REDUCTION OF NEURAL NETWORK COMPLEXITY THROUGH STOCHASTIC NUMBERS

James Ridey

Bachelor of Engineering
Software Engineering

School of Engineering
Macquarie University

October 22, 2020

Supervisor: Associate Professor Tara Hamilton

ACKNOWLEDGMENTS

I would like to acknowledge and thank, Tara Hamilton and Alan Kan for their support and advice during my thesis.

STATEMENT OF CANDIDATE

I, James Ridey, declare that this report, submitted as part of the requirement for the award of Bachelor of Engineering in the School of Engineering, Macquarie University, is entirely my own work unless otherwise referenced or acknowledged. This document has not been submitted for qualification or assessment an any academic institution.

Student's Name: James Ridey

Student's Signature: James Ridey

Date: 7/6/2019

ABSTRACT

Neural networks considered a core foundation of machine learning, has recently ballooned into a large framework that requires vast amounbts of computing resources and time. Here, we examine how to improve these networks, hoping to achieve networks that can calculate the same result with less training time and/or resources using ideas such as stochastic numbers and probalistic results.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Neural networks: introduction

Neural networks are the core structure of what many believe to be the next step in computing, machine learning.

Chapter 2

Background and Related Work

2.1 Neural networks: explanation

Neural networks are a very rough simulation of how the neurons in a human brain actually work. The most common form of a neural network in the feed forward model where the inputs are passed to the neural network and it outputs an answer. The feed forward model consists of an input layer, hidden layers and an output layer each of these layers are composed of varying amounts of neurons depending on the complexity of the data that is trying to be learned. More models of neural networks exist which aim to memorise or improve the feed forward model, however the feed forward model is incredibly ubiquidious and is the main driving force behind machine learning in the industry.

To understand what each layer in a neural network is doing, it is first necessary to understand what a neuron is doing. A neuron is comprised of inputs, a set of weights and one output, as shown below.

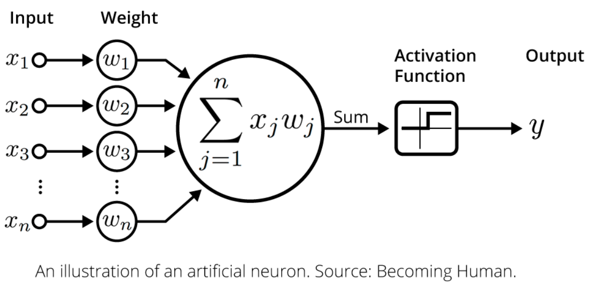

Figure 2.1: Neuron model.

Neurons are fed a set of data through its inputs which are multiplied by a specific weight, all of these numbers are then added up and fed to an activation function. An activation function is a function that maps the input to a value between 0 and 1, a commonly used activation function is the sigmoid function, numbers in the positive infinity direction become increasingly closer to 1, while numbers in the negative infinity direction become increasing closer to 0.

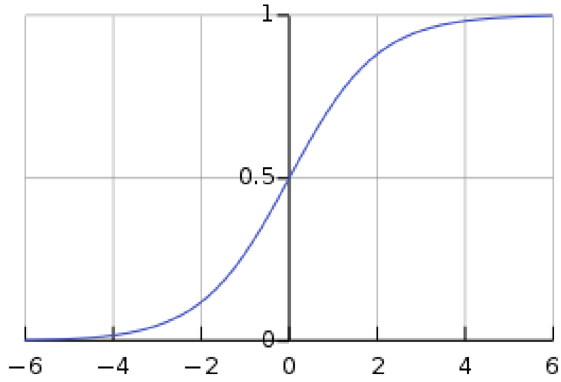

Figure 2.2: Sigmoid graph.

Next is neural network layers where the most common layer is the dense layer, in this layer every neuron connects to every other neuron in the previous layer.

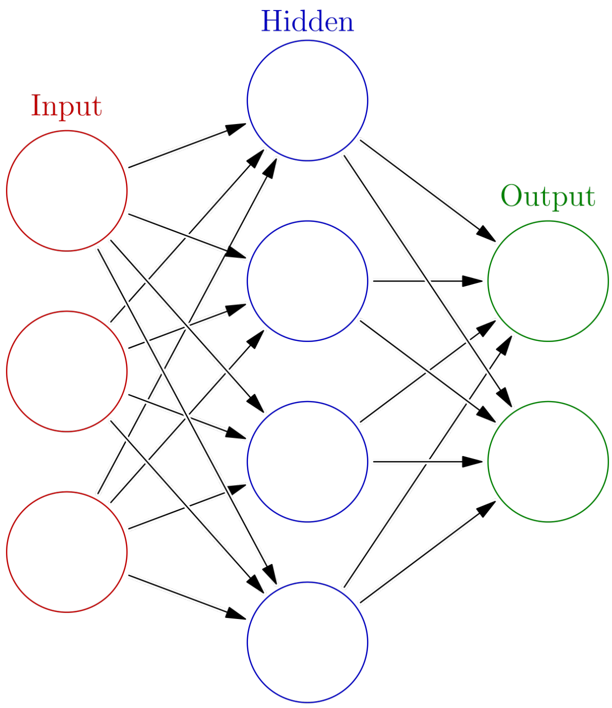

Figure 2.3: Neural network model.

Each circle in this figure represents a neuron and each line in this figure represents a connection between these 2 neurons. By manipulating the weights in the neural network, we change what each neuron calculates and outputs as part of its activation function. In doing so, we now have a framework of which for any given input, the neural network can generate any output (given the right weights), regardless of the complexity of the output [1].

2.1.1 Training

The next question about neural networks are then how are the weights trained for a given input. TODO

2.2 Stochastic numbers

2.3 Tensorflow

Tensorflow is a machine learning framework primarily written in Python. It was released by Google under the Apache License 2.0 on November 9th, 2015. The Tensorflow network employs the idea of creating a neural network model which is then compiled before being handed off to the appropiate processing unit be it CPU or GPU, this is in contrast to other machine learning frameworks like PyTorch that generate their models dynamically on the fly which results in a small performance hit.

Tensorflow continues to grow in support with new frameworks like Kera that aim to simplify common operations and overall streamline the process and show no signs of stopping with a stable release only 2 months ago.

2.4 MNIST dataset

The MNIST dataset is a commonly used dataset for machine learning that consists of 70,000 images of handwritten digits (60,000 training images and 10,000 test images) and the associated labels for each of these images. The dataset is commonly choosen due to its simplicity and many frameworks support it out of the box.

Another common reason that MNIST is choosen is to demonstrate the accuracy of neural networks over more traditional statistical analysis such as linear classifiers and K-nearest neighbours, both of which score around an error rate of 12% and 5% respectively [?] as compared to a simple neural network of 3 layers which can score around 3.05% [?].

However this dataset is not without its faults, there are quite a number of images that are dubiously labeled.

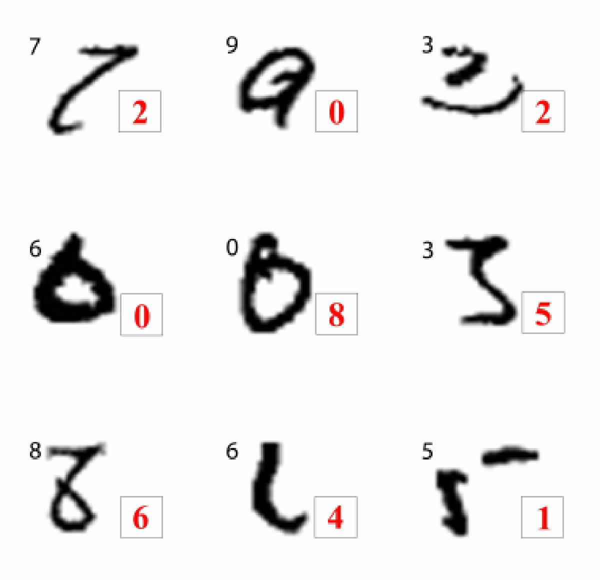

Figure 2.4: Samples of ambigious digits.

The dataset is also often constructed as too small and too simple, consisting of only 70,000 images of greyscale images. Compared to datasets such as CIFAR10 which has the same amount of images but 32x32 color images or another dataset like ImageNet consisting of 1,331,167 images, MNIST is certainly behind. However MNIST has been used and is continued to be used as a way to quickly validate a machine learning algorithm before moving on to larger more complicated datasets that take much more time to compute.

Chapter 3

Preliminary results

3.1 Stochastic numbers

Chapter 4

Conclusions and Future Work

4.1 Conclusions

This

Chapter 5

Abbreviations

AWGN	Additive White Gaussian Noise
SC	Stochastic numbers
NN	Neural network

Appendix A

name of appendix A

A.1 Overview

here is the Overview of appendix A ...

A.2 Name of this section

here is the content of this section ...

Appendix B

name of appendix B

B.1 Overview

here is the Overview of appendix B ...

B.2 Name of this section

here is the content of this section ...

Bibliography

[1] M. A. Nielsen, "Neural networks and deep learning," 2018. [Online]. Available: http://neuralnetworksanddeeplearning.com/chap4.html