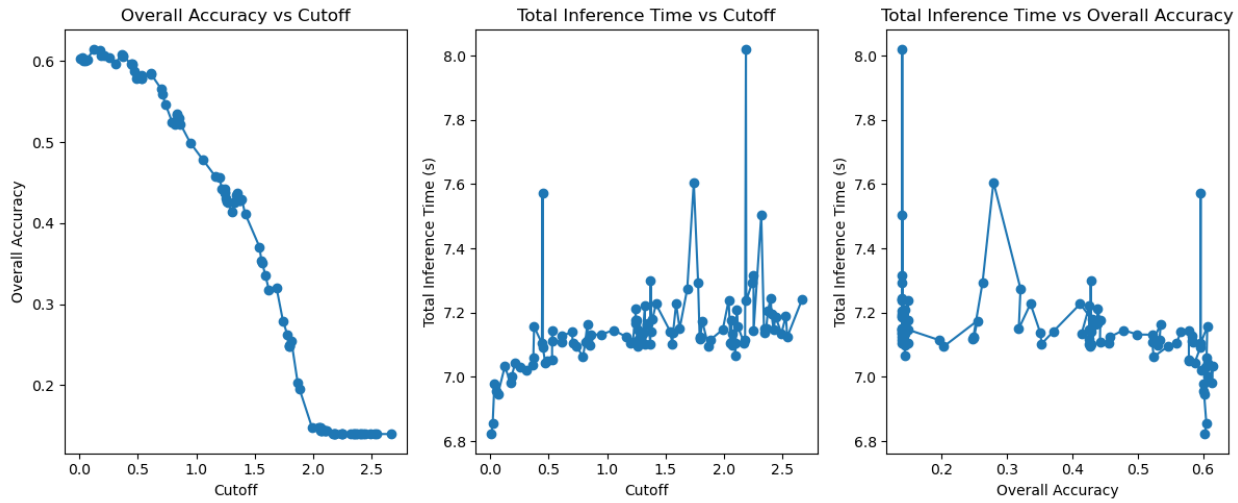
Assignment 3

Shounak Naik

The following graph shows results for Part 1).

We exit a sample if the entropy is greater than the threshold. Greater entropy means greater confidence in prediction.

For plotting the following we have ranged the cutoffs from 0 till log(10) which is 2.71.
For this part we have kept the thresholds the same across all layers. We see an inverse relation between overall accuracy and cutoffs. We use this fact in Part 2. The overall accuracy is calculated as the weighted sum of accuracies across different layers with the weight being the number of samples exited from each layer.



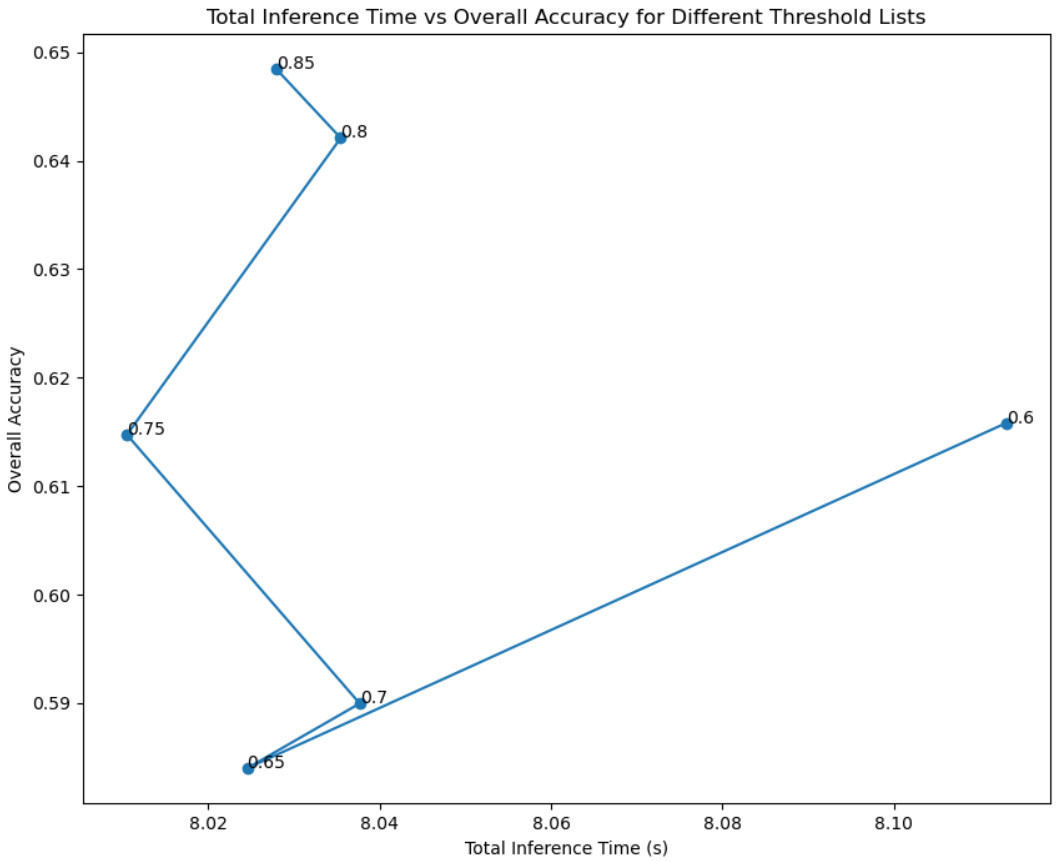
Part 2

The way I have implemented the estimation function is as follows- I iterate over the entire batch for each layer and if I get the accuracy greater than the desired for that layer , I record the entropy threshold for that layer to be 0. This basically means that all samples give the desired accuracy without any threshold.

If this is not the case, I record all the entropies in the validation data for that layer. I approximate by saying that if I want 80% accuracy, I will have to take the threshold to be 20th percentile in all the entropies recorded. This is not foolproof, but this made sense with the figure 1 as well where we see an inverse relation between accuracy and cutoff.

Thus I would get approximate thresholds for each desired accuracy for each layer.

I test this for many different desired accuracies as told in Part 2B) I get the following graph. In this graph the numbers denote the overall weighted accuracy I get when setting the desired accuracy to the number plotted for each layer.



For every desired accuracy we would get a set of 6 parameters which denotes the thresholds required for that configuration. Now we wish to choose the best set of parameters. We will have 6 sets of parameters here since we have tested on 6 different accuracies.

We choose the best parameters by getting the configuration which gives us the best accuracy in the least time(latency). To do this we choose the configuration which has the highest accuracy to latency ratio.

```
Estimated Thresholds for Desired Accuracy 0.8: [0.132112, 0.13962515, 0.033455838, 0.0, 0.0, 0.0]
(DL) dell@dell-G3-3579:~/On-Device-DL$ python3 hw3_submission.py
Files already downloaded and verified
The list denotes the accurary and time at each exit_index
Layerwise Accuracy: [[0.44304311 0.61053008 0.59904672 0.69958333 0.5        0.97962429]]
LayerWise Time: [[1.36704946 2.09311557 0.76246428 1.03013968 0.46919775 0.73644376]]

Overall Accuracy (Fixed Cutoff [0.6, 0.6, 0.6, 0.6, 0.6, 0.6]): 0.5856
Total Inference Time: 6.4584 seconds
Estimated Thresholds for Desired Accuracy 0.8: [0.132112, 0.13962515, 0.033455838, 0.0, 0.0, 0.0]
Best Set is for desired accuracy :0.85
Best Threshold is :[0.07269455, 0.087526366, 0.018168697, 0.018542826, 0.0030984946, 0.004078076]
Test Accuracy using the Best Threshold: 0.6484830217846507
Inference Time on Test Data using the Best Threshold: 7.224052667617798 seconds
(DL) dell@dell-G3-3579:~/On-Device-DL$
```

The terminal shows the best set of parameters chosen layer-wise [0.072, 0.08, 0.01, 0.01, 0.003, 0.004]. This gives us a weighted accuracy of 0.64.