Flu Dashboard Workshop Instructions

Contents

Overview

In this workshop, you will create a dashboard summarizing the 2013 outbreak of influenza A H7N9 in China using data provided by the `outbreaks` package.

Your final product should be similar to the provided `solutions.html`. You can also refer to `sample_dashboard_gapminder.qmd` file for guidance as the dashboard created there is very similar to the one you need to create.

Files in Your Workshop Folder

Below are the key files in your workshop folder:

- **Index.qmd**: This is where you'll build your dashboard.
- **solutions.html**: An example dashboard that demonstrates what your final output should look like. Aim to get close to this example in function and style.
- **sample_dashboard_gapminder.qmd**: A sample dashboard that you can reference for style and code snippets.

Workflow Steps

Step 1: Review Example Outputs

Start by reviewing the `solutions.html` to understand the required outputs and functionalities of your dashboard.

Next, take a look at the `sample_dashboard_gampinder.qmd` file to see how that dashboard was created. Render the `sample_dashboard_gapminder.qmd` file to HTML to see the output.

Then, once you feel you have a good understanding of the required outputs and how to create them, you can start building your dashboard in the `index.qmd` file.

Step 2: Calculate Key Metrics

Start by calculating the main metrics to display on your dashboard:

- **Total Number of Cases**: Count all the rows in the dataset.
- **Total Deaths**: Count the number of deaths.
- **Total Hospitalizations**: Count the number of hospitalizations. (Tip: you can use the date of hospitalization variable to do this.)

Step 3: Map of Cases by Province

- Generate a summary of the number of cases per province using `count()` or `group_by()` and `summarize()`.
- Obtain shapefiles for China using `rgeoboundaries::gb_adm1("china")`.
- Ensure the province names in your dataset match those in the shapefiles. Here's some code to help with harmonizing the names:

```
china_map <- rgeoboundaries::gb_adm1("china")

# Simplify names for matching
china_map <- china_map %>%
  mutate(province = str_trim(str_remove(shapeName, "Province|Municipality|Autonomous Region")))

# Fix any specific mismatches (e.g., Guangdong vs. Guangzhou)
china_map <- china_map %>%
  mutate(province = if_else(province == "Guangzhou", "Guangdong", province))
```

- Join the harmonized data and create a choropleth map displaying the number of cases per province.

Step 4: Plot of Case Counts

- Create a bar chart using `ggplot2` to show the number of cases per province.

Step 5: Download Data Page

- Implement a data download section using `reactable` for an interactive table and a download button. Modify the example code from the `sample_dashboard_gapminder` for this section.

Step 6: About Page

- Include a description of the dataset and the purpose of the dashboard. Copy and adapt the content from the `solutions.html` dashboard.

Step 7: Render and Deploy

- Once your dashboard is complete, knit the `index.qmd` to HTML, review your output, and deploy it to GitHub Pages.