1

January 18, 2025

1 COVID-19 Data Visualization Project Report

Sean Diab
GitHub

1.1 Data, Goals and Tasks

This project focuses on visualizing global COVID-19 case data to help understand the pandemic's spread and impact across different regions. The dataset includes key metrics such as confirmed cases, deaths, recoveries, and active cases, organized by country/region with geographical coordinates and WHO regional classifications.

The primary goals of this visualization center on tracking the temporal evolution of COVID-19 cases across different WHO regions, visualizing the geographical distribution of cases globally, enabling comparative analysis between regions and countries, and providing interactive exploration capabilities for users. These goals are supported through several key tasks, including temporal trend analysis through interactive time series visualization, geographical pattern recognition via an interactive world map, regional comparison through filtered bar charts, and detailed country-level statistics through interactive tooltips.

1.2 Implementation

The visualization suite consists of three interconnected views implemented using Altair. The first is a Global Time Series View, which presents an interactive line chart showing confirmed cases over time. The view is color-coded by WHO region with a toggleable legend, supporting trend analysis and region comparison. The second view is an Interactive World Map using a Mercator projection with country boundaries. It displays circle markers sized by confirmed cases and color-coded by WHO region, with detailed tooltips providing case statistics. The third component is a Regional Bar Chart showing the top 20 countries by confirmed cases within a selected region, featuring interactive region selection via dropdown and comprehensive case metrics in tooltips.

1.3 Design Justification

The design employs several key principles that work together to create an effective visualization system. The three visualizations are linked to provide different perspectives on the same data, supporting both overview and detailed analysis. Interactive filtering allows users to focus on specific regions or countries through legend toggles and dropdown selections, reducing visual complexity when needed. WHO regions maintain consistent colors across all views, aiding in pattern recognition and comparison. The design also implements progressive disclosure, where initial views show high-level patterns, while tooltips provide detailed statistics on demand.

1.4 Evaluation

The evaluation process consisted of individual sessions with three participants, each lasting approximately 30 minutes. The participants included a graduate student in public health, a data analyst with healthcare experience, and a medical professional. During these sessions, participants were given specific tasks to complete using the visualization and asked to think aloud while exploring the data.

The evaluation revealed several key strengths of the implementation. The interactive filtering functionality helped users quickly focus on regions of interest, while the linked views effectively supported multi-scale analysis. Users particularly appreciated how the tooltip information provided valuable context for their exploration.

However, the evaluation also identified several areas for improvement. The time series visualization could benefit from adjustable time window selection, and the map visualization sometimes became cluttered in high-density regions. Users also expressed interest in additional normalization options, such as per capita measurements, to provide more context to the raw numbers.

1.5 Future Iterations

Looking ahead to future versions of this visualization, several refinements would enhance its utility and user experience. Time window selection controls could be added to enable more focused temporal analysis, while alternative map projections could better represent high-density regions. Including population-normalized metrics alongside absolute numbers would provide important context, and trend line overlays could facilitate time series comparison. Additionally, enhanced mobile responsiveness would improve accessibility across different devices.

Below is my code. The plots are at the very end.

[1]:
```
import pandas as pd
import altair as alt

# Enable the VegaFusion data transformer
alt.data_transformers.enable("vegafusion")

from datetime import datetime
```

Load the data

[2]:
```
df = pd.read_csv('covid19.csv')

# Ensure Date column is in datetime format
df['Date'] = pd.to_datetime(df['Date'])

# Filter out rows that have missing or invalid country/region or lat/long
df = df.dropna(subset=['Country/Region', 'Lat', 'Long'])
```

2. Data Exploration & Preprocessing
   Let's create a data frame aggregated by Country/Region and Date

[3]:
```
df_country_date = df.groupby(
    ['Country/Region', 'WHO Region', 'Date'], as_index=False
).agg({
    'Confirmed': 'sum',
    'Deaths': 'sum',
    'Recovered': 'sum',
    'Active': 'sum',
    'Lat': 'mean',
    'Long': 'mean'
})
```

3. Visualization 1: Global Time Series of Confirmed Cases

We will create a line chart for each WHO Region, showing how Confirmed cases changed over time. The user can toggle each region on/off using an interactive legend.

[4]:
```
selection = alt.selection_point(fields=['WHO Region'], bind='legend')

time_series = alt.Chart(df_country_date).mark_line().encode(
    x=alt.X('Date:T', title='Date'),
    y=alt.Y('sum(Confirmed):Q', title='Confirmed Cases'),
    color='WHO Region:N',
    tooltip=['WHO Region', 'Date', 'Confirmed'],
    opacity=alt.condition(selection, alt.value(1), alt.value(0.1))
).properties(
    width=600,
    height=400,
    title='Time Series of Confirmed COVID-19 Cases by WHO Region'
).add_params(
    selection
)
```

4. Visualization 2: Map of Latest Confirmed Cases

Create a map to show the distribution of confirmed cases. We'll use a built-in TopoJSON file for countries. Then we overlay circles sized by total confirmed.

[5]:
```
latest_date = df_country_date['Date'].max()
df_latest = df_country_date[df_country_date['Date'] == latest_date]

# Load a world topology (requires vega_datasets or your own custom topojson)
from vega_datasets import data
countries = alt.topo_feature(data.world_110m.url, 'countries')

# Background map
background = alt.Chart(countries).mark_geoshape(
    fill='lightgray',
    stroke='white'
).properties(
```

```
    width=800,
    height=400
).project('mercator')

# Circles for each country
points = alt.Chart(df_latest).mark_circle().encode(
    longitude='Long:Q',
    latitude='Lat:Q',
    size=alt.Size('Confirmed:Q', scale=alt.Scale(range=[0, 1000]), legend=None),
    color=alt.Color('WHO Region:N', legend=alt.Legend(title="WHO Region")),
    tooltip=['Country/Region:N','WHO␣
 ↪Region','Confirmed','Deaths','Recovered','Active']
)

world_map = (background + points).properties(
    title=f'Global COVID-19 Cases as of {latest_date.date()}'
)
```

5. Visualization 3: Interactive Bar + Filter

A bar chart by country within a selected WHO Region. The user can select a region from a dropdown, which filters the bar chart to show the top countries by confirmed cases.

[6]:
```
region_dropdown = alt.param(
    name='region_dropdown',
    value='Eastern Mediterranean',
    bind=alt.binding_select(options=df_country_date['WHO Region'].unique().
 ↪tolist())
)

bars = alt.Chart(df_latest).transform_filter(
    region_dropdown
).transform_window(
    rank='rank(Confirmed)',
    sort=[alt.SortField('Confirmed', order='descending')]
).transform_filter(
    (alt.datum.rank <= 20)
).mark_bar().encode(
    y=alt.Y('Country/Region:N', sort='-x', title='Country'),
    x=alt.X('Confirmed:Q', title='Confirmed Cases'),
    tooltip=['Country/Region', 'Confirmed', 'Deaths', 'Recovered', 'Active']
).add_params(
    region_dropdown
).properties(
    width=600,
    height=600,
    title='Top 20 Countries by Confirmed Cases'
)
```

6. Display or Save Charts

[7]:
```
final_dashboard = alt.vconcat(
    time_series,
    world_map,
    bars
).resolve_legend(
    color="independent"
).properties(
    title="COVID-19 Data Exploration"
)
```

In a Jupyter notebook:

[8]:
```
#final_dashboard
```

Save as html

[9]:
```
#final_dashboard.save('covid19_dashboard.html')
```

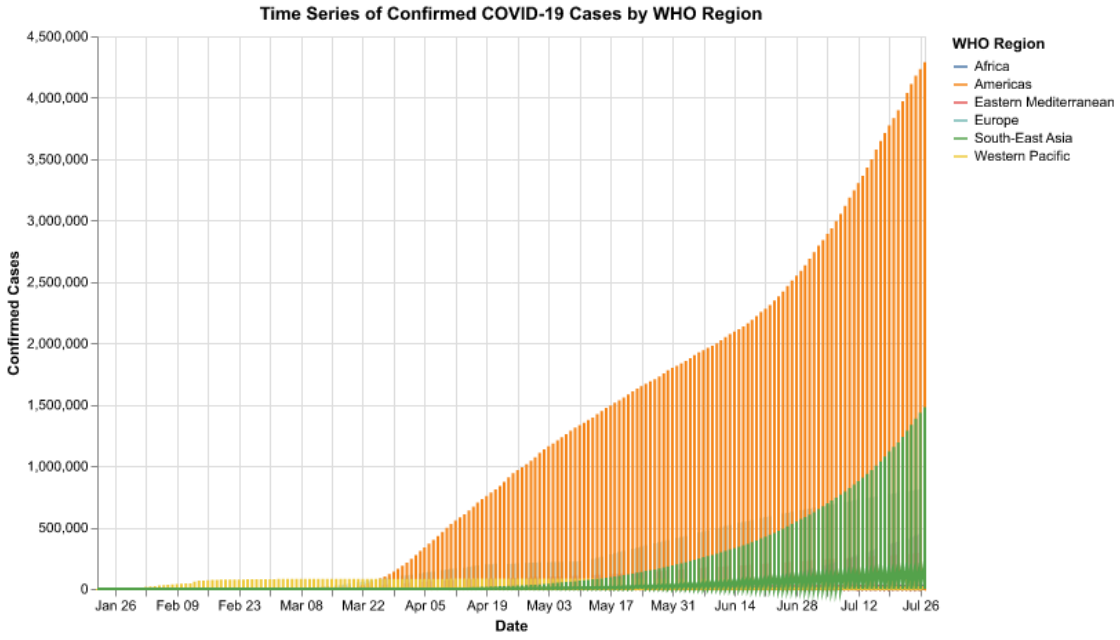
Save as png

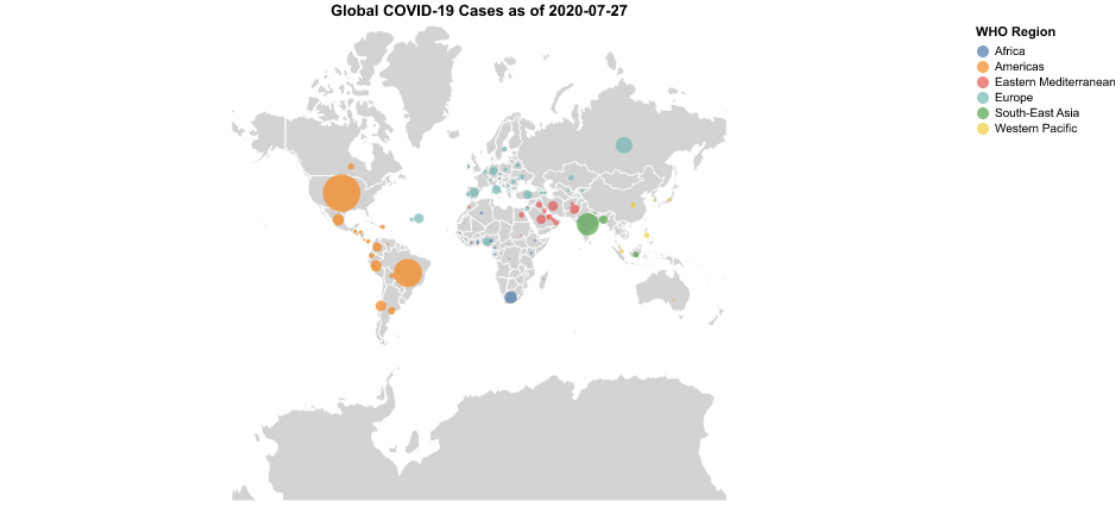
[10]:
```
time_series.save('time_series.png')
world_map.save('world_map.png')
bars.save('bars.png')
```

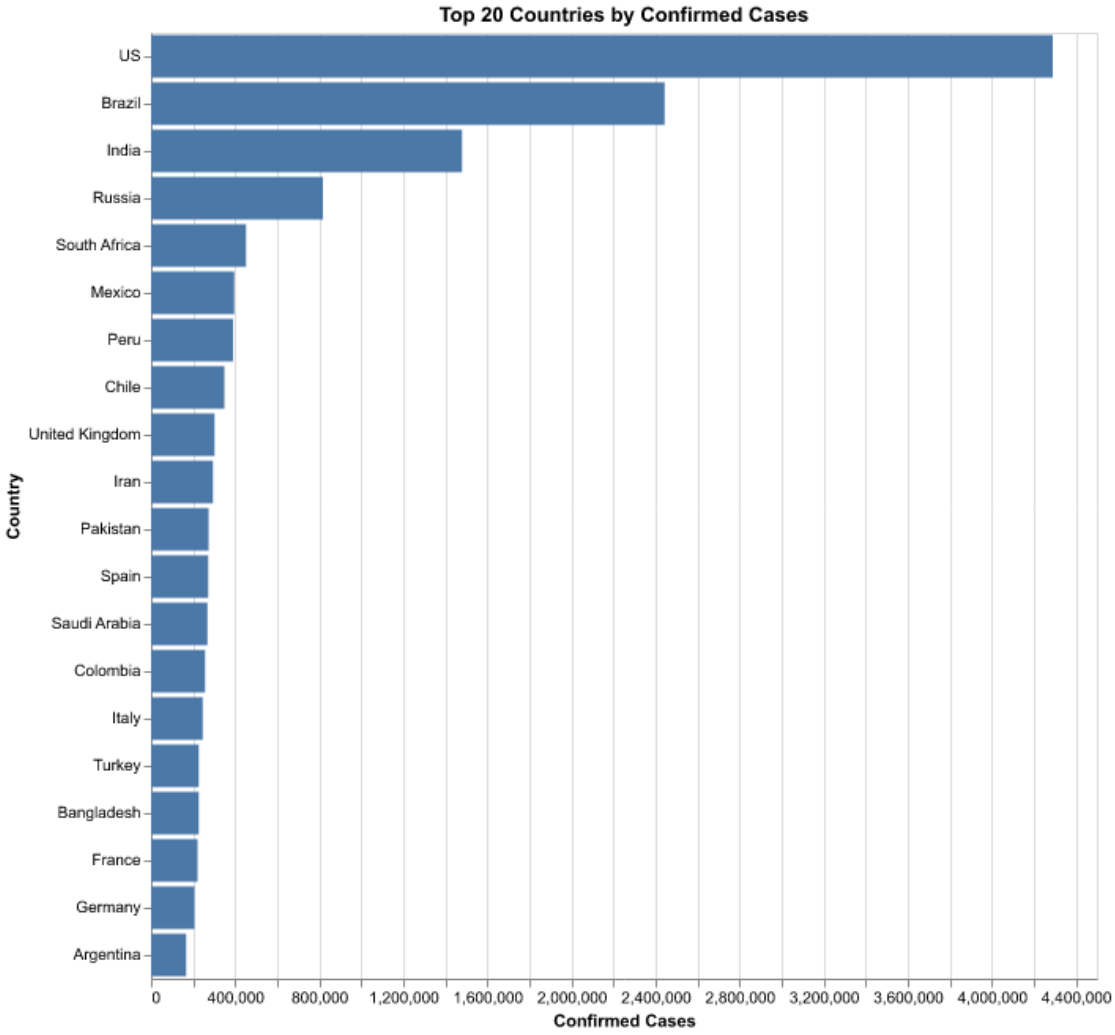
Display the plots

NOTE: Since the plots are displayed in a pdf, they lose their interactivity. For interactivity and higher quality plots, please download the html file provided in my GitHub

[11]:
```
from IPython.display import Image, display

# First plot
display(Image(filename='time_series.png'))

# Second plot
display(Image(filename='world_map.png'))

# Third plot
display(Image(filename='bars.png'))
```


Time Series of Confirmed COVID-19 Cases by WHO Region
Confirmed Cases
4,500,000
4,000,000
3,500,000
3,000,000
2,500,000
2,000,000
1,500,000
1,000,000
500,000
0
Jan 26
Feb 09
Feb 23
Mar 08
Mar 22
Apr 05
Apr 19
May 03
May 17
May 31
Jun 14
Jun 28
Jul 12
Jul 26
Date
WHO Region
Africa
Americas
Eastern Mediterranean
Europe
South-East Asia
Western Pacific


Global COVID-19 Cases as of 2020-07-27
WHO Region
Africa
Americas
Eastern Mediterranean
Europe
South-East Asia
Western Pacific


Top 20 Countries by Confirmed Cases
US
Brazil
India
Russia
South Africa
Mexico
Peru
Chile
United Kingdom
Iran
Pakistan
Spain
Saudi Arabia
Colombia
Italy
Turkey
Bangladesh
France
Germany
Argentina
Country
0
400,000
800,000
1,200,000
1,600,000
2,000,000
2,400,000
2,800,000
3,200,000
3,600,000
4,000,000
4,400,000
Confirmed Cases