

```
1  # Import numpy
2  import numpy as np
3
4  def compute_histogram(image_pixels ):
5      #get dimesnsions and setup vector for histogram
6      rows, cols = image_pixels.shape
7      hist = np.zeros(shape=(256))
8      for i in range(0, rows):
9          for j in range (0, cols):
10             #assign pixel values and it should add up to 1
11             hist[int(image_pixels[i][j])] += 1
12 hist=hist/(rows*cols)
13 return hist
14
15
16
17 def equalize(image_pixels):
18     rows, cols = image_pixels.shape
19     #get dimnesions
20     output=image_pixels
21     #computer the histogram and assign it to value so that when using the formula from
22     3-15
23     hist =compute_histogram(image_pixels)
24     s = np.zeros(shape=(256))
25     L=256
26     #L constant from textbook
27     #second nested for loop through the rows and columns to assign each pixel value in
28     the out image to the transformed pixel
29     for k in range(L):
30         for i in range(k):
31             s[k] += hist[i]
32             s[k] = (L - 1) * s[k]
33     for i in range(rows):
34         for j in range(cols):
35             output[i,j]=s[int(image_pixels[i][j])]
36     return output
37
38
39
40
41
42 def plot_histogram( hist ):
43     # plot_histogram Plots the length 256 numpy vector representing the normalized
44     histogram of a grayscale image.
45     #
46     # Syntax:
47     #   plot_histogram( hist )
48     #
49     # Input:
50     #   hist = The length 256 histogram vector..
51     #
52     # Output:
53     #   none
54     #
55     # History:
56     #   S. Newsam      10/23/2022   created
57
58     # Import plotting functions from matplotlib.
59     import matplotlib.pyplot as plt
60
61     plt.bar( range(256), hist )
62
63     plt.xlabel('intensity value');
64
```

```
65     plt.ylabel('PMF');  
66  
67     plt.show()  
68
```