

---

# Neural Networks and Redundancy in Network Layers

---

Anonymous Author(s)

Affiliation

Address

email

## Abstract

1 This documents is a look into the redundancy of extra neural network layers. This  
2 will be an overview of the benefits or the faults of having an extensive amount of  
3 layers, first layer Neural Networks without any dropout in the neural network.

## 4 1 Deep Learning

5 Deep Learning The study of deep learning is based on the idea of the human brain. It starts off with  
6 the idea of making a computer think like a human. This has been the ideal for many years for society,  
7 and there have been many different approaches to creating a human-like machine.

8 To complete this ideal, the idea of the perceptron was created. The perceptron is an electronic “node”  
9 that takes in multiple inputs, each line consisting of a different value from the input line, and a  
10 weighted value, that the input will be multiplied. This will create a number that would be between  
11 zero and one. This may be used for identification of items, such as handwriting, labels, stock market  
12 trends, and, given a large enough network, the ability to mimic certain human like behaviors. The  
13 perceptron helped bring forward the Artificial Intelligence idea of having human like robots forward,  
14 however, there was a problem with how many perceptron a computer could contain. This has been a  
15 huge wall that the community has not completely overcome, however, over the years, there have been  
16 several breakthroughs. A major one being the development of Neural Networks.

## 17 2 Neural Networks

18 The development of the Neural Network helped push the envelope of what was possible in the world  
19 of Artificial Intelligence. This was completed by taking advantage of the perceptrons that were  
20 created by linking them together and treating them like a set of neurons. This helped push this  
21 forward for learning when it came to Artificial Intelligence as not only were perceptrons able to  
22 identify a single item anymore, but a several layers of them could break up a set of information, be it  
23 an image or a multiple data pieces, so that each perceptron could be used to identify smaller sections  
24 of an item, increasing the likelihood of identification, or trends within a given market.

25 This development lead to several perceptrons that could be used for better identification, such as when  
26 attempting to identify numbers. Instead of a single perceptron looking at the image, broken up into  
27 different inputs, instead you could have several perceptrons that would break up the same image and  
28 attempt to decide if a number is a 4 or a 9, and pull the inputs back in from both perceptrons which  
29 would increase the likely hood that you would correctly identify an image as the correct number. But  
30 with Neural Network development, there was a thought that why not layer these networks, passing  
31 the inputs from the previous layer into the new one to attempt to increase the correct guess. But  
32 before you could attempt a Multi-Layer Neural Network, the network that you were given would  
33 need to be trained first. Training a Neural Network, while not difficult, does take an intense amount  
34 of information, and an even larger set of data to work through.

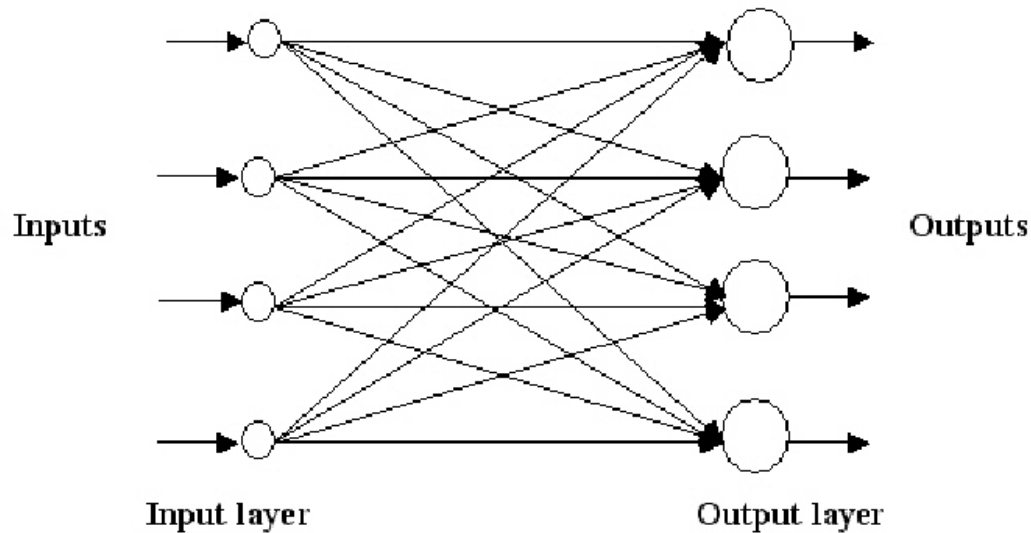


Figure 1: Single Layer Neural Network

### 3 Training a Neural Network

To train a Neural Network, a large dataset is required. When Neural Networks were new, this meant there were many hours spend collating and combining given information to learn to train a Neural Network. To do this, you take part of a given sample set, and run it through the Neural Network, and when the output is given, if it is correct, you do nothing, but if the output is wrong, you run calculations in reverse to help even out the networks weighting system for each input. This is done by calculating the error that comes from each node in the network, and then adjusting it, given a certain training weight or percent, and adjusting the weight number by the amount of error multiplied by the percentage that is taken.

Given that large datasets are required, this lead to many hours of attempting to even out a network to approach that perfect identification, even given a piece of data that was not included in the training dataset. But with the error still being large, even given expansive inputs for a Neural Network, the attempt at making a Multi-Layer Neural Network was the next approach that was taken to attempt to even out this error that single layers still gave.

### 4 Multi-Layer Neural Networks

Multi-Layer Neural Networks were a very good reinvigoration for the fields of Deep Learning and Neural Networks. What a Multi-Layer Neural Networks consists of are several single layers, some completely connected to the next layer, some that only connect to a handful of the nodes in the next layers that could be used to even easier correctly identify items given enough time and training.

The difficult part of using a neural network is not the direct training, but also, how you will use the given information for each of the layers. Unlike a single layer Neural Network, a Multi-Layer Neural Networks does not take in new information for each layer, but instead after the first layer of nodes process the image, the output is then passed forward into the new layer of nodes which then process it forward. This layer effect can be used for between two to fifteen layers, but typically they do not cover more than that.

The largest restriction on how many layers a given project could use was not restricted by the person working on the development of the network, but again, we ran into a hardware issue. Given that at the time of its development, multi-core processors were unheard of this slowed development for several years. However, with the development of multi-core processors and access to GPU processors that

64 could be used for processing, the speed boost to these networks were unbelievable and still being  
65 tested with current technology.

66 The development of the Multi-Layer Neural Networks helped again push the envelope when it came  
67 to identifying a given image. However, this also meant that a new way of training had to be developed,  
68 as adjusting the weights on the final layer of the neural Network would only adjust the entire system  
69 so much.

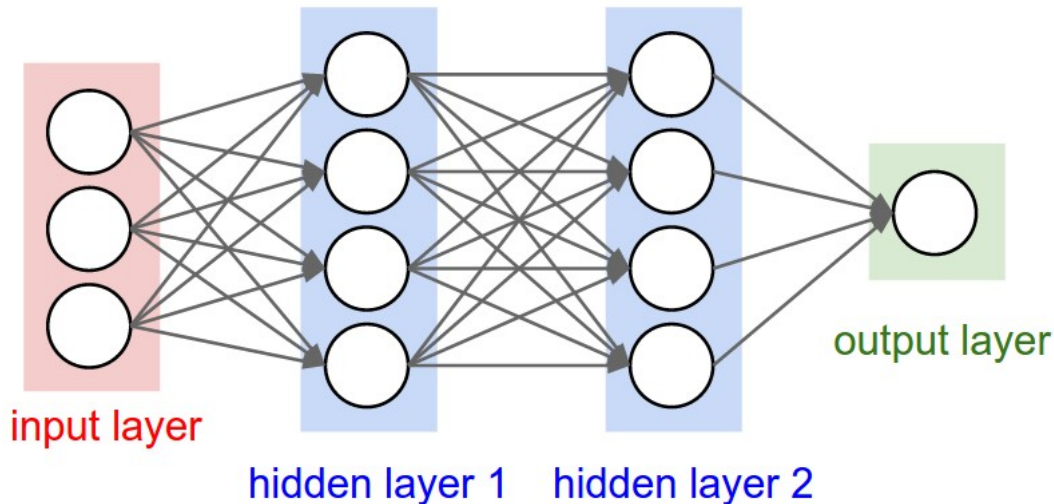


Figure 2: Multi-Layer Neural Network

## 70 5 Training Multi-Layer Neural Networks

71 Since Multi-Layer Neural Networks require several layered Neural Networks, one of the more  
72 efficient ways of enhancing the system was to add adjustment to the previous layers. This seemingly  
73 worked out very easy in the end. The process still involved adjusting the final Neural Network  
74 weights, however, once that is completed, this new error is passed back by each node to the previous  
75 layer, and given different weights adjust the next previous layer's weights. This is repeated for each  
76 successive layer until the first layer is reached. From here the Multi-Layer Neural Networks can  
77 process new training data, and again attempts a correct identification.

78 Single Layer vs Multi-layer Neural Networks The benefits of a Multi-Layer Neural Networks make it  
79 seem like an obvious choice when attempting to train a Neural Network, however, there are some  
80 benefits that a Single Layer may give you.

### 81 5.1 Training

82 Since a Single Layer Neural Network is much faster, completing the training is very simple and helps  
83 with completing the training very quickly if that is needed. However, this also leads to a greater error  
84 chance when it comes to correctly identifying a given image. The benefits of having quicker training  
85 is a system that is up and running much faster.

86 The Multi-Layer Neural Network on the other hand may be more accurate, but the training required  
87 and the back propagation for correctly identifying which node is causing error could take may more  
88 iteration. However, as the graphs below show, the error can be greatly reduced as well.

### 89 5.2 Large Datasets

90 Having to process large sets of data with a Neural Network can lead to a huge amount of downtime.  
91 For a Single Layer Neural Network, this is usually not the problem, as the use of this would indicate a  
92 huge amount of error allowed.

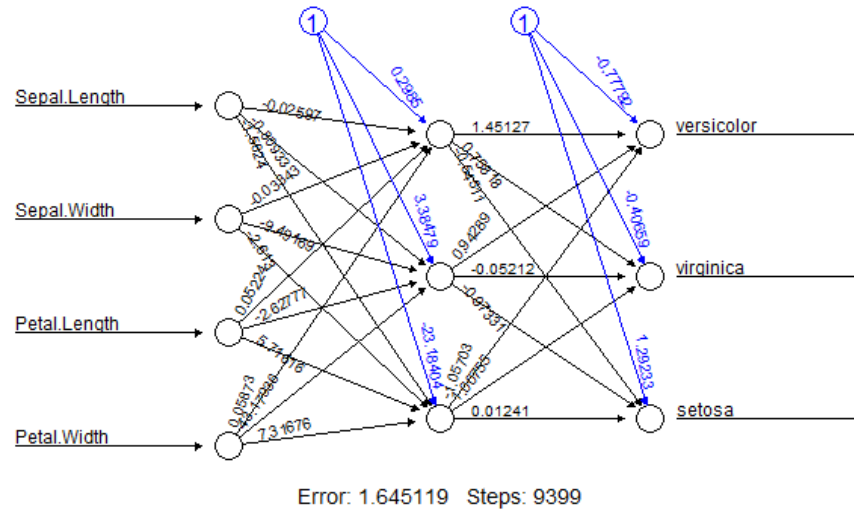


Figure 3: Neural Network Training

93 Meanwhile, Multi-layer Neural Networks would take even longer, but because they have such a large  
 94 dataset, their ability to correctly identify a given image would greatly increase their ability to perform  
 95 on the system over all.

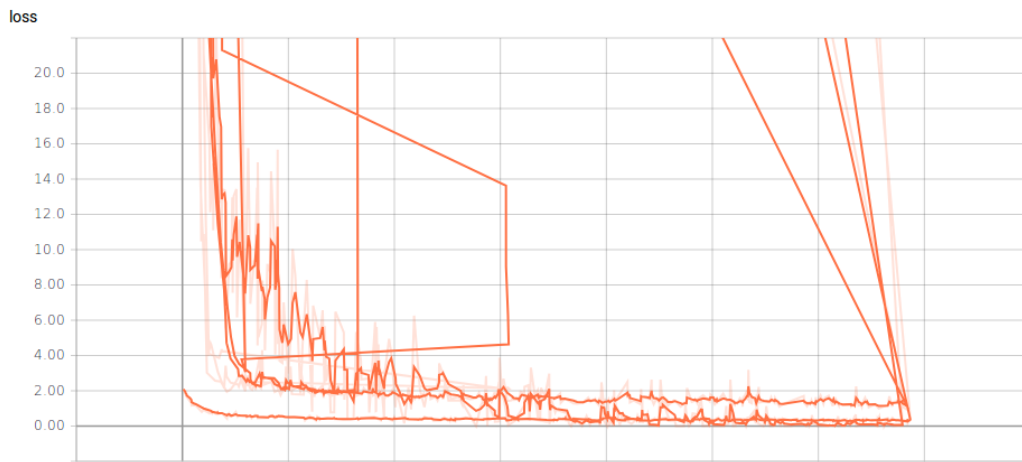


Figure 4: 3 Layer Error Percent

### 96 5.3 Large Training Iteration

97 An easy win is pointed out in this area for the Multi-layer Neural Networks as it does take longer, but  
 98 again, this would be needed to correctly identify a image much more accurately. This is because the  
 99 Single Layer Network would not be able to correctly train even given a large enough data set.

## 100 6 Redundancy

101 Now while Multi-layer Neural Networks have a huge benefits over smaller Single Layer Network,  
 102 they do suffer from one major issue. Redundancy is a huge problem when it comes to making  
 103 Multi-layer Neural Networks as too many layers is just as bad as too few. During testing, there were

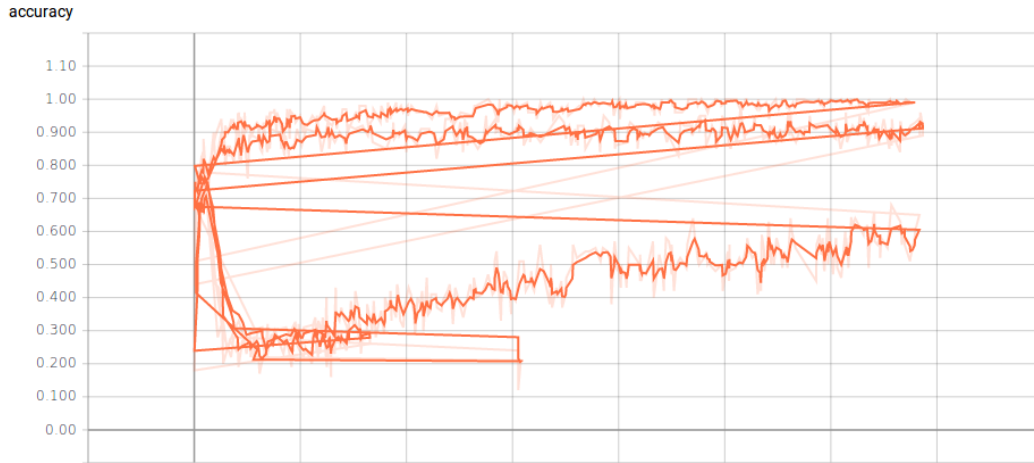


Figure 5: 3 Layer Correct Identification

several different multilayer networks ran, once consisting of 15 layers, one consisting of 5 layers, one of 3 layers, and one being a single layer Neural Network. What was shown is more layers than the 5 actually reduced the accuracy with which a item can be identified and also creates problems in correctly identifying the training data set around 135,000 iteration of training.

## 7 Testing and Conclusion

The Single Layer Neural Network worked well when running on its own, but the performance never made it higher than 80% efficiency when identifying anything outside of the training set.

The 5 Multi-layer Neural Network on the other hand seemed to be able to correctly identify the training dataset at the 134,000 training iteration with 100% accuracy and the actual tested dataset with 98.74% accuracy. This was tested against the MNIST handwriting data set to make iteration of training easier.

The 15 Multi-layer Neural Network when tested worked well when training iteration we arond 10,000. However, after an additional 100,000 training iterations, the percentage of correctly identified images would drop 1-4% on different run. While this may not seem like much of a drop, the 5 Multi-layer Neural Network never dropped below 96% identification, even when run with only 100 training iteration.

This may go to represent that while a Multi-layer Neural Network is better than a Single Layer Neural Network, to much work in a Multi-layer Neural Network may be just as harmful to the overall performance of a system as to little.

## References

- [1] David A. Medler & Michael R. W. Dawson Using Redundancy to Improve the Performance of Artificial Neural Networks\*
- [2] Micheal Cogswell (2016) Thesis submitted to the Faculty of Virginia Polytechnic Institute and State University *Understanding Representations and Reducing their Redundancy in Deep Networks*