

# String Activity

Abbie M Popa

9/12/2018

## Factors

### Preamble

The Oakland street trees data set contains information about trees that grow in the city of Oakland California. Download the data from the github page, then import the file into your current R session using the following line of code, you will need to change “dir” to whichever directory you saved your data in.

```
trees <- read.csv("dir/oakland-street-trees.csv")
```

### Making LOWWELL Ordinal

The column in the tree dataset “LOWWELL” contains information about the size of “well” for the particular tree. I don’t know much about (arboreal) trees, but I assume this has to do with how big the tree is. Examine this factor with the following code:

```
levels(trees$LOWWELL)
```

```
## [1] "High"      "Low"       "Moderate"  "None"
```

```
table(trees$LOWWELL)
```

```
##
```

```
##      High      Low Moderate      None
##      796     5995     2328     29489
```

Is the LOWWELL column ordered? Does the default sorting make sense?

Change the LOWWELL column so that it is an ordered factor by filling in the blank (\_\_\_\_\_) in the following code and removing the # sign. None < Low < Moderate < High

```
# trees$LOWWELL <- factor(as.character(trees$LOWWELL), ordered = T, levels = c(_____))
```

Copy the code from the code block named “looking at lowwell” to see if your re-ordering did what you think it did. If necessary, fix what you put in the blank above.

Use comparison operators (e.g., “<”) to find how many trees need a smaller than “moderate” well.

## Strings

### Preamble

The Questions dataset contains questions that were asked about the R programming language on the popular online forum stack overflow. Download the data from the github page, then import the file with all the R questions into your current R session using the following line of code, you will need to change “dir” to whichever directory you saved your data in.

```
questions <- read.csv("dir/Questions.csv")
```

Load the necessary libraries.

```
library(stringr)
library(tibble)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
## The following object is masked from 'package:base':
##
##      date
```

## Tibble-ify

The file is currently a generic data frame. Print the data frame. Convert the data frame to a tibble and print it again. **Important!** Be sure to comment out the code where you print the data before knitting! The data is large enough that if you don't the file will not knit.

You will notice that the columns "Title" and "Body" have been stored as factors. Convert them to character strings.

You will also notice the "CreationDate" column is stored as a factor. Convert this to a date column (you can lose the time information if you want to)

## Questions from the dataset

What is the earliest date in the dataset? What is the latest date in the dataset?

You have a theory that lists are the worst data type in R. Count the number of times "list" appears in the Body column of this data set. Compare it to how many times "vector," "matrix," "array," "data frame," and "tibble" appear in the dataset. What do you conclude?

What is the earliest year "tibble" appears in the dataset? What about data frame? Why might this be?

Based on this information, do you think you are older than the tibble library?