

PDI - Lab1

Yuri Diego Santos Niitsuma (2011039023)

Importando bibliotecas

```
In [1]: import skimage.io as io
import numpy as np
import skimage.exposure as skie
%matplotlib inline

import matplotlib.pyplot as plt
import matplotlib.image as mpimg
```

Escolha uma imagem em escala de cinza e uma colorida (RGB).

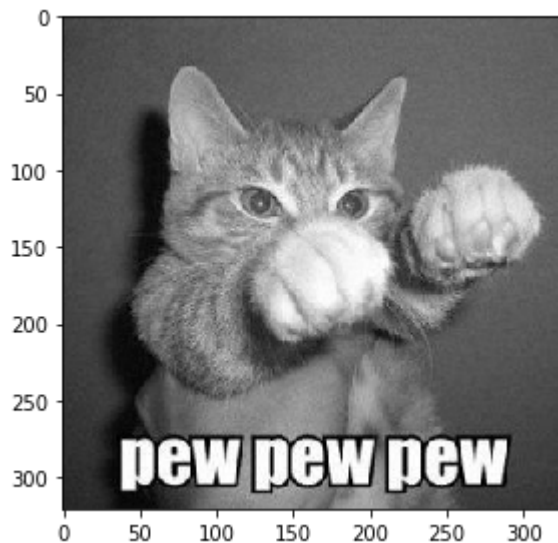
São dois PNGs
'cat_gray.png' é uma imagem com escala de cinza
'ines_colors.png' é uma imagem com cores RGB

Escreva o código para carregar e mostrar ambas imagens.

```
In [2]: img_color = io.imread('ines_colors.png')
io.imshow(img_color)
io.show()

img_gray = io.imread('cat_gray.png')
io.imshow(img_gray, cmap = plt.get_cmap('gray'))
io.show()
```

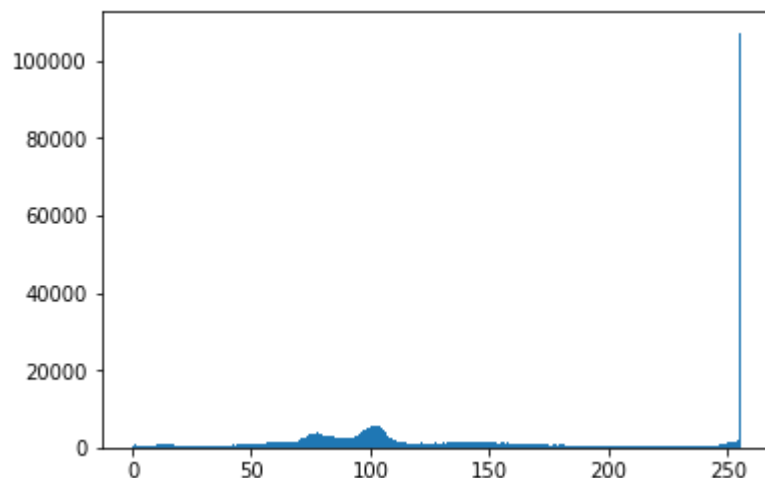
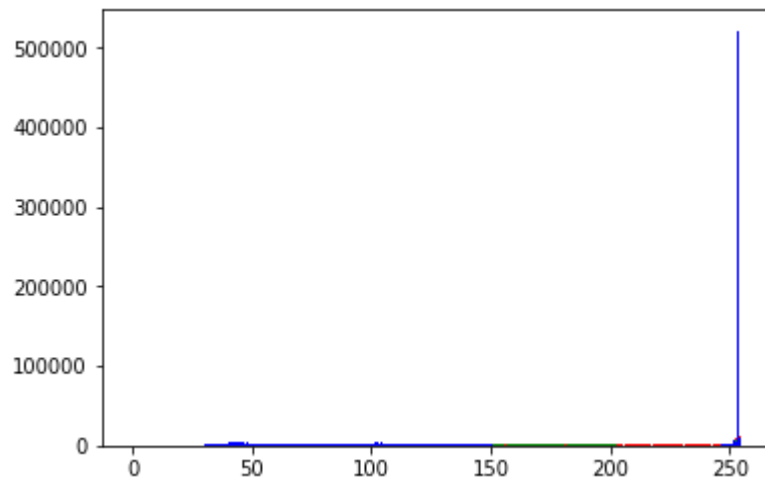
/home/yuri/anaconda3/lib/python3.6/site-packages/skimage/io/_plugins/matplotlib_plugin.py:51: FutureWarning: Conversion of the second argument of issubdtype from `float` to `np.floating` is deprecated. In future, it will be treated as `np.float64 == np.dtype(float).type`.
out_of_range_float = (np.issubdtype(image.dtype, np.float) and



Escreva o código para criar e mostrar o histograma de ambas as imagens.

```
In [3]: # hist in color space
color = ['r','g','b']
for i, c in enumerate(color) :
    plt.hist(img_color[:, :, i].flatten(), 256, color=c)
plt.show()

# hist in grayscale
plt.hist(img_gray.ravel(), 256, [0, 256])
plt.show()
```



Escreva o código e mostre uma das imagens com diferentes operações no histograma (mudança da exposição, equalização).

Função auxiliar contendo a imagem e histograma

```
In [4]: # Retirado dos exemplos nas aulas
def show(img):
    # Display the image.
    fig, (ax1, ax2) = plt.subplots(1, 2,
                                   figsize=(12, 3))

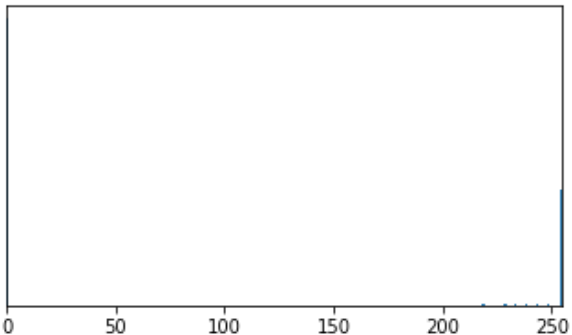
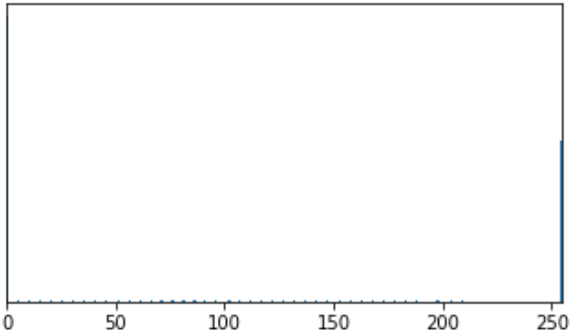
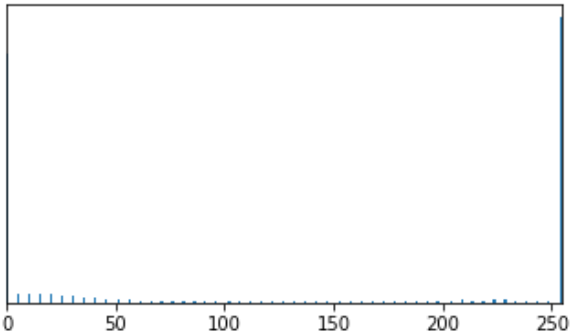
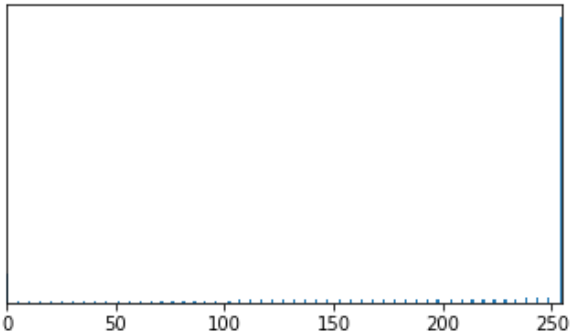
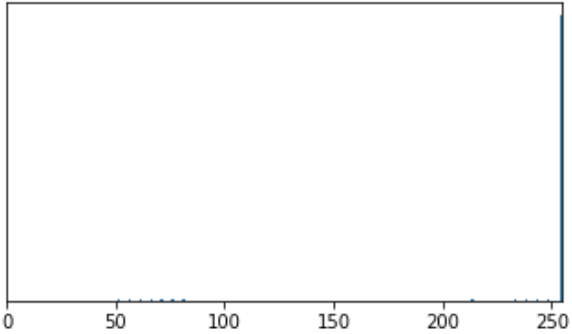
    ax1.imshow(img, cmap=plt.cm.gray)
    ax1.set_axis_off()

    # Display the histogram.
    ax2.hist(img.ravel(), lw=0, bins=256)
    ax2.set_xlim(0, img.max())
    ax2.set_yticks([])

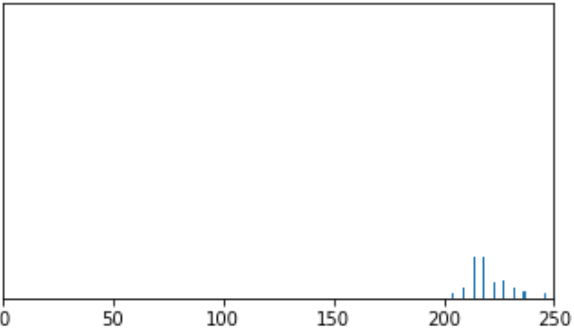
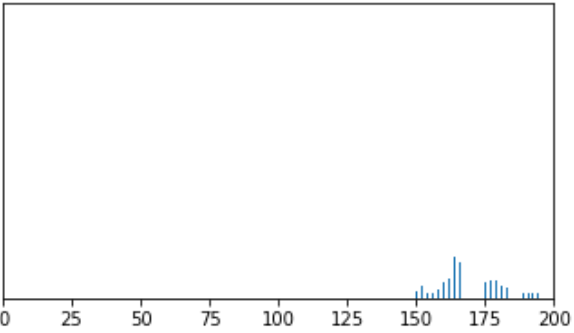
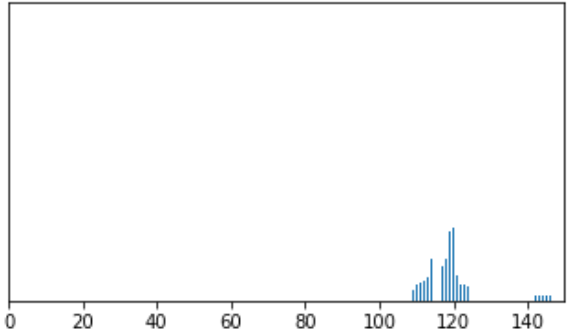
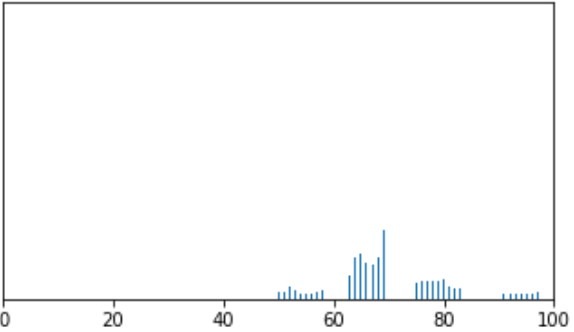
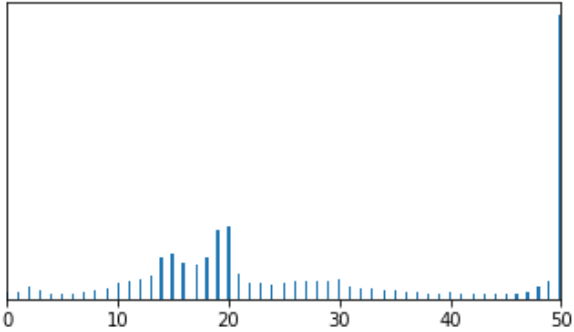
    plt.show()
```

Mudança da exposição

```
In [5]: for i in range(0, 250, 50):  
        show(skie.rescale_intensity(img_gray, in_range=(i,i+50), out_range=(0, 255)))
```



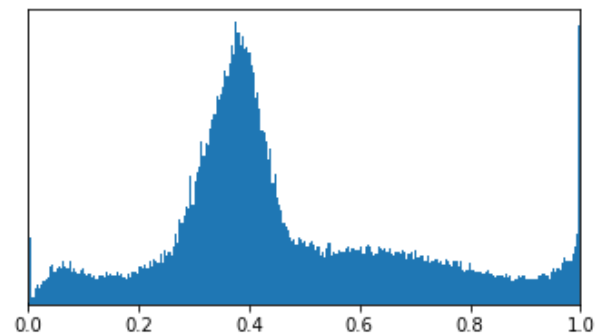
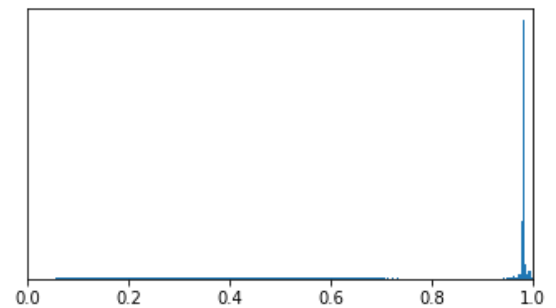
```
In [6]: for i in range(0, 250, 50):  
        show(skie.rescale_intensity(img_gray, in_range=(0,255), out_range  
        =(i, i+50)))
```



Equalização

```
In [7]: show(skie.equalize_adapthist(img_color))  
show(skie.equalize_adapthist(img_gray))
```

```
/home/yuri/anaconda3/lib/python3.6/site-packages/skimage/util/dtype.p  
y:122: UserWarning: Possible precision loss when converting from floa  
t64 to uint16  
.format(dtypeobj_in, dtypeobj_out))
```



Escreva o código que calcule o negativo da imagem.

```
In [9]: # Peguei a imagem da Inês Brasil pois é a mais divertida
# O código é feito em python mesmo então vai demorar uns segundinhos

new_img = img_color.copy()

height, width, channels = new_img.shape
for i in range(height):
    for j in range(width):
        for k in range(channels):
            new_img[i][j][k] = 255 - new_img[i][j][k]

show(new_img)
```

