HW 03 - Nobel laureates

due Tuesday, 9/27 at 11:59pm

Contents

In January 2017, Buzzfeed published an article titled "These Nobel Prize Winners Show Why Immigration Is So Important For American Science". In the article they explore where many Nobel laureates in the sciences were born and where they lived when they won their prize.

In this homework we will work with the data from this article to recreate some of their visualizations as well as explore new questions.

The learning goals of this lab are:

- Manipulate and transform data to prepare it for visualization.
- Recreate visualizations.
- Summarize data.

Getting started

Go to the course GitHub organization and locate your HW 03 repo, which should be named `hw-03-nobel-[GITHUB USERNAME]`. Grab the URL of the repo, and clone it in RStudio. Refer to HW 01 for step-by-step for cloning a repo and creating a new RStudio project.

Configure git

Before we can get started we need to do one more thing. Specifically, we need to configure your git so that RStudio can communicate with GitHub. This requires two pieces of information: your email address and your GitHub username.

To do so, run the following:

```
usethis::use_git_config(user.name = "github username", user.email = "your email")
```

Update YAML

Change the author to your name in YAML.

Packages

We'll use the **tidyverse** package for this analysis. Run the following code in the Console to load this package.

```
library(tidyverse)
```

The data

The dataset for this assignment can be found as a csv file in the `data` folder of your repository. You can read it in using the following.

```
nobel <- read_csv("data/nobel.csv")
```

The variable descriptions are as follows:

- `id`: ID number
- `firstname`: First name of laureate
- `surname`: Surname
- `year`: Year prize won
- `category`: Category of prize
- `affiliation`: Affiliation of laureate
- `city`: City of laureate in prize year
- `country`: Country of laureate in prize year
- `born_date`: Birth date of laureate
- `died_date`: Death date of laureate
- `gender`: Gender of laureate
- `born_city`: City where laureate was born
- `born_country`: Country where laureate was born
- `born_country_code`: Code of country where laureate was born
- `died_city`: City where laureate died
- `died_country`: Country where laureate died
- `died_country_code`: Code of country where laureate died
- `overall_motivation`: Overall motivation for recognition
- `share`: Number of other winners award is shared with
- `motivation`: Motivation for recognition

In a few cases the name of the city/country changed after prize was given (e.g. in 1975 Bosnia and Herzegovina was part of the Socialist Federal Republic of Yugoslavia). In these cases the variables below reflect a different name than their counterparts without the suffix `_original`.

- `born_country_original`: Original country where laureate was born
- `born_city_original`: Original city where laureate was born
- `died_country_original`: Original country where laureate died
- `died_city_original`: Original city where laureate died
- `city_original`: Original city where laureate lived at the time of winning the award
- `country_original`: Original country where laureate lived at the time of winning the award

Exercises

Note that in this lab, the R chunks are not provided for you. Therefore you must create your own code chunks and name them properly. A portion of the lab grade will be based on: - Naming code chunks - Reasonable number of commits to ensure you are tracking your progress - Good coding style

Get to know your data

1. How many observations and how many variables are in the dataset? Use inline code to answer this question.

There are some observations in this dataset that we will exclude from our analysis to match the Buzzfeed results.

```
**Hint:** The lecture about logical operators could be useful here!
```

2. Create a new data frame called `nobel_living` that filters for

- laureates for whom `country` is available
- laureates who are people as opposed to organizations (organizations are denoted with `"org"` as their `gender`)
- laureates who are still alive (their `died_date` is `NA`)

Confirm that once you have filtered for these characteristics you are left with a data frame with 228 observations.

Knit, commit and push your changes to GitHub with an appropriate commit message again. Make sure to commit and push all changed files so that your Git pane is cleared up afterwards.

"Most living Nobel laureates were based in the US when they won their prizes"

… says the Buzzfeed article. Let's see if that's true.

First, we'll create a new variable to identify whether the laureate was in the US when they won their prize. We'll use the `mutate()` function for this. The following pipeline mutates the `nobel_living` data frame by adding a new variable called `country_us`. We use an if/else statement to create this variable. The first argument in the `if_else()` function is the condition we're testing for. If `country` is equal to `"USA"`, we set `country_us` to `"USA"`. If not, we set the `country_us` to `"Other"`.

```
Note that we can achieve the same result using the `fct_other()` function (i.e. with `country_us = fct_
```

```
nobel_living <- nobel_living %>%
  mutate(
    country_us = if_else(country == "USA", "USA", "Other")
  )
```

Next, we will limit our analysis to only the following categories: Physics, Medicine, Chemistry, and Economics.

```
nobel_living_science <- nobel_living %>%
  filter(category %in% c("Physics", "Medicine", "Chemistry", "Economics"))
```

You will work with the `nobel_living_science` data frame you created above for the remainder of the lab. This means you'll need to define this data frame in your R Markdown document.

```
**Hint:** You can change the orientation of the bars using the `coord_flip()` function in ggplot2. Clic
```

3. Create a faceted bar plot visualizing the relationship between the category of prize and whether the laureate was in the US when they won the nobel prize. Note: Your visualization should be faceted by category. For each facet you should have two bars, one for winners in the US and one for Other. Flip the coordinates so the bars are horizontal, not vertical. Interpret your visualization, and say a few words about whether the Buzzfeed headline is supported by the data.

Knit, commit and push your changes to GitHub with an appropriate commit message again. Make sure to commit and push all changed files so that your Git pane is cleared up afterwards.*

"But of those US-based Nobel laureates, many were born in other countries…"

```
**Hint:** You should be able to borrow from code you used earlier to create the `country_us` variable.
```

4. Create a new variable called `born_country_us` that has the value `"USA"` if the laureate is born in the US, and `"Other"` otherwise. Be sure to save the variable to the `nobel_living_science` data frame.
5. Add a second variable to your visualization from Exercise 3 based on whether the laureate was born in the US or not. Your final visualization should contain a facet for each category, within each facet a bar for whether they won the award in the US or not, and within each bar whether they were born in the US or not. Based on your visualization, do the data appear to support Buzzfeed's claim? Explain your reasoning in 1-2 sentences.

Knit, commit and push your changes to GitHub with an appropriate commit message again. Make sure to commit and push all changed files so that your Git pane is cleared up afterwards.

Here's where those immigrant Nobelists were born

```
Note that your bar plot won't exactly match the one from the Buzzfeed article. This is likely because t
```

6. In a single pipeline, filter for laureates who were living in the US when they won their prize, but were born outside of the US, then create a frequency table (with the `count` function) for their birth country (`born_country`), and arrange the resulting data frame in descending order of number of observations for each country.

Knit, commit and push your changes to GitHub with an appropriate commit message again. Make sure to commit and push all changed files so that your Git pane is cleared up afterwards.

Wrapping up

Go back through your write up to make sure you followed the coding style guidelines we discussed in class (e.g. no long lines of code).

Also, make sure all of your R chunks are properly labeled, and your figures are reasonably sized.

Submission

Upload your PDF document to Canvas.

Interested in how Buzzfeed made their visualizations?

The plots in the Buzzfeed article are called waffle plots. You can find the code used for making these plots in Buzzfeed's GitHub repo (yes, they have one!) here. You're not expected to recreate them as part of your assignment, but you're welcomed to do so for fun! © 2020 GitHub, Inc.

This lab was adapted from Data Science in a Box.