Name: Vijayalakshmi Somu

Artificial Neural Networks consists of hyperparameters like learning rate, hidden units, Activation functions and so on...

Hyperparameters are crucial as they control the overall behaviour of a machine learning model. The ultimate goal is to find an optimal combination of hyperparameters that minimizes the loss function to give better results. Hyperparameter tuning is the problem of choosing a set of optimal hyperparameters for a learning algorithm.

The model uses keras Tuner, a library to easily perform hyperparameter tuning with TensorFlow 2.0.

First, a tuner is defined, it's role is to determine which hyperparameter combinations should be tested. The library search function is executed in iteration loop, which evaluates a certain number of hyperparameter combinations.

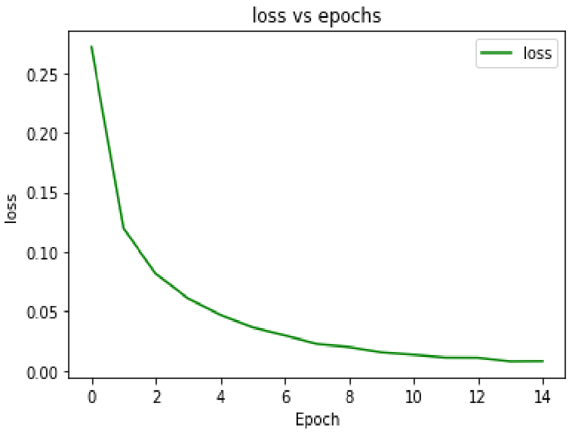
The model uses Hyperband tuning algorithm. It uses adaptive resource allocation and early-stopping to quickly converge on a high-performing model. The algorithm trains a large number of models for a few epochs and carry forward by choosing only the top-half performer models to the next round.

Evaluation is done by computing the trained model's accuracy on a held-out validation set. Finally, the best hyperparameter combination in terms of validation accuracy can be tested on a held-out test set.

The test and train accuracy and the activation function selected by the tuner is as follows.

Results for MNIST Dataset:

```
The optimal units in dense layer is 128, optimal learning rate is 0.001
and The best activation function is relu
```

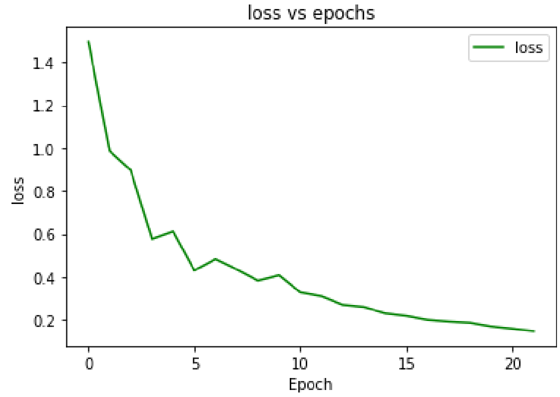



```
Train loss: [0.27808210253715515,
0.12185943871736526, 0.08262238651514053,
0.06037473306059837, 0.04699554294347763,
0.0360846109688282, 0.027921611443161964,
0.021892353892326355, 0.020137075334787737,
0.014695864170789719, 0.013476178981363773,
0.011438588611781597, 0.01041810680180788,
0.008612608537077904, 0.006973332725465298]

Test loss: 0.09290921688079834

Train accuracy: [0.9197962880134583,
0.964388906955719, 0.9756296277046204, 0.9814444184303284,
0.985370397567749, 0.988703727722168, 0.9917407631874084,
0.9938889145851135, 0.9938333630561829, 0.9954814910888672,
0.9959629774093628, 0.9964073896408081, 0.9967407584190369,
0.9974074363708496, 0.9980370402336121]

Test accuracy: 0.9790999889373779
```

F1Score: array([0.98672114, 0.99162627, 0.97831325, 0.97461424, 0.97822785,0.97493036, 0.98484056, 0.97425935, 0.97068063, 0.97496318])

Results for Iris Dataset:

The optimal units in dense layer is 640, optimal learning rate is 0.01 and the best activation function is relu



Train loss [1.496172547340393, 0.9865855574607849, 0.8965694904327393, 0.575973629951477, 0.6117395758628845, 0.4297371208667755, 0.4829946756362915, 0.4349101185798645, 0.3825111985206604, 0.4077788293361664, 0.3279189169406891, 0.31004536151885986, 0.26971110701560974, 0.259765625, 0.2298911213874817, 0.2182309627532959, 0.1996961385011673, 0.191730335354805, 0.18627120554447174, 0.1689901500940323, 0.15848606824874878, 0.14730454981327057]

Test loss: 0.11139528453350067

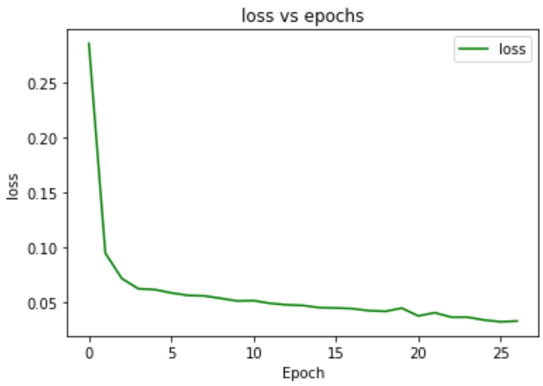
Train accuracy: [0.3142857253551483, 0.5571428537368774, 0.6285714507102966, 0.6714285612106323, 0.6571428775787354, 0.8142856955528259, 0.6571428775787354, 0.7142857313156128, 0.8142856955528259, 0.7285714149475098, 0.8857142925262451, 0.8714285492897034, 0.9571428298950195, 0.9428571462631226, 0.9571428298950195, 0.9428571462631226, 0.9571428298950195, 0.9428571462631226, 0.9428571462631226, 0.9571428298950195, 0.9571428298950195, 0.9428571462631226],

Test accuracy 0.9800000190734863

F1 Score: array([1, 0.96551724, 0.96969697])

Results for Breast Cancer Dataset:

The optimal units in dense layer is 608, optimal learning rate is 0.001 and The best activation function is tanh



Train loss :[0.28519323468208313, 0.0943077877163887, 0.07126504927873611, 0.06192169338464737, 0.06116223707795143, 0.05814981460571289, 0.055895354598760605, 0.05544576421380043, 0.05318291857838631, 0.050821855664253235, 0.05111201852560043, 0.04865354672074318, 0.04728586599230766, 0.0467381477355957, 0.04471301659941673, 0.044385287910699844, 0.0438203401863575, 0.04196745529770851, 0.04133710265159607, 0.04427610710263252, 0.03719068691134453, 0.04010620713233948, 0.03607422113418579, 0.036058843135833374, 0.033420391380786896, 0.0318572036921978, 0.032520685344934464]

Test loss: 0.05041830614209175

Train accuracy: [0.9060150384902954, 0.969924807548523, 0.9849624037742615, 0.981203019618988, 0.9887217879295349, 0.9887217879295349, 0.9924812316894531, 0.9924812316894531, 0.9924812316894531, 0.9924812316894531, 0.9924812316894531,

0.9924812316894531, 0.9924812316894531, 0.9924812316894531,
0.9924812316894531, 0.9924812316894531, 0.9924812316894531,
0.9924812316894531, 0.9887217879295349, 0.9849624037742615,
0.9924812316894531, 0.9924812316894531, 0.9924812316894531,
0.9924812316894531, 0.9924812316894531, 0.9924812316894531,
0.9924812316894531]

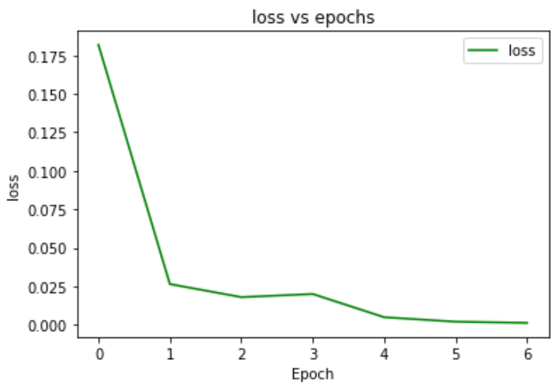
Test accuracy 0.9840425252914429

F1 Score: 0.9876543209876544

Results for Bank-Note Dataset (Kaggle):

The optimal units in dense layer is 960, optimal learning rate is 0.01 and The best activation function is relu

Train loss:[0.1819043606519699, 0.02638547122478485, 0.017887987196445465, 0.01993984542787075, 0.004832809325307608, 0.0019258253742009401, 0.0011715113651007414],



Test loss 0.0010091392323374748

Train accuracy: [0.9020217657089233, 0.9875583052635193, 0.9953343868255615, 0.993779182434082, 1.0, 1.0, 1.0]

Test accuracy :1.0

F1 Score: 1.0

Github Link: https://github.com/vijjisomu/LeaningActivationFunction.git