Assessment 01 - Data Types

Alessandro Corradini - Harvard Data Science Professional

Variable names

The type of data we are working with will often influence the data visualization technique we use. We will be working with two types of variables: categorical and numeric. Each can be divided into two other groups: categorical can be ordinal or not, whereas numerical variables can be discrete or continuous.

We will review data types using some of the examples provided in the `dslabs` package. For example, the `heights` dataset.

```r
library(dslabs)
data(heights)
```

Instructions

Let's start by reviewing how to extract the variable `names` from a dataset using the names function. What are the two variable names used in the `heights` dataset?

```
library(dslabs)
data(heights)
names(heights)
```

Variable type

We saw that sex is the first variable. We know what values are represented by this variable and can confirm this by looking at the first few entires:

```
library(dslabs)
data(heights)
head(heights)
```

What data type is the sex variable?

Instructions

Possible Answers

- Continuous
- Categorical [X]
- Ordinal
- None of the above

Numerical values

Keep in mind that discrete numeric data can be considered ordinal. Although this is technically true, we usually reserve the term ordinal data for variables belonging to a small number of different groups, with each group having many members.

The `height` variable could be ordinal if, for example, we report a small number of values such as short, medium, and tall. Let's explore how many unique values are used by the heights varialbe. For this we can use the unique fuction:

```
x <- c(3, 3, 3, 3, 4, 4, 2)
unique(x)
```

Instructions

Use the `unique` and `length` functions to determine how many unique heights were reported.

```r
library(dslabs)
data(heights)
x <- heights$height
length(unique(x))
```

```
## [1] 139
```

Tables

One of the useful outputs of data visualization is that we can learn about the distribution of variables. For categorical data we can construct this distribution by simply computing the frequency of each unique value. This can be done with the function `table`. Here is an example:

```
x <- c(3, 3, 3, 3, 4, 4, 2)
table(x)
```

Instructions

Use the `table` function to compute the frequencies of each unique height value. Because we are using the resulting frequency table in a later exercise we want you to save the results into an object and call it `tab`.

```r
library(dslabs)
data(heights)
x <- heights$height
tab <- table(x)
```

Indicator variables

To see why treating the reported heights as an ordinal value is not useful in practice we note how many values are reported only once.

Instructions

In the previous exercise we computed the variable `tab` which reports the number of times each unique value appears. For values reported only once `tab` will be 1. Use logicals and the function `sum` to count the number of times this happens.

```r
library(dslabs)
data(heights)
tab <- table(heights$height)
sum(tab == 1)
```

```
## [1] 63
```

Data types - heights

Since there are a finite number of reported heights and technically the height can be considered ordinal, which of the following is true:

Instructions

Possible Answers

- It is more effective to consider heights to be numerical given the number of unique values we observe and the fact that if we keep collecting data even more will be observed. [X]
- It is actually preferable to consider heights ordinal since on a computer there are only a finite number of possibilities.

- This is actually a categorical variable: tall, medium or short.
- This is a numerical variable because numbers are used to represent it.