Lab 8

Ignacio Cabezudo

October 27, 2017

Using your own dataset (which may include more than one table) carry out the following data cleaning steps. Knit together the PDF document and commit both the Lab 8 RMD file and the PDF document to Git. Push the changes to GitHub so both documents are visible in your public GitHub repository.

Before you begin: as many of you have large datasets, you're going to want to select only the variables you're interested in utilizing for this project (ideally no more than twenty columns but perhaps much smaller) so you don't have R Studio's memory working on the entire dataset. The example code provided below can be modified to allow you to subset your data to only the variables you wish to use. First, read in your complete dataset and save it as `data`. Then, add the names of the variables you wish to use for your poster project to the `select` function, separated by commas. Run the two lines of code to save this new, smaller version of your data to `data_subset`. Use this smaller dataset to complete the rest of the lab

```
# Read in your data with the appropriate function
library(tidyverse)
```

```
## Loading tidyverse: ggplot2
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
## Loading tidyverse: dplyr

## Conflicts with tidy packages ----------------------------------------------

## filter(): dplyr, stats
## lag():    dplyr, stats
```

```
library(readr)
library(ggplot2)

ACLED <- read_csv("ACLED-AFRICA-FULL.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   GWNO = col_integer(),
##   YEAR = col_integer(),
##   TIME_PRECISION = col_integer(),
##   INTER1 = col_integer(),
##   INTER2 = col_integer(),
##   INTERACTION = col_integer(),
##   LATITUDE = col_double(),
##   LONGITUDE = col_double(),
##   GEO_PRECISION = col_integer(),
##   FATALITIES = col_integer()
## )

## See spec(...) for full column specifications.

## Warning in rbind(names(probs), probs_f): number of columns of result is not
## a multiple of vector length (arg 2)

## Warning: 1 parsing failure.
## row # A tibble: 1 x 5 col     row       col expected     actual                    file expected   <
```

```r
ACLED_parsed <- ACLED %>%
  select(EVENT_DATE, COUNTRY, YEAR, FATALITIES, LOCATION, LONGITUDE, LATITUDE)
ACLED_parsed <- dplyr::filter(ACLED_parsed, FATALITIES > 0) # Removes 0 Fatality events
#as.Date(ACLED_parsed$EVENT_DATE, "%m/%d/%Y") #make sure date is set correctly
#clearly something is wrong with this. Not sure what's going on
```

1. To get a feel for its structure, look at the class, dimensions, column names, structure, and basic summary statistics of your data.

```r
summary(ACLED_parsed)
```

```
##   EVENT_DATE          COUNTRY               YEAR        FATALITIES
##  Length:4829        Length:4829        Min.   :1997   Min.   :  1.000
##  Class :character   Class :character   1st Qu.:2017   1st Qu.:  1.000
##  Mode  :character   Mode  :character   Median :2017   Median :  2.000
##                                        Mean   :2017   Mean   :  5.101
##                                        3rd Qu.:2017   3rd Qu.:  5.000
##                                        Max.   :2017   Max.   :310.000
##    LOCATION           LONGITUDE        LATITUDE
##  Length:4829        Min.   :-16.38   Min.   :-34.053
##  Class :character   1st Qu.: 13.96   1st Qu.:  2.049
##  Mode  :character   Median : 29.47   Median :  6.698
##                     Mean   : 26.98   Mean   :  8.522
##                     3rd Qu.: 41.08   3rd Qu.: 12.113
##                     Max.   : 50.18   Max.   : 37.006
```

2. Preview the first and last 15 rows of your data. Is you dataset tidy? If not, what principles of tidy data does it seem to be violating?

```r
head(ACLED_parsed, 15)
```

```
## # A tibble: 15 x 7
##    EVENT_DATE COUNTRY  YEAR FATALITIES    LOCATION LONGITUDE LATITUDE
##         <chr>   <chr> <int>      <int>       <chr>     <dbl>    <dbl>
##  1   1/1/1997 Algeria  1997          5    Douaouda   2.78940 36.67250
##  2   1/2/1997 Algeria  1997          2    Hassasna   0.88330 36.13330
##  3   1/3/1997 Algeria  1997          2     Algiers   3.04197 36.75250
##  4   1/6/1997 Algeria  1997          4  Ain Benian   2.92185 36.80277
##  5   1/7/1997 Algeria  1997          7 Ain Fakroun   6.87374 35.97108
##  6   1/7/1997 Algeria  1997          1       Jijel   5.76670 36.80000
##  7  1/10/1997 Algeria  1997          1 Bachdjerrah   3.11833 36.72167
##  8  1/11/1997 Algeria  1997          4       Chlef   1.33452 36.16525
##  9  1/13/1997 Algeria  1997          4  Mostaganem   0.08333 35.93333
## 10  1/13/1997 Algeria  1997          1      Larbaa   3.15654 36.58934
## 11  1/14/1997 Algeria  1997          2    Douaouda   2.78940 36.67250
## 12  1/15/1997 Algeria  1997          1     Algiers   3.04197 36.75250
## 13  1/16/1997 Algeria  1997         15    Boufarik   2.91214 36.57413
## 14  1/19/1997 Algeria  1997          6  Mostaganem   0.08333 35.93333
## 15  1/21/1997 Algeria  1997         19     Algiers   3.04197 36.75250
```

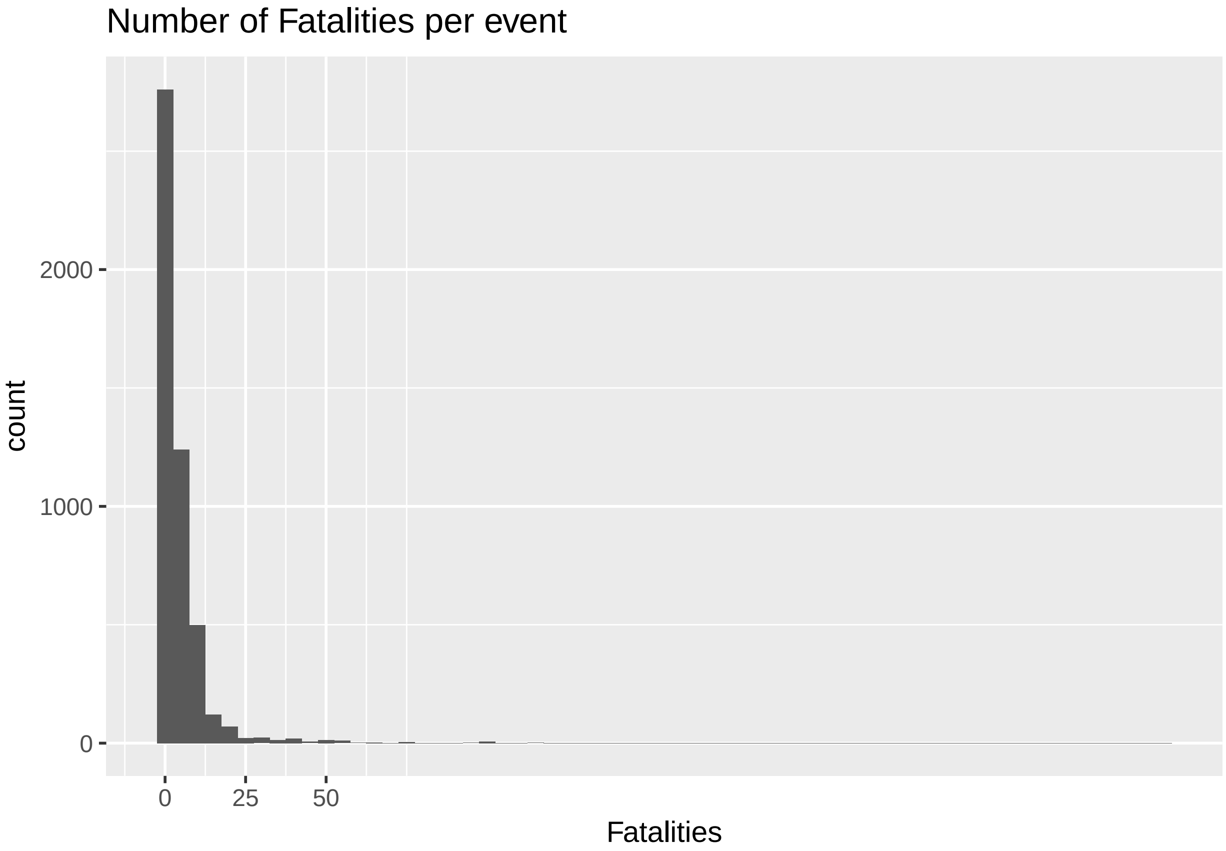
```r
tail(ACLED_parsed, 15)
```

```
## # A tibble: 15 x 7
##    EVENT_DATE  COUNTRY  YEAR FATALITIES    LOCATION LONGITUDE  LATITUDE
##         <chr>    <chr> <int>      <int>       <chr>     <dbl>     <dbl>
##  1   9/7/2017  Tunisia  2017          1     Tabarka   8.75801  36.95442
##  2  9/18/2017  Tunisia  2017          1    Jendouba   8.78024  36.50114
##  3  1/31/2017   Uganda  2017          1        Arua  30.93080   3.01920
##  4  2/28/2017   Uganda  2017          2      Kisoro  29.69830  -1.35390
##  5  3/17/2017   Uganda  2017          2     Kampala  32.56560   0.31560
##  6   4/4/2017   Uganda  2017          1     Kampala  32.56560   0.31560
##  7  4/18/2017   Uganda  2017          5 Chepsikunya  34.53750   1.54670
##  8   6/7/2017   Uganda  2017          3    Adjumani  31.80972   3.36139
##  9   8/1/2017   Uganda  2017          1 Ibanda Town  30.53190  -0.15390
## 10  7/25/2017   Zambia  2017          1    Chinsali  32.06670 -10.55000
## 11  4/18/2017 Zimbabwe  2017          1      Harare  31.05000 -17.83330
## 12  4/21/2017 Zimbabwe  2017          1      Mutare  32.66670 -18.96660
## 13  6/29/2017 Zimbabwe  2017          1      Harare  31.05000 -17.83330
## 14   7/7/2017 Zimbabwe  2017          1    Silobela  29.30000 -18.98330
## 15  7/18/2017 Zimbabwe  2017          1     Marange  32.26670 -19.25000
```

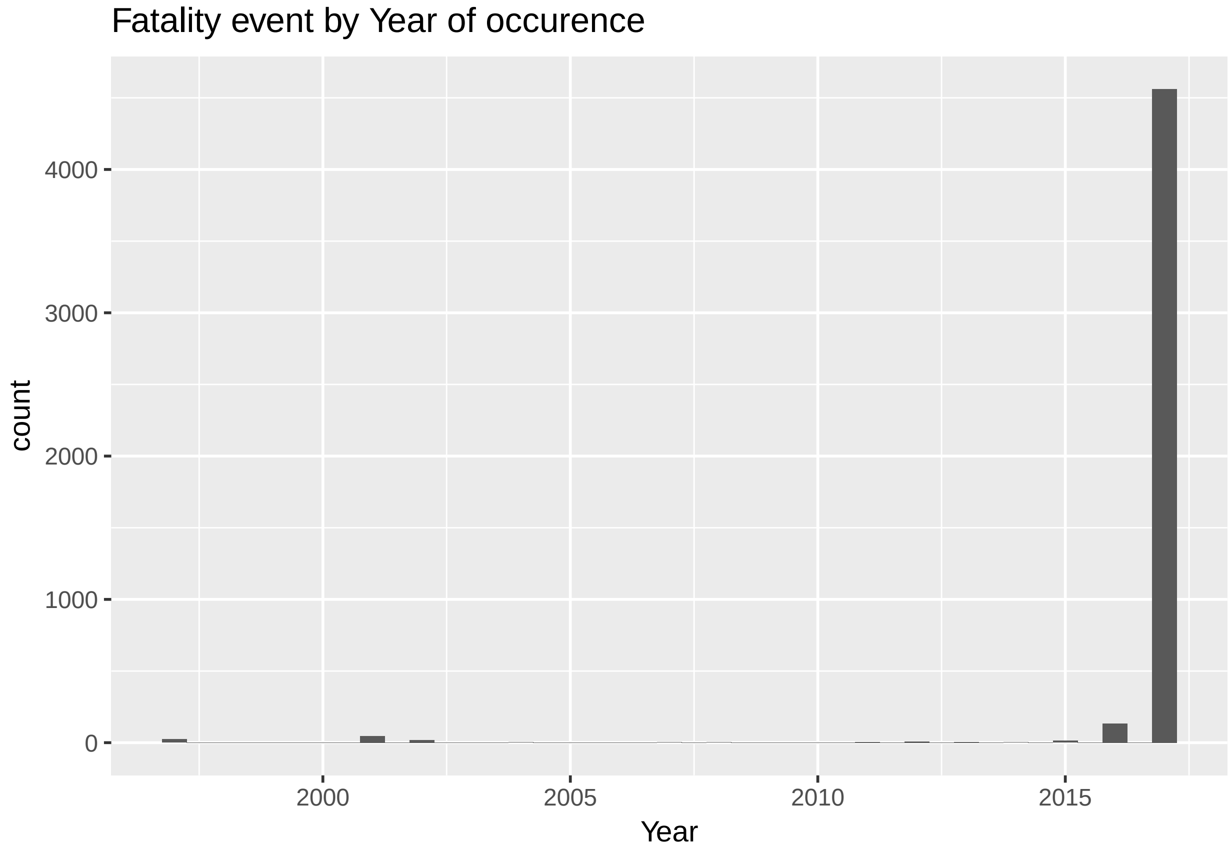
I think my data is quite tidy!

3. Create a histogram for at least two variables you plan to focus on for your study. Describe what these plots show you about these variables.

```
ggplot(ACLED_parsed, aes(FATALITIES)) + geom_histogram(binwidth = 5) + scale_x_continuous(breaks = seq(
```



```r
ggplot(ACLED_parsed, aes(YEAR)) + geom_histogram(binwidth = .5) + scale_x_continuous(breaks = seq(1990,
```

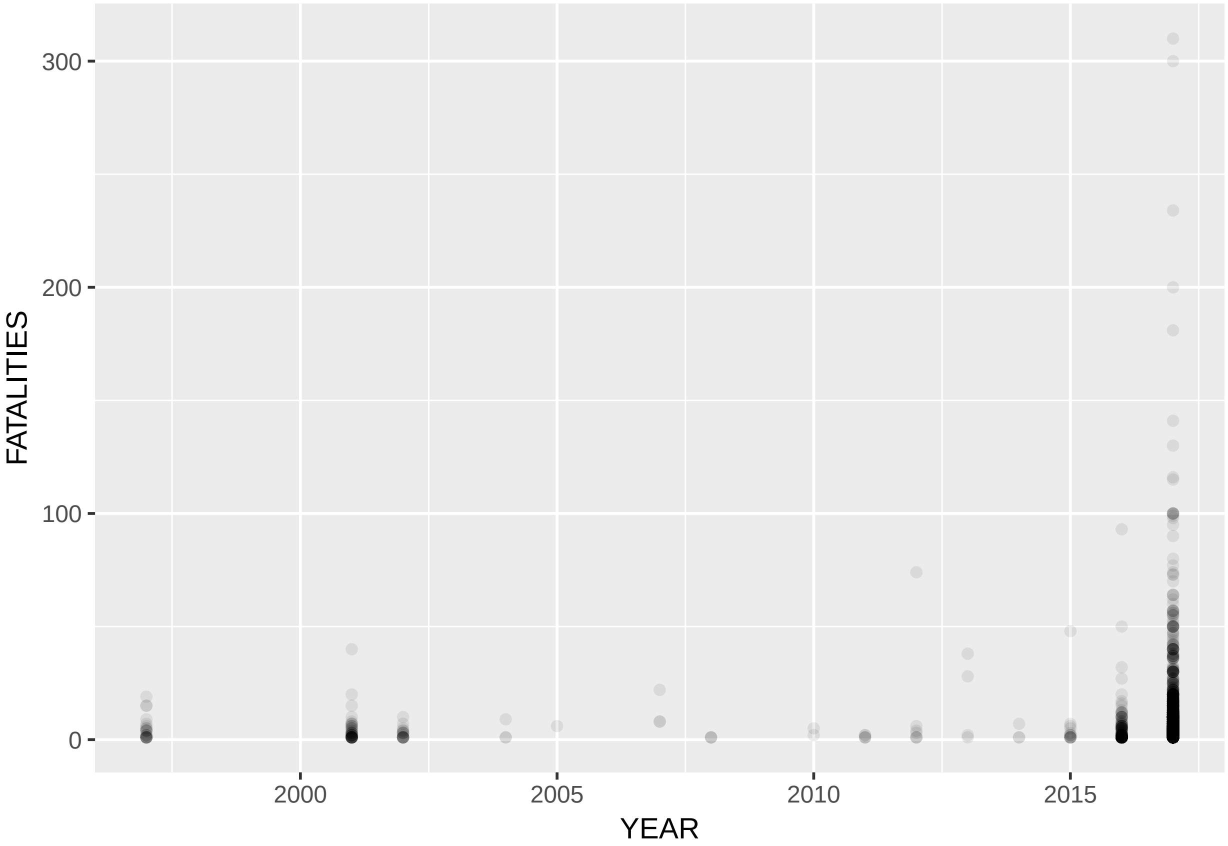


```r
#I'm still working on what I want to do with these

#hist(ACLED_parsed$YEAR, xlim = c(1995, 2018), ylim = c(0, 150), breaks = 50)
```

4. Create at least one bivariate plot showing the relationship between two variables of interest. What does/do the(se) plot(s) tell you about the association between these two variables?

```r
#Still working on trying to make this look better
ggplot(ACLED_parsed, aes(YEAR, y = FATALITIES)) +
  geom_point(alpha = .075)
```



```
#plot(ACLED_parsed$YEAR, ACLED_parsed$FATALITIES)
```

5. Load the `tidyr` package. Do all of your columns correspond to variables? Do any columns represent multiple variables? If your answer is yes to either question, carry out the appropriate `tidyr` function (`gather()` or `spread()` respectively) to tidy your data.

I'm happy with what I've got

6. Do any columns need to be separated into two or more? Do any columns need to be combined into one? If so, carry out the appropriate the appropriate `tidyr` function (`separate()` or `unite()` respectively) to tidy your data.

Not needed for this dataset.

At this stage each row in your data should represent one observation, each column should be a variable, and each table should be observational unit.

7. What is the class of each of the variables in your analysis? Are these classes appropriate for the type of measurement they purport to capture? Explain your reasoning.

character, character, Int, Int, Chr, dbl, dbl

The only one I'm concerned about is Date, as I'm considering breaking the data even further down to by month. In order to do this my idea was to use the commented out as.date() code in the initial cleaning phase, then later use that information to seperate things into monthly. I'm not quite sure how to do that yet, but I don't think it's a difficult operation.

8. Do any of your variables need to be coerced into a different data type? If so, carry out the appropriate coercion methods below. (This includes transformation of any date objects using the `lubridate` package)

Yes! Yes they do!! We just talked about that!

```
#library(lubridate)
```

9. Are there any strings you need to manipulate for your analysis? If so, use the appropriate function from the `stringr` package.

No!

10. Do you have any missing values in your dataset? How many and how are they coded? **Be sure to look out for specific codebook values for missing values (i.e. `-1` for `NA`) as well as empty strings or other software-specific values for `NA`.** Don't worry about removing NAs yet - we'll tackle this question later once discern whether they're random or systematically distributed.

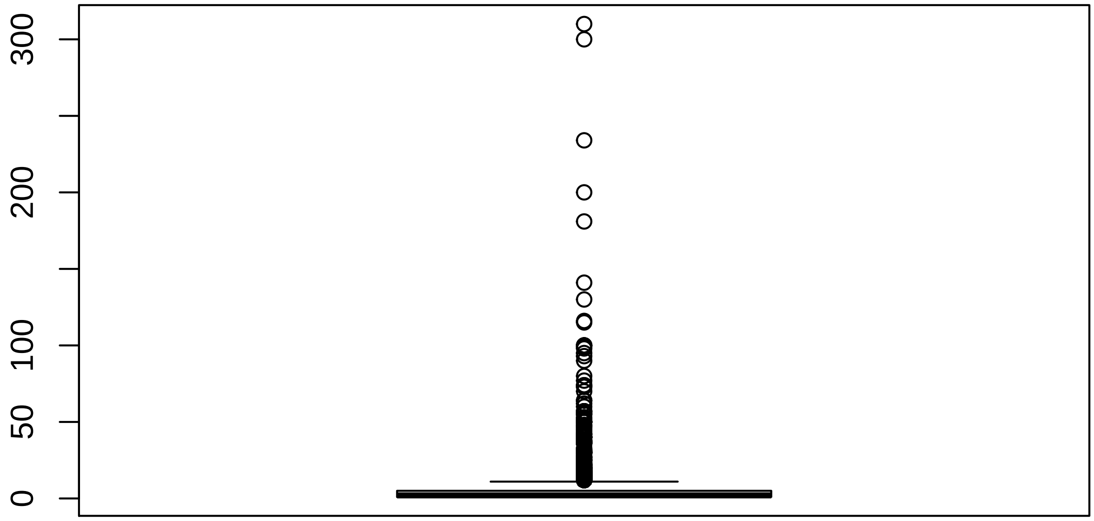
No missing values!

11. Are there any special values in your dataset? If so, what are they and how do you think they got there? *The presence of special values is less likely if you haven't performed any data manipulation yet so you should remember to return to this step each time you carry out a mathematical transformation of any values in your dataset.*

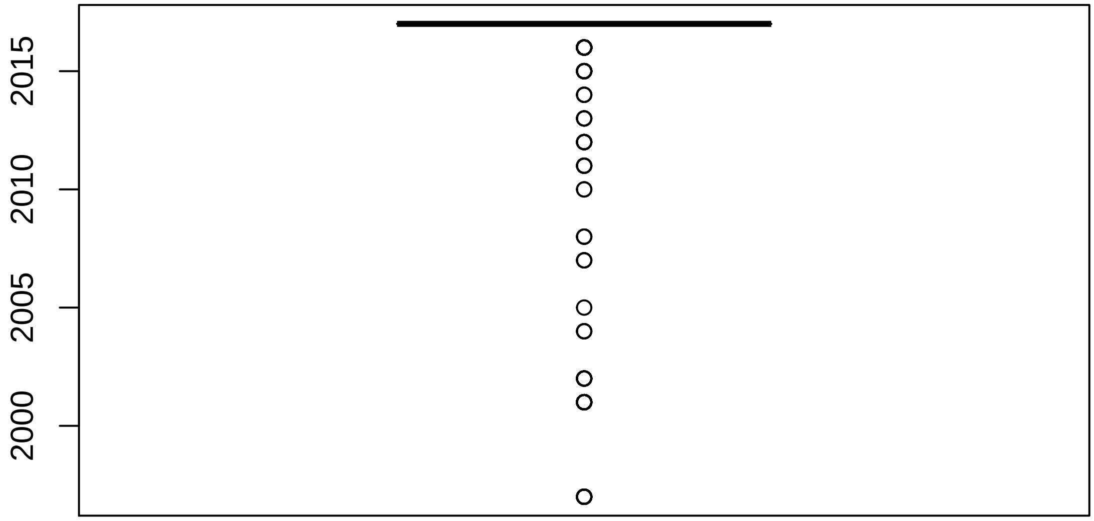
I don't think so.

12. Create a boxplot of your data (you can create an individual boxplot for each variable if there are too many variables in your dataset to meaningfully visualize them all in one plot). Are there any outliers? If so, what are they and to which variable do they correspond? Do any of these outliers seem like obvious errors? If so, why?

```
boxplot(ACLED_parsed$FATALITIES)
```



```r
boxplot(ACLED_parsed$YEAR)
```



13. For any outliers and/or obvious errors, what do you think is the best way to handle them (i.e. remove them entirely, run analyses including and excluding them and compare the results, manually change them to an appropriate measure of center, or something else?).

I have one major outlier. Given that the data I'm looking at is about real people killed in real conflicts, I don't feel right about removing it, I may specifically address it. I haven't decided yet.