Programmes in R – basics.

Example 1. We use R as a simple calculator.

> pi

[1] 3.141593

> sin(pi/2)

[1] 1

> sqrt( 3^2 + 4^2 ) + log(1)\*exp(0)

[1] 5

Example 2. We learn from the help system how to do logarithms in other bases.

> help(log)

...

> log(64, base=4)

[1] 3

Example 3. We define and use a couple of variables.

> a = 5

> a^2

[1] 25

> b <- a

> b^2

[1] 25

Example 4. We define and remove a variable.

> a = 5

> ls()

[1] "a"

> rm( a )

> a

Error: object "a" not found

Example 5. We concatenate several values together into an array a.

> a = c( 2,3,5,7,11 )

> a

[1] 2 3 5 7 11

Example 6. We read in the first ten prime numbers, as they are typed from the keyboard

and then from an existing file called primes.txt.

> primes1 = scan()

1: 2 3 5 7 11

6: 13 17 19 23 29

11:

Read 10 items

> primes2 = scan( file="primes.txt" )

Read 10 items

> primes1

[1] 2 3 5 7 11 13 17 19 23 29

> primes2

[1] 2 3 5 7 11 13 17 19 23 29

Example 7. We do some array arithmetic.

> a = c(1,2,3)

> b = c(5,5,5,5,5,5)

> a^2

[1] 1 4 9

> 4+a

[1] 5 6 7

> a+b

[1] 6 7 8 6 7 8

> sin( a\*pi/2 )

[1] 1.000000e+00 1.224606e-16 -1.000000e+00

Example 8. We compute some statistics for a set of exam scores.

> scores = scan()

1: 81 81 96 77

5: 95 98 73 83

9: 92 79 82 93

13: 80 86 89 60

17: 79 62 74 60

21:

Read 20 items

> range(scores)

[1] 60 98

> median(scores)

[1] 81

> mean(scores)

[1] 81

> sd(scores)

[1] 11.3555

Example 9- vector assessing

>month.abb[c(2,5,7,10)]

## [1] "Feb" "May" "Jul" "Oct"

Example 10 – vector manipulation

>week\_end <- c("Sat","Sun")

more\_days <- c(days,"Thu","Fri",week\_end)

more\_days

## [1] "Mon" "Tue" "Wed" "Thu" "Fri" "Sat" "Sun"

Example 11 – matrix

• Similar knowledge of matrices in R.

> A <- matrix(1:6, 3, 2)

> A

[,1] [,2]

[1,] 1 4

[2,] 2 5

[3,] 3 6

> A[1, ]

[1] 1 4

> A\*A # pointwise product

> t(A) # transpose

> t(A) %\*% A # matrix multiplication

> solve(t(A) %\*% A) # matrix inverse

Example 12- Basics of vectors and matrices

• Vectors: how to create them and extract components and sub-vectors. Operations such as ^, \*, /,

mean, length, sum, exp, log.

> x <- 1:5

> x

[1] 1 2 3 4 5

> x\*x

[1] 1 4 9 16 25

> x^3

[1] 1 8 27 64 125

> sum(x)

[1] 15

> mean(x)

[1] 3

> x <- c(1, 4, 2.8, 3)

> x

[1] 1.0 4.0 2.8 3.0

> x >= 3

[1] FALSE TRUE FALSE TRUE

> x[x >= 3]

[1] 4 3

Example 13-We compute some statistics for a set of exam scores.

> scores = scan()

1: 81 81 96 77

5: 95 98 73 83

9: 92 79 82 93

13: 80 86 89 60

17: 79 62 74 60

21:

Read 20 items

> range(scores)

[1] 60 98

> median(scores)

[1] 81

> mean(scores)

[1] 81

> sd(scores)

[1] 11.3555

Example 14 -We explore a group of exam scores to learn about the shape of the distribution.

> scores = scan()

1: 81 81 96 77

5: 95 98 73 83

9: 92 79 82 93

14.3 How to explore and describe data 153

13: 80 86 89 60

17: 79 62 74 60

21:

Read 20 items

> stem(scores)

The decimal point is 1 digit(s) to the right of the |

6 | 002

7 | 34799

8 | 0112369

9 | 23568

> stem(scores,scale=2) # same plot stretched twice as long

The decimal point is 1 digit(s) to the right of the |

6 | 002

6 |

7 | 34

7 | 799

8 | 01123

8 | 69

9 | 23

9 | 568

Other common ways to explore a set of data are through histograms or box plots. In R,

the hist() function produces histograms, and the boxplot() function creates box plots.

Example 15- We explore the same set of exam grades with a histogram that we have

customized with breaks on the “fives.”

> hist(scores,breaks=c(55,65,75,85,95,105))

Example 15-Finally, we take another look at the same exam grades with a box plot.

> boxplot(scores)