

# Chapter 49

## Correlation Between Number of Hidden Layers and Accuracy of Artificial Neural Network



Purva Raut and Apurva Dani

### 1 Introduction

Artificial neural networks are relatively crude electronic networks of neurons which aims at making machines learn knowledge like a human does. The neural network takes one input at a time and then further processes and learns by comparing the desired output and the result from the neural network. The error which is calculated from the first input is fed back to the network and is used to modify the weights between the neurons. This process of reducing error correction is performed for many iterations. A neuron has two major components:

1. Input values and random weights which are associated with it.
2. Summation function [ $u$ ] that adds the weights together and maps it to an output [ $y$ ].

There are different layers in an artificial neural network which consists of neurons. The input layer is composed just of the input values and not the neurons which act as input to the next layer.

The next layer is hidden layer; there may be several of them and this paper focuses on how varying the number of hidden layers correlates with the accuracy of the model (Fig. 1).

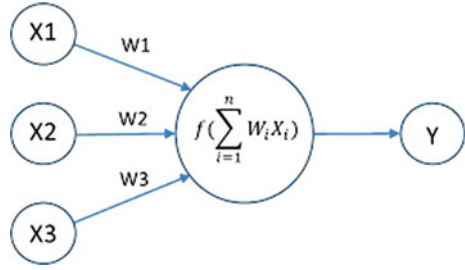
The hidden layers take weighted inputs and the output which is given by this layer is based on an activation function. There is no fixed rule about the number of hidden layers which should be used to create the neural network.

---

P. Raut · A. Dani (✉)  
Dwarkadas J. Sanghvi College of Engineering, Mumbai 400056, India  
e-mail: [apurvadani@gmail.com](mailto:apurvadani@gmail.com)

P. Raut  
e-mail: [purvapraut@gmail.com](mailto:purvapraut@gmail.com)

**Fig. 1** Structure of a simple neural network



## 2 Hidden Layer and Its Working

For training a neural network, the following steps are performed in a loop so that the weights of each input to the hidden neurons can be adjusted to get the least error possible:

1. In first step, forward propagation is implemented.
2. In second step, the loss is computed.
3. In third step, backward propagation is implemented to get the parameters to adjust the weights.
4. In step four, the parameters are updated to reduce the error.
5. In the final step, forward propagation is implemented.

In the first step, generally, the hidden layers use ReLU activation function and the output layer uses sigmoid activation function.

The forward propagation is computed using the following equations:

- Computation at first layer of activation:

$$Y^{[1]} = W^{[1]}X + b^{[1]}, A^{[1]} = \mathbf{ReLU}(Y^{[1]})$$

- Computation at nth activation layer:

$$Y^{[n]} = W^{[n]}A^{[n-1]} + b^{[1]}, A^{[n]} = \mathbf{ReLU}(Y^{[n]})$$

- Computation at last activation layer:

$$Y^{[L]} = W^{[L]}A^{[L-1]} + b^{[1]}, A^{[n]} = \mathbf{sigmoid}(Y^{[L]})$$

- Computation of loss function:

$$\frac{-1}{m} \sum_{i=1}^m ((y^i) \log(a^{[L](i)}) + (1 - y^i) \log(1 - a^{[L](i)}))$$

After implementing forward propagation, backward propagation is calculated using the following steps:

- First, we perform linear backward propagation.
- After that linear to activation backward where the derivative of ReLU or sigmoid activation is computed.
- [linear to ReLU]  $X(N - 1)$  to linear to sigmoid backward (entire model).

After completion of all the above-mentioned steps, we use gradient descent to update the parameters.

### 3 Procedure

In this paper, an artificial neural network [1] was trained for four different datasets and for each model, the number of hidden layers was varied to see the effect of the number of layers on the accuracy of the model. While changing the number of layers, all the other factors such as number of neurons for a level, activation function, and other variables were kept constant. After training each model, the same data was tested for all neural networks for a dataset. Then, for every dataset, a graph was plotted which visualizes accuracy [2] for a different number of hidden layers on a given dataset.

### 4 Dataset

There are four different datasets used for the following experiment. The range of number of rows varies from 1,000 to 10,000 and range of number of columns varies from 4 to 8. The datasets are further divided into 25:75 for training [3] and testing, respectively. Accuracy is used as the performance measure of the neural network in this experiment. The accuracy is defined as the number of predictions of testing set that is correct to that of the total number of cases that are used for testing.

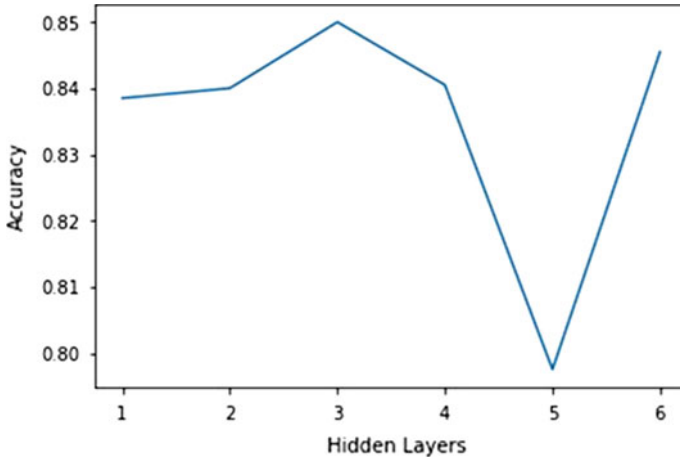


Fig. 2 Accuracy of dataset 1 with varying number of hidden layers

## 5 Results

The following graphs are for four different datasets where *X*-axis shows the number of hidden layers, whereas *Y*-axis shows the accuracy attained. Along with hidden layers, there is an output level which is not considered while plotting the graph.

### 5.1 Dataset 1

The graph below shows the accuracy of dataset 1 with number of hidden layers increased from 1 to 6 (Fig. 2).

### 5.2 Dataset 2

See Fig. 3.

### 5.3 Dataset 3

See Fig. 4.

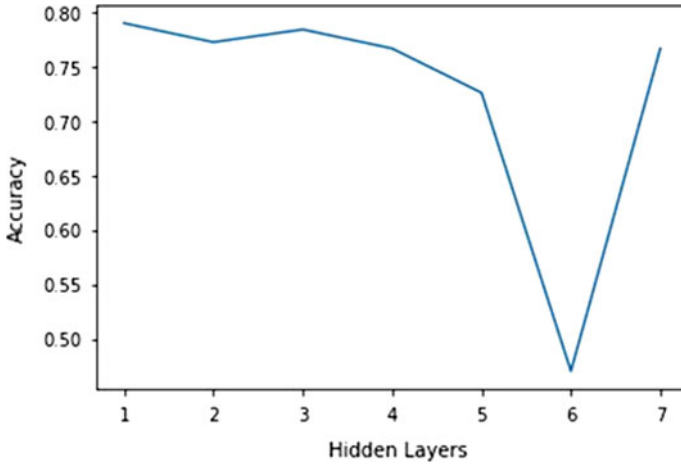


Fig. 3 Accuracy of dataset 2 with varying number of hidden layers

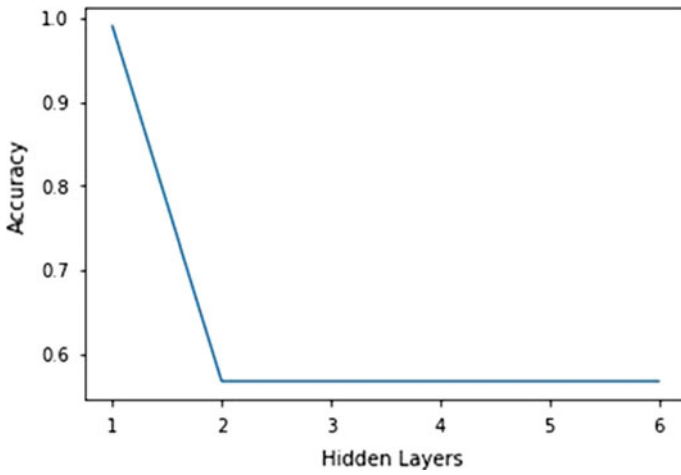


Fig. 4 Accuracy of dataset 3 with varying number of hidden layers

### 5.4 Dataset 4

See Fig. 5.

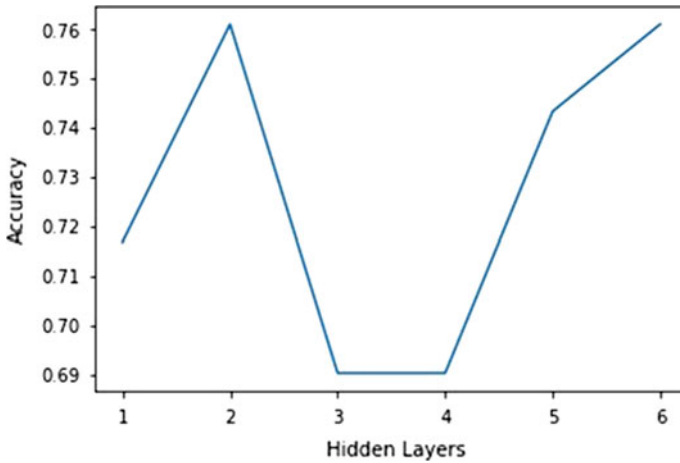


Fig. 5 Accuracy of dataset 4 with varying number of hidden layers

## 6 Observations

The graph above explains how accuracy varies with variation in the number of hidden layers. It can be observed that initially the model’s accuracy increases gradually for certain number of layers and then drops abruptly after reaching a saturation point.

In dataset 1, accuracy starts from around 0.8385, reaches its maximum when there are three hidden layers. All the graphs can be summarized in a similar way, and Table 1 gives an insight about the maximum accuracy and hidden layers.

Theoretically, if appropriate number of neurons is selected for the first hidden layer of the neural network then it can fit most of the hypothesis and hardly there is a need to add more hidden layers for the network.

Table 1 Hidden layer at which maximum accuracy was found

Dataset	Accuracy	Hidden layers
1	0.85	3
2	0.7907	1
3	0.99	1
4	0.761	2

A function that has a continuous mapping from one finite space to another can be approximated with the help of only one hidden layer.

However, one hidden layer can approximate any function that contains a continuous mapping from one finite space to another.

An arbitrary decision boundary to arbitrary accuracy with rational activation can be represented using two hidden layers. It can also be used to approximate smooth mapping to any accuracy.

When you do not use any hidden layer, the network can only be used to represent functions which are linearly separable.

So, from the above observations, it is clear that accuracy can be improved by increasing the number of hidden layers from 2 to 3 or 1 to 2 or 0 to 1 for small datasets.

## 7 Correlation Between Accuracy and Number of Hidden Layers

If the number of hidden layers which are used to build the network are much more than what is required for the given dataset, the accuracy of the test set will decrease. Such networks will overfit the training data, that is, it will perfectly learn the data which is given for training, but it will fail to generalize for the test data.

Figure 6 reflects the problem of underfitting and overfitting. Here, we have a set of data points and we will try to fit the best function we can to fit the data.

In the first figure, we are trying to fit a linear function to fit all the data points. As we can see that the function is not complex enough to fit all the data points and it suffers from the problem of underfitting. In the second figure, we try to generalize the data with a more complex function. It can be seen that the model has learned the trend that the points in the data follows which is a parabola. In the last figure, we have increased the number of hidden layers more than the model required and we can see that it suffers from the problem of overfitting. That is it could not learn

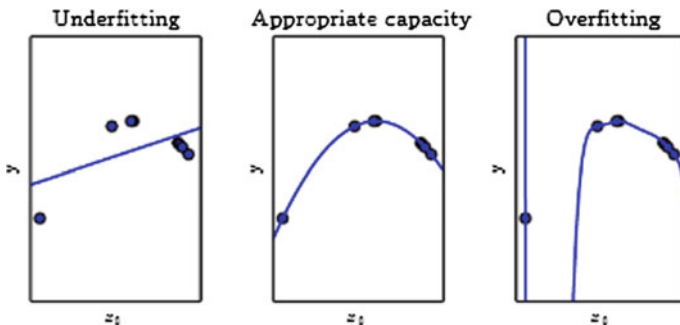


Fig. 6 Simulating the model graph with increasing number of hidden layers [4]

what the trend was and thus it fails to correctly predict the results of test data. Thus, by increasing the number of hidden layers in the neural network, the model fails to generalize the trend to the new data. Thus, it gives poor accuracy with the testing set and this can be reflected in the result section of this paper.

## 8 Conclusion

In most of the practical cases, where we are using small dataset, there is no need of having more than two hidden layers for getting a real good accuracy. Increasing the number of hidden layers will reduce the accuracy of the model since the backpropagation algorithm loses its effectiveness.

When you increase the number of hidden layers in the neural network, the error that you will get while using the model to predict test dataset will increase, even though the model was correctly predicting for the training set due to overfitting.

The accuracy of the model depends upon the performance of the architecture of the network and the algorithm used in its test dataset.

When a network tries to fit the data very closely, it will have a huge generalization error and a very high variance because of overfitting.

Thus, to decrease this variance, we need to smooth the network outputs, but while reducing the variance, the bias may increase to a huge value and the error in generalization will be large again. This is the case of underfitting. Thus, the balance between the bias and variance plays a huge role in applying neural network to practical applications.

Following solutions can be used to avoid the problem of underfitting:

1. There should be enough number of hidden nodes in the network so that the function properly fits the dataset. It should be capable enough to represent the mappings of the data points.
2. To reduce the cost of sum squared error, the network should be trained for enough amount of time.

To prevent overfitting:

1. The network should not be trained for extremely long time that it does not learn the trend and it only fits the training data.
2. The adjustable parameters like number of hidden layers of the network must be restricted so that the chance of overfitting reduces.



## References

1. Da Silva IN et al (2017) Artificial neural networks. Springer International Publishing, Cham
2. Rauber PE et al (2016) Visualizing the hidden activity of artificial neural networks. *IEEE Trans Visualization Comput Graph* 23(1):101–110
3. Günther Frauke, Fritsch Stefan (2010) Neuralnet: training of neural networks. *R J* 2(1):30–38
4. Goodfellow I, Bengio Y, Courville A (2016) Deep learning. MIT Press, Cambridge