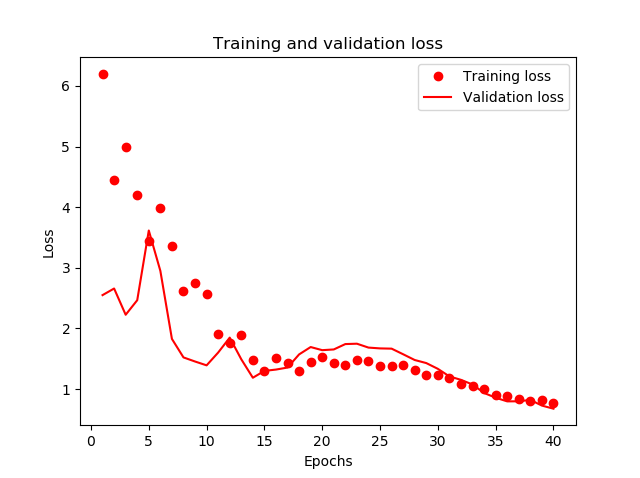
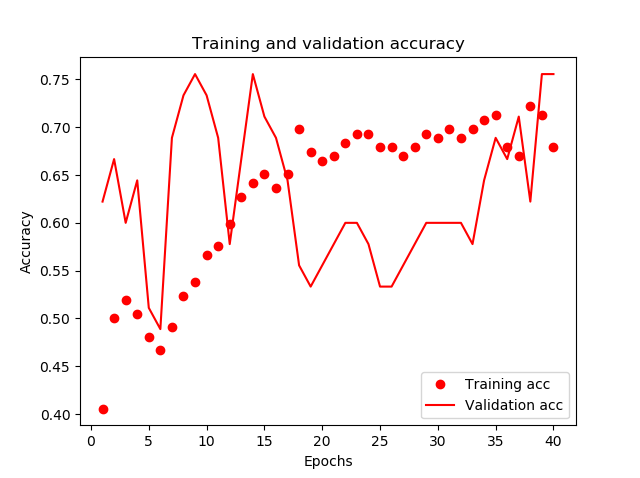
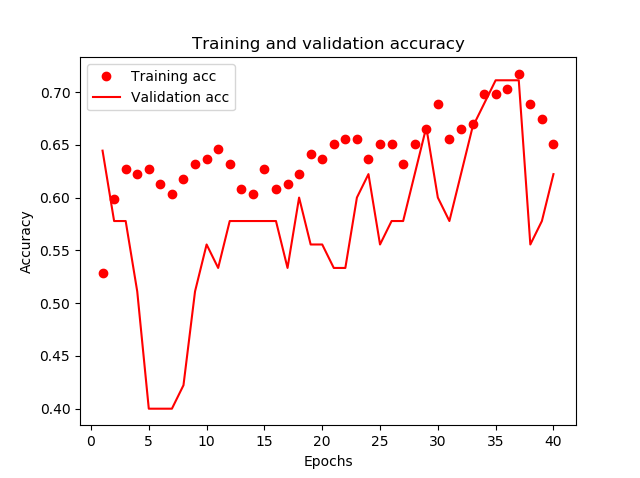
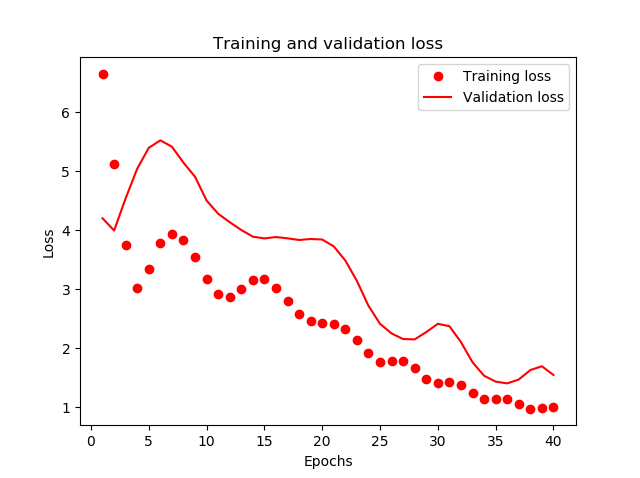
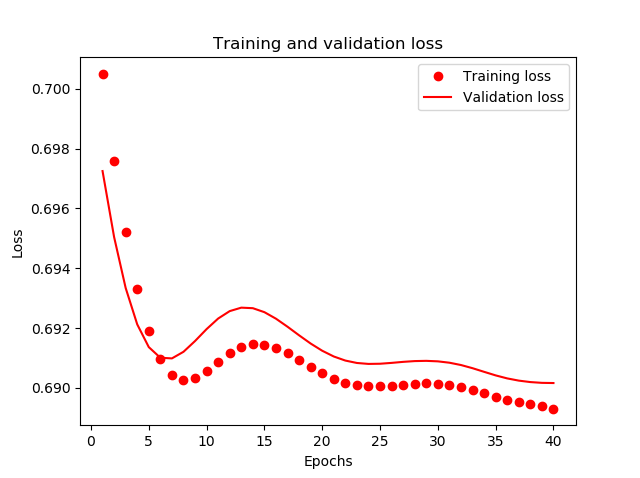
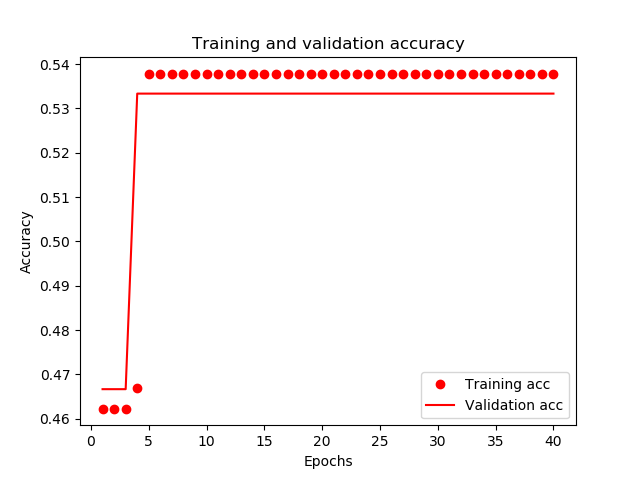
For this project, I decided to use Keras(withthe permission of the instructor) which is capable of running on top of TensorFlow. The full documentation on how to use Keras can be found here(<https://keras.io/>). I am using 2010 laptop that runs on Windows 7 and installed anaconda that had TensorFlow installed in it with Keras. In this project I used two links: one is talking about how to evaluate the model(<https://machinelearningmastery.com/evaluate-performance-deep-learning-models-keras/>)

 At first, I separated the data into groups of 70% for training set, 15% as a test set, and 15% as a validation set. Then I created a model that has in input layer with 13 columns(14th is for output). I also normalize the data using the batch normalization at this layer. Then I add two hidden layers, and my output that indicates whether or not the patient has a heart disease(has 1 neuron). I tried to use softmax on the output layer, but it did not work and gave me an incorrect accuracy of more than 8, so I used sigmoid which seems to give a correct result. For my model I used AdamOptimizer and for loss I used binary crossentropy. The first result I got was from linear layers with loss of 0.90 and accuracy of 0.65. The following are the graphs of training and validation loss. The second graph is accuracy. The format will be followed in the next diagrams as well.

The next model I tried included rectified liner unit layers. This resulted in 0.98 loss and 0.717 accuracy with the following graphs.

And the last model I used included layers with sigmoid functions.

The best I achieved was 0.68 loss and 0.59 accuracy with the following graphs.

I tried different combination of neurons and layers but was getting close to the above results. In fact, those were the best results I have gotten. In the future, I would like to change my computer. I believe my CPU and GPU do not work very well, and since Keras is using both of them, I believe that my fail in accuracy and loss as well as using softmax on the last layer is probably because of them. I also would like to try different configurations of the training strategy, as well as my network. To run the script simply navigate to the folder and run: python proj4.py