Chapter 1

Introduction

Computers are like a bicycle for our minds.

— Steve Jobs, *Michael Lawrence Films*

Our brain is constantly classifying and recognizing. For instance, when we spot a dog on the street, one easy classification we can make is {dog, not dog}, which is probably too easy for our brain—it's almost instantaneous. However, things get a bit more complex when we read the teacher's whiteboard. What happens when we encounter a symbol that confuses us because it resembles another? We can interpret the mathematics behind this reasoning as the brain seeking/creating a function that provides us with the certainty of recognizing that particular letter. Eventually, we reach a point where we feel confident enough to write it down in our notes.

Artificial intelligence aims to replicate the remarkable capabilities of our brains. It seeks to develop computational models and algorithms that can perform tasks such as classification, recognition, and decision-making with a level of accuracy and efficiency comparable to human intelligence. When AI first emerged, one of the initial challenges was actually hand-written digit recognition, exemplified by the MNIST digits dataset. This dataset comprises 60,000 examples of handwritten digits from 0 to 9. To enable a machine learning model to recognize these digits, it must effectively map each image to its corresponding number. This problem naturally aligns with a mathematician's perspective of function learning, where the goal is to approximate a function based on a given dataset consisting of points in space.

Neural networks are widely employed in artificial intelligence to address various problems. The theory of function approximation using neural networks has a long history, dating back to the work by McCulloch and Pitts in the 1940s [?]. These pioneers laid the groundwork for understanding how neural networks can mimic the behavior of biological neurons to compute complex functions. Since then, significant advancements have been made in the design, training, and optimization of neural networks, enabling them to tackle increasingly challenging tasks.

This Bachelor's thesis aims to explore the Leshno-Lin-Pinkus-Schocken Theorem [**?**], a fundamental result in deep learning that establishes the necessary and sufficient conditions for an activation function to enable neural networks to act as universal approximators. Let $\mathcal{M}$ denote certain class of $L_{loc}^{\infty}(\mathbb{R})$ functions (see Definition **??**).

The statement is the following:

> **Theorem 30**
>
> Let $\sigma \in \mathcal{M}$. Set
> $$\Sigma_n = span\{\sigma(w \cdot x + \theta) : w \in \mathbb{R}^n, \theta \in \mathbb{R}\}.$$
> Then Σ_n is dense in $\mathcal{C}(\mathbb{R}^n)$ if and only if σ is not a polynomial.

To motivate this theorem, in the first chapter we start by introducing some basic concepts of machine learning, accompanied by examples of models such as linear regression and logistic regression. We then proceed with a detailed description of a neural network model, accompanied by its purely mathematical definition. Since we take an analytical perspective on machine learning, in the following chapter, we provide definitions and results from functional analysis focused on function approximation. The fourth and final chapter presents the main theorem along with its extensive and detailed proof.