Nonparametric Quantile Regression with ReLU-Activated Recurrent Neural Networks

Hang Yu[1,2,*] **Lyumin Wu**[3,*], **Wen-Xin Zhou**[4], **Zhao Ren**[5,†]

[1] National Key Laboratory for Novel Software Technology, Nanjing University, China
[2] School of Artificial Intelligence, Nanjing University, China
[3] Department of AI and Data Science, The University of Hong Kong, China
[4] Department of Information and Decision Sciences, University of Illinois Chicago, USA
[5] Department of Statistics, University of Pittsburgh, USA

Abstract

This paper investigates nonparametric quantile regression using recurrent neural networks (RNNs) and sparse recurrent neural networks (SRNNs) to approximate the conditional quantile function, which is assumed to follow a compositional hierarchical interaction model. We show that RNN- and SRNN-based estimators with rectified linear unit (ReLU) activation and appropriately designed architectures achieve the optimal nonparametric convergence rate, up to a logarithmic factor, under stationary, exponentially β-mixing processes. To establish this result, we derive sharp approximation error bounds for functions in the hierarchical interaction model using RNNs and SRNNs, exploiting their close connection to sparse feedforward neural networks (SFNNs). Numerical experiments and an empirical study on the Dow Jones Industrial Average (DJIA) further support our theoretical findings.

1 Introduction

Quantile regression (QR) [Koenker and Bassett, 1978] provides a flexible framework for estimating conditional quantiles of a response variable, offering a more comprehensive characterization of the conditional distribution than least squares regression, which focuses solely on the conditional mean. By modeling different quantiles, QR reveals how covariate effects vary across the response distribution, making it particularly valuable when errors are non-normal, heteroscedastic, or when the focus is on tail behavior. Since its introduction, QR has evolved through a wide range of methodological developments, including linear QR [Koenker and Bassett, 1978], quantile autoregression [Koenker and Xiao, 2006], quantile regression forests [Meinshausen, 2006], and quantile boosting [Zheng, 2012], among others. An early step toward integrating neural networks into QR was taken by White [1992], who established theoretical guarantees for single-layer feedforward networks. Building on this foundation, subsequent research has focused on multi-layer feedforward neural networks (FNNs) with ReLU activation functions [Nair and Hinton, 2010]. Assuming that the true conditional quantile function admits a compositional structure consisting of lower-dimensional component functions, Shen et al. [2021] derived a sub-optimal convergence rate for ReLU-based FNN estimators in the presence of heavy-tailed response distributions, which was later refined in a subsequent work [Shen et al., 2025]. Extending this line of work, Padilla et al. [2022] investigated nonparametric QR using ReLU-activated sparse feedforward neural networks (SFNNs) with bounded parameters, achieving optimal convergence rates under Hölder smoothness or Besov-space assumptions on the quantile

[*] Equal contribution.
[†] Correspondence: Zhao Ren <zren@pitt.edu>

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

function. More recently, Feng et al. [2024] addressed the problem of covariate shift in nonparametric QR via ReLU FNNs, obtaining minimax-optimal rates under an adaptive self-calibration condition.

However, FNNs often struggle to capture temporal dynamics and sequential dependencies, thereby limiting their effectiveness in modeling dependent data. In contrast, recurrent neural networks (RNNs) [Rumelhart et al., 1986], specifically designed for sequential data, are naturally better suited to such settings. Their capacity to retain and integrate information across time steps has proven effective in applications such as time series forecasting, sequence-to-sequence learning [Sutskever et al., 2014], and modeling long-term dependencies [Hochreiter and Schmidhuber, 1997, Chung et al., 2014]. Motivated by these strengths, we investigate RNNs and their sparse variants (SRNNs) as predictive function classes and establish their theoretical properties within the framework of nonparametric QR.

Recent studies have investigated the theoretical properties of RNNs in the many-to-one setting. Several works have established PAC-style generalization guarantees for RNNs [Chen et al., 2020, Tu et al., 2020, Cheng et al., 2025]. More closely related to our setting are results on their approximation properties under dependent data. In particular, Jiao et al. [2024] showed that RNNs can attain the optimal convergence rate in nonparametric regression when the data are stationary and β-mixing. Building upon these developments, the present study establishes statistical guarantees for the performance of RNNs in QR, which requires distinct technical tools from those employed in least squares estimation. In what follows, we formally define the model, outline the main contributions, and introduce the notation.

Model. Consider sequentially stationary observations $\{(\mathbf{x}_t, y_t)\}_{t=1}^n$, where any consecutive N observations share the same joint distribution as $Z = ((X_1, Y_1), \ldots, (X_N, Y_N))$. Here, $Y_i \in \mathbb{R}$ denotes the random outcome of interest, and $X_i \in \mathbb{R}^{d_\mathbf{x}}$ is a $d_\mathbf{x}$-dimensional covariate vector for $i \in [N]$. Motivated by the recurrent structure of RNNs, we assume that y_t depends on the sequence of covariates $(\mathbf{x}_{t-N+1}, \ldots, \mathbf{x}_t)$. The formal regularity conditions governing this dependence are specified in Section 3.2. Given a quantile level $\tau \in (0, 1)$ of interest, we define the conditional τ-th quantile of y_t (or Y_N) given $\mathbf{x}_{t-N+1}, \ldots, \mathbf{x}_t$ (or $X_1, \ldots, X_N$) as

$$q_\tau(y_t|\mathbf{x}_{t-N+1}, \ldots, \mathbf{x}_t) = f_0(\mathbf{x}_{t-N+1}, \ldots, \mathbf{x}_t), \quad \mathbf{x}_{t-N+1}, \ldots, \mathbf{x}_t \in \mathbb{R}^{d_\mathbf{x}}, \quad N \le t \le n,$$

where $f_0 : \mathbb{R}^{d_\mathbf{x} \times N} \to \mathbb{R}$ is the unknown conditional quantile function. Equivalently, the relationship between Y_N and $(X_1, \ldots, X_N)$ can be expressed in additive form as

$$Y_N = f_0(X_1, \ldots, X_N) + \epsilon, \tag{1.1}$$

where ϵ denotes the regression error satisfying $\mathbb{P}(\epsilon \le 0|X_1, \ldots, X_N) = \tau$.

A key motivating example is the broad class of nonlinear autoregressive (AR) models with time-varying conditional variance.

Example (Nonlinear AR Model with Time-varying Conditional Variances)**.** Consider the heteroscedastic nonlinear AR model

$$W_t = \phi(W_{t-1}, \ldots, W_{t-p}) + \epsilon_t \sigma(W_{t-1}, \ldots, W_{t-q}),$$

where $\phi : \mathbb{R}^p \to \mathbb{R}$ and $\sigma : \mathbb{R}^q \to \mathbb{R}$ are unknown functions, $\{\epsilon_t\}_{t=-\infty}^{\infty}$ is a sequence of independently and identically distributed (i.i.d.) random variables satisfying $q_\tau(\epsilon_t) = 0$. The integers p and q denote the AR order, with $p \le q$. This model conforms to the general formulation in (1.1) by setting $N = q$, $Y_N = W_t$, and $X_i = W_{t-(N-i+1)}$ for $i \in [q]$. Under this representation, the conditional quantile function $f_0(X_1, \ldots, X_N)$ is given by $\phi(W_{t-1}, \ldots, W_{t-p})$.

Contributions. We summarize the main contributions of this work as follows.

- To the best of our knowledge, this work provides the first approximation error bounds for functions within a hierarchical interaction model [Kohler and Langer, 2021] using both RNNs and SRNNs. These bounds highlight the ability of such architectures to effectively capture the complexity inherent in hierarchical interaction structures. Our analysis builds upon the close connections between SFNNs and SRNNs, as established in Lemma 4 and Lemma 14 of the supplementary material. In particular, we show that any SRNN can be represented by an SFNN with a slightly larger but less sparse architecture, and vice versa, indicating that their respective function classes are comparable in expressive power. This result complements the equivalence between FNNs and RNNs previously demonstrated by Jiao et al. [2024].

- Built upon the established approximation error bounds, we conduct a comprehensive error analysis for nonparametric QR with weakly dependent data using RNNs and SRNNs. Specifically, we estimate the true conditional quantile function f_0 within a hierarchical interaction model characterized by intrinsic smoothness $\gamma^\star$. We show that, for a stationary exponentially $\boldsymbol{\beta}$-mixing sequence of n observations, the empirical risk minimizers based on both RNNs and SRNNs achieve a convergence rate of $n^{-2\gamma^\star/(2\gamma^\star+1)}$ under the squared L_2 norm, up to a logarithmic factor. This rate coincides with the minimax-optimal rate established by Schmidt-Hieber [2020]. Furthermore, we derive a slower convergence rate for the case of algebraically $\boldsymbol{\beta}$-mixing dependence.

Notations. For two sequences $\{a_n\}$ and $\{b_n\}$, we write $a_n \gtrsim b_n$ if there exists a constant $C > 0$, independent of n, such that $a_n \geq C b_n$. We write $a_n \lesssim b_n$ if $b_n \gtrsim a_n$. Additionally, we use the notation $a_n \asymp b_n$ when both $a_n \lesssim b_n$ and $a_n \gtrsim b_n$ hold. For any $\alpha \in \mathbb{R}$, let $\lfloor \alpha \rfloor$ denote the largest integer that is strictly smaller than α, and $\lceil \alpha \rceil$ denote the smallest integer that is strictly larger than α. We denote $\mathbb{N} = \{1, 2, \ldots\}$, $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$, and $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$. For any $N \in \mathbb{N}$, we define $[N]$ as $\{1, \ldots, N\}$. For any set $\mathcal{S}$, we denote its cardinality by $|\mathcal{S}|$. For $\mathbf{x} = (x_1, \ldots, x_d)^\top \in \mathbb{R}^d$, we define $\|\mathbf{x}\|_p = (\sum_{i=1}^d |x_i|^p)^{1/p}$, $\|\mathbf{x}\|_\infty = \max_i |x_i|$, $\|\mathbf{x}\|_0 = \sum_i \mathbb{1}(x_i \neq 0)$. For $A = [a_{i,j}] \in \mathbb{R}^{m \times n}$, we define $\|A\|_0 = \sum_i \sum_j \mathbb{1}(a_{i,j} \neq 0)$ and $\mathrm{vec}(A) = (a_{1,1}, \ldots, a_{1,n}, \ldots, a_{m,1}, \ldots, a_{m,n})^\top \in \mathbb{R}^{mn}$. Moreover, for any real-valued function h defined on a domain $\mathcal{X}$, we define $\|h\|_\infty = \sup_{\mathbf{x} \in \mathcal{X}} |h(\mathbf{x})|$. Let $\mathbb{P}_X$ be a probability measure on $\mathcal{X}$. The L_p norm of h ($1 \leq p < \infty$) with respect to $\mathbb{P}_X$ is defined as $\|h\|_p = (\mathbb{E}_{X \sim \mathbb{P}_X} |h(X)|^p)^{1/p}$.

2 Methodologies

In this section, we formally define the architectures of RNNs and SRNNs, both employing the ReLU activation function, $\sigma(x) = \max\{x, 0\}$, applied elementwise to vector inputs. Building on these architectures, we then introduce nonparametric QR estimators based on RNNs and SRNNs, constructed through empirical risk minimization (ERM) over overlapping subsequences of the data.

2.1 RNNs and SRNNs

An RNN is characterized by the following parameters: the input dimension $d_{\mathbf{x}}$, the output dimension $d_{\mathbf{y}}$, the width W, and the depth L. Given a time horizon N, an RNN processes an input sequence $\boldsymbol{X} := (\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}) \in \mathbb{R}^{d_{\mathbf{x}} \times N}$ sequentially through three types of layers: an input layer p, a sequence of recurrent layers $\{r_l\}_{l=1}^L$, and an output layer q. The architecture generates an output sequence $\boldsymbol{Y} := (\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(N)}) \in \mathbb{R}^{d_{\mathbf{y}} \times N}$ according to

$$\begin{aligned} \boldsymbol{V}_0 &= p(\boldsymbol{X}) = (P\mathbf{x}^{(1)}, \ldots, P\mathbf{x}^{(N)}), \\ \boldsymbol{V}_l &= r_l(\boldsymbol{V}_{l-1}) = (r_l^{(1)}(\boldsymbol{V}_{l-1}), \ldots, r_l^{(N)}(\boldsymbol{V}_{l-1})), \quad l \in [L], \\ \boldsymbol{Y} &= q(\boldsymbol{V}_L) = (Q\mathbf{v}_L^{(1)}, \ldots, Q\mathbf{v}_L^{(N)}), \end{aligned}$$

where $P \in \mathbb{R}^{W \times d_{\mathbf{x}}}$, $Q \in \mathbb{R}^{d_{\mathbf{y}} \times W}$, and $\boldsymbol{V}_l = (\mathbf{v}_l^{(1)}, \ldots, \mathbf{v}_l^{(N)}) \in \mathbb{R}^{W \times N}$ for $l \in [L] \cup \{0\}$. For each time step $t \in [N]$ and each layer $l \in [L]$, the recurrent operation $r_l^{(t)}$ is defined by

$$r_l^{(t)}(\boldsymbol{V}_{l-1}) := \sigma(A_l r_l^{(t-1)}(\boldsymbol{V}_{l-1}) + B_l \mathbf{v}_{l-1}^{(t)} + \mathbf{c}_l),$$

where $A_l, B_l \in \mathbb{R}^{W \times W}$, $\mathbf{c}_l \in \mathbb{R}^W$, and $r_l^{(0)} = \mathbf{0} \in \mathbb{R}^W$.

By composing all $L + 2$ layers, the overall RNN function r_θ can be expressed as

$$r_\theta := q \circ r_L \circ \cdots \circ r_1 \circ p,$$

where $\theta = (\mathrm{vec}(P)^\top, \mathrm{vec}(A_1)^\top, \mathrm{vec}(B_1)^\top, \mathbf{c}_1^\top, \ldots, \mathrm{vec}(A_L)^\top, \mathrm{vec}(B_L)^\top, \mathbf{c}_L^\top, \mathrm{vec}(Q)^\top)^\top \in \mathbb{R}^{d_\theta}$, with $d_\theta = (2W^2 + W)L + W(d_{\mathbf{x}} + d_{\mathbf{y}})$. In the many-to-one RNN setting considered here, the prediction function corresponds to the final element of the output sequence, denoted by $r_\theta^{(N)}$.

We define $\mathcal{RNN}$ as the class of RNN prediction functions with bounded outputs:

$$\mathcal{RNN}_{d_{\mathbf{x}}, d_{\mathbf{y}}}(W, L, K) = \left\{ r_\theta^{(N)} : \mathbb{R}^{d_{\mathbf{x}} \times N} \to \mathbb{R}^{d_{\mathbf{y}}} \,\middle|\, \sup_{\boldsymbol{X} \in \mathbb{R}^{d_{\mathbf{x}} \times N}, t \in [N]} \|r_\theta^{(t)}(\boldsymbol{X})\|_\infty \leq K \right\}.$$

For simplicity, we assume $K \geq 1$ throughout and omit it when boundedness is either understood or not required. Likewise, when the input and output dimensions $d_{\mathbf{x}}$ and $d_{\mathbf{y}}$ are clear from context, we denote the class more compactly as $\mathcal{RNN}(W, L)$.

Such RNN architectures, particularly those with large width or depth, are often substantially overparameterized in practice, which can lead to severe overfitting during training. Introducing sparsity provides an effective mechanism to mitigate this overparameterization. From a theoretical perspective, sparsity reduces the effective hypothesis space, thereby improving generalization bounds. We now formally define RNN sparsity. For a recurrent layer r_l, define its sparsity as $\mathcal{T}(r_l) = \|A_l\|_0 + \|B_l\|_0 + \|\mathbf{c}_l\|_0$. For the input and output layers, we set $\mathcal{T}(p) = \|P\|_0$ and $\mathcal{T}(q) = \|Q\|_0$, respectively. The total sparsity of an RNN r_θ is then given by $\mathcal{T}(r_\theta) = \mathcal{T}(p) + \mathcal{T}(q) + \sum_{l=1}^{L} \mathcal{T}(r_l)$. Using this notation, we define the class of SRNNs as

$$\mathcal{SRNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}(W, L, K, s) = \left\{ r_\theta^{(N)} \,|\, r_\theta^{(N)} \in \mathcal{RNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}(W, L, K) \text{ with } \mathcal{T}(r_\theta) \leq s \right\},$$

where s denotes the sparsity budget. SRNNs combine the sequential representation power of RNNs with the computational and statistical advantages of sparse architectures, making them particularly attractive for both theoretical analysis and practical implementation in resource-constrained environments.

2.2 Nonparametric quantile regression

As is standard in the QR literature, we begin by imposing regularity conditions on the noise variable ϵ in (1.1), conditioned on the covariates $X_1, \ldots, X_N$, as formalized in the following assumption.

Assumption 1. The conditional density of ϵ given $X_1, \ldots, X_N$, denoted by $p_{\epsilon|X_1,\ldots,X_N}$, exists and is continuous over its support. Moreover, it satisfies, almost surely over $X_1, \ldots, X_N$,

$$\underline{p} \leq p_{\epsilon|X_1,\ldots,X_N}(0) \leq \sup_{u\in\mathbb{R}} p_{\epsilon|X_1,\ldots,X_N}(u) \leq \bar{p}$$

for constants $\bar{p} \geq \underline{p} > 0$. In addition, there exists a constant $l_0 > 0$ such that, almost surely over $X_1, \ldots, X_N$, $|p_{\epsilon|X_1,\ldots,X_N}(u_1) - p_{\epsilon|X_1,\ldots,X_N}(u_2)| \leq l_0|u_1 - u_2|$ for all $u_1, u_2 \in \mathbb{R}$.

This assumption is standard in the literature and has been adopted in prior works such as Belloni and Chernozhukov [2011], Belloni et al. [2019], and Padilla et al. [2022]. It plays a crucial role in linking the excess risk to the squared L_2 error of the estimators, as established in Theorem 3.

Under Assumption 1, the target function f_0 is the unique minimizer of the population check loss

$$\mathcal{R}_\tau(f) := \mathbb{E}_{((X_1,\ldots,X_N)\sim\Pi,\, Y_N)}[\rho_\tau(Y_N - f(X_1, \ldots, X_N))],$$

where Π denotes the joint distribution of $X_1, \ldots, X_N$ and $\rho_\tau(u) = (\tau - \mathbb{1}(u < 0))u$ is the check loss. Hereafter, we assume that the joint distribution Π has compact support on $[0, 1]^{d_{\mathbf{x}}\times N}$. Given a dataset $\{(\mathbf{x}_t, y_t)\}_{t=1}^n$, we form overlapping subsequences to construct the training sample

$$\mathcal{S} = \{((\mathbf{x}_1, \ldots, \mathbf{x}_N), y_N), ((\mathbf{x}_2, \ldots, \mathbf{x}_{N+1}), y_{N+1}), \ldots, ((\mathbf{x}_{n-N+1}, \ldots, \mathbf{x}_n), y_n)\}.$$

We then estimate the target function f_0 by ERM over a function class $\mathcal{F}$, yielding the estimator

$$\widehat{f} \in \arg\min_{f\in\mathcal{F}} \mathcal{R}_n(f) := \frac{1}{n - N + 1} \sum_{t=N}^{n} \rho_\tau(y_t - f(\mathbf{x}_{t-N+1}, \ldots, \mathbf{x}_t)). \tag{2.1}$$

In the sections that follow, we derive error bounds for the cases where $\mathcal{F} = \mathcal{RNN}_{d_{\mathbf{x}},1}(W, L, K)$ and $\mathcal{F} = \mathcal{SRNN}_{d_{\mathbf{x}},1}(W, L, K, s)$, respectively.

3 Statistical Theory

In this section, we begin by introducing the hierarchical interaction model and derive error bounds for function approximation under this framework using both RNNs and SRNNs. Building on these results, we first establish oracle-type inequalities and subsequently use them to obtain separate upper bounds on the L_2 error of QR estimators based on RNNs and SRNNs, under the assumption that the data-generating process satisfies a stationary β-mixing condition.

3.1 Approximation error bounds

The theoretical performance of neural networks critically depends on the properties of the underlying function class. A commonly adopted assumption in nonparametric statistics is that the true regression function belongs to a Hölder class. We recall its definition below, following [Stone, 1982].

Definition 1 (Hölder Class of Functions $\mathcal{C}_d^\beta(\mathcal{X}, K)$)**.** Given a domain $\mathcal{X} \subseteq \mathbb{R}^d$, a positive Hölder smoothness parameter β, and a constant $K > 0$, the β-Hölder function class is defined as

$$\mathcal{C}_d^\beta(\mathcal{X}, K) = \left\{ f : \mathcal{X} \to \mathbb{R} \,\middle|\, \sum_{\boldsymbol{\alpha}:\|\boldsymbol{\alpha}\|_1 < \beta} \left\|\partial^{\boldsymbol{\alpha}} f\right\|_\infty + \sum_{\boldsymbol{\alpha}:\|\boldsymbol{\alpha}\|_1 = r} \sup_{\substack{\mathbf{x},\mathbf{y}\in\mathcal{X} \\ \mathbf{x}\neq\mathbf{y}}} \frac{\left|\partial^{\boldsymbol{\alpha}} f(\mathbf{x}) - \partial^{\boldsymbol{\alpha}} f(\mathbf{y})\right|}{\|\mathbf{x}-\mathbf{y}\|_2^s} \leq K \right\},$$

where $r = \lfloor \beta \rfloor$, $s = \beta - r$, $\partial^{\boldsymbol{\alpha}} = \partial^{\alpha_1} \cdots \partial^{\alpha_d}$ with $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d) \in \mathbb{N}_0^d$ and $\|\boldsymbol{\alpha}\|_1 = \sum_{i=1}^d \alpha_i$. Moreover, we refer to $\gamma = \beta/d$ as the dimension-adjusted degree of smoothness of $\mathcal{C}_d^\beta(\mathcal{X}, K)$.

Specifically, any function $f \in \mathcal{C}_d^\beta(\mathcal{X}, K)$ is bounded in magnitude by K. Without loss of generality, we assume throughout the paper that $K \geq 1$. Stone [1982] established that the minimax convergence rate for estimating a regression function under the L_2 norm over the Hölder class $\mathcal{C}_d^\beta(\mathcal{X}, K)$ is $n^{-\gamma/(2\gamma+1)}$. However, in neural network applications, the input dimension d is often large, resulting in a small value of the dimension-adjusted smoothness γ, which in turn leads to slow convergence rates. To mitigate the impact of high dimensionality, we consider a class of functions with a compositional structure, known as the hierarchical interaction model, which captures intrinsic low-dimensional structures and allows for more favorable approximation and estimation properties.

Definition 2 (Hierarchical Interaction Model)**.** Let $l, d \in \mathbb{N}$ be positive integers, and let $K \geq 1$. Suppose $\mathcal{P} \subseteq [1, \infty) \times \mathbb{N}$ is a parameter set such that $\sup_{(\beta,t)\in\mathcal{P}} \max\{\beta, t\} < \infty$. The hierarchical interaction model $\mathcal{H}_d^l(\mathcal{P}, K)$ is defined recursively as follows:

(i) A function $h : \mathbb{R}^d \to \mathbb{R}$ belongs to the first-level model $\mathcal{H}_d^1(\mathcal{P}, K)$ if there exist $(\beta, t) \in \mathcal{P}$, a function $h_1 \in \mathcal{C}_t^\beta(\mathbb{R}^t, K)$, and a set of indices $\{j_1, \ldots, j_t\} \subseteq \{1, \ldots, d\}$ such that $h(\mathbf{x}) = h_1(x_{j_1}, \ldots, x_{j_t})$ for $\mathbf{x} = (x_1, \ldots, x_d)^\top \in \mathbb{R}^d$.

(ii) For $l > 1$, a function h belongs to the hierarchical interaction model $\mathcal{H}_d^l(\mathcal{P}, K)$ if there exist $(\beta, t) \in \mathcal{P}$, a function $h_l \in \mathcal{C}_t^\beta(\mathbb{R}^t, K)$, and functions $u_1, \ldots, u_t \in \mathcal{H}_d^{l-1}(\mathcal{P}, K)$ such that $h(\mathbf{x}) = h_l(u_1(\mathbf{x}), \ldots, u_t(\mathbf{x}))$ for $\mathbf{x} \in \mathbb{R}^d$.

In analogy to the Hölder class, we define the intrinsic smoothness of $\mathcal{H}_d^l(\mathcal{P}, K)$ by

$$\gamma^\star = \beta^\star/t^\star, \quad \text{where } (\beta^\star, t^\star) = \arg\min_{(\beta,t)\in\mathcal{P}} \beta/t.$$

This quantity can be interpreted as the effective smoothness of the least regular component in the composition. Crucially, it does not depend on the ambient input dimension, thereby mitigating the curse of dimensionality. Extensive research has established the minimax-optimal convergence rates for models related to the hierarchical interaction model in nonparametric regression, demonstrating that these models retain favorable statistical performance even in high-dimensional settings [Bauer and Kohler, 2019, Schmidt-Hieber, 2020, Kohler and Langer, 2021]. Moreover, the hierarchical interaction model encompasses a broad class of structured functions, including classical models such as additive models, single-index models, and other compositionally structured function classes [Kohler and Langer, 2021].

Based on the preceding definitions, we impose the following assumption on the true quantile function f_0 introduced in (1.1).

Assumption 2. Let $K \geq 1$, $l \in \mathbb{N}$, and $\mathcal{P} \subset [1, \infty) \times \mathbb{N}$ satisfy $\sup_{(\beta,t)\in\mathcal{P}} \max\{\beta, t\} < \infty$. The true quantile function f_0 belongs to the hierarchical interaction model $\mathcal{H}_{d_\mathbf{x}\times N}^l(\mathcal{P}, K)$.

The following theorem provides an error bound for approximating functions within a hierarchical interaction model using RNNs.

Theorem 1. Under Assumption 2, for any $W_0, L_0 \geq 3$, and a probability measure μ on $[0,1]^{d_\mathbf{x}\times N}$ that is absolutely continuous with respect to the Lebesgue measure, the following inequality holds

$$\inf_{f\in\mathcal{RNN}_{d_\mathbf{x},1}(W,L,K)} \left\{ \int_{[0,1]^{d_\mathbf{x}\times N}} |f(\mathbf{x}) - f_0(\mathbf{x})|^2 \mu(\mathrm{d}\mathbf{x}) \right\}^{1/2} \leq c_1 (W_0 L_0)^{-2\gamma^\star},$$

where

$$W = c_2(d_{\mathbf{x}} + 1)\lceil W_0 \log W_0 \rceil + 1 \quad \text{and} \quad L = 2c_3\lceil L_0 \log L_0 \rceil + 2N. \tag{3.1}$$

Here, the positive constants c_1–c_3 depend on $(l, N, \mathcal{P}, K)$.

Theorem 1 establishes the ability of RNNs to approximate hierarchical functions. In contrast, while Jiao et al. [2024] also explored the approximation capabilities of RNNs, their analysis relied on the restrictive assumption that the true regression function belongs to a Hölder class. As a result, their approximation error bounds exhibit a strong dependence on the input dimension. By leveraging the benefits of the hierarchical interaction model in high dimensions, a more thorough exploration of the theoretical properties of SRNNs becomes possible. This is particularly relevant as SRNNs are commonly used in high-dimensional data scenarios. We then present the following theorem, which provides a result for SRNNs that is not achievable by Jiao et al. [2024].

Theorem 2. Under Assumption 2, for any $W_0 \geq \sup_{(\beta,t)\in\mathcal{P}} \max\{(\beta+1)^t, (K+1)e^t\}$, $L_0 \geq 1$, and a probability measure μ on $[0,1]^{d_{\mathbf{x}}\times N}$ that is absolutely continuous with respect to the Lebesgue measure, the following inequality holds

$$\inf_{f\in\mathcal{SRNN}_{d_{\mathbf{x}},1}(W,L,K,s)} \left\{ \int_{[0,1]^{d_{\mathbf{x}}\times N}} |f(\mathbf{x}) - f_0(\mathbf{x})|^2 \mu(\mathrm{d}\mathbf{x}) \right\}^{1/2} \leq c_4 \left(W_0 2^{-L_0} + W_0^{-\gamma^\star} \right),$$

where

$$\begin{aligned} W &= (d_{\mathbf{x}} + 1)c_5 W_0 + 1, \quad L = 2c_6 L_0 + 2N, \\ s &= c_7 L_0 W_0 + 3(d_{\mathbf{x}} + 1)c_5 W_0 N + 6(c_6 L_0 + N)(d_{\mathbf{x}} + 1)c_5 W_0. \end{aligned} \tag{3.2}$$

Here, the positive constants c_4–c_7 depend on $(l, N, \mathcal{P}, K)$.

The approximation error bound established in Theorem 2 demonstrates the effectiveness of SRNNs in approximating hierarchical functions. The key distinction between the error bounds in Theorem 1 and Theorem 2 lies in the construction of the neural networks. Specifically, Theorem 1 is based on RNNs constructed using the methodology in Theorem 3.3 of Jiao et al. [2023], while Theorem 2 employs SRNNs constructed through local Taylor approximations, following the approach of Yarotsky [2017]. Importantly, the difference in the resulting error bounds becomes negligible when applied to the derivation of convergence rates for QR estimators. As demonstrated in Theorem 4 and Theorem 5, both approximation bounds can be incorporated into oracle inequalities, leading to optimal rates for estimators based on RNNs and SRNNs, respectively.

3.2 Error bounds for QR estimators

The statistical performance of neural network estimators in regression tasks critically depends on the distribution of the observed data. Classical statistical theory typically assumes that observations are i.i.d. In contrast, we consider a more general setting where the observations $\{(\mathbf{x}_t, y_t)\}_{t=1}^n$ form a stationary $\boldsymbol{\beta}$-mixing sequence. To proceed, we introduce the following definitions.

Definition 3 (Stationarity)**.** A sequence of random vectors $\{\mathbf{z}_t\}_{t=-\infty}^{\infty}$ is said to be stationary if, for any given $t \in \mathbb{Z}$ and $m, k \in \mathbb{N}_0$, the distribution of the random matrix $(\mathbf{z}_t, \mathbf{z}_{t+1}, \ldots, \mathbf{z}_{t+m})$ is identical to that of $(\mathbf{z}_{t+k}, \mathbf{z}_{t+k+1}, \ldots, \mathbf{z}_{t+m+k})$.

Definition 4 ($\boldsymbol{\beta}$-mixing [Bradley, 1983])**.** Let $\{\mathbf{z}_t\}_{t=-\infty}^{\infty}$ be a sequence of random vectors. For any $i, j \in \mathbb{Z}\cup\{-\infty, +\infty\}$, define $\sigma_i^j = \sigma(\mathbf{z}_i, \mathbf{z}_{i+1}, \ldots, \mathbf{z}_j)$ as the σ-algebra generated by $\mathbf{z}_k, i \leq k \leq j$. For any $a \in \mathbb{N}$, the $\boldsymbol{\beta}$-mixing coefficient of the stochastic process $\{\mathbf{z}_t\}_{t=-\infty}^{\infty}$ is defined as

$$\boldsymbol{\beta}(a) = \sup_{k\geqslant 1} \mathbb{E}_{B\in\sigma_{-\infty}^k} \left[\sup_{A\in\sigma_{k+a}^{\infty}} |\mathbb{P}(A|B) - \mathbb{P}(A)| \right].$$

We say that $\{\mathbf{z}_t\}_{t=-\infty}^{\infty}$ is algebraically $\boldsymbol{\beta}$-mixing if there exist positive constants β_0 and $r > 1$ such that $\boldsymbol{\beta}(a) \leq \beta_0/a^r$ for all a. Similarly, it is said to be exponentially $\boldsymbol{\beta}$-mixing if there exist positive constants β_0, β_1 and r such that $\boldsymbol{\beta}(a) \leq \beta_0 \exp(-\beta_1 a^r)$ for all a.

The concept of $\boldsymbol{\beta}$-mixing has been extensively studied in the literature [Yu, 1994, Mohri and Rostamizadeh, 2010, Phandoidaen and Richter, 2020, Jiao et al., 2024]. In particular, a number of

procedures have been developed to estimate the mixing rate r [McDonald et al., 2011, 2015]. We emphasize that our data assumption includes the nonlinear AR model with time-varying conditional variances, which was not considered in Jiao et al. [2024] due to their differing stationarity assumption on the sequential observations.

The following theorem presents oracle-type inequalities for the QR estimator when the function class is specified as RNNs and SRNNs.

Theorem 3. Assume Assumption 1 holds, that $\{(\mathbf{x}_t, y_t)\}_{t=1}^n$ is a stationary $\boldsymbol{\beta}$-mixing sequence, and $\|f_0\|_\infty \leq K$ for some $K \geq 1$. For any positive integer ℓ such that $n \geq 4\ell N$, we define $m_{\max} = 2(n-N+1)/(N\ell)$ and $m_{\min} = (n-N+1)/(2N\ell)$.

(i) Let $\mathcal{F} = \mathcal{RNN}_{d_{\mathbf{x}},1}(W, L, K)$ with $W, L \geq 3$,

$$\delta_a = \inf_{f\in\mathcal{F}} \|f - f_0\|_2, \quad \text{and} \quad \delta_b = WL\sqrt{\log(\max\{W,L\})\log(m_{\max})/m_{\min}}.$$

Then, there exists a constant $c_8 > 0$, such that for any $u \geq 1$, the ERM estimator $\widehat{f}$ in (2.1) satisfies

$$\mathbb{P}\left(\|\widehat{f} - f_0\|_2 \geq c_8\big(\delta_a + \delta_b + \sqrt{u/m_{\max}}\big)\right) \lesssim N\ell e^{-u} + n\lfloor \log_2(2K/\delta_b)\rfloor \boldsymbol{\beta}((\ell-1)N+1). \quad (3.3)$$

(ii) Let $\mathcal{F} = \mathcal{SRNN}_{d_{\mathbf{x}},1}(W, L, K, s)$ with $W, L \geq 3$,

$$\delta_a = \inf_{f\in\mathcal{F}} \|f - f_0\|_2, \quad \text{and} \quad \delta_b = \sqrt{sL\log(WL^2)\log(m_{\max})/m_{\min}}.$$

Then, there exists a constant $c_9 > 0$, such that for any $u \geq 1$, the ERM estimator $\widehat{f}$ in (2.1) satisfies

$$\mathbb{P}\left(\|\widehat{f} - f_0\|_2 \geq c_9\big(\delta_a + \delta_b + \sqrt{u/m_{\max}}\big)\right) \lesssim N\ell e^{-u} + n\lfloor \log_2(2K/\delta_b)\rfloor \boldsymbol{\beta}((\ell-1)N+1). \quad (3.4)$$

Remark 1. To the best of our knowledge, the established oracle inequalities for QR under dependence represent a new contribution and are technically nontrivial. To place these contributions in context, we begin with a concise overview of the analytical framework, highlighting the key methodological innovations that arise at each stage of the analysis.

• **Novel decomposition.** First, we introduce a donut-shaped decomposition that has not appeared in the context of nonparametric regression with neural networks. Specifically, we introduce donut-shaped sets, which allow us to decompose the probability bound $\mathbb{P}(\|\widehat{f} - f_0\|_2 > \delta_\star)$ into manageable components by bounding each term separately, where $\delta_\star = c_8\big(\delta_a + \delta_b + \sqrt{u/m_{\max}}\big)$ for RNNs and $\delta_\star = c_9\big(\delta_a + \delta_b + \sqrt{u/m_{\max}}\big)$ for SRNNs. This decomposition differs from the direct argument used in Eq. (16) of Jiao et al. [2024], whose analysis of the excess risk critically relies on the squared loss. To handle the check loss function, we develop the novel decomposition described above.

• **Refined blocking technique.** Next, we introduce a refined blocking technique to relate the mixing sequence to its i.i.d. counterparts, which differs from the approach in Jiao et al. [2024]. In Step 2 of their proof of Theorem 13, part of the data within each partition is discarded to facilitate analysis under the squared loss. In comparison, our new partitioning procedure, i.e., Figure 2, retains all observations, thereby enabling a more comprehensive analysis under dependent data. Moreover, while Lemma 16 in Jiao et al. [2024] plays a central role in their argument, it cannot be directly applied to our setting. Instead, we employ a probabilistic counterpart, i.e., Lemma 5, to carry out our analysis.

• **Sharper inequality**. Finally, we develop a novel and sharp empirical process inequality, i.e., Lemma 7, that underpins the tightness of our oracle inequality. One reason the result of Shen et al. [2021] lacks tightness is that their analysis relies on a non-sharp application of the Bernstein inequality. To address this limitation, we derive Lemma 7 by building on Theorem 7.3 of Bousquet [2003] and Corollary 5.1 of Chernozhukov et al. [2014].

For both function classes, the non-asymptotic bound in Theorem 3 comprises three components: the approximation error δ_a, the stochastic error δ_b, and a dependence-adjustment term that accounts for the discrepancy between dependent and independent sequences. The integer ℓ serves as a key parameter, commonly introduced in time series analysis [Nobel and Dembo, 1993, Yu, 1994], to bridge the behavior of mixing sequences and their i.i.d counterparts. By appropriately tuning the network parameters and selecting ℓ to balance the trade-offs among these components, the ensuing theorems establish the convergence rates of the corresponding estimators when the target function f_0 exhibits a hierarchical interaction structure.

Theorem 4. Let $\mathcal{RNN}_{d_{\mathbf{x}},1}(W,L,K)$ be the hypothesis class $\mathcal{F}$ and assume that the probability measure Π on $[0,1]^{d_{\mathbf{x}}\times N}$ is absolutely continuous with respect to the Lebesgue measure.

(i) Suppose Assumption 1 and Assumption 2 hold and $\{(\mathbf{x}_t,y_t)\}_{t=1}^n$ is a stationary exponentially $\boldsymbol{\beta}$-mixing sequence. Let $W_0,L_0\geq 3$ satisfy $W_0L_0\asymp \left(n/(\log n)^{(6+1/r)}\right)^{1/(4\gamma^\star+2)}$, and define W and L according to (3.1). Then, there exists a constant $c_{10}>0$ such that the ERM estimator $\widehat{f}$ in (2.1) satisfies

$$\mathbb{P}\left[\|\widehat{f}-f_0\|_2\geq c_{10}\left(\left(\frac{(\log n)^{(6+1/r)}}{n}\right)^{\gamma^\star/(2\gamma^\star+1)}+\sqrt{\frac{u(\log n)^{1/r}}{n}}\right)\right]\lesssim \frac{(\log n)^{1/r}}{e^u}+\frac{\log n}{n}.$$

An immediate consequence is that $\|\widehat{f}-f_0\|_2=\mathcal{O}_p\left(n^{-\gamma^\star/(2\gamma^\star+1)}(\log n)^{(6+1/r)\gamma^\star/(2\gamma^\star+1)}\right)$.

(ii) Suppose Assumption 1 and Assumption 2 hold and $\{(\mathbf{x}_t,y_t)\}_{t=1}^n$ is a stationary algebraically $\boldsymbol{\beta}$-mixing sequence. Let $W_0,L_0\geq 3$ satisfy $W_0L_0\asymp \left(n^{(1-1/r)}/(\log n)^7\right)^{1/(4\gamma^\star+2)}$, and define W and L according to (3.1). Then, there exists a constant $c_{11}>0$ such that the ERM estimator $\widehat{f}$ in (2.1) satisfies

$$\mathbb{P}\left[\|\widehat{f}-f_0\|_2\geq c_{11}\left(\left(\frac{(\log n)^{7}}{n^{1-1/r}}\right)^{\gamma^\star/(2\gamma^\star+1)}+\sqrt{\frac{u\log n}{n^{1-1/r}}}\right)\right]\lesssim \frac{n^{1/r}\log n}{e^u}+(\log n)^{(1-r)}.$$

An immediate consequence is that $\|\widehat{f}-f_0\|_2=\mathcal{O}_p\left(n^{-(1-1/r)\gamma^\star/(2\gamma^\star+1)}(\log n)^{(7\gamma^\star/(2\gamma^\star+1))}\right)$.

Theorem 4 shows that the RNN-based QR estimator $\widehat{f}$, with a suitably chosen network architecture, achieves the minimax-optimal convergence rate $n^{-2\gamma^\star/(2\gamma^\star+1)}$ under the squared L_2 norm in a stationary exponentially $\boldsymbol{\beta}$-mixing setting, up to a logarithmic factor. Under stationary algebraically $\boldsymbol{\beta}$-mixing conditions, the convergence rate under squared L_2 norm slows to $n^{-(1-1/r)2\gamma^\star/(2\gamma^\star+1)}$. Nevertheless, as $r\to\infty$, the exponent $-(1-1/r)2\gamma^\star/(2\gamma^\star+1)$ approaches $-2\gamma^\star/(2\gamma^\star+1)$ for fixed $\gamma^\star$, indicating that the convergence rate becomes arbitrarily close to the optimal rate. Our results differ from prior QR studies in two key aspects. First, unlike the FNNs used in previous works [Shen et al., 2021, 2024], we employ RNNs as the approximating function class. Second, rather than assuming independent covariates as in Sangnier et al. [2016] and Padilla et al. [2022], we consider a more general dependence structure, where the sequence $\{(\mathbf{x}_t,y_t)\}_{t=1}^n$ is stationary and $\boldsymbol{\beta}$-mixing. Furthermore, compared to the convergence rate of RNNs in nonparametric regression [Jiao et al., 2024], our results capture intrinsic low-dimensional structures of the target function, thereby achieving optimal performance while circumventing the curse of dimensionality.

The following theorem establishes the convergence rate of the SRNN-based estimator.

Theorem 5. Let $\mathcal{SRNN}_{d_{\mathbf{x}},1}(W,L,K,s)$ be the hypothesis class $\mathcal{F}$ and assume that the probability measure Π on $[0,1]^{d_{\mathbf{x}}\times N}$ is absolutely continuous with respect to the Lebesgue measure.

(i) Suppose Assumption 1 and Assumption 2 hold and $\{(\mathbf{x}_t,y_t)\}_{t=1}^n$ forms a stationary exponentially $\boldsymbol{\beta}$-mixing sequence. Let $W_0\geq \sup_{(\beta,t)\in\mathcal{P}}\max\{(\beta+1)^t,(K+1)e^t\}$ and $L_0\geq 3$ satisfy $W_0\asymp (n/(\log n)^{(4+1/r)})^{1/(2\gamma^\star+1)}$ and $L_0\asymp\log n$. Define W, L, and s according to (3.2). Then, there exists a constant $c_{12}>0$ such that the ERM estimator $\widehat{f}$ in (2.1) satisfies

$$\mathbb{P}\left[\|\widehat{f}-f_0\|_2\geq c_{12}\left(\left(\frac{(\log n)^{(4+1/r)}}{n}\right)^{\gamma^\star/(2\gamma^\star+1)}+\sqrt{\frac{u(\log n)^{1/r}}{n}}\right)\right]\lesssim \frac{(\log n)^{1/r}}{e^u}+\frac{\log n}{n}.$$

An immediate consequence is that $\|\widehat{f}-f_0\|_2=\mathcal{O}_p\left(n^{-\gamma^\star/(2\gamma^\star+1)}(\log n)^{((4+1/r)\gamma^\star/(2\gamma^\star+1))}\right)$.

(ii) Suppose Assumption 1 and Assumption 2 hold and $\{(\mathbf{x}_t,y_t)\}_{t=1}^n$ forms a stationary algebraically $\boldsymbol{\beta}$-mixing sequence. Let $W_0\geq \sup_{(\beta,t)\in\mathcal{P}}\max\{(\beta+1)^t,(K+1)e^t\}$ and $L_0\geq 3$ satisfy $W_0\asymp \left(n^{(1-1/r)}/(\log n)^5\right)^{1/(2\gamma^\star+1)}$ and $L_0\asymp\log n$. Define W, L, and s according to (3.2). Then, there exists a constant $c_{13}>0$ such that the ERM estimator $\widehat{f}$ in (2.1) satisfies

$$\mathbb{P}\left[\|\widehat{f}-f_0\|_2\geq c_{13}\left(\left(\frac{(\log n)^{5}}{n^{1-1/r}}\right)^{\gamma^\star/(2\gamma^\star+1)}+\sqrt{\frac{u\log n}{n^{1-1/r}}}\right)\right]\lesssim \frac{n^{1/r}\log n}{e^u}+(\log n)^{(1-r)}.$$

An immediate consequence is that $\|\widehat{f}-f_0\|_2=\mathcal{O}_p\left(n^{-(1-1/r)\gamma^\star/(2\gamma^\star+1)}(\log n)^{(5\gamma^\star/(2\gamma^\star+1))}\right)$.

Similar to Theorem 4, under stationary exponentially $\boldsymbol{\beta}$-mixing conditions, $\widehat{f}$ attains the minimax-optimal convergence rate, up to a logarithmic factor. Under algebraic $\boldsymbol{\beta}$-mixing, the convergence rate is slower but approaches the optimal rate as $r \to \infty$ for fixed $\gamma^\star$. Compared to Theorem 4, Theorem 5 imposes more restrictions on the choice of width W and length L of SRNNs, owing to technical reasons. However, when fixing the length L to be the same for both theorems, Theorem 5 provides guarantees for wider SRNNs. This aligns with the practical setting better because sparsity is commonly considered for deep and wide neural networks.

4 Numerical Study

In this section, we conduct numerical experiments to evaluate the finite-sample performance of RNN- and SRNN-based QR estimators in comparison to quantile random forest (QRF) and FNN-based estimators. All experiments are implemented in Python. The QRF estimator is trained using the `scikit-garden` package, and the number of trees is set as 100. For all NN-based estimators, we employ early stopping to mitigate overfitting, as proposed by [Raskutti et al., 2014]. Specifically, the dataset is split into training (80%) and validation (20%) sets. At the end of each epoch, the model's validation performance is evaluated using the mean check loss, and training is terminated if the validation loss does not improve for 20 consecutive epochs. For the FNN-based estimator, we adopt $L = 2$ hidden layers with widths $W_1 = W_2 = 200$, following the architecture in Padilla et al. [2022]. For the RNN- and SRNN-based estimators, we use $L = 3$ layers with hidden width $W = 100$ and ReLU activation. To induce sparsity, we prune the smallest 40% of parameters and finetune the remaining weights post-training. The hyperparameters above are fixed throughout the main experiments, and we further investigate the impact of varying these parameters in Appendix D. All neural networks are implemented in `PyTorch`.

We conduct experiments on the following two nonlinear AR models at quantile levels $\tau = 0.1$ and $\tau = 0.5$ to validate the theoretical results established in our analysis.

• **Model 1(SIM$_1$-1):**

$$Y_t = (\Phi(-Z_t) - 0.5)Y_{t-1} + (\Phi(2Z_t) - 0.6)Y_{t-2} + \epsilon_t, \tag{4.1}$$
$$Z_t = \sum_{i=1}^{5} (-1)^{i-1}(Y_{t-2i} + Y_{t-2i+1}),$$

where $\Phi(\cdot)$ is the standard normal distribution function.

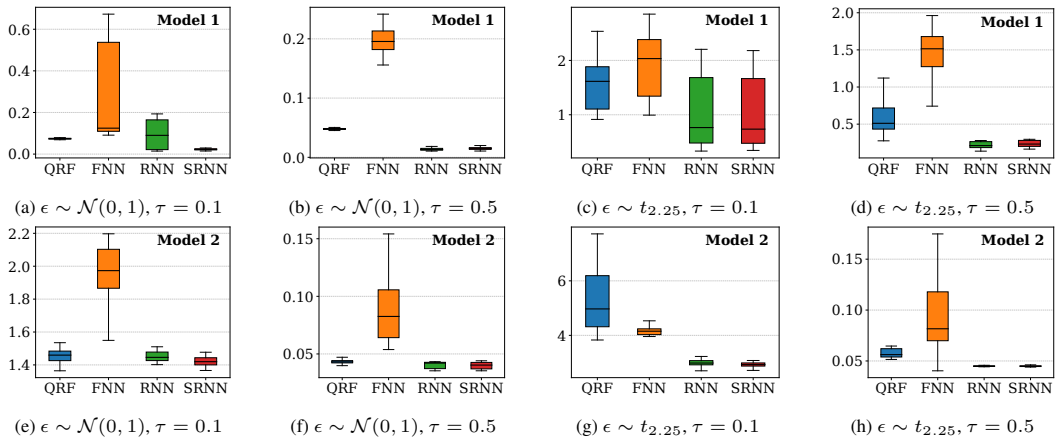
• **Model 2:**

$$Y_t = 0.2h_1(Y_{t-1}, \ldots, Y_{t-10}) + h_2(Y_{t-1}, \ldots, Y_{t-10})\epsilon_t, \tag{4.2}$$

where

$$h_1(Y_{t-1}, \ldots, Y_{t-10}) = \cos(2\pi Y_{t-1}) + \frac{1}{1 + e^{-(Y_{t-2}+Y_{t-3})}} + \frac{1}{(1 + Y_{t-4} + Y_{t-5})^4 + 1}$$
$$+ \frac{Y_{t-6}}{|Y_{t-6}| + e^{Y_{t-7}Y_{t-8}}} + \cos(\pi Y_{t-9})\cos(\pi Y_{t-10}),$$
$$h_2(Y_{t-1}, \ldots, Y_{t-10}) = \sin\left(\frac{\pi(Y_{t-1} + Y_{t-2})}{2}\right) + \frac{\ln\left(1 + Y_{t-3}^2 Y_{t-4}^2 Y_{t-5}^2\right)}{1 + Y_{t-3}^2 Y_{t-4}^2 Y_{t-5}^2}$$
$$+ \frac{1}{1 + e^{-(Y_{t-6}+Y_{t-7}+Y_{t-8})}} + \sin(\pi Y_{t-9})\sin(\pi Y_{t-10}).$$

In the experiments, we examine both light-tailed and heavy-tailed noise distributions: the standard normal distribution $\mathcal{N}(0, 1)$ and a scaled Student's t-distribution with 2.25 degrees of freedom $t_{2.25}$, respectively. Theorem 1 in Chen and Chen [2000] ensures that the data generated from (4.1) and (4.2) are strictly stationary and exponentially $\boldsymbol{\beta}$-mixing. For each model, we generate 110,000 data points, discard the first 100 as a burn-in period, and split the remaining data by assigning the last 100,000 points to the test set and the rest to the training set. For each trained estimator, we compute the mean squared error (MSE) on the test set. We perform 500 Monte Carlo repetitions for each experiment. As shown in Figure 1, the RNN- and SRNN-based estimators exhibit significantly lower variance and superior overall performance compared to the QRF- and FNN-based estimators.



(a) $\epsilon \sim \mathcal{N}(0,1), \tau = 0.1$ (b) $\epsilon \sim \mathcal{N}(0,1), \tau = 0.5$ (c) $\epsilon \sim t_{2.25}, \tau = 0.1$ (d) $\epsilon \sim t_{2.25}, \tau = 0.5$

(e) $\epsilon \sim \mathcal{N}(0,1), \tau = 0.1$ (f) $\epsilon \sim \mathcal{N}(0,1), \tau = 0.5$ (g) $\epsilon \sim t_{2.25}, \tau = 0.1$ (h) $\epsilon \sim t_{2.25}, \tau = 0.5$

Figure 1: MSE comparison for Model 1 and Model 2 with different noise distributions and quantile levels.

5 Application

In this section, we conduct experiments on the DJIA dataset[3] to empirically validate our approach under the stationarity assumption. To further assess robustness in nonstationary settings, we provide a complementary case study on GDP forecasting in Appendix C. Since log-returns of stock prices are widely regarded as approximately stationary, the DJIA dataset is well-suited for evaluation in the stationary environment. We partition the data chronologically, allocating the first 19 years for training and the final year for evaluation. We assess the out-of-sample predictive performance of RNN-, SRNN-, FNN-, and QRF-based estimators in a 30-business-day-ahead forecasting task, with implementations following the specifications in Section 4. All models are trained by minimizing the empirical check loss and evaluated on the test set using the same criterion.

Table 1 presents the mean empirical check loss at five quantile levels. The results show that the RNN and SRNN estimators consistently outperform FNN and QRF methods at all quantile levels. Moreover, the SRNNs achieve predictive accuracy comparable to RNNs while inducing sparsity. These empirical findings are consistent with, and provide validation for, our theoretical results.

Model	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
QRF	0.456	0.698	0.817	0.616	0.365
FNN	0.538	0.735	0.810	0.662	0.404
RNN	0.410	**0.640**	0.760	0.562	0.306
SRNN	**0.406**	0.647	**0.759**	**0.561**	**0.305**

Table 1: Out-of-sample prediction errors at different quantiles for DJIA growth analysis.

6 Conclusion

This study investigates the convergence properties of nonparametric quantile regression using RNNs and SRNNs. Error bounds are derived for the approximation of functions within a hierarchical interaction model using RNNs and SRNNs, respectively. Based on these error bounds, we demonstrate that, for a stationary, exponentially β-mixing sequence of n observations, the empirical risk minimizers of both RNN- and SRNN-based methods achieve the optimal convergence rate.

Future research could explore the approximation capabilities of RNNs and SRNNs for nonparametric quantile process regression. Instead of focusing on a fixed quantile level τ, a useful extension would involve approximating the entire quantile process indexed by $\tau \in [\tau_L, \tau_U] \subseteq (0,1)$, which complements the results of Shen et al. [2024] using FNNs.

[3]We use DJIA data from Jan 1, 2000, to Dec 31, 2020, obtained from `https://www.investing.com`.

Acknowledgment

Zhao Ren was supported in part by the National Science Foundation (DMS-2113568) and the National Institutes of Health (NIGMS R01GM157600). Wen-Xin Zhou was supported in part by the National Science Foundation (DMS-2401268) and the Australian Research Council Discovery Project Grant (DP230100147).

References

T. Adrian, N. Boyarchenko, and D. Giannone. Vulnerable growth. *American Economic Review*, 109 (4):1263–1289, 2019.

M. Anthony and P. L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 2009.

P. L. Bartlett, N. Harvey, C. Liaw, and A. Mehrabian. Nearly-tight VC-dimension and pseudodimension bounds for piecewise linear neural networks. *Journal of Machine Learning Research*, 20(63): 1–17, 2019.

B. Bauer and M. Kohler. On deep learning as a remedy for the curse of dimensionality in nonparametric regression. *The Annals of Statistics*, 47(4):2261–2285, 2019.

A. Belloni and V. Chernozhukov. ℓ1-penalized quantile regression in high-dimensional sparse models. *The Annals of Statistics*, 39(1):82–130, 2011.

A. Belloni, V. Chernozhukov, D. Chetverikov, and I. Fernández-Val. Conditional quantile processes based on series or many regressors. *Journal of Econometrics*, 213(1):4–29, 2019.

O. Bousquet. Concentration inequalities for sub-additive functions using the entropy method. In *Stochastic Inequalities and Applications*, pages 213–247, 2003.

R. C. Bradley. Absolute regularity and functions of Markov chains. *Stochastic Processes and Their Applications*, 14(1):67–77, 1983.

M. Chen and G. Chen. Geometric ergodicity of nonlinear autoregressive models with changing conditional variances. *Canadian Journal of Statistics*, 28(3):605–614, 2000.

M. Chen, X. Li, and T. Zhao. On generalization bounds of a family of recurrent neural networks. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 1233–1243, 2020.

X. Cheng, K. Huang, and S. Ma. Generalization and risk bounds for recurrent neural networks. *Neurocomputing*, 616:128825, 2025.

V. Chernozhukov, D. Chetverikov, and K. Kato. Gaussian approximation of suprema of empirical processes. *The Annals of Statistics*, 42(4):1564–1597, 2014.

J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint*, arXiv:1412.3555, 2014.

J. Fan, Y. Gu, and W.-X. Zhou. How do noise tails impact on deep ReLU networks? *The Annals of Statistics*, 52(4):1845–1871, 2024.

X. Feng, X. He, Y. Jiao, L. Kang, and C. Wang. Deep nonparametric quantile regression under covariate shift. *Journal of Machine Learning Research*, 25(385):1–50, 2024.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

Y. Jiao, G. Shen, Y. Lin, and J. Huang. Deep nonparametric regression on approximate manifolds: Nonasymptotic error bounds with polynomial prefactors. *The Annals of Statistics*, 51(2):691–716, 2023.

Y. Jiao, Y. Wang, and B. Yan. Approximation bounds for recurrent neural networks with application to regression. *arXiv preprint*, arXiv:2409.05577, 2024.

R. Koenker and G. Bassett. Regression quantiles. *Econometrica*, 46(1):33–50, 1978.

R. Koenker and Z. Xiao. Quantile autoregression. *Journal of the American Statistical Association*, 101(475):980–990, 2006.

M. Kohler and S. Langer. On the rate of convergence of fully connected deep neural network regression estimates. *The Annals of Statistics*, 49(4):2231–2249, 2021.

D. J. McDonald, C. R. Shalizi, and M. Schervish. Estimating beta-mixing coefficients. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 516–524, 2011.

D. J. McDonald, C. R. Shalizi, and M. Schervish. Estimating beta-mixing coefficients via histograms. *Electronic Journal of Statistics*, 9(2):2855 – 2883, 2015.

D. J. McDonald, C. R. Shalizi, and M. Schervish. Nonparametric risk bounds for time-series forecasting. *Journal of Machine Learning Research*, 18(32):1–40, 2017.

N. Meinshausen. Quantile regression forests. *Journal of Machine Learning Research*, 7(6):983–999, 2006.

M. Mohri and A. Rostamizadeh. Stability bounds for stationary φ-mixing and β-mixing processes. *Journal of Machine Learning Research*, 11(2):789–814, 2010.

V. Nair and G. E. Hinton. Rectified linear units improve restricted Boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, pages 807–814, 2010.

A. Nobel and A. Dembo. A note on uniform laws of averages for dependent processes. *Statistics & Probability Letters*, 17(3):169–172, 1993.

O. H. M. Padilla, W. Tansey, and Y. Chen. Quantile regression with ReLU networks: Estimators and minimax rates. *Journal of Machine Learning Research*, 23(247):1–42, 2022.

N. Phandoidaen and S. Richter. Forecasting time series with encoder-decoder neural networks. *arXiv preprint*, arXiv:2009.08848, 2020.

G. Raskutti, M. J. Wainwright, and B. Yu. Early stopping and non-parametric regression: An optimal data-dependent stopping rule. *Journal of Machine Learning Research*, 15(1):335–366, 2014.

D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

M. Sangnier, O. Fercoq, and F. d'Alché Buc. Joint quantile regression in vector-valued RKHSs. In *Advances in Neural Information Processing Systems 29 (NIPS)*, pages 3693–3701, 2016.

J. Schmidt-Hieber. Nonparametric regression using deep neural networks with ReLU activation function. *The Annals of Statistics*, 48(4):1875–1897, 2020.

G. Shen, Y. Jiao, Y. Lin, J. L. Horowitz, and J. Huang. Deep quantile regression: Mitigating the curse of dimensionality through composition. *arXiv preprint*, arXiv:2107.04907, 2021.

G. Shen, Y. Jiao, Y. Lin, J. L. Horowitz, and J. Huang. Nonparametric estimation of non-crossing quantile regression process with deep ReQU neural networks. *Journal of Machine Learning Research*, 25(88):1–75, 2024.

G. Shen, R. Dai, G. Wu, S. Luo, C. Shi, and H. Zhu. Deep distributional learning with non-crossing quantile network. *arXiv preprint*, arXiv:2504.08215, 2025.

C. H. Song, G. Hwang, J. H. Lee, and M. Kang. Minimal width for universal property of deep RNN. *Journal of Machine Learning Research*, 24(121):1–41, 2023.

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

I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27 (NIPS)*, pages 3104–3112, 2014.

Z. Tu, F. He, and D. Tao. Understanding generalization in recurrent neural networks. In *Proceedings of the 8th International Conference on Learning Representations (ICLR)*, 2020.

M. Vidyasagar. *Learning and generalisation: with applications to neural networks*. Springer Science & Business Media, 2013.

H. White. Nonparametric estimation of conditional quantiles using neural networks. In *Computing Science and Statistics*, pages 190–199, 1992.

D. Yarotsky. Error bounds for approximations with deep ReLU networks. *Neural Networks*, 94: 103–114, 2017.

B. Yu. Rates of convergence for empirical processes of stationary mixing sequences. *The Annals of Probability*, 22(1):94–116, 1994.

S. Zheng. Qboost: Predicting quantiles with boosting for regression and binary classification. *Expert Systems with Applications*, 39(2):1687–1697, 2012.

Appendix

The appendix is organized as follows: Appendix A provides the proofs of the main theorems presented in Section 3. Appendix B contains the proofs of all auxiliary lemmas referenced in Appendix A. Appendix C presents an additional case study on GDP forecasting that further supports our theoretical findings. Appendix D provides a sensitivity analysis of the hyperparameters for the SRNNs.

A Proofs of Theorems

We first provide the definition of FNNs for our analysis. An FNN $f_\theta : \mathbb{R}^{d_{\mathbf{x}}} \to \mathbb{R}^{d_{\mathbf{y}}}$ can be represented as

$$f_\theta(\mathbf{x}) := f_L \circ \cdots \circ f_1(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^{d_{\mathbf{x}}},$$

where each feedforward layer $f_l(\mathbf{x})$ is defined as

$$f_l(\mathbf{x}) = \begin{cases} \sigma(A_l\mathbf{x} + \mathbf{b}_l), & l \in [L-1], \\ A_l\mathbf{x} + \mathbf{b}_l, & l = L. \end{cases}$$

For each layer l, the weight matrix is $A_l \in \mathbb{R}^{W_l \times W_{l-1}}$ and the bias vector is $\mathbf{b}_l \in \mathbb{R}^{W_l}$. The initial dimension $W_0 = d_{\mathbf{x}}$ and the final dimension $W_L = d_{\mathbf{y}}$ correspond to the input and output sizes, respectively. The complete parameter vector θ is defined as $\theta = (\mathrm{vec}(A_1)^\top, \mathbf{b}_1^\top, \ldots, \mathrm{vec}(A_L)^\top, \mathbf{b}_L^\top)^\top \in \mathbb{R}^{\sum_{l=1}^{L} W_l(W_{l-1}+1)}$. The width W of the network is defined as $\max\{W_1, \ldots, W_{L-1}\}$. We denote $\mathcal{FNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}(W, L, K)$ as a class of FNNs with width W, depth L and bounded output, defined as

$$\mathcal{FNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}(W, L, K) = \left\{ f_\theta : \mathbb{R}^{d_{\mathbf{x}}} \to \mathbb{R}^{d_{\mathbf{y}}} \,\middle|\, \sup_{\mathbf{x} \in \mathbb{R}^{d_{\mathbf{x}}}} \|f_\theta(\mathbf{x})\|_\infty \le K \right\}.$$

Similar to SRNNs, we define $\mathcal{T}(f_l) = \|A_l\|_0 + \|\mathbf{b}_l\|_0$, for $l \in [L]$ and define $\mathcal{T}(f_\theta) = \sum_{l=1}^{L} \mathcal{T}(f_l)$ as the sparsity of the SFNN f_θ. Moreover, we define the class of SFNNs as follows

$$\mathcal{SFNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}(W, L, K, s) = \left\{ f_\theta \,|\, f_\theta \in \mathcal{FNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}(W, L, K) \text{ with } \mathcal{T}(f_\theta) \le s \right\}.$$

To maintain consistent input formatting, if the input to an FNN or SFNN is a sequence $\boldsymbol{X} \in \mathbb{R}^{d_{\boldsymbol{x}} \times N}$, we first stack its columns into a vector and then use this vector as the input to the neural network, that is, $f_\theta(\boldsymbol{X}) = f_\theta(((\mathbf{x}^{(1)})^\top, \ldots, (\mathbf{x}^{(N)})^\top)^\top)$. For notational simplicity, the distinction between representing the input as a sequence $\boldsymbol{X}$ or as its vectorized form is not explicitly made where the context is unambiguous.

For simplicity, we assume $K \ge 1$ throughout and omit it when boundedness is understood or not required in the notation of the function classes $\mathcal{FNN}$, $\mathcal{SFNN}$, $\mathcal{RNN}$ and $\mathcal{SRNN}$.

A.1 Proof of Theorem 1

We first introduce two lemmas: the first establishes an error bound for using FNNs to approximate functions within a hierarchical interaction model, and the second demonstrates that an FNN can be represented by an RNN prediction function.

Lemma 1 (Proposition 3.4 in Fan et al. [2024])**.** Given a hierarchical interaction model $\mathcal{H}_d^l(\mathcal{P}, K)$, for any W_0, $L_0 \ge 3$, and a probability measure μ on $[0,1]^{d_{\mathbf{x}} \times N}$ that is absolutely continuous with respect to the Lebesgue measure, the following inequality holds

$$\sup_{f_0 \in \mathcal{H}_d^l(\mathcal{P},K)} \inf_{\bar{f} \in \mathcal{FNN}_{d,1}(W,L,K)} \left\{ \int_{[0,1]^{d_{\mathbf{x}} \times N}} |\bar{f}(\mathbf{x}) - f_0(\mathbf{x})|^2 \mu(\mathrm{d}\mathbf{x}) \right\}^{1/2} \le c_{14}(W_0 L_0)^{-2\gamma^\star},$$

where $W = c_{15}\lceil W_0 \log W_0 \rceil$ and $L = c_{16}\lceil L_0 \log L_0 \rceil$. Here, the positive constants c_{14}–c_{16} depend on $(l, \mathcal{P}, K)$.

Lemma 2 (Proposition 2 in Jiao et al. [2024])**.** For any FNN $\bar{f} \in \mathcal{FNN}_{d_{\mathbf{x}} \times N, d_{\mathbf{y}}}(W, L)$, there exists an RNN prediction function $f \in \mathcal{RNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}((d_{\mathbf{x}}+1)W + 1, 2L + 2N)$ such that

$$\bar{f}(\boldsymbol{X}) = f(\boldsymbol{X}), \quad \boldsymbol{X} \in [0,1]^{d_{\mathbf{x}} \times N}.$$

Proof of Theorem 1. Applying Lemma 1 to $\mathcal{H}^l_{d_{\mathbf{x}}\times N}(\mathcal{P}, K)$, there exist positive constants c_{14}–c_{16} such that

$$\inf_{\bar{f}\in\mathcal{FNN}_{d_{\mathbf{x}}\times N,1}(W',L',K)} \left\{\int_{[0,1]^{d_{\mathbf{x}}\times N}} |\bar{f}(\mathbf{x}) - f_0(\mathbf{x})|^2 \mu(\mathrm{d}\mathbf{x})\right\}^{1/2} \leq c_{14}(W_0 L_0)^{-2\gamma^\star},$$

where $W' = c_{15}\lceil W_0 \log W_0\rceil$, $L' = c_{16}\lceil L_0 \log L_0\rceil$.

By Lemma 2, we have that for any $\bar{f} \in \mathcal{FNN}_{d_{\mathbf{x}}\times N,1}(W', L', K)$ there exists an RNN prediction function $f \in \mathcal{RNN}_{d_{\mathbf{x}},1}((d_{\mathbf{x}}+1)W'+1, 2L'+2N, K)$ such that $f(\boldsymbol{X}) = \bar{f}(\boldsymbol{X})$, $\boldsymbol{X} \in [0,1]^{d_{\mathbf{x}}\times N}$.

Combining the two results above, we complete the proof of the theorem.

□

A.2 Proof of Theorem 2

Following a similar line of reasoning as in the proof of Theorem 1, we introduce two key lemmas that extend Lemma 1 and Lemma 2 to the sparse setting. The first lemma establishes an error bound for approximating functions under a hierarchical interaction model using SFNNs. The second lemma shows that an SFNN can be equivalently represented by an SRNN prediction function. Proofs of these lemmas are deferred to Appendix B.1 and Appendix B.2, respectively.

Lemma 3. Given a hierarchical interaction model $\mathcal{H}^l_d(\mathcal{P}, K)$, for any $W_0 \geq \sup_{(\beta,t)\in\mathcal{P}} \max\{(\beta+1)^t, (K+1)e^t\}$, $L_0 \geq 1$, and a probability measure μ on $[0,1]^{d_{\mathbf{x}}\times N}$ that is absolutely continuous with respect to the Lebesgue measure, there exist W, L, $s > 0$ such that the following inequality holds

$$\sup_{f_0\in\mathcal{H}^l_d(\mathcal{P},K)} \inf_{\bar{f}\in\mathcal{SFNN}_{d,1}(W,L,K,s)} \left\{\int_{[0,1]^{d_{\mathbf{x}}\times N}} |\bar{f}(\mathbf{x}) - f_0(\mathbf{x})|^2 \mu(\mathrm{d}\mathbf{x})\right\}^{1/2} \leq c_{17}\Big(W_0 2^{-L_0} + W_0^{-\gamma^\star}\Big),$$

where $W = c_{18}W_0$, $L = c_{19}L_0$ and $s = c_{20}L_0W_0$. Here c_{17}–c_{20} are positive constants depending on $(l, \mathcal{P}, K)$.

Lemma 4. For any SFNN $\bar{f} \in \mathcal{SFNN}_{d_{\mathbf{x}}\times N,1}(W, L, s)$, there exists an SRNN prediction function $f \in \mathcal{SRNN}_{d_{\mathbf{x}},1}\left((d_{\mathbf{x}}+1)W+1, 2L+2N, s+3(d_{\mathbf{x}}+1)WN+6(L+N)(d_{\mathbf{x}}+1)W\right)$ such that

$$\bar{f}(\boldsymbol{X}) = f(\boldsymbol{X}), \quad \boldsymbol{X} \in [0,1]^{d_{\mathbf{x}}\times N}.$$

Proof of Theorem 2. Applying Lemma 3 to $\mathcal{H}^l_{d_{\mathbf{x}}\times N}(\mathcal{P}, K)$, there exist positive constants c_{17}–c_{20} such that

$$\inf_{\bar{f}\in\mathcal{SFNN}_{d_{\mathbf{x}}\times N,1}(W',L',K,s')} \left\{\int_{[0,1]^{d_{\mathbf{x}}\times N}} |\bar{f}(\mathbf{x}) - f_0(\mathbf{x})|^2 \mu(\mathrm{d}\mathbf{x})\right\}^{1/2} \leq c_{17}\Big(W_0 2^{-L_0} + W_0^{-\gamma^\star}\Big),$$

where $W' = c_{18}W_0$, $L' = c_{19}L_0$ and $s' = c_{20}L_0W_0$.

Furthermore, Lemma 4 establishes that for any SFNN $\bar{f} \in \mathcal{SFNN}_{d_{\mathbf{x}}\times N,1}(W', L', K, s')$, there exists an SRNN prediction function $f \in \mathcal{SRNN}_{d_{\mathbf{x}},1}((d_{\mathbf{x}}+1)W'+1, 2L'+2N, K, s'+3(d_{\mathbf{x}}+1)W'N+6(L'+N)(d_{\mathbf{x}}+1)W')$ such that $f(\boldsymbol{X}) = \bar{f}(\boldsymbol{X})$, $\boldsymbol{X} \in [0,1]^{d_{\mathbf{x}}\times N}$.

Combining the two results above, we complete the proof of the theorem.

□

A.3 Proof of Theorem 3

As a preparatory step for the proof of Theorem 3, we present the following three lemmas:

Lemma 5 (Lemma 2 in Nobel and Dembo [1993])**.** Let h be a real-valued Borel measurable function, and let $\{\mathbf{z}_t\}_{t=-\infty}^{\infty}$ be a stationary $\boldsymbol{\beta}$-mixing sequence of random vectors. We define a block of length

L as a sequence of L consecutive vectors from $\{\mathbf{z}_t\}_{t=-\infty}^{\infty}$. Consider a sequence $\mathcal{C}$ consisting of m such blocks, where the gap between any two consecutive blocks is $a+1$. That is, for any two consecutive blocks in $\mathcal{C}$, the index of the first vector in the second block exceeds the index of the last vector in the first block by exactly $a+1$. Then for any constant t, the following bound holds

$$\left|\mathbb{P}(h(\mathcal{C}) \geq t) - \mathbb{P}(h(\widetilde{\mathcal{C}}) \geq t)\right| \leq m\boldsymbol{\beta}(a+1).$$

Here $\widetilde{\mathcal{C}}$ comprises m independent blocks, each drawn from the same distribution as in $\mathcal{C}$.

The lemma above enables us to extend concentration results for i.i.d. data to the setting of $\boldsymbol{\beta}$-mixing data. Similar results can be found in Yu [1994], Vidyasagar [2013], and McDonald et al. [2017].

Lemma 6. Under Assumption 1, for any function $f : [0,1]^{d\times N} \to [-K, K]$, the population check loss function satisfies

$$c_{21}\|f - f_0\|_2^2 \leq \mathcal{R}_\tau(f) - \mathcal{R}_\tau(f_0) \leq c_{22}\|f - f_0\|_2^2,$$

where $c_{21} = \min\left\{\underline{p}/(8K), \underline{p}^2/(32Kl_0)\right\}$ and $c_{22} = \bar{p}/2$.

In the analysis of Theorem 3, we consider a generic function class $\mathcal{F}$. The main difference between (i) and (ii) lies in the properties of δ_b, which are determined by whether $\mathcal{F} = \mathcal{RNN}_{d_\mathbf{x},1}(W, L, K)$ or $\mathcal{SRNN}_{d_\mathbf{x},1}(W, L, K, s)$. We present a key concentration result for each of these function classes with their corresponding δ_b. For any $\mathcal{F}$, we define $\mathcal{F}(\delta) = \{f \in \mathcal{F} \mid \|f - f_0\|_2 \leq \delta\}$. For convenience, we denote $\mathbf{z}_t = ((\mathbf{x}_{t-N+1}, y_{t-N+1}), \ldots, (\mathbf{x}_t, y_t))$ and $S' = \{\mathbf{z}_t\}_{t=N}^{n}$. Additionally, for any f, we introduce the following notation:

$$g(f, Z) = \rho_\tau(Y_N - f(X_1, \ldots, X_N)) - \rho_\tau(Y_N - f_0(X_1, \ldots, X_N)). \tag{A.1}$$

With the above definition, we have the following lemma.

Lemma 7. Let $\mathbf{z}_1, \ldots, \mathbf{z}_n$ be i.i.d. random vectors drawn from the distribution of Z, and let g be defined as in (A.1). We choose $\delta_b = WL\sqrt{\log(\max\{W, L\})\log n/n}$ for RNNs, and $\delta_b = WL\sqrt{\log(WL^2)\log n/n}$ for SRNNs. Then there exists a universal constant $c_{23} > 0$ such that for any $\delta \geq \delta_b$ and $0 \leq x \leq n\delta^2$, the following inequality holds uniformly over both $\mathcal{F} = \mathcal{RNN}_{d_\mathbf{x},1}(W, L, K)$ and $\mathcal{F} = \mathcal{SRNN}_{d_\mathbf{x},1}(W, L, K, s)$

$$\mathbb{P}\left[\sup_{f\in\mathcal{F}(\delta)}\left|\frac{1}{n}\sum_{i=1}^{n} g(f, \mathbf{z}_i) - \mathbb{E}_Z[g(f, Z)]\right| \geq c_{23}\delta\left(\delta_b + \sqrt{\frac{x}{n}}\right)\right] \leq e^{-x}.$$

Proof of Theorem 3. First, let $m = n - N + 1$ and, for a given $u \geq 1$, define $\delta_\star = C(\delta_a + \delta_b + \sqrt{2N\ell u/m})$, where C is given by

$$C = \max\left\{\sqrt{8c_{22}/c_{21}}, 32c_{23}/c_{21}\right\} \geq 1. \tag{A.2}$$

Here, the constants c_{21} and c_{22} are specified in Lemma 6 and c_{23} is specified in Lemma 7. We emphasize that in Theorem 3, both c_8 and c_9 can be set to $4C$. Next, for an integer $i \in \mathbb{N}$, we define the donut-shaped sets as

$$\mathcal{D}_i := \mathcal{F}(2^i\delta_\star) \setminus \mathcal{F}(2^{i-1}\delta_\star) = \left\{f \in \mathcal{F} : 2^{i-1}\delta_\star < \|f - f_0\|_2 \leq 2^i\delta_\star\right\}.$$

With the definition of $\mathcal{D}_i$, we can write

$$\mathbb{P}\left(\|\widehat{f} - f_0\|_2 > \delta_\star\right) \leq \sum_{i=1}^{\lfloor\log_2(2K/\delta_\star)\rfloor} \mathbb{P}\left(\widehat{f} \in \mathcal{D}_i\right). \tag{A.3}$$

Therefore, it reduces to bounding each probability $\mathbb{P}\left(\widehat{f} \in \mathcal{D}_i\right)$ separately. Following Lemma 6, for any $f \in \mathcal{D}_i$, we have

$$c_{21}2^{2i-2}\delta_\star^2 \leq c_{21}\|f - f_0\|_2^2 \leq \mathcal{R}_\tau(f) - \mathcal{R}_\tau(f_0). \tag{A.4}$$

We next derive an upper bound of the right-hand side of (A.4). By the definition of δ_a, there exists $f_m \in \mathcal{F}$ such that $\|f_m - f_0\|_2 \leq 2\delta_a$. Now, if $\widehat{f} \in \mathcal{D}_i$, we have

$$\mathcal{R}_\tau(\widehat{f}) - \mathcal{R}_\tau(f_0)$$

$$
\begin{aligned}
=&\mathcal{R}_\tau(\widehat{f}) - \mathcal{R}_n(\widehat{f}) + \big(\mathcal{R}_n(\widehat{f}) - \mathcal{R}_n(f_m)\big) + \mathcal{R}_n(f_m) - \mathcal{R}_\tau(f_m) + \mathcal{R}_\tau(f_m) - \mathcal{R}_\tau(f_0) \\
\leq&\mathcal{R}_\tau(\widehat{f}) - \mathcal{R}_n(\widehat{f}) + \mathcal{R}_n(f_m) - \mathcal{R}_\tau(f_m) + \mathcal{R}_\tau(f_m) - \mathcal{R}_\tau(f_0),
\end{aligned}
$$

where the last inequality follows from the definition of $\widehat{f}$. By Lemma 6, it follows that $\mathcal{R}_\tau(f_m) - \mathcal{R}_\tau(f_0) \leq 4c_{22}\delta_a^2$. For any set S, we denote

$$
\Delta_S(f) = \sum_{\mathbf{z}\in S} g(f, \mathbf{z}) - |S|\mathbb{E}_Z[g(f, Z)].
$$

Then the earlier inequality can be further bounded as

$$
\mathcal{R}_\tau(\widehat{f}) - \mathcal{R}_\tau(f_0) \leq \frac{1}{m}\left(\Delta_{S'}(f_m) - \Delta_{S'}(\widehat{f})\right) + 4c_{22}\delta_a^2. \tag{A.5}
$$

Combining (A.4) and (A.5), we obtain an upper bound of the probability $\mathbb{P}\left(\widehat{f} \in \mathcal{D}_i\right)$ as

$$
\mathbb{P}\left(\widehat{f} \in \mathcal{D}_i\right) \leq \mathbb{P}\left(\exists f \in \mathcal{D}_i \text{ such that } \frac{1}{m}\big(\Delta_{S'}(f_m) - \Delta_{S'}(f)\big) \geq \frac{c_{21}}{4}2^{2i}\delta_\star^2 - 4c_{22}\delta_a^2\right). \tag{A.6}
$$

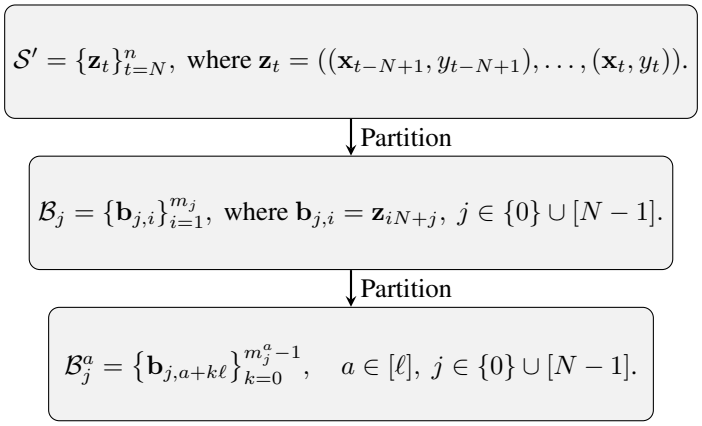


Figure 2: **Illustration of the Partitioning Procedure.** Here, $m_j = |\mathcal{B}_j|$ and $m_j^a = |\mathcal{B}_j^a|$. We first split S' into N sequences. Each sequence $\mathcal{B}_j$ is then partitioned into ℓ equidistant sub-sequences $\mathcal{B}_j^a$.

We next relate the dependent observations to independent observations. We partition S' into $N\ell$ sequences step by step. We first partition it into N equidistant sequences $\{\mathcal{B}_j\}_{j=0}^{N-1}$. For $j \in \{0\} \cup [N-1]$, we define $\mathcal{B}_j := \{\mathbf{b}_{j,i}\}_{i=1}^{m_j}$, where $\mathbf{b}_{j,i} = \mathbf{z}_{iN+j}$, $j \in \{0\} \cup [N-1]$ and $m_j = |\mathcal{B}_j|$. Let $r_1 = m \bmod N$ denote the remainder. Then, for $j < r_1$, $m_j = \lceil m/N \rceil$, and for $j \geq r_1$, $m_j = \lceil m/N \rceil - 1$. We continue partitioning each $\mathcal{B}_j$ into ℓ equidistant sub-sequences $\{\mathcal{B}_j^a\}_{a=1}^{\ell}$. For $a \in [\ell]$, we define $\mathcal{B}_j^a := \{\mathbf{b}_{j,a+k\ell}\}_{k=0}^{m_j^a - 1}$, where $m_j^a = |\mathcal{B}_j^a|$. Let $r_2 = m_j \bmod \ell$ denote the remainder. Then, for $a \leq r_2$, we have $m_j^a = \lceil m_j/\ell \rceil$, and for $a > r_2$, we have $m_j^a = \lceil m_j/\ell \rceil - 1$. The detail is shown in Figure 2. With the partition, we have

$$
\Delta_{S'}(f) = \sum_{j=0}^{N-1}\sum_{a=1}^{\ell}\left(\sum_{\mathbf{b}\in\mathcal{B}_j^a} g(f, \mathbf{b}) - m_j^a\mathbb{E}_Z[g(f, Z)]\right) = \sum_{j=0}^{N-1}\sum_{a=1}^{\ell}\Delta_{\mathcal{B}_j^a}(f).
$$

Plugging it into (A.6) and noting that $f_m \in \mathcal{F}(2^i\delta_\star)$ for all $i \geq 1$ (since $2\delta_a \leq 2^i\delta_\star$ for all $i \geq 1$), we obtain

$$
\begin{aligned}
&\mathbb{P}\left(\widehat{f} \in \mathcal{D}_i\right) \\
\leq&\mathbb{P}\left(\exists f \in \mathcal{D}_i \text{ such that } \frac{1}{m}\sum_{j=0}^{N-1}\sum_{a=1}^{\ell}\left(\Delta_{\mathcal{B}_j^a}(f_m) - \Delta_{\mathcal{B}_j^a}(f)\right) \geq \frac{c_{21}}{4}2^{2i}\delta_\star^2 - 4c_{22}\delta_a^2\right)
\end{aligned}
$$

$$
\begin{aligned}
&\leq \mathbb{P}\left(\exists f \in \mathcal{D}_i \text{ such that } \max_{j,a}\left\{\Delta_{\mathcal{B}_j^a}(f_m) - \Delta_{\mathcal{B}_j^a}(f)\right\} \geq \frac{m}{N\ell}\left(\frac{c_{21}}{4}2^{2i}\delta_\star^2 - 4c_{22}\delta_a^2\right)\right)\\
&\leq \mathbb{P}\left(\max_{j,a}\sup_{f\in\mathcal{F}(2^i\delta_\star)}|\Delta_{\mathcal{B}_j^a}(f)| \geq \frac{m}{N\ell}\left(\frac{c_{21}}{8}2^{2i}\delta_\star^2 - 2c_{22}\delta_a^2\right)\right)\\
&\leq \mathbb{P}\left(\max_{j,a}\sup_{f\in\mathcal{F}(2^i\delta_\star)}|\Delta_{\mathcal{B}_j^a}(f)| \geq \frac{m}{N\ell}\frac{c_{21}}{16}2^{2i}\delta_\star^2\right)\\
&\leq \sum_{j=0}^{N-1}\sum_{a=1}^{\ell}\mathbb{P}\left(\sup_{f\in\mathcal{F}(2^i\delta_\star)}|\Delta_{\mathcal{B}_j^a}(f)| \geq \frac{m}{N\ell}\frac{c_{21}}{16}2^{2i}\delta_\star^2\right),
\end{aligned}
\tag{A.7}
$$

where the fourth inequality follows from the choice of C in (A.2) and the last inequality follows from the union bound.

For each $\mathcal{B}_j^a$, we leverage the property of $\boldsymbol{\beta}$-mixing to relate it to $\widetilde{\mathcal{B}}_j^a$. The key here is that the blocks comprising $\widetilde{\mathcal{B}}_j^a$ are mutually independent, with each block $\widetilde{\mathbf{b}}$ containing exactly N elements. Furthermore, the elements within each block $\widetilde{\mathbf{b}} \in \widetilde{\mathcal{B}}_j^a$ are identically distributed to those in the corresponding block $\mathbf{b} \in \mathcal{B}_j^a$.

For any $j \in \{0\} \cup [N-1]$ and $a \in [\ell]$, we denote $c_i = \frac{m}{N\ell}\frac{c_{21}}{16}2^{2i}\delta_\star^2$. By Lemma 5, we have

$$
\left|\mathbb{P}\left(\sup_{f\in\mathcal{F}(2^i\delta_\star)}|\Delta_{\mathcal{B}_j^a}(f)| \geq c_i\right) - \mathbb{P}\left(\sup_{f\in\mathcal{F}(2^i\delta_\star)}|\Delta_{\widetilde{\mathcal{B}}_j^a}(f)| \geq c_i\right)\right| \leq m_j^a\boldsymbol{\beta}((\ell-1)N+1). \tag{A.8}
$$

For any $j \in \{0\} \cup [N-1]$ and $a \in [\ell]$, we next bound the probability

$$
\begin{aligned}
&\mathbb{P}\left(\sup_{f\in\mathcal{F}(2^i\delta_\star)}|\Delta_{\widetilde{\mathcal{B}}_j^a}(f)| \geq \frac{m}{N\ell}\frac{c_{21}}{16}2^{2i}\delta_\star^2\right)\\
\leq&\mathbb{P}\left(\sup_{f\in\mathcal{F}(2^i\delta_\star)}|\Delta_{\widetilde{\mathcal{B}}_j^a}(f)| \geq m_j^a\frac{c_{21}}{32}2^{2i}\delta_\star^2\right)\\
\leq&\mathbb{P}\left(\sup_{f\in\mathcal{F}(2^i\delta_\star)}|\Delta_{\widetilde{\mathcal{B}}_j^a}(f)| \geq m_j^a\frac{c_{23}}{C}2^{2i}\delta_\star^2\right),
\end{aligned}
$$

where the first inequality follows that $m/(N\ell) \geq m_j^a/2$ and the last inequality follows that $c_{23}/C \leq c_{21}/32$.

To this end, we choose $\delta = 2^i\delta_\star$ and $x = 2^{2i}u$. Since $C \geq 1$, we have $\delta \geq \delta_b$ and $0 \leq x \leq m_j^a\delta^2$. Then, Lemma 7 yields

$$
\begin{aligned}
&\mathbb{P}\left(\sup_{f\in\mathcal{F}(2^i\delta_\star)}|\Delta_{\widetilde{\mathcal{B}}_j^a}(f)| \geq m_j^a\frac{c_{23}}{C}2^{2i}\delta_\star^2\right)\\
=&\mathbb{P}\left(\sup_{f\in\mathcal{F}(2^i\delta_\star)}\left|\frac{1}{m_j^a}\sum_{\mathbf{b}\in\widetilde{\mathcal{B}}_j^a}g(f,\mathbf{b}) - \mathbb{E}_Z[g(f,Z)]\right| \geq \frac{c_{23}}{C}2^{2i}\delta_\star^2\right)\\
\leq&\mathbb{P}\left(\sup_{f\in\mathcal{F}(2^i\delta_\star)}\left|\frac{1}{m_j^a}\sum_{\mathbf{b}\in\widetilde{\mathcal{B}}_j^a}g(f,\mathbf{b}) - \mathbb{E}_Z[g(f,Z)]\right| \geq c_{23}\delta\left(\delta_b + \sqrt{\frac{2N\ell x}{m}}\right)\right)\\
\leq&\mathbb{P}\left(\sup_{f\in\mathcal{F}(2^i\delta_\star)}\left|\frac{1}{m_j^a}\sum_{\mathbf{b}\in\widetilde{\mathcal{B}}_j^a}g(f,\mathbf{b}) - \mathbb{E}_Z[g(f,Z)]\right| \geq c_{23}\delta\left(\delta_b + \sqrt{\frac{x}{m_j^a}}\right)\right)\\
\leq&\exp(-x) = \exp(-2^{2i}u),
\end{aligned}
\tag{A.9}
$$

where the second inequality follows that $m/(N\ell) \leq 2m_j^a$.

Combining (A.9) with (A.3), (A.7) and (A.8) implies

$$\begin{aligned}
&\mathbb{P}\left(\|\widehat{f}-f_0\|_2>\delta_\star\right)\\
\leq& N\ell\sum_{i=1}^{\lfloor\log_2(2K/\delta_\star)\rfloor}\exp(-2^{2i}u)+2m\lfloor\log_2(2K/\delta_\star)\rfloor\boldsymbol{\beta}((\ell-1)N+1)\\
\leq& N\ell\sum_{i=1}^{\infty}\exp(-iu)+2n\lfloor\log_2(2K/\delta_\star)\rfloor\boldsymbol{\beta}((\ell-1)N+1)\\
\lesssim& N\ell e^{-u}+n\lfloor\log_2(2K/\delta_b)\rfloor\boldsymbol{\beta}((\ell-1)N+1),
\end{aligned}$$

where the last inequality uses the fact that $u\geq 1$. This proves the claim.

□

A.4 Proof of Theorem 4

According to Theorem 1, we obtain

$$\delta_a=\inf_{f\in\mathcal{F}}\left\|f-f_0\right\|_2\lesssim(W_0L_0)^{-2\gamma^\star}. \tag{A.10}$$

Recall that

$$\begin{aligned}
\delta_b=&WL\sqrt{\frac{2N\ell\log(\max\{W,L\})\log(2m/(N\ell))}{m}}\\
\lesssim& W_0L_0\log W_0\log L_0\sqrt{\frac{2N\ell\log(\max\{W_0\log W_0,L_0\log L_0\})\log m}{m}}.
\end{aligned} \tag{A.11}$$

We now analyze how these bounds behave under different $\boldsymbol{\beta}$-mixing conditions.

• **Case 1: Exponentially $\boldsymbol{\beta}$-mixing.** Recall that for exponentially $\boldsymbol{\beta}$-mixing, there exist positive constants β_0, β_1, and r such that $\boldsymbol{\beta}(a)\leq\beta_0\exp(-\beta_1a^r)$ for all a. By plugging (A.10) and (A.11) into (3.3), and $\ell\asymp(\log n)^{1/r}$ such that $\boldsymbol{\beta}((\ell-1)N+1)\lesssim 1/n^2$, together with setting $W_0L_0\asymp(n/(\log n)^{(6+1/r)})^{1/(4\gamma^\star+2)}$, we obtain that there exists a constant $c_{10}>0$ such that

$$\mathbb{P}\left[\|\widehat{f}-f_0\|_2\geq c_{10}\left(\left(\frac{(\log n)^{(6+1/r)}}{n}\right)^{\gamma^\star/(2\gamma^\star+1)}+\sqrt{\frac{(\log n)^{1/r}u}{n}}\right)\right]\lesssim\frac{(\log n)^{1/r}}{e^u}+\frac{\log n}{n}.$$

• **Case 2: Algebraically $\boldsymbol{\beta}$-mixing.** Recall that for algebraically $\boldsymbol{\beta}$-mixing, there exist positive constants β_0 and $r>1$ such that $\boldsymbol{\beta}(a)\leq\beta_0/a^r$ for all k. By plugging (A.10) and (A.11) into (3.3), and choosing $\ell\asymp n^{1/r}\log n$, together with setting $W_0L_0\asymp\left(n^{(1-1/r)}/(\log n)^7\right)^{1/(4\gamma^\star+2)}$, we obtain that there exists a constant $c_{11}>0$ such that

$$\mathbb{P}\left[\|\widehat{f}-f_0\|_2\geq c_{11}\left(\left(\frac{(\log n)^7}{n^{1-1/r}}\right)^{\gamma^\star/(2\gamma^\star+1)}+\sqrt{\frac{u\log n}{n^{1-1/r}}}\right)\right]\lesssim\frac{n^{1/r}\log n}{e^u}+(\log n)^{(1-r)}.$$

□

A.5 Proof of Theorem 5

According to Theorem 2, we obtain

$$\delta_a=\inf_{f\in\mathcal{F}}\left\|f-f_0\right\|_2\lesssim W_02^{-L_0}+W_0^{-\gamma^\star}. \tag{A.12}$$

Recall that

$$\delta_b=\sqrt{\frac{2NsL\ell\log(WL^2)\log(2m/(N\ell))}{m}}\lesssim\sqrt{\frac{2NsL_0\ell\log(W_0L_0^2)\log m}{m}}. \tag{A.13}$$

We now analyze how these bounds behave under different $\boldsymbol{\beta}$-mixing conditions.

• **Case 1: Exponentially $\boldsymbol{\beta}$-mixing.** By plugging (A.12) and (A.13) into (3.4), and choosing $\ell \asymp (\log n)^{1/r}$ such that $\boldsymbol{\beta}((\ell-1)N+1) \lesssim 1/n^2$, together with setting $W_0 \asymp (n/(\log n)^{(4+1/r)})^{1/(2\gamma^\star+1)}$, $L_0 \asymp \log n$, and $s \asymp W_0 L_0$, we obtain that there exists a constant $c_{12} > 0$ such that

$$\mathbb{P}\left[\|\widehat{f} - f_0\|_2 \geq c_{12}\left(\left(\frac{(\log n)^{(4+1/r)}}{n}\right)^{\gamma^\star/(2\gamma^\star+1)} + \sqrt{\frac{(\log n)^{1/r} u}{n}}\right)\right] \lesssim \frac{(\log n)^{1/r}}{e^u} + \frac{\log n}{n}.$$

• **Case 2: Algebraically $\boldsymbol{\beta}$-mixing.** By plugging (A.12) and (A.13) into (3.4), and choosing $\ell \asymp n^{1/r}\log n$, together with setting $W_0 \asymp \left(n^{(1-1/r)}/(\log n)^5\right)^{1/(2\gamma^\star+1)}$, $L_0 \asymp \log n$, and $s \asymp W_0 L_0$, we obtain that there exists a constant $c_{13} > 0$ such that

$$\mathbb{P}\left[\|\widehat{f} - f_0\|_2 \geq c_{13}\left(\left(\frac{(\log n)^5}{n^{1-1/r}}\right)^{\gamma^\star/(2\gamma^\star+1)} + \sqrt{\frac{u\log n}{n^{1-1/r}}}\right)\right] \lesssim \frac{n^{1/r}\log n}{e^u} + (\log n)^{(1-r)}.$$

□

B Proofs of Lemmas

B.1 Proof of Lemma 3

This proof is adapted from Schmidt-Hieber [2020]. In preparation for the proof of Lemma 3, we first present two auxiliary lemmas that will be useful in the analysis:

Lemma 8 (Theorem 5 in Schmidt-Hieber [2020])**.** For any function $f_0 \in \mathcal{C}_d^\beta([0,1]^d, K)$, $W_0 \geq \max\{(\beta+1)^d, (K+1)e^d\}$ and $L_0 \geq 1$, the following inequality holds

$$\inf_{\bar{f}\in\mathcal{SFNN}_{d,1}(W,L,s)} \|\bar{f} - f_0\|_\infty \leq c_{24} W_0 2^{-L_0} + c_{25} W_0^{-\frac{\beta}{d}},$$

where $W = c_{26}W_0$, $L = c_{27}L_0$ and $s = c_{28}L_0W_0$. Here, the positive constants c_{24}–c_{28} depend on (β, d, K).

Lemma 9. Let $f_1 \in \mathcal{SFNN}_{d_1,d_2}(W_1, L_1, s_1)$ and $f_2 \in \mathcal{SFNN}_{d_1',d_2'}(W_2, L_2, s_2)$. Their aggregation and composition satisfy the following properties

Aggregation rule: If $L_1 = L_2$ and $d_1 = d_1'$, then $f = (f_1, f_2)^\top \in \mathcal{SFNN}_{d_1,d_2+d_2'}(W_1 + W_2, L_1, s_1 + s_2)$.

Composition rule: If $d_2 = d_1'$, then $f = f_2 \circ \sigma(f_1) \in \mathcal{SFNN}_{d_1,d_2'}(\max\{W_1, W_2\}, L_1 + L_2 + 1, s_1 + s_2)$.

We omit the proof for Lemma 9 since it is straightforward to obtain the result. We refer readers to Section 7 in Schmidt-Hieber [2020] for more details.

Proof of Lemma 3. In this lemma, we consider a fixed input domain within $[0,1]^d$. By the definition of hierarchical interaction model, for any $f_0 \in \mathcal{H}_d^l(\mathcal{P}, K)$, there exists a sequence of functions $\{g_i\}_{i=1}^l$ such that

$$f_0 = g_l \circ \cdots \circ g_1,$$

where $g_1 : [0,1]^{t_1} \to [-K,K]^{t_2}$ and $g_i : [-K,K]^{t_i} \to [-K,K]^{t_{i+1}}$ for $i = 2, \ldots, l$. While this formulation appears slightly different from our original definition of the hierarchical interaction model, they are indeed equivalent. Specifically, for each level $i = 2, \ldots, l$, we have $g_i = (h_{i,1}, \ldots, h_{i,t_{i+1}})^\top$, where each component function $h_{i,j} \in \mathcal{C}_{t_i}^{\beta_i}([-K,K]^{t_i}, K)$ for $j \in [t_{i+1}]$. Similarly, $g_1 = (h_{1,1}, \ldots, h_{1,t_2})^\top$ with $h_{1,j} \in \mathcal{C}_{t_1}^{\beta_1}([0,1]^{t_1}, K)$ for $j \in [t_2]$. Here, $(\beta_i, t_i) \in \mathcal{P}$ for $i \in [l]$, $t_1 \leq d$ and $t_{l+1} = 1$.

To facilitate the application of Lemma 8, the sequence of functions $\{g_i\}_{i=1}^l$ is transformed into another sequence $\{\bar{h}_i\}_{i=1}^l$ as follows

$$\bar{h}_1(\mathbf{x}) = \frac{1}{2K} g_1(\mathbf{x}) + \frac{1}{2}, \quad \forall \mathbf{x} \in [0,1]^{t_1},$$

$$\bar{h}_i(\mathbf{x}) = \frac{1}{2K} g_i(2K\mathbf{x} - K) + \frac{1}{2}, \quad \forall \mathbf{x} \in [0,1]^{t_i},$$
$$\bar{h}_l(\mathbf{x}) = g_l(2K\mathbf{x} - K), \quad \forall \mathbf{x} \in [0,1]^{t_l}.$$

Building upon this transformation, it follows that

$$f_0 = \bar{h}_l \circ \cdots \circ \bar{h}_1,$$

where $\bar{h}_{1,j} \in \mathcal{C}_{t_1}^{\beta_1}\left([0,1]^{t_1}, 1\right)$ for $j \in [t_2]$, $\bar{h}_{i,j} \in \mathcal{C}_{t_i}^{\beta_i}\left([0,1]^{t_i}, (2K)^{\beta_i}\right)$ for $i = 2, \ldots, l-1$ and $j \in [t_{i+1}]$, and $\bar{h}_{l,j} \in \mathcal{C}_{t_l}^{\beta_l}\left([0,1]^{t_l}, K(2K)^{\beta_l}\right)$ for $j \in [t_{l+1}]$.

By Lemma 8, for each $i \in [l]$ and $j \in [t_{i+1}]$, there exist positive numbers c_{24}, c_{25}, c_{26}, c_{27}^i and c_{28} and a function $\bar{f}_{i,j} \in \mathcal{SFNN}(W_i, L_i, s_i)$, such that

$$\|\bar{f}_{i,j} - \bar{h}_{i,j}\|_\infty \leq c_{24} W_0 2^{-L_0} + c_{25} W_0^{-\frac{\beta_i}{t_i}},$$

where $L_i = c_{26} L_0$, $W_i = c_{27}^i W_0$, and $s_i = c_{28} L_0 W_0$. However, $\bar{f}_{i,j}$ may not map into $[0,1]$. For analytical convenience, we further transform it into $\bar{f}'_{i,j}$ defined by

$$\bar{f}'_{i,j}(\mathbf{x}) = \min\left\{\max\left\{\bar{f}_{i,j}(\mathbf{x}), 0\right\}, 1\right\}, \quad \forall \mathbf{x}.$$

This transformation can be implemented by adding two additional layers with four parameters to each $\bar{f}_{i,j}$. Under this transformation, for each $i \in [l-1]$ and $j \in [t_{i+1}]$, there exists a function $\bar{f}'_{i,j} \in \mathcal{SFNN}(W_i + 2, L_i, s_i + 4)$, taking values in $[0,1]$, such that

$$\|\bar{f}'_{i,j} - \bar{h}_{i,j}\|_\infty \leq c_{24} W_0 2^{-L_0} + c_{25} W_0^{-\frac{\beta_i}{t_i}}.$$

We next construct $\bar{f}$, which provides a consistent approximation to f_0 via the aggregation and composition rules in Lemma 9. For $i \in [l-1]$, we aggregate $\bar{f}'_{i,1}, \ldots, \bar{f}'_{i,t_{i+1}}$ into a single function $\bar{f}'_i$ defined by $\bar{f}'_i = \left(\bar{f}'_{i,1}, \ldots, \bar{f}'_{i,t_{i+1}}\right)^\top$, then $\bar{f}'_i \in \mathcal{SFNN}(t_{i+1}(W_i + 2), L_i + 2, t_{i+1}(s_i + 4))$. By composition, we construct $\bar{f} = \bar{f}_l \circ \sigma(\bar{f}'_{l-1}) \circ \cdots \circ \sigma(\bar{f}'_1) = \bar{f}_l \circ \bar{f}'_{l-1} \circ \cdots \circ \bar{f}'_1$. Then $\bar{f} \in \mathcal{SFNN}(W, L, s)$, with $L = 3(l-1) + \sum_{i=1}^l L_i = c_{29} L_0$, $W = \max_i t_{i+1}(W_i + 2) = c_{30} W_0$, and $s = \sum_{i=1}^l t_{i+1}(s_i + 4) = c_{31} L_0 W_0$. The approximation error between f_0 and $\bar{f}$ can then be bounded as follows

$$\begin{aligned}
&\|f_0 - \bar{f}\|_\infty \\
=&\|\bar{h}_l \circ \bar{h}_{l-1} \circ \cdots \circ \bar{h}_1 - \bar{f}_l \circ \bar{f}'_{l-1} \circ \cdots \circ \bar{f}'_1\|_\infty \\
\leq&\|\bar{h}_l \circ \bar{h}_{l-1} \circ \cdots \circ \bar{h}_1 - \bar{h}_l \circ \bar{f}'_{l-1} \circ \cdots \circ \bar{f}'_1\|_\infty + \left\|\bar{h}_l - \bar{f}_l\right\|_\infty \\
\leq&K(2K)^{\beta_l}\|\bar{h}_{l-1} \circ \cdots \circ \bar{h}_1 - \bar{f}'_{l-1} \circ \cdots \circ \bar{f}'_1\|_\infty + \|\bar{h}_l - \bar{f}_l\|_\infty \\
\leq&K(2K)^{\beta_l}\left(\|\bar{h}_{l-1} \circ \cdots \circ \bar{h}_1 - \bar{f}'_{l-1} \circ \cdots \circ \bar{f}'_1\|_\infty + \|\bar{h}_l - \bar{f}_l\|_\infty\right) \\
\leq&K\prod_{j=1}^{l-1}(2K)^{\beta_{j+1}}\left(\sum_{i=1}^{l-1}\|\bar{h}_i - \bar{f}'_i\|_\infty + \|\bar{h}_l - f_l\|_\infty\right).
\end{aligned}$$

where the second inequality follows from the definition of Hölder smoothness.

Since $K\prod_{j=1}^{l-1}(2K)^{\beta_{j+1}}$ is a constant, it follows that

$$\|\bar{f} - f_0\|_\infty^2 \leq c_{32} \max_{i \in [l]}\left(W_0 2^{-L_0} + W_0^{-\frac{\beta_i}{t_i}}\right)^2 \leq c_{32}\left(W_0 2^{-L_0} + W_0^{-\gamma^\star}\right)^2.$$

To ensure that the output lies within $[-K, K]$, we define $\bar{f}'$ by $\bar{f}' = \min\{\|f_0\|_\infty / \|\bar{f}\|_\infty, 1\} \cdot \bar{f}$. It follows that

$$\|\bar{f}' - f_0\|_\infty \leq \|\bar{f}' - \bar{f}\|_\infty + \|\bar{f} - f_0\|_\infty \leq 2\|\bar{f} - f_0\|_\infty.$$

Then, we directly establish the existence of a constant c_{33} such that

$$\left\{\int_{[0,1]^{d_{\mathbf{x}} \times N}} |\bar{f}(\mathbf{x}) - f_0(\mathbf{x})|^2 \mu(\mathrm{d}\mathbf{x})\right\}^{1/2} \leq c_{33}\left(W_0 2^{-L_0} + W_0^{-\gamma^\star}\right).$$

Choosing $c_{17} = c_{33}$, $c_{18} = c_{29}$, $c_{19} = c_{30}$ and $c_{20} = c_{31}$ completes the proof.

□

B.2 Proof of Lemma 4

In this subsection, we first formally define the architectures of modified recurrent neural networks (MRNNs) and their sparse variants (SMRNNs). First, we define a new activation function that operates on specific components instead of all components. Given an index set $I \subseteq \mathbb{N}$, the modified activation function σ_I is defined as

$$\sigma_I(s)_i = \begin{cases} \sigma(s_i) & \text{if } i \in I, \\ s_i & \text{otherwise.} \end{cases}$$

Using this modified activation function, for any $l \in [L]$ we define the modified recurrent operation as

$$\begin{aligned} \widetilde{r}_l^{(t)}(\boldsymbol{V}_{l-1})_i :=& \sigma_I(\widetilde{A}_l \widetilde{r}_l^{(t-1)}(\boldsymbol{V}_{l-1}) + \widetilde{B}_l \mathbf{v}_{l-1}^{(t)} + \widetilde{\mathbf{c}}_l)_i \\ =& \begin{cases} \sigma(\widetilde{A}_l \widetilde{r}_l^{(t-1)}(\boldsymbol{V}_{l-1}) + \widetilde{B}_l \mathbf{v}_{l-1}^{(t)} + \widetilde{\mathbf{c}}_l)_i & \text{if } i \in I, \\ (\widetilde{A}_l \widetilde{r}_l^{(t-1)}(\boldsymbol{V}_{l-1}) + \widetilde{B}_l \mathbf{v}_{l-1}^{(t)} + \widetilde{\mathbf{c}}_l)_i & \text{otherwise.} \end{cases} \end{aligned}$$

Here, $\widetilde{A}_l, \widetilde{B}_l \in \mathbb{R}^{W \times W}$, $\widetilde{\mathbf{c}}_l \in \mathbb{R}^W$, and $\widetilde{r}_l^{(0)} = \mathbf{0} \in \mathbb{R}^W$.

We denote by $\mathcal{MRNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}(W, L, K, s)$ the class of neural network functions defined with the structure of RNNs but composed of modified recurrent layers instead of original recurrent layers. It is clear that $\mathcal{RNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}(W, L, K, s) \subseteq \mathcal{MRNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}(W, L, K, s)$. Similarly, $\mathcal{SMRNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}(W, L, K, s)$ is defined as the class following the structure of SRNNs, also utilizing modified recurrent layers.

The proof builds on Lemma 10 and Lemma 11, which construct two intermediate representations. Specifically, Lemma 10 shows that an SFNN can be equivalently expressed as an SMRNN prediction function, and Lemma 11 further shows that an SMRNN can be represented by an SRNN. Proofs of these lemmas are deferred to Appendix B.5 and Appendix B.6, respectively.

Lemma 10. For any SFNN $\bar{f} \in \mathcal{SFNN}_{d_{\mathbf{x}} \times N,1}(W, L, s)$, there exists an SMRNN prediction function $\widetilde{f} \in \mathcal{SMRNN}_{d_{\mathbf{x}},1}((d_{\mathbf{x}}+1)W, N+L, s+3(d_{\mathbf{x}}+1)WN)$ such that

$$\widetilde{f}(\boldsymbol{X}) = \bar{f}(\boldsymbol{X}), \quad \boldsymbol{X} \in [0,1]^{d_{\mathbf{x}} \times N}.$$

Lemma 11. For any SMRNN prediction function $\widetilde{f} \in \mathcal{SMRNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}(W, L, s)$, there exists an SRNN prediction function $f \in \mathcal{SRNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}(W+1, 2L, s+6LW)$ such that

$$f(\boldsymbol{X}) = \widetilde{f}(\boldsymbol{X}), \quad \boldsymbol{X} \in [0,1]^{d_{\mathbf{x}} \times N}.$$

Proof of Lemma 4. From Lemma 10, we know there exists an SMRNN prediction function $\widetilde{f} \in \mathcal{SMRNN}_{d_{\mathbf{x}},1}((d_{\mathbf{x}}+1)W, N+L, s+3(d_{\mathbf{x}}+1)WN)$ such that

$$\widetilde{f}(\boldsymbol{X}) = \bar{f}(\boldsymbol{X}), \quad \boldsymbol{X} \in [0,1]^{d_{\mathbf{x}} \times N}.$$

Then, applying Lemma 11 to the function $\widetilde{f}$, we obtain an SRNN prediction function $f \in \mathcal{SRNN}_{d_{\mathbf{x}},1}((d_{\mathbf{x}}+1)W+1, 2L+2N, s+3(d_{\mathbf{x}}+1)WN+6(d_{\mathbf{x}}+1)(L+N)W)$, such that

$$f(\boldsymbol{X}) = \widetilde{f}(\boldsymbol{X}), \quad \boldsymbol{X} \in [0,1]^{d_{\mathbf{x}} \times N}.$$

This completes the proof.

□

B.3 Proof of Lemma 6

To establish the lower bound, we utilize the Lipschitz continuity of $p_{\epsilon|X_1,\ldots,X_N}(\cdot)$, which implies that $p_{\epsilon|X_1,\ldots,X_N}(u) \geq \underline{p}/2$ when $|u| \leq \underline{p}/(2l_0)$. By applying Lemma S6 from the supplementary materials of Padilla et al. [2022], we derive the following result:

$$\mathcal{R}_\tau(f) - \mathcal{R}_\tau(f_0) \geq \min\left(\frac{\underline{p}}{4}, \frac{\underline{p}^2}{16 l_0}\right) \mathbb{E} \min\left[|f(X_1 \ldots, X_N) - f_0(X_1, \ldots, X_N)|,\right.$$

$$(f(X_1,\ldots,X_N)-f_0(X_1,\ldots,X_N))^2\big]. \tag{B.1}$$

Furthermore, as both f and f_0 are bounded by K, the squared difference satisfies the inequality $(f(X_1,\ldots,X_N)-f_0(X_1,\ldots,X_N))^2 \le 2K|f(X_1,\ldots,X_N)-f_0(X_1,\ldots,X_N)|$. By incorporating this result into (B.1) and assuming $K \ge 1$, we arrive at the desired lower bound for the excess quantile risk.

For the upper bound, we have the following decomposition:

$$\rho_\tau(u-v)-\rho_\tau(u) = -v\{\tau-\mathbb{1}(u\le 0)\}+\int_0^v \{\mathbb{1}(u\le t)-\mathbb{1}(u\le 0)\}\mathrm{d}t.$$

Recall that $R_\tau(f)=\mathbb{E}[\rho_\tau(Y_N-f(X_1,\ldots,X_N))]$. Choosing $u=\epsilon$ and $v=f(X_1,\ldots,X_N)-f_0(X_1,\ldots,X_N)$, we have

$$\begin{aligned}\mathcal{R}_\tau(f)-\mathcal{R}_\tau(f_0) =& \mathbb{E}[\rho_\tau(Y_N-f(X_1,\ldots,X_N))-\rho_\tau(\epsilon)]\\ =& \mathbb{E}\bigg[\int_0^{f(X_1,\ldots,X_N)-f_0(X_1,\ldots,X_N)}\int_0^t p_{\epsilon|X_1,\ldots,X_N}(s)\mathrm{d}s\mathrm{d}t\bigg]\\ \le& \frac{\bar{p}}{2}\|f-f_0\|_2^2,\end{aligned}$$

where the last inequality follows from Assumption 1.

□

B.4 Proof of Lemma 7

To prove this lemma, we first introduce the definition of the uniform covering number.

Definition 5 (Uniform covering number)**.** Let $d\in\mathbb{N}$ and $\mathcal{F}=\{f\,|\,\mathcal{X}\to\mathbb{R}\}$ be a class of real-valued functions on $\mathcal{X}$. For $\epsilon>0$, the uniform covering number of $\mathcal{F}$ under the supremum norm is defined as

$$\boldsymbol{N}_d(\epsilon,\|\cdot\|_\infty,\mathcal{F}) = \sup_{(\mathbf{x}_1,\ldots,\mathbf{x}_d)\in\mathcal{X}^d} \boldsymbol{N}(\epsilon,\|\cdot\|_\infty,\mathcal{F}|_{\mathbf{x}_1,\ldots,\mathbf{x}_d}),$$

where $\mathcal{F}|_{\mathbf{x}_1,\ldots,\mathbf{x}_d}=\{(f(\mathbf{x}_1),\ldots,f(\mathbf{x}_d))^\top\,|\,f\in\mathcal{F}\}\subseteq\mathbb{R}^d$ and $\boldsymbol{N}(\epsilon,\|\cdot\|_\infty,\mathcal{W})$ is the ϵ-covering number of a subset $\mathcal{W}\subseteq\mathbb{R}^d$ under the supremum norm $\|\cdot\|_\infty$.

Given the above definition, we next present two lemmas characterizing the covering numbers of RNNs and SFNNs.

Lemma 12 (RNN covering number bound)**.** Let $\mathcal{F}_N=\mathcal{RNN}_{d_{\mathbf{x}},1}(W,L,K)$. Then we have

$$\log \boldsymbol{N}_n\big(\epsilon,\|\cdot\|_\infty,\mathcal{F}_N\big)\lesssim W^2L^2\log\max\{W,L\}\log\left(\frac{Kn}{\epsilon}\right).$$

Lemma 13 (SFNN covering number bound)**.** Let $\mathcal{F}_N=\mathcal{SFNN}_{d_{\mathbf{x}},1}(W,L,K,s)$. Then we have

$$\log \boldsymbol{N}_n\big(\epsilon,\|\cdot\|_\infty,\mathcal{F}_N\big)\lesssim sL\log(WL^2)\log\left(\frac{Kn}{\epsilon}\right).$$

The following lemma shows that any SRNN can be represented by an SFNN, which will be used to characterize the covering number of SRNNs.

Lemma 14. For any SRNN prediction function $f\in\mathcal{SRNN}_{d_{\mathbf{x}},d_{\mathbf{y}}}(W,L,s)$, there exists an SFNN $\bar{f}\in\mathcal{SFNN}_{d_{\mathbf{x}}\times N,d_{\mathbf{y}}}\big((2N-1)W,(N+1)L+2,2Ns+2N^2WL\big)$ such that

$$f(\boldsymbol{X})=\bar{f}(\boldsymbol{X}),\quad \boldsymbol{X}\in[0,1]^{d_{\mathbf{x}}\times N}.$$

Finally, we provide two technical lemmas that serve as essential components in the proof.

Lemma 15 (Theorem 7.3 in Bousquet [2003])**.** Let $\mathbf{z}_1,\ldots,\mathbf{z}_n$ be i.i.d. random vectors drawn from the distribution of Z, and $\mathcal{F}$ be a measurable class of functions such that $\mathbb{E}f(Z)=0$ for any $f\in\mathcal{F}$. Assume $\sup_{f\in\mathcal{F}}\|f\|_\infty\le A$ and let σ be a positive constant such that $\sigma^2\ge\sup_{f\in\mathcal{F}}\mathbb{E}f^2(Z)$. Then, for any $x>0$,

$$\mathbb{P}\left\{\sup_{f\in\mathcal{F}}\left|\frac{1}{n}\sum_{i=1}^n f(\mathbf{z}_i)\right|\ge 2\mathbb{E}\left[\sup_{f\in\mathcal{F}}\left|\frac{1}{n}\sum_{i=1}^n f(\mathbf{z}_i)\right|\right]+\sigma\sqrt{\frac{2x}{n}}+\frac{4Ax}{3n}\right\}\le e^{-x}. \tag{B.2}$$

Lemma 16 (Corollary 5.1 in Chernozhukov et al. [2014]). Denote $S = ([0,1]^d \times \mathbb{R})^N$ and let $\mathbf{z}_1, \ldots, \mathbf{z}_n \in S$ be i.i.d. random vectors drawn from the distribution of $Z \in S$. Let $\mathcal{F}$ be a measurable class of functions $S \to \mathbb{R}$, to which a measurable envelope F is attached. Assume that $\|F\|_2 < \infty$, and let $\sigma > 0$ be any positive constant such that $\sup_{f \in \mathcal{F}} \mathbb{E} f^2(Z) \leq \sigma^2 \leq \|F\|_2^2$. Furthermore, we assume that there exist constants $A \geq e$ and $\nu \geq 1$ such that $\sup_Q \boldsymbol{N}(\epsilon \|F\|_{Q_2}, \|\cdot\|_{Q_2}, \mathcal{F}) \leq (A/\epsilon)^\nu$ for any $0 < \epsilon \leq 1$, where the supremum is taken over all n-discrete probability measures Q on S and $\boldsymbol{N}(\epsilon, \|\cdot\|_{Q_2}, \mathcal{F})$ is the ϵ-covering number of $\mathcal{F}$ under the $L_2(Q)$ norm. Then,

$$\mathbb{E}\left[\sup_{f \in \mathcal{F}} \frac{1}{\sqrt{n}} \left| \sum_{i=1}^n (f(\mathbf{z}_i) - \mathbb{E} f(\mathbf{z}_i)) \right| \right] \lesssim \sigma \sqrt{v \log\left(\frac{A\|F\|_2}{\sigma}\right)} + \frac{\nu \|\bar{F}\|_2}{\sqrt{n}} \log\left(\frac{A\|F\|_2}{\sigma}\right),$$

where $\bar{F} = \max_{1 \leq i \leq n} F(\mathbf{z}_i)$.

Proof of Lemma 7. Recall that $\mathcal{F}(\delta) = \{f \in \mathcal{F} \,|\, \|f - f_0\|_2 \leq \delta\}$.

Since $\rho_\tau(\cdot)$ is a Lipschitz function, we have

$$\sup_{f \in \mathcal{F}(\delta)} |g(f, \mathbf{z}_i)| \leq \sup_{f \in \mathcal{F}(\delta)} |f(\mathbf{x}_{i-N+1}, \ldots, \mathbf{x}_i) - f_0(\mathbf{x}_{i-N+1}, \ldots, \mathbf{x}_i)| \leq 2K.$$

Therefore, $\sup_{f \in \mathcal{F}(\delta)} |g(f, \mathbf{z}_i) - \mathbb{E}_Z g(f, Z)| \leq 4K =: A$. Moreover,

$$\sup_{f \in \mathcal{F}(\delta)} \mathbb{E}\big[|g(f, \mathbf{z}_i)|^2\big] \leq \sup_{f \in \mathcal{F}(\delta)} \mathbb{E}\big[|f(\mathbf{x}_{i-N+1}, \ldots, \mathbf{x}_i) - f_0(\mathbf{x}_{i-N+1}, \ldots, \mathbf{x}_i)|^2\big] \leq \delta^2,$$

which further implies

$$\sup_{f \in \mathcal{F}(\delta)} \mathbb{E}[|g(f, \mathbf{z}_i) - \mathbb{E}_Z g(f, Z)|^2] \leq \sup_{f \in \mathcal{F}(\delta)} \mathbb{E}[|g(f, \mathbf{z}_i)|^2] \leq \delta^2 =: \sigma^2.$$

Denoting $E(\delta) = \mathbb{E} \sup_{f \in \mathcal{F}(\delta)} \left| n^{-1} \sum_{i=1}^n g(f, \mathbf{z}_i) - \mathbb{E}_Z g(f, Z) \right|$, Lemma 15 gives

$$\mathbb{P}\left\{ \sup_{f \in \mathcal{F}(\delta)} \left| \frac{1}{n} \sum_{i=1}^n g(f, \mathbf{z}_i) - \mathbb{E}_Z g(f, Z) \right| \geq 2E(\delta) + \sigma \sqrt{\frac{2x}{n}} + \frac{4Ax}{3n} \right\} \leq e^{-x} \tag{B.3}$$

for any $x \geq 0$.

Now, we establish an upper bound of the expectation $E(\delta)$. We denote $\mathcal{M}_n(\delta) = \{g(f, \mathbf{z}_i) \,|\, f \in \mathcal{F}(\delta)\}$.

• **Case 1:** $\mathcal{F} = \mathcal{RNN}_{d_{\mathbf{x}},1}(W, L, K)$ **.**

Combining the Lipschitz continuity of $\rho_\tau(\cdot)$ and Lemma 12 gives that for any $\epsilon \in (0, K)$,

$$\log \boldsymbol{N}_n\left(\epsilon, \|\cdot\|_\infty, \mathcal{M}_n(\delta)\right) \leq \log \boldsymbol{N}_n\left(\epsilon, \|\cdot\|_\infty, \mathcal{F}\right) \lesssim W^2 L^2 \log \max\{W, L\} \log\left(\frac{Kn}{\epsilon}\right).$$

Also, the Lipschitz property of $\rho_\tau(\cdot)$ implies that $F = 2K$ is an envelope function of $\mathcal{M}_n(\delta)$. Thus, for any discrete probability measure Q supported on n points, we have

$$\log \boldsymbol{N}_n(\epsilon \|F\|_2, \|\cdot\|_2, \mathcal{M}_n(\delta)) \lesssim W^2 L^2 \log \max\{W, L\} \log\left(\frac{n}{2\epsilon}\right),$$

where the L_2 norm is taken with respect to Q. Applying Lemma 16, we have

$$\begin{aligned} E(\delta) &\lesssim \sigma \sqrt{\frac{W^2 L^2 \log \max\{W, L\}}{n} \log\left(\frac{nK}{\sigma}\right)} + 2K \frac{W^2 L^2 \log \max\{W, L\}}{n} \log\left(\frac{nK}{\sigma}\right) \\ &\lesssim \delta \delta_b + \delta_b^2 \end{aligned}$$

for any $\delta \geq 1/n$. Thus, when $\delta \geq \delta_b$, we have $E(\delta) \lesssim \delta \delta_b$. By combining this and (B.3), there exists a universal positive constant $c_{34} > 0$ such that

$$\mathbb{P}\left[\sup_{f \in \mathcal{F}(\delta)} \left| \frac{1}{n} \sum_{i=1}^n g(f, \mathbf{z}_i) - \mathbb{E}_Z[g(f, Z)] \right| \geq c_{34} \cdot \delta \left(\delta_b + \sqrt{\frac{x}{n}} \right) \right] \leq e^{-x}.$$

holds for any $0 \le x \le n\delta^2$ and $\delta \ge \delta_b$.

• **Case 2:** $\mathcal{F} = \mathcal{SRNN}_{d_{\mathbf{x}},1}(W, L, K, s)$**.**

Combining the Lipschitz continuity of $\rho_\tau(\cdot)$, Lemma 13, and Lemma 14 yields that for any $\epsilon \in (0, K)$ and $s \asymp WL$,

$$\log \boldsymbol{N}_n\left(\epsilon, \|\cdot\|_\infty, \mathcal{M}_n(\delta)\right) \le \log \boldsymbol{N}_n\left(\epsilon, \|\cdot\|_\infty, \mathcal{F}\right) \lesssim sL\log(WL^2)\log\left(\frac{Kn}{\epsilon}\right).$$

Also, the Lipschitz property of $\rho_\tau(\cdot)$ implies that $F = 2K$ is an envelope function of $\mathcal{M}_n(\delta)$. Thus, for any discrete probability measure Q supported on n points, we have

$$\log \boldsymbol{N}_n(\epsilon\|f\|_2, \|\cdot\|_2, \mathcal{M}_n(\delta)) \lesssim sL\log(WL^2)\log\left(\frac{n}{2\epsilon}\right),$$

where the L_2 norm is taken with respect to Q. Applying Lemma 16, we have

$$\begin{aligned} E(\delta) &\lesssim \sigma\sqrt{\frac{sL\log(WL^2)}{n}\log\left(\frac{nK}{\sigma}\right)} + 2K\frac{sL\log(WL^2)}{n}\log\left(\frac{nK}{\sigma}\right) \\ &\lesssim \delta\delta_b + \delta_b^2 \end{aligned}$$

for any $\delta \ge 1/n$. Thus, when $\delta \ge \delta_b$, we have $E(\delta) \lesssim \delta\delta_b$. By combining this and (B.3), there exists a universal positive constant $c_{35} > 0$ such that

$$\mathbb{P}\left[\sup_{f\in\mathcal{F}(\delta)}\left|\frac{1}{n}\sum_{i=1}^n g(f, \mathbf{z}_i) - \mathbb{E}_Z[g(f, Z)]\right| \ge c_{35}\cdot\delta\left(\delta_b + \sqrt{\frac{x}{n}}\right)\right] \le e^{-x}.$$

holds for any $0 \le x \le n\delta^2$ and $\delta \ge \delta_b$. Choosing $c_{23} = \max\{c_{34}, c_{35}\}$ completes the proof.

□

B.5 Proof of Lemma 10

For any given SFNN $\bar{f}$, we first construct an SMRNN $\widetilde{r}_{\theta_{N+1}}$ such that $\widetilde{r}^{(N)}_{\theta_{N+1}}$ is equivalent to the first layer of $\bar{f}$. Subsequently, building upon $\widetilde{r}_{\theta_{N+1}}$, we sequentially construct $\widetilde{r}_{\theta_{N+L}}$ by directly leveraging the remaining $L-1$ layers of $\bar{f}$, ensuring that $\widetilde{r}_{\theta_{N+L}}$ is equivalent to $\bar{f}$.

Proof of Lemma 10. Our construction begins with a key result for building the initial SMRNN $\widetilde{r}_{\theta_{N+1}}$.

For any given $A_{k,1}, \ldots, A_{k,N} \in \mathbb{R}^{1\times d_{\mathbf{x}}}$ for $k \in [W]$ and a bias vector $\mathbf{c} \in \mathbb{R}^W$, we construct an SMRNN $r_{\theta_{N+1}} = r_{N+1}\circ r_N \circ \cdots \circ r_1 \circ p : \mathbb{R}^{d_{\mathbf{x}}\times N} \to \mathbb{R}^{(d_{\mathbf{x}}+1)W\times N}$ of width $(d_{\mathbf{x}}+1)W$, depth $N+1$. The layers $\{r_l\}_{l=1}^{N+1}$ and p are defined explicitly as follows for each $t \in [N]$ and $l \in [N]$.

$$r_l^{(t)}(\boldsymbol{V}) = \begin{pmatrix} A & & \\ & \ddots & \\ & & A \end{pmatrix} r_l^{(t-1)}(\boldsymbol{V}) + \begin{pmatrix} B_{1,l} & & \\ & \ddots & \\ & & B_{W,l} \end{pmatrix} \begin{pmatrix} \mathbf{v}^{(t)} \\ \vdots \\ \mathbf{v}^{(t)} \end{pmatrix} \in \mathbb{R}^{(d_{\mathbf{x}}+1)W},$$

$$r_{N+1}^{(t)}(\boldsymbol{V}) = \sigma(D\mathbf{v}^{(t)} + \bar{\mathbf{c}}), \quad p^{(t)}(\boldsymbol{X}) = \begin{pmatrix} I_{d_{\mathbf{x}}} \\ O_{1,d_{\mathbf{x}}} \\ \vdots \\ I_{d_{\mathbf{x}}} \\ O_{1,d_{\mathbf{x}}} \end{pmatrix} \mathbf{x}^{(t)} = \begin{pmatrix} \mathbf{x}^{(t)} \\ 0 \\ \vdots \\ \mathbf{x}^{(t)} \\ 0 \end{pmatrix} \in \mathbb{R}^{(d_{\mathbf{x}}+1)W},$$

where $A = \begin{pmatrix} O_{d_{\mathbf{x}},d_{\mathbf{x}}} & O_{d_{\mathbf{x}},1} \\ O_{1,d_{\mathbf{x}}} & 1 \end{pmatrix}, B_{k,l} = \begin{pmatrix} I_{d_{\mathbf{x}}} & O_{d_{\mathbf{x}},1} \\ \mathbf{b}_{k,l} & 1 \end{pmatrix}$, $\mathbf{b}_{k,l} = \sum_{i=1}^N \lambda_{i,l}A_{k,i} \in \mathbb{R}^{1\times d_{\mathbf{x}}}$,

$\left[\lambda_{i,j}\right]_{1\le i,j\le N} = \left[\binom{2N-i-j}{N-i}\right]^{-1}_{1\le i,j\le N}$, $\bar{\mathbf{c}} = \begin{pmatrix} \mathbf{c} \\ O_{d_{\mathbf{x}}W,1} \end{pmatrix}$ and $D_{i,j} = \begin{cases} 1 & i \in [W], j = (d_{\mathbf{x}}+1)i \\ 0 & \text{otherwise} \end{cases}$.

Combining Lemma 6 in Song et al. [2023] and Step 2 of the proof of Lemma 8 in Jiao et al. [2024] gives that $r_{\theta_{N+1}}$ constructed above satisfies the following property:

$$r^{(N)}_{\theta_{N+1}}(\boldsymbol{X}) = \begin{pmatrix} \sigma\Big(\sum_{j=1}^N A_{1,j}\mathbf{x}^{(j)} + c_1\Big) \\ \sigma\Big(\sum_{j=1}^N A_{2,j}\mathbf{x}^{(j)} + c_2\Big) \\ \vdots \\ \sigma\Big(\sum_{j=1}^N A_{W,j}\mathbf{x}^{(j)} + c_W\Big) \\ O_{d_{\mathbf{x}}W,1} \end{pmatrix}. \tag{B.4}$$

In this construction, we have $\mathcal{T}(p) = Wd_{\mathbf{x}}$, $\mathcal{T}(r_{N+1}) \le W + d_{\mathbf{x}}$, and for $l \in [N]$:

$$\begin{aligned} \mathcal{T}(r_l) &= W(1 + d_{\mathbf{x}} + 1 + \|\mathbf{b}_l\|_0) \\ &\le 2W(d_{\mathbf{x}} + 1). \end{aligned}$$

We bound the sparsity of $r_{\theta_{N+1}}$ by

$$\mathcal{T}(r_{\theta_{N+1}}) \le \sum_{l=1}^{N+1} \mathcal{T}(r_l) + \mathcal{T}(p) \le 2WN(d_{\mathbf{x}} + 1) + Wd_{\mathbf{x}} + W + d_{\mathbf{x}}. \tag{B.5}$$

By applying this result to a given SFNN $\bar{f}$, parameterized as $(\mathrm{vec}(\widetilde{B}_1)^\top, \widetilde{\mathbf{c}}_1^\top, \ldots, \mathrm{vec}(\widetilde{B}_L)^\top, \widetilde{\mathbf{c}}_L^\top)^\top$, we proceed to define the corresponding parameters required for initializing the SMRNN $\widetilde{r}_{\theta_{N+1}}$. Here, $\widetilde{B}_1 \in \mathbb{R}^{W\times(d_{\mathbf{x}}N)}$, $\widetilde{B}_l \in \mathbb{R}^{W\times W}$ for $l = 2, \ldots, L-1$, $\widetilde{B}_L \in \mathbb{R}^{1\times W}$, $\widetilde{\mathbf{c}}_l \in \mathbb{R}^W$ for $l \in [L-1]$, and $\widetilde{\mathbf{c}}_L \in \mathbb{R}^{d_{\mathbf{y}}}$. Specifically, we set $\mathbf{c} = \widetilde{\mathbf{c}}_1$, and for each $i \in [W]$ and $j \in [N]$, we define $A_{i,j} = (\widetilde{B}_1)_{i,\,(j-1)d_{\mathbf{x}}+1:jd_{\mathbf{x}}}$. The notation $(\widetilde{B}_1)_{i,\,(j-1)d_{\mathbf{x}}+1:jd_{\mathbf{x}}}$ refers to the sub-vector of the i-th row of $\widetilde{B}_1$ corresponding to entries from column $(j-1)d_{\mathbf{x}}+1$ to $jd_{\mathbf{x}}$. By applying these definitions to the construction described in (B.4), we obtain $\widetilde{r}_{\theta_{N+1}}$. It then follows from (B.4) that

$$\widetilde{r}^{(N)}_{\theta_{N+1}}(\boldsymbol{X}) = \begin{pmatrix} \sigma\Big(\widetilde{B}_1\,\mathrm{vec}(\boldsymbol{X}) + \widetilde{\mathbf{c}}_1\Big) \\ O_{d_{\mathbf{x}}W,1} \end{pmatrix}.$$

We next construct the remaining $L-1$ layers using the 2-nd to L-th layers of the SFNN $\bar{f}$. Define

$$\widetilde{r}^{(t)}_{N+l}(\boldsymbol{V}) = \begin{cases} \sigma\big(\bar{B}_l\mathbf{v}^{(t)} + \bar{\mathbf{c}}_l\big) & l = 2, \ldots, L-1, \\ \bar{B}_l\mathbf{v}^{(t)} + \bar{\mathbf{c}}_l & l = L, \end{cases} \qquad q(\boldsymbol{V}) = (Q\mathbf{v}^{(1)}, \ldots, Q\mathbf{v}^{(N)}),$$

where

$$\begin{aligned} \bar{B}_l &= \begin{pmatrix} \widetilde{B}_l & O_{W,d_{\mathbf{x}}W} \\ O_{d_{\mathbf{x}}W,W} & O_{d_{\mathbf{x}}W,d_{\mathbf{x}}W} \end{pmatrix} && \text{for } l = 2, \ldots, L-1, \\ \bar{B}_L &= \begin{pmatrix} \widetilde{B}_L & O_{d_{\mathbf{y}},d_{\mathbf{x}}W} \\ O_{(d_{\mathbf{x}}+1)W-d_{\mathbf{y}},W} & O_{(d_{\mathbf{x}}+1)W-d_{\mathbf{x}}W} \end{pmatrix}, \\ \bar{\mathbf{c}}_l &= \begin{pmatrix} \widetilde{\mathbf{c}}_l \\ O_{d_{\mathbf{x}}W,1} \end{pmatrix} && \text{for } l = 2, \ldots, L-1, \\ \bar{\mathbf{c}}_L &= \begin{pmatrix} \widetilde{\mathbf{c}}_L \\ O_{(d_{\mathbf{x}}+1)W-d_{\mathbf{y}},1} \end{pmatrix}, \\ Q &= \begin{pmatrix} I_{d_{\mathbf{y}}} & O_{d_{\mathbf{y}},(d_{\mathbf{x}}+1)W-d_{\mathbf{y}}} \end{pmatrix}. \end{aligned}$$

Then SMRNN $\widetilde{r}_{\theta_{N+L}} = q \circ \widetilde{r}_{N+L} \circ \cdots \circ \widetilde{r}_{N+2} \circ \widetilde{r}_{\theta_{N+1}}$ satisfies

$$\widetilde{r}^{(N)}_{\theta_{N+L}}(\boldsymbol{X}) = \bar{f}(\boldsymbol{X}).$$

Finally, we compute the sparsity of $\widetilde{r}_{\theta_{N+L}}$,

$$\mathcal{T}(\widetilde{r}_{\theta_{N+L}}) = \mathcal{T}(\widetilde{r}_{\theta_{N+1}}) + \sum_{l=2}^{L} \mathcal{T}(\widetilde{r}_{N+l}) + \mathcal{T}(q)$$

$$\leq 2WN(d_{\mathbf{x}}+1)+2Wd_{\mathbf{x}}+\sum_{l=2}^{L}\left(\|\widetilde{B}_l\|_0+\|\widetilde{\mathbf{c}}_l\|_0\right)+1$$
$$\leq 3WN(d_{\mathbf{x}}+1)+\mathcal{T}(\bar{f}),$$

where the first inequality follows from (B.5) and $d_{\mathbf{y}}=1$.

□

B.6 Proof of Lemma 11

Proof of Lemma 11. This proof is adapted from the proof of Lemma 3 in Song et al. [2023]. We first show that any modified recurrent layer can be represented by the composition of a sequence of recurrent layers and linear maps. Specifically, for any modified recurrent layer $\widetilde{r}:\mathbb{R}^{d\times N}\to\mathbb{R}^{d\times N}$ satisfying $\widetilde{r}^{(t)}(\boldsymbol{X})=\sigma_I(\widetilde{A}\widetilde{r}^{(t-1)}(\boldsymbol{X})+\widetilde{B}\mathbf{x}^{(t)}+\widetilde{\mathbf{c}})$, there exist linear maps $p:\mathbb{R}^{d\times N}\to\mathbb{R}^{(d+1)\times N}$ and $q:\mathbb{R}^{(d+1)\times N}\to\mathbb{R}^{d\times N}$, along with recurrent layers $r_1,r_2:\mathbb{R}^{(d+1)\times N}\to\mathbb{R}^{(d+1)\times N}$, such that

$$\widetilde{r}(\boldsymbol{X})=q\circ r_2\circ r_1\circ p(\boldsymbol{X}),\ \text{for}\ \boldsymbol{X}\in\mathcal{X}^N. \tag{B.6}$$

where $\mathcal{X}\subseteq\mathbb{R}^d$ is a compact set.

We present the construction of r_1,r_2,p,q directly as follows. For any $t\in[N]$, we choose

$$p^{(t)}(X)=\begin{pmatrix}I_d\\O_{1,d}\end{pmatrix}\mathbf{x}^{(t)}=\begin{pmatrix}\mathbf{x}^{(t)}\\0\end{pmatrix},$$
$$r_1^{(t)}(\boldsymbol{V})=\sigma\left(\begin{pmatrix}\widetilde{B}&\\&0\end{pmatrix}\mathbf{v}^{(t)}+\begin{pmatrix}\widetilde{\mathbf{c}}\\0\end{pmatrix}+z_0\mathbf{1}_{d+1}\right),$$
$$r_2^{(t)}(\boldsymbol{V})=\sigma\left(Ar_2^{(t-1)}(\boldsymbol{V})+\mathbf{v}^{(t)}+\begin{pmatrix}-z_0\mathbf{1}_k\\\mathbf{0}_{d+1-k}\end{pmatrix}\right),$$
$$q^{(t)}(\boldsymbol{V})=\begin{pmatrix}I_d&O_{d,1}\end{pmatrix}\begin{pmatrix}I_k&&\\&I_{d-k}&-\mathbf{1}_{d-k}\\&&0\end{pmatrix}\mathbf{v}^{(t)},$$

where $\boldsymbol{V}\in\mathbb{R}^{(d+1)\times N}$, $\mathbf{v}^{(t)}$ is the t-th column of $\boldsymbol{V}$, $A=\begin{pmatrix}\widetilde{A}&\\&0\end{pmatrix}\begin{pmatrix}I_k&&\\&I_{d-k}&-\mathbf{1}_{d-k}\\&&0\end{pmatrix}$, and $z_0=\max_{t\in[N],i\in[d]}\sup_{\boldsymbol{X}\in\mathcal{X}^N}\left|(\widetilde{A}\widetilde{r}^{(t-1)}(\boldsymbol{X})+\widetilde{B}\mathbf{x}^{(t)}+\widetilde{\mathbf{c}})_i\right|$.

By the choice of z_0, for any $t\in[N]$, we have

$$\begin{aligned}&(r_1\circ p)^{(t)}(\boldsymbol{X})\\=&\sigma\left(\begin{pmatrix}\widetilde{B}&\\&0\end{pmatrix}\begin{pmatrix}\mathbf{x}^{(t)}\\0\end{pmatrix}+\begin{pmatrix}\widetilde{\mathbf{c}}\\0\end{pmatrix}+z_0\mathbf{1}_{d+1}\right)\\=&\begin{pmatrix}z_0\mathbf{1}_d+(\widetilde{B}\mathbf{x}^{(t)}+\widetilde{\mathbf{c}})\\z_0\end{pmatrix}.\end{aligned}$$

We next prove that our construction satisfies the following property by induction

$$(r_2\circ r_1\circ p)^{(t)}(\boldsymbol{X})=\begin{pmatrix}\widetilde{r}^{(t)}(\boldsymbol{X})_{1:k}\\z_0\mathbf{1}_{d-k}+\widetilde{r}^{(t)}(\boldsymbol{X})_{k+1:d}\\z_0\end{pmatrix},\ \text{for}\ t\in[N]. \tag{B.7}$$

For $t=1$, we have

$$\begin{aligned}&(r_2\circ r_1\circ p)^{(1)}(\boldsymbol{X})\\=&\sigma\left((r_1\circ p)^{(1)}(\boldsymbol{X})+\begin{pmatrix}-z_0\mathbf{1}_k\\\mathbf{0}_{d+1-k}\end{pmatrix}\right)\\=&\sigma\left(\begin{pmatrix}(\widetilde{B}\mathbf{x}^{(1)}+\widetilde{\mathbf{c}})+z_0\mathbf{1}_d\\z_0\end{pmatrix}+\begin{pmatrix}-z_0\mathbf{1}_k\\\mathbf{0}_{d+1-k}\end{pmatrix}\right)\end{aligned}$$

$$= \begin{pmatrix} \sigma(\widetilde{B}\mathbf{x}^{(1)} + \widetilde{\mathbf{c}})_{1:k} \\ z_0 \mathbf{1}_{d-k} + (\widetilde{B}\mathbf{x}^{(1)} + \widetilde{\mathbf{c}})_{k+1:d} \\ z_0 \end{pmatrix}$$
$$= \begin{pmatrix} \widetilde{r}^{(1)}(\boldsymbol{X})_{1:k} \\ z_0 \mathbf{1}_{d-k} + \widetilde{r}^{(1)}(\boldsymbol{X})_{k+1:d} \\ z_0 \end{pmatrix},$$

where the second-to-last equality follows from the choice of z_0.

Assuming the induction hypothesis holds for $t-1$, we have

$$(r_1 \circ p)^{(t)}(\boldsymbol{X}) + \begin{pmatrix} -z_0 \mathbf{1}_k \\ \mathbf{0}_{d+1-k} \end{pmatrix} = \begin{pmatrix} (\widetilde{B}\mathbf{x}^{(t)} + \widetilde{\mathbf{c}})_{1:k} \\ z_0 \mathbf{1}_{d-k} + (\widetilde{B}\mathbf{x}^{(t)} + \widetilde{\mathbf{c}})_{k+1:d} \\ z_0 \end{pmatrix}.$$

With the above align, we further obtain

$$\begin{aligned}
&(r_2 \circ r_1 \circ p)^{(t)}(\boldsymbol{X}) \\
=&\sigma\left(A \begin{pmatrix} \widetilde{r}^{(t-1)}(\boldsymbol{X})_{1:k} \\ z_0 \mathbf{1}_{d-k} + \widetilde{r}^{(t-1)}(\boldsymbol{X})_{k+1:d} \\ z_0 \end{pmatrix} + \begin{pmatrix} (\widetilde{B}\mathbf{x}^{(t)} + \widetilde{\mathbf{c}})_{1:k} \\ z_0 \mathbf{1}_{d-k} + (\widetilde{B}\mathbf{x}^{(t)} + \widetilde{\mathbf{c}})_{k+1:d} \\ z_0 \end{pmatrix} \right) \\
=&\sigma\left(\begin{pmatrix} (\widetilde{A}\widetilde{r}^{(t-1)}(\boldsymbol{X}))_{1:k} \\ (\widetilde{A}\widetilde{r}^{(t-1)}(\boldsymbol{X}))_{k+1:d} \\ 0 \end{pmatrix} + \begin{pmatrix} (\widetilde{B}\mathbf{x}^{(t)} + \widetilde{\mathbf{c}})_{1:k} \\ z_0 \mathbf{1}_{d-k} + (\widetilde{B}\mathbf{x}^{(t)} + \widetilde{\mathbf{c}})_{k+1:d} \\ z_0 \end{pmatrix} \right) \\
=&\begin{pmatrix} \widetilde{r}^{(t)}(\boldsymbol{X})_{1:k} \\ z_0 \mathbf{1}_{d-k} + \widetilde{r}^{(t)}(\boldsymbol{X})_{k+1:d} \\ z_0 \end{pmatrix},
\end{aligned}$$

where the last equality follows from the choice of z_0.

Thus, we have demonstrated that our construction satisfies (B.7).

By the definition of q, for any $t \in [N]$ we have

$$\begin{aligned}
&(q \circ r_2 \circ r_1 \circ p)^{(t)}(\boldsymbol{X}) \\
=&\begin{pmatrix} I_d & O_{d,1} \end{pmatrix} \begin{pmatrix} I_k & & \\ & I_{d-k} & -\mathbf{1}_{d-k} \\ & & 0 \end{pmatrix} \begin{pmatrix} \widetilde{r}^{(t)}(\boldsymbol{X})_{1:k} \\ z_0 \mathbf{1}_{d-k} + \widetilde{r}^{(t)}(\boldsymbol{X})_{k+1:d} \\ z_0 \end{pmatrix} \\
=&\widetilde{r}^{(t)}(\boldsymbol{X}).
\end{aligned}$$

Thus, our construction satisfies (B.6).

Applying (B.6) to any SMRNN $\widetilde{r}_{\widetilde{\theta}} = \widetilde{q} \circ \widetilde{r}_L \circ \cdots \circ \widetilde{r}_1 \circ \widetilde{p}$ with depth L and width W and setting $d = W$, we know that there exist $2L$ recurrent layers $r_1, \ldots, r_{2L}$ and $2L$ linear maps $p_1, \ldots, p_L, q_1, \ldots, q_L$ such that for any $\boldsymbol{X} \in [0,1]^{d_{\mathbf{x}} \times N}$

$$\begin{aligned}
\widetilde{r}_{\widetilde{\theta}}(\boldsymbol{X}) =&\widetilde{q} \circ \widetilde{r}_L \circ \cdots \circ \widetilde{r}_1 \circ \widetilde{p}(\boldsymbol{X}) \\
=&\widetilde{q} \circ (q_L r_{2L} r_{2L-1} p_L) \circ \cdots \circ (q_1 r_2 r_1 p_1) \circ \widetilde{p}(\boldsymbol{X}) \\
=&\big(\widetilde{q} q_L\big) \circ r_{2L} \circ (r_{2L-1} p_L q_{L-1}) \circ \cdots \circ (r_3 p_2 q_1) \circ r_2 \circ r_1 \circ \big(p_1 \widetilde{p}\big)(\boldsymbol{X}).
\end{aligned}$$

By setting $q = \widetilde{q} q_L$, $p = p_1 \widetilde{p}$, $\bar{r}_{2l+1} = r_{2l+1} p_{l+1} q_l$ for $l \in [L-1]$, and $r_\theta = q \circ r_{2L} \circ \bar{r}_{2L-1} \circ r_{2L-2} \circ \cdots \circ r_4 \circ \bar{r}_3 \circ r_2 \circ r_1 \circ p$, we obtain

$$r_\theta(\boldsymbol{X}) = \widetilde{r}_{\widetilde{\theta}}(\boldsymbol{X}), \text{ for } \boldsymbol{X} \in [0,1]^{d_{\mathbf{x}} \times N}.$$

It is clear that r_θ has depth $2L$ and width $W+1$.

Finally, we analyze the sparsity, for $l \in [L-1]$,

$$\mathcal{T}(\bar{r}_{2l+1}) = \mathcal{T}(r_{2l+1} p_{l+1} q_l) = \mathcal{T}(r_{2l+1}) = \|\widetilde{B}_l\|_0 + \|\widetilde{\mathbf{c}}_l + z_0 \mathbf{1}_W\|_0 + 1, \tag{B.8}$$

$$\mathcal{T}(r_{2l}) = \|A_l\|_0 + (W + 1 + k), \tag{B.9}$$

$$\|A_l\|_0 \leq \|\widetilde{A}_l\|_0 + W. \tag{B.10}$$

Combining (B.8), (B.9), and (B.10), we obtain

$$\begin{aligned}
\mathcal{T}(r_\theta) =& \mathcal{T}(p_1\widetilde{p}) + \mathcal{T}(\widetilde{q}q_1) + \sum_{l=1}^{L} \mathcal{T}(r_{2l}) + \mathcal{T}(r_1) + \sum_{l=1}^{L-1} \mathcal{T}(\bar{r}_{2l+1}) \\
=& \mathcal{T}(p_1\widetilde{p}) + \mathcal{T}(\widetilde{q}q_1) + \sum_{l=1}^{2L} \mathcal{T}(r_l) \\
\leq& \mathcal{T}(p_1\widetilde{p}) + \mathcal{T}(\widetilde{q}q_1) + \sum_{l=1}^{L} (\|\widetilde{A}_l\|_0 + \|\widetilde{B}_l\|_0 + \|\widetilde{\mathbf{c}}_l\|_0) + 4LW + L \\
=& \mathcal{T}(\widetilde{p}) + \mathcal{T}(\widetilde{q}) + W + \sum_{l=1}^{L} \mathcal{T}(\widetilde{r}_l) + 4LW + L \\
\leq& \mathcal{T}(\widetilde{r}_{\widetilde{\theta}}) + 6LW,
\end{aligned}$$

where the first inequality follows from $k + 1 \leq W$ and $\|\widetilde{\mathbf{c}}_l + z_0\mathbf{1}_W\|_0 \leq \|\widetilde{\mathbf{c}}_l\|_0 + W$.

□

B.7 Proof of Lemma 12

Our covering number bound for RNNs is consistent with Lemma 12 in Jiao et al. [2024], which itself is based on Theorem 7 in Bartlett et al. [2019]. For completeness, we provide a full proof of the covering number bound for RNNs.

Proof of Lemma 12. By Lemma 9 in Jiao et al. [2024], we have

$$\mathcal{F}_N = \mathcal{RNN}_{d_{\mathbf{x}},1}(W, L, K) \subseteq \mathcal{FNN}_{d_{\mathbf{x}}\times N,1}\big((2N-1)W, (N+1)L, K\big).$$

Then it follows that

$$\log \boldsymbol{N}_n\big(\epsilon, \|\cdot\|_\infty, \mathcal{F}_N\big) \leq \log \boldsymbol{N}_n\Big(\epsilon, \|\cdot\|_\infty, \mathcal{FNN}_{d_{\mathbf{x}}\times N,1}\big((2N-1)W, (N+1)L, K\big)\Big). \tag{B.11}$$

By Theorem 7 in Bartlett et al. [2019], for any $W, L, K \geq 0$ we have

$$\log \boldsymbol{N}_n\Big(\epsilon, \|\cdot\|_\infty, \mathcal{FNN}_{d_{\mathbf{x}}\times N,1}\big(W, L, K\big)\Big) \lesssim W^2L^2 \log\max\{W, L\} \log\left(\frac{Kn}{\epsilon}\right). \tag{B.12}$$

Combining (B.11) with (B.12) yields the desired bound and completes the proof.

□

B.8 Proof of Lemma 13

We first present the following lemma and definitions of Vapnik-Chervonenkis dimension and Pseudo-dimension of a real-valued function class .

Lemma 17 (Lemma 17 in Bartlett et al. [2019])**.** Suppose $W \leq M$ and let $P_1, \ldots, P_M$ be polynomials of degree at most D in W variables. Then we have

$$\big|\big\{(\operatorname{sgn}(P_1(a)), \ldots, \operatorname{sgn}(P_M(a))) \,|\, a \in \mathbb{R}^W\big\}\big| \leq 2(2eMD/W)^W.$$

Definition 6 (VC-dimension)**.** Let $\mathcal{F} : \mathbb{R}^{d_{\mathbf{x}}} \to \mathbb{R}$ be a class of real-valued functions. The VC-dimension of $\mathcal{F}$, denoted as $\mathrm{VCdim}(\mathcal{F})$, is the largest integer $m \in \mathbb{N}$ for which there exist points $(\mathbf{x}_1, \ldots, \mathbf{x}_m) \in \mathbb{R}^{d_{\mathbf{x}}\times m}$ such that

$$\big|\big\{\big(\operatorname{sgn}(f(\mathbf{x}_1)), \ldots, \operatorname{sgn}(f(\mathbf{x}_m))\big) \,|\, f \in \mathcal{F}\big\}\big| = 2^m.$$

Here, $\operatorname{sgn}(\cdot)$ is the sign function defined as

$$\operatorname{sgn}(x) = \begin{cases} 1, & \text{if } x > 0, \\ 0, & \text{if } x \leq 0. \end{cases}$$

Furthermore, the quantity

$$\Pi_{\mathcal{F}}(m) = \left| \left\{ \left(\operatorname{sgn}(f(\mathbf{x}_1)), \ldots, \operatorname{sgn}(f(\mathbf{x}_m)) \right) \mid f \in \mathcal{F} \right\} \right|$$

is referred to as the growth function of the function class $\mathcal{F}$.

Definition 7 (Pseudo-dimension)**.** Given a real-valued function class $\mathcal{F} : \mathbb{R}^{d_{\mathbf{x}}} \to \mathbb{R}$, the pseudo-dimension, denoted as Pdim($\mathcal{F}$) is the largest $m \in \mathbb{N}$ for which there exist $(\mathbf{x}_1, \ldots, \mathbf{x}_m) \in \mathbb{R}^{d_{\mathbf{x}} \times m}$ and $(y_1, \ldots, y_m) \in \mathbb{R}^m$ such that for any $(b_1, \ldots, b_m) \in \{0, 1\}^m$, there exists $f \in \mathcal{F}$ such that for any $i \in [m]$:

$$f(\mathbf{x}_i) \geq y_i \iff b_i = 1.$$

Proof of Lemma 13. We prove this bound in three steps. First, we reformulate the problem as an inequality depending only on the VC-dimension, using the relationships between the pseudo-dimension, the VC-dimension, and the covering number. Second, we adapt the proof of Theorem 7 in Bartlett et al. [2019] to establish a bound for the growth function of the prediction function class derived from a sparse neural network with a fixed zero-parameter configuration. Finally, we account for all possible sparse structures to derive the overall VC-dimension bound for the prediction function class.

• Step 1: From Covering Number to VC-dimension.

By Theorem 12.2 in Anthony and Bartlett [2009], for any $\epsilon \in (0, K)$, we have

$$\boldsymbol{N}_n\left(\epsilon, \|\cdot\|_\infty, \mathcal{F}_N\right) \leq \sum_{i=1}^{\text{Pdim}(\mathcal{F}_N)} \binom{n}{i} \left(\frac{K}{\epsilon}\right)^i.$$

If $n < \text{Pdim}(\mathcal{F}_N)$, then

$$\boldsymbol{N}_n\left(\epsilon, \|\cdot\|_\infty, \mathcal{F}_N\right) \leq \sum_{i=1}^{n} \binom{n}{i} \left(\frac{K}{\epsilon}\right)^i = \left(1 + \frac{K}{\epsilon}\right)^n \leq \left(\frac{enK}{\epsilon}\right)^{\text{Pdim}(\mathcal{F}_N)}.$$

On the other hand, if $n \geq \text{Pdim}(\mathcal{F}_N)$, then

$$\boldsymbol{N}_n\left(\epsilon, \|\cdot\|_\infty, \mathcal{F}_N\right) \leq \left(\frac{enK}{\epsilon \text{Pdim}(\mathcal{F}_N)}\right)^{\text{Pdim}(\mathcal{F}_N)} \leq \left(\frac{enK}{\epsilon}\right)^{\text{Pdim}(\mathcal{F}_N)}.$$

Thus, we have

$$\log \boldsymbol{N}_n\left(\epsilon, \|\cdot\|_\infty, \mathcal{F}_N\right) \leq \text{Pdim}(\mathcal{F}_N) \log\left(\frac{enK}{\epsilon}\right). \tag{B.13}$$

By Theorem 14.1 in Anthony and Bartlett [2009], there exists a function class $\mathcal{F}_{N+1}$ generated from the SFNN with width W, length $L + 1$ and sparsity $s + 1$ such that

$$\text{Pdim}(\mathcal{F}_N) \leq \text{VCdim}(\mathcal{F}_{N+1}). \tag{B.14}$$

In the remainder of the proof, we restrict our analysis to the case of N for simplicity. The results obtained can be directly extended to the case of $N + 1$.

• Step 2: Cardinality Bound for SFNNs with fixed sparsity structure.

First, we formally define the function class generated from the SFNN with a fixed sparsity structure as follows:

$$\mathcal{F}_N^{\xi} = \{ f_\theta \in \mathcal{F}_N \mid \theta_i = 0 \text{ if } \xi_i = 0, \ \forall i \in [M] \}.$$

Here, $M = \sum_{l=1}^{L-1} W_l + \sum_{l=0}^{L-1} W_l W_{l+1}$ is the total number of parameters in the SFNN and $\xi \in \{0,1\}^M$ is a binary vector satisfying $\|\xi\|_0 = s$. Each entry of ξ corresponds to an entry of θ in the SFNN: if $\xi_i = 1$, then θ_i is treated as a free parameter; otherwise, if $\xi_i = 0$, θ_i is fixed to zero. Thus, $\mathcal{F}_N^\xi$ represents the subclass of $\mathcal{F}_N$ constrained by the sparsity pattern specified by ξ.

In this step, we consider a function class $\mathcal{F}_N^\xi$ with a fixed parameter ξ. Since $\mathcal{F}_N^\xi$ is a parameterized function class whose parameter vector has $M - s$ zero entries, any function in $\mathcal{F}_N^\xi$ can be represented as $g(\mathbf{x}, \theta')$, where $\theta' \in \mathbb{R}^s$. Given $m \geq s$ and $\mathbf{x}_1, \ldots, \mathbf{x}_m$, our goal is to derive an upper bound for the cardinality of the following set:

$$\left|\left\{\left(\operatorname{sgn}(g(\mathbf{x}_1, \theta')), \ldots, \operatorname{sgn}(g(\mathbf{x}_m, \theta'))\right) \mid \theta' \in \mathbb{R}^s\right\}\right|.$$

Recall that the ReLU activation function $\sigma(x) = \max\{x, 0\}$ is a piecewise-defined function. Consequently, $g(\mathbf{x}, \theta')$ is also a piecewise-defined function with respect to θ'. To apply Lemma 17, we partition $\mathbb{R}^s$ into disjoint regions $S = \{I_1, \ldots, I_D\}$, where $I_i \cap I_j = \emptyset$ for $i \neq j$ and $i, j \in [D]$. Within each region I_i, for all $j \in [m]$, the function $g(\mathbf{x}_j, \theta')$ reduces to a polynomial function. We denote the number of free parameters in the first l layers by s_ξ^l. We construct this partition recursively with the goal for S_l to satisfy the following properties for any $l \in [L]$:

1. If $l = 1$, then $|S_{l-1}| = 1$; otherwise, $|S_{l-1}| \leq |S_{l-2}| \cdot 2\left(2eW_l ml / s_\xi^{l-1}\right)^{s_\xi^{l-1}}$.
2. For any $I \in S_{l-1}$, $w \in [W_l]$, and $j \in [m]$, the input to the w-th neuron at level l is a polynomial function $f_{I,w,j}$ of degree at most l, which depends on s_ξ^l variables.

We first select $S_0 = \mathbb{R}^s$. The input to any neuron at the first layer is an affine function of degree at most 1, depending on at most s_ξ^1 variables. Consequently, S_0 satisfies both required properties.

Given $S_0, \ldots, S_{n-1}$ that satisfy the two properties, we proceed to construct S_n. For any region $I \in S_{n-1}$. Since $f_{I,w,j}$ is a polynomial function of degree at most n for each $w \in [W_n]$ and $j \in [m]$, we derive the following inequality by applying Lemma 17.

$$|\{\operatorname{sgn}(f_{I,w,j}(\theta')) \mid w \in [W_n], j \in [m], \theta' \in I\}| \leq 2(2eW_n mn / s_\xi^n)^{s_\xi^n}.$$

This result implies that we can partition I into at most $2(2eW_l mL/(s_\xi^n))^{s_\xi^n}$ disjoint regions. Within each region, all these polynomials maintain the same sign. Consequently, for any $j \in [m]$ the output of any neuron at the n-th level remains a polynomial of degree at most n.

We define S_n to be the set of all disjoint regions generated from all $I \in S_{n-1}$. Since each layer connecting the n-th and $(n+1)$-th layers can increase the degree by at most one, for any $j \in [m]$, the input to any neuron at the $(n+1)$-th layer is a polynomial of degree at most $n+1$, depending on at most s_ξ^{n+1} variables. Thus, S_n satisfies both of the required properties. By induction, we have

$$|S_{L-1}| \leq \prod_{l=1}^{L-1} 2(2eW_l ml s_\xi^l)^{s_\xi^l} \leq \prod_{l=1}^{L-1} 2(2eW' ml s_\xi^l)^{s_\xi^l},$$

where $W' = \max\{W, d_{\mathbf{y}}\}$. As implied by Lemma 17, the cardinality of any partition in S_{L-1} is bounded by $2\left(2eW'ml/s_\xi^L\right)^{s_\xi^L}$. By AM-GM inequality, we obtain

$$\begin{aligned}
&|\{\operatorname{sgn}(g(\mathbf{x}_1, \theta')), \ldots, \operatorname{sgn}(g(\mathbf{x}_m, \theta')) \mid \theta' \in \mathbb{R}^{s_\xi}\}| \\
\leq &\prod_{l=1}^{L} 2(2eW'ml/s_\xi^l)^{s_\xi^l} \\
\leq &2^L \left(\frac{2eW'm\sum_{l=1}^L l}{\sum_{l=1}^L s_\xi^l}\right)^{\sum_{l=1}^L s_\xi^l}.
\end{aligned} \tag{B.15}$$

• Step 3: Obtain bound on VCdim$(\mathcal{F}_N)$.

In this step, we consider all possible fixed sparsity structures ξ, of which there are at most $\binom{M}{s} \leq M^s$. Summing (B.15) over all such structures yields, for any $(\mathbf{x}_1, \ldots, \mathbf{x}_m) \in \mathbb{R}^{d_{\mathbf{x}} \times m}$,

$$\begin{aligned}
&|\{\operatorname{sgn}(f(\mathbf{x}_1)), \ldots, \operatorname{sgn}(f(\mathbf{x}_m)) \,|\, f \in \mathcal{F}_N\}| \\
&\leq \sum_{\|\xi\|_0 = s} 2^L \left(\frac{2eW'm \sum_{l=1}^L l}{\sum_{l=1}^L s_\xi^l} \right)^{\sum_{l=1}^L s_\xi^l} \\
&\leq M^s 2^L \max_{\|\xi\|_0 = s} \left(\frac{2eW'm \sum_{l=1}^L l}{\sum_{l=1}^L s_\xi^l} \right)^{\sum_{l=1}^L s_\xi^l}. \qquad \text{(B.16)}
\end{aligned}$$

Let $\xi_{\max}$ be the sparsity structure that attains the maximum in (B.16):

$$\xi_{\max} = \arg\max_{\|\xi\|_0 = s} \left(\frac{2eW'm \sum_{l=1}^L l}{\sum_{l=1}^L s_\xi^l} \right)^{\sum_{l=1}^L s_\xi^l}.$$

The upper bound in (B.16) can thus be expressed as $M^s 2^L \left(\frac{2eW'm \sum_{l=1}^L l}{\sum_{l=1}^L s_{\xi_{\max}}^l} \right)^{\sum_{l=1}^L s_{\xi_{\max}}^l}$.

Choosing m as $\operatorname{VCdim}(\mathcal{F}_N)$, we have

$$2^{\operatorname{VCdim}(\mathcal{F}_N)} \leq M^s 2^L \left(\frac{2eW' \operatorname{VCdim}(\mathcal{F}_N) \sum_{l=1}^L l}{\sum_{l=1}^L s_{\xi_{\max}}^l} \right)^{\sum_{l=1}^L s_{\xi_{\max}}^l}.$$

By Lemma 18 in Bartlett et al. [2019], we obtain

$$\begin{aligned}
\operatorname{VCdim}(\mathcal{F}_N) &\leq L + s \log_2 M + (\sum_{l=1}^L s_{\xi_{\max}}^l) \log_2(2eW'L^2 \log_2(eWL^2)) \\
&\lesssim L + s \log_2(W^2 L) + Ls \log_2(2eWL^2 \log_2(eWL^2)) \\
&\lesssim sL \log(WL^2). \qquad \text{(B.17)}
\end{aligned}$$

Finally, to ensure the completeness of this proof, we need to show $\operatorname{VCdim}(\mathcal{F}_N) \geq s$. This result follows directly from Theorem 3 in Bartlett et al. [2019].

Combining (B.13), (B.14), and (B.17), we obtain the desired bound.

□

B.9 Proof of Lemma 14

This proof is adapted from the proof of Lemma 9 in Jiao et al. [2024]. As a first step, we demonstrate that for any recurrent layer $r : \mathbb{R}^{W \times N} \to \mathbb{R}^{W \times N}$ satisfying $r^{(t)}(\boldsymbol{X}) = \sigma(Ar^{(t-1)}(\boldsymbol{X}) + B\mathbf{x}^{(t)} + \mathbf{c})$, there exists an SFNN $\bar{f}$ with width $(2N-1)W$ and length $N+1$, such that

$$\bar{f}(\boldsymbol{X}) = \begin{pmatrix} r^{(1)}(\boldsymbol{X}) \\ r^{(2)}(\boldsymbol{X}) \\ \vdots \\ r^{(N)}(\boldsymbol{X}) \end{pmatrix}. \qquad \text{(B.18)}$$

Here, $\boldsymbol{X} \in \mathbb{R}^{W \times N}$. We construct $\bar{f}$ as $\bar{f} = f_{N+1} \circ f_N \circ \cdots \circ f_1$ with the parameters defined as follows

$$\bar{A}_1 = \begin{pmatrix} B & O & \cdots & O \\ O & I & \cdots & O \\ O & -I & \cdots & O \\ \vdots & \vdots & \ddots & \vdots \\ O & O & \cdots & I \\ O & O & \cdots & -I \end{pmatrix}, \quad \bar{\mathbf{b}}_1 = \begin{pmatrix} \mathbf{c} \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

$$\bar{A}_j = \begin{pmatrix} I & & & & & & & & \\ & \ddots & & & & & & & \\ & & I & & & & & & \\ & & & I & O & O & & & \\ & & & -I & O & O & & & \\ & & & A & B & -B & & & \\ & & & & & & I & & \\ & & & & & & & \ddots & \\ & & & & & & & & I \end{pmatrix}, \quad \bar{\mathbf{b}}_j = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \mathbf{c} \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad j = 2, \ldots, N,$$

$$\bar{A}_{N+1} = \begin{pmatrix} I & -I & & & & & \\ & & I & -I & & & \\ & & & & \ddots & & \\ & & & & & I & -I \\ & & & & & & I \end{pmatrix}, \quad \bar{\mathbf{b}}_{N+1} = \mathbf{0}.$$

In these block matrices, each block is of size $W \times W$. $\bar{A}_1$ has $2N-1$ rows and N columns of blocks. For $j = 2, \ldots, N$, $\bar{A}_j$ has $2j-4$ identity matrices in the upper left corner and $2N-2j$ identity matrices in the lower right corner, and $\bar{\mathbf{b}}_j$ has $(2j-2)W$ zeros above $\mathbf{c}$. $\bar{A}_{N+1}$ has N rows and $2N-1$ columns of blocks.

We verify that this construction satisfies (B.18) through direct calculation.

$$\begin{aligned} \bar{f}(\boldsymbol{X}) &= f_{N+1} \circ f_N \circ \cdots \circ f_1 \begin{pmatrix} \mathbf{x}^{(1)} \\ \mathbf{x}^{(2)} \\ \vdots \\ \mathbf{x}^{(N)} \end{pmatrix} \\ &= f_{N+1} \circ f_N \circ \cdots \circ f_{i+1} \begin{pmatrix} \sigma(r^{(1)}(\boldsymbol{X})) \\ \sigma(-r^{(1)}(\boldsymbol{X})) \\ \vdots \\ \sigma(r^{(i-1)}(\boldsymbol{X})) \\ \sigma(-r^{(i-1)}(\boldsymbol{X})) \\ r^{(i)}(\boldsymbol{X}) \\ \sigma(\mathbf{x}^{(i+1)}) \\ \sigma(-\mathbf{x}^{(i+1)}) \\ \vdots \\ \sigma(\mathbf{x}^{(N)}) \\ \sigma(-\mathbf{x}^{(N)}) \end{pmatrix} \\ &= f_{N+1} \begin{pmatrix} \sigma(r^{(1)}(\boldsymbol{X})) \\ \sigma(-r^{(1)}(\boldsymbol{X})) \\ \vdots \\ \sigma(r^{(N-1)}(\boldsymbol{X})) \\ \sigma(-r^{(N-1)}(\boldsymbol{X})) \\ r^{(N)}(\boldsymbol{X}) \end{pmatrix} \\ &= \begin{pmatrix} r^{(1)}(\boldsymbol{X}) \\ r^{(2)}(\boldsymbol{X}) \\ \vdots \\ r^{(N)}(\boldsymbol{X}) \end{pmatrix}. \end{aligned}$$

This derivation relies on the property $\sigma(\mathbf{x}) - \sigma(-\mathbf{x}) = \mathbf{x}$, which is applied sequentially across the last three equalities. Thus, our construction for $\bar{f}$ satisfies (B.18).

Next, we extend this construction to a complete SRNN $r_\theta = q \circ r_L \circ \cdots \circ r_1 \circ p$ with width W and depth L. For an input sequence $\boldsymbol{X} \in [0,1]^{d_{\mathbf{x}} \times N}$, we can find SFNNs $\bar{f}_1, \cdots, \bar{f}_L$ with width $(2N-1)W$ and length $N+1$, as well as layers $\bar{p}$ and $\bar{q}$, such that

$$\begin{aligned}
\bar{q} \circ \bar{f}_L \circ \cdots \circ \bar{f}_2 \circ \bar{f}_1 \circ \bar{p}(\boldsymbol{X}) &= \bar{q} \circ \bar{f}_L \circ \cdots \circ \bar{f}_2 \circ \bar{f}_1 \begin{pmatrix} p^{(1)}(\boldsymbol{X}) \\ p^{(2)}(\boldsymbol{X}) \\ \vdots \\ p^{(N)}(\boldsymbol{X}) \end{pmatrix} \\
&= \bar{q} \circ \bar{f}_L \circ \cdots \circ \bar{f}_2 \begin{pmatrix} (r_1 \circ p)^{(1)}(\boldsymbol{X}) \\ (r_1 \circ p)^{(2)}(\boldsymbol{X}) \\ \vdots \\ (r_1 \circ p)^{(N)}(\boldsymbol{X}) \end{pmatrix} \\
&= \bar{q} \circ \begin{pmatrix} (r_L \cdots r_1 \circ p)^{(1)}(\boldsymbol{X}) \\ (r_L \cdots r_1 \circ p)^{(2)}(\boldsymbol{X}) \\ \vdots \\ (r_L \cdots r_1 \circ p)^{(N)}(\boldsymbol{X}) \end{pmatrix} \\
&= (q \circ r_L \circ \cdots r_1 \circ p)^{(N)}(\boldsymbol{X}).
\end{aligned}$$

Here, the layer $\bar{p}$ is constructed to map $\boldsymbol{X}$ to the stacked outputs of p applied to each input $\mathbf{x}^{(t)}$, and the final layer $\bar{q}$ applies q to the N-th segment of its input vector. Choosing $\bar{f} = \bar{q} \circ \bar{f}_L \circ \cdots \circ \bar{f}_2 \circ \bar{f}_1 \circ \bar{p}$ completes the construction.

Finally, we analyze the sparsity

$$\begin{aligned}
\mathcal{T}(\bar{f}) &\leq \sum_{i=1}^{L} 2N\mathcal{T}(r_i) + 2N^2WL + N\mathcal{T}(p) + N\mathcal{T}(q) \\
&\leq 2N\mathcal{T}(r_\theta) + 2N^2WL.
\end{aligned}$$

In conclusion, we obtain

$$\bar{f} \in \mathcal{SFNN}_{d_{\mathbf{x}} \times N, d_{\mathbf{y}}}\big((2N-1)W, (N+1)L + 2, 2Ns + 2N^2WL\big).$$

□

C GDP Growth Analysis

Analyzing GDP growth is a cornerstone of economic research, providing crucial insights into an economy's overall health and trajectory. Given its capacity to effectively capture and forecast upside and downside economic risks, QR has become an indispensable tool in GDP growth analysis [Adrian et al., 2019].

To further evaluate the practical advantages of RNN-based QR estimators under potentially nonstationary settings, we conduct an empirical analysis using real GDP data[4]. We compare the out-of-sample prediction performance of RNN-, SRNN-, FNN-, and QRF-based estimators for one-quarter-ahead GDP forecasting. For all models, the input sequence length is fixed at $N = 4$, corresponding to four consecutive quarterly GDP observations $\mathbf{x}_{t-3}$ to $\mathbf{x}_t$, representing one year of lagged values. The target variable $\mathbf{y}_t$ is the GDP level at time $t+1$. All models were implemented according to the specifications detailed in Section 4.

The dataset is partitioned chronologically, with the most recent 30% reserved for testing. For all NN-based estimators, the preceding 70% is used for training and validation. During training, 20% of this subset is held out for validation, and early stopping is applied based on the validation check loss.

[4]We utilize U.S. GDP growth data from April 1947 to December 2024, accessible at `https://fred.stlouisfed.org/series/A191RL1Q225SBEA`.

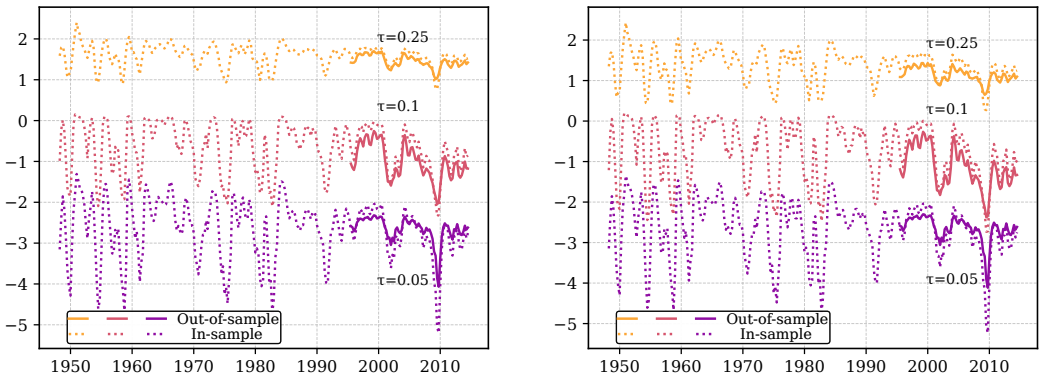
All models are trained by minimizing the empirical check loss and evaluated on the test set using the same metric.

Table 2 presents the mean empirical check loss for each model at quantile levels $\tau = 0.1$, 0.25, 0.5, 0.75, and 0.9. The results indicate that the RNN- and SRNN-based estimators consistently outperform both the FNN and QRF methods across most quantile levels. Furthermore, the SRNN achieves predictive accuracy comparable to that of the standard RNN while inducing sparsity. This suggests that the sparse architecture does not compromise performance.

Model	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
QRF	0.849	1.225	1.410	1.246	0.911
FNN	0.867	1.180	1.773	2.505	2.657
RNN	**0.835**	**1.113**	1.349	**1.154**	0.904
SRNN	0.837	1.700	**1.211**	1.200	**0.898**

Table 2: Out-of-sample prediction errors at different quantiles for GDP growth analysis.

To assess the robustness of RNN-based estimators in QR in the nonstationary environment, we extend our analysis to include in-sample estimators, enabling a direct comparison of in-sample and out-of-sample performance for both RNN and SRNN methods. Adrian et al. [2019] showed that GDP growth volatility is primarily driven by lower quantiles, while upper quantiles remain relatively stable over time. Motivated by this, we focus our comparison on quantile levels $\tau = 0.05$, 0.1, and 0.25, where tail behavior is most prominent. For the in-sample predictions, the full dataset is used for training and validation. The comparative results, shown in Figure 3, reveal a striking alignment between in-sample and out-of-sample quantile estimates. In these figures, the x-axis represents time, while the y-axis denotes the GDP growth rate. This consistency is particularly noteworthy, given that major financial crises (e.g., 2007–2009), which represent substantial tail events, are absent from the data used for out-of-sample estimation. These findings highlight the reliability and generalizability of RNN-based QR methods, even under adverse and previously unseen market conditions.



(a) RNN

(b) SRNN

Figure 3: In-sample and out-of-sample comparison for RNN and SRNN models.

D Hyperparameter Sensitivity Analysis

In this section, we conduct additional experiments varying network depth L, width W, and pruning ratio on Model 1 in Section 4 under the $t_{2.25}$ distribution to assess hyperparameter sensitivity.

We first vary $L \in \{2, 3, 4, 5, 6, 7\}$ and $W \in \{16, 32, 64, 128, 256, 512, 1024\}$, evaluating each (L, W) configuration under the same data and training budget. As shown in Table 3, performance generally improves as both the network depth L and width W increase. When either L or W is small, the model tends to underfit, leading to worse MSE. In contrast, when both are sufficiently large, the model achieves its best performance, suggesting that adequate depth and width are essential for capturing the heavy-tailed characteristics of the $t_{2.25}$ distribution.

$L \setminus W$	16	32	64	128	256	512	1024
2	0.056	0.054	0.054	0.053	0.060	0.065	0.097
3	0.055	0.051	0.051	0.050	0.054	0.058	0.073
4	0.055	0.050	0.052	0.050	0.051	0.052	0.070
5	0.050	0.050	0.050	0.052	0.050	0.050	0.057
6	0.060	0.050	0.053	0.050	**0.049**	**0.049**	0.056
7	0.050	0.050	0.050	0.050	**0.049**	**0.049**	0.056

Table 3: SRNN MSE across different L and W.

We next study sparsity while fixing other settings as in Section 4. As reported in Table 4, the MSE exhibits a non-monotonic trend with respect to the pruning ratio: moderate pruning improves performance, suggesting an effective regularization effect, whereas aggressive pruning degrades performance—likely due to the removal of critical parameters. These results indicate an interior optimum that balances compactness and representational capacity.

Pruning ratio	0.2	0.3	0.4	0.5	0.6	0.7	0.8
MSE	0.046	0.044	0.040	0.039	0.038	**0.036**	0.038

Table 4: Effect of pruning ratio on SRNN MSE.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction accurately stated the contributions and scope of the paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Our work focuses on a fixed quantile level. We discuss this limitation of our results and future direction in Section 6.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: All the theoretical results in the paper are accompanied by a full set of assumptions, with complete and correct proofs included in the Appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We clearly disclose all information necessary for reproducing the presented experimental results, thereby ensuring their reproducibility.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Our experiments are easily reproducible, and we provide all necessary information for their re-implementation in the paper. As this paper focuses on theoretical analysis, the code is not provided.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: This paper provides all details essential for understanding the results in Section 4, Section 5, Appendix C, and Appendix D.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: This paper reports appropriate information about the statistical significance.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: Our experiments utilized relatively small datasets with minimal computational requirements that can be easily reproduced on standard personal computers, making detailed specifications of compute resources unnecessary.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics and ensured that our paper conforms to it.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: Due to its theoretical nature, this paper does not discuss societal impacts.

    Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We do not make publicly available any data or models that pose a high risk of misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The datasets used in this study are the DJIA and GDP, both of which are publicly available.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We don't introduce any new assets in this paper.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: Our research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.