CO544 Lab 01

E/19/166

Jayathunga W.W.K.

Preliminaries

12. The command print(np.dot(U[:,0], U[:,1])) computes the dot product of the first and second column vectors of the matrix U. The result is zero. This is expected because the columns of U are eigenvectors of the matrix B, and for a symmetric matrix like B, the eigenvectors corresponding to different eigenvalues are orthogonal. Orthogonal vectors have a dot product of zero.

<u>Proof</u>

B is symmetric and λ1 and λ2 (with λ1 ≠ λ2) are eigenvalues of B with corresponding eigenvectors v1 and v2, then v1 and v2 are orthogonal.

Bv1 = λ1v1 and Bv2 = λ2v2

(Bv1) . v2 = λ1 (v1 . v2)

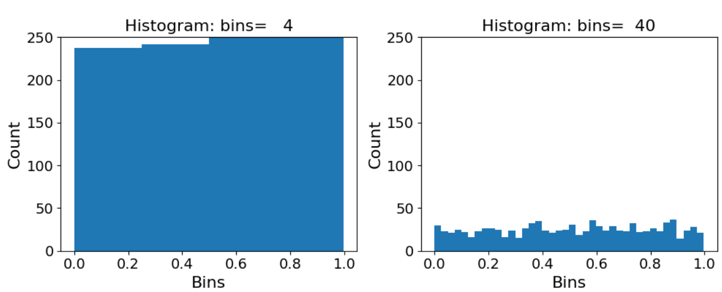
Since B is symmetric, Bv1 is the same as v1B, so we can write this as (v1B) . v2 = λ1 (v1 . v2)

v1B = v1 . Bv2 = v1 . λ2v2 = λ2 (v1 . v2)

λ2 (v1 . v2) = λ1 (v1 . v2)

If λ1 ≠ λ2, this implies that v1 . v2 = 0, i.e., v1 and v2 are orthogonal.

1.



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

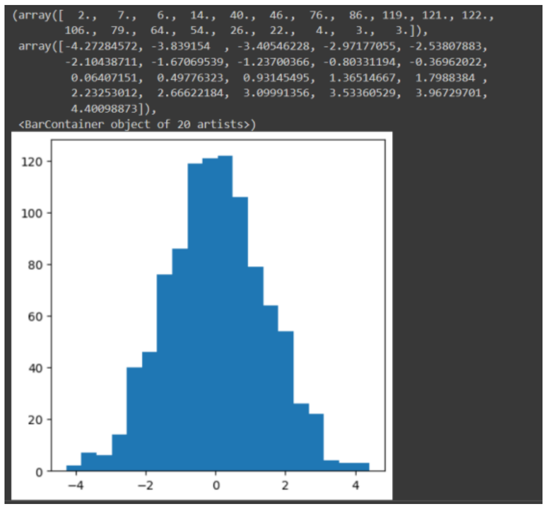
- This is due to the inherent randomness in the data. Even though the data is drawn from a uniform distribution, the number of data points that fall into each bin will vary due to chance. This is especially noticeable when the number of data points is relatively small, or the number of bins is large.

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

- This is because each time we run the code, a new set of random numbers is generated. Since the numbers are random, the exact distribution of numbers in the bins will change each time.

How do the above observations change (if so how) if you had started with more data?

- If we start with more data, the histograms will start to look more like the expected uniform distribution. This is due to the Law of Large Numbers, which states that as the size of our sample increases, the average of our sample gets closer to the average of the whole population. In other words, with more data, the effect of randomness is reduced and the true underlying distribution (in this case, uniform) is more accurately reflected in the histogram.



What do you observe?

- The histogram of these sums will likely resemble a bell curve or a Gaussian distribution. This is due to the Central Limit Theorem (CLT), which states that the sum of a large number of independent and identically distributed random variables tends towards a normal distribution.

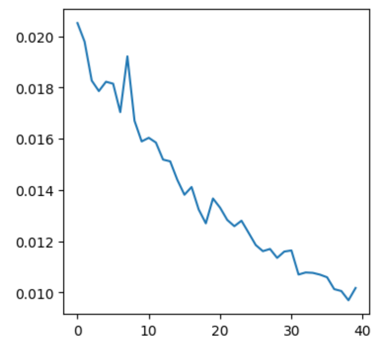
How does the resulting histogram change when you change the number of uniform random numbers you add and subtract?

- If we increase the number of random numbers, the spread of the distribution will increase, and vice versa. This is because the variance of the sum of independent and identically distributed random variables is equal to the sum of their variances.

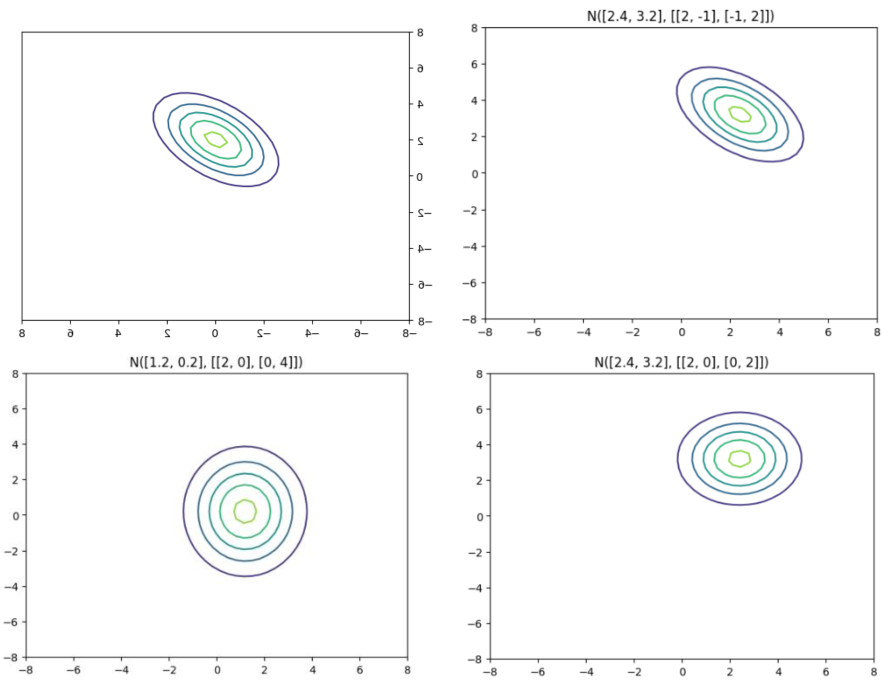
Is there a theory that explains your observation?

- Yes, the Central Limit Theorem (CLT) explains why the histogram of the sums of random numbers follows a Gaussian distribution.

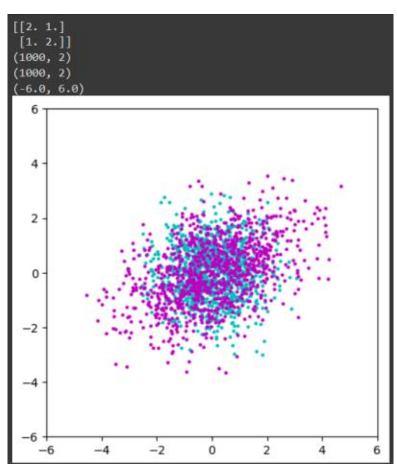
2.


0.020
0.018
0.016
0.014
0.012
0.010
0
10
20
30
40

3.

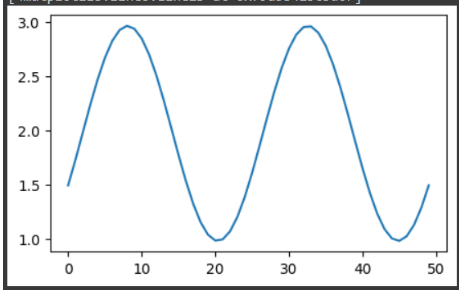

N([2.4, 3.2], [[2, -1], [-1, 2]])
N([1.2, 0.2], [[2, 0], [0, 4]])
N([2.4, 3.2], [[2, 0], [0, 2]])

4.


[[2. 1.]
[1. 2.]]
(1000, 2)
(1000, 2)
(-6.0, 6.0)

5.

```
The vector:  [0.8660254037844386, 0.5000000000000001]
Magnitude :  1.0
Angle :  59.99999999999999
(1000,)
Projected variance:  2.959828119768428
[<matplotlib.lines.Line2D at 0x79a5b415c3d0>]
```



What are the maxima and minima of the resulting plot?

```
max_var_idx = np.argmax(pVars)
min_var_idx = np.argmin(pVars)
print(f"Maxima at theta = {thRange[max_var_idx]}")
print(f"Minima at theta = {thRange[min_var_idx]}")
```

```
Maxima at theta = 1.0258261726007487
Minima at theta = 2.5645654315018716
```

Compute the eigenvalues an eigenvectors of the covariance matrix C

```
C = np.cov(Y.T)  # Note: we transpose Y because np.cov expects rows to represent variables
eigvals, eigvecs = np.linalg.eig(C)
print("Eigenvalues: ", eigvals)
print("Eigenvectors: ", eigvecs)
```

```
Eigenvalues:  [2.94153     1.14298761]
Eigenvectors:  [[ 0.85462256 -0.51924973]
 [ 0.51924973  0.85462256]]
```

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

- Yes, the directions of maximum and minimum variance of the projected data correspond to the eigenvectors of the covariance matrix, and the magnitudes of these variances correspond to the eigenvalues. The maximum variance is along the eigenvector corresponding to the largest eigenvalue, and the minimum variance is along the eigenvector corresponding to the smallest eigenvalue. This is the fundamental concept behind Principal Component Analysis (PCA).

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

- If the data is circularly symmetric about the origin, the variance of the projected data will be the same for all directions, and the plot will be a constant function (not sinusoidal). If the data is not circularly symmetric, the plot could potentially look sinusoidal, but this would depend on the specific distribution of the data.