COMP 6245 Report for Lab one 1

RUNYING JIANG (rj1u20@soton.ac.uk)

January 20, 2021

Preliminaries

Question 1.

The answer I got from the last command(np.dot(U[:,0], U[:,1])) is
-2.7755575615628914e-16 , which is close to zero. This is the result I want from Matrix B.
as $B = B.T$, Matrix B is a **Symmetric Matrix**
for **Symmetric Matrix**, all of the eigenvectors are perpendicular to each other.
as $U \cdot U^T = I$, which we got from U @ U.T,the set of eigenvectors Matrix U is orthogonal.

Random Numbers and Uni-variate Densities

Question 2.

- Because Matrix X is random, the count in different intervals are still different.
- Because of the pseudo random seed
- The more the data, the flatter the histogram

Question 3.

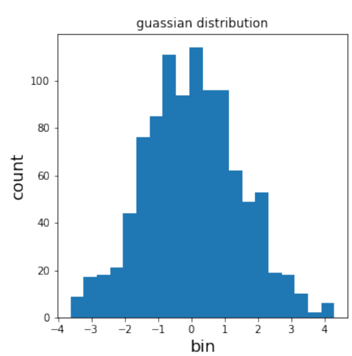


Figure 1: gaussian distribution

- I suppose the histogram in Figure 2 is a Gaussian Distribution.

- The mean of gaussian distribution keeps the same while the variance becomes larger when I add the number and becomes smaller when I substract.
- The theory
  $$X \sim \mathcal{N}(\mu_1, \sigma_1^2) + X \sim \mathcal{N}\mu_2, \sigma_2^2) = X \sim \mathcal{N}(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2)$$
  probably could explain my observation.

Uncertainty in Estimation

There is a line spacing error in the sample code. The right code should be :

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

MaxTrial = 2000
sampleSizeRange = np.linspace(100,200,40)
plotVar = np.zeros(len(sampleSizeRange))
for sSize in range(len(sampleSizeRange)):
    numSamples = int(sampleSizeRange[sSize])
    vStrial = np.zeros(MaxTrial)
    for trial in range(MaxTrial):
        xx = np.random.randn(numSamples,1)
        vStrial[trial] = np.var(xx)
    plotVar[sSize] = np.var(vStrial)
fig, ax = plt.subplots(figsize=(4,4))
ax.plot(plotVar)
```

Figure 3 is what I got from by running the code above, which shows the variance of variance gets smaller as we had more data.

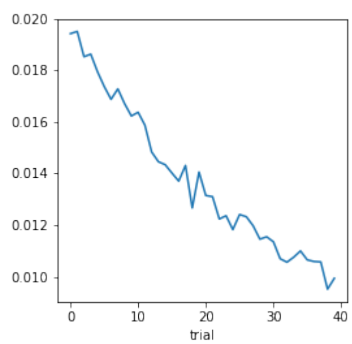


Figure 2: plot Variation

Bi-variate Gaussian Distribution

Compared to the formula of multivariate Gaussian. I made some changes to the code:

```
def gauss2D(x,m,C):
    Ci = np.linalg.inv(C)
```

```
dC = np.linalg.det(C)**(1/2)
#dC = np.linalg.det(C1)
num = np.exp(-0.5 * np.dot((x-m).T, np.dot(Ci,(x-m))))
den = (2* np.pi)**(1/2) *dC
# den = 2*np.pi*dC
return num/den
```

The mesh plot I got from using package Axes3D are as follow:

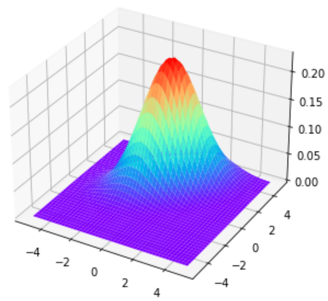
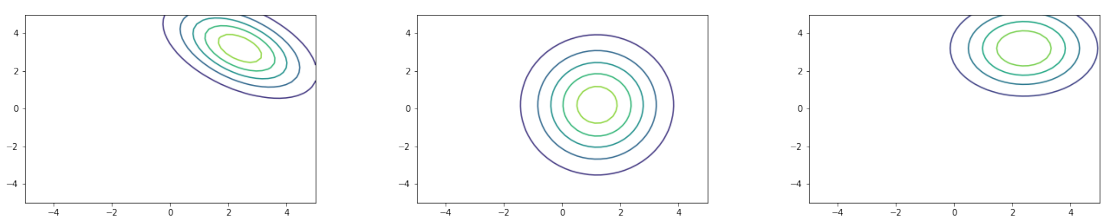


Figure 3: mesh$_{plot}$

Question 4.

My contours of the distributions $N\left(\begin{bmatrix}2.4\\3.2\end{bmatrix}\begin{bmatrix}2 & -1\\-1 & 2\end{bmatrix}\right)$, $N\left(\begin{bmatrix}1.2\\0.2\end{bmatrix}\begin{bmatrix}2 & 0\\0 & 4\end{bmatrix}\right)$ and $N\left(\begin{bmatrix}2.4\\3.2\end{bmatrix}\begin{bmatrix}2 & 0\\0 & 2\end{bmatrix}\right)$ are as follow:



(a) (b) (c)

Figure 4: result of bi-variate Gaussian Distribution

Sampling from a multi-variate Gaussian

Figure 5 shows the scatter plot I got from 5000 bivariate Gaussian random Data X, Y:

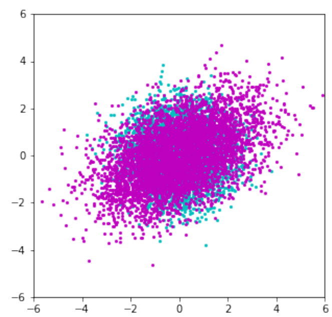


Figure 5: sampling Gaussian

Distribution of Projections

The maxima of the plot is 2.9637 when theta = 50.4
The minima of the plot is 0.9869 when theta = 316.8
Figure 6 below shows the resulting plot:

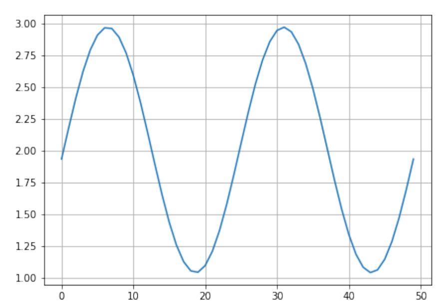


Figure 6: variance of Projection

Question 5.

- The eigenvalues of the covariance Matrix C are 3 and 1

  The eigenvectors of the covariance Matrix C are k $\begin{bmatrix} \sqrt(2)/2 \\ \sqrt(2)/2 \end{bmatrix}$ and k $\begin{bmatrix} -\sqrt(2)/2 \\ \sqrt(2)/2 \end{bmatrix}$
- As I changed the Matrix C ,I found that eigenvalues are the maximum and minimum in the plot, which is the factor stretched and squashed the plot during transformation.
- The theory in The Objective part could confirm it might be true:

  $$x \sim \mathcal{N}(m, C), y = Ax \rightarrow y \sim \mathcal{N}(Am, ACA^T)$$

  as the theory, the variance is uCu^T which is

  $$\begin{bmatrix} \sin\theta \\ \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} \sin\theta \\ \cos\theta \end{bmatrix} = 2 + \sin(2\theta)$$