cloud layout**

When you first log in you should see the following:


Spaces

 Your Workspace


 New Space


Learn


 Guide

 What's New


 Primers


 DataCamp Courses

 Cheat Sheets

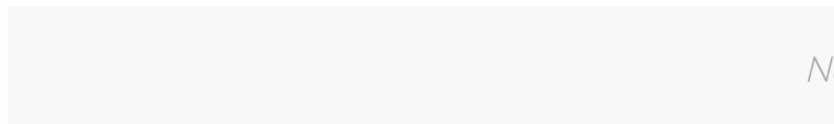
 Feedback and Questions

Info

 Terms and Conditions

 System Status

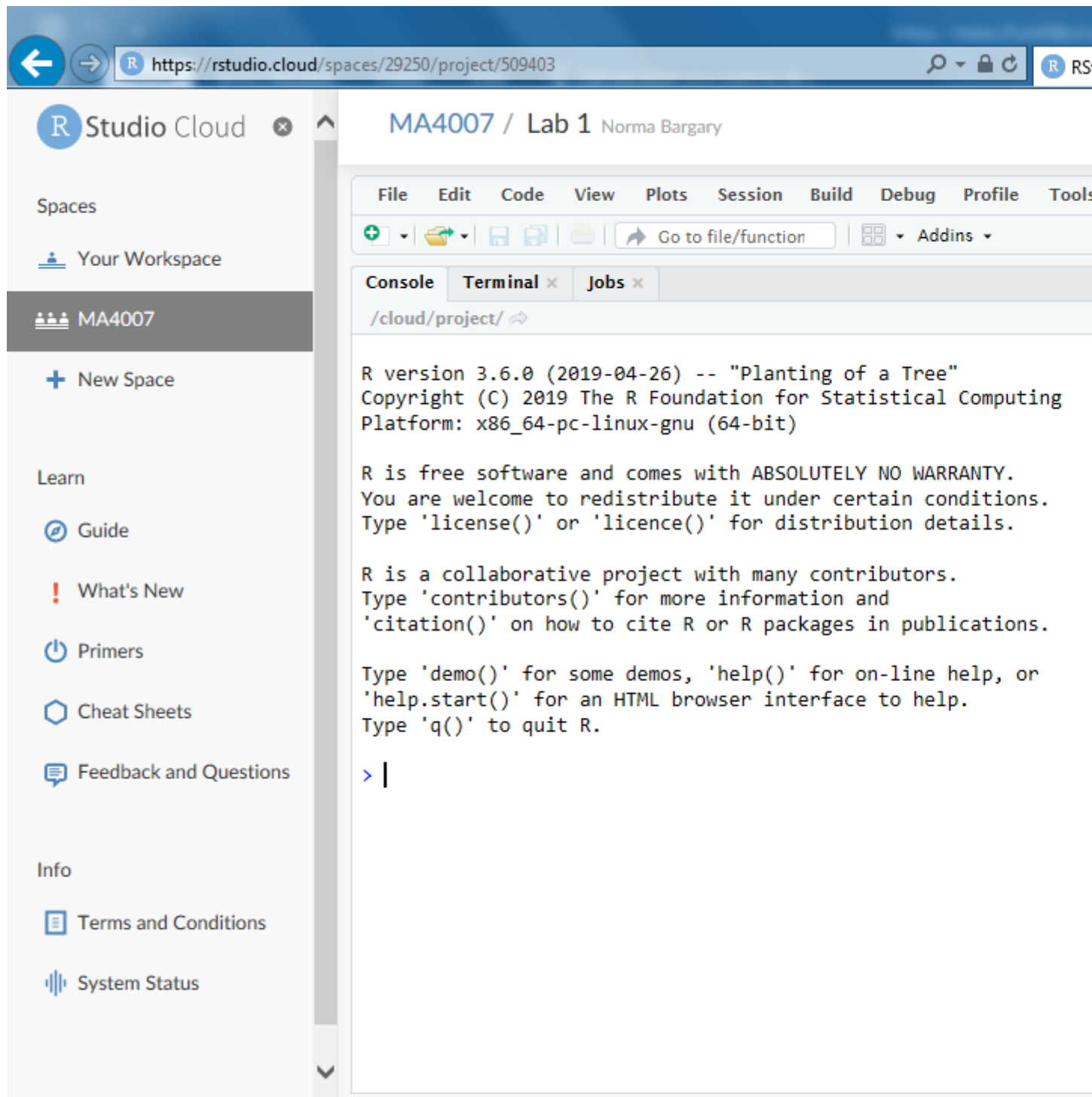
## Your Projects



## Accessing the MA4605 workspace

We will have a shared workspace for the MA4007 module. Once you have logged in to RStudio cloud, copy and paste the link from the text file on SULIS and paste it into your web browser. (You will only have to do this once.)

Click on MA4605 and then click Lab 1. You should see the following:



Click the **Save a Permanent Copy** link to create your own copy of the project with all the files, etc.

- The panel on left is the R Console. At the top of this panel, you will see the version of R that you're running. Below that information (which appears every time you start RStudio), you will see the prompt '>'. The prompt is a request for a command.
- The panel in the upper right contains your **workspace**, i.e. the folder you are working in. It also shows the history of the commands you have typed in.
- The panel in the bottom right has several tabs.
  1. The **Files** tab displays the files in your workspace.
  2. The **Plots** tab is where any plots that you create will appear.
  3. The **Packages** tab is where you can see what packages are available to you to you. This is also where you can download and install new packages with specific code to help with your particular analysis.
  4. The **Help** tab is where you can view the R help files.

## Using the R console

The easiest way to use R is to type commands directly into the console window. For example, type the following into R and press Enter.

```
2 + 2
```

You will see the following:

```
> 2 + 2
[1] 4
>
```

The prompt '>' indicates that R is ready for another command. The [1] indicates that this line contains the first element of the output.

If a command is incomplete at the end of a line, a '+' is displayed on subsequent lines until the command is syntactically complete.

For example, enter 2 +, then press Enter. A '+' sign will appear in the console. When you complete the command by typing a number, e.g. 2, the command is syntactically correct and the result is printed as before.

```
> 2 +
+ 2
[1] 4
>
```

You can also exit the command by hitting the Esc key!

## Reproducible (lab) reports

We will be using R Markdown to type up the lab report. This allows you to complete your lab entirely in RStudio (including code, output and comments) as well as ensuring reproducibility of your analysis and results. R Markdown files have the .Rmd file extension.

You can start a new Markdown file by clicking **File -> New File -> R Markdown -> Document**. However a template (MA4605\_Lab1\_Student.Rmd) has been provided for you to get you started. You should see it in the **Files** tab in the bottom right panel of your screen.

To complete the lab, open this file and type your brief answers and the R code (when necessary) in the spaces provided in the document.

Click on the **Knit** button and your document will appear in a new pop-up window.

## R packages

R is open-source, which means that users can create packages of R code that do specific things. We can use these packages for free. Some packages are part of the basic R installation. Others must be downloaded from the R website.

It is very easy to do this in RStudio. In the bottom right panel, click the **Packages** tab. To download and install a new package, click on **Install**. Type the name of the package(s) you want to download into the box under *Packages* and click **Install**.

Two packages we will use are

- `dplyr`: for data wrangling
- `ggplot2`: for data visualization

These are both contained in the `tidyverse` package. I have already installed this package for you. However, to access all of the functions and data sets in the package, it must be loaded into the workspace by typing the following into the console:

```
library(tidyverse)
```

Note that this line of code also appear at the top of your R Markdown document. We need to load packages both in the console workspace and in the R Markdown environment since these two environments work independently of each other. To run code from R Markdown in the console, simply click the green arrow next to the code in the R Markdown document.

To check what packages are currently loaded into the console workspace you can use

```
search()
```

REMEMBER: If you close your R session and re-open it at a later stage, packages must be re-loaded into the workspace.

## The whiteside data

The whiteside dataset is part of the `MASS` package in R. The `MASS` package must be loaded into R to access the data. (You should have already done this when loading the `tidyverse` package by clicking the green arrow next to the code in the `MA4605_Lab1_Student.Rmd` R Markdown file.)

The whiteside data contains the records of the weekly gas consumption (in 1000s of cubic feet) and average external temperature (in degrees C) at a house in south-east England for two heating seasons, one of 26 weeks before, and one of 30 weeks after cavity-wall insulation was installed. The object of the exercise was to assess the effect of the insulation on gas consumption.

Type the following code in the console

```
whiteside
```

You should see 3 columns of data. The first column contains a *factor* describing if the house was insulated or not. The second and third columns contain numbers which measure the temperature and corresponding gas consumption.

The whiteside data is stored in a *dataframe*. The dataframe contains 56 measurements on 3 variables.

You can see the dimensions of the dataframe using

```
dim(whiteside)
```

```
## [1] 56  3
```

There are 56 rows and 3 columns in this dataframe. You can see the names of these columns (i.e. variables) by typing

```
names(whiteside)
```

```
## [1] "Insul" "Temp"  "Gas"
```

The columns are called `Insul`, `Temp` and `Gas`.

## Some exploration

To access the data in a single column we can use

```
whiteside$Temp
```

This will only display the data in the `Temp` column. The dollar sign `$` says “go to the data frame that comes before me, and find the variable that comes after me”.

**Exercise 1:** What command would you use to extract just the data from the `Insul` column? Try it! You should enter your answer in your R Markdown document and press Knit.

One of R’s biggest strengths is its excellent graphic capabilities. To create a scatterplot of gas consumption versus temperature, type the following command into the console.

```
ggplot(aes(x=Gas, y=Temp), data=whiteside) + geom_point()
```

To add a label on the x-axis and a label on the y-axis use:

```
ggplot(aes(x=Gas, y=Temp), data=whiteside) + geom_point() + ylab("Temperature") + xlab("Gas consumption")
```

It can also be useful to colour the points by whether the house was insulated or not.

**Exercise 2:** Create a plot of gas consumption versus temperature but colour the points by whether the value was measured before or after insulation was added. (HINT: add `colour=Insul` to the `aes()` argument.) Is there an apparent difference in the trend in the gas consumption for the two insulation types? Type your answers in your R Markdown document.

We can create subsets of dataframes very easily using the **pipe** `%>%`. To select the rows of the `whiteside` dataset corresponding to the data measured before insulation was added use:

```
whiteside %>%  
  filter(Insul == "Before") %>%  
  select(Gas, Temp)
```

The `filter` command tells R to keep only the rows where `Insul` has the value “Before”. The `select` command tells R to keep only the `Gas` and `Temp` columns. You can store the smaller dataset in a new dataframe using:

```
whiteside.before = whiteside %>%  
  filter(Insul == "Before") %>%  
  select(Gas, Temp)
```

**Exercise 3:** Use the `filter` and `select` commands to select the rows in the `whiteside` dataset corresponding to the data measure after insulation was added. Store the result into a dataframe called `whiteside.after`. Type your answers in your R Markdown document.

Other useful commands include the `group_by` and `summarize` commands. To calculate the mean gas consumption before and after insulation was added use:

```
whiteside %>%  
  group_by(Insul) %>%  
  summarize(mean_temp = mean(Temp))
```

**Exercise 4:** Calculate the mean gas consumption before and after insulation was added. Comment on the difference between the values. Type your R code and comments in your R Markdown document.



## Getting help

R has a built-in help facility. For example to get help on the `sqrt` command type

```
? sqrt
```

in the console. The help file will appear in the bottom right panel in RStudio. Using Google is also great!

There are some useful cheatsheets for R Markdown, dplyr and ggplot2 available at the following links:

- [R Markdown cheatsheet](#)
- [dplyr cheatsheet](#)
- [ggplot2 cheatsheet](#)

## Summary

This lab provided an introduction to RStudio and R Markdown. You will continue to build on your skills as the course progresses. There are many websites available to learn more about R, RStudio and R Markdown. I would encourage you to look at some of these!