Short Answer Problems (B. Python)

2. Describe (in words) the result of each of the following Python commands.

(a) > x = np.random.permutation(1000)

x is set to be an array of length 1000, with all 1000 of the numbers/elements in range [0, 999] randomly permuted without any duplicates.

(b) > a = np.array([[1,2,3],[4,5,6],[7,8,9]])
> b = a[2,:]

a is set to be a 3x2 matrix:
[[1 2 3]
[4 5 6]
[7 8 9]]
b is assigned to be the third row of a:
[7 8 9]

(c) > a = np.array([[1,2,3],[4,5,6],[7,8,9]])
> b = a.reshape(-1)

a is set to be a 3x2 matrix:
[[1 2 3]
[4 5 6]
[7 8 9]]
a is reshaped to a 1d array, and the result is assigned to b:
[1 2 3 4 5 6 7 8 9]

(d) > f = np.random.randn(5,1)
> g = f[f>0]

f is set to be a 5x1 matrix that has each element randomly generated based on the standard normal distribution
g is set to be an array of all the positive elements from matrix f

(e) > x = np.zeros(10)+0.5
> y = 0.5*np.ones(len(x))
> z = x + y

x is set to be an array of length 10 of all zeros with a 0.5 added to each of them:
[0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5]
y is set to be an array of the same length as x of all ones with a 0.5 multiplied with each of them, which has an equivalent result as x:
[0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5]
z is set to be an array of the sum of x and y, which results in a length 10 array of ones:
[1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]

(f) > a = np.arange(1,100)
> b = a[::-1]

a is set to be an array of length 100, with integers listed out from 1 to 99 inclusive in ascending order:
[1 2 3 … 99]
b is set to be an array of length 100 in reverse order of a:
[99 98 .. 1]

3. Write a few lines of code to do each of the following

(a) Use numpy.random.rand to return the roll of a six-sided die over N trials.
((np.random.rand(N)*6)+1).astype(int)

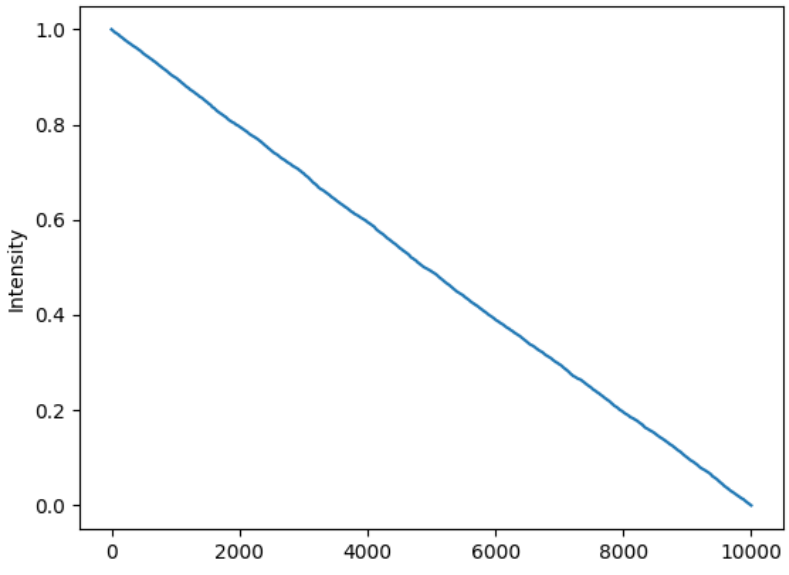
(b) Let y be the vector: y = np.array([1, 2, 3, 4, 5, 6]). Use the reshape command to form a new matrix z that looks like this: [[1,2],[3,4],[5,6]]
z = y.reshape(3,2)

(c) Use the numpy.max and numpy.where functions to set x to the maximum value that occurs in z (above), and set r to the row number (0-indexed) it occurs in and c to the column number (0-indexed) it occurs in.
x = np.max(z)
r = np.where(z == x)[0][0]
c = np.where(z == x)[1][0]

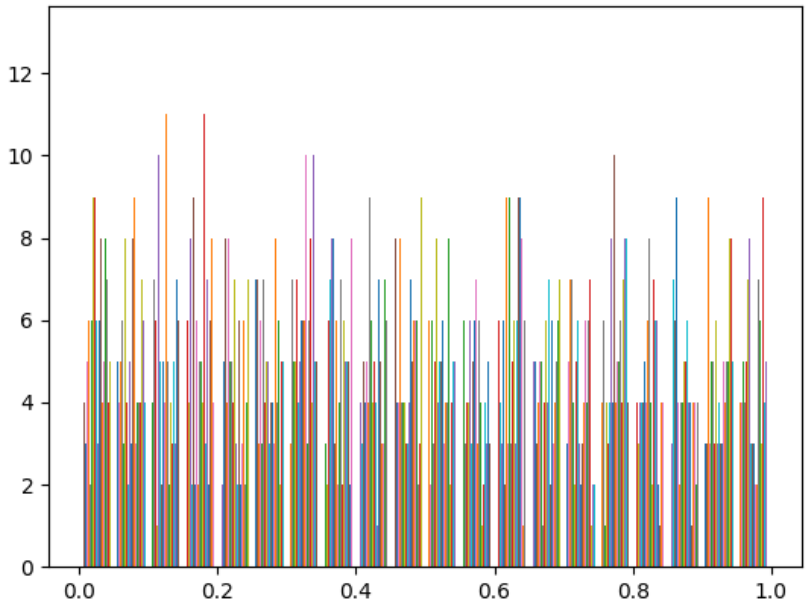
(d) Let v be the vector: v = np.array([1, 8, 8, 2, 1, 3, 9, 8]). Set a new variable x to be the number of 1's in the vector v.
x = np.count_nonzero(v == 1)

4. Create any 100 x 100 matrix A (not all constant). Save A in a .npy file called inputAPS0Q1.npy and submit it. Write a script which loads inputAPS0Q1.npy and performs each of the following actions on A. Name it PS0Q1.py and submit it. [30 points]

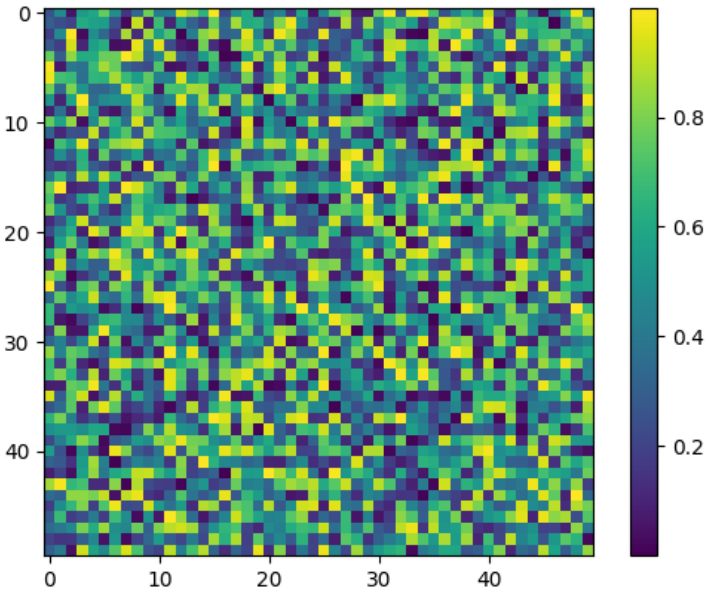
a) Plot all the intensities in A, sorted in decreasing value. Provide the plot in your answer sheet. (Note, in this case we don't care about the 2D structure of A, we only want to sort the list of all intensities.)



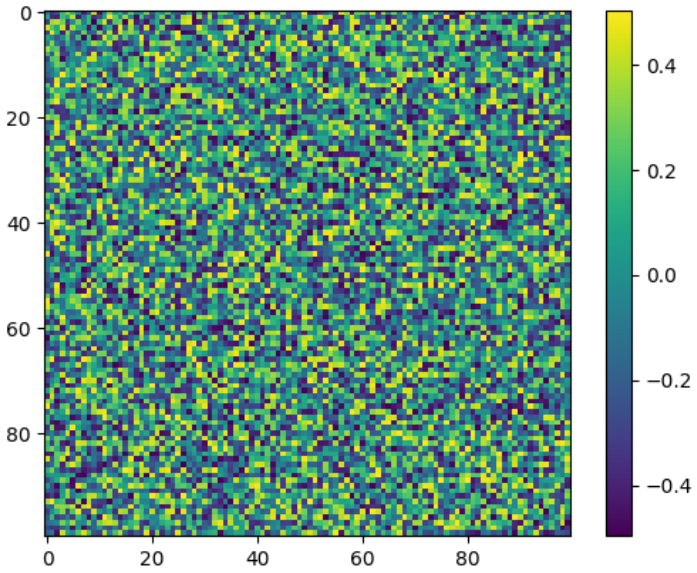
b) Display a histogram of A's intensities with 20 bins. Again, we do not care about the 2D structure. Provide the histogram in your answer sheet.



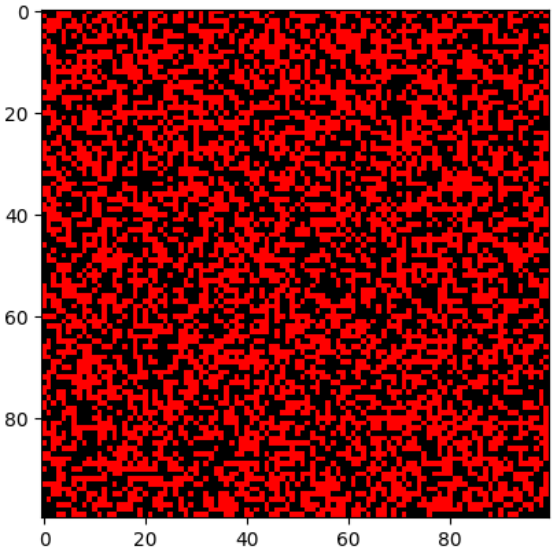
c) Create a new matrix X that consists of the bottom left quadrant of A. Display X as an image in your answer sheet using matplotlib.pyplot.imshow with no interpolation (blurry effect). Look at the documentation for matplotlib.pyplot.imshow. Save X in a file called outputXPS0Q1.npy and submit the file.



d) Create a new matrix Y, which is the same as A, but with A's mean intensity value subtracted from each pixel. Display Y as an image in your answer sheet using matplotlib.pyplot.imshow. Save Y in a file called outputYPS0Q1.npy and submit the file.



e) Create a new matrix Z that represents a color image the same size as A, but with 3 channels to represent R, G and B values. Set the values to be red (i.e., R = 1, G = 0, B = 0) wherever the intensity in A is greater than a threshold t = the average intensity in A, and black everywhere else. Display Z as an image in your answer sheet using matplotlib.pyplot.imshow. Save Z as outputZPS0Q1.png and submit the file. Be sure to view outputZPS0Q1.png in an image viewer to make sure it looks right



Subplot for Short Programming Example