Online Appendix 1.A: Introduction to R/RStudio, Part I

Download **R** and the helper software **RStudio**. You must have Internet access and administrative privileges on your computer and it is assumed that you are reasonable familiarity with working with software in your preferred computing environment. The code in this book should run on almost any computing environment: (e.g., Windows, OSX, Linux, etc.). That is one reason for choosing it. The other reason is that **R** is an open source platform, so one has access to the *entire* code; nothing is hidden; with sufficient knowledge one can modify and improve the code; of so, the author would appreciate hearing from the reader so that he may incorporate the improvements in future versions of the book/code.

Online Appendix 1.A.1: Windows computer

On a Windows PC do a Google search for "**Install R Windows**", and one should be able to find the appropriate link. Download the software and follow the installation instructions. One needs to also install **RStudio** after installing **R**. This too is free software; think of it as a user interface for **R** (**R** already has a basic user interface; **RStudio** adds many powerful features which make life a lot easier). Do a Google search for "**Install RStudio Windows**" and find the link. Download the software and follow the installation instructions.

Online Appendix 1.A.2: OS X (MAC) machine

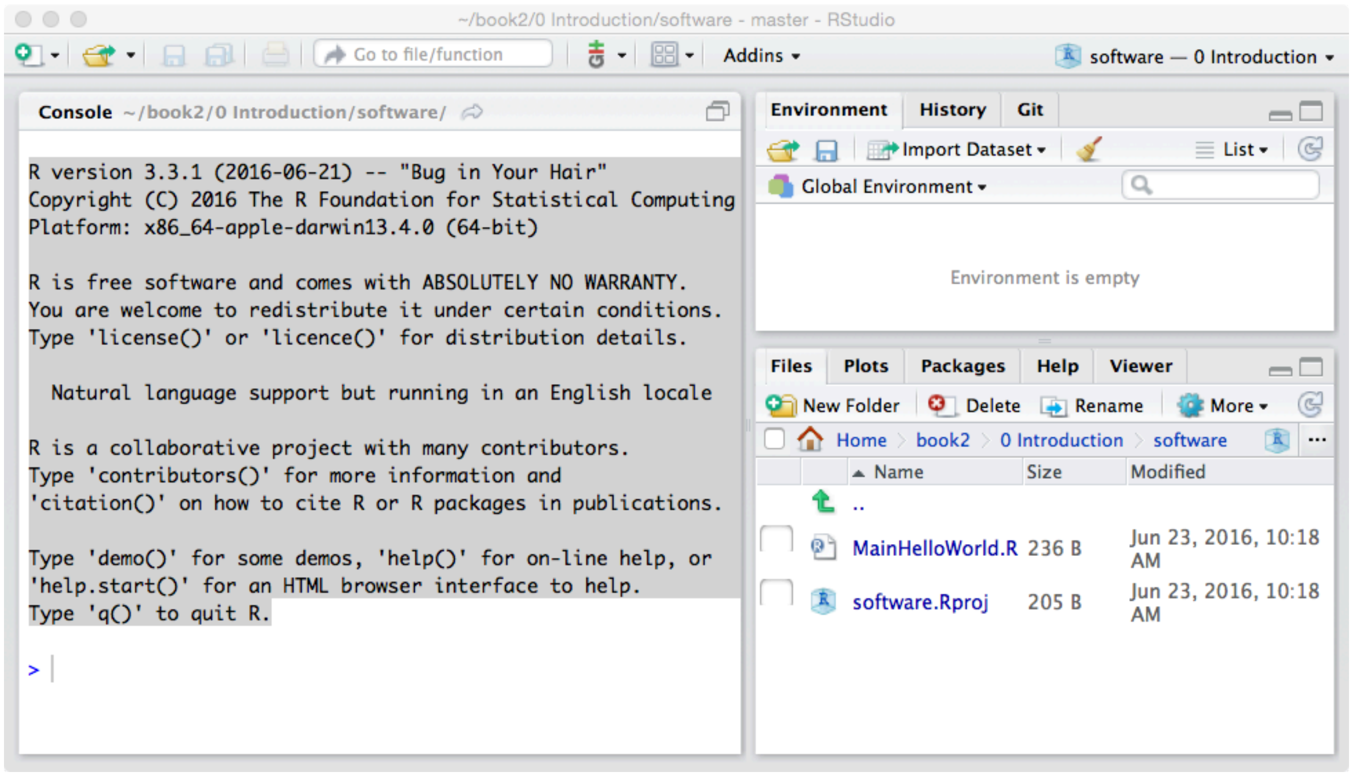
(This is the author's platform.) Search for "**Install R MAC**". Click on the downloaded software and follow the standard OS X installation procedure. After the **R** installation is complete install **RStudio** for the Macintosh operating system. (Do a search for "**Install RStudio Mac**" and find the link. Download the software and follow the installation instructions.)

Online Appendix 1.A.3: Book software

You should have download three files: **Ch01_Text.pdf**, **Ch01_OnlineAppendices.pdf** (this file) and **software.zip**. Copy the files to a suitable location on your computer. In the author's computer the location is **Desktop/book2**). Unzip any compressed files. Feel free to play with the code. Do not worry about "messing it up": if one does "mess up" one can always download the files again. *The author cannot emphasize strongly enough that the only way to really learn is by doing.*

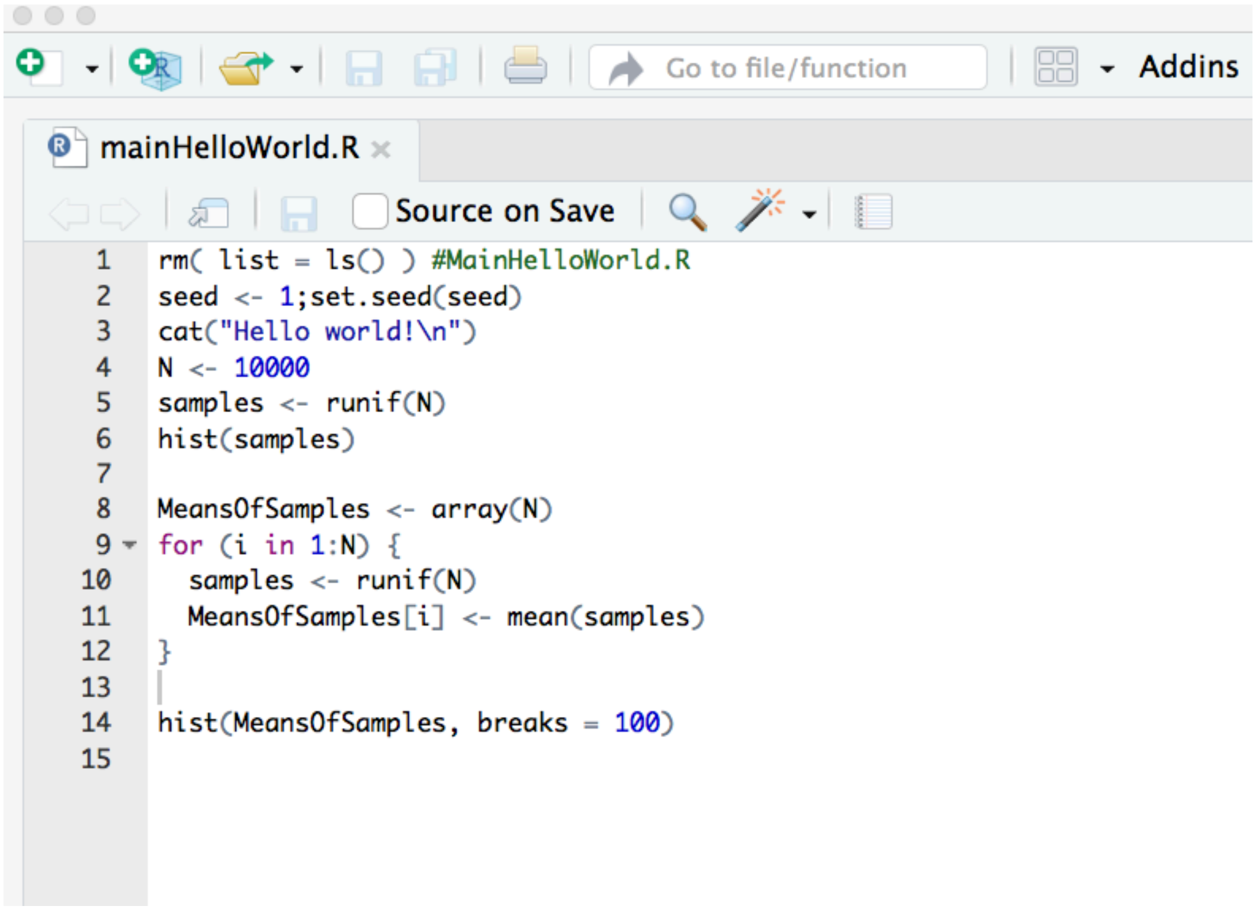
Online Appendix 1.A.4: A simple example using R/RStudio

This section is intended as a gentle introduction to the software. The screen-shots that follow are for the author's iMAC installation (the author's operating system is **OS X Yosemite 10.10.5**). If your screen shots look different these are due to differences in platforms (Windows, OSX, Linux, etc.) and version numbers. With a little faith and inquisitiveness one should be able to follow the brief tutorial below. Follow these directions carefully: find the folder corresponding to the chapter heading of this chapter and open it; in the author's computer it is **Desktop\book2**. In the folder named **software** you should see a file named **software.Rproj**. The extension means that this is an **R** project file. A project file organizes all the files in your project for easy viewing, editing and compilation; one does not need to know its contents. Open this file by clicking (or double clicking, depending on your computer settings) its icon. Your screen should resemble Fig. 1.A.1.


~/book2/0 Introduction/software - master - RStudio
Go to file/function
Addins
software — 0 Introduction
Console ~/book2/0 Introduction/software/
Environment
History
Git
Import Dataset
List
Global Environment
Environment is empty
Files
Plots
Packages
Help
Viewer
New Folder
Delete
Rename
More
Home
book2
0 Introduction
software
Name
Size
Modified
MainHelloWorld.R 236 B Jun 23, 2016, 10:18 AM
software.Rproj 205 B Jun 23, 2016, 10:18 AM

```
R version 3.3.1 (2016-06-21) -- "Bug in Your Hair"
Copyright (C) 2016 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin13.4.0 (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.

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.

> 
```

Fig. 1.A.1.a: Screen shot upon clicking **`software.Rproj`**. Note organization into 3 panels: **`Console`**, **`Environment`** etc. and **`Files`** etc. One may see 4 panels if any of the source code files - i.e., those ending in **`.R`**. - are open; simply close all such files (see below) to get a display similar, but perhaps not identical to this figure.



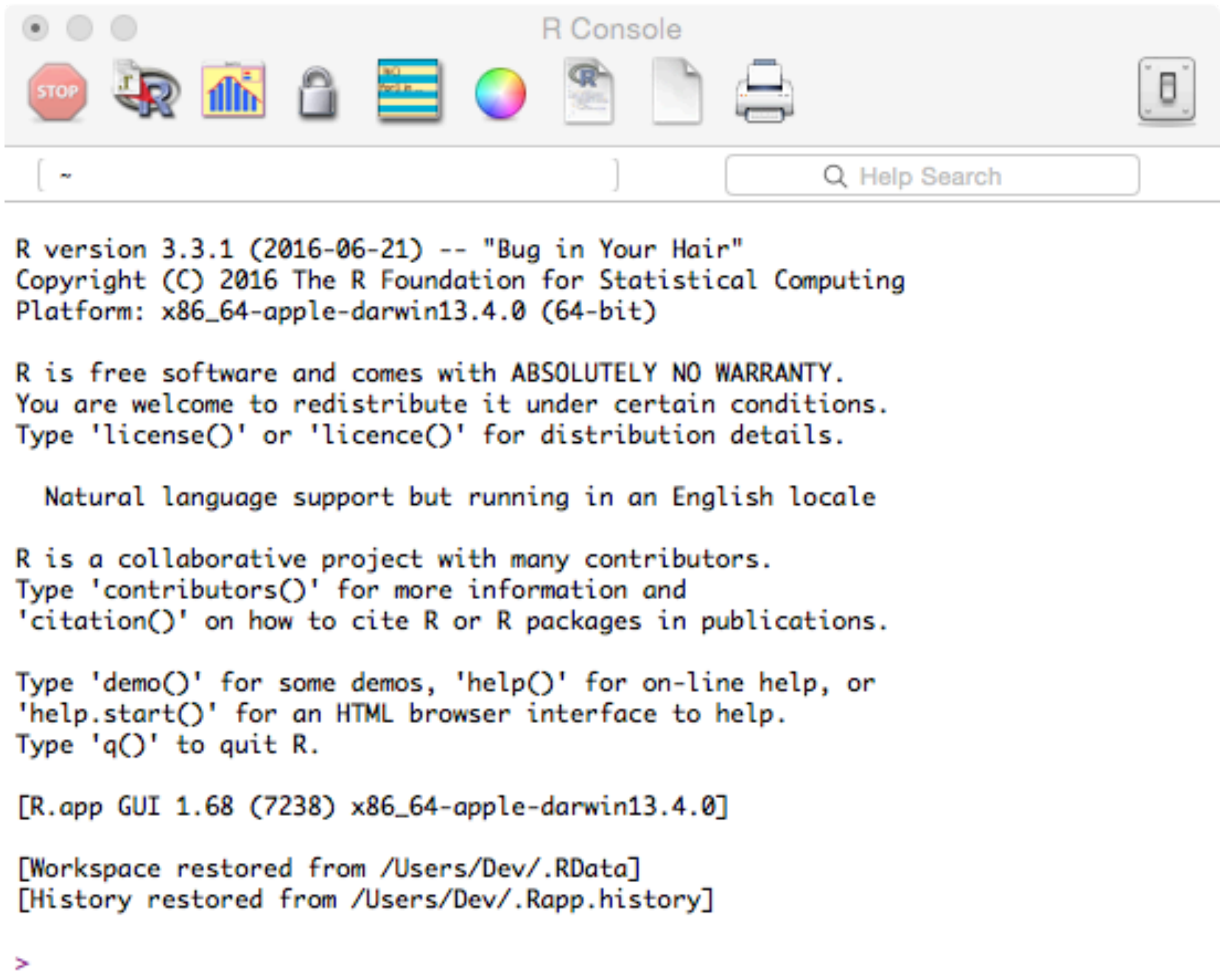
```r
rm( list = ls() ) #MainHelloWorld.R
seed <- 1;set.seed(seed)
cat("Hello world!\n")
N <- 10000
samples <- runif(N)
hist(samples)

MeansOfSamples <- array(N)
for (i in 1:N) {
  samples <- runif(N)
  MeansOfSamples[i] <- mean(samples)
}

hist(MeansOfSamples, breaks = 100)

```

Fig. 1.A.1.b: The x button next to the file name closes the window.

Notice the organization of the display into 3 major panels (one may see 4 panels if any of the source code files - i.e., those ending in **`.R`**. - are open; simply close (click the x button shown in Fig. 1.A.1.b) all such files to get a display similar, but perhaps not identical to Fig. 1.A.1). The left panel is labeled **`Console`** and the right consists of two panels: the top-right panel consists of two sub-panels labeled **`Environment, History`** and **`Git`**, and the **`Environment`** sub-panel is in focus (notice the brighter background of the text in **`Environment`** compared to **`History`** or **`Git`**). The bottom-right panel consists of five sub-panels labeled **`Files`**, **`Plots`**, **`Packages, Help`** and **`Viewer`** and the **`Files`** sub-panel is in focus. The actual number of menu items may have changed by the time this book is published, because the developers of **`RStudio`** and **`R`** are hard at work improving the software; ignore the extra ones for now.

The left window, termed **`Console`**, shows the "**`>`**" symbol. This is an invitation to the user to enter commands, so this symbol is the *command prompt* for **`R`**. Sometimes, when debugging a program, one might see this symbol preceded by a **`Browse[2]`**, or **`Browse[3]`**, as in , **`Browse[2]>`**. This is a left-over from previous versions of **`RStudio`**, when debugging was initiated by a **`browse()`** function. Debugging capabilities have improved dramatically in the author's five-year experience with **`R`**.

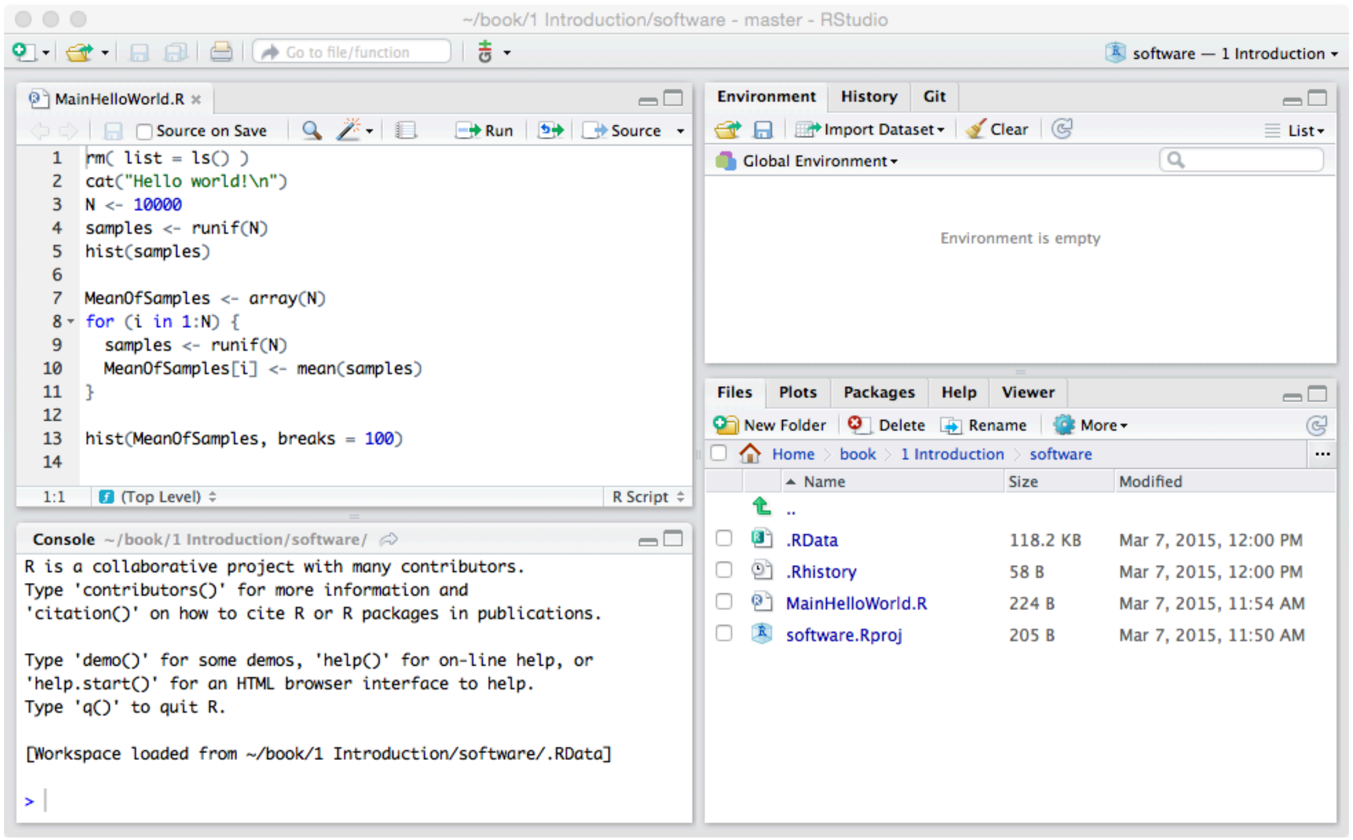
The **`Console`** panel is currently simply showing the output screen one gets from starting **`R`**. Try it! Locate **`R`** using **`Spotlight`** (the magnifying glass symbol in the very top-right of the **`OS X`** display) and open it (i.e., click on it). The **`R`** application is in the **`Applications`** folder. This is what the author sees, Fig. 1.A.2. [On **`Windows`** go to **`Program`** and find the **`R`** executable file.]



```
R version 3.3.1 (2016-06-21) -- "Bug in Your Hair"
Copyright (C) 2016 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin13.4.0 (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.

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.

[R.app GUI 1.68 (7238) x86_64-apple-darwin13.4.0]

[Workspace restored from /Users/Dev/.RData]
[History restored from /Users/Dev/.Rapp.history]

>
```

Fig. 1.A.2: Screen-shot obtained by opening the **R** application (i.e., independent of **RStudio**). Note that it is practically identical to the shaded text in Fig. 1.A.1. Think of **RStudio** as a helper application that makes it easier to work with **R**.

Excepting for the last few lines, the two screens (the left panel of Fig. 1.A.1 and Fig. 1.A.2) are identical. Now quit **R** - in this book we will always use **RStudio** and never open **R** directly. Returning to Fig. 1.A.1, the **Files** sub-panel contains all the files in your project. Take some time to examine this screen. Below the **Files** sub-panel one sees **New Folder**, **Delete**, **Rename** and **More**. The next row shows the complete directory path: **Home/book2/0 Introduction/software**. The next row is labeled **Name**, **Size** and **Modified**, clicking on any of which sorts the files according to the chosen label. Click on the file named **mainHelloWorld.R** in this panel. *In this book any file name that starts with* ***main*** *contains directly executable code; all the other files with the* ***.R*** *extension are functions (called subroutines in some programming languages) that may be called by directly executable code.* This is what the author sees, Fig. 1.A.3:



```
rm( list = ls() )
cat("Hello world!\n")
N <- 10000
samples <- runif(N)
hist(samples)

MeanOfSamples <- array(N)
for (i in 1:N) {
  samples <- runif(N)
  MeanOfSamples[i] <- mean(samples)
}

hist(MeanOfSamples, breaks = 100)
```

```
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 ~/book/1 Introduction/software/.RData]

>
```

Name	Size	Modified
..		
.RData	118.2 KB	Mar 7, 2015, 12:00 PM
.Rhistory	58 B	Mar 7, 2015, 12:00 PM
MainHelloWorld.R	224 B	Mar 7, 2015, 11:54 AM
software.Rproj	205 B	Mar 7, 2015, 11:50 AM

Fig. 1.A.3: screen-shot obtained by opening the file **MainHelloWorld.R**. Note the additional window showing the code file. Note the locations of the **Run** and **Source** buttons. The former button executes the line the cursor happens to be on, or a highlighted section of code. The latter executes the entire code in the file.

Notice the additional **MainHelloWorld.R** window occupying the top-left **RStudio** window, in addition to the **Console** window which now occupies the bottom-left **RStudio** window; the additional window organizes your *source* code files (these are files with the **.R** extension) into one window for convenient viewing and editing. Whenever one creates new code for a project one should save it to the directory indicated on the bottom-right panel. To distinguish it easily from regular text, all code shown in this book will look like the one shown below (i.e., using the same font type and size, background, shading and borders). A listing of the file **MainHelloWorld.R** follows:

Online Appendix 1.A.2: Code Listing

```
rm( list = ls() ) #MainHelloWorld.R
seed <- 1;set.seed(seed)
cat("Hello world!\n")
N <- 10000
samples <- runif(N)
hist(samples)

MeansOfSamples <- array(N)
for (i in 1:N) {
  samples <- runif(N)
  MeansOfSamples[i] <- mean(samples)
}

hist(MeansOfSamples, breaks = 100)
```

First, notice that the existence of helpful line numbers in Fig. 1.A.3 (bottom left panel); these are generated by **RStudio**. The hash symbol (**#**) is used in **R** to insert comments: the compiler[1] ignores anything appearing on a line after the hash symbol. This is a convenient way to insert explanations of what we are doing and/or why we are doing it this way. Ignoring the description that follows the hash symbol, the command **rm(list = ls())** on line 1 says (in effect) that we are deleting ("removing") all existing variables so as to start with a "clean slate". [It will be shown later how one uses the **Help** system to find out exactly what the code does.] Blank lines of code, which are ignored by the compiler, are inserted for improved readability. Multiple commands, separated by semi-colons (**;**) can be put on one line to preserve "vertical real estate". **R** does not have a continuation symbol for long lines - we will see later how to split up long lines across several lines without using any special continuation symbol. Also, **R** commands are case sensitive: as one example, the command **Rm(list = ls())**will not work [try it! type **Rm(list = ls())**into the **Console** window (bottom-left panel in Fig. 1.A.3) and press the **return/enter** key]:

Online Appendix 1.A.2.1: Code Snippets

```
> Rm(list = ls())
Error: could not find function "Rm"}
hist(MeansOfSamples, breaks = 100)
```

Line 2 also uses the **set.seed()** function to set the seed of the random number generator to unity, to force repeatable "random" sequences[2]. Line 3 uses the **cat()** function, which stands for *concatenate and print*. In the present example the **cat()** function simply prints the string variable **"Hello world!"**. Line 4 initializes the variable **N** to 10,000. Unlike some other programming languages (e.g., **IDL** and **C**), **R** uses the **<-** symbol to *assign the right hand side to the left hand side* (keyboard shortcut: use **Alt-** to enter this quickly in the **Console** window; the shortcut even inserts a "readability" space before the value that follows). Line 5 obtains 10,000 samples from the standard random uniform distribution (i.e., uniformly distributed in the closed range 0 to 1). The function used is called **runif(N)** for "**N** random samples from the random uniform distribution". The 10,000 samples are assigned to the variable **samples**. Line 6 uses the **hist()** function to display a histogram of these samples. Line 8 allocates an **array** variable **MeansOfSamples** to hold 10,000 numbers. The **for**-loop beginning on line 9 and ending on line 12 means, effectively, "repeatedly execute the statements in the enclosed curly brackets for all values of the integer variable **i** in the range 1 to 10,000". On each pass through this loop, at line 10, a fresh sample of **N** random values from the uniform distribution are generated, which overwrites the existing variable **samples**, and line 11 uses the **mean()** function to calculate the average of these samples, which is saved to **MeansOfSamples[i]**. The square brackets denote array indexing: the first time through the loop the average is saved to **MeansOfSamples[1]**, the second time it is saved to **MeansOfSamples[2]**, and so on. Line 14 displays a histogram of the **MeansOfSamples** array. Click on the **Source** button on the top-right corner of the source-file window; this causes all lines in the current file to be executed sequentially. You should get the following output in the **Console** window:

Online Appendix 1.A.2.2: Code Output

```
> source('~/book/1 Introduction/software/MainHelloWorld.R')
Hello world!
```

The reassuring message came from line 3. Your **RStudio** window should look like that shown in Fig. 1.A.4:

[1] A compiler is software the converts the more-or-less English-like instructions that make some sense to the programmer to sequences of binary instructions that the computer understands, but which would be gibberish to the programmer. When one executes a program, the computer sequentially executes the binary instructions.
[2] Allowing the period (**.**) as part of a variable name causes the author to cringe; but in life one cannot have everything ones way.

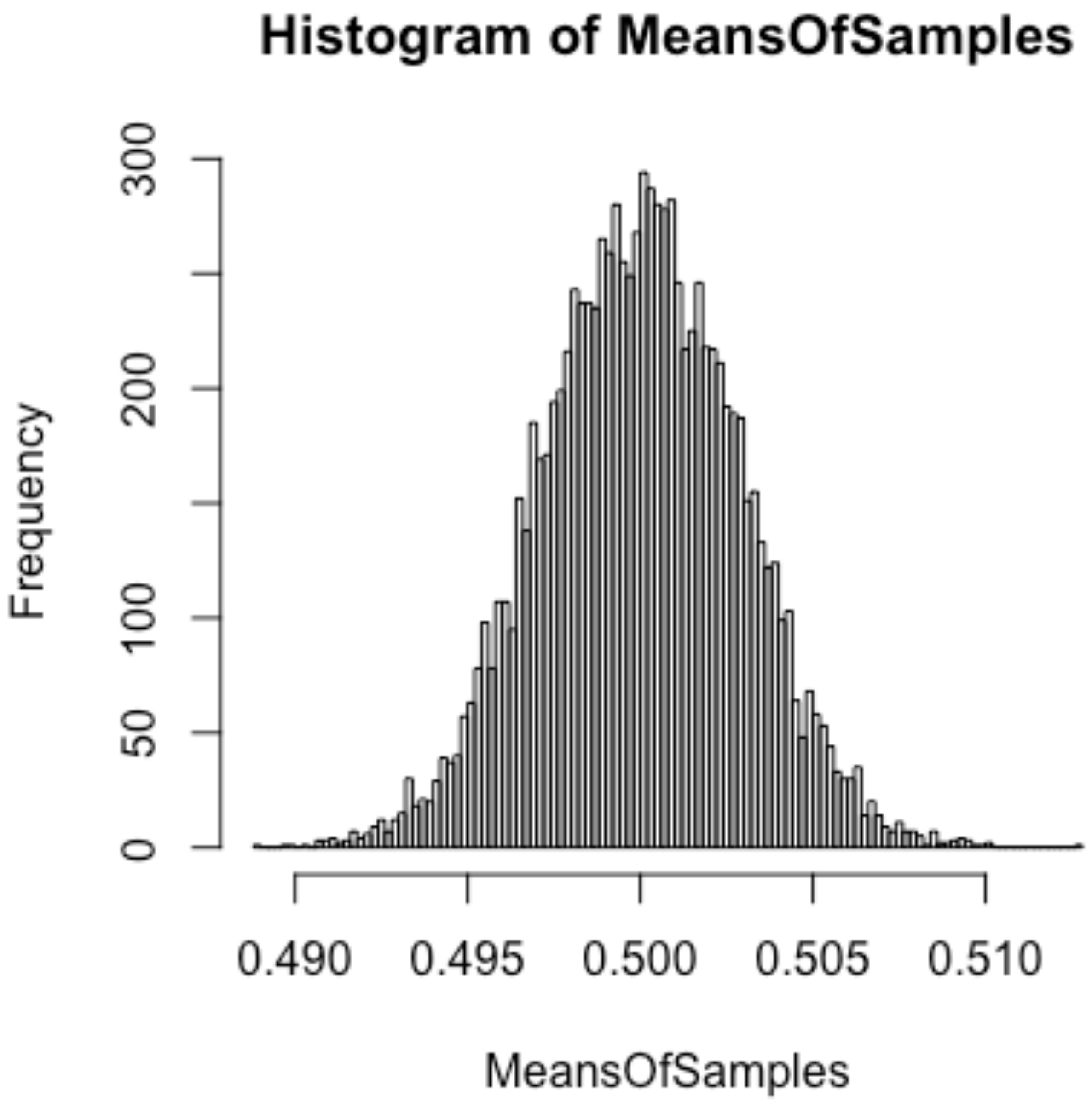
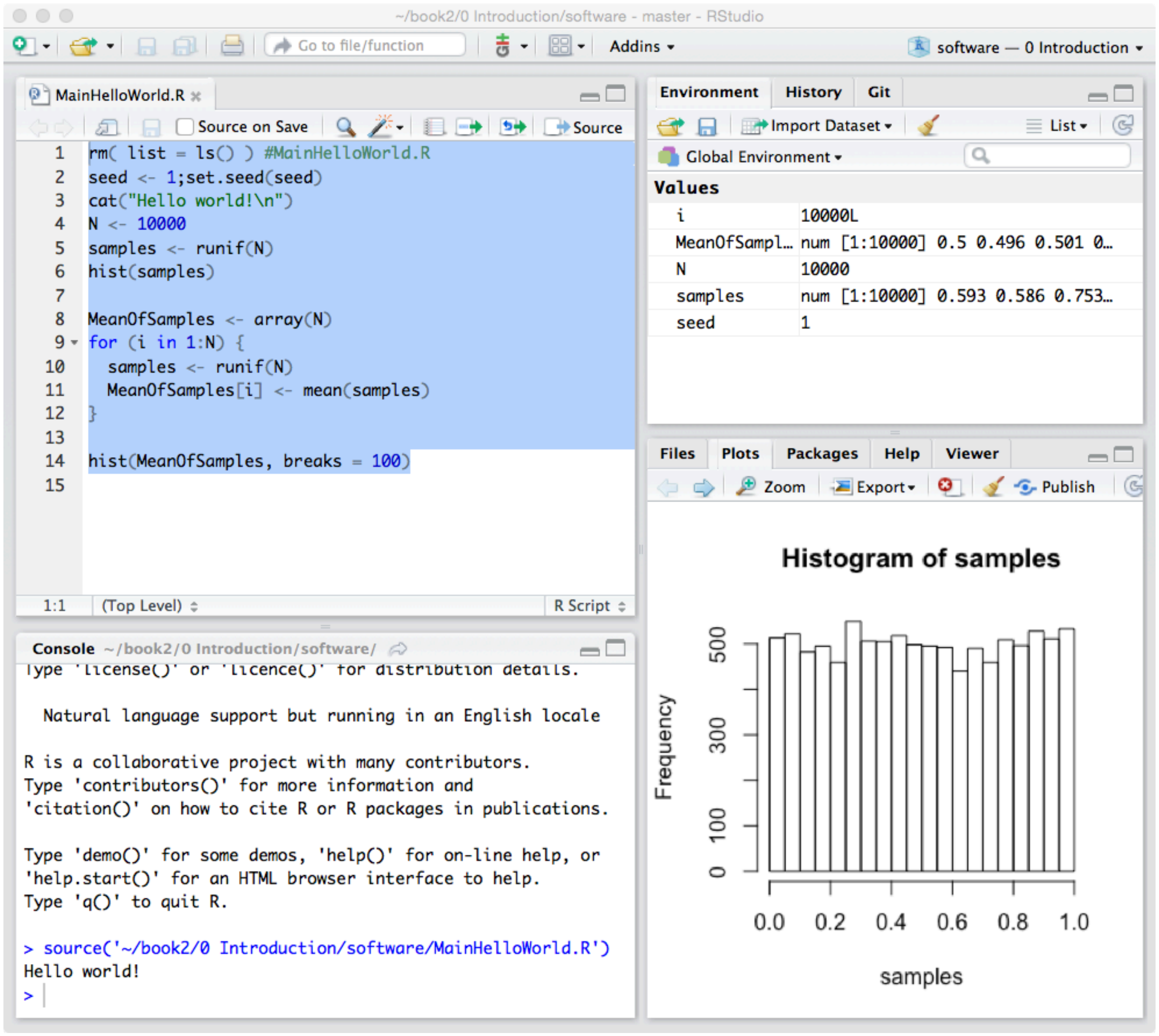


Fig. 1.A.4: Screen-shot obtained by sourcing the file **mainHelloWorld.R**. Note the Histogram plot, resulting from line 14.

Notice that the lower-right window is focused on the **Plots** tab, and a histogram of the variable **MeansOfSamples** is shown, which came from line 14, that looks like a normal distribution centered at 0.5, a result following from the central limit theorem of statistics, which states that the mean of a sufficiently large number of independent random samples, will be approximately normally distributed, regardless of the underlying distribution (see any standard statistics book).

Notice that the top-right window is focused on the **Environment** tab, which is filled in with values (later we will demonstrate many uses for this window, basically a convenient way of examining variables without having to print them out). For example, one sees that **i** is 10,000 (the last value used in the **for** loop), **MeansOfSamples** is an array with 10,000 values the first few are 0.5, 0.496, 0.501, 0.504, etc. all values being near 0.5. The variable **`N`** is 10,000 (from line 4) and finally the variable **`samples`** is an array with 10,000 values, the first few being 0.593, 0.586, 0.753, 0.964, etc., which correspond to the last execution of line 10. If one looks carefully at the top-left end of the **`Plots`** window one sees that the blue *left* arrow button is active, indicating that there is another plot *hiding* behind the one being shown; clicking on this button Clicking on the blue *left* arrow should yield Fig. 1.A.5.



```
1   rm( list = ls() ) #MainHelloWorld.R
2   seed <- 1;set.seed(seed)
3   cat("Hello world!\n")
4   N <- 10000
5   samples <- runif(N)
6   hist(samples)
7
8   MeanOfSamples <- array(N)
9   for (i in 1:N) {
10    samples <- runif(N)
11    MeanOfSamples[i] <- mean(samples)
12  }
13
14  hist(MeanOfSamples, breaks = 100)
15
```

```
Values
i               10000L
MeanOfSampl…    num [1:10000] 0.5 0.496 0.501 0…
N               10000
samples         num [1:10000] 0.593 0.586 0.753…
seed            1
```

```
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.

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.

> source('~/book2/0 Introduction/software/MainHelloWorld.R')
Hello world!
>
```

Fig. 1.A.5: screen-shot obtained by clicking on the left arrow button in the plots window, which shows the plot generated at line 6.

The histogram, which is due to line 6, is nearly flat, as one expects with a uniform distribution. This time the *right* blue arrow is active, clicking on which will bring up the previous figure. So one can switch between all the plots (not just two) generated by one's code.

Click on the **`Clear All`** button (with the broom symbol next to it) to delete all plots.
That ends this brief tutorial.

Last modified:
9/29/18 11:53 AM