Introduction to R Software
Swayam Prabha

Lecture 3

Help, Demonstration, and Examples

Shalabh

Department of Mathematics and Statistics

Indian Institute of Technology Kanpur
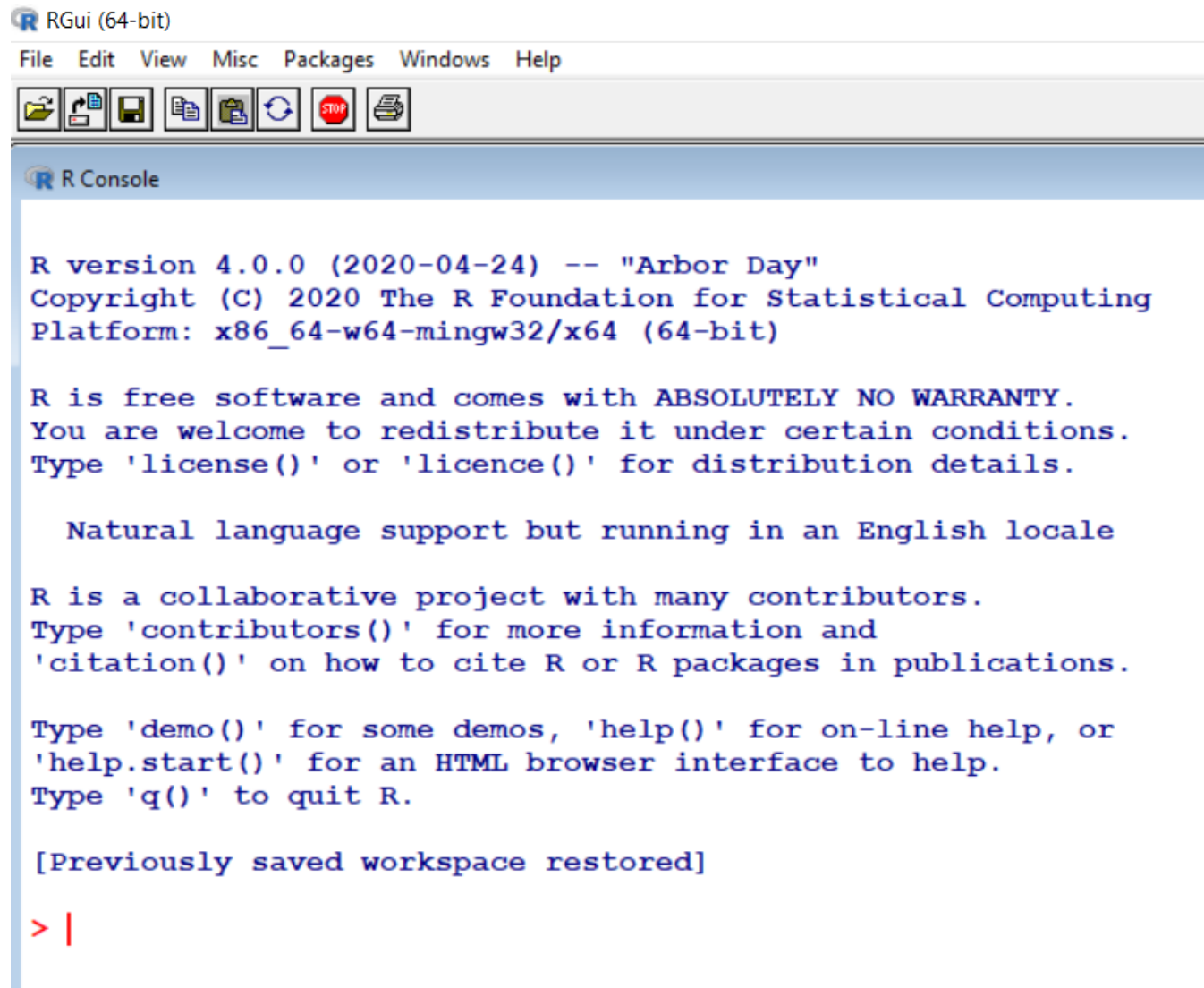
Slides can be downloaded from

http://home.iitk.ac.in/~shalab/sp

Starting with R

To start R, double click on the icon .

Then we get the following Gui (Graphic user interface) window screen

Getting Help in R

This can be done in one of the following ways:

1) **Start R software and click the help button in the toolbar of the R Gui (Graphic user interface) window.**
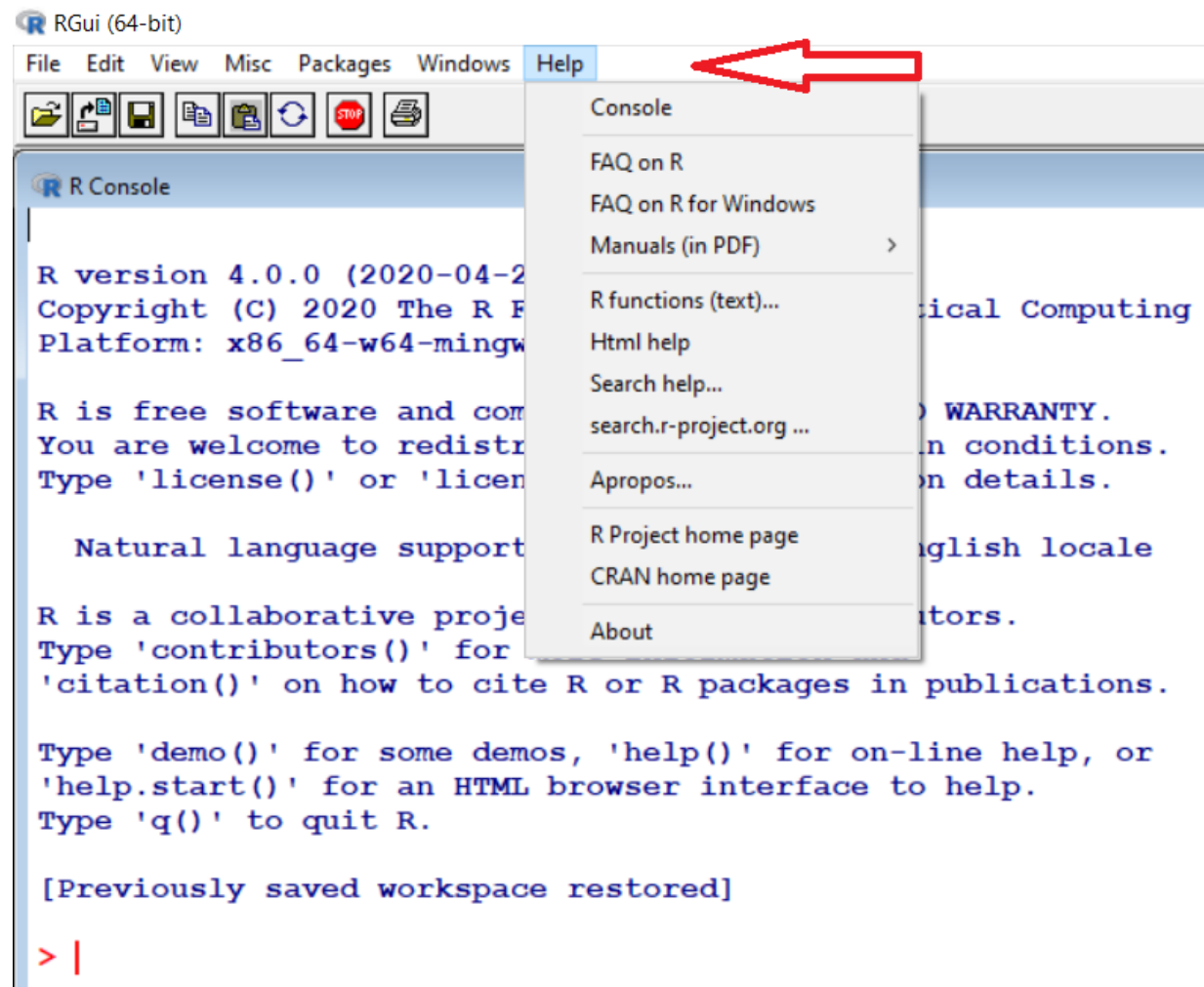
Getting Help in R

2. Search for help in Google `www.google.com`

3. If you need help with a function, then type question mark followed by the name of the function. For example, `?read.table` to get help for function `read.table`.

```
R Console

R version 4.0.0 (2020-04-24) -- "Arbor Day"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (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.

[Previously saved workspace restored]

> ?read.table
starting httpd help server ... done
> |
```

read.table {utils}

Description

Reads a file in table format and creates a data frame from it, with cases corresponding to lines and variables to fields in the file.

Usage

```
read.table(file, header = FALSE, sep = "", quote = "\"'",
           dec = ".", row.names, col.names,
           as.is = !stringsAsFactors,
           na.strings = "NA", colClasses = NA, nrows = -1,
           skip = 0, check.names = TRUE, fill = !blank.lines.skip,
           strip.white = FALSE, blank.lines.skip = TRUE,
           comment.char = "#",
           allowEscapes = FALSE, flush = FALSE,
           stringsAsFactors = default.stringsAsFactors(),
           fileEncoding = "", encoding = "unknown", text)

read.csv(file, header = TRUE, sep = ",", quote = "\"",
         dec = ".", fill = TRUE, comment.char = "", ...)

read.csv2(file, header = TRUE, sep = ";", quote = "\"",
          dec = ",", fill = TRUE, comment.char = "", ...)

read.delim(file, header = TRUE, sep = "\t", quote = "\"",
           dec = ".", fill = TRUE, comment.char = "", ...)

read.delim2(file, header = TRUE, sep = "\t", quote = "\"",
            dec = ",", fill = TRUE, comment.char = "", ...)
```

Arguments

- •
- •
- •
- •

…continued

Arguments

file
the name of the file which the data are to be read from. Each row of the table appears as one line of the file. If it does not contain an *absolute* path, the file name is *relative* to the current working directory, getwd(). Tilde-expansion is performed where supported. This can be a compressed file (see file).

Alternatively, file can be a readable text-mode connection (which will be opened for reading if necessary, and if so closed (and hence destroyed) at the end of the function call). (If stdin() is used, the prompts for lines may be somewhat confusing. Terminate input with a blank line or an EOF signal, Ctrl-D on Unix and Ctrl-Z on Windows. Any pushback on stdin() will be cleared before return.)

file can also be a complete URL. (For the supported URL schemes, see the 'URLs' section of the help for url.)

header
a logical value indicating whether the file contains the names of the variables as its first line. If missing, the value is determined from the file format: header is set to TRUE if and only if the first row contains one fewer field than the number of columns.

sep
the field separator character. Values on each line of the file are separated by this character. If sep = "" (the default for read.table) the separator is 'white space', that is one or more spaces, tabs, newlines or carriage returns.

quote
the set of quoting characters. To disable quoting altogether, use quote = "". See scan for the behaviour on quotes embedded in quotes. Quoting is only considered for columns read as character, which is all of them unless colClasses is specified.

dec
the character used in the file for decimal points.

row.names
a vector of row names. This can be a vector giving the actual row names, or a single number giving the column of the table which contains the row names, or character string giving the name of the table column containing the row names.

If there is a header and the first row contains one fewer field than the number of columns, the first column in the input is used for the row names. Otherwise if row.names is missing, the rows are numbered.

Using row.names = NULL forces row numbering. Missing or NULL row.names generate row names that are considered to be 'automatic' (and not preserved by as.matrix).

col.names
a vector of optional names for the variables. The default is to use "V" followed by the column number.

as.is
the default behavior of read.table is to convert character variables (which are not converted to logical, numeric or complex) to factors. The variable as.is controls the conversion of columns not otherwise specified by colClasses. Its value is either a vector of logicals (values are recycled if necessary), or a vector of numeric or character indices which specify which columns should not be converted to factors.

Note: to suppress all conversions including those of numeric columns, set colClasses = "character".

All minor details and explanations of all arguments are given.

How to clear the screen in R

Press ctrl + L to clear the screen of R console.

and

Getting Help in R
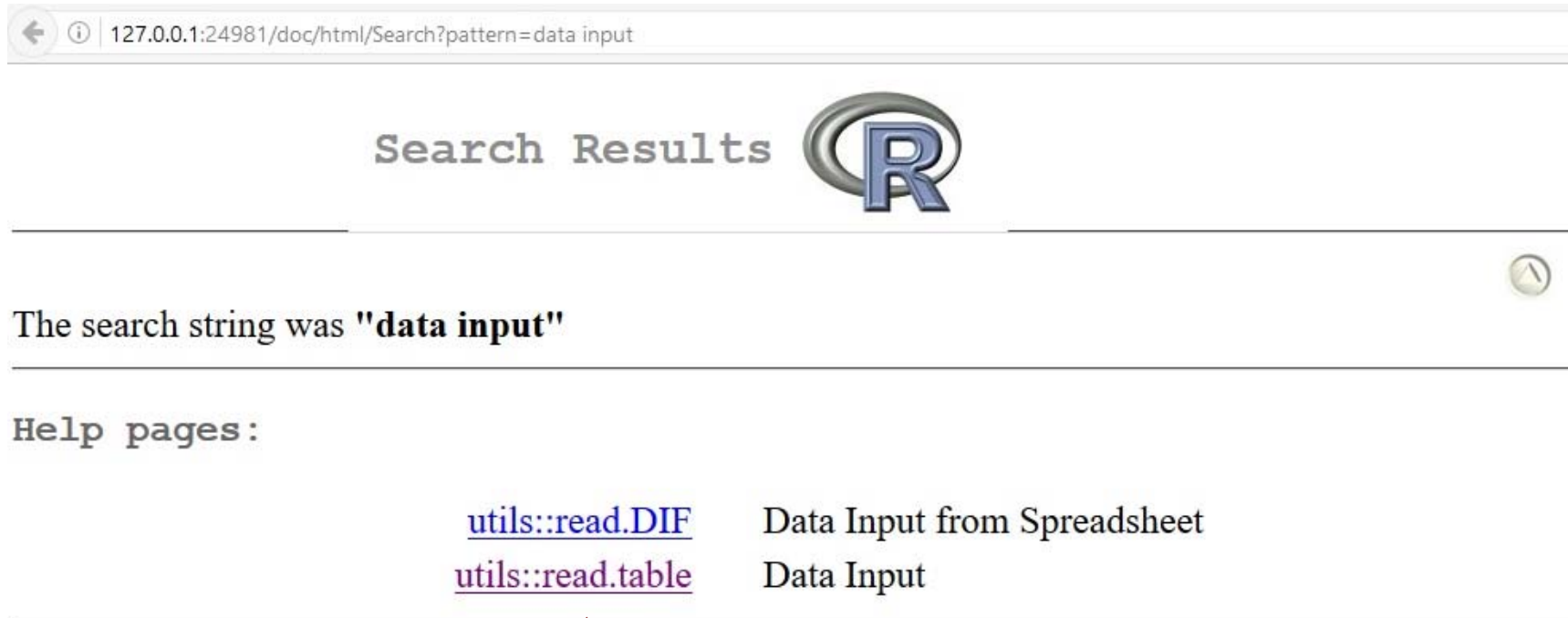
4. Sometimes, you want to search by the subject on which we want help (e.g. data input). In such a case, type `help.search("data input")`

```
R Console

> help.search("data input")
starting httpd help server ... done
>
```

Then we get....

Then we get....

127.0.0.1:24981/doc/html/Search?pattern=data input

Search Results

The search string was **"data input"**

Help pages:

utils::read.DIF	Data Input from Spreadsheet
utils::read.table	Data Input

Clicking over the link give required information
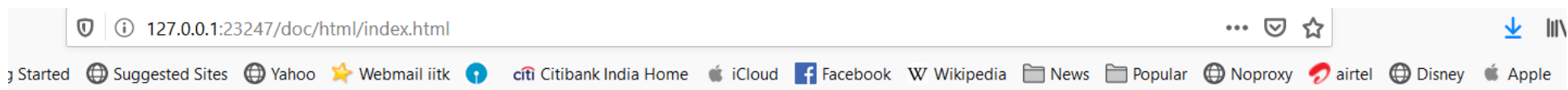
Getting Help in R

4. '`help()`' for on-line help,

   or '`help.start()`' for an HTML browser interface to help.

```
R  R Console

> help()
> help.start()
If nothing happens, you should open
'http://127.0.0.1:13077/doc/html/index.html' yourself
> |
```

Then we get….

Getting Help in R

Getting Help in R

5) Other useful functions are `find` and `apropos`.

6) The `find` function tells us what package something is in.

For example

```
> find("lowess")  returns
    [1] "package:stats"
```

```
R Console
> find("lowess")
[1] "package:stats"
>
>
```

Getting Help in R

7) The apropos returns a character vector giving the names of all objects in the search list that match your enquiry.

`apropos("lm")` returns

```
R Console

> apropos("lm")
 [1]  ".__C__anova.glm"        ".__C__anova.glm.null"  ".__C__glm"
 [4]  ".__C__glm.null"         ".__C__lm"              ".__C__mlm"
 [7]  ".__C__optionalMethod"   ".colMeans"             "anova.glm"
[10]  "anova.glmlist"          "anova.lm"              "anova.lmlist"
[13]  "anova.mlm"              "colMeans"              "contr.helmert"
[16]  "getAllMethods"          "glm"                   "glm.control"
[19]  "glm.fit"                "hatvalues.lm"          "KalmanForecast"
[22]  "KalmanLike"             "KalmanRun"             "KalmanSmooth"
[25]  "kappa.lm"               "lm"                    "lm.fit"
[28]  "lm.influence"           "lm.wfit"               "model.frame.glm"
[31]  "model.frame.lm"         "model.matrix.lm"       "nlm"
[34]  "nlminb"                 "plot.lm"               "plot.mlm"
[37]  "predict.glm"            "predict.lm"            "predict.mlm"
[40]  "print.glm"              "print.lm"              "residuals.glm"
[43]  "residuals.lm"           "rstandard.glm"         "rstandard.lm"
[46]  "rstudent.glm"           "rstudent.lm"           "summary.glm"
[49]  "summary.lm"             "summary.mlm"
```

Worked Examples of Functions

To see a worked `example` just type the function name, e.g., `lm` for linear models:

`example(lm)`

and we see the printed and graphical output produced by the `lm` function.

```
R R Console

> example(lm)

lm> require(graphics)

lm> ## Annette Dobson (1990) "An Introduction to Generalized Linear Models".
lm> ## Page 9: Plant Weight Data.
lm> ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)

lm> trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)

lm> group <- gl(2, 10, 20, labels = c("Ctl","Trt"))

lm> weight <- c(ctl, trt)

lm> lm.D9 <- lm(weight ~ group)

lm> lm.D90 <- lm(weight ~ group - 1) # omitting intercept

lm> ## No test:
lm> anova(lm.D9)
Analysis of Variance Table

Response: weight
          Df Sum Sq Mean Sq F value Pr(>F)
group      1 0.6882 0.68820  1.4191  0.249
Residuals 18 8.7292 0.48496

lm> summary(lm.D90)
```

…and other details follow further

Demonstration of R Functions

This can be useful for seeing the type of things that R can do.

`demo(persp)`   [`persp` is a command for 3d surface plots]

…and it continues

Demonstration of R Functions

This can be useful for seeing the type of things that R can do.

`demo(graphics)`

```
R R Console                                                          □

> demo(graphics)

        demo(graphics)
        ---- ~~~~~~~~

Type  <Return>   to start :

> #  Copyright (C) 1997-2009 The R Core Team
>
> require(datasets)

> require(grDevices); require(graphics)

> ## A little color wheel.        This code just plots equally spaced hues in
> ## a pie chart.        If you have a cheap SVGA monitor (like me) you will
> ## probably find that numerically equispaced does not mean visually
> ## equispaced.  On my display at home, these colors tend to cluster at
> ## the RGB primaries.  On the other hand on the SGI Indy at work the
> ## effect is near perfect.
>
> par(bg = "gray")

> pie(rep(1,24), col = rainbow(24), radius = 0.9)
Waiting to confirm page change...

> title(main = "A Sample Color Wheel", cex.main = 1.4, font.main = 3)

> title(xlab = "(Use this as a test of monitor linearity)",
+         cex.lab = 0.8, font.lab = 3)

> ## We have already confessed to having these.  This is just showing off X11
> ## color names (and the example (from the postscript manual) is pretty "cute".
```
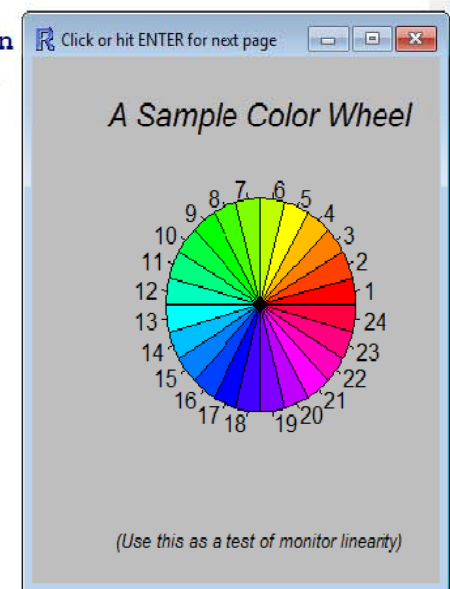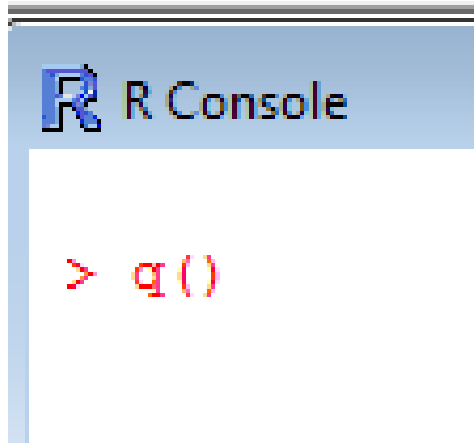

A Sample Color Wheel

(Use this as a test of monitor linearity)

How to quit in R

Type 'q ( )' to quit R.