Noboru Hayashi

EN 605.662 Data Visualization

Project #4

# Interactive Visualization using JavaScript, R, or Python

## I. Introduction

From our previous exercise, we had an opportunity to get exposed and utilize the visualization tool: Tableau. But recently there are some libraries supported for many computing languages such as JavaScript, R, or Python, having more flexibility and capability than some of the off-the-shelf tools. In this paper, the ways to create interactive visualizations using Python will be demonstrated. The main library I used for this practice is plotly (Express & Graph Objects) [1], and codes were developed on Jupyter Notebook.

## II. Dataset & Approach

While developing visualizations with Plotly, I used three different datasets. The one for the first visualization is Gapminder[2], which is a built-in dataset of Plotly Express. This dataset contains countries' historical information such as continent, life expectancy, population, and etc. The code for the first visualization is using an existing sample from the official document, then some user interactive components are added to the figure.

For the second visualization, the dataset of COVID-19 World Vaccination Progress provided by Kaggle is used[3]. This dataset has over time records of vaccination counts and stats for each country. With this characteristic of the dataset, the visualization #2 is a time series visual which illustrates the total vaccination count for each country over time.

Finally, Stroke Prediction Dataset by Kaggle [4] is used in the third visualization. This dataset contains healthcare information of each patient such as gender, age, smoking_status, had stroke or not, and etc. In the process of developing the visualization, creating subplots and pie charts will be demonstrated.

## III. Visualization #1

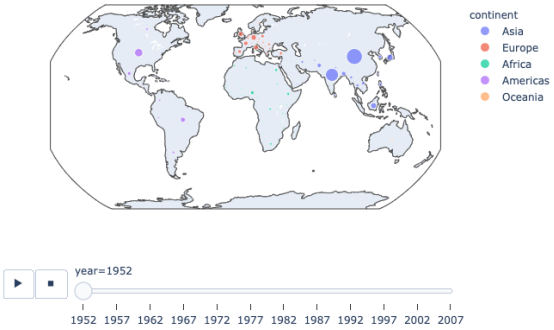



Figure 1

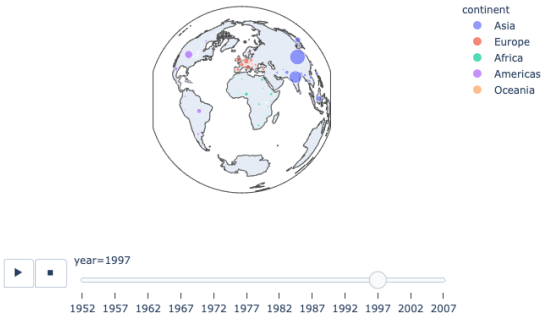



Figure 2

In the first visualization, the code snippet from the tutorial document below is used first[5], and figure 1 is shown:

```
import plotly.express as px
df = px.data.gapminder().query("year==2007")
fig = px.scatter_geo(df, locations="iso_alpha", color="continent",
                     hover_name="country", size="pop",
                     projection="natural earth")
fig.show()
```

This code snippet visualizes each country's population with a bubble map having animation, the actual number will be shown if the mouse hovers over the bubble. Then I changed the way of projection from "natural earth" to "azimuthal equal area", which is a type of mapping projection

that maintains the areas of the countries but distorts other properties. With the projection type, the visual figure 2 is shown.

## IV. Visualization #2

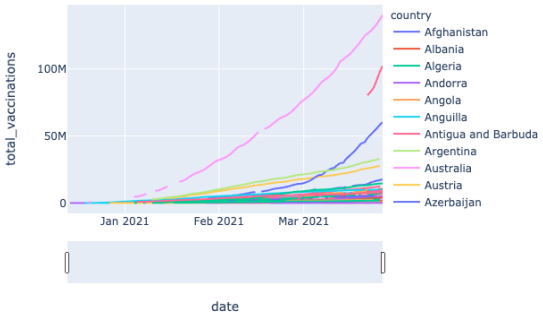



Figure 3

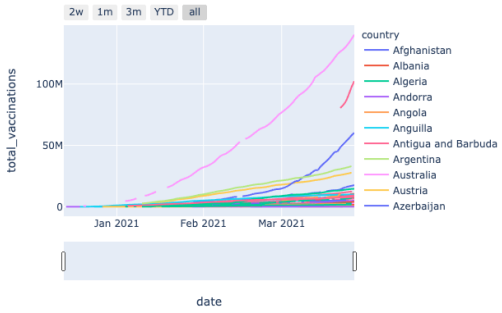
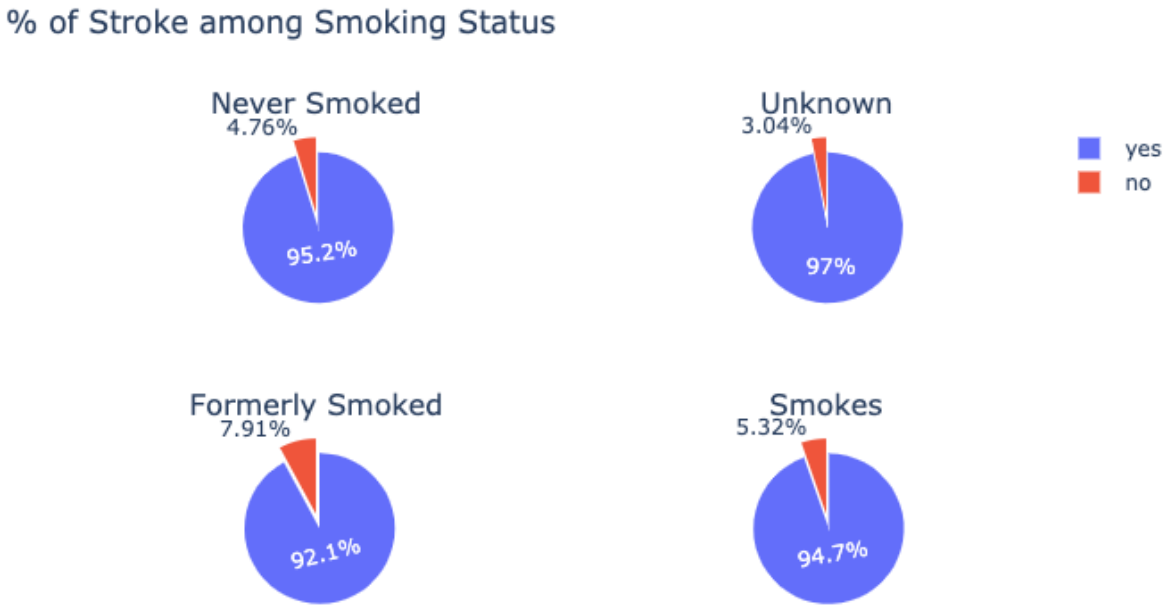



Figure 4

For the second visualization, I utilized the sample snippet for time series plots from the tutorial[6], and applied the code to Kaggle's COVID-19 Vaccination Progress dataset. Since Plotly also provides a feature of range slider and range selector, which can be added to the visual with a couple lines of code, I added the slider and selector for the date field of the dataset.

V. Visualization #3




In order to visualize the proportions of people who had a stroke before by splitting the data into groups of smoking status: Never smoked, Unknown, Formerly smoked and Smokes, a 2x2 subplot grid is created. Then for each slot in the grid, a pie chart is placed illustrating the corresponding stroke percentages.

VI. Conclusion

In this project, I developed three visualizations with the use of three different datasets and documentations for three types of visualizations (Map, Time Series, Pie) by Plotly. After some research on the documents and development, I got familiar with Plotly and I feel I now have a basic understanding of its capabilities.

## VII. References


1. Plotly Python Open Source Graphing Library: https://plotly.com/python/
2. Gapminder: https://www.gapminder.org/data/
3. COVID-19 World Vaccination Progress: https://www.kaggle.com/gpreda/covid-world-vaccination-progress
4. Stroke Prediction Dataset: https://www.kaggle.com/fedesoriano/stroke-prediction-dataset
5. Bubble Maps in Python: https://plotly.com/python/bubble-maps/
6. Time Series and Date Axes in Python: https://plotly.com/python/time-series/