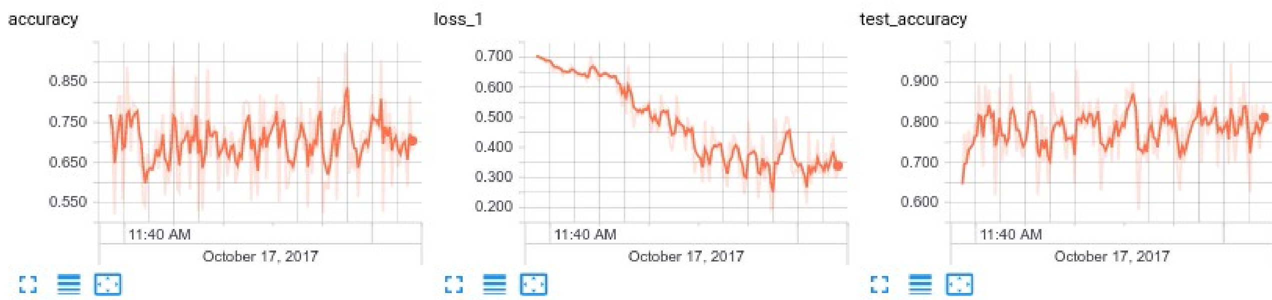
Nathan Day
CS 501R – Deep Learing
Lab 6 – Write Up
October 17, 2017

Lab 6 – Cancer Detection

What is the final classification accuracy on the test set before and after regularization?

The following plot is of the training and test accuracy without an regularization:



Final Training Accuracy: 72%

Final Test accuracy: 82%

The following is a plot of training and test accuracy using the the dropout method with a dropout probability of 0.6:

Final Training Accuracy: 92%

Final Test accuracy: 84%

The following is a plot of accuracy using L2 regularization with a lambda of 0.001:

Final Training Accuracy: 91%

Final Test accuracy: 86%

Is the generalization error better or worse?

The generalization error for the L2 regularization was better than the Dropout method. It seems that the dropout method should be more accurate, so it seems that there is a problem with me network that isn't very apparent.

If you used dropout, what dropout probability did you use?

I used a dropout probability of 0.6.

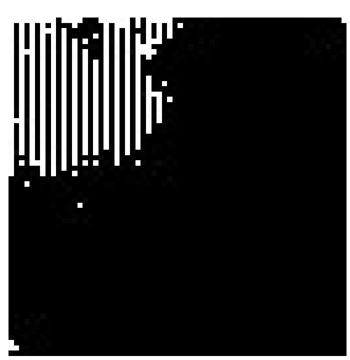
If you used L1/L2, how did you pick lambda?

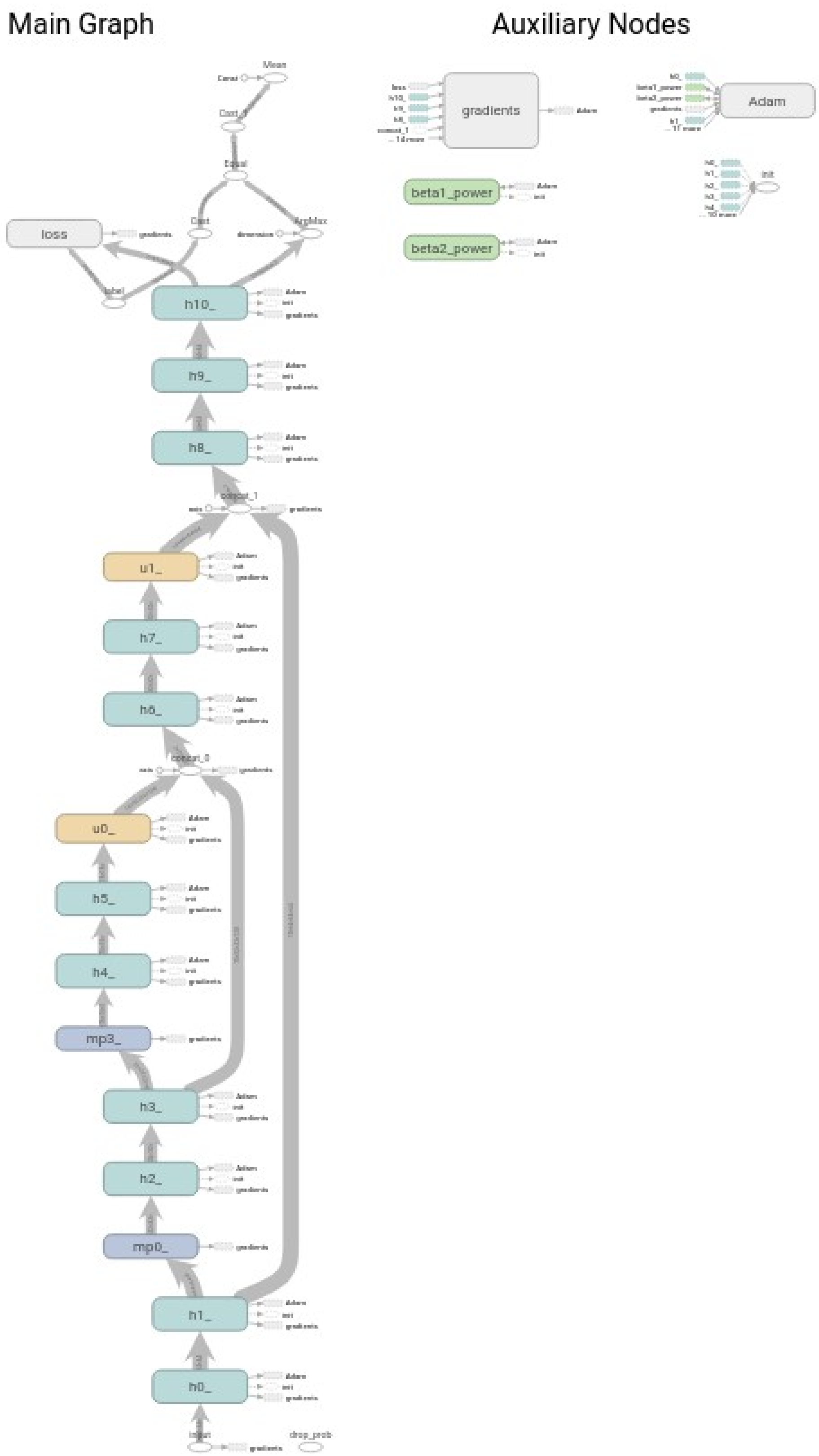
I used a lambda of 0.001. I chose 0.001 because anything larger lambdas made the output masks all black.

What is the form of your network?

I used the U-net structure recommended by Dr. Wingate. I ended up using an image size of 64x64 because of my lack of computing power. The form of the neural network can be seen on the following page. I also trained using 150 random images in batches of 10, and tested using 10 random images. I calculated the accuracy based on the accuracy of each pixel. The accuracy is somewhat high, because most of the images are black, so even a prediction of a completely black image would have high accuracy. The loss function was calculated using the softmax function on the labels and the last layer of the neural network.

What does the final prediction for the specified image look like?




Main Graph
Auxiliary Nodes
Mean
Const
Cast_1
Equal
loss
Cast
dimension
ArgMax
label
h10_
h9_
h8_
concat_1
axis
u1_
h7_
h6_
concat_0
u0_
h5_
h4_
mp3_
h3_
h2_
mp0_
h1_
h0_
input
drop_prob
gradients
Adam
init
beta1_power
beta2_power