Practice2

Kishore Hari

2022-11-24

This is an example for cleaning data. Download the text file at this link: https://michaelgastner.com/DAVisR_data/homicides.txt. If you open it in a text editor like wordpad, you will see that there is a consistent structure. However, the delimiters are not consistent. Therefore, your task is to convert the text data into a dataframe. You can read the file using the function .

```
x <- read_lines("https://michaelgastner.com/DAVisR_data/homicides.txt")
```

a. *What is the class of x?*

```
class(x)
```

```
## [1] "character"
```

b. *What are the possible delimiters that you see in the data?*

Possible delimiters: comma, “<”, “>”, “<dd>”, “</dd>”, “<dl>”, “</dl>”. Note that a delimiter doesn’t have to be a single character.

```
write_lines(x[1], file = stdout())
```

```
## 39.311024, -76.674227, iconHomicideShooting, ’p2’, ’<dl><dt>Leon
Nelson</dt><dd class="address">3400 Clifton Ave.<br />Baltimore, MD
21216</dd><dd>black male, 17 years old</dd><dd>Found on January 1,
2007</dd><dd>Victim died at Shock Trauma</dd><dd>Cause: shooting</dd></dl>’
```

c. *How many columns can the data be divided into, given that each column must only have one category of information?*

   Minimum 5. I count 12 in total. But there can be more?

d. *Can you use comma as a delimiter to split the data? If you do so, will there by any inconsistencies?*

   There are two ways to do this: using read_csv and splitting the strings based on commas, like we discussed in class.

```
# Method 1: read_csv
df <- read_csv("https://michaelgastner.com/DAVisR_data/homicides.txt",
    col_names = F)
```

```
## Warning: One or more parsing issues, see ‘problems()‘ for details
## Rows: 1249 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (6): X3, X4, X5, X6, X7, X8
## dbl (2): X1, X2
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

e. *Explore the "quote" argument in read_csv. See that it helps reduce the inconsistencies found in the previous question.*

f. *Write a code to split each row in 12 parts: Longitude, Lattitude, Category_of_homicide, case_number, Victim_name, Address, Victim_description, Gender, Victim_age, Date_of_homicide, Place_of_death, Cause_of_death. Hint: Use the function* `str_split`*.*

g. *Using the code above, convert the text file into a dataframe of 12 columns.*

h. *Using a barplot of the gender column, find out which gender has the highest number of homicides? Similarly find out, which cause of death is most common? Which month had the highest number of homicides? (Note that you would need to split the date column into day, month and year to do this. Use the* `separate` *function in tidyr package)*