3.3: Getting Started with R

When we work with R, we often do this using a *command line* in which we type commands and it responds to those commands. In the simplest case, if we just type in a number, it will simply respond with that number. Go into the R console and type the number 3. You should see somethign like this:

```
> 3
[1] 3
```

The `>` symbol is the *command prompt*, which is prompting you to type something in. The next line (`[1] 3`) is R's answer. Let's try something a bit more complicated:

```
> 3 + 4
[1] 7
```

R spits out the answer to whatever you type in, as long as it can figure it out. Now let's try typing in a word:

```
> hello
Error: object 'hello' not found
```

What? Why did this happen? When R encounters a letter or word, it assumes that it is referring to the name of a *variable* — think of `X` from high school algebra. We will return to variables in a little while, but if we want R to print out the word *hello* then we need to contain it in quotation marks, telling R that it is a *character string*.

```
> "hello"
[1] "hello"
```

There are many types of variables in R. You have already seen two examples: integers (like the number 3) and character strings (like the word "hello"). Another important one is *real numbers*, which are the most common kind of numbers that we will deal with in statistics, which span the entire number line including the spaces in between the integers. For example:

```
> 1/3
[1] 0.33
```

In reality the result should be 0.33 followed by an infinite number of threes, but R only shows us two decimal points in this example.

Another kind of variable is known as a *logical* variable, because it is based on the idea from logic that a statement can be either true or false. In R, these are capitalized (`TRUE` and `FALSE`).

To determine whether a statement is true or not, we use *logical operators*. You are already familiar with some of these, like the greater-than (`>`) and less-than (`<`) operators.

```
> 1 < 3
[1] TRUE
> 2 > 4
[1] FALSE
```

Often we want to know whether two numbers are equal or not equal to one another. There are special operators in R to do this: `==` for equals, and `!=` for not-equals:

```
> 3 == 3
[1] TRUE
> 4 != 4
[1] FALSE
```