**Programming Test: Learning Activations in Neural**

**Networks**

**Name: Shwetha S Kulloli**

**Contact:9035252757**

**Email id:** [**shwethak205@gmail.com**](mailto:shwethak205@gmail.com)

---------------------------------------------------------------------------------------------------------------------

**Bank Note Authentication**

**PROBLEM DEFINITION:** It aims to detect fraudulent notes**.**

**OBJECTIVE:** The goal here is to model is to classify that a banknote is fraudulent or not.

**TYPE OF PROBLEM:**

It is a classification problem as the output class has a Binary data two values: 0 (Legal) or 1 (fraudulent).

**DATASET INFORMATION:**

Features: 5

Input Features: variance, skewness, curtosis, entropy

Output Features: Class

Size: Rows🡪1372, Columns🡪5

**BASIC EXPLORATORY DATA ANALYSIS:**

1. Imported required Libraries (Numpy, Pandas, Matplotlib.pyplot, Seaborn, SKlearn, TensorFlow)
2. Checked for the basic information regarding the features such as column names, datatypes of the column names, null values and information about the column names
3. Checked missing values in the data and found that there are no missing data
4. To calculate the data distributions and count plot is used which shows of instances for each class, which also indicates dataset is balanced.

**DATA PRE-PROCESSING:**

1. **STANDARD SCALER was used.** This method assumes data to be normally distributed within each feature and scales them in such a way that the distribution becomes centred around 0 with a standard deviation of 1.
2. **SPLITTING DATASET INTO DEPENDENT AND INDEPENDENT VARIABLES**

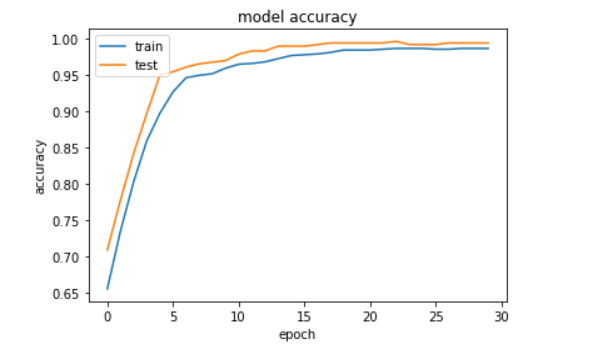
**ALGORITHM :**

The model was built using basic Keras neural network using the Sequential model API.

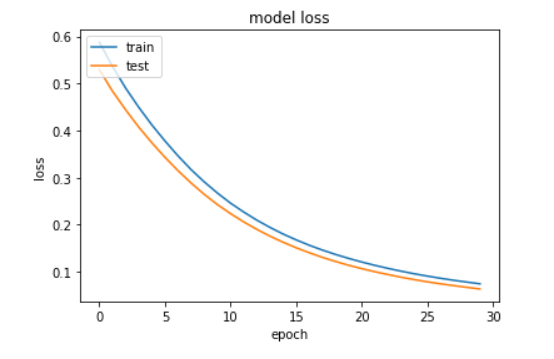
**MODEL BUILDING:**

1. Initialised the sequential model.
2. Added the input layer and the first hidden layer using activation function “Relu” with kernel\_initializer='he\_normal'.
3. Added the output layer with activation function “Sigmoid”.
4. Compiled with optimizer = 'adam', loss = 'binary\_crossentropy' and metrics as accuracy,f1 score, precision and recall
5. Trained the model on the training set with validation\_data = (X\_test,Y\_test),batch\_size = 32, epochs = 30, verbose=2.
6. Evaluated Train and test accuracy and loss.

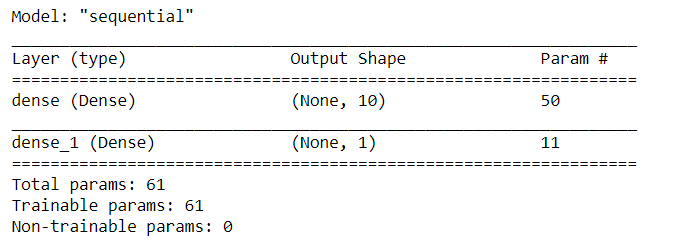
**Plot of Model Accuracy Vs Epoch**



**Plot of Model Loss Vs Epoch**



**MODEL SUMMARY:**



**RESULT OBTAINED:**

Training Accuracy obtained is **0.9858**

Testing Accuracy obtained is **0.9933**

**Other Metrics:**

|  |  |
| --- | --- |
| F1 Score | 0.9926 |
| Precision | 0.9860 |
| Recall | 1.0 |
| Cohens kappa | 0.9867 |
| ROC AUC | 0.9999 |

**Conclusion:**

The accuracy of the model is 99% which indicates that model efficiently detect that a banknote is fraudulent or not.