

Technical Report on Programming Test: Learning

Activations in Neural Networks

Prepared by: Sumantha.NTS
Bengaluru, Karnataka

Background:

Selection of the best performing Activation Function (AF) for classification task is essentially a naive (or brute-force) procedure wherein, a popularly used AF is picked and used in the network for approximating the optimal function. If this function fails, the process is repeated with a different AF, till the network learns to approximate the ideal function. It is interesting to inquire and inspect whether there exists a possibility of building a framework which uses the inherent clues and insights from data and bring about the most suitable AF. The possibilities of such an approach could not only save significant time and effort for tuning the model, but will also open up new ways for discovering essential features of not-so-popular AFs.

Neural Network Modelling:

With the understanding of background, neural network has been built with the help of tensorflow library with 3 layers i.e., one input layer, one hidden layer and one output layer. The weights for the neural network are randomly initialized with the help of **uniform** kernel initializer. The weights will get updated in back propagation in each epoch. To generalize the model, **Categorical crossentropy** is considered to be the loss function as it is a classification problem statement and **Adam** optimizer is considered to update the weights in back propagation. **Accuracy** is considered to be the evaluation metric. Number of neurons in layers, number of epochs and batch size is considered to be user defined.

The main objective is to select the best activation function for the given dataset. 7 activation functions are considered for the same purpose. They are **Relu**, **LeakyRelu**, **Selu**, **Elu**, **Softplus**, **Exponential**, **Tanh**. Since the classification problem statement is considered for experiments, the activation function considered for output layer is Softmax which gives the probability distribution of the classes.

There are 4 datasets given for the above experiments and results are given in next section. Also the python file has been uploaded to Git-hub repository as given in the below link.

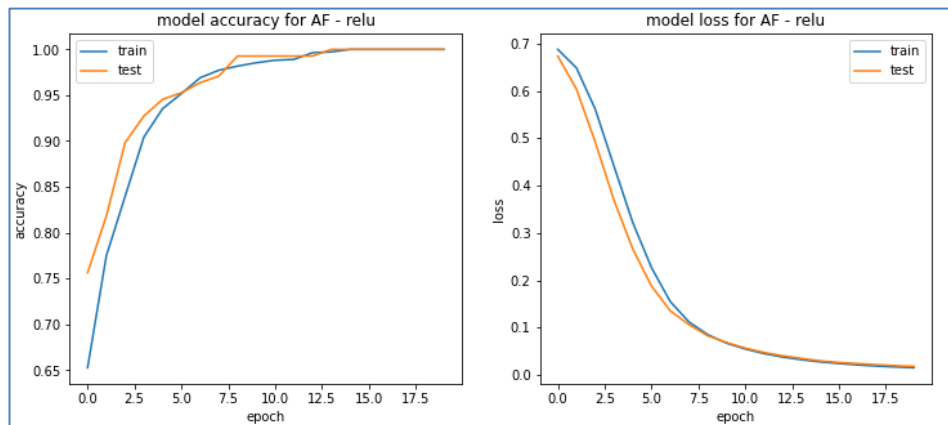
<https://github.com/sumantha-NTS/Learning-Activations-in-Neural-Networks>

Bank-Note Authentication dataset:

Bank note authentication dataset consists of 4 inputs and 1 output. Class is considered to be output variable. The user defined parameters are input layer neurons = 4, hidden layer neurons=8, epoch = 20 and batch size = 32.

The results for Bank note authentication dataset is mentioned below

- Best Activation function is **relu** with accuracy of **1.0**
- F1 score for model with relu activation function is **1.0**
- Graph showing the model accuracy and loss with respect to epochs are shown below

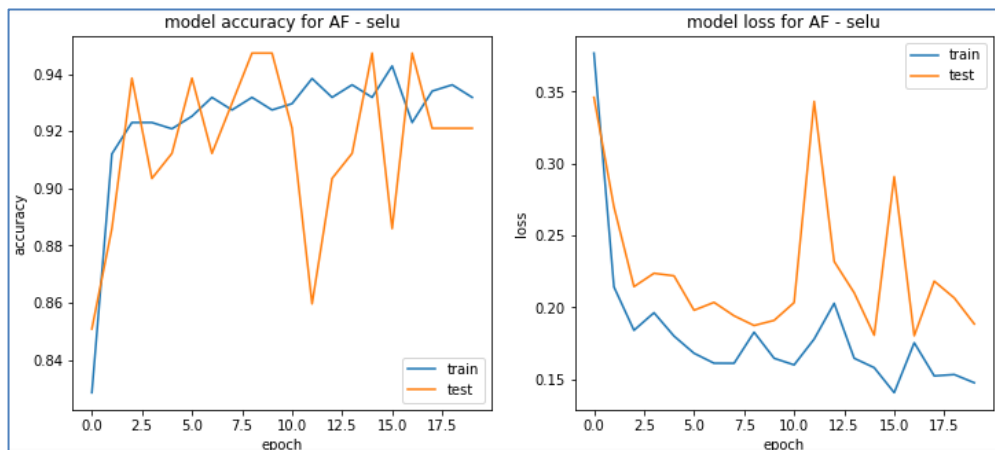


Wisconsin Breast Cancer dataset:

Wisconsin Breast Cancer dataset consists of 30 inputs and 1 output. Diagnosis is considered to be output variable. The user defined parameters are input layer neurons = 32, hidden layer neurons=64, epoch = 20 and batch size = 32.

The results for Bank note authentication dataset is mentioned below

- Best Activation function is **selu** with accuracy of **0.943**.
- F1 score for model with selu activation function is **0.93**.
- Graph showing the model accuracy and loss with respect to epochs are shown below

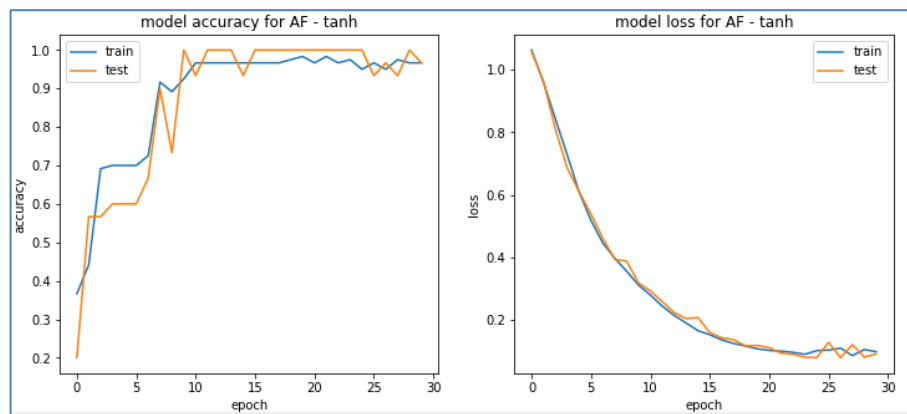


Iris dataset:

Iris dataset consists of 4 inputs and 1 output. Species is considered to be output variable. The user defined parameters are input layer neurons = 64, hidden layer neurons=300, epoch = 30 and batch size = 32.

The results for Bank note authentication dataset is mentioned below

- Best Activation function is **tanh** with accuracy of **0.983**.
- F1 score for model with tanh activation function is **1**.



MNIST Handwritten digit dataset:

MNIST Handwritten digit dataset consists of 60000 training records and 10000 testing records. 0 to 9 digit is considered to be output variable. The user defined parameters are input layer neurons = 64, hidden layer neurons=128, epoch = 20 and batch size = 32.

The results for Bank note authentication dataset is mentioned below

- Best Activation function is **tanh** with accuracy of **0.997**.
- F1 score for model with tanh activation function is **0.976**.
- Graph showing the model accuracy and loss with respect to epochs are shown below

