

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт перспективной инженерии  
Департамент цифровых, робототехнических систем и электроники

**ОТЧЕТ**  
**ПО ПРАКТИЧЕСКОЙ РАБОТЕ №3**  
**дисциплины**  
**«Искусственный интеллект и машинное обучение»**  
**Вариант 12**

Выполнил:  
Рябинин Егор Алексеевич  
2 курс, группа ИВТ-б-о-23-2,  
09.03.01 «Информатика и  
вычислительная техника»,  
направленность (профиль)  
«Программное обеспечение средств  
вычислительной техники и  
автоматизированных систем», очная  
форма обучения

---

(подпись)

Проверил:  
Доцент департамента цифровых,  
робототехнических систем и  
электроники института перспективной  
инженерии  
Воронкин Роман Александрович

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

Ставрополь, 2025 г

**Тема:** Основы работы с библиотекой matplotlib.

**Цель:** исследовать базовые возможности библиотеки matplotlib языка программирования Python.

**Порядок выполнения работы:**

Ссылка на репозиторий GitHub:

[https://github.com/EgorGorilla/Lab3\\_Artificial-Intelligence-and-Machine-Learning](https://github.com/EgorGorilla/Lab3_Artificial-Intelligence-and-Machine-Learning)

1. Задания практической работы.

✓ Задание 1. Построение простого графика

Напишите код, который строит график функции  $y = x^2$  на интервале  $[-10,10]$ . Добавьте заголовок, подписи осей и сетку.

```
[20] import matplotlib.pyplot as plt
import matplotlib inline
import numpy as np
x = np.linspace(-10,10)
y = x**2
plt.plot(x,y)
plt.title("График функции y = x^2")
plt.xlabel("x")
plt.ylabel("y")
plt.grid()
```

Рисунок 1 – Задание 1. Построение простого графика

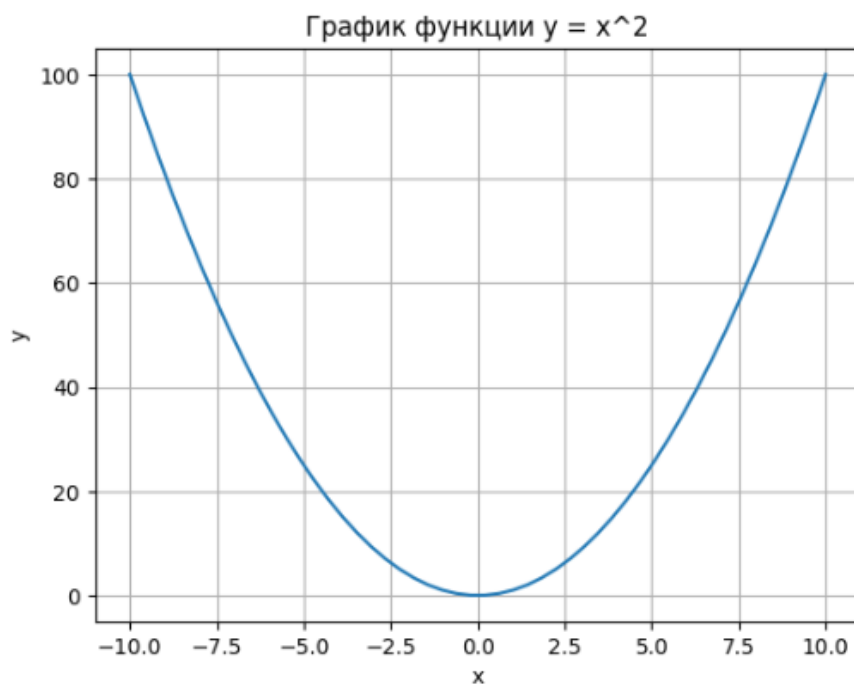


Рисунок 2 – Результат работы программы к заданию №1

## ✓ Задание №2. Настройка стилей и цветов

Постройте три линии на одном графике:

- $y = x$  (синяя, пунктирная линия),
- $y = x^2$  (зеленая, штрих-пунктирная линия),
- $y = x^3$  (красная, сплошная линия).

Добавьте легенду и сделайте оси одинакового масштаба

```
0 2сек
x = np.linspace(-1,1)
y = x**2
y1 = x**3
y2 = x
plt.plot(x,y,"g-.",label="x = y^2")
plt.plot(x,y1,"r",label="x = y^3")
plt.plot(x,y2,"b--",label="x = y")
plt.xlabel("x")
plt.ylabel("y")
plt.title("Графики функций y = x, y = x^2, y = x^3")
plt.legend()
plt.grid()
plt.axis("equal")
```

Рисунок 3 – Задание №2. Настройка стилей и цветов

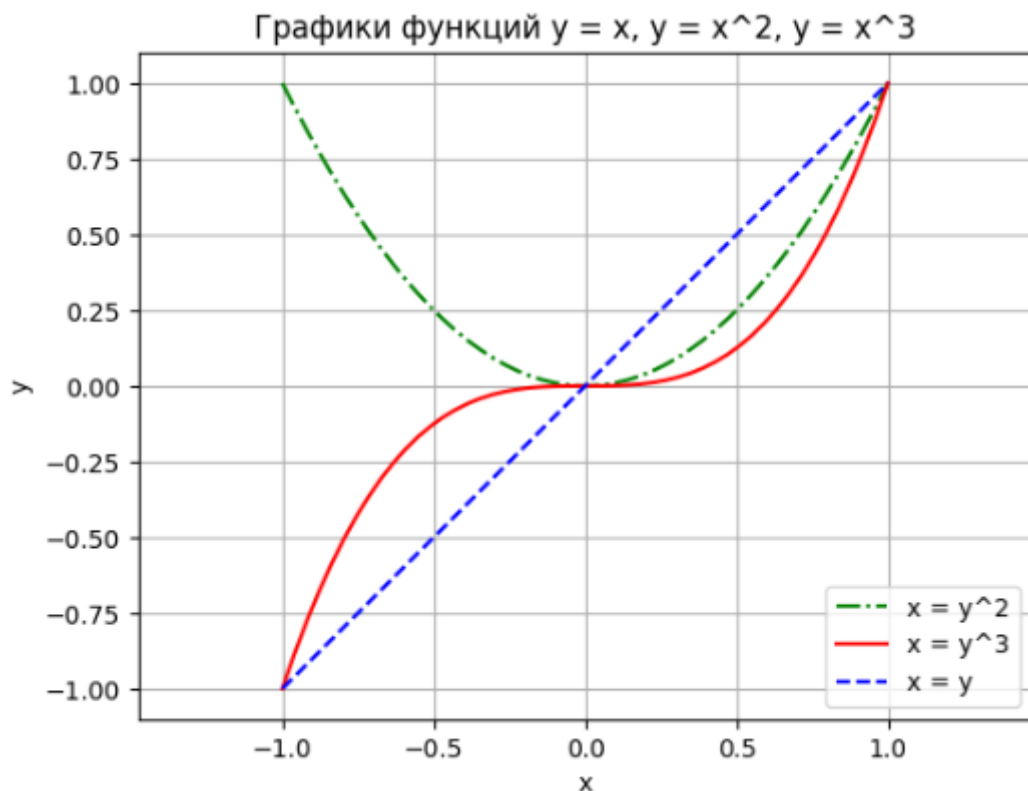


Рисунок 4 – Результат работы программы к заданию №2

### ✓ Задание №3. Использование различных типов графиков

Сгенерируйте 50 случайных точек и постройте диаграмму рассеяния (scatter plot), где цвет точек зависит от их координаты по оси x, а размер точек зависит от координаты по оси y.

```
[76] x = np.random.uniform(-10, 10, 50)
      y = np.random.uniform(-10, 10, 50)
      colors = x
      sizes = (y - min(y)) / (max(y) - min(y)) * 200 + 50
      plt.figure(figsize=(8, 6))
      plt.scatter(x, y, c=colors, s=sizes, cmap='viridis', alpha=0.7, edgecolors='k')
      plt.title('Диаграмма рассеяния (scatter plot)')
      plt.xlabel('x')
      plt.ylabel('y')
      plt.grid()
      plt.colorbar(label='x')
```

Рисунок 5 – Задание №3. Использование различных типов графиков

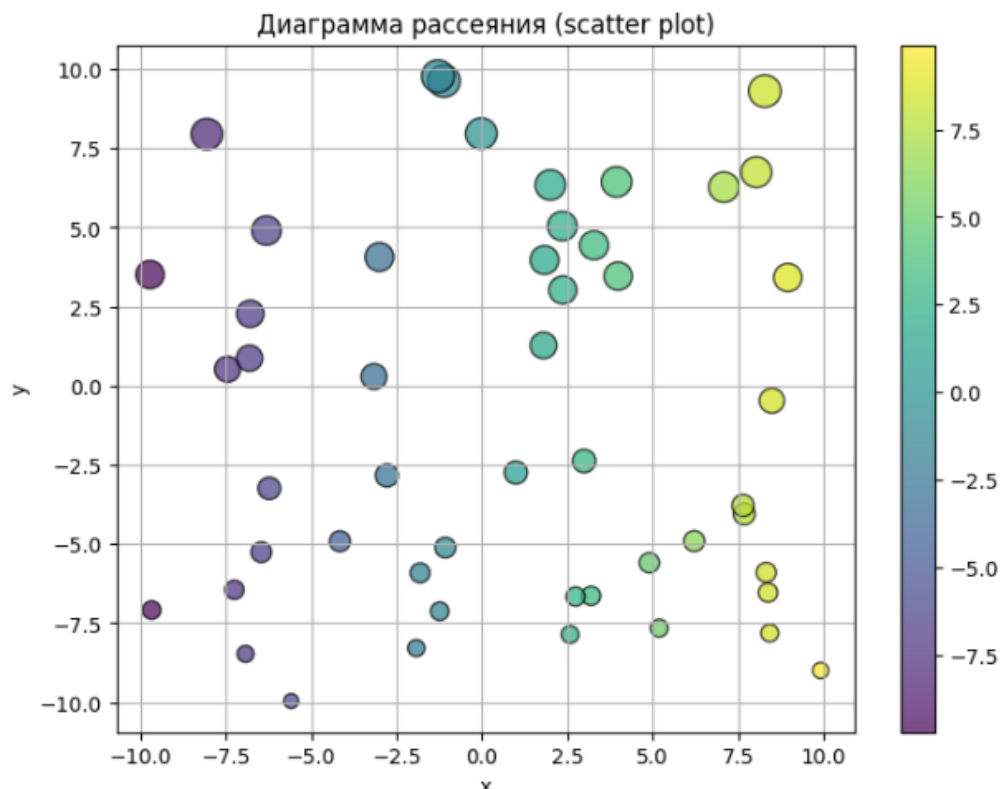


Рисунок 6 – Результат выполнения программы к заданию №3

### ✓ Задание №4. Гистограмма распределения

Сгенерируйте 1000 случайных чисел из нормального распределения с параметрами  $\mu = 0$ ,  $\sigma = 1$  и постройте их гистограмму с 30 бинами. Добавьте вертикальную линию в среднем значении.

```
[87] norm = np.random.normal(0,1,1000)
      plt.hist(norm,bins=30,color='blue',edgecolor='black',alpha=0.7)
      plt.axvline(np.mean(norm), color='red')
```

Рисунок 7 – Задание №4. Гистограмма распределения

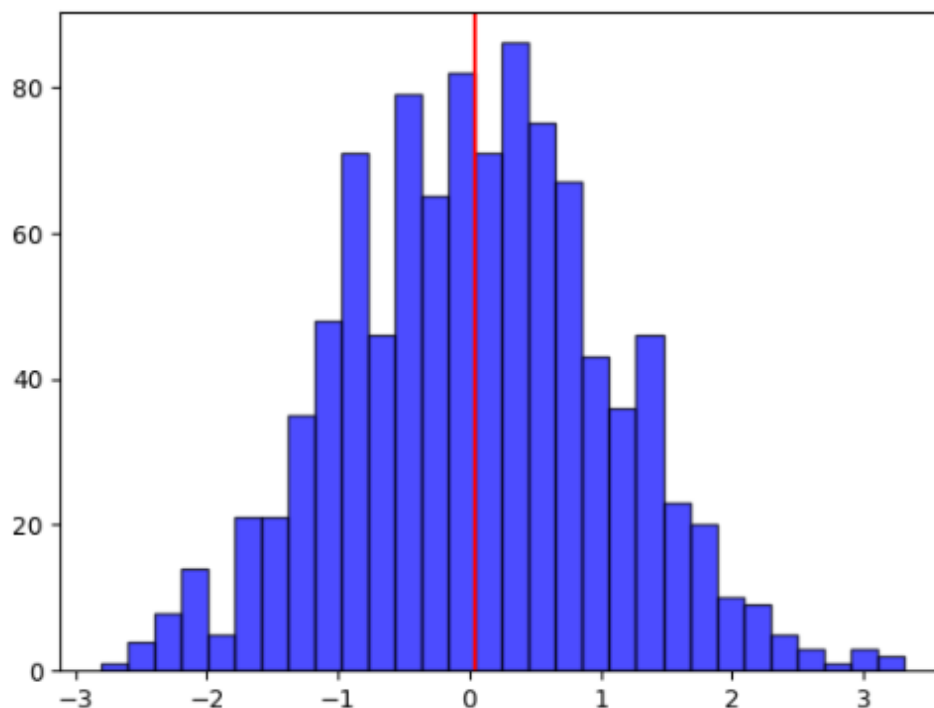


Рисунок 8 – Результат работы программы к заданию №4

#### ✓ Задание №5. Столбчатая диаграмма

Создайте столбчатую диаграмму, которая показывает количество студентов, получивших оценки:

- "Отлично" - 20 человек,
- "Хорошо" - 35 человек,
- "Удовлетворительно" - 30 человек,
- "Неудовлетворительно" - 15 человек.

Добавьте подписи к осям и заголовок.

```
[92] marks = ["Отлично", "Хорошо", "Удовлетворительно", "Неудовлетворительно"]
      guys = [20, 35, 30, 15]
      plt.bar(marks, guys, color=['green', 'yellow', 'orange', 'red'])
      plt.xlabel("Оценки")
      plt.ylabel("Количество студентов")
      plt.title("Успеваемость студентов")
```

Рисунок 9 – Задание №5. Столбчатая диаграмма

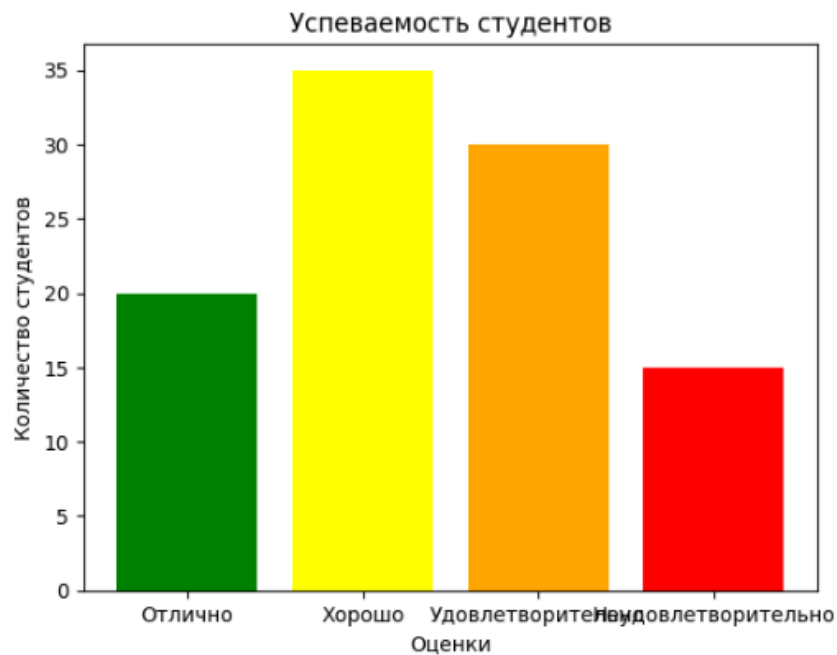


Рисунок 10 – Результат работы программы к заданию №5

#### ✓ Задание №6. Круговая диаграмма

Используя данные предыдущей задачи, постройте круговую диаграмму с процентными подписями секторов.

```
[102] marks = ["Отлично", "Хорошо", "Удовлетворительно", "Неудовлетворительно"]
guys = [20, 35, 30, 15]
plt.figure(figsize=(6, 6))
plt.pie(guys, labels=marks, autopct='%1.1f%%', colors=['green', 'yellow', 'orange', 'red'], startangle=140)
plt.title("Успеваемость студентов")
```

Рисунок 11 – Задание №6. Круговая диаграмма

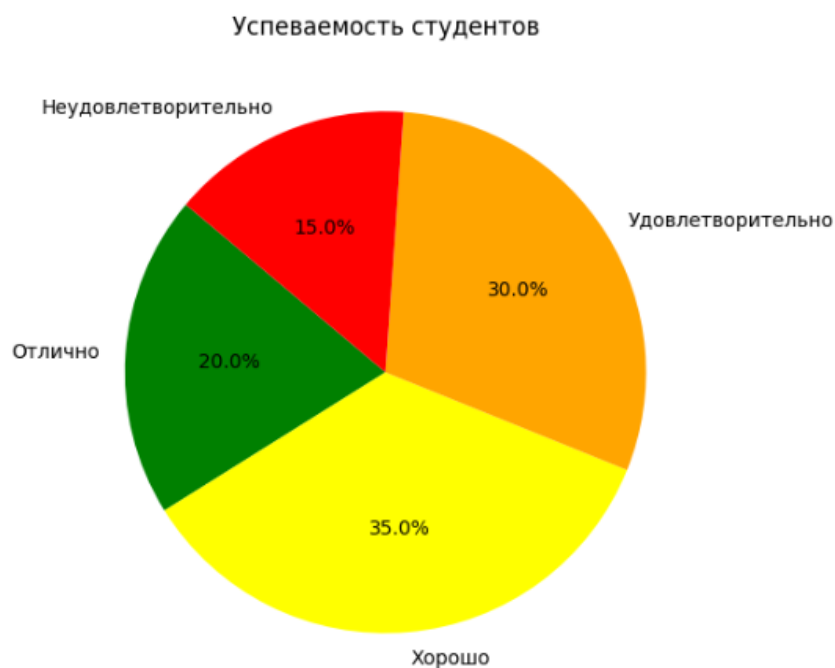


Рисунок 12 – Результат работы программы к заданию №6

### ✓ Задание №7. Трехмерный график поверхности

Используя `mpl_toolkits.mplot3d`, постройте 3D-график функции  $z = \sin(\sqrt{x^2 + y^2})$  на сетке значений  $x, y$  в диапазоне  $[-5, 5]$ .

```
[165] from mpl_toolkits.mplot3d import Axes3D
x = np.linspace(-5,5,100)
y = np.linspace(-5,5,100)
X, Y = np.meshgrid(x, y)
Z = np.sin(np.sqrt(X**2 + Y**2))
fig = plt.figure(figsize=(8,6))
a = fig.add_subplot(111,projection='3d')
a.plot_surface(X,Y,Z,cmap='viridis')
a.set_xlabel('x')
a.set_ylabel('y')
a.set_zlabel('z')
a.set_title('3D-график z = sin(√(x² + y²))')
```

Рисунок 13 – Задание №7. Трехмерный график поверхности

3D-график  $z = \sin(\sqrt{x^2 + y^2})$

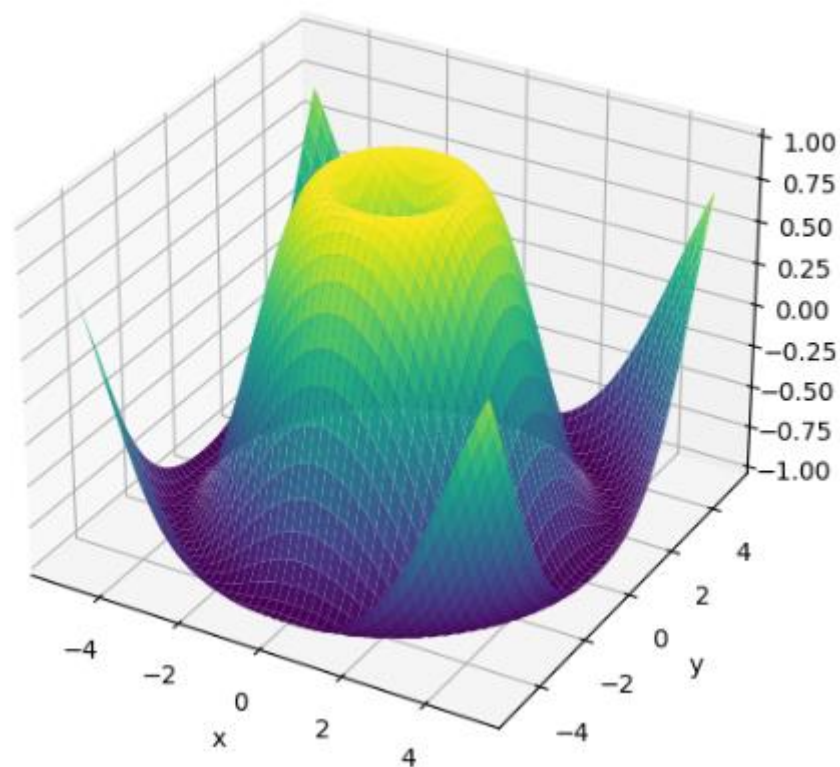


Рисунок 14 – Результат работы программы к заданию №7

## ✓ Задание №8. Множественные подграфики [subplots]

Постройте четыре графика в одной фигуре (2x2 сетка):

1. Линейный график  $y = x$
2. Парабола  $y = x^2$
3. Синус  $y = \sin(x)$
4. Косинус  $y = \cos(x)$

Добавьте заголовки к каждому подграфику.

```
[106] x = np.linspace(-10,10)
      fig, axes = plt.subplots(2,2,figsize=(10,8))
      axes[0,0].plot(x,x,color='blue')
      axes[0,0].set_title('y = x')
      axes[0,1].plot(x,x**2,color='green')
      axes[0,1].set_title('y = x^2')
      axes[1,0].plot(x,np.sin(x),color='red')
      axes[1,0].set_title('y = sin(x)')
      axes[1,1].plot(x,np.cos(x), color='purple')
      axes[1,1].set_title('y = cos(x)')
```

Рисунок 15 – Задание №8. Множественные подграфики [subplots]

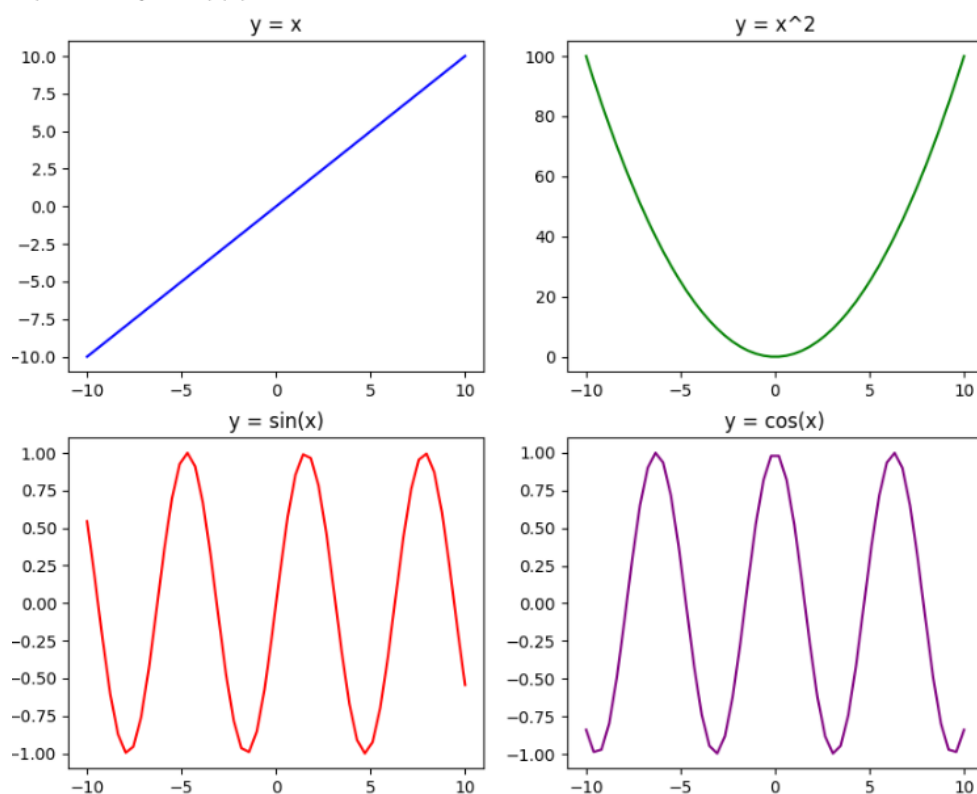


Рисунок 16 – Результат работы программы к заданию №8



## ✓ Задание №9. Тепловая карта (imshow)

Создайте случайную матрицу 10x10 с элементами от 0 до 1 и визуализируйте её как тепловую карту с цветовой шкалой.

```
[108] matrix = np.random.rand(10,10)
      plt.imshow(matrix,cmap='viridis',aspect='auto')
      plt.colorbar()
      plt.title("Тепловая карта (imshow)")
```

Рисунок 17 – Задание №9. Тепловая карта (imshow)

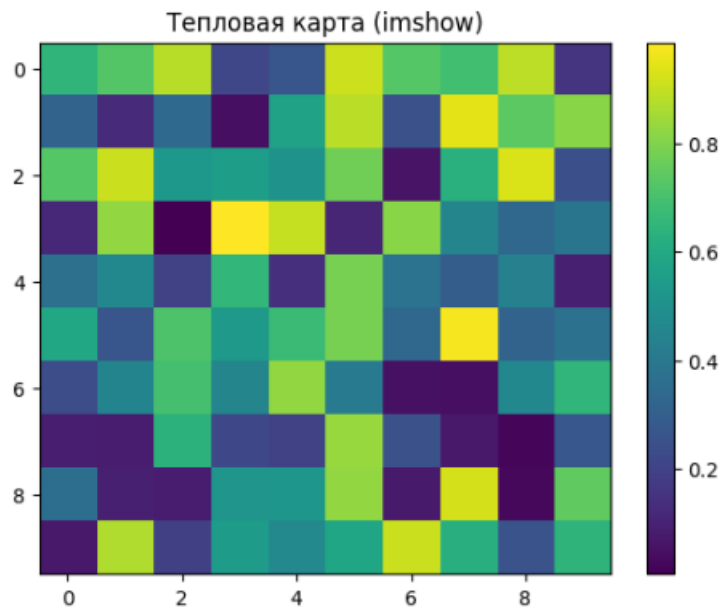


Рисунок 18 – Результат работы программы к заданию №9

## 2. Индивидуальные практические задания.

### Задача на построение линейных графиков в Matplotlib

#### Вариант 12. Изменение уровня освещенности в течении дня

Фиксировался уровень освещенность в люксах (lx) на улице в зависимости от времени суток:

- Время суток (часы): [0,3,6,9,12,15,18,21,24]
- Освещенность (lx): [0,5,500,15000,40000,35000,10000,100,0]

Используйте заливку области под графиком.

```
hours = [0, 3, 6, 9, 12, 15, 18, 21, 24]
lx = [0, 5, 500, 15000, 40000, 35000, 10000, 100, 0]
plt.plot(hours, lx, color='orange', linewidth=2)
plt.fill_between(hours, lx, color='orange', alpha=0.5)
plt.xlabel("Время суток (часы)")
plt.ylabel("Освещенность (lx)")
plt.title("Изменение уровня освещенности в течение дня")
```

Рисунок 19 – Индивидуальное задание №1. Задача на построение линейных графиков в Matplotlib

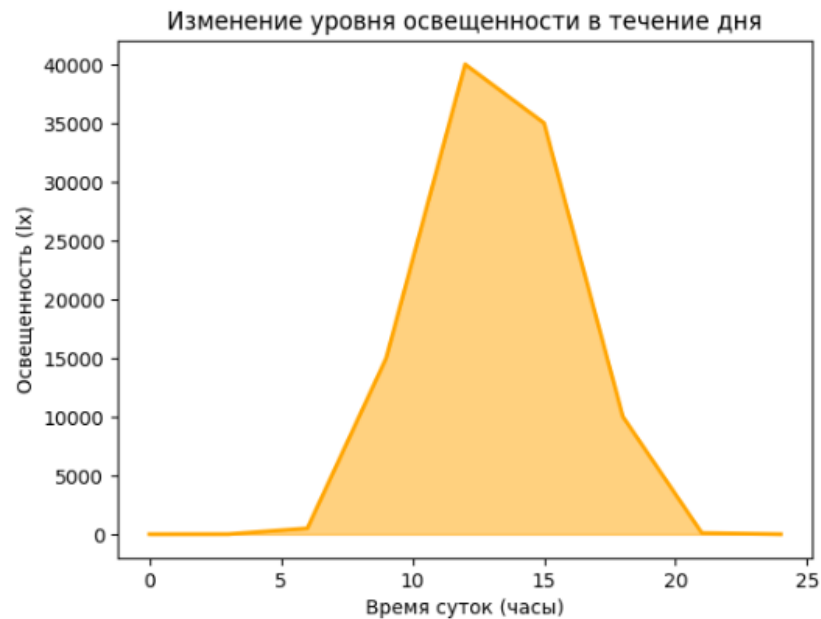


Рисунок 20 – Результат работы программы к индивидуальному заданию №1

### ✓ Задача на построение столбчатых диаграмм в Matplotlib

#### Вариант 12. Расход воды в разных городах

Средний расход воды на человека в день (литры):

- Города: ["Москва", "Берлин", "Лондон", "Париж", "Нью-Йорк"]
- Расход (л): [250, 160, 150, 140, 300]

Добавьте пунктирную линию, обозначающую средний расход по всем городам.

```
[121] city = ['Москва', 'Берлин', 'Лондон', 'Париж', 'Нью-Йорк']
      rasxod = [250, 160, 150, 140, 300]
      average_rasxod = np.mean(rasxod)
      plt.bar(city, rasxod, color='lightblue')
      plt.axhline(y=average_rasxod, color='red', linestyle='--')
      plt.title('Расход воды в разных городах')
      plt.xlabel('Города')
      plt.ylabel('Расход воды (л)')
```

Рисунок 21 – Индивидуальное задание №2. Задача на построение столбчатых диаграмм в Matplotlib

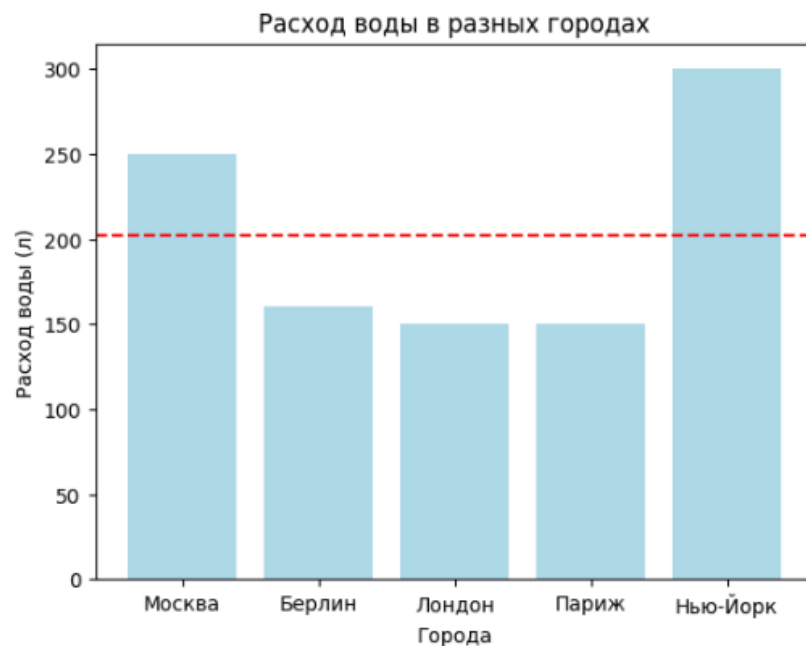


Рисунок 22 – Результат работы программы к индивидуальному заданию №2

#### ✓ Задачи на вычисление определенного интеграла с помощью Matplotlib

В каждой задаче требуется:

1. Построить график подинтегральной функции.
2. Вычислить площадь под кривой на заданном отрезке как значение определенного интеграла.
3. Закрасить область под графиком, чтобы визуализировать интеграл.

#### Вариант 12. Площадь под модулем синуса

Определите интеграл  $f(x) = |\sin(x)|$  на отрезке  $[0,1]$ .

#### Дополнительные требования

- В каждой задаче используйте функцию `fill_between()` для закрашивания площади под кривой.
- Добавьте подписи осей и заголовков.
- При необходимости, отобразите оси X и Y для лучшей визуализации.

```
[143] def f(x):
      return np.abs(np.sin(x))
      a = 0
      b = 1
      x = np.linspace(a,b,1000)
      y = f(x)
      S = np.trapz(y,x)
      plt.plot(x,y,label=r'$f(x)=|\sin(x)|$', color='blue')
      plt.fill_between(x,y,color="lightblue",alpha=0.5)
      plt.text(0.5,0.5,f'Площадь = {S:.5f}',ha='center')
      plt.title(r'Площадь под графиком $f(x) = |\sin(x)|$ на отрезке [0,1]')
      plt.xlabel('x')
      plt.ylabel('y')
      plt.grid()
```

Рисунок 23 – Индивидуальное задание №3. Задача на вычисление определенного интеграла с помощью Matplotlib

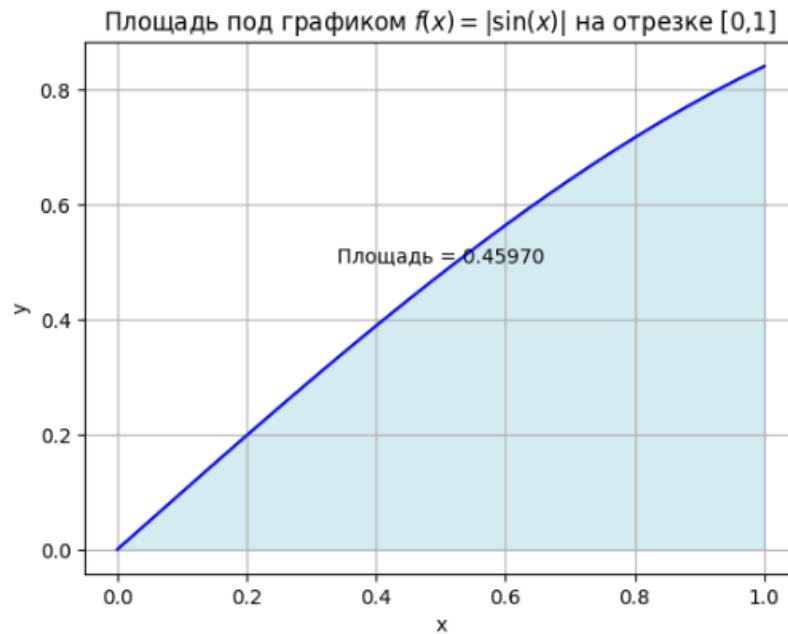


Рисунок 24 – Результат работы программы к индивидуальному заданию №3

#### ✓ Задачи на построение 3D-графиков с помощью Matplotlib

Во всех задачах требуется:

1. Построить трехмерный график функции  $f(x, y)$  в заданных пределах.
2. Использовать библиотеку Matplotlib для визуализации.
3. Оформить график: добавить заголовок, подписи осей и цветовую карту (если уместно).

#### Вариант 12. Двухгорбая поверхность (бигауссовский холм)

Постройте поверхность:  $f(x, y) = e^{((x-1)^2+y^2)} + e^{-((x+1)^2+y^2)}$  на  $x, y \in [-5, 5]$

#### Дополнительные требования

- Используйте `plot_surface()` для плотных поверхностей.
- Используйте `plot_wireframe()` для каркасных моделей.
- Настройте цветовую карту через `cm`.
- Добавьте подписи осей и заголовки.

```
[163] def f(x, y):
      return np.exp(-(x - 1)**2 - y**2) + np.exp(-((x + 1)**2) - y**2)
x = np.linspace(-5, 5)
y = np.linspace(-5, 5)
X, Y = np.meshgrid(x, y)
Z = f(X, Y)
fig = plt.figure(figsize=(8, 6))
ax = fig.add_subplot(111, projection='3d')
surf = ax.plot_surface(X, Y, Z, cmap='viridis')
ax.set_title('Двухгорбая поверхность (бигауссовский холм)')
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('f(x, y)')
```

Рисунок 25 – Индивидуальное задание №4. Задача на построение 3D-графиков с помощью Matplotlib

Двухгорбая поверхность (бигауссовский холм)

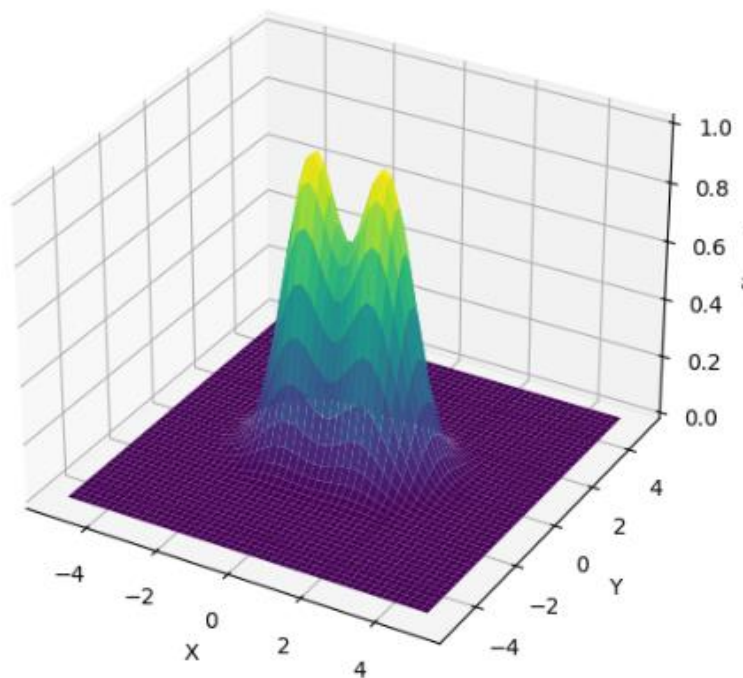


Рисунок 26 – Результат работы программы к индивидуальному заданию №4

#### **Контрольные вопросы:**

##### **1. Как осуществляется установка пакета matplotlib?**

Установка пакета matplotlib осуществляется с помощью команды `pip install matplotlib` в командной строке или терминале.

##### **2. Какая "магическая" команда должна присутствовать в ноутбуках Jupyter для корректного отображения графиков matplotlib?**

Для корректного отображения графиков в ноутбуках Jupyter используется магическая команда `%matplotlib inline`.

##### **3. Как отобразить график с помощью функции plot ?**

Для отображения графика с помощью функции `plot` нужно использовать команду `plt.plot(x, y)` и затем `plt.show()`.

##### **4. Как отобразить несколько графиков на одном поле?**

Чтобы отобразить несколько графиков на одном поле, можно использовать функцию `plt.subplot(rows, cols, index)` перед каждым графиком.

##### **5. Какой метод Вам известен для построения диаграмм категориальных данных?**

Для построения диаграмм категориальных данных используется метод `plt.bar()` для столбчатых диаграмм или `plt.boxplot()` для коробчатых диаграмм.

#### **6. Какие основные элементы графика Вам известны?**

Основные элементы графика включают оси (x, y), метки осей, легенду, заголовок, сетку и данные.

#### **7. Как осуществляется управление текстовыми надписями на графике?**

Для управления текстовыми надписями на графике используется метод `plt.text(x, y, 'text')` для добавления текста в указанные координаты.

#### **8. Как осуществляется управление легендой графика?**

Для управления легендой графика используется метод `plt.legend()`.

#### **9. Как задать цвет и стиль линий графика?**

Цвет и стиль линий графика задаются с помощью параметров, таких как `color`, `linestyle` в функции `plot()`, например, `plt.plot(x, y, color='r', linestyle='--')`.

#### **10. Как выполнить размещение графика в разных полях?**

Для размещения графиков в разных полях используется метод `plt.subplot()`.

#### **11. Как выполнить построение линейного графика с помощью matplotlib?**

Линейный график строится с помощью функции `plt.plot(x, y)`.

#### **12. Как выполнить заливку области между графиком и осью? Между двумя графиками?**

Заливка области между графиком и осью или между двумя графиками выполняется с помощью метода `plt.fill_between(x, y1, y2)`.

#### **13. Как выполнить выборочную заливку, которая удовлетворяет некоторому условию?**

Для выборочной заливки, которая удовлетворяет определенному условию, используется метод `plt.fill_between(x, y, condition)`.

#### **14. Как выполнить двухцветную заливку?**

Двухцветную заливку можно сделать с помощью `plt.fill_between(x, y1, y2, where=condition, color='color1', alpha=0.5)`.

### **15. Как выполнить маркировку графиков?**

Для маркировки графиков используются функции `plt.text()` или `plt.annotate()` для добавления меток на график.

### **16. Как выполнить обрезку графиков?**

Обрезка графиков осуществляется с помощью метода `plt.xlim()` и `plt.ylim()` для ограничения диапазонов осей.

### **17. Как построить ступенчатый график? В чем особенность ступенчатого графика?**

Ступенчатый график строится с помощью функции `plt.step(x, y)`, и его особенность в том, что линии соединяют данные ступенями.

### **18. Как построить стековый график? В чем особенность стекового графика?**

Стековый график строится с помощью функции `plt.stackplot(x, y)`, и его особенность в том, что области под графиком накладываются друг на друга.

### **19. Как построить stem-график? В чем особенность stem-графика?**

Стем-график строится с помощью `plt.stem(x, y)`, его особенность в том, что он отображает данные в виде вертикальных линий с маркерами на вершинах.

### **20. Как построить точечный график? В чем особенность точечного графика?**

Точечный график строится с помощью функции `plt.scatter(x, y)`, и его особенность в том, что отображает данные в виде точек.

### **21. Как осуществляется построение столбчатых диаграмм с помощью matplotlib?**

Столбчатая диаграмма строится с помощью `plt.bar(x, height)`.

### **22. Что такое групповая столбчатая диаграмма? Что такое столбчатая диаграмма с `errorbar` элементом?**

Групповая столбчатая диаграмма строится с использованием нескольких `plt.bar()` с раздвигом на оси `x`. Столбчатая диаграмма с элементом `errorbar` добавляется через `plt.errorbar()`.

### **23. Как выполнить построение круговой диаграммы средствами matplotlib?**

Круговую диаграмму можно построить с помощью метода `plt.pie(data)`

### **24. Что такое цветовая карта? Как осуществляется работа с цветовыми картами в matplotlib?**

Цветовая карта — это способ отображения данных через цвета. В `matplotlib` она используется с помощью функций типа `plt.imshow()` или `plt.contourf()`, где можно задавать различные схемы цветов.

### **25. Как отобразить изображение средствами matplotlib?**

Изображение отображается с помощью `plt.imshow(image)`.

### **26. Как отобразить тепловую карту средствами matplotlib?**

Тепловая карта отображается с помощью `plt.imshow(data, cmap='hot')`

### **27. Как выполнить построение линейного 3D-графика с помощью matplotlib?**

Для построения линейного 3D-графика используется `ax.plot(x, y, z)` с осью 3D.

### **28. Как выполнить построение точечного 3D-графика с помощью matplotlib?**

Для построения точечного 3D-графика используется `ax.scatter(x, y, z)` с осью 3D.

### **29. Как выполнить построение каркасной поверхности с помощью matplotlib?**

Каркасную поверхность можно построить с помощью `ax.plot_wireframe(X, Y, Z)`.

### **30. Как выполнить построение трехмерной поверхности с помощью matplotlib?**



Трехмерную поверхность можно построить с помощью `ax.plot_surface(X, Y, Z)` для 3D-графиков.

**Вывод:** в ходе практической работы мы исследовали базовые возможности библиотеки `matplotlib` языка программирования Python.