Języki i techniki programowania

Wykład 7

Maciej Rybczyński

matplotlib

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

plt.rcParams['font.size'] = 18

x = np.random.normal(0,4,50000)
y = np.random.normal(0,4,50000)


plt.plot(x,y,'b,')              # niebieskie piksele
plt.axis([-15,15,-15,15])
plt.show()
```

50000 licz losowych
Z rozkładu normalnego
średnia = 0, szerokość = 4

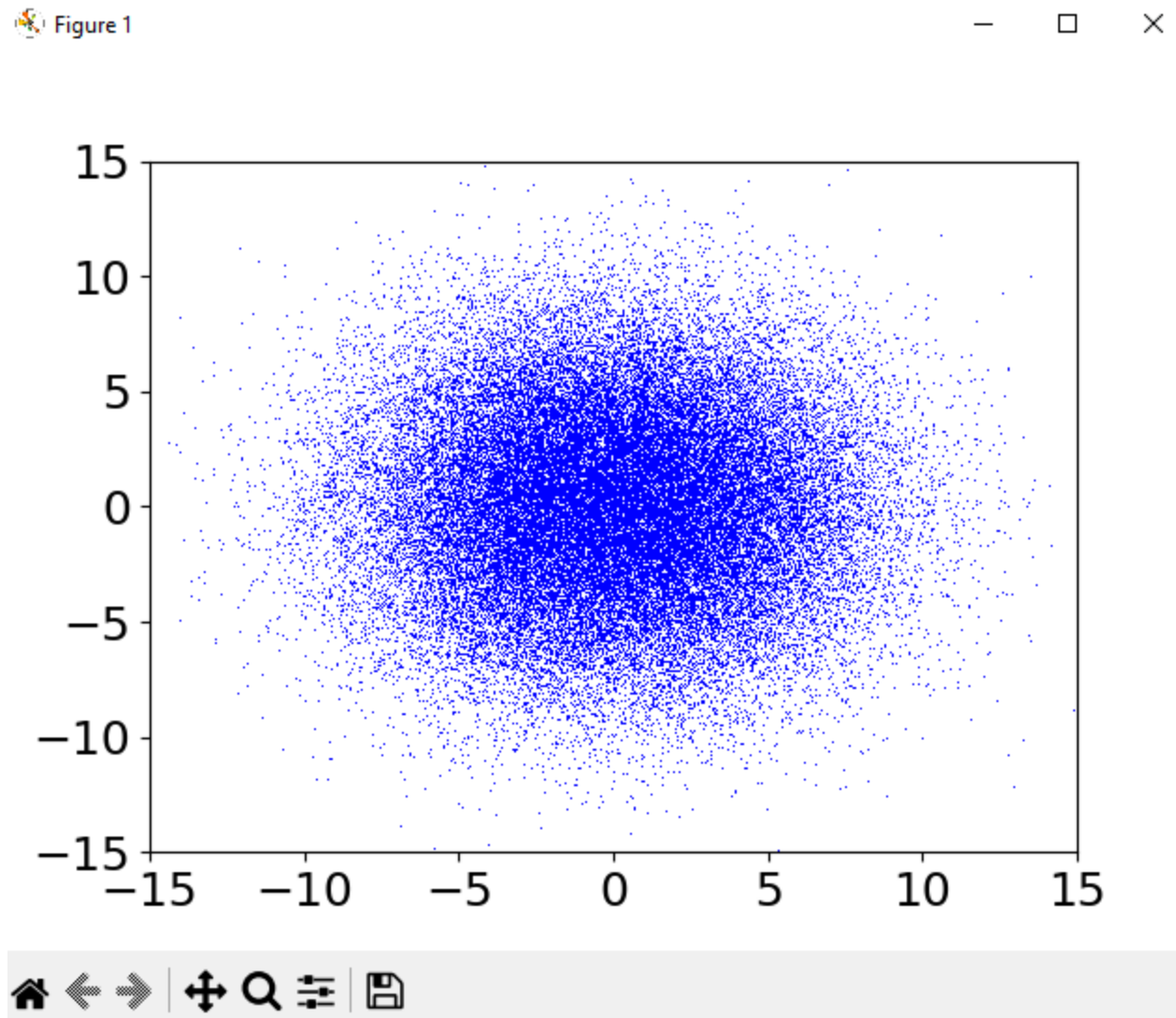


Figure 1
15
10
5
0
−5
−10
−15
−15
−10
−5
0
5
10
15

Z-order

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

plt.rcParams['font.size'] = 18

x = np.random.normal(0,4,100)
y = np.random.normal(0,4,100)


plt.plot(x,y,'b*', ms=5, zorder=2)
plt.plot([-10,10],[0,0],'r-',lw=10, zorder=1)

plt.axis([-15,15,-15,15])
plt.show()
```

Kolejność rysowania:
-zorder=1
-zorder=2
-zorder=3, etc

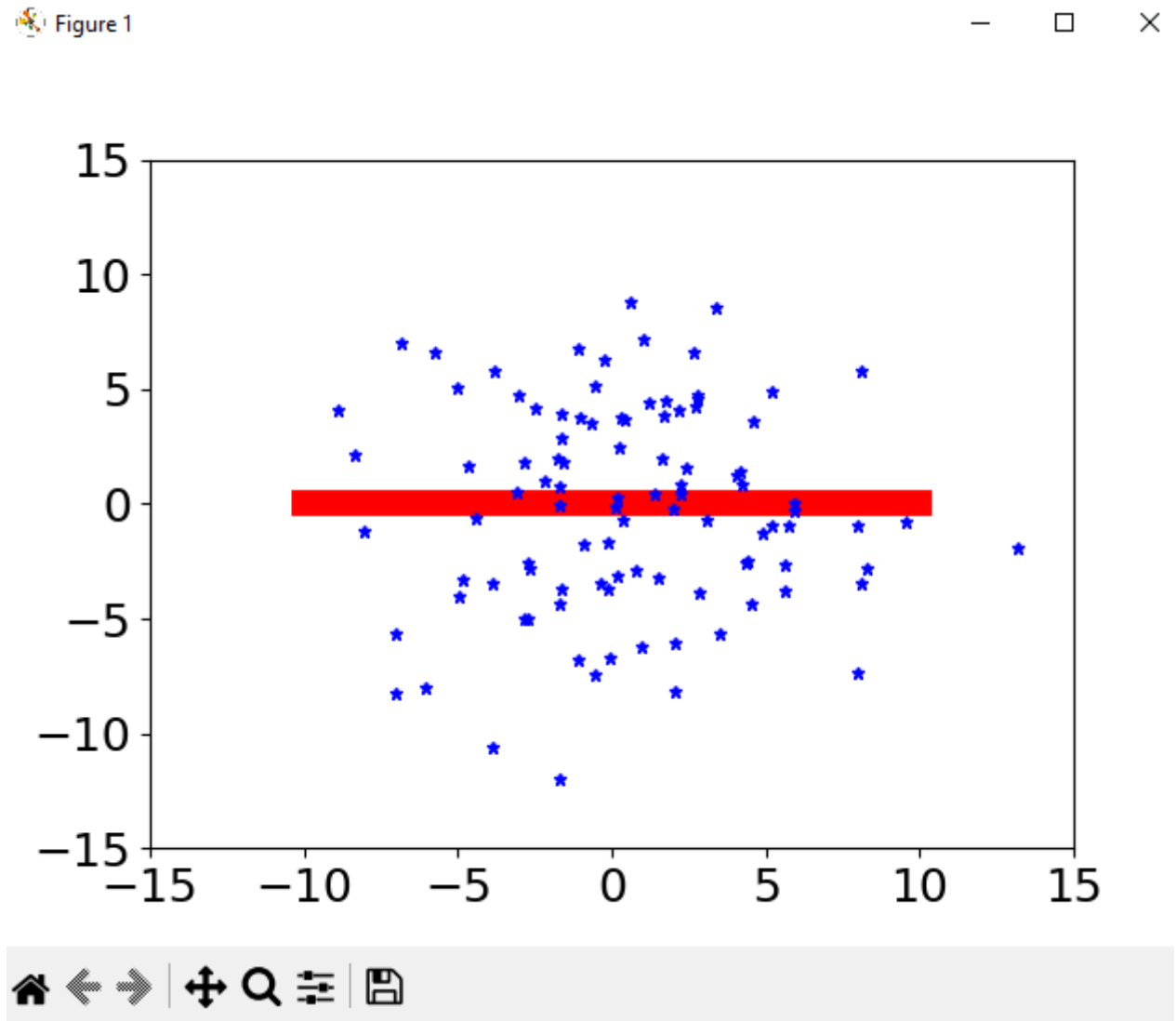


Figure 1
15
10
5
0
−5
−10
−15
−15
−10
−5
0
5
10
15

Histogram

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

plt.rcParams['font.size'] = 18

L = np.random.uniform(0,10,1500)

plt.hist(L)          # histogram

plt.show()
```

1500 liczb losowych z przedziału [0,10)

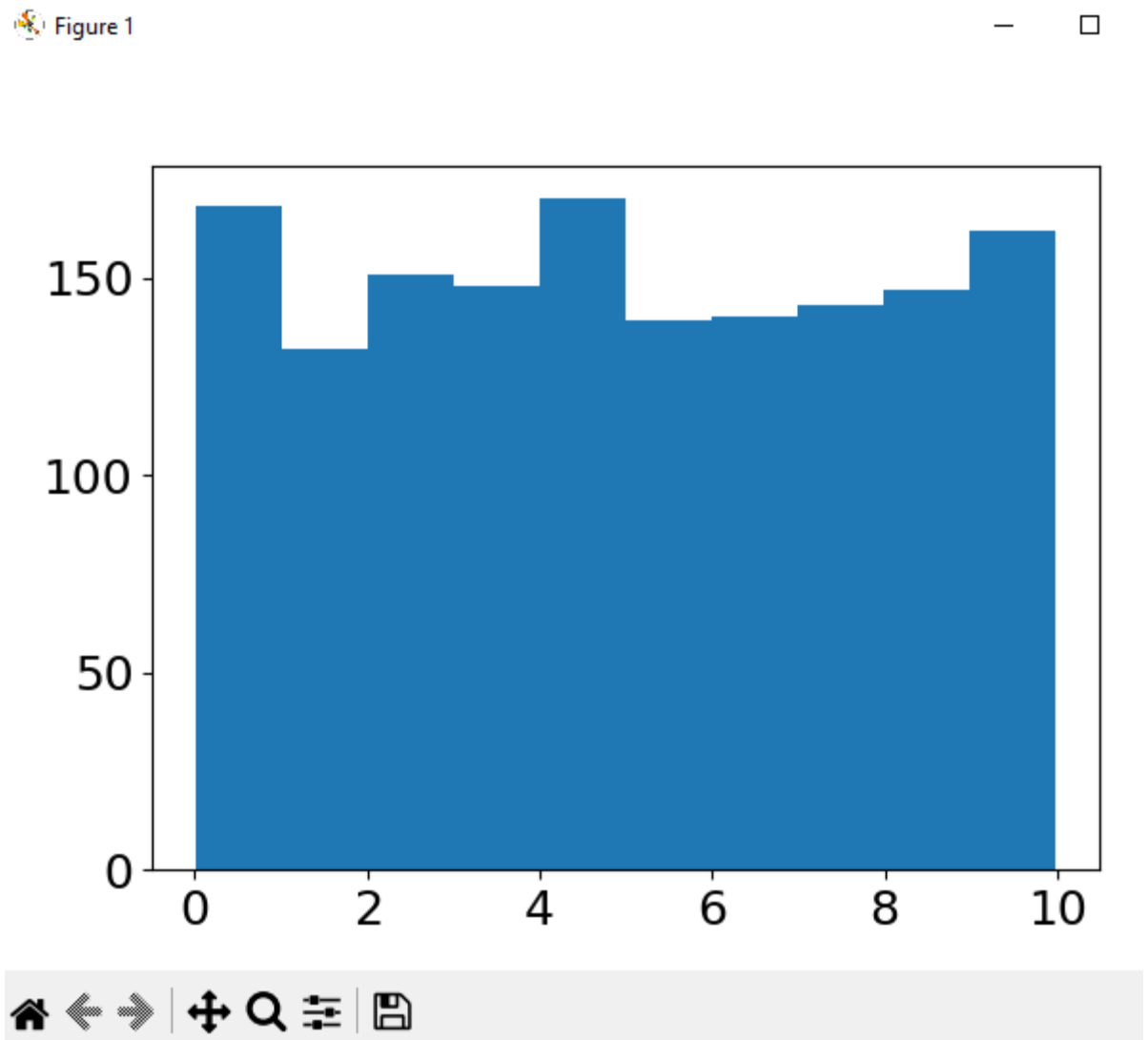
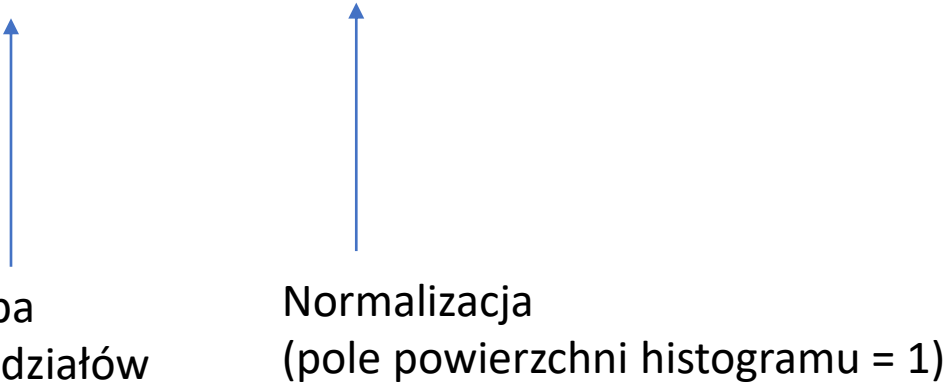


Figure 1
150
100
50
0
0
2
4
6
8
10

Histogram

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

# Fixing random state for reproducibility
np.random.seed(19680801)

plt.rcParams['font.size'] = 18

L = np.random.uniform(0,10,1500)

plt.hist(L, color='green', bins=40, density=True)

plt.show()
```



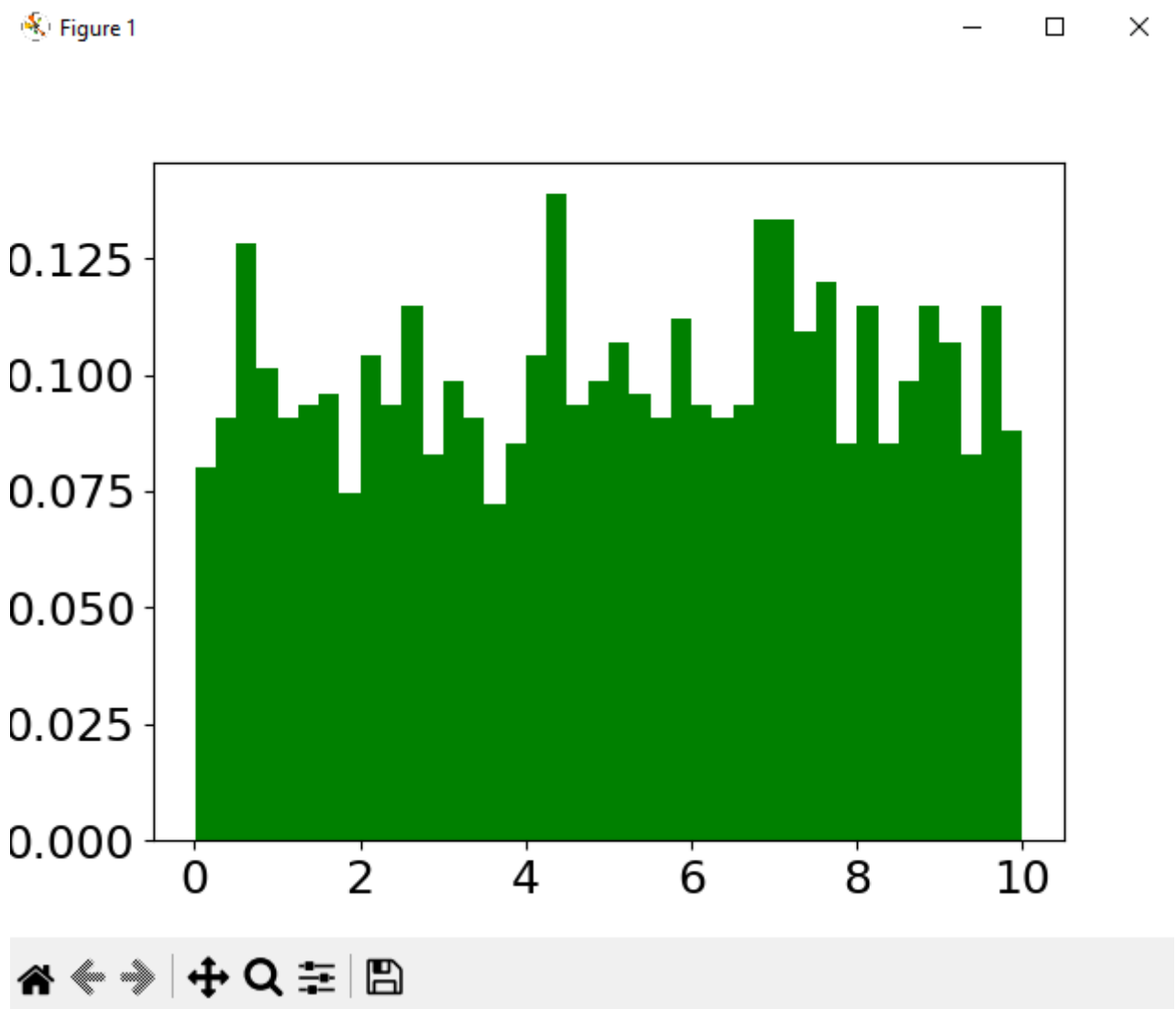


Figure 1
0.125
0.100
0.075
0.050
0.025
0.000
0
2
4
6
8
10

Histogram

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

plt.rcParams['font.size'] = 14
plt.rcParams['legend.fontsize'] = 14

L1 = np.random.uniform(0,10,9500)
L2 = np.random.normal(0,3,9500)

plt.hist(L1, color='g', bins=50, density=True, alpha=0.8,
         label='uniform')
plt.hist(L2, color='b', bins=50, density=True, alpha=0.5,
         label='normal')

plt.axis([-10,10,0,0.16])
plt.xlabel('x')
plt.ylabel('y')
plt.legend(loc='upper left', frameon=False)
plt.show()
```

Przezroczystość

Sprawdź, zmieniając wartości:
plt.legend(bbox_to_anchor=(0.5,0.75), frameon=False)

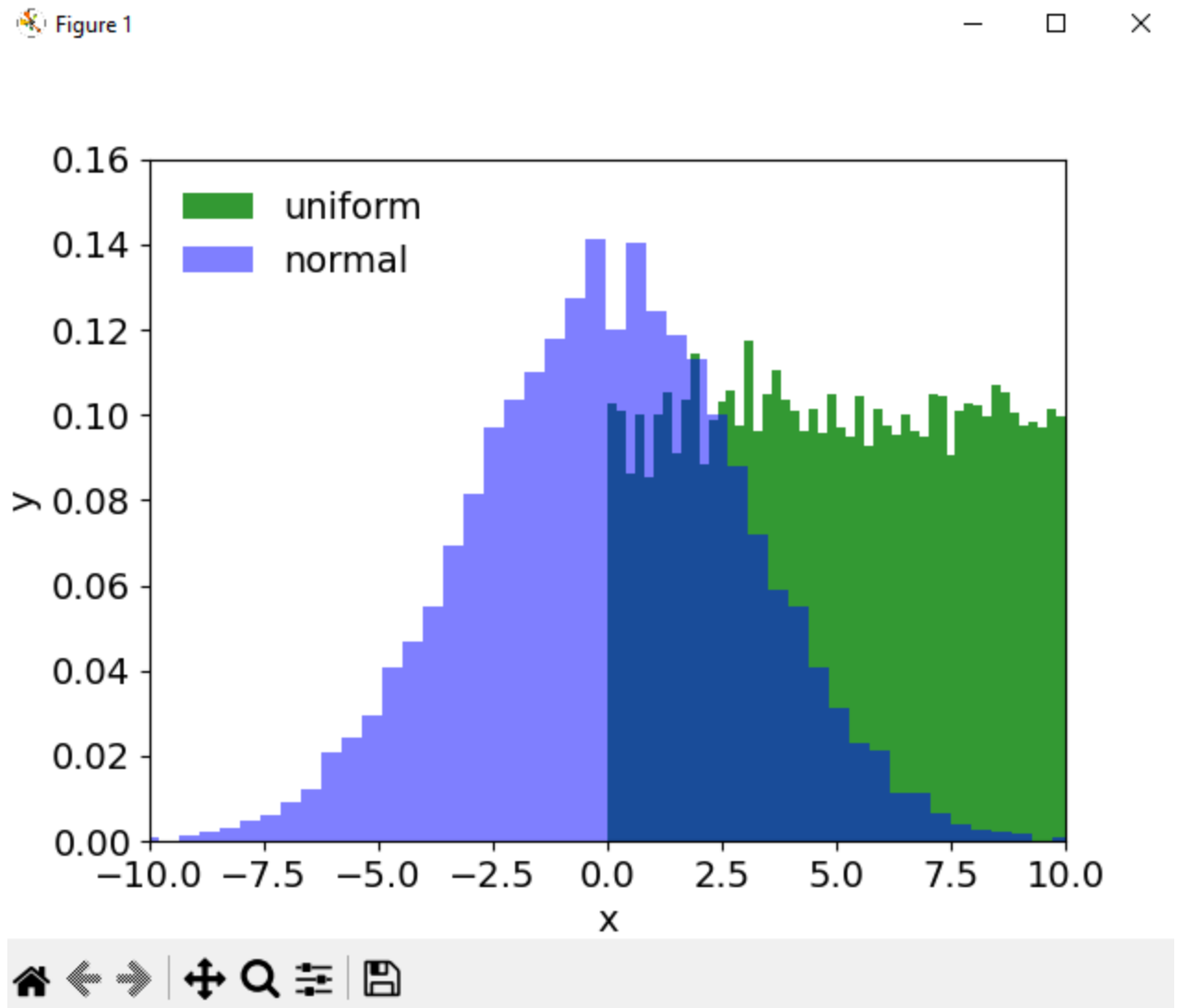


Figure 1
uniform
normal
0.16
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
y
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0
x

Histogram

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

plt.rcParams['font.size'] = 18

L = np.random.normal(0,3,10**6)

mybins = np.arange(-10,11,2)
# [-10, -8, -6, -4, -2, 0, 2, 4, 6, 8, 10]

plt.hist(L, color='r', bins=mybins, density=True)

plt.show()
```

Możemy zdefiniować własne przedziały. W tym przypadku:
przedział 1: od -10 do -8
przedział 2: od -8 do -6
przedział 3: od -6 do -4
etc.
ostatni przedział: od 8 do 10

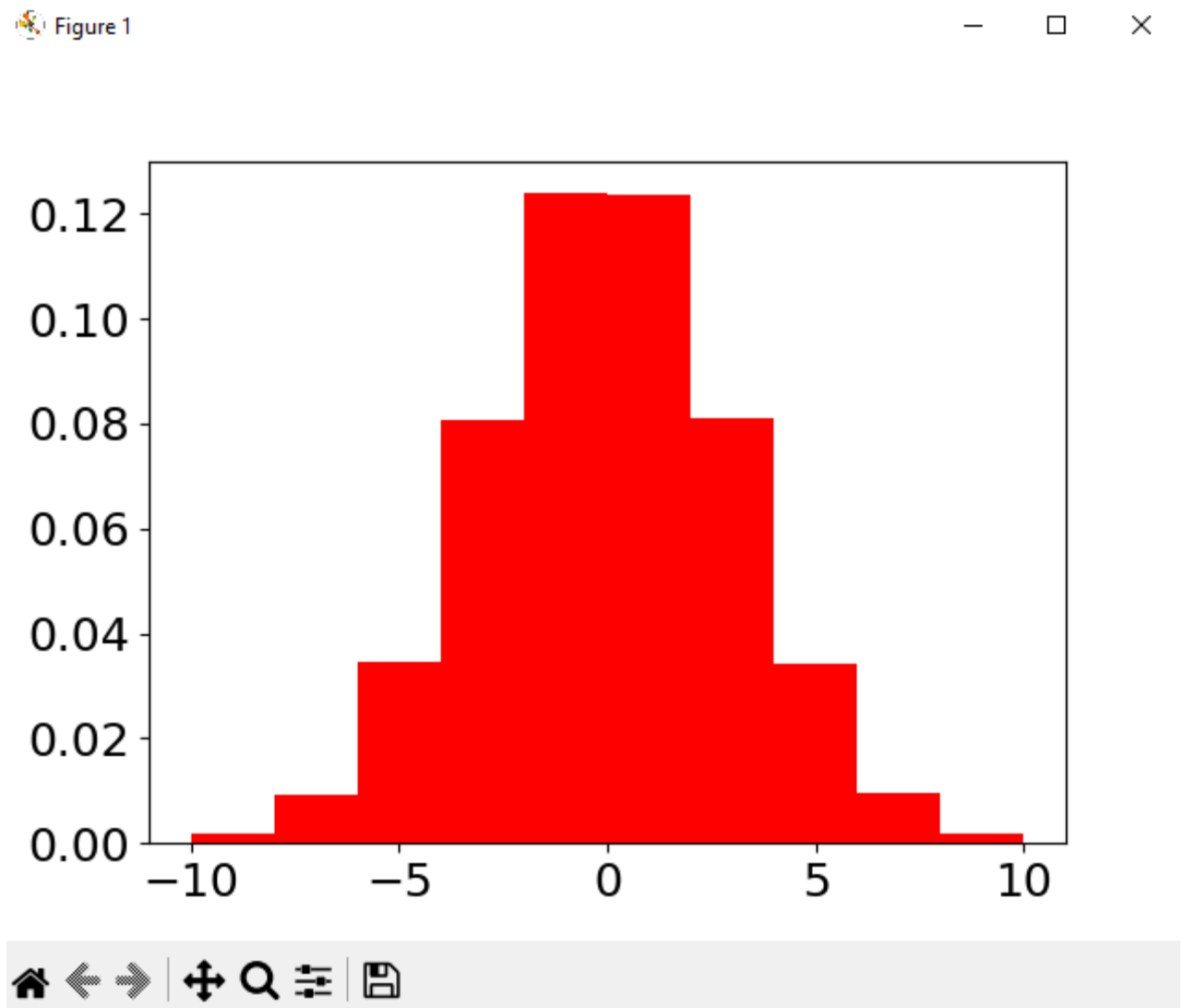


Figure 1
0.12
0.10
0.08
0.06
0.04
0.02
0.00
−10
−5
0
5
10

Histogram

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

plt.rcParams['font.size'] = 18

L = np.random.poisson(5,10**6)    # średnia = 5

mybins = np.arange(-0.5,20,1)

plt.hist(L, color='b', bins=mybins, density=True,
         histtype='step', lw=2)

plt.axis([-1,16,0,0.2])

plt.show()
```

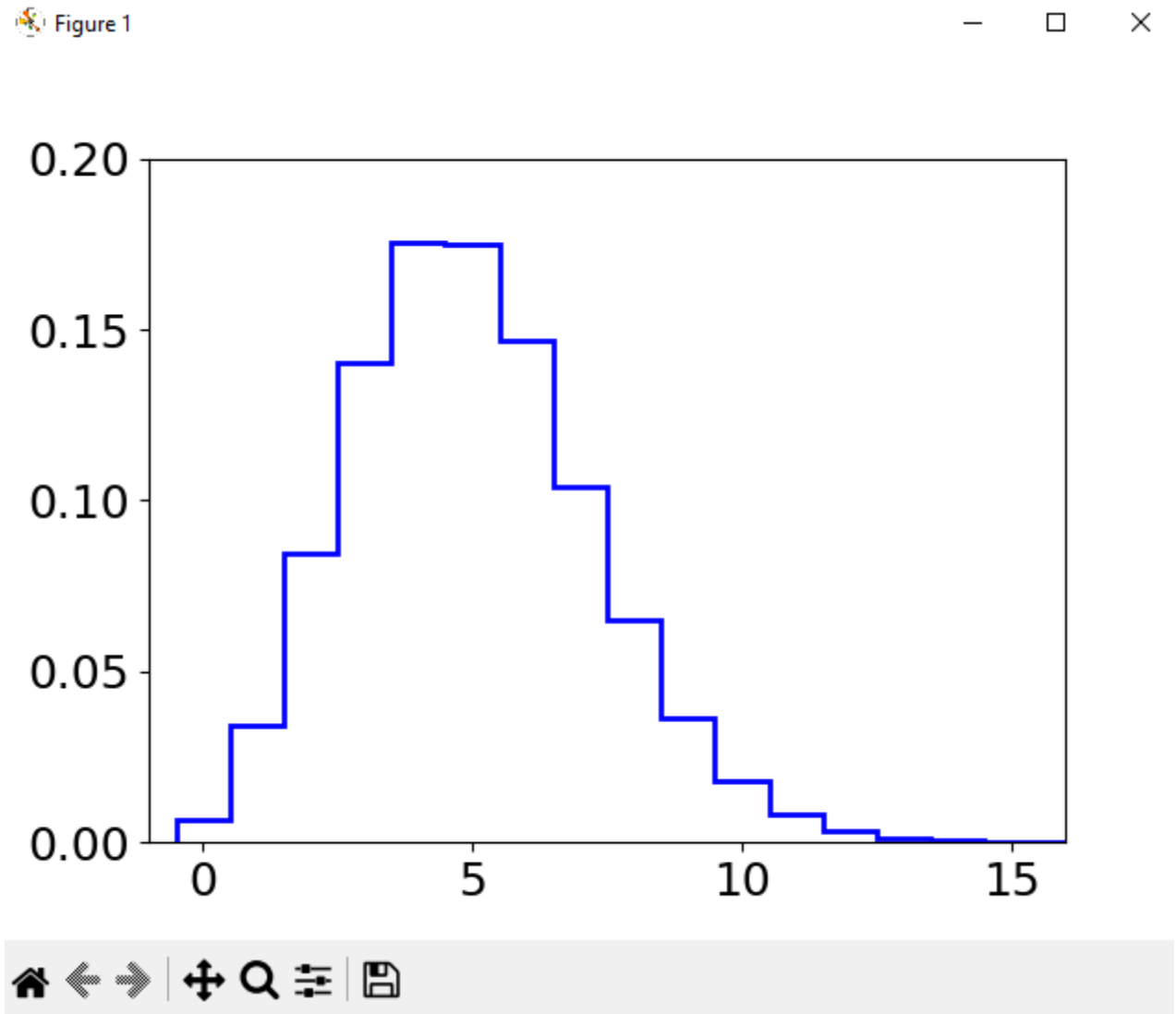


Figure 1
0.20
0.15
0.10
0.05
0.00
0
5
10
15

Subplots

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

plt.rcParams['font.size'] = 14

x = np.arange(0,10,0.1)

plt.subplots_adjust(hspace=0.2, wspace=0.5)

plt.subplot(2,2,1)
plt.plot(x, np.sin(x), 'g-', lw=1.8)

plt.subplot(2,2,2)
plt.semilogy(x, np.exp(x), 'r-', lw=1.8)

plt.subplot(2,2,3)
plt.loglog(x, np.exp(-x), 'k--', lw=2.5)

plt.subplot(2,2,4)
plt.hist(np.random.normal(0,2,10**5), bins=40, density=True)

plt.show()
```

Odległość między rysunkami

dwa wiersze, dwie kolumny, pierwszy rysunek

Logarytmiczna pionowa oś

Wykres podwójnie logarytmiczny

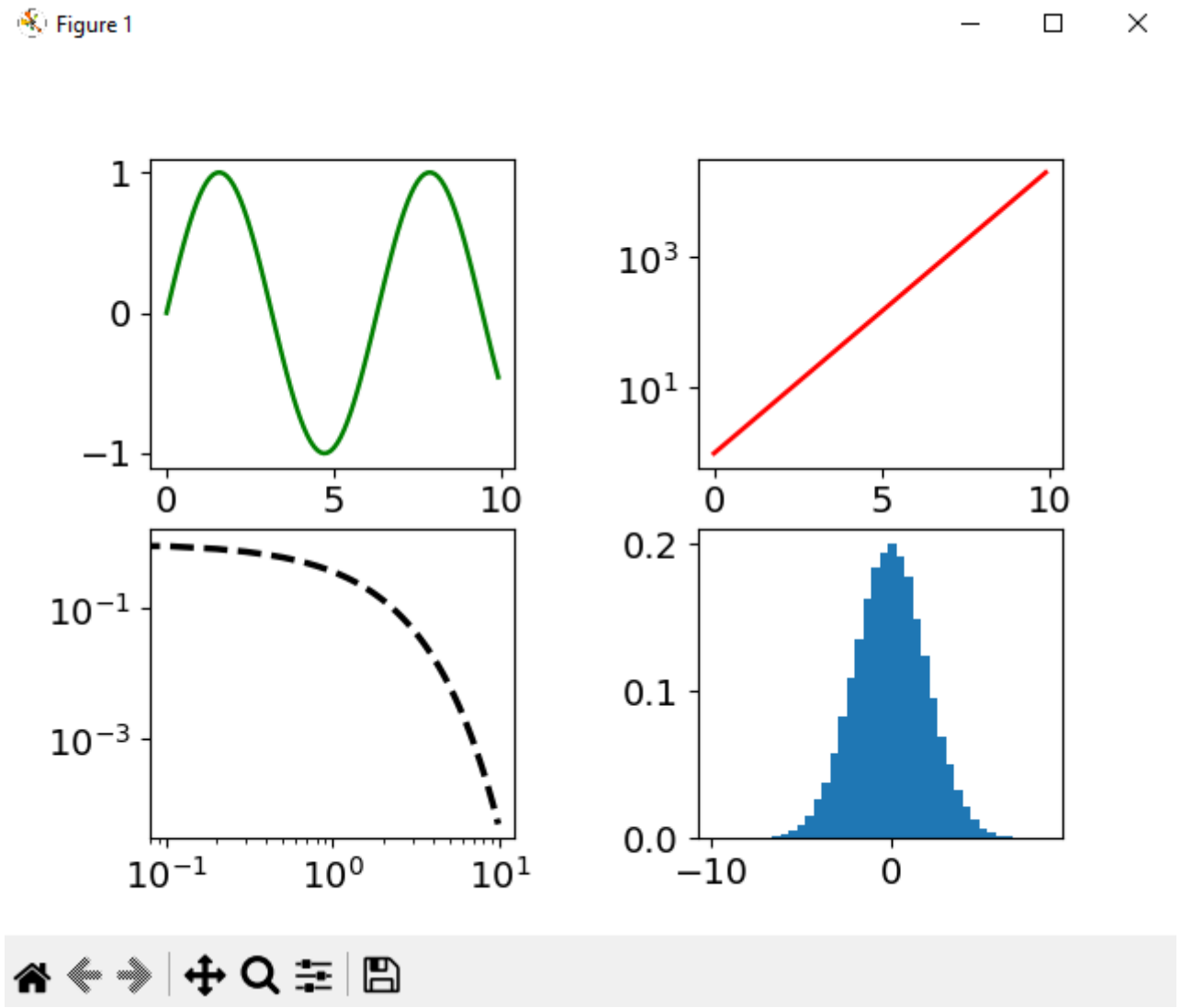


Figure 1
1
0
−1
0
5
10
10^3
10^1
0
5
10
10^{-1}
10^{-3}
10^{-1}
10^0
10^1
0.2
0.1
0.0
−10
0

meshgrid

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

xvalues = np.array([0, 1, 2, 3, 4])
yvalues = np.array([0, 1, 2, 3, 4])

xx, yy = np.meshgrid(xvalues, yvalues)

plt.plot(xx, yy, marker='.', color='k', linestyle='none')

plt.show()
```

Tworzy wszystkie pary (xvalues, yvalues)

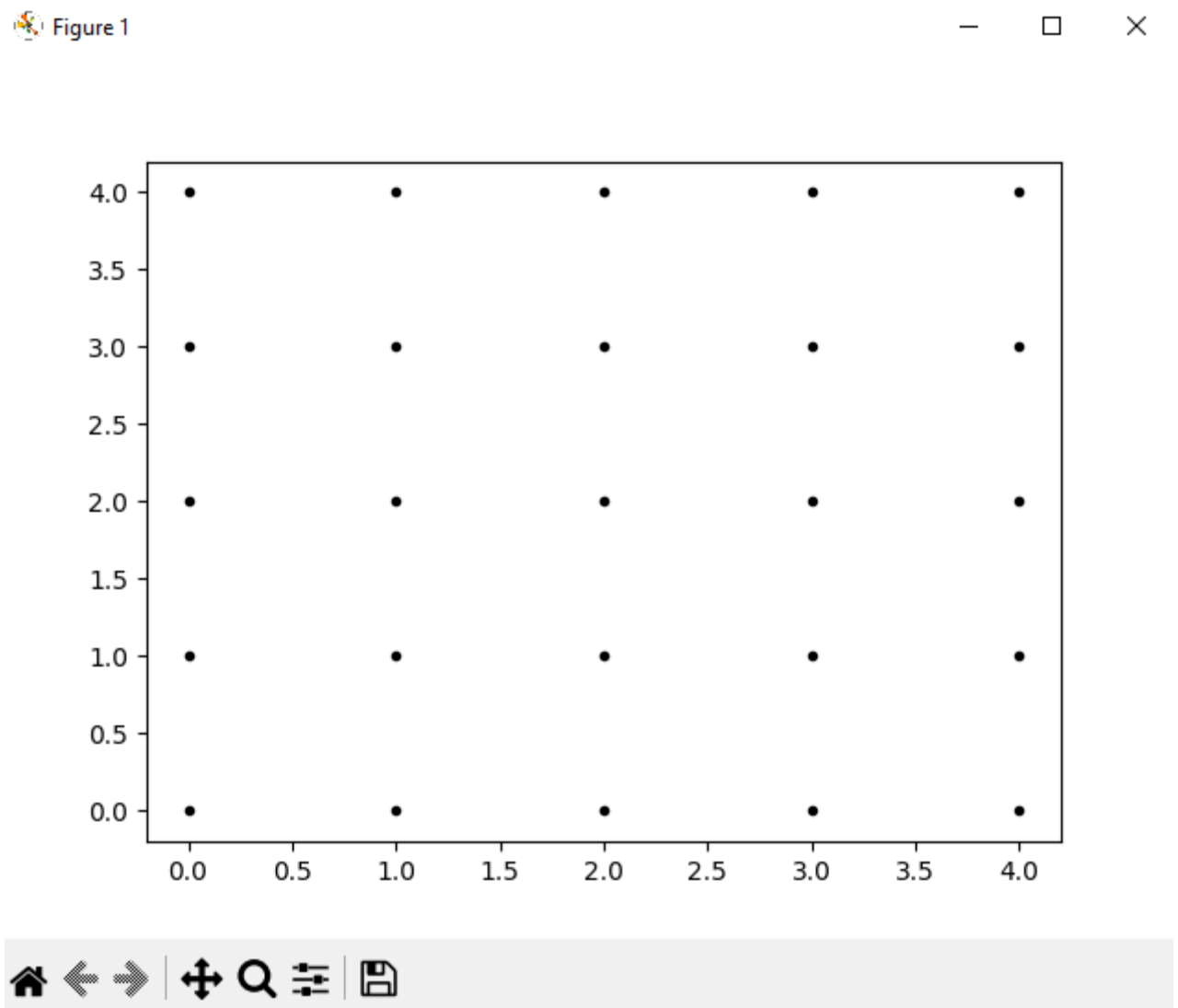


Figure 1
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0

2D, contour plot

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

plt.rcParams['font.size']=14

x = np.linspace(0,2,50)
y = np.linspace(0,2,50)

(X,Y) = np.meshgrid(x,y)
Z = np.exp(-X**2) * np.exp(-Y**4)

levels = [0.03, 0.1, 0.3, 0.5, 0.7, 0.9, 0.99]

a = plt.contour(X,Y,Z,levels,linewidths=1.8)

plt.clabel(a)

plt.xlabel('x')
plt.ylabel('y')
plt.axis([0,2,0,2])
plt.show()
```

kontury

clabel podaje wartości konturów na rysunku, tutaj 0.03,0.1,0.3 etc.

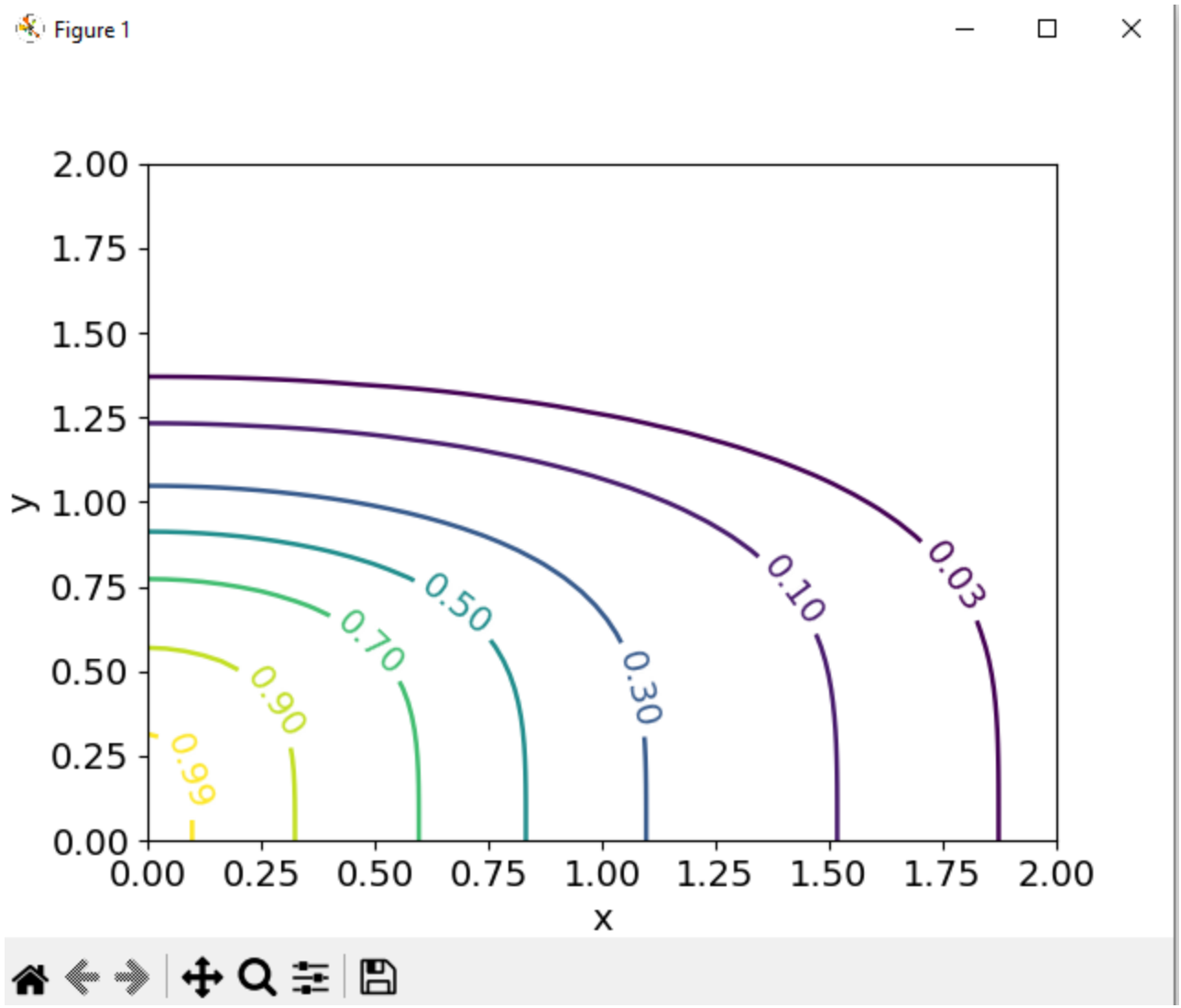


Figure 1
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
y
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
x
0.03
0.10
0.30
0.50
0.70
0.90
0.99

2D, filled contour

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

plt.rcParams['font.size']=14

x = np.linspace(0,2,50)
y = np.linspace(0,2,50)

(X,Y) = np.meshgrid(x,y)
Z = np.exp(-X**2) * np.exp(-Y**4)

levels = np.linspace(0,1,11)

plt.contourf(X,Y,Z,levels,cmap='YlGnBu')

plt.colorbar(orientation='vertical')

plt.axis([0,2,0,2])
plt.show()
```

Mapa kolorów, tu: żółto-zielono-niebieska

spróbuj: orientation='horizontal'

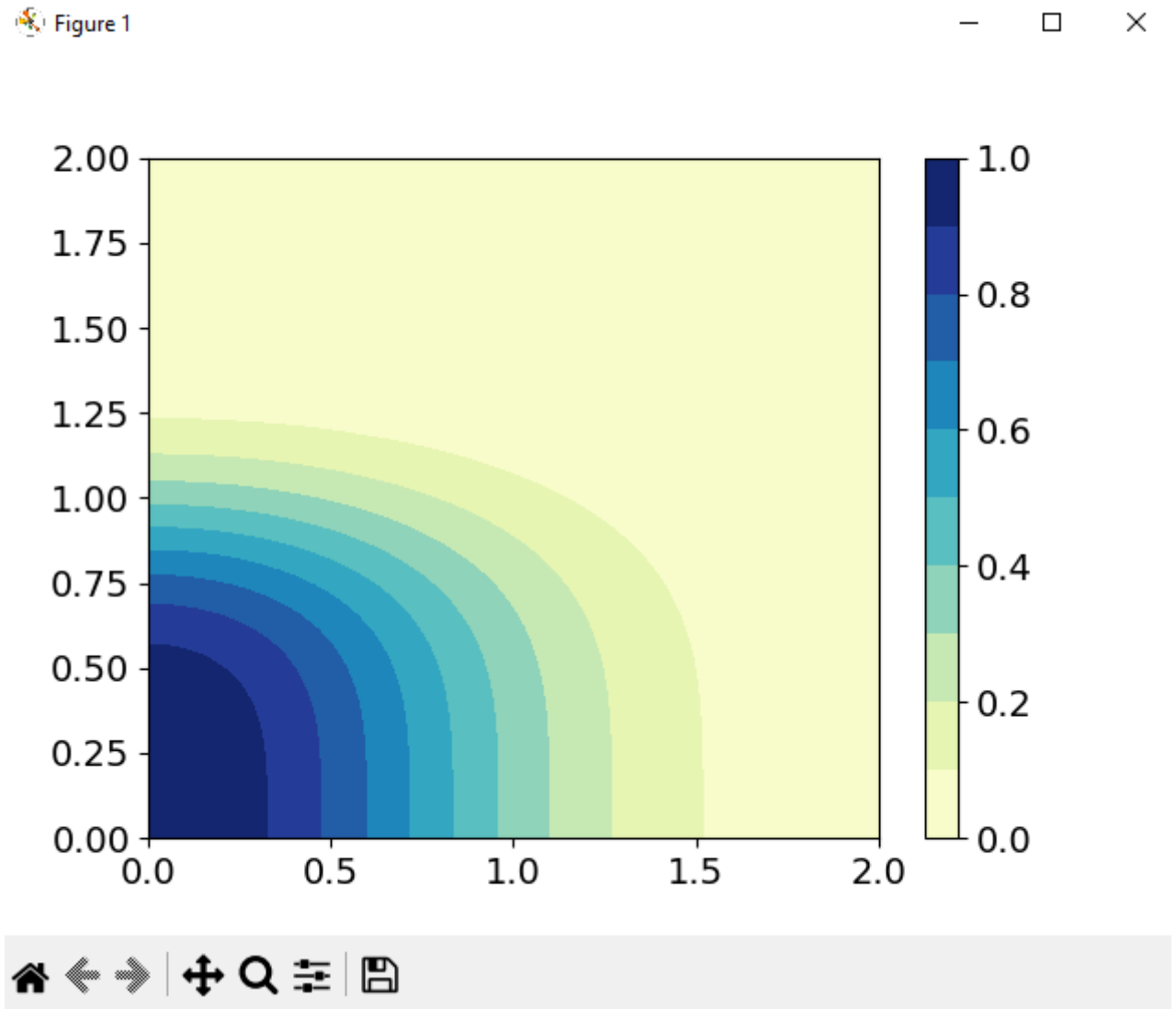


Figure 1
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0.0
0.5
1.0
1.5
2.0
1.0
0.8
0.6
0.4
0.2
0.0

2D plot

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

plt.rcParams['font.size']=14

x = np.linspace(0,2,50)
y = np.linspace(0,2,50)

(X,Y) = np.meshgrid(x,y)
Z = np.exp(-X**2) * np.exp(-Y**4)

plt.pcolor(X,Y,Z)

plt.colorbar(ticks=np.linspace(0,1,5))   # !!!

plt.axis([0,2,0,2])
plt.show()
```

można dodać cmap

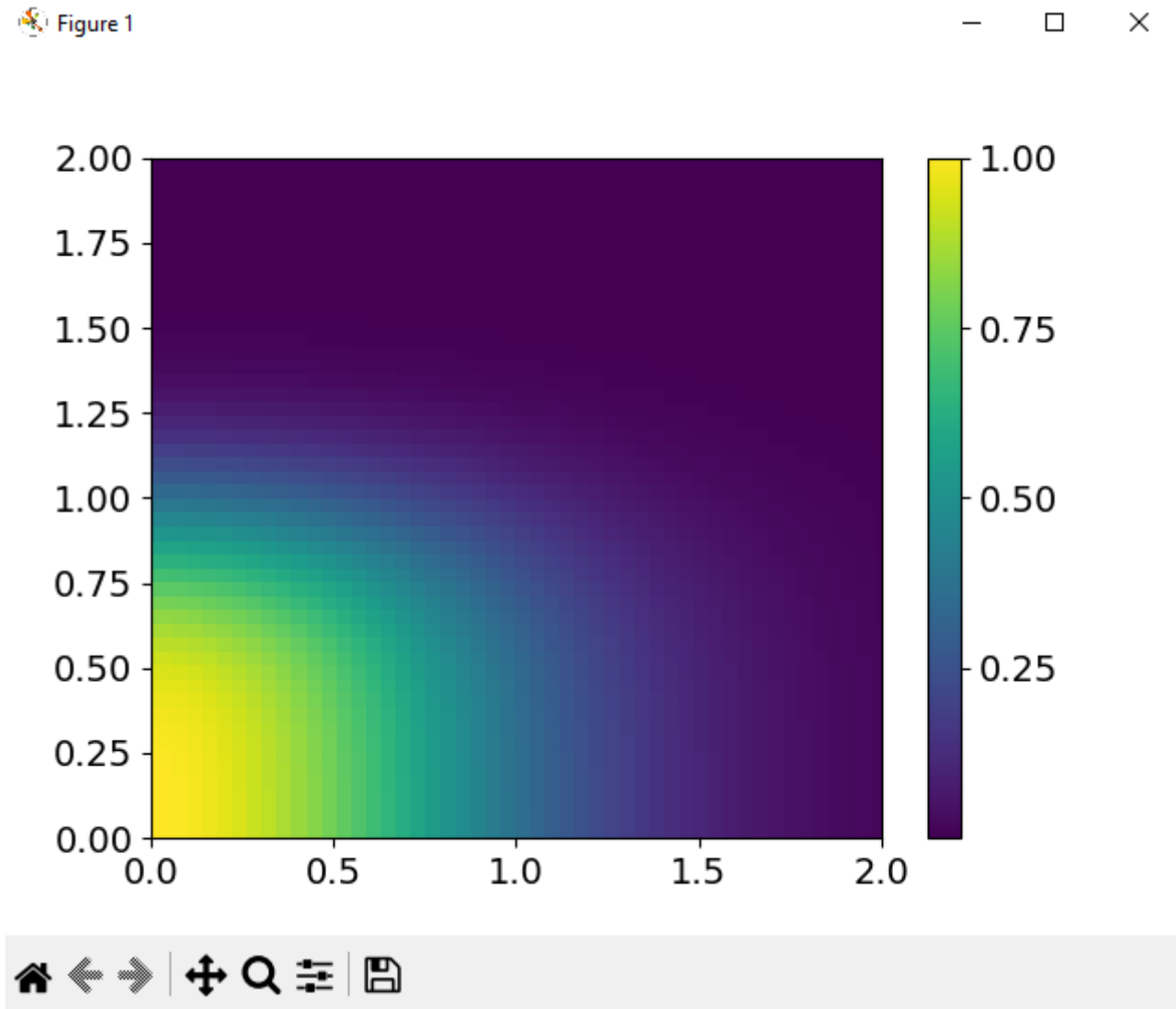


Figure 1
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0.0
0.5
1.0
1.5
2.0
1.00
0.75
0.50
0.25

2D histogram

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

plt.rcParams['font.size'] = 14
plt.rcParams['xtick.direction'] = 'out'    # !!!
plt.rcParams['ytick.direction'] = 'out'    # !!!

x = np.random.random(1000)
y = np.random.random(1000)

plt.hist2d(x,y,bins=15,density=True,cmap='PuBu')

plt.colorbar(ticks=np.linspace(0,2.5,6))

plt.show()
```

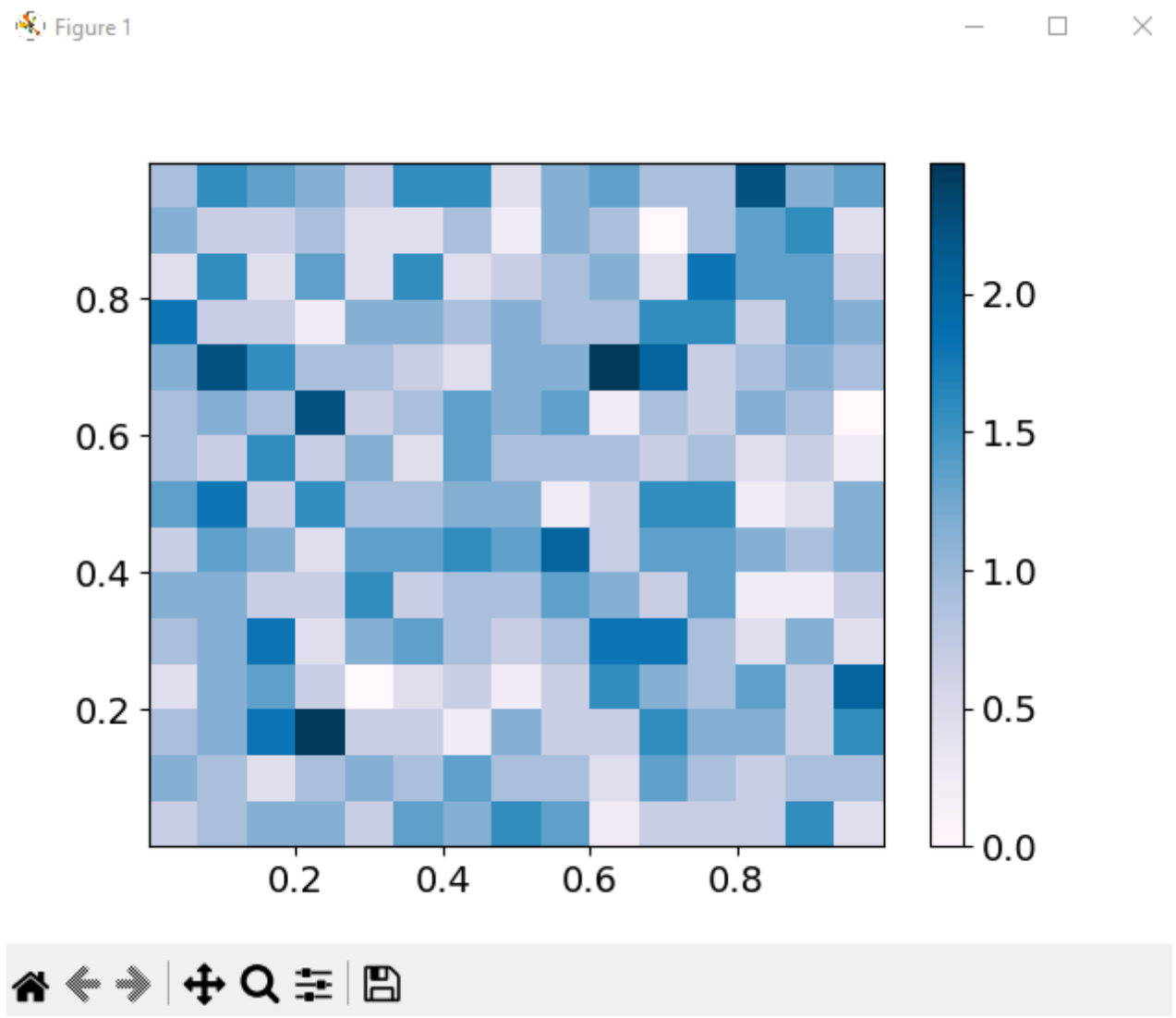
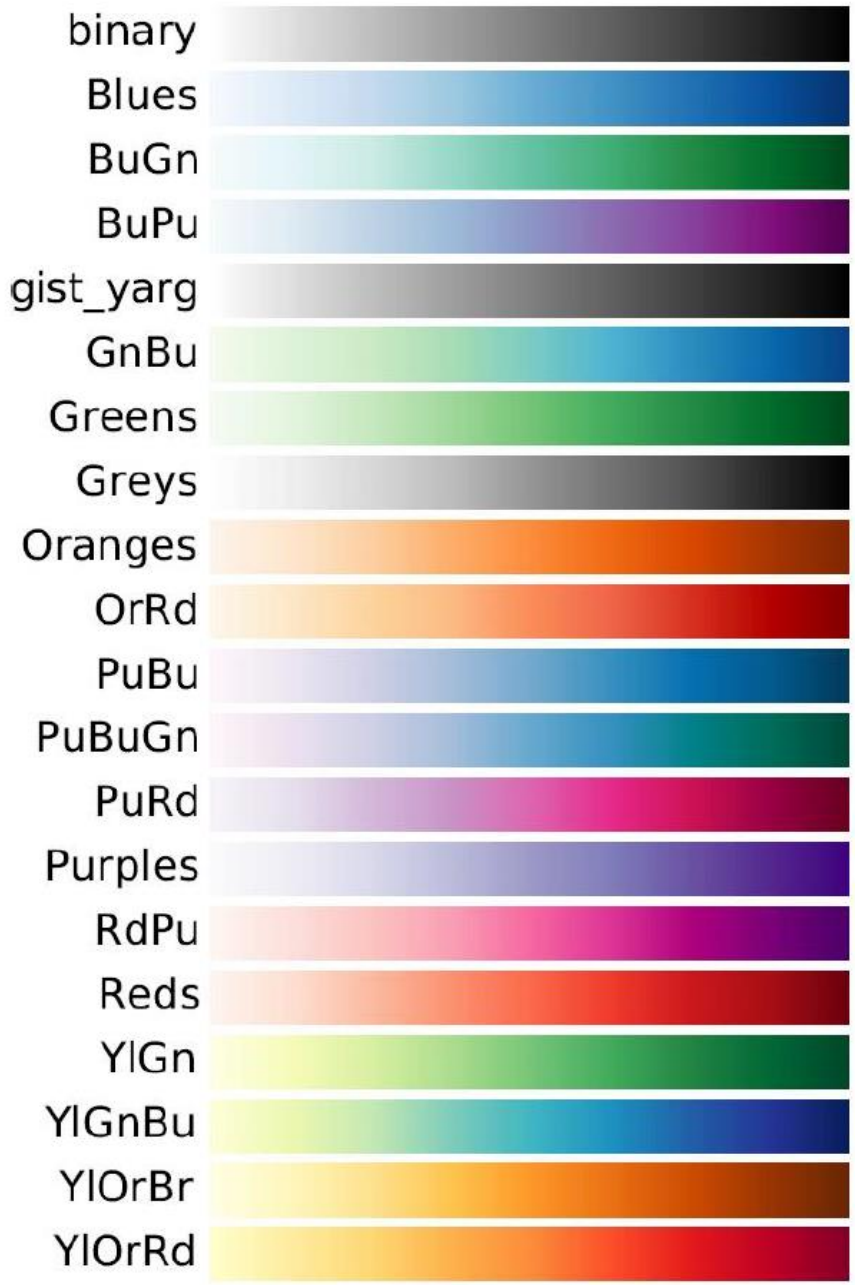
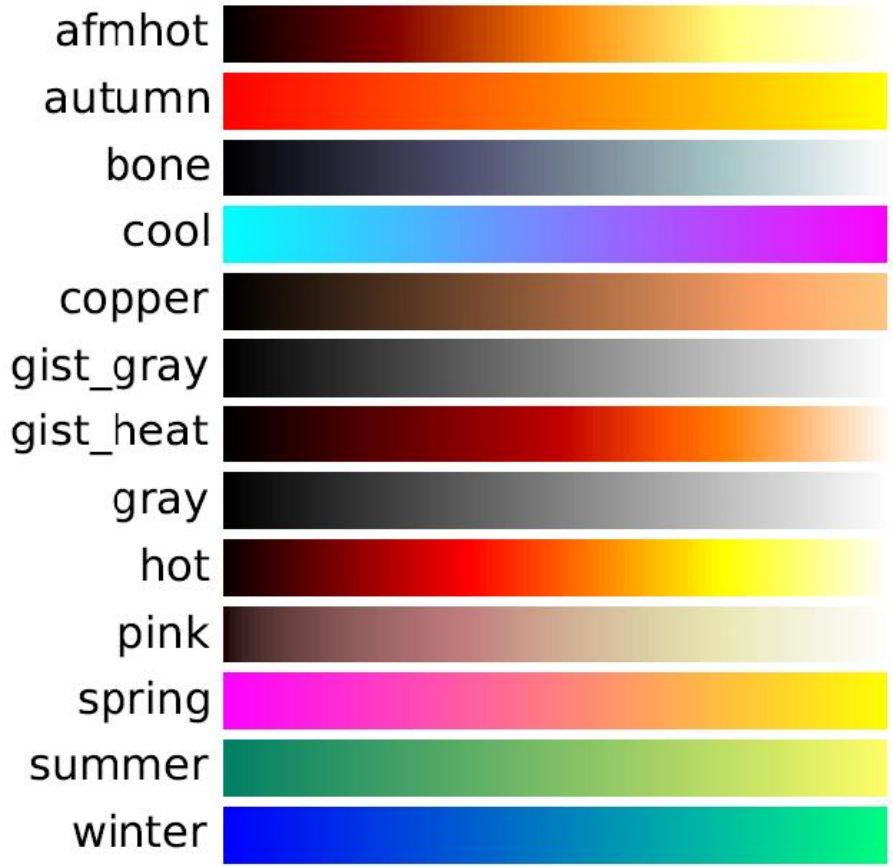
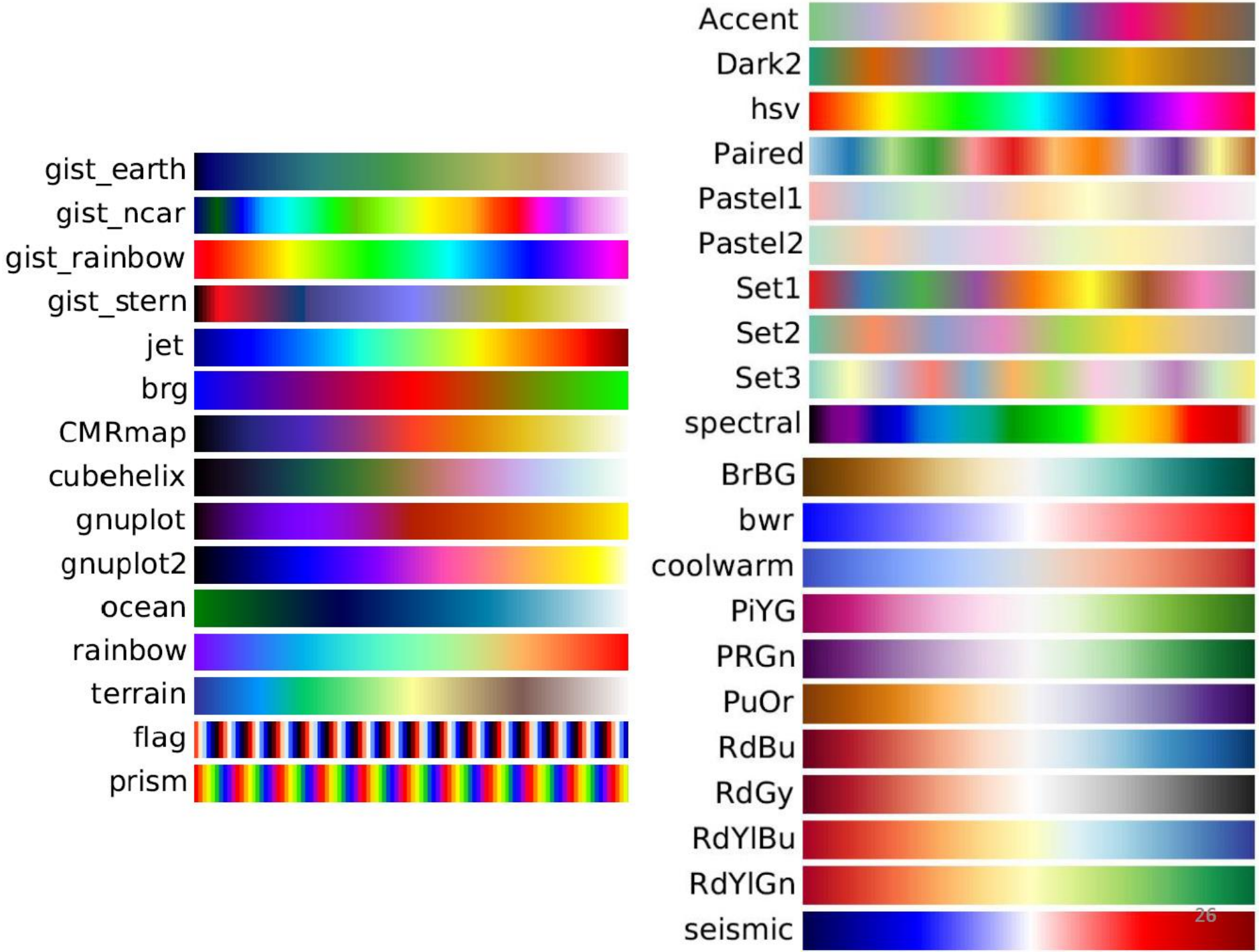


Figure 1
0.8
0.6
0.4
0.2
0.2
0.4
0.6
0.8
2.0
1.5
1.0
0.5
0.0

Color map


binary
Blues
BuGn
BuPu
gist_yarg
GnBu
Greens
Greys
Oranges
OrRd
PuBu
PuBuGn
PuRd
Purples
RdPu
Reds
YlGn
YlGnBu
YlOrBr
YlOrRd


afmhot
autumn
bone
cool
copper
gist_gray
gist_heat
gray
hot
pink
spring
summer
winter


gist_earth
gist_ncar
gist_rainbow
gist_stern
jet
brg
CMRmap
cubehelix
gnuplot
gnuplot2
ocean
rainbow
terrain
flag
prism
Accent
Dark2
hsv
Paired
Pastel1
Pastel2
Set1
Set2
Set3
spectral
BrBG
bwr
coolwarm
PiYG
PRGn
PuOr
RdBu
RdGy
RdYlBu
RdYlGn
seismic