Онлайн-университет «УРБАН»

URBAN UNIVERSITY

ООО «Эдекс»

**ДИПЛОМНАЯ РАБОТА**

**Сравнение различных библиотек для визуализации данных:**

**Matplotlib, Seaborn и Plotly**

### направление: Python-developer

### выполнил: студент группы Python-разработчик

### Кузнецов Александр Владимирович

### 2024

### год

Оглавление

[**Введение** 4](#_Toc181182460)

[Анализ современных подходов и инструментов в визуализации данных в Python: 4](#_Toc181182461)

[**Обоснование выбора темы:** 5](#_Toc181182462)

[**1.** **Актуальность:** 5](#_Toc181182463)

[**2.** **Потребность рынка:** 5](#_Toc181182464)

[**3.** **Доступность источников:** 6](#_Toc181182465)

[**4.** **Потенциальные перспективы:** 6](#_Toc181182466)

[**Цели и задачи исследования:** 6](#_Toc181182467)

[**Теоретический обзор и анализ литературы:** 6](#_Toc181182468)

[1. **Matplotlib** 6](#_Toc181182469)

[Вывод по обзору официального сайта Matplotlib: 10](#_Toc181182470)

[2. **Seaborn** 11](#_Toc181182471)

[Вывод по обзору официального сайта Seaborn: 17](#_Toc181182472)

[3. **Plotly** 17](#_Toc181182473)

[Вывод по обзору официального сайта Plotly: 23](#_Toc181182474)

[**Практическая часть исследования:** 24](#_Toc181182475)

[Визуализация линейных графиков 25](#_Toc181182476)

[Результат построения линейного графика Matplotlib: 25](#_Toc181182477)

[Результат построения линейного графика Seaborn: 25](#_Toc181182478)

[Результат построения линейного графика Plotly: 27](#_Toc181182479)

[Визуализация столбчатых диаграмм 27](#_Toc181182480)

[Результат построения столбчатой диаграммы Matplotlib: 28](#_Toc181182481)

[Результат построения столбчатой диаграммы Seaborn: 29](#_Toc181182482)

[Результат построения столбчатой диаграммы Plotly: 30](#_Toc181182483)

[Визуализация круговых диаграмм 30](#_Toc181182484)

[Результат построения Круговой диаграммы Matplotlib: 31](#_Toc181182485)

[Результат построения Круговой диаграммы Seaborn: 32](#_Toc181182486)

[Результат построения Кольцевой диаграммы Plotly: 33](#_Toc181182487)

[Визуализация 3D-моделей 33](#_Toc181182488)

[Результат построения 3D модели Matplotlib: 34](#_Toc181182489)

[Результат построения 3D модели Seaborn: 35](#_Toc181182490)

[Результат построения 3D модели Plotly: 36](#_Toc181182491)

[Сводная таблица по реализованному объему кода: 36](#_Toc181182492)

[Сводная таблица по временным показателям построения графиков: 37](#_Toc181182493)

[Сводная таблица оценки по практической части исследования: 37](#_Toc181182494)

[**Вывод по библиотеке Matplotlib**: 37](#_Toc181182495)

[**Вывод по библиотеке Seaborn**: 38](#_Toc181182496)

[**Вывод по библиотеке Plotly**: 38](#_Toc181182497)

[**Общий вывод:** 38](#_Toc181182498)

# **Введение**

В современном мире нас окружает такое количество данных, что даже если не брать конкретно сферу IT, то их невозможно представить без приближенного и понятного визуализированного вида. Визуализация процессов в большинстве случаев доносит информацию более эффективно, чем текст или аудио. В отличие от этих подходов, где возможны разночтения, визуальную презентацию все понимают примерно одинаково, и это очень удобно.

### Анализ современных подходов и инструментов в визуализации данных в Python:

Наиболее популярными инструментами в разработке визуализации данных на языке Python являются:

**Taipy**, - Библиотека с ориентированным на пользователя подходом к визуализации данных. Делает динамические и интерактивные визуализации доступными для тех, кто имеет минимальный опыт веб-разработки.

**Plotly**, - Библиотека для создания интерактивных графиков издательского качества в режиме онлайн. Позволяет создавать широкий спектр графиков, включая сложные 3D-визуализации, географические карты и интерактивные временные ряды.

**Matplotlib**, - Библиотека предлагает контроль над каждым элементом графика, что делает её идеальной для создания готовых к публикации графиков и рисунков.

**Seaborn**, - Библиотека, построенная на основе Matplotlib, расширяет её функциональность, облегчая создание сложных визуализаций. Особенно хорошо подходит для статистического анализа.

**Bokeh**, - С помощью этой библиотеки можно создавать интерактивные графики и информационные панели непосредственно в браузере.

**Gradio**, - Библиотека позволяет создавать интерактивные пользовательские интерфейсы для скриптов Python. Подходит для сценариев, в которых важна визуализация результатов моделирования, сравнение различных моделей или демонстрация возможностей модели в режиме реального времени.

**Streamlit**, - Библиотека позволяет пользователям создавать высоко интерактивные приложения для исследования и визуализации данных. Упор делается на скорость, простоту и возможность быстрой итерации.

Ниже приведены некоторые недостатки современных подходов к визуализации данных:

**- Отнимает много времени**. Создание визуализаций может быть трудоёмким процессом, особенно при работе с большими и сложными наборами данных.

**- Может вводить в заблуждение**. Если выполнена неправильно, визуализация может создавать впечатление закономерностей или тенденций, которых, возможно, не существует.

**- Может быть трудно интерпретировать**. Некоторые типы визуализаций, например, те, которые включают 3D или интерактивные элементы, могут быть трудны для интерпретации и понимания.

**- Может подходить не для всех типов данных**. Определённые типы данных, такие как текстовые или аудиоданные, могут плохо поддаваться визуализации.

**- Могут быть доступны не всем пользователям**. Некоторые пользователи могут иметь нарушения зрения или другие нарушения, которые затрудняют или делают невозможным интерпретацию визуализаций.

# **Обоснование выбора темы:**

## **Актуальность:**

В условиях динамичных изменений современного мироустройства как в политической, экономической, продовольственной сферах так и в логистике, включением в эти процессы все новых и новых государств происходит активный рост документального потока. Все больше возрастает потребность в преобразовании информации, ее сжатии и наглядном представлении.

## **Потребность рынка:**

Потребность рынка в специалистах визуализации данных высока и продолжает расти, так как объём данных в мире постоянно увеличивается. По прогнозам, к 2025 году он вырастет до 180 зеттабайт. Наиболее востребованы специалисты в розничной торговле, электроэнергетике и металлургической промышленности, так как именно в этих отраслях чаще внедряют BI-инструменты (Business Intelligence**)**. Также системы по сбору и обработке массивов данных используют в банках, логистике, нефтегазовой и транспортной сфере, пищевой промышленности.

## **Доступность источников:**

Официальная документация абсолютного большинства библиотек, предназначенных для визуализации данных размещена на официальных сайтах в свободном доступе.

## **Потенциальные перспективы:**

Использование различных инструментов для визуализации данных позволяет развить и наработать опыт создания интересных и разносторонних проектов. Кроме того, такие навыки востребованы на рынке труда, что открывает перспективы для карьерного роста и развития.

Таким образом, выбор темы «**Сравнение различных библиотек для визуализации данных: Matplotlib, Seaborn и Plotly**» обусловлен ее актуальностью, потребностями рынка, доступностью источников и потенциальными перспективами, что делает эту тему подходящей для проведения дипломной работы.

# **Цели и задачи исследования:**

1. Провести теоретический обзор и анализ литературы
   1. Определить следующие критерии оценки, - легкость поиска необходимой информации (API), наличие обучающих материалов, структура подачи материала, наличие примеров кода, наличие релизов.
   2. Установить оценку по 5-бальной шкале
   3. Оформить выводы по каждой библиотеке
2. Провести практическое исследование: Создать набор визуализаций с использованием Matplotlib, Seaborn и Plotly, сравнить их функциональность и удобство использования.
   1. Определить критериями оценки, - объем кода (без учета стиля написания и импортов), скорость построения графиков, дополнительные возможности.
   2. Установить оценку по 5-бальной шкале
   3. Оформить выводы по каждой библиотеке
3. Провести итоговую оценку исследования (оформить заключением)

# **Теоретический обзор и анализ литературы:**

## **Matplotlib**

Официальный сайт [matplotlib.org](https://matplotlib.org/) cодержит всю необходимую информацию о продукте, начиная с истории создания в 2008 году, полной документации, обучающие материалы, отсылки на свои страницы в **GitHub** и **Twitter.** (рис.1)

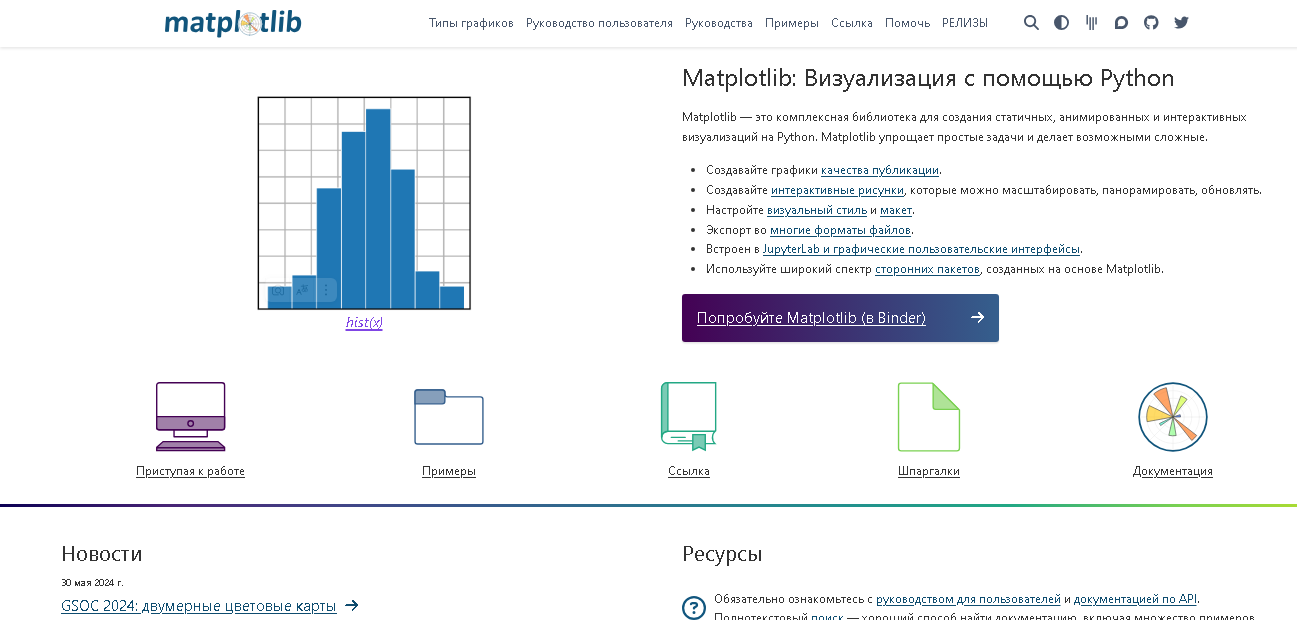


рис.1

Содержит разделы, - «Plot Types» (Типы графиков), «User guide» (Руководство пользователя), «Tutorials» (Руководства), «Exemples» (Примеры), «Reference» (Ссылка), «Contribute» (помощь), «Reliases» (релизы).

Раздел «User guide» (руководство пользователя) содержит всеобъемлющую информацию об использовании инструментов библиотеки, начиная с установки библиотеки (рис.2)

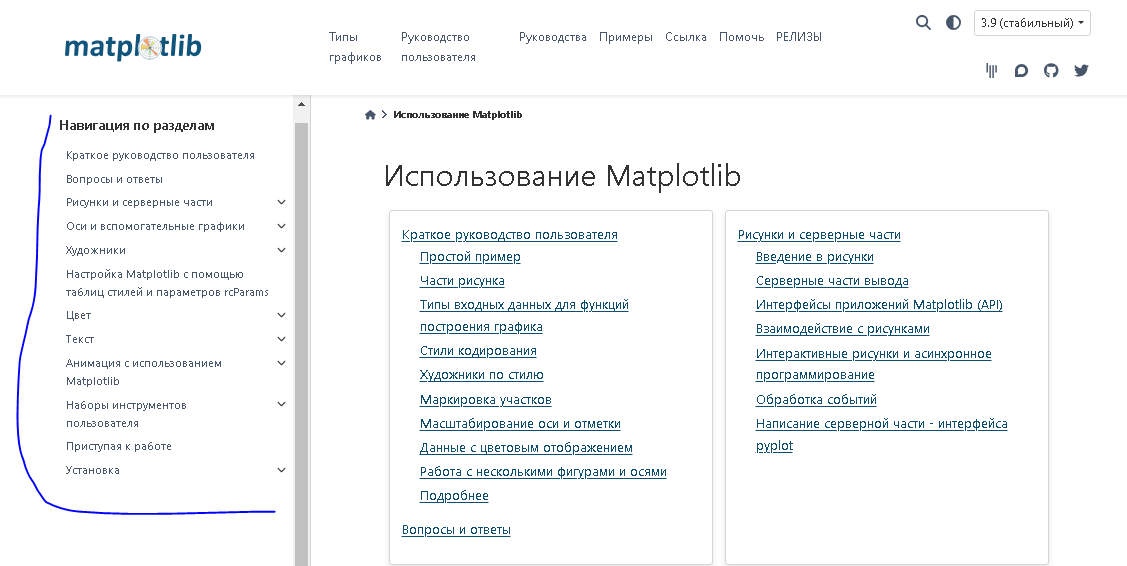


рис.2

При переходе к интересующему инструменту, выводится его полное описание, функционал и примеры кода (рис. 2.1, 2.2)



рис.2.1

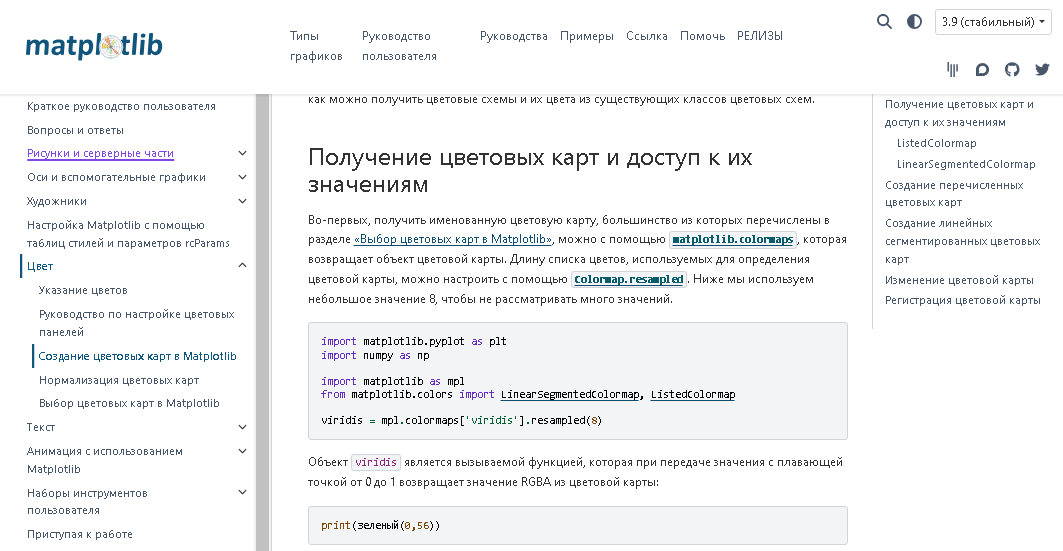


рис.2.2

Раздел «Plot types» (Типы графиков) содержит список доступных визуализаций (рис.3)

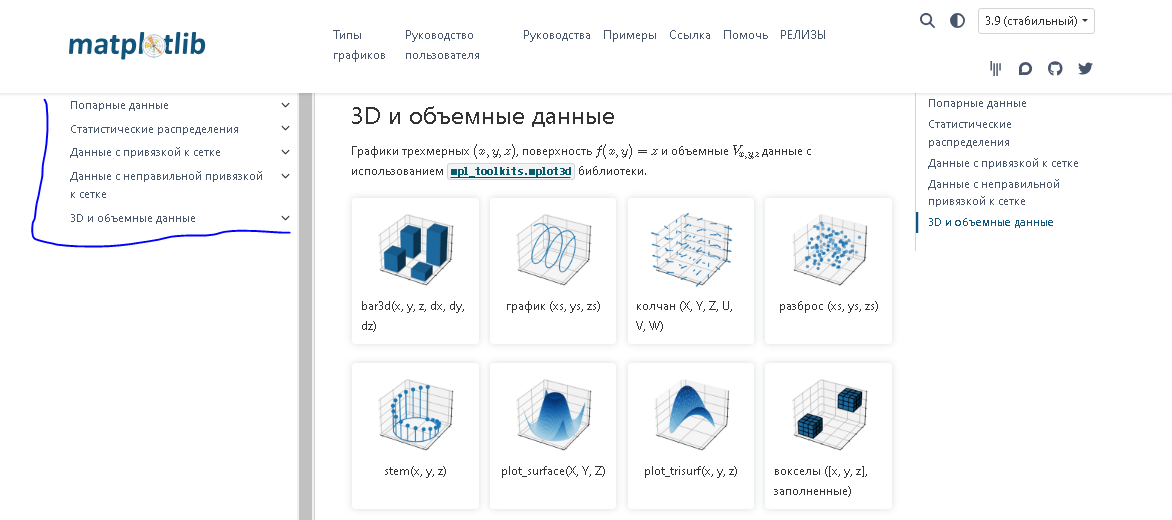


рис.3

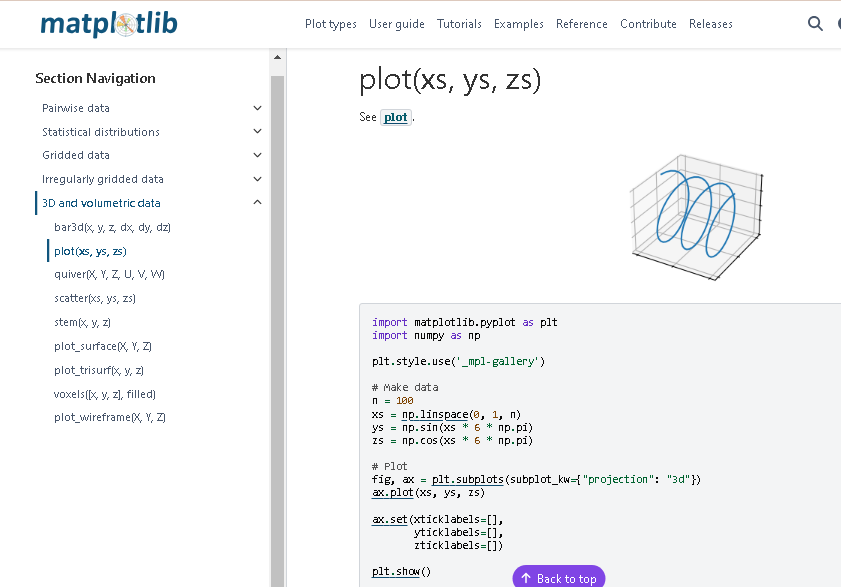
При переходе к интересующему графику отражается его визуализация и минимальный код Python необходимый для его построения (рис.3.1).

рис. 3.1

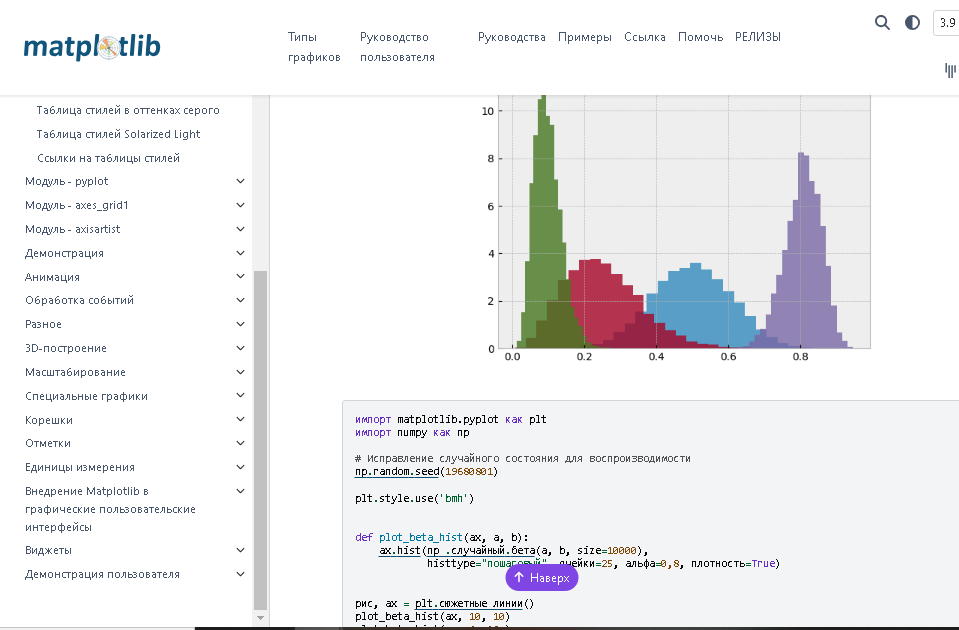
На сайте имеется отдельный раздел «Example» с примерами различных инструментов, стилей, коллекций, виджетов, вплоть до внедрения Matplotlib в различные пользовательские интерфейсы также содержащими код на языке Python (рис. 4).

рис.4

В разделе «Tutorials» (Руководства) содержатся учебные пособия с возможностью их скачивания для Python и Jupiter в формате .zip. (рис.5).

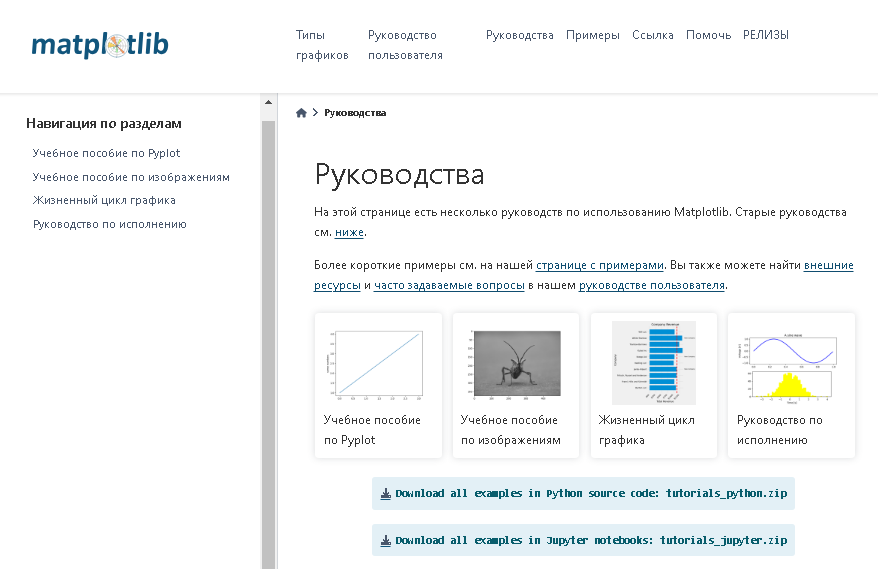


Рис.5

### Вывод по обзору официального сайта Matplotlib:

- легкость поиска необходимой информации (API), - отсутствует на первой странице сайта, находится в разделе документация, необходимо прокручивать, однако применяемые в примерах кода методы интерактивны и переводят в соответствующий раздел;

- наличие обучающих материалов и возможность их скачивания, - доступны на первой странице сайта в разделе «Tutorials» (Руководства);

- структура подачи материала, - интуитивно понятен, легко воспринимается;

- наличие примеров кода, - да.

|  |  |  |
| --- | --- | --- |
| № | критерий | оценка |
| 1. | легкость поиска необходимой информации (API) | 4 |
| 2. | наличие обучающих материалов и возможность их скачивания | 5 |
| 3. | структура подачи материала | 5 |
| 4. | наличие примеров кода | 5 |
| 5. | Наличие релизов | 5 |
| 5. | Итоговая оценка | 4.8 |

Сайт содержит максимально полную информацию о библиотеке, ее инструментах и методах, обучающую литературу для новичков, примеры кода для каждого вида визуализации данных, рекомендации по устранению возможных ошибок. В целом, пригоден для самостоятельного изучения библиотеки

## **Seaborn**

Официальный сайт [seaborn.pydata.org](https://seaborn.pydata.org/) на главной странице дает краткое описание библиотеки, имеет содержание, дает ссылку на свои страницы в **GitHub, stackoverflow, Twitter** создан в 2012 году(рис.1)

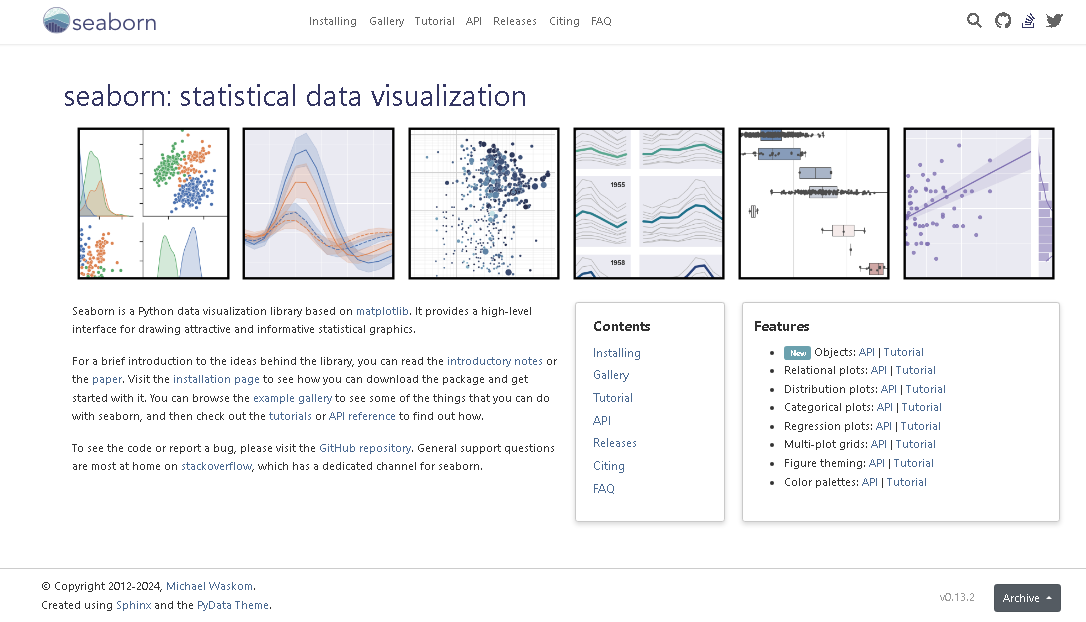


рис.1

Содержит разделы, - «Installing» (Установка), «Gallery» (Галерея), «Tutorial» (Учебник), API, «Releases» (Релизы), «Citing» (Цитаты), FAQ.

Раздел «Installing» (Установка) содержит информацию по установке библиотеки, зависимости ПО, возможные проблемы при отладке, и вызов справки (рис.2).

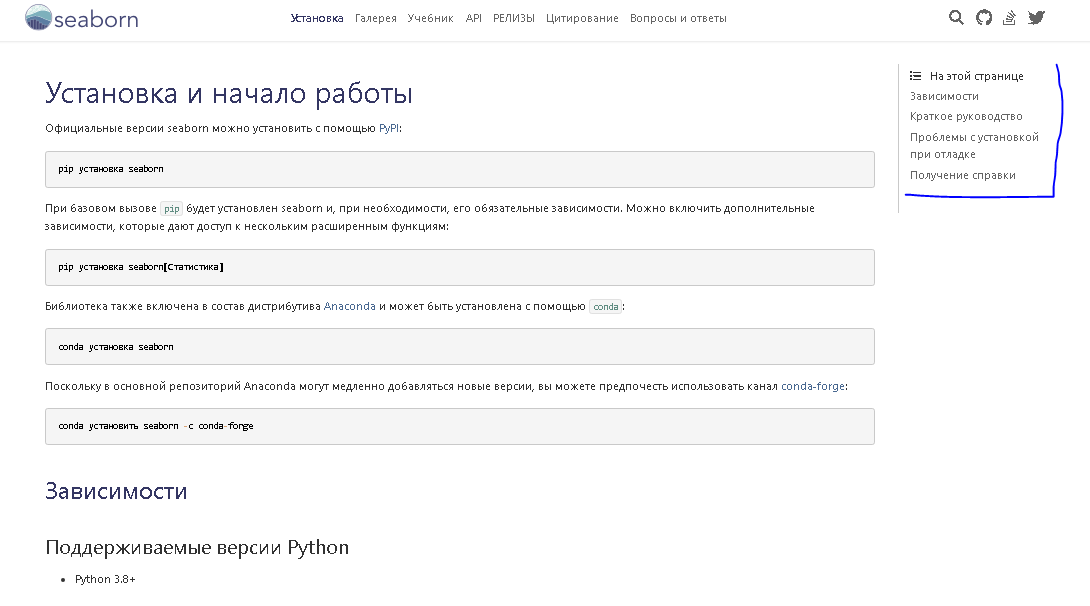


Рис.2

Раздел «Gallery» (Галерея) содержит интерактивные иконки с изображением примеров визуализации (рис.3), при нажатии на которые происходит переход на страницу с данной визуализацией, кроме того содержащей код на языке Python (рис.3.1) также под графиком отображены ссылки на используемые компоненты (рис 3.2) переход по которым открывает их описание (рис.3.3) включая примеры использования (рис. 3.4).

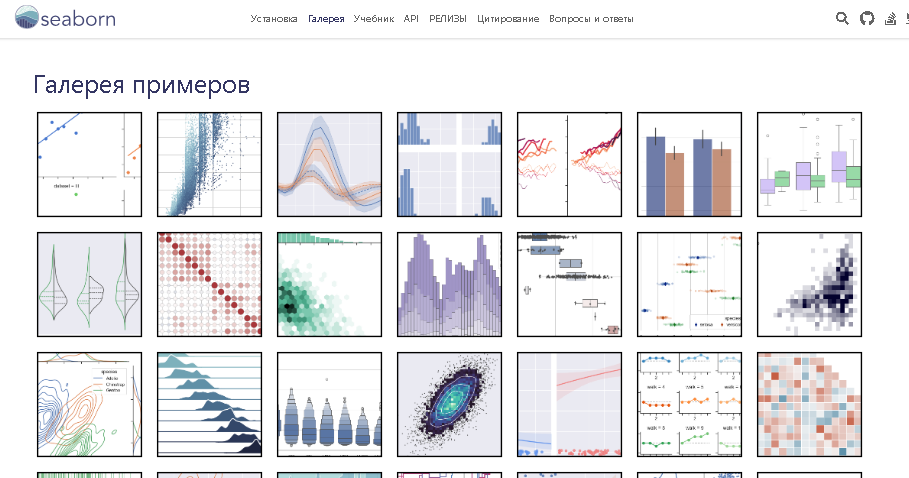


Рис.3

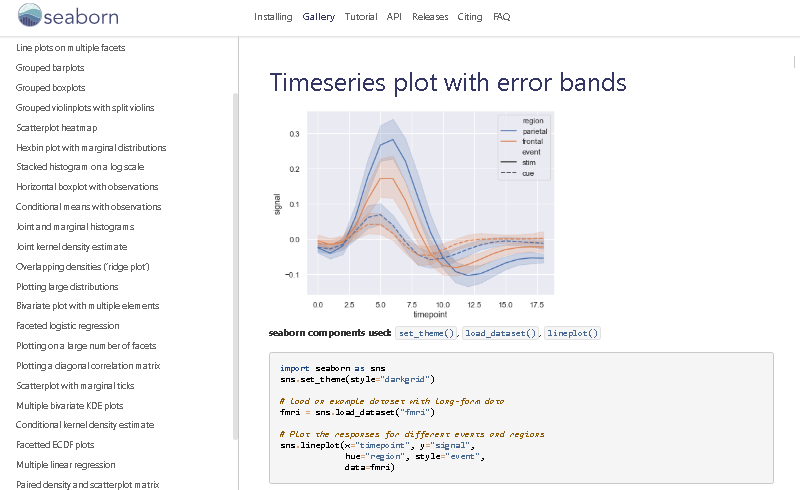


Рис. 3.1



Рис. 3.2

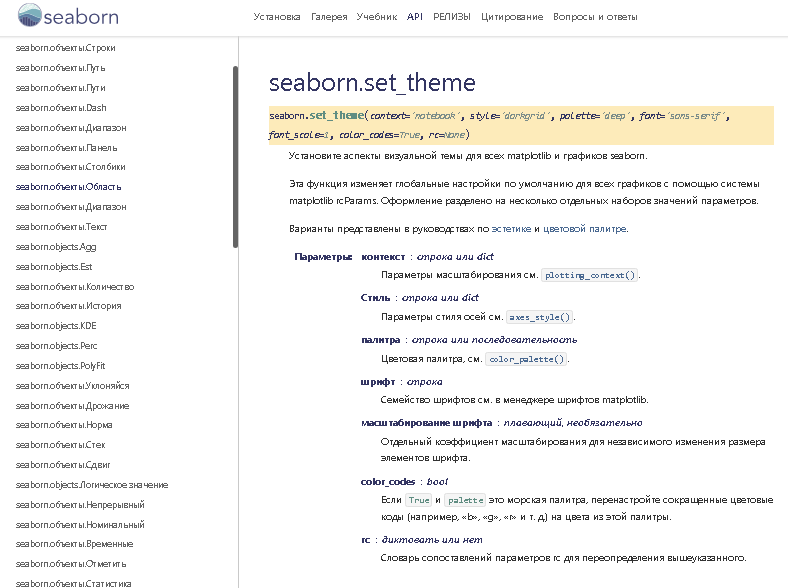


Рис. 3.3

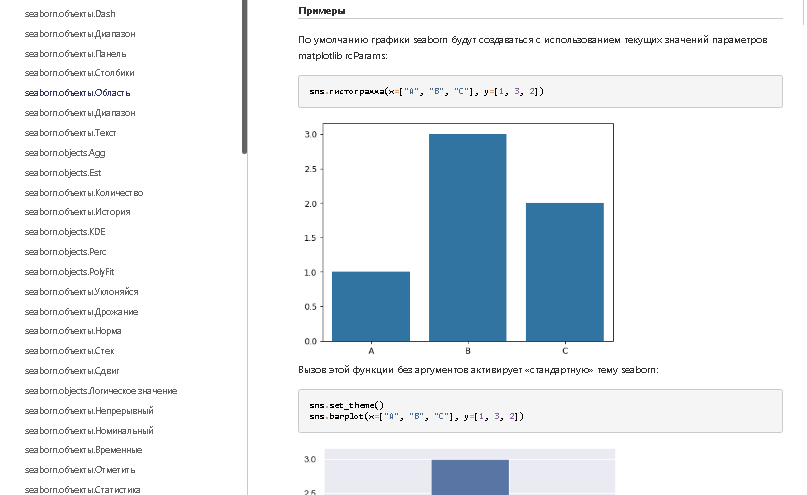


Рис. 3.4

Отдельно стоит отметить отображение в левой части документа при каждом переходе отображается список альтернативных вариантов, - на странице с графиками – другие варианты графиков, на странице с компонентами фигур – другие варианты компонентов.

Раздел «Tutorial» (Учебник) содержит учебные материалы, включая самоучитель (рис.4)

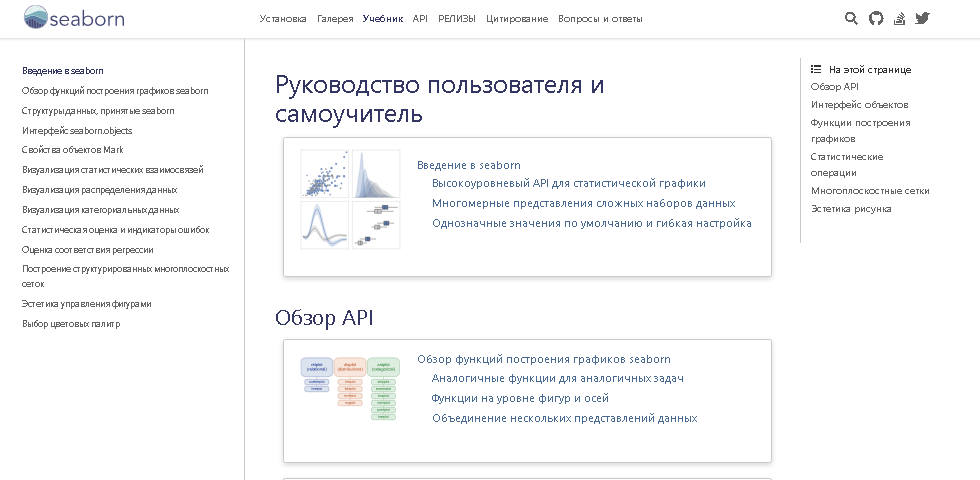


Рис.4

Раздел API содержит ссылки на объекты Seaborn, слева, как было отмечено выше отражен активный список всех объектов (рис. 5). При переходе к объекту отражается полная информация о нем, применимых к нему методов и конфигурациях, а также примеры кода на языке Python (рис. 5.1)

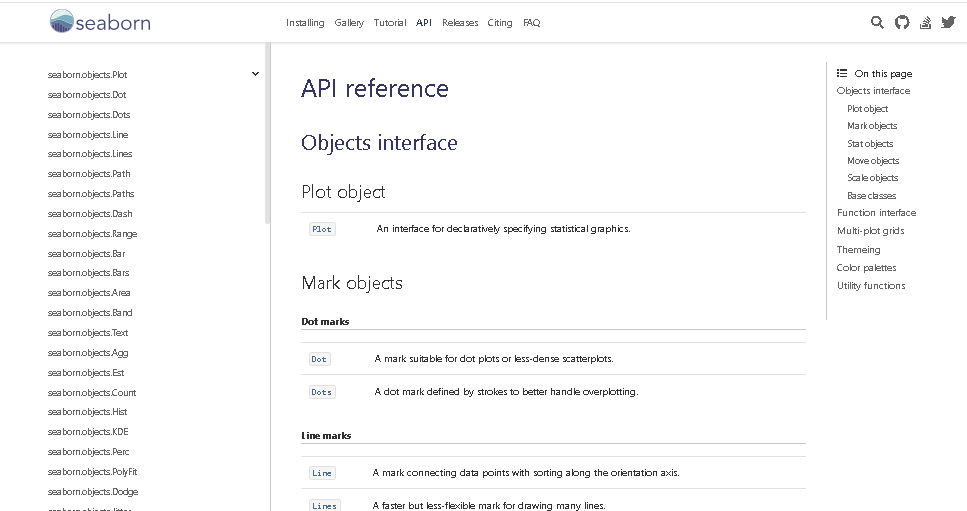


Рис. 5

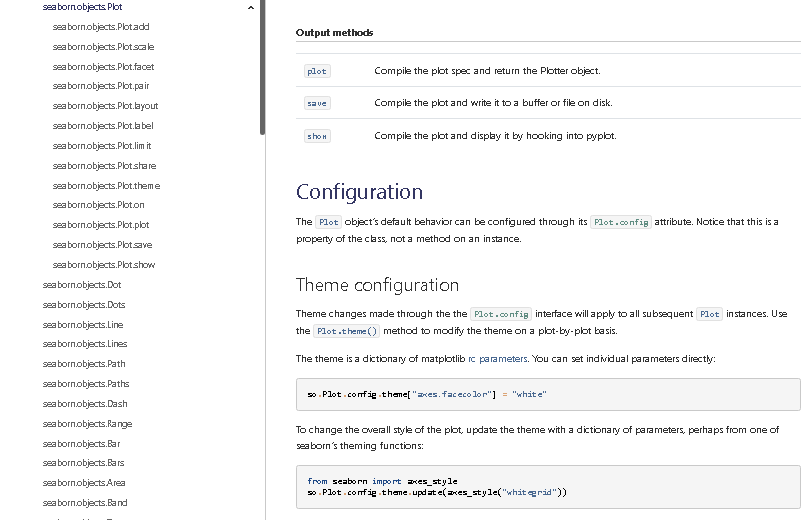


Рис. 5.1

Раздел «Releases» (Релизы) содержит все версии библиотеки, начиная с декабря 2013 года (рис.6)

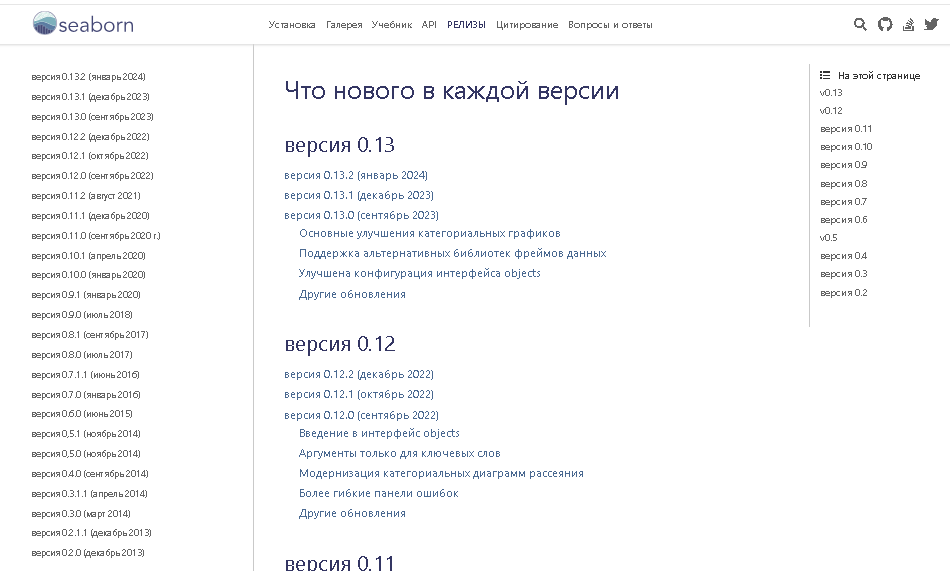


Рис.6

Раздел FAQ содержит ответы на часто задаваемые вопросы.

### Вывод по обзору официального сайта Seaborn:

- легкость поиска необходимой информации (API), - находится на первой странице сайта в блоках «**Features**» (Характеристики) и «**Contents**» (Содержание);

- наличие обучающих материалов и возможность их скачивания, - доступны на первой странице сайта в разделе в блоке «**Features**» (Характеристики) и «**Contents**» (Содержание);

- структура подачи материала, - интуитивно понятен, легко воспринимается;

- наличие примеров кода, - да, применяемые в коде методы не интерактивны, что при обучении вынуждает переходить в раздел сайта с API самостоятельно.

|  |  |  |
| --- | --- | --- |
| № | критерий | оценка |
| 1. | легкость поиска необходимой информации (API) | 4 |
| 2. | наличие обучающих материалов и возможность их скачивания | 5 |
| 3. | структура подачи материала | 5 |
| 4. | наличие примеров кода | 3 |
| 5. | Наличие релизов | 5 |
| 6. | Итоговая оценка | 4.4 |

Сайт имеет довольно непривычную навигацию, но понимание установленных связей происходит достаточно быстро. Возможно у начинающих специалистов вызовет затруднения. Содержит максимально полную информацию о библиотеке, ее инструментах и методах, обучающую литературу для новичков, примеры кода для каждого вида визуализации данных, рекомендации по устранению возможных ошибок. В целом, пригоден для самостоятельного изучения библиотеки.

## **Plotly**

Документация библиотеки для работы с Python размещена по адресу [plotly.com/python/](https://plotly.com/python/) на главной странице (рис.1) дает краткое описание библиотеки, имеет содержание (в левой части страницы) с интерактивными сносками, в верхней части страницы можно выбрать язык программирования, ссылки на форум и фреймворк Dash, - созданный компанией Plotlyдля анализа/визуализации данных на языках Python, R или Julia, в нижней правой части страницы встроен ASK AI, - инновационная функция на базе искусственного интеллекта которая использует передовую технологию обработки естественного языка (NLP) для понимания и интерпретации вопросов, задаваемых в повседневной разговорной речи. (рис.1.1), данный функционал, что очень важно принимает и обрабатывает запросы на русском языке (рис. 1.2)

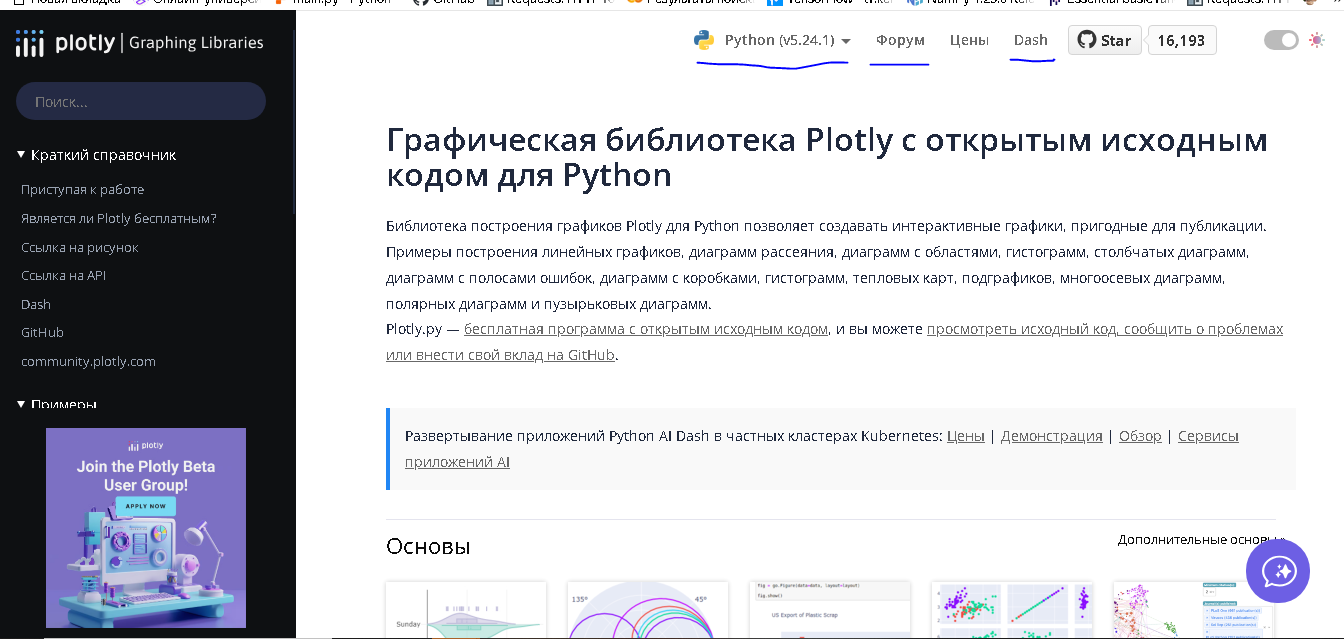


Рис.1



Рис.1.1

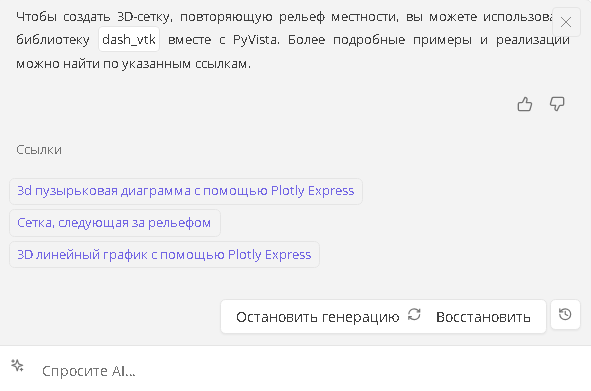


Рис. 1.2

Навигация по главной странице осуществляется прокруткой, что не совсем удобно для обзора, в каждом блоке отражается по пять примеров визуализации (рис.2) каждый является интерактивным. При переходе открывается страница с подробным описанием конкретного вида графиков, множеством примеров кода на выбранном языке программирования, а также результатами их выполнения в виде интерактивных графиков (рис.2.1), которые можно сохранить в формате .png (подчеркнуто на рисунке)

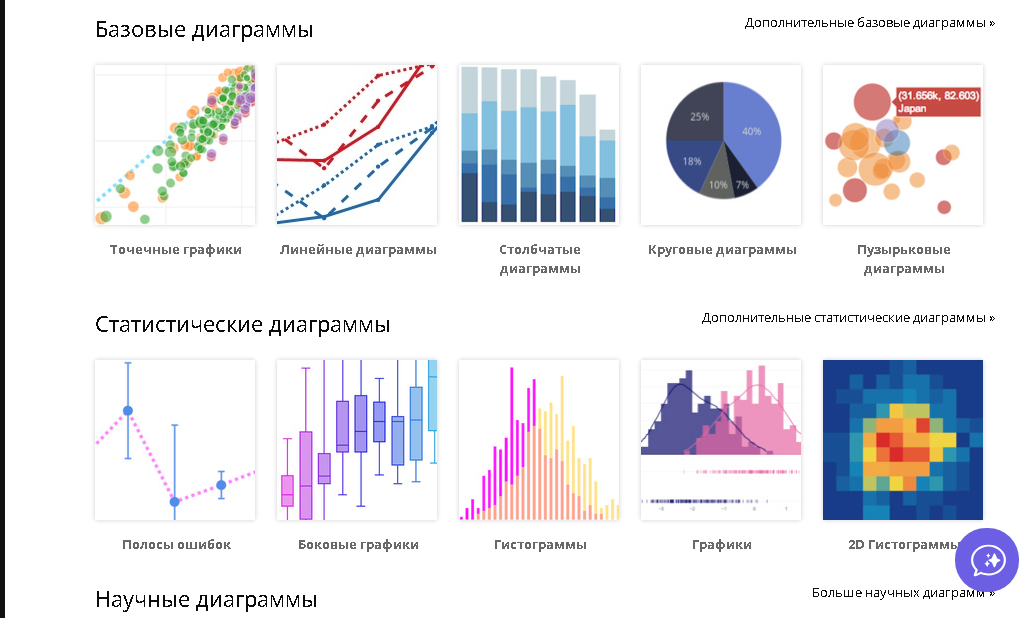


Рис. 2

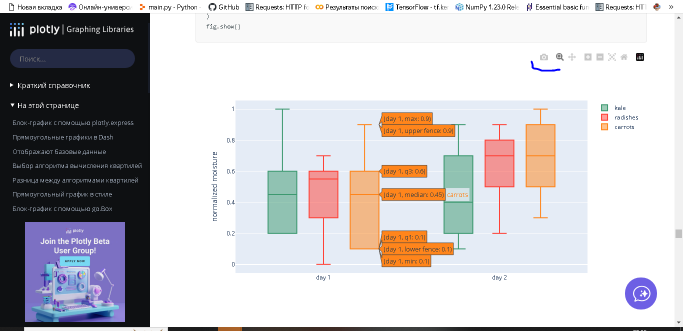


Рис. 2.1

Краткий справочник, в левой части экрана (рис.3) содержит ссылки на блоки с учебными материалами, подробным описанием построения визуализаций (рис.3.1, 3.1.1), интерактивными ссылками на API (рис.3.2) переводящими к их полному описанию (рис.3.2.1), Dash, GitHub и тематический форум разработчиков community.plotly.com

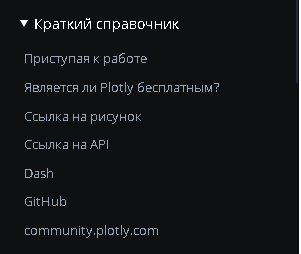


Рис.3



Рис.3.1

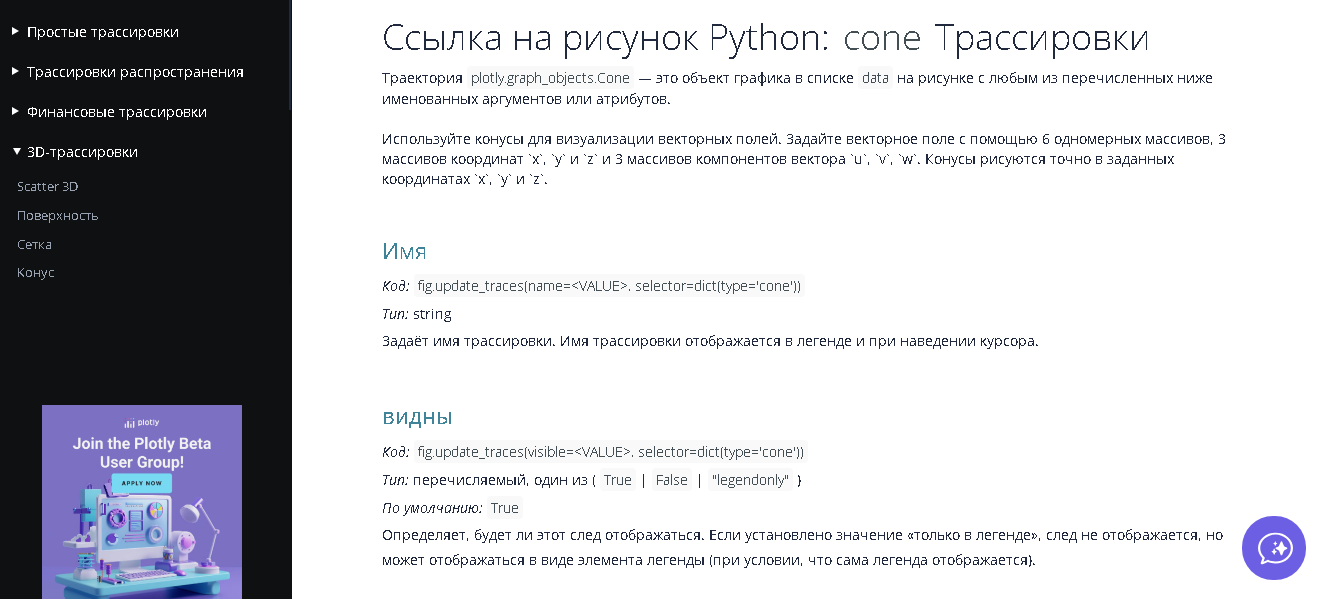


Рис.3.1.1

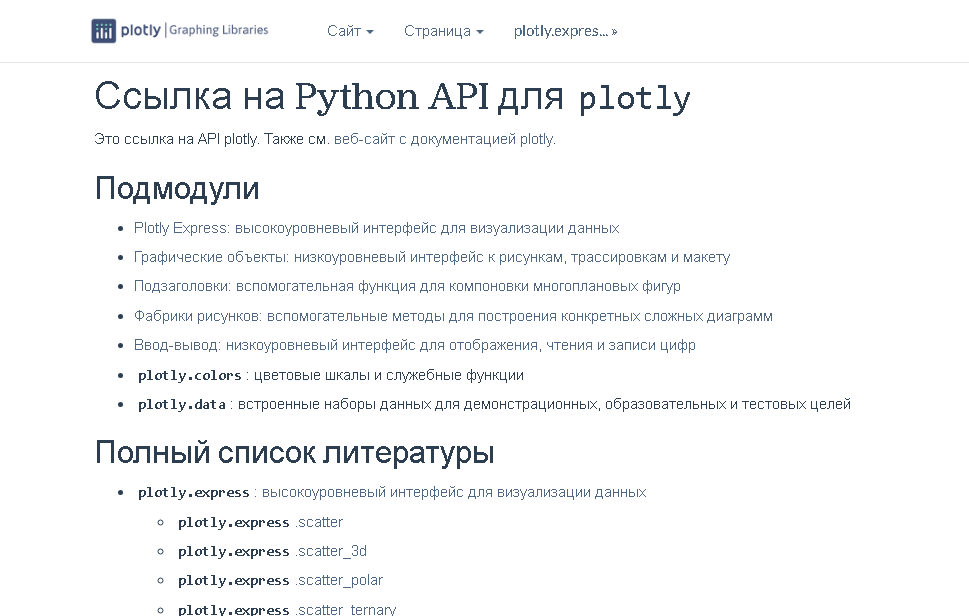


Рис. 3.2

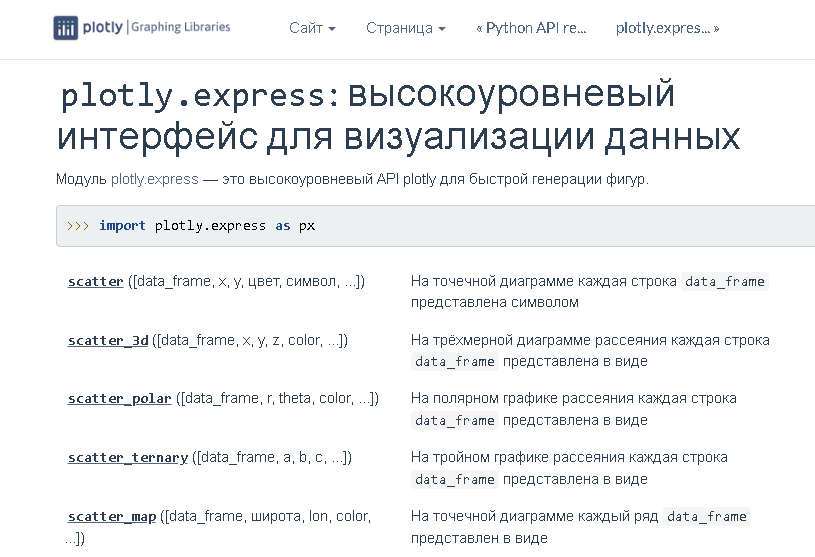


Рис. 3.2.1

Блок Dash переводит на сайт dash.plotly.com, который так же обладает встроенным ASK AI (рис. 4)

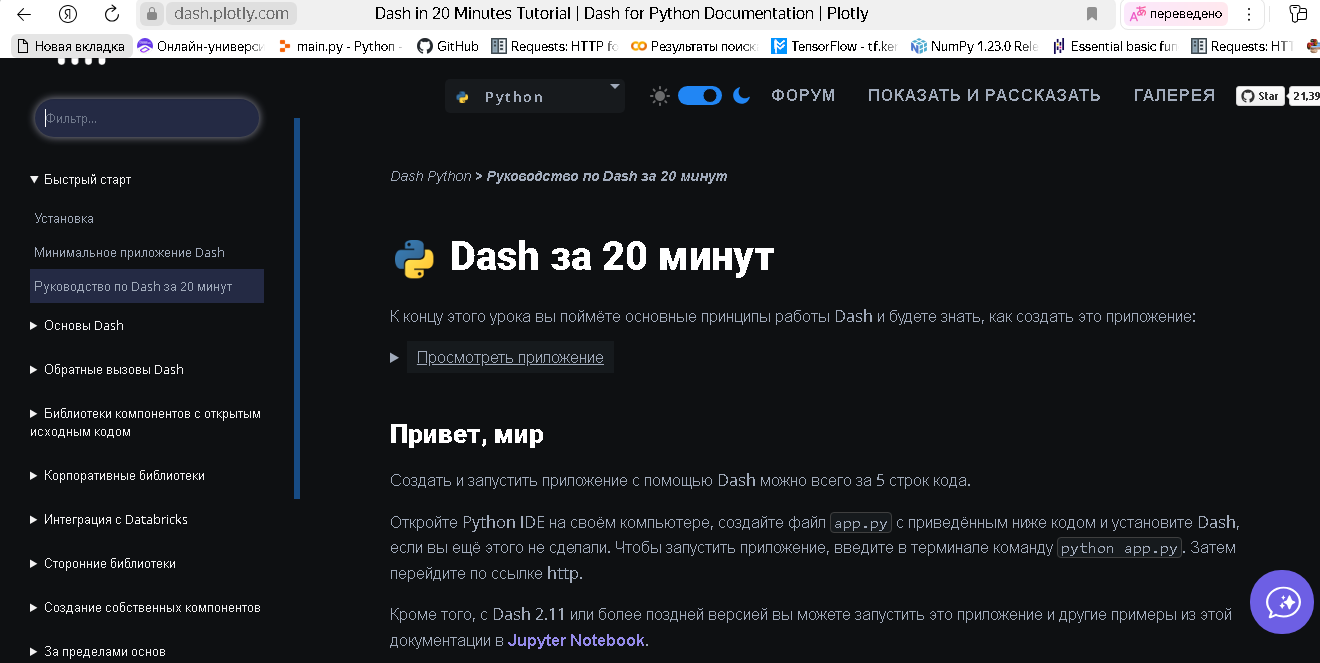


Рис.4

### Вывод по обзору официального сайта Plotly:

- легкость поиска необходимой информации (API), - находится на первой странице сайта в блоке «**Quick Reference**» (Краткий справочник);

- наличие обучающих материалов и возможность их скачивания, - доступны на первой странице сайта в разделе в блоке «**Quick Reference**» (Краткий справочник), возможность скачивания отсутствует;

- структура подачи материала, - интуитивно понятен, легко воспринимается, возможна обработка запросов с помощью встроенного искусственного интеллекта ASK AI;

- наличие примеров кода, - да, применяемые в коде методы интерактивны, что облегчает переход в необходимый раздел сайта с API.

|  |  |  |
| --- | --- | --- |
| № | критерий | оценка |
| 1. | легкость поиска необходимой информации (API) | 5 |
| 2. | наличие обучающих материалов и возможность их скачивания | 4 |
| 3. | структура подачи материала | 5 |
| 4. | наличие примеров кода | 5 |
| 5. | Наличие релизов | 5 |
| 6. | Итоговая оценка | 4.8 |

Сайт, его архитектура, наполнение и возможности превзошли все ожидания. Учитывая встроенный ASK AI изучение документации, а также поиск необходимого решения задач по силам даже новичкам в области визуализации.

# **Практическая часть исследования:**

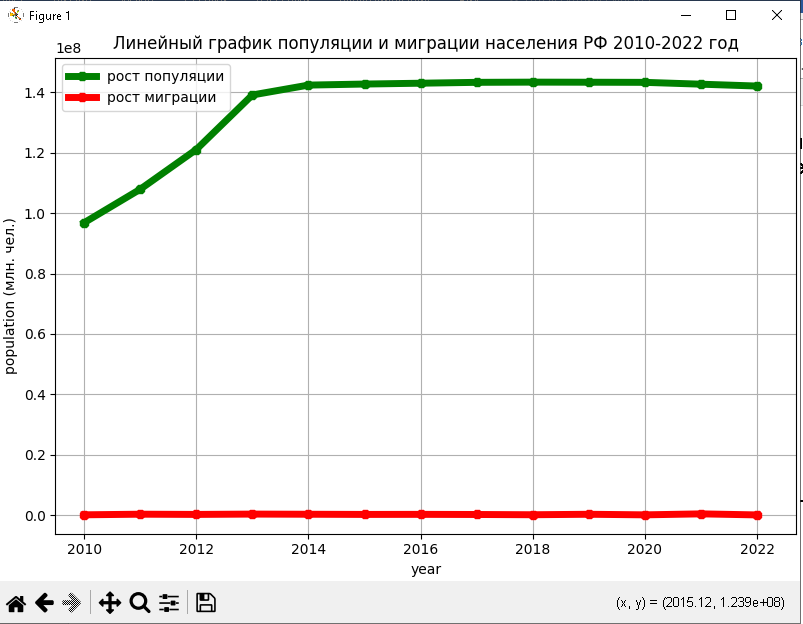
Для реализации практической части с использованием всех исследуемых библиотек, с целью построения наиболее распространенных графиков, в качестве объекта визуализации, был сформирован датасет **data\_mun\_demography\_123\_v20240612.csv** содержащий сведения о популяции, смертности, рождаемости и миграции в России в период с 2010 по 2022 год:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| year | population | deaths | births | migration |
| 2010 | 96835169 | 1721108 | 1473894 | 88582 |
| 2011 | 107915196 | 1777845 | 1571089 | 288555 |
| 2012 | 121076849 | 1745423 | 1684264 | 215920 |
| 2013 | 139146879 | 1843328 | 1825672 | 324983 |
| 2014 | 142418256 | 1865506 | 1899250 | 272014 |
| 2015 | 142763919 | 1861645 | 1897339 | 213128 |
| 2016 | 143040123 | 1845888 | 1848712 | 237789 |
| 2017 | 143332001 | 1785678 | 1657729 | 193544 |
| 2018 | 143397127 | 1789372 | 1574306 | 110299 |
| 2019 | 143365227 | 1759010 | 1452994 | 265918 |
| 2020 | 143325162 | 2093277 | 1409152 | 76256 |
| 2021 | 142718676 | 2389085 | 1371462 | 399792 |
| 2022 | 142103031 | 1857505 | 1280079 | 51887 |

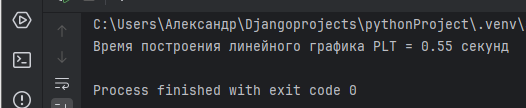
При написании кода для измерения временного интервала формирования визуализации использовалась библиотека **time.**

# Визуализация линейных графиков

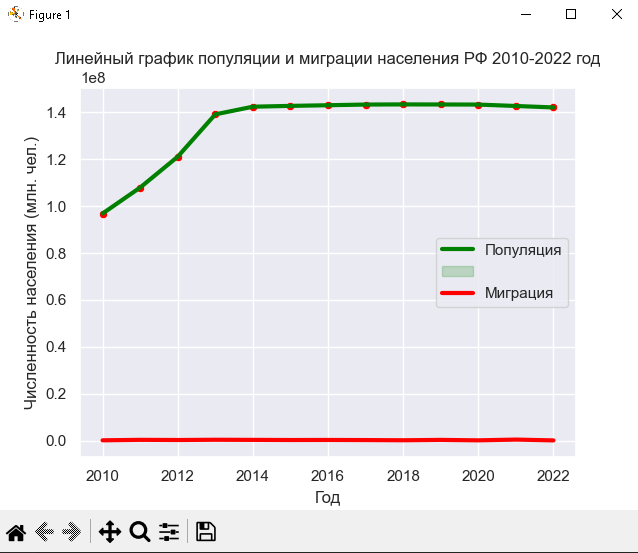
## Результат построения линейного графика Matplotlib:



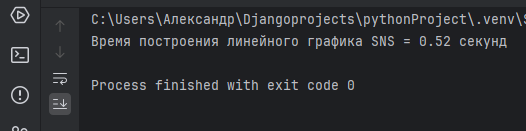
Объем кода составляет 8 строк.



## Результат построения линейного графика Seaborn:



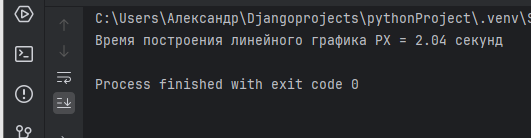
Объем кода составляет 7 строк.



## Результат построения линейного графика Plotly:

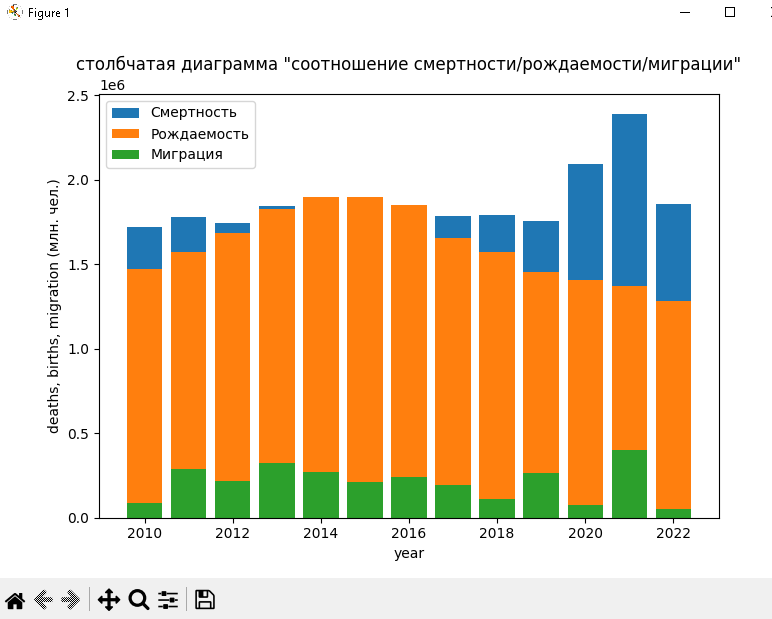


Объем кода составляет 7 строк.

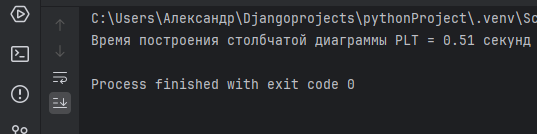


# Визуализация столбчатых диаграмм

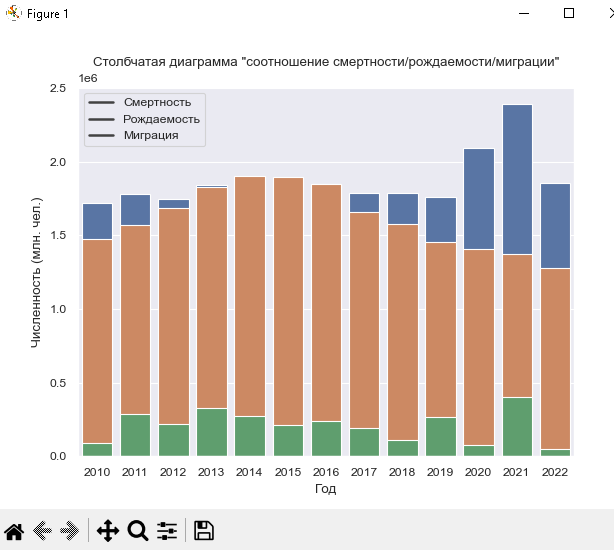
## Результат построения столбчатой диаграммы Matplotlib:



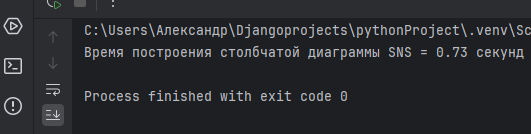
Объем кода составляет 8 строк.



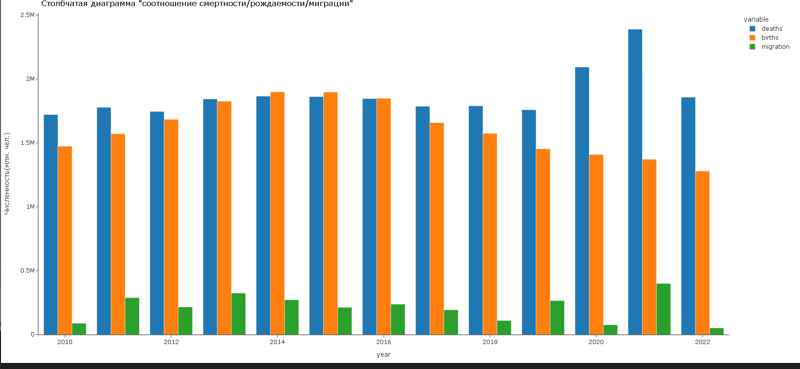
## Результат построения столбчатой диаграммы Seaborn:



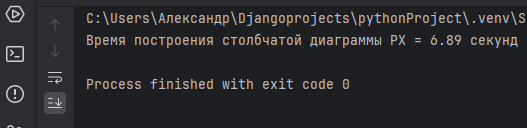
Объем кода составляет 8 строк.



## Результат построения столбчатой диаграммы Plotly:

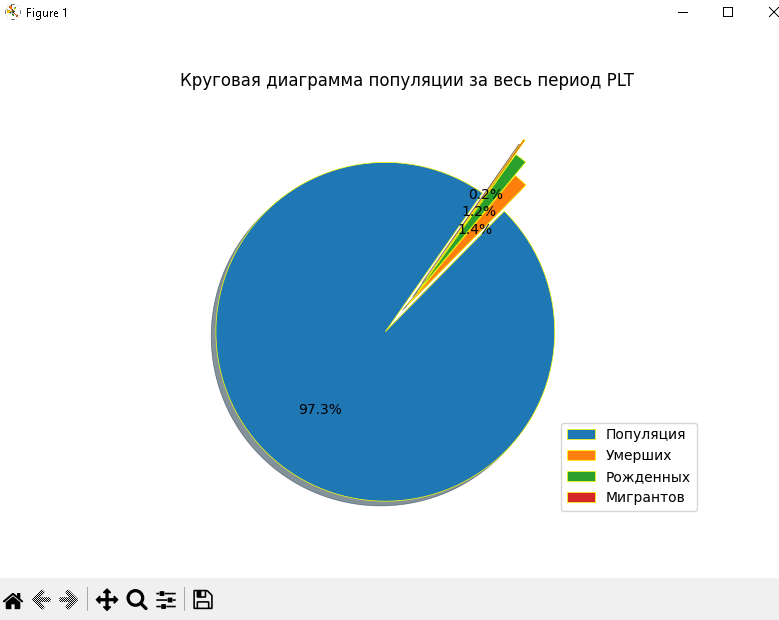


Объем кода составляет 2 строки.

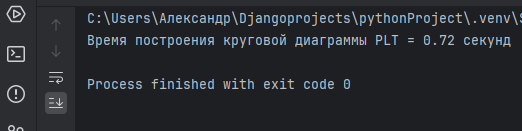


# Визуализация круговых диаграмм

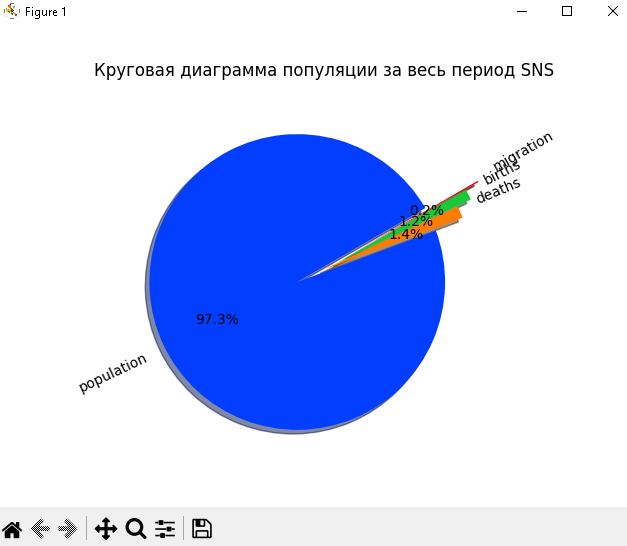
## Результат построения Круговой диаграммы Matplotlib:



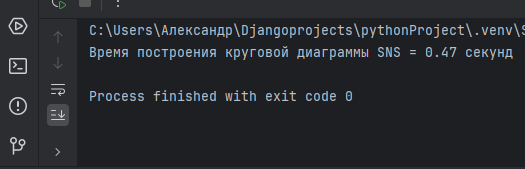
Объем кода составляет 6 строк.



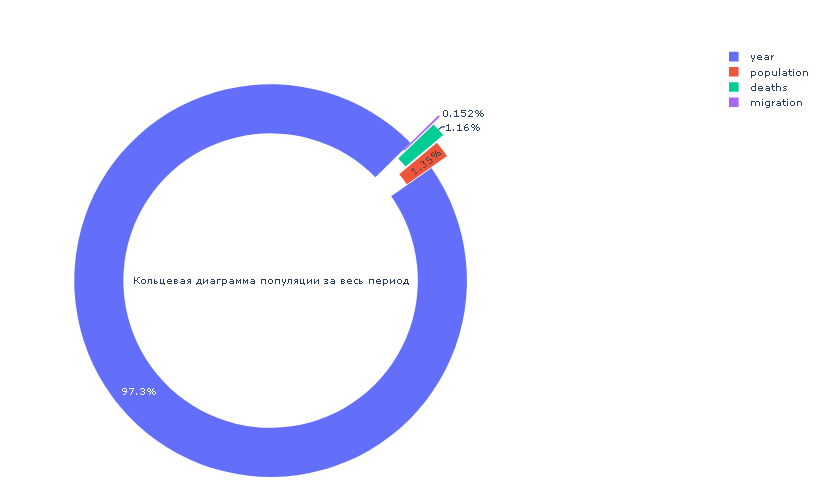
## Результат построения Круговой диаграммы Seaborn:



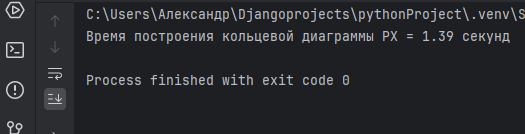
Объем кода составляет 6 строк.



## Результат построения Кольцевой диаграммы Plotly:



Объем кода составляет 4 строки.

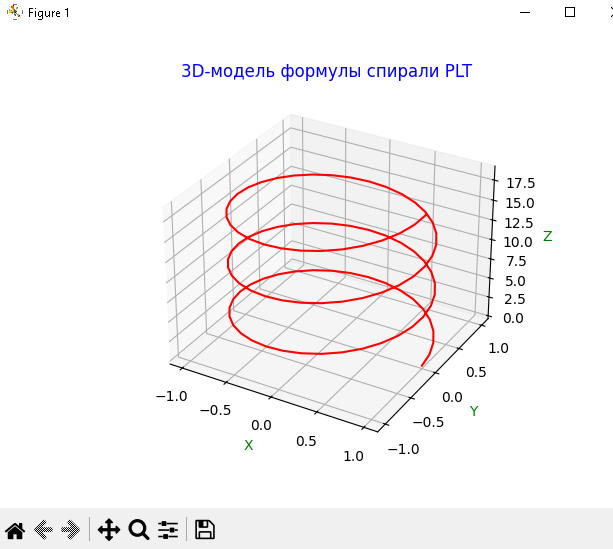


# Визуализация 3D-моделей

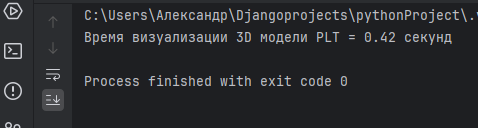
Для построения 3D-визуализации использована формула спирали в декартовой системе координат:

**x = ρ cos ϕ, y = ρ sin ϕ**.

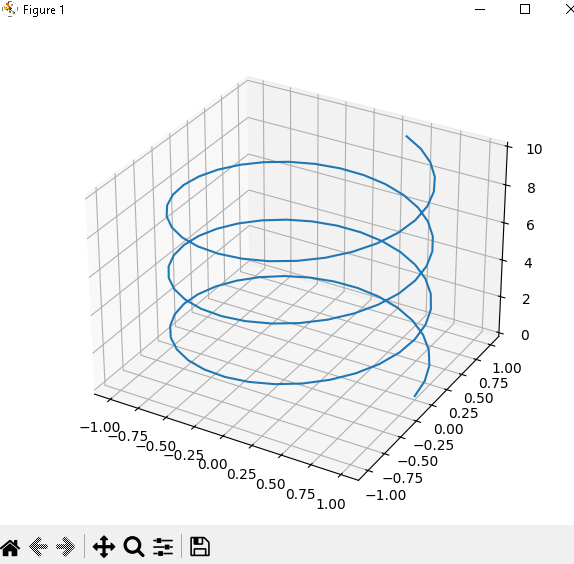
## Результат построения 3D модели Matplotlib:



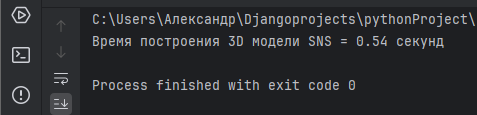
Объем кода составляет 15 строк.



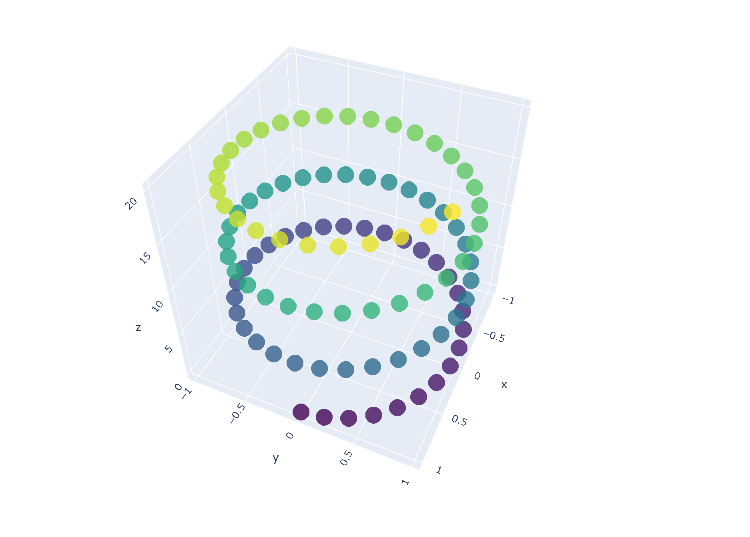
## Результат построения 3D модели Seaborn:



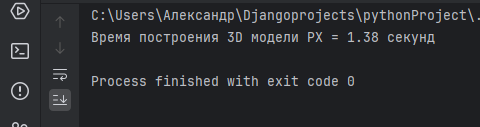
Объем кода составляет 8 строк.



## Результат построения 3D модели Plotly:



Объем кода составляет 4 строки.



## Сводная таблица по реализованному объему кода:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | библиотека | Объем кода линейный график | Объем кода столбчатая диаграмма | Объем кода круговая диаграмма | Объем кода 3D-модель | Общий  объем кода |
| 1. | Matplotlib | 8 | 8 | 6 | 15 | 39 |
| 2. | Seaborn | 7 | 8 | 6 | 8 | 31 |
| 3. | Plotly | 7 | 2 | 4 | 4 | 17 |

## Сводная таблица по временным показателям построения графиков:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | библиотека | Затраченное время линейный график (сек.) | Затраченное время столбчатая диаграмма (сек.) | Затраченное время круговая диаграмма(сек.) | Затраченное время 3D-модель (сек.) | Общее время (сек.) |
| 1. | Matplotlib | 0.55 | 0.51 | 0.72 | 0.42 | 2.2 |
| 2. | Seaborn | 0.52 | 0.73 | 0.47 | 0.54 | 2.26 |
| 3. | Plotly | 2.04 | 6.89 | 1.39 | 1.38 | 11.7 |

Возможность взаимодействия с построенными визуализациями как то, - изменение размера, зум, перемещение, отображение информации в конкретной точке графика присутствует только со стороны разработчика до сохранения его в формат «png» во всех исследуемых библиотеках. Для сохранения графиков в иных форматах требуется либо написание дополнительного кода, либо установка дополнительных пакетов или расширений библиотек, например, для Matplotlib и Seaborn matplotlib.animation, для Plotly, - plotly-gif. Учитывая, что у Matplotlib и Seaborn, - объем кода более громоздкий итоговый балл по данному пункту снижается.

## Сводная таблица оценки по практической части исследования:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | критерий | Оценка Matplotlib | Оценка Seaborn | Оценка Plotly |
| 1. | объем кода | 3 | 4 | 5 |
| 2. | скорость построения графиков | 5 | 4 | 3 |
| 3. | дополнительные возможности | 4.5 | 4.5 | 5 |
| 4. | Итоговая оценка | 4.16 | 4.16 | 4.3 |

# **Вывод по библиотеке Matplotlib**:

Matplotlib — универсальная библиотека для визуализации данных на Python. Она позволяет создавать статические, интерактивные и анимированные визуализации. Эффективна для обработки больших наборов данных, т.к. построена поверх NumPy. Готовые результаты визуализации можно экспортировать в форматах PDF, SVG, JPG, PNG, BMP и GIF. является основой для других библиотек, например, Seaborn, который упрощает процедуру построения графиков и имеет больше возможностей для улучшения внешнего вида графиков.

# **Вывод по библиотеке Seaborn**:

Seaborn — мощный инструмент для визуализации данных. Он позволяет легко создавать как простые, так и сложные графики. Библиотека помогает визуализировать статистические взаимосвязи. Она позволяет понять, как переменные в наборе данных связаны друг с другом и как эта взаимосвязь зависит от других переменных, включает в себя множество встроенных функций для статистического анализа, таких как регрессия и распределение данных. Seaborn интегрируется с библиотекой pandas. Это позволяет легко работать с DataFrame и другими структурами данных. позволяет совмещать несколько графиков для более сложного анализа данных.

# **Вывод по библиотеке Plotly**:

В целом, Plotly — мощный интерактивный инструмент для визуализации данных, который приносит значительную ценность при работе с информацией и обеспечивает возможность создания красивых и понятных графиков и диаграмм, позволяет создавать графики и диаграммы, которые реагируют на действия пользователя, такие как наведение курсора, щелчок и масштабирование.  Plotly поддерживает различные платформы и языки программирования, включая Python, JavaScript, R и MATLAB.  Графики, созданные с помощью Plotly, можно экспортировать в различные форматы (например, PNG, SVG) и встраивать в веб-страницы, блокноты Jupyter и другие приложения. Имеет онлайн-платформу Plotly Chart Studio, где пользователи могут создавать, редактировать и делиться графиками.

# **Общий вывод:**

Проведенное исследование показало, что все библиотеки по праву занимают верхние строчки по популярности в среде разработчиков и удобны в использовании. Доступность литературы обеспечивает пользователям возможность самостоятельного поиска и выбора документов. Наиболее функциональной себя показала библиотека Plotly, которая отличается своей интерактивностью и гибкостью. Выбор использования конкретной библиотеки в большей степени зависит от требований заказчика и того, какие данные подлежат визуализации.