

DC for Social Science: Exercises for Day 2

Lachlan Deer

Before We Start

You will need to use the following libraries:

```
library(readr)
library(dplyr)
library(lubridate)
library(tidyr)
library(ggplot2)
```

And you will have needed to download the **SAFI** data.

The code below will download the cleaned SAFI data as a csv file into a subdirectory called **data** if you have not already downloaded the data, otherwise it will print a message saying the data doesn't need to be downloaded.

```
dest_file <- "data/SAFI_clean.csv"
if(!file.exists(dest_file)){
  dir.create('data')
  download.file("https://ndownloader.figshare.com/files/11492171", destfile = dest_file, mode = "wb")
} else {
  message('data already downloaded')
}
```

Load the data that was downloaded as follows:

```
interviews <- read_csv('data/SAFI_clean.csv', na = "NULL")
```

Solutions: Will be made available at the end of the session [here](#).

Part 1 - Working With Data

Exercise 1

- Create a data frame (`interviews_100`) containing only the data in row 100 of the `interviews` dataset.
- Create a new data frame (`interviews_last`) from that last row. (*Hint:* Refer to `?n()` for help)
- Extract the row that is in the middle of the dataset and store the content of this middle row in an object named `interviews_middle`. (*hint:* Use the `median()` function and what you've learned about `n()` to find the middle row!)
- Combine `n()` with `slice()` to reproduce the behavior of `head(interviews)`, keeping just the first through 6th rows of the `interviews` dataset. Name the resulting dataset `interviews_head`.

Exercise 2

- Using pipes, subset the interviews data to include interviews where respondents were members of an irrigation association (`memb_assoc`) and retain only the columns `affect_conflicts`, `liv_count`, and `no_meals`.
- Create a new dataframe from the interviews data that meets the following criteria: contains only the village column and a new column called `total_meals` containing a value that is equal to the total number of meals served in the household per day on average (`no_membrs` times `no_meals`). Only the rows where `total_meals` is greater than 20 should be shown in the final dataframe.
- Use `group_by()` and `summarize()` to find the mean, min, and max number of household members for each village. Also add the number of observations (hint: see `?n`).
- What was the largest household interviewed in each month?

Exercise 3

- Create a new dataframe (named `interviews_months_lack_food`) that has one column for each month and records `TRUE` or `FALSE` for whether each interview respondent was lacking food in that month.
- How many months (on average) were respondents without food if they did belong to an irrigation association? What about if they didn't?

Part 2 - Plotting

Exercise 1

Boxplots are useful summaries, but hide the shape of the distribution. For example, if the distribution is bimodal, we would not see it in a boxplot.

- An alternative to the boxplot is the violin plot, where the shape (of the density of points) is drawn.. Replace the box plot with a violin plot; use `?geom_violin()` for help as needed.
- Create a boxplot for `liv_count` for each wall type. Overlay the boxplot layer on a jitter layer to show actual measurements.
- Add colour to the data points on your previous boxplot according to whether the respondent is a member of an irrigation association (`memb_assoc`).

Exercise 2

- Create a plot that visualizes the proportion of respondents in each village who owned a particular item?
 - *HINT:* To complete this exercise you will need a count of the number of interviewees in each village. The following lines of code will give you this. You need to join new column of data into the existing dataset to proceed (search `?inner_join` to explore one way to do this).

```
village_pop <- interviews_plotting %>%  
  group_by(village) %>%  
  count() %>%  
  rename(people_in_village = n)
```