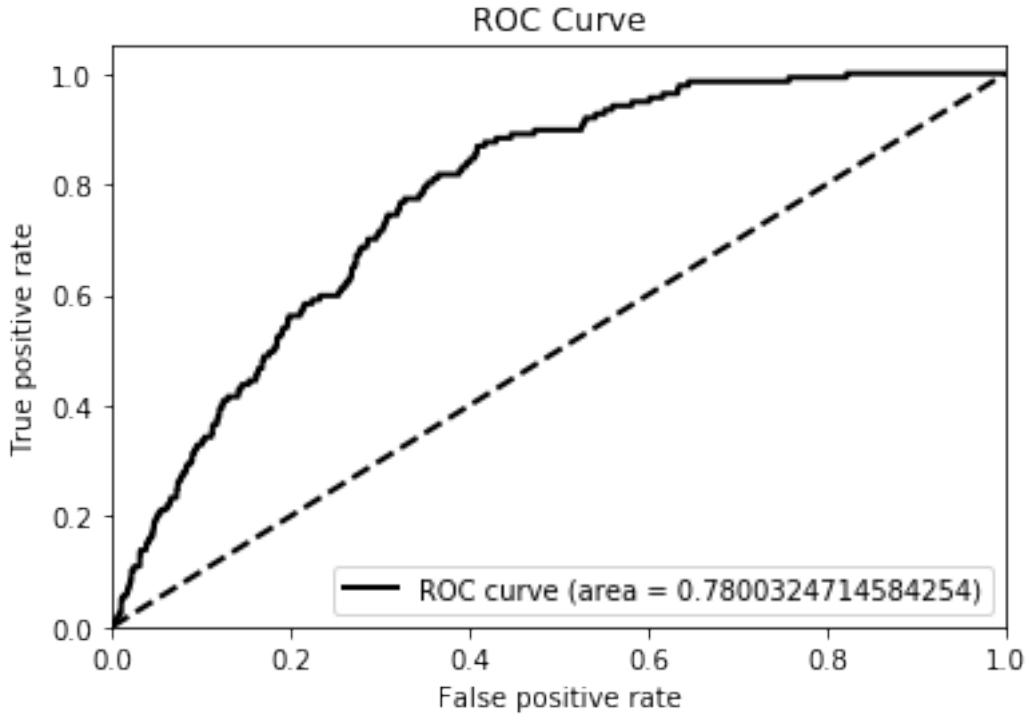
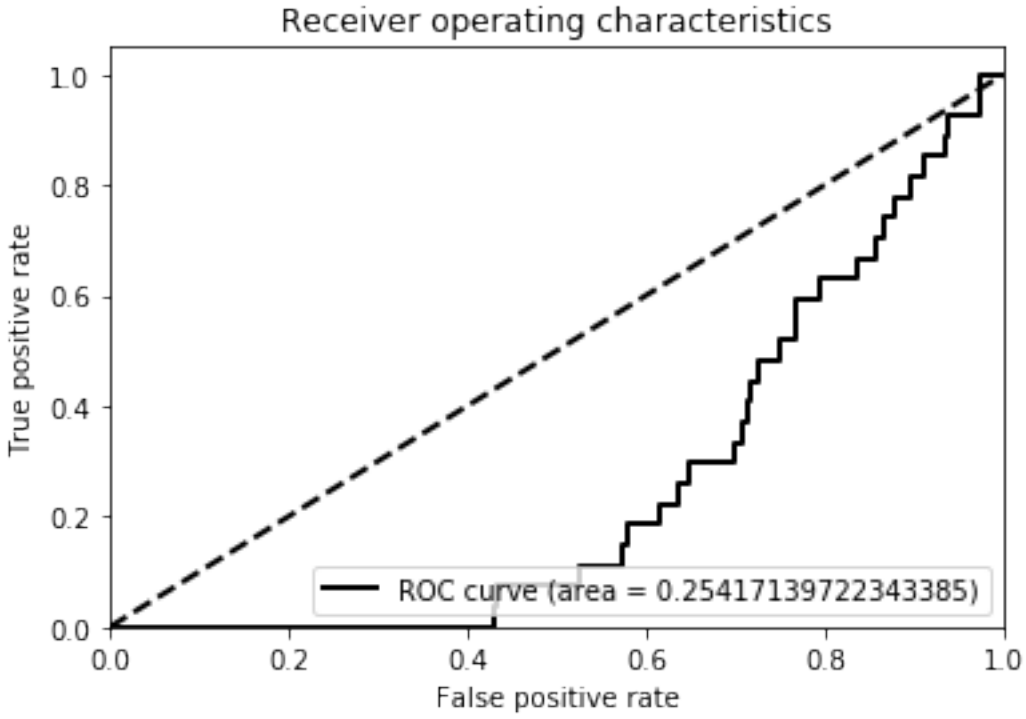
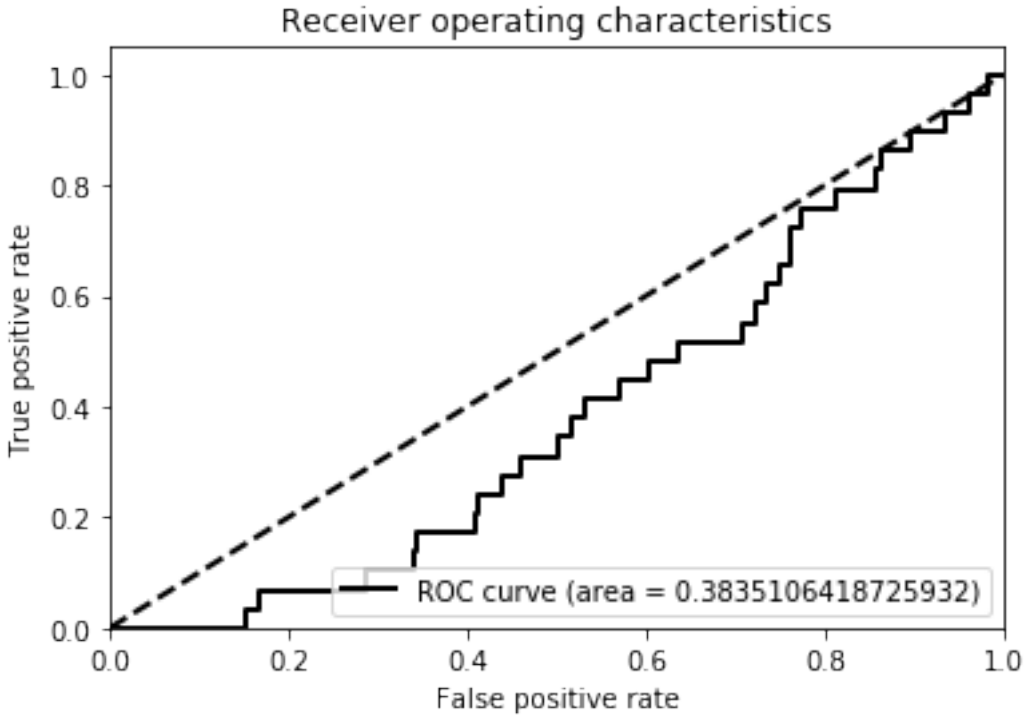
Snow Naing
BMI203
Final Project Redo: Neural Net - Distinguishing binding sites of a transcription factor, RAP1

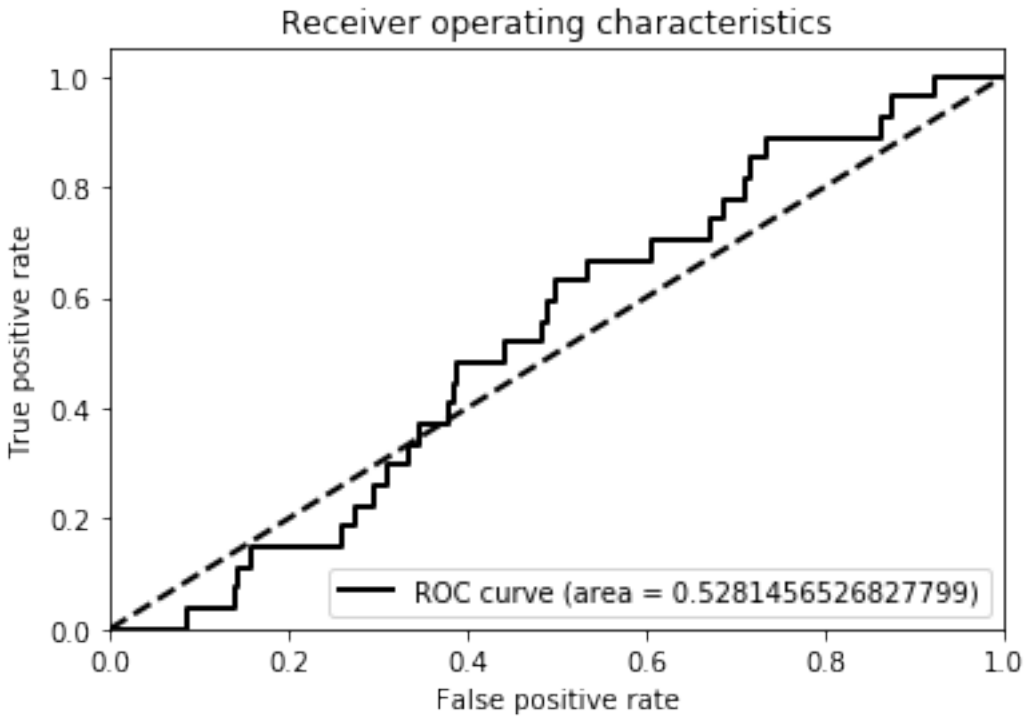
- My codes for neural net can be found in "neuralnet.py" and the input for neural net can be found in "io.py". For my neural net, there are three different layers – input, hidden and output. The layer sizes can be adjusted.
- For identity matrix autoencoder, I used input, hidden and output layer size as 8,3,8 respectively. For input, I chagned the DNA string to a binary 4-bit vector, which makes the input into my neural net 68 nodes (17bp x 4-bit per bp). I also got rid of the positive binding site from the whole genome and the remaining sequences are used as negative test data.
- Weight matrices are placed between input and hidden layers, and between hidden and output layers. For activation function, I used a sigmoid function. Depending on the weight matrix and activation functions in each layer, the output is calculated in "forward" function. To avoid overfitting, I add a regularization parameter to the cost function. For gradient descent, I take derivatives of sigmoid function and continue doing more derivatives for batch gradient descents.
- For training, my function takes in number of iterations, regularization parameter, true positive and true negative files and output a predicted output file. To take into considerations of the background errors, I take stochastic training. I trained all my samples on five batches of samples for both positive and negatives by dividing the length of the positive and negative files by five.
- For the testing data, I tested with 100 iterations and used 200 hidden layers. The scoring for the testing data can be found in "yhat_test_snow.txt". The ROC curve for my training data can be seen as below.

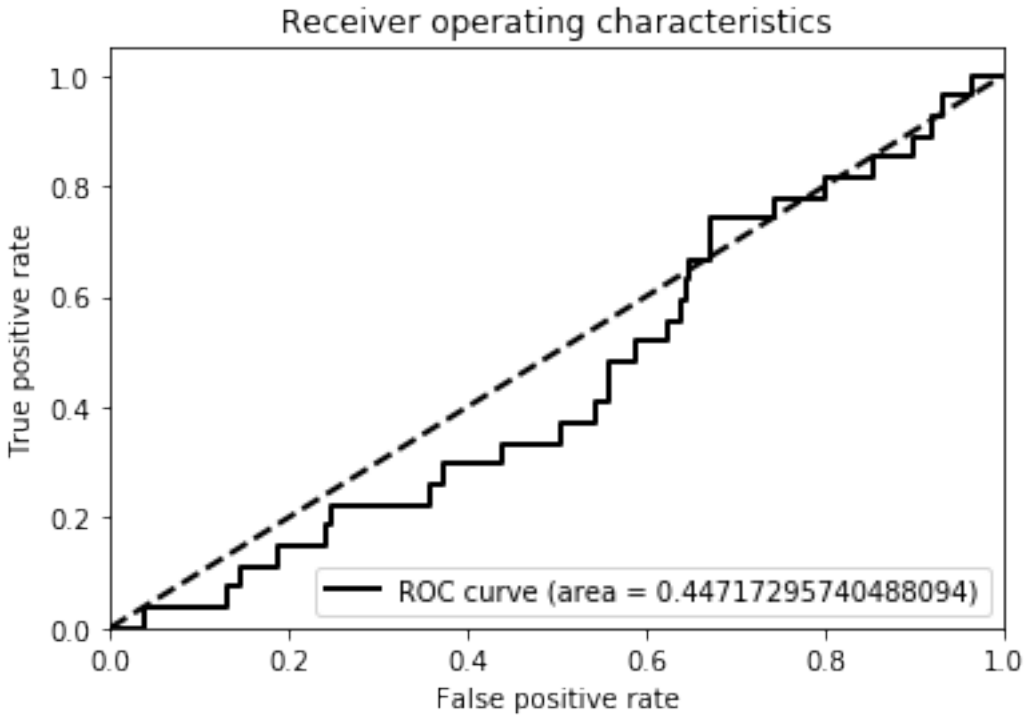


- To create the datasets for cross-validation, I shuffled positive datasets randomly and make into five sub-datasets. Below are the ROC values for the five cross-validation datasets. The ROC values are very poor. One thing I could change is having a more robust regularization parameter and find an optimal value for it.




Receiver operating characteristics
True positive rate
False positive rate
ROC curve (area = 0.5281456526827799)
0.0
0.2
0.4
0.6
0.8
1.0


Receiver operating characteristics
True positive rate
False positive rate
ROC curve (area = 0.44717295740488094)
0.0
0.2
0.4
0.6
0.8
1.0