

Article

Sinusoidal Approximation Theorem for Kolmogorov–Arnold Networks

Sergei Gleyzer¹, Hanh Nguyen² , Dinesh P. Ramakrishnan¹  and Eric A. F. Reinhardt^{1,*} ¹ Department of Physics and Astronomy, University of Alabama, Tuscaloosa, AL 35405, USA² Department of Mathematics, University of Alabama, Tuscaloosa, AL 35405, USA

* Correspondence: eareinhardt@crimson.ua.edu

Abstract

The Kolmogorov–Arnold representation theorem states that any continuous multivariable function can be exactly represented as a finite superposition of continuous single-variable functions. Subsequent simplifications of this representation involve expressing these functions as parameterized sums of a smaller number of unique monotonic functions. Kolmogorov–Arnold Networks (KANs) have been recently proposed as an alternative to multilayer perceptrons. KANs feature learnable nonlinear activations applied directly to input values, modeled as weighted sums of basis spline functions. This approach replaces the linear transformations and sigmoidal post-activations used in traditional perceptrons. In this work, we propose a novel KAN variant by replacing both the inner and outer functions in the Kolmogorov–Arnold representation with weighted sinusoidal functions of learnable frequencies. We particularly fix the phases of the sinusoidal activations to linearly spaced constant values and provide a proof of their theoretical validity. We also conduct numerical experiments to evaluate its performance on a range of multivariable functions, comparing it with fixed-frequency Fourier transform methods, basis spline KANs (B-SplineKANs), and multilayer perceptrons (MLPs). We show that it outperforms the fixed-frequency Fourier transform B-SplineKAN and achieves comparable performance to MLP.

Keywords: approximation; machine learning; representation; periodic functions**MSC:** 26-08; 41-04; 41A58; 42A10

Academic Editor: Xiangmin Jiao

Received: 6 August 2025

Revised: 2 September 2025

Accepted: 19 September 2025

Published: 2 October 2025

Citation: Gleyzer, S.; Nguyen, H.; Ramakrishnan, D.P.; Reinhardt, E.A.F. Sinusoidal Approximation Theorem for Kolmogorov–Arnold Networks. *Mathematics* **2025**, *13*, 3157. <https://doi.org/10.3390/math13193157>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The quest to represent complex mathematical functions in terms of simpler constituent parts is a central theme in analysis and applied mathematics. A pivotal moment in this endeavor arose from the list of 23 problems presented by David Hilbert in 1900, which profoundly shaped the mathematical landscape of the 20th century. Hilbert’s 13th problem, in particular, posed a fundamental question about the limits of function composition. The problem seeks solutions to a general 7th-degree polynomial, which are finite combinations of functions of a single variable each, i.e., if $f(x_1, \dots, x_7) = g(f_1(x_1), \dots, f_7(x_7))$, where g is a simple known function. [1,2].

In the late 1950s, Andrei Kolmogorov and his graduate student Vladimir Arnold provided a powerful and affirmative answer to this problem. Their work culminated in the Kolmogorov–Arnold representation theorem (KART) [3,4], which states that any multivariable continuous function f defined on a compact domain, such as the n -dimensional unit

cube, can be represented exactly by a finite superposition of single variable functions and the single binary operation of addition. The canonical form of the theorem is expressed as

$$f(x_1, \dots, x_n) = \sum_{q=0}^{2n} \phi_q \left(\sum_{p=1}^n \psi_{p,q}(x_p) \right), \quad (1)$$

where $\psi_{p,q} : \mathbb{R} \rightarrow \mathbb{R}$ are the innerfunctions, and $\phi_q : \mathbb{R} \rightarrow \mathbb{R}$ are the outer functions, which are assumed to be continuous. Lorentz [5] and Sprecher [6] showed that the functions ϕ_q can be replaced by only one function ϕ and that the functions $\psi_{p,q}$ can be replaced by $\lambda^{pq}\psi_q$, where λ is a constant and ψ_q are monotonic and Lipschitz-continuous functions.

Perceptrons were introduced around the same time by Frank Rosenblatt [7] for simple classification tasks. However, their limitations were famously highlighted in the 1969 book “*Perceptrons*” by Marvin Minsky and Seymour Papert [8]. They proved that a single-layer perceptron could not solve problems that were not linearly separable, such as the simple XOR logical function. This critique led to a significant decline in neural network research for over a decade, often termed the first “AI winter.” The revival came in the 1980s with the popularization of the multilayer perceptron (MLP), which was introduced in the late 1960s [9]. By adding one or more “hidden” layers between the input and output layers, MLPs could overcome the limitations of the single-layer perceptron. The key breakthrough that made these deeper networks practical was the backpropagation algorithm. While developed earlier, its popularization by David Rumelhart, Geoffrey Hinton, and Ronald Williams in 1986 [10] provided an efficient method to train the weights in these new, multilayered networks, sparking a renewed wave of interest in the field.

With MLPs and derived neural network architectures demonstrating practical success [11–13], the next crucial step was to understand their theoretical capabilities. The Universal Approximation Theorem provides this foundation, showing that a standard MLP with just one hidden layer can, in principle, approximate any continuous function to any desired degree of accuracy. George Cybenko [14] used tools from functional analysis, like the Hahn–Banach theorem and the Riesz representation theorem, and showed that, given any multivariable continuous function in a compact domain, there exist values of the weights and biases of a multilayer perceptron with a sigmoidal model whose loss with respect to the function is bound by any given positive value ϵ . Almost concurrently, Kurt Hornik, Maxwell Stinchcombe, and Halbert White [15] provided a different, more general proof for the universal approximation property of a multilayer perceptron with any bounded, non-constant, and continuous activation function.

The applicability of KART to continuous functions in a compact domain was first highlighted by Hecht-Nielsen [16,17]. However, the functions constructed in Kolmogorov’s proofs, as well as in their later simplifications or improvements, are highly complex and non-smooth, which were very different from the much simpler activation functions used in MLPs [18]. Soon, a series of papers by Věra Kůrková [19,20] showed that the one-dimensional inner and outer functions could be approximated using two-layer MLPs and obtained a relationship between the number of hidden units and the loss based on the properties of the function being approximated.

It was not until much later in the 2020s that KART had a direct practical application in the domain of neural networks and machine learning. Kolmogorov–Arnold networks (KANs) were first proposed by Liu et al. [21,22] as an alternative to MLPs, which demonstrated interpretable hidden activations and higher regression accuracy. They used learnable basis splines to model the constituent hidden values, similar to KART, and their stacked representation is simplified compared to the original form used in KART. Subsequent works retained the same representation but replaced splines with other series of functions, such as radial basis functions and Chebyshev polynomials [23–26], which were

computationally faster and demonstrated superior scaling properties with larger numbers of learnable parameters. However, issues of speed and numerical stability on smaller floating-point types remain, especially compared to the well-established MLPs [27,28].

To address these challenges, we propose a variant of the representation theorems by Lorentz [5] and Sprecher [6]. We set the phases of the sinusoidal activations to linearly spaced constant values and establish their mathematical foundation to confirm their validity. Previous work has explored this approximation series compared to MLPs and basis-spline approximations and showed competitive performance on the inherently discontinuous domain task of labeling hand-written numerical characters [29]. We extend this work by providing a constructive proof of the approximation series for single-variable and multivariable functions and evaluating performance on several such functions with features such as rapid and changing oscillation frequencies. We also extend the comparison from [29] to include fixed-frequency Fourier transform methods and MLP with piecewise-linear and periodic activation functions. Table 1 summarizes the advancements in this area.

Table 1. Simplifications proposed for the Kolmogorov–Arnold representation theorem at different points in time.

Version	Representation
Kolmogorov (1957)	$\sum_{q=0}^{2d} g_q \left(\sum_{p=1}^d \psi_{pq}(x_p) \right)$
Lorentz–Sprecher (1965)	$\sum_{q=0}^{2d} g \left(\sum_{p=1}^d \lambda_p \psi(x_p + qa) + c_q \right)$
KAN, B-spline basis (2024)	$y_i = \sum_{j=1}^d \sum_{k=1}^{G+p} w_{ijk} B_{k,p}(x_j) + b_i$ $\implies \sum_{i=1}^H \sum_{l=1}^{G+p} W_{il} B_l(y_i) + \beta$
KAN, sinusoidal basis (ours)	$y_q = \sum_{p=1}^d \sum_{k=0}^N A_{pq,k} \sin(\omega_{1k} x_p + \varphi_{1k})$ $\implies \sum_{q=1}^{2d+1} \sum_{j=0}^N B_{qj} \sin(\omega_{2j} y_q + \varphi_{2j})$

The remainder of this paper is organized as follows. In Section 2, we establish the necessary auxiliary results to prove the approximation theorem for one-dimensional functions. Section 3 extends the results from the previous section to derive a universal approximation theorem for two-layer neural networks. In Section 4, we test several functions and compare the numerical performance of our proposed networks with the classical Fourier transform. Finally, in Section 5, we discuss potential applications and future research directions.

2. Sinusoidal Universal Approximation Theorem

The concept of approximating any function using simple, manageable functions has been widely utilized in deep learning and neural networks. In addition to the multilayer perceptron (MLP) approach, the Kolmogorov–Arnold representation theorem (KART) has gained increasing attention from data scientists, who have developed Kolmogorov–Arnold Networks (KANs) to create more interpretable neural networks. In the classical Fourier transform, any continuous, piecewise continuously differentiable (C^1) function on a compact interval can be expressed as a Fourier series of sine and cosine functions. In

contrast, Kolmogorov demonstrated that any continuous multivariable function can be represented as form (1).

By extending Kolmogorov's idea and refining both outer and inner functions using sine terms, we can demonstrate that any continuous function in $[0, 1]$ can be approximated by a finite sum of sinusoidal functions with varying frequencies and linearly spaced phases.

Theorem 1. Let f be a continuous function on $[0, 1]$ and $0 < \alpha \leq \frac{\pi}{2}$. For any $\epsilon > 0$, there exist $N \in \mathbb{N}$, frequencies $\omega_k \in [0, 2\pi]$, and coefficients $A_k \in \mathbb{R}$ such that

$$\sup_{0 \leq x \leq 1} \left| f(x) - \sum_{k=0}^N A_k \sin\left(\omega_k x + \frac{k\alpha}{N+1}\right) \right| < \epsilon. \quad (2)$$

To prove this theorem, we need a series of lemmas to approximate the sine function with its Taylor series and then apply the Weierstrass approximation theorem to bring the sum of sinusoidal functions arbitrarily close to any continuous function.

First, we need to approximate the sinusoidal function using its Taylor polynomial.

Lemma 1. Let $\omega \in [0, 2\pi]$ and $\alpha \in \mathbb{R}$. For every $N \in \mathbb{N}$,

$$\sup_{0 \leq x \leq 1} |\sin(\omega x + \alpha) - T_N(\omega, \alpha, x)| \leq \frac{(2\pi)^{N+1}}{(N+1)!},$$

where

$$T_N(\omega, \alpha, x) = \sum_{l=0}^N \frac{\omega^l}{l!} \sin\left(\alpha + \frac{l\pi}{2}\right) x^l. \quad (3)$$

Proof. Set $g(x) = \sin(\omega x + \alpha)$. We can rewrite

$$g(x) = \sin(\omega x + \alpha) = T_N(\omega, \alpha, x) + R_N(x),$$

where $T_N(\omega, \alpha, x)$, determined by (3), is the Taylor polynomial of the function $g(x)$ centered at $x = 0$, and the remainder

$$R_N(x) = \frac{g^{(N+1)}(\xi)}{(N+1)!} x^{N+1},$$

for some $\xi \in (0, 1)$.

Since $\omega \in [0, 2\pi]$ and $\xi \in (0, 1)$,

$$|g^{(N+1)}(\xi)| = |\omega^{N+1} \sin(\omega\xi + (N+1)\pi/2)| \leq \omega^{N+1} \leq (2\pi)^{N+1}.$$

Therefore,

$$|\sin(\omega x + \alpha) - T_N(\omega, \alpha, x)| = |R_N(x)| = \frac{|g^{(N+1)}(\xi)|}{(N+1)!} |x|^{N+1} \leq \frac{(2\pi)^{N+1}}{(N+1)!}$$

for all $x \in [0, 1]$. This completes the proof of Lemma 1. $\square$

Lemma 2. For every $\epsilon > 0$, there exists $N_0 \in \mathbb{N}$ such that for all $N > N_0$,

$$\sup_{0 \leq x \leq 1} \left| \sum_{k=0}^N A_k \sin(\omega_k x + \alpha_k) - \sum_{k=0}^N A_k T_N(\omega_k, \alpha_k, x) \right| < \epsilon, \quad (4)$$

where $\omega_k \in [0, 2\pi]$, $\alpha_k, A_k \in \mathbb{R}$, and $T_N(\omega_k, \alpha_k, x)$ is determined by (3).

Proof. From Lemma 1, we have

$$\sup_{0 \leq x \leq 1} |\sin(\omega_k x + \alpha_k) - T_N(\omega_k, \alpha_k, x)| \leq \frac{(2\pi)^{N+1}}{(N+1)!}$$

for $\omega_k \in [0, 2\pi]$, $\alpha_k \in \mathbb{R}$, $0 \leq k \leq N$, and $N \in \mathbb{N}$. Therefore,

$$\begin{aligned} & \sup_{0 \leq x \leq 1} \left| \sum_{k=0}^N A_k \sin(\omega_k x + \alpha_k) - \sum_{k=0}^N A_k T_N(\omega_k, \alpha_k, x) \right| \\ & \leq \sum_{k=0}^N |A_k| \sup_{0 \leq x \leq 1} |\sin(\omega_k x + \alpha_k) - T_N(\omega_k, \alpha_k, x)| \\ & \leq \sum_{k=0}^N \frac{(2\pi)^{N+1} |A_k|}{(N+1)!} \leq \frac{(2\pi)^{N+1}}{N!} \max_k |A_k|. \end{aligned}$$

Since $\lim_{N \rightarrow \infty} \frac{(2\pi)^{N+1}}{N!} = 0$, for a given $\epsilon > 0$, we can choose $N_0 \in \mathbb{N}$ such that for all $N > N_0$, we obtain (4) to complete the proof of Lemma 3. $\square$

Lemma 3 (Weierstrass Approximation [30]). *Let f be a continuous function on $[0, 1]$. Then, for any $\epsilon > 0$, there exists N_0 such that for all $N > N_0$*

$$\sup_{0 \leq x \leq 1} |f(x) - B_N(f, x)| < \epsilon,$$

where $B_N(f, x)$ is the Bernstein polynomial of f determined by

$$B_N(f, x) = \sum_{l=0}^N f\left(\frac{l}{N}\right) \binom{N}{l} x^l (1-x)^{N-l}. \quad (5)$$

Lemma 4. *Let $p(x) = \sum_{l=0}^N b_l x^l$ be any polynomial, and let $0 < \alpha_k < \frac{\pi}{2}$ for all $0 \leq k \leq N$. Then, there exists $\omega_k \in [0, 2\pi]$ and $A_k \in \mathbb{R}$ such that*

$$p(x) = \sum_{k=0}^N A_k T_N(\omega_k, \alpha_k, x) \quad (6)$$

for all $x \in [0, 1]$, where $T_N(\omega_k, \alpha_k, x)$ is determined by (3).

Proof. Recall the formula determined by (3); then, we have

$$\begin{aligned} \sum_{k=0}^N A_k T_N(\omega_k, \alpha_k, x) &= \sum_{k=0}^N \sum_{l=0}^N A_k \frac{\omega_k^l \sin(\alpha_k + \frac{l\pi}{2}) x^l}{l!} \\ &= \sum_{l=0}^N \left(\sum_{k=0}^N \frac{A_k \omega_k^l \sin(\alpha_k + \frac{l\pi}{2})}{l!} \right) x^l. \end{aligned}$$

To obtain (6), we need to choose ω_k and A_k such that

$$\sum_{k=0}^N \frac{A_k \omega_k^l \sin(\alpha_k + \frac{l\pi}{2})}{l!} = b_l, \quad 0 \leq l \leq N.$$

Equivalently, we need to find ω_k and A_k such that

$$\sum_{k=0}^N A_k \omega_k^l \sin(\alpha_k + \frac{l\pi}{2}) = b_l \cdot l!, \quad 0 \leq l \leq N. \quad (7)$$

Consider the following $(N+1) \times (N+1)$ matrix:

$$M = \begin{bmatrix} \sin(\alpha_0) & \sin(\alpha_1) & \cdots & \sin(\alpha_N) \\ \omega_0 \sin(\alpha_0 + \frac{\pi}{2}) & \omega_1 \sin(\alpha_1 + \frac{\pi}{2}) & \cdots & \omega_N \sin(\alpha_N + \frac{\pi}{2}) \\ \vdots & \vdots & \ddots & \vdots \\ \omega_0^N \sin(\alpha_0 + \frac{N\pi}{2}) & \omega_1^N \sin(\alpha_1 + \frac{N\pi}{2}) & \cdots & \omega_N^N \sin(\alpha_N + \frac{N\pi}{2}) \end{bmatrix}.$$

Since $0 < \alpha_k < \frac{\pi}{2}$, $\sin(\alpha_k + \frac{l\pi}{2}) \neq 0$ for all $0 \leq l \leq N$. By induction in N , we can select $\omega_0, \omega_1, \dots, \omega_N \in [0, 2\pi]$ such that $\det(M) \neq 0$. Therefore, the system of Equation (7) with the augmented matrix M has a solution for $A_0, A_1, \dots, A_N$. This completes the proof of Lemma 4. $\square$

Proof of Theorem 1. For a given $\epsilon > 0$, according to Lemmas 2 and 3, there exists $N_0 \in \mathbb{N}$ such that for all $N > N_0$, we have

$$\sup_{0 \leq x \leq 1} |f(x) - p(x)| < \epsilon/2, \quad (8)$$

where $p(x) = B_N(f, x)$ is the Bernstein polynomial of f , and

$$\sup_{0 \leq x \leq 1} \left| \sum_{k=0}^N A_k \sin(\omega_k x + \alpha_k) - \sum_{k=0}^N A_k T_N(\omega_k, \alpha_k, x) \right| < \epsilon/2 \quad (9)$$

for some ω_k, A_k (to be chosen later) and $\alpha_k = \frac{k\alpha}{N+1} \in (0, \frac{\pi}{2})$.

By virtue of Lemma 4, we can find $\omega_k \in [0, 2\pi]$ and $A_k \in \mathbb{R}$ such that

$$p(x) = \sum_{k=0}^N A_k T_N(\omega_k, \alpha_k, x).$$

Now, combining (8) and (9), we obtain the conclusion of Theorem 1. $\square$

3. Sinusoidal Approximation Theorem for Two-Layer Neural Networks

We next extend the result of Theorem 1 to approximate any continuous function defined on a compact domain in $\mathbb{R}^n$. To simplify the treatment of function extension, we restrict our attention to functions supported within the unit cube $I^n = [0, 1]^n$. Under this setting, a two-layer neural network with sinusoidal activation functions can approximate any continuous function on the unit cube. A similar result for sigmoidal activation functions was established by Kůrková in [20]. The main goal of this section is to approximate a continuous function on a compact domain by a finite sum of nested sinusoidal functions of the form

$$\sum_{j=0}^M B_j \sin \left(v_j \sum_{p=1}^n \sum_{k=0}^N A_{pk}^j \sin(\omega_k x_p + \varphi_k) + \gamma_j \right), \quad (10)$$

where $B_j, A_{pk}^j \in \mathbb{R}$ are coefficients, $v_j, \omega_k \in [0, 2\pi]$ are frequencies, and $\varphi_k, \gamma_j \in (0, \frac{\pi}{2}]$ are phases.

Theorem 2. Let $f : I^n \rightarrow \mathbb{R}$ be a continuous function. For any $\varepsilon > 0$, there exists $M, N \in \mathbb{N}$, $A_{pq,k}, B_{qj} \in \mathbb{R}$, $\omega_{1k}, \omega_{2j} \in [0, 2\pi]$, and $\varphi_{1k}, \varphi_{2j} \in (0, \frac{\pi}{2}]$ such that

$$\left| f(x_1, \dots, x_n) - \sum_{q=1}^{2n+1} \sum_{j=0}^M B_{qj} \sin(\omega_{2j} \sum_{p=1}^n \sum_{k=0}^N A_{pq,k} \sin(\omega_{1k} x_p + \varphi_{1k}) + \varphi_{2j}) \right| < \varepsilon. \quad (11)$$

Proof. According to the Kolmogorov Theorem, any continuous function $f : I^n \rightarrow \mathbb{R}$ can be written as

$$f(x_1, \dots, x_n) = \sum_{q=1}^{2n+1} \phi_q \left(\sum_{p=1}^n \psi_{pq}(x_p) \right),$$

where ϕ_q and ψ_{pq} are continuous functions on the number line.

Each function ψ_{pq} is continuous, and the sum $\sum_{p=1}^n \psi_{pq}(x_p)$ is bounded for all $x \in I^n$. This means that

$$a \leq \sum_{p=1}^n \psi_{pq}(x_p) \leq b$$

for some $a, b \in \mathbb{R}$ and for all $(x_1, \dots, x_n) \in I^n$.

Now, the function ϕ_q is uniformly continuous on $[a, b]$, so for the given ε , there exists $\delta_1 > 0$ such that

$$|\phi_q(u) - \phi_q(v)| < \varepsilon / (4n + 2) \quad (12)$$

whenever $|u - v| < \delta_1$.

For each function $\psi_{pq}(x_p)$, according to Theorem 1, there exists $A_{pq,k}$, w_{1k} and φ_{1k} such that

$$|\psi_{pq}(x_p) - \sum_{k=0}^N A_{pq,k} \sin(w_{1k} x_p + \varphi_{1k})| < \delta_1 / n.$$

Therefore,

$$\begin{aligned} & \left| \sum_{p=1}^n \psi_{pq}(x_p) - \sum_{p=1}^n \left(\sum_{k=0}^N A_{pq,k} \sin(w_{1k} x_p + \varphi_{1k}) \right) \right| \\ & \leq \sum_{p=1}^n \left| \psi_{pq}(x_p) - \left(\sum_{k=0}^N A_{pq,k} \sin(w_{1k} x_p + \varphi_{1k}) \right) \right| \\ & < \sum_{p=1}^n \delta_1 / n = \delta_1. \end{aligned}$$

Now, we apply (12) to $u = \sum_{p=1}^n \psi_{pq}(x_p)$ and $v = \sum_{p=1}^n \left(\sum_{k=0}^N A_{pq,k} \sin(w_{1k} x_p + \varphi_{1k}) \right)$ to obtain

$$\left| \phi_q \left(\sum_{p=1}^n \psi_{pq}(x_p) \right) - \phi_q \left(\sum_{p=1}^n \left(\sum_{k=0}^N A_{pq,k} \sin(w_{1k} x_p + \varphi_{1k}) \right) \right) \right| < \varepsilon / (4n + 2). \quad (13)$$

By applying Theorem 1 to ϕ_q , we obtain the coefficients B_{qj} , frequencies w_{2j} , and phases φ_{2j} such that

$$\left| \phi_q \left(\sum_{p=1}^n \sum_{k=0}^N A_{pq,k} \sin(w_{1k} x_p + \varphi_{1k}) \right) - \sum_{j=0}^M B_{qj} \sin(w_{2j} \left(\sum_{p=1}^n \left(\sum_{k=0}^N A_{pq,k} \sin(w_{1k} x_p + \varphi_{1k}) \right) \right) + \varphi_{2j}) \right| < \delta_2 \quad (14)$$

where $\delta_2 = \varepsilon / (4n + 2)$.

Combining (13) and (14) yields

$$\left| \phi_q \left(\sum_{p=1}^n \psi_{pq}(x_p) \right) - \sum_{j=0}^M B_{qj} \sin \left(w_{2j} \left(\sum_{p=1}^n \left(\sum_{k=0}^N A_{pq,k} \sin(w_{1k}x_p + \varphi_{1k}) \right) \right) + \varphi_{2j} \right) \right| < \varepsilon / (2n + 1).$$

The last inequality follows that

$$\begin{aligned} & \left| f(x_1, \dots, x_n) - \sum_{q=1}^{2n+1} \sum_{j=0}^M B_{qj} \sin \left(w_{2j} \left(\sum_{p=1}^n \left(\sum_{k=0}^N A_{pq,k} \sin(w_{1k}x_p + \varphi_{1k}) \right) \right) + \varphi_{2j} \right) \right| \\ &= \left| \sum_{q=1}^{2n+1} \phi_q \left(\sum_{p=1}^n \psi_{pq}(x_p) \right) - \sum_{q=1}^{2n+1} \sum_{j=0}^M B_{qj} \sin \left(w_{2j} \left(\sum_{p=1}^n \left(\sum_{k=0}^N A_{pq,k} \sin(w_{1k}x_p + \varphi_{1k}) \right) \right) + \varphi_{2j} \right) \right| \\ &\leq \sum_{q=1}^{2n+1} \left| \phi_q \left(\sum_{p=1}^n \psi_{pq}(x_p) \right) - \sum_{j=0}^M B_{qj} \sin \left(w_{2j} \left(\sum_{p=1}^n \left(\sum_{k=0}^N A_{pq,k} \sin(w_{1k}x_p + \varphi_{1k}) \right) \right) + \varphi_{2j} \right) \right| \\ &< \sum_{q=1}^{2n+1} \varepsilon / (2n + 1) = \varepsilon. \end{aligned}$$

This completes the proof of Theorem 2. $\square$

4. Numerical Analysis

To evaluate the performance of our sinusoidal approximation, we compared it to the Fourier series for approximating one-dimensional functions. For higher-dimensional problems with two inputs and one output, we benchmark our approach against two-layer (MLPs) [10], using either ReLU [31] or sine activation functions, basis-spline KANs (B-SplineKANs), and against multi-dimensional truncated Fourier series. In both cases, we consider functions with a single output and compute the relative L^2 error as follows:

$$\text{Relative } L^2 \text{ Error} = \frac{\|\mathbf{y}_{\text{values}} - \mathbf{y}_{\text{fit}}\|_2}{\|\mathbf{y}_{\text{values}}\|_2}. \quad (15)$$

Based on Theorem 1, we constructed our SineKAN model and analyzed its performance numerically using the following neural network formulation for one-dimensional functions:

$$y = \sum_k^G A_k \sin(\omega_k x + k / (G + 1)) + b, \quad (16)$$

where ω_k , A_k , and b are the learnable frequency parameters, amplitude functions, and a bias term, respectively. For multi-dimensional functions, Theorem 2 guides the construction of SineKAN layers as follows:

$$y_j = \sum_k^G \sum_l^N A_{jkl} \sin\left(\omega_k x_l + \frac{k}{G+1} + \frac{l\pi}{N+1}\right) + b_j, \quad (17)$$

$$z_m = \sum_n^G \sum_j^N B_{mnj} \sin\left(v_n y_j + \frac{j}{G+1} + \frac{n\pi}{N+1}\right) + c_m, \quad (18)$$

where A_{jkl} and B_{mnj} are learnable amplitude tensors, b_j and c_m are learnable bias vectors, and ω_k and v_n are learnable frequency vectors. For the one-dimensional case, we consider functions defined on a uniform grid of input values from 0.01 to 1. These functions pose challenges for convergence in Fourier series due to their singularities or non-periodicity:

The first function,

$$f(x) = e^{-\frac{1}{x}} \sin\left(\frac{1}{x}\right), \quad (19)$$

is non-periodic, has a small magnitude across the domain, and exhibits a strong singularity at $x = 0$.

The second function,

$$f(x) = \sum_k e^{\frac{kx}{\pi}} \sin(kx), \quad (20)$$

shows rapid growth and high-frequency oscillations near $x = 0$.

The third function,

$$f(x) = \sum_k e^{-\frac{1}{x}} \sin\left(kx + \frac{\pi}{k}\right), \quad (21)$$

incorporates phase shifts to evaluate the model's performance and convergence with respect to linearly spaced phases.

The final two functions are particularly challenging for Fourier series convergence, allowing us to test our model's convergence behavior:

$$f(x) = x^{\frac{1}{5}} \sin\left(\frac{1}{x}\right), \quad (22)$$

$$f(x) = x^{\frac{4}{5}} \sin\left(\frac{1}{x}\right). \quad (23)$$

All models are fitted using the Trust Region Reflective algorithm for least-squares regression from the `scipy` package [32]. Each function is fitted for a default of 100 steps per fitted parameter. The hyperparameters used for model dimensionality on this task are provided in Table 2, where H is hidden layer dimensionality, Grid In and Grid Out are KAN expansion grid sizes, and Degree is the polynomial degree for B-Splines. Grid In and Grid Out entries are scaled together as pairs.

Table 2. Hyperparameter combinations explored for two-dimensional function fitting. The sets list the values used.

Model	H	Grid In	Grid Out	Degree
SineKAN	{5, 10, 15}	{2, 4, 6}	{1, 2, 3}	N/A
B-SplineKAN	{5, 10, 15}	{2, 4, 6}	{2, 4, 6}	{2, 3}
ReLU MLP	{20, 40, 60, 80, 100}	N/A	N/A	N/A
Sine MLP	{10, 20, 30, 40, 50, 60}	N/A	N/A	N/A

Results for 1D functions are shown in Figure 1. For (19)–(21), the SineKAN approximation significantly outperforms the Fourier series approximation. In (22), performance is roughly comparable between the two. We observe that the function in (22) has less regularity, which causes both the SineKAN and the Fourier series to converge slowly.

For multi-dimensional functions, we benchmarked the following two equations on a 100 by 100 mesh grid of input values ranging from 0.01 to 1:

The first function,

$$f(x, y) = x^2 + y^2 - ae^{-\frac{(x-1)^2+y^2}{c}} - be^{-\frac{(x+1)^2+y^2}{d}}, \quad (24)$$

with parameters $a = \frac{3}{2}$, $b = 1$, $c = 0.5$, and $d = 0.5$, features Gaussian-like terms that create a complex surface, suitable for testing convergence on smooth but non-trivial landscapes.

The second function (Rosenbrock function),

$$f(x, y) = (a - x)^2 + b(y - x^2)^2, \quad (25)$$

with parameters $a = 1$ and $b = 2$, represents a nonlinear, non-symmetric surface, ideal for evaluating convergence in challenging multi-dimensional optimization problems.

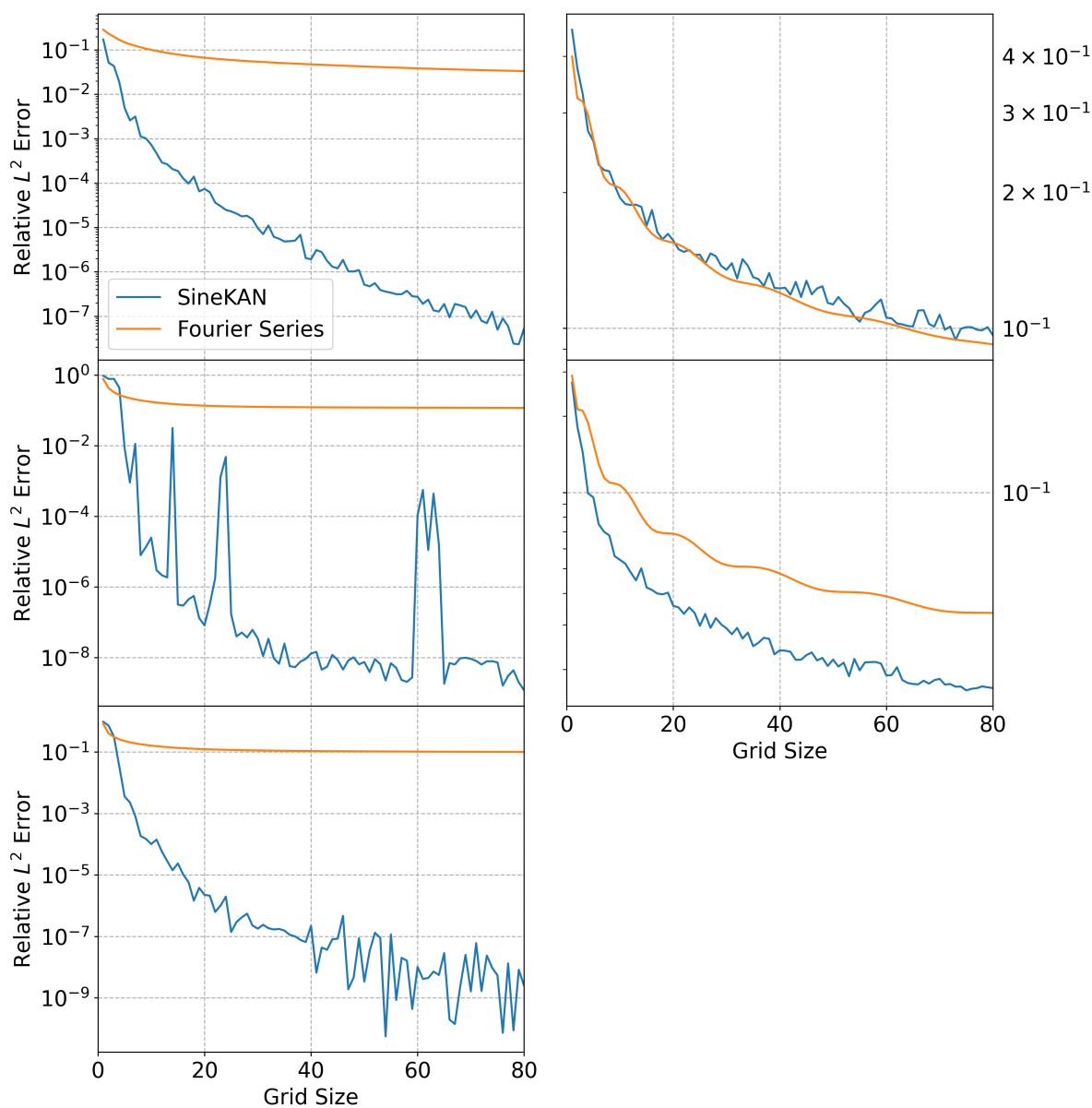


Figure 1. Approximation error as a function of grid size: Top left: Equation (19); middle left: Equation (20); bottom left: Equation (21); top right: Equation (22); middle right: Equation (23).

We show in Figure 2 that the two-layer SineKAN outperforms the two-layer MLP with sinusoidal activation functions as a function of the number of parameters. MLP with ReLU activations performs substantially worse, with an error several orders of magnitude higher as a function of the number of parameters and Fourier series, with a characteristic error roughly one to two orders of magnitude greater than the two-layer MLP with ReLU.

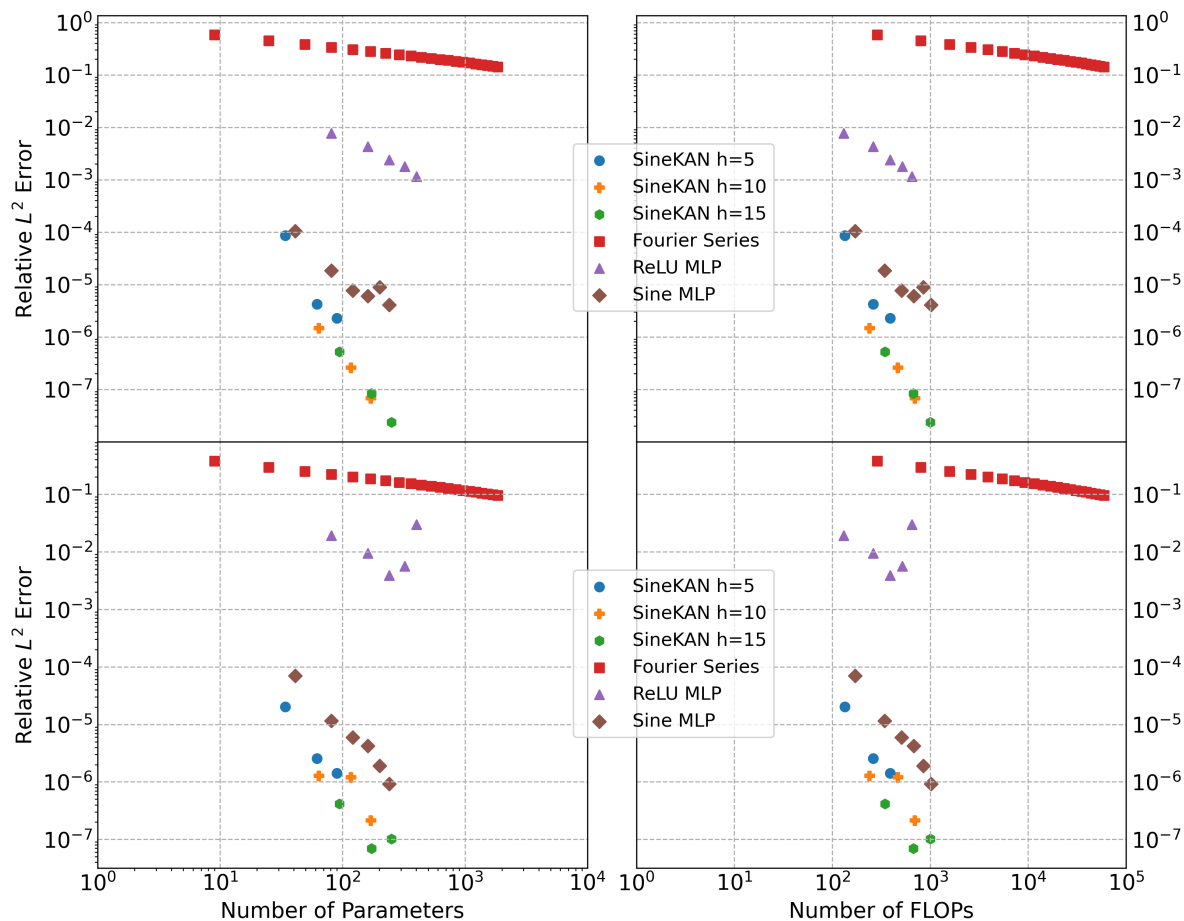


Figure 2. Approximation error of SineKAN, MLP, and Fourier series models as a function of the number of parameters (**Left**) and FLOPs (**Right**) for Equation (24) (**Top**) and Equation (25) (**Bottom**).

We also computed performance as a function of the number of relative FLOPs or compute units. To do this calculation, we ran 10 million iterations using numpy arrays of size 1024 to estimate the relative compute time of addition, multiplication, ReLU, and sine; we found that, when setting addition and multiplication to approximately 1 FLOP, ReLU costs an estimated 1.5 FLOPs, and sine functions cost 12 FLOPs. We carried out similar estimates in PyTorch (<https://pytorch.org/>, accessed on 15 September 2025) and found that the relative FLOPs for sine would be closer to 3.5 in PyTorch, and the relative FLOPs for ReLU would be around 1 FLOP. Figure 2 is based on numpy estimates.

Further, in Figure 3, we show that the SineKAN model scales well compared to B-SplineKAN. The characteristic error with comparable numbers of parameters is typically two full orders of magnitude less for SineKAN, while B-SplineKAN models require substantially more FLOPs for the same number of parameters. Moreover, we find that, when compared to MLP and SineKAN models, B-SplineKAN models have significantly worse consistency in convergence with larger models' dimensionalities, which may necessitate much more precise tuning of hyperparameters or higher-order optimization algorithms.

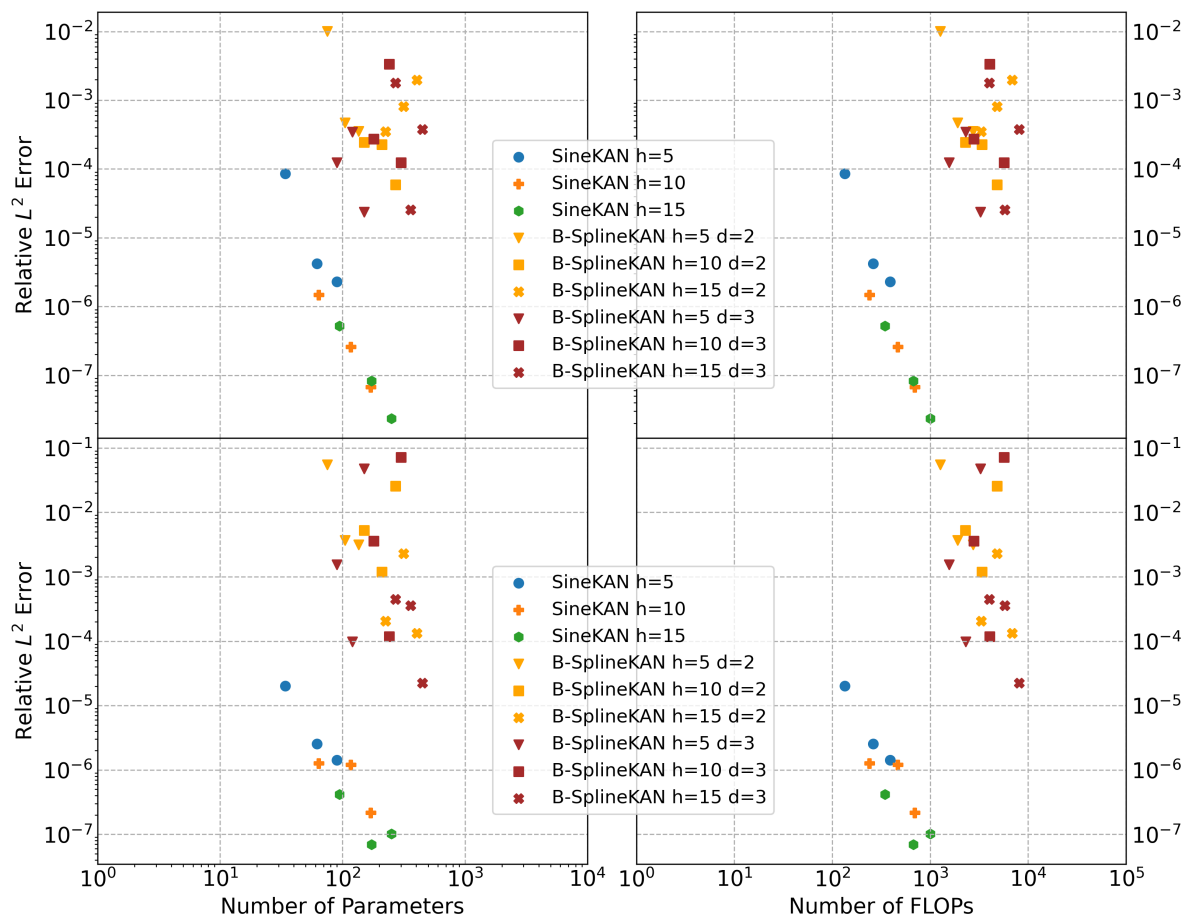


Figure 3. Approximation error of SineKAN and B-SplineKAN models as a function of the number of parameters (Left) and FLOPs (Right) for Equation (24) (Top) and Equation (25) (Bottom).

In Table 3, we show the best-performing models explored. Here, we also evaluate (for comparison) the number of nodes in the computational graphs of the models that can result in a significant bottleneck in backpropagation optimization. The characteristic error for SineKAN is substantially lower than for all other models, including the second-best MLP with sine activations, when controlling for similar numbers of parameters and FLOPs. MLP models with sine activations still achieve comparable performance that is within one to two orders of magnitude of error but require a smaller number of nodes, meaning their optimal training times could potentially be lower as a result.

Table 3. Characteristic parameters, FLOPs, and errors for Equations (24) and (25) for best-performing models explored.

Model	Params	FLOPs	Nodes	Func. 1 Acc.	Func. 2 Acc.
MLP (ReLU)	241	391	61	2.399×10^{-3}	3.916×10^{-3}
KAN (B-Spline)	450	8090	169	3.786×10^{-4}	2.238×10^{-5}
MLP (Sine)	241	1021	61	4.083×10^{-6}	9.250×10^{-7}
KAN (Sine)	250	1000	73	2.343×10^{-8}	1.022×10^{-7}

5. Discussion

The original implementation of the KAN model, developed by Liu et al., used basis-spline functions [22]. These were proposed as an alternative to MLP due to their improved explainability, domain segmentation, and strong numerical performance in modeling functions. However, later work showed that, when accounting for increases in time

and space complexity, the basis-spline KAN underperformed compared to MLP [27]. Previous work on the Sirens model has shown that, for function modeling, particularly for continuously differentiable functions, sinusoidal activations can improve the performance of MLP architectures [33]. This motivated the development of the SineKAN architecture, which builds on both concepts by combining the learnable on-edge activation functions and on-node weighted summation of KAN with the periodic activations from Sirens [29].

Previous work in [29,34] has established that, on tasks that require large models and with discontinuous space mappings, the SineKAN model can perform comparably to, if not better than, other commonly used fundamental machine learning models, including LSTM [12], MLP [10], and B-SplineKAN [22]. We extend these two previous works by providing a robust constructive proof for the approximation power of the SineKAN model. We show that a single layer is sufficient for the approximation of arbitrary one-dimensional functions and that a two-layer SineKAN is sufficient for the approximation of arbitrary multivariable functions bounded by the same constraints as the original KART [35].

In Figures 1–3, we show that these functions can achieve low errors in modeling mathematical functions with features such as rapid- and variable-frequency oscillations. For two-dimensional D functions, we show that SineKAN outperforms MLP, including MLP with sinusoidal activations, with flexible model parameter combinations when accounting for both time and space complexity of the models. This strongly motivates further exploration of this model for numerical approximation tasks.

Given the inherent periodic nature of sinusoidal functions, our approximation framework (2) shows strong potential for modeling periodic and time-series data. Future work will explore the extension of SineKAN to continual learning tasks, particularly in scenarios involving dynamic environments or non-stationary data. Further directions include theoretical analysis of generalization bounds, integration with neural differential equations, and applications in signal processing and real-time prediction systems.

6. Conclusions

In this paper, we build upon Lorentz’s and Sprecher’s foundational work [5,6] to establish two main theorems for approximating single and multivariable functions using the sinusoidal activation function. Our proposed SineKAN models introduce learnable frequencies, amplitudes, and biases, offering a flexible and expressive framework for function approximation. Through numerical experiments, we demonstrate that SineKAN outperforms the classical Fourier series, B-SplineKAN, and MLP models in accuracy across a variety of test cases. In the task of modeling two input and one output functions, the SineKAN model achieves a characteristic relative L^2 error that is half an order of magnitude lower than MLP with sine activations when controlling for similar numbers of parameters and FLOPs. Compared to B-SplineKAN, this extends to roughly three orders of magnitude, and compared to MLP with ReLU activations, this extends to roughly four orders of magnitude. The SineKAN model also significantly outperforms one-dimensional and multi-dimensional Fourier transforms with the sole exception of Equation (22), where large-amplitude oscillations are densely packed near the lower bound of the domain. To support reproducibility and ease of experimentation, we provide a link to our code below.

Author Contributions: Conceptualization, E.A.F.R.; Methodology, H.N. and D.P.R.; Software, E.A.F.R.; Formal analysis, H.N.; Investigation, D.P.R. and E.A.F.R.; Writing—original draft, H.N., D.P.R., and E.A.F.R.; Writing—review & editing, S.G.; Funding acquisition, S.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the US Department of Energy (DOE) under Award No. DE-SC0012447 (D.P.R., E.A.F.R., and S.G.).

Data Availability Statement: The source code for our implementation is available in a GitHub repository: <https://github.com/ereinha/SinusoidalApproximationTheorem> (accessed on 15 September 2025).

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

- Hilbert, D. Über die Gleichung neunten Grades. *Math. Ann.* **1927**, *97*, 243–250. [\[CrossRef\]](#)
- Sternfeld, Y. Hilbert’s 13th problem and dimension. In *Geometric Aspects of Functional Analysis: Israel Seminar (GAFA) 1987–1988*; Springer: Berlin/Heidelberg, Germany, 1989; pp. 1–49. [\[CrossRef\]](#)
- Kolmogorov, A.N. *On the Representation of Continuous Functions of Several Variables by Superpositions of Continuous Functions of a Smaller Number of Variables*; American Mathematical Society: Providence, RI, USA, 1961.
- Somvanshi, S.; Javed, S.A.; Islam, M.M.; Pandit, D.; Das, S. A survey on kolmogorov-arnold network. *Acm Comput. Surv.* **2024**, *58*, 1–35. [\[CrossRef\]](#)
- Lorentz, G. Approximation of functions, athena series. *Sel. Top. Math.* **1966**, *s1-43*, 570–571.
- Sprecher, D.A. On the structure of continuous functions of several variables. *Trans. Am. Math. Soc.* **1965**, *115*, 340–355. [\[CrossRef\]](#)
- Rosenblatt, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychol. Rev.* **1958**, *65*, 386. [\[CrossRef\]](#)
- Minsky, M.; Papert, S.A. *Perceptrons, Reissue of the 1988 Expanded Edition with a New Foreword by Léon Bottou: An Introduction to Computational Geometry*; MIT Press: Cambridge, MA, USA, 2017.
- Amari, S. A theory of adaptive pattern classifiers. *IEEE Trans. Electron. Comput.* **2006**, *EC-16*, 299–307. [\[CrossRef\]](#)
- Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [\[CrossRef\]](#)
- Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 84–90. [\[CrossRef\]](#)
- Sherstinsky, A. Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. *Phys. Nonlinear Phenom.* **2020**, *404*, 132306. [\[CrossRef\]](#)
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.U.; Polosukhin, I. Attention is All you Need. In *Proceedings of the Advances in Neural Information Processing Systems*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30.
- Cybenko, G. Approximation by superpositions of a sigmoidal function. *Math. Control Signals Syst.* **1989**, *2*, 303–314. [\[CrossRef\]](#)
- Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *Neural Netw.* **1989**, *2*, 359–366. [\[CrossRef\]](#)
- Hecht-Nielsen, R. Kolmogorov’s mapping neural network existence theorem. In *Proceedings of the International Conference on Neural Networks*; IEEE Press: New York, NY, USA, 1987; Volume 3, pp. 11–14.
- Hecht-Nielsen, R. *Neurocomputing*; Addison-Wesley Longman Publishing Co., Inc.: Boston, MA, USA, 1989.
- Girosi, F.; Poggio, T. Representation properties of networks: Kolmogorov’s theorem is irrelevant. *Neural Comput.* **1989**, *1*, 465–469. [\[CrossRef\]](#)
- Kůrková, V. Kolmogorov’s Theorem Is Relevant. *Neural Comput.* **1991**, *3*, 617–622. [\[CrossRef\]](#)
- Kůrková, V. Kolmogorov’s theorem and multilayer neural networks. *Neural Netw.* **1992**, *5*, 501–506. [\[CrossRef\]](#)
- Liu, Z.; Ma, P.; Wang, Y.; Matusik, W.; Tegmark, M. Kan 2.0: Kolmogorov-arnold networks meet science. *arXiv* **2024**, arXiv:2408.10205. [\[CrossRef\]](#)
- Liu, Z.; Wang, Y.; Vaidya, S.; Ruehle, F.; Halverson, J.; Soljačić, M.; Hou, T.Y.; Tegmark, M. Kan: Kolmogorov-arnold networks. *arXiv* **2024**, arXiv:2404.19756. [\[PubMed\]](#)
- Li, Z. Kolmogorov-arnold networks are radial basis function networks. *arXiv* **2024**, arXiv:2405.06721. [\[CrossRef\]](#)
- Qiu, Q.; Zhu, T.; Gong, H.; Chen, L.; Ning, H. Relu-kan: New kolmogorov-arnold networks that only need matrix addition, dot multiplication, and relu. *arXiv* **2024**, arXiv:2406.02075.
- Sidharth, S.S.; Keerthana, A.R.; Gokul, R.; Anas, K.P. Chebyshev polynomial-based kolmogorov-arnold networks: An efficient architecture for nonlinear function approximation. *arXiv* **2024**, arXiv:2405.07200.
- Xu, J.; Chen, Z.; Li, J.; Yang, S.; Wang, W.; Hu, X.; Ngai, E.C.H. FourierKAN-GCF: Fourier Kolmogorov-Arnold Network—An Effective and Efficient Feature Transformation for Graph Collaborative Filtering. *arXiv* **2024**, arXiv:2406.01034.
- Shukla, K.; Toscano, J.D.; Wang, Z.; Zou, Z.; Karniadakis, G.E. A comprehensive and fair comparison between mlp and kan representations for differential equations and operator networks. *Comput. Methods Appl. Mech. Eng.* **2024**, *431*, 117290. [\[CrossRef\]](#)
- Yu, R.; Yu, W.; Wang, X. Kan or mlp: A fairer comparison. *arXiv* **2024**, arXiv:2407.16674. [\[CrossRef\]](#)

29. Reinhardt, E.; Ramakrishnan, D.; Gleyzer, S. SineKAN: Kolmogorov-Arnold Networks using sinusoidal activation functions. *Front. Artif. Intell.* **2025**, *7*, 1462952. [[CrossRef](#)]
30. Franklin, P. The Weierstrass Approximation Theorem. *J. Math. Phys.* **1925**, *4*, 148–152. [[CrossRef](#)]
31. Householder, A.S. A theory of steady-state activity in nerve-fiber networks: I. Definitions and preliminary lemmas. *Bull. Math. Biophys.* **1941**, *3*, 63–69. [[CrossRef](#)]
32. Coleman, T.F.; Li, Y. An Interior Trust Region Approach for Nonlinear Minimization Subject to Bounds. *SIAM J. Optim.* **1996**, *6*, 418–445. [[CrossRef](#)]
33. Sitzmann, V.; Martel, J.N.P.; Bergman, A.W.; Lindell, D.B.; Wetzstein, G. Implicit Neural Representations with Periodic Activation Functions. *arXiv* **2020**, arXiv:2006.09661. [[CrossRef](#)]
34. Shamim, M.A.; Reinhardt, E.A.F.; Chowdhury, T.A.; Gleyzer, S.; Araujo, P.T. Probing Quantum Spin Systems with Kolmogorov-Arnold Neural Network Quantum States. *arXiv* **2025**, arXiv:2506.01891. [[CrossRef](#)]
35. Kolmogorov, A.N. On the representations of continuous functions of many variables by superposition of continuous functions of one variable and addition. *Dokl. Akad. Nauk USSR* **1957**, *114*, 953–956. Available online: <https://cs.uwaterloo.ca/~y328yu/classics/Kolmogorov56.pdf> (accessed on 15 September 2025).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.