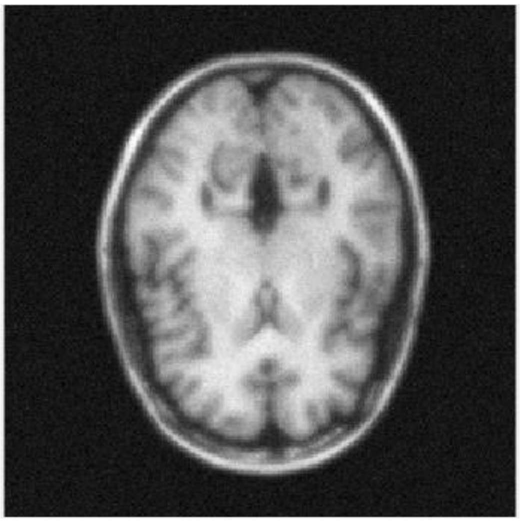
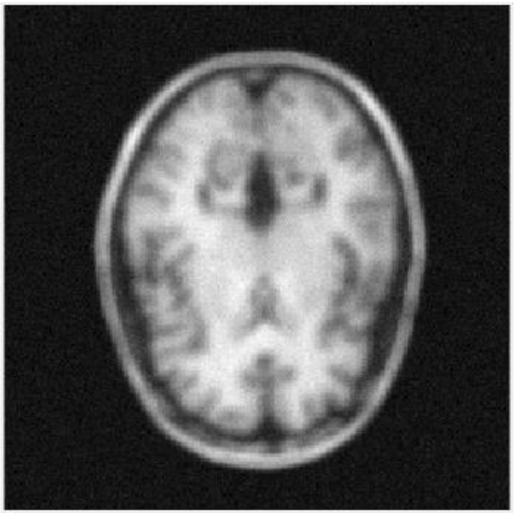
Image Deconvolution: Wiener-Hunt Method

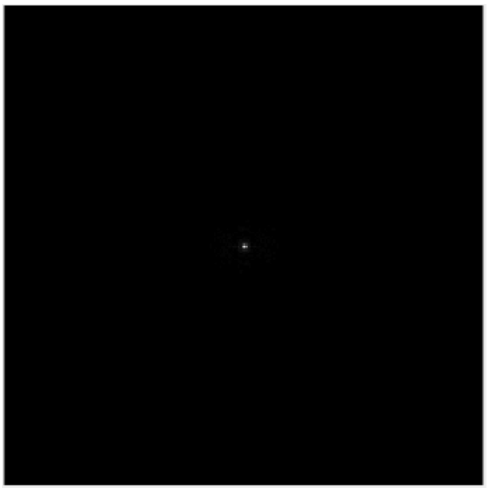
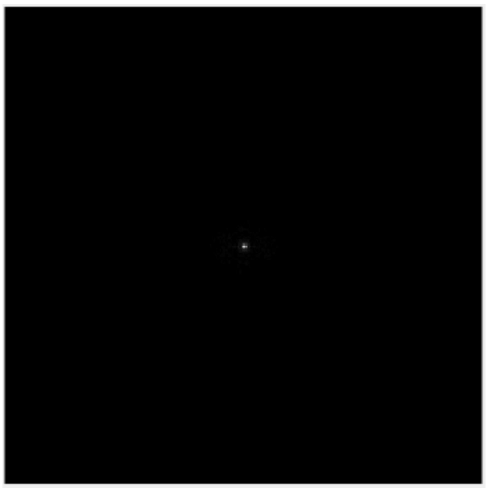
Santiago Herrero Melo

The main task of this lab session is to deblur certain provided blurred images, which are the following ones:

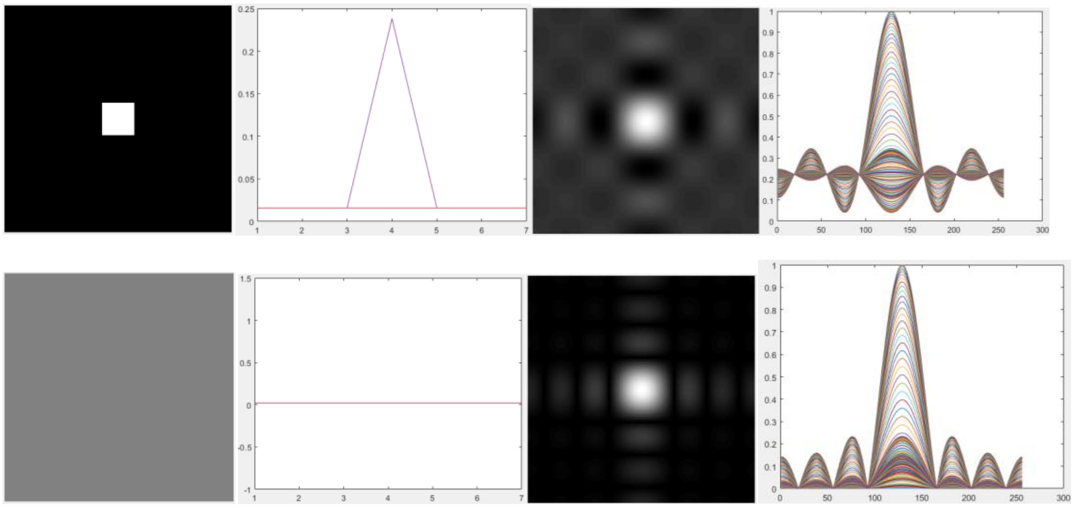



By taking a quick look, we can easily state that the first image is less blurred than the second one. When comparing the blurred images with the original one, we can confirm that theory as the difference of the second image respect to that one y way bigger than in the case of the blurred image of DataOne.

For performing the deconvolution, we will use linear methods, more especifically the Penalized Least Squares method. In order to perform it, we have to work in the frequency domain. The corresponding frequency domain images are the following ones:

In order to deblur the images, we need the impulse response, which is also provided. We have to transform this response into the frequency domain and make its size match with the one of the blurred image. For doing so, we do the N-point Fourier Transformation of the impulse response, where N is the size of the first row of the given image (in this case, it is 256). The obtained results are the following, being the first row the results for DataOne (impulse response in spatial domain and its plotted graph, and impulse response in the frequency domain and its plotted graph) and the second row for DataTwo:



In this case we can very well appreciate that the impulse responses given for each image are very different.

Once this is obtained, we should set the regularization term. This term is based on the following filter and its transpose:

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

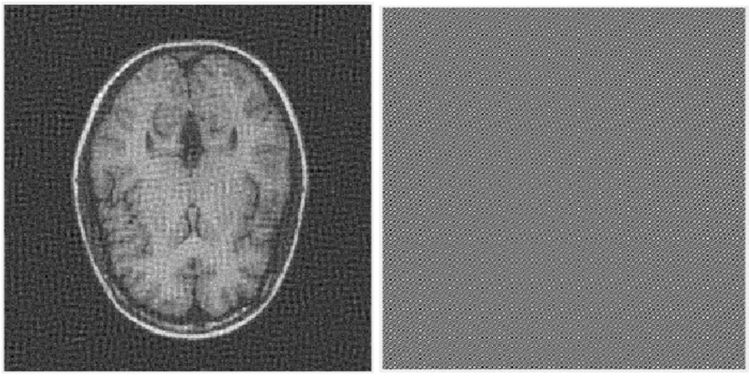
For calculating the deconvolution, the *deconvolusion* function has been created, which takes as input the original blurred image, its corresponding impulse response, the regularization filter and the μ parameter. In order to perform the deconvolution, we follow the following steps:

1. Construct the convolution matrix *h* as the N-point FFT of the impulse response, for which we will use the provided *MyFFT2RI* function.
2. Construct *dx* and *dy*, which are the horizontal and vertical regularization parameters in the frequency domain. For constructing them, we will calculate, separately, the N-point FFT of the regularization parameter and its transpose.
3. Construct the vector g_{PLS} with the following formula:

$$gPLS = \frac{h\,*}{|h|^2 + \mu\,|\mathrm{d}|^2}$$

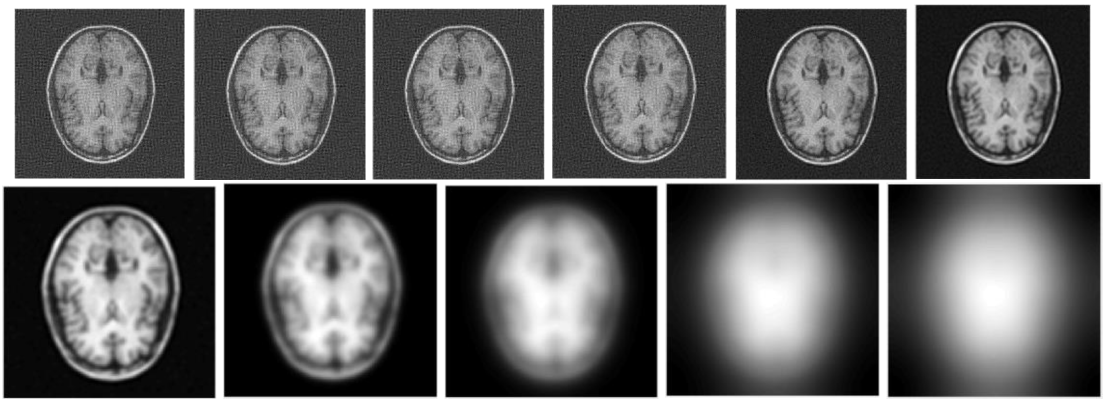
4. Construct *y* as the FFT of the given blurred image, using the provided function *MyFFT2*.
5. Compute *bx* as the point-wise product between g_{PLS} and *y*.
6. Compute the inverse transform of *bx* in order to obtain *x*, the answer in the spatial domain. For doing so, we will use the provided *MyIFFT2* function.

In the first place, we will test the results for the particular case in which μ=0, both for DataOne and DataTwo:

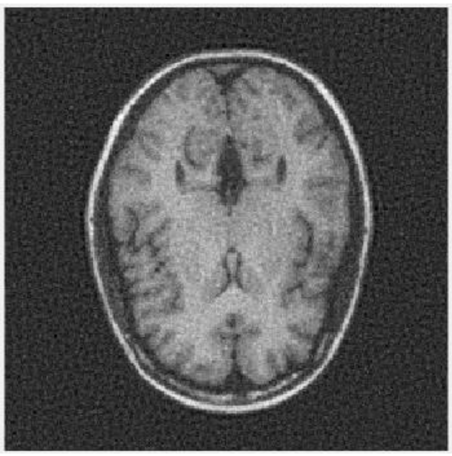


While in the first case we obtain a more nitid image, in the second case we only obtain noise because of the lack of regularization parameter. These images proove that the tuning of the μ parameter is very important.

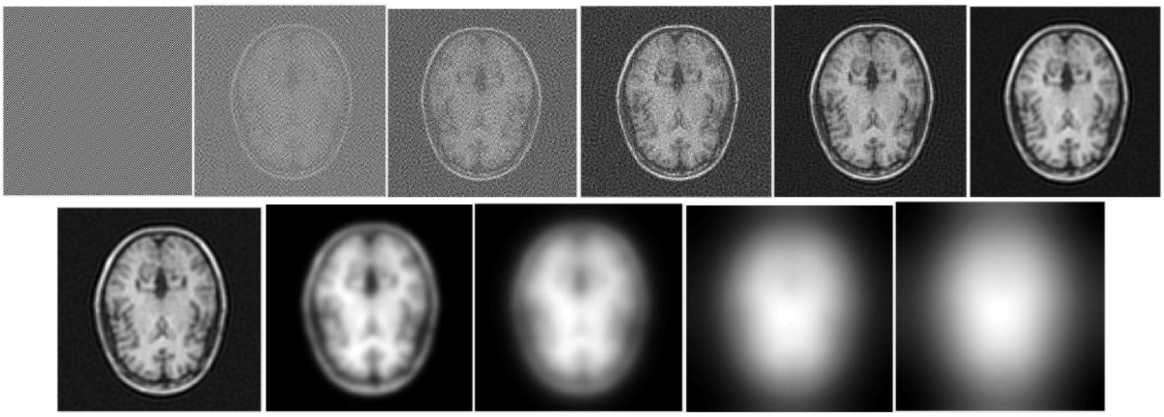
We will test the different possible outputs with different μ values in a logarithmic scale. The presented images have been obtained for values of μ between 10^{-5} and 10^{5}. The results for the first image are:



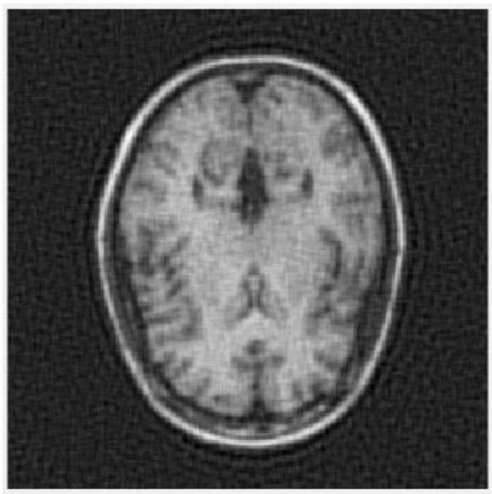
Visually, we can see a significant difference between the values of μ below and over 1. Once its value goes over 1, the images are very blurry, while for inferior values, the results are visually quite similar. Anyway, we can say that the best μ value will be located between 0.001 and 0.01, and by trial and error we obtained the bes μ as μ=0.006, with this result:



In the case of the second data provided, the results are, for the same μ values:



By observing the obtained results, we can determine that the best μ value will be located between 0.01 and 0.1. By trial and error, the best μ value for this dataset has been determined to be μ=0.02, with the following visual result:



Finally, we will compute different distance equations to check how different the deconvolved image (^x) is from the original image (x*):

$$\Delta_1(\mu) = \|\widehat{\boldsymbol{x}}(\mu) - \boldsymbol{x}^\star\|^2 = \frac{\sum_{p,q}(\widehat{x}_{p,q}(\mu) - x^\star_{p,q})^2}{\sum_{p,q}(x^\star_{p,q})^2}$$

$$\Delta_2(\mu) = \|\widehat{\boldsymbol{x}}(\mu) - \boldsymbol{x}^\star\|_1 = \frac{\sum_{p,q}|\widehat{x}_{p,q}(\mu) - x^\star_{p,q}|}{\sum_{p,q}|x^\star_{p,q}|}$$

$$\Delta_\infty(\mu) = \|\widehat{\boldsymbol{x}}(\mu) - \boldsymbol{x}^\star\|_\infty = \frac{\max_{p,q}(|\widehat{x}_{p,q}(\mu) - x^\star_{pq}|)}{\max_{p,q}|x^\star_{pq}|}$$

For DataOne, the best μ obtained with each of the distances, respectively, is μ_1=0.01, μ_2=0.1 and μ_{inf}=0.01. Going deeper, we can determine the values of μ_1=0.027, μ_2=0.1021 and μ_{inf}=0.0089. In this first case, the result of μ obtained visually is not very close to the ones obtained with the distance calculation, just remotely near form the third distance one, It is also important to highlight that the μ values obtained with the three distances are quite different.

For DataTwo, the best μ obtained with each of the distances, respectively, is μ_1=0.01, μ_2=0.1 and μ_{inf}=0.01. Going deeper, we can determine the values of μ_1=0.0236, μ_2=0.0763 and μ_{inf}=0.0074. As seen, although in the previous part of the exercise we didn't explore such deeply the parameters, we got a μ value of 0.2, which is very similar to the one obtained by calculating the first distance.