FROM GENERALIZATION ANALYSIS TO OPTIMIZATION DESIGNS FOR STATE SPACE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

A State Space Model (SSM) is a foundation model in time series analysis, which has recently been shown as an alternative to transformers in sequence modeling. In this paper, we theoretically study the generalization of SSMs and propose improvements to training algorithms based on the generalization results. Specifically, we give a *data-dependent* generalization bound for SSMs, showing an interplay between the SSM parameters and the temporal dependencies of the training sequences. Leveraging the generalization bound, we (1) set up a scaling rule for model initialization based on the proposed generalization measure, which significantly improves the robustness of SSMs to different temporal patterns in the sequence data; (2) introduce a new regularization method for training SSMs to enhance the generalization performance. Numerical results are conducted to validate our results.

1 INTRODUCTION

Sequence modeling has been a long-standing research topic in many machine learning areas, such as speech recognition (Hinton et al., 2012), time series prediction (Li et al., 2019), and natural language processing (Devlin et al., 2019). Various machine learning models have been successfully applied in sequence modeling to handle different types of sequence data, ranging from the (probabilistic) Hidden Markov model (Baum & Petrie, 1966) to deep learning models, e.g., Recurrent Neural Networks (RNNs), Long Short-Term Memory units (Hochreiter & Schmidhuber, 1997), Gated Recurrent Unit (Chung et al., 2014), and transformers (Vaswani et al., 2017). In this paper, we focus on the state space model (SSM), which has a simple mathematical expression[1]: $h'(t) = Ah(t) + Bx(t), y(t) = Ch(t) + Dx(t)$ where $h(t)$ is the hidden state, $x(t)$ is the input sequence, $y(t)$ is the output sequence and A, B, C, D are trainable parameters. Recent studies have demonstrated the power of SSMs in deep learning. For example, it was shown in Gu et al. (2022a) that by a new parameterization and a carefully chosen initialization, the structured state space sequence (S4) model achieved strong empirical results on image and language tasks. Following the S4 model, more variants of SSMs are proposed, e.g., the diagonal SSM (Gu et al., 2022b; Gupta et al., 2022) the S5 model (Smith et al., 2023), the H3 model (Fu et al., 2023), the GSS model (Mehta et al., 2023), and the Hyena Hierarchy (Poli et al., 2023).

Theoretical analysis and understanding of the approximation and optimization of SSMs are well studied in the literature such as (Li et al., 2021; 2022; Gu et al., 2022a; 2023). Since the SSM can be regarded as a continuous linear RNN model (Li et al., 2022), most generalization analysis of SSMs is based on the generalization theory of RNNs (Zhang et al., 2018; Chen et al., 2019; Tu et al., 2019). However, these previous works did not study the effects of the temporal dependencies in the sequence data on the SSM generalization (See more details on the comparison in Section 4.1). As an attempt to understand the relationship between the temporal dependencies and the generalization performance, this paper aims to provide a generalization bound that connects the memory structure of the model with the temporal structure of the data. We can, in turn, use the proposed bound to guide us in designing new algorithms to improve optimization and generalization. Specifically, we discover two roles for the proposed generalization measure: (1) generalization bound as an *initialization scheme*; (2) generalization bound as a *regularization method*. The common initialization method for the S4 model and its variants follows from the HiPPO framework (Gu et al., 2022a;

[1]To simplify the analysis, we omit the skip connection by letting $D = 0$

2023), which is based on the prerequisite that the training sequence data is stable. To improve the robustness of SSMs to different temporal patterns in the sequence data, we consider to rescale the initialization of SSMs with respect to the generalization measure. This new initialization scheme makes the SSMs more resilient to variations in the temporal patterns of the training data. Except for the initialization setup, our generalization bound can also be served as a regularizer. Regularization methods like weight decay and dropout are widely applied to training SSMs, but the hidden state matrix A is not regularized because its imaginary part controls the oscillating frequencies of the basis function $e^{At}B$ (Gu et al., 2022b). By taking into account the interaction between the SSM structure and the temporal dependencies, we introduce a new regularization method based on our bound, and it can be applied to the hidden state space to improve the generalization performance. When combining the initialization scheme and the regularization method, our method is applicable to various tasks, ranging from image classification to language processing, while only introducing a minimal computational overhead. To summarize, our contributions are as follows:

- We provide a data-dependent generalization bound for SSMs by taking into account the temporal structure. Specifically, the generalization bound correlates with the memory structure of the model and the (auto)covariance process of the data. It indicates that instead of the weight or the data norm, it is the interplay between the memory structure and the temporal structure of the sequence data that influences the generalization.
- Based on the proposed generalization bound, we setup an initialization scaling rule by adjusting the magnitude of the model parameters with respect to the generalization measure at initialization. This scaling rule improves the robustness of SSMs across different temporal patterns of the sequence data.
- Apart from the initialization scheme, we design a new regularizer for the hidden state matrices of SSMs. Unlike weight decay, our regularizer does not penalize the parameter norm but encourages the model to find a minimizer with lower generalization bound to improve the generalization performance.

2 Related Works

Since a SSM is also a continuous linear RNN, there are three lines of research that are related to our work: generalization of RNNs, temporal structure analysis on RNNs, and optimization of SSMs.

Generalization of RNNs Existing works on the generalization of RNNs focus on the generalization error bound analysis. Specifically, in the early two works of Dasgupta & Sontag (1995) and Koiran & Sontag (1998), VC dimension-based generalization bounds were provided to show the learnability of RNNs. In recent studies, Zhang et al. (2018); Chen et al. (2019); Tu et al. (2019) proved norm-based generalization bounds, improving the VC dimension-based bounds by the Rademacher complexity technique (Bartlett & Mendelson, 2002) under the uniform-convergence framework. In the overparameterization settings, it was shown in Allen-Zhu & Li (2019) that RNNs can learn some concept class in polynomial time given that the model size is large enough. These generalization bounds, however, do not take into account the temporal dependencies and their effects on generalization. In this work, we provide a new generalization bound by combining the memory structure of the model and the temporal structure of the data.

Temporal structure analysis on RNNs Sequence data has long-range temporal dependencies across the time domain, which notably set it apart from non-sequence data. Recent studies have studied the effects of such temporal dependencies on the approximation and optimization of RNNs. For example, in the two works of Li et al. (2021; 2022), a "curse of memory" phenomenon was discovered when using linear RNNs to model the temporal input-output relationships. Particularly, when the target relationship between the input and output has a long-term memory, then both approximation and optimization become extremely challenging. In Wang et al. (2023), the "curse of memory" phenomenon on approximation and optimization was extended to non-linear RNNs based on the temporal relationships. In this paper, we conduct a fine-grained analysis on the effects of the temporal structure analysis on the *generalization* of RNNs.

Optimization of SSMs RNN optimization is known for two issues: training stability and computational cost (Bengio et al., 1994; Pascanu et al., 2013). To address these two issues and capture the long dependencies more efficiently in sequence modeling, the S4 model was proposed by in-

troducing new paraemterization, initialization and discretization (Gu et al., 2022a). Recent variants for the S4 model simplified the hidden state matrix by a diagonal matrix to enhance computational efficiency (Gu et al., 2022b; Gupta et al., 2022; Smith et al., 2023; Orvieto et al., 2023). Regularization methods are also applied for SSMs to prevent overfitting, such as dropout, weight decay and the data continuity regularizer (Qu et al., 2023). However, the principled way to regularize and initialize the parameters still remains to be explored. In this study, we design a new regularization and initialization scheme to improve both optimization and generalization.

3 PRELIMINARIES

In this section, we briefly introduce the SSM in Section 3.1 and the motivation for optimization designs based on the generalization analysis in Section 3.2.

3.1 INTRODUCTION TO SSMS

In this paper, we consider the following single-input single-output SSM,

$$h'(t) = Ah(t) + Bx(t), \quad y(t) = Ch(t), \quad t \geq 0 \tag{1}$$

where x is the input from an input space[2] $\mathcal{X} := C_0(\mathbb{R}_{\geq 0}, \mathbb{R})$; $y(t) \in \mathbb{R}$ is the output at time t; $h(t) \in \mathbb{R}^m$ is the hidden state with $h(0) = 0$; $A \in \mathbb{R}^{m\times m}, B \in \mathbb{R}^{m\times 1}, C \in \mathbb{R}^{1\times m}$ are trainable parameters. Then (1) has an explicit solution $y(t) = \int_0^t \rho_\theta(s)x(t-s)ds$, where $\rho_\theta(s) := Ce^{As}B$ with $\theta = (C, A, B)$. The function $\rho_\theta(s)$ captures the memory structure of the model and the temporal input-output relationship (Li et al., 2022). For the remainder of this paper, we assume that the input sequence $x(t)$ follows a Gaussian process $\mathcal{GP}(\mu(t), K(s,t))$ with

$$\mu(t) = \mathbb{E}[x(t)], \quad K(s,t) = \mathbb{E}[(x(s) - \mu(s))(x(t) - \mu(t))]. \tag{2}$$

Discretization For the S4 model and its variants (Gu et al., 2022a;b; Gupta et al., 2022; Gu et al., 2023), (1) is usually discretized by the Zero-Order Hold method, i.e., given a timescale $\Delta \in \mathbb{R}$,

$$h_{k+1} = \bar{A}h_k + \bar{B}x_k, \quad y_k = \bar{C}h_k, \quad k = 0, 1, \ldots$$

where $\bar{A} = e^{\Delta\cdot A}, \bar{B} = (\bar{A} - \mathbb{I}_m)A^{-1}B, \bar{C} = C$. Then, y_k can be written as the convolution between a kernel and the input sequence, i.e., $y_k = \bar{C}\bar{A}^k\bar{B}x_0 + \bar{C}\bar{A}^{k-1}\bar{B}x_1 + \ldots + \bar{C}\bar{B}x_k = [\bar{K} * x]_k$, where $\bar{K} = (\bar{C}\bar{B}, \bar{C}\bar{A}\bar{B}, \ldots, \bar{C}\bar{A}^k\bar{B})$.

3.2 MOTIVATION: A LINEAR REGRESSION MODEL

In this subsection, we use a linear regression model on non-sequential data as an example to illustrate the connection between the generalization analysis and the optimization designs. This example then motivates us to extend the connection to SSMs on sequential data.

Linear regression We consider a simple linear model $y = \theta^\top x$ with input $x \in \mathbb{R}^d$, output $y \in \mathbb{R}$ and parameter $\theta \in \mathbb{R}^d$. Let the training data $\{(x_i, y_i)\}_{i=1}^n$ be i.i.d. sampled from a distribution $\mathcal{D}$ such that $\|x_i\|_2 = r, |y_i| \leq 1 (\forall i \in [1:n])$. Define the empirical risk $\mathcal{L}_n(\theta) := \frac{1}{n}\sum_{i=1}^n (\theta^\top x_i - y_i)^2$ and the population risk $\mathcal{L}_\mathcal{D}(\theta) := \mathbb{E}_{x,y}[(\theta^\top x - y)^2]$. Then given a norm-constrained space $\Theta := \{\theta \in \mathbb{R}^d : \|\theta\|_2 \leq R\}$, with probability at least $1 - \delta$ over $\mathcal{D}$,

$$\sup_{\theta\in\Theta} |\mathcal{L}_n(\theta) - \mathcal{L}_\mathcal{D}(\theta)| \leq (rR + 1)^2 \cdot \mathcal{O}\left(\sqrt{\frac{\log(1/\delta)}{n}}\right). \tag{3}$$

This is a well-known norm-based generalization bound based on the Rademacher theory (Mohri et al., 2012), and we provide a proof in Appendix B for completeness. Notice that the key term r^2R^2 in the generalization bound (3) is also an upper bound for the magnitude of the linear model output, i.e., $\sup_{\theta\in\Theta}(\theta^\top x_i)^2 \leq r^2R^2$. Thus, we connect the model stability with the generalization bound stability, and this connection induces an initialization scheme for the initialization $\theta^{(0)}$ by setting $\|\theta^{(0)}\|_2 \sim \mathcal{O}(1/r)$. In particular, if we normalize each input x_i such that r is also $\mathcal{O}(1)$,

[2] A linear space of continuous functions from $\mathbb{R}_{\geq 0}$ to $\mathbb{R}$ that vanishes at infinity.

then $\|\theta^{(0)}\|_2 \sim \mathcal{O}(1)$. Since $\theta^{(0)} \in \mathbb{R}^d$, one possible initialization scheme is that $\theta^{(0)}$ follows a Uniform distribution $U[-1/\sqrt{d}, 1/\sqrt{d}]$, which corresponds to the Kaiming initialization (up to some constant) (He et al., 2015). When treating the term r^2R^2 as a regularizer to improve the generalization, we get the weight decay method, i.e., the ℓ_2 regularization w.r.t. $\|\theta\|_2^2$. We summarize the above logic chain that connects the generalization analysis with optimization designs in Figure 1. Now for SSMs, we extend the generalization analysis from non-sequential data to sequential data

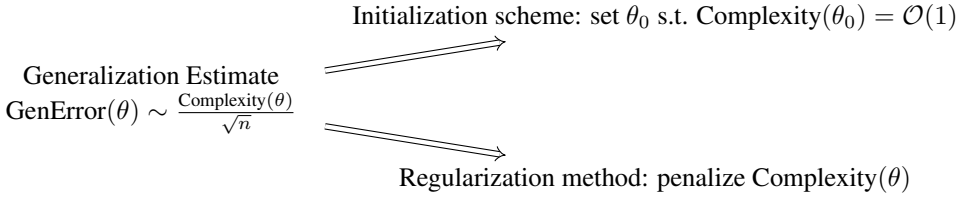

Figure 1: The logic diagram goes from generalization analysis to optimization designs.

by taking into account the temporal structure of the data. This linear regression example motivates us to apply the same logic diagram (Figure 1) to the SSMs, and this is exactly what we are going to present in the following part of this paper.

4 Main results

In this section, we first give a generalization bound for SSMs in Section 4.1, then we design a new initialization scheme in Section 4.2 based on this proposed bound. Apart from the initialization scheme, we introduce a new regularization method in Section 4.3. Finally, we conduct experiments to test the initialization scheme and the regularization method in Section 4.4.

4.1 A generalization bound of SSMs

In this section, we present a generalization bound for the SSM (1) and reveal the effects of the temporal dependencies on the generalization performance. We show that our bound gives a tighter estimate compared with previous norm-based bounds through a toy example. Following the same notation in Section 3.1, we define the empirical risk $R_n(\theta)$ and the population risk $R_x(\theta)$ as

$$R_n(\theta) := \frac{1}{n}\sum_{i=1}^{n}\left|\int_0^T \rho_\theta(T-s)x_i(s)ds - y_i\right|^2, \quad R_x(\theta) := \mathbb{E}_x\left|\int_0^T \rho_\theta(T-s)x(s)ds - y\right|^2$$

where $T > 0$ is some finite terminal time, the training sequence data $\{x_i(t)\}_{i=1}^n$ are independently sampled from a Gaussian process $\mathcal{GP}(\mu(t), K(s,t))$ that satisfies (2), and the label y is generated by some underlying functional $H_T : \mathcal{X} \to \mathbb{R}$, i.e., $y = H_T(x)$. We assume that $|y| \le 1$ for any $x \in \mathcal{X}$, otherwise, we truncate the value of the label to 1. In the next, we make the following assumption on the normalized Gaussian process of $x(t)$:

Assumption 1. *The normalized Gaussian process $\tilde{x}(t) := \frac{x(t)-\mu(t)}{\sqrt{K(t,t)}}$ is almost surely finite and Hölder continuous, i.e., $P(\sup_{t\in[0,T]}|\tilde{x}(t)| < \infty) = 1$, and there exists constants $c_\alpha, H > 0$ such that $\mathbb{E}\left[(\tilde{x}(t) - \tilde{x}(s))^2\right] \le c_\alpha(t-s)^{2\alpha}$ for any $\alpha \in (0, H)$.*

The almost surely boundness assumption covers a large class of Gaussian process, including any stationary Gaussian process with continuous $\mu(t)$ and $K(s,t)$ (Adler et al., 2007). For the stationary Gaussian process with mean 0 and covariance $K(s-t)$, the Hölder continuity assumption is equivalent to $K(0) - K(s-t) \le \tilde{c}_\alpha(t-s)^{2\alpha}$ for any $\alpha \in (0, H)$. Examples for the stationary Gaussian process that satisfy Assumption 1 include: (1) identical sequences: $x(t) = x$ for all $t \in [0,T]$, where $x \sim \mathcal{N}(0,1)$; (2) Gaussian white noise: $\mu(t) = 0$, $K(s,t) = \frac{1}{|b|\sqrt{\pi}}e^{-((s-t)/b)^2}$ for some $b \neq 0$; (3) Ornstein-Uhlenbeck process: $\mu(t) = 0$, $K(s,t) = e^{-|s-t|}$.

We now proceed to bound generalization gap $|R_x(\theta) - R_n(\theta)|$ by establishing uniform convergence of the empirical risk to its corresponding population risk, as stated in following theorem:

Theorem 1. *For a SSM $\int_0^T \rho_\theta(T-s)x(s)ds$, following the notations and settings in Section 3.1 & 4.1, then under Assumption 1, given a parameter space Θ for θ, there exists a constant C_T that depends on T such that for any $\delta \in (0,1)$, with probability at least $1-\delta$ over the training sequences,*

$$\sup_{\theta\in\Theta} |R_x(\theta) - R_n(\theta)|$$
$$\leq \left(\sup_{\theta\in\Theta}\int_0^T |\rho_\theta(T-s)|\sqrt{K(s,s)}ds + \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-s)\mu(s)ds\right| + 1\right)^2 \cdot \widetilde{\mathcal{O}}\left(C_T\sqrt{\frac{\log(n/\delta)}{n}}\right) \tag{4}$$

where $\widetilde{\mathcal{O}}(\cdot)$ hides the logarithmic factor.

The proof is given in Appendix D.4. We see that this generalization bound decreases to zero as the sample size n goes to infinity, provided that the terminal time T is finite and the supremum term in (4) is bounded. Theorem 1 captures the temporal dependencies of the sequence data on the SSM generalization, yielding that the mean and variance at each length position together play important roles in the generalization analysis. Specifically, the generalization gap is small if the convolutions between the SSM function $\rho_\theta(s)$ and the sequence statistics $\mu(s)$, $\sqrt{K(s,s)}$ have small magnitude.

Proof sketch The proof for Theorem 1 is based on Rademacher complexity analysis (Bartlett & Mendelson, 2002). The main difficulty is on bounding the Rademacher complexity of the SSM function $\int_0^T \rho_\theta(T-s)x(s)ds$ for a Gaussian process $x(s)$. We first use the Hölder inequality to give an upper bound for the Rademacher complexity w.r.t. the normalized Gaussian process, then use the Borell-TIS inequality (Lemma 4) to show the finiteness of the normalized Gaussian process. Combining with the Hölder continuity property (Assumption 1), we use an ε-net argument to bound the Rademacher complexity, which then finishes the proof.

Remark 1. *Theorem 1 relies on Assumption 1, which does not apply to all Gaussian processes. In Appendix D.5 (see Theorem 2), we also give another generalization bound that does not require Assumption 1 and can be applied to any Gaussian process, but is looser than the bound (4).*

Comparison Since a SSM is also a continuous linear RNN, we compare (4) with previous bounds for linear RNNs. In Chen et al. (2019), a generalization bound $\widetilde{\mathcal{O}}\left(\|x\|_2\|B\|_2\|C\|_2\|A\|_2/\sqrt{n}\right)$ is provided, where $\|x\|_2$ is the 2-norm of the discrete input sequence. In the continuous case, $\|x\|_2$ corresponds to the L^2 norm w.r.t. a Dirac measure. By changing the matrix 2-norm to matrix 1-norm, Tu et al. (2019) shows another similar generalization bound. These bounds separate the data complexity and the model complexity by the data norm and the model parameter norm individually, and do not account for the temporal dependencies across the time domain. In this work, instead, we incorporate the temporal dependencies via the sequence statistics (mean and variance) to get a generalization bound. Next, we use a toy example to illustrate that our bound gives a tighter estimation. Given a Gaussian process $\{x(t)\}_{t\in[0,T]} \sim \mathcal{GP}(\mu(t), K(s,t))$, we consider the following two upscale transformations (by increasing T to $2T$):

1. left zero padding: $x_1(t) = 0,\ t \in [0,T);\quad x_1(t) = x(t-T),\ t \in [T,2T]$
2. right zero padding: $x_2(t) = x(t),\ t \in [0,T];\quad x_2(t) = 0,\ t \in (T,2T]$

For both cases, we calculate the SSM outputs $y_1(2T)$ and $y_2(2T)$ as

$$y_1(2T) = \int_0^{2T} \rho_\theta(2T-s)x_1(s)ds = \int_0^T \rho_\theta(T-s)x(s)ds = C\int_0^T e^{A(T-s)}Bx(s)ds$$
$$y_2(2T) = \int_0^{2T} \rho_\theta(2T-s)x_2(s)ds = \int_0^T \rho_\theta(2T-s)x(s)ds = Ce^{AT}\int_0^T e^{A(T-s)}Bx(s)ds$$

Then the magnitude of $y_1(2T)$ and $y_2(2T)$ differs with an exponential factor e^{AT}. Since all the eigenvalues of A have negative real part, $y_2(2T) \to 0$ as T increases. Hence, the right zero padding transformation degenerates the SSM function class to a zero function class for large T, inducing a *minimal* generalization gap that only contains the statistical sampling error (see (4) by letting $K(s,s) = \mu(s) = 0$). Therefore, a desired generalization bound should reflect such a difference caused by the different temporal dependencies. However, previous norm-based generalization bounds do not capture such a difference for these two transformations as they produce the same L^2

norm for the input sequence. Let us see what happens for our proposed generalization measure. For the left zero padding, the key term in (4) becomes

$$\int_0^T \left|Ce^{A(T-s)}B\right| \sqrt{K(s,s)}ds + \left|\int_0^T Ce^{A(T-s)}B\mu(s)ds\right| + 1 \tag{5}$$

For the right zero padding, the key term in (4) becomes

$$\int_0^T \left|Ce^{AT}e^{A(T-s)}B\right| \sqrt{K(s,s)}ds + \left|\int_0^T Ce^{AT}e^{A(T-s)}B\mu(s)ds\right| + 1 \tag{6}$$

The detailed derivations are given in Appendix C. By the same argument, our bound (4) indeed captures the difference on the magnitude of the generalization performance for these two sequence transformations. In particular, as $T \to \infty$, (6) reduces to 1, which yields a minimal generalization gap as expected for the zero function class. In that sense, we get a tighter bound for the SSMs.

4.2 Generalization bound as an initialization scheme

In this section, we apply our generalization bound (4) to the practical training. Specifically, we design a scaling rule for the SSM parameters at initialization. This new initialization scheme improves the robustness of SSMs across different temporal patterns of the sequence data.

Previous initialization scaling In the S4 model and its variants, the initialization for the hidden state matrices A, B is based on the HiPPO framework (Gu et al., 2023) to produce orthogonal basis functions. For the matrix C, the initialization is set to be the standard normal distribution $\mathcal{N}(0,1)$ in order to ensure stability and variance-preserving on the input sequences. However, the argument the stability property relies on the prerequisite that the input sequence is constant along the length direction (Gu et al. (2023, Corollary 3.4)), which is restricted as the long-range dependencies may lead to very different temporal patterns on the input sequence, thus it is necessary to adjust the scaling of C to enhance the robustness of the SSMs on different temporal dependencies.

Following the logic diagram in Figure 1 and the linear regression example in Section 3.2, we first extract the dominant term in the generalization bound (4) as

$$\tau(\theta) := \left(\int_0^T |\rho_\theta(T-s)| \sqrt{K(s,s)}ds + \left|\int_0^T \rho_\theta(T-s)\mu(s)ds\right|\right)^2. \tag{7}$$

Then notice that $\rho_\theta(s) = Ce^{As}B$, if we rescale C to ξC for some $\xi \in \mathbb{R}$, we have $\tau(\tilde{\theta}) = \xi^2 \cdot \tau(\theta)$ for $\tilde{\theta} = (\xi C, A, B)$. This induces a new initialization scheme, i.e., once the parameters $\theta = (C, A, B)$ are initialized by default, we rescale C to $\tilde{C}$ such that

$$\tilde{C} = \frac{1}{\sqrt{\tau(\theta)}} \cdot C = \frac{1}{\int_0^T |\rho_\theta(T-s)| \sqrt{K(s,s)}ds + \left|\int_0^T \rho_\theta(T-s)\mu(s)ds\right|} \cdot C \tag{8}$$

The new initialization scheme (8) guarantees that the SSM output is finite for *any* Gaussian process, ensuring the stability of the SSM at initialization in a general case. We formalize it in Proposition 1.

Proposition 1. *For a SSM $\int_0^T \rho_\theta(T-s)x(s)ds$, following the notations and settings in Section 3.1 & 4.1, under Assumption 1, for any fixed θ, let $\tilde{C}$ given by the rescale method (8), then for $\tilde{\theta} := (\tilde{C}, A, B)$, there exists a constant β_T depends on T such that $\mathbb{E}_x\left[\left|\int_0^T \rho_{\tilde{\theta}}(T-s)x(s)ds\right|\right] \leq \beta_T$.*

The proof is provided in Appendix E. Proposition 1 only requires the normalized Gaussian process to be stable (Assumption 1) but does not have any constraint on the original Gaussian process, which improves the robustness of SSMs across different temporal structures in a broader sense. When the input sequence is constant, i.e., $\mu(s) = c, K(s,s) = 0$, the initialization scheme (8) reduces to the default initialization (up to some constant) in (Gu et al., 2023). It is worth noting that different from the data normalization methods such as min-max normalization and standardization, our rescaling method (8) only changes the model parameters. This is important because normalization on the data

numerical values in language tasks can lead to loss of crucial information. For example, mathematical expressions like "$\max(1, 9) = 9$" have a contextual meaning where normalizing could result in the loss of structured information essential to understand.

Implementation In the practical training, the SSMs used for tasks such as image classification or language processing are usually deep and high dimensional ($d > 1$), while our initialization scheme (8) is designed based on the one-dimensional shallow SSM. To extend the initialization scheme to high-dimensional deep SSMs, we (1) treat all features to be independent and calculate $\tau(\theta)$ by its average along the feature dimension d; (2) rescale the initialization C for each layer via (8) for deep SSMs. Also, the calculation of the sequence statistics $\mu(s)$ and $K(s, s)$ is based on the input sequence, which is not the same as the raw sequence data if there is an encoder (or embedding) layer before the SSM layer. To reduce the computational cost, we conduct the rescaling operation (8) based on the *first* batch of the training sequence. We summarize the procedures for one-layer SSMs in Algorithm 1, where the $|\cdot|$ and $\sqrt{\cdot}$ in Line 5 represent to element-wise absolute value and element-wise square root respectively. $[\cdot]_L$ extracts the last position of an element obtained from the convolution. The $\mathsf{Mean}(\cdot)$ operation in Line 6 calculates the mean value of a vector.

Algorithm 1 Training one-layer SSMs with the initialization scheme (8)

Input: Training sequences $x_1, \ldots, x_n \in \mathbb{R}^{L\times d}$ with length L and dimension d, a SSM initialization $\theta_0 = (C, A, B)$, a SSM kernel function $k(\theta) \in \mathbb{R}^{L\times d}$, number of epochs s

1: **for** $i = 0$ to $s - 1$ **do**
2: **if** $i = 0$ **then**
3: Sample a minibatch sequence $x = (x^{(1)}, \ldots, x^{(B)}) \in \mathbb{R}^{B\times L\times d}$
4: Compute the mean $\mu \in \mathbb{R}^{L\times d}$ and variance $K \in \mathbb{R}^{L\times d}$ for x along the batch dimension
5: Compute $\tau(\theta_i)$ via convolution: $\tau(\theta_i) \leftarrow \left[|k(\theta_i)| * \sqrt{K} + |k(\theta_i) * \mu|\right]_L \in \mathbb{R}^d$
6: Average over the feature dimension: $\tau(\theta_i) \leftarrow \mathsf{Mean}^2(\tau(\theta_i))$
7: Rescale by the initialization scheme (8): $\tilde{C} \leftarrow C/\sqrt{\tau(\theta_i)}$
8: Start to train with the updated initialization $(\tilde{C}, A, B)$
9: **end if**
10: Regular training procedure
11: **end for**

Output: Updated model parameter θ_s

4.3 Generalization bound as a regularization method

In addition to its role as an initialization scheme, the generalization measure can also be regarded as a regularizer. In this section, we utilize the bound (4) to design a regularization method to improve the generalization performance, and simultaneously bring a little extra computational cost. For the generalization bound (4), we consider to use the dominant term (for large T) $\tau(\theta)$ defined in (7) as a regularizer. Then, the new empirical risk with regularization is given by

$$\tilde{R}_n(\theta) := \frac{1}{n}\sum_{i=1}^{n}\left|\int_0^T \rho_\theta(T-s)x_i(s)ds - y_i\right|^2 + \lambda\cdot\tau(\theta), \tag{9}$$

where $\lambda \geq 0$ is the regularization coefficient. Following the implementation details in Algorithm 1, we also adopt a mini-batch version of the sequence statistics $(\mu(s), K(s, s))$ to replace the full-batch version to reduce the computational cost. For multi-layer SSMs, we calculate the layer-wise regularization measure $\tau(\theta)$ w.r.t. the SSM input at each layer and sum them together. Again, we describe the training procedures for one-layer SSMs in Algorithm 2, where the $|\cdot|$ and $\sqrt{\cdot}$ in Line 5 represent to element-wise absolute value and element-wise square root respectively. $[\cdot]_L$ extracts the last position of an element obtained from the convolution. The $\mathsf{Mean}(\cdot)$ operation in Line 6 calculates the mean value of a vector.

Computational cost analysis From the training procedures in Algorithm 2, we can see that the newly introduced training complexity mainly comes from the calculation for the convolution between the SSM kernel and the sequence statistics (μ, K). Since the convolution can be conducted by the fast Fourier transform (Gu et al., 2022a) with complexity $\mathcal{O}(BdL\log L)$, then the new complexity for Algorithm 2 becomes $\mathcal{O}((B + 2)dL\log L)$, which is acceptable in the practical training.

Algorithm 2 Training one-layer SSMs with the regularization method (9)

Input: Training sequences $x_1, \ldots, x_n \in \mathbb{R}^{L\times d}$ with length L and dimension d, a SSM initialization θ_0, a SSM kernel function $k(\theta) \in \mathbb{R}^{L\times d}$, loss function $\tilde{R}(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, regularization coefficient λ, optimizer OPT, number of epochs s

1: **for** $i = 0$ to $s-1$ **do**
2: Sample a minibatch input $x = (x^{(1)}, \ldots, x^{(B)}) \in \mathbb{R}^{B\times L\times d}$ with labels $(y^{(1)}, \ldots, y^{(B)})$
3: Calculate the mean $\mu \in \mathbb{R}^{L\times d}$ and variance $K \in \mathbb{R}^{L\times d}$ for x along the batch dimension
4: Compute the SSM output via convolution: $y \leftarrow [k(\theta_i) * x]_L \in \mathbb{R}^{B\times d}$
5: Compute the regularization via convolution: $\tau(\theta_i) \leftarrow \left[|k(\theta_i)| * \sqrt{K} + |k(\theta_i) * \mu|\right]_L \in \mathbb{R}^d$
6: Average over the feature dimension: $\tau(\theta_i) \leftarrow \mathsf{Mean}^2(\tau(\theta_i))$
7: Compute the total loss $\mathcal{L} \leftarrow \frac{1}{B}\sum_{i=1}^{B} \tilde{R}(y_i, y^{(i)}) + \lambda \cdot \tau(\theta_i)$
8: Parameters update: $\theta_{i+1} \leftarrow \mathsf{OPT}(\theta_i, \mathcal{L})$
9: **end for**

Output: Updated model parameter θ_s

4.4 Experiments

This section contains experiments to demonstrate the effectiveness of the proposed initialization scheme (8) and the regularization method (9). We use a synthetic sequence dataset and the Long Range Arena (LRA) benchmark (Tay et al., 2021) for numerical validations.

A synthetic dataset We consider a synthetic sequence dataset generated by a centered Gaussian white noise with the autocovariance function $K(s,t) = \frac{1}{|b|\sqrt{\pi}} e^{-((s-t)/b)^2}$, which is a stationary Gaussian process and satisfies Assumption 1. Then we can get different temporal dependencies by varying the coefficient b, i.e., as the magnitude of b decreasing, the temporal dependence of the corresponding Gaussian white noise decreases as well. In particular, as $b \to 0$, $\frac{1}{|b|\sqrt{\pi}} e^{-(x/b)^2}$ becomes a delta function $\delta(x)$, entailing a zero temporal dependence for the sequence data.

In the following experiment, we generate the sequence data by the Gaussian white noise with $b = [1, 0.1, 0.01]$. For each input sequence $(x_1, \ldots, x_L)$, its corresponding label is obtained by $\sin(x_{[L/2]})$, i.e., the sine value of the time-lagged input. We use the S4 model (that only contains the convolution layer) to train the sequence data. More details about the experiment setup are provided in Appendix A.1. In Figure 2, we plot the model output $\mathbb{E}_x[|y_L|]$ and the gradient norm $\|\nabla R_n(\theta)\|$

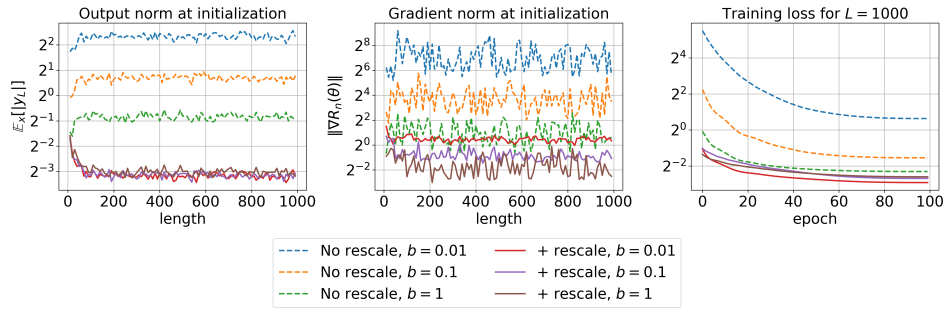

Figure 2: Effects of the initialization scheme (8) on the model output, the gradient norm and the optimization under different temporal dependencies. (Left) The output $\mathbb{E}_x[|y_L|]$ at initialization w.r.t. the Gaussian white noise sequence $(x_1, \ldots, x_L)$ for length L from 1 to 1000; (Middle) The gradient norm $\|\nabla R_n(\theta)\|$ at initialization w.r.t. the mean squared error (MSE) for varied sequence length; (Right) The training MSE curve for the Gaussian white noise with length $L = 1000$.

at initialization, and the training loss with different temporal patterns by varying the Gaussian white noise parameter b. We see that the initialization scheme (8) enhances the stability for the SSM output (matches with Proposition 1) and gradient norm at initialization across different temporal structures. For the optimization, (8) makes the initial training loss more stable and as a result, improving the training performance. This is also verified in Table 1, where we report the final training loss and

	Training loss (MSE)				Test loss (MSE)			
	w/o (8), (9)	w (8)	w (9)	(8) + (9)	w/o (8), (9)	w (8)	w (9)	(8) + (9)
$b = 1$	$0.21_{\pm 0.02}$	$\mathbf{0.16}_{\pm 0.02}$	$0.28_{\pm 0.02}$	$0.24_{\pm 0.02}$	$0.54_{\pm 0.04}$	$0.57_{\pm 0.07}$	$0.42_{\pm 0.009}$	$\mathbf{0.42}_{\pm 0.02}$
$b = 0.1$	$0.35_{\pm 0.01}$	$\mathbf{0.16}_{\pm 0.007}$	$0.51_{\pm 0.01}$	$0.32_{\pm 0.01}$	$1.00_{\pm 0.05}$	$1.20_{\pm 0.12}$	$0.71_{\pm 0.03}$	$\mathbf{0.65}_{\pm 0.02}$
$b = 0.01$	$1.20_{\pm 0.33}$	$\mathbf{0.14}_{\pm 0.02}$	$1.94_{\pm 0.33}$	$0.32_{\pm 0.04}$	$3.37_{\pm 0.50}$	$1.43_{\pm 0.13}$	$2.22_{\pm 0.16}$	$\mathbf{0.67}_{\pm 0.03}$

Table 1: Training and test loss on the Gaussian white noise sequences with different coefficients b. Under the initialization scheme (8), SSMs achieve better optimization performance and are more robust across different temporal dependencies. With both the initialization scheme (8) the regularization method (9), the generalization performance gets significantly improved.

test loss with mean and standard error over 3 independent runs. From the training loss results, we observe that the initialization scheme (8) improves the robustness of SSMs to different temporal patterns (by varing b), and the optimization performance is better compared with training without (8). From the test loss results, we can see that the regularization method (9) improves the generalization performance. By combining the initialization scheme (8) and the regularization method (9), both the robustness and generalization get enhanced across various temporal structures of the sequence data.

LRA benchmark We investigate the effects of the initialization scheme (8) and the regularization method (9) on the LRA benchmark. We use 5 out of 6 tasks excluding the PathX dataset due to the limitation of the computational resource. We use the S4 model and follow the training rules in Gu et al. (2023). See more details on the dataset description and the experiment setup in Appendix A.2. Similar to the experiment of the synthetic dataset, we consider 4 training settings: w/o (8), (9); w (8); w (9); (8) + (9). As shown in Table 2, We find that both the initialization scheme (8) and the regularization (9) improve the SSM performance in the LRA benchmark. Specifically, when comparing w/o (8), (9) vs w (8), and w/o (8), (9) vs w (9), we can see that separately using the initialization scheme (8) or the regularization method (9) improves the generalization performance individually. When these two optimization algorithms are combined, one can get a better test accuracy in 3 tasks. This indicates that our proposed optimization designs effectively improve the generalization performance. It is also worth noting that the magnitude of the regularization coefficients in Table 2 are different for different tasks. This is because the generalization measure (7) varies across different tasks. We include more discussions and experiment results in Appendix A.2.

	w/o (8), (9)	w (8)	w (9)	(8) + (9)
ListOps (L = 2048)	59.45	60.30	**60.65** ($\lambda = 10^{-3}$)	60.40 ($\lambda = 10^{-3}$)
Text (L = 2048)	79.27	81.44	81.45 ($\lambda = 10^{-2}$)	**82.56** ($\lambda = 10^{-2}$)
Retrieval (L = 4000)	88.28	89.38	89.21 ($\lambda = 10^{-3}$)	**90.13** ($\lambda = 10^{-3}$)
Image (L = 1024)	87.99	88.11	87.79 ($\lambda = 10^{-5}$)	**88.28** ($\lambda = 10^{-2}$)
Pathfinder (L = 1024)	87.84	87.95	**90.36** ($\lambda = 10^{-5}$)	90.03 ($\lambda = 10^{-6}$)

Table 2: Test accuracy on the 5 tasks of the LRA benchmark under different settings. λ is the regularization coefficient.

5 Discussion

In this work, we study the optimization and the generalization for SSMs. Specifically, we give a data-dependent generalization bound, revealing an effect of the temporal dependencies of the sequence data on the generalization. Based on the bound, we design two algorithms to improve the optimization and generalization for SSMs across different temporal patterns. The first is a new initialization scheme, by which we normalize the generalization measure at initialization, improving the robustness of SSMs to various temporal structures. The second is a new regularization method, which enhances the generalization performance in sequence modeling. However, in this paper we do not address the feature dependencies when calculating the generalization measure (7) for high-dimensional SSMs, but simply treat all the features are independent. It is interesting to understand the effects of feature structures on the optimization and generalization of SSMs, which we leave for future work.

Reproducibility The the generalization bound (3) for linear regression is proved in Appendix B. The proof for Theorem 1 and another generalization bound that does not rely on Assumption 1 are provided in Appendix D.4 and Appendix D.5 respectively. The derivations for (5) and (6) in Section 4.1 are given in Appendix C. The proof for Proposition 1 is in Appendix E. The details for the experiment settings are shown in Appendix A.1 and Appendix A.2.

REFERENCES

Robert J Adler, Jonathan E Taylor, et al. *Random fields and geometry*, volume 80. Springer, 2007.

Zeyuan Allen-Zhu and Yuanzhi Li. Can sgd learn recurrent neural networks with provable generalization? *Advances in Neural Information Processing Systems*, 32, 2019.

Ehsan Azmoodeh, Tommi Sottinen, Lauri Viitasaari, and Adil Yazigi. Necessary and sufficient conditions for hölder continuity of gaussian processes. *Statistics & Probability Letters*, 94:230–235, 2014.

Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

Leonard E Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state markov chains. *The annals of mathematical statistics*, 37(6):1554–1563, 1966.

Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

Minshuo Chen, Xingguo Li, and Tuo Zhao. On generalization bounds of a family of recurrent neural networks. *arXiv preprint arXiv:1910.12947*, 2019.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

Bhaskar Dasgupta and Eduardo Sontag. Sample complexity for learning recurrent perceptron mappings. *Advances in Neural Information Processing Systems*, 8, 1995.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186. Association for Computational Linguistics, June 2019.

Daniel Y Fu, Tri Dao, Khaled Kamal Saab, Armin W Thomas, Atri Rudra, and Christopher Re. Hungry hungry hippos: Towards language modeling with state space models. In *The Eleventh International Conference on Learning Representations*, 2023.

Albert Gu, Karan Goel, and Christopher Re. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022a.

Albert Gu, Ankit Gupta, Karan Goel, and Christopher Ré. On the parameterization and initialization of diagonal state space models. *Advances in Neural Information Processing Systems*, 35, 2022b.

Albert Gu, Isys Johnson, Aman Timalsina, Atri Rudra, and Christopher Re. How to train your HIPPO: State space models with generalized orthogonal basis projections. In *International Conference on Learning Representations*, 2023.

Ankit Gupta, Albert Gu, and Jonathan Berant. Diagonal state spaces are as effective as structured state spaces. In *Advances in Neural Information Processing Systems*, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 29(6):82–97, 2012.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Pascal Koiran and Eduardo D Sontag. Vapnik-chervonenkis dimension of recurrent neural networks. *Discrete Applied Mathematics*, 86(1):63–79, 1998.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Michel Ledoux and Michel Talagrand. *Probability in Banach Spaces: isoperimetry and processes*. Springer Science & Business Media, 2013.

Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyou Zhou, Wenhu Chen, Yu-Xiang Wang, and Xifeng Yan. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. *Advances in neural information processing systems*, 32, 2019.

Zhong Li, Jiequn Han, Weinan E, and Qianxiao Li. On the curse of memory in recurrent neural networks: Approximation and optimization analysis. In *International Conference on Learning Representations*, 2021.

Zhong Li, Jiequn Han, Weinan E, and Qianxiao Li. Approximation and optimization theory for linear continuous-time recurrent neural networks. *The Journal of Machine Learning Research*, 23 (1):1997–2081, 2022.

Drew Linsley, Junkyung Kim, Vijay Veerabadran, Charles Windolf, and Thomas Serre. Learning long-range spatial dependencies with horizontal gated recurrent units. *Advances in neural information processing systems*, 31, 2018.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150. Association for Computational Linguistics, June 2011.

Harsh Mehta, Ankit Gupta, Ashok Cutkosky, and Behnam Neyshabur. Long range language modeling via gated state spaces. In *The Eleventh International Conference on Learning Representations*, 2023.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. The MIT Press, 2012.

Nikita Nangia and Samuel R Bowman. Listops: A diagnostic dataset for latent tree learning. *arXiv preprint arXiv:1804.06028*, 2018.

Antonio Orvieto, Samuel L Smith, Albert Gu, Anushan Fernando, Caglar Gulcehre, Razvan Pascanu, and Soham De. Resurrecting recurrent neural networks for long sequences. *arXiv preprint arXiv:2303.06349*, 2023.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pp. 1310–1318. Pmlr, 2013.

Michael Poli, Stefano Massaroli, Eric Nguyen, Daniel Y Fu, Tri Dao, Stephen Baccus, Yoshua Bengio, Stefano Ermon, and Christopher Ré. Hyena hierarchy: Towards larger convolutional language models. *arXiv preprint arXiv:2302.10866*, 2023.

Eric Qu, Xufang Luo, and Dongsheng Li. Data continuity matters: Improving sequence modeling with lipschitz regularizer. In *The Eleventh International Conference on Learning Representations*, 2023.

Dragomir R. Radev, Pradeep Muthukrishnan, and Vahed Qazvinian. The ACL Anthology network corpus. In *Proceedings of the 2009 Workshop on Text and Citation Analysis for Scholarly Digital Libraries (NLPIR4DL)*, pp. 54–61. Association for Computational Linguistics, August 2009.

Jimmy T.H. Smith, Andrew Warrington, and Scott Linderman. Simplified state space layers for sequence modeling. In *The Eleventh International Conference on Learning Representations*, 2023.

Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. Long range arena : A benchmark for efficient transformers. In *International Conference on Learning Representations*, 2021.

Zhuozhuo Tu, Fengxiang He, and Dacheng Tao. Understanding generalization in recurrent neural networks. In *International Conference on Learning Representations*, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Shida Wang, Zhong Li, and Qianxiao Li. Inverse approximation theory for nonlinear recurrent neural networks. *arXiv preprint arXiv:2305.19190*, 2023.

Jiong Zhang, Qi Lei, and Inderjit Dhillon. Stabilizing gradients for deep neural networks via efficient svd parameterization. In *International Conference on Machine Learning*, pp. 5806–5814. PMLR, 2018.

A EXPERIMENTS DETAILS

In this section, we provide more details for the experiments of the synthetic dataset and the LRA benchmark in Section 4.4.

A.1 THE SYNTHETIC EXPERIMENT

For the Gaussian white noise sequences, we generate 100 i.i.d. sequences for training and 1000 i.i.d. sequences for test. The timescale for the discrete sequences is set to be 1, i.e., to generate a Gaussian white noise sequence with length L, we sample from a multivariate normal distribution with mean zero and covariance matrix $K_{i,j} = h(i-j)$ for $i, j \in [1 : L]$, where $h(t) = \frac{1}{|b|\sqrt{\pi}} e^{-(t/b)^2}$. The model that we use is the one-layer S4 model that only contains the FFTConv (fast Fourier transform convolution) layer and without activation and the skip connection ($D = 0$) (Gu et al., 2022a). The state space dimension for the FFTConv layer is 64, other settings such as the discretization, the initialization and the parameterization follow the default settings in Gu et al. (2023), i.e., we use the ZOH discretization, the LegS initialization and the exponential parameterization for the hidden state matrix A.

For the optimizer, we follow Gu et al. (2023) to set the optimizer by groups. For the (ZOH) timescale Δ, the hidden state matrices A, B, we use Adam optimizer with learning rate 0.001, while for the matrix C, we use AdamW with learning rate 0.01 and decay rate 0.01. For all the parameters, we use the cosine annealing schedule. The batch size is set to be 100 (full batch) and the training epochs is 100. The regularization coefficient λ used for training with (9) is set to be 0.01 across all the temporal patterns.

A.2 LRA BENCHMARK

Datasets The datasets in the LRA benchmark contain (1) ListOps (Nangia & Bowman, 2018), a dataset that is made up of a list of mathematical operations with answers; (2) Text (Maas et al., 2011), a movie review dataset collected from IMDB, which is used for sentiment analysis; (3) Retrieval (Radev et al., 2009), a task of retrieving documents utilizing byte-level texts from the ACL Anthology Network. (4)Image (Krizhevsky et al., 2009), a sequential CIFAR10 dataset used for sequence classification; (5) Pathfinder (Linsley et al., 2018), a task that requires a model to tell whether two points in an image are connected by a dashed path.

Models For the model architectures, we use the standard S4 model with the default LegS initialization, ZOH discretization and exponential parameterization as in Gu et al. (2023). For the optimizer, we also follow the standard setup in Gu et al. (2023) that the hidden state matrices are trained in a relatively small learning rate with no weight decay, while the other parameters are trained with AdamW with a larger learning rate. We use a reduce learning rate schedule with different patience for diffrent tasks. Let D, H, N denote the depth, feature dimension and hidden state space dimension respectively, we summarize all the hyperparameters for the S4 model in Table 3.

	D	H	N	Dropout	Learning rate	Batch size	Epochs	Weight decay	Patience
ListOps	6	128	64	0	0.01	50	50	0.01	5
Text	4	128	64	0	0.01	50	40	0	10
Retrieval	6	256	64	0	0.002	64	25	0	20
Image	6	512	64	0.2	0.004	50	100	0.01	10
Pathfinder	6	256	64	0.1	0.004	100	200	0	10

Table 3: List of the S4 model hyperparameters for the LRA benchmark.

Regularization coefficients When training with the regularization method (9), we choose the regularization coefficient λ from $\{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$ when the model performs best on the validation set. We notice that in the 4 tasks that regularization helps (when comparing w/o (8), (9) vs w (9) in Table 2), the optimal choice of λ varies. Especially for the Pathfinder dataset, the optimal λ is much smaller than the other datasets, but the improvement on the generalization is the largest. To investigate the magnitude of the optimal λ, we plot the generalization measure (7) during

the training process. As shown in Figure 3, the generalization measure when training the Pathfinder dataset without regularization increases to 4000 after 75 epochs, making the generalization measure very sensitive to the magnitude of λ, thus it should be set with a relatively small value.

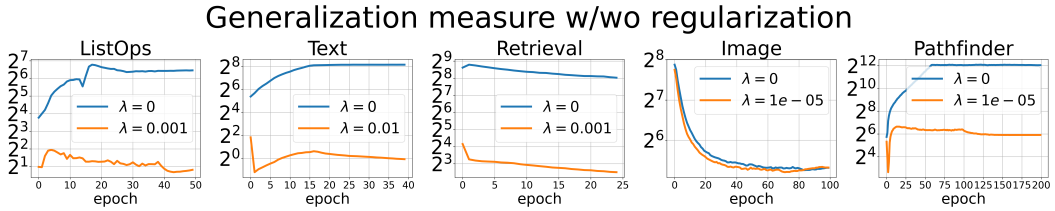

Figure 3: Generalization measure (7) of S4 models on the 5 tasks of the LRA benchmark with or without regularization for different regularization coefficient λ in (9).

We also plot the generalization measure (7) when training only with the initialization scheme (8) and training with both the initialization scheme (8) and the regularization method (9) in Figure 4. It is interesting to find that only using the initialization scheme (8) only guarantees that the generalization measure is small at initialization (which is as expected because we only rescale the model parameters at initialization). After training with some epochs, the generalization measure starts to increase. If we use both the initialization scheme and the regularization method, we get a much lower generalization measure, and as Table 2 shows, combining (8) and (9) produces better generalization performance in some tasks.

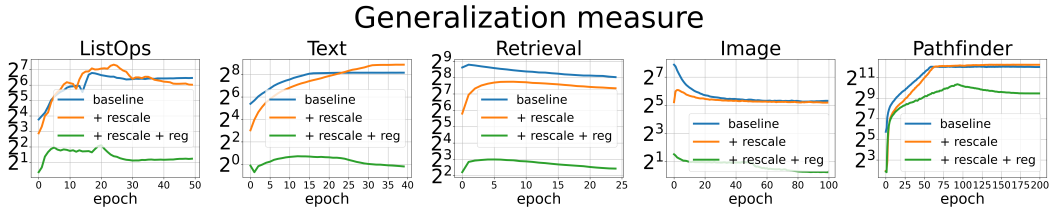

Figure 4: Generalization measure (7) of S4 models on the 5 tasks of the LRA benchmark with or without the regularization (9) and the rescaling method (8) during training. 'baseline' means training without regularization nor rescaling; '+ rescale' represents training with rescaling but without regularization; '+ rescale + reg' corresponds to training with rescaling and regularization, where the regularization coefficient λ is adopted based on the '(8) + (9)' column in Table 2.

B Proof for the linear regression result in Section 3.2.

In this section, we give the proof for the generalization bound (3). The proof is based on the following uniform-convergence generalization bound in Mohri et al. (2012).

Lemma 1. *Consider a family of functions $\mathcal{F}$ mapping from $\mathcal{Z}$ to $[a, b]$. Let $\mathcal{D}$ denote the distribution according to which samples are drawn. Then for any $\delta > 0$, with probability at least $1 - \delta$ over the draw of an i.i.d. sample $S = \{z_1, \ldots, z_n\}$, the following holds for all $f \in \mathcal{F}$:*

$$\mathbb{E}_{z\sim\mathcal{D}}\left[f(z)\right] - \frac{1}{n}\sum_{i=1}^{n} f(z_i) \leq 2\mathcal{R}_S(\mathcal{F}) + 3(b-a)\sqrt{\frac{\log(2/\delta)}{2n}},$$

where $\mathcal{R}_S(\mathcal{F})$ is the empirical Rademacher complexity with respect to the sample S, defined as: $\mathcal{R}_S(\mathcal{F}) = \mathbb{E}_\sigma\left[\sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} \sigma_i f(z_i)\right]$. $\{\sigma_i\}_{i=1}^n$ are i.i.d. random variables drawn from $U\{-1, 1\}$ with $P(\sigma_i = 1) = P(\sigma_i = -1) = 0.5$.

And the Talagrand's contraction lemma Ledoux & Talagrand (2013).

Lemma 2. *Let H be a hypothesis set of functions mapping $\mathcal{X}$ to $\mathbb{R}$ and $\Psi_1, \ldots, \Psi_m$, μ-Lipschitz functions for some $\mu > 0$. Then, for any sample S of m points $x_1, ..., x_m \in \mathcal{X}$, the following*

inequality holds

$$\frac{1}{m}\mathbb{E}_\sigma\left[\sup_{h\in H}\sum_{i=1}^{m}\sigma_i\left(\Psi_i\circ h\right)(x_i)\right]\leq\frac{\mu}{m}\mathbb{E}_\sigma\left[\sup_{h\in H}\sum_{i=1}^{m}\sigma_i h\left(x_i\right)\right]$$

Now we begin our proof:

Proof. First, notice for any $i\in[1:n]$ and $\theta\in\Theta$, we have

$$(\theta^\top x_i-y_i)^2\leq 2(\theta^\top x_i)^2+2y_i^2\leq 2r^2R^2+2$$

Second, note that $(\theta^\top x_i-y_i)^2$ is $2\sup_{\theta\in\Theta,i\in[1:n]}|\theta^\top x_i-y_i|$-Lipschitz (the maximum gradient norm) with respect to $\theta^\top x_i-y_i$, and we can bound the Lipschitz constant as

$$2\sup_{\theta\in\Theta,i\in[1:n]}|\theta^\top x_i-y_i|\leq 2rR+2$$

Then by Lemma 2, the Rademacher complexity for the linear model is bounded as

$$\begin{aligned}
\mathcal{R}_S(\mathcal{F})&=\frac{1}{n}\mathbb{E}_\sigma\left[\sup_{\|\theta\|_2\leq R}\sum_{i=1}^{n}\sigma_i(\theta^\top x_i-y_i)^2\right]\\
&\leq\frac{2rR+2}{n}\mathbb{E}_\sigma\left[\sup_{\|\theta\|_2\leq R}\sum_{i=1}^{n}\sigma_i(\theta^\top x_i-y_i)\right]\\
&=\frac{2rR+2}{n}\mathbb{E}_\sigma\left[\sup_{\|\theta\|_2\leq R}\sum_{i=1}^{n}\sigma_i\theta^\top x_i\right]\\
&\leq\frac{2R(rR+1)}{n}\mathbb{E}_\sigma\left\|\sum_{i=1}^{n}\sigma_i x_i\right\|\\
&\leq\frac{2R(rR+1)}{n}\sqrt{\mathbb{E}_\sigma\left\|\sum_{i=1}^{n}\sigma_i x_i\right\|^2}\\
&=\frac{2R(rR+1)}{n}\sqrt{\sum_{i=1}^{n}\|x_i\|^2}\\
&\leq\frac{2rR(rR+1)}{\sqrt{n}}
\end{aligned}$$

Combining with the function value bound, we get the desired bound (3) by Lemma 1. □

C Derivations for (5) and (6) in Section 4.1

For the left zero padding transformation, the key term in (4) becomes

$$\begin{aligned}
&\int_0^{2T}|\rho_\theta(2T-t)|\sqrt{K_1(t,t)}dt+\left|\int_0^{2T}\rho_\theta(2T-t)\mu_1(t)dt\right|+1\\
=&\int_0^{T}|\rho_\theta(T-t)|\sqrt{K(t,t)}dt+\left|\int_0^{T}\rho_\theta(T-t)\mu(t)dt\right|+1
\end{aligned}$$

For the right zero padding transformation, the key term in (4) becomes

$$\begin{aligned}
&\int_0^{2T} |\rho_\theta(2T-t)| \sqrt{K_2(t,t)}dt + \left| \int_0^{2T} \rho_\theta(2T-t)\mu_2(t)dt \right| + 1 \\
=&\int_0^{T} |\rho_\theta(2T-t)| \sqrt{K(t,t)}dt + \left| \int_0^{T} \rho_\theta(2T-t)\mu(t)dt \right| + 1 \\
=&\int_0^{T} \left| Ce^{AT}e^{A(T-t)}B \right| \sqrt{K(t,t)}dt + \left| \int_0^{T} Ce^{AT}e^{A(T-t)}B\mu(t)dt \right| + 1
\end{aligned}$$

Then we get (5) and (6).

D Proof for Theorem 1 and Theorem 2

In this section, we will prove Theorem 1 and Theorem 2. Before moving into the formal proof, we first introduce some important lemmas that help to build the proof.

D.1 Sub-exponential random variables

In this subsection, we introduce the sub-exponential random variable and its properties.

Definition 1. *A random variable X is (τ^2, b) sub-exponential if*

$$\mathbb{E}[e^{\lambda(X-\mathbb{E}[X])}] \leq e^{\frac{\tau^2\lambda^2}{2}}, \quad \forall|\lambda| < \frac{1}{b}$$

Second is the concentration inequality for sub-exponential random variables:

Proposition 2. *If X is (τ^2, b) sub-exponential, then*

$$P\left(|X-\mathbb{E}[X]| \geq t\right) \leq 2\exp\left(-\min\left\{\frac{t^2}{2\tau^2}, \frac{t}{2b}\right\}\right)$$

Example 1. *Let $Z \sim \mathcal{N}(0, \sigma^2)$, then $X = Z^2$ is $(4\sigma^4, 4\sigma^2)$ sub-exponential. This is because that*

$$\mathbb{E}[e^{\lambda(X-\sigma^2)}] = \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{\infty} e^{\lambda(z^2-\sigma^2)}e^{-\frac{z^2}{2\sigma^2}}dz = \frac{e^{-\lambda\sigma^2}}{\sqrt{1-2\lambda\sigma^2}} \leq \exp\left(2\lambda^2\sigma^4\right), \quad \forall|\lambda| < \frac{1}{4\sigma^2}$$

Proposition 3. *If X is (τ^2, b) sub-exponential, then $\forall\alpha \in \mathbb{R}$, αX is $(\alpha^2\tau^2, |\alpha|b)$ sub-exponential. If X_1 is (τ_1^2, b_1) sub-exponential and X_2 is (τ_2^2, b_2) sub-exponential (not necessarily independent), then $X_1 + X_2$ is $((\tau_1+\tau_2)^2, \max\left((1+\tau_2/\tau_1)b_1, (1+\tau_1/\tau_2)b_2\right))$ sub-exponential. Specifically, if $X_1, \ldots, X_n$ are sub-exponential with coefficients $(\tau_1^2, b_1), \ldots, (\tau_n^2, b_n)$ respectively, suppose that there exists a constant c such that $\frac{b_1}{\tau_1} = \cdots = \frac{b_n}{\tau_n} = c$, then $X_1 + \cdots + X_n$ is $((\tau_1 + \cdots + \tau_n)^2, c(\tau_1 + \cdots + \tau_n))$ sub-exponential.*

Proof. Since X is (τ^2, b) sub-exponential, then

$$\mathbb{E}[e^{\lambda(\alpha X-\mathbb{E}[\alpha X])}] \leq e^{\frac{(\alpha\tau)^2\lambda^2}{2}}, \quad \forall|\lambda| < \frac{1}{|\alpha|b}$$

Therefore, αX is $(\alpha^2\tau^2, |\alpha|b)$ sub-exponential.

For any $p, q > 1$ such that $\frac{1}{p} + \frac{1}{q} = 1$, using Hölder's inequality, we have

$$\begin{aligned}
\mathbb{E}\exp\{\lambda\left(X_1 + X_2 - \mathbb{E}[X_1 + X_2]\right)\} &\leq \left[\mathbb{E}e^{p\lambda(X_1-\mathbb{E}[X_1])}\right]^{1/p}\left[\mathbb{E}e^{q\lambda(X_2-\mathbb{E}[X_2])}\right]^{1/q} \\
&\leq \exp\left\{\frac{\lambda^2}{2}\left(p\tau_1^2 + q\tau_2^2\right)\right\}, \quad \forall|\lambda| < \min\left(\frac{1}{pb_1}, \frac{1}{qb_2}\right)
\end{aligned}$$

Minimizing over $\frac{1}{p}+\frac{1}{q}=1$, we get $p=1+\tau_2/\tau_1, q=1+\tau_1/\tau_2$, then

$$\mathbb{E}\exp\{\lambda\left(X_1+X_2-\mathbb{E}[X_1+X_2]\right)\}\leq\exp\left\{\frac{\lambda^2}{2}\left(\tau_1+\tau_2\right)^2\right\},\quad\forall|\lambda|<\min\left(\frac{\tau_1}{(\tau_1+\tau_2)b_1},\frac{\tau_2}{(\tau_1+\tau_2)b_2}\right)$$

For the case when $\frac{b_1}{\tau_1}=\cdots=\frac{b_n}{\tau_n}=c$, one can prove the result by induction. □

Remark 2. *Similar results can also be obtained for linear combination of sub-Gaussian random variables. Also note that if X_1 and X_2 are independent, then X_1+X_2 is $\left(\tau_1^2+\tau_2^2,\max(b_1,b_2)\right)$ sub-exponential.*

Proposition 4. *Suppose that $X_1,\ldots,X_n$ are independent (τ^2,b) sub-exponential random variables, then the average $Z=\frac{X_1+\cdots+X_n}{n}$ is $(\tau^2/n,b/n)$ sub-exponential. This is because that*

$$\mathbb{E}[e^{\lambda(Z-\mathbb{E}[Z])}]=\prod_i^n\mathbb{E}[e^{\lambda(X_i-\mathbb{E}[X_i])/n}]\leq\exp\left(\frac{\tau^2\lambda^2}{2n}\right),\quad\forall|\lambda|<\frac{n}{b}$$

Then by the concentration inequality, we have

$$P\left(\left|\frac{X_1+\cdots+X_n}{n}-\mathbb{E}[X]\right|\geq t\right)\leq 2\exp\left(-\min\left\{\frac{nt^2}{2\tau^2},\frac{nt}{2b}\right\}\right)$$

D.2 Integral of Gaussian processes

In this subsection, we show that the integral of a Gaussian process is also Gaussian, and the integral of square of a Gaussian process is sub-exponential, which is formalized in the following lemma.

Lemma 3. *If $\{x(t)\}_{t\geq0}$ is a centered and square integrable Gaussian process with the kernel function $\mathbb{E}[x(s)x(t)]=K(s,t)$, then for any $T>0$, the integral $\int_0^T x(t)dt$ is Gaussian with mean 0 and variance $\int_0^T\int_0^T K(s,t)dsdt$. Moreover, $\int_0^T x^2(t)dt$ is $\left(4\left(\int_0^T K(t,t)dt\right)^2,4\int_0^T K(t,t)dt\right)$ sub-exponential.*

Proof. Note that the (Riemann) integral $\int_0^T x(t)dt$ and $\int_0^T x^2(t)dt$ can be approximated by the Riemann sums:

$$\int_0^T x(t)dt=\lim_{N\to\infty}\frac{T}{N}\sum_{j=1}^N x\left(\frac{jT}{N}\right)$$
$$\int_0^T x^2(t)dt=\lim_{N\to\infty}\frac{T}{N}\sum_{j=1}^N x^2\left(\frac{jT}{N}\right)$$

Since $\{x(t)\}_{t\geq0}$ is a Gaussian process, then $x\left(\frac{jT}{N}\right)$ is a Gaussian random variable with

$$x\left(\frac{jT}{N}\right)\sim\mathcal{N}\left(0,K\left(\frac{jT}{N},\frac{jT}{N}\right)\right).$$

By the definition of Gaussian process, the linear combination $\frac{T}{N}\sum_{j=1}^N x\left(\frac{jT}{N}\right)$ is also a Gaussian random variable with mean 0 and variance

$$\begin{aligned}\mathbb{E}\left[\left(\frac{T}{N}\sum_{j=1}^N x\left(\frac{jT}{N}\right)\right)^2\right]&=\frac{T^2}{N^2}\sum_{i,j=1}^N\mathbb{E}\left[x\left(\frac{iT}{N}\right)x\left(\frac{jT}{N}\right)\right]\\&=\frac{T^2}{N^2}\sum_{i,j=1}^N K\left(\frac{iT}{N},\frac{jT}{N}\right)\end{aligned}$$

When $n \to \infty$,

$$\lim_{n\to\infty} \mathbb{E}\left[\left(\frac{T}{N}\sum_{j=1}^{N} x\left(\frac{jT}{N}\right)\right)^2\right] = \int_0^T \int_0^T K(s,t)dsdt$$

Therefore, $\int_0^T x(t)dt$ is Gaussian with mean 0 and variance $\int_0^T \int_0^T K(s,t)dsdt$.

Also, since $x\left(\frac{jT}{N}\right)$ is Gaussian, we have $x^2\left(\frac{jT}{N}\right)$ is $\left(4K^2\left(\frac{jT}{N},\frac{jT}{N}\right), 4K\left(\frac{jT}{N},\frac{jT}{N}\right)\right)$ sub-exponential. Notice that $\frac{4K\left(\frac{jT}{N},\frac{jT}{N}\right)}{\sqrt{4K^2\left(\frac{jT}{N},\frac{jT}{N}\right)}} = 2$, then by Proposition 3, $\frac{T}{N}x^2\left(\frac{jT}{N}\right)$ is $\left(\frac{4T^2K^2\left(\frac{jT}{N},\frac{jT}{N}\right)}{N^2}, \frac{4TK\left(\frac{jT}{N},\frac{jT}{N}\right)}{N}\right)$ sub-exponential, and the sum $\frac{T}{N}\sum_{j=0}^{N-1} x^2\left(\frac{jT}{N}\right)$ is sub-exponential with coefficients

$$\left(\sum_{j=1}^{N} \frac{2TK\left(\frac{jT}{N},\frac{jT}{N}\right)}{N}\right)^2, \quad 2\sum_{j=1}^{N} \frac{2TK\left(\frac{jT}{N},\frac{jT}{N}\right)}{N}$$

Taking the limit, we have

$$\lim_{N\to\infty} \sum_{j=1}^{N} \frac{TK\left(\frac{jT}{N},\frac{jT}{N}\right)}{N} = \int_0^T K(t,t)dt.$$

Therefore, $\int_0^T x^2(t)dt$ is $\left(4\left(\int_0^T K(t,t)dt\right)^2, 4\int_0^T K(t,t)dt\right)$ sub-exponential. □

D.3 Some useful lemmas

In this section, we introduce some useful lemmas to build the proof.

The first is the Borell-TIS Inequality (Adler et al., 2007), which is used to prove exceedence probabilities for Gaussian process.

Lemma 4. *Let $\{f_t\}_{t\in T}$ be a centered (i.e., mean zero) Gaussian process on T, with $\|f\|_T := \sup_{t\in T}|f_t|$ almost surely finite, and let $\sigma_T^2 := \sup_{t\in T}\mathbb{E}|f_t|^2$. Then $\mathbb{E}(\|f\|_T)$ and σ_T are both finite, and, for each $u > 0$,*

$$P\left(\|f\|_T > \mathbb{E}(\|f\|_T) + u\right) \le \exp\left(\frac{-u^2}{2\sigma_T^2}\right)$$

The second lemma is about necessary and sufficient condition for the Hölder continuity of the Gaussian process (Azmoodeh et al., 2014, Theorem 1.).

Lemma 5. *A centered (mean zero) Gaussian process X is Hölder continuous of any order $a < H$, i.e.,*

$$|X_t - X_s| \le C_\varepsilon |t-s|^{H-\varepsilon}, \quad \forall \varepsilon \in (0, H)$$

if and only if there exists constants c_ε such that

$$\mathbb{E}\left[(X_t - X_s)^2\right] \le c_\varepsilon (t-s)^{2H-2\varepsilon}, \quad \forall \varepsilon \in (0, H)$$

The third lemma is the Massart Lemma for the Rademacher complexity with finite class.

Lemma 6. *Let $\mathcal{A}$ be some finite subset of R^m and $\sigma_1, \ldots, \sigma_m$ be independent Rademacher random variables. Let $r = \sup_{a\in\mathcal{A}}\|a\|$. Then, we have,*

$$\mathbb{E}_\sigma\left[\sup_{a\in\mathcal{A}} \sum_{i=1}^{m} \sigma_i a_i\right] \le r\sqrt{2\log|\mathcal{A}|}$$

D.4 PROOF OF THEOREM 1

In this subsection, we are ready to prove the main result Theorem 1.

Proof. We let $g_\theta(x) := \int_0^T \rho_\theta(T-t)x(t)dt - y$, then the generalization gap is given by

$$R_x(\theta) - R_n(\theta) = \mathbb{E}_x[g_\theta^2(x)] - \frac{g_\theta^2(x_1) + \ldots + g_\theta^2(x_n)}{n}.$$

Now let hypothesis space $\mathcal{F} = \{x \mapsto g_\theta^2(x) : \theta \in \Theta\}$, then its empirical Rademacher complexity is given by

$$\begin{aligned}\mathcal{R}_S(\mathcal{F}) &= \mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\frac{1}{n}\sum_{i=1}^n \sigma_i g_\theta^2(x_i)\right] \\ &= \frac{1}{n}\mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\sum_{i=1}^n \sigma_i \left|\int_0^T \rho_\theta(T-t)x_i(t)dt - y_i\right|^2\right]\end{aligned}$$

By the Talagrand's contraction Lemma 2, since $g_\theta^2(x_i)$ is $2\sup_{\theta\in\Theta, i\in[1:n]}|g_\theta(x_i)|$ Lipschitz, we have

$$\begin{aligned}\mathcal{R}_S(\mathcal{F}) &\le 2\sup_{\theta\in\Theta, i\in[1:n]}|g_\theta(x_i)| \cdot \frac{1}{n}\mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\sum_{i=1}^n \sigma_i\left(\int_0^T \rho_\theta(T-t)x_i(t)dt - y_i\right)\right] \\ &= \frac{2\sup_{\theta\in\Theta, i\in[1:n]}|g_\theta(x_i)|}{n}\mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\int_0^T \rho_\theta(T-t)\sum_{i=1}^n \sigma_i x_i(t)dt\right]\end{aligned}$$

Now we separate the expectation into two parts: the unbiased part invovled with $x_i(t) - \mu(t)$ and the biased part $\mu(t)$, by noticing that

$$\begin{aligned}&\mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\int_0^T \rho_\theta(T-t)\sum_{i=1}^n \sigma_i x_i(t)dt\right] \\ =&\mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\int_0^T \rho_\theta(T-t)\sum_{i=1}^n \sigma_i (x_i(t)-\mu(t))dt + \int_0^T \rho_\theta(T-t)\sum_{i=1}^n \sigma_i\mu(t)dt\right] \\ \le&\mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\int_0^T \rho_\theta(T-t)\sum_{i=1}^n \sigma_i (x_i(t)-\mu(t))dt\right] + \mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\int_0^T \rho_\theta(T-t)\sum_{i=1}^n \sigma_i\mu(t)dt\right]\end{aligned}$$

For the unbiased part, by the Hölder's inequality, for any $p, q \in [1, \infty]$ such that $\frac{1}{p} + \frac{1}{q} = 1$,

$$\begin{aligned}&\mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\int_0^T \rho_\theta(T-t)\sum_{i=1}^n \sigma_i (x_i(t)-\mu(t))dt\right] \\ \le&\sup_{\theta\in\Theta}\left(\int_0^T |\rho_\theta^p(T-t)|\, K^{p/2}(t,t)dt\right)^{1/p}\mathbb{E}_\sigma\left[\left(\int_0^T\left|\sum_{i=1}^n \sigma_i\frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right|^q dt\right)^{1/q}\right]\end{aligned} \tag{10}$$

For the biased part,

$$\begin{aligned}\mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\int_0^T \rho_\theta(T-t)\sum_{i=1}^n \sigma_i\mu(t)dt\right] &\le \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|\mathbb{E}_\sigma\left[\left|\sum_{i=1}^n \sigma_i\right|\right] \\ &\le \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|\sqrt{\mathbb{E}_\sigma\left[\left|\sum_{i=1}^n \sigma_i\right|^2\right]} \\ &= \sqrt{n}\sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|\end{aligned} \tag{11}$$

Now for the unbiased part (10), we take $p = 1, q = \infty$. Then we have

$$\begin{aligned}&\mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\int_0^T \rho_\theta(T-t)\sum_{i=1}^n \sigma_i(x_i(t)-\mu(t))dt\right]\\ \leq&\sup_{\theta\in\Theta}\left(\int_0^T |\rho_\theta(T-t)|\sqrt{K(t,t)}dt\right)\mathbb{E}_\sigma\left[\sup_{t\in[0,T]}\left|\sum_{i=1}^n \sigma_i\frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right|\right]\end{aligned} \tag{12}$$

Also by the same argument, note that

$$\begin{aligned}&\sup_{\theta\in\Theta,i\in[1:n]}|g_\theta(x_i)|\\ =&\sup_{\theta\in\Theta,i\in[1:n]}\left|\int_0^T \rho_\theta(T-t)x_i(t)dt - y_i\right|\\ \leq&\sup_{\theta\in\Theta,i\in[1:n]}\left|\int_0^T \rho_\theta(T-t)(x_i(t)-\mu(t))dt\right| + \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right| + 1\\ \leq&\sup_{\theta\in\Theta}\left(\int_0^T |\rho_\theta(T-t)|\sqrt{K(t,t)}dt\right)\sup_{i\in[1:n],t\in[0,T]}\left|\frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right| + \sup_{\theta\in\Theta}\left|\int_0^T \Re(\rho_\theta(T-t))\mu(t)dt\right| + 1\end{aligned} \tag{13}$$

Thus, there are two terms that we need to bound:

$$\sup_{i\in[1:n],t\in[0,T]}\left|\frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right|,\quad \mathbb{E}_\sigma\left[\sup_{t\in[0,T]}\left|\sum_{i=1}^n \sigma_i\frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right|\right]$$

For the first term, notice that the normalized Gaussian process $\frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}$ is centered, and by Assumption 1, it is almost surely finite on $t \in [0, T]$. Therefore, we may apply the Borell–TIS inequality (Lemma 4) by letting $u = \sigma_T\sqrt{2\log(3n/\delta)}$. Then we have for any $\delta \in (0, 1)$, with probability at least $1 - \delta/3n$,

$$\sup_{t\in[0,T]}\left|\frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right| \leq \mathbb{E}_x\left(\sup_{t\in[0,T]}\left|\frac{x(t)-\mu(t)}{\sqrt{K(t,t)}}\right|\right) + \sigma_T\sqrt{2\log(3n/\delta)},\quad \forall i = 1,\ldots,n$$

where $\sigma_T := \sup_{t\in[0,T]}\mathbb{E}_x\left|\frac{x(t)-\mu(t)}{\sqrt{K(t,t)}}\right|^2$ and the expectation $\mathbb{E}_x\left(\sup_{t\in[0,T]}\left|\frac{x(t)-\mu(t)}{\sqrt{K(t,t)}}\right|\right)$ are both finite. Therefore, there exists a constant $\tilde{C}_T$ that depends on the terminal time T such that with probability at least $1 - \delta/3n$,

$$\sup_{t\in[0,T]}\left|\frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right| \leq \tilde{C}_T\sqrt{2\log(3n/\delta)},\quad \forall i = 1,\ldots,n$$

Now by taking a union bound over $i = 1, \ldots, n$, we get with probability at least $1 - \delta/3$,

$$\sup_{i\in[1:n],t\in[0,T]}\left|\frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right| \leq \tilde{C}_T\sqrt{2\log(3n/\delta)} \tag{14}$$

For the second term, by Assumption 1, the normalized Gaussian process is Hölder continuous with order in $(0, H)$, then by applying $a = H/2$ in Lemma 5 we may directly get the Hölder continuity for the normalized Gaussian process, i.e.,

$$\left|\frac{x(t)-\mu(t)}{\sqrt{K(t,t)}} - \frac{x(s)-\mu(s)}{\sqrt{K(s,s)}}\right| \leq L|t-s|^{H/2} \tag{15}$$

for some constant L.

Now we discretize the time interval $[0,T]$ into N parts $[0,T/N] \cup [T/N, 2T/N] \cdots \cup [(N-1)T/N, T]$, then for any $t \in [0,T]$, there exists a sub-interval such that $t \in [(k-1)T/N, kT/N]$ for some $k \in [1:N]$. Therefore, $\forall t \in [0,T]$ such that $t \in [(k-1)T/N, kT/N]$ for some $k \in [1:N]$, by the Hölder continuity (15), we have

$$\begin{aligned}
\left|\sum_{i=1}^n \sigma_i \frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right| &\le \left|\sum_{i=1}^n \sigma_i \frac{x_i\left(\frac{(k-1)T}{N}\right)-\mu\left(\frac{(k-1)T}{N}\right)}{\sqrt{K\left(\frac{(k-1)T}{N},\frac{(k-1)T}{N}\right)}}\right| + \left|\sum_{i=1}^n \sigma_i \left(\frac{x_i\left(\frac{(k-1)T}{N}\right)-\mu\left(\frac{(k-1)T}{N}\right)}{\sqrt{K\left(\frac{(k-1)T}{N},\frac{(k-1)T}{N}\right)}} - \frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right)\right| \\
&\le \max_{k=1,\dots,N}\left|\sum_{i=1}^n \sigma_i \frac{x_i\left(\frac{(k-1)T}{N}\right)-\mu\left(\frac{(k-1)T}{N}\right)}{\sqrt{K\left(\frac{(k-1)T}{N},\frac{(k-1)T}{N}\right)}}\right| + \|\sigma\|\sqrt{n}L\left(\frac{T}{N}\right)^{H/2} \\
&= \max_{k=1,\dots,N}\left|\sum_{i=1}^n \sigma_i \frac{x_i\left(\frac{(k-1)T}{N}\right)-\mu\left(\frac{(k-1)T}{N}\right)}{\sqrt{K\left(\frac{(k-1)T}{N},\frac{(k-1)T}{N}\right)}}\right| + nL\left(\frac{T}{N}\right)^{H/2}
\end{aligned}$$

Then by the Massart Lemma 6 and the sup norm bound (14), with probability at least $1-\delta/3$,

$$\begin{aligned}
\mathbb{E}_\sigma\left[\sup_{t\in[0,T]}\left|\sum_{i=1}^n \sigma_i \frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right|\right] &\le \mathbb{E}_\sigma\left[\max_{k=1,\dots,N}\left|\sum_{i=1}^n \sigma_i \frac{x_i\left(\frac{(k-1)T}{N}\right)-\mu\left(\frac{(k-1)T}{N}\right)}{\sqrt{K\left(\frac{(k-1)T}{N},\frac{(k-1)T}{N}\right)}}\right|\right] + nL\left(\frac{T}{N}\right)^{H/2} \\
&\le \sqrt{2n\log N}\cdot \sup_{i\in[1:n],t\in[0,T]}\left|\frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right| + nL\left(\frac{T}{N}\right)^{H/2} \\
&\le \tilde{C}_T\sqrt{2n\log N}\sqrt{2\log(3n/\delta)} + nL\left(\frac{T}{N}\right)^{H/2}
\end{aligned}$$

Since N is an arbitrary integer number, we let $N = \left[Tn^{1/H}\right]+1$, then there exists another constant C_T such that

$$\mathbb{E}_\sigma\left[\sup_{t\in[0,T]}\left|\sum_{i=1}^n \sigma_i \frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right|\right] \le \widetilde{\mathcal{O}}\left(C_T\sqrt{n\log(n/\delta)}\right) \tag{16}$$

Combining (14), (16), (11) and (12), we can further bound (13) as

$$\sup_{\theta\in\Theta,i\in[1:n]}|g_\theta(x_i)| \le \sup_{\theta\in\Theta}\left(\int_0^T |\rho_\theta(T-t)|\sqrt{K(t,t)}dt\right)\tilde{C}_T\sqrt{2\log(3n/\delta)} + \sup_{\theta\in\Theta}\left|\int_0^T \Re(\rho_\theta(T-t))\mu(t)dt\right| + 1 \tag{17}$$

And the Rademacher complexity is further bounded as

$$\begin{aligned}
&\mathcal{R}_S(\mathcal{F}) \\
\le& \frac{2\sup_{\theta\in\Theta,i\in[1:n]}|g_\theta(x_i)|}{n}\mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\int_0^T \rho_\theta(T-t)\sum_{i=1}^n \sigma_i x_i(t)dt\right] \\
\le& \frac{2\sup_{\theta\in\Theta,i\in[1:n]}|g_\theta(x_i)|}{n}\left(\sup_{\theta\in\Theta}\int_0^T |\rho_\theta(T-t)|\sqrt{K(t,t)}dt + \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|\right)\cdot\widetilde{\mathcal{O}}\left(C_T\sqrt{n\log(n/\delta)}\right) \\
\le& \left(\sup_{\theta\in\Theta}\int_0^T |\rho_\theta(T-t)|\sqrt{K(t,t)}dt + \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right| + 1\right)^2\cdot\widetilde{\mathcal{O}}\left(\bar{C}_T\sqrt{\frac{\log(n/\delta)}{n}}\right)
\end{aligned}$$

for some constant $\bar{C}_T$ that depends on T.

Finally, by the symmetrization of $R_x(\theta) - R_n(\theta)$, combining it with (17) and (1), we have with probability at least $1-\delta$,

$$\sup_{\theta\in\Theta}|R_x(\theta)-R_n(\theta)| \leq \left(\sup_{\theta\in\Theta}\int_0^T |\rho_\theta(T-t)|\sqrt{K(t,t)}dt + \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right| + 1\right)^2 \cdot \widetilde{\mathcal{O}}\left(C_T\sqrt{\frac{\log(n/\delta)}{n}}\right)$$

for some constant C_T that depends on T. □

D.5 Another generalization bound that does not rely on Assumption 1

In this subsection, we will give another generalization bound that does not need to use Assumption 1, but is looser than the original bound (4). The key step is to use the Cauchy Schwarz inequality instead of the Hölder inequality, and then we can use the tools for sub-exponential random variables. We first give the statement for the Theorem.

Theorem 2. *For a SSM $\int_0^T \rho_\theta(T-s)x(s)ds$, following the notations and settings in Section 3.1 & 4.1, given a parameter space Θ for θ, for any $\delta \in (0,1)$, if $n > 8\log(6/\delta)$, then we have with probability at least $1-\delta$ over the training sequences,*

$$\begin{aligned}
&\sup_{\theta\in\Theta}|R_x(\theta)-R_n(\theta)| \\
\leq &\left(\sup_{\theta\in\Theta}\int_0^T T\rho_\theta^2(T-t)K(t,t)dt + \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|^2 + 1\right)\cdot\widetilde{\mathcal{O}}\left(\sqrt{\frac{\log(n/\delta)}{n}}\right)
\end{aligned}$$

We can see that comparing the above bound with (4)

Now we comparing the two generalization measures in Theorem 1 and Theorem 2 for finite time T:

$$\begin{aligned}
\text{Theorem 1}: &\quad \left(\sup_{\theta\in\Theta}\int_0^T |\rho_\theta(T-t)|\sqrt{K(t,t)}dt + \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right| + 1\right)^2 \\
\text{Theorem 2}: &\quad \sup_{\theta\in\Theta}\int_0^T T\rho_\theta^2(T-t)K(t,t)dt + \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|^2 + 1
\end{aligned}$$

Notice that

$$\begin{aligned}
&\left(\sup_{\theta\in\Theta}\int_0^T |\rho_\theta(T-t)|\sqrt{K(t,t)}dt + \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right| + 1\right)^2 \\
\leq &3\left(\sup_{\theta\in\Theta}\left(\int_0^T |\rho_\theta(T-t)|\sqrt{K(t,t)}dt\right)^2 + \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|^2 + 1\right) \\
\leq &3\left(\sup_{\theta\in\Theta}\int_0^T T\rho_\theta^2(T-t)K(t,t)dt + \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|^2 + 1\right)
\end{aligned}$$

In that sense, the generalization measure in Theorem 2 is looser than the one in Theorem 1. Next, we give the proof for Theorem 2.

Proof. The proof follows from the the proof of Theorem 1 until the inequality (12), where we use the Cauchy Schwarz inequality ($p = q = 2$) and then we get

$$\begin{aligned}
&\mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\int_0^T \rho_\theta(T-t)\sum_{i=1}^n \sigma_i(x_i(t)-\mu(t))dt\right] \\
\leq &\sup_{\theta\in\Theta}\sqrt{\int_0^T \rho_\theta^2(T-t)K(t,t)dt}\sqrt{\mathbb{E}_\sigma\int_0^T\left|\sum_{i=1}^n \sigma_i\frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\right|^2 dt} \\
= &\sup_{\theta\in\Theta}\sqrt{\int_0^T \rho_\theta^2(T-t)K(t,t)dt}\sqrt{\sum_{i=1}^n\int_0^T\frac{(x_i(t)-\mu(t))^2}{K(t,t)}dt}
\end{aligned} \tag{18}$$

Also note that

$$\begin{aligned}
&\sup_{\theta\in\Theta,i\in[1:n]}|g_\theta(x_i)| \\
= &\sup_{\theta\in\Theta,i\in[1:n]}\left|\int_0^T \rho_\theta(T-t)x_i(t)dt - y_i\right| \\
\leq &\sup_{\theta\in\Theta,i\in[1:n]}\left|\int_0^T \rho_\theta(T-t)(x_i(t)-\mu(t))dt\right| + \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right| + 1 \\
\leq &\sup_{\theta\in\Theta}\sqrt{\int_0^T \rho_\theta^2(T-t)K(t,t)dt}\sup_{i\in[1:n]}\sqrt{\int_0^T\frac{(x_i(t)-\mu(t))^2}{K(t,t)}dt} + \sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right| + 1
\end{aligned} \tag{19}$$

Thus, there are two terms that we need to bound:

$$\sqrt{\frac{1}{n}\sum_{i=1}^n\int_0^T\frac{(x_i(t)-\mu(t))^2}{K(t,t)}dt}, \quad \sqrt{\sup_{i\in[1:n]}\int_0^T\frac{(x_i(t)-\mu(t))^2}{K(t,t)}dt}$$

For the first term, notice that the covariance function for the normalized Gaussian process $\frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}$ is given by

$$\mathbb{E}\left[\frac{x_i(t)-\mu(t)}{\sqrt{K(t,t)}}\frac{x_i(s)-\mu(s)}{\sqrt{K(s,s)}}\right] = \frac{K(s,t)}{\sqrt{K(t,t)K(s,s)}}$$

By Lemma 3, $\int_0^T\frac{(x_i(t)-\mu(t))^2}{K(t,t)}dt$ is $(4T^2, 4T)$ sub-exponential. By Proposition 4, for $0 < s < T$,

$$P\left(\frac{1}{n}\sum_{i=1}^n\int_0^T\frac{(x_i(t)-\mu(t))^2}{K(t,t)}dt - T \geq s\right) \leq \exp\left(-\frac{ns^2}{8T^2}\right) \tag{20}$$

For the second term $\sup_{i\in[1:n]}\left(\int_0^T\frac{(x_i(t)-\mu(t))^2}{K(t,t)}dt\right)^{1/2}$, since $\int_0^T\frac{(x_i(t)-\mu(t))^2}{K(t,t)}dt$ is $\left(4T^2, 4T\right)$ sub-exponential, then by Proposition 2, for $s > T$,

$$P\left(\int_0^T\frac{(x_i(t)-\mu(t))^2}{K(t,t)}dt - T \geq s\right) \leq \exp\left(-\frac{s}{8T}\right)$$

Taking the union bound over $i \in [1:n]$, we get for $s > T$,

$$P\left(\sup_{i\in[1:n]}\int_0^T\frac{(x_i(t)-\mu(t))^2}{K(t,t)}dt - T \geq s\right) \leq n\exp\left(-\frac{s}{8T}\right) \tag{21}$$

Now for bound (20), we solve $\delta/6 = \exp\left(-\frac{ns^2}{8T^2}\right)$, i.e., $s = T\sqrt{\frac{8\log(6/\delta)}{n}}$, by the precondition that $0 < s < T$, we get for any $\delta \in (0,1)$, if $n > 8\log(6/\delta)$, then w.p.a. $1-\delta/6$,

$$\frac{1}{n}\sum_{i=1}^{n}\int_0^T \frac{(x_i(t)-\mu(t))^2}{K(t,t)}dt \le T\left(1+\sqrt{\frac{8\log(6/\delta)}{n}}\right)$$

For bound (21), we solve $\delta/6 = n\exp\left(-\frac{s}{8T}\right)$, i.e., $s = 8T\log(6n/\delta)$. Since $8\log(6n/\delta) > 8\log 6 > 1$, which guarantees the precondition that $s > T$, hence for any $\delta \in (0,1)$, w.p.a. $1-\delta/6$,

$$\sup_{i\in[1:n]}\int_0^T \frac{(x_i(t)-\mu(t))^2}{K(t,t)}dt \le T\left(1+8\log(6n/\delta)\right)$$

Therefore, for any $\delta \in (0,1)$, if $n > 8\log(6/\delta)$, then w.p.a. $1-\delta/3$,

$$\sup_{\theta\in\Theta, i\in[1:n]}|g_\theta(x_i)| \le \sup_{\theta\in\Theta}\sqrt{\int_0^T \rho_\theta^2(T-t)K(t,t)dt}\sqrt{T\left(1+8\log\frac{6n}{\delta}\right)}+\sup_{\theta\in\Theta}\left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|+1$$

Hence, by combining the two parts (18) & (11), we can get

$$\begin{aligned}
&\mathcal{R}_S(\mathcal{F})\\
\le&\frac{2\sup_{\theta\in\Theta,i\in[1:n]}|g_\theta(x_i)|}{n}\left(\sup_{\theta\in\Theta}\sqrt{\textstyle\int_0^T \rho_\theta^2(T-t)K(t,t)dt}\sqrt{\textstyle\sum_{i=1}^n\int_0^T\frac{(x_i(t)-\mu(t))^2}{K(t,t)}dt}+\sqrt{n}\sup_{\theta\in\Theta}\left|\textstyle\int_0^T\rho_\theta(T-t)\mu(t)dt\right|\right)\\
\le&\frac{\sup_{\theta\in\Theta}\sqrt{T\int_0^T\rho_\theta^2(T-t)K(t,t)dt}+\sup_{\theta\in\Theta}\left|\int_0^T\rho_\theta(T-t)\mu(t)dt\right|+1}{n}\\
&\left(\sup_{\theta\in\Theta}\sqrt{T\int_0^T\rho_\theta^2(T-t)K(t,t)dt}+\sup_{\theta\in\Theta}\left|\int_0^T\rho_\theta(T-t)\mu(t)dt\right|\right)\cdot\widetilde{\mathcal{O}}\left(\sqrt{n\log(n/\delta)}\right)\\
\le&\left(\sup_{\theta\in\Theta}\int_0^T T\rho_\theta^2(T-t)K(t,t)dt+\sup_{\theta\in\Theta}\left|\int_0^T\Re(\rho_\theta(T-t))\mu(t)dt\right|^2+1\right)\cdot\widetilde{\mathcal{O}}\left(\sqrt{\frac{\log(n/\delta)}{n}}\right)
\end{aligned}$$

Finally, by the symmetrization of the generalization gap and Lemma 1, we have

$$\begin{aligned}
&\sup_{\theta\in\Theta}|R_x(\theta)-R_n(\theta)|\\
\le&\left(\sup_{\theta\in\Theta}\int_0^T T\rho_\theta^2(T-t)K(t,t)dt+\sup_{\theta\in\Theta}\left|\int_0^T\rho_\theta(T-t)\mu(t)dt\right|^2+1\right)\cdot\widetilde{\mathcal{O}}\left(\sqrt{\frac{\log(n/\delta)}{n}}\right)
\end{aligned}$$

□

E PROOF FOR PROPOSITION 1

Proof. First, notice that by the Hölder's inequality with $p = 1, q = \infty$, we have

$$
\begin{aligned}
&\mathbb{E}_x\left[\left|\int_0^T \rho_{\tilde{\theta}}(T-t)x(t)dt\right|\right] \\
=&\frac{\mathbb{E}_x\left[\left|\int_0^T \rho_\theta(T-t)x(t)dt\right|\right]}{\int_0^T |\rho_\theta(T-t)|\sqrt{K(t,t)}dt + \left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|} \\
\leq&\frac{\mathbb{E}_x\left[\left|\int_0^T \rho_\theta(T-t)(x(t)-\mu(t))dt\right|\right] + \left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|}{\int_0^T |\rho_\theta(T-t)|\sqrt{K(t,t)}dt + \left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|} \\
\leq&\frac{\int_0^T |\rho_\theta(T-t)|\sqrt{K(t,t)}dt \cdot \mathbb{E}_x\left[\sup_{t\in[0,T]}\left|\frac{x(t)-\mu(t)}{\sqrt{K(t,t)}}\right|\right] + \left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|}{\int_0^T |\rho_\theta(T-t)|\sqrt{K(t,t)}dt + \left|\int_0^T \rho_\theta(T-t)\mu(t)dt\right|} \\
\leq&\mathbb{E}_x\left[\sup_{t\in[0,T]}\left|\frac{x(t)-\mu(t)}{\sqrt{K(t,t)}}\right|\right] + 1
\end{aligned}
$$

Under Assumption 1, by the Borell–TIS inequality (Lemma 4), the expectation $\mathbb{E}_x\left[\sup_{t\in[0,T]}\left|\frac{x(t)-\mu(t)}{\sqrt{K(t,t)}}\right|\right]$ is finite. Therefore, there exists a constant β_T that depends on T such that

$$
\mathbb{E}_x\left[\left|\int_0^T \rho_{\tilde{\theta}}(T-t)x(t)dt\right|\right] \leq \beta_T
$$

□