Kolmogorov-Arnold Networks for High-Dimensional Estimation: a Method of Sieves Approach

Sami Abdurahman

January 15, 2026

Abstract

This paper introduces a novel sieve extremum estimator based on Kolmogorov-Arnold Networks (KANs), designed for nonparametric estimation in high-dimensional settings involving time series data. By integrating KANs with the method of sieves and leveraging sparsity, our method achieves asymptotic rates of convergence in the mean-square norm that are independent of the covariate dimension, thereby mitigating the curse of dimensionality. Specifically, our estimator yields an explicit convergence rate of $o_P(n^{-1/4})$. This framework accommodates diverse applications such as high-dimensional conditional density estimation and nuisance function estimation within Double/Debiased Machine Learning, areas where existing deep neural network frameworks are less suited for due to their "black box" nature, reliance on i.i.d. assumptions, and slower convergence rates. Our proposed KAN sieve extremum estimator overcomes these limitations by learning activation functions using B-splines, and its theoretical framework rigorously permits stationary mixing processes.

Keywords: Artificial neural networks, causal inference, curse of dimensionality, deep learning, double/debiased machine learning, Group Lasso, Kolmogorov-Arnold networks, nonparametric/semiparametric estimation, sieve estimation, sparse neural networks, time series.
JEL classification: C14, C32, C45, G12.

Author Affiliation: Toronto Metropolitan University, Department of Economics.
Email: sami.abdurahman@torontomu.ca; all errors are my own.
Computations were performed on the Trillium supercomputer at the SciNet HPC Consortium. SciNet is funded by Innovation, Science and Economic Development Canada; the Digital Research Alliance of Canada; the Ontario Research Fund: Research Excellence; and the University of Toronto.
Full code with documentation is available at: `https://github.com/samiabd8/SieveKAN`

1 Introduction

In modern statistical and economic applications, flexible nonparametric function estimation methods with wide-applicability are increasingly important for modeling complex relationships arising from high-dimensional datasets. Examples of these methods include regression models based on kernels and penalized splines. However, a fundamental challenge facing the efficacy of these methods in high-dimensional settings is the curse of dimensionality (CoD). This phenomenon dictates that the sample size required for a given estimation accuracy grows exponentially with the dimensionality of the covariate space. As a direct consequence of this, derived rates of convergence will be prohibitively slow, rendering the use of a wide range of traditional nonparametric estimation techniques impractical in these settings. Although approaches such as generalized additive models (GAMs) may seek to mitigate the CoD through imposing structural assumptions (e.g., additivity), these assumptions will generally fail to capture the compositional and interactive nature inherent in many complex relationships arising from real-world phenomena.

One of these flexible methods for nonparametric estimation that has demonstrated strong empirical performance in a variety of machine learning tasks (e.g., regression, classification, natural language processing) in recent years is deep neural networks (DNNs), largely due to their powerful approximation capabilities for complex functions. Despite their demonstrated empirical excellence (see Goodfellow et al., 2016 for a comprehensive overview), the application of DNNs for rigorous statistical inference and econometric analysis presents many significant challenges. The most critical of which arises from the "black box" nature of these network architectures, which hinders tractability[1] in a manner that models such as GAMs do not. Furthermore, much of the recent work in the literature that attempts to bridge DNNs with statistical theory for inference (typically falling under the category of "Deep ReLU networks") rely on independent and identically distributed (i.i.d.) assumptions. While such

[1]From the perspective of controlling the metric entropy of the network in high-dimensions, which is a measure of the function space complexity (e.g., in terms of covering numbers).

assumptions simplify the analysis on the approximation capabilities of these structures, they render DNNs unsuited for time series and panel data structures common in economic applications. Lastly, obtaining sufficiently fast convergence rates for the application of these architectures under high-dimensionality is a highly nontrivial task due to the CoD. As we will show in this paper, commonly derived convergence rates in the literature require extremely restrictive conditions that break down in these settings. Such conditions tie the smoothness of the function to be estimated to the dimension of the covariates (e.g., minimax optimal convergence rates in nonparametric regression); requiring degrees of differentiability that are generally unrealistic for even a modest number of regressors, in order to deliver the convergence rate needed for semiparametric inference.

Introduced by Liu et al. (2024), Kolmogorov-Arnold Networks (KANs) represent a promising alternative to the general DNN architecture (multilayer perceptrons, or MLPs) that inherently addresses some of these limitations. By replacing fixed activations in MLP structures with learned spline functions, KANs do not suffer from the "black box" nature associated with DNNs. In MLPs, complex relationships are obscured within learnable weight matrices and fixed nonlinearities in the form of commonly used activation functions. As the name suggests, the universal approximation capabilities of KANs arise from the Kolmogorov-Arnold Representation theorem (KART). The KART (Kolmogorov, 1956) posits that continuous multivariate functions can be represented as a finite composition of univariate functions and summation operations, regardless of the degree of interaction between covariates in the multivariate function.

This paper develops a novel and theoretically grounded sieve extremum estimator based on deep KANs for nonparametric estimation in high-dimensional settings with time series data. By integrating KANs with the sieve estimation framework of Chen and Shen (1998) and leveraging a sparse version of the KART (where only a small number of univariate components are non-zero), we achieve explicit asymptotic convergence rates in the L^2-norm that do not depend on the dimension of the covariates (d). This directly mitigates the CoD in

a manner similar to sparse GAMs when the target function admits a sparse compositional form. Although we assume the true sparse compositional target function is fixed for a given sample size n, our novel regularization method centered around Group Lasso[2] can be viewed as consistent with a data-driven form of model selection from the perspective of Barron et al. (1999). Our sieve estimation approach relies on a finite-dimensional space that grows with n in order to control network complexity growth in addition to sparsity, while simultaneously ensuring sufficient network approximation power. Unlike MLP architectures where the approximation power is tied to the network width (i.e., the number of neurons in the hidden layers, necessitating proportional increases in the total number of learnable weights) leading to intractable complexity with respect to metric entropy, the expressiveness of KANs hinges on the flexibility of the splines (governed by the grid resolution and smoothness) in our learnable activation functions.

Our proposed KAN sieve extremum estimator θ_n achieves a dimension-independent convergence rate of $o_P(n^{-1/4})$, the aforementioned rate required for delivering $\sqrt{n}$-consistency in semiparametric inference. By satisfying the necessary conditions outlined in Chen and Shen (1998)[3], we establish the asymptotic normality of smooth (linear or otherwise) plug-in functionals of $\hat{\theta}_n$. As a result, we are able to accommodate a wide range of practical applications involving time series and semiparametric inference. Such applications include the nonparametric regression model example that we consider in simulations, as well as high-dimensional conditional density estimation. We also demonstrate how our estimator can be used for nuisance function estimation in the Double/Debiased Machine Learning (DML) framework of Chernozhukov et al. (2018). DML has emerged as an indispensable tool in high-dimensional statistics for constructing valid inference for finite-dimensional parameters. We discuss how this framework offers a bias-robust alternative in ultra-high-dimensional regimes

[2]Specifically, we apply a Group Lasso penalty to the vector of B-spline coefficients for each learned activation function, pruning redundant univariate functions.

[3]We demonstrate how the KAN sieve extremum estimator satisfies these high-level assumptions (pertaining to the Riesz representation theorem and stochastic equicontinuity) by adapting the verification approach of Chen and White (1999) to sieve KANs.

($d \gg n$), although the integration of sample-splitting for time series data is left to future work[4] as it is beyond the scope of this paper. The KAN sieve extremum estimator consistently demonstrates strong performance across Monte Carlo simulations and an empirical application involving financial factors time series data. Our novel regularization method involving Group Lasso effectively prunes redundant univariate functions in the architecture, crucial for controlling the metric entropy. Through the tractability of our sieve KAN architecture featuring learned B-spline activation functions, our proposed framework bridges the gap between deep (otherwise black box) compositional networks and GAMs.

1.1 Related Literature

Many of the recent contributions to the literature on DNNs revolved around networks using the Rectified Linear Units (ReLU) activation function, $\sigma(x) = \max\{0, x\}$, due to their convenient and relatively simple properties. The manner in which this activation function truncates negative inputs can be viewed as a natural induction of sparsity[5] in activation maps (as opposed to in weights, see the Appendix for an illustrative example), in addition to mitigating the vanishing gradient problem (discussed later on in the context of our sieve KAN optimization approach). Yarotsky (2017, 2018) develops the approximation theory for this class of DNNs, while Farrell et al. (2021) and Schmidt-Hieber (2020) propose rigorous frameworks for their application in semiparametric inference and nonparametric regression, respectively. The former derives nonasymptotic high probability bounds for ReLU DNNs, while the latter considers a sparsely connected[6] architecture and obtains an oracle inequality. In an earlier study addressing the CoD and potential avenues for circumventing it in the context of ReLU DNNs[7], Mhaskar and Poggio (2016) define *relative dimensions*. This

[4]e.g., the "neighbors-left-out" cross-fitting method of Semenova et al. (2023), which would significantly complicate the independent blocks construction our main results are built upon.

[5]Ensuring that only neurons with non-negative inputs ($x \geq 0$) remain active, preserving important features across forward passes. This differs from the *structural* sparsity relevant to this framework and methods such as weight decay.

[6]With respect to the network weights, where a compositional structure assumption is imposed on the target function.

[7]With a focus on deep convolutional networks, see LeCun et al. (2015) for an overview.

concept provides quantitative measurements of sparsity tied to parameters of the DNN and leverages compositional function assumptions to provide a plausible explanation for the improved performance of DNNs relative to "shallow" networks (single hidden layer feedforward networks, or SLFNs). We contrast this with our proposed sieve KAN using a conditional density estimation example in the Appendix, highlighting its limited applicability for general DNNs.

Linking the improved performance of artificial neural networks (aNNs) when additional layers are added in MLPs to the KART, Schmidt-Hieber (2021) presents an interpretation of this representation theorem using deep ReLU networks under simple modifications (namely, smoothness of the target function; resulting in learnable Lipschitz continuous inner functions). This constituted a breakthrough in the application of the KART in aNNs, once thought to be irrelevant[8] in this context. Although several papers in the time between Kolmogorov (1956) and Schmidt-Hieber (2021) have attempted to show the usefulness of the KART in aNNs (e.g., Sprecher, 1996, 1997), these were limited to network structures with two hidden layers and fixed activation functions.

Applying this crucial insight, Liu et al. (2024) propose an alternative to MLPs in the form of KANs, which derive their universal approximation capabilities from the KART rather than the universal approximation theorem[9]. While both of these DNN architectures are fully-connected, KANs replace the fixed activation functions in MLPs with learnable activation functions in the form of B-splines. The authors demonstrate how by defining the layers of KANs in terms of matrices of univariate functions, the KART can be generalized to arbitrary widths and depths. As a result, KANs are effectively models with MLPs on the outside and splines on the inside, the latter contributing to significantly improved accuracy as shown by Liu et al. (2024) in a wide range of applications. Deriving generalization bounds for KANs, Zhang and Zhou (2024) show how it is possible to guarantee that the bound scales

[8] e.g., Girosi and Poggio (1989), titled: "Representation Properties of Networks: Kolmogorov's Theorem Is Irrelevant" attributed to the otherwise fractal nature of the inner functions.

[9] Introduced in Hornik et al. (1989), a foundational paper in the literature on DNNs and machine learning.

with the L^1-norm of the coefficients when the activation function is composed of B-spline basis functions. In both of these KAN papers, ℓ_1-regularization plays an essential role. While these papers lay the groundwork for KANs with learned activation functions in the form of B-splines, our proposed framework introduces a new regularization procedure centered around Group Lasso and achieves explicit rates of convergence for the sieve KAN M-estimator.

In a major contribution to the literature on sieve extremum estimation (Grenander, 1981), Chen and Shen (1998) obtain convergence rates for these nonparametric and semiparametric estimators involving time series data; in addition to establishing the $\sqrt{n}$-consistency and asymptotic normality of plug-in sieve extremum estimates with respect to smooth functionals[10]. The method of sieves entails estimating a function belonging to a possibly infinite-dimensional parameter space Θ using a sequence of approximating parameter spaces (growing in complexity as $n \to \infty$, defined as the sieve space Θ_n) that are dense in Θ. This sieve M-estimator framework derives theoretical results using a quasi-maximum-likelihood estimation (QMLE) method (White, 1982; Gourieroux et al., 1984) which does not suffer from the issues associated with infinite-dimensional maximum likelihood (namely, slow convergence rates and potential inconsistency under misspecification). Crucially, the extension of sieve estimation to time series data in Chen and Shen (1998) allows for the application of this method to a wide range of economic applications that would otherwise not be possible under the i.i.d assumptions made in the existing literature (e.g., Shen, 1997). Among the applications involving time series models considered to demonstrate this sieve extremum estimator, rates of convergence for nonlinear state-space models using aNNs are obtained. Chen and White (1999) obtain improved approximation rates for shallow sieve aNN estimators building directly upon this framework, see the Appendix for a full exposition. An outline of regularization approaches in nonlinear (finite-dimensional or otherwise) sieve can be found in Chen (2007). Shen et al. (2023) investigate the asymptotic properties of shallow sieve network estimators and discuss how their results can potentially be extended to DNNs with

[10]a "plug-in" estimator is a method of estimating a functional of interest by substituting unknown population distributions or nuisance functions with their estimated counterparts.

Lipschitz continuous activation functions. To the best of our knowledge, there does not exist a framework integrating the method of sieves with modern DNN architectures in order to establish the consistency of such networks in nonparametric regression.

The remainder of this paper is organized as follows. Section 2 defines the target function space and outlines our proposed KAN sieve extremum estimator, including the key parameters of the sieve KAN architecture and its assumptions. Section 3 presents our main theoretical results on approximation error and convergence rates. We present results of our empirical example and Monte Carlo simulations for our DML (semiparametric mean regression) model in Section 4. Section 5 concludes with a summary and a discussion of directions for future research.

2 Sieve Kolmogorov-Arnold Networks

We begin by defining the relevant metric spaces and their properties, in addition to the notation used throughout this paper. While our notation and norm definitions loosely follow Chen and White (1999), we deviate in our characterization of the target function space. Rather than assuming functions in this space admit a Fourier representation[11], we simply assume that it takes on the form of a Sobolev space. As we discuss in Section 3, we impose a relatively strong local structural assumption on the univariate functions (Lipschitz continuity), rather than the global spectral/smoothness assumption of Chen and White (1999).

Given a strictly stationary sequence $\{Z_t\}_{t=1}^n = (Y_t, X_t')'$ with dimension $(1+d)$, where $\{X_t\}_{t=1}^n$ is a sequence of covariates (potentially including lags), let μ be a probability measure on $\mathbb{R}^d$ (the distribution of $X_t \in \mathbb{R}^d$). For the semiparametric structural models of interest, define the partition $X_{t,\mathcal{I}_D} \in \mathbb{R}^{d_1}$ and $X_{t,\mathcal{I}_X} \in \mathbb{R}^{d_2}$, where $\mathcal{I}_D = \{1, \ldots, d_1\}$ and $\mathcal{I}_X = \{d_1+1, \ldots, d\}$ and $d_1 + d_2 = d \in \mathbb{N}$. For simplicity, let $D_t \equiv X_{t,\mathcal{I}_D}$ (representing the linear component) and $W_t \equiv X_{t,\mathcal{I}_X}$ (noting that μ now corresponds to $W_t \in \mathbb{R}^{d_2}$) in semiparametric

[11] Key for obtaining dimension-independent rates in their framework, see Section A.3 in the Appendix for a full exposition.

applications.

We define $L_2(\mu)$ as the Hilbert space of all measurable functions $h : \mathbb{R}^d \to \mathbb{R}$ such that $\int_{\mathbb{R}^d} |h(x)|^2 d\mu(x) < \infty$, equipped with the following norm:

$$\|h\|_{L_2(\mu)} = \left(\int_{\mathbb{R}^d} |h(x)|^2 d\mu(x) \right)^{1/2}$$

To simplify the notation throughout this paper, let $\|\cdot\| \equiv \|\cdot\|_{L_2(\mu)}$ unless stated otherwise. In order to characterize the space of the target function ($f_0 \equiv \theta_0 \in \Theta$ for non-parametric regression[12], where Θ may more generally represent a class of regression functions, probability densities, and other structural mappings of interest), we define the multi-index[13] $\lambda = (\lambda_1, \ldots, \lambda_d)^\intercal$, which represents a vector of non-negative integers with order $|\lambda| = \sum_{i=1}^d \lambda_i$. Let D^λ represent the weak partial derivative operator corresponding to λ, where:

$$D^\lambda h(x) = \frac{\partial^{|\lambda|} h(x)}{\partial x_1^{\lambda_1} \ldots \partial x_d^{\lambda_d}}$$

The target function space is defined as a Sobolev space $\mathcal{F}_d^{m+1} \equiv W_2^{m+1}(\mu)$. In addition to consisting of all measurable functions $h : \mathbb{R}^d \to \mathbb{R}$ such that $h \in L_2(\mu)$, this space contains its weak derivatives $D^\lambda h \in L_2(\mu)$ for all multi-indices λ with $|\lambda| \leq m + 1$. The norm for a Sobolev space $W_2^{m+1}(\mu)$ is given by:

$$\|h\|_{W_2^{m+1}(\mu)} = \left[\sum_{|\lambda|=0}^{m+1} \left(\|D^\lambda h\|_{L_2(\mu)} \right)^2 \right]^{1/2}$$

As we demonstrate in our model assumptions, this definition allows us to quantify the smoothness of the true function we seek to estimate. As a result, we are able to link the properties of the proposed sieve KAN architecture that features (learned) spline activation functions to this function space. Critically, this target space allows us to work towards

[12]Whereas $\theta_0 \equiv (\alpha_0, f_0)$ in the aforementioned semiparametric applications, e.g., the partially linear regression model (Robinson, 1988).

[13]A vector specifying the number of partial derivatives to be taken with respect to each corresponding input variable.

convergence rates where the dependence on the smoothness parameter m is absorbed into constants; unlike typical "minimax" optimal convergence rates[14] (Stone, 1982) commonly found in the literature on deep ReLU networks. For example, a common minimax estimation rate in nonparametric estimation takes on the form of $n^{-m/(2m+d)}$, which is prohibitively slow in high-dimensional settings. By avoiding such a dependency, our approach is a key step towards mitigating the CoD in DNN architectures.

Although our analysis relies on the smoothness provided by the Sobolev space, we are only able to obtain convergence rates in the L^2-norm rather than in the stronger Sobolev norm (the latter implying convergence in the former). We will demonstrate why this is the case when applying spline approximation theory in our key results, as rates of convergence in the Sobolev norm would not be independent of m under the sieve KAN architecture. This is one of the main ways our approach differs from Chen and White (1999), where results for consistent nonparametric sieve network estimators (in the form of single hidden layer aNNs) are obtained in the Sobolev norm. Nonetheless, this weaker form of convergence is sufficient for the applications we consider, as well as for important extensions[15]. Furthermore, given how quickly convergence rates involving non-static conditions on m breakdown in high-dimensional settings (see the Appendix for an illustrative example), this tradeoff is justified by how it enables the proper theoretical grounding for analysis in these settings.

We assume the target function $f_0 \in \mathcal{F}_d^{m+1} \equiv W_2^{m+1}$ has a sparse compositional structure (Assumption **K.3**), where only a small subset (of cardinality s_0) of the univariate functions in the KART are non-zero. This form of sparsity arises naturally in many applications involving high-dimensional data. For example, in a financial risk model predicting the probability of a given borrower defaulting, a researcher might collect hundreds of variables based on available data. It is reasonable to expect that only a few covariates (e.g., including payment history and debt-to-income ratios) truly impact odds of default, with complex (not necessarily additive)

[14]Where the smoothness degree of the target function m is tied to d; see Schmidt-Heiber (2020) for a full discussion in relation to DNNs.

[15]Namely, obtaining asymptotic normal linear plug-in estimators in a semiparametric model framework, where the nonparametric component is estimated using our proposed KAN sieve extremum estimator.

nonlinear relationships arising between this small subset of variables. Formally, the KART states:

Definition 1 (Kolmogorov-Arnold Representation Theorem): for a multivariate smooth $f : [0,1]^d \to \mathbb{R}$,

$$f(X_t) = f(X_{t,1}, X_{t,2}, \ldots, X_{t,d}) = \sum_{i=1}^{2d+1} \Phi_i \left(\sum_{j=1}^{d} \phi_{i,j}(X_{t,j}) \right) \tag{1}$$

In the context of the possibly infinite-dimensional parameter space $\mathcal{F}_d^{m+1}$, this sparse compositional assumption is analogous to those made in sparse GAM frameworks[16], as it entails approximating a multivariate function using a collection of univariate functions[17]. As we demonstrate throughout this paper, a sieve KAN can be viewed as a generalization of these models to sparse compositional structures (thus allowing for complex interactions), embedded within a DNN architecture. In terms of multiplicative interactions, Liu et al. (2024) demonstrate how even a simple two-layer KAN structure is capable of approximating functions such as $f(x_1, x_2) = x_1 x_2$. In this example, it can be observed that the KAN computes $x_1 x_2$ by leveraging $2x_1x_2 = (x_1 + x_2)^2 - (x_1^2 + x_2^2)$, where the spline functions learn the individual parts of this decomposition. The main consequence of this sparse compositional assumption imposed on the target function $f_0 \in \mathcal{F}_d^{m+1}$ is that it allows for the dimension of the covariates, d, to grow with the sample size at even a polynomial rate: $d = O(n^{\zeta_d})$ for any $\zeta_d > 0$.

The main consequence of this structural assumption is we are able to derive explicit asymptotic convergence rates in the L_2 norm that do not depend on d at all. The ability of our proposed framework to permit dimensions that scale as the sample increases without causing the metric entropy to grow uncontrollably[18] makes it a very promising direction for

[16] Such as Raskutti et al. (2011), Meier et al. (2009) and Friedman (1991).

[17] Also referred to one dimensional, or "1D" functions.

[18] By leveraging spline approximation theory in addition to the method of sieves and structural sparsity, as we discuss in the next section.

the application of sparse deep networks in high-dimensional settings; namely due to how assumptions commonly made in high-dimensional models such as $s_0 = O(1)$ (which implies that the target function can be approximated by a fixed number of univariate components: e.g., $f(X_t) = \sum_{j=1}^{s_0} f_j(X_{t,j})$ in GAMs) can be overly restrictive in many practical applications. In GAMs, s_0 represents the number of covariates that have a non-zero functional effect on the response variable. As a result, we mitigate the CoD by exploiting sparsity in a manner similar to sparse additive models and bound the network entropy by using a sieve estimation approach. We note that a similar claim is also made in Liu et al. (2024), one that is debatable for general KAN classes in the absence of sparsity and appropriate mechanisms for controlling complexity[19]. Without these theoretical underpinnings, the network complexity *implicitly scales with* d and renders general KAN classes vulnerable to exponential growth of said complexity, especially in high-dimensional settings. Going forward, we use the symbol $\asymp$ to denote asymptotic equivalence[20], which is important for our network parameter definitions.

2.1 KAN Sieve Extremum Estimator

We integrate our proposed KAN architecture in the sieve extremum estimator framework of Chen and Shen (1998), which allows for time series data (Assumption **K.1**) and avoids the reliance on the otherwise restrictive i.i.d assumption (e.g., as in Schmidt-Heiber, (2020) and Farrell et al., (2021), among many other works on deep ReLU networks). This is of significance as many of the potential high-dimensional applications a researcher might be interested in using deep networks for generally involve time series data. In addition to obtaining asymptotic normal plug-in estimators, we consider semiparametric applications in the double/debiased machine learning framework introduced by Chernozhukov et al. (2018). Our proposed KAN sieve extremum estimator[21], $\theta_n \in \Theta_n$, achieves the required rate of con-

[19] i.e., the sparse compositional form imposed on the target function and our sieve estimation approach.

[20] Where $a_n \asymp b_n \iff a_n \gtrsim b_n$ and $b_n \gtrsim a_n$ as $n \to \infty$. This is equivalent to $b_n = O(a_n)$ and $a_n = O(b_n)$.

[21] We refer to θ_n as such due to how the sieve space Θ_n is constructed using the outputs of our deep sieve network. To avoid confusion, "sieve KAN" refers to the *architecture* with network weights ω_n.

vergence (precisely $o(n^{-1/4})$ in the L^2-norm) for nuisance function estimation under Neyman-orthogonal moments/scores in DML.

For the strictly stationary process $\{Z_t\}_{t=1}^n = (Y_t, X_t')'$ and the true probability measure P_0, where $\theta_0 : \mathbb{R}^d \to \mathbb{R}$ is defined as the maximizer of the population expected criterion function (identified by the functional $\mathcal{T}$ under P_0), we define the following:

$$\theta_0 = \mathcal{T}(P_0) = \arg\sup_{\theta \in \Theta} E_{P_0}[l(Z_t, \theta)] \tag{2}$$

$$\hat{\theta}_n \equiv \arg\sup_{\theta \in \Theta_n} \mathcal{L}_n(\theta) - \lambda_n R_n(\theta) \equiv \arg\sup_{\theta \in \Theta_n} \frac{1}{n} \sum_{t=1}^{n} l(Z_t, \theta) - \lambda_n R_n(\theta) \tag{3}$$

$$\lambda_n \asymp O\left(\sqrt{\frac{\log p_n}{n}}\right) \tag{4}$$

Here, $l(\cdot)$ is a quasi-log-likelihood function belonging to a class of general loss functions, permitting a variety of nonparametric regression problems. For example, $l(Z_t, \theta) = -\frac{1}{2}(Y_t - \theta(X_t))^2$ in the case of nonparametric mean regression and $l(Z_t, \theta) = -|Y_t - \theta(X_t)|[\tau \mathbf{1}_{Y_t \geq \theta(X_t)} + (1-\tau)\mathbf{1}_{Y_t < \theta(X_t)}]$ when applied in quantile regression models (see Koenker, 2005). The regularization penalty term λ_n is asymptotically equivalent to a term that accounts for the (log) total number of network parameters (p_n) relative to the sample size, selected via cross-validation (CV). Lastly, $R_n(\theta)$ is the regularizer term used to represent the constrained coefficients of our learned B-spline activation functions, which we expand upon in the following section.

Example 1 (Partially Linear Regression): Suppose $\{Y_t, D_t, W_t'\}$ satisfies the model[22]

$$Y_t = D_t \alpha_0 + g_0(W_t) + U_t, \qquad \mathbb{E}[U_t | D_t, W_t] = 0,$$

where D_t is the (treatment) variable of interest and g_0 is an unknown nonparametric function of controls W_t. Assume $g_0 \in W_2^{m+1}$ admits a sparse compositional form (**Assumption K.3(a)**), where $\theta \equiv (\alpha_0, g_0) \in \Theta$ and μ is the distribution of W_t. Let $\hat{g}_n \in \mathcal{K}_n$ be the sieve KAN estimator for g_0 obtained using $l(Z_t, \theta) = -(Y_t - D_t\alpha - g(W_t))^2/2$ after profiling out

[22]See the partition of X_t defined earlier in the context of semiparametric structural models.

α. Then, under **Assumption K.1-4**, we have:

$$(i) \quad \|\hat{\theta}_n - \theta_0\|_{L_2(\mu)} = o_P(n^{-1/4}), \text{ and}$$
$$(ii) \quad \sqrt{n}\big(\hat{\alpha}_n - \alpha_0\big) \rightsquigarrow \mathcal{N}(0, \Sigma_\alpha),$$

Example 2 (Average Partial Derivative Estimation): Suppose $\{Y_t, X_t'\}$ satisfies $Y_t = \theta_0(X_t) + \epsilon_t$, $\mathbb{E}[\epsilon_t|X_t] = 0$, $\text{Var}[\epsilon_t|X_t] = \sigma^2(X_t) > 0$, with $\theta_0 \equiv f_0 \in W_2^{m+1}$ admitting a sparse compositional form. Let $\hat{\theta}_n \in \mathcal{K}_n$ be the sieve KAN estimator obtained using $l(Z_t, \theta) = -(Y_t - \theta(X_t))^2/2$. For a covariate $X_{t,j}$ with $j \in \{1, \ldots, d\}$ and under **Assumption K.1-4**:

$$(i) \quad \|\hat{\theta}_n - \theta_0\|_{L_2(\mu)} = o_P(n^{-1/4}), \text{ and}$$
$$(ii) \quad \sqrt{n}\left(\frac{1}{n}\sum_{t=1}^{n} \frac{\partial \hat{\theta}_n(X_t)}{\partial x_j} - \mathbb{E}\left[\frac{\partial \theta_0(X_t)}{\partial x_j}\right]\right) \rightsquigarrow \mathcal{N}(0, \Sigma_{\gamma_j}).$$

Example 3 (Dynamic Discrete Choice Model Estimation): Consider a dynamic discrete choice model[23] where an agent chooses an action $d_t \in \{1, \ldots, J\}$ each period to maximize expected discounted utility. The per-period utility is $u(s_t, d_t; q)$, where s_t is the state vector and q is a vector of parameters. The value function satisfies

$$V(s_t; q) = \max_{d_t}\big\{u(s_t, d_t; q) + \beta\, \mathbb{E}[V(s_{t+1}; q) \mid s_t, d_t]\big\},$$

with discount factor $\beta \in (0, 1)$. Let $F(s_t, d_t; q) \equiv \mathbb{E}[V(s_{t+1}; q) \mid s_t, d_t]$ be the expected value function. Then under **Assumption K.1-4**, we have:

$$(i) \quad \|\hat{F}_n - F_0\|_{L_2(\mu)} = o_P(n^{-1/4})$$

[23]See Norets (2012) for an example of the application of aNNs in this setting. An extension of Norets (2012) centered around inference procedures facilitated by the KAN sieve extremum estimator is left to future work.

where μ is the joint distribution of (s, d, q).

2.2 Sieve KAN Architecture

The network architecture of our sieve KAN estimator has three key parameters: the number of layers (L_n), the network width (W_n; corresponding to the number of nodes W_n^l in a given layer $l \in [1, \ldots, L_n]$) and the size of the B-spline grid (G_n) for the learnable activation functions which makes the network fully tractable. Unlike how MLPs place fixed activation functions on nodes, the learnable activation functions belong on *edges* (or, "in-between" layers) so that the KAN nodes simply perform summation. This compositional structure can be represented as follows, in contrast with the structure of a deep MLP:

$$\text{KAN}(\boldsymbol{x}) = \big(\boldsymbol{\Phi}_{L_n-1} \circ \boldsymbol{\Phi}_{L_n-2} \circ \ \ldots \ \circ \boldsymbol{\Phi}_1 \circ \boldsymbol{\Phi}_0\big)\boldsymbol{x} \tag{5}$$

$$\text{MLP}(\boldsymbol{x}) = \big(\boldsymbol{W}_{L-1} \circ \sigma \circ \boldsymbol{W}_{L-2} \circ \sigma \circ \ \ldots \ \circ \boldsymbol{W}_1 \circ \sigma \circ \boldsymbol{W}_0\big)\boldsymbol{x} \tag{6}$$

where σ denotes a fixed activation (e.g., ReLU) and $\boldsymbol{\Phi}_l$ represents a collection of learnable univariate spline functions at layer l in the form of a matrix. In other words, the typical weight matrices on the edges of MLP structures are replaced with univariate functions in the form of B-splines. To reflect this, we define $r_n \coloneqq L_n W_n^2$ as a key parameter[24] representing the number of total *potential* edges in our KAN architecture.

The difference between r_n and the number of *active* edges[25] can be seen in Figure 1, where the solid lines represent learned activation functions with non-zero B-spline coefficients under ℓ_1-regularization. In this example, the second input variable (x_2) does not contribute to the true target function[26] in the DGP and there is an inactive node present in the second hidden layer, lacking any activation functions to sum as a result of the Group Lasso regularization.

[24] Where $r_n \equiv W_n$ simply represents the number of hidden units in the case of the shallow sieve networks from Chen and White (1999), as outlined in the Appendix.

[25] Denoted by s_n and determined by s_0 in addition to the ℓ_1 constraint Δ_n: another key parameter in our sieve space definition

[26] Throughout this paper, we refer to covariates upon which f_0 depends on as 'relevant' for brevity.

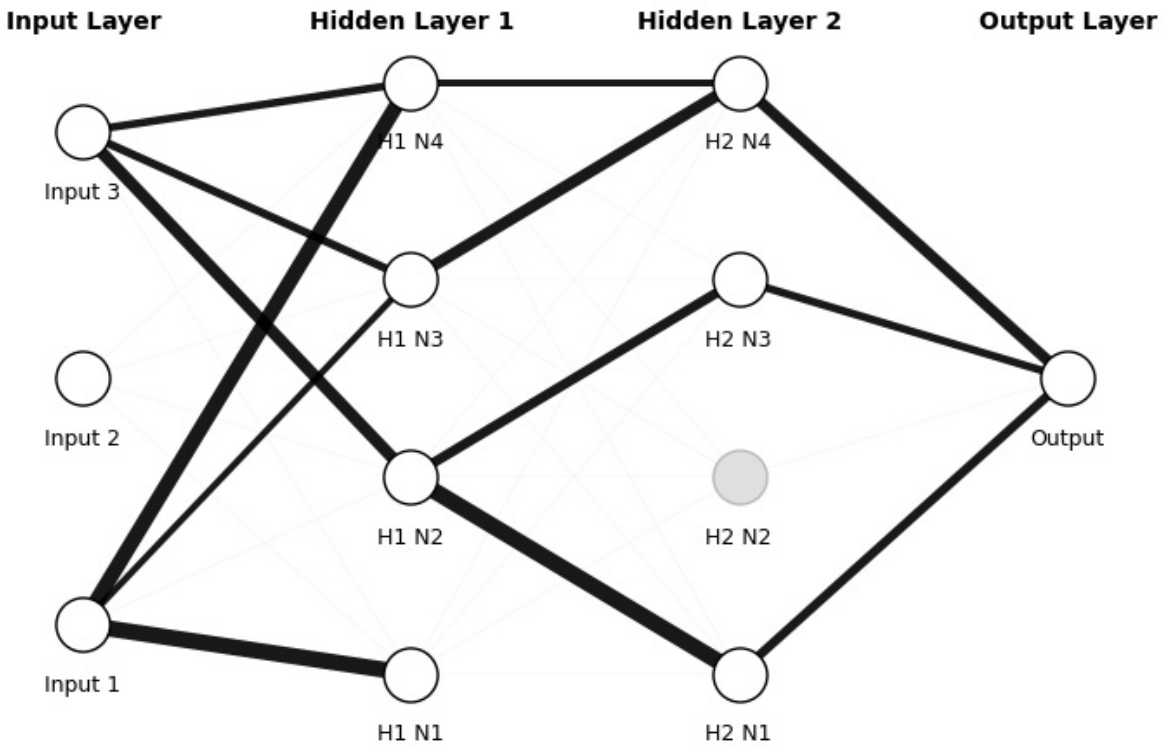

Figure 1: A Sieve KAN With Active Edges Highlighted

Using r_n, we define our total parameters in the KAN as $p_n := O(r_n G_n)$ as in Liu et al. (2024) where $p = O(LW^2G)$.

The fact that the sieve KAN learns activation functions instead of weight matrices is what allows us to fix W_n across hidden layers in the first place, attempting to do so with MLP structures[27] is incompatible with our sieve estimation approach. Under a hypothetical framework incorporating MLPs with sieve estimation, ensuring that the MLP simultaneously possesses a bounded entropy and a degree of approximation power sufficient for high-dimensional estimation is an extremely challenging task. This is due to how the approximation power of an MLP is tied to its width and depth, which are typically required to grow with the sample size in order to achieve convergence rates comparable to the aforementioned minimax rates in nonparametric regression. As a result of this inherent dependency, it is difficult to control the metric entropy of such a sieve without inordinately restricting the approximation

[27]See Farrell et al. (2021) for a discussion on this class of "fixed width" networks, which are especially likely to struggle in high-dimensions.

capabilities of the MLP.

To the contrary, due to how the expressiveness KANs are instead linked to the flexibility of the splines (i.e., is capable of being increased by simply expanding the grid size), this architecture guarantees the controlled growth of the sieve space. In these structures, the width determines the number of interactions between input variables and by extension, the diversity of interactions within a given layer. This makes KANs well-suited for sieve estimation in high-dimensions, in addition to how they do not suffer from the black box nature of MLPs. Likewise, stacking additional layers in a KAN permits for richer representations of f_0. As Schmidt-Hieber (2020) points out, allowing the length of deep networks to grow concurrently with the sample size is intuitive. Similarly, the concluding remarks of Hornik et al. (1989) addresses the natural connection between multilayer feedforward networks and the method of sieves. KAN architectures significantly contribute to this framework that harmonizes DNNs with a method of sieves estimation approach for nonparametric regression.

Let $\Theta \equiv A \times \mathcal{F}_d^{m+1}$ be the parameter space where $A \subset \mathbb{R}^a$: a compact set representing the finite-dimensional space of our linear parameters of interest and Θ is equipped with the pseudometric $\|\cdot\|$. The sieve space is given by $\Theta_n \equiv A \times \mathcal{K}_n$, where for a vector of network "weights" $\boldsymbol{\omega}_n \equiv (p_n, \{\boldsymbol{c}_j\}_{j=1}^{r_n})$, we define the sieve KAN class:

$$\mathcal{K}_n = \Bigg\{ f(\tilde{X}_t; \boldsymbol{\omega}_n) \mid f(\boldsymbol{x}) = \Phi_{L_n} \circ \big(\Phi_{L_n - 1} + h\big) \circ \cdots \circ \big(\Phi_1 + h\big) \circ \Phi_0\big(\tilde{X}_t\big), \\ \boldsymbol{\omega}_n \equiv \big(p_n, \{c_{j,k}\}_{k=1,j=1}^{G_n, r_n}\big),\ \sum_{j=1}^{r_n} \|\boldsymbol{c}_j\|_2 \leq \Delta_n \Bigg\} \tag{7}$$

Here, $\boldsymbol{\omega}_n$ consists of the network architecture parameters[28] $p_n = O(L_n W_n^2 G_n)$ and a collection of all B-spline coefficients across the r_n potential edges within the KAN. We note that it is possible for activation functions with non-zero coefficients that do not belong to the active paths (i.e., edges disconnected from the network output) to be pruned. See Figure 1 for

[28]More specifically, $p_n = \sum_{l=0}^{L_n - 1} W_l W_{l+1}(G_n + k - 1) \approx L_n W_n^2 G_n$, where k represents the B-spline order (degree $k - 1$).

an example of such a function, corresponding to the first input edge. In practice, we only prune the $(r_n - s_n)$ inactive edges to highlight the robustness of our Group Lasso ($\ell_{1,2}$-regularization) approach centered around $R_n(\theta) \coloneqq \sum_{j=1}^{r_n} \|\boldsymbol{c}_j\|_2 \equiv \|\boldsymbol{c}\|_{1,2}$ (the regularizer term introduced earlier, where $R_n(\theta) \leq \Delta_n$); as the vast majority of active edges after training contribute directly to the network output. We define the following adaptive threshold for what constitutes an active edge: Δ_n / r_n, which corresponds to a baseline network where coefficients are evenly distributed across all potential edges. This addresses the potential issues associated with arbitrary fixed thresholds commonly applied in KAN architectures, particularly in the presence of many covariates.

In order to ensure the domain is bounded[29] between $[0, 1]$, a CDF normalization is applied to inputs $X_t \coloneqq [X_{t,1}, \dots X_{t,d}]' \in \mathbb{R}^d$ such that:

$$\tilde{X}_{t,j} = \hat{F}_j(X_{t,j}), \quad j = 1, \dots, d$$
$$\hat{F}_j(z) \coloneqq \frac{1}{n} \sum_{t=1}^{n} \mathbf{1}\{X_{t,j} \leq z\}$$

Where $\hat{F}_n^j(\cdot) \equiv \hat{F}_j(\cdot)$ is the (marginal) empirical distribution function. By Theorem 19.3 of van der Vaart (1998) and Lemma 1 of Chen and Shen (1998), this introduces an approximation error of $O(n^{-1/2})$: a term negligible with respect to the resulting KAN sieve extremum approximation error (**Theorem 1**). Although commonly applied in empirical finance applications (e.g., copula models), the use of CDF normalization of inputs in deep learning architectures remains underexplored. Employing this normalization in the place of commonly used standardization techniques facilitates the use of a fixed spline grid $[-\epsilon, 1+\epsilon]$, for a small $\epsilon > 0$ padding term.

The learnable activation functions $\phi(\cdot)$ can include a basis function $b(x)$ in order to provide increased stability as part of our network initialization, as outlined in Liu et al. (2024). During training, the initial activation function would take on a weighted sum of the

[29] e.g., as required by the KART.

basis function and the B-spline function as follows:

$$\phi(x) = w_b b(x) + w_s \text{spline}(x) \tag{8}$$

$$b(x) = \text{silu}(x) = x/(1 + e^{-x}), \quad \text{spline}(x) = \sum_{k=1}^{G_n} c_k B_k(x) \tag{9}$$

where the weights (w_b, w_s) are updated during training in addition to the learned B-spline coefficients: with $w_b \to 0$ and $w_s \to 1$ early on in the training process, so that the resulting network derives its approximation capabilities purely from B-splines. Note that the basis function $b(x)$ is simply a combination of the ReLU and sigmoid activation functions (resulting in the "sigmoid linear unit", or "SiLU" activation). See the Appendix for a discussion on the useful properties of this "hybrid" activation function with respect to gradient stabilization in the context of (single hidden layer) sieve networks.

We find that across Monte Carlo simulations, this optimization approach occasionally leads to a persistent degree overfitting, one that cannot be addressed adequately by $\lambda_n R_n(\theta)$ once the transition to $w_s = 1$ is complete. To avoid this, we instead introduce residual connections in the form of the identity mapping[30] $h : \mathbb{R}^{W_n^{l-1}} \mapsto \mathbb{R}^{W_n^l}$, throughout the $l = 1, \ldots L_n - 1$ hidden layers. The residual connection $h(x) = x$ passes the input from the previous hidden layer, improving stability during training by mitigating the vanishing gradient problem[31] in deep networks similar to the ReLU/SiLU activation functions. For this reason, the implementation of residual connections is common in many deep MLP architectures (inspired by He et al., 2016) and the growing literature on KANs (e.g., Yu et al., 2024). Furthermore, the presence of this identity mapping eliminates the necessity for carefully selecting appropriate spline initializations, allowing for a wide range of initialization techniques to be applied in practice. Given the relatively aggressive regularization approach of our framework, ensuring gradient stability is crucial with respect to learning the relevant true $\phi(x) = \sum_{k=1}^{G_n} c_k B_k(x)$ edge functions.

[30] As denoted in the definition of the sieve KAN class $\mathcal{K}_n$ earlier.

[31] Through bounding the gradient magnitude away from zero.

The B-spline functions $B_{j,k}(x)$ are defined recursively[32]: for order k and a (non-decreasing) sequence of knot point t_j, the B-spline $B_{j,k}(x)$ is given by:

$$B_{j,k}(x) = \frac{x - t_j}{t_{j+k} - t_p} B_{j,k-1}(x) + \frac{t_{j+k+1} - x}{t_{j+k+1} - t_{j+1}} B_{j+1,k-1}(x)$$

with the base case for order $k = 1$ defined as:

$$B_{j,1}(x) = \begin{cases} 1 & \text{if } t_j \leq x < t_{j+1} \\ 0 & \text{otherwise} \end{cases}$$

In order to simplify the analysis in our key theorems/results, we fix the polynomial degree of our splines as order $k = 4$ (degree 3), with G_n intervals accordingly ($G_n + 1$ knot points in total: $t_0 < t_1 < ... < t_{G_n}$). We note that in practice, KANs typically use cubic splines for a wide range of general applications, so this formalization does not incur additional costs or tradeoffs beyond in certain settings outside of the scope of nonparametric/semiparametric regression (e.g., symbolic regression). Furthermore, our theoretical results extend to any $k \geq m + 1$, although the stability of higher order B-splines in this setting remains untested.

2.3 Key Parameters and Model Assumptions

For cases where A is a nonempty set (e.g., in semiparametric mean regression, where $\alpha_0 \in A$ is the coefficient for the linear component), we have: $\theta_0 \equiv (\alpha_0, f_0) \in \Theta$ and $\theta_n \equiv (\alpha_n, f(\cdot; \boldsymbol{\omega}_n)) = (\alpha_n, \pi_n f_0) \in \Theta_n$, where π_n is a projection operator representing the best possible approximation of f_0 in the sieve space. Specifically, π_n is the orthogonal projection under the L^2-norm which ensures that the approximation is the unique element in the sieve space closest to the true θ_0. Under the pseudometric[33] $\|\cdot\|$ on Θ, we define Θ_n as dense in

[32]This definition is commonly referred to as the Cox-de Boor recursion formula: de Boor (1978).

[33]Recall that a pseudometric satisfies the axioms of a metric with the exception that $\|\theta_1 - \theta_2\|$ does not necessarily imply that $\theta_1 = \theta_2$. In the method of sieve literature, $\|\cdot\|$ is typically used interchangeably with distance $d(\cdot, \cdot)$.

Θ: for any $\theta \in \Theta$, there exists a $\pi_n\theta \in \Theta_n$ such that $\|\theta - \pi_n\theta\| \to 0$ as $n \to \infty$. This ensures that the approximation error induced by the sieve space Θ_n becomes asymptotically negligible relative to the estimation error. Furthermore, we define $\pi_n\theta_0 \equiv \inf_{\theta\in\Theta_n}\|\theta - \theta_0\|$. Let $E[\cdot]$ denote the expectation under measure P_0 (the same applies to $Var[\cdot]$ and $Cov[\cdot]$) henceforth.

We define our KAN parameters $p_n \equiv (L_n, W_n, G_n)$ as follows:

$$L_n \asymp O(log\ n), \quad W_n \asymp s_0 \asymp O(\log n), \quad G_n \asymp n^{\gamma} \tag{10}$$

$$\Delta_n \asymp O(n^{\zeta_\delta}), \quad \zeta_\delta > 0, \quad \frac{1}{16} < \gamma < 1 \tag{11}$$

where γ is a parameter required to guarantee sufficient approximation power in order to deliver our $o_P(n^{-1/4})$ convergence rate. With respect to selecting a suitable width such that $W_n \geq s_0$, the use of CV is practical and consistent with existing deep learning approaches. In theory, overparameterization in any of the parameters p_n does not harm the model performance due to the adaptability of our sieve space and regularization approach. We verify this using our sparsity ratio (defined in terms of the Δ_n/r_n threshold for active edges) in the simulation analysis across many DGPs, defining $W_n = C_w \log n$ for some constant $C_W > 0$ and evaluating a wide range of potential values. In the case of $s_0 = O(1)$, we simply let $W_n = C_W$ and focus our attention on the case above in proofs involving the metric entropy of this KAN architecture. The requirement that the width of the sieve KAN grows at a rate equivalent to that of s_0 arises from our sparse compositional structure on $\theta_0 \in \mathcal{F}_d^{m+1}$ (Assumption **K.3(a)** below). This ensures that we have enough potential univariate functions to approximate the active edge functions in θ_0, even as $n \to \infty$. For simplicity, we set $L_n = \lfloor \log n \rfloor$ in practice.

Sieve KAN assumptions:

K.1(a): The strictly stationary process $\{Z_t\}_{t=1}^n = (Y_t, X_t')'$ is i.i.d or m-dependent (for $m \in \mathbb{N}$), or

K.1(b): β-mixing stationary with $\beta(j) \leq \beta_0 j^{-\xi}, \xi > 2$ for some $\beta_0 > 0,\ \xi > 2$, or

K.1(c): ϕ-mixing (uniform mixing) stationary with $\psi(j) \leq \psi_0 j^{-\xi}$ for some $\psi_0 > 0,\ \xi > 2$.

K.2(a): The parameter space Θ is equipped with the pseudometric $\|\cdot\|$, Θ_n (with network weights ω_n defined earlier) is dense in Θ and $\mathcal{F}_d^{m+1} \equiv W_2^{m+1}(\mu)$ is a Sobolev space. Furthermore, restrict $G_n \geq 2$.

K.2(b): The quasi-log-likelihood function $l(Z_t, \theta)$ satisfies:

$$\sup_{\theta \in \Theta_n, \|\theta - \theta_0\| \leq \epsilon} \text{Var}[l(Z_t, \theta_0) - l(Z_t, \theta)] \geq C_1 \epsilon^{2\rho}, \quad \text{for all small } \epsilon > 0$$

where $C_1 > 0$ and $\rho = 1$ for quadratic losses.

K.2(c): (Entropy) There exists a function $U(\cdot)$ and some $\nu > 0$ such that for any $\delta > 0$:

$$\sup_{\theta \in \Theta_n, \|\theta - \theta_0\| \leq \delta} |l(\theta) - l(\theta_0)| \leq \delta^\nu U(Z_t)$$

Our initial set of assumptions are standard in the literature on sieve extremum estimators. Assumption **K.1** permits data dependence structures beyond the baseline i.i.d case used to derive the entropy bounds of DNNs. Since m-dependence is generally an unrealistic assumption for economic and financial time series[34], the β-mixing and uniform mixing assumptions (see the Appendix for full definitions) are necessary for controlling serial dependence in the context of empirical processes. The margin condition in Assumption **K.2(b)** guarantees that θ_0 is identifiable and that $l(\cdot)$ is well-behaved in a neighborhood surrounding it. The size of these neighborhoods in Θ_n is dictated by the term δ^ν in Assumption **K.2(c)**, thus preventing the sieve from becoming overly fine.

Sieve KAN assumptions (continued):

K.3(a): (Sparse Compositional Structure) The target function $f_0 \in \mathcal{F}_d^{m+1} \equiv W_2^{m+1},\ f_0 :$

[34]To see why this is the case, consider a Moving Average process of order m (denoted $MA(m)$) with a sequence of i.i.d shocks ϵ_t defined by $X_t = \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots + \theta_n \epsilon_{t-m}$. For a finite m, this implies that the autocorrelation of this process is zero for all lags greater than m, thus ruling out the long-term persistence common in macroeconomic shocks, to give an example.

$[0,1]^d \to \mathbb{R}$ takes on the following compositional form:

$$f_0(X_t) = \sum_{i=1}^{s_0} \Phi_i \left(\sum_{j \in \mathcal{J}_i} \phi_{ij}(X_{t,j}) \right)$$

with $|\mathcal{J}_i| \ll d$, and $d = O(n^{\zeta_d})$ for some $\zeta_d > 0$.

K.3(b): (Sparsity Growth) For the $s_n \geq s_0$ corresponding to the best approximation $\pi_n \theta_0$:

i. $s_0 = O(1)$, or

ii. $s_0 = O(\log n)$

K.4(a): Let $\mathcal{I}_\epsilon = [-\epsilon, 1+\epsilon]$ for some $\epsilon > 0$. Each univariate edge function $\phi_{i,j}, \Phi_i$ in the representation of f_0 belongs to the Sobolev space $W_2^{m+1}(\mathcal{I}_\epsilon)$ with a uniformly bounded norm (given a constant $B < \infty$).

K.4(b): The univariate edge functions $\phi_{i,j}, \Phi_i$ and their derivatives (up to order $m+1$) are Lipschitz continuous (with uniform constant $L < \infty$).

The second set of assumptions focuses on the structural properties of the target function and the regularity of its components. Assumption **K.3(a)** formalizes the sparse compositional structure that allows the KAN sieve extremum to bypass the CoD. The main advantage of our deep architecture is in the use of additional layers to achieve more optimal representations of the outer functions Φ_i. As a result, our estimator benefits from the improved performance of deeper representations (as discussed in Liu et al. 2024 in comparison to shallow KANs) and the metric entropy is controlled through sparsity. This is a key focus in our Monte Carlo simulations with DGPs that involve complex, compositional functions. Assumption **K.3(b)** constrains s_0 to be either constant or at most logarithmic in the sample size. This is consistent with the sieve KAN parameters, as the architecture size (governed by W_n and L_n) also grow logarithmically in n. As emphasized throughout this study, the expressiveness of KANs is uniquely tied to the grid resolution (G_n). A slowly growing architecture size facilitates the freedom[35] to select a sufficiently large grid (through $\frac{1}{16} < \gamma < 1$)

[35]Which in general, is crucial for network parameters selected via CV.

and B-spline coefficient constraint (Δ_n, through $\zeta_\delta > 0$). Lastly, Assumptions **K.4(a)** and **K.4(b)** are high-level assumptions that impose the necessary regularity on the univariate edge functions. The Lipschitz continuity of the derivatives further ensures that the sieve extremum objective is stable[36].

Under these assumptions, the KAN sieve extremum estimator delivers the convergence rate: $\|\hat{\theta}_n - \theta_0\| = o_P(n^{-1/4})$. We note that while this estimator does not suffer from the COD as demonstrated in the proofs, d is naturally still relevant to our network parameters through s_0 and the subset $\mathcal{J}_i$. This is despite how estimation methods relying on splines are notorious for failing to produce precise estimates (Chen, 2007) in high-dimensional settings due to the COD. Although there are existing spline estimation approaches that attempt to exploit sparsity in high-dimensional environments[37] similar to our proposed KAN sieve estimator, these methods are likely to perform poorly on the truly compositional structures (i.e., where imposing additivity is insufficient) found in economic and financial applications.

In practice, we do not apply entropy regularization, a method that penalizes the average magnitude of the activation functions commonly applied in KANs (see the approach outlined in Liu et al., 2024, Appendix C therein). While this may assist in reducing the number of redundant functions in general KAN architectures (in settings where $n \gg d$), this form of regularization effectively distributes the task of learning the structure of f_0 across the many potential edges. This ultimately results in dense networks that will fail to prevent the explosion of the metric entropy in high-dimensions, unlike under structural sparsity. Instead, our constraint Δ_n promotes the opposite[38], forcing the network to use each active edge s_n as efficiently as possible *without* sacrificing approximation power.

[36] Aiding in the mitigation of the aforementioned exploding gradients problem, as well as functional oscillations that can occur in unconstrained KANs.

[37] e.g., Friedman (1991), Stone (1985).

[38] In addition to penalizing entire vectors of B-spline coefficients across edges under our Group Lasso method.

3 Theoretical Results

To simplify the notation, we assume $\theta_0 \equiv f_0 \in \Theta$ and $\theta_n \equiv f(\cdot; \boldsymbol{\omega}_n) \in \Theta_n$ (i.e., A is an empty set) for Theorems 1-3. A potential extension of the result from Chen and Shen (1998) concerning linear plug-in estimators is discussed in the following section. Proofs of each of the following theorems can be found in the Appendix, in addition to a restated key result from Chen and Shen (1998). WLOG, let $\mu \equiv \tilde{\mu}$ as defined in the previous section.

3.1 Convergence Rates

Theorem 1 (Sieve KAN Approximation Error): Under Assumptions **K.1-K.4**, there exists a sieve approximation $\pi_n\theta_0 \in \mathcal{K}_n$ such that under $G_n \asymp n^{\gamma}$ and given cubic B-splines activation functions of fixed interval length (with uniform knots):

$$\|\pi_n\theta_0 - \theta_0\| \leq Cs_0G_n^{-k} + C's_0O(n^{-1/2}) = o(n^{-1/4})$$

The proof of **Theorem 1** relies on the sparse compositional structure imposed on the target function and the smooth univariate edge functions (Assumptions **K.3** and **K.4**). Using results from spline approximation theory (DeVore and Lorentz, 1993), we show that each edge function is approximated with error $O(G_n^{-k})$. The resulting approximation function of the sieve KAN is independent of d, a key result for mitigating the CoD. We note that due to the results from approximation theory applied in the proof of this theorem, we cannot apply the adaptive grid approach of Liu et al. (2024). However, due to our optimization approach that relies on CDF normalization, this is not problematic. As discussed earlier, freedom to choose a sufficiently large $\frac{1}{16} < \gamma < 1$ in addition to the constraint Δ_n on the spline coefficients leads to a similar result in practice.

Corollary 1: Under the same conditions as **Theorem 1**, given $\frac{1}{4} < \gamma < 1$ and assuming

$m \leq 3$:

$$\|\pi_n\theta_0 - \theta_0\|_{W_2^{m+1}(\mu)} \leq C_k s_0 G_n^{-1} + C' s_0 O(n^{-1/2}) = o(n^{-1/4})$$

Theorem 2 (Sieve KAN Entropy Bound): Under Assumptions **K.1-K.3**, and given the sieve space $\mathcal{K}_n$:

$$\log \mathcal{N}(\epsilon, \mathcal{K}_n, \|\cdot\|_{L_2(\mu)}) \leq C_1 s_n \log\left(\frac{r_n}{s_n}\right) + C_2 s_n G_n \log\left(\frac{\Delta_n}{\epsilon}\right)$$

In this proof, the metric entropy bound is broken down into two components. The first, $s_n \log(r_n/s_n)$, is a combinatorial term (by Lemma 17.5.1 of Cover and Thomas, 1991; where $p = s_n/r_n$. A similar argument appears in Raskutti et al., 2011) arising from the active edge selection which is central to our sparse compositional form assumption. The B-spline complexity term $s_n G_n \log(\Delta_n/\epsilon)$ arises via the $\ell_{1,2}$-constraint (the Group Lasso norm) imposed on the spline coefficients (by Corollary 4.2.11 of Vershynin, 2018), which encourages sparsity across the r_n potential edges. This highlights the role of Δ_n in controlling the overall complexity of $\mathcal{K}_n$ in the sieve KAN architecture and the dependence on effective parameters ($s_n G_n$) only, crucial for high-dimensional estimation.

Theorem 3 (Sieve KAN Estimation Error): Under Assumptions **K.1-K.3**, given the sieve space $\mathcal{K}_n$ and letting $\frac{1}{16} < \gamma < 1$:

$$\|\hat{\theta}_n - \pi_n\theta_0\| = o_P(n^{-1/4})$$

The proof of **Theorem 3** involves splitting the entropy integral at $\epsilon_0 = n^{-1}$ at showing that each term is $o_P(n^{-1/4})$. Obtaining convergence rates in the Sobolev norm would lead to a condition that $m > d$ as we show in the Appendix, which would re-introduce the CoD. Instead, the estimation error achieves a convergence rate of $o_P(n^{-1/4})$ which holds uniformly over $d = O(n^{\zeta_d})$. Combining this result with the result from **Theorem 1** earlier, we have:

Corollary 2 (Sieve KAN Convergence Rate): Under Assumptions **K.1-K.4**, the KAN sieve extremum estimator achieves $\|\hat{\theta}_n - \theta_0\| = o_P(n^{-1/4})$.

To see this, note that by the triangle inequality:

$$\|\hat{\theta}_n - \theta_0\| \leq \|\hat{\theta}_n - \pi_n\theta_0\| + \|\pi_n\theta_0 - \theta_0\|$$
$$\|\hat{\theta}_n - \theta_0\| = o_P(n^{-1/4}) + o(n^{-1/4}) = o_P(n^{-1/4}) \tag{12}$$

Remark: We note that by letting $\omega_n \to \infty$ and $\Delta_n \to \infty$ as $n \to \infty$, we obtain a sequence of increasingly flexible network architectures; making this analogous to the result from White (1990) for KANs under sparsity, a foundational result originally in terms of universal approximator MLPs. In establishing the consistency of sieve aNN estimators, White (1990) demonstrates how the approximation power of the network increases in tandem with the "experience" of the aNN[39] while controlling network complexity sufficiently well. In the context of sparse compositional target functions, our framework represents a class of *connectionist sieve networks* with multiple hidden layers that guarantees a dimension-free convergence rate, the first of its kind.

In **Theorem 4**, we establish the asymptotic normality of plug-in estimators using sieve KANs. Following Chen and Shen (1998, Theorem 2), we consider a linear functional $\Gamma : \Theta \to \mathbb{R}$ with $v^* \in \bar{V}$ (by the Riesz representation theorem) satisfying $\Gamma'_{\theta_0} = \langle \theta - \theta_0, v^* \rangle$ for all $\theta \in \Theta$. The existence of v^*, the Gateaux derivative of Γ, ensures that the functional is bounded (where $\bar{V}$ denotes the completion of the space spanned by $\Theta - \{\theta_0\}$ and the inner product $\langle \cdot, \cdot \rangle$ is induced by $\|\cdot\|$) and identifies the direction in the sieve space that characterizes the asymptotic variability of the estimator. We expand on these definitions and list the relevant conditions from Chen and Shen (1998) in the Appendix, before verifying each condition for the sieve KAN.

Theorem 4 (Asymptotic Normality): Under Assumptions **K.1-K.4** and **B.1-B.5** (in

[39] Referring to the sample size of the dataset the network is trained on.

the Appendix), the plug-in sieve KAN estimator satisfies:

$$\sqrt{n}\Big(\Gamma(\hat{\theta}_n) - \Gamma(\theta_0)\Big) \rightsquigarrow N(0, \sigma_*^2) \tag{13}$$

$$\sigma_*^2 = \lim_{n\to\infty} n^{-1} Var\Big(\sum_{t=1}^{n} l'_{\theta_0}[v^*, Z_t]\Big) \tag{14}$$

noting that under Assumption K.1: $\sigma_*^2 = Var\big(l'_{\theta_0}[v^*, Y_1]\big) + 2\sum_{j=2}^{\infty} Cov\big(l'_{\theta_0}[v^*, Z_1], l'_{\theta_0}[v^*, Z_t]\big)$. In ultra-high-dimensional settings ($d \gg n$), estimates of this linear component (e.g., the parameter of interest in partially linear regression models) are likely to suffer from regularization bias, leading us to consider an alternative approach centered around DML, designed to address this.

3.2 Double/Debiased Machine Learning

In this section, we show how our KAN sieve extremum estimator can be applied within the DML framework of Chernozhukov et al. (2018) to obtain $\sqrt{n}$-consistent and asymptotic normal estimates of linear parameters of interest. A similar DML application appears in Farrell et al. (2025), although the convergence rate of the ReLU DNN they propose[40] explicitly depends on the heterogeneity dimension (d_c). In addition to the presence of the CoD, their near-minimax $O(n^{-m/(m+d_c)} \log^8 n)$ convergence rate depends on the smoothness of the target function; highlighting the tradeoff arising from the very useful economic interpretability of their proposed estimator. To the best of our knowledge, the sieve KAN is the first and only network architecture since Chen and White (1999) capable of delivering a dimension-free convergence rate of $o_P(n^{-1/4})$.

Chernozhukov et al. (2018) address how the naive application of machine learning algorithms in semiparametric models with low-dimensional causal parameters can lead to biased estimates. This arises from the regularization methods commonly employed in these models for nuisance function estimation, which perform well by reducing variance (overfitting) in a

[40] Which unlike Farrell et al. (2021), takes a quasi-likelihood approach.

manner consistent with the standard variance-bias tradeoff. As a result of this overfitting and regularization bias, estimates of the causal parameter obtained by using the fitted nuisance function as a plug-in estimator will suffer from a heavy degree of bias. Furthermore, they demonstrate how the Donsker conditions typically imposed in semiparametric models to control complexity break down in ultra-high-dimensional settings. The DML framework resolves these issues through two key ingredients: Neyman orthogonality and sample splitting.

We consider the partially linear regression model[41] (Robinson, 1988) in our simulations and assume $Z_i = \{Y_i, X_i\} = \{Y_i, D_i, W_i\}$ is i.i.d for this application. This model is defined in terms of the following equations:

$$Y_i = D_i\alpha_0 + g_0(W_i) + U_i, \qquad E[U_i|W_i, D_i] = 0 \tag{15}$$

$$D_i = m_0(W_i) + V_i, \qquad E[V_i|W_i] = 0 \tag{16}$$

where the sample splitting procedure (in the case of two folds) entails estimating m_0 using half of the dataset and then estimating g_0 with the remaining half. In this model, D_i represents the treatment variable (or more generally, the primary regressor of interest) with a linear causal effect α_0 estimated after controlling for the confounding effects of W_i. Neyman orthogonality is the condition that ensures moment conditions defining α_0 are insensitive to perturbations[42] in the nuisance functions $\eta_0 = (g_0, m_0)$.

Definition 2 (Neyman Orthogonality): The score $\psi(\cdot;\cdot)$ satisfies Neyman orthogonality if:

$$\partial_\eta E[\psi(Z; \alpha_0, \eta_0)][\eta - \eta_0] = 0 \quad \text{for all } \eta \in \mathrm{T}_n$$

The main advantage of our procedure estimating nuisance functions using sieve KANs over widely used machine learning methods (e.g., Lasso, random forests) is how it is capable of capturing complex compositional structures[43] in the data-generating process (DGP). In their

[41] Which is the leading example in Chernozhukov et al. (2018); see our partition of X_i defined earlier.
[42] With respect to errors in estimating both the outcome model and the treatment/propensity model.
[43] Nonlinear or otherwise.

empirical example, Chernozhukov et al. (2018) discuss how "Deep Learning methods" were experimented with but that the results were omitted due to computational and stability issues (presumably related to the aforementioned persistent issues involving gradient stability) encountered. We observe no such issues across all of our Monte Carlo simulations and compare the results against those of a Lasso model using a penalty λ_L selected via cross-validation. By Theorem 4.1 of Chernozhukov et al. (2018), the following result holds when σ^2 is replaced with $\hat{\sigma}^2$.

Corollary 3 (DML Estimation): Under Assumptions **K.1-K.4**, the DML estimator $\hat{\alpha}$ where $(\hat{g}_n, \hat{m}_n) \in \mathcal{K}_n$ satisfies:

$$\sigma^{-1}\sqrt{n}(\hat{\alpha} - \alpha_0) \rightsquigarrow N(0,1), \quad \sigma^2 = (E[V^2])^{-1}E[V^2U^2](E[V^2])^{-1}$$

4 Results

In this section, we analyze the performance of the sieve KAN in Monte Carlo simulations and an empirical example involving macro-finance factors. In our simulations, we first consider a simple additive nonparametric regression model in a moderately high-dimensional setting. We then evaluate our proposed estimator in a DML application (using the PLR model outlined earlier) against the performance of other ML methods. Both involve running $N = 1000$ total simulations and evaluating the aggregate results, with careful attention paid to the crucial sparsity ratio. This is the ratio of inactive edges relative to the total number of potential edges (r_n), using the Δ_n/r_n threshold for what constitutes an active edge (i.e., learned univariate functions with a magnitude greater than those in a hypothetical fully dense network) defined earlier. We also report results obtained using the sieve network from Chen and White (1999) denoted by "SLFN", which provides a shallow network model comparison for our sieve KAN.

4.1 Monte Carlo Simulations

In our baseline nonparametric regression, we evaluate ability of the sieve KAN to recover a sparse additive signal with a target function dependent on $S = 30$ out of $d = 100$ total covariates and $n = 5000$ observations. The sieve KAN parameters are set consistent with the ones outlined in Section 2.3, where we manually select $\gamma = 0.35$.

$$f_0(X) = \sum_{j=1}^{S} \sin(X_j) + \epsilon, \quad \epsilon \sim N(0, 0.25) \tag{17}$$

As seen in Table 1, the sieve KAN achieves a Test R^2 of 0.78 and a Test RMSE of 1.87, where both metrics are comparable to the performance on the training data and significantly outperforming the SLFN. The gap in performance of these models is indicative of the superior approximation power possessed by the sieve KAN, demonstrating how the learned B-spline edges directly adapt to the smoothness and periodicity of the target function. With respect to performance on unseen data, we observe minimal overfitting of the sieve KAN (despite our novel and relatively aggressive regularization method) based on the differences between the aforementioned metrics. On the other hand, the SLFN reliant on a fixed activation function suffers from underfitting as it lacks the localized flexibility of the B-spline grid.

Table 1: Nonparametric Regression Results

Metric	**KAN**	**SLFN**
Train RMSE	1.66	2.34
Train R^2	0.83	0.66
Test RMSE	1.87	2.39
Test R^2	0.78	0.64
Runtime (s)	2513.88	0.33
Sparsity (%)	97.61	N/A
Model Size (Active Parameters)	4339.34	213.0

The most critical finding in this setting is the effective dimensionality reduction achieved through the Group Lasso penalization. Out of the many r_n potential edges, the sieve KAN

achieves a sparsity ratio of 97.61%. This means that the network successfully identified the narrow subset of active edges required to represent the $S = 30$ relevant covariates, resulting in $s_n = 4339$ active (unpruned) univariate functions. This prevents the overfitting noise common in insufficiently penalized deep networks and is crucial for maintaining a controlled metric entropy as discussed throughout this paper. The relatively higher computational runtime (2514 seconds) is consistent with the tradeoff associated with deeper networks, as the sieve KAN demonstrates significantly higher statistical efficiency. These results verify that even in cases where the true model is additive rather than deeply compositional (e.g., in our next simulation setup), the KAN sieve extremum estimator is robust and highly accurate; with an underlying architecture that is naturally robust to the CoD.

In our DML simulations, we compare the performance of sieve KANs against Lasso in estimating $\alpha_0 = 1.5$. Across each DGP, we generate $d = 100$ covariates in total with $S = 30$ relevant covariates and set use five folds in the sample spliting procedure. After implementing this split for each fold and an additional split to obtain the validation set, we are left with a training sample size of $n_{\text{train}} = 360$ observations for the first two DGPs and $n_{\text{train}} = 1440$ for the DGP #3. The validation set serves an important purpose in making sure our fitted models generalize well to unseen data and we report these loss metrics. We fix the sieve KAN parameter determining the grid size as $\gamma = 0.5$ for all simulations. For our main simulation DGP, we have a sieve KAN with $L_n = 6$ and $W_n = 201$. In each DGP, the true partially linear regression model is generated using the following nonlinear compositional functions and computed orthogonal residuals:

$$m_0(\boldsymbol{X}) = \sin\left(\sum_{j=1}^{S} \frac{1}{1+e^{-X_j}}\right) + 0.1\sum_{j=1}^{S} X_j^2 + \epsilon_m, \quad \epsilon_m \sim N(0, 0.25) \tag{18}$$

$$g_0(\boldsymbol{X}) = \frac{1}{1+e^{-\sum_{j=1}^{S}\sin(X_j)}} + 0.1\sum_{j=1}^{S} |X_j| + \epsilon_g, \qquad \epsilon_g \sim N(0, 0.25) \tag{19}$$

where $\hat{V} = D - \hat{m}(\boldsymbol{X})$, and $\hat{U} = Y - \hat{g}(\boldsymbol{X})$ are used to obtain $\hat{\alpha} = \frac{\sum \hat{V}_i(Y_i - \hat{g})}{\sum \hat{V}_i D_i}$. For our main

simulation DGP, we have a sieve KAN with $L_n = 6$ and $W_n = 201$ (KAN #1). We also consider an additional sieve KAN with $W_n = 101$ (KAN #2).

Table 2: DML Simulation Results: ($\alpha_0 = 1.5$)

Metric	**KAN #1**	**KAN #2**	**SLFN**	**Lasso**
Causal Parameter ($\hat{\alpha}$)				
$\hat{\alpha}$	1.5024	1.4804	1.6660	1.7318
RMSE	0.0657	0.0461	0.1672	0.2323
Runtime (s)	2302.98	1070.20	0.17	0.04
Nuisance Function Performance				
Train R^2 ($\hat{m}_0$ / $\hat{g}_0$)	0.80 / 0.82	0.84 / 0.85	0.09 / 0.07	0.18 / 0.16
Test R^2 ($\hat{m}_0$ / $\hat{g}_0$)	0.39 / 0.45	0.36 / 0.43	-0.04 / -0.06	-0.13 / -0.17
Sparsity (Δ_n) ($\hat{m}_0$ / $\hat{g}_0$)	0.925 / 0.929	0.888 / 0.890	NA / NA	NA / NA

Across the DML simulations, the sieve KAN demonstrates a strong ability to recover the true causal parameter $\alpha_0 = 1.5$ despite the complex compositional structure of the nuisance functions in the DGP. The wider sieve KAN (KAN #1 in Table 2) achieves a near-zero bias, yielding $\hat{\alpha} = 1.5024$ while the narrower architecture (KAN #2) yields $\hat{\alpha} = 1.4804$. In contrast, the SLFN and Lasso estimators exhibit substantial upward bias, with estimates of $\hat{\alpha} = 1.666$ and $\hat{\alpha} = 1.7318$, respectively. Despite the relatively strong performance of the SLFN in the earlier nonparametric regression (including its ability to generalize to unseen data, captured by the Test R^2), it fails to learn the compositional structures of m_0 and g_0. This highlights how shallow networks generally struggle to decompose complex hierarchical nonlinearities, similar to the the Lasso model which imposes a linear structure: resulting in a failure to eliminate regularization bias even under Neyman Orthogonality and sample-splitting. Conversely, both iterations of the sieve KAN display significant predictive power on the test set; confirming our theory that the learned B-spline activations effectively decomposes compositions into univariate sieve components, whereas the internal structure is unidentifiable to other architectures.

We observe that overparameterization in W_n does not degrade performance. Despite the

larger width of sieve KAN #1, it maintains high sparsity ratios, indicating the success of our regularization method combining Group Lasso with the constraint coefficient Δ_n. As emphasized throughout this paper and in our theoretical results, this structural sparsity is crucial for controlling the sieve KAN metric entropy in high-dimensions while preserving approximation power; thus preventing the explosion in complexity that standard MLPs are prone to. Furthermore, the stability of our estimator across different widths reinforces the robustness of the sieve grid resolution G_n. Even when the width is halved, the KAN sieve extremum estimator maintains strong performance and ability to generalize to unseen data. These results provide empirical evidence that the sieve KAN is a superior ML method for estimating nuisance functions in high-dimensional settings characterized by compositional covariate effects.

4.2 Empirical Application

We evaluate our proposed KAN sieve extremum estimator in an empirical application featuring a large set of covariates relative to the overall sample size. Rapach and Zhou (2021) demonstrate how principal component analysis (PCA) can be applied to 118 macroeconomic and financial variables in order to identify interpretable risk factors. The resulting components (e.g., housing, yields, credit spreads) are linear combinations of subsets of the raw variables obtained from the FRED-MD monthly database (1963-2024), which are then applied to estimate risk premia for a large set of test assets. Their findings suggests the existence of sparse linear relationships between the raw macro-finance variables, which we use (in addition to the aforementioned key sparse components) to evaluate our model against Lasso in a simple nonparametric regression. Given how this is the type of setting that the Lasso model traditionally excels in, this serves as a robustness test for the sieve KAN involving noisy time series financial data. We split the $n = 726$ observations using a $0.8 : 0.1 : 0.1$ train/test/validation ratio. Selecting $\gamma = 0.5$ and $\zeta_\delta = 0.9$, the sieve KAN parameters are given by: $L_n = 6$, $W_n = 243$, $G_n = 24$ and $\Delta_n = 307.91$.

Table 3: Sparse Macro-Finance Factors

Metric	**LASSO**	**SLFN**	**Sieve KAN**
Train RMSE	4.49	3.85	3.94
Train R^2	0.38	0.54	0.52
Test RMSE	4.70	4.53	4.75
Test R^2	0.52	0.56	0.51
Runtime (s)	1.85	58.11	534.46
Sparsity (%)	89.3	N/A	97.7

As shown in Table 2, the sieve KAN achieves a Test R^2 of 0.51, which is nearly identical to the that of the Lasso model. This is of significant as it provides empirical evidence of the robustness of our sieve extremum estimator in high-dimensional settings characterized by (approximate) linearity. In regimes involving linear or additive DGPs, overparameterized MLP architectures generalize poorly (an issue addressed by Schmidt-Hieber, 2020) due to the unconstrained parameter space. Conversely, the sieve KAN architecture effectively contracts or collapses to the linear approximation. Combined with our DML simulation results, this displays robustness to both extremes (with respect to the target function complexity) in its ability to generalize. Given this demonstrated robustness of our proposed estimator, the tradeoff resulting in increased computation time is justified. The Lasso model produces non-zero coefficient estimates for 12 covariates, while the sieve KAN demonstrates a sparsity ratio of 97.7%. By pruning a significant proportion of redundant edges, the sieve KAN minimizes the risk of capturing spurious nonlinearities; which is crucial for preserving consistency in finite-samples, especially when n is small relative to the total number of covariates as it is in this application.

5 Conclusion

This paper introduces the KAN sieve extremum estimator, a novel estimator that integrates the adaptive functional flexibility of KANs with the method of sieves. Our framework addresses a critical gap in the deep learning literature by providing an architecture that is

innately robust to the CoD under structural sparsity. This is achieved through the use of learnable B-spline activation functions and a novel Group Lasso regularization method.

The established theoretical results demonstrate that the sieve KAN maintains a controlled entropy under a sparse compositional assumption imposed on the target function. This enables consistent estimation even in high-dimensional settings, where the number of covariates is large relative to the sample size. Unlike standard MLPs, which often suffer from overparameterization and poor generalization in sparse environments, the sieve KAN effectively prunes redundant univariate functions to identify/recover the underlying signal of the target function.

Our Monte Carlo simulations and empirical application further substantiates these theoretical claims, with the sieve KAN consistently producing a high ratio of inactive edges relative to the total potential edges (across a wide range of architectural sizes, dictated by network lengths and widths). The sieve KAN performs especially well in DML simulations involving highly complex compositional nuisance functions relative to other ML methods. An extension integrating a sample-splitting method suitable for time series data is left to future work, although this is a very promising finding with respect to the rapidly-growing literature on DML.

In summary, the sieve KAN represents a tractable and statistically efficient alternative for nonparametric estimation. By bridging the gap between GAMs and deep compositional networks, it provides a robust and flexible approach to approximation that rigorously controls the complexity of the function space in high-dimensional settings.

References

[1] Barron, A., Birgé, L., & Massart, P. (1999). Risk bounds for model selection via penalization. *Probability Theory and Related Fields*, *113*(3), 301-413.

[2] Belloni, A., & Chernozhukov, V. (2011). L1-penalized quantile regression in high-dimensional sparse models. *Annals of Statistics*, *39*(1), 82–130.

[3] Chen, X. (2007). Large sample sieve estimation of semi-nonparametric models. In J. Heckman & E. Leamer (Eds.), *Handbook of Econometrics*, *6B*, 5549-5632. Elsevier.

[4] Chen, X., Linton, O., & Van Keilegom, I. (2003). Estimation of semiparametric models when the criterion function is not smooth. *Econometrica*, *71*(5), 1591-1608.

[5] Chen, X., & Shen, X. (1998). Sieve extremum estimates for weakly dependent data. *Econometrica*, *66*(2), 289-314.

[6] Chen, X., & White, H. (1999). Improved rates and asymptotic normality for nonparametric neural network estimators. *IEEE Transactions on Information Theory*, *45*(2), 682-691.

[7] Cover, T. M., & Thomas, J. A. (1999). *Elements of Information Theory*. John Wiley & Sons.

[8] De Boor, C. (1978). *A Practical Guide to Splines* (Vol. 27). Springer-Verlag, New York.

[9] DeVore, R. A. (1998). Nonlinear approximation. *Acta Numerica*, *7*, 51-150.

[10] Farrell, M. H., Liang, T., & Misra, S. (2021). Deep neural networks for estimation and inference. *Econometrica*, *89*(1), 181-213.

[11] Farrell, M. H., Liang, T., and Misra, S. (2025). Deep learning for individual heterogeneity. *arXiv preprint arXiv:2010.14694*.

[12] Friedman, J. H. (1991). Multivariate adaptive regression splines. *The Annals of Statistics*, *19*(1), 1-67.

[13] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

[14] Gourieroux, C., Monfort, A., & Trognon, A. (1984). Pseudo maximum likelihood methods: Theory. *Econometrica*, *52*(3), 681-700.

[15] Grenander, U. (1981). *Abstract Inference*. Wiley, New York.

[16] Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359-366.

[17] Hornik, K., Stinchcombe, M., & White, H. (1990). Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks. *Neural Networks*, *3*(5), 551-560.

[18] Koenker, R. (2005). *Quantile regression* (Vol. 38). Cambridge University Press.

[19] Kolmogorov, A. N. (1956). On the representation of continuous functions of several variables as superpositions of continuous functions of a smaller number of variables. *Dokl. Akad. Nauk*, *108*(2), 179-182.

[20] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436-444.

[21] Liu, Z., Wang, Y., Vaidya, S., Ruehle, F., Halverson, J., Soljačić, M., ... & Tegmark, M. (2024). KAN: Kolmogorov-Arnold networks. *arXiv preprint arXiv:2404.19756*.

[22] Meier, L., Van de Geer, S., & Bühlmann, P. (2008). The group lasso for logistic regression. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *70*(1), 53-71.

[23] Meier, L., Van de Geer, S., & Bühlmann, P. (2009). High-dimensional additive modeling. *The Annals of Statistics*, *37*(6B), 3730-3751.

[24] Mhaskar, H. N., & Poggio, T. (2016). Deep vs. shallow networks: An approximation theory perspective. *Analysis and Applications*, *14*(06), 829-848.

[25] Norets, A. (2012). Estimation of Dynamic Discrete Choice Models Using Artificial Neural Network Approximations. *Econometric Reviews*, 31(1), 84-106.

[26] Rapach, D., & Zhou, G. (2021). Sparse macro factors. *Available at SSRN 3259447*.

[27] Raskutti, G., Wainwright, M. J., & Yu, B. (2011). Minimax rates of estimation for high-dimensional linear regression over ℓ_q-balls. *IEEE Transactions on Information Theory*, *57*(10), 6976-6994.

[28] Robinson, P. M. (1988). Root-N-consistent semiparametric regression. *Econometrica*, *56*(4), 931-954.

[29] Schmidt-Hieber, J. (2020). Nonparametric regression using deep neural networks with ReLU activation function. *The Annals of Statistics*, *48*(4), 1851-1875.

[30] Schmidt-Hieber, J. (2021). The Kolmogorov–Arnold representation theorem revisited. *Neural Networks*, *137*, 119-126.

[31] Schumaker, L. (2007). *Spline Functions: Basic Theory*. Cambridge University Press.

[32] Semenova, V., Goldman, M., Chernozhukov, V., & Taddy, M. (2023). Inference on heterogeneous treatment effects in high-dimensional dynamic panels under weak dependence. *Quantitative Economics*, *14*(2), 471-510.

[33] Shen, X. (1997). On methods of sieves and penalization. *The Annals of Statistics*, *25*(6), 2555-2591.

[34] Shen, X., Jiang, C., Sakhanenko, L., & Lu, Q. (2023). Asymptotic properties of neural network sieve estimators. *Journal of Nonparametric Statistics*, *35*(4), 839-868.

[35] Sprecher, D. A. (1996). A numerical implementation of Kolmogorov's superpositions. *Neural Networks*, *9*(5), 765-772.

[36] Sprecher, D. A. (1997). A numerical implementation of Kolmogorov's superpositions II. *Neural Networks*, *10*(3), 447-457.

[37] Stone, C. J. (1982). Optimal global rates of convergence for nonparametric regression. *The Annals of Statistics*, *10*(4), 1040-1053.

[38] Stone, C. J. (1985). Additive regression and other nonparametric models. *The Annals of Statistics*, *13*(2), 689-705.

[39] Vershynin, R. (2018). *High-Dimensional Probability: An Introduction with Applications in Data Science* (Vol. 47). Cambridge University Press.

[40] White, H. (1990). Connectionist nonparametric regression: Multilayer feedforward networks can learn arbitrary mappings. *Neural Networks*, *3*(5), 535-549.

[41] White, H. (1992). Nonparametric estimation of conditional quantiles using neural networks. In C. Page & R. LePage (Eds.), *Computing Science and Statistics* (pp. 190-199). Springer.

[42] Yarotsky, D. (2017). Error bounds for approximations with deep ReLU networks. *Neural Networks*, *94*, 103-114.

[43] Yarotsky, D. (2018). Optimal approximation of continuous functions by very deep ReLU networks. In *Conference on Learning Theory* (COLT 2018) (pp. 639-649). PMLR.

[44] Yokoyama, R. (1980). Moment bounds for stationary mixing sequences. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 52(1), 45-57.

[45] Zhang, X., & Zhou, H. (2024). Generalization bounds and model complexity for Kolmogorov-Arnold networks. *arXiv preprint arXiv:2410.08026*.

A Appendix

A.1 Proofs of Main Results

Proof of Theorem 1: Under **K.3(a)**, θ_0 admits a sparse compositional structure with s_0 active edges given by ϕ_q and $\phi_{q,p}$ which are sufficiently smooth (**K.3(b)**) to be approximated by B-splines of order $k = 4$. Let $\delta_T = \max_{0 \leq j \leq N_j} t_{j+1} - t_j$ where the number of intervals is given by $N_j = G_n - k + 1$. By Theorem 7.3 of DeVore and Lorentz (1993) we obtain an approximation error of $O(G_n^{-4})$, since $\delta_T = O(G_n^{-1})$.

The constants arising from errors terms propagated across each layer[44] (Lipschitz constants) in addition to smoothness parameter m are absorbed by the constant $C > 0$. To see this, consider the following bound of the $L_2(\mu)$ error of the approximation of $\phi \in C^k$ (DeVore and Lorentz, 1993):

$$\|\phi - \pi_n \phi\| \leq C(\|\phi^k\|_\infty) G_n^{-k}$$

Since the overall approximation error accumulates across every active edge, it is bounded by $Cs_0 G_n^{-4} + C' s_0 O(n^{-1/2})$, where the $O(n^{-1/2})$ term arises from the EDF: $\|\hat{F} - F\|_\infty$ (by Theorem 19.3 of van der Vaart, 1998) uniformly over d. In the case of logarithmic sparsity growth where $s_0 = \log n$ and $W_n = C_W \log n$, we have

$$\|\pi_n \theta_0 - \theta_0\| = O(\log n \cdot n^{-4\gamma}) + O(\log n \cdot n^{-1/2})$$

For this approximation error term to be $o(n^{-1/4})$, we require $-4\gamma < -\frac{1}{4} \Rightarrow \gamma > \frac{1}{16}$ (the first of two key conditions on γ) guaranteeing the following limit

$$\lim_{n \to \infty} \frac{\log n \cdot n^{-4\gamma}}{n^{-1/4}} = 0$$

[44] By an argument identical to the one in the proof of Corollary 1 (using the error decomposition $\Pi_i(X_t)$) below, under **Assumption K.4(b)**.

noting that the polynomial term dominates here and the case where $s_0 = O(1)$ follows trivially.

Proof of Corollary 1:

Given cubic B-splines with uniform knots $t_0 < \cdots < t_{G_n}$, for any $\phi \in W_2^{m+1}(\mathcal{I}_\epsilon)$ and $0 \leq l \leq m+1$, there exists an approximation $\pi_n\phi$ such that for $l \in \{0, \ldots, m\}$:

$$\|\phi^{(l)} - \pi_n\phi^{(l)}\| \leq C_k G_n^{-(m+1-l)} \|\phi\|_{W_2^{m+1}(\mu)}$$

by Theorem 12.8 of Schumaker (2007), where C_k depends only on the spline order. Applying this to each inner edge function ϕ_{ij} and outer edge function Φ_i, we obtain the following spline approximations $\pi_n\phi_{ij}$ and $\pi_n\Phi_i$ satisfying:

$$\|\phi_{ij}^{(l)} - \pi_n\phi_{ij}^{(l)}\| \leq C_k B G_n^{-(m+1-l)}, \quad \|\Phi_i^{(l)} - \pi_n\Phi_i^{(l)}\| \leq C_k B G_n^{-(m+1-l)}$$

noting that $B < \infty$ is the bound from Assumption K.4(a).

Let $X_{t,j} \equiv X_j$. Given the sieve approximation

$$(\pi_n\theta_0)(X_t) = \sum_{i=1}^{s_0} \pi_n\Phi_i\left(\sum_{j\in\mathcal{J}_i} \pi_n\phi_{i,j}(X_{t,j}\right)$$

define the following error decomposition:

$$\Pi_i(X_t) = \pi_n\Phi_i\left(\sum_j \pi_n\phi_{ij}(X_{t,j})\right) - \Phi_i\left(\sum_j \phi_{ij}(X_{t,j})\right) = A_i(X_t) + B_i(X_t)$$

where $A_i(X_t) = \pi_n\Phi_i(v_i) - \Phi_i(v_i)$ and $B_i(X_t) = \Phi_i(v_i) - \Phi_i(u_i)$, with $u_i = \sum_j \phi_{ij}(X_{t,j})$ and $v_i = \sum_j \pi_n\phi_{ij}(X_{t,j})$. For the inner sum, (noting that $G_n \geq 2$ and re-indexing $r = m - l$ in order to form a geometric series) we have:

$$\|u_i - v_i\|_{W_2^m}^2 = \sum_{|\alpha|\leq m} \|D^\alpha(u_i - v_i)\|_{L_2(\mu)}^2 \leq s_1 \sum_{j\in\mathcal{J}_i} \sum_{l=0}^{m} \|(\phi_{ij} - \pi_n\phi_{ij})^{(l)}\|_{L_2(\mu)}^2$$
$$\leq s_1^2 C_k^2 B^2 \sum_{l=0}^{m} G_n^{-2(m+1-l)}$$
$$\leq s_1^2 C_k^2 B^2 G_n^{-2} \sum_{r=0}^{m} G_n^{-2r}$$
$$\leq s_1^2 C_k^2 B^2 G_n^{-2} (\frac{4}{3}) \tag{20}$$

Taking the square root, we obtain: $\|u_i - v_i\|_{W_2^m} \leq \sqrt{(\frac{4}{3})} s_1 C_k B G_n^{-1}$. By assumption **K.4**, the mapping $\Phi_i : W_2^m \to W_2^m$ is Lipschitz continuous (with bounded derivatives), so there exists a constant $L_1 > 0$ (dependent on the constants from the previous step) such that:

$$\|\Phi_i(v_i) - \Phi_i(u_i)\|_{W_2^m} \leq L_1 \|v_i - u_i\|_{W_2^m}$$

thus ensuring the boundedness of $\|B_i\|_{W_2^m}$. Similarly, the error $A_i = \pi_n\Phi_i(v_i) - \Phi_i(v_i)$ satisfies:

$$\|A_i\|_{W_2^m} \leq L_2 \|\pi_n\Phi_i - \Phi_i\|_{W_2^m}$$

where the bounded derivatives of v_i are absorbed by the constant $L_2 > 0$. By an identical argument from the inner sum error case and noting that edges in the remaining $L_n - 1$ layers in the sieve KAN are used to approximate each Φ_i, we have

$$\|\pi_n\Phi_i - \Phi_i\|_{W_2^m} = \sum_{l=0}^{m} \|(\pi_n\Phi_i - \Phi_i)^{(l)}\|_{L_2}^2 \leq C_1^2 B^2 \sum_{l=0}^{m} G_n^{-2(m+1-l)} \leq C_5 B^2 G_n^{-2}$$

therefore, $\|A_i\|_{W_2^m} \leq C_6 B G_n^{-1}$. Combining the bounds, we have:

$$\|\Pi_i\|_{W_2^m} \leq \|A_i\|_{W_2^m} + \|B_i\|_{W_2^m} \leq C_6 B G_n^{-1} + C_3 C_2 s_1 B G_n^{-1}$$

summing over $i = 1, \ldots, s_0$:

$$\|\pi_n\theta_0 - \theta_0\|_{W_2^m} \leq \sum_{i=1}^{s_0} \|\Pi_i\|_{W_2^m} + C' s_0 O(n^{-1/2}) \leq C s_0 s_1 B G_n^{-1} + C' s_0 O(n^{-1/2})$$

where $C > 0$ depends on the constants from each previous step.

Proof of Theorem 2: In this proof we account for two components, the term arising from selection of (active) edges and the basis function approximation. For the former, we have the term $s_n \log\left(\frac{r_n}{s_n}\right)$ which is a generalization bound of the combinatorial complexity of sparse function classes (by Lemma 17.5.1 of Cover and Thomas, 1991); highlighting the s_n active edges from the r_n potential edges in total. As for the latter, we note that across each active edge, the activation function is approximated by G_n B-spline functions with coefficients constrained $\ell_{1,2}$-norm: $R_n(\theta) = \sum_{j=1}^{r_n} \|\boldsymbol{c}_j\|_2 \leq \Delta_n$. By Theorem 4.2.11 of Vershynin (2018), the complexity of the coefficients within an $\ell_{1,2}$ ball is bounded by $s_n G_n \log\left(\frac{\Delta_n}{\epsilon}\right)$; where the $G_n \log\left(\frac{\Delta_n}{\epsilon}\right)$ factor scales with the metric entropy of a Euclidean ball of radius Δ_n in $\mathbb{R}^{G_n}$. Given constants $C_1 > 0$ and $C_2 > 0$ and combining both components, the metric entropy is bounded.

Proof of Theorem 3: We begin by letting $\delta_n = n^{-1/4-\eta}$ for some arbitrarily small $\eta > 0$ and verify the following entropy integral condition:

$$\int_0^{\delta_n} \sqrt{\frac{\log \mathcal{N}(\epsilon, \mathcal{K}_n, \|\cdot\|_{L_2(\mu)})}{n}} d\epsilon \leq C_3 \delta_n^2$$

From **Theorem 2**, we have: $\log \mathcal{N}(\epsilon, \mathcal{K}_n, \|\cdot\|_{L_2(\mu)}) \leq C_1 s_n \log\left(\frac{r_n}{s_n}\right) + C_2 s_n G_n \log\left(\frac{\Delta_n}{\epsilon}\right)$, consider the case where $s_n \asymp s_0 = O(\log n)$ for which we defined $W_n = C_W \log n$ accordingly

earlier. Then we have $r_n = O((\log n)^3)$ by definition, leading to the expression:

$$\log \mathcal{N}(\epsilon, \mathcal{K}_n, \|\cdot\|_{L_2(\mu)}) \leq C_1(\log n) \log\left(\frac{O((\log n)^3)}{O(\log n)}\right) + C_2(\log n) n^\gamma \log(\Delta_n/\epsilon)$$
$$= O(\log n \log(\log n)) + O(\log n \cdot n^\gamma \log(\Delta_n/\epsilon))$$

where in the case of $s_0 = O(1)$, the RHS term is simply equal to $O(\log(\log n)) + O(n^\gamma \log(\Delta_n/\epsilon))$. In both cases, the dominant term of $\log \mathcal{N}(\epsilon, \mathcal{K}_n, \|\cdot\|_{L_2(\mu)})$ is $O(s_0 G_n \log(\Delta_n/\epsilon))$. We decompose this by splitting the integral at $\epsilon_0 = n^{-1}$ and show that both terms are $o_P(n^{-1/4})$:

$$\int_0^{\delta_n} \sqrt{\frac{\log \mathcal{N}(\epsilon, \mathcal{K}_n, \|\cdot\|_{L_2(\mu)})}{n}} d\epsilon = \underbrace{\int_0^{\epsilon_0} \cdots d\epsilon}_{(I)} + \underbrace{\int_{\epsilon_0}^{\delta_n} \cdots d\epsilon}_{(II)} \tag{21}$$

Term (I):

$$(I) \leq \epsilon_0 \sqrt{\frac{\log \mathcal{N}(\epsilon_0, \mathcal{K}_n, \|\cdot\|_{L_2(\mu)})}{n}}$$

For $s_0 = O(\log n)$, we have $\log \mathcal{N}(\epsilon_0, \mathcal{K}_n, \|\cdot\|_{L_2(\mu)}) = O(\log n \cdot n^\gamma \log n) = O(n^\gamma (\log n)^2)$. Therefore:

$$(I) = O\left(n^{-1}\sqrt{n^\gamma (\log n)^2}\right) = O\left(n^{\frac{\gamma}{2}-1} \log n\right)$$

and in order for this term to be $o_P(n^{-1/4})$, we require $\frac{\gamma}{2} - 1 < -1/4 \implies \frac{\gamma}{2} < 3/4 \implies \gamma < 3/2$, which is satisfied by $\gamma < 1$. The same condition arises when $s_0 = O(1)$ since we end up with $(I) = O\left(n^{\frac{\gamma}{2}-1}(\log n)^{1/2}\right)$ for this case.

Term (II):

Using the aforementioned dominant term of $\log \mathcal{N}(\epsilon, \mathcal{K}_n, \|\cdot\|_{L_2(\mu)})$, we have:

$$\begin{aligned}
(II) &\leq \sqrt{\frac{Cs_0}{n}} \int_{\epsilon_0}^{\delta_n} \sqrt{G_n \log(\Delta_n/\epsilon)} d\epsilon \\
&= O\left(\sqrt{\frac{s_0}{n}} \sqrt{G_n} \delta_n \sqrt{\log(\Delta_n/\delta_n)} \right) \\
&= O\left(n^{\frac{\zeta_s}{2} - \frac{1}{2}} (n^{\gamma})^{1/2} n^{-1/4-\eta} (\log n)^{1/2} \right) \\
&= O\left(n^{\frac{\zeta_s + \gamma}{2} - \frac{3}{4} - \eta} (\log n)^{1/2} \right)
\end{aligned}$$

For term (II) to be $o_P(n^{-1/4})$, we require $\frac{\zeta_s+\gamma}{2} - \frac{3}{4} - \eta < -\frac{1}{4} \implies \frac{\zeta_s+\gamma}{2} < \frac{1}{2} + \eta \implies \zeta_s + \gamma < 1 + 2\eta$. For the cases considered here we have $\zeta_s = 0$ for $s_0 = O(1)$ and $\zeta_s \to 0$ for $s_0 = O(\log n)$ (by L'Hopital's rule, $\forall \zeta_s > 0$). Thus for a sufficiently small $\eta > 0$, the key condition for guaranteeing our $o_P(n^{-1/4})$ rate is that $\gamma < 1$. By Theorem 1 and Lemma 1 of Chen and Shen (1998) (given $a_{n1} = n^{1/(1+\xi)}$ and $a_{n2} = \lfloor n/(2a_{n1}) \rfloor$, and noting that Assumptions K.1-3 imply a.1-3 of the latter), we conclude that $\|\hat{\theta}_n - \pi_n \theta_0\| = o_P(n^{-1/4})$.

A.2 Condition B (Chen and Shen 1998)

Condition B.1: Let $r[\theta - \theta_0, Z_t] \equiv l(\theta, Z_t) - l(\theta_0, Z_t) - l'_{\theta_0}[\theta - \theta_0, Z_t]$. We require:

$$\sup_{\{\theta \in \Theta_n : \|\theta - \theta_0\| \leq \delta_n\}} \mu_n \left(r[\theta - \theta_0, Z_t] - r[P_n(\theta^*(\theta, \epsilon_n)) - \theta_0, Z_t] \right) = O_P(\epsilon_n^2)$$

where $\theta^*(\theta, \epsilon_n) = (1 - \epsilon_n)\theta + \epsilon_n(u^* + \theta_0)$ with $u^* = \pm v^*$, $\mu_n = \frac{1}{n}\sum_{t=1}^n \delta_{Z_t}$ and $\epsilon_n = o(n^{-1/2})$.

Condition B.2:

$$\sup_{\{\theta \in \Theta_n : 0 < \|\theta - \theta_0\| \leq \delta_n\}} \left| K(\theta_0, P_n\theta^*(\theta, \epsilon_n)) - K(\theta_0, \theta) - \frac{1}{2}\left[\|\theta^*(\theta, \epsilon_n) - \theta_0\|^2 - \|\theta - \theta_0\|^2 \right] \right| = O(\epsilon_n^2)$$

where $K(\theta_0, \theta) = \mathbb{E}[l(\theta_0, Z_t) - l(\theta, Z_t)]$.

Condition B.3:

$$(i) \sup_{\{\theta \in \Theta_n : 0 < \|\theta - \theta_0\| \leq \delta_n\}} \|\theta^*(\theta, \epsilon_n) - P_n(\theta^*(\theta, \epsilon_n))\| = O(\delta_n^{-1}\epsilon_n^2)$$

$$(ii) \sup_{\{\theta \in \Theta_n : \|\theta - \theta_0\| \leq \delta_n\}} \mu_n \left(l'_{\theta_0}[\theta^*(\theta, \epsilon_n) - P_n(\theta^*(\theta, \epsilon_n)), Z_t] \right) = O_P(\epsilon_n^2)$$

Condition B.4:

$$\sup_{\{\theta \in \Theta_n : \|\theta - \theta_0\| \leq \delta_n\}} \mu_n \left(l'_{\theta_0}[\theta - \theta_0, Z_t] \right) = O_P(\epsilon_n)$$

Condition B.5:

$$n^{1/2}\mu_n \left(l'_{\theta_0}[v^*, Z_t] \right) \xrightarrow{d} \mathcal{N}(0, \sigma_v^2)$$

where $\sigma_v^2 = \text{Var}_0(l'_{\theta_0}[v^*, Z_1]) + 2\sum_{j=2}^{\infty} \text{Cov}_0(l'_{\theta_0}[v^*, Z_1], l'_{\theta_0}[v^*, Z_j])$.

Proof of Theorem 4:

By Theorem 2 of Chen and Shen (1998), it is sufficient to verify Conditions B.1-B.5 therein. By Assumption K.2(c), the class $\mathcal{F}_n = \{r[\theta - \theta_0, \cdot] - r[P_n(\theta^*(\theta, \epsilon_n)) - \theta_0, \cdot] : \theta \in \Theta_n, \|\theta - \theta_0\| \leq \delta_n\}$ has bounded envelope and the entropy bound is given by Theorem 2. Applying Lemma 1 of Chen and Shen (1998) with $a_n = n^{1/(1+\xi)}$, we obtain the bound in Condition B.1 above. For the quadratic loss function, Condition B.2 holds exactly. By Theorem 1, for any $h \in \Theta$, $\inf_{g \in \Theta_n} \|g - h\| \leq C s_0 G_n^{-(m+1)}$. With $\epsilon_n = o(n^{-1/2})$ and $\delta_n = o(n^{-1/4})$, Condition B.3(i) holds. Condition B.3(ii) is satisfied by an identical argument as B.1 (combining entropy bound with Lemma 1). The class $\mathcal{G}_n = \{l'_{\theta_0}[\theta - \theta_0, \cdot] : \theta \in \Theta_n, \|\theta - \theta_0\| \leq \delta_n\}$ has envelope $U(Y)\delta_n^{\kappa}$ by Assumption K.2(c). Given $\log N_{[]}(\epsilon, \mathcal{G}_n, L_2(P)) \leq C s_n G_n \log(\Delta_n/\epsilon)$ from Theorem 2, we apply Lemma 1 and B.4 is satisfied. Under Assumptions 1-2 and applying a CLT for β-mixing sequences (Yokoyama, 1980), Condition B.5 is satisfied trivially.

A.3 Conditional Density Estimation Example

We now consider a conditional density estimation (CDE) example to motivate a discussion on the shortcomings of deep MLPs in sparse high-dimensional settings, as well as how these are addressed by sieve KANs. Suppose we seek to employ a deep MLP in a sparse estimation problem involving covariates with high-dimensions and naively attempt to implement ℓ_1 regularization (or the more common "elastic net" method combining it with ℓ_2 regularization). This applies a penalty $\sum_{i,j} |w_{ij}|$ on all weights across every layer, in theory reducing the contributions of weights significantly influenced by irrelevant features. Achieving this in practice with single hidden layer networks is more feasible relative to DNNs[45], due to the complex interconnected nature of the weights in the latter architecture. Consider the

[45] See Shen et al. (2023).

following function:

$$f_0(x_1, x_2, \ldots x_{10}) = -3.0 + 2\sin(x_1) + 0.1(x_2^2) + \epsilon, \quad \epsilon \sim N(0, 0.25)$$

where the true density of the relevant variables (x_1 and x_2) takes on a bullseye shape, $d = 10$ and $n = 10,000$.

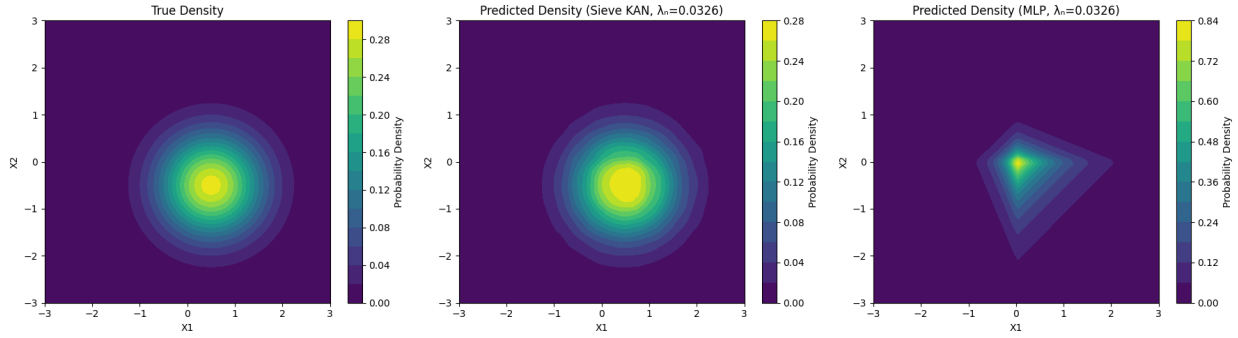

Figure 2: CDE Example Predicted Densities

As seen in Figure 2, a deep MLP (in this case, a ReLU DNN) applying ℓ_1 regularization will struggle to learn the shape of the true density. However in the absence of this regularization, the MLP is more prone to overfitting in noisy DGPs[46], highlighting a key tension in the potential application of deep MLPs in high-dimensions. As for our sieve KAN, the predicted density is much closer to the true density in terms of shape and scale, despite relying only the learned univariate functions preserved under ℓ_1-regularization. We note that due to how $n \gg d$, this prediction can easily be improved further by increasing λ_n and/or G_n through γ (initially set to $\gamma = 0.3$ to obtain a grid size similar to those in DGP #1 and #2 in the earlier simulations). In line with the philosophy of truly data-adaptive approaches to nonparametric estimation, these parameters can also be automatically optimized using cross-validation[47] in a manner similar to the Lasso penalty selection offered by packages across various programming languages.

[46]See Schmidt-Hieber (2020)

[47]See the documentation for several examples of this in practice, in addition to standard hyperparameter tuning; e.g., setting the learning rate.

A.4 Shallow Sieve Networks (Chen and White, 1999)

Let d_x denote the dimension of the covariates. Let $\mathcal{B}_{d_x}^m$ denote the weighted Sobolev space comprising all functions on $\mathbb{R}^{d_x}$ whose partial derivatives up to order m are both continuous and uniformly bounded (see the definitions in Section 2). The target function $f : \mathbb{R}^{d_x} \to \mathbb{R}$ is assumed to admit a Fourier representation such that it belongs to the functional class:

$$\mathcal{F}_{d_x}^{m+1} = \left\{ f : \mathbb{R}^d \to \mathbb{R} \;\middle|\; f(x) = \int \exp\big(ia^\top x\big)\, d\sigma_f(a),\ \|\sigma_f\|_{m+1} \equiv \int I(a)^{m+1}\, d|\sigma_f|_{\text{tv}}(a) < \infty \right\} \tag{22}$$

This assumption imposes a general smoothness constraint[48] required by the class of SLFNs utilized in Chen and White (1999), denoted by $\mathcal{G}_n \equiv G_{d_x}^m(\psi, r_n, \Delta_n)$ and defined as:

$$\mathcal{G}_n = \Bigg\{ g : g(x) = \sum_{j=1}^{r_n} v_j l(a_j)^{-m} \psi(a_j' x + w_j), \quad a_j \in \mathbb{R}^{d_x},\ (w_j, v_j) \in \mathbb{R},\ \sum_{j=1}^{r_n} |v_j| \leq \Delta_n \Bigg\} \tag{23}$$

Here, $\psi \in \mathcal{B}_1^m$ represents the selected activation function (satisfying a Hilbert condition) that is k-finite for some $k \geq m$. The parameter vector $\omega_j \equiv (a_j, w_j, v_j)$ contains the network weights, where r_n specifies the number of hidden units as discussed earlier in this paper. For any weight vector a_j, the term $l(a_j) \equiv \max\{(a_j' a_j)^{\frac{1}{2}}, 1\}$ normalizes the input layer weights, thereby controlling model complexity and enforcing the required smoothness of the approximation.

[48] Specifically, this ensures the Fourier transform possesses a bounded first moment.