**Setup**

First, install spyder (or any python IDE). If you are on windows, you will probably want to install anaconda first: https://docs.anaconda.com/anaconda/install/

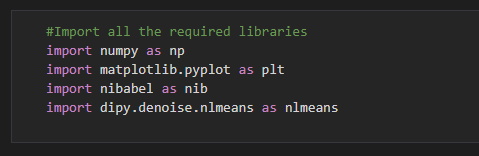
• Note – you don’t need to use spyder (any python IDE is fine) but it will help a lot because the attached console is very useful for debugging and visualization of intermediate results.

Second, get the images (images folder). This directory contains 6 images:

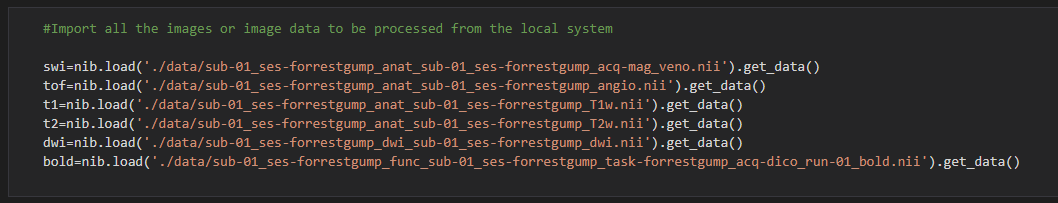
* SWI (susceptibility weighted image): sub-01\_ses-forrestgump\_anat\_sub-01\_ses-forrestgump\_acq-mag\_veno.nii.gz
* TOF (time of flight): sub-01\_ses-forrestgump\_anat\_sub-01\_ses-forrestgump\_angio.nii.gz
* T1 (T1-weighted image): sub-01\_ses-forrestgump\_anat\_sub-01\_ses-forrestgump\_T1w.nii.gz
* T2 (T2-weighted image): sub-01\_ses-forrestgump\_anat\_sub-01\_ses-forrestgump\_T2w.nii.gz
* DWI (Diffusion weighted image): sub-01\_ses-forrestgump\_dwi\_sub-01\_ses-forrestgump\_dwi.nii.gz
* BOLD (Blood oxygen level dependent image): sub-01\_ses-forrestgump\_func\_sub-01\_ses-forrestgump\_task-forrestgump\_acq-dico\_run-01\_bold.nii.gz

**Topics:** Basic numpy, medical imaging modalities, SNR, and denoising. In this assignment we will familiarize ourselves with basic numpy array manipulation, different imaging modalities, and gain understanding of some basic image features (contrast and SNR).

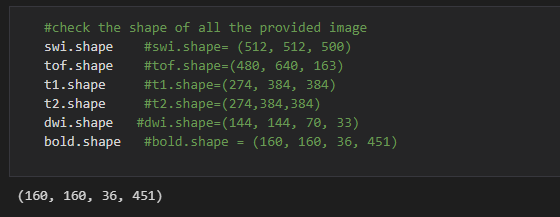
Import all the required libraries:



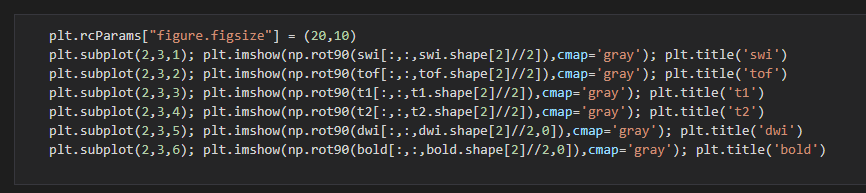
Import all the images or image data to be processed from the local system:

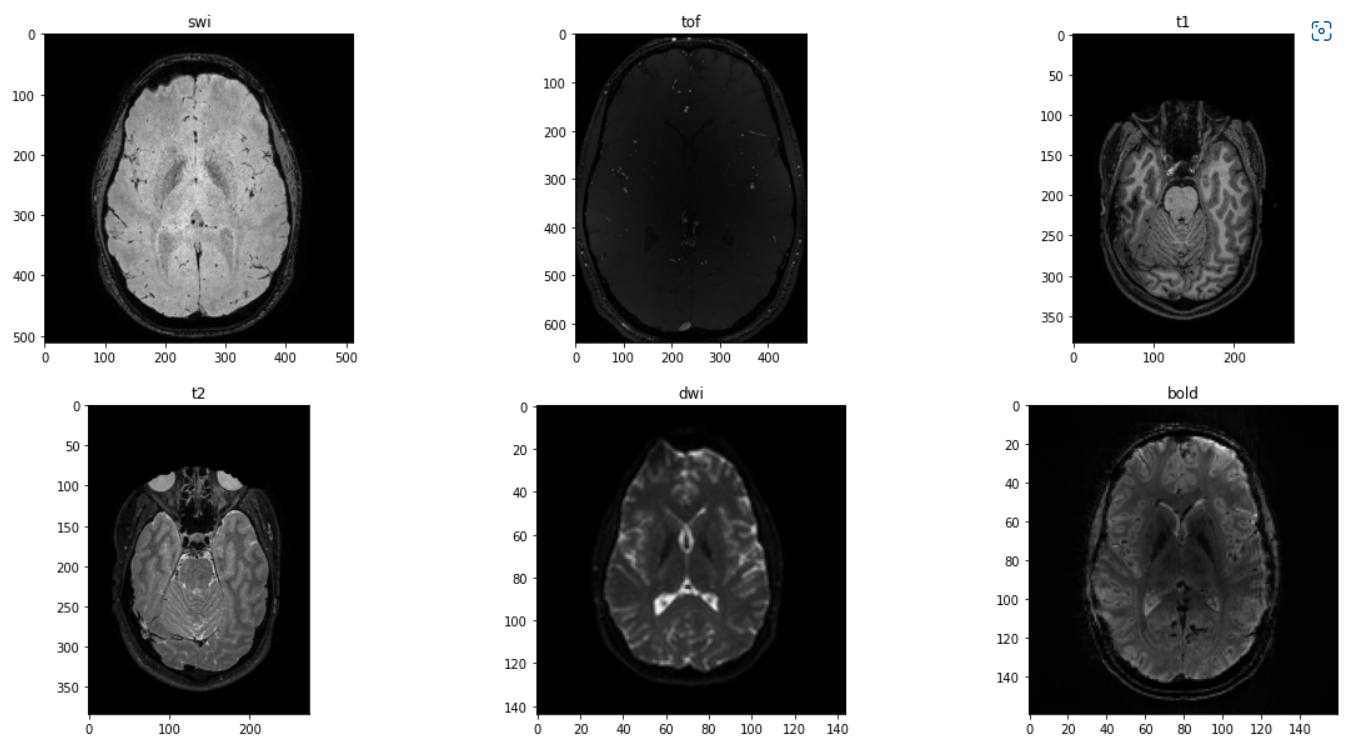


Check the shape of all the provided images:



Plotting the middle z-slice of every image in grey colour map:

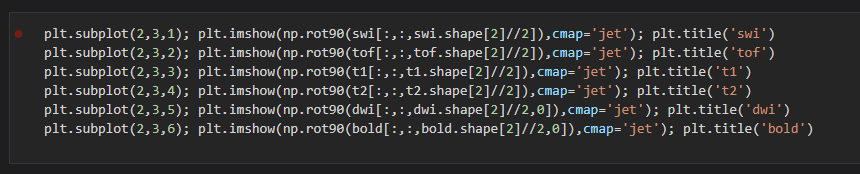




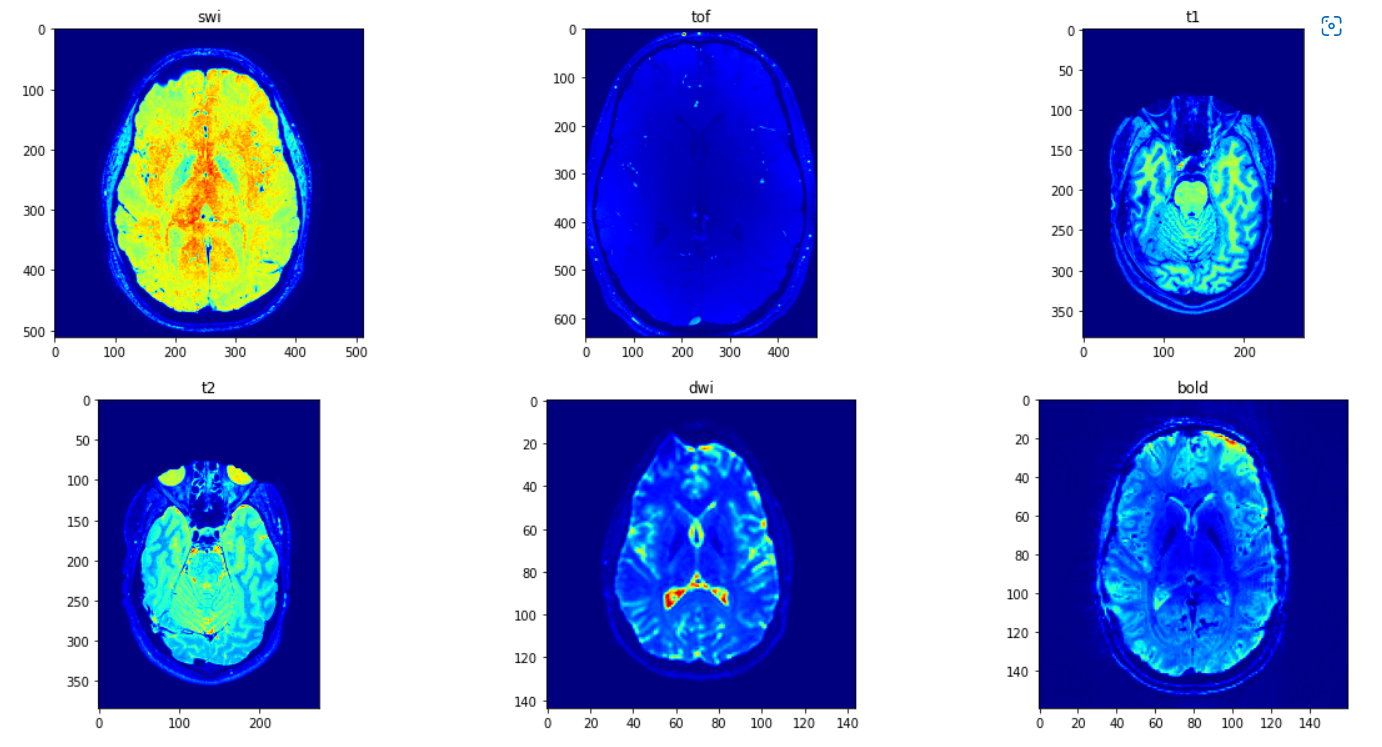
Part 1**: Simple plotting with matplotlib**

1. Basic plotting: replicate the figure below, but use the ‘jet’ colormap instead of the ‘gray’ colormap (which I used to create the figure below). Part 1a: plotting the middle z-slice for each image type.
2. Use the np.min() and np.max() functions to create minimum and maximum intensity projections respectively (see below) of the SWI and the TOF. Show views from all 3 axes and use jet colormap to display your results. \*hint for the SWI, you will probably want to restrict the MIP to ~50-100 slices, and use vmax=300 for the TOF (to better highlight the arteries).

1. Code Snippet for plotting the middle z-slice for each image type using jet colour map:

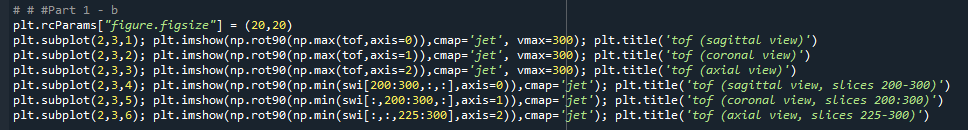


Plot:

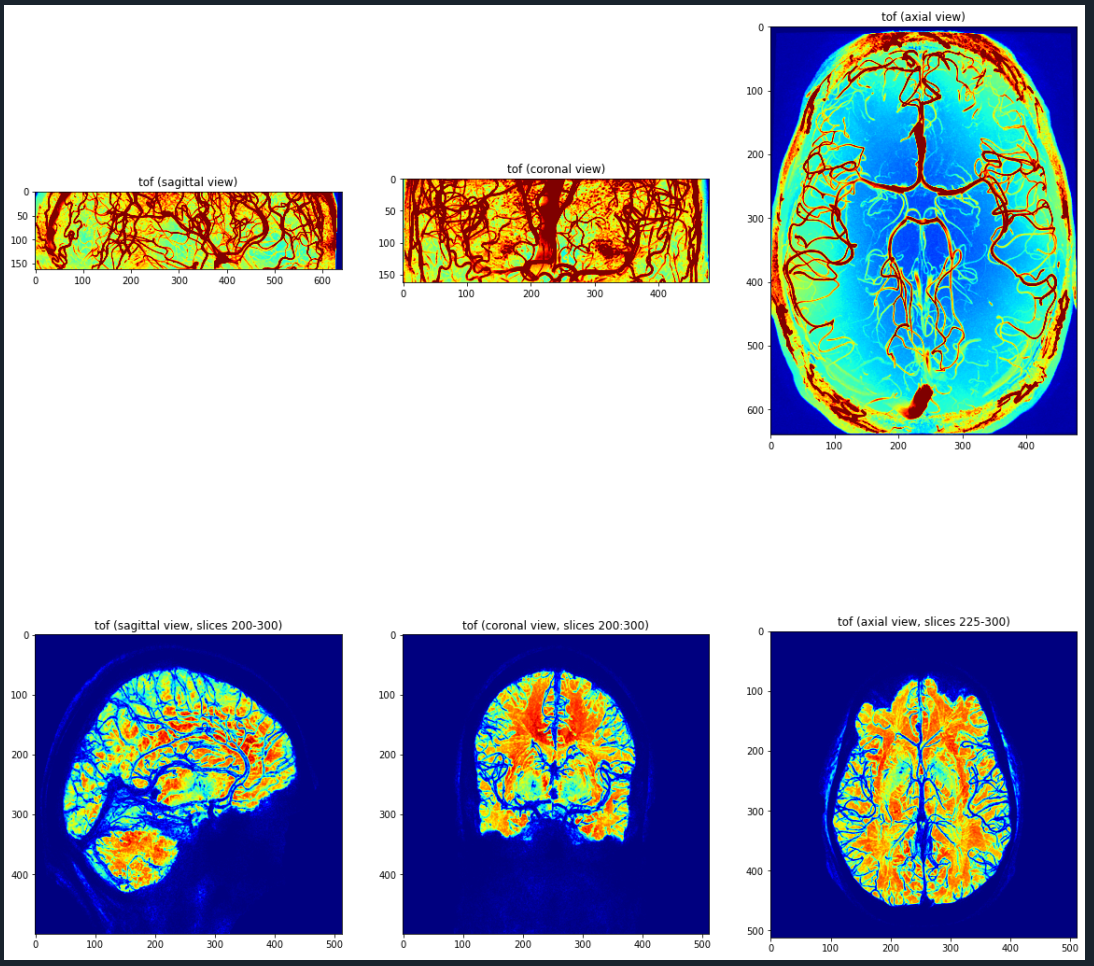


1. Plotting the maximum intensity projection and minimum intensity projection for TOF and SWI.

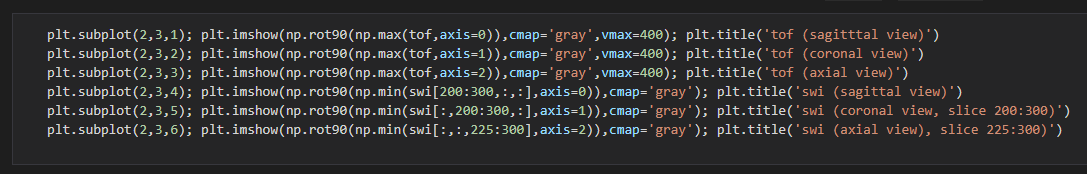
Code Snippet for jet colour map:



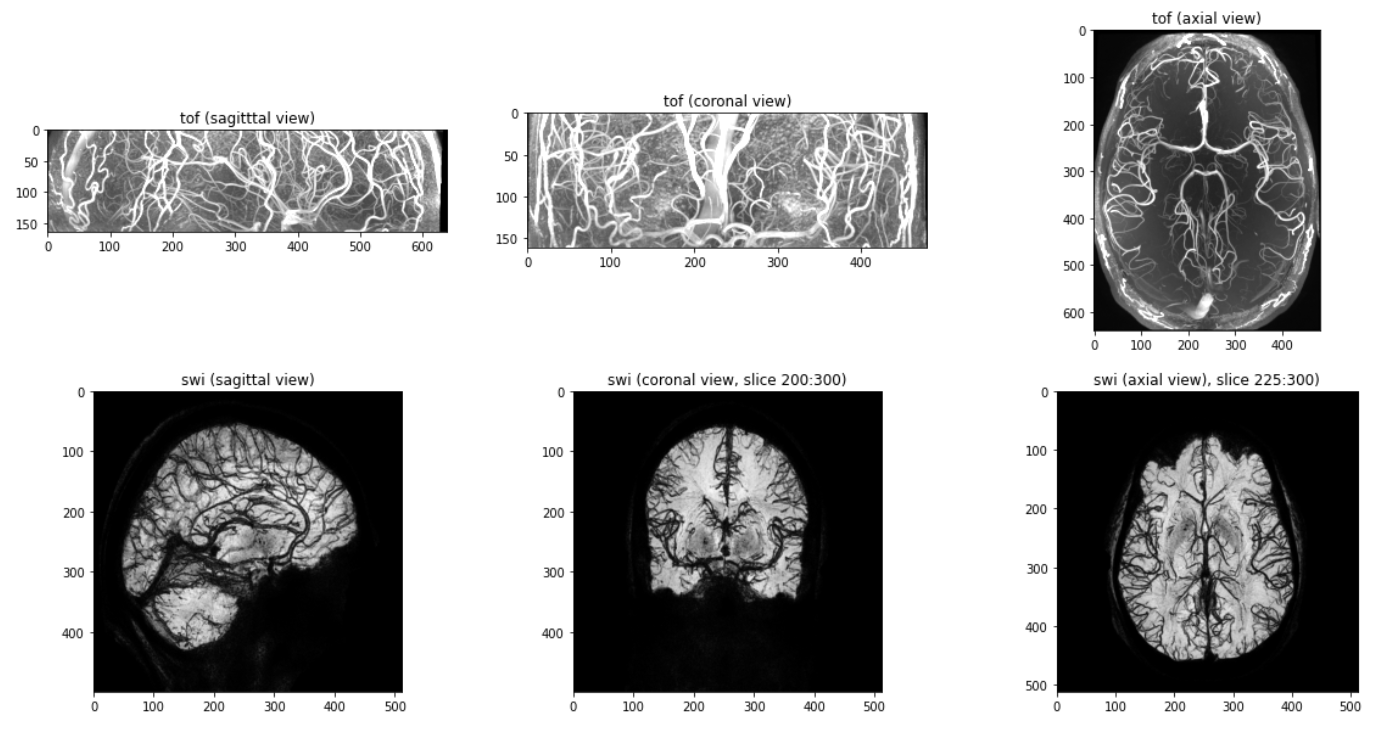
Plot:



Using Code snippet for gray colour map:



Plot:



Part 2: Contrast estimation

Using numpy, get 3 different contrast measures for each image (root mean square, Michelson, and entropy, see lecture 3 slide 4, 5). Report the contrast (all 3 values) in the title of the plots in figure 1a. base your contrast estimation on the entire 3D or 4D image (not just the slice shown in the figures).

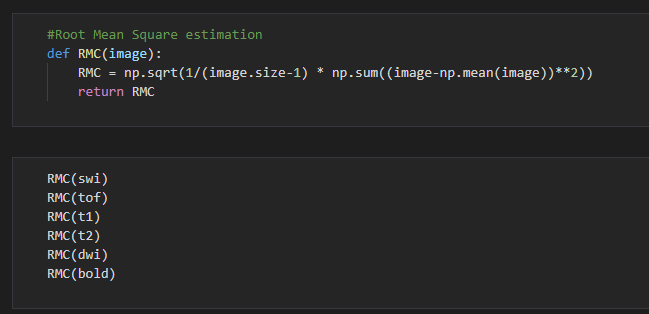
There are 3 types of contrast root mean square, Michelson, and entropy

MC = (image.max()-image.min())/(image.max()+image.min())

RMS = np.sqrt(1/(image.size-1)\*np.sum(image-np.mean(image))\*\*2)

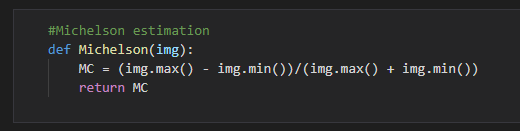
**Method for Michelson Contrast of each Images:**

Code Snippet:

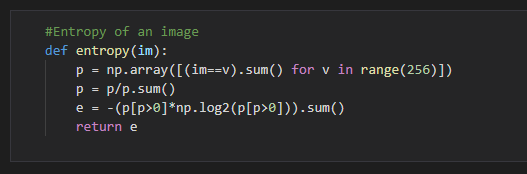


**Method for Root Mean Square Contrast of each Images:**

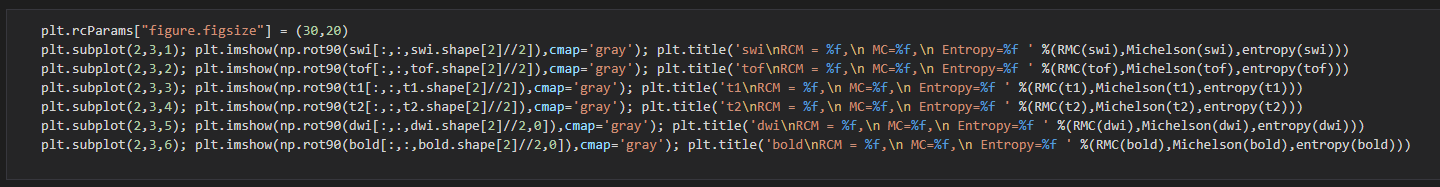
Code Snippet:

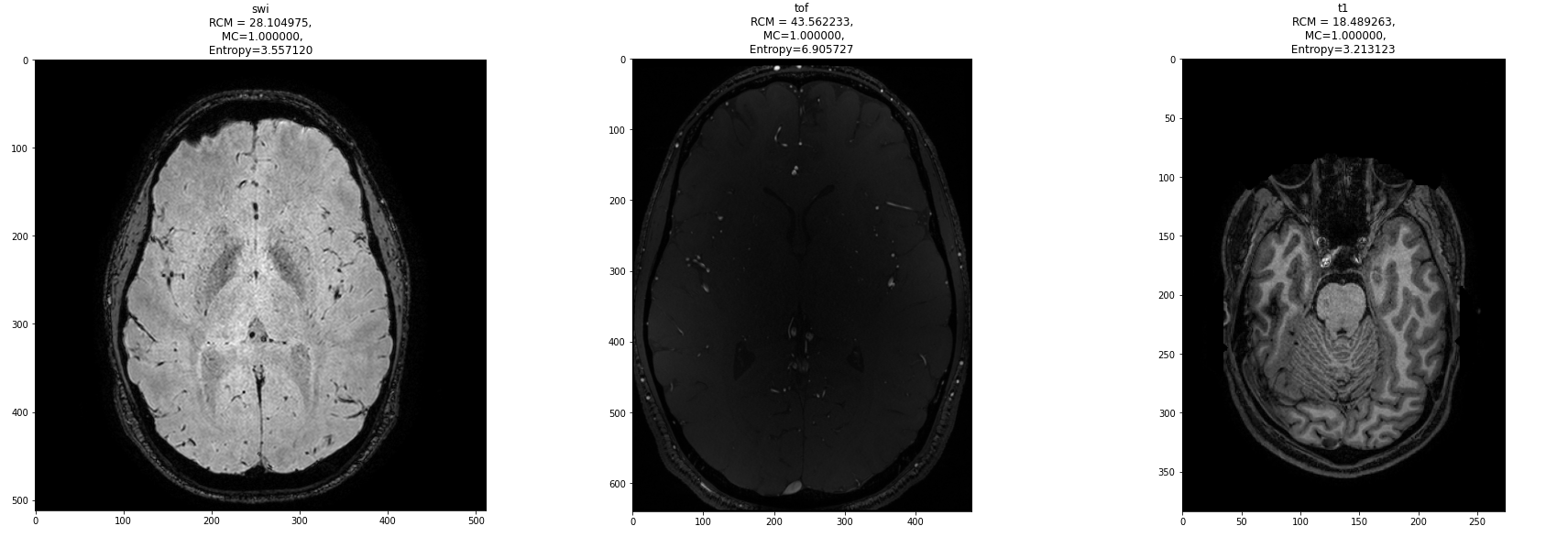


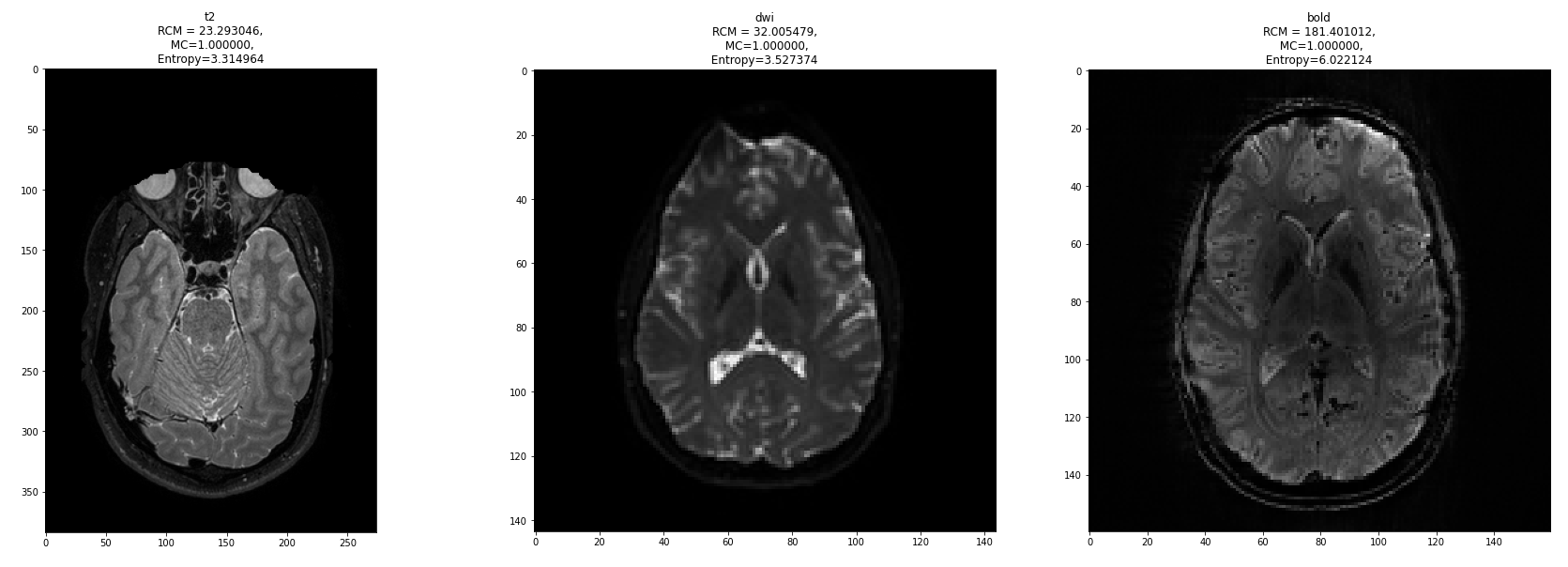
**Method for Entropy Contrast of each Images:**



Plotting Part:







Part 3: SNR estimation

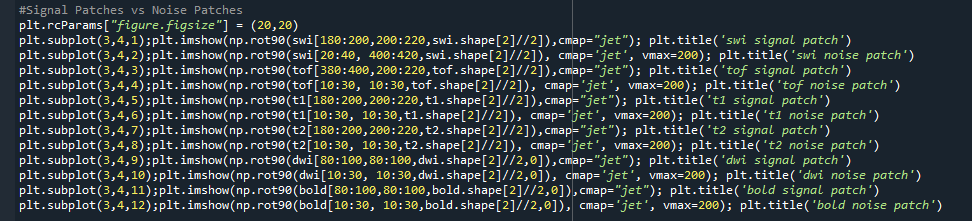
a) Using the method outlined in the lecture 3 slide 7, report the SNR for each of the modalities. Which modality has the highest SNR and which has the lowest?

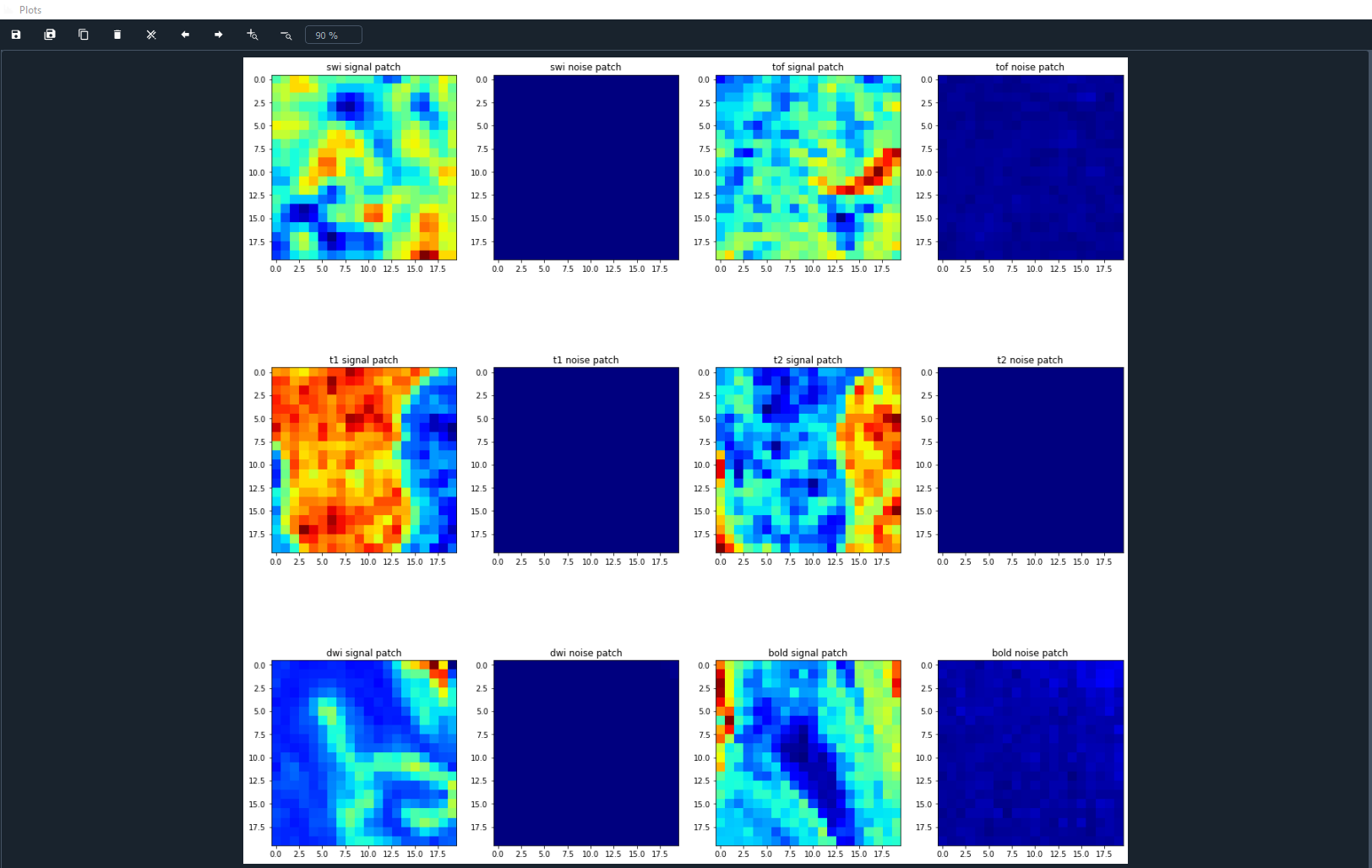
b) Plot histograms of the noise in each image. What type of distribution does the noise follow?

To display the solution to part 3, create a new figure (as in part 1) and display the noise histogram of each image (instead of the image itself) in each sub plot. Show the SNR as the title above each histogram (along with the image name). \*Caution – when selecting your noise patch, be sure the patch isn’t all zeros, otherwise your noise will be estimated as 0 and the SNR will be infinite\*

Signal patches vs Noise patches of each image.

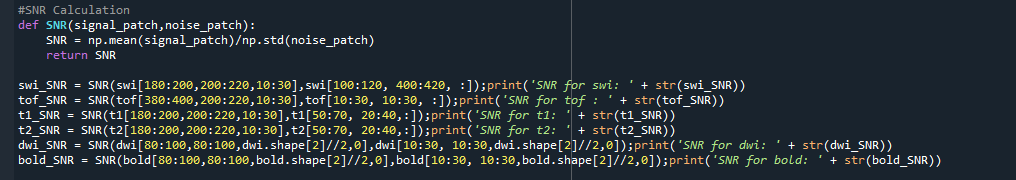
Code Snippet:

****

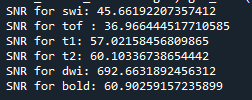
****

1. Report SNR for each modality

Code Snippet:



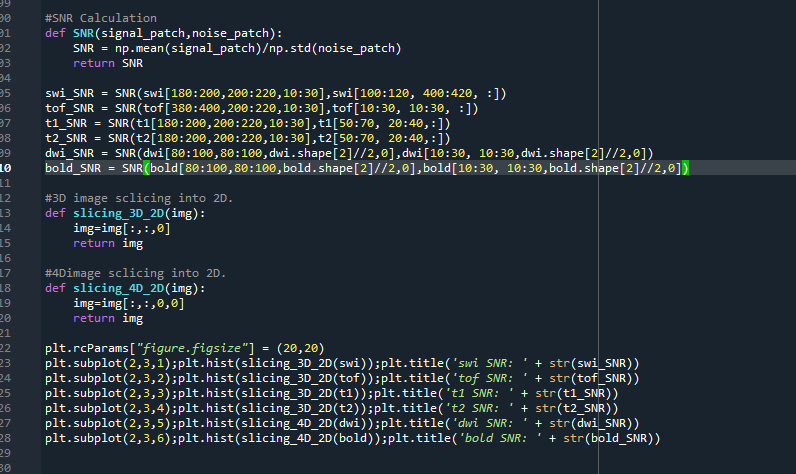
SNR Report:



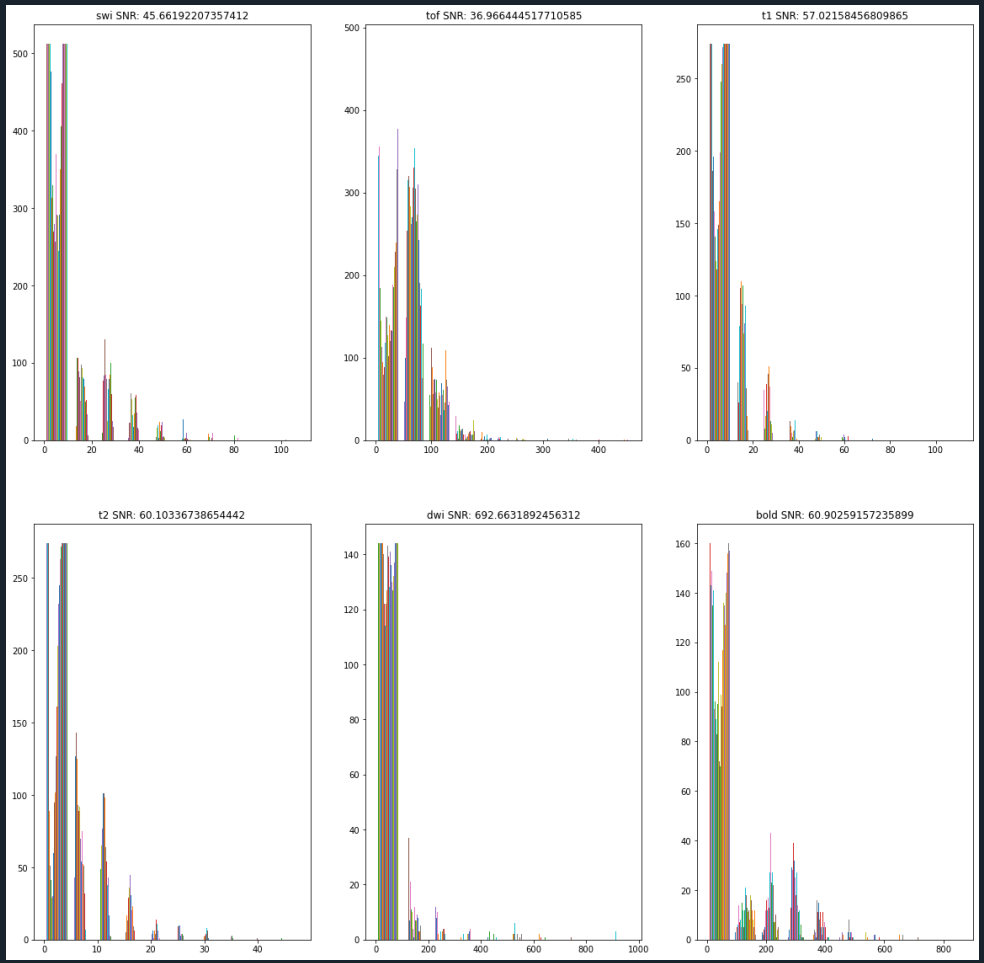
Highest SNR of “692.6631892456312” has taken by “dwi” modality while lowest SNR of “36.966444517710585” has taken by “tof” modality.

1. Plotting histograms of the noise in each image

Code snippet:



Histogram Plot:



According to the above histograms noise follows a “Gaussian” distribution.

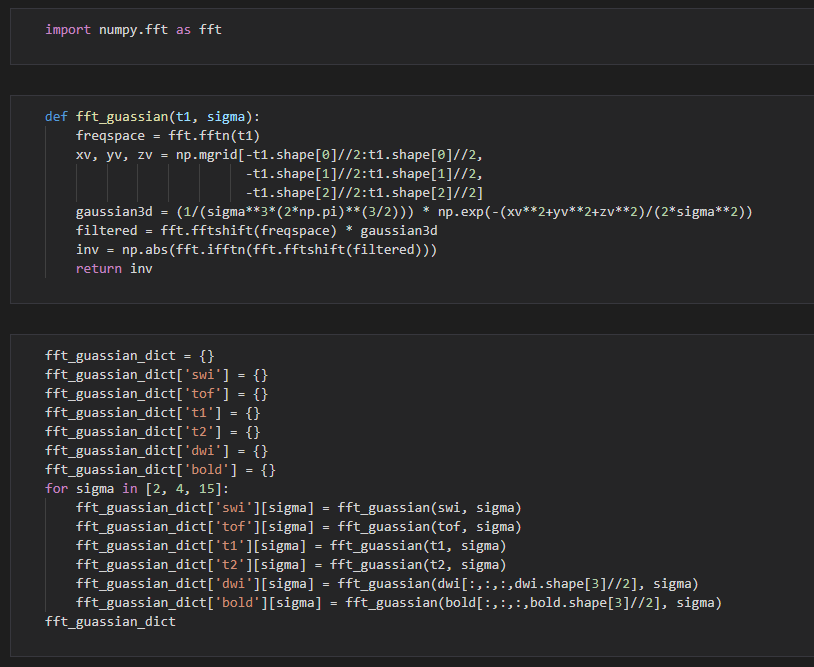
Part 4: Denoising

a) Using the Fourier transform method shown at end of assignment 1 video and the 3d gaussian, apply linear filtering to each image for 𝜎 = 2, 𝜎 = 4, and 𝜎 = 15. Create 3 new versions of the figure in part 1a, one figure for each sigma. Show the middle z-slice of the filtered image in all subplots.

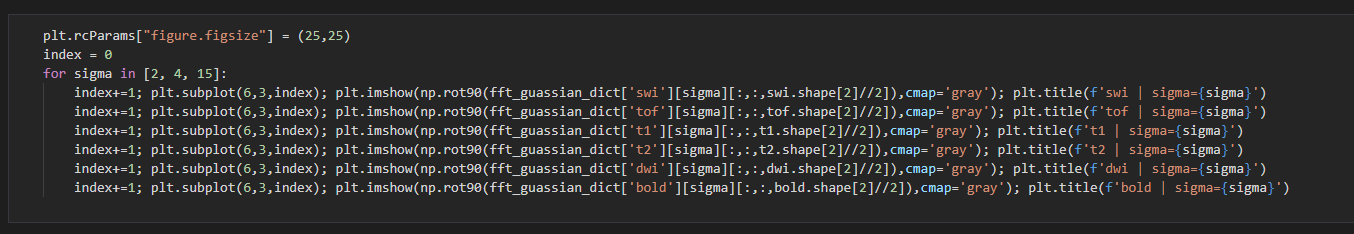
b) Install the ‘dipy’ package and use nlmeans to denoise the images. Show, for each image, the following plots 1) noisy image 2) denoised image 3) method noise (noisy minus denoised), see below for an example on the T1.

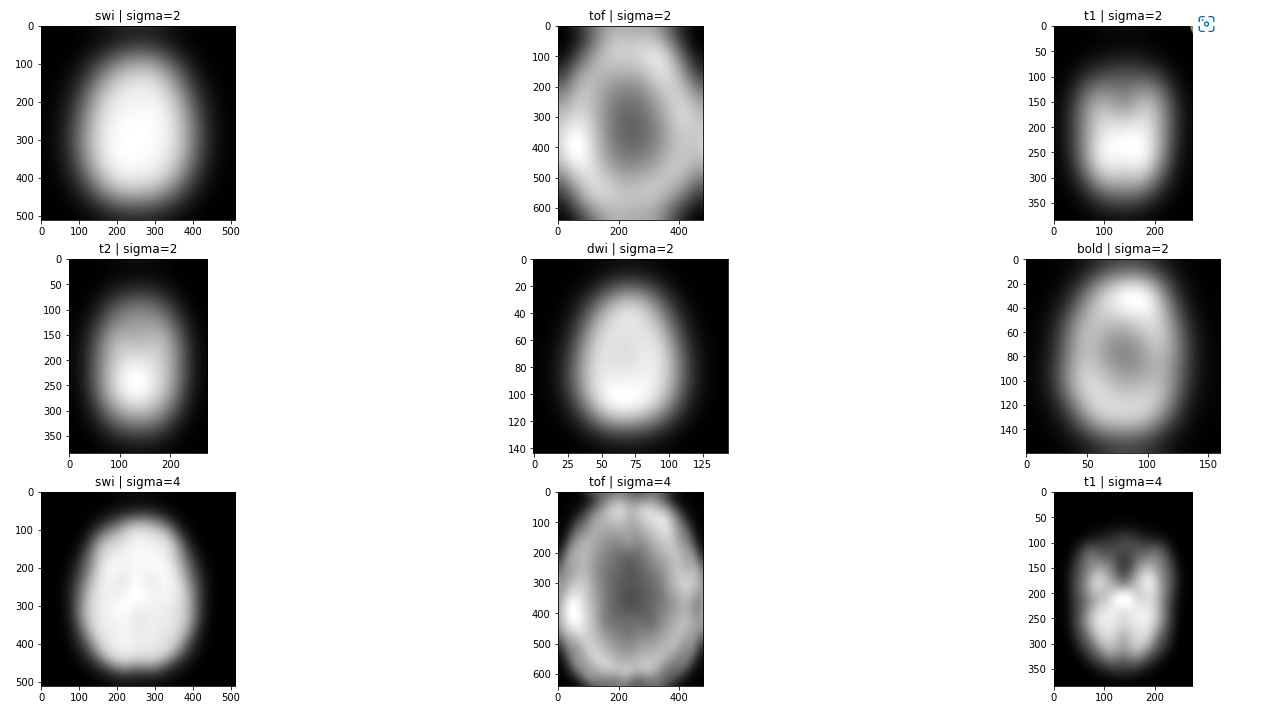
1. Apply linear filtering to each image for 𝜎 = 2, 𝜎 = 4, and 𝜎 = 15.

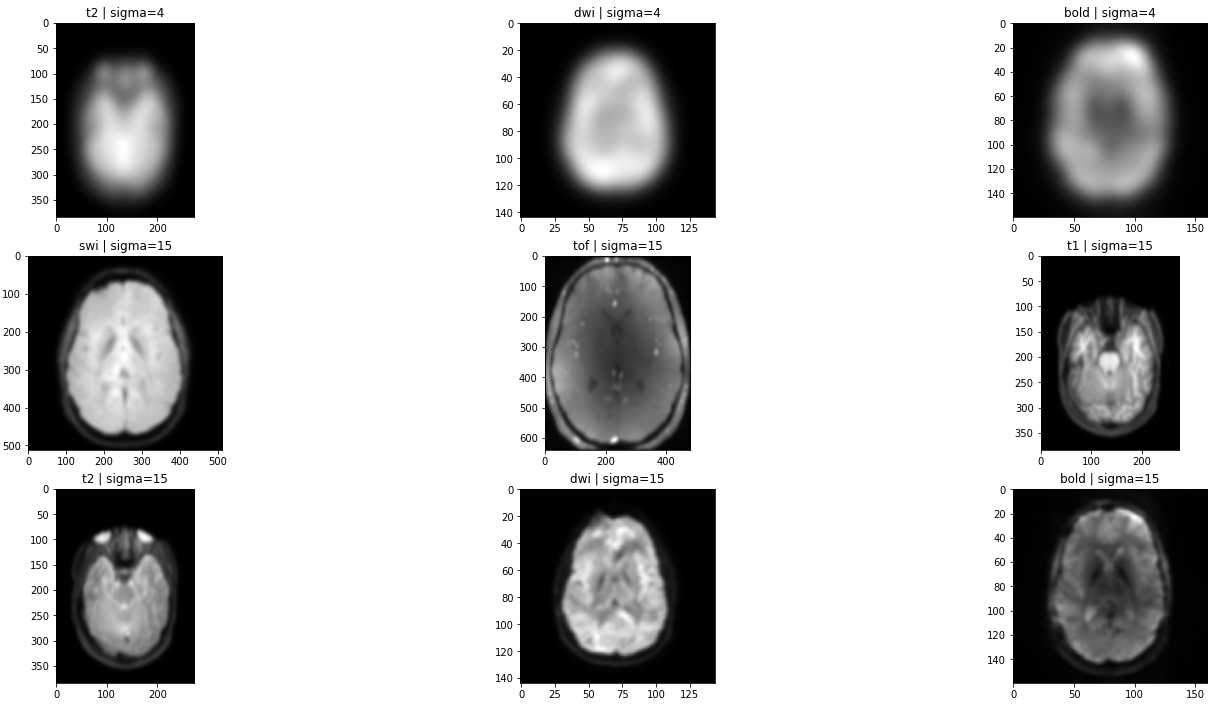
Code Snippet:



Plotting:

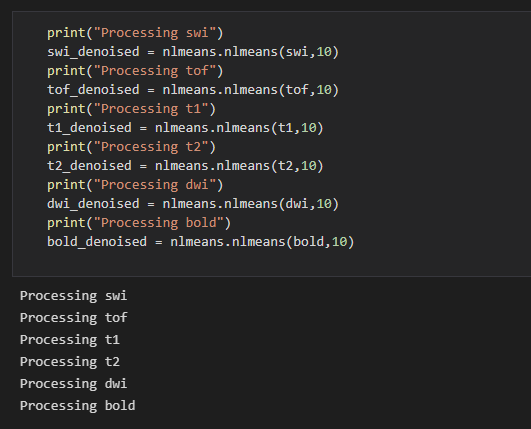






1. Noisy Image, Denoised Image and method noise – Denoised using nlmeans

Code Snippet:



Plot:

