A Comparative Analysis of KAN Variants Based on Different Basis Function Families with MLP

Araştırma Makalesi/Research Article

Zeynep HABER[1]

[1]Konya Teknik Üniversitesi, Bilgisayar Mühendisliği, Konya, Türkiye

zortacay@ktun.edu.tr

(Geliş/Received:11.08.2026; Kabul/Accepted:27.10.2026)
DOI: 10.17671/gazibtd.2015870

Abstract— This study systematically compares five Kolmogorov-Arnold Network (KAN) variants based on different basis function families (B-Spline, RBF, Chebyshev, Fourier, and Wavelet) against the classical Multi-Layer Perceptron (MLP) under a parameter-matched configuration. Model performance was evaluated across ten mathematical test functions, including six Genz benchmark functions, over 30 repetitions, considering overall approximation performance, dimensionality scaling, noise robustness, and hyperparameter sensitivity. The Friedman test revealed a statistically significant performance difference among the six models ($\chi^2(5) = 29.03$, $p = 0.00002$); the Nemenyi post-hoc test showed that RBF was significantly superior to Fourier, Wavelet, and MLP, while the difference with B-Spline and Chebyshev could not be statistically confirmed. Under matched parameter budgets, MLP's general performance ranked among the lowest of the six models (mean rank 5.0), marginally ahead of Wavelet (mean rank 5.1), which was the lowest-ranked model overall; this indicates that MLP's earlier competitiveness under unequal parameter settings was partly attributable to its larger capacity. Nevertheless, under increasing dimensionality, MLP achieved the lowest scaling exponent ($b = 0.017$) and remained the most stable model under noise, even after the parameter budget was equalized. The findings indicate that KAN variants, particularly RBF, Chebyshev, and B-Spline, achieved the best average ranks under fair comparison conditions, although the Nemenyi test could not statistically distinguish these three from one another; MLP retains an architectural advantage in dimensional scaling and noise robustness; the choice of basis function family should therefore be made carefully according to the characteristics of the target problem.

Keywords— kolmogorov-arnold networks, multi-layer perceptron, basis function comparison, function approximation, statistical significance test

Farklı Temel Fonksiyon Ailelerine Dayalı KAN Varyantlarının MLP ile Karşılaştırmalı Analizi

Özet— Bu çalışmada, Kolmogorov-Arnold Ağları'nın (KAN) farklı temel fonksiyon ailelerine dayalı beş varyantı (B-Spline, RBF, Chebyshev, Fourier ve Wavelet), klasik çok katmanlı algılayıcı (MLP) ile parametre sayısı eşitlenmiş bir yapılandırma altında karşılaştırılmıştır. Altısı Genz test fonksiyonları olmak üzere toplam on farklı matematiksel test fonksiyonu üzerinde, 30 tekrarlı deneylerle modellerin performansı; genel yaklaşım performansı, boyut ölçeklenebilirliği, gürültü dayanıklılığı ve hiperparametre duyarlılığı açısından değerlendirilmiştir. Friedman testi, altı model arasında istatistiksel olarak anlamlı bir performans farkı ortaya koymuş ($\chi^2(5) = 29{,}03$, $p = 0{,}00002$); Nemenyi post-hoc testine göre RBF, Fourier, Wavelet ve MLP'ye karşı anlamlı üstünlük göstermiş, ancak B-Spline ve Chebyshev ile aradaki fark istatistiksel olarak doğrulanamamıştır. Parametre bütçesi eşitlendiğinde MLP, altı model arasında en düşük ortalama sıralardan birinde (5,0) yer almıştır. Bu sıra, en son sırada yer alan Wavelet'in sırasına (5,1) çok yakındır; MLP teknik olarak son sırada değildir, ancak bu sonuç yine de MLP'nin eşit olmayan parametre koşullarındaki eski rekabetçi görünümünün kısmen daha büyük kapasiteden kaynaklandığını göstermektedir. Buna karşın, artan boyutlulukta MLP en düşük ölçeklenme katsayısına ($b = 0{,}017$) sahip olup, parametre bütçesi eşitlenmiş olmasına rağmen gürültü altında da en kararlı model olarak kalmıştır. Bulgular, adil karşılaştırma koşullarında özellikle RBF ve Chebyshev varyantlarının daha düşük yaklaşım hatası sağladığını, buna karşın MLP'nin boyutsal ölçeklenme ve gürültü dayanıklılığında mimari bir avantaj koruduğunu; dolayısıyla temel fonksiyon seçiminin problem karakteristiğine göre dikkatle yapılması gerektiğini göstermektedir.

Anahtar Kelimeler— kolmogorov-arnold ağları, çok katmanlı algılayıcı, temel fonksiyon karşılaştırması, fonksiyon yaklaşıklama, istatistiksel anlamlılık testi

1. INTRODUCTION

Deep learning methods are widely used in many fields [1, 2]. Multi-Layer Perceptrons (MLPs) form the foundation of these methods. Thanks to the universal approximation theorem, MLPs can theoretically approximate any continuous function [3, 4]. This property has made MLP the most preferred architecture for classification and regression problems.

However, despite its success, the MLP architecture has several limitations. Activation functions (ReLU, sigmoid, etc.) are fixed and predetermined throughout the network [5]. This limits the learning capacity [6, 7]. Approximating functions with complex or local features requires a large number of neurons and layers [8, 9]. Moreover, the learned weights do not carry an interpretable meaning. Therefore, MLP models are generally considered "black boxes" [10].

To address these limitations, alternative architectures such as RBF networks [11], extreme learning machine [12], learnable activation functions [13] and, in recent years, Kolmogorov-Arnold Networks (KAN) [14] have been proposed in the literature. The Kolmogorov-Arnold Network (KAN), proposed by Liu et al., presents a new neural network architecture that, unlike classical MLPs, defines learnable activation functions on the edge [14]. In the original study, it was shown that by using B-Spline-based basis functions, the KAN architecture could achieve higher interpretability and more successful results in some function approximation problems. Following this study, numerous alternative basis function families have been proposed to improve the performance of KAN across different function classes. Among these, prominent approaches include FastKAN [15], based on Gaussian radial basis functions, Chebyshev-KAN [16], which uses Chebyshev polynomials, Wavelet-based Wav-KAN [17], FourierKAN [18], based on Fourier series, sinusoidal-based SineKAN [19], rKAN [20], which uses rational functions, and fractional Jacobi-based fKAN [21]. In addition, studies such as BSRBF-KAN [22], which combines B-Spline and RBF bases, FC-KAN [23], which uses different combinations of basis functions, and ReLU-KAN [24], developed for faster computation, have also focused on advancing the KAN architecture from different perspectives in the literature.

Although the KAN architecture has been developing rapidly, recent studies show that its performance can vary considerably depending on the basis function family used, the application domain, and the experimental setup. In particular, studies emphasizing the need for fair comparison [25] reveal that KAN is not superior to MLP in every problem. Comprehensive benchmarking studies also support this picture: Poeta et al. [26] on tabular data, and Shukla et al.[27] showed that the superiority of KAN in differential equation solving and operator learning tasks is task-dependent; Hou et al. [28], through a broader critical evaluation, systematically demonstrated that KAN outperforms MLP only in symbolic regression tasks, while lagging behind MLP in fields such as machine learning, computer vision, and natural language processing. Similarly, Seydi [29] compared 18 different polynomial-based KAN variants only; the X-KAN study [30] proposed a new hybrid architecture combining KAN with evolutionary rule-based learning and evaluated it using statistical methods such as the Friedman test and Holm-corrected Wilcoxon tests. However, most existing studies either focus on a single basis function family or evaluate the performance of a newly proposed variant in specific application domains.

Despite these developments in the literature, studies that systematically compare structurally different basis function families under the same architecture, similar parameter budget, and controlled experimental conditions remain quite limited. Furthermore, in most existing research, performance evaluation is limited to RMSE-based performance metrics alone; important analyses such as dimensionality scaling, noise robustness, hyperparameter sensitivity, and multiple statistical validation are not addressed together. This study aims to fill this gap in the literature by comparing B-Spline, RBF, Chebyshev, Fourier, and Wavelet-based KAN variants together with the classical MLP within the same experimental framework, examining the effect of basis function choice on function characteristics (including different mathematical behaviors such as oscillatory, discontinuous, and sharp-peak), problem dimensionality, noise level, and hyperparameter settings through multi-faceted statistical analyses such as the Friedman test, paired t-tests, and the Wilcoxon signed-rank test.

The main contributions of this study are summarized below:

- B-Spline, RBF, Chebyshev, Fourier, and Wavelet-based KAN variants, together with the classical MLP, were systematically compared.
- The effect of basis function choice on different function characteristics (such as oscillatory, discontinuous, sharp-peak, and non-linear functions) was comprehensively analyzed.
- The robustness of the models was evaluated under different input dimensions, noise levels, and hyperparameter settings.
- The statistical significance of the experimental results was validated using the Friedman test, paired t-test, and Wilcoxon signed-rank test.
- Based on the findings obtained, practical recommendations are provided to guide practitioners on which basis function family is more suitable for different problem types.
- Unlike prior studies that typically examine a single basis function family or a limited subset of evaluation criteria, this study provides a systematic comparison of five structurally different basis function families under a unified, multi-criteria evaluation protocol.

2. MATERIALS AND METHODS

In this study, the classical MLP and KAN based on different basis function families were compared under the same experimental conditions. To ensure a fair comparison among the models, the architectural structure, parameter budget, training algorithm, and training procedure were kept as equal as possible for all models. In this way, it was aimed that the resulting performance differences would stem from the basis function family used, rather than from the architectural design.

2.1. Classic MLP

The MLP is one of the most widely used artificial neural network architectures. It consists of input, hidden, and output layers. The neurons in each layer are fully connected to all neurons in the previous layer. The output of an MLP layer is defined as follows:

$$h^l = \sigma(W^l h^{l-1} + b^l) \tag{1}$$

Here, W^l denotes the weight matrix, b^l denotes the bias vector, and σ denotes the activation function. The activation function (ReLU, sigmoid, tanh, etc.) is fixed throughout the entire network. Only the parameters W and b are learned. This structurally limits the expressive capacity of the model. The universal approximation theorem guarantees that MLP can represent any continuous function. However, this approximation may require a large number of neurons and layers in practice. Furthermore, the fixed activation structure restricts the interpretability of the model.

2.2. Kolmogorov-Arnold Networks

KAN is a neural network architecture proposed by Liu et al. as an alternative to classical MLP [14]. At the core of KAN lies the Kolmogorov-Arnold representation theorem, which states that any continuous multivariate function can be expressed as a composition and sum of univariate functions. This theorem is expressed as in Equation 2:

$$f(x_1, \ldots, x_n) = \sum_{q=0}^{2n} \Phi_q \left(\sum_{p=1}^{n} \varphi_{q,p}(x_p) \right) \tag{2}$$

Here, $\phi_{q,p}$ denotes the inner functions, and Φ_q denotes the outer functions.

The fundamental difference between MLP and KAN lies in the point at which non-linear transformations are applied. In MLP, non-linear activation functions (ReLU, sigmoid, etc.) are located at the nodes and remain fixed throughout training. In KAN, instead of these activations, learnable univariate functions are used on each connection (edge). The nodes only perform the summation of the values coming into them. The output of a KAN layer is computed in Equation 3.

$$\mathrm{h}_j^{(l+1)} = \sum_i \phi_{j,i}^l \, h_i^{\mathrm{l}} \tag{3}$$

Here, $\phi_{j,i}^l$ is a function learned separately for each edge. In the original study, these functions were represented by B-Spline curves. Each edge function consists of the sum of a base contribution and a learnable spline contribution:

$$\phi(x) = w_b \mathrm{SiLU}(x) + w_s \mathrm{spline}(x) \tag{4}$$

Here, SiLU denotes the fixed base function, and *spline* denotes the B-Spline component learned during training. The coefficients w_b and w_s are learnable weights that determine the contribution of these two components. The fundamental difference between MLP and KAN emerges at this point. In MLP, the weights are learnable but the activations are fixed. In KAN, all edge functions are learned from the data. This structure both increases the expressive capacity and contributes to the interpretability of the model. For clarity, this manuscript uses edge function to refer to the full learnable function ϕ on a given KAN connection (Equation 3), basis function to refer to the specific mathematical building block (e.g., a B-Spline curve, a radial basis function, a Chebyshev polynomial term, a Fourier component, or a Morlet Wavelet) from which the edge function is constructed, and basis function family to refer to the class of basis function that characterizes each of the five KAN variants compared in this study.

2.3. KAN Variants

The original KAN architecture uses B-Spline basis functions. Subsequently, different basis function families have been proposed. These studies aim to improve representation performance in specific function classes. In this study, 5 different KAN variants commonly used in the literature are compared.

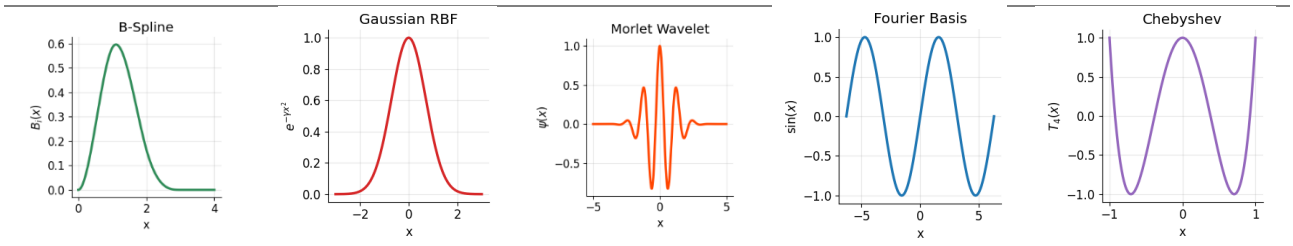

Figure 1. Representative basis functions used in KAN variants

1. B-Spline: In the original KAN architecture proposed by Liu et al. [14], B-Spline functions are used as the basis function. Thanks to its piecewise polynomial structure, B-Spline generates local and smooth functions. Each basis function is active only within a specific region. This property enables local variations to be modeled successfully. Grid size (G) and polynomial degree (k) are the main hyperparameters.

2. Gaussian RBF: FastKAN, proposed by Li [15], uses Gaussian radial basis functions (RBF) instead of B-Spline functions. RBF generates a bell-shaped basis function around each center point.

3. Chebyshev: This variant, proposed by Sidharth et al. [16], uses Chebyshev polynomials as the basis function instead of B-Spline. Chebyshev polynomials provide numerical stability owing to their orthogonal structure.

4. Fourier: This variant, proposed by Xu et al. [18], uses sine and cosine functions as the basis function. Each edge learns a linear combination of Fourier components at different frequencies. This structure provides an advantage particularly in representing periodic functions.

5. Morlet wavelet: This variant, proposed by Bozorgasl and Chen [17], uses Morlet wavelet functions as the basis function. The Morlet wavelet can jointly represent both spatial (or temporal) and frequency information. Each edge learns the contributions of wavelet functions at different positions and scales.

These variants were included in the study because they represent basis function families with different mathematical properties and constitute representative KAN architectures based on different basis function families.

2.4. Test Functions

In this study, two different function groups were used. The first group consists of the Genz test functions [31]. These functions are considered standard benchmark problems in the numerical analysis literature. Each function carries a different mathematical property. This allows the models to be evaluated on different function types.

The second group consists of complementary functions. These functions represent mathematical properties not covered by the Genz tests. The Genz functions were evaluated over the domain $[0,1]^d$. For the complementary functions, the interval $[-1,1]$was used. The difficulty parameters were generated randomly. The random seed was kept fixed to allow the same experiments to be repeated.

Table 1. Mathematical Definitions and Properties of the Benchmark Test Functions Used in the Experiments

Function	**Group**	**Mathematical Expression**	**Mathematical Property**
Oscillatory	Genz	$\cos\left(2\pi w + \sum_{i=1}^{d} c_i\, x_i\right)$	Periodic, regular oscillatory structure
Product Peak	Genz	$\prod_{i=1}^{d} \frac{1}{c_i^{-2} + (x_i - w_i)^2}$	Single, narrow, and high peak
Corner Peak	Genz	$\left(1 + \sum_{i=1}^{d} c_i\, x_i\right)^{-(d+1)}$	Peak concentrated at the corner of the domain
Gaussian	Genz	$\exp\left(-\sum_{i=1}^{d} c_i^2\,(x_i - w_i)^2\right)$	Smooth, symmetric
Continuous (C0)	Genz	$\exp\left(-\sum_{i=1}^{d} c_i \mid x_i - w_i \mid\right)$	Continuous, corner-shaped structure
Discontinuous	Genz	$\exp\left(\sum_{i=1}^{d} c_i\, x_i\right) \mathbf{1}_{\{x_i \le w_i\}}$	Function with a real jump
Runge	Complementary	$\frac{1}{1 + 25\sum_{i=1}^{d} x_i^2}$	Rapid change near the boundary regions
Weierstrass	Complementary	$\sum_{n=0}^{\infty} a^n \cos(b^n \pi x)$	Non-differentiable, high-frequency structure
Multitone	Complementary	$\sum_{k=1}^{m} A_k \sin(2\pi f_k x)$	Multiple frequency components
Bessel J0	Complementary	$J_0(x)$	Damped oscillatory structure

3. RESULTS AND DISCUSSION

3.1. Experimental Setup

All models were implemented using the PyTorch library. Each KAN variant consists of one hidden layer (i.e., two KAN edge-function layers: input to hidden and hidden to output), with a fixed hidden width of 8 neurons and G = 10 basis functions per edge for all variants. Among the five KAN variants, only Fourier and Wavelet have an explicit frequency-related hyperparameter (freq_scale and ω, respectively); the other three variants share the basis-function-count hyperparameter G. These frequency-related hyperparameters were likewise held at their default values throughout the main comparison in Sections 3.2–3.5 (ω = 5.0 for Wavelet); their sensitivity to alternative settings is examined separately in Section 3.6, which also includes a supplementary check of B-Spline, RBF, and Chebyshev's sensitivity to G, together with an independent validation-split replication confirming these results are not a test-set-leakage artifact. The classical MLP baseline was also implemented with a single hidden layer, but its width was chosen dynamically for each input dimension d so that its total parameter count matches that of the non-Fourier KAN variants as closely as possible (within 1%), rather than using a fixed width. Exact parameter counts per model and dimension are reported in Table 3. This parameter matching does not extend to the Fourier variant, whose edge functions carry roughly twice as many coefficients, and comparisons involving Fourier should be interpreted with this difference in mind.

The Adam algorithm was used for optimization. The learning rate was set to 0.02. Each model was trained for 300 epochs. The training dataset contains 250 samples for each function, and the test dataset contains 150 samples. Data points were sampled from a random uniform distribution over the relevant domain in each trial.

To increase the statistical reliability of the results, each experiment was repeated with 30 different random initialization points (seeds). The experimental design consists of four stages. In the first stage, a general performance comparison was conducted on 10 test functions. In the second stage, the effect of increasing dimensionality was examined over $d = \{1,2,5,10\}$ values. In the third stage, noise robustness was tested at $\sigma = \{0,0.01,0.05,0.1\}$ levels. In the fourth stage, hyperparameter sensitivity was evaluated by scanning the basis parameters of the Fourier and Wavelet variants. The main parameters related to the experimental setup are summarized in Table 2.

Table 2. Experimental settings used in all experiments

Parameter	**Value**
Optimizer	Adam
Learning rate	0.02
Number of epochs	300
Hidden layer size (KAN)	8
MLP hidden width (parameter-matched, per d)	62 (d=1), 68 (d=2), 77 (d=5), 82 (d=10)
Number of basis functions (G)	10
Number of training samples	250
Number of test samples	150
Number of repetitions (seed)	30

While Table 2 summarizes the training settings shared across all models, it does not by itself indicate how many trainable parameters each model actually has. Table 3 reports these exact counts for every model at each input dimension, taken directly from the training code's parameter counter. As the table shows, the Fourier variant carries substantially more parameters than the other KAN variants at every dimension (for example, 345 versus 185 at d = 1 and 1857 versus 977 at d = 10), consistent with its edge functions using twice as many coefficients, as noted above.

Table 3. Exact trainable parameter counts per model and input dimension, taken directly from the training code's parameter counter

Model	**d=1**	**d=2**	**d=5**	**d=10**
B-Spline	185	273	537	977
RBF	185	273	537	977
Fourier	345	513	1017	1857
Chebyshev	185	273	537	977
Wavelet	185	273	537	977
MLP	187	273	540	985

3.2. General Performance Comparison

The Friedman test was applied to evaluate the overall performance differences among the models. The Friedman test is a non-parametric statistical method that evaluates differences among multiple related groups based on ranking information.

In this study, 6 models were compared on 10 different functions. Since the RMSE scale differs across functions, ranking information within each function was used instead of raw values. Table 4 presents the model RMSE values and within-function ranking information together for each function. The experimental unit for the Friedman test is therefore the function (N = 10 functions, k = 6 models): the RMSE value used for each function is the mean across the 30 repetitions, and the six models are ranked on this per-function mean.

Table 4. RMSE values and model rankings for the benchmark functions

Functions	B-Spline	RBF	Fourier	Chebyshev	Wavelet	MLP
Bessel J0	0.0149	0.0096	0.0079	0.0076	**0.0067**	0.2036
	5	4	3	2	**1**	6
Continuous C0	**0.0142**	0.0163	0.0293	0.0173	0.0455	0.0351
	1	2	4	3	6	5
Corner Peak	0.0081	**0.0067**	0.0279	0.0136	0.0254	0.0078
	3	**1**	6	4	5	2
Discontinuous	0.7230	**0.6738**	0.9193	0.7910	1.2671	1.0097
	2	**1**	4	3	6	5
Gaussian	0.0118	**0.0057**	0.0342	0.0127	0.0627	0.0407
	2	**1**	4	3	6	5
Multitone	0.0092	**0.0028**	0.0073	0.0043	0.0120	0.1122
	4	**1**	3	2	5	6
Oscillatory	0.0150	**0.0073**	0.0382	0.0197	0.0806	0.0352
	2	**1**	5	3	6	4
Product Peak	0.2899	0.0857	0.2411	**0.0835**	2.0748	1.4946
	4	2	3	**1**	6	5
Runge	0.0049	0.0037	0.0067	**0.0030**	0.0099	0.0239
	3	2	4	**1**	5	6
Weierstrass	0.2207	**0.2180**	0.2405	0.2230	0.2503	0.4044
	2	**1**	4	3	5	6
Mean Rank	2.8	**1.6**	4.0	2.5	5.1	5.0

The Friedman test based on the rankings in Table 4 revealed a statistically significant difference in performance among the six models ($\chi^2(5) = 29.03$, $p = 0.00002$). Among the evaluated models, RBF achieved the best overall performance with the lowest average rank (1.60), whereas Wavelet showed the poorest overall performance with the highest average rank (5.10).

Since the test result was significant, the Nemenyi post-hoc test was applied to determine which pairs of models differed from each other. According to the results, RBF is statistically significantly superior to Wavelet ($p = 0.0004$), MLP ($p = 0.0007$), and Fourier ($p = 0.0474$). The difference with B-Spline and Chebyshev could not be statistically confirmed ($p = 0.706$ and $p = 0.891$). Figure 2 presents the critical difference (CD) diagram based on the Nemenyi post-hoc test.

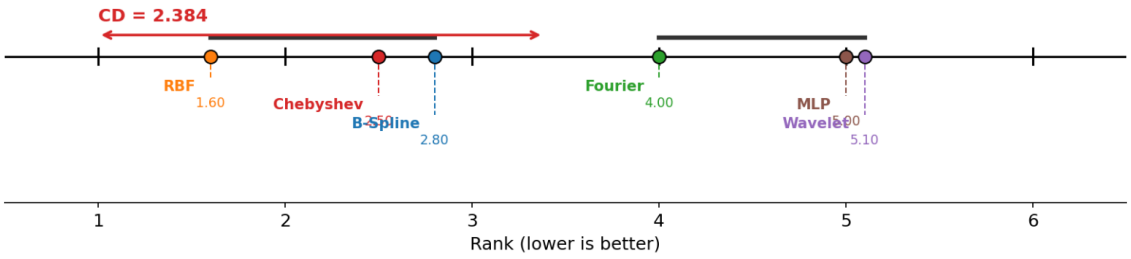

Figure 2. Critical Difference Diagram of Nemenyi post-hoc Test

When the diagram is examined, the models show a tendency to separate into two groups, although the boundary between them is not sharp. The first group includes RBF, Chebyshev, and B-Spline; the average ranking difference among these models remains below the CD threshold (2.384). The second group includes Fourier, MLP, and Wavelet, and these models also cannot be statistically distinguished from one another. The difference between the two groups exceeds the CD threshold for Wavelet and MLP ($p = 0.0004$ and $p = 0.0007$, respectively); the difference with Fourier is comparatively weaker ($p = 0.0474$), close to the conventional significance threshold. To be explicit about which comparisons were statistically tested: Nemenyi p-values were computed for RBF against each of the other five models, since RBF obtained the lowest mean rank; among these, the difference between RBF and B-Spline ($p = 0.706$) or Chebyshev ($p = 0.891$) was not significant, whereas the difference between RBF and Fourier ($p = 0.0474$), MLP ($p = 0.0007$), or Wavelet ($p = 0.0004$) was significant. The remaining pairs (for example, B-Spline versus Wavelet, or Chebyshev versus MLP) were not individually tested; the two-group separation shown in Figure 2 reflects the standard critical-difference-diagram convention, in which models whose mean-rank difference falls below the CD threshold are treated as not significantly distinguishable, and should not be read as a claim that every pair spanning the two groups is individually significant.

The Friedman and Nemenyi tests reveal the overall performance differences among the models. However, these tests do not indicate on which functions the

differences occur. Therefore, the function-based performance differences between the RBF model and the other models were further examined using Holm–Bonferroni corrected paired t-tests. The results are presented in Table 5. Unlike the Friedman test, which operates at the function level (N = 10), these paired t-tests use the seed as the experimental unit: for each function, the 30 paired RMSE differences between RBF and each competing model across the 30 repetitions are tested, and the resulting p-values are Holm–Bonferroni corrected across the 5 × 10 = 50 comparisons.

Table 5. Pairwise comparison of RBF with the other models using Holm–Bonferroni corrected paired t-tests (p-values)

Functions	B-Spline	Fourier	Chebyshev	Wavelet	MLP
Bessel J0	**0.0001 (*)**	0.1163	0.1163	0.0545	**<0.0001 (*)**
Continuous C0	0.0764	**<0.0001 (*)**	0.4680	**<0.0001 (*)**	**<0.0001 (*)**
Corner Peak	0.0735	**<0.0001 (*)**	**<0.0001 (*)**	**<0.0001 (*)**	0.0974
Discontinuous	0.1865	**0.0025 (*)**	**0.0362 (*)**	**<0.0001 (*)**	**0.0025 (*)**
Gaussian	**<0.0001 (*)**	**<0.0001 (*)**	**<0.0001 (*)**	**<0.0001 (*)**	**<0.0001 (*)**
Multitone	**<0.0001 (*)**	**0.0005 (*)**	**0.0014 (*)**	**<0.0001 (*)**	**<0.0001 (*)**
Oscillatory	**<0.0001 (*)**	**<0.0001 (*)**	**<0.0001 (*)**	**<0.0001 (*)**	**<0.0001 (*)**
Product Peak	**<0.0001 (*)**	**<0.0001 (*)**	0.7904	**<0.0001 (*)**	**<0.0001 (*)**
Runge	0.2301	**0.0040 (*)**	0.2301	**0.0055 (*)**	**<0.0001 (*)**
Weierstrass	0.1783	**<0.0001 (*)**	**0.0054 (*)**	**0.0002 (*)**	**<0.0001 (*)**
Wins (p<0.05)	5/10	9/10	6/10	9/10	9/10

Table 5 shows that the RBF model is significantly more successful than the other models, particularly on the Gaussian, Oscillatory, and Multitone functions, where all pairwise comparisons reached significance. In contrast, no significant difference was observed with B-Spline on the Continuous C0, Corner Peak, Discontinuous, Runge, and Weierstrass functions, and with Chebyshev on the Bessel J0, Continuous C0, Product Peak, and Runge functions. Notably, under the parameter-matched configuration, MLP differed significantly from RBF on 9 out of 10 functions (all except Corner Peak), whereas B-Spline and Chebyshev differed on only 5 and 6 functions, respectively. This result indicates that RBF does not provide absolute superiority on all functions, and that its success varies depending on the function characteristics; it also confirms that MLP's relative disadvantage in general performance is broad-based rather than confined to a few functions.

Overall, the basis function family used directly affects function approximation performance. The RBF, B-Spline, and Chebyshev variants produced more consistent results on the examined functions. Fourier and Wavelet-based models showed lower performance particularly on functions containing local variations. The paired t-tests (Holm–Bonferroni corrected) identified significant differences in 38 of 50 comparisons, statistically confirming the observed performance ranking.

3.3. Function-Based Results

The winning pattern varies depending on the function characteristics. Figure 3 shows the summary win-rate matrix of 300 pairwise comparisons (10 functions × 30 seeds). Unlike average RMSE values, this analysis reveals how consistent the relative superiority between models is. In other words, it evaluates not only which model produces lower error, but also the extent to which this superiority is preserved across different functions and different random initializations. The row model shows the proportion of trials in which it produced lower error than the column model.

The pairwise win-rate matrix was constructed as follows. For each pair of models (A, B) and each combination of test function and random seed, a binary outcome was recorded: model A wins if RMSE_A < RMSE_B, and loses otherwise. Since 10 test functions and 30 independent seeds were used, each model pair yields 300 pairwise comparisons. The win rate of model A against model B is defined as the proportion of these 300 comparisons in which model A achieved a lower RMSE:

$$WinRate(A \rightarrow B) = (Number\ of\ trials\ where\ RMSE_A < RMSE_B) / 300 \times 100 \quad (5)$$

Note that Win-Rate(A→B) + Win-Rate(B→A) = 100% by construction, since ties did not occur in continuous-valued outputs. Diagonal entries are undefined and marked with a dash.

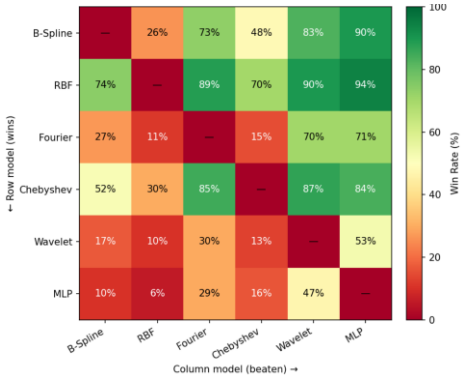

Figure 3. Pairwise win-rate matrix (300 comparisons). Green: row model frequently outperforms column model; red: rarely outperforms

RBF outperforms Wavelet by 90%, Fourier by 89%, and MLP by 84%. The win rate between RBF and B-Spline remains at 74%. This rate is consistent with the fact that the Nemenyi test could not statistically confirm the difference between the two models. This result shows that RBF exhibits consistently superior performance not only on a few specific functions, but across a broad set of functions with different mathematical characteristics. The 74% win rate obtained against B-Spline points to a relatively limited difference. Although both models share a local representation approach, RBF basis functions offer a more flexible approximation capacity.

3.4. Dimensionality Analysis

The dimensionality analysis was performed on three functions representing different mathematical characteristics: Gaussian, with a smooth local structure; Oscillatory, which is periodic; and Discontinuous. This selection aims to confirm that the results are not specific to a single function type and that consistent patterns are observed across functions with different characteristics.

The performance change with increasing dimensionality was analyzed using the log-log regression method. In this approach, log(RMSE) values were linearly modeled over log(d), and the resulting slope coefficient (b) was used as the scaling exponent indicating the model's sensitivity to increasing dimensionality. Low b values indicate that model performance is less affected by increasing dimensionality, whereas high b values indicate that the error increases more rapidly. Figure 4 presents the average RMSE changes of the models at d = 1, 2, 5, and 10 dimensions for three representative functions.

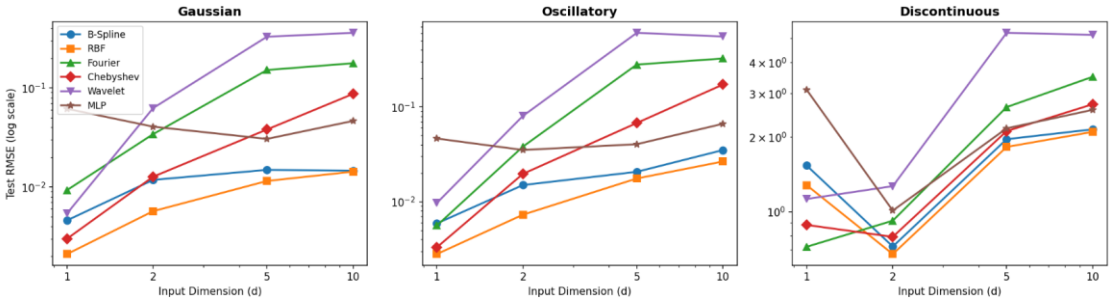

Figure 4. Test RMSE curves with increasing dimensionality

MLP has the lowest mean scaling exponent (b = 0.017), far below the second-lowest, B-Spline (b = 0.486). Table 6 summarizes the function-based scaling exponents. The classical MLP had the lowest average scaling exponent among all models (b = 0.017), while Wavelet had the highest (b = 1.475); this indicates that the rate of error increase with growing dimensionality remains far more limited for MLP than for any KAN variant. Among the KAN variants, B-Spline (b = 0.486) and RBF (b = 0.711), which use local basis functions, scaled noticeably better than Fourier (b = 1.300) and Wavelet (b = 1.475), which experienced a marked performance loss at higher dimensions. Notably, for the Gaussian function, MLP exhibited a negative scaling exponent (b = -0.149), meaning its error decreased rather than increased as dimensionality grew; this pattern is plausibly explained by the relatively unfavorable parameter-to-sample-size ratio available to MLP at low dimensions, which constrains its generalization there, whereas at higher dimensions this ratio becomes comparatively more favorable and improves generalization.

Table 6. Log-log regression scaling exponent

Model	Discontinuous	Gaussian	Oscillatory	Mean	Rank
MLP	0.044	-0.149	0.155	**0.017**	**1**
B-Spline	0.275	0.468	0.717	0.486	2
RBF	0.334	0.824	0.975	0.711	3
Chebyshev	0.567	1.427	1.668	1.221	4
Fourier	0.750	1.329	1.820	1.300	5
Wavelet	0.785	1.823	1.819	1.475	6

This finding does not support the general expectation that the KAN architecture scales better than MLP under all conditions. Notably, this result persists even under the parameter-matched configuration used in this study (Section 3.1), ruling out unequal parameter budget as the explanation. A more likely reason is architectural: KAN assigns a separate learnable function to each input-hidden connection, so as input dimensionality increases, the number of functions that must be learned grows proportionally, and each individual edge function must be fit from a similar amount of training data regardless of d. In contrast, MLP's scalar weights are combined linearly across the entire input space, allowing it to exploit shared structure across dimensions more efficiently even at a comparable total parameter count. The fact that RBF and B-Spline models, which use local basis functions, have lower scaling coefficients than global basis functions indicates that the local representation approach is more stable at high dimensions. In contrast, the global or semi-global basis functions used in Fourier and Wavelet-based

models create more complex optimization problems as dimensionality increases, accelerating the error growth.

These results indicate that the success of the KAN architecture strongly depends not only on the basis function family used, but also on the problem dimensionality. While KAN-based models can provide significant advantages in low-dimensional function approximation problems, this advantage diminishes as dimensionality increases, and in some cases the classical MLP is observed to achieve better generalization. Therefore, in high-dimensional problems, design decisions such as network width, number of basis functions, and parameter budget need to be considered together with the choice of basis function. This finding is consistent with recent studies in the literature emphasizing the need for fair comparison between KAN and MLP, and shows that the superiority of KAN is not absolute, but may vary depending on the experimental setup and problem characteristics. These findings are based on the three functions examined in this analysis (Gaussian, Oscillatory, and Discontinuous) and should not be generalized to the remaining benchmark functions or to other function classes without further verification.

3.5. Noise Analysis

Figure 5 shows the change in test RMSE of each model under increasing noise levels. This analysis was conducted to evaluate not only the approximation performance of the models on clean data, but also their robustness against measurement errors and random perturbations.

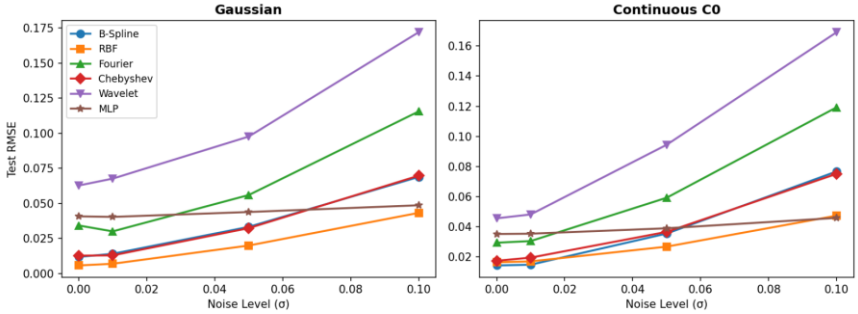

Figure 5. Test RMSE curves with increasing noise level

The relationship between average RMSE on clean data and the noise slope was examined using Spearman rank correlation. In this way, whether a possible trade-off exists between approximation performance and noise robustness was investigated. Spearman correlation measures the monotonic relationship between two variables and does not require the assumption of normal distribution.

MLP has the lowest noise slope on both test functions (Gaussian: 0.083, Continuous C0: 0.109), noticeably lower than the second-most robust model, RBF (0.380 and 0.314, respectively). This indicates that as random perturbations in the input data increase, the rate of error increase for MLP is lower compared to the other models. In other words, although the classical MLP is not the most successful model in terms of approximation performance, it exhibits the most stable generalization behavior under noise. Among the KAN variants, RBF and B-Spline are also the most noise-robust (slopes of 0.347–0.380 and 0.575–0.637, respectively), while Fourier and, especially, Wavelet degrade considerably faster under noise (slopes above 0.84 and 1.09). A possible reason for this is that global, high-frequency basis function families (Fourier, Wavelet) are more sensitive to random perturbations in the input, whereas local, smoother basis functions (RBF, B-Spline) and the fixed-activation MLP generalize more robustly under noisy conditions.

Figure 6 shows the approximation performance of the models on clean data together with their noise fragility. Models located in the lower-left region are considered more advantageous in terms of both high approximation performance and low noise sensitivity. In contrast, models located in the upper-left region achieve high approximation performance but lose performance more rapidly under noise.

Under the parameter-matched configuration, the relationship between clean-data RMSE and noise slope was weak and positive rather than showing a clear trade-off (Spearman r = 0.371, p = 0.468, n = 6). This indicates that, once models are compared under a comparable parameter budget, lower RMSE on clean data is not systematically associated with lower noise robustness; RBF, for instance, combines the lowest noise slope among the KAN variants with the best clean-data RMSE. The relationship did not reach statistical significance given the small number of models (n = 6), so this observation should be treated as descriptive rather than conclusive.

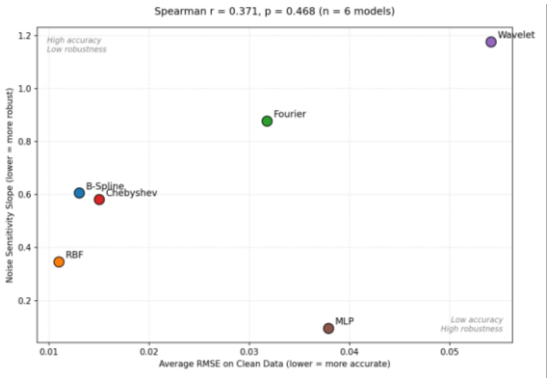

Figure 6. Approximation performance–robustness relationship

This finding shows that the model achieving the best approximation performance may not always be the most reliable model. Particularly in real-world applications where uncertainties such as sensor noise, measurement errors, or missing data exist, focusing only on RMSE-based metrics is not sufficient. Noise robustness should also be considered as an important evaluation criterion in model selection.

MLP constitutes a notable example in this regard. Although it lags behind RBF in terms of average approximation performance, it exhibited the most stable behavior under noise. This suggests that MLP may have a stronger generalization tendency despite its lower representational capacity. In contrast, while KAN-based models achieve superior performance on clean data due to their higher expressive power, the same flexibility can lead to over-sensitivity on noisy data.

Overall, the results obtained show that no single model achieves absolute superiority in both approximation performance and robustness. Therefore, the model selection criterion varies depending on the application. RBF appears to be a more suitable option for clean and controlled datasets, while MLP can be considered a more reliable alternative for noisy or uncertain applications.

3.6. Hyperparameter Analysis

In this section, hyperparameter analysis was performed primarily for the Fourier and Wavelet variants. These two models have frequency-based hyperparameters (freq_scale and ω) that directly affect representation behavior. The B-Spline, RBF, and Chebyshev variants do not have a similar frequency parameter; their sensitivity to the basis-function-count hyperparameter (G) is examined together with the Fourier and Wavelet results in this section, as a supplementary check illustrated jointly in Figure 7 and Table 7. Two representative functions were selected for the analysis. The Gaussian function shows the largest performance difference between the default and best hyperparameter settings. The Multitone function is an example sensitive to frequency scale due to its multi-frequency structure.

Figures 7–8 and Table 7 present an exploratory analysis in which the best settings were identified using test RMSE. We also repeated hyperparameter selection using an independent validation split and evaluated the selected settings on the held-out test set, as reported below. The significance of the performance difference between the default and best hyperparameter settings was evaluated using the Wilcoxon signed-rank test. The Wilcoxon test is a non-parametric method that tests the difference between two paired conditions without requiring the assumption of normality. For each of the 30 seeds, two settings obtained with the same data were compared. Figure 7 shows the performance of all five KAN variants under their default and best/selected hyperparameter settings: frequency-based sweeps for Fourier and Wavelet, and basis-function-count (G) sweeps for B-Spline, Chebyshev, and RBF.

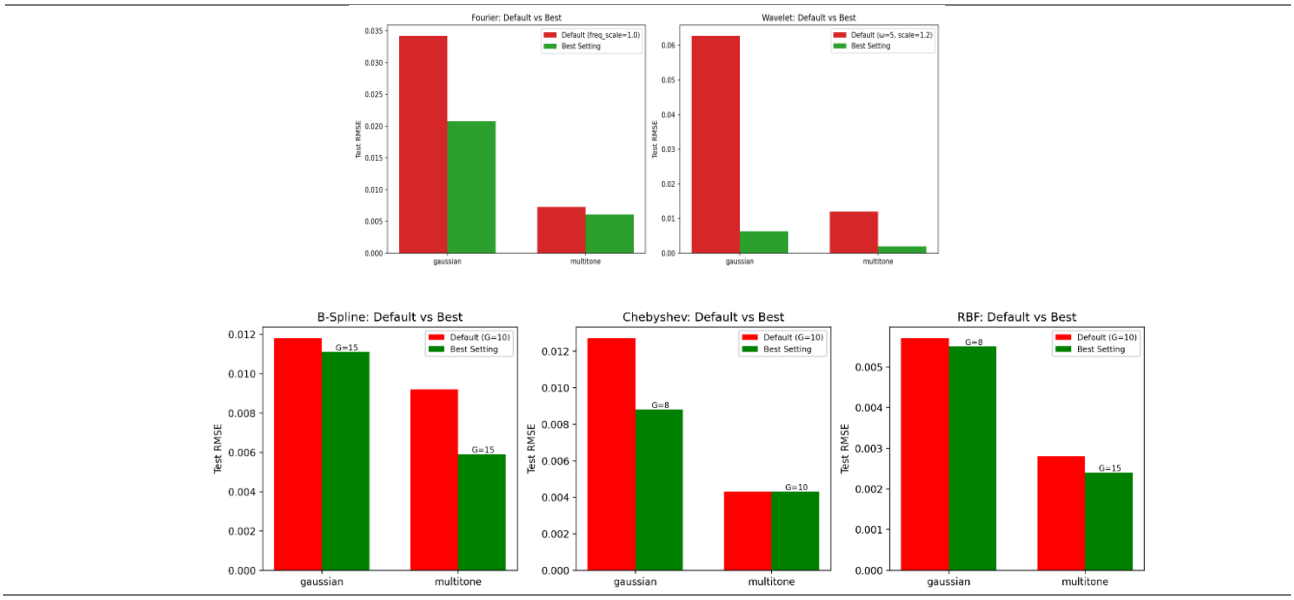

Figure 7. Test RMSE, averaged over 30 seeds, across the tested hyperparameter values for each of the five KAN variants; the red dashed line marks the default setting

Figure 7 shows that hyperparameter selection affects performance in all five variants, but in two qualitatively different ways. For Fourier and Wavelet, test RMSE increases monotonically as the frequency parameter moves away from its best setting; this effect is particularly evident for Wavelet, where trials with the default setting (ω = 5.0) improve substantially as ω decreases toward the selected value of 2.0, while the Fourier curve is comparatively flatter. For B-Spline, Chebyshev, and RBF, the relationship between G and test RMSE is instead non-monotonic: error tends to be higher at both very small (G = 5) and very large (G = 20) values of G than at intermediate values, producing

a U-shaped curve, though the location and sharpness of the minimum vary by model and function. This pattern is most pronounced for Chebyshev, whose Gaussian-function RMSE rises from 0.0088 at G = 8 to 0.0778 at G = 20 (an almost ninefold increase), and whose Multitone-function RMSE is similarly elevated at G = 5 (0.0334) relative to its minimum at G = 10 (0.0043); this suggests that both too few and, especially for Chebyshev, too many basis functions can degrade performance, plausibly reflecting underfitting at low G and numerical instability or overfitting at high G. The same qualitative U-shape is visible for B-Spline and RBF, though with milder degradation at the extremes.

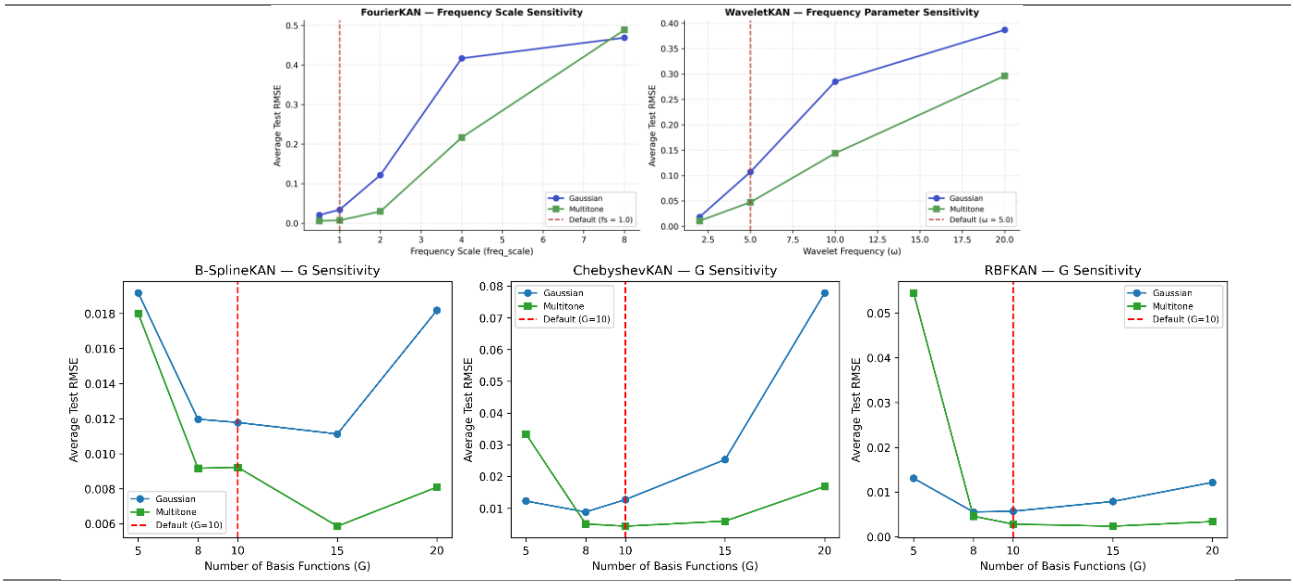

Figure 8. Default versus test-set-selected hyperparameter setting, averaged over 30 seeds, summarized as a single comparison per model and function, corresponding to the values reported in Table 7

Figure 8 summarizes the average performance change across all five variants. The visual results confirm that Wavelet is affected by hyperparameter selection far more than Fourier, and that B-Spline, Chebyshev, and RBF are, in comparison, only modestly sensitive to their basis-function-count hyperparameter, consistent with the smaller improvement factors reported in Table 7. To quantitatively verify these observations, the Wilcoxon test was applied. The results are presented in Table 7, which reports both the Fourier/Wavelet frequency-sensitivity results and the supplementary G-sensitivity results for B-Spline, RBF, and Chebyshev.

Table 7. Comparison of default and best hyperparameter settings

Model	Function	Default RMSE	Best RMSE	Improvement	Wilcoxon p
Fourier	Gaussian	0.0342	0.0208 (fs=0.5)	1.6×	0.009
	Multitone	0.0073	0.0061 (fs=0.5)	1.2×	0.641
Wavelet	Gaussian	0.0627	0.0063 (ω=2.0)	10.0×	<0.001
	Multitone	0.012	0.0020 (ω=2.0)	6.0×	<0.001
B-Spline	Gaussian	0.0118	0.0111 (G=15)	1.06×	0.393
	Multitone	0.0092	0.0059 (G=15)	1.57×	<0.001
Chebyshev	Gaussian	0.0127	0.0088 (G=8)	1.44×	0.002
	Multitone	0.0043	0.0043 (G=10)	1.00× (tie)	n/a
RBF	Gaussian	0.0057	0.0055 (G=8)	1.03×	0.761
	Multitone	0.0028	0.0024 (G=15)	1.20×	0.096

For Wavelet, the ω = 2.0 setting provided the highest improvement on the Gaussian function. The average RMSE decreased by approximately tenfold ($p < 0.001$). On the Multitone function, the error decreased by approximately sixfold. This result shows that Wavelet performance is highly sensitive to the selected Wavelet frequency.

For the Fourier model, freq_scale = 0.5 provided a significant performance improvement on the Gaussian function ($p = 0.009$). However, the improvement obtained on the Multitone function is not statistically significant ($p = 0.641$). This indicates that Fourier is not as sensitive to hyperparameter changes as Wavelet.

Overall, these results show that the low Wavelet performance observed in the previous sections is not solely due to the architecture. When an appropriate hyperparameter is selected, Wavelet can achieve RMSE levels similar to RBF on some functions. Therefore, using only default hyperparameters when comparing KAN variants may not provide a fair evaluation. In particular, optimizing basis-function-specific hyperparameters in Wavelet and Fourier-based models can significantly change performance.

To check whether test-set-based selection affected these findings, we selected hyperparameters on an independent validation split and evaluated them on the held-out test set. The improvements remained similar for Wavelet (9.89× on Gaussian and 5.95× on Multitone) and Fourier (1.65× and 1.21×, respectively). Applying the same validation-based protocol to $G \in \{5, 8, 10, 15, 20\}$ yielded smaller improvements for B-Spline, Chebyshev, and RBF (at most 1.57×). Thus, Wavelet's pronounced sensitivity was also observed when hyperparameter selection and test evaluation were separated.

4. LIMITATIONS

This study has several limitations. First, hyperparameter selection for the Fourier and Wavelet variants (Section 3.6) was originally performed by comparing candidate settings on the same test set used for the main comparison; to verify this did not bias the results, we repeated the selection using an independent validation split and confirmed nearly identical improvement magnitudes and significance levels on the held-out test set (Section 3.6), indicating that the reported sensitivity is not a leakage artifact. Figures 7–8 and Table 7 therefore serve as exploratory sensitivity results, while the independent validation-based check supports the main findings. Second, while the MLP baseline was parameter-matched to the non-Fourier KAN variants, the Fourier variant itself retains roughly twice as many parameters as the other KAN variants because its edge functions use twice as many coefficients; exact parameter counts per model and dimension are reported in Table 3. Third, all KAN variants were evaluated with a fixed hidden width (8 neurons) and a fixed number of basis functions ($G = 10$); a supplementary sensitivity analysis (Section 3.6, Table 7) shows that B-Spline, RBF, and Chebyshev are only mildly affected by G ($\leq$1.6×, mostly not significant), so this choice is unlikely to materially change their relative ranking, though results for other capacity budgets were not exhaustively explored. Fourth, the dimensionality and noise analyses were conducted on partially different subsets of test functions (three and two functions, respectively), selected to keep the experimental grid computationally tractable; this choice may limit direct comparability between the two analyses. Fifth, the comparison relies on synthetic mathematical test functions, including the widely used Genz benchmark family, rather than real-world datasets, which may limit generalizability to applied domains.

5. CONCLUSION

In this study, the classical MLP and KAN based on different basis function families were systematically compared within the same experimental framework. In this context, the original B-Spline-based KAN together with RBF, Chebyshev, Fourier, and Wavelet-based KAN variants were evaluated under the same network architecture and common training strategy. Performance comparisons were not limited to average error values alone; they were comprehensively examined from different perspectives, such as function characteristics, input dimensionality, noise robustness, and hyperparameter sensitivity. In addition, the obtained results were statistically validated using the Friedman test, the Wilcoxon signed-rank test, and paired t-tests.

The experimental results show that the basis function family used has a substantial effect on KAN performance. In terms of overall performance, RBF emerged as the most successful model, producing lower error values than the other KAN variants and the classical MLP, particularly on functions exhibiting local variation. The Chebyshev and B-Spline-based KAN variants closely followed RBF in terms of overall success (with average rankings of 2.5 and 2.8, respectively), while the Fourier and Wavelet-based models, together with MLP, produced competitive results only on certain function types. Notably, under the parameter-matched configuration used in this study, MLP ranked among the lowest of the six models in general performance (mean rank 5.0), marginally ahead of Wavelet (mean rank 5.1), which ranked lowest overall; this indicates that MLP's earlier competitiveness in unequal-parameter settings was partly attributable to its larger capacity rather than its architecture. These findings show that no single basis function family is the most suitable solution for all problem types, and that model success varies depending on the mathematical properties of the function to be represented.

The dimensionality analysis results revealed that the classical MLP exhibits more stable performance as the input dimensionality increases. In contrast, scalability in KAN-based models varies depending on the basis function family used. While the RBF and B-Spline-based models, which use local basis functions, behaved relatively more stably at high dimensions, the Fourier and Wavelet-based models experienced a faster performance loss. This result shows that the superiority of the KAN architecture is not absolute, and that the architectural choice must be made carefully in high-dimensional problems. This dimensionality-scaling finding is based on the subset of benchmark functions used in the dimensionality analysis and should not be generalized to all function classes without further verification.

The noise analysis showed a weak, statistically non-significant positive relationship between clean-data RMSE and noise slope (Spearman $r = 0.371$, $p = 0.468$), rather than a trade-off. RBF combined the best clean-data RMSE with the lowest noise slope among the KAN variants, and B-Spline was also comparatively robust; only Fourier and,

especially, Wavelet showed both weaker approximation performance and markedly higher noise sensitivity. MLP remained the most stable model under noise overall, despite ranking among the lowest of the six models in general performance. This finding is based on the two benchmark functions (Gaussian and Continuous C0) used in the noise-robustness analysis and should be interpreted within that scope rather than as a general claim across all function types. This finding shows that in real-world applications, not only RMSE-based performance but also robustness properties should be considered in model selection, and that, at least under the basis function families examined here, strong approximation performance does not necessarily come at the cost of robustness.

It should be noted that the model rankings reported in Sections 3.2–3.5 reflect performance under fixed, default hyperparameter settings. The hyperparameter analyses showed that the performance of the Wavelet and Fourier models, in particular, largely depends on the appropriate hyperparameter selection. For example, the performance improvement obtained when an appropriate Wavelet frequency is selected for Wavelet revealed that the low performance observed in previous experiments stems not only from the architecture but also from the default hyperparameter selection. This result shows that ensuring fair hyperparameter optimization for all models is of critical importance in future comparative studies.

In conclusion, this study presents a systematic comparison in which KAN variants are comprehensively evaluated not only in terms of average approximation performance, but also in terms of function characteristics, scalability, noise robustness, and hyperparameter sensitivity. Within the scope of the synthetic benchmark functions considered, the findings obtained provide practical recommendations for selecting the appropriate basis function family for specific problem types, while also forming an experimental reference for new KAN architectures to be developed in the future; their applicability to real-world datasets remains to be established in future work.

In future studies, it is planned to extend this comparison to higher-dimensional problems, real-world datasets (particularly given the reliance on synthetic benchmark functions in the present study), and recently proposed Rational-KAN, Jacobi-KAN, Bernstein-KAN, and hybrid KAN architectures. In addition, examining the effect of approaches such as automatic hyperparameter optimization and adaptive basis function selection on KAN performance will more comprehensively reveal the potential of these architectures in different application domains.

REFERENCES

[1] J. Ahmad, H. Farman, Z. Jan, "Deep Learning Methods and Applications," in Deep Learning: Convergence to Big Data AnalyticsSingapore: Springer 2019, pp. 31-42.

[2] S. R. Kadam, M. P. Dhore, "Review of Deep Learning Methods", 2024 8th International Conference on Computing, Communication, Control and Automation (ICCUBEA), 1-8, 2024. DOI:10.1109/ICCUBEA61740.2024.10774807

[3] G. Cybenko, "Approximation by superpositions of a sigmoidal function", Mathematics of Control, Signals and Systems, 2(4), 303-314, 1989. DOI:10.1007/BF02551274

[4] K. Hornik, M. Stinchcombe, H. White, "Multilayer feedforward networks are universal approximators", Neural Networks, 2(5), 359-366, 1989. DOI:10.1016/0893-6080(89)90020-8

[5] I. Goodfellow, Y. Bengio, A. Courville, Deep learning, Cilt·1, MIT press Cambridge, 2016.

[6] M. M. Hammad, "Deep learning activation functions: Fixed-shape, parametric, adaptive, stochastic, miscellaneous, non-standard, ensemble", 2024. DOI:10.48550/arXiv.2407.11090

[7] A. Omidvar, "SmartMixed: A Two-Phase Training Strategy for Adaptive Activation Function Learning in Neural Networks", 2025. DOI:10.48550/arXiv.2510.22450

[8] R. Eldan, O. Shamir, "The power of depth for feedforward neural networks", 29th Annual Conference on Learning Theory, 907-940, 2016.

[9] M. Telgarsky, "Benefits of depth in neural networks", 29th Annual Conference on Learning Theory, 1517-1539, 2016.

[10] I. C. Yeh, W. L. Cheng, "First and second order sensitivity analysis of MLP", Neurocomputing, 73(10), 2225-2233, 2010. DOI:10.1016/j.neucom.2010.01.011

[11] D. S. Broomhead, D. Lowe, "Multivariable Functional Interpolation and Adaptive Networks," 1988, vol. 2.

[12] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, "Extreme learning machine: theory and applications", Neurocomputing, 70(1-3), 489-501, 2006. DOI:10.1016/j.neucom.2005.12.126

[13] F. Agostinelli, M. Hoffman, P. Sadowski, P. Baldi, "Learning activation functions to improve deep neural networks", ICLR 2015 Workshop, 2015.

[14] Z. Liu, Y. Wang, S. Vaidya, F. Ruehle, J. Halverson, M. Soljacic, T. Hou, M. Tegmark, "KAN: Kolmogorov–arnold networks", International conference on learning representations, 70367-70413, 2025.

[15] Z. Li, "Kolmogorov-arnold networks are radial basis function networks", 2024. DOI:10.48550/arXiv.2405.06721

[16] S. Sidharth, A. Keerthana, R. Gokul, K. Anas, "Chebyshev polynomial-based kolmogorov-arnold networks: An efficient architecture for nonlinear function approximation", 2024. DOI:10.48550/arXiv.2405.07200

[17] Z. Bozorgasl, H. Chen, "Wav-kan: Wavelet kolmogorov-arnold networks", 2024. DOI:10.48550/arXiv.2405.12832

[18] J. Xu, Z. Chen, J. Li, S. Yang, W. Wang, X. Hu, E. Ngai, "Enhancing graph collaborative filtering with FourierKAN feature transformation", Proceedings of the 34th ACM International Conference on Information and Knowledge Management, 5376-5380, 2025. DOI:10.1145/3746252.3760909

[19] E. Reinhardt, D. Ramakrishnan, S. Gleyzer, "SineKAN: Kolmogorov-Arnold Networks using sinusoidal activation functions", Frontiers in artificial intelligence, 7, 1462952, 2025. DOI:10.3389/frai.2024.1462952

[20] A. A. Aghaei, M. Hosseinzadeh, K. Parand, "rkan: Rational kolmogorov-arnold networks", Neural Networks, 108888, 2026. DOI:10.1016/j.neunet.2026.108888

[21] A. A. Aghaei, "fkan: Fractional kolmogorov–arnold networks with trainable jacobi basis functions", Neurocomputing, 623, 129414, 2025. DOI:10.1016/j.neucom.2025.129414

[22] H.-T. Ta, "BSRBF-KAN: A combination of B-splines and radial basis functions in Kolmogorov-Arnold networks", **International Symposium on Information and Communication Technology**, 3-15, 2024.

[23] H.-T. Ta, D.-Q. Thai, A. B. S. Rahman, G. Sidorov, A. Gelbukh, "Fc-kan: Function combinations in kolmogorov-arnold networks", Information Sciences, 123103, 2026. DOI:10.1016/j.ins.2026.123103

[24] Q. Qiu, T. Zhu, H. Gong, L. Chen, H. Ning, "Relu-kan: New kolmogorov-arnold networks that only need matrix addition, dot multiplication, and relu", **2025 IEEE Smart World Congress (SWC)**, 1686-1694, 2025. DOI:10.1109/SWC65939.2025.00262

[25] R. Yu, W. Yu, X. Wang, "Kan or mlp: A fairer comparison", 2024. DOI:10.48550/arXiv.2407.16674

[26] E. Poeta, F. Giobergia, E. Pastor, T. Cerquitelli, E. Baralis, "A benchmarking study of Kolmogorov-Arnold networks on tabular data", **2024 IEEE 18th International Conference on Application of Information and Communication Technologies (AICT)**, 1-6, 2024. DOI:10.1109/AICT61888.2024.10740444

[27] K. Shukla, J. D. Toscano, Z. Wang, Z. Zou, G. E. Karniadakis, "A comprehensive and FAIR comparison between MLP and KAN representations for differential equations and operator networks", Computer Methods in Applied Mechanics and Engineering, 431, 117290, 2024. DOI:10.1016/j.cma.2024.117290

[28] Y. Hou, T. Ji, D. Zhang, A. Stefanidis, "Kolmogorov-Arnold Networks: A Critical Assessment of Claims, Performance, and Practical Viability", 2024. DOI:10.48550/arXiv.2407.11075

[29] S. T. Seydi, "Exploring the potential of polynomial basis functions in Kolmogorov-Arnold networks: A comparative study of different groups of polynomials", 2024. DOI:10.48550/arXiv.2406.02583

[30] H. Shiraishi, H. Ishibuchi, M. Nakata, "X-kan: Optimizing local kolmogorov-arnold networks via evolutionary rule-based machine learning", 2025. DOI:10.24963/ijcai.2025/993

[31] A. Genz, "Testing multidimensional integration routines", **Proc. of international conference on Tools, methods and languages for scientific and engineering computation**, Paris, France, 81–94, 1984.