# Classical Results in Approximation Theory

Aravinth Krishnan

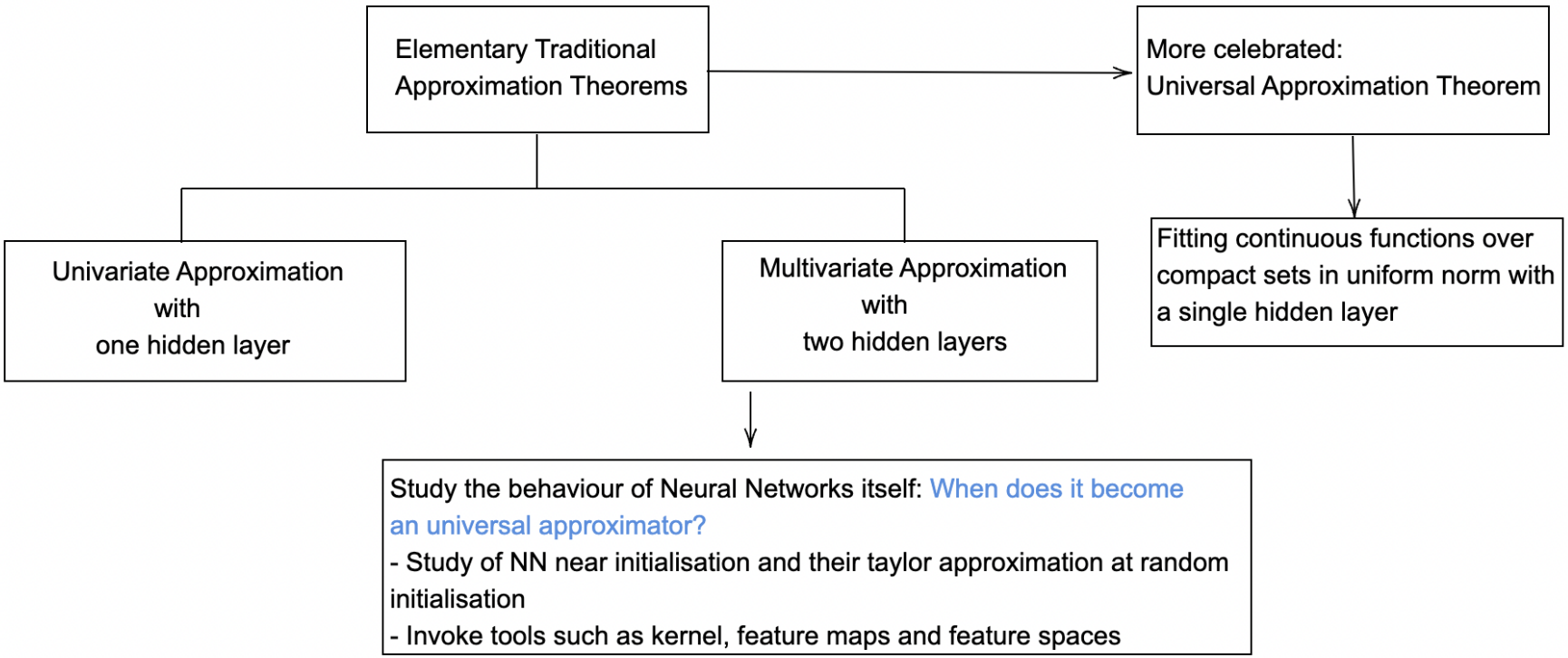
Kansas State University

**QEII Minor Exam**

Dec 2024

# Goal of this Presentation

Provide a roadmap of different classical and modern theorems in the Approximation theory of Neural Networks

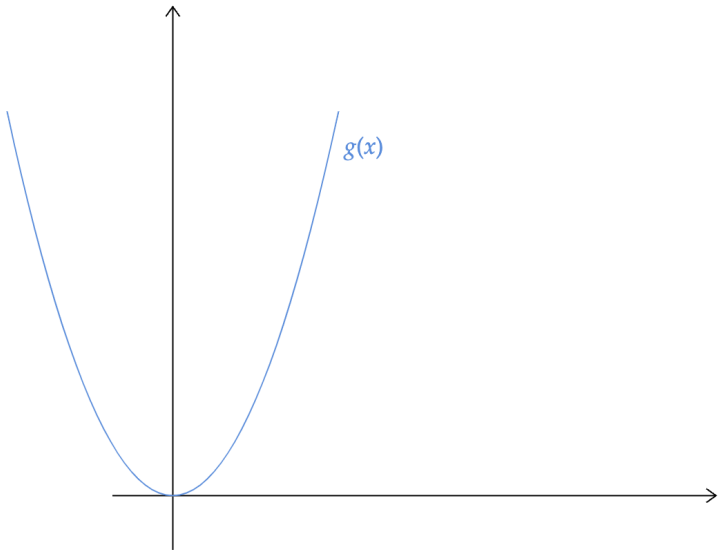


# Approximation of univariate real-valued functions with neural networks

## Theorem 1

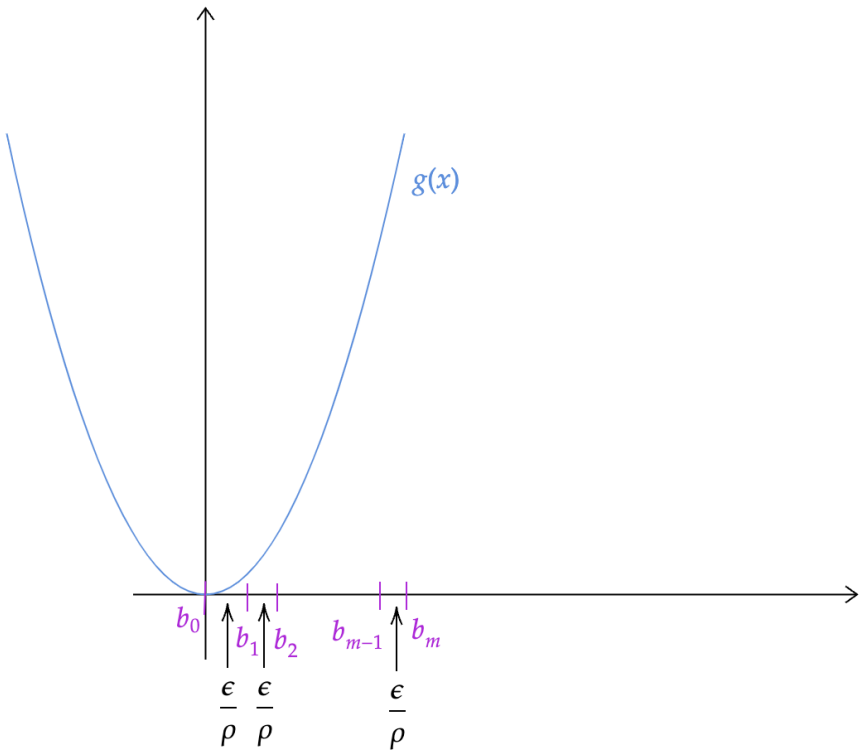
*Suppose* $g : \mathbb{R} \to \mathbb{R}$ *is* $\rho$*-Lipschitz. For any* $\epsilon > 0$*, there exists a 2 layer network* $f$ *with* $\lceil \frac{\rho}{\epsilon} \rceil$ *threshold nodes* $z \mapsto \mathbf{1}_{[z \geq 0]}$ *such that*

$$\sup_{x \in [0,1]} |f(x) - g(x)| \leq \epsilon.$$



# Proof

- Discretise the $x$-axis interval $[0, 1]$ using the step size $\frac{\epsilon}{\rho}$
- Let $m$ be the number of subintervals in $[0, 1]$. So, $m := \lceil \frac{\rho}{\epsilon} \rceil$
- Let $b_i := \frac{i\epsilon}{\rho}$. So, the interval $[0, 1]$ is partitioned by $P = \{b_0, b_1, b_2, ..., b_{m-1}\}$ for $i \in \{0, 1, 2, ..., m-1\}$.



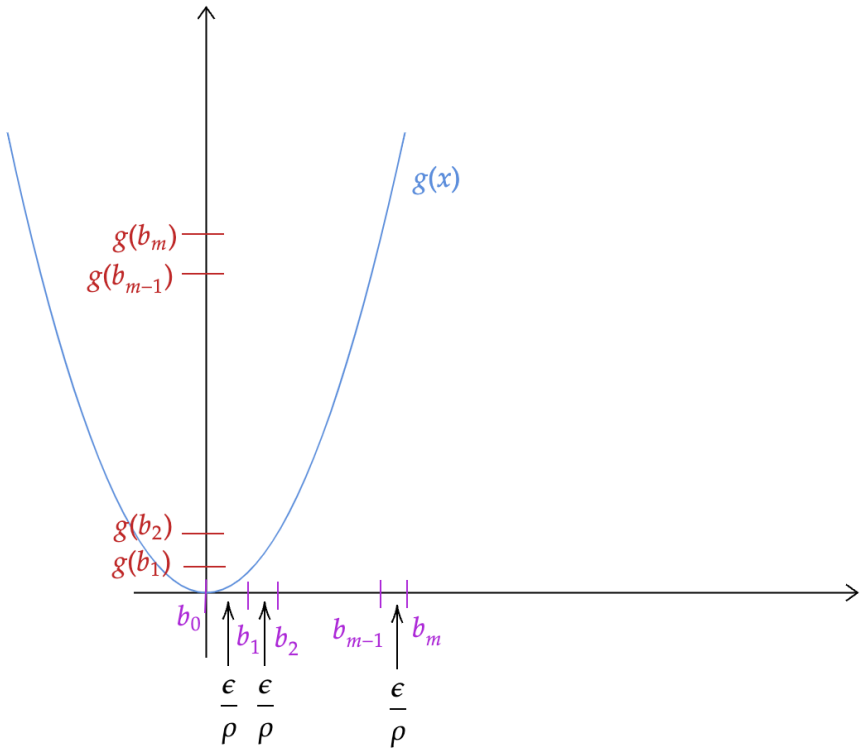
# Proof

Define:

$$a_0 := g(0),$$

and

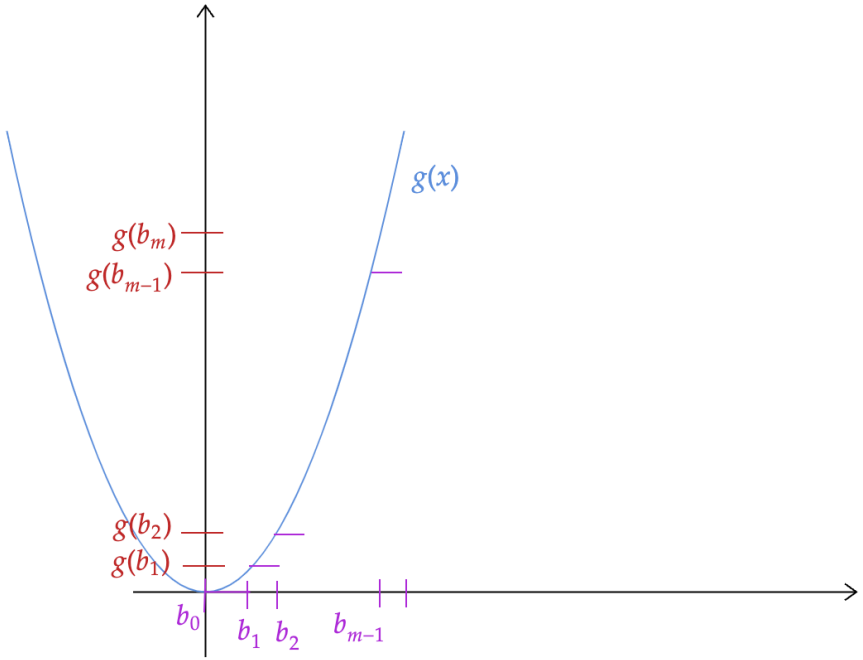
$$a_i := g(b_i) - g(b_{i-1}).$$



# Proof

We define $f$ as follows

$$f(x) := \sum_{i=0}^{m-1} a_i \mathbf{1}_{x \geq b_i}$$



# Proof

We shall prove the following:

- satisfies the condition

$$\sup_{x \in [0,1]} |f(x) - g(x)| \leq \epsilon,$$

- $f(x)$ can be represented as a 2 layer network with $\lceil \frac{\rho}{\epsilon} \rceil$ threshold nodes.

# Proof

$$
\begin{aligned}
|g(x) - f(x)| &= |g(x) - g(b_k) + g(b_k) - f(b_k) + f(b_k) - f(x)| \\
&\leq |g(x) - g(b_k)| + |g(b_k) - f(b_k)| + |f(b_k) - f(x)| \\
&= \rho|x - b_k| + |g(b_k) - \sum_{i=0}^{k} a_i| + 0 \\
&\leq \rho(\frac{\epsilon}{\rho}) + |g(b_k) - g(b_0) - \sum_{i=1}^{k}(g(b_i) - g(b_{i-1}))| \\
&= \epsilon.
\end{aligned}
$$

Hence, we have showed that $f$ satisfies the condition

$$\sup_{x \in [0,1]} |f(x) - g(x)| \leq \epsilon.$$

# f as a Neural Network

$\mathbf{1}_{x \geq b} = H(x - b)$, where $H(x)$ denotes the Heaviside activation function:

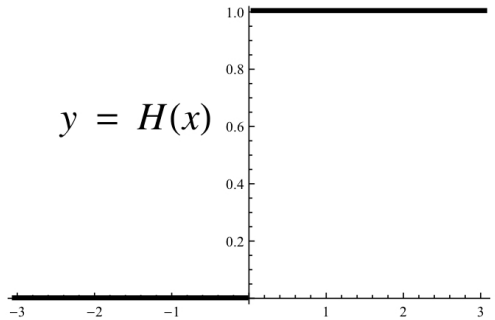


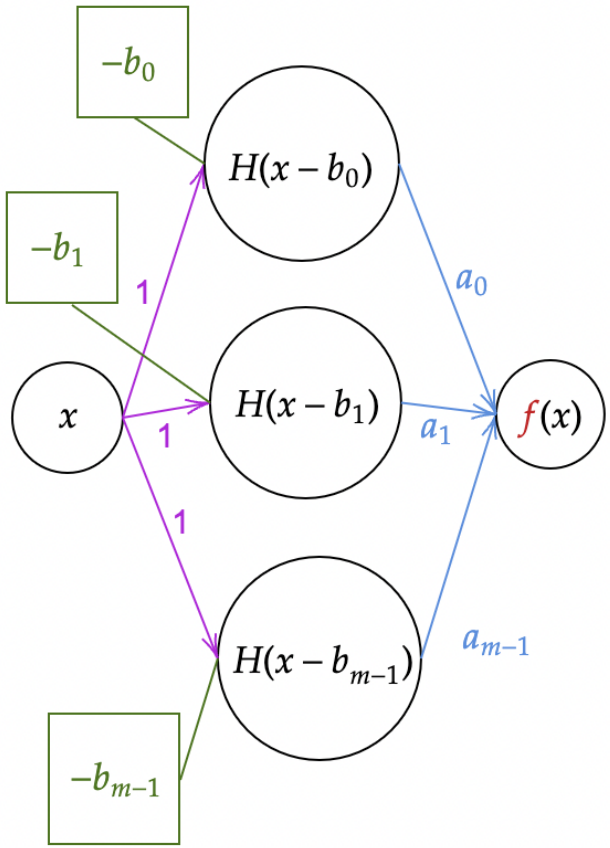

Figure 1: Heaviside Function

So,

$$f(x) := \sum_{i=0}^{m-1} a_i \mathbf{1}_{x \geq b_i}$$

$$= \sum_{i=0}^{m-1} a_i H(x - b_i)$$

# Visual Representation of $f$ as a Neural Network




We can see that there are $m$ neurons in the hidden layer. Thus, the depth of the network is $m = \lceil \frac{\rho}{\epsilon} \rceil$.

# Building a Step function for the Multivariate Case

## Theorem 2

*Let $g : \mathbb{R}^d \to \mathbb{R}$ be a continuous function and an $\epsilon > 0$ be given, and choose $\delta > 0$ so that $||x - x^{'}||_\infty \leq \delta$ implies $|g(x) - g(x^{'})| \leq \epsilon$. Let any set $U \subset \mathbb{R}^d$ be given, along with a partition $P$ of $U$ into rectangles (product of intervals) $P = (R_1, R_2, ..., R_N)$ with all sides lengths not exceeding $\delta$. Then, there exist scalars $(\alpha_1, ..., \alpha_N)$ such that*

$$\sup_{x \in U} |g(x) - h(x)| \leq \epsilon,$$

*where $h(x) = \sum_{i=1}^{N} \alpha_i \mathbf{1}_{R_i}(x)$.*

# Intuition

For each $R_i$ in the partition $P$, pick an arbitrary $x_i \in R_i$ and set $\alpha_i := g(x_i)$. Then,

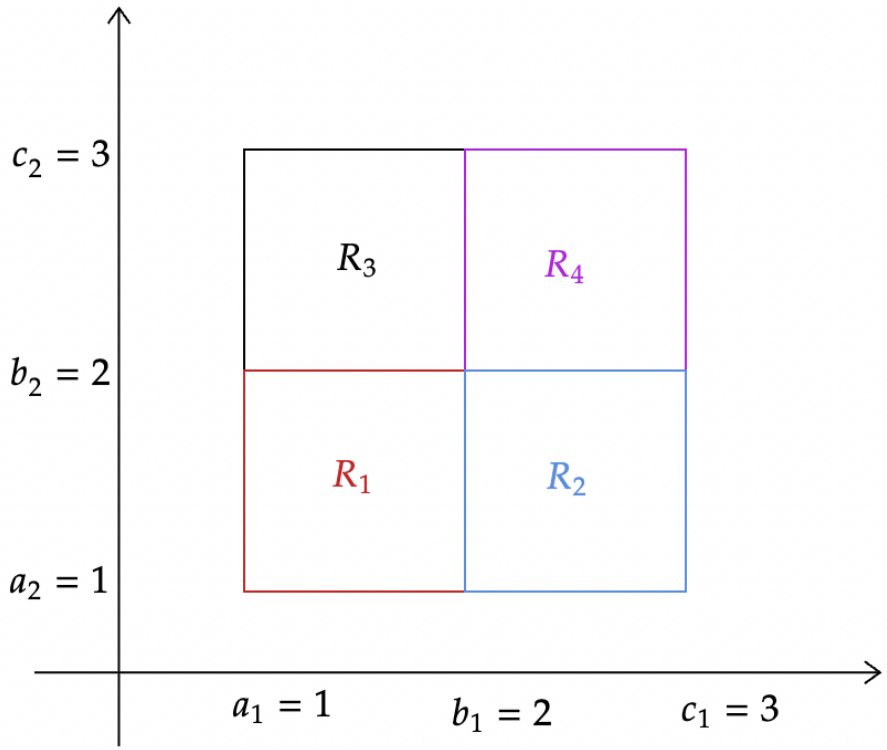
$$h(x) = \sum_{i=1}^{N} \alpha_i \mathbf{1}_{R_i}(x) = \sum_{i=1}^{N} g(x_i) \mathbf{1}_{R_i}(x)$$

Now, we have to show that the function $h$ constructed from the set of $\alpha_i$s arbitrarily picked satisfies the condition:

$$\sup_{x \in U} |g(x) - h(x)| \leq \epsilon.$$

# Proof

$$\sup_{x \in U} |g(x) - h(x)| = \sup_{i \in \{1,..,N\}} \sup_{x \in R_i} |g(x) - h(x)|$$



# Proof

Thus, we have:

$$\begin{aligned}
\sup_{x\in U} |g(x) - h(x)| &= \sup_{i\in\{1,..,N\}} \sup_{x\in R_i} |g(x) - h(x)| \\
&= \sup_{i\in\{1,..,N\}} \sup_{x\in R_i} |g(x) - g(x_i) + g(x_i) - h(x)| \\
&\leq \sup_{i\in\{1,..,N\}} \sup_{x\in R_i} (|g(x) - g(x_i)| + |g(x_i) - h(x)|) \\
&\leq \sup_{i\in\{1,..,N\}} \sup_{x\in R_i} (\epsilon + |g(x_i) - \alpha_i|) \\
&= \epsilon.
\end{aligned}$$

# Theorem

## Theorem 3

*Let $g : \mathbb{R}^d \to \mathbb{R}$ be a continuous function and an $\epsilon > 0$ be given, and choose $\delta > 0$ so that $||x - x^{'}||_\infty \leq \delta$ implies $|g(x) - g(x^{'})| \leq \epsilon$. Then, there exists a 3-layered network $f$ with $\Omega(\frac{1}{\delta^d})$ ReLU with*

$$\int_{[0,1]^d} |f(x) - g(x)| dx \leq 2\epsilon.$$

# Proof

Let $P$ denote a partition of $[0,2)^d$ into rectangles of the form $\Pi_{j=1}^{d}[a_j, b_j)$, with $b_j - a_j \leq \delta$. The final result will work when we restrict the considerations to $[0,1]^d$, but we include an extra regions to work with half-open intervals in a lazy way.
From theorem 2.2, there exist scalars $(\alpha_1, ..., \alpha_N)$ so that

$$\sup_{x \in U} |g(x) - h(x)| \leq \epsilon,$$

where $h = \sum_{i=1}^{N} \alpha_i \mathbf{1}_{R_i}$.

# Proof

Our final constructed network $f$ will be of the form:

$$f(x) := \sum_i \alpha_i g_i(x),$$

where each $g_i$ will be a ReLU Network with 2 hidden layers and $\mathcal{O}(d)$ neurons. Our goal is to show $\int_{[0,1]^d} |f(x) - g(x)| dx \leq 2\epsilon$. That is to say:

$$||f - g||_1 \leq 2\epsilon$$

# Proof

To this end, note that:

$$\begin{aligned} ||f - g||_1 &= ||f - h + h - g||_1 \\ &\leq ||f - h||_1 + ||h - g||_1 \\ &= ||\sum_i \alpha_i(\mathbf{1}_{R_i} - g_i)||_1 + \epsilon \\ &\leq \sum_i |\alpha_i| \cdot ||\mathbf{1}_{R_i} - g_i||_1 + \epsilon \end{aligned}$$

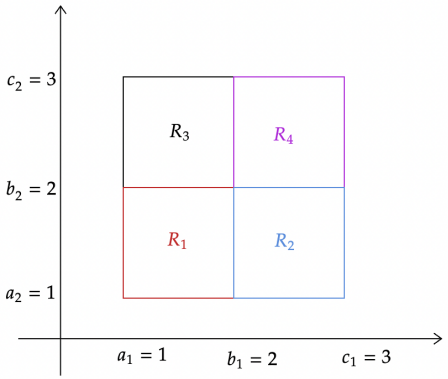
Then, we need to construct each $g_i$ such that $||\mathbf{1}_{R_i} - g_i||_1 \leq \frac{\epsilon}{\sum_i \alpha_i}$.

# Proof

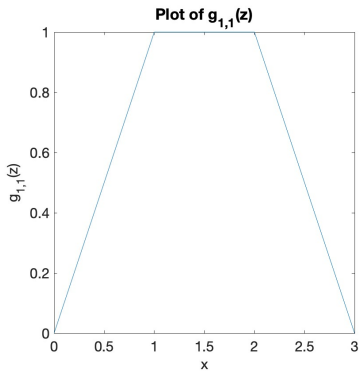
Fix the rectangle $R_i$ selected from the partition $P$. Then, Let $R_i := [a_1, b_1) \times [a_2, b_2) \times ... \times [a_d, b_d)$.
Set $\gamma > 0$ to be a hyperparameter. For each $j \in \{1, 2, 3, ..., d\}$,

$$g_{\gamma,j}(z) = \sigma(\frac{z - (a_j - \gamma)}{\gamma}) - \sigma(\frac{z - a_j}{\gamma}) - \sigma(\frac{z - b_j}{\gamma}) + \sigma(\frac{z - (b_j + \gamma)}{\gamma}),$$

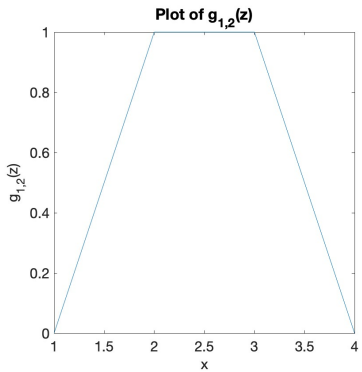
$$= \begin{cases} 1, & \text{if } z \in [a_j, b_j) \\ 0, & \text{if } z \notin [a_j - \gamma, b_j + \gamma) \\ [0, 1], & \text{otherwise} \end{cases}$$




(a) Partition P of U




(b) Plot of $g(1, 1)(z)$




(c) Plot of $g(1, 2)(z)$

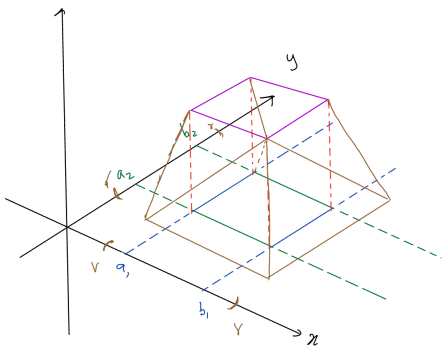
# Proof

Then, we define $g_i$ as:

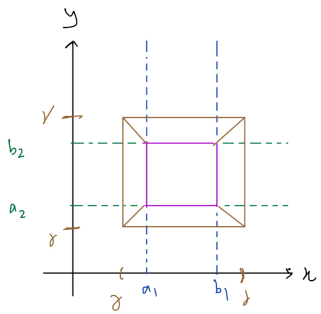
$$g_\gamma = \sigma\left(\sum_j g_{\gamma,j}(x_j) - (d-1)\right)$$

Note that

$$\mathbf{1}_{R_i}(x) \approx g_i(x) = \begin{cases} 1, & \text{if } x \in [a_1, b_1) \times [a_2, b_2) \times \ldots \times [a_d, b_d) \\ 0, & \text{if } x \notin [a_1 - \gamma, b_1 + \gamma) \times \ldots \times [a_d - \gamma, b_d + \gamma) \\ [0,1], & \text{otherwise} \end{cases}$$



(a) $g_\gamma(3D)$

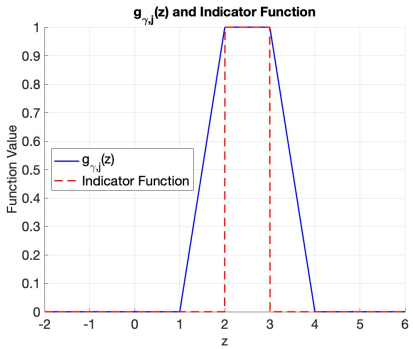


(b) $g_\gamma(2D)$

# Proof

$$
\begin{aligned}
&||1_{R_i} - g_i||_1 \\
=&\int_{[0,2)^d} |\mathbf{1}_{R_i} - g_i| dx \\
=&\int_{R_i} |\mathbf{1}_{R_i} - g_i| dx + \int_{B \setminus R_i} |\mathbf{1}_{R_i} - g_i| dx + \int_{[0,2)^d \setminus B} |\mathbf{1}_{R_i} - g_i| dx \\
\leq& 0 + \Pi_{j=1}^{d}(b_j - a_j + 2\gamma) + \Pi_{j=1}^{d}(b_j - a_j) + 0 \\
=&\mathcal{O}(\gamma)
\end{aligned}
$$

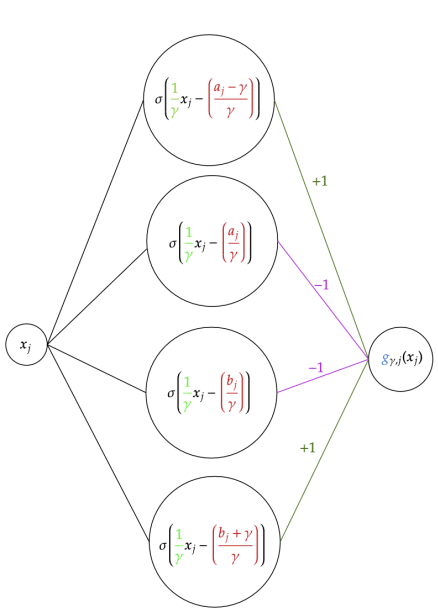
where $B = [a_1 - \gamma, b_1 + \gamma) \times ... \times [a_d - \gamma, b_d + \gamma)$.



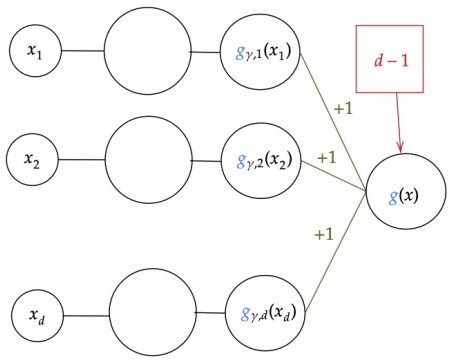
# Proof

This means we can ensure $||1_{R_i} - g_i||_1 \leq \frac{\epsilon}{\sum_i \alpha_i}$ by choosing sufficiently small $\gamma$, thus completing the proof.

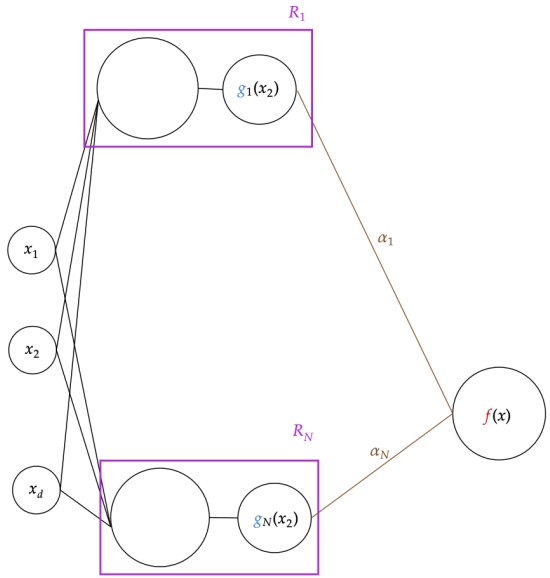
# Visualisation




(a) $g_{\gamma,j}(x_j)$




(b) $g(x)$




(c) $f(x)$

# Weakness of Previous Proof

The theorem above has 2 weakness:

- 2 Hidden layers are used in the neural network
- A specific activation function is used to approximate $g$

# Improvements on the previous theorem

In the previous theorem, we used 2 hidden layers to construct $g_\gamma$. In constructing $f$, we had to approximate

$$x \mapsto \mathbf{1}_{R_i}(x) = \mathbf{1}_{[a_1,b_1]\times\ldots\times[a_d,b_d]}(x).$$

If we had a way to approximate multiplication, we could instead approximate

$$x \mapsto \mathbf{1}_{[a_1,b_1]}(x) \times \mathbf{1}_{[a_2,b_2]}(x) \times \ldots \times \mathbf{1}_{[a_d,b_d]}(x).$$

# Introducing Universal Approximators

Can we approximate multiplication and then form a linear combination, all with just one hidden layer?

YES!

# Definition of Universal Approximators

## Definition 4 (Universal Approximators)

A class of functions $\mathcal{F}$ is an Universal Approximator over a compact set $S$ if for every continuous function $g$ and a target accuracy $\epsilon > 0$, there exists $f \in \mathcal{F}$ with

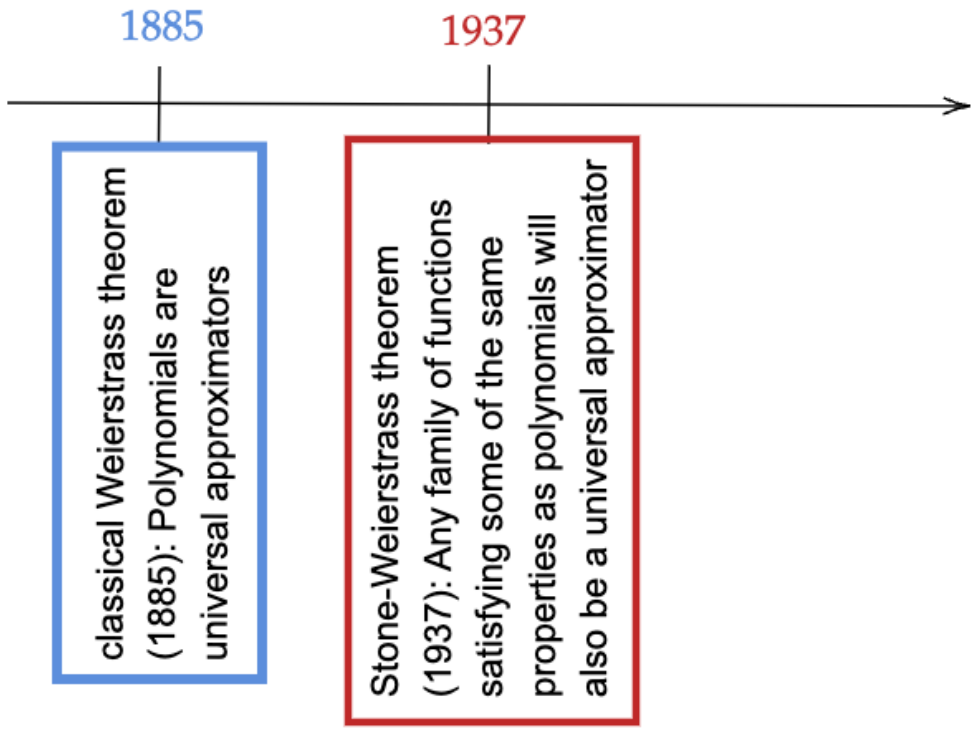
$$\sup_{x \in S} |f(x) - g(x)| \leq \epsilon.$$

Notes:

- Compactness is necessary ($sin(x)$)
- Can be more succinctly written as some class being dense in all continuous functions over compact sets.

How do we know if $\mathcal{F}$ is an universal approximator?

# Basis of Universal Approximation Theorem




The Stone-Weierstrass theorem serves as a good tool to show if some $\mathcal{F}$ is a universal approximator.

# Stone-Weierstrass Theorem (Folland 1999, Theorem 4.45)

## Theorem 5 (Stone-Weierstrass)

*Let $\mathcal{F}$ denote a class of functions and $f \in \mathcal{F}$ be given as follows:*

1. *Each $f \in \mathcal{F}$ is continuous*
2. *For every $x \in X$, there exists $f \in \mathcal{F}$ with $f(x) \neq 0$*
3. *For every $x \neq x'$, there exists $f \in F$ with $f(x) \neq f(x')$ (That is to say $F$ separates points)*
4. *$\mathcal{F}$ is closed under multiplication and vector space operations ($\mathcal{F}$ is an algebra)*

*Then, $\mathcal{F}$ is an universal approximator: For every continuous $g : \mathbb{R}^d \to \mathbb{R}$ and $\epsilon > 0$, there exists $f \in \mathcal{F}$ with $\sup_{x \in [0,1]^d} |f(x) - g(x)| \leq \epsilon$.*

# Representation of Universal Approximators

Let

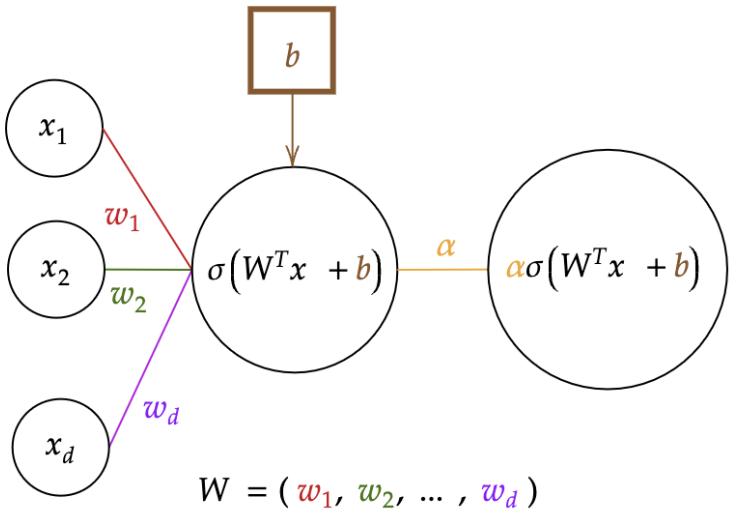
- $\sigma \rightarrow$ Activation Function
- $d \rightarrow$ Input Dimension
- $m \rightarrow$ Depth of Neural Network

Then, $\mathcal{F}_{\sigma,d,m}$ and $\mathcal{F}_{\sigma,d}$ be defined as follows:

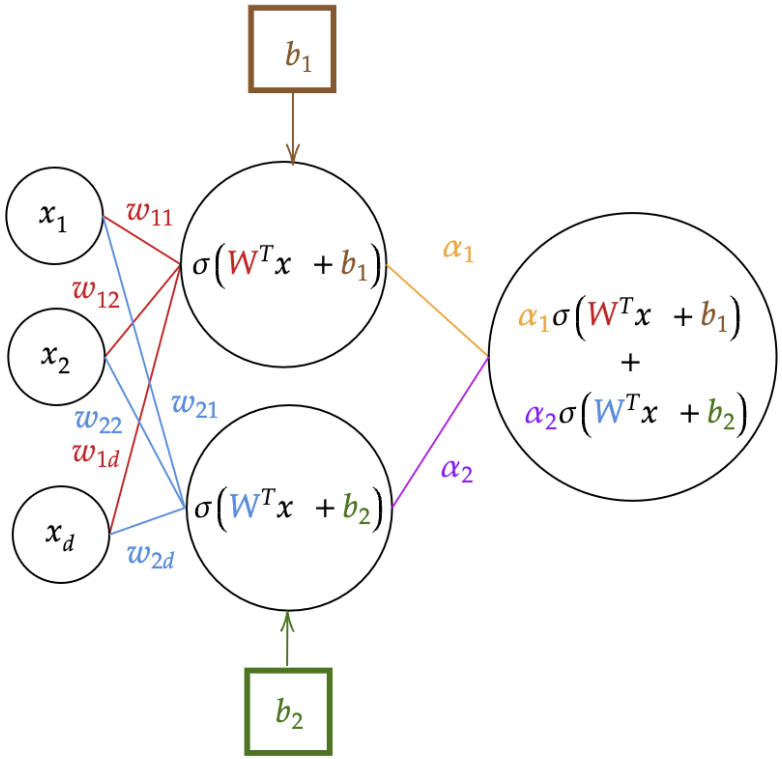
$$\mathcal{F}_{\sigma,d,m} := \mathcal{F}_{d,m} := \{x \mapsto a^T \sigma(Wx + b) : a \in R^m, W \in \mathbb{R}^{m \times d}, b \in \mathbb{R}^m\}$$

$$\mathcal{F}_{\sigma,d} := \mathcal{F}_d := \cup_{m \geq 0} \mathcal{F}_{\sigma,d,m}$$

# Visualising $\mathcal{F}_{\sigma,d,1}$ and $\mathcal{F}_{\sigma,d,2}$




(a) $\mathcal{F}_{\sigma,d,1}$




(b) $\mathcal{F}_{\sigma,d,2}$

# Examples of Universal Approximators

- Example 1: $\mathcal{F}_{cos,d}$ is an universal approximator
- Example 2: $\mathcal{F}_{exp,d}$ is an universal approximator

# Approximation near initialization and the Neural Tangent Kernel

Now, we will consider networks close to their random initialisation. The core idea is to compare a network:

$$f : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}$$
$$(x, W) \mapsto f_W(x)$$

to its first order Taylor approximation at a random initialization $W_0$:

$$f_0(x; W) := f(x; W_0) + \langle \nabla_W f(x; W_0), W - W_0 \rangle.$$

# Goals

The goal of this subsection is to:

- We will show that near initialisation, with large width, $f \approx f_0$ ($f$ is effectively linear near initialisation)
- Show these neural networks near initialisation are already universal approximators

# The Shallow Case

This is our shallow neural network:

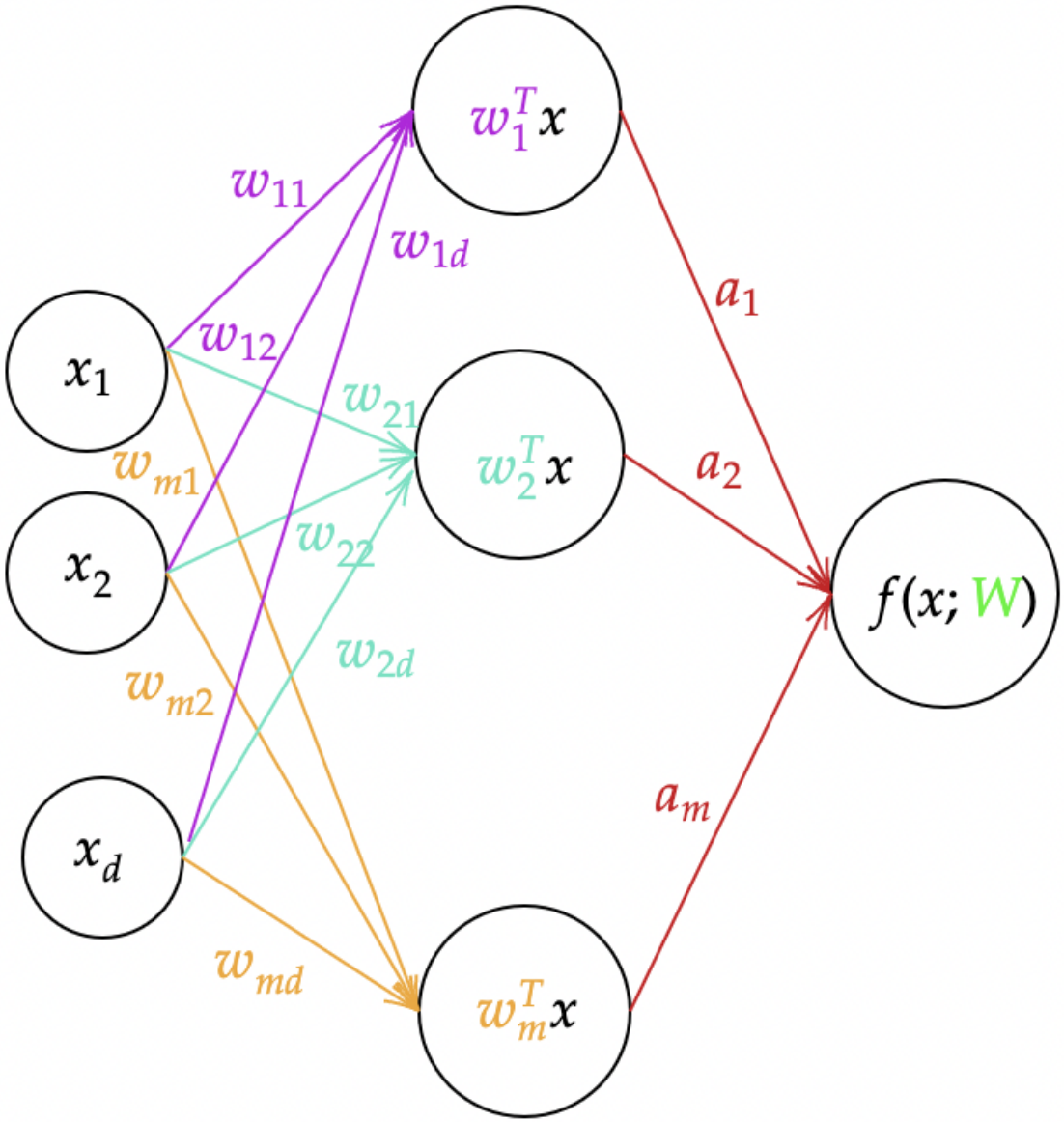
$$f(x; W) := \frac{1}{\sqrt{m}} \sum_{j=1}^{m} a_j \sigma(w_j^T x)$$
$$= \frac{1}{\sqrt{m}} \left( a_1 \sigma(w_1^T x) + a_2 \sigma(w_2^T x) + \ldots + a_m \sigma(w_m^T x) \right)$$

where

$$W := \begin{pmatrix} \leftarrow & w_1^T & \rightarrow \\ \leftarrow & w_2^T & \rightarrow \\ & \cdot & \\ & \cdot & \\ \leftarrow & w_m^T & \rightarrow \end{pmatrix} \in \mathbb{R}^{m \times d},$$

where $\sigma$ will either be a smooth activation or the ReLU, and we will treat $a \in \mathbb{R}^m$ as fixed and only allow $W \in \mathbb{R}^{m \times d}$ to vary.

# Visualisation



# The first order Taylor Approximation at initialisation

Assume $\sigma$ is any univariate activation which is differentiable except on a set of measure 0, and let $W_0$ be the Gaussian initialisation. Then, the first order Taylor Approximation at $W = W_0$ is:

$$
\begin{aligned}
f_0(x; W) &= f(x; W_0) + \langle \nabla_W f(x; W_0), W - W_0 \rangle \\
&= \frac{1}{\sqrt{m}} \sum_{j=1}^{m} a_j (\sigma(w_{0,j}^T x) + \sigma'(w_{0,j} x^T)(w_j - w_{0,j})) \\
&= \frac{1}{\sqrt{m}} \sum_{j=1}^{m} a_j ([\sigma(w_{0,j}^T x) - \sigma'(w_{0,j}) w_{0,j}^T x] + \sigma'(w_{0,j}) w_j^T x).
\end{aligned}
$$

# Theorem

Now, we will see that $f - f_0 \to 0$ as $m \to \infty$.

## Theorem 6

*If $\sigma : \mathbb{R} \to \mathbb{R}$ is $\beta$-smooth and $|a_j| \leq 1$, and $||x||_2 \leq 1$, then for any parameters $W, V \in \mathbb{R}^{m \times d}$,*

$$|f(x; W) - f_0(x; V)| \leq \frac{\beta}{2\sqrt{m}} ||W - V||_F^2.$$

Set $V = W_0$. Small $||W - W_0||$ means that the weight $W$ is close to the initialisation weights $W_0$. Then, the theorem tells us that as $m \to \infty$, our neural network $f$ at weight $W$ gets closer and closer to the Taylor approximation of our neural network initialised at weight $W_0$.

# Proof

$$|f(x;W) - f_0(x;V)| = |f(x;W) - f(x;V_0) + < \nabla_W f(x;V), W - V >$$

$$\leq \frac{1}{\sqrt{m}} \sum_{j=1}^{m} |a_j| \cdot |\sigma(w_j^T x) - \sigma(v_j^T x)$$

$$- \sigma'(v_j^T x) x^T (w_j - v_j)|$$

$$\leq \frac{1}{\sqrt{m}} \sum_{j=1}^{m} \frac{\beta (w_j^T x - v_j^T x)^2}{2}$$

$$\leq \frac{\beta}{2\sqrt{m}} \sum_{j=1}^{m} ||w_j - v_j||^2$$

$$= \frac{\beta}{2\sqrt{m}} ||W - V||_F^2$$

# Next Goal

So far, we have said that $f - f_0$ is small when the width is large.

QUESTION: We know that neural networks are universal approximators. But when does it start having this property?

We will show that when the width is large, neural networks close to initialisation, $f$, are already universal approximators:

- We saw that $f$ is approximately equal to some linear space, $f_0$, which is can be seen as a feature space
- This allows us to consider the kernel corresponding to said feature space and these allows us to bring in new tools to establish our claim above.

# Key Definitions

## Definition 7 (Kernel, Feature Map and Feature Space)

Let $X$ be a non-empty set. Then, a function $k : X \times X \to \mathbb{R}$ is called a **kernel** on $X$ if there exists a $\mathbb{R}$-Hilbert Space $\mathcal{H}$ and a map $\Phi : X \to \mathcal{H}$ such that for all $x, x^{'} \in X$, we have

$$k(x, x^{'}) = \left\langle \Phi(x), \Phi(x^{'}) \right\rangle .$$

We call $\Phi$ a **feature map** and $\mathcal{H}$ a **feature space** of $k$.

# Feature Map (Neural Network Setting)

$\nabla f(\cdot; W_0) : x \mapsto \nabla f(x; W_0)$ defines a feature mapping:

$$\nabla f(x; W_0) = \begin{pmatrix} \leftarrow a_1 \sigma'(w_{0,1}^T x) x^T \rightarrow \\ \cdot \\ \cdot \\ \cdot \\ \leftarrow a_m \sigma'(w_{0,m}^T x) x^T \rightarrow \end{pmatrix}.$$

Note that $x \in \mathbb{R}^d$ and $f(x; W_0) \in \mathbb{R}^{m \times d} \cong \mathbb{R}^{md}$ $(d << md)$

# Kernel (Neural Network Setting)

$$k_m(x, x') := \langle \nabla_W f(x, W_0), \nabla_W f(y, W_0) \rangle$$

$$= \left\langle \begin{pmatrix} a_1 x^T \sigma'(w_{1,0}^T x)/\sqrt{m} \\ \cdot \\ \cdot \\ \cdot \\ a_m x^T \sigma'(w_{m,0}^T x)/\sqrt{m} \end{pmatrix}, \begin{pmatrix} a_1 y^T \sigma'(w_{1,0}^T y)/\sqrt{m} \\ \cdot \\ \cdot \\ \cdot \\ a_m y^T \sigma'(w_{m,0}^T y)/\sqrt{m} \end{pmatrix} \right\rangle$$

$$= \frac{1}{m} \sum_{j=1}^{m} a_j^2 \left\langle x\sigma'(w_{j,0}^T x), y\sigma'(w_{j,0}^T y) \right\rangle$$

$$= x^T y \left[ \frac{1}{m} \sum_{j=1}^{m} \sigma'(w_{j,0}^T x)\sigma'(w_{j,0}^T y) \right] \in \mathbb{R}$$

Justification for $\frac{1}{\sqrt{m}}$: Kernel is now an average, not a sum. We can expect a limit as $m \to \infty$.

# Theorem

TASK: Show functions near initialisation are universal approximators.
Define $\mathcal{H}$ as follows:

$$\mathcal{X} := \left\{ x \in \mathbb{R}^d : ||x|| = 1, x_d = \frac{1}{\sqrt{2}} \right\}$$

$$\mathcal{H} := \left\{ x \mapsto \sum_{j=1}^{m} \alpha_j k(x, x_j) : m \geq 0, \alpha_j \in \mathbb{R}, x_j \in \mathcal{X} \right\}$$

$\mathcal{H}$ is nothing more than the set of infinite width neural networks near its initialization, each infinite width neural network represented as a linear combination of kernels. (Showing why that's the case it beyond the scope of the minor.)

# Theorem

## Theorem 8

*$\mathcal{H}$ is a universal approximator over $\mathcal{X}$; that is to say, for every continuous $g : \mathbb{R}^d \to R$ and every $\epsilon > 0$, there exists a $f \in \mathcal{H}$ with* $\sup_{x \in \mathcal{X}} |g(x) - f(x)| \leq \epsilon$.

# Proof

Let $U := \{u \in \mathbb{R}^{d-1} : ||u||^2 \leq \frac{1}{2}\}$, and $k$ be the kernel function as defined below:

$$k(u, u') := f(u^T u')$$

$$f(z) := \frac{z + \frac{1}{2}}{2} - \frac{(z + \frac{1}{2})arccos(z + \frac{1}{2})}{2\pi}.$$

We shall show that $k$ is an universal approximator over $U$.

# Proof

Note that *arccos* has the maclaurin series

$$arccos(z) = \frac{\pi}{2} - \sum_{k \geq 0} \frac{(2k)!}{2^{2k}(k!)^2} \frac{z^{2k+1}}{2k+1},$$

which is convergent over $z \in [-1, 1]$. Note every term is positive (adding the bias term ensured this).

# Proof

Using the following collary,

## Theorem 9 (Universal Taylor Kernels)

*Fix an* $r \in (0, \infty]$ *and a* $C^\infty$ *function* $f : (-r, r) \to \mathbb{R}$ *that can be expanded into its taylor series at* 0,

$$f(t) = \sum_{n=0}^{\infty} a_n t^n, t \in (-r, r).$$

*Let* $\mathcal{X} := \{x \in \mathbb{R}^d : ||x||_2 < \sqrt{r}\}$. *If we have* $a_n > 0$ *for all* $n \geq 0$, *then* $k$ *given by:*

$$k(x, x^{'}) := f(< x, x^{'} >)$$

*is a universal kernel on every compact subset of* $\mathcal{X}$.

we can see that $k$ is an universal approximator on $U$.

# Proof

Since $k$ is an universal approximator on $U$, $k$ is also an universal approximator on $\partial U$ and thus, the kernel is an universal approximator over $\mathcal{X}$.

# The end