European Journal of Technique

journal homepage: https://dergipark.org.tr/en/pub/ejt

Vol.15, No.1, 2025

Research Article

A Perspective View on Data Visualization Libraries Used in Data Analytics

Faruk Aksoy[1*], Mehmet Ozdem[2], Resul Das[3]

[1*]Dicle University, Ergani Vocational School, Department of Computer Technologies, Diyarbakir, Türkiye. (e-mail: faruk.aksoy@dicle.edu.tr).
[2] Türk Telekom, Ankara, Türkiye. (e-mail:mehmet.ozdem@turktelekom.com.tr)
[3]Firat University, Faculty of Technology, Department of Software Engineering, 23119 Elazig, Türkiye. (e-mail: rdas@firat.edu.tr).

ARTICLE INFO

Received: Jan., 9. 2025
Revised: Jan., 25. 2025
Accepted: Feb., 16. 2025

Keywords:
Data Visualization
Visualization Libraries
Data Visualization Libraries
Data Analytics

Corresponding author: *Faruk Aksoy*

ISSN: *2536-5010* | e-ISSN: *2536-5134*

DOI: https://doi.org/10.36222/ejt.1616824

ABSTRACT

In today's data-driven world, one of the cornerstones of data science is data visualization. Data visualization makes information more understandable by presenting complex data sets through graphs, tables, and other visual tools. Thus, trends, relationships, and anomalies in the data can be easily detected. In this way, more intuitive and meaningful results are obtained from the data. In addition, decision makers are able to make the right decisions faster thanks to these visualization tools. Graphics libraries created and used in programming languages are used intensively to perform these operations. This article explores programming language-based data visualization libraries, which are widely utilized in the field of data science, and examines the key features and functionalities of these libraries. In this context, 76 visualization libraries were examined and evaluated in terms of customization level, interactive features, and the data types they support, and presented separately in tables according to programming languages. The study areas of the articles in which these libraries were used in the literature and the results obtained from the studies were also presented. In addition, the directions of future studies in this field were added to the conclusion.

1. INTRODUCTION

Data visualization is constantly changing as a developing field of data science. The amount of data loaded into data stores and flow rates contribute greatly to this rapid change [1]. In the contemporary data-driven business landscape, visualization tools are integral to facilitating decision-making processes that directly influence the considerable revenues of various industrial enterprises, thereby underscoring their critical importance in both academic research and industry applications. Because the transformation of abstract data into physical images (length, location, shape, color, etc.) is achieved through visualization [2]. Data-driven solutions that will be carried out in analytical processes related to data received from smart cities, smart systems, web-based systems and wireless sensors in today's technology world, which operates with the production and consumption [3] of such abstract data, are seen as an important field of study in the future [4], [5]. Since the data produced on these platforms is within the scope of big data [6], it is necessary to work with a structure where the data is constantly updated. For this reason, the data is structured and analyzed first. Artificial intelligence-supported intelligent systems can be used to collect data and provide meaningful inferences. In addition, appropriate mechanisms are applied to detect anomalies in the visualized information [7].

When we look at the studies in the field of data visualization, it is seen that various visualization libraries are used. However, there is no specific standard for which libraries to be used in which situations. This article seeks to address the gap in existing literature by examining visualization libraries across various programming languages, emphasizing their advantages and limitations in the context of specific use cases. For this purpose, it is aimed to guide the library and purpose matching in future visualization studies by classifying the visualization libraries according to their years and usage purposes. As far as we are aware, no such study has been conducted previously. The process from obtaining the data to producing the image outputs is shown in Figure 1 below. The first step is carried out by analyzing the raw data from different sources. The data that needs to be worked on is filtered and sorted out from the prepared data. After matching the data to be focused on with each other with meaningful relationships, the image data is produced by processing the geometric data obtained on the graph. These processes can also be used as a common workflow that can be used in all data visualization steps [8], [9], [10].

This study consists of four main parts. In Section 2, data visualization tools/libraries that are used extensively in programming languages are presented. Comparisons of the advantages and disadvantages of these tools are also presented in tables. In Section 3, the findings from the literature review with application examples of visualization tools are presented. In Section 4, the analysis of visualization tools and possible future directions are presented and discussed.

2. DATA VISUALIZATION TOOLS

This section discusses libraries used for visualization based on programming languages. Visualization libraries are used in industrial/academic fields and continue to be developed depending on the changing data structure. It is seen in the literature that visualization libraries are applied in very different disciplines such as health, economy, finance, cyber security [11], social media, and construction [12]. It can be used to analyze social networks [13] within the scope of big data, as well as for real-time modeling [14]. Visualization libraries in programming languages used in visualization processes are shown in Figure 1. On the other hand, studies are also being carried out to realize solutions related to all these steps with generative artificial intelligence [15]. Interactive visualization with generative artificial intelligence is one of these areas [16].

2.1. Javascript Based Tools

When we look at the studies conducted in the field of data visualization, there are many tools based on programming languages. These languages include JavaScript, R, Python, Java, Scala, Perl, Julia. The overwhelming majority of programming language-based visualization tools, particularly those that are most commonly used, are built on JavaScript. For this reason, tools based on JavaScript will be presented first in this section. JavaScript allows users to design their charts and graphs flexibly by processing raw data with their own style of code, but these design steps can be time-consuming for those new to programming languages. Some visualization tools are integrated with JavaScript, whereas others rely on programming languages like Python, Java, PHP and R. Visualization libraries in JavaScript are explained below.

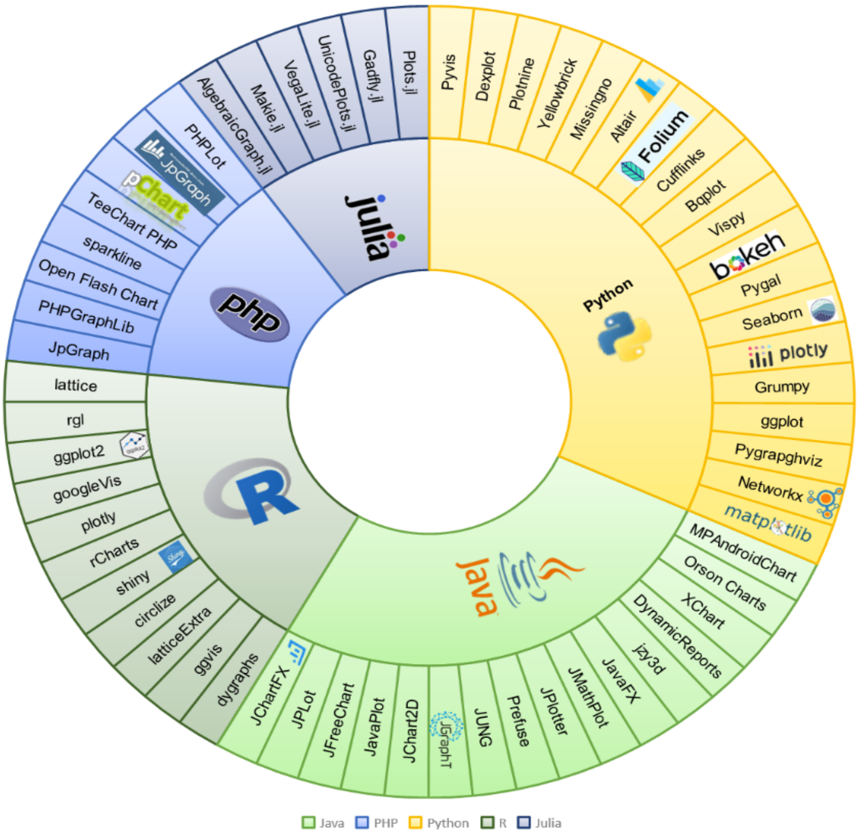

Figure 1. Distribution of graphics libraries by programming languages

TABLE 1 Javascript graphics libraries

References	Year	Tool	Features	Advantages	Disadvantages
[17]	2002	FusionCharts	Basic chart types, indicators, maps, custom visualization features	Rich chart options, easy integration	Paid license, some features limited
[18]	2004	Dojo	Charts, form elements, animations, data management	Versatile library, wide range of features	Performance issues with large data sets
[19], [20]	2005	Prototype.js	DOM manipulation, AJAX, form elements	Simple DOM manipulation, good performance	Limited chart and visualization support
[21]	2006	MooTools	Charts, animations, form elements, data management	Flexible and extendable, modern API	More complex configuration
[22]	2009	Protovis	Line charts, stacked bar charts, pie charts	Powerful data visualization, open source	Limited modern features, less up-to-date
[23]	2009	ZingChart	Various chart types, maps, network graphs, custom visualization features	Wide chart types, strong customization	Paid license, some features limited
[24]	2010	Flot	Basic chart types (line, bar, area, etc.)	Simple to use, good performance	Limited chart types and customization
[25]	2011	D3.js	Data visualization, animations, interactive charts, maps, network graphs	High customization, large community support	Learning curve, complex configuration
[26]	2013	Chart.js	Basic chart types (line, bar, pie, radar, etc.)	Easy to use, lightweight	Fewer customization options
[26]	2013	EChart.js	Various chart types, maps, network graphs, custom visualization features	Strong chart features, extensible	Performance limitations with large data sets
[27]	2014	C3.js	Built on D3.js, provides a simple API interface, line charts, bar charts, pie charts	Simple API, D3.js support	Does not support all D3.js features
[28]	2015	Vega	Data visualization, declarative charts, interactive charts, customization	Modern API, strong interaction	Learning curve, some performance issues
[29]	2010	Highcharts	Interactive charts (line, bar, pie, scatter, etc.), 3D charts, maps	Easy to integrate, rich features, good documentation	Paid license for commercial use

2.1.1. Protovis

Protovis enables the creation of custom data visualizations using basic markers such as bars and dots. It defines these markers through dynamic properties that represent the data, and incorporates inheritance, scales, and layouts to streamline the structure. Protovis is free and open-source, licensed under the BSD License. It utilizes JavaScript and SVG for web-based visualizations, requiring no plugins, though a modern web browser is necessary. Protovis is no longer in active development. The last version of Protovis was v3.3.1. The Protovis team has been developing the visualization library D3.js since June 28, 2011. D3 is based on many concepts from Protovis [22], [30], [31].

2.1.2. D3.js

D3.js is an open-source JavaScript library that combines HTML and CSS techniques to create dynamic, interactive data visualizations [32]. It leverages HTML, CSS, and SVG to generate visual representations of data, accessible in any browser. Moreover, the study [30] detailing the development of D3 demonstrates that it outperforms Protovis, achieving at least twice the speed in performance benchmarks. The D3.js website provides numerous examples, which can either inspire users to create their own visualizations or be used directly as templates [33].

2.1.3. Chart.js

Chart.js, launched in 2013, is an open-source library licensed under the MIT license and is actively maintained by a dynamic community. It includes built-in TypeScript support and is compatible with major JavaScript frameworks, such as React, Vue, Svelte, and Angular. The library is well-suited for handling large datasets, which can be efficiently retrieved in its native format, eliminating the need for data parsing and normalization. Alternatively, users can configure data sampling to reduce the size of the dataset before rendering [34]. In one paper, the JSTrace application designed to identify error sources in web applications evaluated Chart.js as the JavaScript library with the fewest errors [35].

2.1.4. Dojo

It is an open source, modular JavaScript library, launched in 2004, designed for developing cross-platform Ajax-based applications and websites [36]. Dojo's components, known as widgets, combine HTML, JavaScript, and CSS. They provide a wide range of interactive features, including menus, tabs, tooltips, tables, dynamic charts, 2D and 3D vector graphics, and tree-like widgets with drag-and-drop functionality.

A key feature of Dojo is its support for asynchronous communication between the browser and the server, a critical aspect of Ajax applications. Dojo provides an abstracted wrapper (dojo.xhr) for different web browser implementations, enabling seamless data communication between the client and server. This facilitates the storage of user input from forms on the server, with the server returning JavaScript code that dynamically updates the content of the page [37]. Additional capabilities of Dojo include both client- and server-side data storage, along with support for the Adobe Integrated Runtime (AIR) environment. The toolkit is officially supported by major companies such as IBM, Sun Microsystems (now Oracle), and Zend Technologies, the creators of PHP.

2.1.5. Prototype.js

The Prototype JavaScript framework was developed by Sam Stephenson in February 2005 to provide Ajax support for

Ruby on Rails. Typically implemented as a single JavaScript file, prototype.js, it is designed to simplify the development of dynamic web applications. Prototype, as an object-oriented framework, offers extensive support for Ajax, sophisticated programming components, and efficient DOM manipulation, thereby simplifying the process for developers to build interactive and responsive web interfaces [38], [39], [40].

2.1.6. Echart.js

Echart.js is a JavaScript library for charting and data visualization, designed to create intuitive, interactive, and highly customizable charts. It can run on PC and mobile devices. It depends on ZRender, a charting engine. It is compatible with web browsers such as Safari, IE8-11, Firefox, Chrome, etc. It supports charting in ECharts, Canvas, SVG (v4.0+) and VML formats [41].

2.1.7. FusionCharts

FusionCharts is a proprietary JavaScript library that integrates multiple technologies, including JavaScript and ActionScript 3.0. The library is compatible with a diverse array of devices, browsers, and platforms. It provides over 90 unique chart types and more than 1,000 maps, covering regions from every continent. FusionCharts supports the processing of data in .xml and .json formats and allows users to export the generated charts in .jpg, .png, and .pdf file formats. Furthermore, its functionality can be extended through various official plugins and wrappers, enabling the integration of interactive charts into applications built with technologies such as JSP, PHP, jQuery, and Django [42].

2.1.8. C3.js

C3 is a reusable D3-based charting library that enables the integration of charts into web applications [43]. C3 makes it easy to create D3-based charts by packaging the code required to create the entire chart. C3.js is a JavaScript library that enables the creation of visualizations through the use of HTML, SVG, and CSS. It proposals a vast range of APIs and callbacks, allowing users to interact with and access the state of the chart [44].

2.1.9. Flot

Flot is a JavaScript library for jQuery that emphasizes ease of use, customizable styling, and interactive chart features. It supports HTML5-based charts, leveraging Canvas and VML technologies. The library distinguishes the functional logic from the HTML structure, leveraging DOM elements to generate the charts. Flot includes pre-built components for four fundamental chart types: bar charts, line charts, point charts, and segment charts. Additionally, users can easily customize and extend these charts by modifying a wide array of configuration options [45].

2.1.10. ZingChart

ZingChart integrates seamlessly with various frameworks such as Angular, React, jQuery, PHP, Ember, and Backbone within its JavaScript library, allowing for a declarative approach to chart creation and management. ZingChart supports over 35 chart and model types. It allows users to export visualization graphs in .png, .jpg, and .pdf formats. It offers integrated graph editing capabilities. It has a drill-down function where users can select a data element within the graph [23].

2.1.11. Vega

Vega is a declarative language used to define, store, and share interactive visualization designs. With Vega, data visualizations are defined using a JSON style, allowing for the creation of interactive views through HTML5 Canvas or SVG.Vega-Lite, a higher-level language built upon Vega, enables the rapid creation of statistical graphs. [28]. If the working environment is Python instead of JavaScript, the Altair Python API is used for Vega-Lite [46].

2.1.12. Highcharts.js

Highcharts.js is a powerful and flexible data visualization library for modern web applications. It can process large datasets efficiently, thanks to its robust feature set and high performance. Its ability to generate interactive graphs makes it widely applicable in the fields of data science and software development.

The library offers a wide range of graph types. It offers a wide range of data visualization opportunities, from basic line and bar charts to complex financial and geographic maps. The dynamic and interactive charts it offers, especially Highcharts' Highstock and Highmaps add-ons, offer special solutions in the fields of financial data analysis and geographic data visualization. Thanks to these features, it is ideal for large data analysis and global data sets. Its extensible API and customizable building blocks allow users to control every aspect of the charts in detail. In addition, thanks to its mobile-friendly design and high-performance processing capabilities, it offers data visualizations seamlessly on every platform. Highcharts' commercial license model offers a suitable option for projects that require advanced features and support. These advantages make the library attractive for large-scale and commercial data visualization solutions [47].

2.2. Tools Based on Other Programming Languages

Besides JavaScript, there are a wide variety of data visualization tools and libraries available in other programming languages. There are widely used libraries in Java (Table 2), PHP (Table 3), R (Table 4), Julia (Table 5) and Python (Table 6), Scala, and Actionscript. These libraries are listed according to their production dates and the languages they belong to. Some of the most widely used libraries in different languages are described below.

EUROPEAN JOURNAL OF TECHNIQUE, Vol.15, No.1, 2025

TABLE 2 Java graphics libraries

Ref	Year	Tool	Features	Advantages	Disadvantages
[48]	1996	TeeChart for Java	Rich chart types, interactive visualization	Wide variety of chart types, high performance	Paid license, complex configuration
[49]	2005	JChartFX	Customizable charts, user-friendly interface	Easy to use, good performance	Limited customization options
[50]	2008	JPlot	Basic chart types, fast performance	Lightweight and fast	Few chart types and features
[51]	2000	JFreeChart	Wide range of chart types, good documentation	Widely used, open-source, large community support	Performance limitations with large data sets
[52]	2003	JavaPlot	Simple charts, mathematical functions	Simple and lightweight	Limited chart types and features
[53]	2004	JChart2D	2D charts, fast performance	Easy to use, fast rendering	Limited chart types and customization
[54]	2003	JGraphT	Focused on graph theory, supports weighted graphs	Strong graph theory processing, open-source	Limited visualization support
[55]	2004	JUNG	Network analysis, various graph algorithms	Comprehensive network analysis features	Complex usage, limited visualization
[56]	2005	Prefuse	Data visualization, dynamic interactions	Good visualization and interaction	Inadequate documentation, complex configuration
[57]	2007	JPlotter	Basic 2D and 3D charts, speed	Simple and fast	Limited features and chart types
[58]	2007	JMathPlot	Mathematical charts, 2D drawings	Suitable for mathematical data, easy to use	Few chart types and customization options
[59]	2008	JavaFX	Rich user interfaces, 2D and 3D charts	Modern interfaces, rich visualization	Performance issues with large data sets
[60]	2008	Jzy3d	3D charts, high performance	Impressive 3D charts, fast rendering	Complex configuration, learning curve
[61]	2011	XChart	Simple, lightweight charts	Easy to use, low memory usage	Few chart types and customization
[62]	2012	Orson Charts	2D and 3D charts, high-quality rendering	High-quality charts, flexible customization	Smaller community support
[63]	2013	MPAndroidChart	Rich chart types, interactive features	Optimized for Android applications	Only available for the Android platform

TABLE 3
Php graphics libraries

Ref	Year	Tool	Features	Advantages	Disadvantages
[64]	2004	PHPLot	Basic 2D charts, various chart types	Simple and lightweight, easy integration	Limited chart types and customization
[65]	2001	jpGraph	Rich chart types, high customization	Wide range of chart types, strong customization	Paid license, complex configuration
[66]	2005	pChart	Dynamic charts, various data formats	Strong visualization, good performance	Learning curve, some features are paid
[67]	2009	TeeChart PHP	Wide range of chart options, interactive charts	Wide variety of chart types, user-friendly	Paid license, complex configuration
[68]	2009	Sparkline	Small, minimalist charts	Lightweight, fast rendering	Limited chart types and features
[69]	2008	Open Flash Chart	Flash-based charts, interactive features	Web-based interaction, various chart types	Flash-based, limited mobile compatibility
[70]	2001	JpGraph	Rich chart types, high customization	Wide range of chart types, strong customization	Paid license, complex configuration

TABLE 4 R graphic libraries

Ref	Year	Tool	Features	Advantages	Disadvantages
[71]	2004	lattice	Multiple 2D charts, multidimensional data	Strong multidimensional data visualization	Learning curve, sometimes complex configuration
[72]	2008	rgl	3D charts, interactive visualization	Interactive 3D visualization, powerful	Performance issues with large data sets
[73]	2011	latticeExtra	Extra chart types, compatible with lattice	Additional features for lattice charts	More complex configuration
[74]	2005	ggplot2	Grammar of Graphics, rich chart types	Widely used, powerful customization	Performance limitations with large data sets
[75]	2010	googleVis	Google Charts integration, interactive charts	Web-based interaction, easy to use	Dependency on Google services
[76]	2013	plotly	Interactive charts, multi-language support	Rich interaction features, multi-language support	Paid plans and some features are limited
[77]	2014	rCharts	Various JavaScript libraries, interactive charts	JavaScript integration, interactive charts	Steeper learning curve, some features are paid
[78], [79]	2012	shiny	Web app development, interactive data visualization	Dynamic web applications, powerful interaction	Performance issues with large data sets
[80]	2014	circlize	Circular charts, complex data visualization	Advanced circular charts, customizable	Complex configuration, learning curve
[74]	2015	ggvis	Interactive charts, Grammar of Graphics	Interactive visualization, modern API	Advanced features are limited, some performance issues
[81]	2013	dygraphs	Time series charts, interactive	Strong time series visualization, interactive	Customization options are limited

TABLE 5 Julia graphic libraries

Ref	Year	Tool	Features	Advantages	Disadvantages
[82]	2015	Plots.jl	Multiple backend support (GR, PyPlot, Plotly, etc.), interactive and static charts	Easy to use, many backend supports, large user community	Performance can be limited in some cases, limited features for complex charts
[83]	2015	Gadfly.jl	Uses the Grammar of Graphics approach, offers extensive data visualization capabilities	Powerful and flexible, creates aesthetic charts	Performance can be low in some cases, setup and configuration can be complex
[84]	2018	Makie.jl	2D and 3D charts, high performance, interactive visualization	High performance, a wide variety of chart types, interactive capabilities	Learning and usage difficulty, potential library compatibility issues
[85]	2017	VegaLite.jl	Compatible with Vega-Lite, uses the Grammar of Graphics approach	High customization, ability to create interactive charts	Customization and performance limitations
[86]	2016	UnicodePlots.jl	Creates charts with ASCII and Unicode characters	Lightweight, fast, terminal-based visualization	Visually limited, lacks advanced chart features

TABLE 6 Python graphics libraries

Ref	Year	Tool	Features	Advantages	Disadvantages
[87]	2003	Matplotlib	2D charts, extensive customization options	Strong community support, wide range of chart types	Performance limitations, some features may be complex
[88]	2004	Networkx	Graph theory and network visualization	Strong network analysis tools, well documented	Not for general data visualization, network-focused
[89]	2006	PyGraphviz	Graph theory and network visualization, Graphviz interface	Graphviz integration, strong network visualization	Setup and dependencies may be complex
[90]	2010	ggplot	Grammar of Graphics approach, 2D charts	Strong and aesthetic charts, easy to use	Performance limitations, limited 3D support
[91]	2013	Glumpy	Fast 2D and 3D charts, uses OpenGL	High performance, interactive charts	Difficult to use, limited community support
[92]	2013	Plotly	Interactive charts, multiple chart types	Highly interactive, web-based visualization	Limited features in free version
[93]	2012	Seaborn	Statistical data visualization, built on Matplotlib	Aesthetic and understandable charts, easy to use	Limited customization options, dependency on Matplotlib
[94]	2012	Pygal	Web-based SVG charts, easy to use	Lightweight, aesthetic, and dynamic charts	Limited chart types and customization options
[95]	2013	Bokeh	Interactive charts, large data support	Web-based interactions, high performance	May have a learning curve, complex setup
[96]	2014	Vispy	High-performance 2D and 3D charts, uses OpenGL	Performance, large data, and interactive charts	Learning curve, limited community support
[97]	2015	Bqplot	Interactive charts for Jupyter Notebooks	Jupyter-compatible, interactive visualization	Limited chart types, low community support
[98]	2015	Cufflinks	Integrated with Pandas DataFrames, Plotly-based	Integration with Pandas, interactive charts	Limited customization, dependent on Plotly
[99], [100]	2014	Folium	Map visualization, based on Leaflet.js	Map-focused, web-based visualization	Not for general data visualization, map-focused
[101]	2016	Altair	Declarative data visualization, based on Vega-Lite	High-level API, aesthetic charts	Limited 3D and interactive chart support
[102]	2017	Missingno	Missing data visualization, simple charts	Specialized for missing data analysis and visualization	Limited chart types, only for missing data analysis
[103]	2016	Yellowbrick	Machine learning visualizations, integrated with Scikit-learn	ML model evaluation tools, easy to use	Limited data visualization, ML-focused
[104]	2018	Plotnine	Grammar of Graphics-based, similar to ggplot2	Strong and aesthetic charts, ggplot2-like in Python	Performance limitations, some features missing
[105]	2018	Pyvis	Interactive network visualization, integrated with Networkx	Easy interactive network graphs, good visualization	Limited data types, performance limitations
[106]	2014	Holoviews	Interactive and large data visualization	Easy to use, powerful interactive charts	Performance limitations, learning curve
[107]	2016	Datashader	Optimized for large data visualization	Fast large data visualization, effective rendering	Limited interactivity, integration issues with other libs
[108]	2012	Geopandas	Geographic data visualization, Pandas-based	Integration with geographic data, powerful features	Not for general data visualization, geography-focused

2.2.1. Gephi

Gephi is a free, open-source software developed for the purpose of network visualization and analysis. It is developed in Java, utilizing the NetBeans platform. Gephi can display the spatialization process in real time. The default layout algorithm employed is ForceAtlas2, a continuous force-directed layout method. Users have the option to import .csv data files or directly input their data into the Gephi spreadsheet. The data file is structured into two sections: the edge table and the node table. Consequently, users must prepare and organize their data into these two components in advance. It supports up to one million edges and nodes. Once the data is imported, the visualization is automatic. Users can choose an algorithm (ForceAtlas, ForceAtlas2, Fruchterman Reingold, Noverlap, etc.) to analyze the network. The generated network graph can be exported directly in .svg or .pdf formats [25], [109], [110].

2.2.2. jpGraph

jpGraph is an object-oriented library for creating graphs. If server-side graphs need to be generated, the library is ready to be used in any language. The library, which has free and paid (pro) versions, is based on PHP5 (versions above 5.1) and PHP7. It was written for PHP by Asbjørn Ulsberg. The commercial professional version of jpGraph supports additional chart types such as odometer, wind rose (a special diagram representing the distribution of meteorological data, wind speeds by class and direction, also known as a polar rose chart), and barcodes [111].

2.2.3. Matplotlib

Matplotlib is an extensive library in Python used for generating static, animated, and interactive visualizations. Some of the visualization libraries written in Python include Plotly [112], Seaborn [113], Altair [114], Pygal [115], Bokeh [116], Gleam [117], and Matplotlib [118]. Matplotlib allows you to visualize data interactively and prepare high-quality outputs suitable for publication. Both two-dimensional and three-dimensional charts can be produced. It is also a popular data visualization package that works well, including ipython shell, web application server, graphical UI toolkit, and Jupyter notebook [119], [120].

2.2.4. Folium

Folium is a library for interactive map drawing based on leaflet.js module in JavaScript language, but performs data manipulation in Python. Folium simplifies the process of visualizing data manipulated in Python on an interactive leaflet map. The library offers several built-in tile sets from sources like OpenStreetMap, Mapbox, and Stamen, and it also supports custom tile sets using Mapbox or Cloudmade API keys. Additionally, Folium accommodates various textures, including image, video, GeoJSON, and TopoJSON formats [121]. It is built with simplicity, performance, and utility in mind. It is highly performant, has a user-friendly API, and can be enhanced with various plugins [99], [100].

2.2.5. GraphX in Spark

Apache Spark is an open source library developed in Scala language that enables parallel processing on large data sets. Spark can run faster than Apache Hadoop in big data applications with its in-memory data processing feature. Therefore, it is possible to say that Spark's analytics engine can perform faster operations by keeping data in memory while processing instead of reading or writing data from disk. Apache Spark has various libraries for visualization, machine learning, and streaming operations. One of these libraries, GraphX, is a built-in library for graph analytics and graph parallel computing in Apache Spark [13], [122].

2.2.6. Plotly

Developed by a company called Plotly Inc. It is a Python library designed for generating interactive visualizations. It is also a free and open source library compatible with JavaScript and R. Its source can be viewed on GitHub [123]. Plotly Inc. is a technology company specializing in the development of online data analytics and visualization tools. The company offers online tools for graphing, analytics, and statistics, along with scientific graph libraries for various programming languages including Python, R, MATLAB, Perl, Julia, Arduino, and REST [76].

2.2.7. Prefuse

Prefuse is a Java-based tool for creating interactive information visualization applications. It supports a rich feature set for data modeling, visualization, and interaction. It provides support for tables, graphs, trees, animation, dynamic queries, integrated search and database connectivity, and optimized data structures. It is written in Java 1.4 using the Java 2D graphics library. It can be easily integrated into Java Swing applications or web applications. Prefuse is a product of the UC Berkeley Visualization Lab and is licensed under the terms of the BSD license. It may be used freely for both commercial and non-commercial purposes [124].

2.2.8. Flare

Flare is an ActionScript-based visualization library, built on the Prefuse framework, designed to create a diverse range of interactive visualizations that operate within the Adobe Flash Player. Flare, developed by the UC Berkeley Visualization Lab, is an ActionScript 3-based toolkit designed to support data management, visual coding, animation, and interaction techniques [125]. It enables the creation of a wide range of visualizations, from basic tables and charts to intricate interactive graphics [126]. Created by Jeff Heer, Flare is used to produce everything from simple charts and complex animations to network diagrams, tree maps, and more.

2.2.9. Weave

The **We**b-based **A**nalysis and **V**isualization **E**nvironment offers a range of visualization tools that enable users to rapidly and effectively expand their analysis. It is open sourced by the OIC (Open Indicators Consortium). It is maintained in partnership with the Institute for Visualization and Perception Research (IVPR) at the University of Massachusetts Lowell. Weave is designed to enable users to analyze, visualize, and disseminate data and indicators at geographically nested levels, from anywhere, at any time [127] [128]. It presents data in multiple formats as an intuitive application. This feature enables users to easily observe the relationships between datasets. When a user selects a data value in one of the visualization windows, the corresponding value is highlighted on the map, as well as in other tables and visualizations. Users can also incorporate additional data, access a range of tools, and save Weave sessions for future use or sharing with other users [129], [130].

2.2.10. Ggplot2

Ggplot2 is an open-source R language software package for statistical data visualization and graphics. It is grounded in the concepts of the "Grammar of Graphics [131]," a framework developed by Hadley Wickham. This approach defines the building blocks of data visualization and makes the graphic creation process modular and reusable. Graphics are a combination of components such as aesthetics (aes), geoms, stats, coordinates, and facets in ggplot2 [132].

2.2.11. Seaborn

Seaborn is a highly influential data visualization library in the Python ecosystem, playing a key role in the visualization of statistical analysis. It is built on top of Matplotlib and offers a high-level API that facilitates the creation of more complex and aesthetically superior plots. It provides comprehensive tools for deep understanding of the internal structure and relationships of data sets. Functions such as pairplot, heatmap, violin plot provide advanced capabilities for multidimensional data analysis and pattern discovery. Since pairplot allows multidimensional visualization of relationships and distributions between variables, it provides a critical advantage in modeling and anomaly detection processes. The heatmap function is used to visualize dense data structures such as correlation matrices, allowing visual analysis of relationships between data [133].

2.2.12. Ggvis

ggvis is a library that can create interactive and dynamic graphics among the data visualization tools in the R language. While ggvis is built on the Grammar of Graphics [131] principles established by ggplot2, it enhances the visualization functions by making them interactive, enabling users to engage with the data in real time. The ability to integrate with shiny [79] offered by ggvis provides a great advantage in web-based data analysis and visualization applications. In addition to creating interactive graphics in data visualization processes, it offers a powerful API that supports user interactions and dynamic data updates. ggvis's high-performance and fast graphics using WebGL and SVG technologies in the background provide a significant performance advantage when working with large data sets. The dynamic features of ggvis offer data scientists the ability to explore the internal structure of datasets and conduct more detailed analysis by receiving immediate feedback during decision-making. This makes ggvis a powerful tool, particularly for exploratory data analysis and user-focused data visualization projects [74].

TABLE 7 Types supported by visualization libraries

Library	Bar	Line	Pie	Radar	Scatter	Bubble	Heatmap	3D	Other Graphic Types	WebGL Support
FusionCharts	✓	✓	✓	✓	✓	✓	✓	✓	Fuzzy Charts, Fast Data Visualization	✓
Dojo	✓	✓	✓	✓	✓	✓	✓	✓	Conditional Labeling	✓
Prototype.js	✓	✓	✓	✓	✓	✓	✓		Data Binding	
MooTools	✓	✓	✓	✓	✓	✓	✓		Animations	
Protovis	✓	✓	✓	✓	✓	✓	✓		Clustering, Hierarchy	✓
ZingChart	✓	✓	✓	✓	✓	✓	✓	✓	Fast Data Binding	✓
Flot	✓	✓	✓		✓	✓	✓		Live Data Loading	
D3.js	✓	✓	✓	✓	✓	✓	✓	✓	Network Charts, Maps	✓
Chart.js	✓	✓	✓		✓	✓	✓		Innovative Animations	
ECharts	✓	✓	✓	✓	✓	✓	✓	✓	Map Visualization	✓
C3.js	✓	✓	✓		✓	✓	✓		Hierarchical Data	
Vega	✓	✓	✓	✓	✓	✓	✓	✓	Conditional Data Visualization	✓
Plots.jl	✓	✓	✓	✓	✓	✓	✓		Conditional Data	
Gadfly.jl	✓	✓	✓	✓	✓		✓		Conditional Data	
Makie.jl	✓	✓	✓	✓	✓	✓	✓	✓	3D, Interactive Data	✓
VegaLite.jl	✓	✓	✓	✓	✓	✓	✓		Conditional Data	
AlgebraicGraph.jl					✓				Network Charts	
UnicodePlots.jl	✓	✓	✓		✓				Time Series	
Plots.jl	✓	✓	✓	✓	✓	✓	✓		Conditional Data	
Gadfly.jl	✓	✓	✓	✓	✓		✓		Conditional Data	
Makie.jl	✓	✓	✓	✓	✓	✓	✓	✓	3D, Interactive Data	✓
VegaLite.jl	✓	✓	✓	✓	✓	✓	✓		Conditional Data	
AlgebraicGraph.jl					✓				Network Charts	
UnicodePlots.jl	✓	✓	✓		✓				Time Series	
lattice	✓	✓	✓	✓	✓		✓		Conditional Data	
rgl	✓	✓	✓		✓			✓	3D, Interactive Data	✓
latticeExtra	✓	✓	✓		✓				Conditional Data	
ggplot2	✓	✓	✓	✓	✓	✓	✓		Conditional Data	
googleVis	✓	✓	✓	✓	✓	✓	✓		Map Visualization	
plotly	✓	✓	✓	✓	✓	✓	✓	✓	3D, Map Visualization	✓
rCharts	✓	✓	✓	✓	✓	✓	✓		Map Visualization	
shiny	✓	✓	✓		✓				Real Time Data	

circlize	✓	✓	✓		✓				Pie Charts	
ggvis	✓	✓	✓	✓	✓	✓	✓		Conditional Data	
dygraphs	✓	✓	✓		✓		✓		Time Series	
TeeChart for Java	✓	✓	✓	✓	✓	✓	✓	✓	3D, Network, Hierarchy	✓
JChartFX	✓	✓	✓	✓	✓	✓	✓		3D, Fast Data	✓
JPlot	✓	✓	✓	✓	✓	✓	✓		Conditional Data	
JFreeChart	✓	✓	✓	✓	✓	✓	✓		3D, Network, Hierarchy	
JavaPlot	✓	✓	✓		✓	✓			Conditional Data	
JChart2D	✓	✓	✓		✓				Conditional Data	
JGraphT					✓				Network Chart Types	
JUNG					✓				Network Chart Types	
Prefuse	✓	✓	✓	✓	✓	✓	✓		Conditional Data	
JPlotter	✓	✓	✓		✓				3D, Animation	
JMathPlot	✓	✓	✓		✓				Hierarchy Visualization	
JavaFX	✓	✓	✓	✓	✓	✓	✓	✓	3D, Network, Hierarchy	✓
Jzy3d	✓	✓	✓		✓		✓	✓	3D, Fast Data	✓
DynamicReports	✓	✓	✓	✓	✓	✓			Conditional Data	
XChart	✓	✓	✓		✓				Conditional Data	
Orson Charts	✓	✓	✓		✓			✓	3D, Fast Data	✓
MPAndroidChart	✓	✓	✓		✓	✓	✓		Fast Data, Real Time	
Matplotlib	✓	✓	✓	✓	✓	✓	✓		Conditional Data	
Networkx					✓				Network Charts	
PyGraphviz					✓				Network Charts	
ggplot	✓	✓	✓	✓	✓	✓	✓		Conditional Data	
Glumpy	✓	✓	✓		✓			✓	3D, Fast Data	✓
Plotly	✓	✓	✓	✓	✓	✓	✓	✓	3D, Map Visualization	✓
Seaborn	✓	✓	✓		✓	✓	✓		Time Series	
Pygal	✓	✓	✓		✓				Conditional Data	
Bokeh	✓	✓	✓	✓	✓	✓	✓		Map Visualization	
Vispy	✓	✓	✓		✓			✓	3D, Interactive Data	✓
Bqplot	✓	✓	✓		✓				Time Series	
Cufflinks	✓	✓	✓	✓	✓	✓	✓		Conditional Data	
Folium			✓		✓				Map Visualization	
Altair	✓	✓	✓	✓	✓	✓	✓		Conditional Data	
Missingno	✓	✓	✓		✓		✓		Data Cleaning	
Yellowbrick	✓	✓	✓		✓				Machine Learning	
Plotnine	✓	✓	✓		✓				Conditional Data	
Dexplot	✓	✓	✓		✓				Time Series	
Pyvis					✓				Network Charts	
Holoviews	✓	✓	✓	✓	✓	✓	✓		Conditional Data	
Datashader	✓	✓	✓		✓				Big Data Visualization	
Geopandas			✓		✓				Map Visualization	

3. DATA ANALYTIC APPLICATION EXAMPLES

In this section, studies using visualization libraries are examined. As seen below, these results are presented in Table 8. When these studies are examined, it is seen that these visualization libraries are used in many different disciplines and for various purposes. Application articles have been produced in areas such as computer systems, web page design, web security, chemistry, astronomy, data science, smart systems, smart cities, biology, agriculture, microbiology, psychology, big data, network security, health, and mobile applications. It can be said that Python, R, and Julia tools have been used more intensively in these studies in recent years. It can also be seen that Java libraries are used for mobile applications.

TABLE 8. Application examples of visualization tools

Ref.	Date	Library	Application Field	Conclusion
[1]	2012	FusionCharts	Mainframe performance analysis system	The degree of regression has been improved to more than 97% correlation
[2]	2011	Dojo	Dynamic tree menus and categorical classification	Tree nodes can have dynamic effects such as adding, deleting, and dragging.
[19]	2015	Prototype.js	Open online courses (MOOCs) and semantic web technologies	A MOOC management system prototype powered by semantic technologies has been developed
[20]	2011	MooTools	Web security and digital rights management	Development of anti-hotlinking framework
[22]	2009	Protovis	Data visualization	Production of the Protovis tool

[23]	2022	ZingChart	Astronomy and astrophysics	A system for finding the recorded flux at a particular sky position for X-ray observations.
[24]	2014	Flot	Fusion science and remote control systems	Remote access to a web-based user interface, solving the problem of high network latency
[26]	2018	EChart.js	Web-based interactive visualization tools and visualization development frameworks	Development of ECharts framework
[27]	2019	C3.js, D3.js	Developing web-based applications for honeybee colony monitoring and data visualization	Development of the Beemon system
[28]	2017	Vega-Lite	Interactive data visualization	Evaluation of the visualization grammar Vega-Lite
[29]	2017	Highcharts	Visualization of micro video big data	Visualizing micro video data with Highcharts
[64]	2016	PHPLot	Forecasting water levels, currents and waves for web-based information systems and coastal management	Development of Jeddah CIS system
[65]	2006	jpGraph	Electronic logbook and experimental data management	The web based logbook and run information interface
[66]	2012	pChart	Analysis and visualization of multivariate health data with geographic information systems	Development of the web application Community Health Map
[67]	2015	TeeChart PHP	Water pollution control and environmental decision support systems (EDSS)	Integration of EDSS environmental simulation models into the Beiyun River
[68]	2015	Sparkline	Smart grid data visualization and analysis of electrical grid data	Effective visualization of large data sets with Sparklines
[69]	2021	Open Flash Chart	Genomic analysis	Analyses on the Sorghum genome and transcriptome
[70]	2012	JpGraph	Graphical web presentation of thermochemical properties	Two auxiliary functions have been developed for solving graphics courses with Jpgraph support.
[48]	2018	TeeChart for Java	Sentiment analysis and comparison of Lexicon-based approach and Supervised Machine Learning approaches in analysis	The accuracy of Urdu Sentiment Analyzer is 89.03% with 0.86 precision, 0.90 recall and 0.88 F-measure
[49]	2019	JChartFX	Smart cities	Drafting a suitable Enterprise Architecture (EA) for Mauritius Fire Rescue Services (MFRS)
[50]	2019	JPlot	Internet of Things (IoT)	Images taken with Raspberry Pi Zero and Raspberry Pi Camera were processed with artificial neural networks (ANN) to estimate blood glucose levels with mean absolute error (10.37%) and Clarke Grid error (90.32% Region A).
[62]	2016	JFreeChart	Relational databases and tabular data	Introduces a new software called Thoth that facilitates the visualization and analysis of data in various ways
[52]	2012	JavaPlot	Medical electronics and ECG analysis	Non-linear transformations and artificial neural networks are used for ECG signal recognition and classification
[53]	2014	JChart2D	Examining the accuracy of SLOC (source lines of code) prediction models	The prediction model provides the highest accuracy (average MMRE = 0.19 and average Pred(25) = 0.74)
[54]	2022	JGraphT	Protection against phishing threats	Proposes a fake site detection method that checks domain similarity using the Levenshtein metric
[55]	2009	JUNG	Analysis of semantic networks	Semantic network analysis with graphOnt library and JUNG framework
[56]	2012	Prefuse	Interactive visualization of academic research	It offers an information visualization system developed using XML and Java.
[57]	2019	JPlotter	Analyze nonlinear constraint optimization problems	Provides a visual representation of the evolution of the optimization process
[58]	2015	JMathPlot	Toxicity of nanomaterials	Development of nano-QSAR modeling software
[59]	2019	JavaFX	Framework for forensic investigation of drone-related crimes	Framework for forensic investigation of drone-related crimes
[60]	2017	Jzy3d	Modelling flow dynamics in lowlands	Development of Two-Scale Cellular Automation
[61]	2023	XChart	Operation and maintenance (O&M) processes of urban underground pipe networks	It presents a mobile augmented reality (MAR) based visualization framework.
[62]	2016	Orson Charts	To query, display, visualize and analyze tablesal data stored in relational databases and data files	It offers an easy-to-use data visualization and statistics software called Thoth
[63]	2023	MPAndroidChart	Collection of patient -induced health data (PGHD) and integration of these data into the hospital information system	It provided easy integration of patient data by establishing a two-way connection between the mobile application and the hospital information system via QR code.
[71]	2009	lattice	Represent biological data	Generalized lattice –like graph approach
[72]	2020	rgl	Investigation of the structural features of open source software package ecosystems using complex network analysis (CNA) methods	Complex network analysis tools provide an effective method for evaluating software package ecosystems.
[73]	2024	latticeExtra	Mapping of the water status of citrus gardens to determine field -based management zones in precision agriculture.	By reliably estimating the spatial variability of water potential in orchards, scientifically based classification has been shown to be important for precise irrigation strategies and decision support processes.
[74]	2024	ggplot2, ggvis	Data visualization	The developed VisAhoi library supports the creation of placement elements for different visualization types and datasets.
[75]	2017	googleVis	Modeling and analyzing the interactions between intestinal microbiota and its surroundings	The developed compuGUT simulation tool offered an effective calculating approach to examine the intestinal microbiota dynamics
[76]	2023	plotly	Developing web -based interfaces in order to facilitate data visualization processes of users with limited programming knowledge.	The developed web-based application allowed users to easily visualize data sets and analyze correlations between variables, providing an accessible and effective solution, especially for users with limited programming knowledge.

[77]	2022	rCharts	Performing daily global solar radiation estimates for different cities in Europe and creating future projections according to climate change scenarios.	Six different machine-learning algorithms were generally able to estimate global solar radiation with high accuracy.
[78]	2024	shiny	Developing an indicator panel that associates academic programs with career and income data	Detailed explanation of how to program and publish a dashboard that correlates academic programs with career and income data using R and Shiny
[80]	2022	circlize	Analysis of the molecular structure of consumed carbohydrates	An open-access database called the "Davis Food Glycopedia (DFG)" was created by determining the monosaccharide compositions of more than 800 foods commonly consumed in complementary foods and adult diets.
[81]	2022	dygraphs	Measurement and verification of building energy efficiency and energy savings	A web application has been developed that continuously monitors and documents energy savings.
[82]	2023	Plots.jl	Modeling the growth dynamics of Bio-Film communities, biotechnology, environmental engineering, and microbial ecology	A Julia-based software called Biofilm.jl was developed for simulation and analysis
[84]	2024	Makie.jl	Lithium-ion Battery (LIB) Research, Machine Learning (ML) Applications	Evaluating the applicability of the Variational Autocoder (VAE) model helps identify dataset features for data preprocessing planning and discusses the interpretability of the ML model
[86]	2024	UnicodePlots.jl	Programming languages, scientific calculation, software engineering, performance analysis.	Compares the potential of Julia and Rust languages in scientific computing
[87]	2024	Matplotlib	Artificial intelligence, health informatics, eye movement tracking, autism spectrum disorder (ASD) diagnosis, neurological disorders	It offers an efficient, low-cost solution for ASD diagnosis using the Convolutional Neural Network (CNN) model.
[88]	2023	Networkx	Electricity transmission networks planning, network analysis	It presents a methodology with the NetworkX library to reduce the number of connection options of electrical transmission networks.[17]
[89]	2025	PyGraphviz	Cybersecurity, malware analysis	A new analysis technique is proposed for malware analysis using graph embedding networks.
[90]	2018	ggplot	Cyber security, malware detection (malware), graphic neural networks	The detection model developed by combining deep neural network and graph embedding methods gives 97.7% accuracy, 96.6% sensitivity, 96.8% recall and 96.4% F1-score.
[91]	2014	Glumpy	Parallel Calculation, High Performance Calculation	With the use of local memory, it aimed to improve performance by eliminating global memory access delay and analyzed the performance differences between local memory and global memory on eight different calculation devices.
[92]	2024	Plotly	The spread of sound waves, material science	Provides information such as phase and group velocities, power flow angle, enhancement factors, and polarization vectors based on elastic moduli and density of the material with the Python code SEISMIC
[93]	2024	Seaborn	Psychology, Machine Learning (ML) Techniques	Decision trees, random forests, gradient boosting, stochastic gradient boosting and XGBoost algorithms and their applications in psychology are explained and visualized in the Python environment.
[94]	2020	Pygal	Materials science, thermochemistry	Converting data into thermochemical properties and performing thermodynamic analyses with the Python-based pMuTT (multiscale thermochemistry toolbox) software library
[95]	2023	Bokeh	Decision support systems, fuzzy cognitive mapping	An open source web-based application called In-Cognitive has been developed using Bokeh, which guarantees solution convergence and is capable of performing Monte Carlo uncertainty analysis.
[96]	2022	Vispy	Structural analysis, modeling of historical structures	The software, Cloud2FEM, has been developed as an open source Python-based tool that automatically generates finite element meshes (FEM) from point cloud data of historical and existing structures.
[97]	2021	Bqplot	Hydrology, water resources management, reservoir modeling and optimization	Python-based tool called iRONS (Interactive Reservoir Operations Notebooks and Software) was developed for reservoir modeling and optimization.
[98]	2022	Cufflinks	Road safety, winter road maintenance management (WRM), data envelopment analysis	A method to estimate road surface temperature using data envelopment analysis (DEA) and machine learning techniques is proposed to improve the effectiveness of winter road maintenance management (WRM). [18]
[120]	2022	Folium	Big data analytics, data visualization, exploratory data analysis (EDA)	Data visualization and analysis techniques were examined with Python, and exploratory data analysis (EDA) methods were applied. [19,20]
[100]	2021	Folium	Human mobility analysis, social media data mining, big data analytics	The spread of Covid-19 from China and neighboring East Asian countries to the world has been studied using real-time travel data from social media platforms.
[101]	2022	Altair	Hydrology and climate change, water resources management	It shows that the Autocorrelation Function (ACF) method has the highest accuracy rate and especially the Boosted Decision Tree Regression (BDTR) model provides high R^2 values in all scenarios.
[102]	2023	Missingno	Network Security and intrusion detection, detection of DDoS attacks	Random Forest algorithm showed the best performance with 97.6% accuracy rate, K-Nearest Neighbors (KNN) and Logistic Regression with 97% and 91.1% accuracy rate respectively.

[103]	2024	Yellowbrick	Waste plastic management and energy production	It shows that the Random Forest model exhibited the best performance on the test data, obtaining an R2 value of 0.941, an RMSE value of 14.69 and an MAE value of 8.66
[104]	2021	Plotnine	Heart rate (HR) detection and bioengineering	The CWT (derivative Gaussian) method achieved superior results compared to MODWT-MRA, CWT (frequency B-spline) and CWT (Shannon) methods.
[105]	2024	Pyvis	Health data analysis, interactive visualization and data analytics	Proposes a disease monitoring and impact analysis methodology based on 71,849 patients
[106]	2024	Holoviews	Genomic research and neuroscience	Presented transcriptomic analysis of 5-HT receptors, revealing differential distribution and prevalence of each Htr subtype across cell classes
[107]	2024	Datashader	Proteomic data analysis and visualization	An open source software package called AlphaRms has been developed to process high-resolution MS data.
[108]	2023	Geopandas	Human mobility analysis and data modeling	Provides the open source Trackintel Python library for human mobility analysis that adapts to different tracking data types

The integration capabilities of data visualization libraries are essential, particularly in the context of big data processing and interdisciplinary projects, as they facilitate seamless data analysis, enhance collaboration across diverse platforms, and support the synthesis of complex information from multiple domains. Future research should focus on developing solutions that enable libraries in languages like Python, Julia, and R to achieve greater integration with other data processing and analytical platforms, such as Apache Spark and TensorFlow. Such integrations can make ETL processes, data processing and visualization processes more fluid. Therefore, in this study, data visualization libraries that are widely used in the literature were examined. Thus, it is anticipated that more efficient data analysis can be achieved. Python libraries such as Matplotlib, Seaborn and Plotly are quite effective in data visualization thanks to their user-friendly interfaces and wide community support. Matplotlib's customization flexibility, Seaborn's powerful analysis capabilities on statistical data and Plotly's interactive graphics make Python a versatile tool. However, there are some limitations in terms of performance on large data sets. The ggplot2 library in the R language draws attention with the aesthetic and flexible design options it offers in data visualization. ggplot2's "Grammar of Graphics" approach allows users to easily create complex visualizations. However, R's adaptation process may take longer than Python. JavaScript libraries such as D3.js, Chart.js, and Three.js offer very convenient options for web-based interactive visualizations. The powerful data binding and dynamic visualization capabilities of D3.js, the simplicity of Chart.js, and the 3D visualization features of Three.js allow JavaScript to find a wide range of use, especially in web applications. However, JavaScript libraries usually require more technical knowledge and coding. Julia's Plots.jl and Makie.jl libraries are recommended for high-performance computing and working with large data sets. Plots.jl's wide format support and Makie.jl's advanced visualization capabilities make Julia a powerful tool in data science and scientific computing. However, Julia's ecosystem is not as mature as other languages, and community support is considered more limited.

Improving the user experience of visualization tools and increasing accessibility will support their adoption by a wide range of users. This requires libraries to be equipped with more user-friendly interfaces and to make visualizations accessible to individuals at different levels. In particular, studies should be conducted on the integration of visualization and interactive graphic design that comply with accessibility standards. Integrating visualization libraries with artificial intelligence and machine learning can enhance the presentation of data analysis and model outcomes, making it easier to interpret and communicate complex insights. For example, libraries can be enriched with visualization tools that can automatically identify trends and anomalies within the data set. Such integrations can enable users to obtain data insights more quickly and accurately. Data security and privacy are important issues in visualization processes.

4. CONCLUSION

Studies on ergonomic improvements for the user-side use of visualization tools, providing interactive data analysis opportunities, and integrating different data sources are important. More work needs to be done on the performance and scalability of visualization libraries to cope with large data sets and real-time data streams. In particular, future research could focus on optimizing the implementation of interactive and dynamic visualizations for large datasets. Developing algorithms and techniques that enable faster rendering and minimize latency will be essential to improving the efficiency and responsiveness of these visualizations.

This paper provides critical comparisons of visualization libraries based on programming languages to researchers, data analysts, and data scientists working on data visualization libraries, and it is also anticipated that it will contribute to future studies.

REFERENCES

[1] I. H. Sarker, ‘Data Science and Analytics: An Overview from Data-Driven Smart Computing, Decision-Making and Applications Perspective’, *SN Computer Science*, vol. 2, no. 5, p. 377, Sep. 2021, doi: 10.1007/s42979-021-00765-8.

[2] X. Qin, Y. Luo, N. Tang, and G. Li, ‘Making Data Visualization More Efficient and Effective: A Survey’, *The VLDB Journal*, vol. 29, no. 1, pp. 93–117, Jan. 2020, doi: 10.1007/s00778-019-00588-3.

[3] J. Nayak *et al.*, ‘Intelligent Computing in IoT-Enabled Smart Cities: A Systematic Review’, in *Green Technology for Smart City and Society*, R. Sharma, M. Mishra, J. Nayak, B. Naik, and D. Pelusi, Eds., in Lecture Notes in Networks and Systems. Singapore: Springer, 2021, pp. 1–21. doi: 10.1007/978-981-15-8218-9_1.

[4] I. H. Sarker, ‘Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions’, *SN computer science*, vol. 2, no. 6, p. 420, 2021, doi: 10.1007/s42979-021-00815-1.

[5] V. Mathur, Y. Saini, V. Giri, V. Choudhary, U. Bharadwaj, and V. Kumar, ‘Weather Station Using Raspberry Pi’, in *2021 Sixth International Conference on Image Information Processing (ICIIP)*, Shimla, India: IEEE, Nov. 2021, pp. 279–283. doi: 10.1109/ICIIP53038.2021.9702687.

[6] D. Demirol, R. Das, and D. Hanbay, ‘A Key Review on Security and Privacy of Big Data: Issues, Challenges, and Future Research Directions’, *Signal, Image and Video Processing*, pp. 1–9, Sep. 2022, doi: 10.1007/s11760-022-02341-w.

[7] Z. Il-Agure and J. Dempere, ‘Review of Data Visualization Techniques in IoT Data’, in *2022 8th International Conference on Information Technology Trends (ITT)*, May 2022, pp. 167–171. doi: 10.1109/ITT56123.2022.9863948.

[8] R. B. Haber and D. A. McNabb, ‘Visualization Idioms: A Conceptual Model for Scientific Visualization Systems’, *Visualization in Scientific Computing*, vol. 74, p. 93, 1990.

[9] N. Boukhelifa, J. C. Roberts, and P. J. Rodgers, ‘A coordination model for exploratory multiview visualization’, in *Proceedings International Conference on Coordinated and Multiple Views in Exploratory Visualization - CMV 2003 -*, Jul. 2003, pp. 76–85. doi: 10.1109/CMV.2003.1215005.

[10] S. dos Santos and K. Brodlie, ‘Gaining Understanding of Multivariate and Multidimensional Data Through Visualization’, *Computers & Graphics*, vol. 28, no. 3, pp. 311–325, Jun. 2004, doi: 10.1016/j.cag.2004.03.013.

[11] M. Sülü and R. Daş, ‘Graph Visualization of Cyber Threat Intelligence Data for Analysis of Cyber Attacks’, *Balkan Journal of Electrical and Computer Engineering*, vol. 10, no. 3, pp. 300–306, Jul. 2022, doi: 10.17694/bajece.1090145.

[12] ‘A Survey of Scholarly Data Visualization | IEEE Journals & Magazine | IEEE Xplore’. Accessed: Nov. 26, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/8314667/similar#similar

[13] I. Soric, D. Dinjar, M. Stajcer, and D. Orescanin, ‘Efficient social network analysis in big data architectures’, in *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, May 2017, pp. 1397–1400. doi: 10.23919/MIPRO.2017.7973640.

[14] Z. Wang, N. Ferreira, Y. Wei, A. S. Bhaskar, and C. Scheidegger, ‘Gaussian Cubes: Real-Time Modeling for Visual Exploration of Large Multidimensional Datasets’, *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 681–690, Jan. 2017, doi: 10.1109/TVCG.2016.2598694.

[15] Y. Ye *et al.*, ‘Generative AI for visualization: State of the art and future directions’, *Visual Informatics*, vol. 8, no. 2, pp. 43–66, Jun. 2024, doi: 10.1016/j.visinf.2024.04.003.

[16] Y. Hou, M. Yang, H. Cui, L. Wang, J. Xu, and W. Zeng, ‘C2Ideas: Supporting Creative Interior Color Design Ideation with Large Language Model’, in *Proceedings of the CHI Conference on Human Factors in Computing Systems*, May 2024, pp. 1–18. doi: 10.1145/3613904.3642224.

[17] L. Shi, Z. Gao, M. Wang, and A. Liu, ‘A mainframe CPU capacity prediction system based on the mathematical modeling’, in *ICSSSM12*, Jul. 2012, pp. 537–540. doi: 10.1109/ICSSSM.2012.6252295.

[18] X. Zhang, Y. Cao, and X. Mu, ‘The Dynamic Retrieval Tree Menu Based on Dojo’, in *2011 International Conference of Information Technology, Computer Engineering and Management Sciences*, Sep. 2011, pp. 202–205. doi: 10.1109/ICM.2011.116.

[19] L. Zhuhadar, S. R. Kruk, and J. Daday, ‘Semantically enriched Massive Open Online Courses (MOOCs) platform’, *Computers in Human Behavior*, vol. 51, pp. 578–593, Oct. 2015, doi: 10.1016/j.chb.2015.02.067.

[20] Z. Chu and H. Wang, ‘An investigation of hotlinking and its countermeasures’, *Computer Communications*, vol. 34, no. 4, pp. 577–590, Apr. 2011, doi: 10.1016/j.comcom.2010.05.007.

[21] M. del P. Salas-Zárate, G. Alor-Hernández, R. Valencia-García, L. Rodríguez-Mazahua, A. Rodríguez-González, and J. L. López Cuadrado, ‘Analyzing best practices on Web development frameworks: The lift approach’, *Science of Computer Programming*, vol. 102, pp. 1–19, May 2015, doi: 10.1016/j.scico.2014.12.004.

[22] M. Bostock and J. Heer, ‘Protovis: A Graphical Toolkit for Visualization’, *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1121–1128, Nov. 2009, doi: 10.1109/TVCG.2009.174.

[23] R. D. Saxton *et al.*, ‘HILIGT, upper limit servers I-Overview’, *Astronomy and Computing*, vol. 38, p. 100531, 2022, doi: https://doi.org/10.1016/j.ascom.2021.100531.

[24] R. R. Zhang *et al.*, ‘The web-based user interface for EAST plasma control system’, *Fusion Engineering and Design*, vol. 89, no. 5, pp. 558–562, May 2014, doi: 10.1016/j.fusengdes.2014.02.070.

[25] D. Han, J. Pan, X. Zhao, and W. Chen, ‘NetV.js: A web-based library for high-efficiency visualization of large-scale graphs and networks’, *Visual Informatics*, vol. 5, no. 1, pp. 61–66, Mar. 2021, doi: 10.1016/j.visinf.2021.01.002.

[26] D. Li *et al.*, ‘ECharts: A declarative framework for rapid construction of web-based visualization’, *Visual Informatics*, vol. 2, no. 2, pp. 136–146, Jun. 2018, doi: 10.1016/j.visinf.2018.04.011.

[27] G. Buchanan and R. Tashakkori, ‘A Web-App for Analysis of Honey Bee Hive Data’, in *2019 SoutheastCon*, Apr. 2019, pp. 1–6. doi: 10.1109/SoutheastCon42311.2019.9020286.

[28] A. Satyanarayan, D. Moritz, K. Wongsuphasawat, and J. Heer, ‘Vega-Lite: A Grammar of Interactive Graphics’, *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 341–350, Jan. 2017, doi: 10.1109/TVCG.2016.2599030.

[29] R. Lu and S. Cao, ‘Highcharts-based implementation of the micro video big data visualization system’, in *2017 2nd IEEE International Conference on Computational Intelligence and Applications (ICCIA)*, Sep. 2017, pp. 1–5. doi: 10.1109/CIAPP.2017.8167049.

[30] M. Bostock, V. Ogievetsky, and J. Heer, ‘D3 Data-Driven Documents’, *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2301–2309, Dec. 2011, doi: 10.1109/TVCG.2011.185.

[31] T. B. Chandra and A. K. Dwivedi, ‘Chapter 10 - Data visualization: existing tools and techniques’, in *Advanced Data Mining Tools and Methods for Social Computing*, S. De, S. Dey, S. Bhattacharyya, and S. Bhatia, Eds., in Hybrid Computational Intelligence for Pattern Analysis. , Academic Press, 2022, pp. 177–217. doi: 10.1016/B978-0-32-385708-6.00017-5.

[32] M. Bostock, ‘D3.js - Data-Driven Documents’. Accessed: Nov. 25, 2022. [Online]. Available: https://d3js.org/

[33] https://www.ibm.com/cloud/learn/data-visualization, ‘What is Data Visualization?’ Accessed: Dec. 08, 2022. [Online]. Available: https://www.ibm.com/cloud/learn/data-visualization

[34] https://www.chartjs.org/, ‘Chart.js \textbar Open source HTML5 Charts for your website’. Accessed: Nov. 25, 2022. [Online]. Available: https://www.chartjs.org/

[35] J. Wang, W. Dou, C. Gao, and J. Wei, ‘JSTrace: Fast reproducing web application errors’, *Journal of Systems and Software*, vol. 137, pp. 448–462, Mar. 2018, doi: 10.1016/j.jss.2017.06.038.

[36] https://dojotoolkit.org/, ‘Dojo Toolkit’. Accessed: Dec. 15, 2022. [Online]. Available: https://dojotoolkit.org/

[37] A. Papantoniou, V. Loumos, M. Poulizos, and G. Rigas, ‘A Framework for Visualizing the Web of Data: Combining DBpedia and Open APIs’, in *2011 15th Panhellenic Conference on Informatics*, 2011, pp. 240–244. doi: 10.1109/PCI.2011.32.

[38] prototypejs, ‘Prototype JavaScript framework: a foundation for ambitious web applications’. Accessed: Dec. 09, 2022. [Online]. Available: http://prototypejs.org/

[39] Prototype, ‘Prototype’. Accessed: Dec. 15, 2022. [Online]. Available: https://github.com/prototypejs/prototype

[40] ‘Frameworks for RIAs Development’, in *https://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/978-1-4666-6437-1.ch002*, IGI Global, 1AD. Accessed: Nov. 27, 2024. [Online]. Available: https://www.igi-global.com/chapter/frameworks-for-rias-development/www.igi-global.com/chapter/frameworks-for-rias-development/117376

[41] EChartsJS, ‘EChartsJS \textbar Home Page’. Accessed: Dec. 29, 2022. [Online]. Available: https://www.echartsjs.com/index.html

[42] Idera Inc Company, ‘Fusioncharts’. Accessed: Nov. 25, 2022. [Online]. Available: https://www.fusioncharts.com

[43] https://c3js.org/, ‘C3.js \textbar D3-based reusable chart library’. Accessed: Nov. 30, 2022. [Online]. Available: https://c3js.org/

[44] X. Li, A. Kuroda, H. Matsuzaki, and N. Nakajima, ‘Advanced Aggregate Computation for Large Data Visualization’, in *2015 IEEE 5th Symposium on Large Data Analysis and Visualization (LDAV)*, Oct. 2015, pp. 137–138. doi: 10.1109/LDAV.2015.7348086.

[45] O. Laursen, ‘Flot: Attractive JavaScript plotting for jQuery’. Accessed: Mar. 20, 2023. [Online]. Available: https://www.flotcharts.org/

[46] vega.github.io/, ‘Vega: A Visualization Grammar’, Vega. Accessed: Jan. 29, 2023. [Online]. Available: https://vega.github.io/

[47] Z. Fang *et al.*, ‘A Java web application based MDSplus data analysis and real-time monitoring system for EAST’, *Fusion Engineering and Design*, vol. 147, p. 111215, 2019, doi: https://doi.org/10.1016/j.fusengdes.2019.05.034.

[48] N. Mukhtar, M. A. Khan, and N. Chiragh, ‘Lexicon-based approach outperforms Supervised Machine Learning approach for Urdu Sentiment Analysis in multiple domains’, *Telematics and Informatics*, vol. 35, no. 8, pp. 2173–2183, Dec. 2018, doi: 10.1016/j.tele.2018.08.003.

[49] N. Manick, A. Appigadoo, and B. Z. Cadersaib, ‘Enterprise Architecture for the Fire Rescue Services’, in *2019 7th International Conference on Information and Communication Technology (ICoICT)*, Jul. 2019, pp. 1–6. doi: 10.1109/ICoICT.2019.8835254.

[50] A. Alarcón-Paredes, V. Francisco-García, I. P. Guzmán-Guzmán, J. Cantillo-Negrete, R. E. Cuevas-Valencia, and G. A. Alonso-Silverio, 'An IoT-Based Non-Invasive Glucose Level Monitoring System Using Raspberry Pi', *Applied Sciences*, vol. 9, no. 15, Art. no. 15, Jan. 2019, doi: 10.3390/app9153046.
[51] R. R. Laher, 'Thoth: Software for data visualization & statistics', *Astronomy and Computing*, vol. 17, pp. 177–185, Oct. 2016, doi: 10.1016/j.ascom.2016.09.003.
[52] L. N. SHORNA, 'Printing and signing ECG leads in PDF files', Jul. 2012, Accessed: Feb. 09, 2025. [Online]. Available: https://www.politesi.polimi.it/handle/10589/56923
[53] Y. Zhou, Y. Yang, B. Xu, H. Leung, and X. Zhou, 'Source code size estimation approaches for object-oriented systems from UML class diagrams: A comparative study', *Information and Software Technology*, vol. 56, no. 2, pp. 220–237, Feb. 2014, doi: 10.1016/j.infsof.2013.09.003.
[54] A. N. Privalov and V. A. Smirnov, 'Detection of Fake Educational Sites Using Fuzzy String Match', in *2022 2nd International Conference on Technology Enhanced Learning in Higher Education (TELE)*, May 2022, pp. 31–36. doi: 10.1109/TELE55498.2022.9800955.
[55] S. Kelley, M. Goldberg, M. Magdon-Ismail, K. Mertsalov, W. Wallace, and M. Zaki, 'graphOnt: An ontology based library for conversion from semantic graphs to JUNG', in *2009 IEEE International Conference on Intelligence and Security Informatics*, Jun. 2009, pp. 170–172. doi: 10.1109/ISI.2009.5137292.
[56] J. Chen, Y. Li, and J. Song, 'Research on Interactive Information Visualization System in Special Academic Discussion', in *2012 Fourth International Symposium on Information Science and Engineering*, Dec. 2012, pp. 59–63. doi: 10.1109/ISISE.2012.22.
[57] D. Hägele, 'Visualizing Optimization Trajectories', masterThesis, 2019. Accessed: Feb. 09, 2025. [Online]. Available: https://elib.uni-stuttgart.de/handle/11682/10841
[58] P. Ambure, R. B. Aher, A. Gajewicz, T. Puzyn, and K. Roy, '"NanoBRIDGES" software: Open access tools to perform QSAR and nano-QSAR modeling', *Chemometrics and Intelligent Laboratory Systems*, vol. 147, pp. 1–13, Oct. 2015, doi: 10.1016/j.chemolab.2015.07.007.
[59] A. Renduchintala, F. Jahan, R. Khanna, and A. Y. Javaid, 'A comprehensive micro unmanned aerial vehicle (UAV/Drone) forensic framework', *Digital Investigation*, vol. 30, pp. 52–72, Sep. 2019, doi: 10.1016/j.diin.2019.07.002.
[60] H. Kassogué, A. Bernoussi, M. Maâtouk, and M. Amharref, 'A two scale cellular automaton for flow dynamics modeling (2CAFDYM)', *Applied Mathematical Modelling*, vol. 43, pp. 61–77, Mar. 2017, doi: 10.1016/j.apm.2016.10.034.
[61] M. Li, X. Feng, Y. Han, and X. Liu, 'Mobile augmented reality-based visualization framework for lifecycle O&M support of urban underground pipe networks', *Tunnelling and Underground Space Technology*, vol. 136, p. 105069, Jun. 2023, doi: 10.1016/j.tust.2023.105069.
[62] R. R. Laher, 'Thoth: Software for data visualization & statistics', *Astronomy and Computing*, vol. 17, pp. 177–185, Oct. 2016, doi: 10.1016/j.ascom.2016.09.003.
[63] C. Song *et al.*, 'Collection of patient-generated health data with a mobile application and transfer to hospital information system via QR codes', *Computer Methods and Programs in Biomedicine Update*, vol. 3, p. 100099, Jan. 2023, doi: 10.1016/j.cmpbup.2023.100099.
[64] R. Mayerle *et al.*, 'Development of a coastal information system for the management of Jeddah coastal waters in Saudi Arabia', *Computers & Geosciences*, vol. 89, pp. 71–78, Apr. 2016, doi: 10.1016/j.cageo.2015.12.006.
[65] T. W. Fredian and J. A. Stillerman, 'Web based electronic logbook and experiment run database viewer for Alcator C-Mod', *Fusion Engineering and Design*, vol. 81, no. 15, pp. 1963–1967, Jul. 2006, doi: 10.1016/j.fusengdes.2006.04.009.
[66] A. Sopan, A. S.-I. Noh, S. Karol, P. Rosenfeld, G. Lee, and B. Shneiderman, 'Community Health Map: A geospatial and multivariate data visualization tool for public health datasets', *Government Information Quarterly*, vol. 29, no. 2, pp. 223–234, Apr. 2012, doi: 10.1016/j.giq.2011.10.002.
[67] S. Zhang, Y. Li, T. Zhang, and Y. Peng, 'An integrated environmental decision support system for water pollution control based on TMDL – A case study in the Beiyun River watershed', *Journal of Environmental Management*, vol. 156, pp. 31–40, Jun. 2015, doi: 10.1016/j.jenvman.2015.03.021.
[68] T. J. Overbye and J. Weber, 'Smart Grid Wide-Area Transmission System Visualization', *Engineering*, vol. 1, no. 4, pp. 466–474, Dec. 2015, doi: 10.15302/J-ENG-2015098.
[69] P. Rajendrakumar, 'The Sorghum Genome: Current Status and Future Prospects', 2021. doi: 10.1007/978-981-15-8249-3_20.
[70] P. Horovčák, J. Terpák, and B. Stehlíková, 'Selected thermochemical properties of substances and their graphic web presentation', in *Proceedings of the 13th International Carpathian Control Conference (ICCC)*, May 2012, pp. 221–226. doi: 10.1109/CarpathianCC.2012.6228643.
[71] H. González-Díaz *et al.*, 'Generalized lattice graphs for 2D-visualization of biological information', *Journal of Theoretical Biology*, vol. 261, no. 1, pp. 136–147, Nov. 2009, doi: 10.1016/j.jtbi.2009.07.029.
[72] M. Mora-Cantallops, S. Sánchez-Alonso, and E. García-Barriocanal, 'A complex network analysis of the Comprehensive R Archive Network (CRAN) package ecosystem', *Journal of Systems and Software*, vol. 170, p. 110744, Dec. 2020, doi: 10.1016/j.jss.2020.110744.
[73] G. Longo-Minnolo *et al.*, 'Delineating citrus management zones using spatial interpolation and UAV-based multispectral approaches', *Computers and Electronics in Agriculture*, vol. 222, p. 109098, Jul. 2024, doi: 10.1016/j.compag.2024.109098.
[74] C. Stoiber *et al.*, 'VisAhoi: Towards a library to generate and integrate visualization onboarding using high-level visualization grammars', *Visual Informatics*, vol. 8, no. 3, pp. 1–17, 2024, doi: https://doi.org/10.1016/j.visinf.2024.06.001.
[75] A. S. Moorthy and H. J. Eberl, 'compuGUT: An *in silico* platform for simulating intestinal fermentation', *SoftwareX*, vol. 6, pp. 237–242, Jan. 2017, doi: 10.1016/j.softx.2017.06.004.
[76] Z. K. Mundargi, K. Patel, A. Patel, R. More, S. Pathrabe, and S. Patil, 'Plotplay: An Automated Data Visualization Website using Python and Plotly', in *2023 International Conference for Advancement in Technology (ICONAT)*, 2023, pp. 1–4. doi: 10.1109/ICONAT57137.2023.10079977.
[77] M. K. Nematchoua, J. A. Orosa, and M. Afaifia, 'Prediction of daily global solar radiation and air temperature using six machine learning algorithms; a case of 27 European countries', *Ecological Informatics*, vol. 69, p. 101643, Jul. 2022, doi: 10.1016/j.ecoinf.2022.101643.
[78] M. A. Perkins and J. W. Carrier, 'Protocol for programing a degree-to-careers dashboard in R using Posit and Shiny', *STAR Protocols*, vol. 5, no. 1, p. 102902, Mar. 2024, doi: 10.1016/j.xpro.2024.102902.
[79] M. Figueira, D. Conesa, and A. López-Quílez, 'A shiny R app for spatial analysis of species distribution models', *Ecological Informatics*, vol. 80, p. 102542, May 2024, doi: 10.1016/j.ecoinf.2024.102542.
[80] J. J. Castillo *et al.*, 'The Development of the Davis Food Glycopedia—A Glycan Encyclopedia of Food', *Nutrients*, vol. 14, no. 8, p. 1639, Apr. 2022, doi: 10.3390/nu14081639.
[81] A. Severinsen and Ø. Myrland, 'ShinyRBase: Near real-time energy saving models using reactive programming', *Applied Energy*, vol. 325, p. 119798, Nov. 2022, doi: 10.1016/j.apenergy.2022.119798.
[82] M. Owkes, K. Coblentz, A. Eriksson, T. Kammerzell, and P. S. Stewart, 'Biofilm.jl: A fast solver for one-dimensional biofilm chemistry and ecology', *Computer Physics Communications*, vol. 293, p. 108890, Dec. 2023, doi: 10.1016/j.cpc.2023.108890.
[83] X. Pu and M. Kay, 'How Data Analysts Use a Visualization Grammar in Practice', in *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, in CHI '23. New York, NY, USA: Association for Computing Machinery, Apr. 2023, pp. 1–22. doi: 10.1145/3544548.3580837.
[84] A. Yamashita, S. Berg, and E. Figgemeier, 'Unsupervised learning of charge-discharge cycles from various lithium-ion battery cells to visualize dataset characteristics and to interpret model performance', *Energy and AI*, vol. 17, p. 100409, Sep. 2024, doi: 10.1016/j.egyai.2024.100409.
[85] I. Morales Becerra, E. Anguiano Mondragón, I. Morales Becerra, and E. Anguiano Mondragón, 'Matías Romero Avendaño ve 19. yüzyılın sonlarında Meksika'nın Oaxaca kentindeki kahve üretimi', Accessed: Feb. 09, 2025. [Online]. Available: https://www.scielo.sa.cr/scielo.php?pid=S1409-469X2022000100117&script=sci_arttext&tlng=en
[86] M. Bitar, 'Rust and Julia for Scientific Computing', *Computing in Science & Engineering*, vol. 26, no. 1, pp. 72–76, Jan. 2024, doi: 10.1109/MCSE.2024.3369988.
[87] H. N, P. P. Teggi, M. S. Bharadwaj, and R. K, 'Detection of Brain Disorders Using Ocular Responses', in *2024 IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER)*, Oct. 2024, pp. 304–308. doi: 10.1109/DISCOVER62353.2024.10750701.
[88] M. T. Zonin and M. Sperandio, 'Methodology for Reducing the Connection Possibilities in Transmission Network Expansion

Planning', in *2023 15th Seminar on Power Electronics and Control (SEPOC)*, Oct. 2023, pp. 1–5. doi: 10.1109/SEPOC58810.2023.10322600.

[89] H. J. Hadi, Y. Cao, S. Li, N. Ahmad, and M. A. Alshara, 'FCG-MFD: Benchmark function call graph-based dataset for malware family detection', *Journal of Network and Computer Applications*, vol. 233, p. 104050, Jan. 2025, doi: 10.1016/j.jnca.2024.104050.

[90] S. K. A. Fahad and A. E. Yahya, 'Big Data Visualization: Allotting by R and Python with GUI Tools', in *2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)*, Jul. 2018, pp. 1–8. doi: 10.1109/ICSCEE.2018.8538413.

[91] M. Molero-Armenta, U. Iturrarán-Viveros, S. Aparicio, and M. G. Hernández, 'Optimized OpenCL implementation of the Elastodynamic Finite Integration Technique for viscoelastic media', *Computer Physics Communications*, vol. 185, no. 10, pp. 2683–2696, Oct. 2014, doi: 10.1016/j.cpc.2014.05.016.

[92] G. Ulian and G. Valdrè, '*SEISMIC*, a Python-based code of the quantas package to calculate the phase and group acoustic velocities in crystals', *Computers & Geosciences*, vol. 188, p. 105615, Jun. 2024, doi: 10.1016/j.cageo.2024.105615.

[93] N. Kovač, K. Ratković, H. Farahani, and P. Watson, 'A practical applications guide to machine learning regression models in psychology with Python', *Methods in Psychology*, vol. 11, p. 100156, Dec. 2024, doi: 10.1016/j.metip.2024.100156.

[94] J. Lym, G. R. Wittreich, and D. G. Vlachos, 'A Python Multiscale Thermochemistry Toolbox (pMuTT) for thermochemical and kinetic parameter estimation', *Computer Physics Communications*, vol. 247, p. 106864, Feb. 2020, doi: 10.1016/j.cpc.2019.106864.

[95] T. Koutsellis *et al.*, 'In-Cognitive: A web-based Python application for fuzzy cognitive map design, simulation, and uncertainty analysis based on the Monte Carlo method', *SoftwareX*, vol. 23, p. 101513, Jul. 2023, doi: 10.1016/j.softx.2023.101513.

[96] G. Castellazzi, N. Lo Presti, A. M. D'Altri, and S. de Miranda, 'Cloud2FEM: A finite element mesh generator based on point clouds of existing/historical structures', *SoftwareX*, vol. 18, p. 101099, Jun. 2022, doi: 10.1016/j.softx.2022.101099.

[97] A. Peñuela, C. Hutton, and F. Pianosi, 'An open-source package with interactive Jupyter Notebooks to enhance the accessibility of reservoir operations simulation and optimisation', *Environmental Modelling & Software*, vol. 145, p. 105188, Nov. 2021, doi: 10.1016/j.envsoft.2021.105188.

[98] M. Hatamzad, G. C. P. Pinerez, and J. Casselgren, 'Intelligent cost-effective winter road maintenance by predicting road surface temperature using machine learning techniques', *Knowledge-Based Systems*, vol. 247, p. 108682, Jul. 2022, doi: 10.1016/j.knosys.2022.108682.

[99] G. Singh, D. K. Misra, C. Tanwar, and P. Narang, 'Visualization and Analysis of Populations', in *2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON)*, May 2022, pp. 364–368. doi: 10.1109/COM-IT-CON54601.2022.9850886.

[100] A. Lavanya, P. Darsha, P. Akhil, J. Lloret, and N. Yogeshwar, 'A Real-Time Human Mobility Visualization of Covid-19 Spread from East Asian Countries', in *2021 Eighth International Conference on Social Network Analysis, Management and Security (SNAMS)*, Gandia, Spain: IEEE, Dec. 2021, pp. 1–8. doi: 10.1109/SNAMS53716.2021.9732103.

[101] 'Rainfall Forecasting Using Machine Learning Algorithms for Localized Events', *Computers, Materials and Continua*, vol. 71, no. 3, pp. 6333–6350, Jan. 2022, doi: 10.32604/cmc.2022.023254.

[102] A. Fathima, G. S. Devi, and M. Faizaanuddin, 'Improving distributed denial of service attack detection using supervised machine learning', *Measurement: Sensors*, vol. 30, p. 100911, Dec. 2023, doi: 10.1016/j.measen.2023.100911.

[103] S. Hooda and P. Mondal, 'Predictive modeling of plastic pyrolysis process for the evaluation of activation energy: Explainable artificial intelligence based comprehensive insights', *Journal of Environmental Management*, vol. 360, p. 121189, Jun. 2024, doi: 10.1016/j.jenvman.2024.121189.

[104] I. Sadek and B. Abdulrazak, 'A comparison of three heart rate detection algorithms over ballistocardiogram signals', *Biomedical Signal Processing and Control*, vol. 70, p. 103017, Sep. 2021, doi: 10.1016/j.bspc.2021.103017.

[105] J. Kerexeta-Sarriegi *et al.*, 'Analysing disease trajectories in a cohort of 71,849 Patients: A visual analytics and statistical approach', *International Journal of Medical Informatics*, vol. 188, p. 105466, Aug. 2024, doi: 10.1016/j.ijmedinf.2024.105466.

[106] R. De Filippo and D. Schmitz, 'Transcriptomic mapping of the 5-HT receptor landscape', *Patterns*, vol. 5, no. 10, p. 101048, Oct. 2024, doi: 10.1016/j.patter.2024.101048.

[107] S. Willems, E. Voytik, P. Skowronek, M. T. Strauss, and M. Mann, 'AlphaTims: Indexing Trapped Ion Mobility Spectrometry–TOF Data for Fast and Easy Accession and Visualization', *Molecular & Cellular Proteomics*, vol. 20, p. 100149, Jan. 2021, doi: 10.1016/j.mcpro.2021.100149.

[108] H. Martin, Y. Hong, N. Wiedemann, D. Bucher, and M. Raubal, 'Trackintel: An open-source Python library for human mobility analysis', *Computers, Environment and Urban Systems*, vol. 101, p. 101938, Apr. 2023, doi: 10.1016/j.compenvurbsys.2023.101938.

[109] X. Chen *et al.*, 'Simulation study on water yield service flow based on the InVEST-Geoda-Gephi network: A case study on Wuyi Mountains, China', *Ecological Indicators*, vol. 159, p. 111694, 2024, doi: https://doi.org/10.1016/j.ecolind.2024.111694.

[110] J. Liu, T. Tang, W. Wang, B. Xu, X. Kong, and F. Xia, 'A Survey of Scholarly Data Visualization', *IEEE Access*, vol. 6, pp. 19205–19221, 2018, doi: 10.1109/ACCESS.2018.2815030.

[111] P. Paschalis *et al.*, 'Online application for the barometric coefficient calculation of the NMDB stations', *New Astronomy*, vol. 19, pp. 10–18, 2013, doi: https://doi.org/10.1016/j.newast.2012.08.003.

[112] https://plotly.com/, 'Plotly: Low-Code Data App Development'. Accessed: Nov. 29, 2022. [Online]. Available: https://plotly.com/

[113] M. L. Waskom, 'seaborn: statistical data visualization'. Accessed: Feb. 02, 2023. [Online]. Available: https://joss.theoj.org/papers/10.21105/joss.03021

[114] J. VanderPlas *et al.*, 'Altair: Interactive Statistical Visualizations for Python', *Journal of Open Source Software*, vol. 3, no. 32, p. 1057, Dec. 2018, doi: 10.21105/joss.01057.

[115] F. M. / Kozea, 'Pygal: A Python Svg Graph Plotting Library'. Accessed: Nov. 29, 2022. [Online]. Available: https://www.pygal.org/

[116] Bokeh, 'Bokeh Documentation'. Accessed: Nov. 29, 2022. [Online]. Available: https://docs.bokeh.org/en/latest/index.html

[117] M. J. MacLeod *et al.*, 'Invited Review: A Position on the Global Livestock Environmental Assessment Model (gleam)', *Animal*, vol. 12, no. 2, pp. 383–397, Jan. 2018, doi: 10.1017/S1751731117001847.

[118] https://matplotlib.org, 'Matplotlib 3.6.3 documentation'. Accessed: Feb. 04, 2023. [Online]. Available: https://matplotlib.org/stable/index.html

[119] J. D. Hunter, 'Matplotlib: A 2D Graphics Environment', *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007, doi: 10.1109/MCSE.2007.55.

[120] G. Singh, D. K. Misra, C. Tanwar, and P. Narang, 'Visualization and Analysis of Populations', in *2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON)*, May 2022, pp. 364–368. doi: 10.1109/COM-IT-CON54601.2022.9850886.

[121] Folium, 'Folium 0.14.0 documentation'. Accessed: Feb. 05, 2023. [Online]. Available: https://python-visualization.github.io/folium/

[122] M. S. Kahil, A. Bouramoul, and M. Derdour, 'Multi Criteria-Based Community Detection and Visualization in Large-Scale Networks Using Label Propagation Algorithm', in *2021 International Conference on Recent Advances in Mathematics and Informatics (ICRAMI)*, Sep. 2021, pp. 1–6. doi: 10.1109/ICRAMI52622.2021.9585964.

[123] MIT, 'https://github.com/plotly/plotly.py'. Accessed: Feb. 05, 2023. [Online]. Available: https://github.com/plotly/plotly.py

[124] J. Heer, S. K. Card, and J. A. Landay, 'prefuse: a toolkit for interactive information visualization', in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, in CHI '05. New York, NY, USA: Association for Computing Machinery, Apr. 2005, pp. 421–430. doi: 10.1145/1054972.1055031.

[125] Blokt, 'Flare \textbar Data Visualization for the Web', Blokt - Privacy, Tech, Bitcoin, Blockchain & Cryptocurrency. Accessed: Dec. 02, 2022. [Online]. Available: https://blokt.com/tool/prefuse-flare

[126] X. Zhang, C. Zhao, and X. Wang, 'A Survey on Knowledge Representation in Materials Science and Engineering: An Ontological Perspective', *Computers in Industry*, vol. 73, pp. 8–22, Oct. 2015, doi: 10.1016/j.compind.2015.07.005.

[127] S. Purushe, S. K. Anbalagan, and G. Grinstein, 'Development of an Interactive Ramachandran Plot in Weave', in *2011 15th International Conference on Information Visualisation*, Jul. 2011, pp. 232–236. doi: 10.1109/IV.2011.109.

[128] C. Dempsey, 'Weave - Open Source Data Visualization', GIS Lounge. Accessed: Dec. 23, 2022. [Online]. Available: https://www.gislounge.com/weave-open-source-data-visualization/

[129] A. Baumann, S. Shams, M. Ross, W. Mass, and G. Grinstein, 'Enhancing STEM classes using weave: A collaborative web-based visualization environment', in *2011 Integrated STEM Education Conference (ISEC)*, Apr. 2011, pp. 2A-1-2A – 4. doi: 10.1109/ISECon.2011.6229637.

[130] R. Agrawal, A. Kadadi, X. Dai, and F. Andres, 'Challenges and Opportunities with Big Data Visualization', in *Proceedings of the 7th International Conference on Management of computational and collective intElligence in Digital EcoSystems*, Caraguatatuba Brazil: ACM, Oct. 2015, pp. 169–173. doi: 10.1145/2857218.2857256.

[131] L. Wilkinson, *The Grammar of Graphics*. Springer Science & Business Media, 2013.

[132] N. Kruchten, A. M. McNutt, and M. J. McGuffin, 'Metrics-Based Evaluation and Comparison of Visualization Notations', *IEEE Transactions on Visualization and Computer Graphics*, vol. 30, no. 1, pp. 425–435, 2024, doi: 10.1109/TVCG.2023.3326907.

[133] S. G, A. Swaminathan, J. B. J, S. R, and L. Nelson, 'IPL Data Analysis and Visualization for Team Selection and Profit Strategy', in *2023 7th International Conference on Computing Methodologies and Communication (ICCMC)*, 2023, pp. 592–598. doi: 10.1109/ICCMC56507.2023.10083736.

BIOGRAPHIES

Faruk Aksoy works as a lecturer at Dicle University Ergani Vocational School Computer Technologies Department. He graduated from Selçuk University Computer Systems Teaching Department and Batman University Computer Engineering Department. He has a master's degree in Electronics-Computer Education from Fırat University. He continues his doctoral studies at Fırat University Software Engineering Department. His current research interests include data science, data visualization, and big data analysis.

Mehmet Özdem received the B.S. degree in Electrical and Electronics Engineering from Başkent University, Ankara, Turkey, and received the M.S degree in Information Systems from the Middle East Technical University Informatics Institute Ankara, Turkey. Currently, he is a Ph.D. student at Gazi University Faculty of Technology, and his research interests are applying open-source SDN/NFV Technologies to the services used in network infrastructure and observing effects on system performance. "Performance of distribution Networks under VBR video traffic" paper was presented at IEEE SoutheastCon 2007, Richmond, VA, USA. Since 2003, he has been with the Turk Telekom, Ankara, Turkey where he is currently network architecture and design group manager. During this time, he is involved in huge projects like NFV/SDN, IP/MPLS network design, IPTV, NGN Transformation, ATM optimization, DPI, L2/L3 VPN service developments, managing end to end architectural design for both mobile/fixed networks. He is closely working with international organizations (ITU, ETSI, ONF) about network architecture, evolution and convergence of future networks. He works as a Vice Chair of ITU-T SG12 (Performance, QoS and QoE) group. He also has CCIE R&S, PMP and ITIL certifications

Resul Daş is a full professor in the Department of Software Engineering at the Faculty of Technology, Firat University. He received his B.Sc. and M.Sc. degrees in Computer Science from Firat University in 1999 and 2002, respectively, and completed his Ph.D. in Electrical and Electronics Engineering in 2008. Between 2000 and 2011, he served as a lecturer in the Department of Informatics and concurrently worked as a network and system administrator at the University's IT Center. Since 2002, he has been an instructor in the Cisco Networking Academy Program, delivering CCNA and CCNP courses. From September 2017 to June 2018, he conducted research as a visiting professor at the University of Alberta, Edmonton, Canada, under the TUBITAK-BIDEB 2219 Postdoctoral Research Fellowship program. He also held the position of Head of the Department of Software Engineering from March 2020 to April 2023.

Professor Das has served in editorial roles for several prestigious academic journals. He was an Associate Editor for IEEE Access and the Turkish Journal of Electrical Engineering and Computer Science. Currently, he is an Associate Editor for several Elsevier journals, including Internet of Things, Alexandria Engineering Journal, and Telematics and Informatics Reports, as well as the IEEE Open Journal of the Communications Society (OJ-COMS) and the International Journal of Grid and Utility Computing (Inderscience).

Globally recognized for his academic contributions, Prof. Das has been consistently listed among the top 2\% of the "World's Most Influential Scientists," compiled by Stanford University researchers, for five consecutive years (2019-2024). His research interests include computer networks, cybersecurity, IoT and systems engineering, data science and visualization, and software quality assurance and testing.