Assignment 1

The objective of this Assignment is to build neural network model on MNIST (digit) data. So, first we loaded the dataset from Keras using load model.

Text

Description automatically generated

The dataset consists of 60k Training data and 10k Testing data. The data consists of handwritten digits from 0 to 9.

Text, shape

Description automatically generatedThis is just a sample output to display first 25 images along with labels. To start with basic model, I have created a neural network having hidden layers of 128 neurons and 10 dense layers with activation function as “relu”. The metrics used to measure the performance of the model is Accuracy.

Chart, line chart

Description automatically generatedI have split the data set into Train, test and Validation in a ration of 0.33. The number of epochs are 10. The optimizer I have used is an Adam optimizer. With the above mentioned parameters, I received an accuracy of 94.45%.

This is the plot of model accuracy vs model loss during the epochs. We can observe that there is increase in the Train accuracy simultaneously there’s drastic decrease in model loss. To start with experimenting, I tried out 3 different variations in the number of dense layers.

Initially, I had 10 dense layers, other variations, I looked into were 15 layers and 25 layers. The major observations, I found was in the initial model error, for 10 dense layers, the model loss started from 0.5 and boils down to 0.15. But, for other number of layers i.e for model with 15 layers, the initial error was 0.35 till 0.5. And for layers with 25 the error was 0.14. Out of all the variations the model having more dense layers outperformed with accuracy of 95.5%.

The other part of assignment involved to perform different variations in the number of Hidden layers and number of neurons in each layer. To start with I tried out different combinations having different number of hidden layers such as 2 Hidden layers having 16 neurons each. As, a result there was significant decline in the accuracy dropping to 87 % from 95.5%. By just increasing the number of neurons in the layers , I could see that accuracy was increased from 87% to 92%, on moving ahead by increasing it to 64, the accuracy attained was 94.8%, with 128 neurons 95.7%. So, we can conclude that number of neurons in hidden layers contributes majorly in the accuracy of the model.

Chart, line chart

Description automatically generatedChart, line chart

Description automatically generated

2 Hidden layers, 16 neurons 2 Hidden layers, 128 neurons

Now, after having variations in the number of the neurons in hidden layers with same count. I tried different combinations of the number of neurons in each layer. Different combinations, I tried were 2 layers having 16 in first layer and 32 in other, 32-128, 64-128,128-64 and so on.

We can see that as the no. of Units increases from 16, 32, 64 and 128, There is drastic increase in the accuracy with highest accuracy of 95.74. Here, we have taken 2 layers with same number of neurons. But, if we take combinations of neurons at each layer, we can see increase in the Accuracy. The Highest Accuracy achieved is **96.43%** with 3 layers and decreasing number of neurons at each layer.Funnel chart

Description automatically generated