

# Introduction to R and RStudio - I

Dr. Amar Ahmad  
NYU Abu Dhabi

**LIBRARY**  
NYU ABU DHABI

February 15, 2023

# Outline

# Outline

- ▶ The R Story

# Outline

- ▶ The R Story
- ▶ Working with R and RStudio

# Outline

- ▶ The R Story
- ▶ Working with R and RStudio
- ▶ Online help

# Outline

- ▶ The R Story
- ▶ Working with R and RStudio
- ▶ Online help
- ▶ Read and Write Data in RStudio

# Outline

- ▶ The R Story
- ▶ Working with R and RStudio
- ▶ Online help
- ▶ Read and Write Data in RStudio
- ▶ Basic Arithmetic Operators

# Outline

- ▶ The R Story
- ▶ Working with R and RStudio
- ▶ Online help
- ▶ Read and Write Data in RStudio
- ▶ Basic Arithmetic Operators
- ▶ Some Statistical Functions



# Outline

- ▶ The R Story
- ▶ Working with R and RStudio
- ▶ Online help
- ▶ Read and Write Data in RStudio
- ▶ Basic Arithmetic Operators
- ▶ Some Statistical Functions
- ▶ Graphs in R

# The R Story - What are R, S and S-PLUS?

# The R Story - What are R, S and S-PLUS?

- ▶ **S** is a statistical programming language developed primarily between 1975-1976 by John Chambers (**statistician**) at Bell Laboratories for data analysis and visualization (earlier contributors were Rick Becker and Allan Wilks.)

# The R Story - What are R, S and S-PLUS?

- ▶ **S** is a statistical programming language developed primarily between 1975-1976 by John Chambers (**statistician**) at Bell Laboratories for data analysis and visualization (earlier contributors were Rick Becker and Allan Wilks.)
- ▶ **S-PLUS** is a commercial implementation of **S**, which is implement, distributed and maintained from the company (TIBCO Software)

# The R Story - What are R, S and S-PLUS?

- ▶ **S** is a statistical programming language developed primarily between 1975-1976 by John Chambers (**statistician**) at Bell Laboratories for data analysis and visualization (earlier contributors were Rick Becker and Allan Wilks.)
- ▶ **S-PLUS** is a commercial implementation of **S**, which is implement, distributed and maintained from the company (TIBCO Software)
- ▶ **R** is a free implementation of a dialect of the **S** language

# Is R Similar to S-PLUS?

# Is R Similar to S-PLUS?

- ▶ R and S-PLUS are very similar for basic command-line

# Is R Similar to S-PLUS?

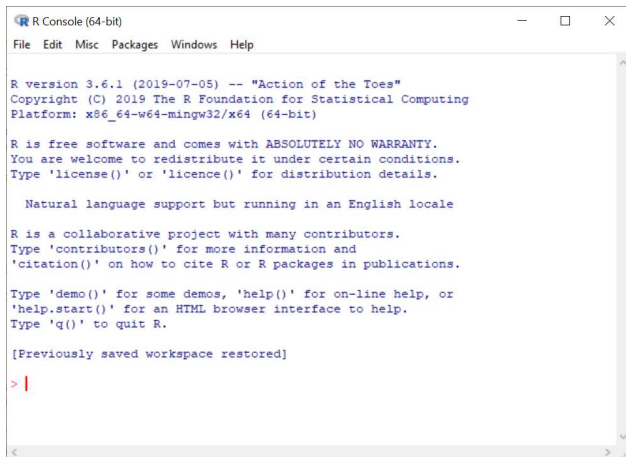
- ▶ R and S-PLUS are very similar for basic command-line
- ▶ Most written programs in R and S-PLUS can be translated straightforwardly to the other



# Is R Similar to S-PLUS?

- ▶ R and S-PLUS are very similar for basic command-line
- ▶ Most written programs in R and S-PLUS can be translated straightforwardly to the other
- ▶ However, large programs need some translation

# R Console - R version 3.6.1



```
R Console (64-bit)
File Edit Misc Packages Windows Help

R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 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

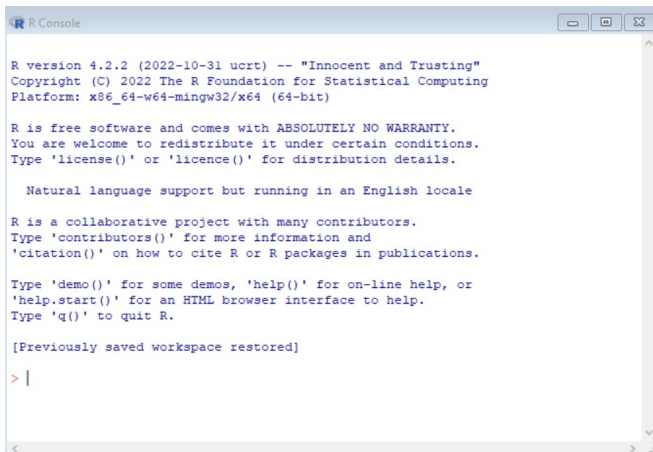
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

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]

> |
```

# R Console - R version 4.2.2



```
R Console

R version 4.2.2 (2022-10-31 ucrt) -- "Innocent and Trusting"
Copyright (C) 2022 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

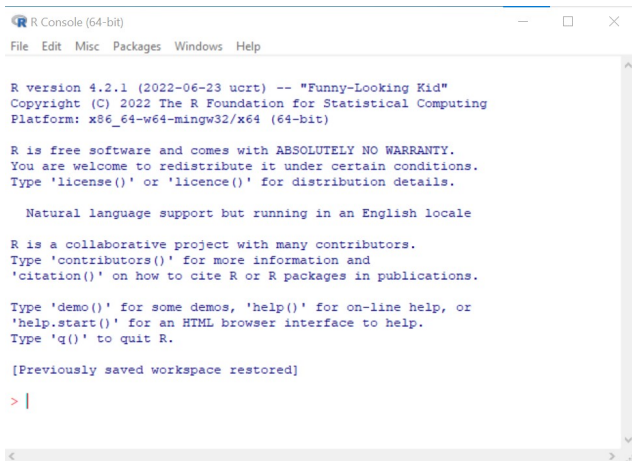
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

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]

> |
```

# R Console - R version 4.2.1



```
R Console (64-bit)
File Edit Misc Packages Windows Help

R version 4.2.1 (2022-06-23 ucrt) -- "Funny-Looking Kid"
Copyright (C) 2022 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]

> |
```

# R versions

```
> Object[1:10,]
  version      date      nickname
1    0.60 1997-12-04 08:47:58    <NA>
2    0.61 1997-12-21 13:09:22    <NA>
3    0.61.1 1998-01-10 00:31:55    <NA>
4    0.61.2 1998-03-14 19:25:55    <NA>
5    0.61.3 1998-05-02 07:58:17    <NA>
6    0.62 1998-06-14 12:56:20    <NA>
7    0.62.1 1998-06-14 22:13:25    <NA>
8    0.62.2 1998-07-10 11:13:45    <NA>
9    0.62.3 1998-08-28 09:02:19    <NA>
10   0.62.4 1998-10-23 12:08:41    <NA>
> Object[79:89,]
  version      date      nickname
79  2.14.0 2011-10-31 08:09:09    Great Pumpkin
80  2.14.1 2011-12-22 08:10:18    December Snowflakes
81  2.14.2 2012-02-29 08:10:10    Gift-Getting Season
82  2.15.0 2012-03-30 07:16:05    Easter Beagle
83  2.15.1 2012-06-22 07:09:44    Roasted Marshmallows
84  2.15.2 2012-10-26 07:11:16    Trick or Treat
85  2.15.3 2013-03-01 08:28:29    Security Blanket
86  3.0.0 2013-04-03 07:12:36    Masked Marvel
87  3.0.1 2013-05-16 07:11:33    Good Sport
88  3.0.2 2013-09-25 07:11:09    Frisbee Sailing
89  3.0.3 2014-03-06 08:12:33    Warm Puppy
> Object[120:129,]
  version      date      nickname
120 4.0.3 2020-10-10 07:05:24    Bunny-Wunnies Freak Out
121 4.0.4 2021-02-15 08:05:13    Lost Library Book
122 4.0.5 2021-03-31 07:05:15    Shake and Throw
123 4.1.0 2021-05-18 07:05:22    Camp Pontanezen
124 4.1.1 2021-08-10 07:05:06    Kick Things
125 4.1.2 2021-11-01 08:05:12    Bird Hippie
126 4.1.3 2022-03-10 08:05:38    One Push-Up
127 4.2.0 2022-04-22 07:05:41    Vigorous Calisthenics
128 4.2.1 2022-06-23 07:05:33    Funny-Looking Kid
129 4.2.2 2022-10-31 08:05:54    Innocent and Trusting
```

# Advantages of R

# Advantages of R

- ▶ No license fee

# Advantages of R

- ▶ No license fee
- ▶ Open Source, therefore NO Black Box



# Advantages of R

- ▶ No license fee
- ▶ Open Source, therefore NO Black Box
- ▶ All algorithms and Functions in the source code are available.

# Advantages of R

- ▶ No license fee
- ▶ Open Source, therefore NO Black Box
- ▶ All algorithms and Functions in the source code are available.
- ▶ However, for some functions written in C, C++ or Fortran, there is de facto a Black Box

# Advantages of R

# Advantages of R

- ▶ Rapid development

# Advantages of R

- ▶ Rapid development
- ▶ Cross-platform: Macintosh, Windows, Linux, etc

# Advantages of R

- ▶ Rapid development
- ▶ Cross-platform: Macintosh, Windows, Linux, etc
- ▶ New statistical methods are mostly offered as a package

# Advantages of R

- ▶ Rapid development
- ▶ Cross-platform: Macintosh, Windows, Linux, etc
- ▶ New statistical methods are mostly offered as a package
- ▶ Faster than **S-Plus**

# Disadvantages of R



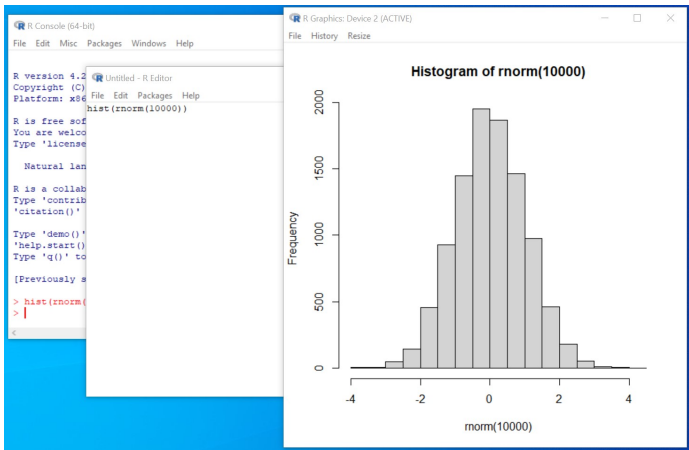
# Disadvantages of R

- ▶ Error messages can be difficult to understand

# Disadvantages of R

- ▶ Error messages can be difficult to understand
- ▶ No completed graphical user interface (Many tries)

# R Ties



# S-Plus

The screenshot displays the S-Plus Workbench interface. The main window shows a script for an option pricing function. The script defines a function `OptionPricing` that takes `sources.dir`, `results.dir`, and `properties.list` as arguments. It includes a `library(winjava)` call and a `source` statement to load `simoptions.ssc`. The script sets parameters for the number of simulations (`nsims`), option days (`optiondays`), strike price (`strike`), volatility (`vol`), and start price (`startprice`). It then uses a `method` switch to select between "Gaussian" and "Mixture" models. The script reads data from a file and prints the `properties.list`. Finally, it uses `filename` to save the `properties.list` to a file named `SP500.xls`.

```
1>OptionPricing <- function(sources.dir, results.dir, properties.list, par
2
3   library(winjava)
4   source(paste(sources.dir,"simoptions.ssc",sep=""))
5
6   ## parameters
7   nsims<-1000 ## number of simulations
8
9   optiondays <- switch(as.character(parameter.list$Expiration),
10                        "3 Months"=90, "1 Year"=250, "2 Years"=500) ## length of option
11
12   strike <- as.numeric(as.character(parameter.list$Strike))
13
14   vol <- 40 ## returns volatility, annualized (40 = 40%) [Should be on
15   startprice <- 100 ## option start price
16
17   method <- switch(as.character(parameter.list$Model),
18                  "Gaussian"="gbm", "Mixture"="normmix") ## method to estimate opt
19
20   ## read in data
21   print(properties.list)
22   filename<-paste(properties.list$inputs.root, "/OptionPricing/SP500.xls")
```

The object browser at the bottom shows the contents of the `object()` function, listing various attributes and their values:

Attribute	Value	Object
"last value"	"Random seed"	"OptionPricing"
"attribute vec"	"barsData"	"dat"
"dsornix"	"file vec"	"g mod"
"grp na"	"intra"	"isep"
"local dir"	"lookback.len"	"listy"
"asit ret"	"nu"	"parameter list"
"path obj"	"path_sornix"	"pdData"
"pred"	"pred.len"	"price"
"q.59"	"results dir"	"ret"
"ad"	"server dir"	"sin_option"
"source vec"	"sources dir"	"stk"
"sys"	"today"	"var 95"
"x"	"x"	"xmax"
"yax"	"ymax"	"ymin"

## RStudio IDE

The most popular coding environment for R, built with love by Posit.

Used by millions of people weekly, the RStudio integrated development environment (IDE) is a set of tools built to help you be more productive with R and Python. It includes a console, syntax-highlighting editor that supports direct code execution. It also features tools for plotting, viewing history, debugging and managing your workspace.

# RStudio Interface

The screenshot shows the RStudio interface with the following components:

- Program Window:** Located at the top left, it contains a script with two lines of R code:

```
1 ## Combine values into a vector
2 x <- c(2,3,1,5,4) |
```
- Environment Window:** Located at the top right, it displays "Environment is empty".
- Console Window:** Located at the bottom left, it shows the R startup message:

```
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]

> |
```
- Other Windows:** Located at the bottom right, it shows the "Files", "Plots", "Packages", "Help", and "Viewer" tabs.

# R License

# R License

- ▶ Type `license()` in the R console



# R License

- ▶ Type `license()` in the R console

```
This software is distributed under the terms of the GNU General  
Public License, either Version 2, June 1991 or Version 3, June 2007.  
The terms of version 2 of the license are in a file called COPYING  
which you should have received with  
this software and which can be displayed by RShowDoc("COPYING").  
Version 3 of the license can be displayed by RShowDoc("GPL-3").
```

Copies of both versions 2 and 3 of the license can be found  
at <https://www.R-project.org/Licenses/>.

```
A small number of files (the API header files listed in  
R_DOC_DIR/COPYRIGHTS) are distributed under the  
LESSER GNU GENERAL PUBLIC LICENSE, version 2.1 or later.  
This can be displayed by RShowDoc("LGPL-2.1"),  
or obtained at the URI given.  
Version 3 of the license can be displayed by RShowDoc("LGPL-3").
```

```
'Share and Enjoy.'
```

# How to Cite R in Publications

# How to Cite R in Publications

To cite R in publications use: R Core Team (2022). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.

# How to Cite R in Publications

To cite R in publications use: R Core Team (2022). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.

We have invested a lot of time and effort in creating R, please cite it when using it for data analysis. See also 'citation("pkgname")' for citing R packages.

# Basic Notations

# Basic Notations

- ▶  $< -$  is the original assignment operator, which is written by typing two keystrokes the less than symbol ( $<$ ) followed by a hyphen (minus) symbol ( $-$ )

# Basic Notations

- ▶ `< -` is the original assignment operator, which is written by typing two keystrokes the less than symbol (`<`) followed by a hyphen (minus) symbol (`-`)
- ▶ The assignment operator `=` is also used in **R** and was added in 2001 mainly as it is the accepted assignment operator in many other languages

# Basic Notations

- ▶  $< -$  is the original assignment operator, which is written by typing two keystrokes the less than symbol ( $<$ ) followed by a hyphen (minus) symbol ( $-$ )
- ▶ The assignment operator  $=$  is also used in R and was added in 2001 mainly as it is the accepted assignment operator in many other languages
- ▶ It makes it easier for R beginners who came from other languages to use  $=$



# Reading Data Into R: RStudio

# Reading Data Into R: RStudio

- ▶ We can read a publicly available data from "winter olympics medals" website into R

# Reading Data Into R: RStudio

- ▶ We can read a publicly available data from "winter olympics medals" website into R
- ▶ This can be done by using the R function `read.csv`

# Reading Data Into R: RStudio

- ▶ We can read a publicly available data from "winter olympics medals" website into R
- ▶ This can be done by using the R function `read.csv`

```
> Dat<-read.csv("http://winterolympicsmedals.com/medals.csv")
```

# Reading Data Into R: RStudio

- ▶ We can read a publicly available data from "winter olympics medals" website into R
- ▶ This can be done by using the R function `read.csv`

```
> Dat<-read.csv("http://winterolympicsmedals.com/medals.csv")
```

- ▶ And view it as

# Reading Data Into R: RStudio

▶ We can read a publicly available data from "winter olympics medals" website into R

▶ This can be done by using the R function `read.csv`

```
> Path <- "https://nyuad.nyu.edu/content/dam/nyuad/research/centers-labs-projects/  
public-health-research-center/statslab/training/datasets/  
WinterOlympicsMedalsAPI.csv"
```

```
> Dat<-read.csv(Path)
```

▶ And view it as

```
> View(Dat)
```

# Start of an R code

# Start of an R code

- ▶ The prompt symbol `>` is not part of an R code



# Start of an R code

- ▶ The prompt symbol `>` is not part of an R code
- ▶ `>` should not be used at the beginning of an R code

# The Winter Olympics Medals API

# The Winter Olympics Medals API

- **year** - the year of the olympics
- **city** - the city it was held in, like "Lillehammer" or "Sarajevo"
- **sport** - the sport
- **discipline** - the sub-discipline
- **country** - the country as an NOC code
- **event** - the event, like "alpine combined" or "two-man"
- **gender** - the athlete's gender, X is for pair sports
- **type** - the medal type (Gold, Silver, Bronze)

Any of these could also be used as a wildcard, so `country="g"` would find GBR, FRG, GDR, YUG and GER.

# Reading Data Into R: RStudio

The screenshot shows the RStudio interface. At the top, the menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. Below the menu bar is a toolbar with icons for file operations and a search box. The main workspace displays a data table with the following columns: Year, City, Sport, Discipline, NOC, Event, Event.gender, and Medal. The table contains 5 visible rows of data for the year 1924. Below the table, it indicates 'Showing 1 to 5 of 2,311 entries, 8 total columns'. The console window at the bottom shows the R version (4.2.2) and the command used to read the data from a CSV file.

	Year	City	Sport	Discipline	NOC	Event	Event.gender	Medal
1	1924	Chamonix	Skating	Figure skating	AUT	individual	M	Silver
2	1924	Chamonix	Skating	Figure skating	AUT	individual	W	Gold
3	1924	Chamonix	Skating	Figure skating	AUT	pairs	X	Gold
4	1924	Chamonix	Bobsleigh	Bobsleigh	DEL	four-man	M	Bronze
5	1924	Chamonix	Bobsleigh	Bobsleigh	DEL	two-man	M	Gold

```
R R 4.2.2 · ~/r
R version 4.2.2 (2022-10-31 ucrt) -- "Innocent and Trusting"
Copyright (C) 2022 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.

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]

> Dat<-read.csv("http://winterolympicsmedals.com/medals.csv")
> View(Dat)
> |
```

# Column Names of an Object

# Column Names of an Object

- ▶ The R function `colnames` can be used to retrieve or set the column names of the dataset `Dat`

# Column Names of an Object

- ▶ The `R` function `colnames` can be used to retrieve or set the column names of the dataset `Dat`

```
> colnames(Dat)
[1] "Year"    "City" "Sport" "Discipline" "NOC"
[6] "Event"   "Event.gender" "Medal"
```

```
> colnames(Dat)[7] <- "EventGender"
colnames(Dat)
[1] "Year"    "City" "Sport" "Discipline" "NOC"
[6] "Event"   "EventGender" "Medal"
```

# Dimensions of an Object



# Dimensions of an Object

- ▶ The `R` function `dim` can be used to get the dimension of the define object `dat` as

# Dimensions of an Object

- ▶ The R function `dim` can be used to get the dimension of the define object `dat` as

```
> dim(Dat)
[1] 2311    8
```

# Data Output

# Data Output

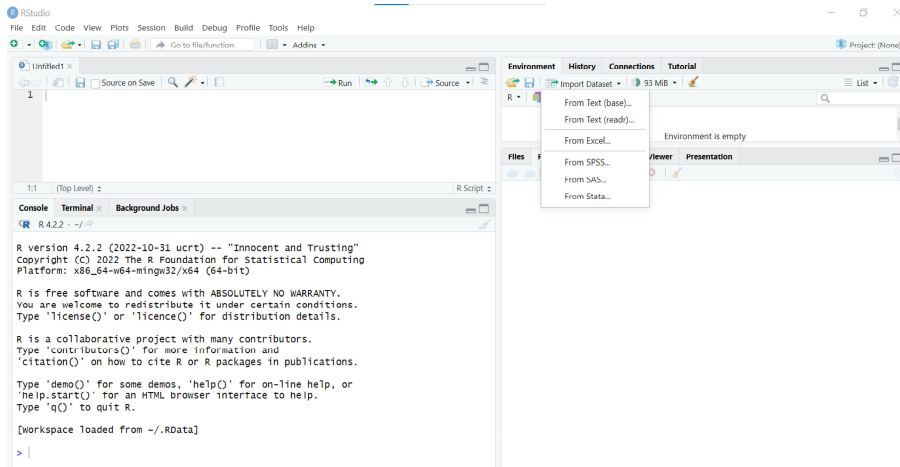
- ▶ The dataset `Dat` can be saved as a CSV (Comma Separated Values) file using the R function `write.csv`

# Data Output

- ▶ The dataset `Dat` can be saved as a CSV (Comma Separated Values) file using the R function `write.csv`

```
> Path <- "C:\\Users\\asa12\\Downloads\\WinterOlympicsMedalsAPI.csv"  
> write.csv(Dat,Path,row.names=F)
```

# Reading Data Into R: RStudio



The screenshot displays the RStudio environment. The main editor shows a script with a single line of code: `1`. The console window at the bottom shows the R startup message for version 4.2.2, including copyright information and instructions on how to use help and quit. The Environment pane on the right shows the current workspace is empty, with a context menu open over the 'R' object. The menu options are: 'From Text (base)...', 'From Text (readr)...', 'From Excel...', 'From SPSS...', 'From SAS...', and 'From Stata...'.

```
1
```

R version 4.2.2 (2022-10-31 ucrt) -- "Innocent and Trusting"  
Copyright (C) 2022 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.

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]

> |

Environment History Connections Tutorial  
R Import Dataset 93 MiB  
Environment is empty  
Files  
Viewer Presentation

- From Text (base)...
- From Text (readr)...
- From Excel...
- From SPSS...
- From SAS...
- From Stata...

# Reading Data Into R: RStudio

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

Project (None)

Environment History Connections Tutorial

Import Text Data

File/URL:  Browse...

Data Preview:

Import Options:

Name:   First Row as Names Delimiter:  Escape:

Skips:   Trim Spaces Quotes:  Comment:

Open Data Viewer Locale:  NA:

Code Preview:

```
library(readr)
dataset <- read_csv(NULL)
View(dataset)
```

Console

R 4.2.2

R version  
copyright  
Platform:

R is free  
You are w  
Type 'lic

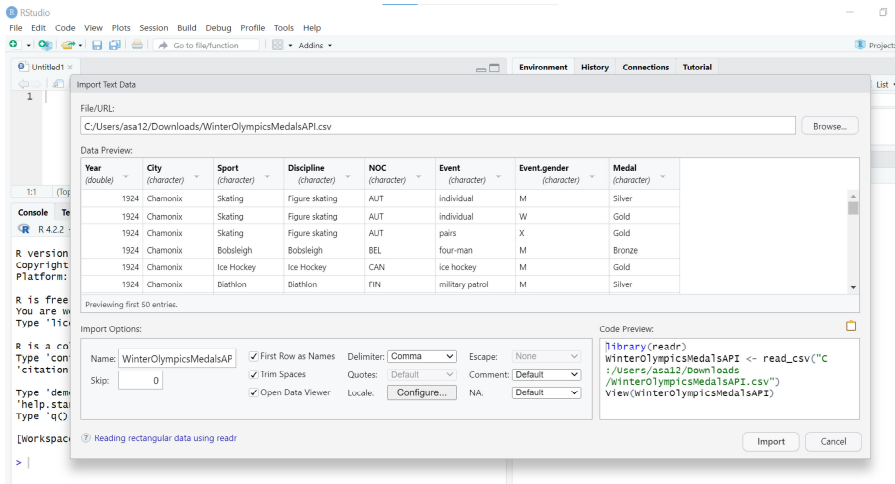
R is a co  
Type 'con  
'citation

Type 'dem  
'help, sta  
Type 'q()

[Workspace] [? Reading rectangular data using readr](#)

Import Cancel

# Reading Data Into R: RStudio



File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

Environment History Connections Tutorial

Import Text Data

File/URL: C:/Users/asa12/Downloads/WinterOlympicsMedalsAPI.csv

Data Preview:

Year (double)	City (character)	Sport (character)	Discipline (character)	NOC (character)	Event (character)	Event Gender (character)	Medal (character)
1924	Chamonix	Skating	figure skating	AUT	individual	M	Silver
1924	Chamonix	Skating	Figure skating	AUT	individual	W	Gold
1924	Chamonix	Skating	Figure skating	AUT	pairs	X	Gold
1924	Chamonix	Bobsleigh	Bobsleigh	BEL	four-man	M	Bronze
1924	Chamonix	Ice Hockey	Ice Hockey	CAN	ice hockey	M	Gold
1924	Chamonix	Diathlon	Diathlon	FIN	military patrol	M	Silver

Previewing first 50 entries.

Import Options:

Name: WinterOlympicsMedalsAPI  First Row as Names Delimiter: Comma Escape: None

Skip: 0  Trim Spaces Quotes: Default Comment: Default

Open Data Viewer Locale: Configure... NA: Default

Code Preview:

```
library(readr)
winterOlympicsMedalsAPI <- read_csv("C:/Users/asa12/Downloads/WinterOlympicsMedalsAPI.csv")
View(winterOlympicsMedalsAPI)
```

Import Cancel



# Reading Data Into R: RStudio

The screenshot shows the RStudio interface with a CSV file loaded into a data frame. The Environment pane on the right shows the data frame with 2311 observations and 8 variables. The Console pane shows the R code used to load the data.

Year	City	Sport	Discipline	NOC	Event	Event.gender	Medal
1	1924	Chamonix	Skating	Figure skating	AUT	individual	Silver
2	1924	Chamonix	Skating	Figure skating	AUT	individual	Gold
3	1924	Chamonix	Skating	Figure skating	AUT	pairs	Gold
4	1924	Chamonix	Bobsleigh	Bobsleigh	BEL	four-man	Bronze

```
R 4.2.2 ~ /  
type 'contributions()' 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]  
  
> library(readr)  
> WinterOlympicsMedalsAPI <- read_csv("C:/Users/asai2/Downloads/WinterOlympicsMedalsAPI.csv")  
Rows: 2311 Columns: 8  
— Column specification —  
Delimiter: ","  
chr (7): City, Sport, Discipline, NOC, Event, Event.gender, Medal  
dbl (1): Year  
  
i Use 'spec()' to retrieve the full column specification for this data.  
i Specify the column types or set 'show_col_types = FALSE' to quiet this message.  
> View(WinterOlympicsMedalsAPI)  
> |
```

# Reading Data Into R: RStudio

# Reading Data Into R: RStudio

- ▶ We can see that the following command was used to import the data into **RStudio**

# Reading Data Into R: RStudio

- ▶ We can see that the following command was used to import the data into **RStudio**

```
> library(readr)
> dat <- read_csv("C:/Users/asa12/Downloads/WinterOlympicsMedalsAPI.csv")
> View(dat)
```

# Cross Tabulation Creation

# Cross Tabulation Creation

- ▶ We can use the `R` function `table` to build a contingency table of Year by medal type

# Cross Tabulation Creation

- ▶ We can use the `R` function `table` to build a contingency table of Year by medal type

```
> Tab<- table(dat$Medal,dat$Year)
> # print the first five columns
> Tab[,1:5]
```

	1924	1928	1932	1936	1948
Bronze	17	15	14	17	22
Gold	16	14	14	17	22
Silver	16	12	14	17	24

# Winter Olympics Medals API



# Winter Olympics Medals API

- ▶ The `R` function `barplot` can be used to create a bar plot using the defined `Tab` object

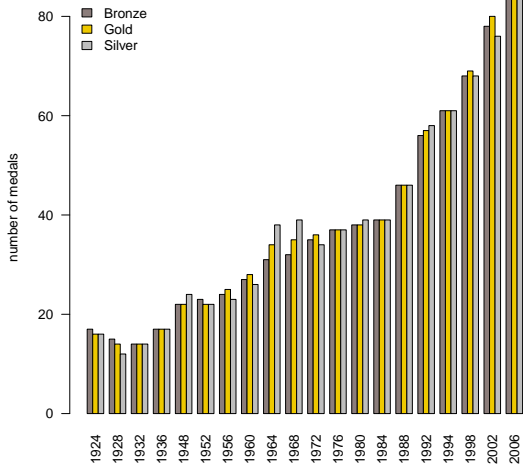
# Winter Olympics Medals API

- ▶ The R function `barplot` can be used to create a bar plot using the defined `Tab` object

```
> barplot(Tab,beside = T,las=2,ylab="number of medals",col=c("mistyrose4","gold2","mistyrose4"))  
> legend("topleft",c("Bronze", "Gold", "Silver"),fill =c("mistyrose4","gold2","mistyrose4"))
```

# Winter Olympics Medals API

# Winter Olympics Medals API



# Getting Help with R

# Getting Help with R

- ▶ Ask **R** before asking a question

# Getting Help with R

- ▶ Ask **R** before asking a question
- ▶ This can be done using the **R help()** or simply by using question mark **?**

# Getting Help with R

- ▶ Ask **R** before asking a question
- ▶ This can be done using the **R help()** or simply by using question mark **?**
- ▶ For example, to read the documentation of the **arithmetic mean** function in **R** we use the following

```
> ?mean  
# or  
> help(mean)
```



# R Documentation - Arithmetic Mean

mean {base}

R Documentation

## Arithmetic Mean

### Description

Generic function for the (trimmed) arithmetic mean.

### Usage

```
mean(x, ...)
```

## Default S3 method:

```
mean(x, trim = 0, na.rm = FALSE, ...)
```

### Arguments

- x**  
An R object. Currently there are methods for numeric/logical vectors and [date](#), [date-time](#) and [time interval](#) objects. Complex vectors are allowed for `trim = 0`, only.
- trim**  
the fraction (0 to 0.5) of observations to be trimmed from each end of `x` before the mean is computed. Values of `trim` outside that range are taken as the nearest endpoint.
- na.rm**  
a logical evaluating to `TRUE` or `FALSE` indicating whether `NA` values should be stripped before the computation proceeds.
- ...**  
further arguments passed to or from other methods.

### Value

If `trim` is zero (the default), the arithmetic mean of the values in `x` is computed, as a numeric or complex vector of length one. If `x` is not logical (coerced to numeric), numeric (including integer) or complex, `NA_real_` is returned, with a warning.

# Adding comments

# Adding comments

- ▶ The hash symbol (#) can be used to start of a comment

# Adding comments

- ▶ The hash symbol (#) can be used to start of a comment
- ▶ One can type the comments, i.e. the information wants to add, after the hash symbol

# Adding comments

- ▶ The hash symbol (`#`) can be used to start of a comment
- ▶ One can type the comments, i.e. the information wants to add, after the hash symbol
- ▶ `R` will ignore everything written after the hash symbol

# Speed and Stopping Distances of Cars

> ?cars

cars {datasets}

R Documentation

## Speed and Stopping Distances of Cars

### Description

The data give the speed of cars and the distances taken to stop. Note that the data were recorded in the 1920s.

### Usage

**cars**

### Format

A data frame with 50 observations on 2 variables.

[,1] speed numeric Speed (mph)

[,2] dist numeric Stopping distance (ft)

### Source

Ezekiel, M. (1930) *Methods of Correlation Analysis*. Wiley.

### References

McNeil, D. R. (1977) *Interactive Data Analysis*. Wiley.

### Examples

# Speed and Stopping Distances of Cars

# Speed and Stopping Distances of Cars

- ▶ One uses the tilde symbol  $\sim$  within formulas of statistical models in blueR



# Speed and Stopping Distances of Cars

- ▶ One uses the tilde symbol  $\sim$  within formulas of statistical models in blueR
- ▶ The left side of the tilde symbol the target variable and the right side is the predictor

# Speed and Stopping Distances of Cars

- ▶ One uses the tilde symbol  $\sim$  within formulas of statistical models in blueR
- ▶ The left side of the tilde symbol the target variable and the right side is the predictor

```
> plot(dist~speed, data=cars, xlab = "Speed (mph)",  
+ ylab = "Stopping distance (ft)",las = 1,pch=19)
```

# Start of an R code

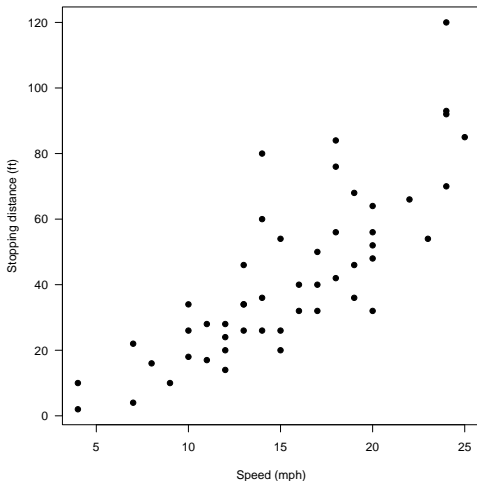
# Start of an R code

- ▶ The prompt symbol `+` is a continuation symbol

# Start of an R code

- ▶ The prompt symbol `+` is a continuation symbol
- ▶ It indicates that this line of code still belongs to the previous line of code

# Speed and Stopping Distances of Cars



# Speed and Stopping Distances of Cars

```
> summary(cars)
```

speed		dist	
Min.	: 4.0	Min.	: 2.00
1st Qu.:	12.0	1st Qu.:	26.00
Median	:15.0	Median	: 36.00
Mean	:15.4	Mean	: 42.98
3rd Qu.:	19.0	3rd Qu.:	56.00
Max.	:25.0	Max.	:120.00

# Basic Notations



# Basic Notations

- ▶ The colon operator `:` is an R function that generates regular sequences

# Basic Notations

- ▶ The colon operator `:` is an R function that generates regular sequences
- ▶ The function `c()` makes a single vector from its arguments

# Basic Notations

- ▶ The colon operator `:` is an R function that generates regular sequences
- ▶ The function `c()` makes a single vector from its arguments

```
# create a sequence of year variable
> Year <- 1999:2022
# generate a vector which contains
# the number of R packages
> Npackages <- c(56,41,65,66,101,144,210,285,332,383,523,
+               585,773,1214,1116,1378,1654,2144,2385,
+               2091,2171,2391,3232,6977)
```

# Basic Notations - Vector

# Basic Notations - Vector

- ▶ We can ask R whether `Npackages` is a vector using the function `is.vector()` which returns `TRUE` if `x` is a vector and `FALSE` otherwise

# Basic Notations - Vector

- ▶ We can ask R whether Npackages is a vector using the function `is.vector()` which returns TRUE if x is a vector and FALSE otherwise

```
> is.vector(Npackages)
```

```
[1] TRUE
```

```
> is.vector(Year)
```

```
[1] TRUE
```

# Basic Notations - Numeric Vector

# Basic Notations - Numeric Vector

- ▶ We can check whether `Npackages` is a numerical vector using the function `is.numeric()` which returns `TRUE` if `x1` is a numerical vector and `FALSE` otherwise



# Basic Notations - Numeric Vector

- ▶ We can check whether `Npackages` is a numerical vector using the function `is.numeric()` which returns `TRUE` if `x1` is a numerical vector and `FALSE` otherwise

```
> is.numeric(Npackages)
```

```
[1] TRUE
```

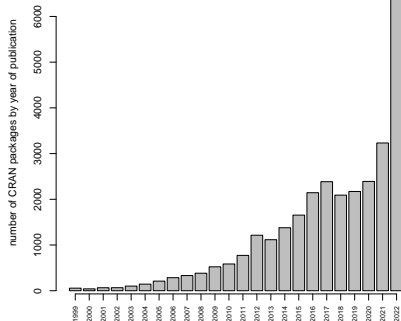
```
> is.numeric(Year)
```

```
[1] TRUE
```

# Number of CRAN Packages By Year of Publication

# Number of CRAN Packages By Year of Publication

```
> hp<-barplot(Npackages,ylab="number of CRAN packages by year of publication")  
> axis(1,at=hp,lab=1999:2022,las=2,cex.axis=0.7)
```



# Some Statistical Functions

# Some Statistical Functions

**Mean** `mean()`

# Some Statistical Functions

**Mean** `mean()` **Variance**

# Some Statistical Functions

**Mean** `mean()` **Variance** `var()`

# Some Statistical Functions

**Mean** `mean()` **Variance** `var()` and **Standard Deviation** `sd()`



# Some Statistical Functions

**Mean** `mean()` **Variance** `var()` and **Standard Deviation** `sd()`

```
> c(mean(Npackages), var(Npackages), sd(Npackages))  
[1] 1263.208 2387336.433 1545.101
```

# Some Statistical Functions

# Some Statistical Functions

**Median:** `median()`

# Some Statistical Functions

**Median:** `median()` **Range:** `range()`

# Some Statistical Functions

**Median:** `median()` **Range:** `range()` and **Interquartile Range:** `IQR()`

# Some Statistical Functions

**Median:** `median()` **Range:** `range()` and **Interquartile Range:** `IQR()`

```
> c(median(Npackages), range(Npackages), IQR(Npackages))  
[1] 679.00 41.00 6977.00 1910.75
```

Thank You For Your Attention!

# References

- ▶ The Book of R: A First Course in Programming and Statistics 1st Edition. Tilman M. Davies 2016. ISBN-10: 9781593276515, ISBN-13: 978-1593276515
- ▶ The Art of R Programming: A Tour of Statistical Software Design 1st Edition. Norman Matloff, 2011. ISBN-10: 1593273843, ISBN-13: 978-1593273842
- ▶ Learning R 1st Edition. Richard Cotton 2013. ISBN-10: 1449357105, ISBN-13: 978-1449357108