Preliminaries

What do you observe for the last command above? Can you formally prove that this is the result you would expect for the specific structure in the matrix B?

I found that **np.dot(U[:,0], U[:,1])** $\approx$ **0**, which means that the dot product of the first and second column of eigenvector matrix ≈ 0. So I could assume that the product of eigenvectors from the symmetric matrix is orthogonal.
Let v_1, v_2 be two eigenvectors of matrix B and λ_1, λ_2 be the corresponding two eigenvalues. So

$$B \cdot v_1 = \lambda_1 v_1, B \cdot v_2 = \lambda_2 v_2$$

$$v_2^\top \cdot B \cdot v_1 = \lambda_1 v_2^\top \cdot v_1$$

Since B is symmetric, so $B = B^\top$

$$v_2^\top \cdot B \cdot v_1 = v_2^\top \cdot B^\top \cdot v_1 = (B \cdot v_2)^\top \cdot v_1 = \lambda_2 v_2^\top \cdot v_1$$

$$\lambda_2 v_2^\top \cdot v_1 = \lambda_1 v_2^\top \cdot v_1$$

Since $\lambda_1 \neq \lambda_2 \neq 0$

$$v_2^\top \cdot v_1 = 0$$

$$v_1 \cdot v_2 = 0$$

So we could draw a conclusion that eigenvetors of a symmetric matrix are orthogonal.

1 Random Numbers and Uni-variate Densities

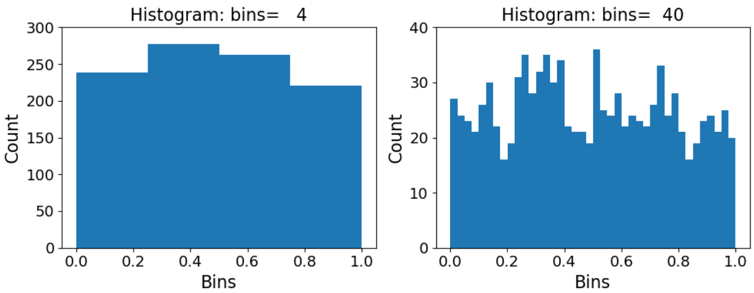

Figure 1: 1000 uniform random numbers

Though the data is from a uniform distribution, the histogram does not appear flat. Why?

That is because we are sampling from the uniform distribution, and when the amount of data is small, the noise plays a large role to make it uneven. When the amount of data is very large, the data we obtain will significantly satisfy the uniform distribution, thus making the histogram more flat.

Every time you run it, the histogram looks slightly different? Why?

Yes. The histogram looks slightly different is because the data used to draw the histogram is regenerated every time randomly by the function **np.random.rand()**.

Do the above observations change (if so how) if you had started with more data (i.e. 100, 000 instead of 1000?)

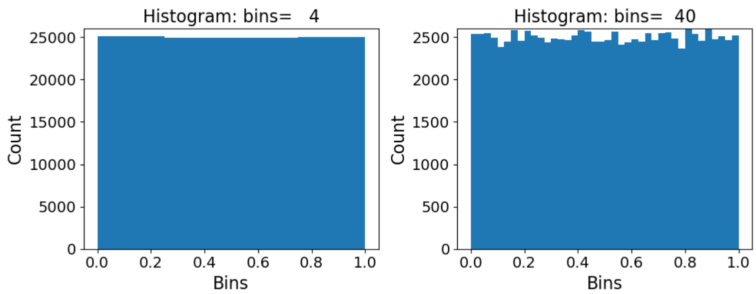

Figure 2: 100000 uniform random numbers

Yes, it tends to be "flatter" from my observation since the scale of the data is greater, the noise from the **np.random.rand()** is relatively smaller.

What do you observe? How does the resulting histogram change when you change the number of uniform random numbers you add and subtract (i.e. fewer numbers than 12)?

The numbers seem to obey Gaussian Distribution, which could be explained by Central limit theorem, that is when groups of random numbers are summed up, the sum of each group tends to subject to Gaussian distribution.
However, when I changed the first parameter of **np.random.rand(x1,x2)**, the histogram didn't change much. But for the parameter **N** which is the number of samples, when I increase the value of it, the distribution obeys Gaussian distribution better as Figure 3 and Figure 4 present.

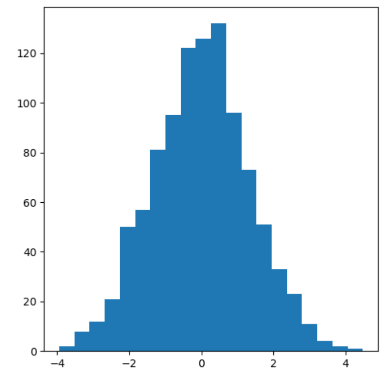

Figure 3: 1000 samples

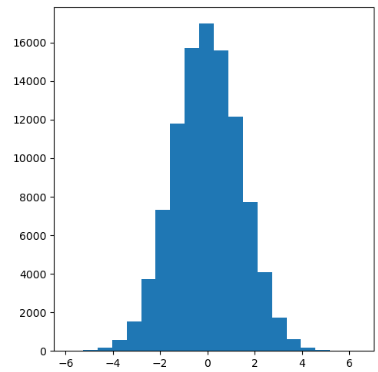

Figure 4: 100000 samples

2 Uncertainty in Estimation

As the sample size in each group gradually increases, the variance of each group begins to gradually decrease, which means that the distribution of the samples obeys the Gaussian distribution better.

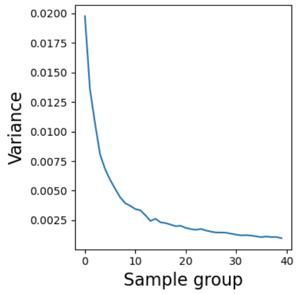

Figure 5: variance as sample size evenly increased

3 Bi-variate Gaussian Distribution

Figure 6 shows the contours of 3 different distributions.

The black contour represents: The blue contour represents: The red contour represents:

$$\mathcal{N}(\begin{bmatrix}2.4\\3.2\end{bmatrix}\begin{bmatrix}2 & -1\\-1 & 2\end{bmatrix}) \qquad \mathcal{N}(\begin{bmatrix}1.2\\0.2\end{bmatrix}\begin{bmatrix}2 & 0\\0 & 4\end{bmatrix}) \qquad \mathcal{N}(\begin{bmatrix}2.4\\3.2\end{bmatrix}\begin{bmatrix}2 & 0\\0 & 2\end{bmatrix})$$

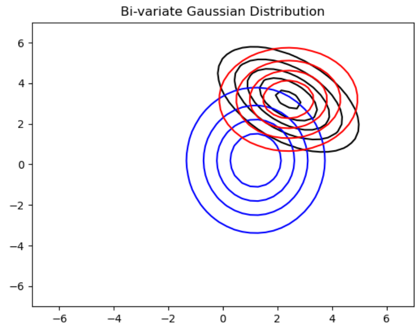

Figure 6: Bi-variate Gaussian Distribution

4 Sampling from a multi-variate Gaussian

Figure 7 shows an isotropic Gaussian distribution and its linear transformed distribution(which is a correlated Gaussian distribution, by using a covariance matrix).

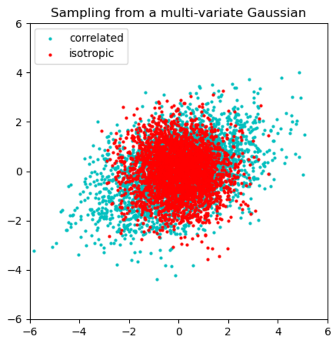

Figure 7: Sampling from a multi-variate Gaussian

5 Distribution of Projections

Figure 8 shows that variance changes as theta changes.

What are the maxima and minima of the resulting plot?

The maxima and minima of the resulting plot are 3.02 and 1.00 respectively.

Compute the eigenvalues an eigenvectors of the covariance matrix C

By using **np.linalg.eig(C)**, we could get the eigenvalues λ_1, λ_2 and eigenvectors μ_1, μ_2 of the matrix C:

$$\lambda_1 = 3, \lambda_2 = 1$$

$$\mu_1 = \begin{bmatrix} 0.70710678 & 0.70710678 \end{bmatrix}, \mu_2 = \begin{bmatrix} -0.70710678 & 0.70710678 \end{bmatrix}$$

Can you see a relationship between the eignevalues and eigenvectors and the maxima and minima of the way the projected variance changes?

The larger eigenvalue is always close to the maxima while the smaller one is close to the minima.

The shape of the graph might have looked sinusoidal for this two dimensional problem. Can you analytically confirm if this might be true?

Since $yp = uY, Y \sim N(m, C)$

$$yp \sim N(um, uCu^\top)$$

So the variance of yp is:

$$uCu^\top = \begin{bmatrix} sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} sin\theta \\ cos\theta \end{bmatrix} = 2 + sin2\theta$$

That's why the shape of the graph looks sinusoidal.

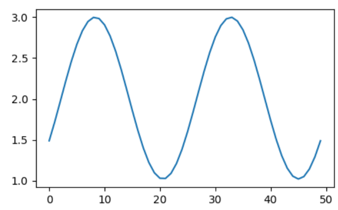

Figure 8: projected variance