STATS 32: Variable Types

Kenneth Tay

This document describes the different types of variables one might encounter in different contexts. It is not meant to be a comprehensive reference; it only contains enough information to get by in this class.

Before we talk about types… what are variables in the first place?

In the programming context, you can think of a variable as an "envelope" or "bucket" where information can be maintained and referenced.[1] Each bucket has a name on the outside, and contains information on the inside. When we want to refer to a bucket, we use the name on the outside, *not* the information on the inside.

What does the type of a variable refer to?

The type of a variable means different things in different contexts.

Types of variables (Statistics)

In statistics, there are broadly 2 types of variables:

Numerical variables: Numbers which should be treated as they usually are in mathematics. For example, age and weight would be considered numerical variables, while phone number and ZIP code would not be considered numerical variables. There are 2 types of numerical variables:

- **Continuous variable:** A numerical variable that can take values on a continuous scale (e.g. age, weight).
- **Discrete variable:** A numerical variable that only takes on whole numbers (e.g. number of visits).

For R, the distinction between continuous and discrete variables is not an important one.

Categorical variables: Variables which should not be treated like numbers (as in mathematics), and whose values come from a list of possibilities.

- **Nominal variable:** A categorical variable where the categories do not have a natural ordering (e.g. gender, ethnicity, country).
- **Ordinal variable:** A categorical variable where the categories have a natural ordering (e.g. age group, income level, educational status).

Types of variables (Programming)

In the programming context, a variable's type defines what operations the program can do with it, as well as the specifics of that operation.

These are the common variable types that we see across programming languages:

[1] https://www.cs.utah.edu/~germain/PPS/Topics/variables.html

- **Integer:** …, 2, -1, 0, 1, 2, ...
- **Double:** Real numbers. This is sometimes referred to as the “float” data type as well.
- **Character:** This is what we commonly think of as text.
- **Boolean:** This has only 2 possible values: TRUE or FALSE.

Types in R

This is where things start to get confusing! As R is a programming language written “by statisticians for statisticians”, some of the terminology for types can get mixed up. Here are the types in R:

- **Numeric:** This matches with numerical variables in the statistics context and both the integer and double types in the programming context.
- **Character:** This matches with the character type in the programming context. It does not match with anything in the statistics context.
- **Boolean:** This matches with the Boolean type in the programming context. In the statistics context, Boolean variables are considered categorical variables (nominal or ordinal depends on the context).
- **Factor:** This matches with categorical variables in the statistics context. It does not match with anything in the programming context. **Factor variables are unique to R.**

How do I decide what type a variable should have in R?

The main confusion is typically between numeric variables and character variables which have digits, and between character variables and factor variables.

To differentiate between numeric variables and character variables which have digits, ask yourself if we should treat the variable like a number as we do in mathematics. Does it make sense to add two of them together? Does it make sense to take the sum or the mean of this variable? Does it make sense to compare them with > and < operators?

To differentiate between character variables and factor variables, ask yourself if you are trying to model some other variable based on the value that this variable takes. If you are, it should be a factor variable, otherwise it should be a character variable. For example, if ZIP code happens to be in your dataset but you’re not using it for a model, it is OK to leave it as a character variable. If you are using it in a model (e.g. to predict weather), then it should be a factor variable.

Why do I need to know about types in R?

While R generally does a good job of guessing what type your variable should be, it sometimes gets it wrong. For example, if your dataset contains phone numbers simply as a string (e.g. 6507231111, as opposed to 650-723-1111 or (650)-723-1111), R will interpret these as numeric variables instead of character variables. To change that, you will have to use the `as.character()` function. The process of changing the type of a variable is called **coercion**.