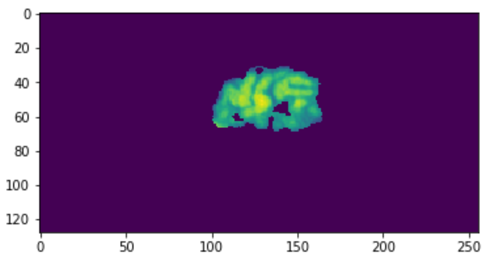
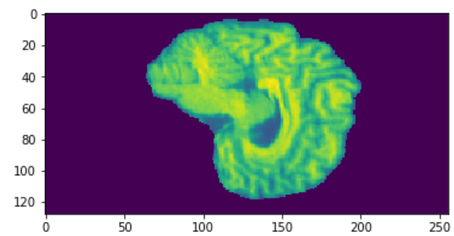
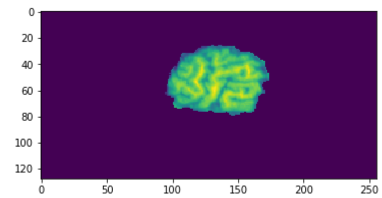
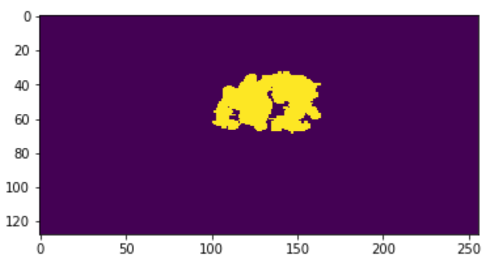
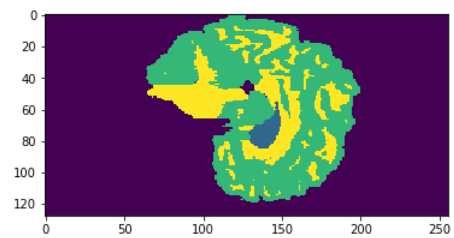
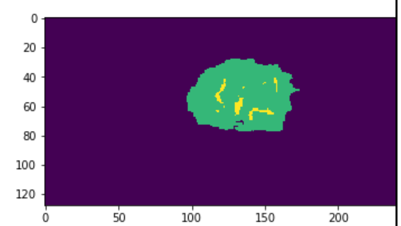
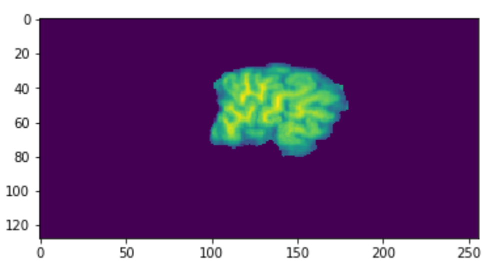
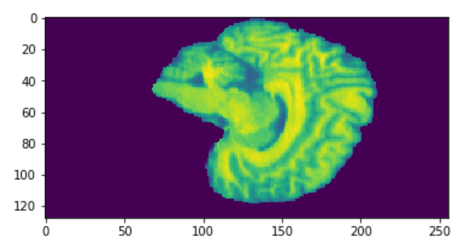
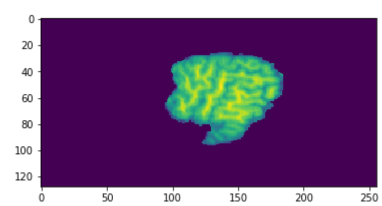
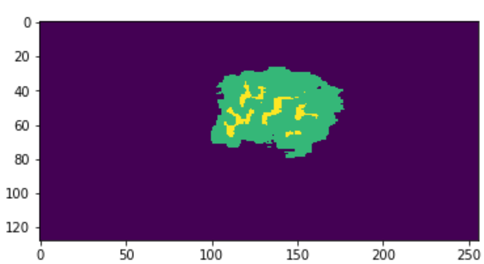
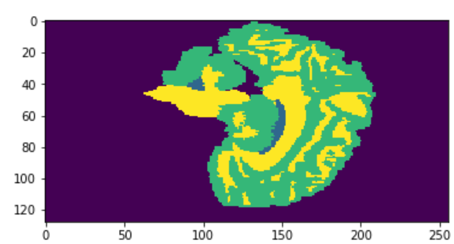
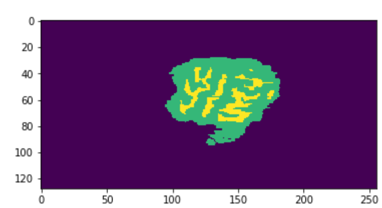
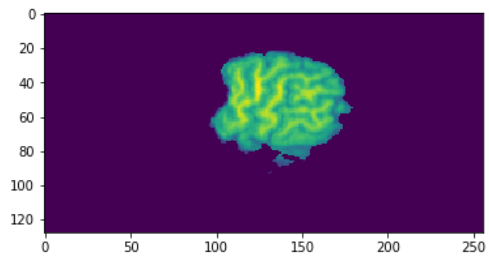
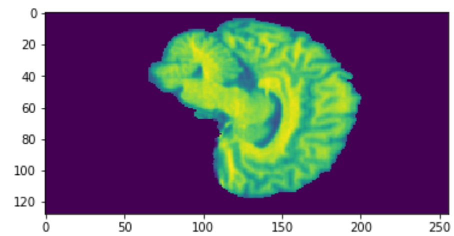
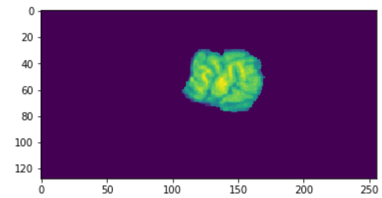
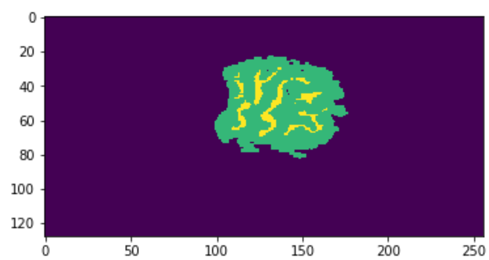
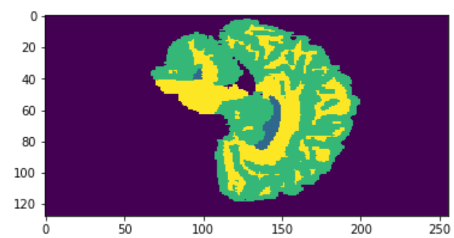
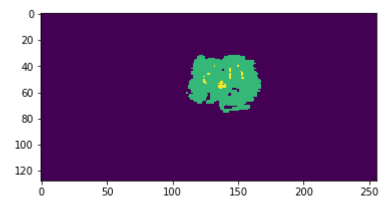
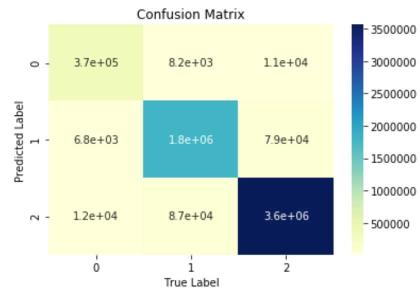
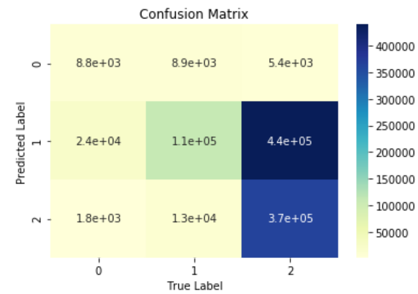
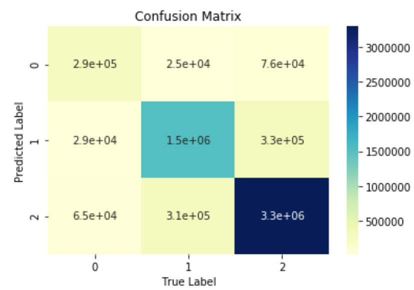
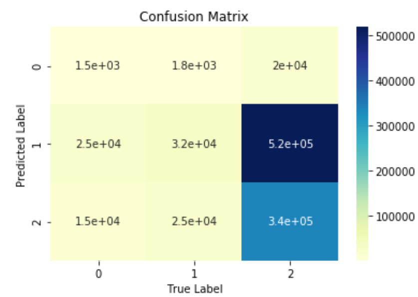
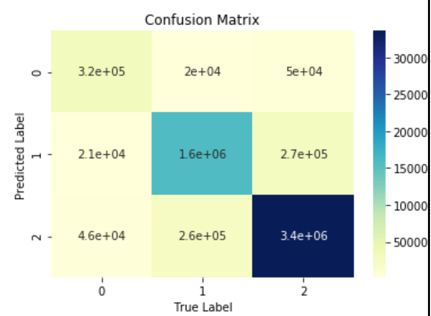
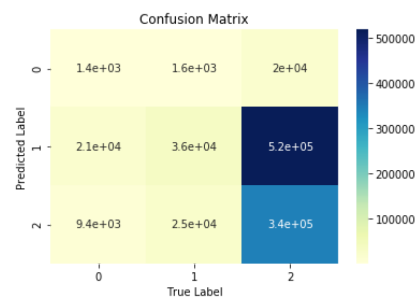
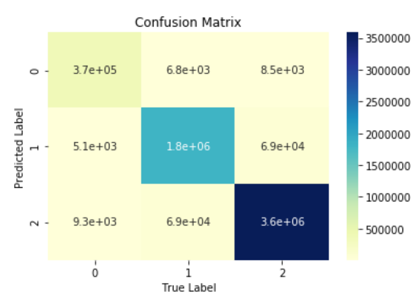
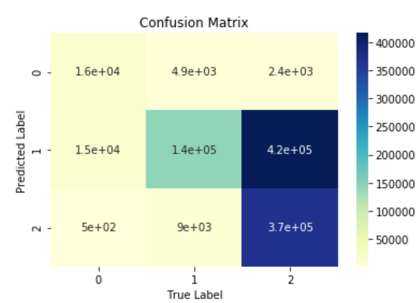
2.1
Qualitative analysis of dataset

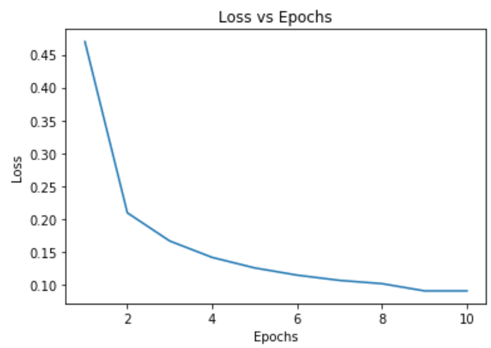
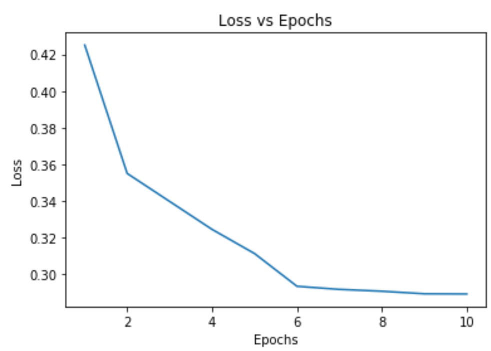
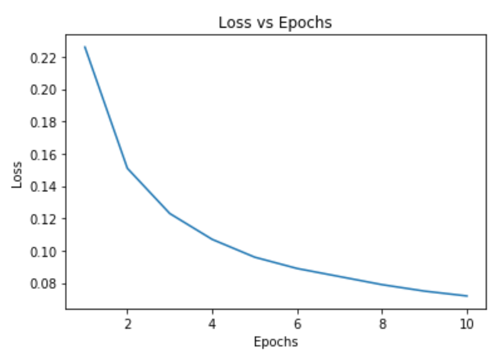
	Cross section 1	Cross section 2	Cross section 3
	Data		
Vol 1			
	Ground Truth		
			
	Data		
Vol2			
	Ground Truth		
			

	Data		
Vol3			
	Ground Truth		
			

2.3

	Dice Coefficient	
Model	Training(Best Model)	Validation
1.	0.96	0.34
2.	0.86	0.31
3.	0.88	0.32
4.	0.96	0.37

<table>
<tr><td></td><td colspan="2">Confusion Matrix</td></tr>
<tr><td>Model</td><td>Training</td><td>Validation</td></tr>
<tr><td>1.</td><td>
</td><td>
</td></tr>
<tr><td>2.</td><td>
</td><td>
</td></tr>
<tr><td>3.</td><td>
</td><td>
</td></tr>
<tr><td>4.</td><td>
</td><td>
</td></tr>
</table>

<table>
<tr><td></td><td>Convergence Plot</td></tr>
<tr><td>Model</td><td></td></tr>
<tr><td>1.</td><td>
</td></tr>
<tr><td>3.</td><td>
</td></tr>
<tr><td>4.</td><td>
</td></tr>
</table>

Models :

1. Baseline Architecture
2. Baseline with Upsampling done via strided, 0 padded Deconvolution Layers
3. 2 + Downsampling done via strided Convolution
4. 1 + Skip Connections

Although training results showed good confusion matrices and dice coefficient. Result in validation data was disappointing. The best confusion matrix and dice coefficient for validation set was the one with skip connections. Skip connections + 3rd model also gave better results.

DC is not improved on replacing pooling and unpooling layers with convolution and deconvolution. Maybe this is because the number of parameters to train greatly increases with the use of convolution and deconvolution layers. Model complexity increases and there is overfitting.
When only deconvolution is used we get poorer results than using both deconvolution and convolution. This may be because of the symmetric kind of architecture with conv layers for downsampling and deconv layers for upsampling.
With the use of skip connections there has been an increase in dice coefficient. Since there is information loss in downsampling while pooling, when we concatenate layers from encoder to decoder, there is better recovery of the information hence giving better segmentation results.

Confusion matrix is not improved with replacing of pooling with conv and deconv. But on adding skip connections even the validation set has a better confusion matrix than obtained otherwise.

While looking at convergence plot model from 6 - 8 epoch seems like the best one

2.2
Choice of Architecture-
The architecture chosen is the skeletal architecture of Segnet. Segnet is a deep full CNN architecture which is adapted for semantic segmentation. Downsampling has been done using max pooling and upsampling has been done using unpooling in the baseline architecture. The skeletal architecture consists of 6 blocks, 3 of encoder and 3 of decoder. It was neither too simple nor too deep for our dataset. A softmax layer is added at end to obtain the probabilities of each pixel to belong to different classes.
For part 3. Unpooling layers are replaced by Deconv
For part 4 Pooling layers are replaced by Conv
For part 5 skip connections added between corresponding encoder and decoder

Preprocessing-
Data set was given in the form of brain volumes in .mat format. After initial steps of preprocessing to obtain data in the form numpy arrays, dataset is iterated along the third dimension to obtain 2D images of size 256 X 128.
There are 4 classes in this image segmentation problem. 0 class is assigned for background. As found from analysis pixel from class 0 are huge in number dominating other classes. Some images had only background class pixels. Moreover there was huge class imbalance as very few class 1 pixels are present in the images. On top of this we have a very small dataset(3 volumes, 786 images) which makes the model prone to overfitting.
To overcome the above problems, 2D patches have been made from each image where we have at least 1 pixel of ground truth 1 and no pixels of background class 0. Using this technique we have 5798 patches. In this way we perform data augmentation and handle class imbalance to a great extent.

Different preprocessing implementation gave different results. One such implementation gave a dice coefficient of 0.186 on baseline architecture.

Loss Function- Cross Entropy loss has been used. It is a good loss for classification. In this problem we classify pixels into 3 classes. Cross entropy minimizes the difference between predicted and true probabilities.

Training Strategy-
Epochs - Until convergence
Learning rate - 0.0001
Evaluation avoiding background- During training while evaluating for dice coefficient and confusion matrix we consider only 3 classes since we have removed background class 0. After forward propagation is performed for each sample, backward propagation is done.
Optimiser used - SGD