

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ
АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
ИНСТИТУТ ЦИФРОВОГО РАЗВИТИЯ**

**Отчет по лабораторной работе №4
«Основы работы с пакетом matplotlib»**

по дисциплине «Технологии распознавания образов»

Выполнила:
Первых Дарья Александровна,
2 курс, группа ПИЖ-б-о-20-1,

Проверил:
Доцент кафедры инфокоммуникаций,
Воронкин Р.А.

Ставрополь, 2022 г.

ВЫПОЛНЕНИЕ

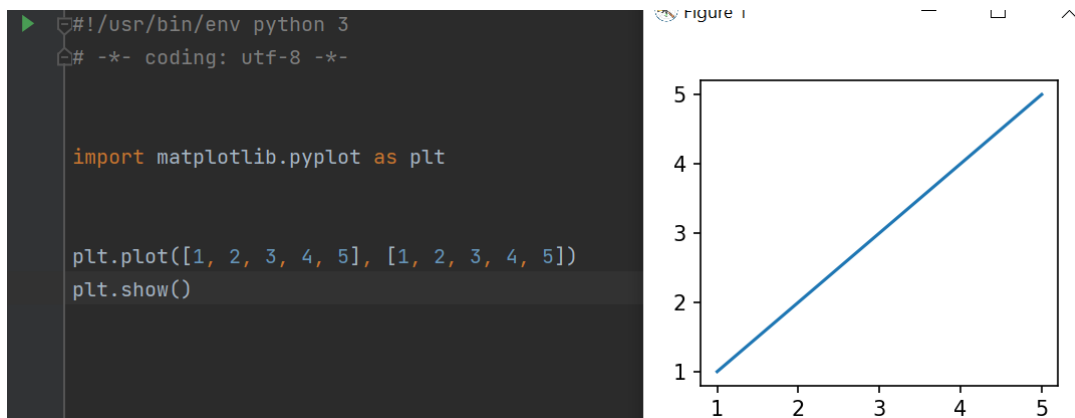


Рисунок 1 – Пример задания графика

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

In [2]: 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[2]: [matplotlib.lines.Line2D at 0x23044ad13d0>]
```

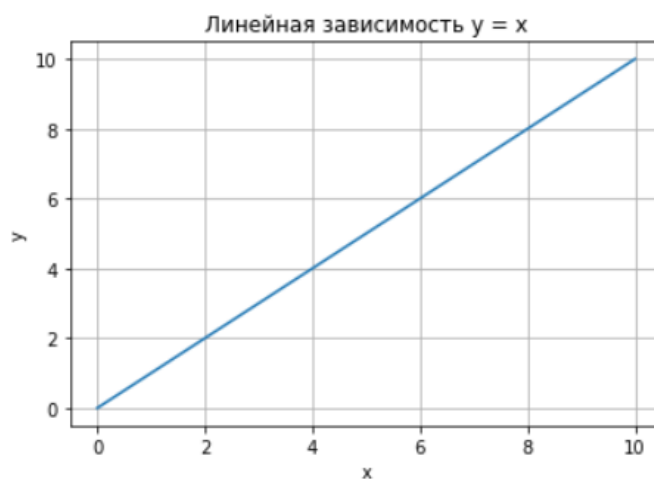


Рисунок 2 – Пример построения простой линейной зависимости

```
In [3]: # 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[3]: [

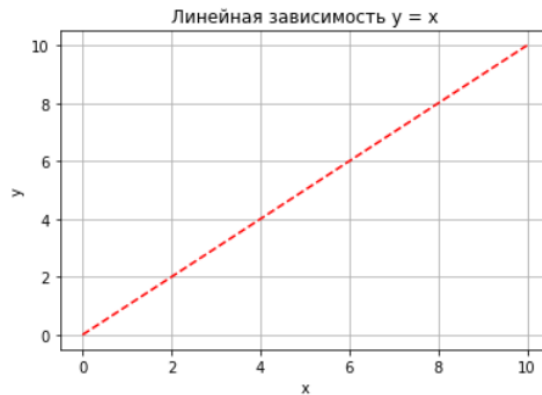


Рисунок 3 – Пример задания графика красной пунктирной линией

```
In [4]: # Линейная зависимость
x = np.linspace(0, 10, 50)
y1 = x
# Квадратичная зависимость
y2 = [i**2 for i in x]
# Построение графика
plt.title("Зависимости:  $y1 = x$ ,  $y2 = x^2$ ") # заголовок
plt.xlabel("x") # ось абсцисс
plt.ylabel("y1, y2") # ось ординат
plt.grid() # включение отображение сетки
plt.plot(x, y1, x, y2) # построение графика
```

Out[4]: [<matplotlib.lines.Line2D at 0x23044c21dc0>]

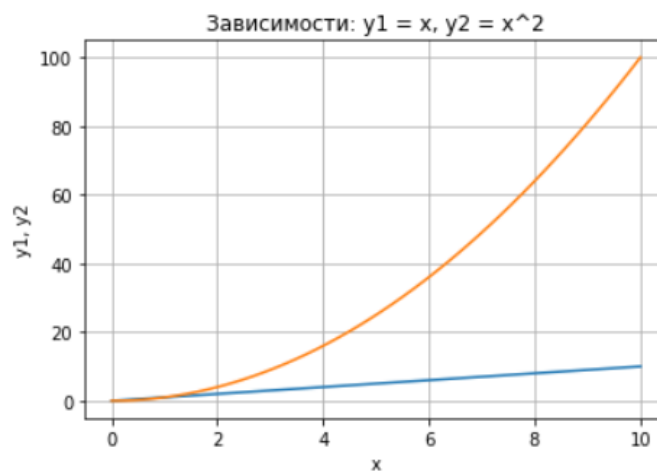


Рисунок 4 – Пример задания нескольких графиков на одном поле

```
plt.grid(True) # включение отображения сетки
plt.subplot(2, 1, 2)
plt.plot(x, y2) # построение графика
plt.xlabel("x", fontsize=14) # ось абсцисс
plt.ylabel("y2", fontsize=14) # ось ординат
plt.grid(True) # включение отображения сетки
```

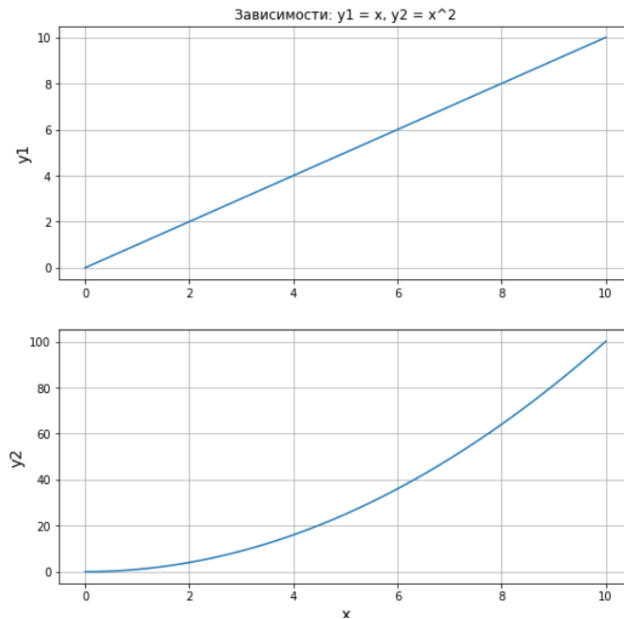


Рисунок 5 – Пример задания графика на нескольких разделённых полях

```
In [6]: fruits = ["apple", "peach", "orange", "bannana", "melon"]
counts = [34, 25, 43, 31, 17]
plt.bar(fruits, counts)
plt.title("Fruits!")
plt.xlabel("Fruit")
plt.ylabel("Count")
```

Out[6]: Text(0, 0.5, 'Count')

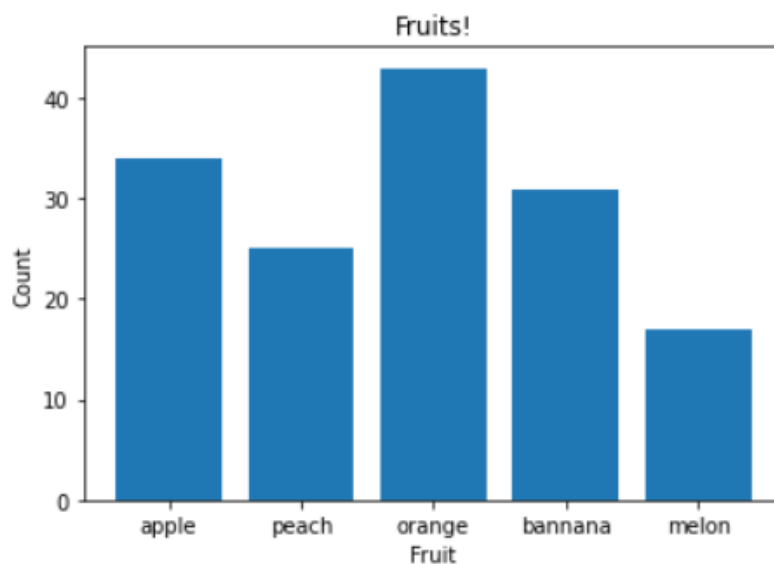


Рисунок 6 – Пример построения диаграммы для категориальных данных

```

ax.set_title("Графики зависимостей: y1=4*x, y2=x^2", fontsize=16)
ax.set_xlabel("x", fontsize=14)
ax.set_ylabel("y1, y2", fontsize=14)
ax.grid(which="major", linewidth=1.2)
ax.grid(which="minor", linestyle="--", color="gray", linewidth=0.5)
ax.scatter(x, y1, c="red", label="y1 = 4*x")
ax.plot(x, y2, label="y2 = x^2")
ax.legend()
ax.xaxis.set_minor_locator(AutoMinorLocator())
ax.yaxis.set_minor_locator(AutoMinorLocator())
ax.tick_params(which='major', length=10, width=2)
ax.tick_params(which='minor', length=5, width=1)
plt.show()

```

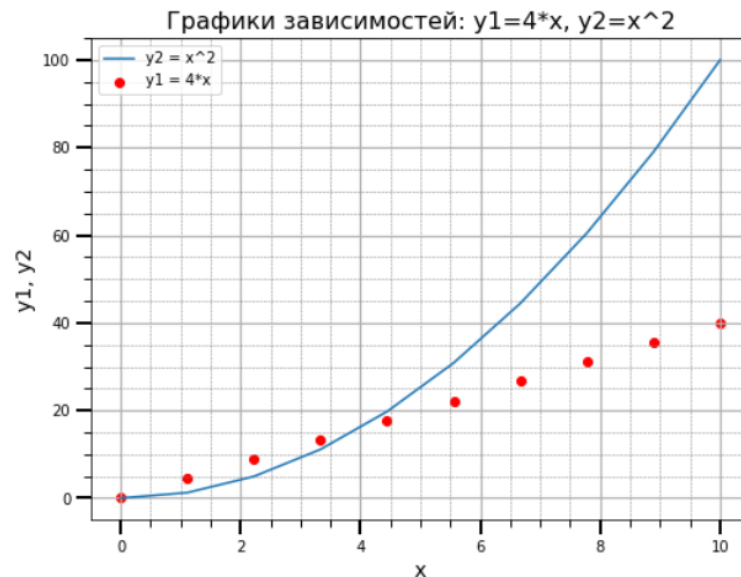


Рисунок 7 – Пример построения графика с основными элементами

```

In [8]: x = [1, 5, 10, 15, 20]
y = [1, 7, 3, 5, 11]
plt.plot(x, y, label='steel price')
plt.title('Chart price', fontsize=15)
plt.xlabel('Day', fontsize=12, color='blue')
plt.ylabel('Price', fontsize=12, color='blue')
plt.legend()
plt.grid(True)
plt.text(15, 4, 'grow up!')

```

Out[8]: Text(15, 4, 'grow up!')

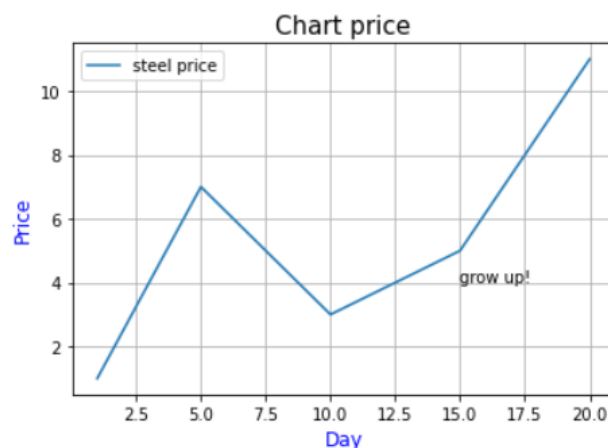
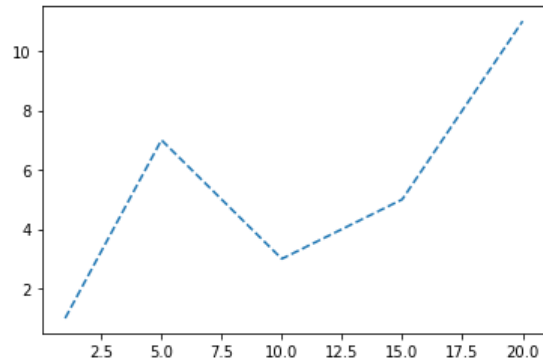


Рисунок 8 – Пример текстовых надписей на графике

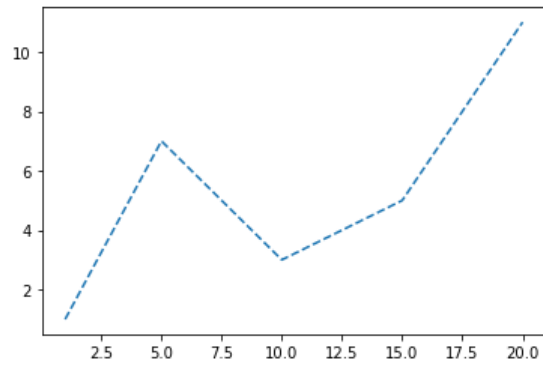
```
In [9]: x = [1, 5, 10, 15, 20]
        y = [1, 7, 3, 5, 11]
        plt.plot(x, y, '--')
```

Out[9]: [



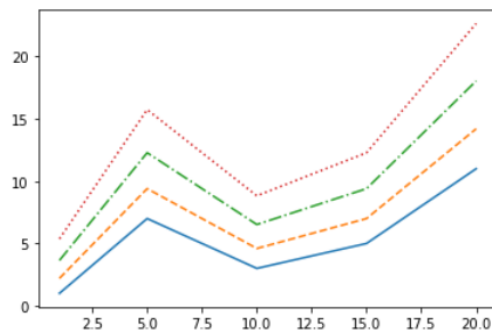
```
In [10]: x = [1, 5, 10, 15, 20]
          y = [1, 7, 3, 5, 11]
          line = plt.plot(x, y)
          plt.setp(line, linestyle='--')
```

Out[10]: [None]



```
In [11]: x = [1, 5, 10, 15, 20]
          y1 = [1, 7, 3, 5, 11]
          y2 = [i*1.2 + 1 for i in y1]
          y3 = [i*1.2 + 1 for i in y2]
          y4 = [i*1.2 + 1 for i in y3]
          plt.plot(x, y1, '-', x, y2, '--', x, y3, '-.', x, y4, ':')
```

Out[11]: [<matplotlib.lines.Line2D at 0x230450f3c40>,
<matplotlib.lines.Line2D at 0x230450f3af0>,
<matplotlib.lines.Line2D at 0x230450f3c70>]



```
In [12]: plt.plot(x, y1, '-')
plt.plot(x, y2, '--')
plt.plot(x, y3, '-.')
plt.plot(x, y4, ':')
```

```
Out[12]: [<matplotlib.lines.Line2D at 0x230451637f0>]
```

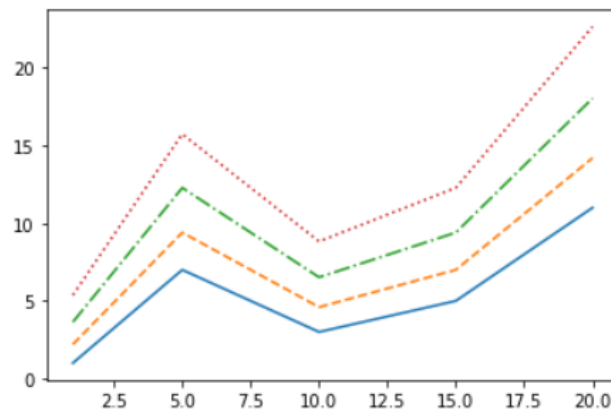


Рисунок 9 – Пример работы с линейным графиком

```
In [13]: x = [1, 5, 10, 15, 20]
y = [1, 7, 3, 5, 11]
plt.plot(x, y, '--r')
```

```
Out[13]: [<matplotlib.lines.Line2D at 0x23044d42130>]
```

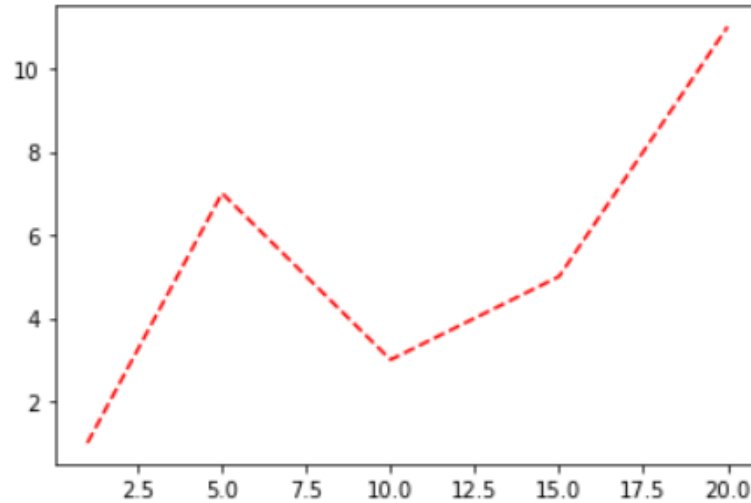
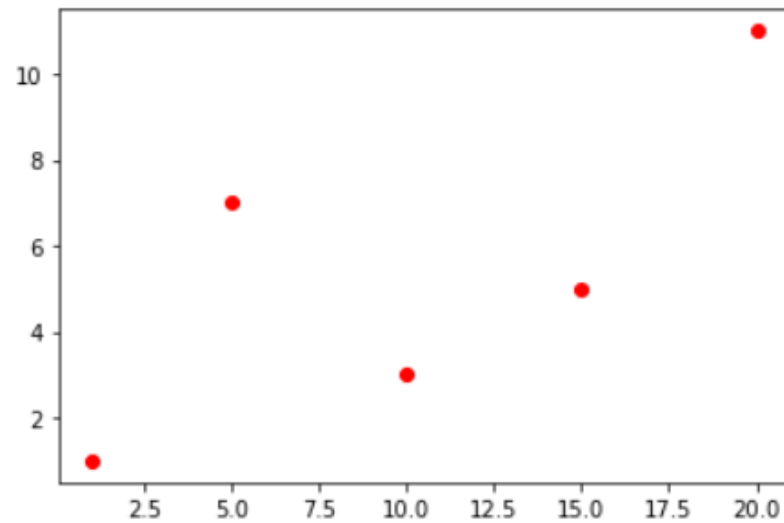


Рисунок 9 – Пример работы с цветом линии

```
In [14]: plt.plot(x, y, 'ro')
```

```
Out[14]: [<matplotlib.lines.Line2D at 0x23044e76130>]
```



```
In [15]: plt.plot(x, y, 'bx')
```

```
Out[15]: [<matplotlib.lines.Line2D at 0x23044cbef40>]
```

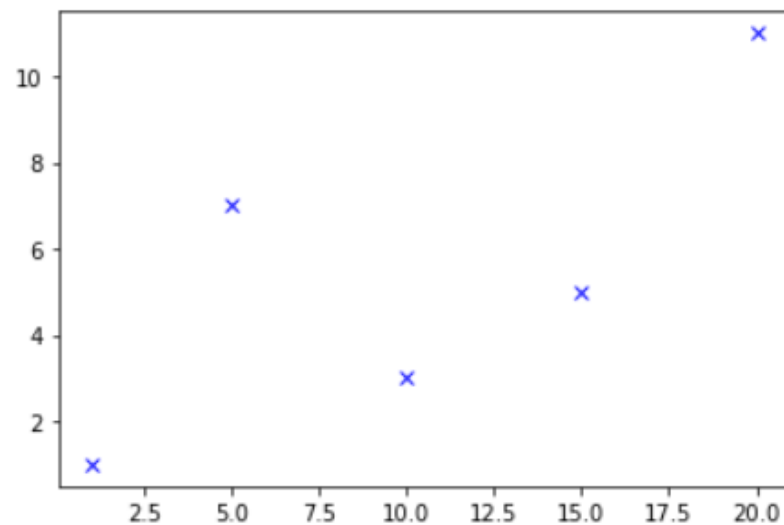
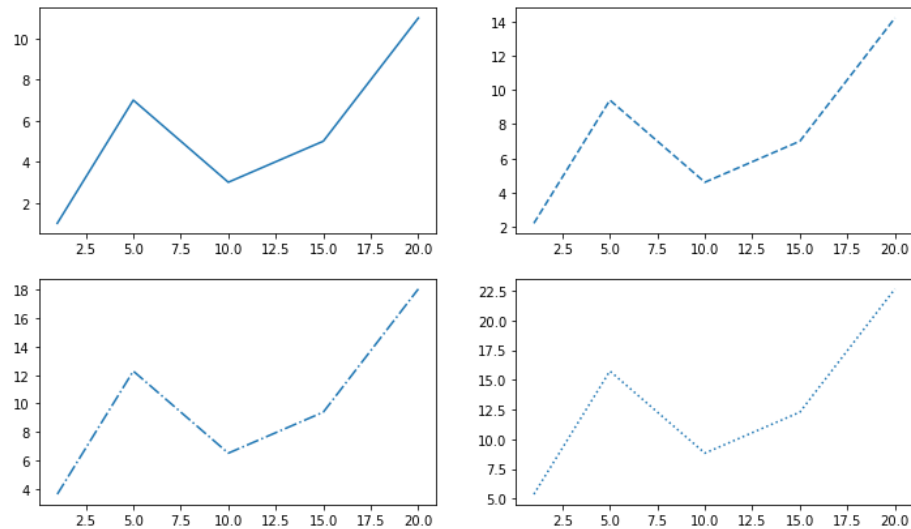


Рисунок 10 – Пример работы с типом графика


```
In [16]: # Исходный набор данных
x = [1, 5, 10, 15, 20]
y1 = [1, 7, 3, 5, 11]
y2 = [i*1.2 + 1 for i in y1]
y3 = [i*1.2 + 1 for i in y2]
y4 = [i*1.2 + 1 for i in y3]
# Настройка размеров подложки
plt.figure(figsize=(12, 7))
# Вывод графиков
plt.subplot(2, 2, 1)
plt.plot(x, y1, '-')
plt.subplot(2, 2, 2)
plt.plot(x, y2, '--')
plt.subplot(2, 2, 3)
plt.plot(x, y3, '-.')
plt.subplot(2, 2, 4)
plt.plot(x, y4, ':')
```

Out[16]: [



```
In [17]: # Вывод графиков
plt.subplot(221)
plt.plot(x, y1, '-')
plt.subplot(222)
plt.plot(x, y2, '--')
plt.subplot(223)
plt.plot(x, y3, '-.')
plt.subplot(224)
plt.plot(x, y4, ':')
```

Out[17]: [

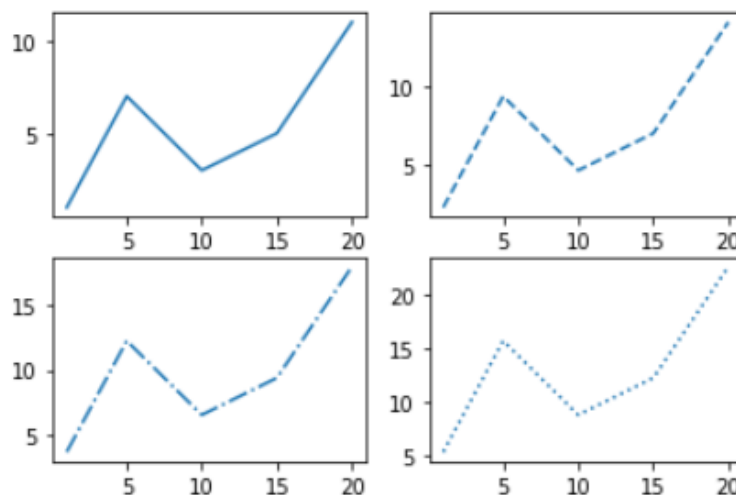


Рисунок 11 – Пример работы с функцией subplot()

```
In [18]: fig, axs = plt.subplots(2, 2, figsize=(12, 7))
axs[0, 0].plot(x, y1, '-')
axs[0, 1].plot(x, y2, '-')
axs[1, 0].plot(x, y3, '-')
axs[1, 1].plot(x, y4, ':')
```

```
Out[18]: [<matplotlib.lines.Line2D at 0x23046511f10>]
```

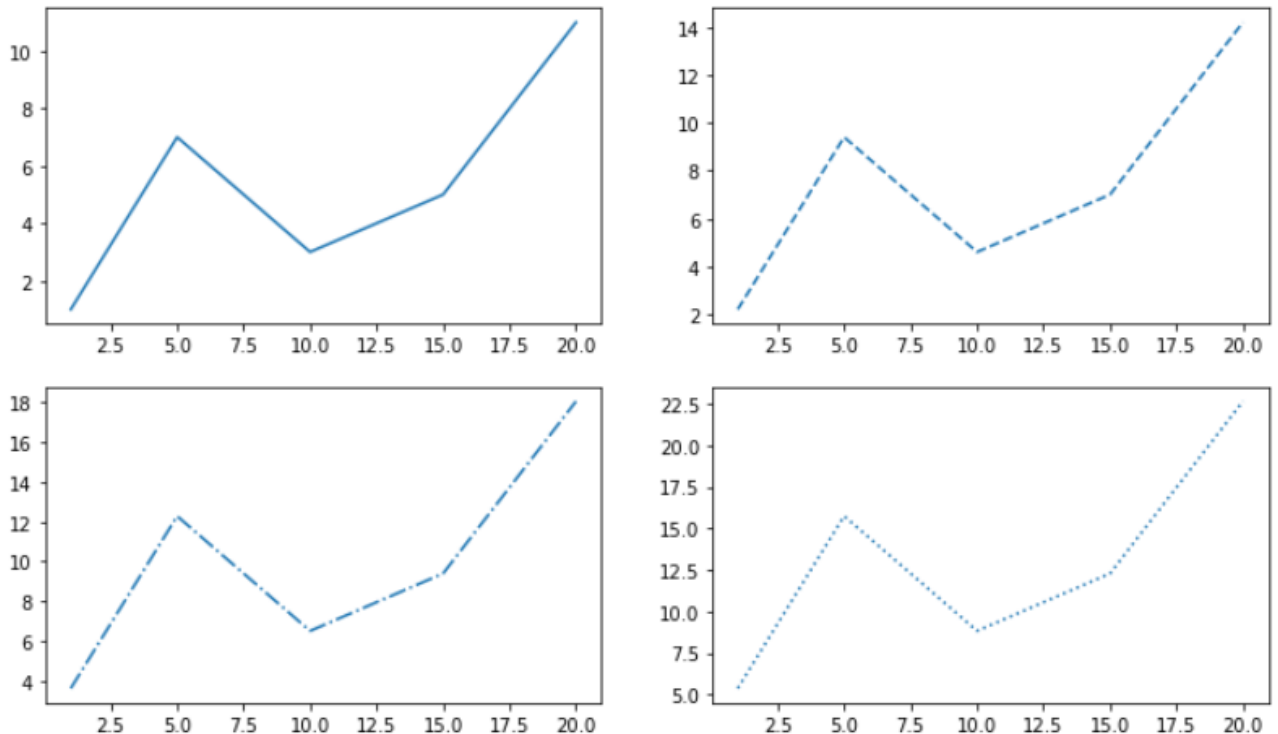


Рисунок 12 – Пример работы с функцией subplots()

ВОПРОСЫ

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.

```
plt.plot([1, 7, 3, 5, 11, 1])
```

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

```
plt.plot([1, 5, 10, 15, 20], [1, 7, 3, 5, 11])
```

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(). Построим несколько наборов данных и выведем их с использованием различных стилей линии.

```
x = [1, 5, 10, 15, 20]
y1 = [1, 7, 3, 5, 11]
y2 = [i*1.2 + 1 for i in y1]
y3 = [i*1.2 + 1 for i in y2]
y4 = [i*1.2 + 1 for i in y3]

plt.plot(x, y1, '-', x, y2, '--', x, y3, '-.', x, y4, ':')
```

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)

Основные тики (major ticks)

Подпись оси x (x label)

Дополнительные тики (minor ticks)

Фигура (figure)

Дополнительная сетка (minor grid)

Подпись оси y (y label)

7. Как осуществляется управление текстовыми надписями на графике?

Наименование осей:

```
plt.xlabel('Day', fontsize=15, color='blue')
```

Заголовок графика:

```
plt.title('Chart price', fontsize=17)
```

Текстовое примечание:

```
plt.text(1, 1, 'type: Steel')
```

Легенда:

```
plt.legend()
```

8. Как осуществляется управление легендой графика?

```
plt.legend()
```

9. Как задать цвет и стиль линий графика?

Задание цвета:

```
plt.plot(x, y, color='red')
```

```
plt.setp( color='red', linewidth=1)
```

Задание стиля линии:

```
plt.plot(x, y, '--')
```

```
line = plt.plot(x, y)  
plt.setp(line, linestyle='--')
```

10. Как выполнить размещение графика в разных полях?

Способ представления графика в отдельных полях – это использование функции `subplot()` для задания их мест размещения.

```
# Исходный набор данных  
x = [1, 5, 10, 15, 20]  
y1 = [1, 7, 3, 5, 11]  
y2 = [i*1.2 + 1 for i in y1]  
y3 = [i*1.2 + 1 for i in y2]  
y4 = [i*1.2 + 1 for i in y3]  
  
# Настройка размеров подложки  
plt.figure(figsize=(12, 7))  
  
# Вывод графиков  
plt.subplot(2, 2, 1)  
plt.plot(x, y1, '-')  
  
plt.subplot(2, 2, 2)  
plt.plot(x, y2, '--')  
  
plt.subplot(2, 2, 3)  
plt.plot(x, y3, '-.')  
  
plt.subplot(2, 2, 4)  
plt.plot(x, y4, ':')
```