**Machine Learning Task Report**

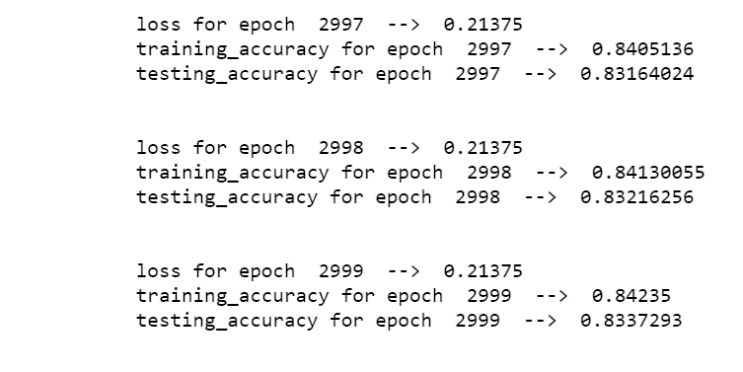
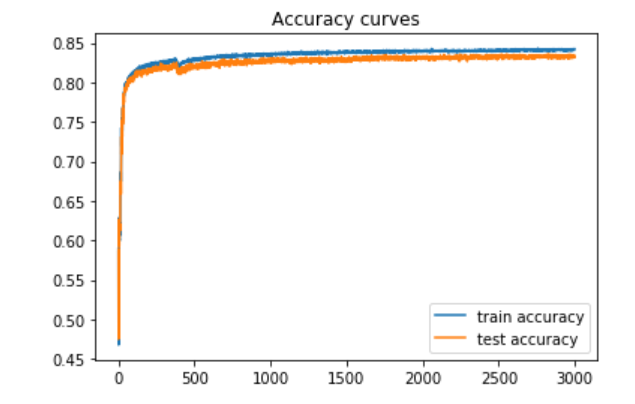
I did this task by the following two approaches:

1. Using Support Vector Machines

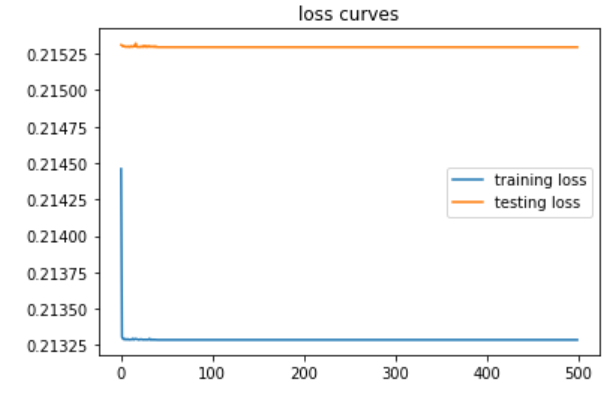
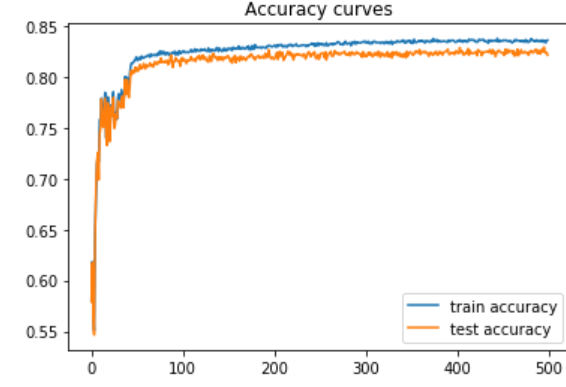
* I used this approach as we all know that SVM is known for working best when there are ample of features and less number of examples.
* This model worked pretty well as it is giving accuracy about 98-99% accuracy and I don’t think that its overfitting as if it would have been the case then I would have got a very bad test accuracy but I am getting very good accuracy around 98-99% on test data.
* I have scaled the data to normalize it.
* I have taken 1017 instances of both classes as instances of non-musk are very much greater than instances of musk.
* You can also check predictions by pasting values in the live predictor in list form.
* I have split the data in 80:20 proportion as said.
* I have saved the model as ‘svm\_model.h5’.

2. Using neural networks (tensorflow)

* I have made a neural network having layer structure [200-100-2] using tanh, tanh, sigmoid activations respectively.
* I have scaled the data to normalize it.
* This model worked pretty well as it is giving accuracy of 84-85 percent.
* I have split the data in 80:20 proportion as said.
* I have used both RMSPropOptimizer and AdamOptimizer and both are giving really descent accuracies and also I have saved both the models.
* I have also used all other optimizers also and they worked really bad and I have mentioned them in the notebook.
* I have declared the batching function to generate batches.
* I have trained it on 500 epochs.
* I also trained it for 3000 epochs but it was giving the almost same accuracy as of 500 epochs.



* Also I have plotted the graphs for both the accuracy and losses for both training and testing.



* Also I tried with many other hyper parameters but these one worked best for me.