

MATPLOTLIB

1. Базовый

1. `plot(x, y)`

код:

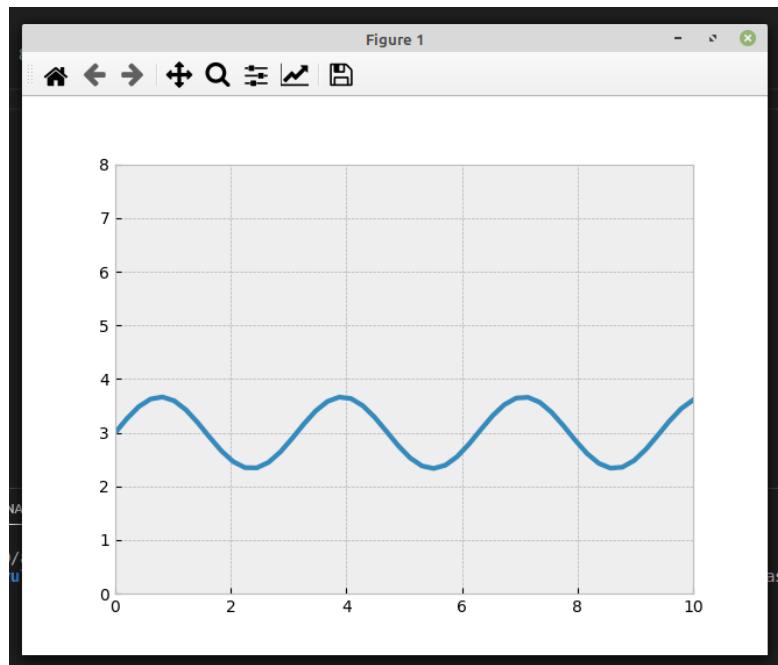
```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('bmh')
x = np.linspace(0, 10)
y = 3 + 2/3 * np.sin(2 * x)
fig, ax = plt.subplots()

ax.plot(x, y, linewidth=3.0)
ax.set(xlim=(0, 10), ylim=(0, 8))

plt.show()
```

результат:



2. scatter(x, y)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('bmh')

np.random.seed(3)
x = 4 + np.random.normal(0, 2, 24)
y = 4 + np.random.normal(0, 2, len(x))

sizes = np.random.uniform(15, 80, len(x))
colors = np.random.uniform(15, 80, len(x))

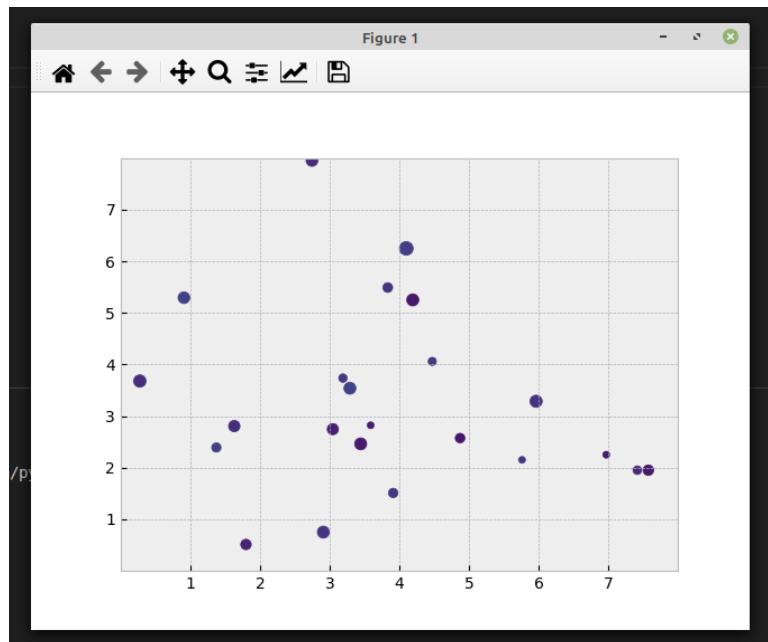
fig, ax = plt.subplots()

ax.scatter(x, y, s=sizes, c=colors, vmin=0, vmax=390)

ax.set(xlim=(0, 8), xticks=np.arange(1, 8),
       ylim=(0, 8), yticks=np.arange(1, 8))

plt.show()
```

результатом:



3. bar (x, y)

Код:

```
import matplotlib.pyplot as plt
import numpy as np
plt.style.use('default')

np.random.seed(6)
x = 0.5 + np.arange(8)
y = np.random.uniform(2, 7, len(x))

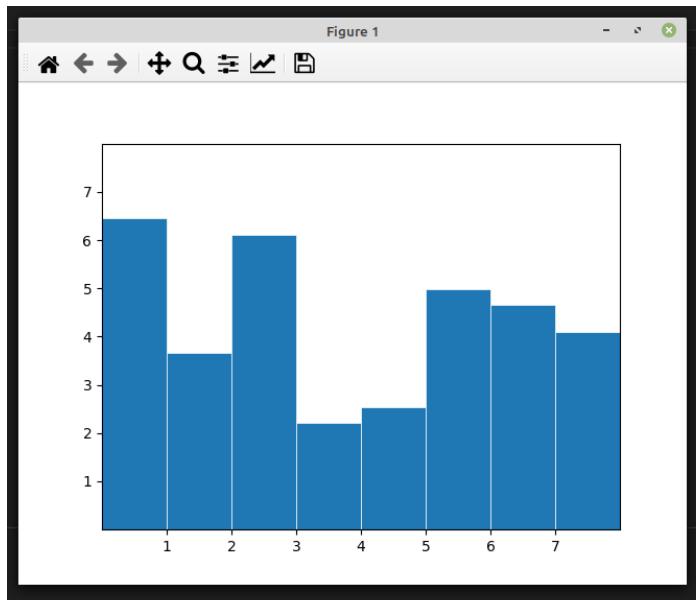
fig, ax = plt.subplots()

ax.bar(x, y, width=1, edgecolor="white", linewidth=0.5)

ax.set(xlim=(0, 8), xticks=np.arange(1, 8),
       ylim=(0, 8), yticks=np.arange(1, 8))

plt.show()
```

результат:



4. stem (x, y)

Код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('classic')
```

```

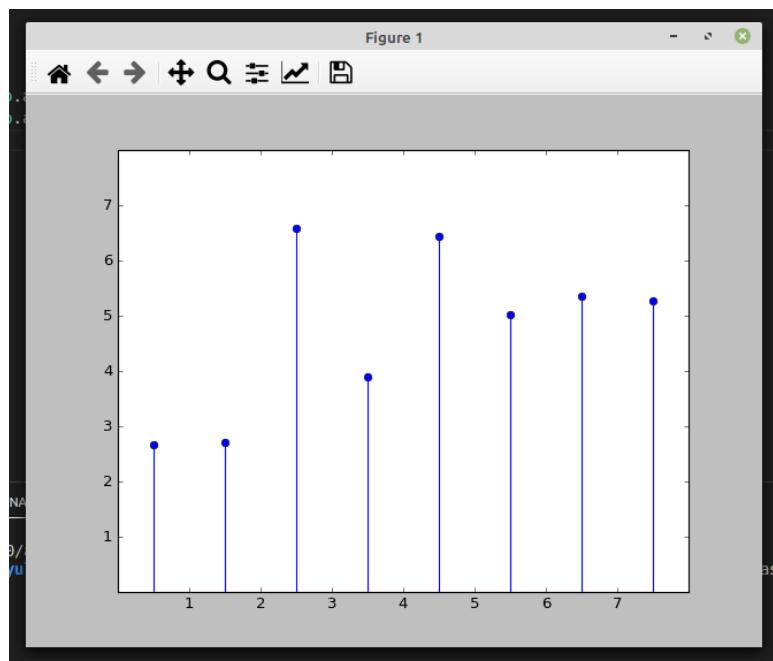
x = 0.5 + np.arange(8)
y = np.random.uniform(2, 7, len(x))

fig, ax = plt.subplots()
ax.stem(x, y)

ax.set(xlim=(0, 8), xticks=np.arange(1, 8),
       ylim=(0, 8), yticks=np.arange(1, 8))
plt.show()

```

результат:



5. step (x, y)

код:

```

import matplotlib.pyplot as plt
import numpy as np

plt.style.use('fast')

x = 0.5 + np.arange(8)
y = np.random.uniform(2, 7, len(x))

fig, ax = plt.subplots()

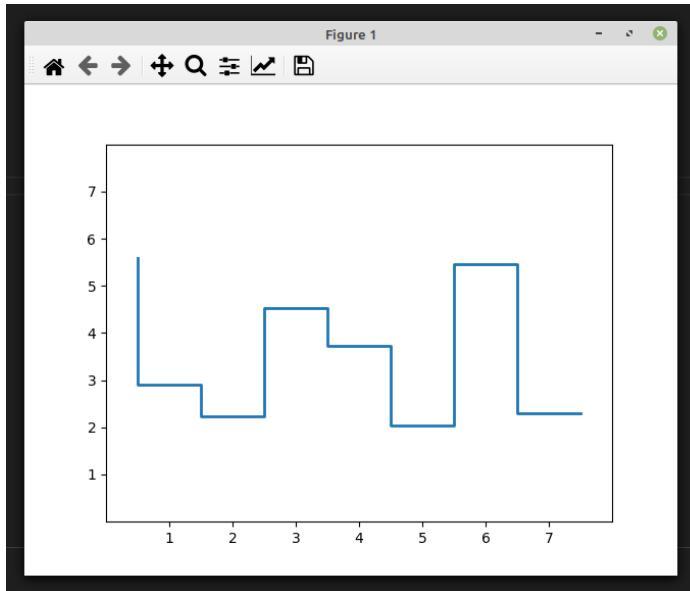
ax.step(x, y, linewidth=2)

```

```
ax.set(xlim=(0, 8), xticks=np.arange(1, 8),
       ylim=(0, 8), yticks=np.arange(1, 8))

plt.show()
```

результатом:



6. fill_between(x, y1, y2)

ког:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('Solarize_Light2')

x = np.linspace(0, 8, 16)
y1 = 3 + 4*x/8 + np.random.uniform(0.0, 0.5, len(x))
y2 = 1 + 2*x/8 + np.random.uniform(0.0, 0.5, len(x))

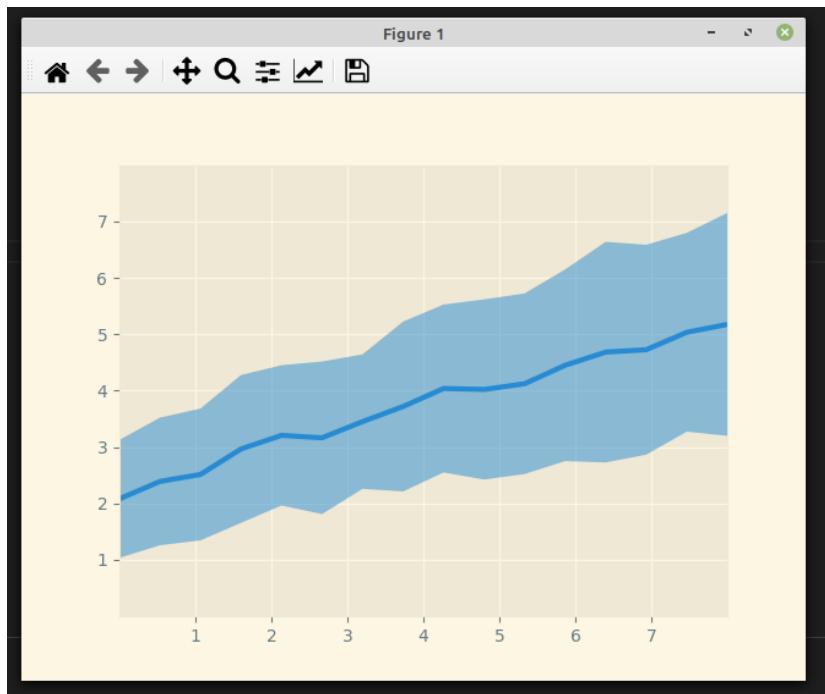
fig, ax = plt.subplots()

ax.fill_between(x, y1, y2, alpha=.5, linewidth=0)
ax.plot(x, (y1 + y2)/2, linewidth=3)

ax.set(xlim=(0, 8), xticks=np.arange(1, 8),
       ylim=(0, 8), yticks=np.arange(1, 8))

plt.show()
```

результат:



7. stackplot(x, y)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('dark_background')

x = np.arange(0, 10, 2)
ay = [4, 1.25, 3, 2.75, 3.5]
by = [1, 4, 1, 3, 1]
cy = [2, 1, 2, 4, 2]
y = np.vstack([ay, by, cy])

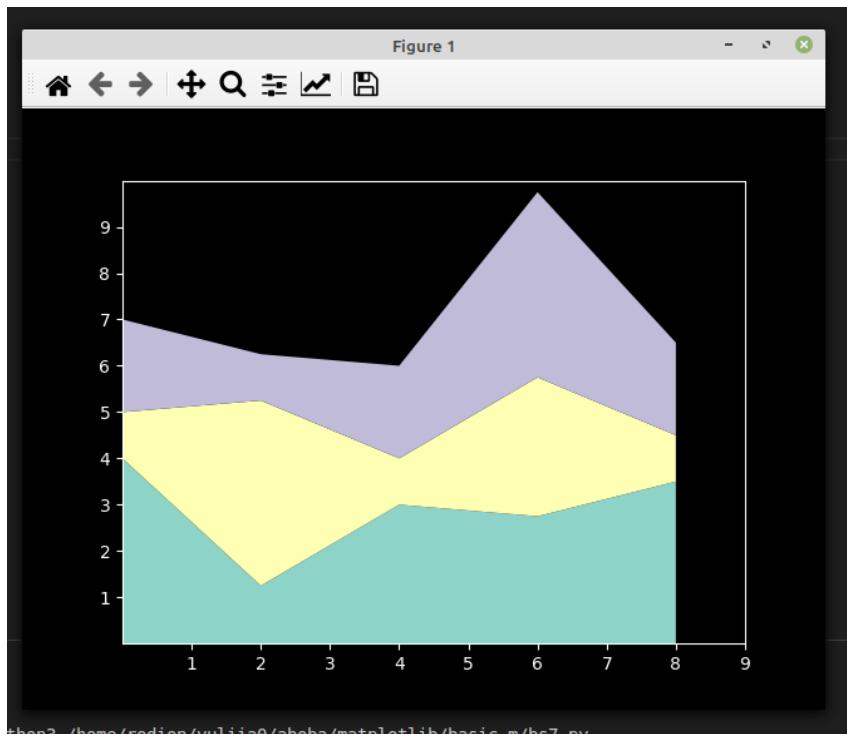
fig, ax = plt.subplots()

ax.stackplot(x, y)

ax.set(xlim=(0, 8), xticks=np.arange(1, 10),
       ylim=(0, 10), yticks=np.arange(1, 10))

plt.show()
```

результатом:



2. Ділянки масивів і полів

1. imshow(Z)

код:

```
import matplotlib.pyplot as plt
import numpy as np

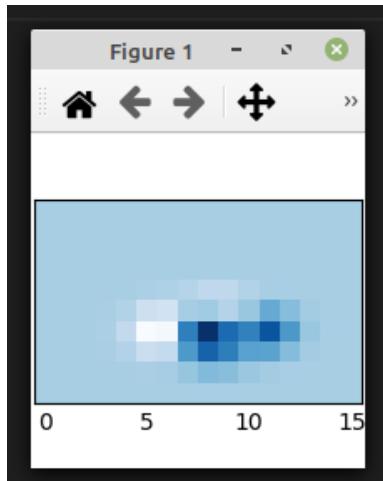
plt.style.use('_mpl-gallery-nogrid')

X, Y = np.meshgrid(np.linspace(-5, 4, 16), np.linspace(-6, 3, 10))
Z = (1 - X/2 + X**5 + Y**3) * np.exp(-X**2 - Y**2)

fig, ax = plt.subplots()

ax.imshow(Z)
plt.show()
```

результат:



2. pcolormesh(X, Y, Z)

код:

```
import matplotlib.pyplot as plt
import numpy as np

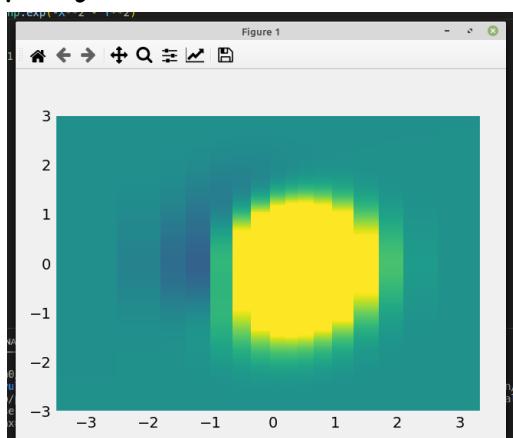
plt.style.use('fivethirtyeight')

x = [-3, -2, -1.6, -1.2, -.8, -.5, -.2, .1, .3, .5, .8, 1.1, 1.5, 1.9,
2.3, 3]
X, Y = np.meshgrid(x, np.linspace(-3, 3, 128))
Z = (5 + X/2 + X**5 - Y**3) * np.exp(-X**2 - Y**2)

fig, ax = plt.subplots()
ax.pcolormesh(X, Y, Z, vmin=-1, vmax=1.0)

plt.show()
```

результат:



3. contour(X, Y, Z)

код:

```
import matplotlib.pyplot as plt
import numpy as np

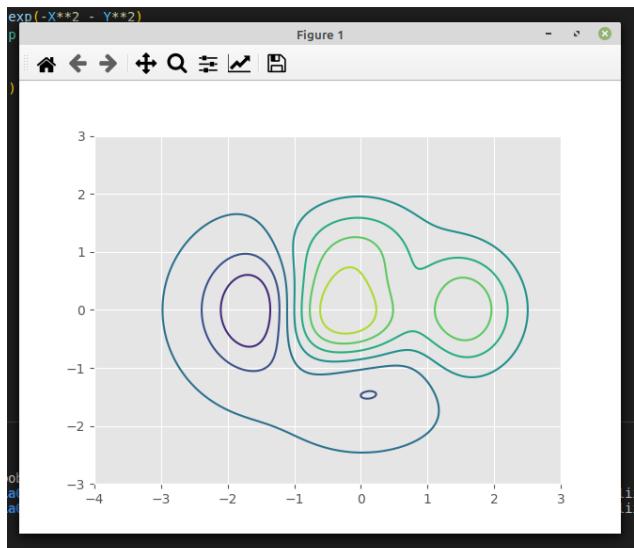
plt.style.use('ggplot')

X, Y = np.meshgrid(np.linspace(-4, 3, 256), np.linspace(-3, 3, 256))
Z = (1 - X/2 + X**5 + Y**3) * np.exp(-X**2 - Y**2)
levels = np.linspace(np.min(Z), np.max(Z), 9)

fig, ax = plt.subplots()
ax.contour(X, Y, Z, levels=levels)

plt.show()
```

результат:



4. contourf(X, Y, Z)

код:

```
import matplotlib.pyplot as plt
import numpy as np

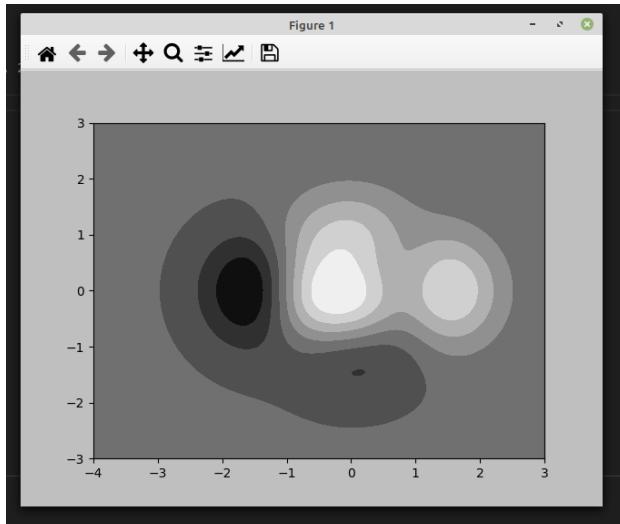
plt.style.use('grayscale')

X, Y = np.meshgrid(np.linspace(-4, 3, 256), np.linspace(-3, 3, 256))
Z = (1 - X/2 + X**5 + Y**3) * np.exp(-X**2 - Y**2)
levels = np.linspace(Z.min(), Z.max(), 9)
```

```
fig, ax = plt.subplots()
ax.contourf(X, Y, z, levels=levels)

plt.show()
```

результатом:



5. barbs (X, Y, U, V)

ког:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('ggplot')

X, Y = np.meshgrid([1, 2, 3, 4], [1, 2, 3, 4])
angle = np.pi / 180 * np.array([[15, 30, 35, 45],
[25, 40, 55, 60],
[35, 50, 65, 75],
[45, 60, 75, 90]])
amplitude = np.array([[5, 10, 25, 50],
[10, 15, 30, 60],
[15, 26, 50, 70],
[20, 45, 80, 100]])
U = amplitude * np.sin(angle)
V = amplitude * np.cos(angle)

fig, ax = plt.subplots()
```

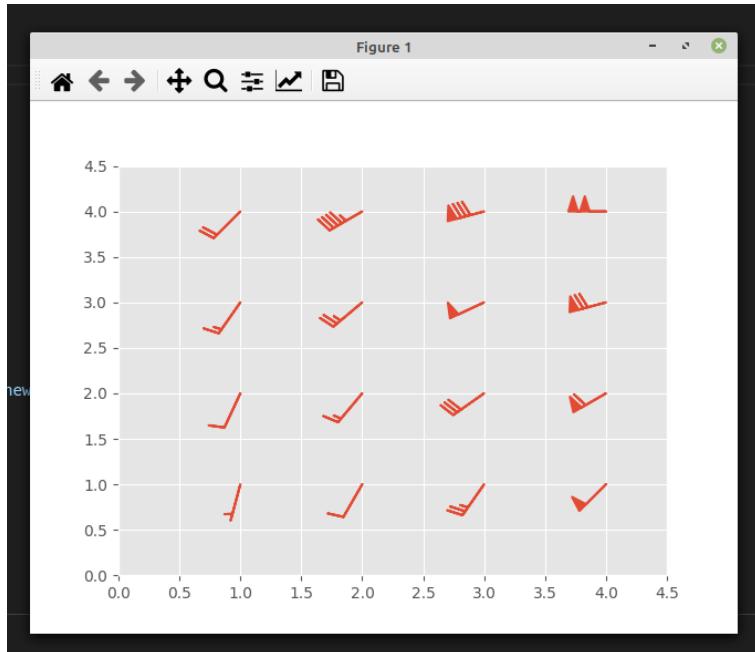
```

ax.barbs(X, Y, U, V, barbcolor='C0', flagcolor='C0', length=7,
linewidth=1.5)
ax.set(xlim=(0, 4.5), ylim=(0, 4.5))

plt.show()

```

результатом:



6. quiver (X, Y, U, V)

код:

```

import matplotlib.pyplot as plt
import numpy as np

plt.style.use('fast')

x = np.linspace(-4, 4, 6)
y = np.linspace(-4, 4, 6)
X, Y = np.meshgrid(x, y)

V = X + Y
U = Y - X

fig, ax = plt.subplots()

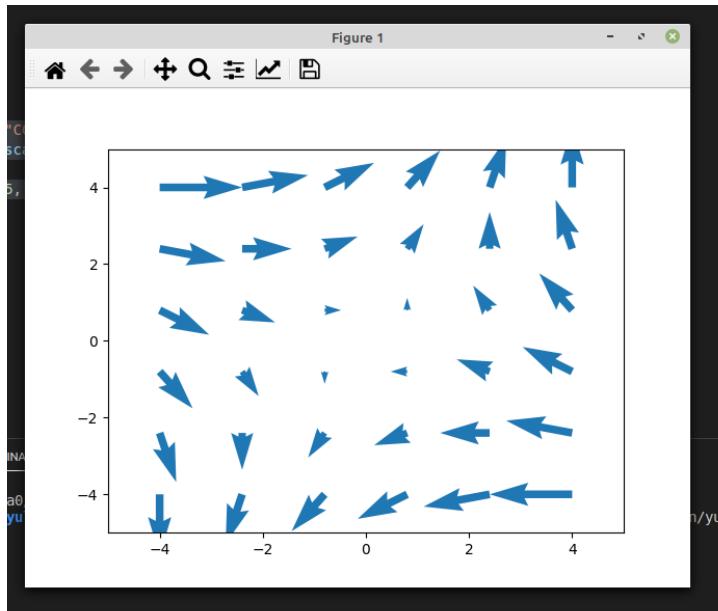
ax.quiver(X, Y, U, V, color="C0", angles='xy',
scale_units='xy', scale=5, width=.015)

```

```
ax.set(xlim=(-5, 5), ylim=(-5, 5))

plt.show()
```

результат:



7. streamplot(X, Y, U, V)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('grayscale')

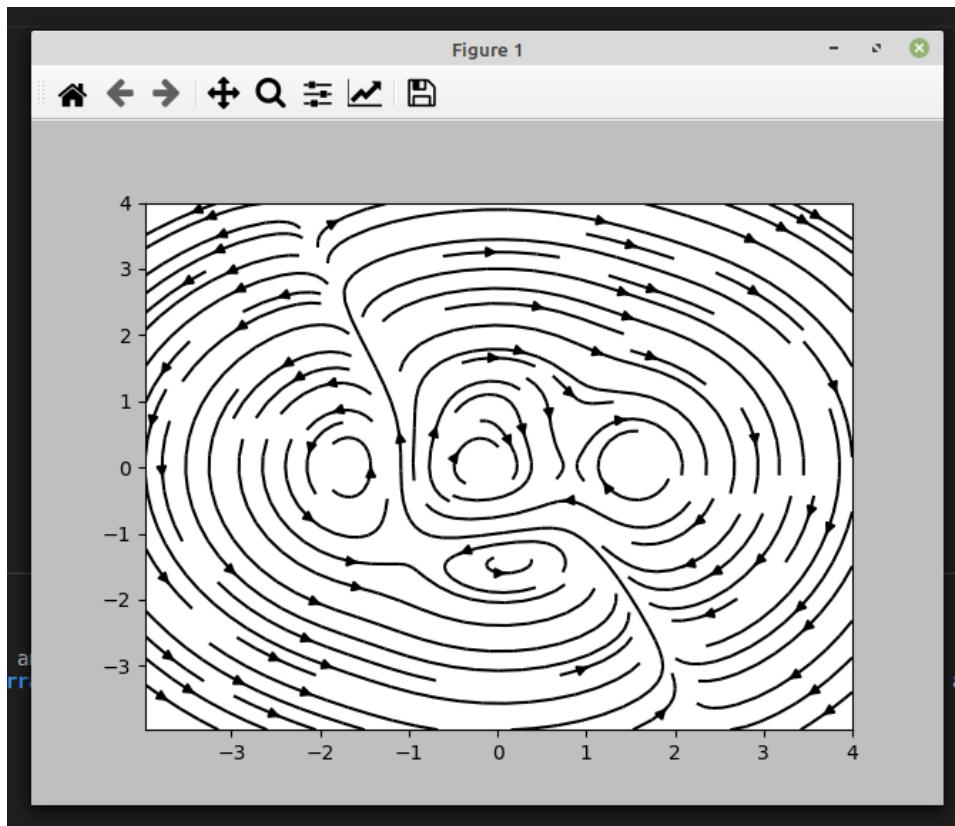
X, Y = np.meshgrid(np.linspace(-4, 4, 256), np.linspace(-4, 4, 256))
Z = (1 - X/2 + X**5 + Y**3) * np.exp(-X**2 - Y**2)

V = np.diff(Z[1:, :], axis=1)
U = -np.diff(Z[:, 1:], axis=0)

fig, ax = plt.subplots()
ax.streamplot(X[1:, 1:], Y[1:, 1:], U, V)

plt.show()
```

результатом:



3. Статистичні графіки

1. hist(x)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('default')

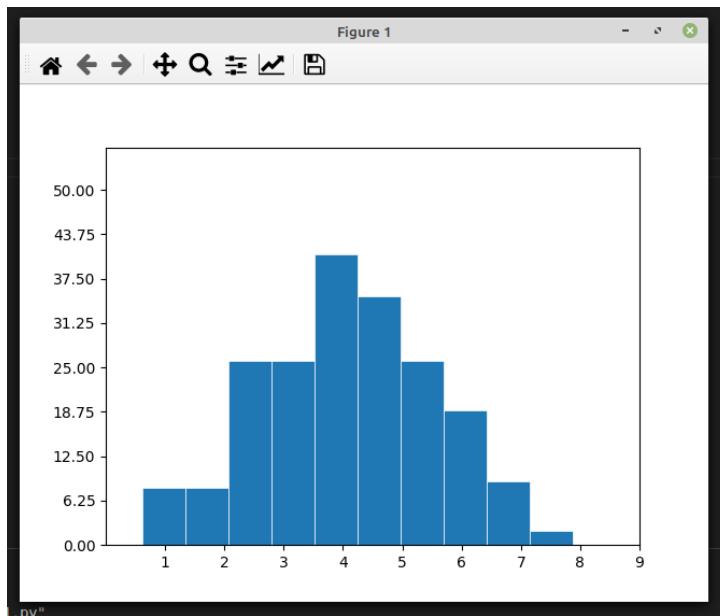
x = 4 + np.random.normal(0, 1.5, 200)
fig, ax = plt.subplots()

ax.hist(x, bins=10, linewidth=0.5, edgecolor="white")

ax.set(xlim=(0, 8), xticks=np.arange(1, 10),
       ylim=(0, 56), yticks=np.linspace(0, 50, 9))
```

```
plt.show()
```

результат:



2. boxplot(X)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('classic')

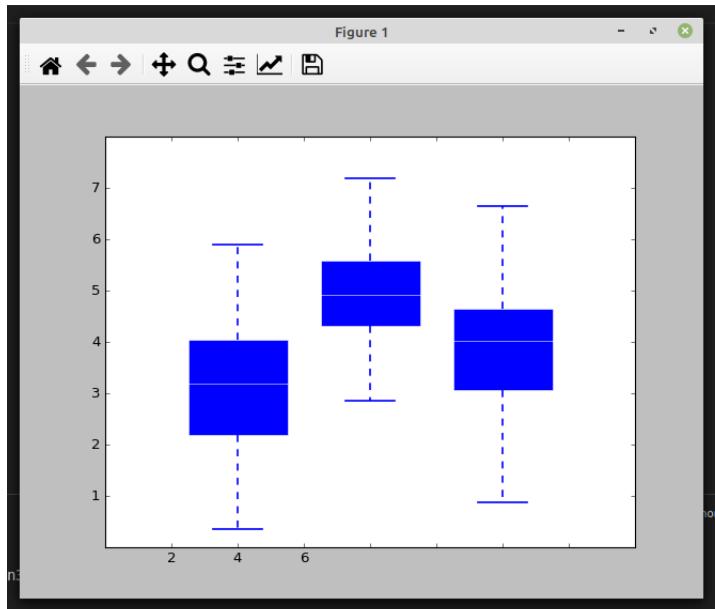
D = np.random.normal((3, 5, 4), (1.25, 1.00, 1.25), (100, 3))

fig, ax = plt.subplots()
VP = ax.boxplot(D, positions=[2, 4, 6], widths=1.5, patch_artist=True,
showmeans=False, showfliers=False,
medianprops={"color": "white", "linewidth": 0.5},
boxprops={"facecolor": "C0", "edgecolor": "white",
"linewidth": 0.5},
whiskerprops={"color": "C0", "linewidth": 1.5},
capprops={"color": "C0", "linewidth": 1.5})

ax.set(xlim=(0, 8), xticks=np.arange(1, 8),
ylim=(0, 8), yticks=np.arange(1, 8))

plt.show()
```

результат:



3. errorbar(x, y, yerr, xerr)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('fast')

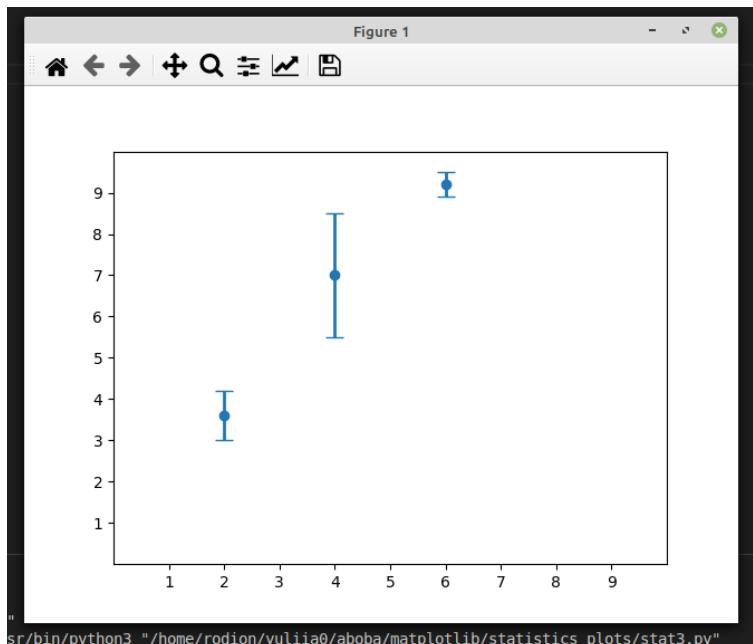
x = [2, 4, 6]
y = [3.6, 4.5, 5.2]
yerr = [0.6, 1.5, 0.3]
fig, ax = plt.subplots()

ax.errorbar(x, y, yerr, fmt='o', linewidth=2, capsize=6)

ax.set(xlim=(0, 10), xticks=np.arange(1, 10),
       ylim=(0, 10), yticks=np.arange(1, 10))

plt.show()
```

результат:



4. violinplot(D)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('default')

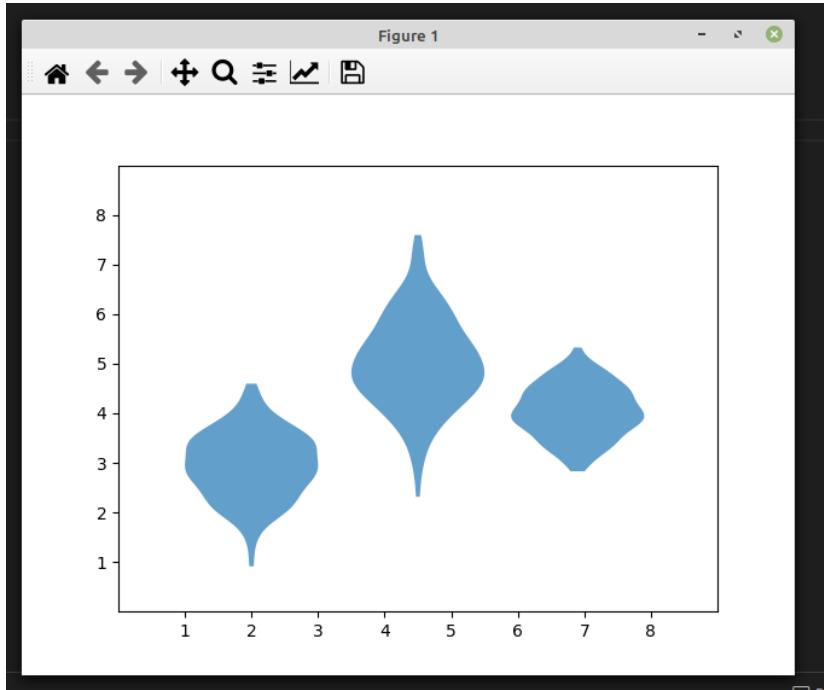
D = np.random.normal((3, 5, 4), (0.7, 1.00, 0.5), (200, 3))

fig, ax = plt.subplots()

vp = ax.violinplot(D, [2, 4.5, 6.9], widths=2,
showmeans=False, showmedians=False, showextrema=False)
for body in vp['bodies']:
body.set_alpha(0.7)
ax.set(xlim=(0, 9), xticks=np.arange(1, 9),
ylim=(0, 9), yticks=np.arange(1, 9))

plt.show()
```

результат:



5. eventplot(D)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('fast')

np.random.seed(1)
x = [2, 4, 6]
D = np.random.gamma(3, size=(3, 50))

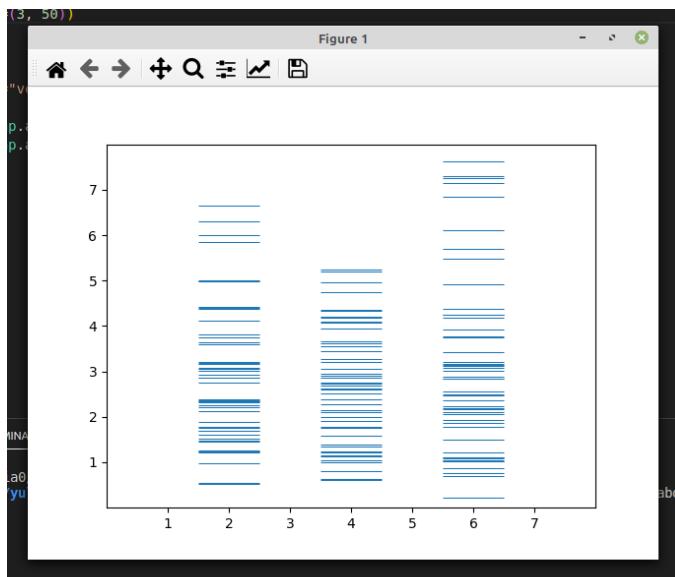
fig, ax = plt.subplots()

ax.eventplot(D, orientation="vertical", lineoffsets=x, linewidth=0.75)

ax.set(xlim=(0, 8), xticks=np.arange(1, 8),
       ylim=(0, 8), yticks=np.arange(1, 8))

plt.show()
```

результат:



6. hist2d(x, y)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('_mpl-gallery-nogrid')

x = np.random.randn(5000)
y = 1.2 * x + x / 3

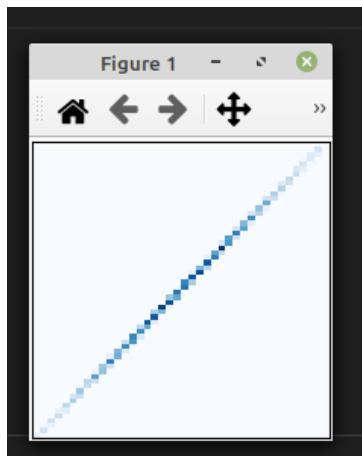
fig, ax = plt.subplots()

ax.hist2d(x, y, bins=(np.arange(-3, 3, 0.1), np.arange(-3, 3, 0.1)))

ax.set(xlim=(-2, 2), ylim=(-3, 3))

plt.show()
```

результат:



7. hexbin(x, y, C)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('_mpl-gallery-nogrid')

x = np.random.randn(5000)
y = 1.2 * x + np.random.randn(5000) / 3

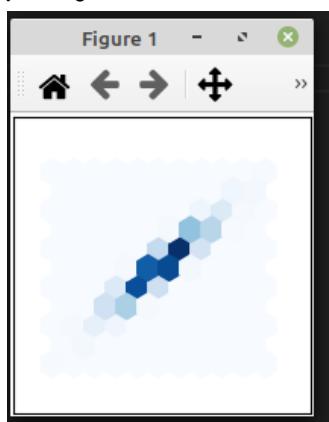
fig, ax = plt.subplots()

ax.hexbin(x, y, gridsize=10)

ax.set(xlim=(-5, 5), ylim=(-7, 7))

plt.show()
```

результат:



8. pie(x)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('classic')

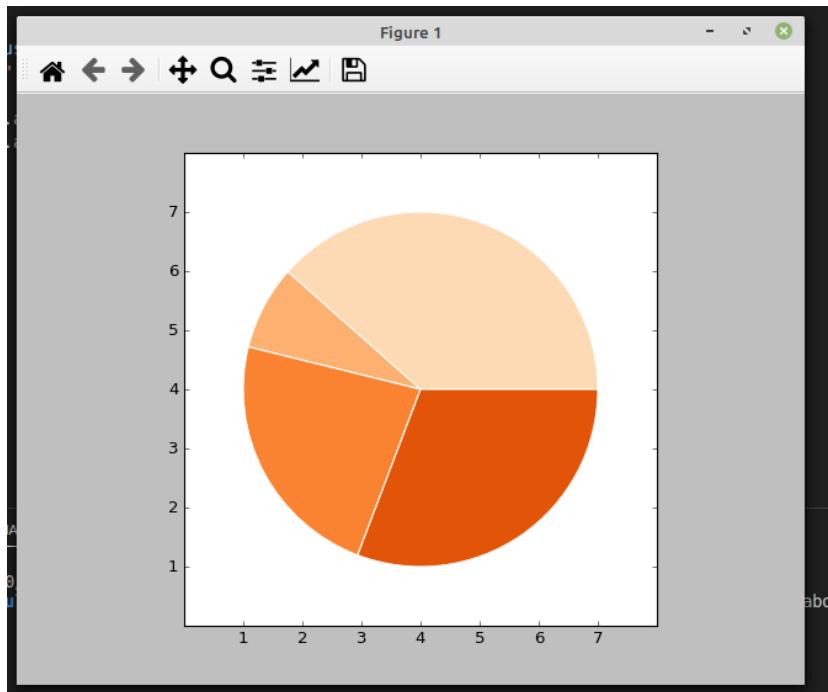
x = [5, 1, 3, 4]
colors = plt.get_cmap('Oranges')(np.linspace(0.2, 0.7, len(x)))

fig, ax = plt.subplots()
ax.pie(x, colors=colors, radius=3, center=(4, 4),
wedgeprops={"linewidth": 1, "edgecolor": "white"}, frame=True)

ax.set(xlim=(0, 8), xticks=np.arange(1, 8),
ylim=(0, 8), yticks=np.arange(1, 8))

plt.show()
```

результат:



4. Неструктуровані координати

1. tricontour(x, y, z)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('_mpl-gallery-nogrid')

x = np.random.uniform(-6, 6, 256)
y = np.random.uniform(-6, 6, 256)
z = (1 - x/2 + x**5 + y**3) * np.exp(-x**2 - y**2)
levels = np.linspace(z.min(), z.max(), 7)

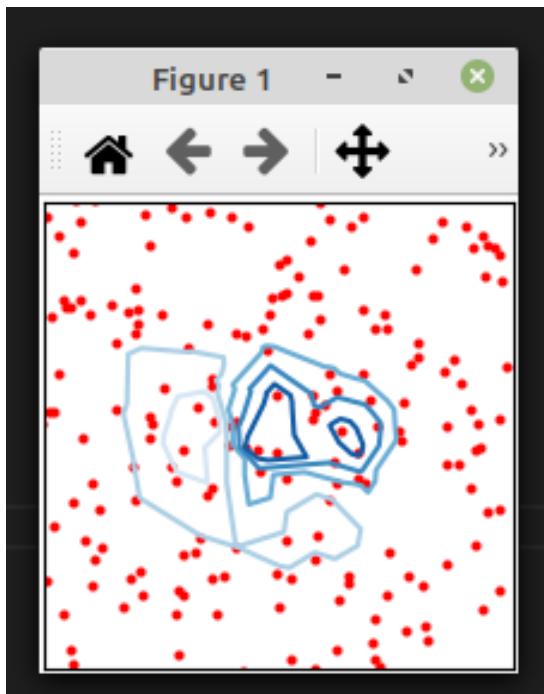
fig, ax = plt.subplots()

ax.plot(x, y, 'o', markersize=2, color='red')
ax.tricontour(x, y, z, levels=levels)

ax.set(xlim=(-5, 5), ylim=(-5, 5))

plt.show()
```

результатом:



2. tricontourf(x, y, z)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('_mpl-gallery-nogrid')

x = np.random.uniform(-10, 10, 256)
y = np.random.uniform(-10, 10, 256)
z = (1 - x/2 + x**5 + y**3) * np.exp(-x**2 - y**2)
levels = np.linspace(z.min(), z.max(), 7)

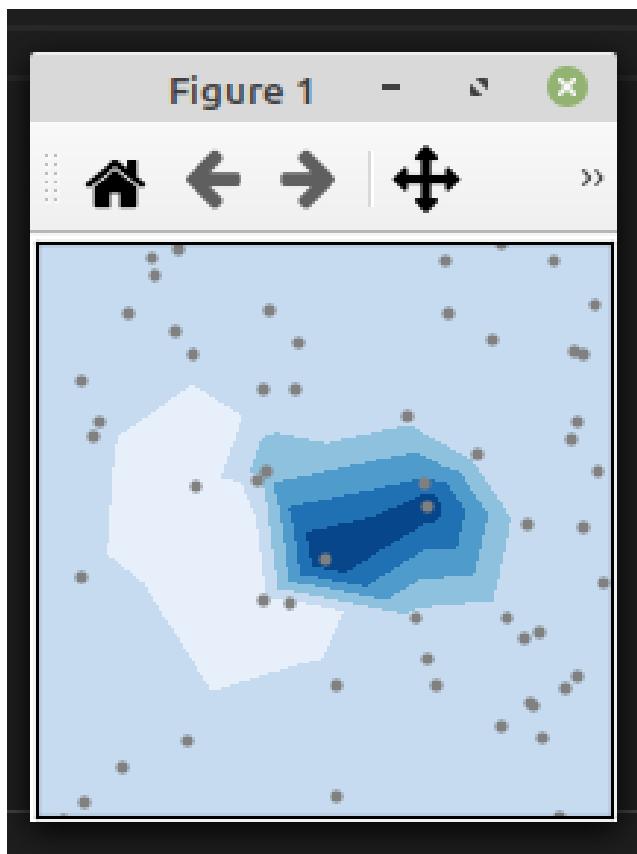
fig, ax = plt.subplots()

ax.plot(x, y, 'o', markersize=2, color='grey')
ax.tricontourf(x, y, z, levels=levels)

ax.set(xlim=(-5, 5), ylim=(-5, 5))

plt.show()
```

результат:



3. tripcolor(x, y, z)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('_mpl-gallery-nogrid')

x = np.random.uniform(-3, 3, 256)
y = np.random.uniform(-3, 3, 256)
z = (1 - x/2 + x**5 + y**3) * np.exp(-x**2 - y**2)

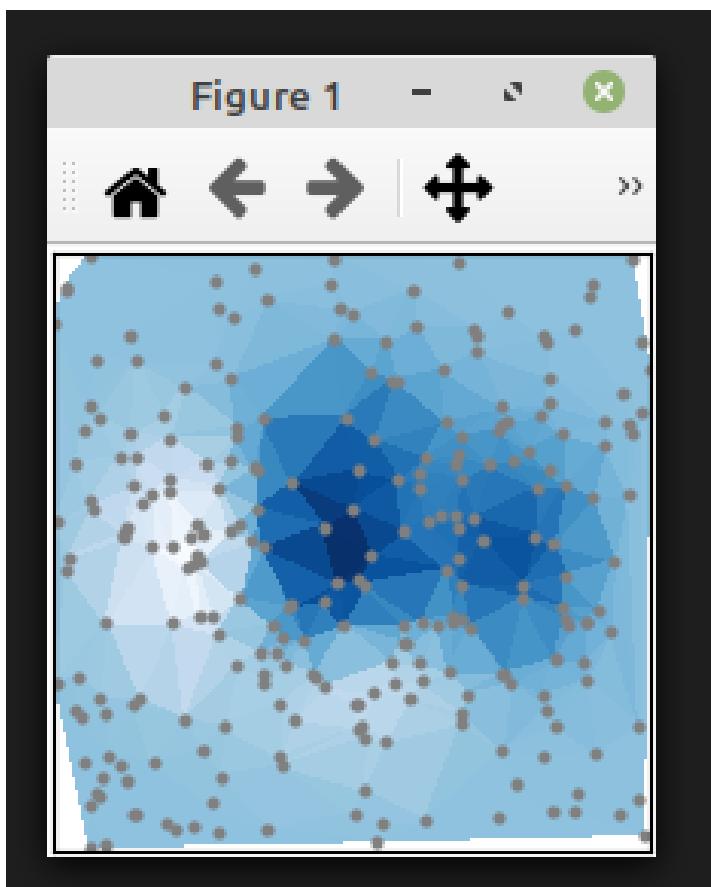
fig, ax = plt.subplots()

ax.plot(x, y, 'o', markersize=2, color='grey')
ax.tripcolor(x, y, z)

ax.set(xlim=(-3, 3), ylim=(-3, 3))

plt.show()
```

результатом:



4. triplot(x, y)

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('_mpl-gallery-nogrid')

x = np.random.uniform(-10, 10, 256)
y = np.random.uniform(-12, 15, 256)
z = (1 - x/2 + x**5 + y**3) * np.exp(-x**2 - y**2)

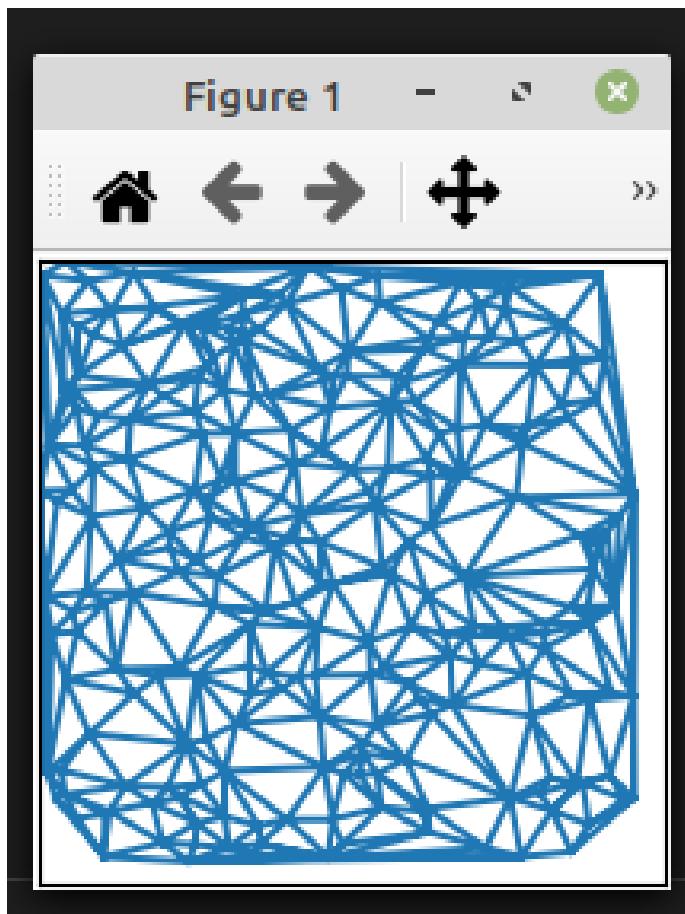
fig, ax = plt.subplots()

ax.triplot(x, y)

ax.set(xlim=(-10, 11), ylim=(-13, 15))

plt.show()
```

результат:



5. 3D

1. 3D scatterplot

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('classic')

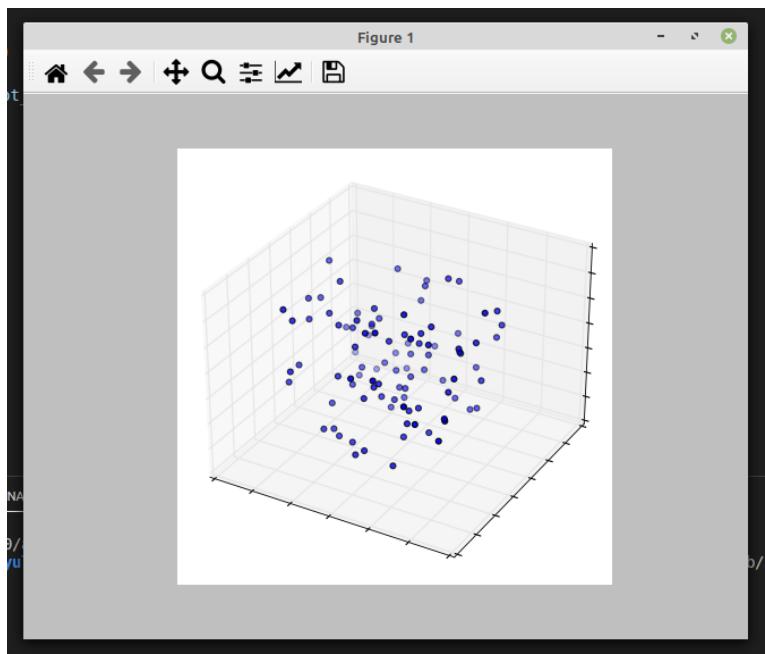
n = 100
rng = np.random.default_rng()
xs = rng.uniform(23, 32, n)
ys = rng.uniform(0, 100, n)
zs = rng.uniform(-50, -25, n)

fig, ax = plt.subplots(subplot_kw={"projection": "3d"})
ax.scatter(xs, ys, zs)

ax.set(xticklabels=[],
       yticklabels=[],
       zticklabels=[])

plt.show()
```

результатом:



2. 3D surface

код:

```
import matplotlib.pyplot as plt
from matplotlib import cm
import numpy as np

plt.style.use('fast')

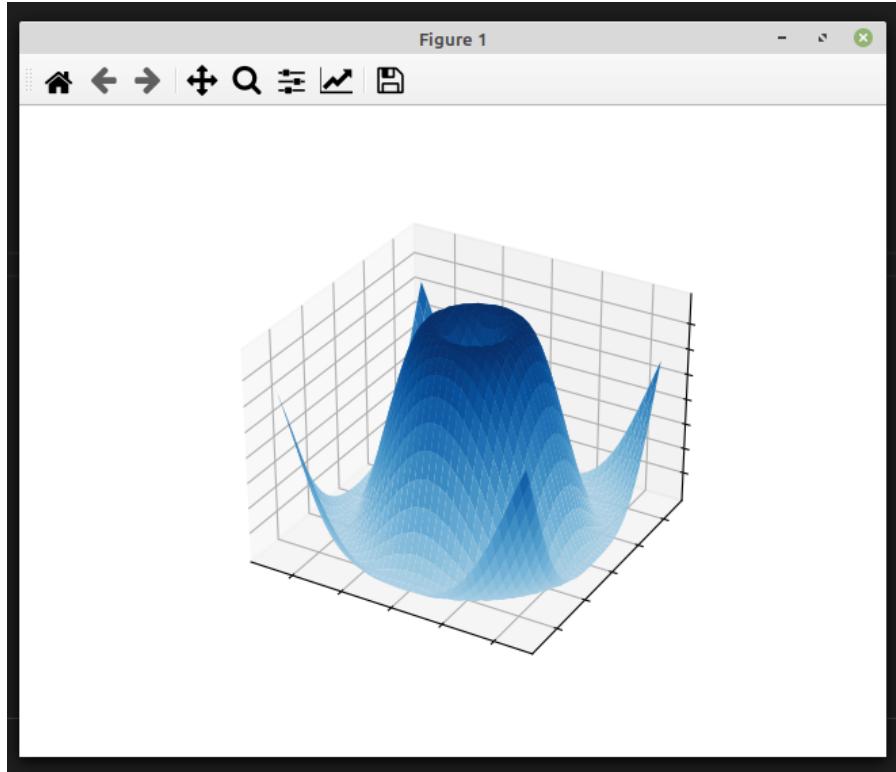
X = np.arange(-5, 5, 0.25)
Y = np.arange(-5, 5, 0.25)
X, Y = np.meshgrid(X, Y)
R = np.sqrt(X**2 + Y**2)
Z = np.sin(R)

fig, ax = plt.subplots(subplot_kw={"projection": "3d"})
ax.plot_surface(X, Y, Z, vmin=Z.min() * 2, cmap=cm.Blues)

ax.set(xticklabels=[],
       yticklabels=[],
       zticklabels=[])

plt.show()
```

результатом:



3. Triangular 3D surfaces

Код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('fast')

radii = np.linspace(0.125, 1.0, 8)
angles = np.linspace(0, 2*np.pi, 36, endpoint=False)[..., np.newaxis]

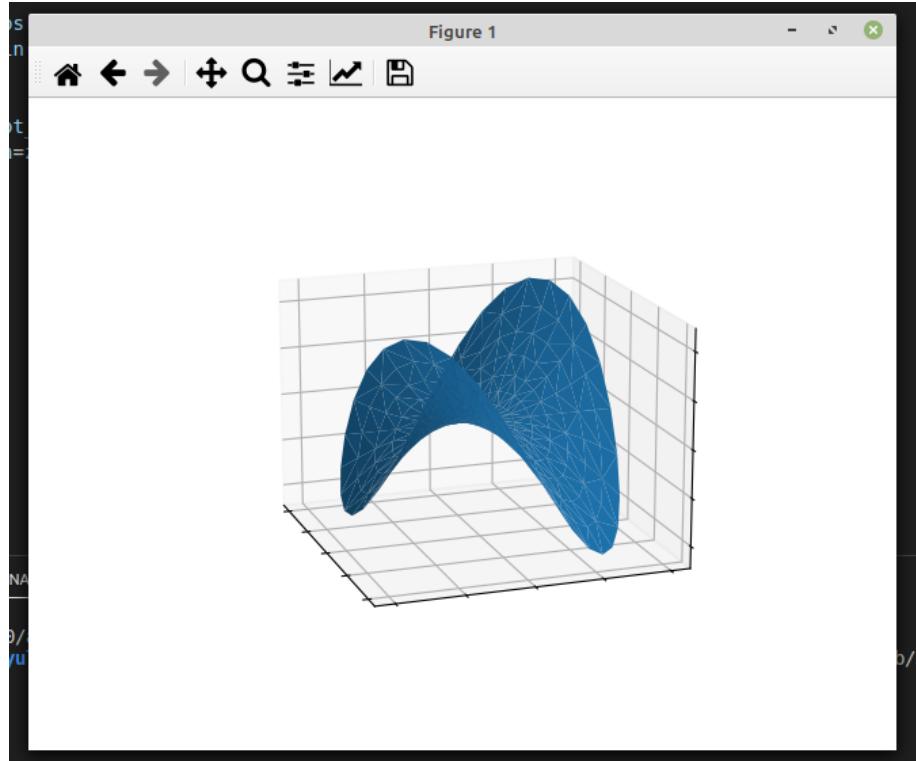
x = np.append(0, (radii*np.cos(angles)).flatten())
y = np.append(0, (radii*np.sin(angles)).flatten())
z = np.sin(-x*y)

fig, ax = plt.subplots(subplot_kw={'projection': '3d'})
ax.plot_trisurf(x, y, z, vmin=z.min() * 2)

ax.set(xticklabels=[],
       yticklabels=[],
       zticklabels=[])

plt.show()
```

результатом:



4. 3D voxel / volumetric plot

код:

```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('_mpl-gallery')

x, y, z = np.indices((8, 8, 8))

cube1 = (x < 4) & (y < 5) & (z < 3)
cube2 = (x >= 5) & (y >= 5) & (z >= 5)

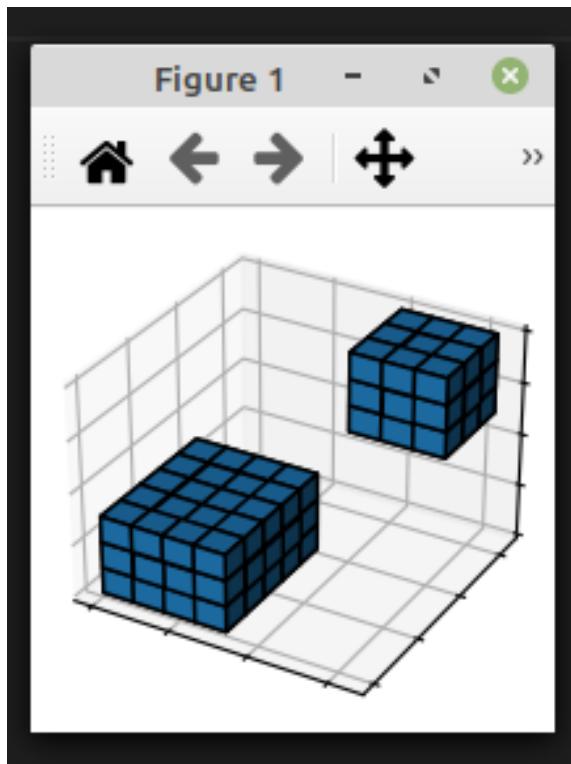
voxelarray = cube1 | cube2

fig, ax = plt.subplots(subplot_kw={"projection": "3d"})
ax.voxels(voxelarray, edgecolor='k')

ax.set(xticklabels=[],
       yticklabels=[],
       zticklabels=[])

plt.show()
```

результатом:



5. 3D wireframe plot

код:

```
from mpl_toolkits.mplot3d import axes3d
import matplotlib.pyplot as plt

plt.style.use('classic')

X, Y, Z = axes3d.get_test_data(0.015)
fig, ax = plt.subplots(subplot_kw={"projection": "3d"})
ax.plot_wireframe(X, Y, Z, rstride=15, cstride=10)

ax.set(xticklabels=[],
       yticklabels=[],
       zticklabels=[])

plt.show()
```

результат:

