

HOMEWORK 3

1. D3

1.1 Global Terrorism Heatmap

Design Choices

- The JavaScript has been imbedded directly in the html file for quick running without setting up a project/app.
- Certain country mappings have to be updated to make them consistent with the geojson data.
- Used d3.rollup to create a nested Map, grouping incidents by country and year, with v.length counting incidents per group. The +d.iyear ensures years are numeric for proper keying.
- Calculate the maximum incident count across all countries and years to set the scale's domain. d3.interpolateBlues provides a gradient from light to dark blue, intuitively representing intensity.
- Sequential scaling is apt for count data, making it easy to spot high-incident areas.
- Display country name and incident count on hover, with smooth transitions (duration(200) for fade-in, 500 for fade-out). Positioning uses mouse coordinates (event.pageX, event.pageY) for responsiveness.
- The slider updates the displayed year and map colors in real-time. The 200ms transition smooths color changes, enhancing the user experience. The range (1970–2020) is hardcoded, which works if it matches the data's scope but could be dynamic for broader applicability.

Insights

- Most attacks seem to have been concentrated in UK, Spain, France, Italy and Germany
- UK has had 200+ attacks 4 times.
- There were a lot of attacks across Europe during 1979.
- Recently the attacks seem to be have been reduced.

How to run?

1. Make sure to have "data/region_08.csv" file. Or change the location accordingly.
2. Launch a https server in the directory with the file. (ex.: python -m http.server 8000)
3. Visit: <http://localhost:8000/>. Open the html file in the browser.

1.2 Top 10 Terrorist Groups Bar Chart

Design Choices

- A dropdown that allows users to filter data by year, making the visualization interactive and explorable.
- Provides context by showing the total number of incidents in the selected year.
- Displays appropriate feedback when there's insufficient data (fewer than 10 groups) or errors.
- **Sequential Color Scale:** The visualization uses `d3.interpolateBlues` to create a gradient from light to dark blue based on rank.
- **Transitions:** The code implements smooth animations when updating the chart.
- **Hover Effects:** The CSS includes hover states for bars.
- The code uses D3's `rollup` function to count incidents by group.
- **Sorting and Limiting:** The data is sorted by incident count and limited to the top 10.
- **Dynamic Scale Adjustment:** The scales adapt to the data range for each year:

```
x.domain([0, d3.max(sortedData, d => d.incidents)]);  
y.domain(sortedData.map(d => d.gname)).padding(0.1);
```

- The code implements error handling:
 - Data Validation: Checks if data exists for the selected year
 - Group Count Checking: Displays a warning if fewer than 10 groups are found
 - Try-Catch Blocks: Catches and displays errors that occur during chart updates
 - CSV Loading Error: Handles cases where the data file can't be loaded

Insights

- Most Terrorist attacks have been from Unknown groups.
- IRA has been responsible for a lot of attacks.
- ETA has also been responsible for significant attacks.
- Neo-nazi extremist had a lot of attacks in 1992.
- PKK conducted lot of attacks in 1995
- Most attack groups have been unknown since 1996.

How to run?

4. Make sure to have "data/region_08.csv" file. Or change the location accordingly.
5. Launch a https server in the directory with the file. (ex.: `python -m http.server 8000`)
6. Visit: <http://localhost:8000/>. Open the html file in the browser.

2. Plotly Visualization: Attack Types Over Time

Design Choices

- The data is made continuous for all years. For attack types that didn't occur in specific years, have been added with count=0.
- Yearly distribution is made by grouping the data by attack type and year.
- For each attack type 3 traces are added:
 - Stacked Graph
 - Grouped Graph
 - 100% Stacked
- A dropdown is added to select the trace type.

Insights

- Most area seems to be covered by Bombing and Explosions, indicating the most attacks.
- Assassinations, Armed Assaults & Facility/Infra Attack seem to be prominent as well.

How to run?

1. Install plotly, pandas and numpy.
2. Make sure to have "data/region_08.csv" file. Or change the location accordingly.
3. Simply run the jupyter notebook.

3. Bokeh Visualization: Target Types and Casualties

Design Choices

- The data is grouped by Target Types for every range of years. The graph will always display aggregated results for the entire range.
- The size of the scatter points is dynamic (Min: 10, Max: 30). And is scaled based on the total consequences (wounded + killed)
- An hover tool is added to display the stats.
- Range slider is added to select a range of years.
- Bokeh does not support python callbacks for standalone runs, so custom JS callback has been used.
- The custom callback filters the data based on the year range and aggregates the statistics grouped by target type.

Insights

- Most fatalities have been when the target type Private Citizens & Property.
- Most wounds have been occurred when the target were Businesses.

- Transportation & Business targets have consistently wounded across the years.
- For the early years Business targets produced the most wounded, while Transportation has produced the most wounded in recent years.

How to run?

1. Install plotly, pandas and numpy.
2. Make sure to have "data/region_08.csv" file. Or change the location accordingly.
3. Simply run the jupyter notebook.