R Control Flow Statements: Building blocks for automated Decision Making

Hoda Haeri (Matriculation:400963829)

Hochschule Fresenius - University of Applied Science

Author Note

The authors have no conflicts of interest to disclose.

Correspondence concerning this article should be addressed to Hoda Haeri (Matriculation:400963829), Email: haeri.hoda@stud.hs-fresenius.de

Abstract

R is a powerful programming language widely used for data analysis and visualization. Control flow statements in R—such as if, else, for, while, and repeat—allow users to automate decision-making and repetitive tasks. These statements are the core building blocks that enable scripts to respond to data, adapt to changing situations, and streamline complex analytical processes. This document provides a clear overview of the main control flow statements in R, their syntax, and practical examples to illustrate their role in automating decision-making.

Keywords: R programming, Control flow statements, Automation, Data analysis, Decision-making

R Control Flow Statements: Building blocks for automated Decision Making

Table of contents

R Control Flow Statements: Building blocks for automated Decision Making

R is a widely used language for statistical analysis, data manipulation, and visualization. One of its most powerful features is the ability to automate decisions and repetitive tasks through control flow statements, such as if, else, for, while, and repeat. These statements act like logic gates in a program, directing how and when different parts of the code are executed, based on specific conditions or data values. This approach allows R scripts to efficiently manage various types of data, respond to unexpected situations like missing values, and repeat processes without manual effort, making analysis more robust and scalable

What Are Control Flow Statements in R?

Control flow statements in R work much like traffic signals, determining which sections of code should run, depending on the data or situation. By using these statements, programs can automatically make decisions, choose different paths of analysis, and repeat tasks as needed. This makes R code not only dynamic and efficient but also reliable, even when faced with real-world data challenges

Types of Control Flow Statements

1- If, Else If, and Else Statements:

If statements execute a block of code only if a specified condition is true.

Else statements execute **if** the if condition is false.

Else if statements check further conditions if the previous ones are not met.

2- Repeat Loops:

Repeat loops execute a block of code multiple times until a certain condition is met, using a break statement to stop the loop.

3- While Loops:

While loops continue to run as long as a specific condition remains true, making them useful for processing unknown or varying amounts of data.

4- For Loops:

For loops repeat a block of code a set number of times, usually once for each value in a

vector or list, making them ideal for systematic, repeated tasks.

Why Are Control Flow Statements Important?

Control flow statements are essential for making R scripts flexible, automated, and intelligent. They enable automation in common tasks such as data cleaning, handling missing values, and applying decision rules. This means R can process and analyze data on its own, reducing the need for manual oversight

Conclusion

Control flow statements are the building blocks that transform simple R scripts into dynamic, responsive tools. By using statements like if, else, for, while, and repeat, R users can automate decisions and repetitive processes, making their data analysis workflows more powerful and efficient. This adaptability is crucial for real-world data analysis, where data and requirements often change

Affidavit

I hereby affirm that this submitted paper was authored unaided and solely by me. Additionally, no other sources than those in the reference list were used. Parts of this paper, including tables and figures, that have been taken either verbatim or analogously from other works have in each case been properly cited with regard to their origin and authorship. This paper either in parts or in its entirety, be it in the same or similar form, has not been submitted to any other examination board and has not been published.

I acknowledge that the university may use plagiarism detection software to check my thesis. I agree to cooperate with any investigation of suspected plagiarism and to provide any additional information or evidence requested by the university.

Checklist:

- ☒ The handout contains 3-5 pages of text.
- ☒ The submission contains the Quarto file of the handout.
- ☒ The submission contains the Quarto file of the presentation.
- ☒ The submission contains the HTML file of the handout.

- ☒ The submission contains the HTML file of the presentation.
- ☒ The submission contains the PDF file of the handout.
- ☒ The submission contains the PDF file of the presentation.
- ☒ The title page of the presentation and the handout contain personal details (name, email, matriculation number).
- ☒ The handout contains a abstract.
- ☒ The presentation and the handout contain a bibliography, created using BibTeX with APA citation style.
- ☒ Either the handout or the presentation contains R code that proof the expertise in coding.
- ☒ The handout includes an introduction to guide the reader and a conclusion summarizing the work and discussing potential further investigations and readings, respectively.
- ☒ All significant resources used in the report and R code development.
- ☒ The filled out Affidavit.
- ☒ A concise description of the successful use of Git and GitHub, as detailed here: https://github.com/hubchev/make_a_pull_request.
- ☒ The link to the presentation and the handout published on GitHub.

[Hoda Haeri,] [06/18/2025,] [Köln]