

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет»**

Кафедра инфокоммуникаций

**Отчёт по практическому занятию №3.4
«Основы работы с пакетом matplotlib»**

по дисциплине «Теории распознавания образов»

Выполнил студент группы ПИЖ-б-о-21-1
Коновалова В.Н. « » _____ 20__ г.
Подпись студента _____
Работа защищена « » _____ 20__ г.
Проверил Воронкин Р.А. _____
(подпись)

Ставрополь 2023

1. Создать общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT и выбранный Вами язык программирования (выбор языка программирования будет доступен после установки флажка Add .gitignore). Выполните клонирование созданного репозитория на рабочий компьютер. Организуйте свой репозиторий в соответствии с моделью ветвления git-flow. Дополните файл .gitignore необходимыми правилами для выбранного языка программирования, интерактивной оболочки Jupyter notebook и интегрированной среды разработки.

2. Проработать примеры лабораторной работы.

```
In [1]: import matplotlib.pyplot as plt
        %matplotlib inline
```

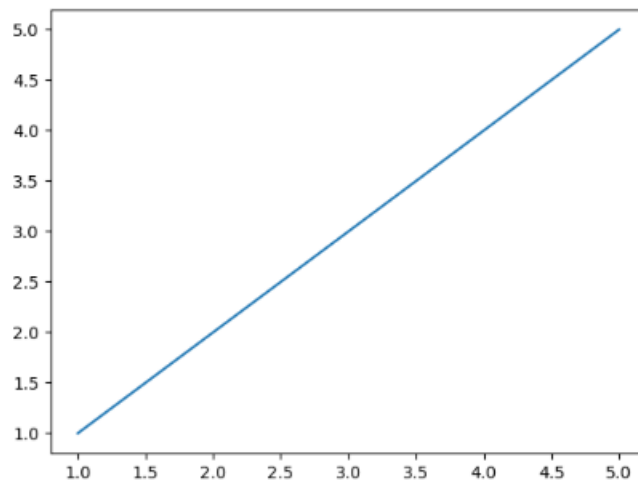
```
In [2]: import matplotlib
        matplotlib.__version__
```

```
Out[2]: '3.5.2'
```

Построение графика

```
In [3]: plt.plot([1, 2, 3, 4, 5], [1, 2, 3, 4, 5])
```

```
Out[3]: [<matplotlib.lines.Line2D at 0x1d969ad43a0>]
```

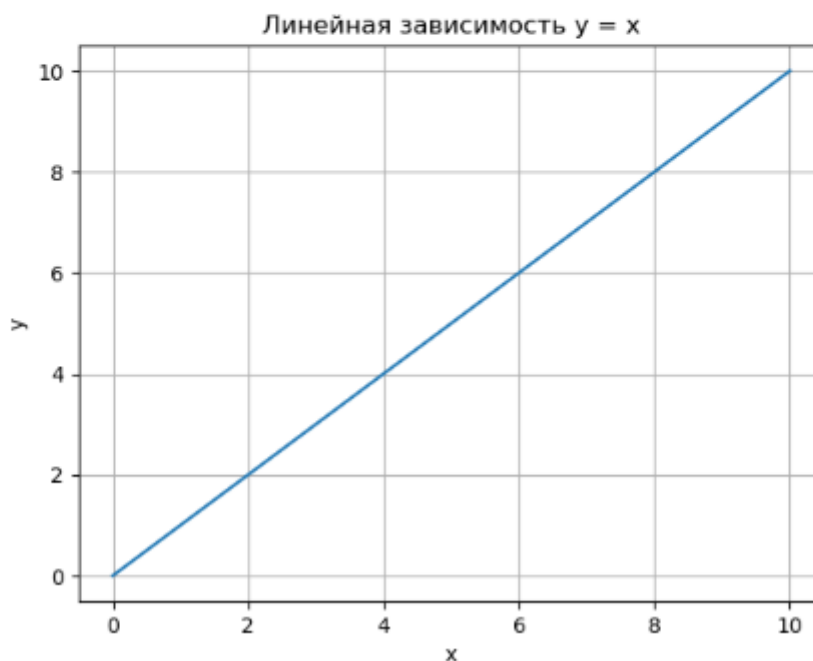


```
In [4]: import numpy as np

# Независимая (x) и зависимая (y) переменные
x = np.linspace(0, 10, 50)
y = x

# Построение графика
plt.title("Линейная зависимость y = x") # заголовок
plt.xlabel("x") # ось абсцисс
plt.ylabel("y") # ось ординат
plt.grid() # включение отображение сетки
plt.plot(x, y) # построение графика
```

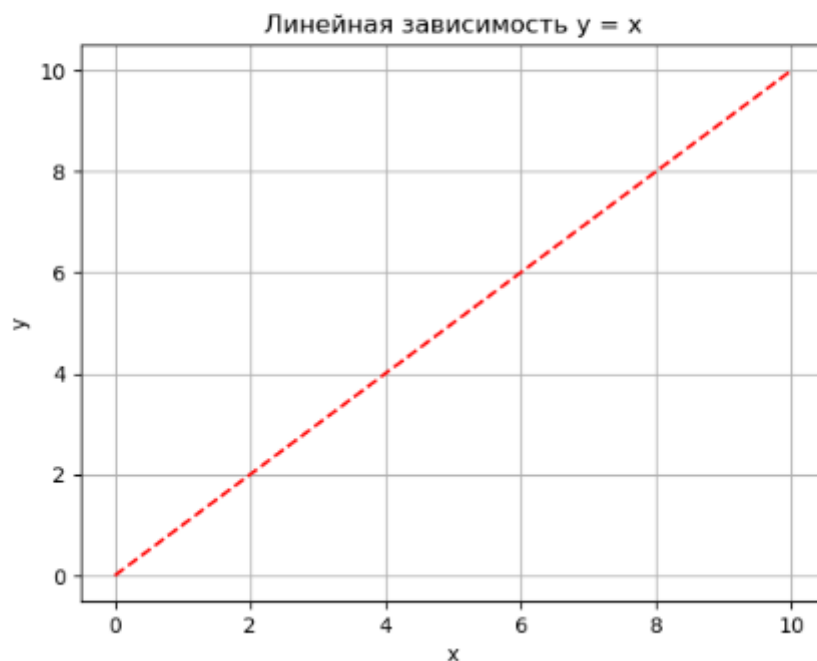
Out[4]: [



```
In [5]: # Построение графика
plt.title("Линейная зависимость y = x") # заголовок
plt.xlabel("x") # ось абсцисс
plt.ylabel("y") # ось ординат
plt.grid() # включение отображение сетки

plt.plot(x, y, "r--") # построение графика
```

Out[5]: [



Вопросы для защиты работы

1. Как осуществляется установка пакета matplotlib?

Существует два основных варианта установки этой библиотеки: в первом случае вы устанавливаете пакет Anaconda, в состав которого входит большое количество различных инструментов для работы в области машинного обучения и анализа данных (и не только); во втором – установить Matplotlib самостоятельно, используя менеджер пакетов.

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

```
%matplotlib inline
```

3. Как отобразить график с помощью функции plot?

Для построения графика используется команда plot(). Если в качестве параметра функции plot() передать список, то значения из этого списка будут отложены по оси ординат (ось y), а по оси абсцисс (ось x) будут отложены индексы элементов массива.

Для того, чтобы задать значения по осям x и y необходимо в plot() передать два списка.

```
plt.plot([1, 2, 3, 4, 5], [1, 2, 3, 4, 5])
```

4. Как отобразить несколько графиков на одном поле?

Для того, чтобы вывести несколько графиков на одном поле необходимо передать соответствующие наборы значений в функцию plot().

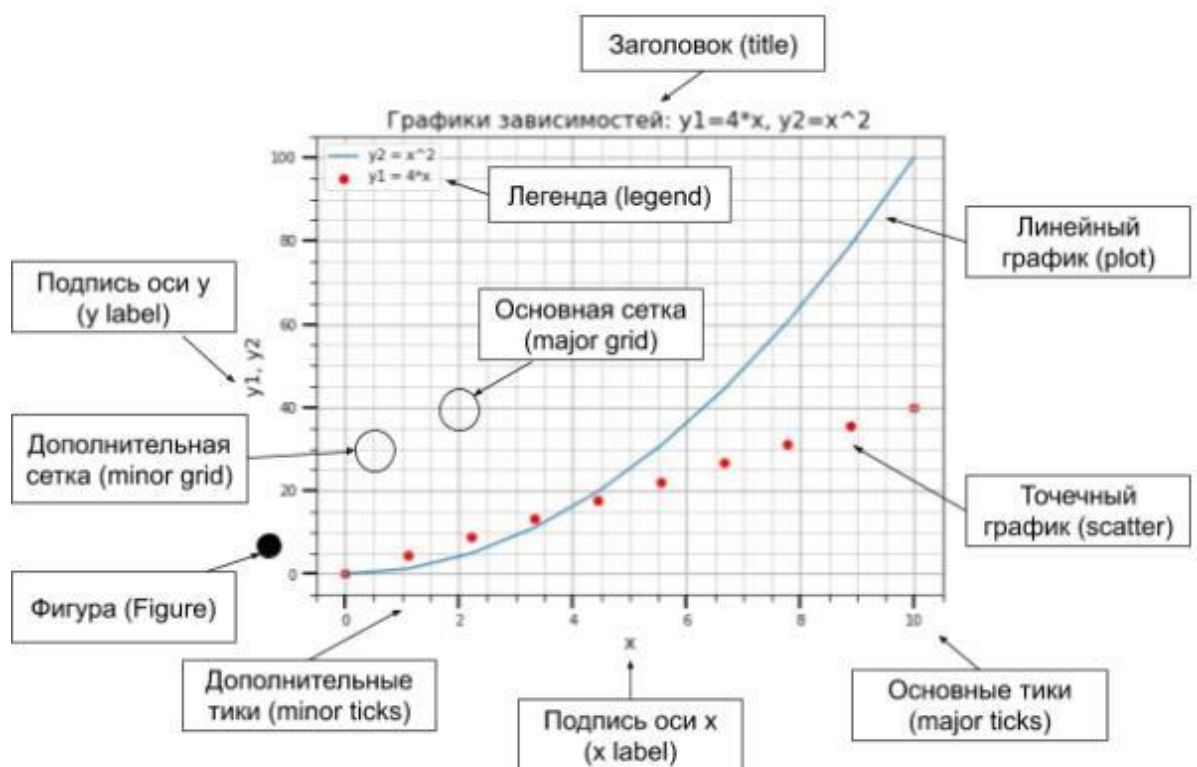
```
plt.plot(x, y1, x, y2)
```

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

Метод bar()

6. Какие основные элементы графика Вам известны?

- Заголовок (title)
- Легенда (legend)
- Основная сетка (major grid)
- Линейный график (plot)
- Точечный график (scatter)
- Дополнительные тики (minor ticks) Фигура (figure)
- Дополнительная сетка (minor grid) Подпись оси y (y label)
- Основные тики (major ticks) Подпись оси x (x label)



7. Как осуществляется управление текстовыми надписями на графике?

Наименование осей: `plt.xlabel()`, `plt.ylabel()`

Заголовок графика: `plt.title()`

Текстовое примечание: `plt.text()`

Легенда: `plt.legend()`

8. Как осуществляется управление легендой графика?

Легенда будет размещена на графике, если вызвать функцию `legend()`

9. Как задать цвет и стиль линий графика?

Задание цвета: `plt.plot(x, y, color='red')`, `plt.setp(color='red', linewidth=1)`

Задание цвета линии графика производится через параметр `color` (или `c`, если использовать сокращенный вариант). Значение может быть представлено в одном из следующих форматов:

- RGB или RGBA кортеж значений с плавающей точкой в диапазоне [0, 1] (пример: (0.1, 0.2, 0.3))
- RGB или RGBA значение в hex формате (пример: '#0a0a0a')
- строковое представление числа с плавающей точкой в диапазоне [0, 1] (определяет цвет в шкале серого) (пример: '0.7')
- символ из набора {'b', 'g', 'r', 'c', 'm', 'y', 'k', 'w'}
- имя цвета из палитры X11/CSS4
- цвет из палитры `xkcd` (<https://xkcd.com/color/rgb/>), должен начинаться с префикса 'xkcd:'
- цвет из набора Tableau Color (палитра T10), должен начинаться с префикса 'tab:'

Если цвет задается с помощью символа из набора {'b', 'g', 'r', 'c', 'm', 'y', 'k', 'w'}, то он может быть совмещен со стилем линии в рамках параметра `fmt` функции `plot()`. Например штриховая красная линия будет задаваться так: '-r', а штрих пунктирная зеленая так '-.g'

Задание стиля линии: `plt.plot(x, y, '--')`

10. Как выполнить размещение графика в разных полях?

Существуют три основных подхода к размещению нескольких графиков на разных полях:

- использование функции `subplot()` для указания места размещения поля с графиком;

- использование функции `subplots()` для предварительного задания сетки, в которую будут укладываться поля;
- использование `GridSpec`, для более гибкого задания геометрии размещения полей с графиками в сетке.

Самый простой способ представить графики в отдельных полях – это использовать функцию `subplot()` для задания их мест размещения.

После задания размера, указывается местоположение, куда будет установлено поле с графиком с помощью функции `subplot()`. Чаще всего используют следующие варианты вызова `subplot`:

`subplot(nrows, ncols, index)`

`nrows (int)` – количество строк.

`ncols (int)` – количество столбцов.

`index(int)` – местоположение элемента

`subplot(pos)`

`pos (int)` – позиция, в виде трехзначного числа, содержащего информацию о количестве строк, столбцов и индексе, например 212, означает подготовить разметку с двумя строками и одним столбцов, элемент вывести в первую позицию второй строки. Этот вариант можно использовать, если количество строк и столбцов сетки не более 10, в ином случае лучше обратиться к первому варианту.