clase 22 de agosto 2023

Susana Hernández

Invalid Date

Table of contents

Preface **3**

1 Introduction **4**

2 Summary **5**

3 plt.show() **6**

References **17**

Preface

This is a Quarto book.

To learn more about Quarto books visit https://quarto.org/docs/books.

1 Introduction

This is a book created from markdown and executable code.

See Knuth (1984) for additional discussion of literate programming.

2 Summary

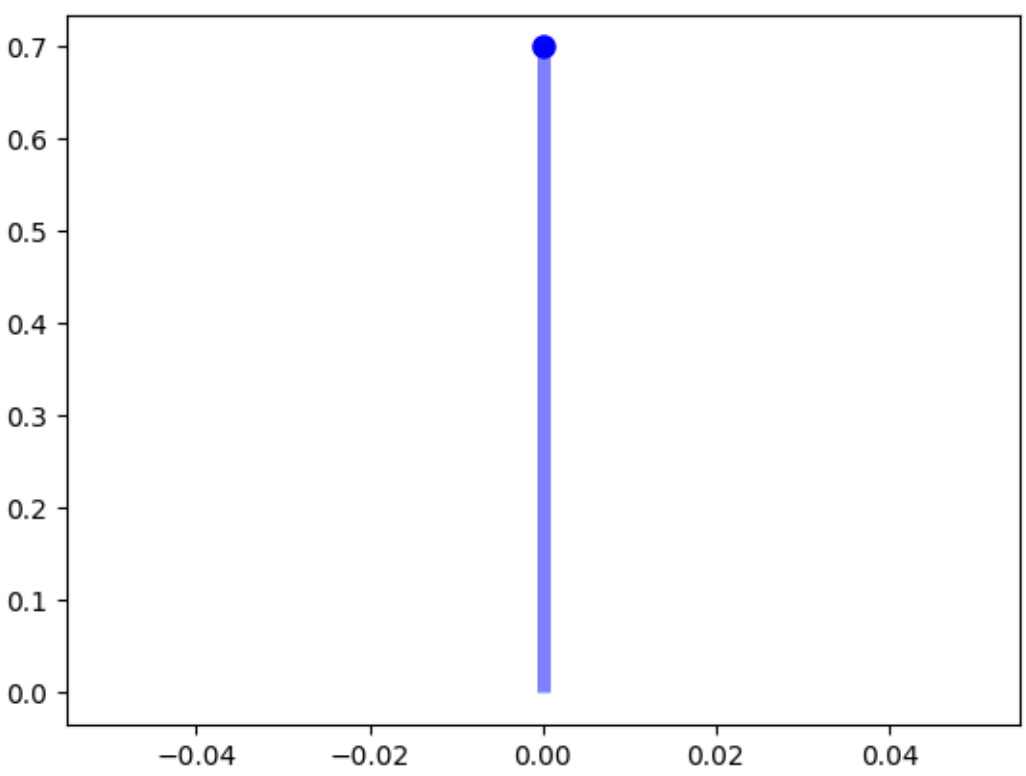
In summary, this book has no content whatsoever.

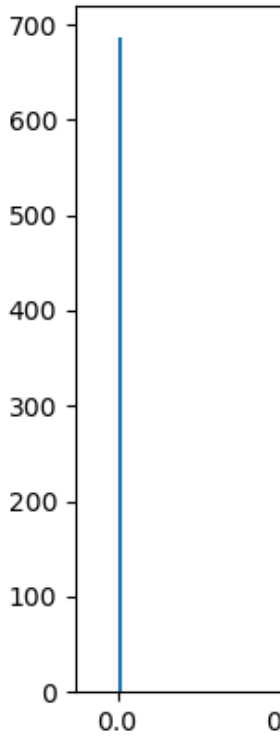
3 plt.show()

Listing 3.1 Exploring functions to generate random variables with a Bernoulli distribution.py

```
import numpy as np
from scipy.stats import bernoulli
import matplotlib.pyplot as plt
fig_01, ax_01 = plt.subplots(1, 1)
fig_02, ax_02 = plt.subplots(1, 1)
p = 0.3
mean, var, skew, kurt = bernoulli.stats(p, moments='mvsk')
print(mean, var, skew,kurt)

x = np.arange(bernoulli.ppf(0.01, p),
              bernoulli.ppf(0.99, p))
ax_01.plot(x, bernoulli.pmf(x, p), 'bo', ms=8, label='bernoulli pmf')
ax_01.vlines(x, 0, bernoulli.pmf(x, p), colors='b', lw=5, alpha=0.5)
r = bernoulli.rvs(p, size=1000)
ax_02.hist(r, bins=200)
plt.show()
```


0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−0.04
−0.02
0.00
0.02
0.04


700
600
500
400
300
200
100
0
0.0

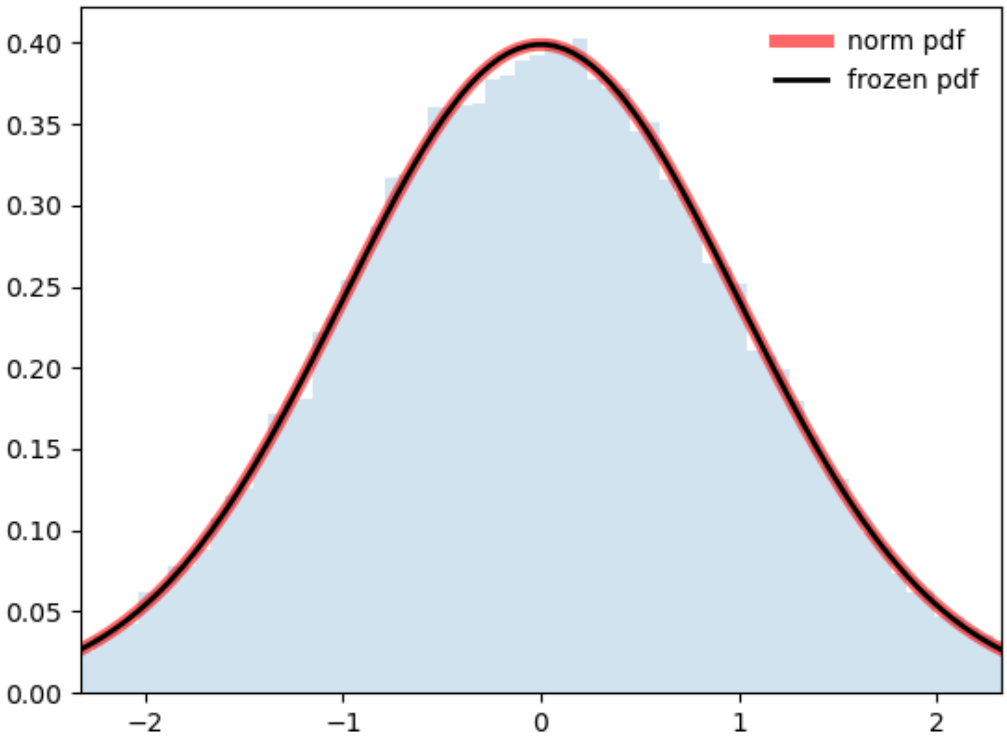
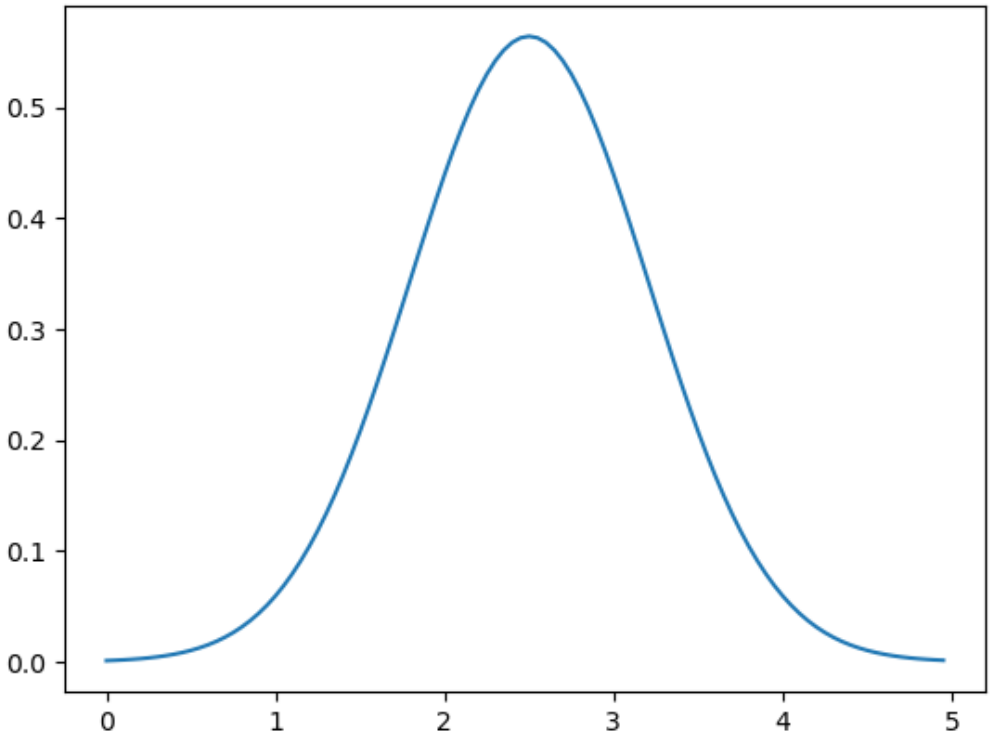
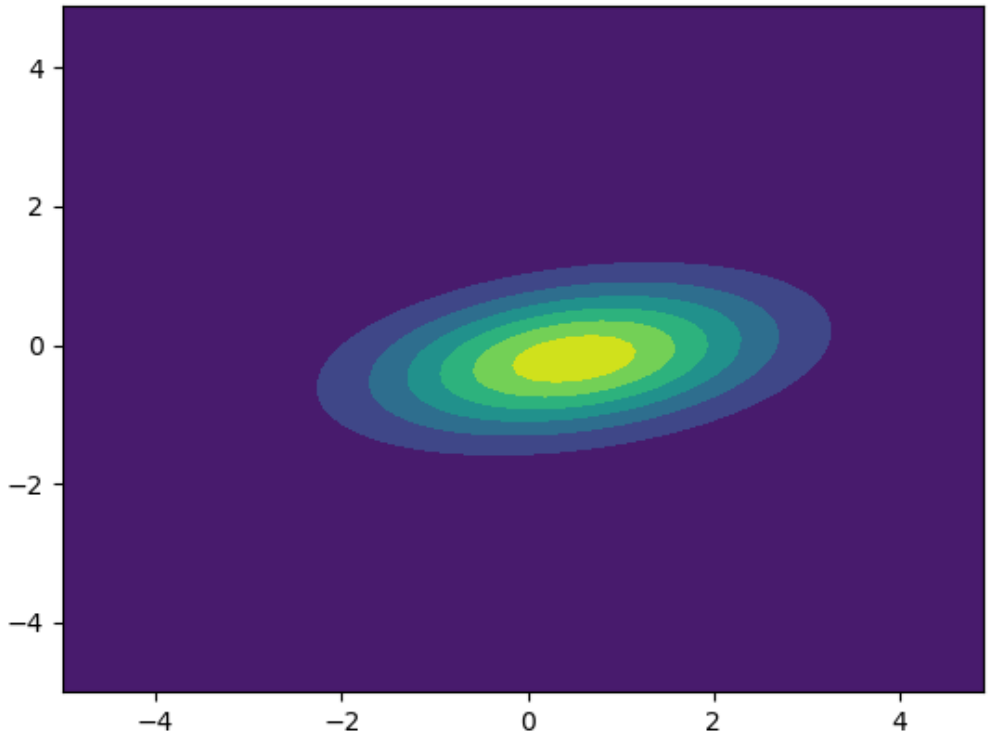
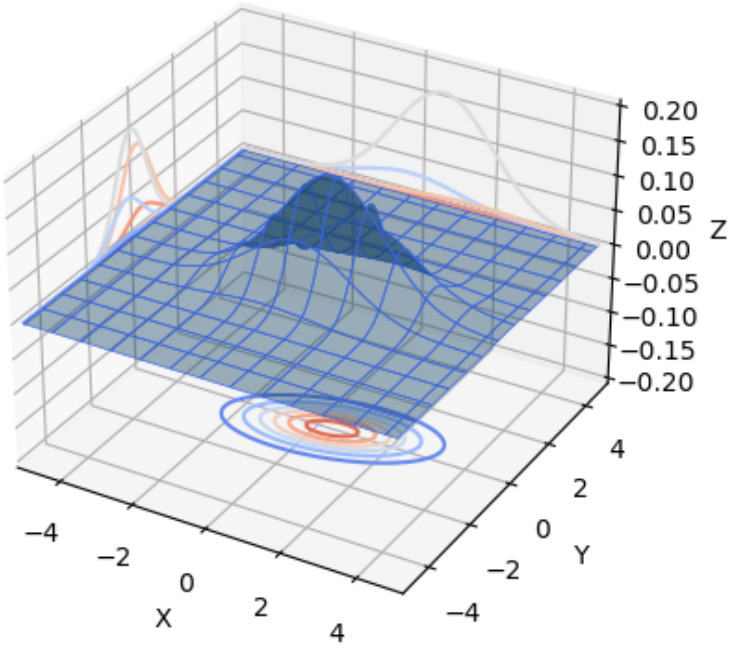


Figure 3.1: Figura 3


0.5
0.4
0.3
0.2
0.1
0.0
0
1
2
3
4
5


4
2
0
−2
−4
−4
−2
0
2
4


0.20
0.15
0.10
0.05
0.00
−0.05
−0.10
−0.15
−0.20
Z
−4
−2
0
2
4
X
4
2
0
−2
−4
Y

Listing 3.2 Exploring functions to generate random variables with a Gaussian distribution.py

Listing 3.3 Revising multivariate Gaussian.py

```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import axes3d
from scipy.stats import multivariate_normal

x = np.linspace(0, 5, 100, endpoint=False)
y = multivariate_normal.pdf(x, mean=2.5, cov=0.5);

fig1 = plt.figure()
ax = fig1.add_subplot(111)
ax.plot(x, y)

x, y = np.mgrid[-5:5:.1, -5:5:.1]
pos = np.dstack((x, y))
rv = multivariate_normal([0.5, -0.2], [[2.0, 0.3], [0.3, 0.5]])
fig2 = plt.figure()
ax2 = fig2.add_subplot(111)

ax2.contourf(x, y, rv.pdf(pos))
# plt.show()

ax = plt.figure().add_subplot(projection='3d')
ax.plot_surface(
    x,
    y,
    rv.pdf(pos),
    edgecolor='royalblue',
    lw=0.5,
    rstride=8,
    cstride=8,
    alpha=0.4
)
ax.contour(x, y, rv.pdf(pos), zdir='z', offset=-.2, cmap='coolwarm')
ax.contour(x, y, rv.pdf(pos), zdir='x', offset=-5, cmap='coolwarm')
ax.contour(x, y, rv.pdf(pos), zdir='y', offset=5, cmap='coolwarm')


ax.set(
    xlim=(-5, 5),
    ylim=(-5, 5),
    zlim=(-0.2, 0.2),
    xlabel='X',
    ylabel='Y',
    zlabel='Z'
)
plt.show()
```

4

5

6

References

Knuth, Donald E. 1984. “Literate Programming.” *Comput. J.* 27 (2): 97–111. https://doi.org/10.1093/comjnl/27.2.97.