Warmup 1: Basic Manipulation of Data Tables

Stat 133, Spring 2022

Introduction

The main goal of this warmup is to give you practice on working with data frames (and tibbles), by using a couple of *tidyverse* packages:

- manipulation with `"dplyr"`
- basic graphs with `"ggplot2"`

This assignment is heavily based on chapters

- https://www.gastonsanchez.com/intro2cwd/eda-dplyr.html
- https://www.gastonsanchez.com/intro2cwd/ggplot1.html

General Instructions

- Write your narrative and code in an `Rmd` (R markdown) file.
- Name this file as `hw1-first-last.Rmd`, where `first` and `last` are your first and last names (e.g. `hw1-gaston-sanchez.Rmd`).
- Please do not use code chunk options such as: `echo = FALSE`, `eval = FALSE`, `results = 'hide'`. All chunks must be visible and evaluated.

1) Data `"storms"`

You will be working with the data set `storms` from the R package `"dplyr"`. This data is a subset of the NOAA Atlantic hurricane database best track data, http://www.nhc.noaa.gov/data/#hurdat. The data includes the positions and attributes of 198 tropical storms, measured every six hours during the lifetime of a storm. For more information, read the documentation for `storms`.

Recall that you need to load the *tidyverse* packages (`"dplyr"` and `"ggplot2"` among them) in a code chunk; use the command:

```
library(tidyverse)
```

2) Storms in 2015

Use `"dplyr"` functions to answer the following parts.

a) Write a `"dplyr"` command to create a table (i.e. tibble) `storms2015` containing the storms that took place in the year 2015.

b) With `storms2015`, write a command that returns only the name of unique storms in 2015. In other words, the output should display only the unique names and nothing else.

c) With `storms2015`, write a command that returns a table with the name of each unique storm and the number of times it appears. In other words, this is a table with two columns: 1) name of storm, and 2) the number of counts of each storm.

d) With `storms2015`, write a command that gives you the name, month and day of the first storm recorded in 2015.

e) With `storms2015`, write a command that gives you the number of hurricanes that occurred in 2015. *Hint*: How does a storm acquire hurricane status?

f) With `storms2015`, write a command that gives you the names of the hurricanes that occurred during that year. *Hint*: How does a storm acquire hurricane status?

3) More manipulation

a) Use `"dplyr"` functions/commands to create a table (e.g. tibble) `storm_names_2010s` containing columns `name` and `year` of storms recorded from 2010 to 2015. To clarify, this table should contain **only one occurrence** of each storm. Use `head()` and `tail()` to display its first 5 rows, and also its last 5 rows.

b) With the entire `storms` data, use `"dplyr"` functions/commands to create a table (or tibble) `storm_counts_per_year` containing the number of unique storms in each year (i.e. counts of storms in each year). This table should contain two columns: year values in the first column, and the number of unique storms in the second column. Display its last 15 rows.

c) With the entire `storms` data, use `"dplyr"` functions/commands to create a table (e.g. tibble) `max_wind_per_storm` containing three columns: 1) `year` of storm, 2) `name` of storm, and 3) `max_wind` maximum wind speed record (for that storm). Display its first 10 rows, and also its last 10 rows.

4) Some basic plots with "ggplot2"

a) Make a barchart for the number of (unique) storms in each year during the period 2010 to 2015. Make sure that the axis-label of each bar indicates the associated year. Also, add a meaningful title to the plot.

b) Using the entire `storms` table, make a density graph for the variable `wind`, adding color to the border line, as well as the filling color of the density curve. Also, add a meaningful title to the plot, and choose the "Black-White" theme for the background of the graph.

c) Make boxplots for the variable `pressure` of storms in each year during the period 2000 to 2011. Use facets for year (i.e. one facet per year). Also, add a meaningful title to the plot, and choose the "Minimal" theme for the background of the graph.

d) Using the entire `storms` table, graph a timeline of the median wind speed by year. That is: years in the x-axis, median wind-speed in the y-axis, timeline connecting the dots for median wind speed in each year. Also, add a meaningful title to the plot.

5) Wind Speed and Pressure

a) With the entire `storms` table, use `"ggplot2"` functions to make a scatterplot of `wind` (x-axis) and `pressure` (y-axis). Because of the large number of dots, add an `alpha` value in order to make the dots somewhat transparent. Likewise, see how to add a "smoother" with the function `stat_smooth()`. Also, add a meaningful title to the plot.

b) With the entire `storms` table, use `"ggplot2"` functions to make the previous scatterplot of `wind` (x-axis) and `pressure` (y-axis). This time don't include a smoother; instead use the variable `category` to color-code the dots in the scatterplot. Also, add a meaningful title to the plot.

6) Storm Categories

a) Use `"dplyr"` functions/commands to display, in ascending order, the different (unique) types of storm categories (using the entire `storms` table).

b) With the entire `storms` tabl, use `"dplyr"` functions/commands to display a table showing the `category`, `avg_pressure` (average pressure), and `avg_wind` (average wind speed), for each type of storm `category`. This table should contain three columns: 1) `category`, 2) `avg_pressure`, and 3) `avg_wind`.

c) With the entire `storms` table, make a chart to visualize `pressure` in terms of the different category values. For example, you can use either histograms, or density curves, or boxplot, or violin plots. Add a meaningful title to the plot.