

R and R Studio®

Programming Basics

Understanding R code

Object assignment

```
my_var <- 4      b <- "banana"
my_var          b
[1] 4           [1] "banana"
```

Object_name <- value(s)

Objects of single values, data structures, or function results are assigned to a name by the assignment operator (<-).

Variable names may contain letters, numbers, periods, and underscores but must begin with a letter.

Commenting

```
n_st <- 100 #Number
of students
n_st
[1] 100
```

Comments follow a # and are not evaluated by R.

Getting help

args (function)

Display the argument names and default values of a function

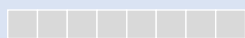
?function or ??function
Get help for a particular function

example (function)

Show example of how to use a function

Data Structures

Vector



One-dimensional data of the same data type

```
> c("A", "Z", "H")
[1] "A" "Z" "H"
```

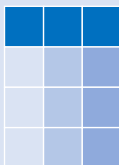
Join elements into a vector

```
> seq(from=2, to=3,
+ by=0.5)
[1] 2 2.5 3
```

Sequence

```
> rep(1, times=3)
[1] 1 1 1
```

Repeat a vector



Data Frame

Two-dimensional of the same or different data type

```
> df<-data.frame(Var1=c("a", "b", "c"),
+ Var2=seq(5, 15, by=5))
```

```
> df
```

	Var1	Var2
1	a	5
2	b	10
3	c	15

```
df$Var1
```

Vectorize a column with \$

```
summary(df)
```

Summary of each column

```
nrow(df)
```

Number of rows

```
ncol(df)
```

Number of columns

```
rename(df, New_name = Old_name)
```

Change name of individual columns (dplyr)

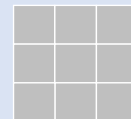
Keyboard shortcuts

Ctrl	+	R	Run selected line(s), R	
Ctrl	+	Enter	Run selected line(s), RStudio	
Ctrl	+	Shift	+ C	(Un)Comment line
Alt	+	-	Insert <- (macOS: Option + -)	
For macOS, replace Ctrl with Cmd				

Matrix

Two-dimensional data of the same data type

```
> matrix(seq(0, 9, by=3),
+ nrow=2, ncol=2)
     [,1] [,2]
[1,]    0    6
[2,]    3    9
```



Array

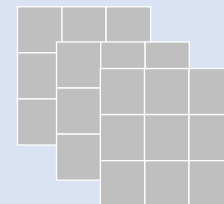
```
> array(c(m1, m2, m3), dim=c(2, 2, 3))
```

```
,,1
     [,1] [,2]
[1,]    0    6
[2,]    3    9

,,2
     [,1] [,2]
[1,]    1    7
[2,]    4   10

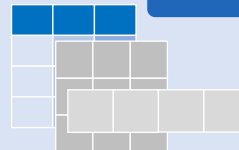
,,3
     [,1] [,2]
[1,]    3    9
[2,]    6   12
```

Three-dimensional matrix (rows, columns, and depth)



List

Lists can be composed of different data types and structures



```
> list(1:10, matrix(c("a", "b",
+ "c", "d"), nrow=2, ncol=2))
[[1]]
[1] 1 2 3 4 5 6 7 8 9 10
```

```
[[2]]
     [,1] [,2]
[1,] "a"  "c"
[2,] "b"  "d"
```



Major Data Types

Numeric

Numbers with decimal values, also known as Double.

- Integer: numbers without decimal values

Character

Text values defined within " or ""

- Factor: text categories for statistical analysis
- Date: calendar values

Logical

Boolean values (TRUE or FALSE)

Missing

Not available / missing values.
Can be present in all data types

Determine data type/structure

class(x)

Get the class of an object

```
var1 <- "pizza"
```

```
class(var1)
[1] "character"
```

Check if data is a specific type
is.<data type>(x)

```
is.character(var1)
```

```
[1] TRUE
```

Or data structure
is.<structure>(x)

```
is.data.frame(var1)
```

```
[1] FALSE
```

Assign data type/structure

Data types may be specified with
as.<data type>(x)

```
var2 <- "8"
```

```
as.numeric(var2)
[1] 8
```

Change data structure with
as.<structure>(x)

```
as.data.frame(var2)
```

```
var2
1 8
```

Math functions

x + y

Addition

x - y

Subtraction

x/y

Division

x^y

Exponentiation

x %% y

Modulo (division remainder)

abs(x)

Absolute value of x

sqrt(x)

Square root of x

log(x)

Natural logarithm of x

exp(x)

Exponential (e^x) of x

max(x)

Largest element in x

min(x)

Smallest element in x

mean(x)

Average value of x

median(x)

Median value of x

sum(x)

Sum of all elements in x

sd(x)

Standard deviation of x

var(x)

Variance of x

round(x, n)

Round x to n decimal places

sin(x)

Sine of x

cos(x)

Cosine of x

tan(x)

Tangent of x

R environment

ls()

List all variables in the environment

rm(x)

Remove x from the environment

rm(list=ls())

Remove all variables from the environment

rm(list=setdiff(ls(), c("x", "y")))

Remove all variables from the environment except variables x and y

Using Packages

install.packages('package')

Download and install a package

library(package)

Load the package, making all its functions available to use

help(package = 'package')

View R help documentation for the package

browseVignettes('package')

Package tutorials or workflows

package::function()

Use a function from a specific package

Character functions

toupper(x) Convert to uppercase

tolower(x) Convert to lowercase

nchar(x) Number of characters in a string

substr(x, start, finish)

Subset a string from start to finish character number

Also see the **stringr** library

See the data

head(x)

Show the first six values

tail(x)

Show the last six values

View(x)

Show all data

str(x)

Compact display of data types and values

Contact:

madgichelp@trentu.ca