

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

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

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

по дисциплине «Технологии распознавания образов»

Выполнил студент группы ПИЖ-б-о-20-1

Бокань И.П. « » _____ 2022г.

Подпись студента _____

Работа защищена « » _____ 2022г.

Проверил Воронкин Р.А. _____
(подпись)

Ставрополь 2022

1. Вывод (примеры)

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

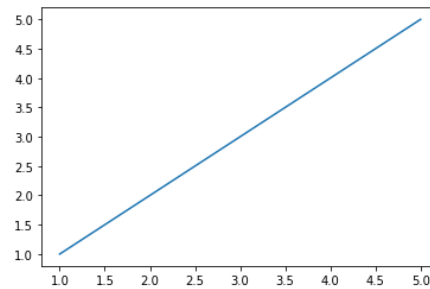


Рисунок 1.1 - Результаты примера простое построение графика

```
# 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  
plt.show()
```

```
# 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  
plt.show()
```

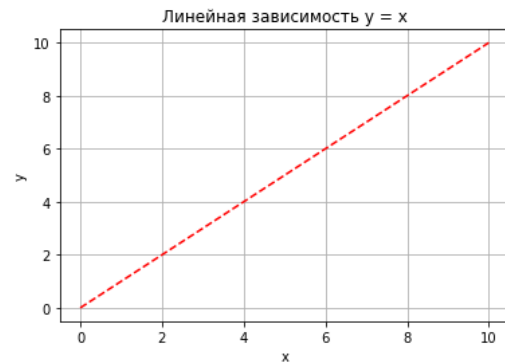
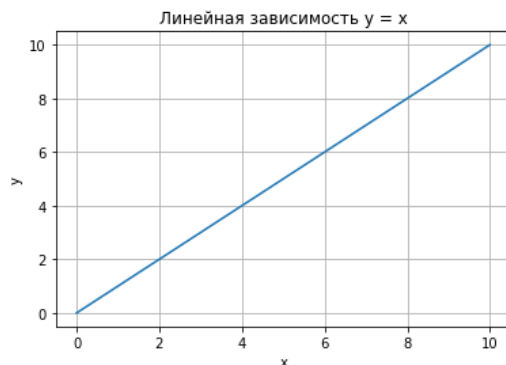


Рисунок 1.2 - Результаты примера линейное построение графика

```
# 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

plt.show()
```

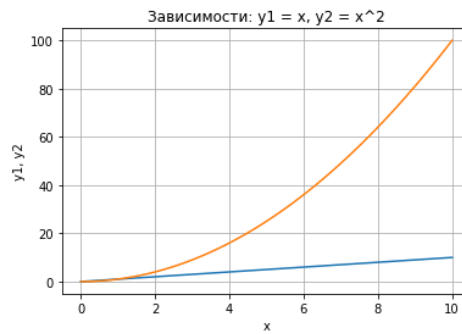


Рисунок 1.3 - Результаты примера несколько графиков на одном поле

```
# Линейная зависимость
x = np.linspace(0, 10, 50)
y1 = x

# Квадратичная зависимость
y2 = [i**2 for i in x]

# Построение графиков
plt.figure(figsize=(9, 9))

plt.subplot(2, 2, 1)
plt.plot(x, y1) # построение графика

plt.title("Зависимости: y1 = x, y2 = x^2") # заголовок
plt.ylabel("y1", fontsize=14) # ось ординат
plt.grid(True) # включение отображение сетки

plt.subplot(2, 2, 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 отображение сетки

plt.show()
```

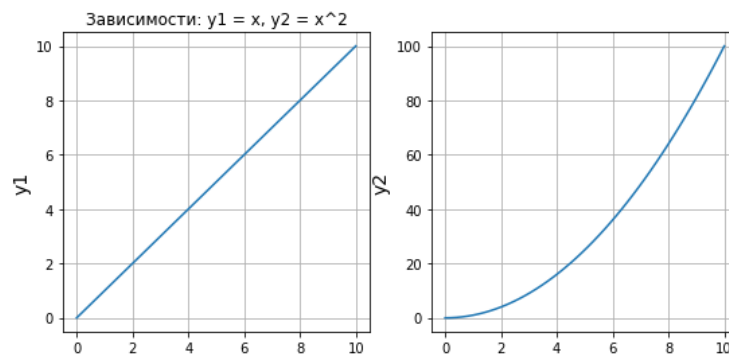


Рисунок 1.4 - Результаты примера несколько разделенных полей с графиками

```
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")

plt.show()
```

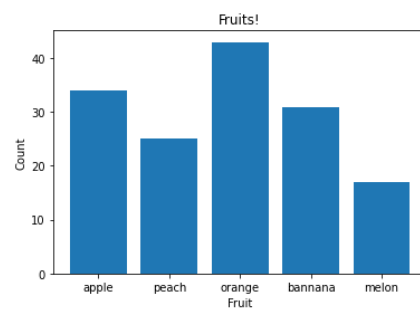


Рисунок 1.5 - Результаты примера построение диаграммы для категориальных данных

```

x = np.linspace(0, 10, 10)
y1 = 4*x*x
y2 = [(i**2)-i for i in x]

fig, ax = plt.subplots(figsize=(8, 6))

# Заголовок(title)
ax.set_title("Графики зависимостей: y1=4*x, y2=x^2", fontsize=16)

# Подпись оси (x Label)
ax.set_xlabel("x", fontsize=14)

# Подпись оси (y Label)
ax.set_ylabel("y1, y2", fontsize=14)

# Основная сетка (major grid)
ax.grid(which="major", linewidth=1.2)

# Дополнительная сетка (minor grid)
ax.grid(which="minor", linestyle="--", color="gray", linewidth=0.5)

# Точечный график (scatter)
ax.scatter(x, y1, c="red", label="y1 = 4*x")

# Линейный график (plot)
ax.plot(x, y2, label="y2 = x^2")

# Легенда (Legend)
ax.legend()

ax.xaxis.set_minor_locator(AutoMinorLocator())
ax.yaxis.set_minor_locator(AutoMinorLocator())

# Основные тики (major ticks)
ax.tick_params(which='major', length=10, width=2)

# Дополнительные тики (minor ticks)
ax.tick_params(which='minor', length=5, width=1)

# Вывод на экран
plt.show()

```

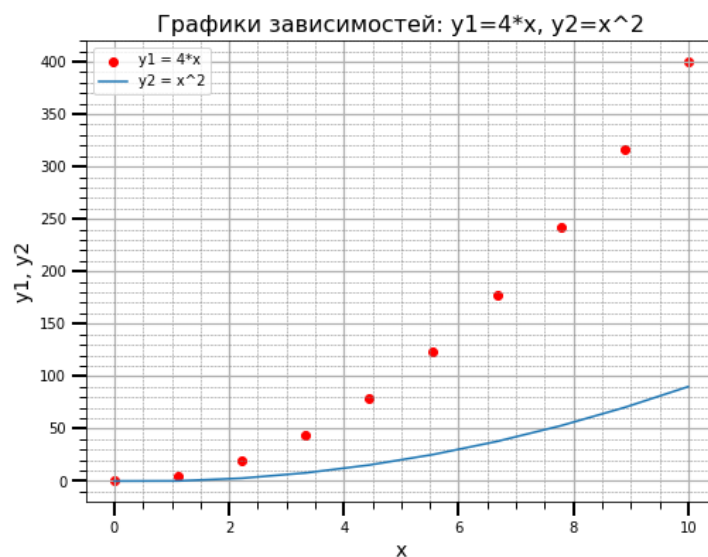


Рисунок 1.6 - Результаты примера построение диаграммы для категориальных данных

```
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]
y5 = [i*1.2 + 1 for i in y4]

# где x, y5 (нет видимость линия)
plt.plot(x, y1, '-r', x, y2, '--g', x, y3, '-.b', x, y4, ':y', x, y5, 'None')
plt.show()
```

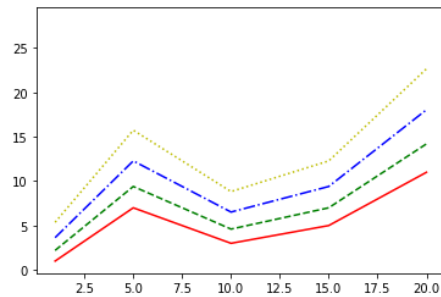


Рисунок 1.7 - Результаты примера стиль линии графика и Цвет линии

```
plt.plot(x, y, 'ro')
plt.show()

plt.plot(x, y, 'bx')
plt.show()
```

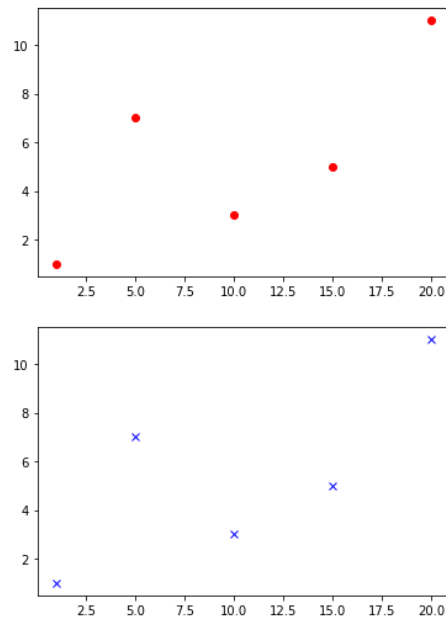


Рисунок 1.8 - Результаты примера тип графика

```
# Исходный набор данных
x = [1, 5, 10, 15, 20]
y = np.array([1, 2, 3, 4, 5])
y1 = (np.sin(y ** 2)).tolist()
y2 = (np.cos(y ** 3)).tolist()
y3 = (np.tan(y ** 2)).tolist()
y4 = (np.arctan(y ** 3)).tolist()

# Настройка размеров подложки
plt.figure(figsize=(12, 7))

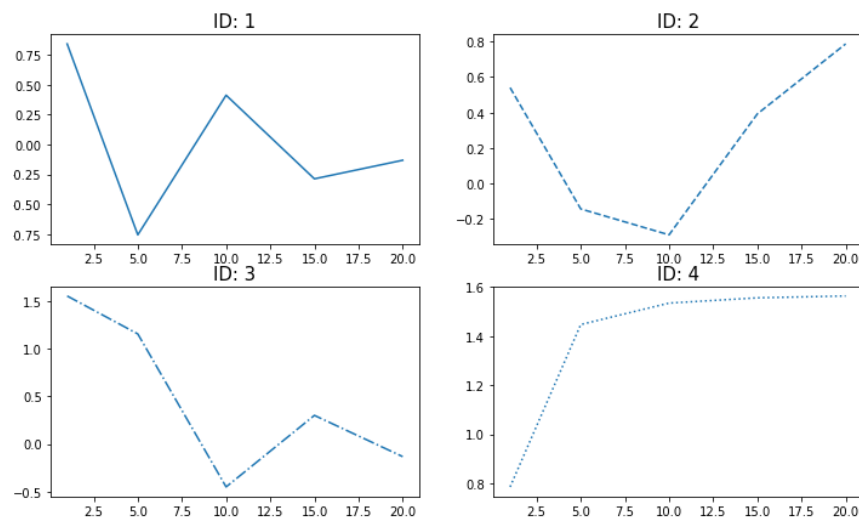
# Вывод графиков
plt.subplot(2, 2, 1)
plt.plot(x, y1, '-')
plt.title('ID: 1', fontsize=15)

plt.subplot(2, 2, 2)
plt.plot(x, y2, '--')
plt.title('ID: 2', fontsize=15)

plt.subplot(2, 2, 3)
plt.plot(x, y3, '-.')
plt.title('ID: 3', fontsize=15)

plt.subplot(2, 2, 4)
plt.plot(x, y4, ':')
plt.title('ID: 4', fontsize=15)

plt.show()
```



```
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, ':')

plt.show()
```

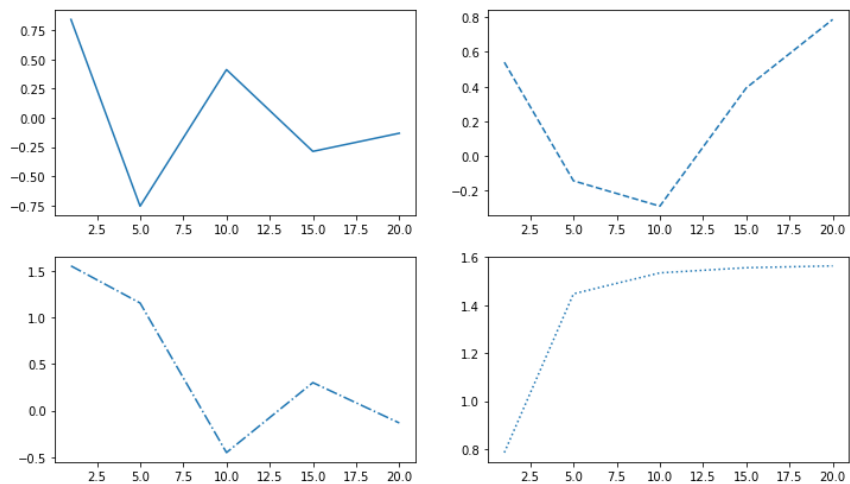


Рисунок 1.9 - Результаты примера работа с функцией subplot