Young, James Yang
20740589

ELEC4840 Assignment 1 Report

Problem 1

In problem 1, my main task was to implement the ResNet-50 classification network to predict the lesion disease state of benign or malignant using the ISIC Challenge 2016 dataset.

The first step was to split the dataset into training, validation, and testing sets. The dataset already has a training and testing split (900 training and 379 testing), so for creating the validation set I split the testing dataset into around half (190 validation, 189 testing) so that my dataset ratio is around 70% training, 15% validation, and 15% testing. Then I applied random cropping and some flipping to augment the dataset to create variation into the dataset for improving the model accuracy.

After implementing the ResNet-50 model in Pytorch, I trained the model on the skin lesion dataset using SGD as my optimizer and cross entropy for my loss function. For other hyperparameters, I set the learning rate to 1e-3 and SGD momentum to 0.9. Figure 1 shows the loss and accuracy curves during training and validation for 10 epochs. My best model result on the test set was an accuracy of 83.07% and an AUC of 68.4%.

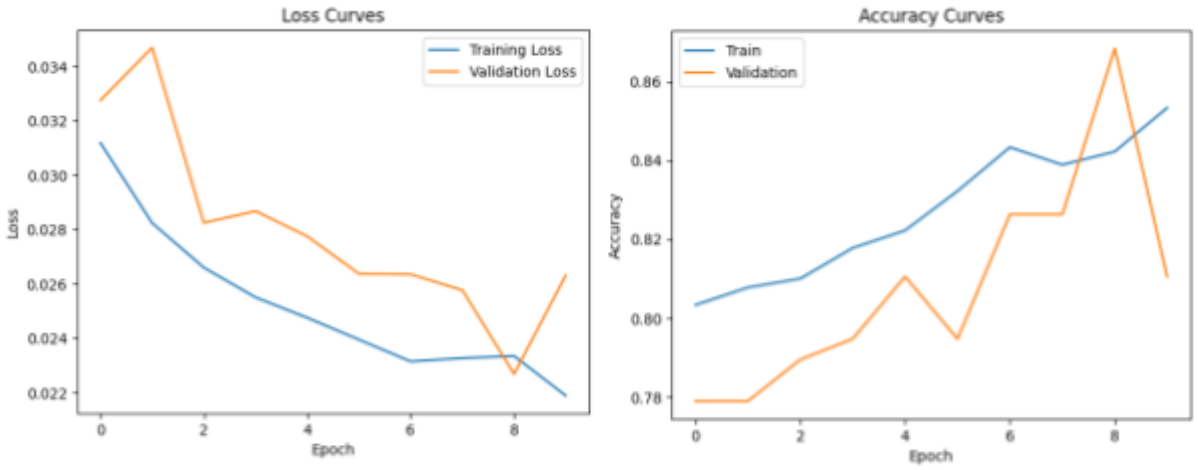

Figure 1: Loss and Accuracy Curves

However, the dataset has a class imbalance issue, where the number of benign data is much higher than the number of malignant data. To deal with this issue, I decided to add weights to the cross entropy loss function. The idea is to give more weight to the minority class so that

the model learns more equally from the classes. The loss and accuracy curves of using the weighted loss function is shown in Figure 2. With this implementation, the best model was able to get an accuracy score of 84.13% and an AUC of 71.27%. As seen in the results, this method does tackle the class imbalance as seen by the increase in the AUC score.

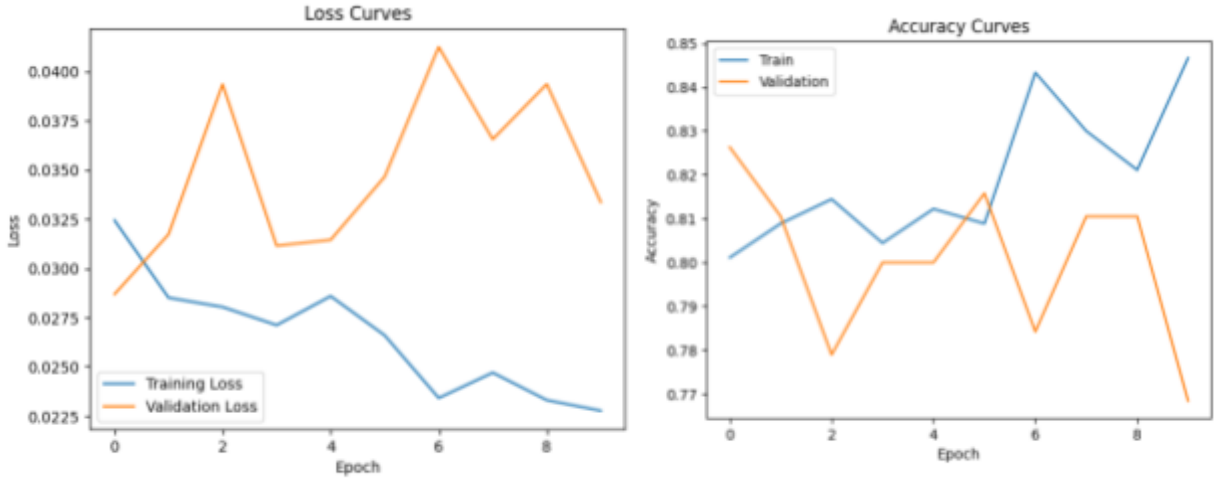

Figure 2: Loss and Accuracy Curves with Weighted Loss Function

Problem 2

In problem 2, our task was to segment 3D abdomen CT images that contain spleen using 3D-UNet. We had to first implement the data loader and augment the data. Then we had to implement the 3D-UNet model before setting up the training, validation, and testing loop.

The table below (Figure 3) shows the 4 evaluation metrics for the dice, jaccard, the average surface distance (ASD), and the 95% Hausdorff Distance (95HD) on the test set. Figure 4 shows the loss curves and dice curves which were generated using Tensorboard.

Figure 3: 4 Evaluation Metrics

Dice	Jaccard	Average Surface Distance (ASD)	95% Hausdorff Distance (95HD)
82.3%	0.713	3.7163	26.55

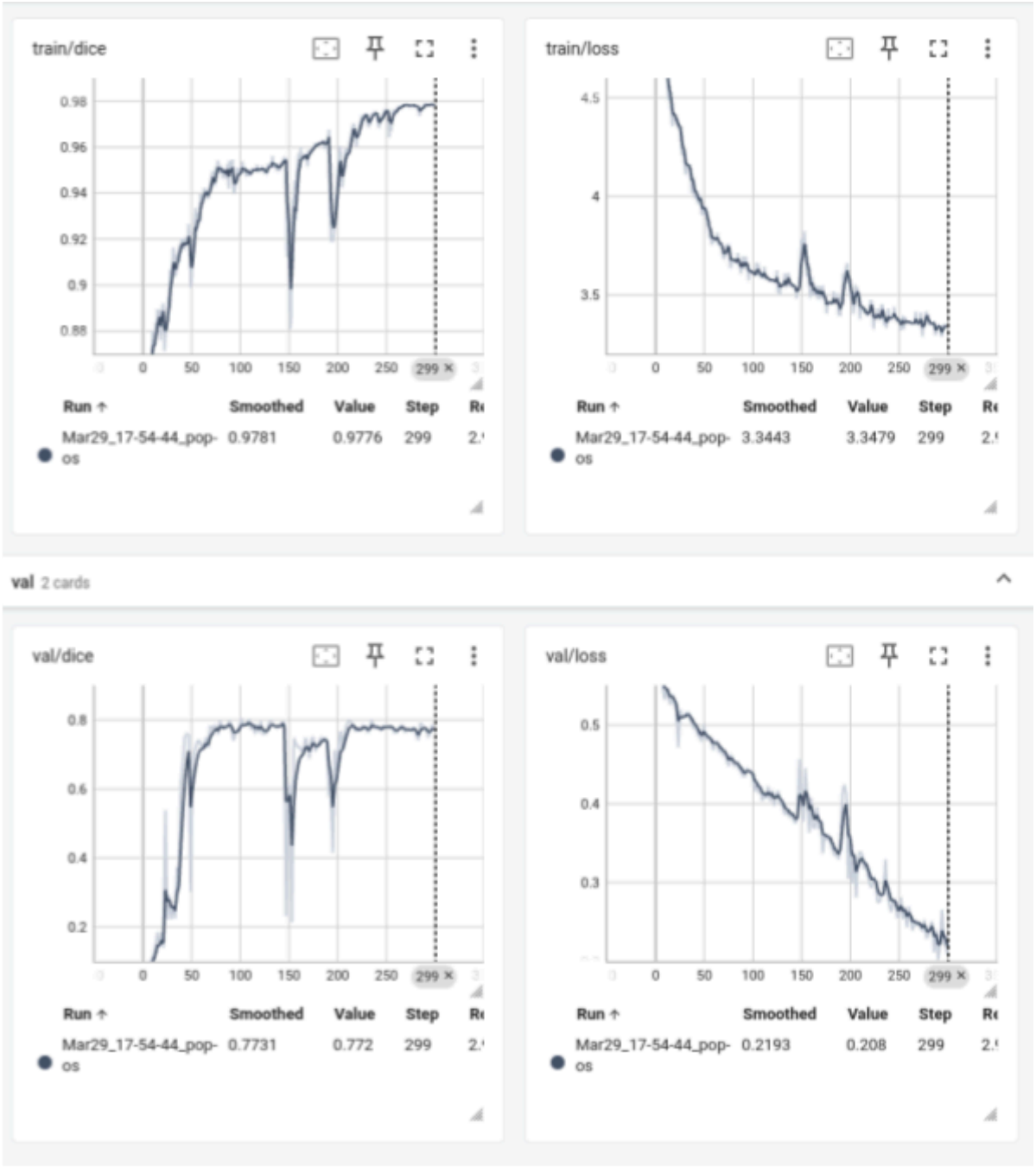

Figure 4: Dice and Loss Curves

The images below show 4 segmentation results compared with the ground-truth label.

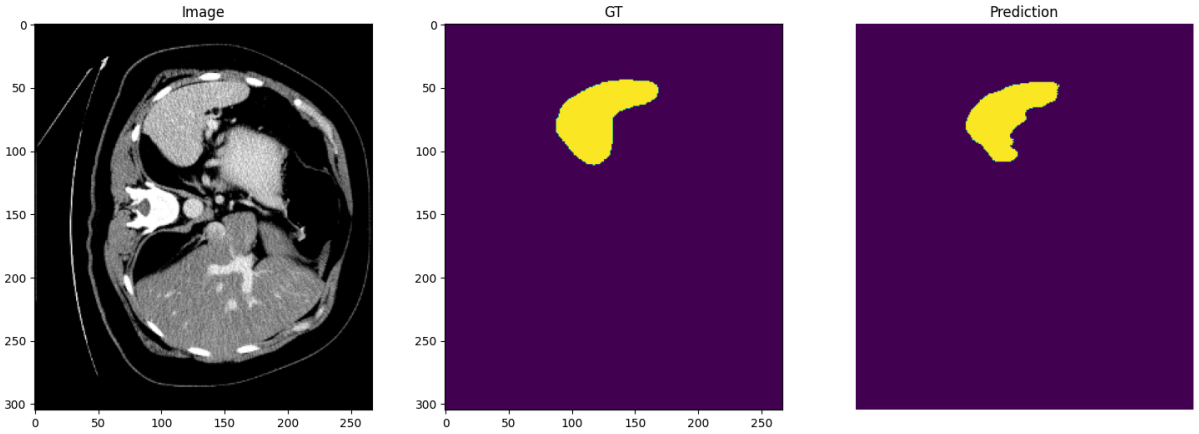
Image
GT
Prediction

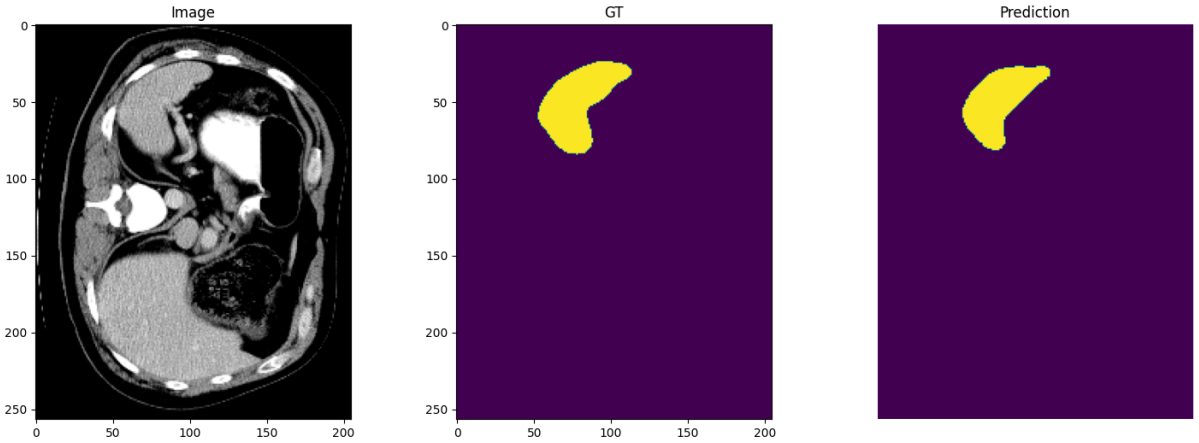
Image
GT
Prediction

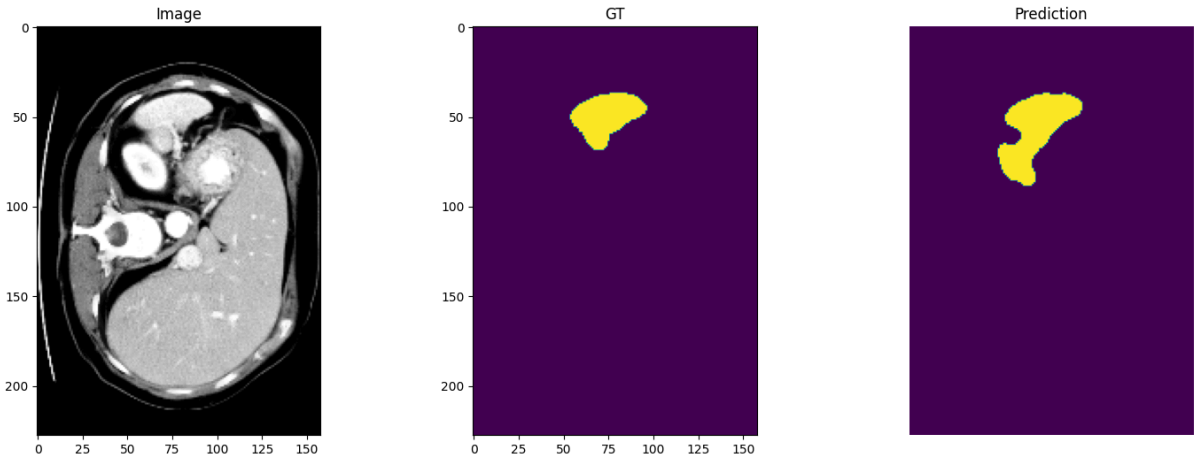
Image
GT
Prediction

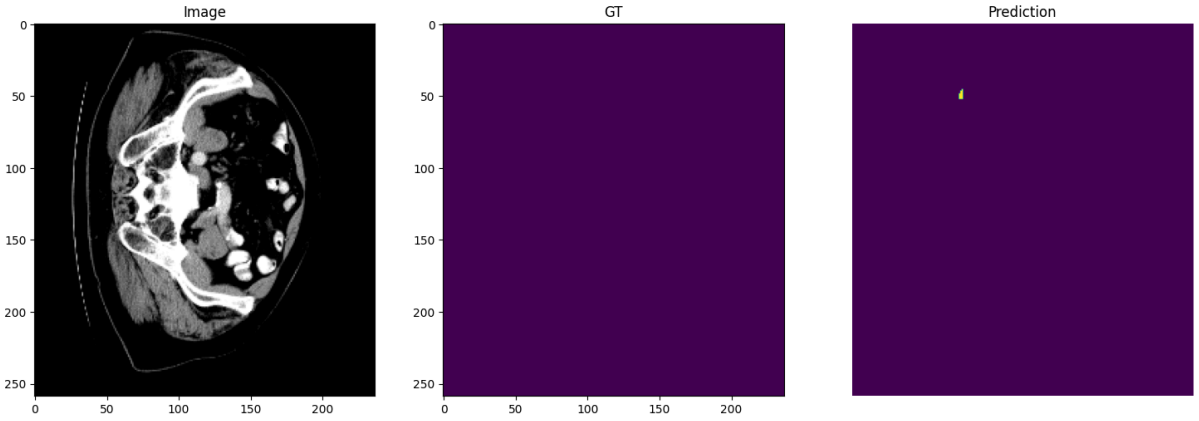
Image
GT
Prediction
0
50
100
150
200
250