**Programming Test: Learning Activations in NeuralNetworks**

- I have used iris dataset for the the test.

- downloaded and load iris dataset in jupyter notebbok

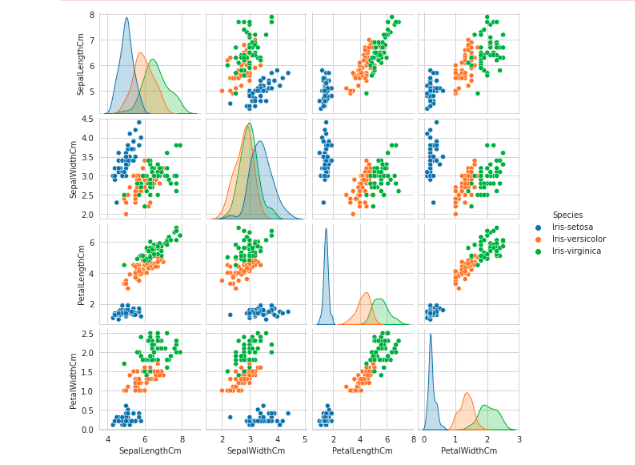
- In iris we have 4 independent features and 1 dependent feature.

- target feature has 3 classes (Iris-setosa, Iris-versicolor, Iris-virginica)

- did some check for null values and checked statastic information with .describe function

- checked wether dataset is balanced or not by doing groupby operation on target feature

- plotted graph of scatter plot to check lenearity of features



- After this worked on convert the species into each respective category to be feed into the neural network also did shuffling so that for test dataset won’t be of same category.

- Now created X and y i.e input and output feature data

- Then for better accuracy and and fast processing I am normalizing input data

- Train and test data splitted in 80-20% ratio

- changed the output feature to one hot vector

- now craeted neaural network model with activation function ReLU in hidden layer and softmax for output layer

Model has **3 hidden layers** with **activation function ReLU**  
- because sigmoid has vnishing gradient problem and data is not that much complex that we will have dying ReLU problem.

I also used **kernel\_intializer as ‘he\_uniform’** for initializing weights for better accuracy.

- for **output layer ‘softmax’ activation** function is used as it’s multi class classification.

- used ‘**categorical crossentropy’ as loss function** and ‘**adam’ optimizer**.

- Then model trained with **batch size of 15 and 20 epochs**

**Results**

- Train acuuracy – 100%

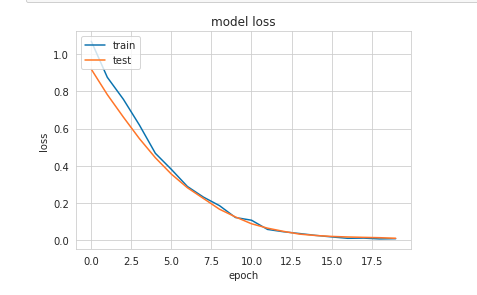
- Test Accuracy -- 100%

- Train loss -- 0.003978

- Test loss -- 0.009745

- F1 score – 1.0

**Loss function vs. Epochs graph**



Code base github link : - https://github.com/nikhil93-ds/Assignment