

# 1 Introduction

The "curse of dimensionality" is perhaps mythologised slightly in how it is named, but to those who encounter it, the description likely seems appropriate. The "curse" refers to various phenomena that arise when analysing and organising data in high-dimensional spaces (often with hundreds or thousands of dimensions) that do not occur in lower-dimensional settings. This term is widely used in fields such as data analysis, machine learning, and statistics. In particular it refers to the difficulties that arise as an increasing volume of space results in increasingly sparse data. [?]. . . This phenomena was the primary motivator for this report.

Neural networks, a class of machine learning models inspired by the human brain, are particularly affected by this curse. These networks are designed to recognise patterns and make decisions based on high-dimensional input data. However, as the dimensionality of the input data increases, the amount of data required to train the model and the complexity of the model required to achieve a given accuracy effectively also increase, leading to challenges in model training and generalisation.

We aim to address the curse of dimensionality in neural networks by exploring existing theoretical results and translating them into practical, discretised frameworks. Specifically, we will investigate how these results, which often pertain to specific classes of functions, can be adapted and applied to more grounded, discrete mathematical structures. The objectives of this research include:

- Analysing the existing literature on approximation theory and neural networks, focusing on the classes of functions they attempt to approximate.
- Creating analogous mathematical structures to continuous ones, ensuring they are well-defined and exploring their properties.
- Conducting experiments to verify if analogous theoretical results hold true in these newly defined discrete spaces.

By translating theoretical results from analytic approximations in continuous function spaces to discrete analogs, this research seeks to find how our original results translate to this new framework and hopefully find a solid practical methodology to mitigate the curse of dimensionality in neural networks by use of smarter network architectures, providing a more practical and robust framework for high-dimensional data analysis.

We begin our discussion first with a conversation on philosophy and our underlying motivations - exploring what classes of functions neural networks can already approximate well and making an attempt to bridge the "*real world*" and theoretical.

Neural networks attempt to approximate some function. Much of the theory starts from the premise our target function is a member of some continuous function space. However, in practice, the training data we are working with is discrete and often the functions we attempt to approximate are too. This naturally leads to the question of how well the theory holds up in practice and if we can actually make any guarantees about the performance of our neural networks in these cases.

When we refer to the complexity of a neural network, we are often referring to the number of parameters/neurons it has. Some popular networks such as OpenAI's GPT-4 model are rumoured to have up to 1.7 trillion parameters - this is a huge number and whilst it is great for the ability to learn complex patterns, it also comes at huge computational cost, estimates say that a single GPT-4 query can cost up to a maximum of 2-3¢ (USD) per query in just electricity costs alone, with a large volume of requests, it is easy to see how with increasing demand this number too becomes huge. It is in our interest as a world entering this age of greater AI prevalence to find ways to make these models leaner and more efficient. This piece of work aims to be part of a larger shift in culture towards creating more efficient models by use

of smarter mathematical methods, to enable us to create more capable neural networks, rather than trying to improve performance via unsustainable methods of "throwing more neurons" at a problem to solve it, which in turn contributes to larger problems like our current climate crisis.

Neural networks have the wondrous property of being universal approximators - however this too may be a curse - we see neural network implementations widely today but they are becoming increasingly over-used, ill-suited and vastly inefficient to their respective tasks. If we are to continue to see this trend, we must look for methods to make them more efficient.