# STAT 471: Introduction to R Programming Lecture

## Lecture 2: Loops and Writing Functions

Cory Suzuki

Department of Mathematics & Statistics
California State University, Long Beach

27 August 2025

# Overview

1. Loops

2. Writing Functions & Built-In Functions

# For Loops

In math and stats, we have dealt with summations in the past. For example,

**Summation**

$$\sum_{i=1}^{10} i = 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10 = 55$$

- Notice how this sum iterates addition each time (starts at 1, increases by 10 up until 10)
- This is exactly how a loop works!

# For Loops Continued

This is a for loop in R. Believe it or not, it does the same exact thing as the summation above!

```
sum = 0
for (i in 1:10) {
sum = sum + i
}
sum
```

Test it out in R!

# For Loops Continued and Syntax

The great thing about for loops is that they are iterative, they perform the same operation until you stop it.

```
for (i in __:__) {
Code goes here
}
```

- i is the loop index, you can also use different letters, but not numbers.
- The first space is the starting index, and the second space is the ending index

# Example: Sorting a Vector without Built-In Functions

# While Loops

While loops are iterative as well, however they are written differently and the logic is dependent on the conditions you define.

- These conditions are sometimes controlled with logical values like TRUE or FALSE
- For loops will automatically end once the maximum number of iterations are reached, while loops may run forever until a condition is reached

# While Loop Example

```
while (flag == TRUE) {
code goes here
flag = FALSE
}
```

- The flag variable here is set to TRUE at first to continue the loop
- The flag variable being set to FALSE will end the loop

# Calculating Sum of Elements in a Vector (By hand) [Example 1]

# Calculating Accuracy of a Machine Learning Model (By hand) [Example 2]

# General Way of Writing Functions

Functions can organize multiple lines of code if you are handling multiple sub-tasks within a single task. There are built-in functions in R that simplify tasks, such as mean(), paste(), min(), max(), median(), etc. You can also create your own customized functions with the below format.

```
function_name = function(...) {
code goes here
}
```

# Writing a Function to Calculate Mean and Variance of a Vector [Example 1]

# Writing a Function to Standardize a Vector [Example 2]

# Next Time...

Bisection and Newton-Raphson Algorithms (Application to Computational Statistics)

# Announcements

Programming Assignment 1 is due this Saturday at 11:59pm!