**Python/Deep Learning\_Lab2 Report**

**Team Members (10)**

1. Chaitanya Mallepudi (14) – 34%
2. Sai Rohith Guntupally (8) – 33%
3. Madhu Varma Rudraraju (20) – 33%

***Introduction***

This lab focuses on Deep Learning concepts. In this, we had implemented image classification with CNN, text classification with CNN and LSTM, Logistic Regression and Linear Regression.

***Objective****:*

To implement below tasks:

1. Linear Regression
2. Logistic Regression
3. Image Classification
4. Text Classification with CNN
5. Text Classification with LSTM
6. Compare 4 and 5 results
7. Image Classification with Autoencoding

***Requirements****:*

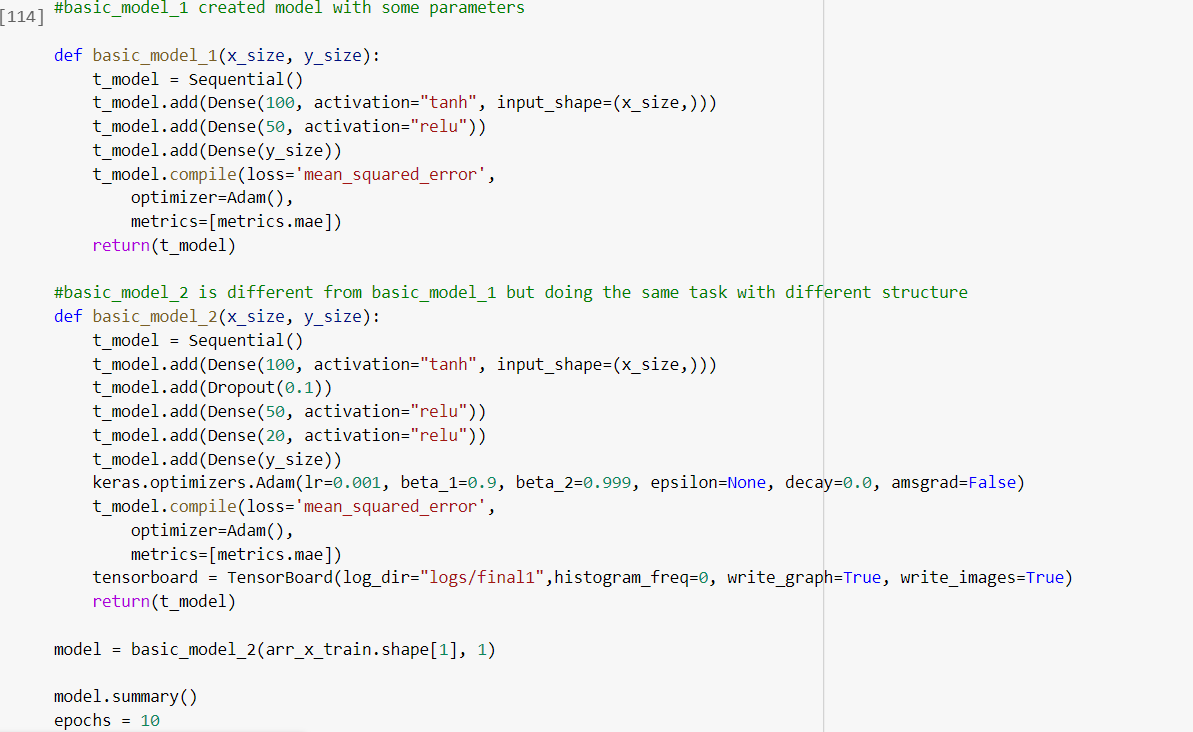
1. Google Colab
2. Python 3.7
3. Tensor Flow
4. Keras
5. TensorBoard

***WorkFlow****:*

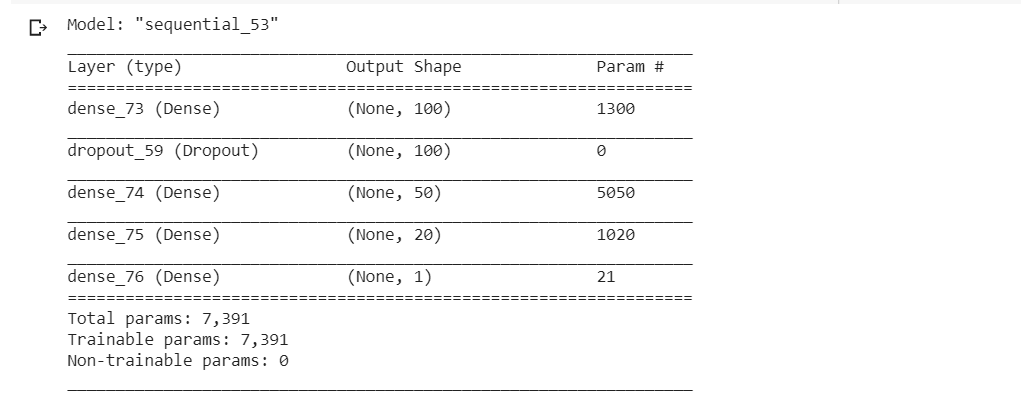
**Task 1:-**

Building Sequential model using keras implement Linear Regression & plotting a Tensor-Board & plotting loss and tuning hyperparameters lets us note how accuracy and mean absolute error changes for each case.

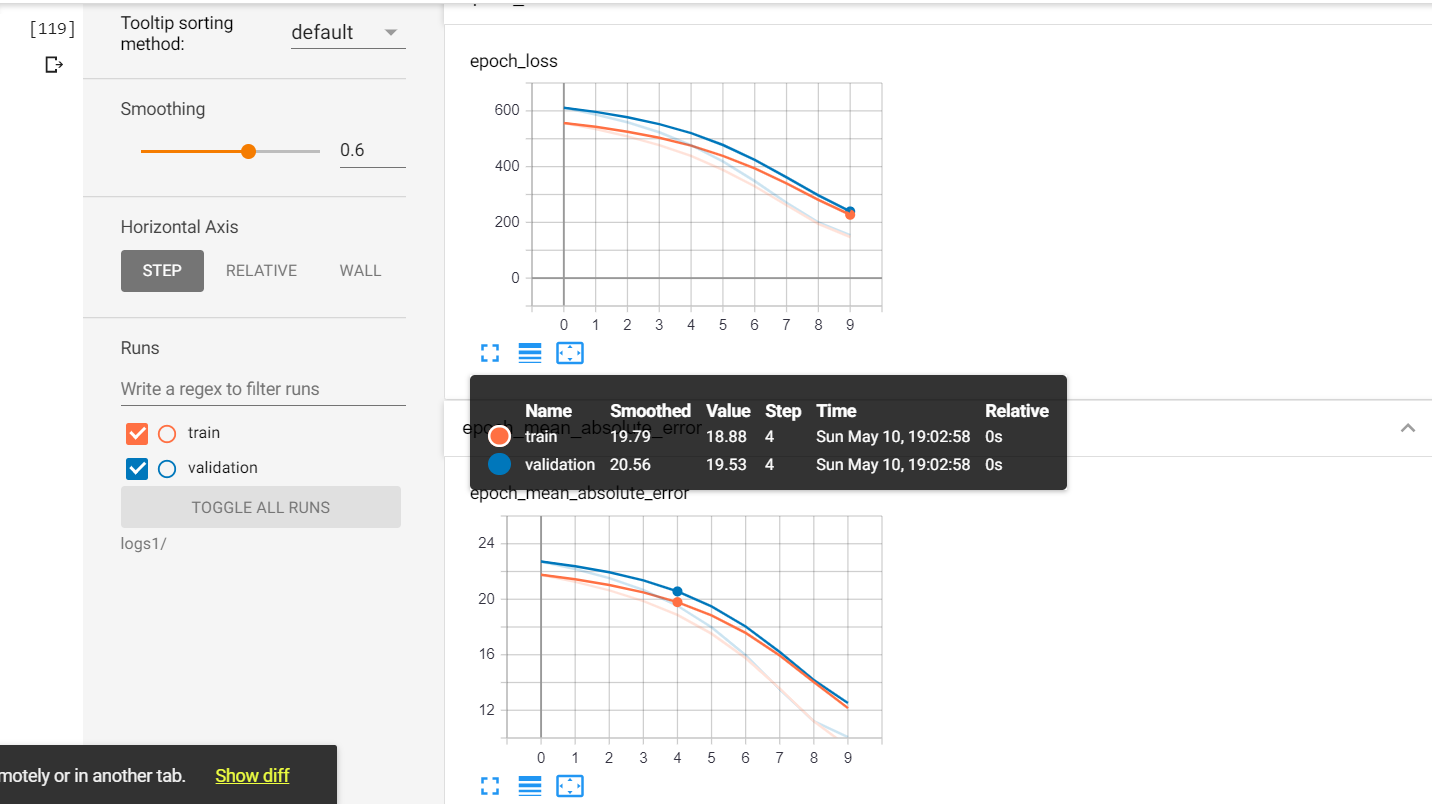
**Code & Output :--**



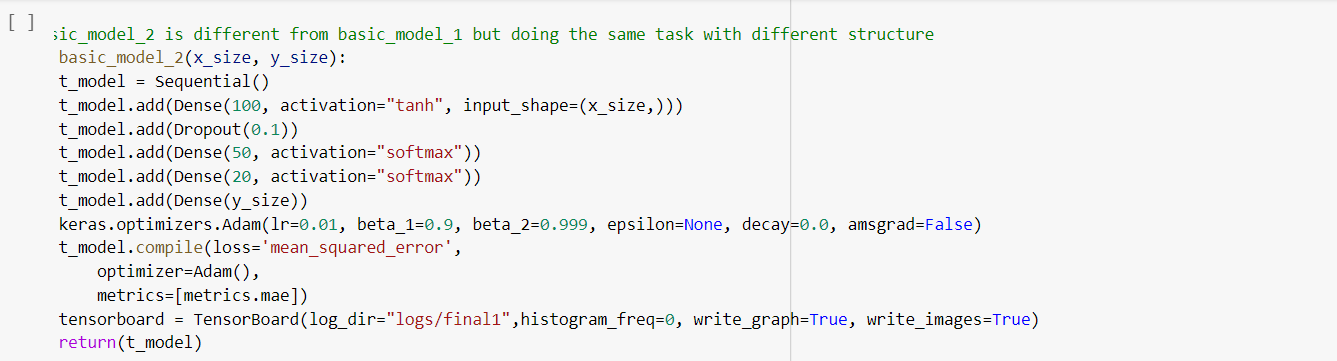


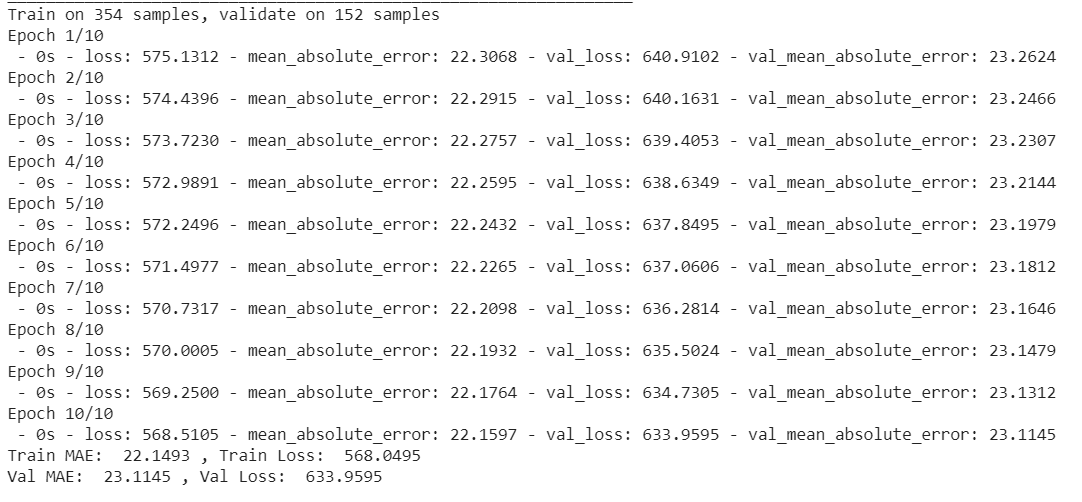


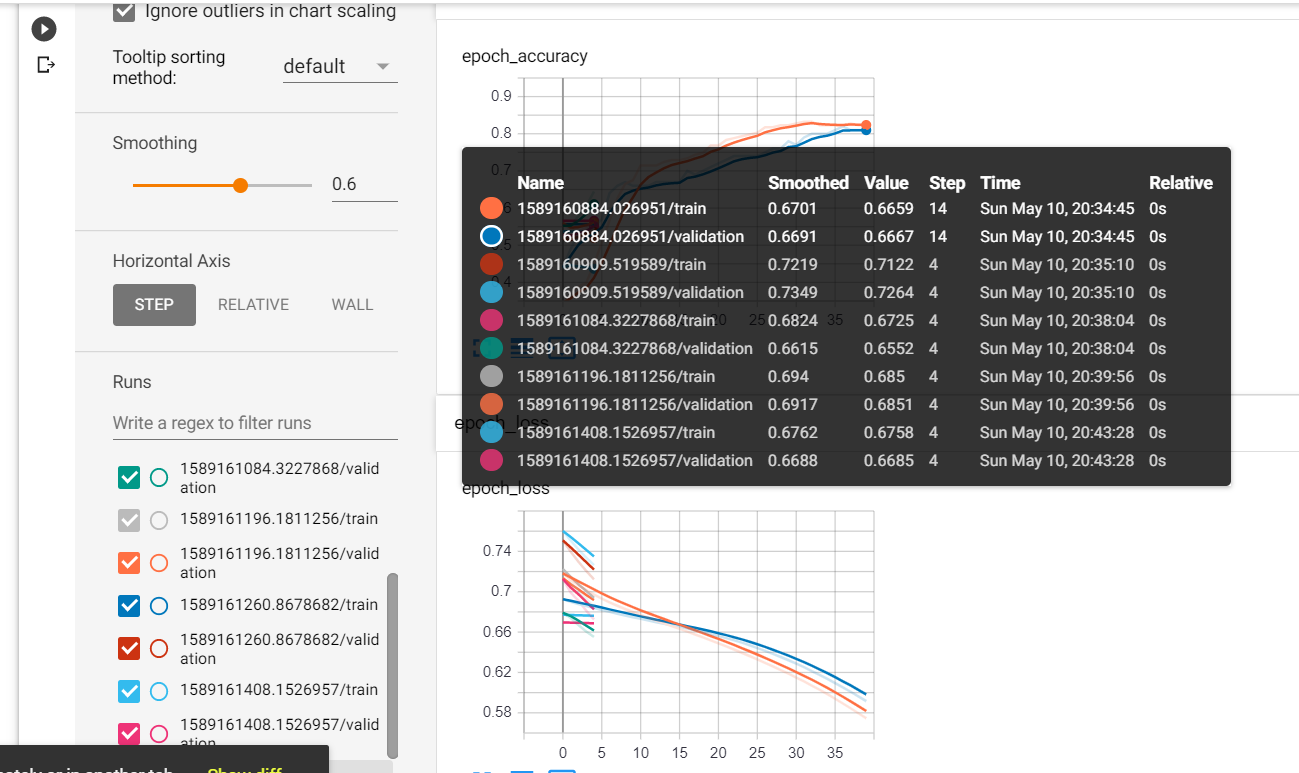




Changing Learning rate to 0.01

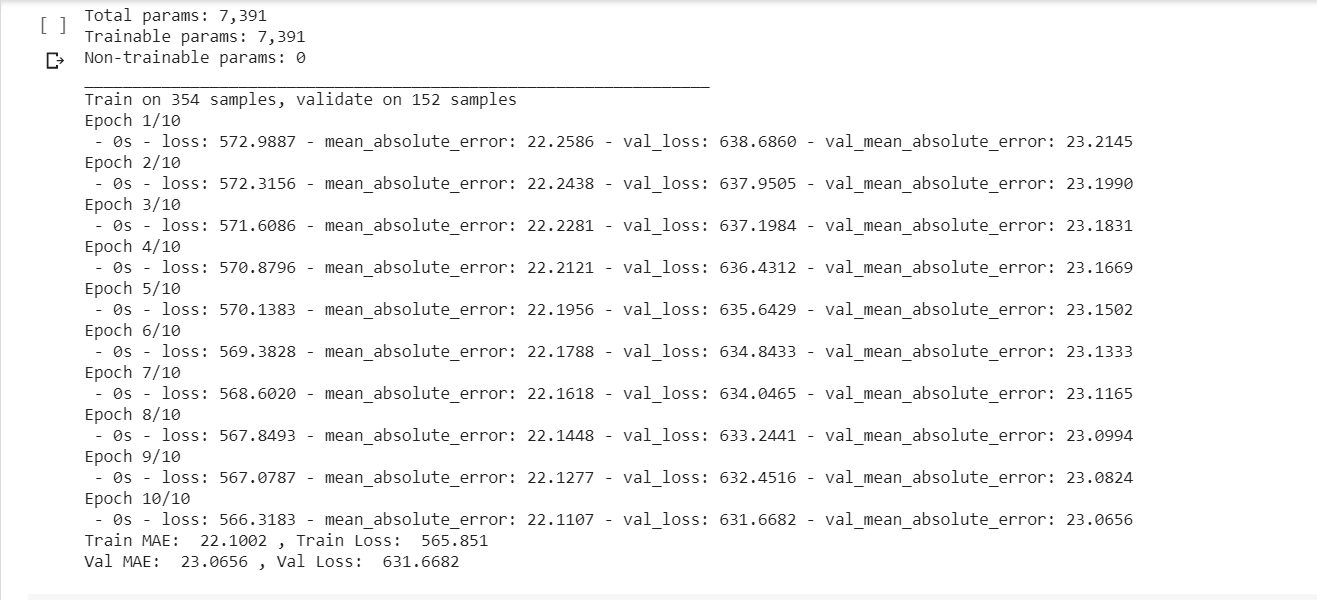


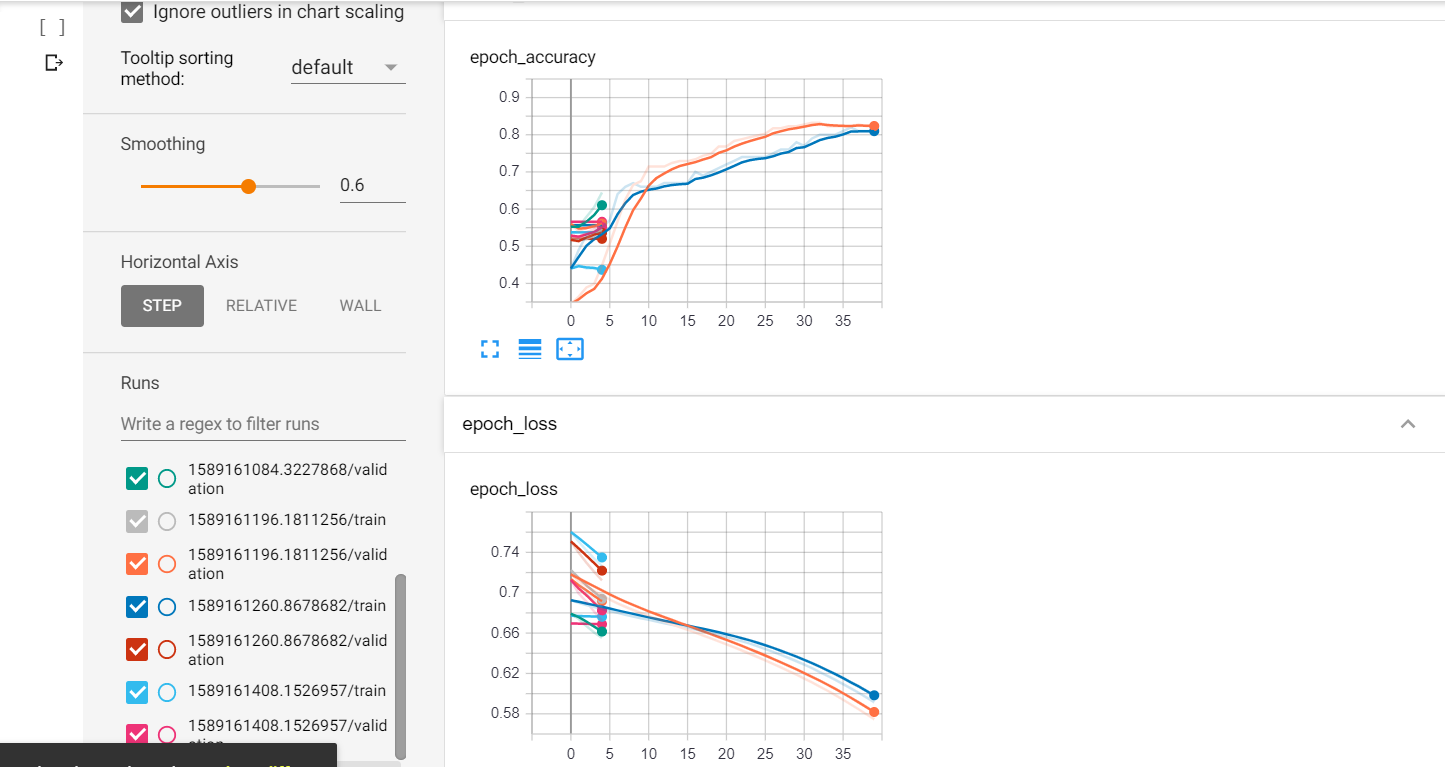




Changing activation layersoftmax:



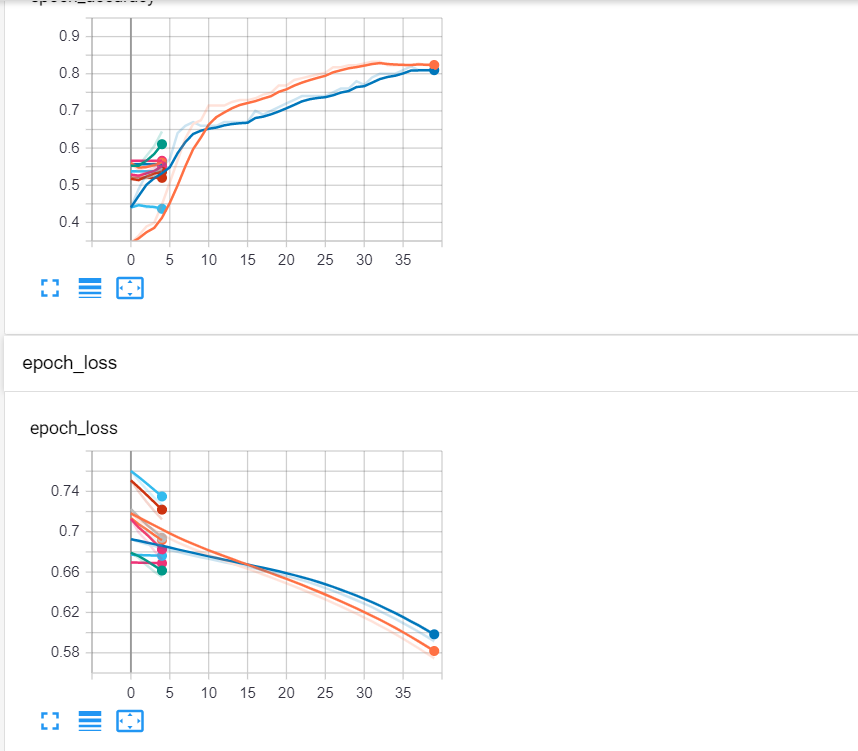




Changing Batch Size to 128:





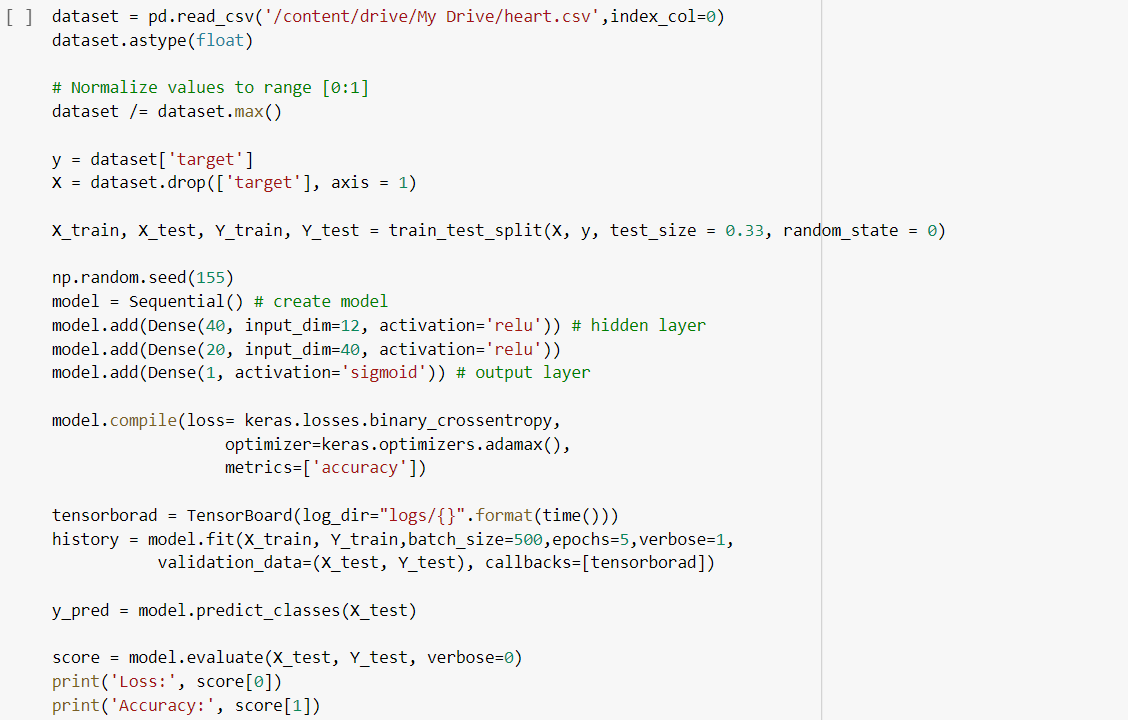


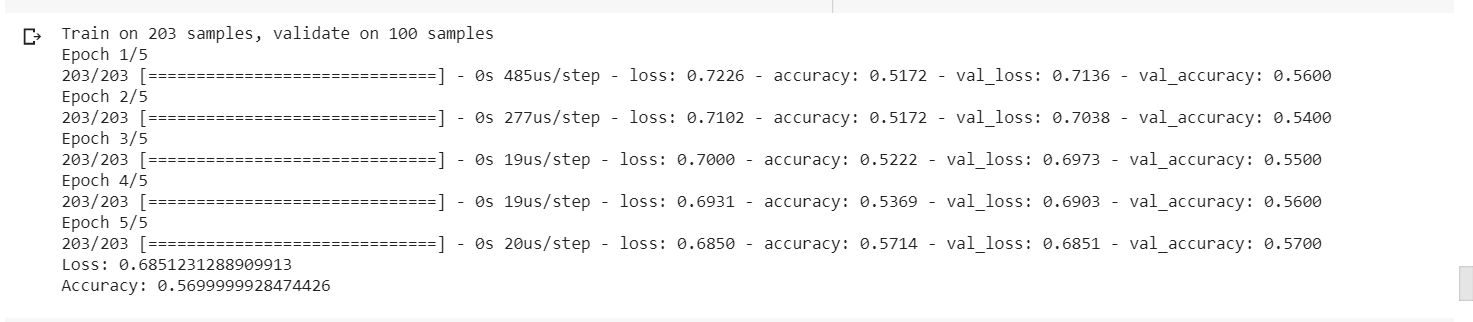
Hence, we can clearly say that by tuning each hyper parameter the loss and error varies. So, the accuracy also varies.

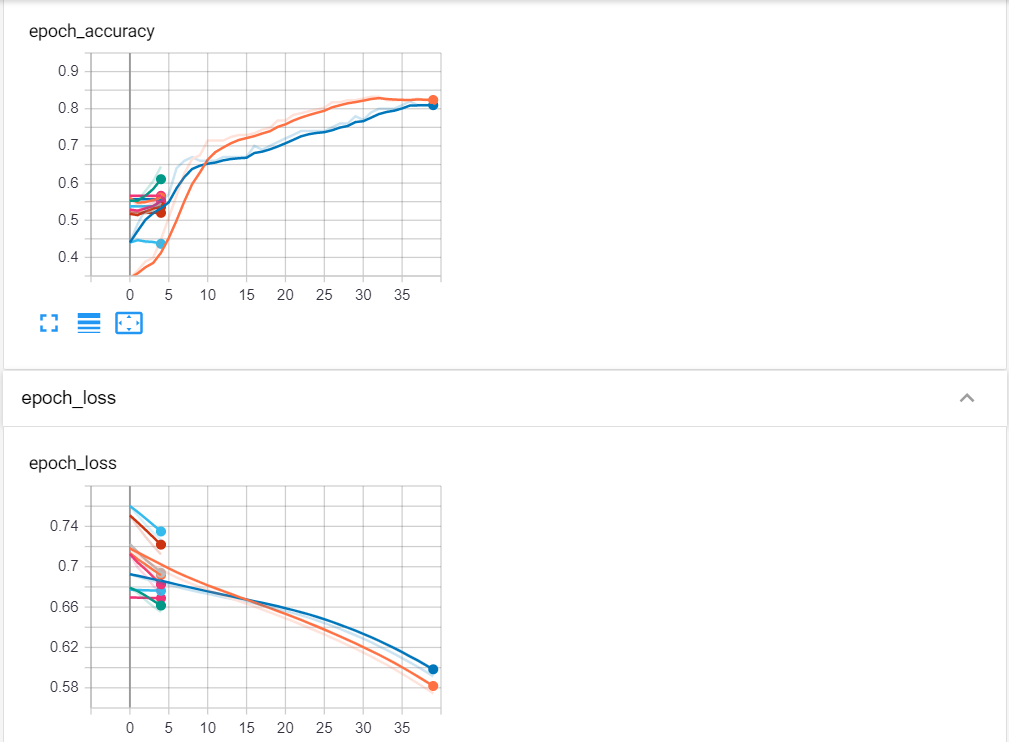
**Task 2:-**

Implementing Logistic Regression and by tuning the hyper parameters in each case lets see how the loss and accuracy changes in each case.

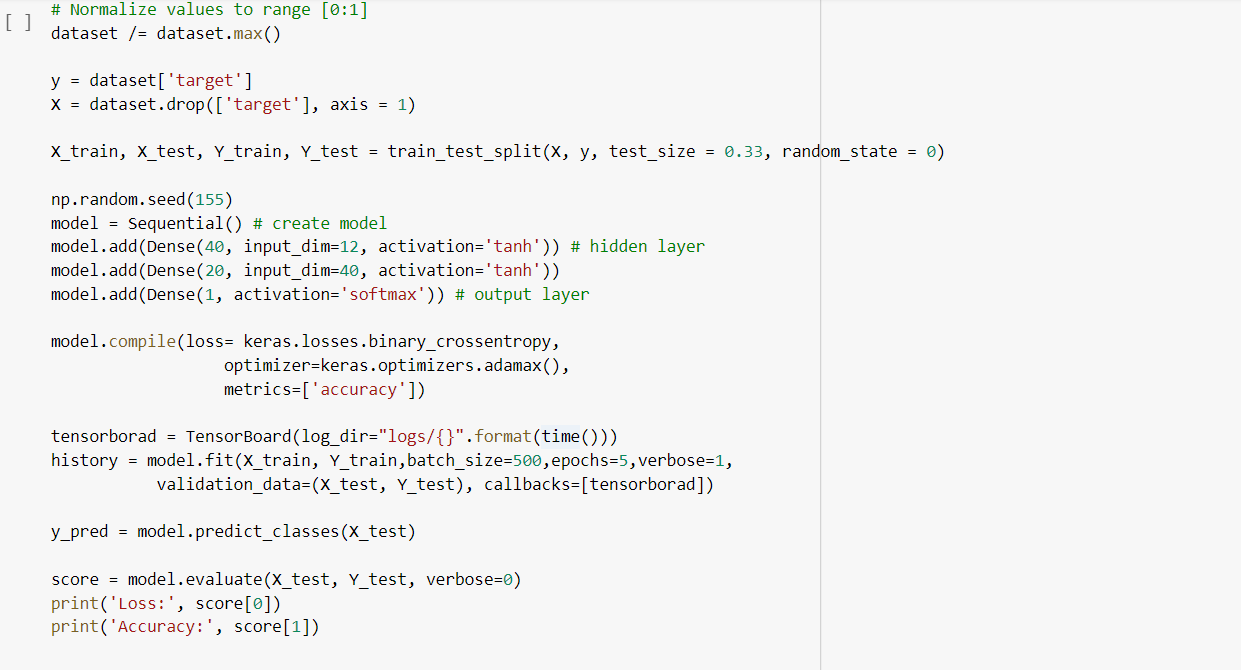
**Code & Output :--**

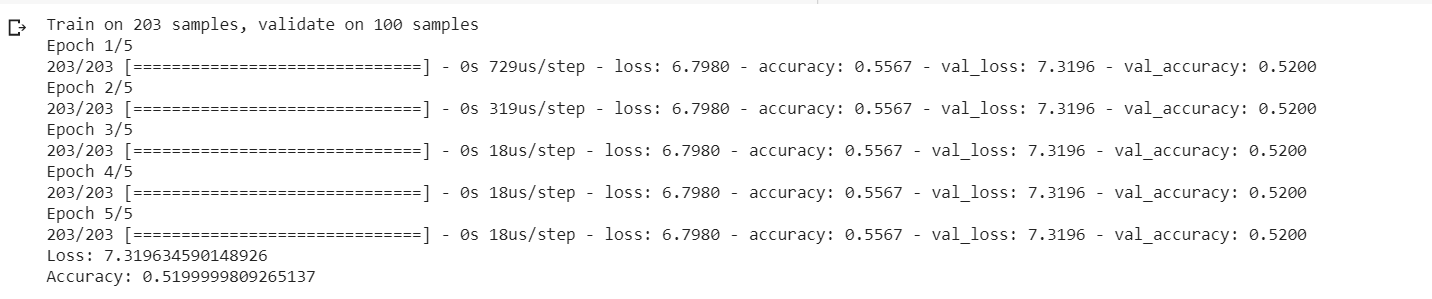


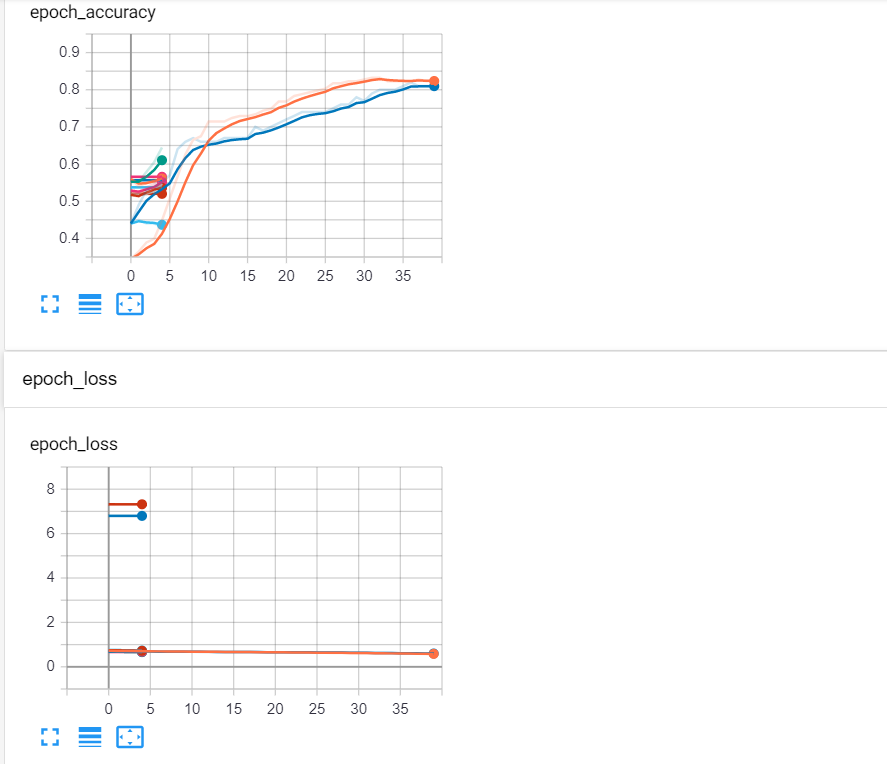




By Changing the Activation function to Tanh:

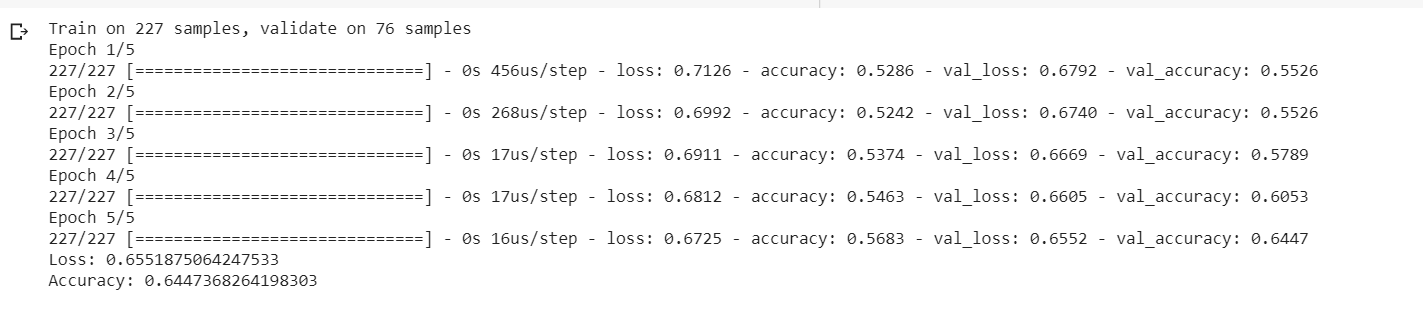


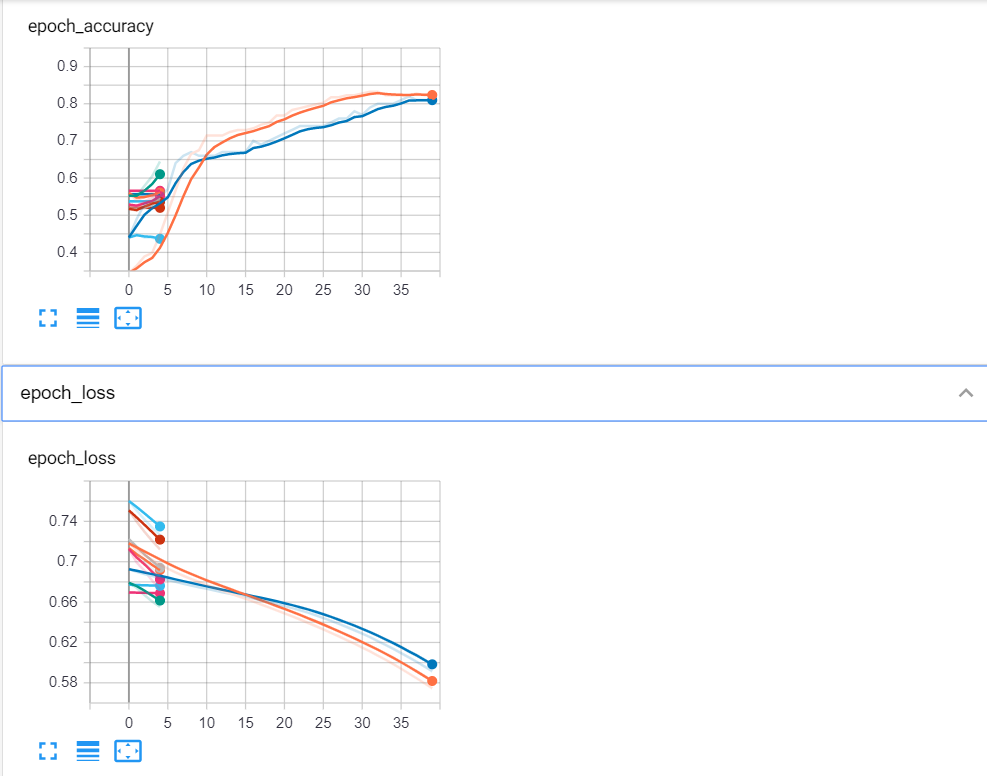




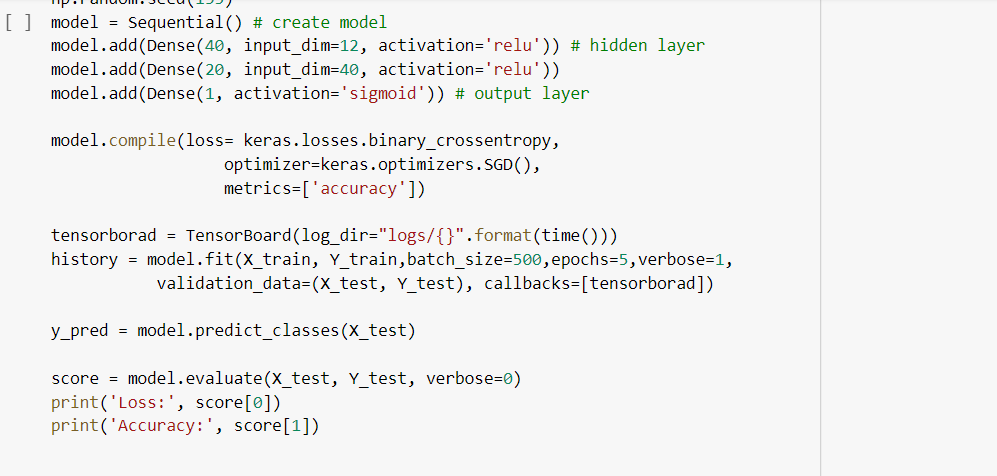
By Changing the test size to 0.25:

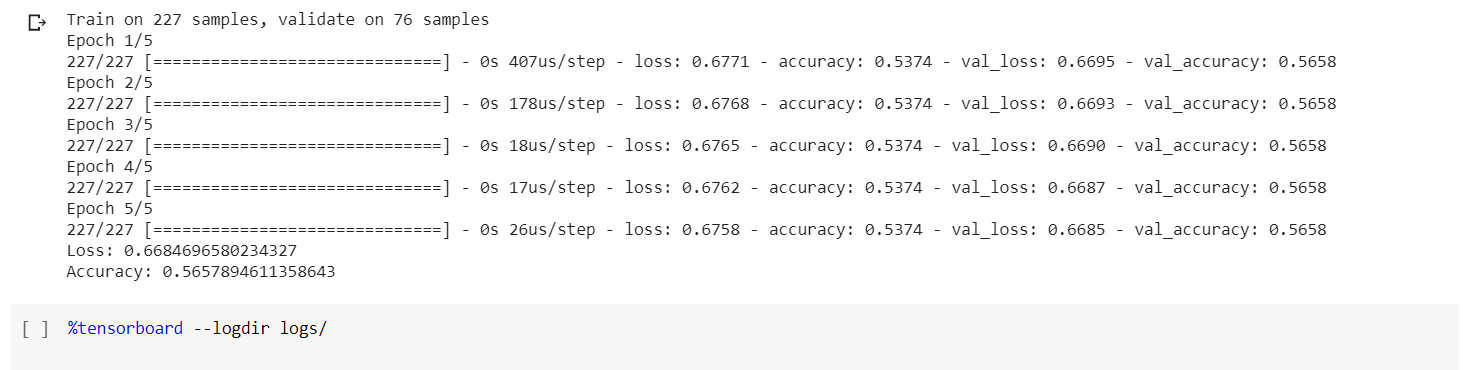


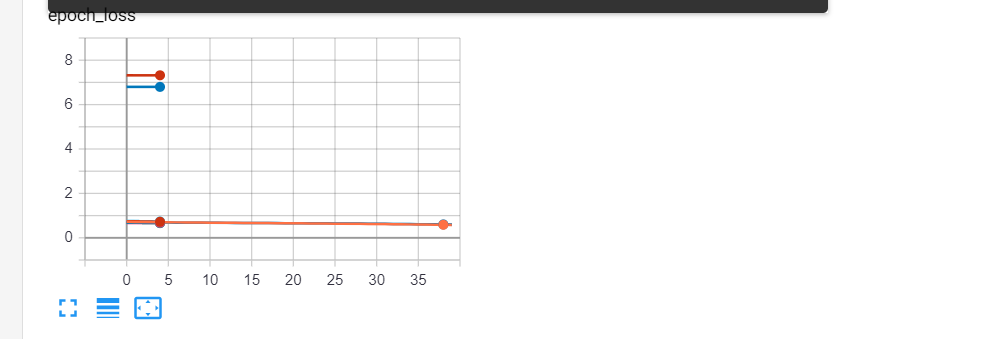




Changing the optimizer to SGD:







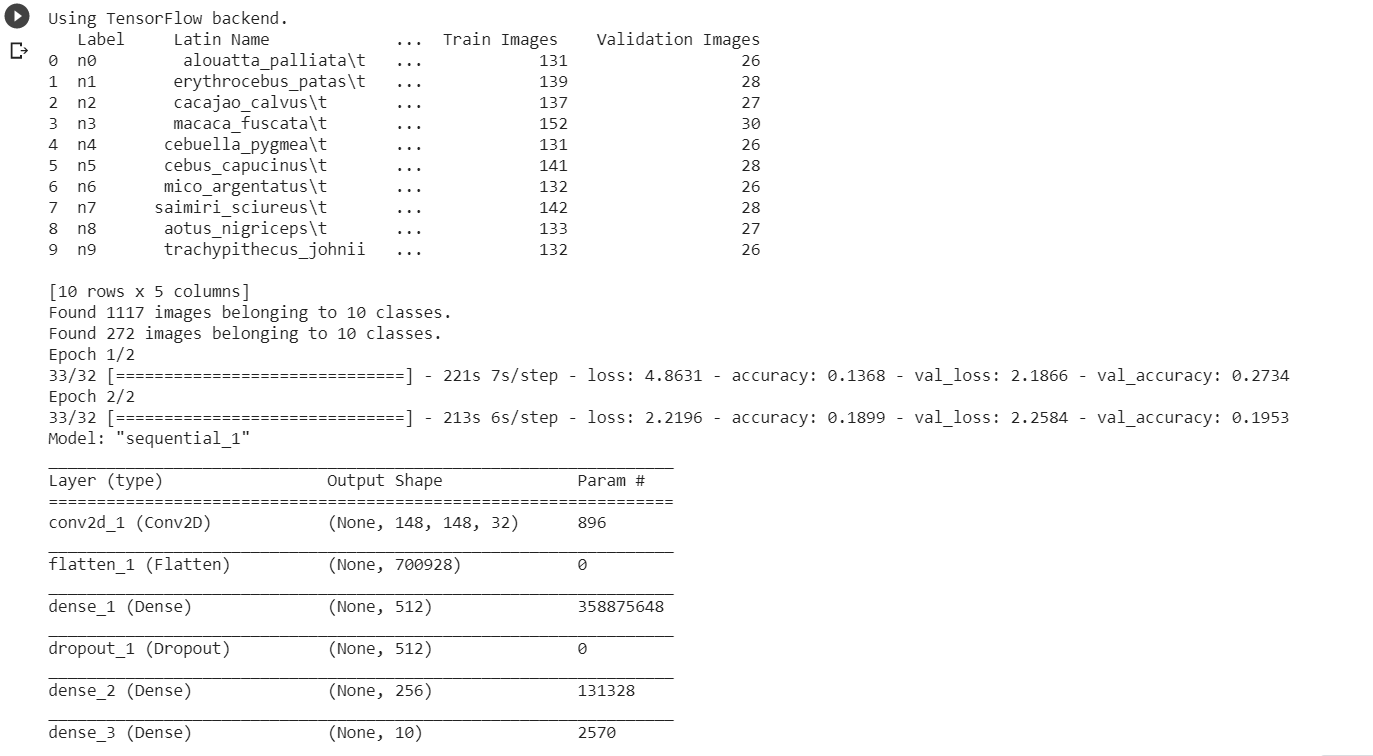
**Task3:-**

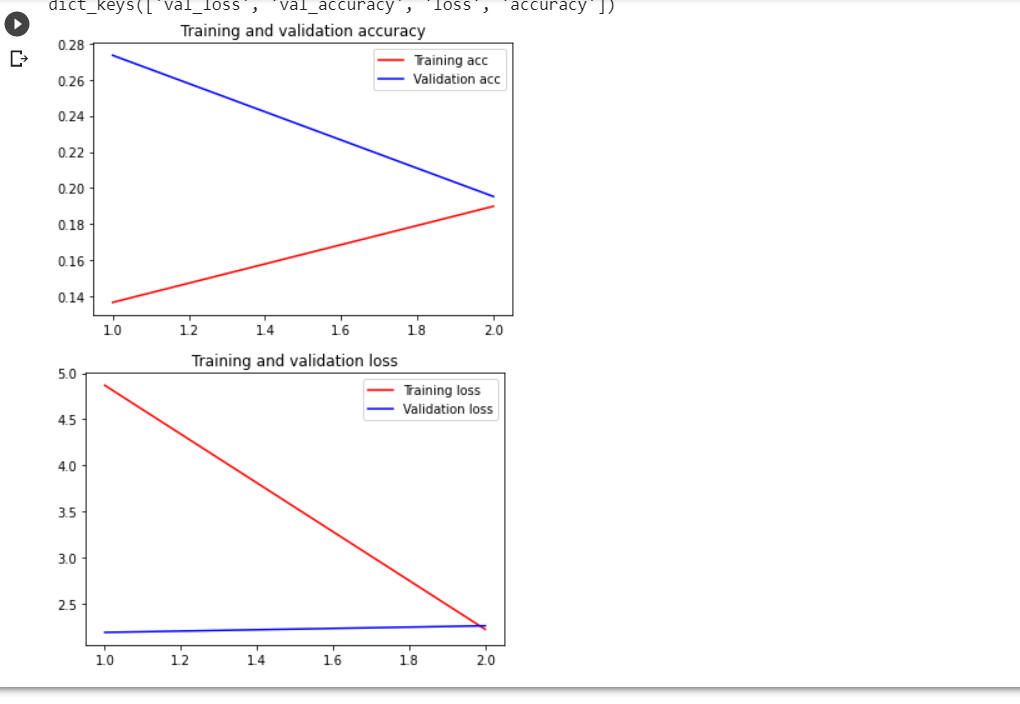
Implementing Image Classification with CNN Model with Monkey Variety Data set from Kaggle.

We had classified the different variety of models by building a conv\_2d model.

**Code & Output:-**







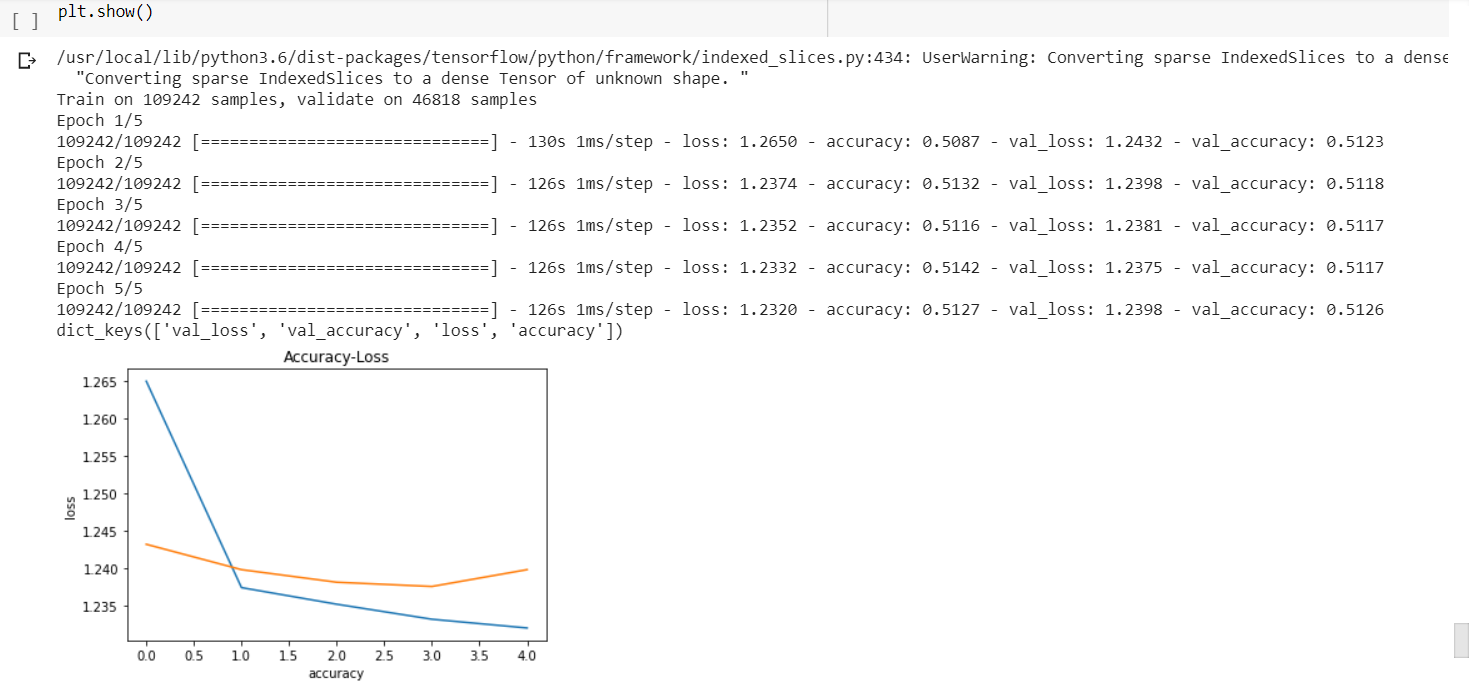
**Task4:--**

Implementing\_text\_Classification\_with\_CNN\_Model of movie review dataset.

**Code & Output:-**







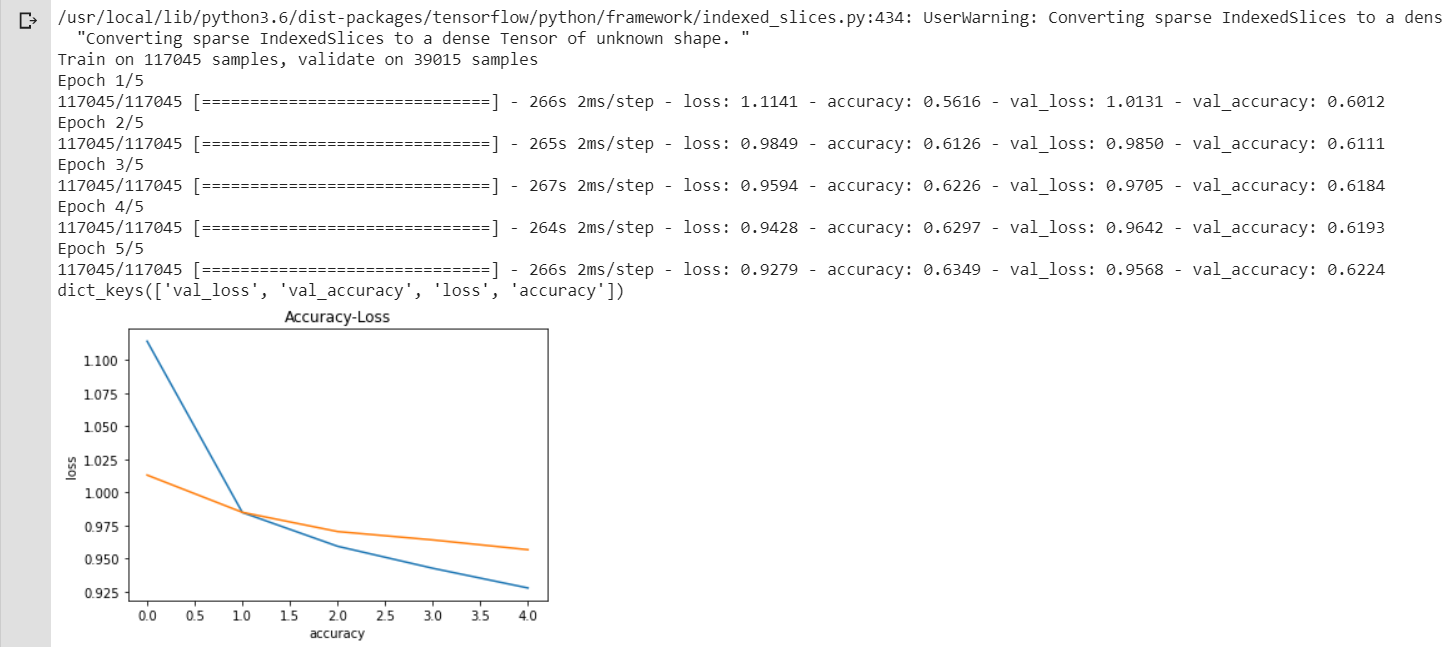
**Task5:-**

Implementing text Classification with LSTM Model and performing sentimental Analysis:

**Code & Output:-**







**Task6:--**

Comparing\_the\_results\_of\_CNN \_ and\_LSTM\_Model\_for\_text classification.

Tune\_Parameters\_to\_Attain\_Best\_Accuracy.

From the above 2 results we can see that the accuracy for LSTM model is higher compared to CNN model.

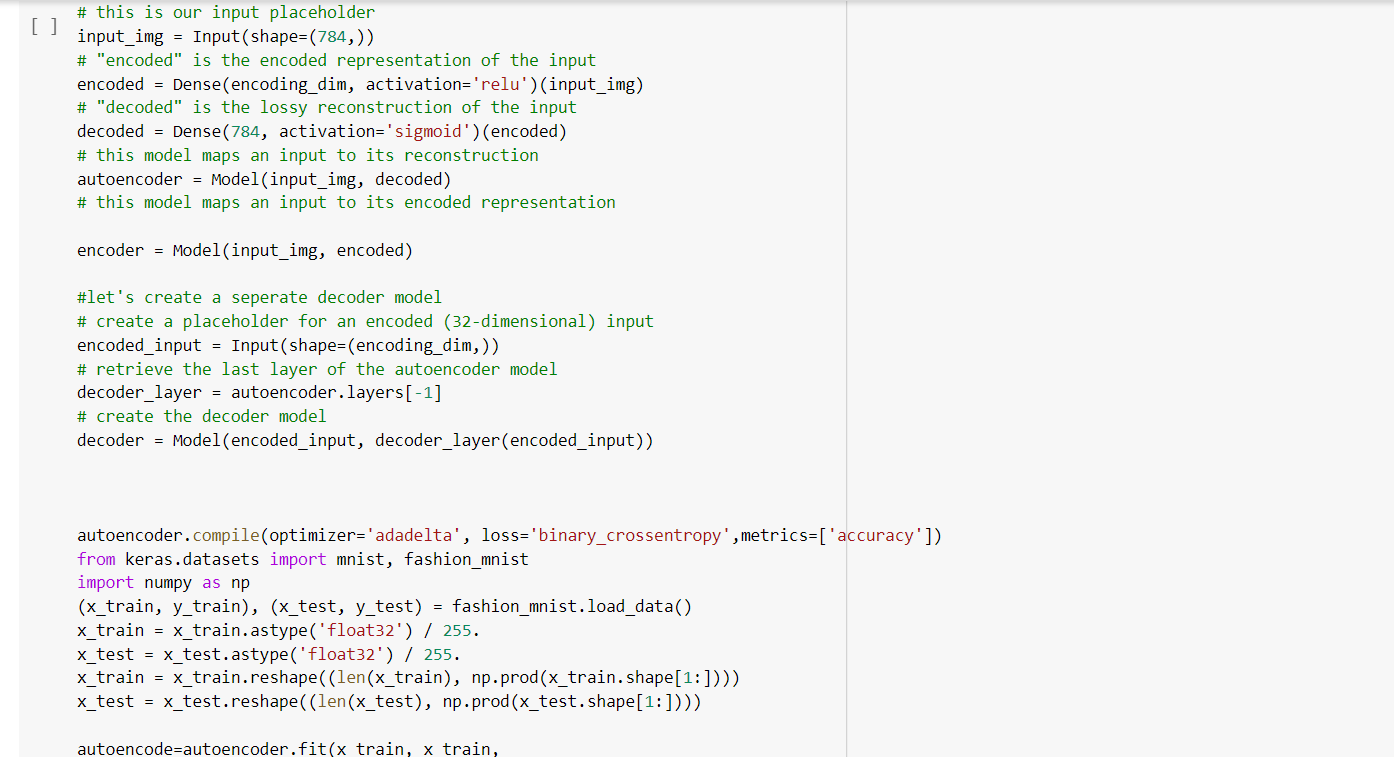
We can see that accuracy for Conv model as 0.5 and accuracy for LSTM is 0.9 for each continuous text data feed. So, LSTM model is better for text classification compared to CNN.

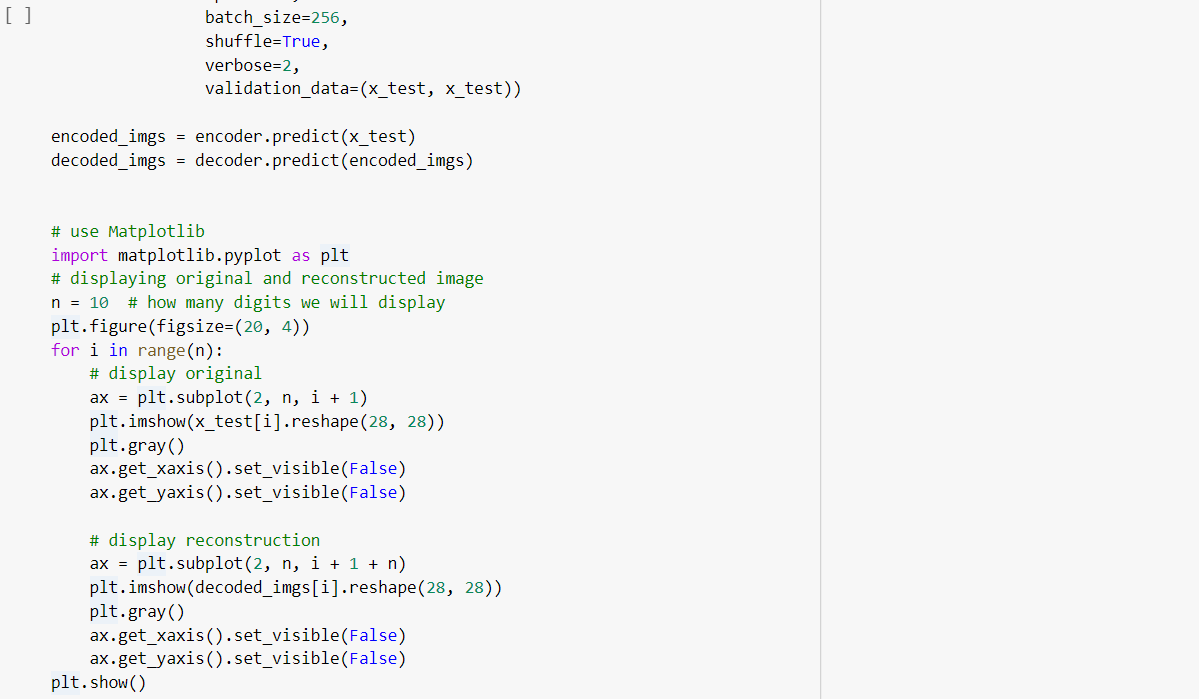
Also, the loss for Conv model is 1.23 and loss for LSTM model is 0.51 .

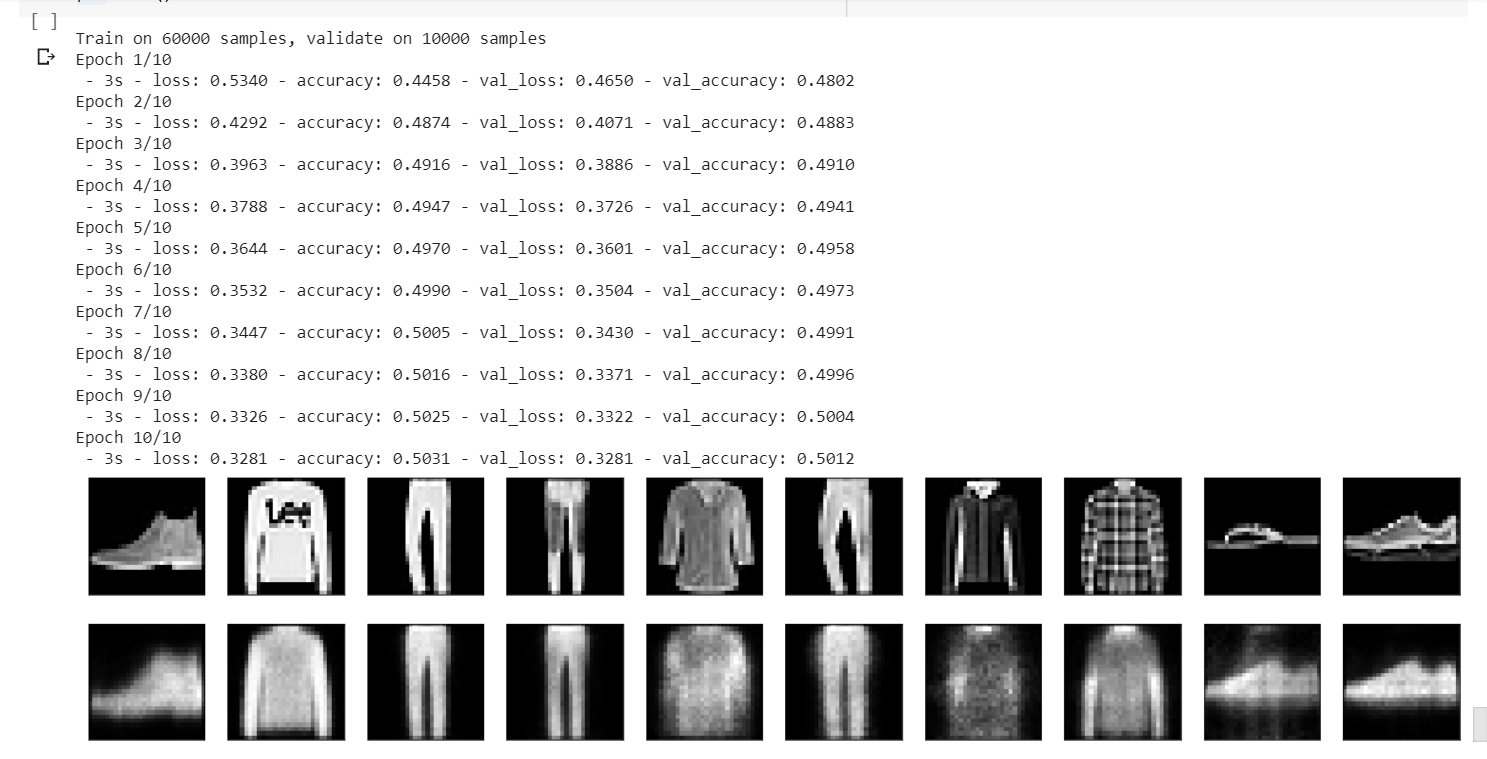
**Task 7:--**

Applying\_Encoding\_And\_Showing\_Encoding\_&\_Decoding\_on Particular\_Image with mnist dataset in keras.

**Code & Output :-**







***Conclusion****:*

We have understood and implemented the above-mentioned concepts. Text Classification is implemented using CNN and LSTM models and the accuracies are compared.

The loss and accuracy graphs are plotted .Linear Regression, Logistic Regression models are evaluated. Text Classification is implemented using CNN and LSTM models and the accuracies are compared. Image classification is implemented using CNN.

References:

<https://www.kaggle.com/>

<https://stackoverflow.com/>

<https://www.tensorflow.org/api_docs/python/tf/keras/Model>