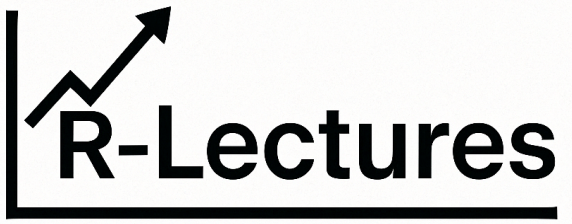


# Quantitative Methods Edition

Timur Öztürk

**Lecture 1: Getting Started with R**

April 25, 2025

# Introduction

- **What is R?** R is a programming language and software environment for statistical computing and graphics. It is a free software environment for statistical computing and graphics. The R language is widely used among economists and data scientists for data analysis, visualization, and statistical modeling. R is an open source project supported by the R Foundation for Statistical Computing. The R language is widely used among economists and data scientists for data analysis, visualization, and statistical modeling.

- **What do we do today?**:
  - We will install `R` and `RStudio`. (This is done for you in the computer labs.)
  - We will talk about the `RStudio` interface.
  - We will learn about OOP, functions, and packages.
  - We will learn how to load and plot spatial data in R.

# Understanding the Basics

Let's start with the basic uses of R. Our learning outcomes for this section are:

- **Calculator.** We could use it as a calculator.
- **Objects.** We could assign and manipulate objects.
  - These objects could be anything: numbers, strings, vectors, matrices, data frames, etc.
  - We will mostly use dataframes, vectors and polygons.
- **Functions.** We could use functions to manipulate objects.
- **Packages.** We could use packages to extend the functionality of R.
  - Packages are collections of functions. They are going to be our main tools for spatial analysis.

# Understanding the Basics

- Internet is your friend; we encourage you to use it.
  - Google, StackOverflow, ChatGPT will help you.
- If you are complete beginner to R and/or programming, or looking at these slides later, check this website out! `https://moderndive.com/1-getting-started.html`
- RStudio also has an introduction: `https://education.rstudio.com/learn/`

# Basics of R: Basic Calculations

```r
# Basic calculations in R
2 + 3
4 * 5
timur <- 10
numbers <- c(1, 2, 3, 4, 5)
object <- c("timur", TRUE, 4)
standard_deviation <- sd(numbers)
sigma_sq <- var
quantile(numbers)
hist(numbers)

cat("Total: ", total, "\n")
cat("Difference: ", diff, "\n")
cat("Product: ", product, "\n")
cat("Quotient: ", quot, "\n")
```

# Basics of R: Functions

```r
# Function to calculate the mean of a vector of numbers.
my_mean <- function(x) {
  sum(x) / length(x)
}

# Now, call the function with a vector of numbers.
my_vector <- c(1, 2, 3, 4, 5)
result <- my_mean(my_vector)
# You can display the result again, try in the console by writing result.
```

# Basics of R: Data Types

```r
# Data Types

x           <- 10                       # numeric
x_num       <- 10L                      # integer literal
x_chr       <- "Gandalf"                # character
x_logical   <- TRUE                     # logical

print(class(x_logical))
```

# Basics of R: Data Structures

```
# Structuring Data
vec     <- c(1, 2, 3, 4, 5)                    # atomic vector
lst     <- list("apple", 10, TRUE)             # heterogeneous list
mat     <- matrix(1:9, nrow=3)                 # 3x3 matrix
arr     <- array(1:24, dim=c(2,3,4))           # 3D array
fctr    <- factor(c("yes","no","yes","yes"))   # categorical
df      <- data.frame(Name=c("Frodo","Sam","Merry"),
Age=c(51,39,36),
City=c("Shire","Shire","Shire"))
```

# Basics of R: Control Flows

```
a <- 5

if (a > 10) {
 message("a is greater than 10")
} else {
 message("a is not greater than 10")
}

for (i in 1:5) message(i)
i <- 1
while (i <= 5) {
 message(i)
 i <- i + 1
}
```

# Basics of R: Data Manipulation

```r
# How would your excel files look like in R?
library(dplyr)
data <- data.frame(
  ID   = 1:5,
  Name = c("Aragorn", "Frodo", "Sam", "Legolas", "Gimli"),
  Age  = c(88, 51, 39, 2931, 140)
)
# Some basic operations, like in Excel.
filtered_data <- data %>%
  mutate(Group = if_else(Age < 40, "Young", "Old")) %>%
  filter(Group == "Young")
print(filtered_data)
```

# Basics of R: Data Visualization

```
# What if you want to do some basic plots?
library(ggplot2)

# Create a data frame for plotting
library(ggplot2)
plot_df <- data.frame(x = 1:5, y = 2 * (1:5))

# static scatter + histogram
ggplot(plot_df, aes(x, y)) + geom_point() + labs(title="Scatter")
ggplot(plot_df, aes(x))   + geom_histogram(bins = 5) + labs(title =
    "Histogram")

# bar chart example
bar_df <- data.frame(Category = LETTERS[1:5], Value=c(10, 25, 15, 30, 20))
ggplot(bar_df, aes(Category, Value, fill = Category)) +
  geom_col(width = 0.7) + theme_minimal() + labs(title = "Bar Plot")
```