Parameter Reduction using Generalized Neural Networks

William Guss

Machine Learning at Berkeley

Abstract

Classification and prediction tasks on high resolution continuous data require models with *exponentially many parameters*. In this paper we generalize artificial neural networks to infinite dimensional Banach spaces to attack the curse of dimensionality. Using this new class of algorithms, $\{\mathcal{G}\}$, we prove a new universal approximation theorem for bounded continuous operators and show that this new functional representation of weights is invariant to the number of samples.

The Problem with High Resolution

Computationally, we deal with *discrete data*, but most of the time this data is sampled from a *continuous process*. For example,

- *Audio*: Inherently a continuous $f : \mathbb{R} \to \mathbb{R}$ sampled as a vector $v \in \mathbb{R}^{44{,}100 \times t}$
- *Images*: Truthfully a function $f : \mathbb{R}^2 \to \mathbb{R}^3$, but sampled as $v \in \mathbb{R}^{3872 \times 2592}$

However, performing tractable machine learning on this data almost always requires some *lossy preprocessing* like PCA or Discrete Fourier Analysis[1]. Even the state of the art approaches, convolutional neural networks, *do not escape the dimensionality issues associated with high resolution data* [1,2].

Our Solution

In answer to this problem, we assume the data is a continuous $f : X \to \mathbb{R}$.

- This leads to a powerful generalization of ANNs, $\{\mathcal{G}\}$ which are *universal approximators of* $K : L^p(X) \to L^q(X)$
- Assuming continuity gives *invariance to input resolution* and a *massive reduction of parameters.*

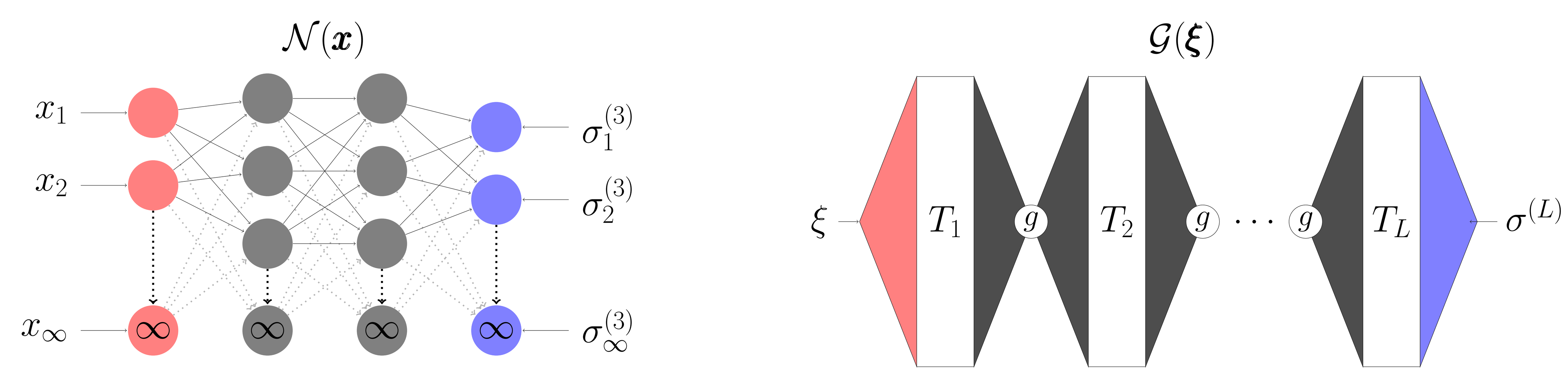

Figure 1: Left: A neural network $\mathcal{N}$ as the number of nodes $\to \infty$. Right: A generalized neural network $\mathcal{G}$.

Operator Neural Networks

Definition 1. *We say $\mathcal{N} : \mathbb{R}^n \to \mathbb{R}^m$ is a feed-forward neural network if for an input vector $\boldsymbol{x}$,*

$$\begin{aligned} \mathcal{N} : \sigma_j^{(l+1)} &= g\left(\sum_i w_{ij}^{(l)} \sigma_i^{(l)} + \beta^{(l)}\right) \\ \sigma_i^{(0)} &= x_i. \end{aligned} \tag{1}$$

Furthermore we say $\{\mathcal{N}\}$ is the set of all neural networks.

$$\Downarrow$$

Definition 2. *We call $\mathcal{O} : L^p(X) \to L^q(Y)$ an operator neural network if,*

$$\begin{aligned} \mathcal{O} : \sigma^{(l+1)}(j) &= g\left(\int_X \sigma^{(l)}(i) w^{(l)}(i,j)\, di\right) \\ \sigma^{(0)}(i) &= f(i). \end{aligned} \tag{2}$$

Furthermore let $\{\mathcal{O}\}$ denote the set of all operator neural networks.

Generalized Neural Networks

Both $\mathcal{O}$ and $\mathcal{N}$ look really similar. Is there some more general category or structure containing them?

Definition 3. *If A, B are (possibly distinct) Banach spaces over a field $\mathbb{F}$, we say $\mathcal{G} : A \to B$ is a generalized neural network if and only if*

$$\begin{aligned} \mathcal{G} : \sigma^{(l+1)} &= g\left(T_l\left[\sigma^{(l)}\right] + \beta^{(l)}\right) \\ \sigma^{(0)} &= \xi \end{aligned} \tag{7}$$

for some input $\xi \in A$, and a linear form T_l. Denote the set of all such networks, $\{\mathcal{G}\}$

Remark. G is a *category*, and we can write neural networks as *commutative diagrams.*

Layer Types

We suggest several types of layers in the category.

T_l is $\mathfrak{o}$*-operational* if

$$\begin{aligned} \mathfrak{o} : L^p(X) &\to L^q(Y) \\ \sigma &\mapsto \int_X \sigma(i) w^{(l)}(i,j)\, di. \end{aligned} \tag{3}$$

T_l is $\mathfrak{n}$*-discrete* if

$$\begin{aligned} \mathfrak{n} : \mathbb{R}^n &\to \mathbb{R}^m \\ \vec{\sigma} &\mapsto \sum_j^m \vec{e}_j \sum_i^n \sigma_i w_{ij}^{(l)} \end{aligned} \tag{4}$$

T_l is $\mathfrak{n}_1$*-transitional* if

$$\begin{aligned} \mathfrak{n}_1 : \mathbb{R}^n &\to L^q(Y) \\ \vec{\sigma} &\mapsto \sum_i^n \sigma_i w_i^{(l)}(j). \end{aligned} \tag{5}$$

T_l is $\mathfrak{n}_2$*-transitional* if

$$\begin{aligned} \mathfrak{n}_2 : L^p(X) &\to \mathbb{R}^m \\ \sigma(i) &\mapsto \sum_j^m \vec{e}_j \int_X \sigma(i) w_j^{(l)}(i)\, di \end{aligned} \tag{6}$$

ANNs as Commutative Diagrams

This generalization is nice from a creative standpoint. We make new "classifiers" as we like.

Examples:

- A three-layer neural network is just
$$\mathcal{N}_3 : \mathbb{R}^{10000} \xrightarrow{g \circ \mathfrak{n}} \mathbb{R}^{30} \xrightarrow{g \circ \mathfrak{n}} \mathbb{R}^3.$$
- A three-layer operator network is simply
$$\mathcal{O}_3 : L^p(R) \xrightarrow{g \circ \mathfrak{o}} L^1(R) \xrightarrow{g \circ \mathfrak{o}} C(R).$$
- We can even classify functions!
$$\mathcal{C} : L^p(X) \xrightarrow{g \circ \mathfrak{o}} L^1(X) \xrightarrow{g \circ \mathfrak{o}} L^1(X) \xrightarrow{g \circ \mathfrak{n}_2} \mathbb{R}^n.$$

Results

Theorem 1. (Inclusion) It follows that

$$\{\mathcal{N}\} \subset \{\mathcal{O}\} \subset \{\mathcal{G}\}.$$

Inclusion is the first and most important result to this generalization. For every $\mathcal{N}$ there exists an $\mathcal{O}$ such that $\mathcal{O} \simeq \mathcal{N}$.

Theorem 2. (Universality) Let $F : A \to B$ be a continuous operator between Banach spaces. For every $\epsilon > 0$, there exist a GANN

$$\mathcal{G}_2 : A \xrightarrow{g \circ T_1} C \xrightarrow{g \circ T_2} B$$

such that for all ξ

$$\|C(\xi) - F(\xi)\| < \epsilon.$$

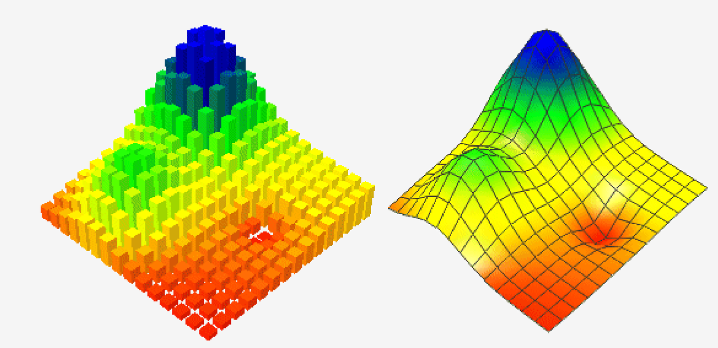

Figure 2: Parameter reduction using weight polynomials.

Theorem 3. (Parameter Reduction) Let $\mathcal{C}$ be a continuous classifier

$$\mathcal{C} : L^p(X) \xrightarrow{g \circ \mathfrak{o}} L^q(Y) \xrightarrow{g \circ \mathfrak{n}_2} \mathbb{R}^n.$$

with $O(1)$ weight polynomials. If a continuous function, say $f(t)$ is sampled uniformly from $t = 0$, to $t = N$, such that $x_n = f(n)$, then there exists a unique $\mathcal{N} \simeq \mathcal{C}$ with $O(N^2)$ weights.

Generalized Backpropagation. If $\mathcal{G}$ is parameterized by $W^l \in \mathbb{R}^{n \times m}$ then

$$B \otimes A \ni \frac{\partial \mathcal{G}}{\partial W^l} = \underbrace{\left[\bigcirc_{k=L}^{l} Dg \circ T_k\right]}_{\delta_{l+1} \text{ from BP}} \circ Dg \circ D\pi_l.$$

References

[1] Burch, Carl (2012)
A survey of machine learning
International Conference on Artificial Intelligence and Statistics

[2] Roux, Nicolas L and Bengio, Yoshua (2007)
Continuous Neural Networks
Journal of Machine Learning Research