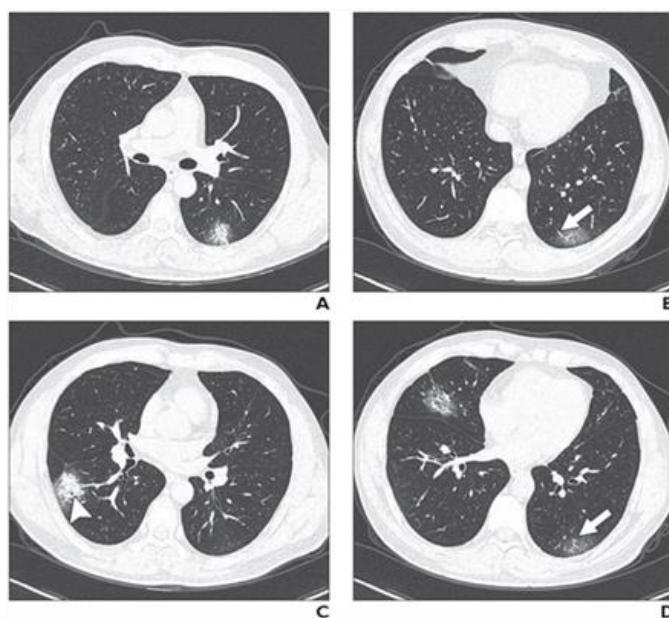

MEDICAL IMAGE ANALYSIS IN PYTHON

by Renat Alimbekov



First edition
2021

```
from matplotlib import cm # color schemes for visualization
from matplotlib import pyplot as plt # library for visualization
```

We read the file using the load method, and to get the image, we need to use the get_fdata function. We will work with the file volume_pt5 / volume-44.nii

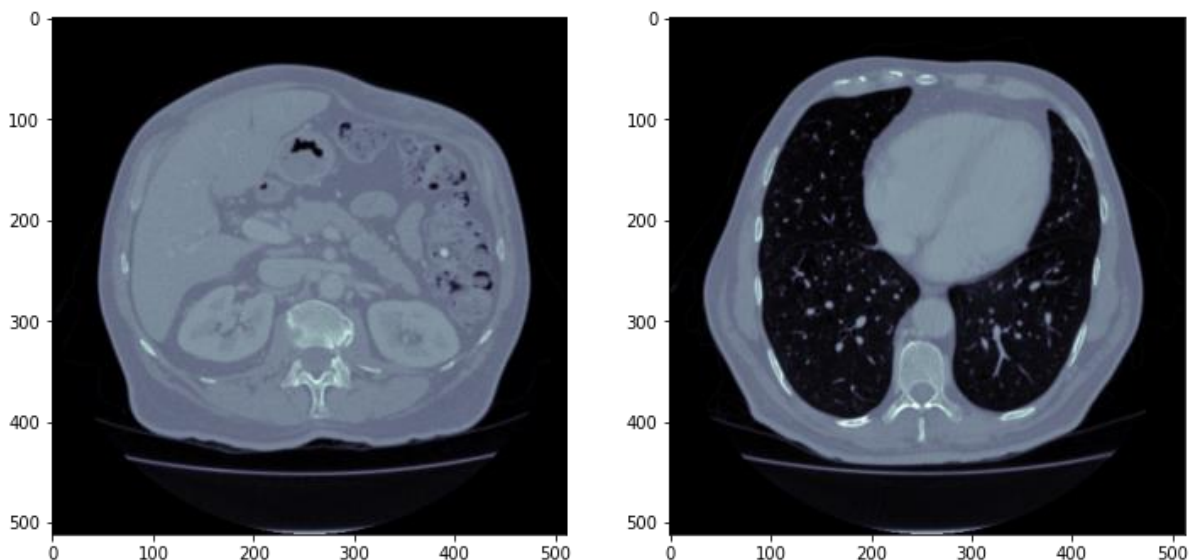
```
filepath = 'volume_pt5/volume-44.nii'
imagedata=nibabel.load(filepath)
array = imagedata.get_fdata()
```

turn the image 90 degrees and look at the shape of the file contents.

```
array = np.rot90(np.array(array))
print(array.shape)
```

The output will be Height x Width x Depth - (512, 512, 119). Depth here means CT slices. Let's visualize a 50 and 118 CT slice.

```
f = plt.figure(figsize=(12,12))
ax = f.add_subplot(121)
ax2 = f.add_subplot(122)
ax.imshow(array[:,50,:].astype(np.float32), cmap=plt.cm.bone)
ax2.imshow(array[:,118,:].astype(np.float32), cmap=plt.cm.bone)
```



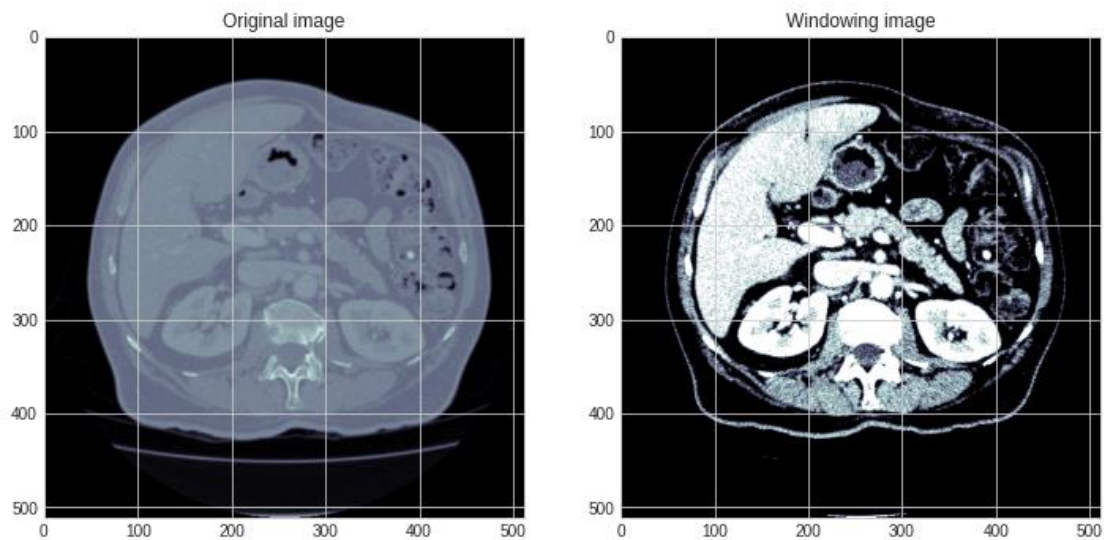
Self-study task: visualize other CT slices

Explore of metadata

As I said before, DICOM files contain metadata in addition to the image. For example: age, gender, study modality, name, body part and body position.

Let's try to explore our metadata. Let's start with what we already know, the file reads.

```
example = 'stage_2_images/ID_01fe90211.dcm'
```

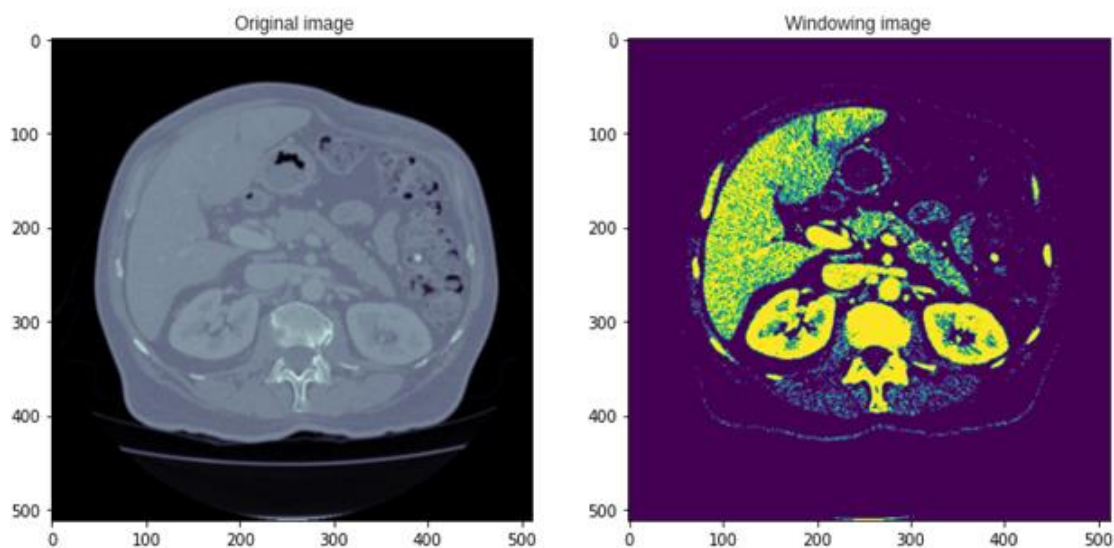


Indeed, after applying Windowing, it highlights structures more.

We've reviewed the recommended window settings. Let's take a look at a few more exotic options. For example, you can mix Windowing.

```
def rainbow_bsb_window(img):
    brain_img = windowed(img, 40, 80)
    subdural_img = windowed(img, 80, 200)
    bone_img = windowed(img, 600, 2000)
    combo = (brain_img*0.3 + subdural_img*0.5 + bone_img*0.2)
    return combo

f = plt.figure(figsize=(12,12))
ax = f.add_subplot(121)
ax2 = f.add_subplot(122)
ax.imshow(array[...,:50].astype(np.float32), cmap=plt.cm.bone)
ax.title.set_text('Original image')
ax2.imshow(rainbow_bsb_window(array[...,:50].astype(np.float32)))
ax2.title.set_text('Windowing image')
```



Self-study task: try other Windowing. Evaluate the result

