**Implementation of a Basic Neural Network in C**

Mario Martínez Vitutia

November 2024

Table of contents

[1 Introduction 5](#_Toc182580822)

[1.1 Project Objective 5](#_Toc182580823)

[1.2 Context and motivation 5](#_Toc182580824)

[1.3 Overview of the Document 5](#_Toc182580825)

[2 Theory and Background 6](#_Toc182580826)

[2.1 Neural Network Overview 6](#_Toc182580827)

[2.2 Key Algorithms Used 6](#_Toc182580828)

[3 Design and Implementation 6](#_Toc182580829)

[3.1 Project Structure 6](#_Toc182580830)

[3.2 Definition of Structures 6](#_Toc182580831)

[3.2.1 Neurons 6](#_Toc182580832)

[3.2.2 Connections 6](#_Toc182580833)

[3.2.3 Layers 6](#_Toc182580834)

[4 Training the Neural Network 6](#_Toc182580835)

[4.1 Creating Connections Between Layers 6](#_Toc182580836)

[4.2 Data Normalization 6](#_Toc182580837)

[4.3 Forward Propagation 6](#_Toc182580838)

[4.4 Backpropagation 6](#_Toc182580839)

[4.5 Training Functions 6](#_Toc182580840)

[5 Challenge and Difficulties 6](#_Toc182580841)

[6 Conclusion 6](#_Toc182580842)

[6.1 Summary of Results 6](#_Toc182580843)

[6.2 Possible Improvements 6](#_Toc182580844)

[6.3 Reflection on the Project’s Impact 6](#_Toc182580845)

# 1 Introduction

This project implements a basic neural network using the C programming language. It is well known that compiled programming languages (such as C, C#, or Rust) are faster than interpreted ones (Python, PHP, JavaScript…), and this speed difference is more noticeable in programs that require significant computational calculations. However, in fields such as Artificial Intelligence, where techniques like machine learning involve complex computational tasks, Python is the most widely used language. This is due to the availability of powerful libraries and frameworks, even though Python itself may not always be the fastest language. Specifically, this project aims to provide a deep understanding of how neural networks work at a lower level, using C as a low-level language to learn the fundamentals of their structure and training process.

## Project Objective

The goal of this project is to design and implement a basic artificial neural network from scratch capable of solving regression problems. The primary objective is to explore and understand the core concepts of neural networks, in order to reflect on the possibility of programming faster neural networks.

## Context and motivation

I have always been interested in understanding how things work, with focus on their internal functioning and structure. Furthermore, I have been interested in AI for some time, working on various technical projects, with the intention of specializing in this area. Yet, it is not hard to question: why Python is the predominant language for AI programming? Later, we will answer that question and reflect on it. So, basically, the reason I started this project was to study and learn in detail how neural networks function at an internal level. Additionally, I aimed to open a discussion on whether it might be worthwhile to focus more efforts on creating libraries in low-level languages.

## Overview of the Document

Throughout this document, various aspects will be explained, including the theory and fundamentals of neural networks, beginning with what a neural network is, and the algorithms used. The project’s structure will be detailed, covering the custom-built data structures, the functions used to train the neural network, as well as the challenges encountered during development and the conclusions drawn upon project completion.

# Theory and Background

Before explaining the inner workings of a neural network a reviewing the program in C, it is helpful to first clarify what a neural network is, how it operates, and the algorithms it uses.

## Neural Network Overview

Neural networks are computational models inspired by the human brain, designed to process and learn from data. In this context, we define ‘learning’ as the process of minimizing error. Neural networks consist of artificial neurons organized into layers, where these layers are connected by weighted connections. These connections adjust their parameters to reduce the error with each epoch (we will examine this in detail later on). Thanks to this learning, it is not necessary to explicitly indicate to the program what operations, formulas or methods it has to use, the network itself identifies a valid combination of operations that reach the correct answer. This is also known as *soft computing.*

## Key Algorithms Used

# Design and Implementation

## Project Structure

## Definition of Structures

### Neurons

### Connections

### Layers

# Training the Neural Network

## Creating Connections Between Layers

## Data Normalization

## Forward Propagation

## Backpropagation

## Training Functions

# Challenge and Difficulties

# Conclusion

## Summary of Results

## Possible Improvements

## Reflection on the Project’s Impact