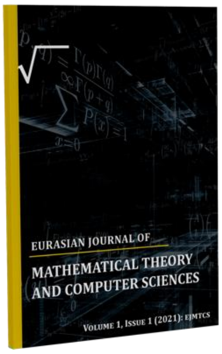
EURASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES
Innovative Academy Research Support Center
IF = 7.906 www.in-academy.uz



# DATA VISUALIZATION IN PYTHON

**Shoyqulov Shodmonkul Qudratovich**
Acting Associate Professor, department of Applied Mathematics,
Karshi State university, Karshi, Republic of Uzbekistan

https://doi.org/10.5281/zenodo.13892777

ARTICLE INFO
Received: 01st October 2024
Accepted: 05th October 2024
Online: 05th October 2024
KEYWORDS
*Data visualization, Python, Matplotlib, Seaborn, Plotly, Bokeh, data analysis, charts, graphs, interactive visualization, Python libraries, machine learning.*

ABSTRACT

*This article discusses the main approaches and tools for data visualization in Python, a popular programming language in the field of data analysis and science. Data visualization is an important step in analytical processes, allowing you to clearly present information, identify trends and relationships. Python offers a wide range of libraries for creating graphs, charts, and interactive visualizations, including Matplotlib, Seaborn, Plotly, and Bokeh. The article also provides examples of practical application of data visualization, from time series analysis to interactive and animated graphs. The advantages and limitations of Python for data visualization are considered, as well as prospects for its further development in this direction.*

## INTRODUCTION

Data visualization plays a key role in analyzing and interpreting information, allowing researchers and analysts to present their findings more effectively and find hidden patterns. In today's world, where data volumes are growing at an incredible rate, high-quality visualization is becoming an essential tool for analysis and decision making. It not only simplifies the understanding of complex data, but also provides the ability to analyze it in real time. Python is one of the most popular programming languages for data visualization, due to its simplicity and a rich set of libraries that cover a wide range of tasks: from plotting simple graphs to creating interactive and animated visualizations. Python tools such as Matplotlib, Seaborn, Plotly, and Bokeh provide flexible capabilities for working with graphics and help create visualizations for scientific research, business, and education.

The purpose of this article is to explore the main Python libraries for data visualization, consider their capabilities, advantages, and limitations, and show examples of practical application.

## RESULTS and DISCUSSIONS

Data visualization is a key stage of analysis that allows researchers and analysts not only to present data in an easy-to-understand form, but also to identify hidden patterns and anomalies. Using visualization helps to understand trends, relationships between variables, and potential problems in a data set faster and more deeply. In the context of working with big data, visualization becomes especially relevant, as it allows to significantly reduce the time for analysis and facilitates the interpretation of complex multidimensional data.

Data visualization in Python is the process of presenting data in a graphical format that allows you to better understand data structures, identify patterns, trends, and dependencies. Python offers many libraries for data visualization, each of which has its own advantages depending on the goals and objectives [2].

Python has long established itself as one of the leading programming languages for data analysis and machine learning. Its popularity is due not only to its convenient syntax and huge community, but also to a powerful set of libraries for working with data and their visualization. This article covered libraries such as Matplotlib, Seaborn, Plotly, Bokeh, and Pandas, each of which provides unique capabilities for creating graphs and charts[4].

Matplotlib remains one of the most popular libraries for plotting in Python. It is versatile and supports the creation of almost any type of graph: from line to three-dimensional visualizations. Although Matplotlib has a somewhat outdated syntax compared to modern libraries, it remains the basis of most visualization tools in Python.

Seaborn simplifies the process of creating more complex and statistically significant visualizations. With its help, you can easily build distribution graphs, heat maps, and scatter plots. While Matplotlib requires more work to style graphs, Seaborn automatically creates visually appealing and informative graphs.

Plotly and Bokeh significantly expand the possibilities of data visualization, offering support for interactive graphs that can be used in both web applications and reports. These libraries allow researchers to create complex and interactive visualizations with a minimum amount of code. In particular, Plotly supports the creation of 3D plots, which is especially useful for working with data in areas where it is important to take into account spatial parameters (for example, in geographic studies or bioinformatics).

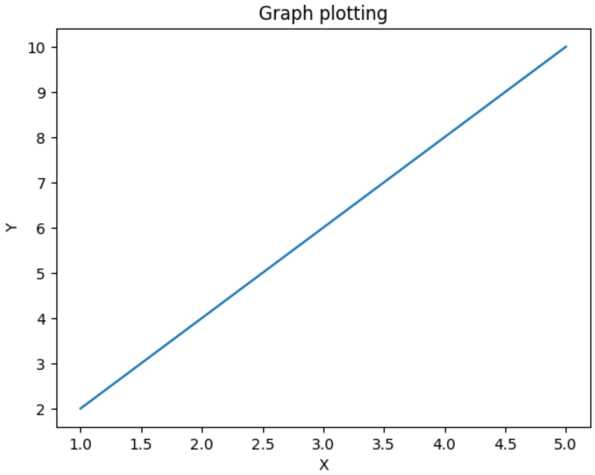
Pandas Visualization offers basic visualization tools built right into the DataFrame. This allows users to quickly build simple graphs without the need for external libraries. Although Pandas Visualization does not provide advanced features compared to Matplotlib or Seaborn, it is great for quick data analysis and creating preliminary visualizations.

Now let's look at the main libraries and their use cases.

1. Matplotlib. One of the most popular libraries for creating static, animated and interactive graphs. It is used to build simple graphs: line graphs, histograms, pie charts, etc.

Example #1.

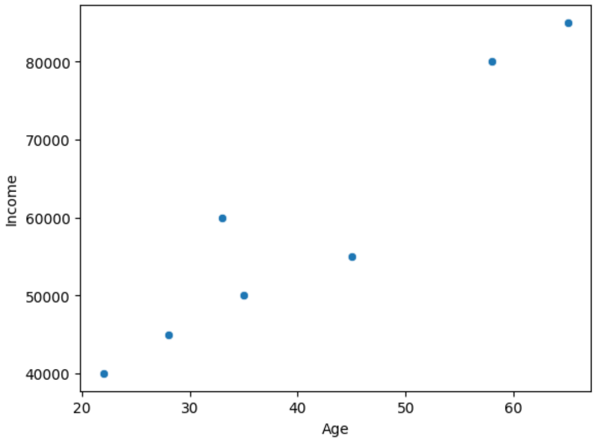
```
import matplotlib.pyplot as plt
x = [1, 2, 3, 4, 5]
y = [2, 4, 6, 8, 10]
plt.plot(x, y)
plt.xlabel('X')
plt.ylabel('Y')
plt.title('Graph plotting')
plt.show()
```



2. Seaborn. An extension to Matplotlib that provides a more convenient interface and styles for creating beautiful and informative visualizations. Seaborn is often used to create heatmaps, distributions, and multivariate plots.

Example #2.

```
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd
data = pd.DataFrame({
'Age': [22, 35, 58, 45, 28, 65, 33],
'Income': [40000, 50000, 80000, 55000, 45000, 85000, 60000]
})
sns.scatterplot(x='Age', y='Income', data=data)
plt.show()
```

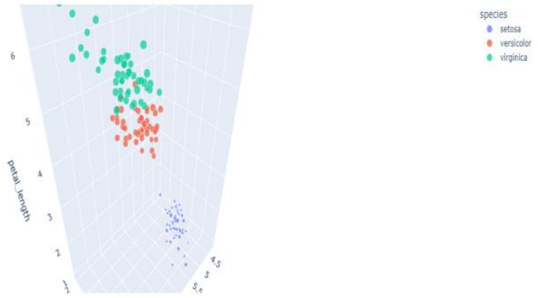


3. Plotly. A library for creating interactive plots that supports web visualization and can be used both in the browser and in Jupyter Notebook. It is used to create interactive plots and visualizations of 3D data.

Example #3.

```
import plotly.express as px
df = px.data.iris()
fig = px.scatter_3d(df, x='sepal_length', y='sepal_width', z='petal_length', color='species', size='petal_width')
```

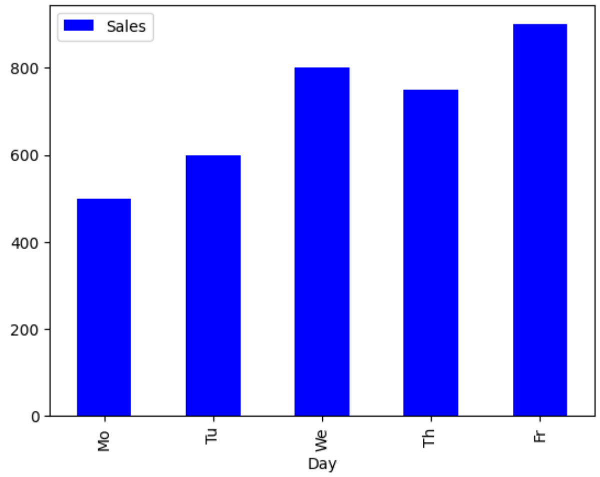
```
fig.show()
```



4. Pandas Visualization. Built-in data visualization tools in the Pandas library. This library is used to quickly build simple graphs directly from a DataFrame.

Example #4.

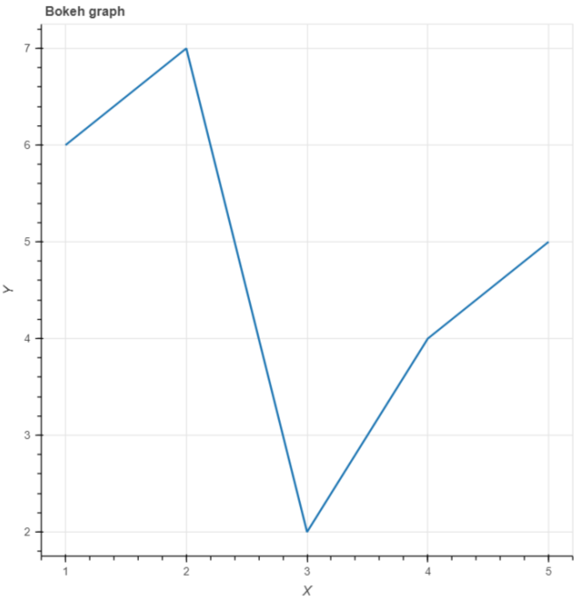
```
import pandas as pd
import matplotlib.pyplot as plt
data = {'Day': ['Mo', 'Tu', 'We', 'Th', 'Fr'],
'Sales': [500, 600, 800, 750, 900]}
df = pd.DataFrame(data)
df.plot(kind='bar', x='Day', y='Sales', color='blue')
plt.show()
```



5. Bokeh. Library for creating interactive graphs with the ability to change data in real time

Example #5.

```
from bokeh.plotting import figure, show
from bokeh.io import output_notebook
output_notebook()
p = figure(title=" Bokeh graph", x_axis_label='X', y_axis_label='Y')
p.line([1, 2, 3, 4, 5], [6, 7, 2, 4, 5], line_width=2)
show(p)
```



Data visualization is a powerful tool for presenting and analyzing information, especially when it comes to large amounts of data. Different types of graphs and charts help identify trends, patterns, and anomalies. Depending on the task and the nature of the data, different types of visualizations are used. Let's look at the main types of visualizations used in data analytics and supported by Python libraries such as Matplotlib, Seaborn, Plotly, and others:

- Line charts. Used to visualize time series or to track changes in data over time.
- Histograms. Used to display the distribution of data.
- Scatter plots. Help visualize relationships between two or more variables.
- Pie charts. Show proportions or percentage distribution of categories.
- Heatmaps. Used to display data intensity in the form of colored matrices.
- Box plots. Used to analyze the spread and outliers in a data set.

Each type of data visualization has its own unique purpose and application. Choosing the right type of plot depends on the purpose of the analysis and the nature of the data. Line plots are ideal for tracking changes over time, scatter plots are great for analyzing correlations, and box plots and histograms are great for revealing the distribution of data and its features. Python, with its variety of libraries, provides researchers with all the necessary tools for effective and clear data visualization in any application area[12].

Despite the wide range of capabilities of Python for data visualization, there are still certain challenges associated with the use of these tools. One of the main problems is the difficulty of creating complex plots for multivariate data. Even with libraries such as Plotly or Seaborn, researchers sometimes face complex problems in choosing appropriate visualization methods for large data sets with many variables.

Another important aspect is performance. When working with large amounts of data, especially in real time, the performance of visualizations can become a bottleneck. For example, interactive graphs created with Plotly or Bokeh may be slow if the data set is large or contains many complex visual elements.

In addition, data interpretation remains an important challenge. Although visualization greatly simplifies the perception of information, the wrong choice of graph type or styling can lead to a distorted interpretation of the data. For example, incorrectly displayed axes, scales, or colors can create false impressions and lead to incorrect conclusions. This highlights the importance of choosing the right visualization methods depending on the research objectives.

Modern data visualization tools such as Plotly and Bokeh allow you to create interactive graphs, which is becoming increasingly important in data analytics. Interactive elements such as filters, pop-ups with information, and dynamic graph changes allow users to explore and interact with the data. This is especially useful when creating reports for executives, who can change visualization parameters in real time to obtain the desired information[1].

The future of data visualization in Python will likely be associated with the development of tools for working with big data and the implementation of artificial intelligence. The integration of libraries for machine learning and visualization will allow you to create more intelligent graphs that will automatically suggest better ways to present data based on their analysis. This, in turn, will increase the efficiency of analysts and speed up the decision-making process.

In practice, data visualization in Python is already actively used in a variety of industries. For example, in finance, visualization allows you to explore the dynamics of stock prices, conduct risk analysis, and evaluate investment strategies. In science, visualization helps to explore statistical patterns and identify correlations between variables. In business, data visualization plays a key role in making decisions based on KPIs (key performance indicators), which helps improve operational processes and increase profits.

Interactive visualizations find their application in presentations and web applications, where end users can analyze the data themselves, change visualization parameters, and receive updated results in real time. This approach is actively used in marketing, where data visualization helps track customer behavior and analyze the effectiveness of advertising campaigns[2].

Research and development of data visualization tools will continue to be an important area, as the volumes of data that researchers and companies work with grow every year. The introduction of new visualization methods, such as multidimensional interactive graphs, and improving the performance of existing tools are key areas of development in this area.

In the future, more and more advanced data visualization libraries will likely appear, allowing the creation of more complex and beautiful graphs with minimal effort from users. An important aspect will also be the integration of big data tools, which will allow researchers to effectively visualize even the most voluminous and complex data sets[4].

## CONCLUSIONS

Python provides powerful data visualization tools that allow analysts and researchers to effectively present data in a convenient and understandable way. Using libraries such as Matplotlib, Seaborn, Plotly, Bokeh, and Pandas Visualization, it is possible to create both simple and complex graphs for deep data analysis and interaction.

Data visualization is a vital analysis tool that helps analysts, researchers, and business users make informed decisions based on the data obtained. Using the Python programming language for data visualization significantly expands the capabilities of users thanks to powerful libraries and frameworks such as Matplotlib, Seaborn, Plotly, Bokeh, and Pandas. These tools not only simplify the process of creating visualizations, but also allow for deeper analysis of data due to flexible and interactive graphical capabilities[6].

Each of the libraries considered has its own features and applications. Matplotlib offers wide functionality for creating static graphs and is the basis for most visualization libraries. Seaborn extends Matplotlib to provide more intuitive and aesthetically pleasing visualization tools, especially useful for working with statistical data. Plotly and Bokeh support the creation of interactive plots, allowing users to explore data in more detail and interact with visualizations in real time. Pandas provides built-in capabilities for quickly and easily plotting directly from a DataFrame, speeding up data analysis.

Using these tools provides a wide range of visualization options for a variety of data types, including time series, categorical and numeric data, geographic information, and multidimensional datasets. Python’s ability to combine powerful data analysis and visualization tools makes it a versatile solution for both simple and complex analytical tasks. As a result, data visualization is becoming an integral part of the data analysis process in industries ranging from scientific research to business intelligence and finance. In the future, we can expect data visualization tools to continue to evolve, offering more intuitive interfaces, better capabilities for working with big data, and deeper integration with artificial intelligence and machine learning. The Python platform, with its flexibility and openness, plays a leading role in this process and continues to be one of the best tools for data analysts around the world. Thus, using Python and its libraries for data visualization not only simplifies the analysis process, but also significantly improves the quality and accuracy of the conclusions that can be drawn from the data. The ability to visually represent information allows for more informed decisions and provides a better understanding of the structure and behavior of the data.

## References:

1. Shoyqulov Sh. Q.METHODS FOR PLOTTING FUNCTION GRAPHS IN COMPUTERS USING BACKEND AND FRONTEND INTERNET TECHNOLOGIES. European Scholar Journal (ESJ). Vol. 2 No. 6, June 2021, ISSN: 2660-5562. P.161-165, https://scholarzest.com/index.php/esj/article/view/964/826

2. Shoyqulov Sh. Q., Bozorov A. A. Methods for graphing functions in computers using Web technologies. Journal of Information Computational Science. Journal Vol. 1 Issue 1, JUNE 2021. Urgench., https://www.sciencepublish.org/index.php/ics/article/view/79

3. Shoyqulov Sh. Q. Wonderful multimedia - applying in areas outside of teaching. “Innovations in technology and science education” scientific journal, Volume #2, issue#7, Publication: february 2023, p. 700-708, SJIF-5.305, ISSN 2181-371X, https://humoscience.com/index.php/itse/index

4. Sh.Q. Shoyqulov. (2021). Methods for plotting function graphs in computers using backend and frontend internet technologies. European Scholar Journal, 2(6), 161-165. Retrieved from https://scholarzest.com/index.php/esj/article/view/964

5. Sh.Q. Shoyqulov, A. M. Shukurov. Propagation of Non-Stationary Waves Of Transverse Displacement from a Spherical Cavity in an Elastic Half-Space.

6. International Journal of Advanced Research in Science, Engineering and Technology. 13291-13299. Vol. 7, Issue 4 , April 2020. http://www.ijarset.com/upload/2020/april/13-shshovqulov-02-1.pdf

7. Shoyqulov Sh. Q., Bozorov A. A. Methods for plotting function graphs in computers using modern software and programming languages. ACADEMICIA: An International Multidisciplinary Research Journal. 321-329. 2021, Volume : 11, Issue : 6. ISSN : 2249-7137. DOI : 10.5958/2249-7137.2021.01619.0. Online published on 22 July, 2021.

8. Bozorov Abdumannon, & Shoyqulov Shodmonkul Qudratovich. (2022). MULTIMEDIA SURVEILLANCE CAMERAS AND THEIR FEATURES IN USING. Open Access Repository, 9(10), 29–34. https://doi.org/10.17605/OSF.IO/4EV75

9. Bozorov Abdumannon, Nodirbek Abdulkhayev, Shoyqulov Shodmonkul Qudratovich. (2022). MODERN TECHNOLOGIES OF VIRTUAL REALITY– A NEW MULTIMEDIA OPPORTUNITIES. EURASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES, 2(11), 85–90. https://doi.org/10.5281/zenodo.7251370

10. Qudratovich, S. S. (2022). The Role and Possibilities of Multimedia Technologies in Education. International Journal of Discoveries and Innovations in Applied Sciences, 2(3), 72–78. Retrieved from http://openaccessjournals.eu/index.php/ijdias/article/view/1148

11. Qudratovich, S. S. (2022). Technical and Software Capabilities of a Computer for Working with Multimedia Resources. International Journal of Discoveries and Innovations in Applied Sciences, 2(3), 64–71. Retrieved from http://openaccessjournals.eu/index.php/ijdias/article/view/1147

12. Sh.Q. Shoyqulov. (2022). The text is of the main components of multimedia technologies. Academicia Globe: Inderscience Research, 3(04), 573–580. https://doi.org/10.17605/OSF.IO/VBY8Z

13. Sh.Q. Shoyqulov. EditorJournals and Conferences. (2022, May 3). The graphics- is of the main components of multimedia technologies. https://doi.org/10.17605/OSF.IO/2KAM8

14. https://wos.academiascience.org/index.php/wos/article/view/1427

15. Shoyqulov, S.Q. and Bozorov, A.A. 2022. The Audio- Is of the Main Components of Multimedia Technologies. International Journal on Integrated Education. 5, 5 (May 2022), 263-268.

16. Shoykulova Dilorom Kudratovna, & Sh.Q. Shoyqulov. (2022). PHP is one of the main tools for creating a Web page in computer science lessons. Texas Journal of Engineering and Technology, 9, 83–87. Retrieved from https://zienjournals.com/index.php/tjet/article/view/2000