

Exercises:

1. Evaluate the following expressions:

- 1) $11 + 22 + 33 + 44 + 55 + 66 + 77$
- 2) $\frac{11 \times 22 - 33 \times 44}{55 \times 66}$
- 3) $2 \sin^2(\pi/3)$
- 4) $\frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\right)$

2. Evaluate the square root of all the odd numbers between 1 and 25.

3. Create a vector that repeats 10 times a sequence of integers between 1 and 5.

4. Generate:

- (a) a sequence of integers between 5 and 10,
- (b) a sequence of integers of length 5 starting at 41,
- (c) a sequence of real numbers spaced uniformly at 0.2 intervals between 0 and 2,
- (d) a sequence of even integers between 2 and 14.

5. Define an object called **a** and make it an array with four elements which are the integers 5 to 8, inclusive.

- (a) Find element 4 of **a**.
- (b) Calculate the product of elements 2 and 4 of **a**.
- (c) Assign the integers 3 and 4 to an object **b**, then use **b** to find elements 3 and 4 of object **a**.
- (d) Remove the second element of **a** (and call this **d**).
- (e) Remove the third and fourth elements of **a**.

6. Define a matrix with three rows and three columns, the elements being the integers from 1 to 9 inclusive, such that the first row contains the integers 1 to 3 inclusive, the second 4 to 6 inclusive, and the third 7 to 9.

- (a) Find the value on the second row, third column.
- (b) Extract the whole second row and the whole third column.

7. Define an object which is a data frame having the same values as the matrix defined in the above question.

- (a) Extract the value of third variable of the second item.
- (b) Extract the data for the second item.

- (c) Extract the third variable.
8. Create an object with (any) four numeric values, and another object with (any) two character values.
 - (a) Use the `list` function to create an object which is a list of the two objects you have just created.
 - (b) Use the `names` function to change the names of the sub-objects in the list you have just created.
 9. Load the `MASS` package.
 10. Install the `oz` package.
 11. Simulate a vector `x.norm` of 1000 independent $N(2, 4)$ variables (recall that 4 is the *variance*). Find the sum, mean, median and variance of the sample. Plot a histogram of the simulated data using the function `hist`. Repeat the above (using different variable names!) for a uniform distribution bounded between 0 and 1.
 12. Simulate a sample of size 100 from a Gamma distribution with parameters 3 and 1. Find the mean and standard deviation of the sample. Use the `hist` function to plot a histogram of the variable, adding a suitable title and axis labels. Using the `qgamma`, find the true upper and lower quartiles. Use the `abline` function to draw vertical lines on the histogram representing the quartiles.
 13.
 - Simulate a vector `z` of 1000 $N(0, 1)$ random variables and plot a histogram of these.
 - Create a logical vector `g` indicating those values of `z` which are greater than 1.3.
 - Create a new vector `ztrunc` of all those `z` values which are greater than 1.3. Plot a histogram of these and find their average (mean).
 14. Create a function called `none` which has one parameter `x`, and simply returns the value of `x` sent to it. Run this function, sending the value 3 to it.
 15. Create a function called `mulp` which multiplies two numbers. Use 4 and 8 as inputs to this function.
 16. Write a function called `sum.uniforms` which takes a single argument `n` and returns 1000 numbers, each being the sum of `n` draws from the $\text{uniform}(0,1)$ distribution.
 17. Modify your function `sum.uniforms` so that it only prints a warning message if the user inputs `n < 1`. In such cases it should not proceed to draw the samples or draw a histogram.
 18. Use the `while` function to:
 - (a) Generate a sequence of integers between 0 and 5.

- (b) Generate a sequence of random draws from a $N(0, 1)$ until a generated value is less than 0.7. How many of those draws did you get? Repeat the procedure. Do you get the same number?