Chapter 11
Perspectives on the Definition of Data Visualization: A Mapping Study and Discussion on Coronavirus (COVID-19) Dataset

Ajantha Devi and Anand Nayyar

Abstract  Data visualization is an arrangement that presents data in manners that enable the utilization of human subjective and visual capacities. It is the method used to convert crude information into some visual structure. It uses designs and visuals to help with the psychological burden of comprehending big data. The burden put on general well-being because of the coronavirus is exacerbated by the ceaseless rise of new strains and unanswered inquiries concerning viral spread inside the host. Observations regarding the course of coronaviruses in people have been step by step expanded and extended to numerous regions worldwide. These observation programs have created an enormous amount of genomic information regarding coronaviruses, which encourages the investigation of the infection by computational strategies that are proficient and cost effective.

The main focus of this chapter is the development of visualization techniques to comprehend the advancement of coronaviruses. The strategies depend on unaided dimensional decrease methods, which can be applied to every individual genome fragment or to the total genome succession of the infection. These strategies are a takeoff from the customary phylogenetic tree development worldview in light of the fact that an exceptionally enormous number of high-dimensional info arrangements can be prepared and results are seen legitimately in an a few-dimensional Euclidean space.

Keywords  Data visualization · Coronavirus · Public health · Disease surveillance · Social network analysis

A. Devi (*)
Research Head, AP3 Solutions, Chennai, Tamil Nadu, India
e-mail: ap3solutionsresearch@gmail.com

A. Nayyar (*)
Graduate School, Faculty of Information Technology, Duy Tan University, Da Nang, Viet Nam
e-mail: anandnayyar@duytan.edu.vn

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021
F. Al-Turjman et al. (eds.), Emerging Technologies for Battling Covid-19, Studies in Systems, Decision and Control 324, https://doi.org/10.1007/978-3-030-60039-6_11
11.1 Introduction

With the broad utilization of computers, huge amounts of data are generated every day in many different disciplines, including scientific and engineering fields, economics, and social sciences. The need to understand and develop insights from big data is urgent. However, the data are large, complex, and often unstructured, which makes it difficult for us to discover the underlying patterns and reveal the hidden relationships.

Dissimilar to the conventional investigation method, which combines the information into a few numbers, visualization approaches [1] couple human and machine analysis by utilizing the human visual framework to aid information revelation. Visualization permits more parts of the information to be watched and enables collaborations for increasingly deliberate investigation. The utilization of graphical diagrams gives a general way to change the information and its connections into a theoretical view for investigating complex connections and improving information understanding.

Producing graph-based portrayals for various types of data is challenging due to the following reasons. As a matter of first importance, users are keen on particular connections derived from different types of data. For multivariate data collections, they might be keen on considering the connections among factors. For instance, when the auto showcase is considered, they might need to know the correlation between price and brand.

11.2 Visualization of Scientific Data

Wang and Tao classified the methods utilizing graph-based portrayals in logical perception depending on their applications into four categories: partition-wise, relationship-wise, structure-wise, and provenance-wise. Works identified within this chapter fall into the categories of partition-wise and relationship-wise. In particular, the four connections reviewed here are as follows: hierarchical relationships, data evolution, variable relationships, and relationships among field lines.

11.2.1 Exploring Hierarchical Relationships

A tree is a unique sort of graph that does not contain cycles. It is often used to represent hierarchical relationships. In this subsection, we focus on two sorts of chains of important developments: data partitioning and data clustering.

Data Partitioning Data partitioning progressively parcels the spatiotemporal data into smaller pieces, which enables clients to analyze the data in a versatile way. Octree is a broadly utilized structure to segment a solitary volume. During prepro-
cessing, they originally divided the data set into space-time squares utilizing an octree for singular compression. At that point, they pressed information checks into the illustration’s memory, then remade and rendered the data in the GPU. In addition to being the inward structures for parcels, trees act as interfaces that guide clients in their investigation. Wang and Shen presented the hierarchical navigation interface, an interface with numerous planned perspectives for level-of-detail (LOD) choice and rendering. There are three perspectives: the volume rendering view, the tree view that displays the chain of command cut, and a tree map that assists clients with pinpointing the objective districts for exploration.

Data Clustering  Other than dividing the area, different methodologies apply clustering techniques to group data pieces and structure chains of command that could be normally spoken to by trees. Gu and Wang applied three hierarchal clustering algorithms to gather the example focuses dependent on their connections and eliminate the pecking order when utilizing trees. Equal directions are used to visualize the connections between example focuses for each degree of the tree.

11.2.2  Understanding Data Evolution

Contemplating the advancement in a time-varying volumetric data is significant in numerous scientific and engineering fields. It enables researchers to watch the progressions and check their hypotheses. Graph-based perception aims at helping researchers to realize the development in the information through disentangling the fundamental advancement connections and picturing them as charts. A significant test is to distinguish highlights and concentrate the connections. One commonly utilized arrangement is distinguishing client-characterized highlights and matching them in neighboring time steps. Another arrangement is to examine the likelihood of changes starting with one component then onto the next, which permits emotional element changes.

Feature Matching  For the component coordinating methodologies in 3D volume tracking, features are coordinated through their qualities or areas, under the assumption that the progressions in neighboring time steps are small. A commonly embraced technique extricates old limited features from each time step and afterward relates them over time.

11.3  Data Visualization Regarding the COVID-19 Pandemic

Data visualization [1–3] in its most fundamental structure is essentially mapping data to geometry and shading. In addition, a significant part of visuals is to have the option to guide to the information, ensuring that its substance is unblemished, and
in any case, the visualizations will show up as only shapes. Having the option to pick a specific viewable signal is essential. These signs by and large change dependent on the job needing to be done, which relies upon how different shapes, sizes, and shades are seen. Figure 11.1 presents the ten most common prompts.

The main obvious sign is position, which helps in spotting clusters, patterns, and outliers by plotting all the information immediately. One model type is a scatterplot, where data centers, addressed as spots, are settled on a choice reliant on their X and Y arrangements and where they are comparative with others. Scatterplots can be valuable when the information is huge in size, because it draws all the information inside the X- and Y-plane, therefore requiring less space.

COVID-19 [4–6] is a fatal pandemic that the world is currently facing. The principle reason for coronavirus infection data-visualization [1–3] is to convey data unmistakably and viably utilize distinctive graphical presentations. Visualization is a helpful medium for examining, comprehending, and transmitting data because it has a few potential uses in the domain.

Python is viewed as one of the top programming languages for handling data visualization because it is recognized for its huge and dynamic logical figuring network and has numerous libraries that take into consideration more prominent adaptability. It can likewise control the particular components of the charts that are made and make those particulars repeatable through code. In addition, python is extraordinary at taking care of information and can handle a lot of information without crashing; it is particularly valuable for analyses and heavy computation. Finally, Python has a spotless and simple to-peruse sentence structure that software programmers like, and it can work off of a lot of modules to make information illustrations (Fig. 11.2).

1. Data acquisition: is the place where you read information from different well-springs of unstructured information, semi structured information, or fully organized information that may be put in a spreadsheet, comma-isolated document, website page, database, and so forth.
2. Data cleaning: is the place where you expel uproarious information and set tasks expected to keep just the significant information.
3. Exploratory analysis: is the place where you inspect your cleaned information and make factual preparing fits for explicit examination purposes.
4. An analysis model should be made. Progressed apparatuses, for example, AI calculations, can be utilized in this progression.
5. Data visualization is the place the outcomes are plotted utilizing different frameworks provided by Python to help in the dynamic procedure.
### 11.3.1 Data Visualization Libraries

**Matplotlib** Matplotlib is one of the most famous Python bundles utilized for data visualization. It is a cross-platform library for making 2D plots from data in arrays. It gives an object-oriented API that helps in installing plots in applications utilizing Python GUI.

![Type of plots on visualization](image-url)
Tool boxes, for example, PyQt and WxPythonotTkinter. It tends to be utilized in Python and IPython shells, Jupyter note pad and web application servers as well.

**Seaborn** It is a library based on Matplotlib. It enables one to make their representations prettier, and furnishes us with a portion of the basic data visualization needs (such as mapping a shading to a variable or utilizing faceting). Seaborn is increasingly incorporated for working with Pandas Data Frames.

**Plotly** Plotly.js is an explanatory JavaScript data visualization library based on D3 and WebGL that bolsters a wide scope of statistical, scientific, financial, geographic, and three-dimensional perceptions. Backing for making Plotly.js perceptions from Python is given by the plotly.py library.

**Folium** It tells the best way to make a Leaflet web map from scratch with Python and the Folium library. The process should create a map.html document. Afterward, you can essentially put that HTML document on a live server and have the guide on the web.

### 11.4 Data Visualization: Case Study on COVID-19

COVID-19 has affected more than 11 million individuals worldwide and in excess of 500,000 lives in Europe and the United States, overwhelming the numbers in China [7–10] where the pandemic began last December. The World Health Organization (WHO) announced the coronavirus a pandemic because of the across the board size of the episode and has cautioned that the most extreme period of COVID-19 [4–6] is yet to come. The United Nations said the coronavirus pandemic is the most noticeably awful worldwide emergency since World War II.

In this examination, we perform a time-intermittent observation of COVID-19 in the adjoining United States [11] utilizing an imminent space-time check...
measurement. We recognize developing groups of COVID-19 [4, 5] in the United States at the province level, and give day-by-day results [12] during the investigation time of January 22, 2020–June 25, 2020.

11.4.1 COVID-19 Confirmed Cases Using Plotly with Scattergeo

See Figs. 11.3 and 11.4.

Fig. 11.3 COVID-19 confirmed cases: code using Plotly

![COVID-19 confirmed cases: code using Plotly](image)

Fig. 11.4 COVID-19 confirmed cases worldwide as of April 13, 2020

![COVID-19 confirmed cases worldwide as of April 13, 2020](image)
11.4.2 COVID-19 Confirmed Cases Using Plotly with Express

Plotly with Express is a high-level API that wraps a graph object, where a multi-bar plot can be implemented in a single line (Figs. 11.5 and 11.6).

- All the nations began to recognize COVID-19 cases from the end of January with the exception of China [7–10], which revealed to the world that the focal point of the infection was in China.
- China showed a development in number of cases from January till mid-February, and after that, the quantity of cases has been consistent in China.
- The United States [11] has been the most noticeably affected nation in the world, and as of April 19 the quantity of cases that had been contracted was 700,000 and it had expanded to in excess of 800,000 on April 22, 2020.
- Italy, Spain, and France [11, 13–17] nearly follow a similar example and the quantity of cases contracted there was approximately 200,000 as of April 22, 2020.
- In India, the condition was better contrasted with different nations [18, 19] as the quantity of cases contracted was just 22,000, which is extremely few when contrasted with different nations as of April 22, 2020.

![Code using Plotly with Express](image1)

**Fig. 11.5** COVID-19 confirmed cases: code using Plotly with Express

![Graph of COVID-19 confirmed cases](image2)

**Fig. 11.6** COVID-19 confirmed cases worldwide as of June 29, 2020
11.4.3 Reported COVID-19 Cases Using Plotly with Choropleth

- A geometry as a geojson object. This is the place where things are somewhat befuddling and not obviously referenced in its documentation. You might give a geojson object. On the off chance that you give a geojson object, at that point, that item will be utilized to plot the earth highlights, yet in the event that you do not give a geojson object, at that point, the capacity will, as a matter of course, utilize one of the inherent geometries.
- A pandas DataFrame object for the characteristic data_frame. Here we label our DataFrame, new_df, as we did before.
- We will utilize the information of the confirmed section to choose the shade of every nation polygon.
- Further, we will utilize the date segment to make the animation_frame. In this way, as we slide over the dates, the shades of the nations will change according to the qualities in the confirmed segment (Figs. 11.7 and 11.8).

Instances of coronavirus disease 2019 (COVID-19) are expanding exponentially in Europe and North America, as shown in Fig. 11.9. These settings have among the most vigorous readiness plans; in any case, most have been not able to satisfy the requirements placed on their well-being frameworks by this pandemic. In total, 33% of nations have abilities to react to a well-being crisis in accordance with the International Health Guidelines. Struggling nations are lopsidedly influenced

![Fig. 11.7 Reported COVID-19 cases: code using Plotly](image-url)
because of the results of contention on their well-being frameworks, foundations, establishments, economies, and general public well-being, leaving them badly arranged to oversee pandemics, for example, COVID-19 (Fig. 11.10).
11.4.4 COVID-19 Daily Confirmed and Recovery Cases in India: Bar Chart

See Figs. 11.11, 11.12, 11.13, and 11.14.

11.4.5 Recovery Rate in Italy: Using Plotly with Indicator and Gauge Mode

The reason for “indicator” is to picture a solitary worth determined by the “value” property. Three unmistakable visual components are accessible to speak to that value: number, delta, and check. Any mix of them can be determined through the “mode” attribute. High level attributes are:

```python
In [7]: filterIn = df_filtered["Country/Region"] == "India"
filterIn = df_filtered[filterIn]
fig = px.bar(df_In, title="COVID-19 Daily Confirmed and Recovery Cases in India", x="Date", y="Confirmed_Plus",
               color="Recovered_Plus",
               labels={"Confirmed_Plus": "New Confirmed Cases"},
               hover_data={"Confirmed_Plus", "Recovered_Plus", "Deaths_Plus"})
fig.write_html('first_figure.html', auto_open=True)
```

![Fig. 11.11](image-url) COVID-19 daily confirmed and recovery: code for Bar chart—India

```python
In [11]: filterIt = df_filtered["Country/Region"] == "Italy"
filterIt = df_filtered[filterIt]
fig = px.bar(df_It, title="COVID-19 Daily Confirmed and Recovery Cases in Italy", x="Date", y="Confirmed_Plus",
               color="Recovered_Plus",
               labels={"Confirmed_Plus": "New Confirmed Cases"},
               hover_data={"Confirmed_Plus", "Recovered_Plus", "Deaths_Plus"})
fig.write_html('first_figure.html', auto_open=True)
```

![Fig. 11.13](image-url) COVID-19 daily confirmed and recovery: code for Bar chart—Italy
11.4.6 COVID-19 Confirmed Cases Using Seaborn

Seaborn gives profoundly appealing and useful charts/plots. It is anything but difficult to utilize and is blazingly quick. It tends to be utilized to manufacture practically every statistical chart. It is based on matplotlib, which is likewise a visualization library. Seaborn is a dataset [12] with arranged plotting capacity that can be utilized regarding the two information types: casings and exhibits. It improves the representation intensity of matplotlib, which is only utilized for fundamental plotting such as visual diagrams, line graphs, and pie charts (Figs. 11.19 and 11.20).

Fig. 11.14 COVID-19 daily confirmed and recovery: cases in India—Bar chart—Italy

In [26]:

```python
import plotly.graph_objects as go
fig = go.Figure(go.Indicator(
    mode = "gauge+number",
    value = 48877,
    domain = {'x': [0, 1], 'y': [0, 1]},
    title = {'text': "Recovery in Italy"},
    gauge = {
        'axis': {'range': [None, 160000], 'tickwidth': 1, 'tickcolor': "darkblue"},
        'bar': {'color': "darkblue"},
        'steps': [
            {'range': [0, 40000], 'color': "lightcoral"},
            {'range': [40000, 80000], 'color': "orange"},
            {'range': [80000, 120000], 'color': "yellow"},
            {'range': [120000, 160000], 'color': "green"}
        ],
        'threshold': {'line': {'color': "red", 'width': 4}, 'thickness': 0.75, 'value': 157114}
    })
))
fig.update_layout(paper_bgcolor = "lavender", font = {'color': "darkblue", 'family': 'Arial'})
fig.write_html('first_figure.html', auto_open=True)
```

Fig. 11.15 Recovery rate in Italy: code using Plotly with Indicator and Gauge mode

• value: the incentive to picture
• mode: which visual components to draw
• align: how to align number and delta (left, focus, right)
• domain: the degree of the figure (Figs. 11.15, 11.16, 11.17 and 11.18)
Fig. 11.16  Recovery rate in Italy: using Plotly with Indicator and Gauge mode

Fig. 11.17  Recovery rate in Spain: code using Plotly with Indicator and Gauge mode

Fig. 11.18  Recovery rate in Spain: using Plotly with Indicator and Gauge mode
Fig. 11.19  Top 20 countries COVID-19 confirmed cases: code

![Top 20 countries COVID-19 confirmed cases: code](image)

**Fig. 11.20**  Top 20 countries COVID-19 confirmed cases worldwide as of June 24, 2020

The nation names on the x pivot all overlap each other. Furthermore, the bars are for the most part various colors. More often than not, the axis bars should be a similar shading, except if we are attempting to feature a couple of specific bars, or we have to bunch the bars into classifications of some sort. Here, we will make a horizontal bar chart with the bars on different shading. To do this, we will map the nation name to the y-axis and the confirmed count to the x-axis. The horizontal bar outline is greatly improved in many cases this way, in light of the fact that the names do not overlap when we put them on the y-axis. The entire thing is a lot simpler to peruse (Figs. 11.21, 11.22, 11.23, 11.24, 11.25, and 11.26).
Fig. 11.21  Top 20 countries COVID-19 active cases: code

Fig. 11.22  Top 20 countries COVID-19 active cases worldwide as of June 24, 2020

Fig. 11.23  Top 20 countries COVID-19 death numbers: code
In this chapter, initially we showed the complete number of instances of COVID-19 alongside the absolute recovered and losses over the continents with the assistance of plots and outlines. From the data introduced and cases assessed above, we infer that among various countries, the USA has been seriously affected by COVID-19 as contrasted with different nations [18, 19]. Various elements have driven a few nations to endure a higher infection rate than others. Subsequent to contemplating, we have arrived at the resolution that lockdown and travel limitations have assumed a significant role in containing COVID-19.

**Fig. 11.24** Top 20 countries COVID-19 death numbers worldwide as of June 24, 2020

**Fig. 11.25** Top 20 countries COVID-19 recovered cases: code

### 11.5 Conclusion

In this chapter, initially we showed the complete number of instances of COVID-19 alongside the absolute recovered and losses over the continents with the assistance of plots and outlines. From the data introduced and cases assessed above, we infer that among various countries, the USA has been seriously affected by COVID-19 as contrasted with different nations [18, 19]. Various elements have driven a few nations to endure a higher infection rate than others. Subsequent to contemplating, we have arrived at the resolution that lockdown and travel limitations have assumed a significant role in containing COVID-19.
We have likewise presumed that because COVID-19 has an elevated number of cases that this implies it has a noteworthy infectiousness as compared with other plagues and has a lower death rate.

References

1. Mohammed, T.T., Al-ameen, S.: SSCSMCS 2019 data visualization with real world data using Python. ResearchGate, April (2019). https://www.researchgate.net/publication/333671090_Data_Visualization_with_Real_World_Data_Using_Python
2. Oberoi, A., Chauhan, R.: Visualizing data using Matplotlib and Seaborn libraries in Python for data science. Int. J. Sci. Res. Publ. 9(3), p8733 (2019). https://doi.org/10.29322/ijsrp.9.03.2019.p8733
3. Weihs, C., Ickstadt, K.: Data science: the impact of statistics. Int. J. Data Sci. Anal. 6(3), 189–194 (2018). https://doi.org/10.1007/s41060-018-0102-5
4. Sajadi, M.M., Habibzadeh, P., Vintzileos, A., Shokouhi, S., Miralles-Wilhelm, F., Amoroso, A.: Temperature and latitude analysis to predict potential spread and seasonality for COVID-19. [ISSRN 3550308] (2020)
5. Bukhari, Q., Jameel, Y.: Will coronavirus pandemic diminish by summer? [ISSRN 3556998] (2020)
6. Wang, J., Tang, K., Feng, K., Lv, W.: High temperature and high humidity reduce the transmission of COVID-19. [ISSRN 3551767] (2020)
7. Zhou, F., Yu, T., Du, R., Fan, G., Liu, Y., Liu, Z., Xiang, J., Wang, Y., Song, B., Gu, X., et al.: Clinical course and risk factors for mortality of adult inpatients with COVID-19 in Wuhan, China: a retrospective cohort study. Lancet. 395, 1054–1062 (2020)
8. Leung, C.: Risk factors for predicting mortality in elderly patients with COVID-19: a review of clinical data in China. Mech. Ageing Dev. 188, 111255 (2020)
9. Zhang, J., Litvinova, M., Liang, Y., Wang, Y., Wang, W., Zhao, S., Wu, Q., Merler, S., Viboud, C., Vespignani, A., et al.: Changes in contact patterns shape the dynamics of the COVID-19 outbreak in China. Science. 368(6498), 1481–1486 (2020)
10. Dandekar, R., Barbastathis, G.: Neural network aided quarantine control model estimation of COVID spread in Wuhan, China. arXiv, arXiv:2003.09403 (2020)
11. Covid, I., Murray, C.J.: Forecasting COVID-19 impact on hospital bed-days, ICU-days, ventilator-days and deaths by US state in the next 4 months. medRxiv (2020)
12. COVID19_line_list_data.csv. https://www.kaggle.com/sudalairajkumar/novel-corona-virus-2019-dataset
13. Giordano, G., Blanchini, F., Bruno, R., Colaneri, P., Di Filippo, A., Di Matteo, A., Colaneri, M.: Modelling the COVID-19 epidemic and implementation of population-wide interventions in Italy. Nat. Med. 6(6), 855–860 (2020)
14. Onder, G., Rezza, G., Brusaferro, S.: Case-fatality rate and characteristics of patients dying in relation to COVID-19 in Italy. J.A.M.A. 323(18), 1775–1776 (2020)
15. Pepe, E., Bajardi, P., Gauvin, L., Privitera, F., Lake, B., Cattuto, C., Tizzoni, M.: COVID-19 outbreak response: a first assessment of mobility changes in Italy following national lockdown. medRxiv (2020)
16. Gómez Expósito, A., Rosendo Macías, J.A., González Cagigal, M.Á.: Modelado y Análisis de la Evolución de una Epidemia Virica Mediante Filtros de Kalman: El Caso del COVID-19 en España; Technical report; Universidad de Sevilla: Sevilla, Spain (2020)
17. Oliver, N., Barber, X., Roomp, K., Roomp, K.: The covid19 impact survey: assessing the pulse of the COVID-19 pandemic in Spain via 24 questions. arXiv, arXiv:2004.01014 (2020)
18. Sayre, R., Karagulle, D., Frye, C., Boucher, T., Wolff, N.H., Breyer, S., Touval, J.: An assessment of the representation of ecosystems in global protected areas using new maps of world climate regions and world ecosystems. Global Ecol. Conserv. 21, e00860 (2020). https://doi.org/10.1016/j.gecco.2019.e00860
19. Average yearly temperature by country, Lebanese Economy Forum. https://web.archive.org/web/20150905135247/http://lebanese-economyforum.com/wdi-gdf-advanced-datadisplay/show/EN-CLC-AVRT-C/. Last Accessed 13 Apr 2020