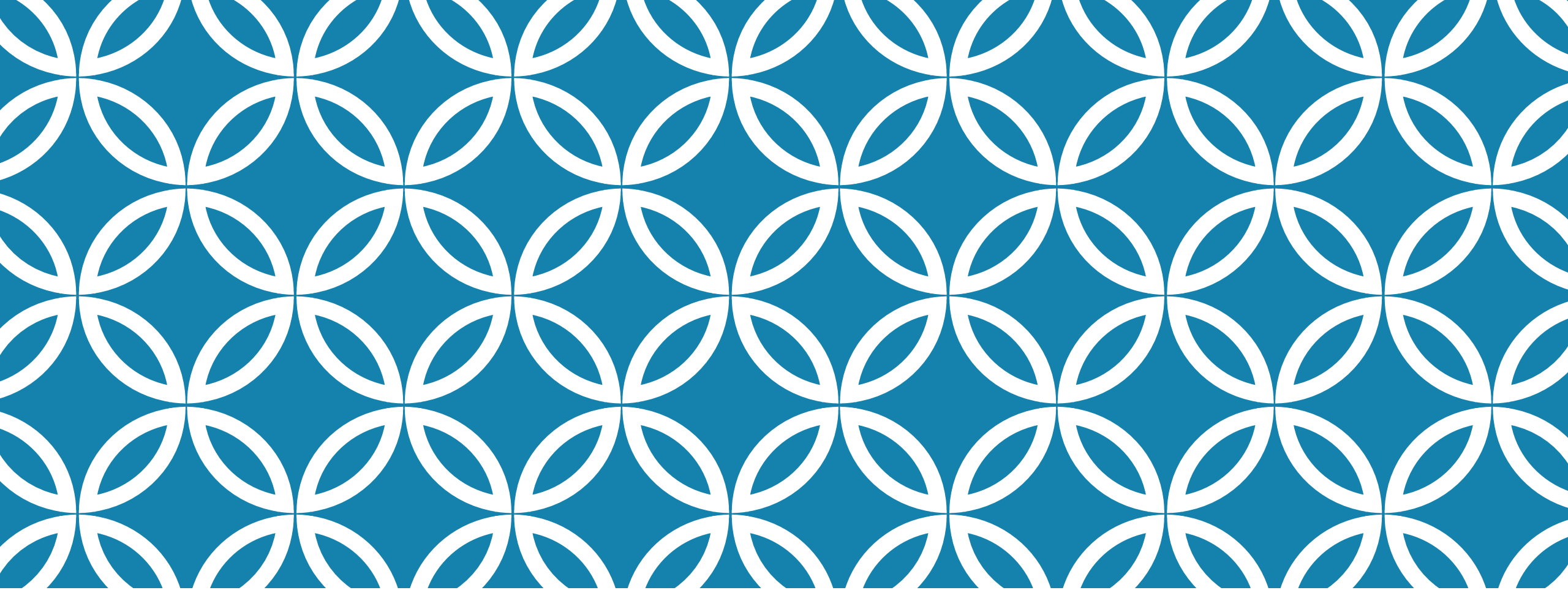
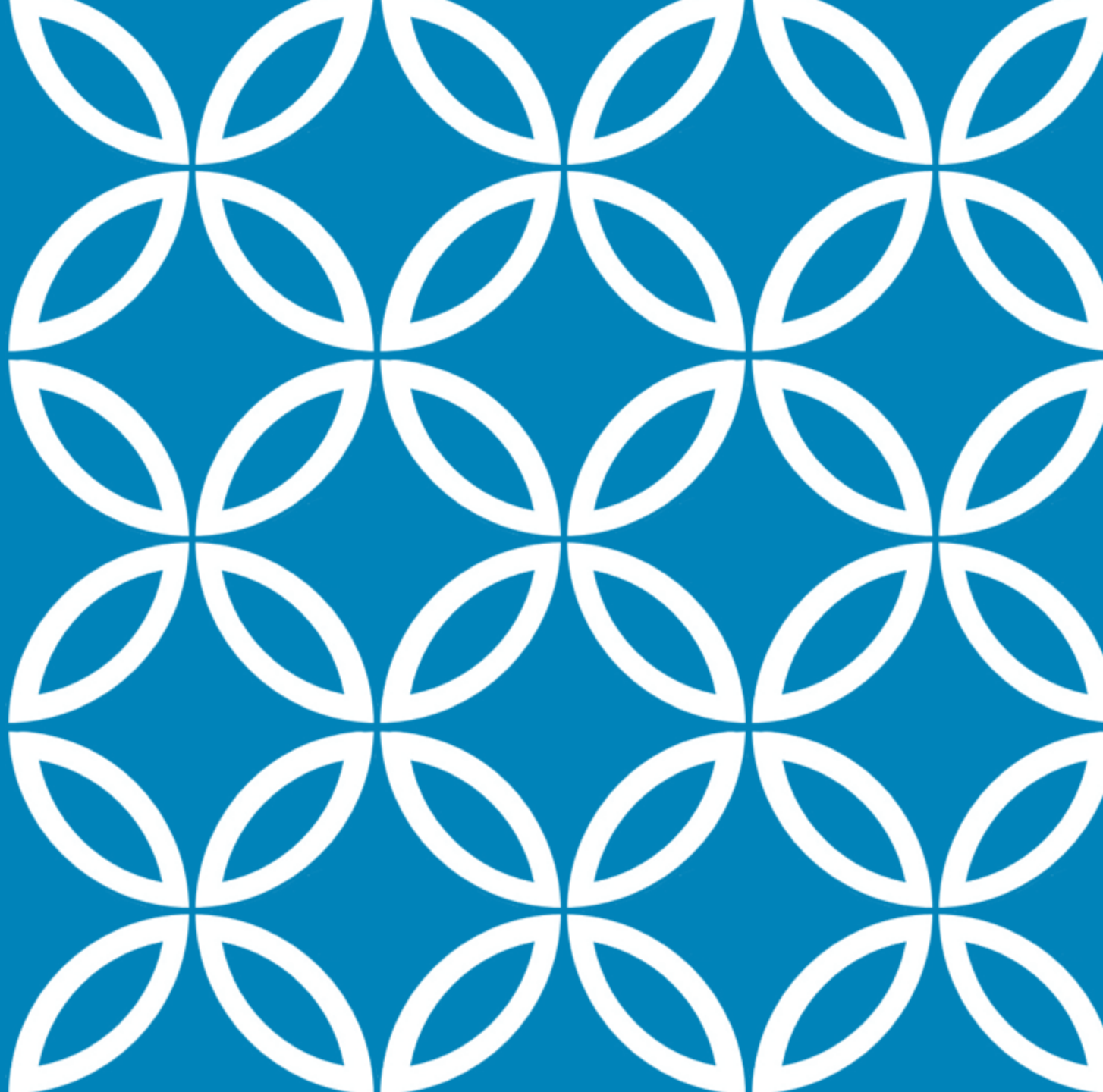


LOOPS AND FUNCTIONS

dr Maria Kubara, WNE UW

WRITING YOUR OWN FUNCTION IN R

User defined function in R (UDF)



FUNCTION – WHAT IS IT?

Function - a sequence of instructions constituting a block of code that can be used repeatedly in different code scripts or coding projects. A function is called by using its name and supplying a list of arguments to it. A function often returns a value when it has completed its operation.

In R we can write and use our own functions!

HOW TO WRITE YOUR OWN FUNCTION - SYNTAX

Defining that 'myRange' will be now a function name

```
myRange <- function(variable){
```

Function 'myRange' takes only one argument - variable

```
  rangeNum <- max(variable) - min(variable)
```

In the body of the function, we need to define which operations are to be executed

```
  return(rangeNum)
}
```

We return the results. After calling 'myRange' we will get the 'rangeNum' of given variable.

INSTRUCTION IF

Condition of *if*. If it returns the value TRUE….

```
if(condition which needs to be checked){

      operation1()
      operation2()

   }
operation3()
operation()
```

… R is executing the code written within the curly brackets

After that we are moving onto the execution of the following code

ELEGANT WAY TO READ PACKAGES (PART 1)

When sharing your script with other people or when trying to make sure that you script works across all R updates, you should make sure, that the installation and loading of packages works automatically.

```
if (!require("pacman")) install.packages("pacman")

pacman::p_load(package1, package2, package_n)
```

Another way with the librarian package, which works with GitHub and Bioconductor repos as well is discussed here: https://statsandr.com/blog/an-efficient-way-to-install-and-load-r-packages/#inefficient-way-to-install-and-load-r-packages

ELEGANT WAY TO READ PACKAGES (PART 2)

Using if statement:

You may also write your own if statement or functions for automation

The most efficient solution though, will be to use pacman or librarian ☺

```
if(!require(somepackage)){
    install.packages("somepackage")
    library(somepackage)
}
```

Creating a function which works for packages:

LOOPS

Loops allow you to automatically execute repeatable operations in R.

Many loop versions in R:

- For
- While
- Repeat

Details: *https://www.geeksforgeeks.org/loops-in-r-for-while-repeat/*

The most popular is the **for** loop.

You can manipulate the flow of your loop execution with next and break statements. Next allows you to skip to the next iteration, while break makes you exit the execution altogether.

FOR LOOP

for (value in sequence){
operation
}

Examples:

```
for(i in 1:10){
   print(i)
}

for(index in seq(from=2, to=10, by=2){
   print(index)
}
```