Transforming Data in R

2025-01-28

Contents

Reading:

Required Reading (everyone):

- R for Data Science, Ch. 2–3.

Reading Notes:

Overview of the reading

- See the homework assignment for the exercises you whould do.

Chapter 2

This very short chapter goes over the basics of coding in R. It discusses creating variables to store data or other information, and calling functions to transform data.

Chapter 3

This chapter is longer and gets into how we organize data in R. There are many ways to organize and work with data, and for this book and this course, we will follow Hadley Wickham's principles of tidy data. The principles of tidy data are explained in section 5.2, which we will read for Thu., Jan. 30.

This chapter focuses on the `dplyr` package, which is part of the larger `tidyverse` package.

The `dplyr` package (pronounced "Dee-Plier", like "pliers") defines a data structure called a `tibble` (which stands for "Tidy Table"), and a suite of functions for transforming `tibbles`. A `tibble` is a variation on an R `data.frame`, and you can mostly consider the two to be the same.

In a `tibble`, each row represents a different measurement or observation, and it contains columns, which represent the different variables you record for each observation:

The `dplyr` package defines many functions to manipulate and transform `tibbles`. The most important for this chapter are: * `filter` to create a new `tibble` by choosing certain rows that match conditions given in `filter`. * `arrange` to rearrange a tibble by putting the rows in order of different columns, from smallest to largest or largest to smallest. * `select` to create a new `tibble` by selecting a subset of the columns, and optionally renaming and re-ordering them.

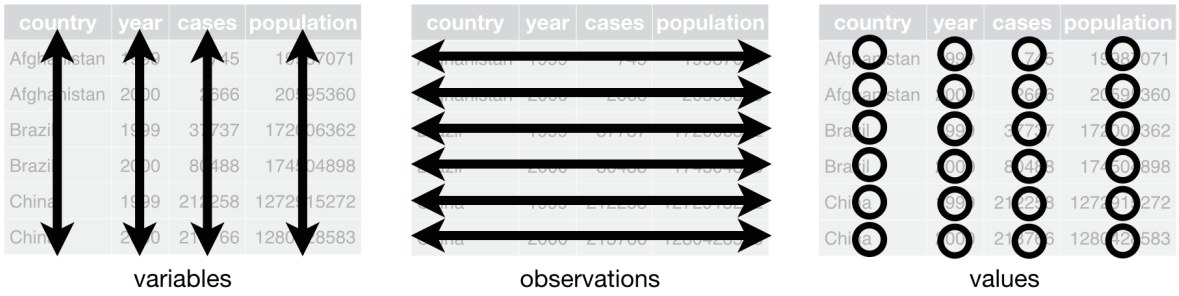

Figure 1: Diagram showing the structure of a `tibble`

* `rename` and `rerlocate` can change the names of columns or the order of columns in the table. * `mutate` to change the values of certain columns or create new columns. You can use other columns to set the value of the new column to things like the sum, difference, product, ratio, or other functions of other columns. * The pipe `|>` lets you easily connect multiple functions to create a new `tibble` from a combination of these functions. * `group_by` and `ungroup` let you perform operations on groups of rows in a `tibble`. For instance, if a `tibble` contains measurements of rocks retrieved from five different sites, you can use `group_by` to report the average silica content of rocks at each site. * `summarize` lets you generate summary statistics, such as mean, median, standard deviation, maximum, or minimum, for selected columns. * `slice` functions let you select certain rows from a `tibble`, such as the 5 rows with the greatest values for a certain column.

There is a lot more to `dplyr`, but this will get us started.