CS590 – Cortical thickness challenge

Group Partner	Student ID
Pratik More	002308580

Email: pmore21@ubishops.ca

Introduction

In this challenge we have to generate the cortical thickness map of the brain image data that been provided. Cortical thickness is the thickness of the gray matter of the brain at every point. It is the distance between the white matter surface and the pial surface.

Steps

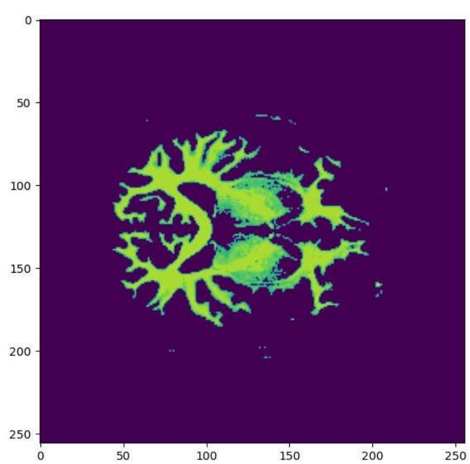
1) Loading and Reading image

We have used the nibabel libraries to load the nifty image, and by using matplotlib libraries we are displaying the plots.

2) Segment the white matter

We start by segmenting the white matter by using a specific tolerance value.

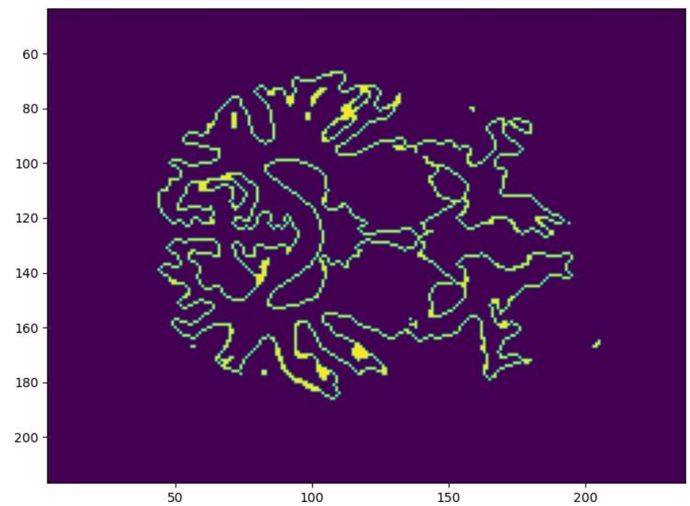
Following is the plot for white matter



3) Segment the white boundary

From the white matter, we use binary dilation which is a morphology technique in which we dilate the white matter and then subtract the dilated part from the white matter. Then we select all the values which are 1 and rest all we changed it to 0, from this we get the white matter boundary.

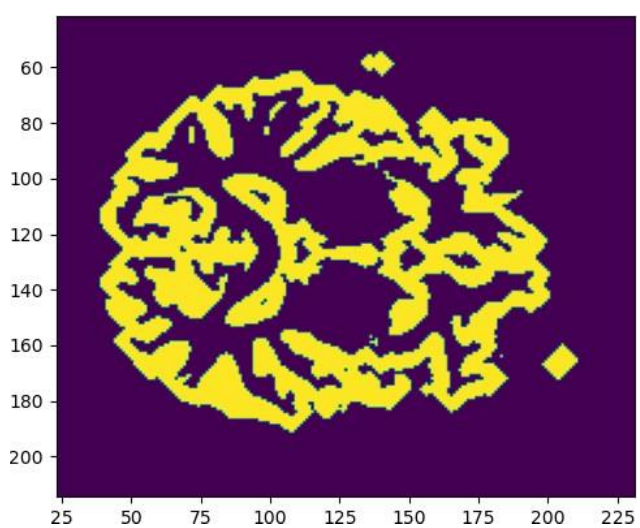
Following is the plot for the white boundary



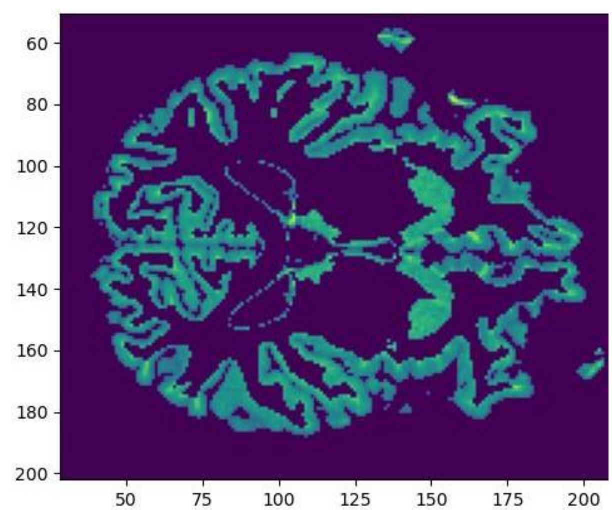
4) Segmenting Gray matter

Now, we dilate the white matter till it covers all the grey matter and a little bit outside of the gray matter. Then by using tolerance we removed the outside voxels of gray matter to get the actual gray matter.

Gray matter before tolerance



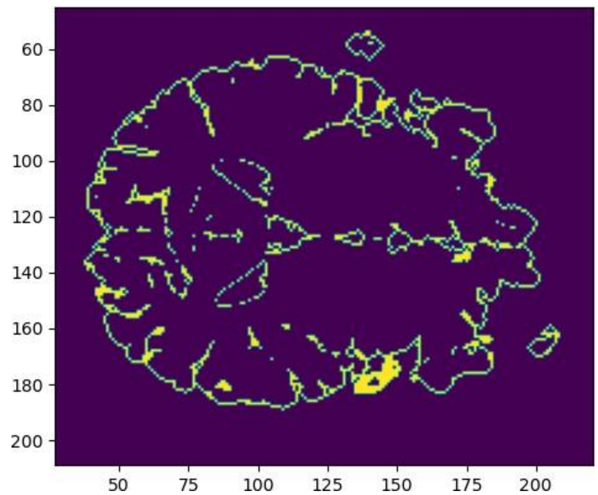
Gray matter after tolerance



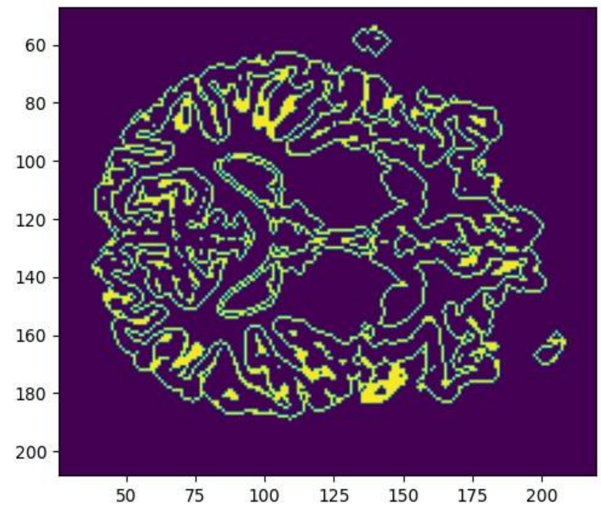
5) Pial boundary

To get the Pial boundary, we dilate the gray matter and then subtract the original gray matter from the dilated gray matter.

Pial Boundary



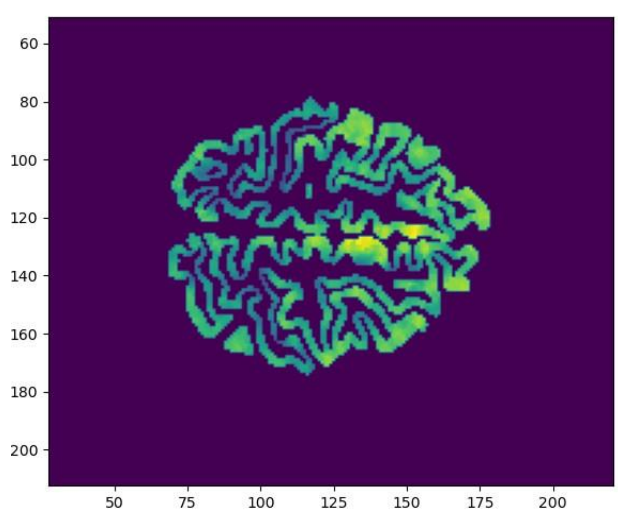
Pial and white matter boundary



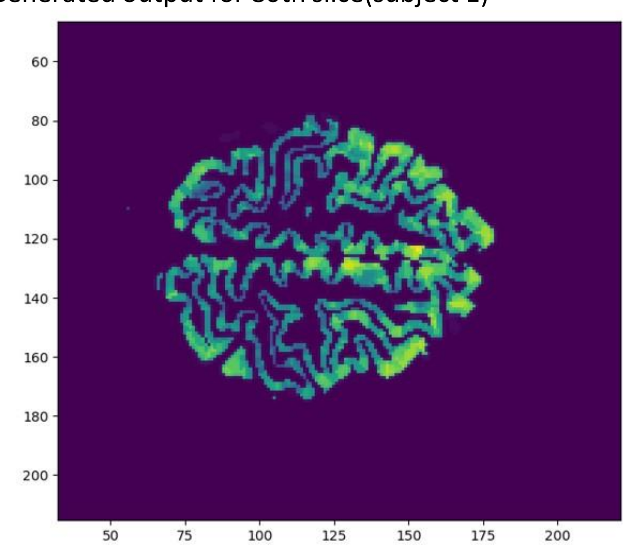
6) Thickness estimation

For estimating the thickness, we first need to find the smallest distance of every white matter boundary voxel with the pial boundary voxel. By using KDTree we find out the smallest distance between two points. Then, we replaced the WM boundary co-ordinate value with the smallest distance we found. Next, we calculated the smallest distance for each gray matter voxel with WM boundary voxel, and then replaced the gray matter voxel value with the WM boundary voxel value.

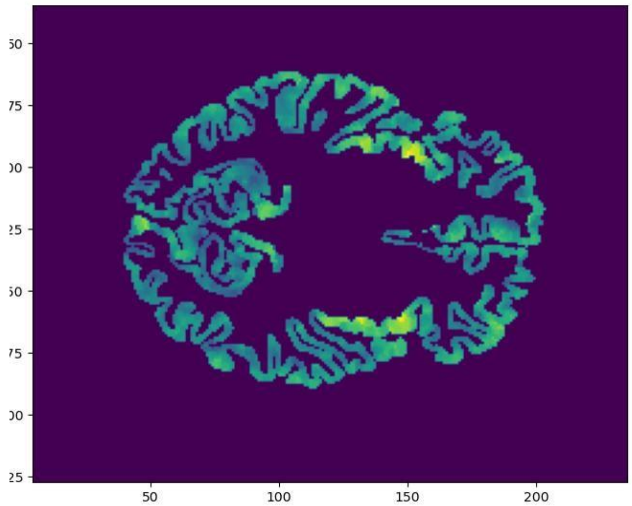
Expected output for 80^{th} slice(subject 1)



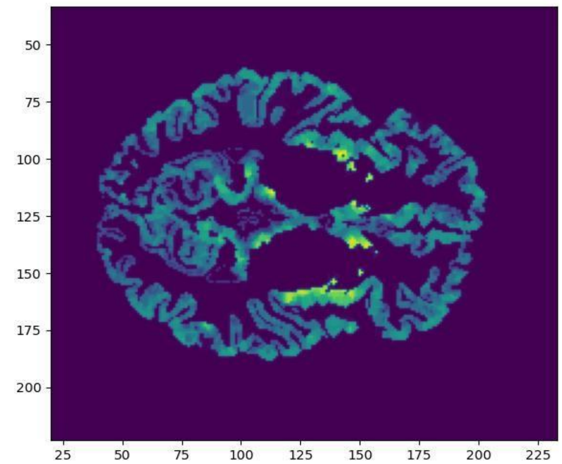
Generated output for 80th slice(subject 1)



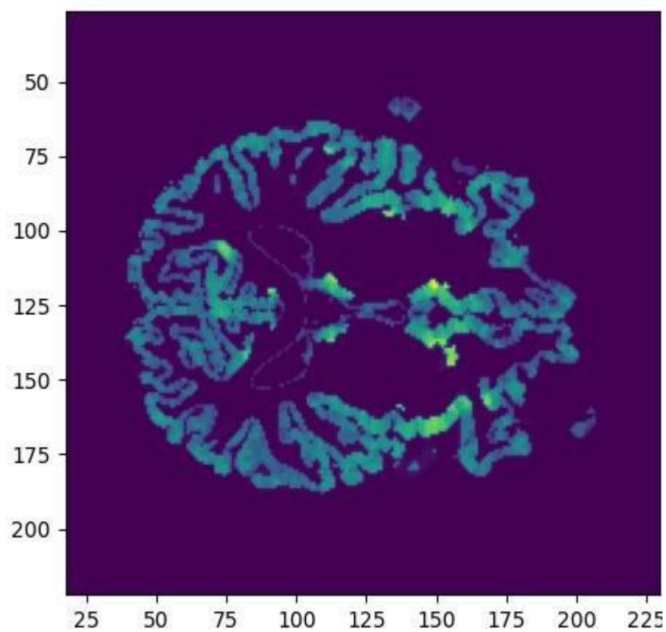
Expected output for 128th slice(subject 1)



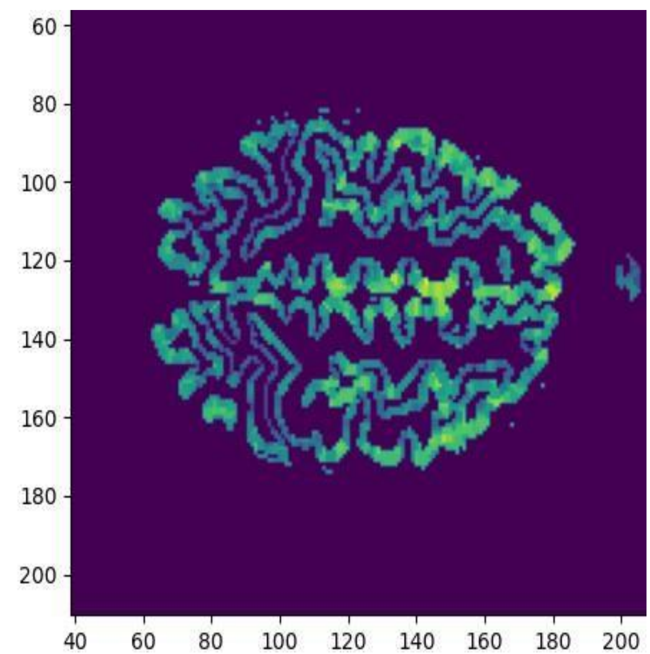
Generated output for 128th slice(subject 1)



Final Output for Subject 2 (Slice 128)



Final Output for Subject 2 (Slice 80)



Challenges

1) Calculating Euclidean distance using two nested loops for all the points was taking lot of time which was not a feasible solution.

2) Selecting the tolerance value for the segmentation to get the accurate results was bit challenging.