PAC-Bayes Generalisation Bounds for Dynamical Systems Including Stable RNNs

Eringis, Deividas; Leth, John-Josef; Tan, Zheng-Hua; Wisniewski, Rafal; Petreczky, Mihaly

Published in:
Proceedings of the AAAI Conference on Artificial Intelligence

DOI (link to publication from Publisher):
10.1609/aaai.v38i11.29076

Publication date:
2024

Document Version
Accepted author manuscript, peer reviewed version

PAC-Bayes Generalisation Bounds for Dynamical Systems including Stable RNNs

Deividas Eringis[1], **John Leth**[1], **Zheng-Hua Tan**[1], **Rafal Wisniewski**[1], **Mihály Petreczky**[2]

[1]Department of Electronic Systems, Aalborg University, Denmark
[2] Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France
{der,jjl,zt,raf}@es.aau.dk, mihaly.petreczky@centralelille.fr

Abstract

In this paper, we derive a PAC-Bayes bound on the generalisation gap, in a supervised time-series setting for a special class of discrete-time non-linear dynamical systems. This class includes stable recurrent neural networks (RNN), and the motivation for this work was its application to RNNs. In order to achieve these results, we impose some stability constraints, on the allowed models. Here, stability is understood in the sense of dynamical systems. For RNNs, these stability conditions can be expressed in terms of conditions on the weights. We assume the processes involved are essentially bounded and the loss functions are Lipschitz. The proposed bound on the generalisation gap depends on the mixing coefficient of the data distribution, and the essential supremum of the data. Furthermore, the bound converges to zero as the dataset's size increases. In this paper, we 1) formalise the learning problem, 2) derive a PAC-Bayesian error bound for such systems, 3) discuss various consequences of this error bound, and 4) show an illustrative example, with discussions on computing the proposed bound. Unlike other available bounds the derived bound holds for non i.i.d. data (time-series) and it does not grow with the number of steps of the RNN.

Introduction

The Probably Approximately Correct (PAC)-Bayesian framework has been a popular tool for obtaining generalisation bounds and to derive efficient learning algorithms, see (Alquier 2021; Dziugaite and Roy 2017).

Contribution. In this paper we develop PAC-Bayesian inequalities for a class of discrete-time dynamical systems with hidden (unobserved) states. This class includes a wide variety of dynamical systems, ranging from linear time-invariant state-space representations (LTIs) to recurrent neural networks (RNNs). We view dynamical systems as hypotheses (predictors) which transform past inputs and outputs (labels) to estimates of the current output (label). That is, our framework captures both time-series forecasting and learning models which causally transform sequences of inputs to sequences of outputs. In the latter case, the dynamical system at hand uses only past inputs. Furthermore, training data represents a single time-series sampled from the input and output processes. That is, the training data is not i.i.d.

The PAC-Bayesian inequality of this paper proposes a bound on the difference between the generalisation loss and empirical loss. This bound holds with high probability, and it depends on the number of data points N and on the parameter (learning rate) λ. Moreover, for a suitable choice of λ the bound converges to zero at the rate $\mathcal{O}(1/\sqrt{N})$. The latter rate is consistent with most of finite-sample bounds available in the literature for various types of models.

In order to consider non i.i.d. data we assumed that inputs and outputs are bounded and they are weakly dependent. The latter represents a type of mixing condition. Moreover, we had to restrict attention to dynamical systems which transform bounded and weakly dependent inputs to outputs having the same properties. To this end, we required that the dynamical systems satisfy the *exponential convergence* (Pavlov and van de Wouw 2012) property. The latter concept originates from control theory.

Motivation. PAC and PAC-Bayesian bounds are a major tool for analysing learning algorithms. Moreover, by minimizing the error bound, new, theoretically well-founded learning algorithms can be formulated. In particular, PAC-Bayesian error bounds turned out to be useful for providing non-vacuous error bounds for neural networks (Dziugaite and Roy 2017). While there is a wealth of literature on PAC (Shalev-Shwartz and Ben-David 2014) and PAC-Bayesian (Alquier 2021; Guedj 2019) bounds, for static models, much less is known on dynamical systems.

Generalization bounds for RNNs. PAC bounds for RNN were developed in (Koiran and Sontag 1998; Sontag 1998; Chen, Li, and Zhao 2020) using VC dimension, in (Wei and Ma 2019; Akpinar, Kratzwald, and Feuerriegel 2020; Joukovsky et al. 2021; Chen, Li, and Zhao 2020) using Rademacher complexity, and in (Zhang, Lei, and Dhillon 2018) using PAC-Bayesian approach. All the cited papers assume noiseless models, a fixed number of time-steps, that the training data are i.i.d sampled time-series, and the signals are bounded. In contrast, we consider (1) noisy models, (2) generalisation loss defined on infinite time horizon, (3) only one single time series available for training data.

Our contribution is consistent with recent results for linear RNNs (Emami et al. 2021; Cohen-Karlik et al. 2023), on their ability to extrapolate to longer sequences by training on short sequences with stochastic gradient descent. Thus we provide a different perspective, while extending to a more

general class of models.

PAC and PAC-Bayesian bounds for autoregressive and linear models. In (Alquier and Wintenberger 2012; Alquier, Li, and Wintenberger 2013) auto-regressive models without exogenous inputs were considered, and the variables were either assumed to be bounded or the loss function was assumed to be Lipschitz. In contrast, we consider non-linear state-space models with inputs. That is, the learning problem considered in this paper is different from that of (Alquier and Wintenberger 2012; Alquier, Li, and Wintenberger 2013). However, we were inspired by (Alquier and Wintenberger 2012; Alquier, Li, and Wintenberger 2013). In (Eringis et al. 2021, 2023b) PAC-Bayesian bounds for linear state-space models were derived. In contrast, in this paper we derive bounds for non-linear state-space models. Moreover, the bound of (Eringis et al. 2021, 2022) does not converge to zero for systems with unbounded noise and inputs. The result of (Eringis et al. 2023b) for systems with bounded noise and inputs, where the PAC-Bayesian bound converges to zero, is a special case of the result of the present paper.

Finite-sample bounds for system identification. Guarantees for asymptotic convergence of learning algorithms are a classical topic in system identification (Ljung 1999). Recently, several publications on finite-sample bounds for learning dynamical systems were derived, without claiming completeness (Simchowitz et al. 2018; Simchowitz, Boczar, and Recht 2019; Simchowitz 2021; Oymak and Ozay 2022; Lale et al. 2020; Foster, Sarkar, and Rakhlin 2020; Hazan et al. 2018; Tsiamis and Pappas 2019; Sarkar, Rakhlin, and Dahleh 2021). First, all the cited papers propose a bound which is valid only for models generated by a specific learning algorithm. In particular, these bounds do not characterise the generalisation gap for arbitrary models, i.e., they are not PAC(-Bayesian) bounds. Second, many of the cited papers do not derive bounds on the infinite horizon prediction error. More precisely, (Oymak and Ozay 2022; Sarkar, Rakhlin, and Dahleh 2021; Lale et al. 2020; Tsiamis and Pappas 2019; Simchowitz and Foster 2020) provided error bounds for the difference of the first T Markov-parameters of the estimated and true system for a specific identification algorithm. However, in order to characterise the infinite horizon prediction error, we need to take $T = \infty$. For $T = \infty$ the cited bounds become infinite, i.e., vacuous. Error bounds for certain classes of non-linear dynamical systems were also derived in (Sattar, Oymak, and Ozay 2022; Sattar and Oymak 2022; Blanke and Lelarge 2023; Foster, Sarkar, and Rakhlin 2020; Mania, Jordan, and Recht 2022; Sayedana et al. 2022; Shi, Mazhar, and De Schutter 2022; Roy, Balasubramanian, and Erdogdu 2021; Ziemann, Sandberg, and Matni 2022; Ziemann and Tu 2022; Li et al. 2023), but they assume full state observation and they provide an error bound for a specific learning algorithm. In contrast, we consider models with unobserved (hidden) states.

Outline.We start off the paper with informally presenting the main result. We then define the class of dynamical systems which plays the role of the hypotheses class. In the following section we define the learning problem formally. Finally, we state the main results of the paper, where the PAC-Bayesian bound on the generalisation gap of dynamical systems is stated. At the end, we present a short illustrative numerical example. The detailed proofs can be found in (Eringis et al. 2023a, Appendix).

Notation. Note that unless otherwise defined, this paper will follow the notation defined by Goodfellow et al. (2016), i.e. $\boldsymbol{x}$ is a vector, and $\mathbf{x}$ is a random vector. Let $\mathbf{F}$ denote a σ-algebra on the set Ω and $\mathbf{P}$ be a probability measure on $\mathbf{F}$. Unless otherwise stated all probabilistic considerations will be with respect to the probability space $(\Omega, \mathbf{F}, \mathbf{P})$, and we let $\mathbb{E}(\mathbf{z})$ denote expectation of the stochastic variable $\mathbf{z} : \Omega \to \mathbb{R}^{n_{\mathbf{z}}}$. We shall denote the realisation of a stochastic variable $\mathbf{z}$ as $\mathbf{z}(\omega)$, with $\omega \in \Omega$. Each Euclidean space is associated with the topology generated by the 2-norm $\|\cdot\|_2$, and the Borel σ-algebra generated by the open sets. The induced matrix 2-norm is also denoted $\|\cdot\|_2$. We use $\triangleq$ to denote "defined by", and $\overset{\text{a.s.}}{=}$ to denote that the equality holds almost surely with respect to some underlying probability measure.

Problem Formulation and PAC-Bayesian Setting

In this paper we will consider time-series supervised learning problem. The goal will be to optimise a posterior distribution defined over some set of predictors (hypotheses) $\mathcal{H}$. To this end we assume that we have only one sequence of training data. Let us fix *bounded* stochastic processes $\mathbf{y}(t) \in \mathbb{Y} \subset \mathbb{R}^{n_y}, \mathbf{x}(t) \in \mathbb{X} \subset \mathbb{R}^{n_x}$ that share the time-axis $t \in \mathbb{Z}$, i.e. $\mathbf{y}(t), \mathbf{x}(t)$ are random vectors on $(\Omega, \mathbf{F}, \mathbf{P})$. The goal of each predictor $h \in \mathcal{H}$ is to estimate $\mathbf{y}(t)$ based on current and past values $\{\mathbf{x}(s)\}_{s=0}^{t}$ of $\mathbf{x}$. Formally, we can think of h as a function $h : \bigcup_{k=1}^{\infty} \mathbb{X}^k \to \mathbb{Y}$, such that the prediction $\hat{\mathbf{y}}(t)$ generated by h satisfies $\hat{\mathbf{y}}(t) = h(\{\mathbf{x}(s)\}_{s=0}^{t})$.

We allow the process $\mathbf{x}$ to contain $\mathbf{y}$ as a component, i.e., $\mathbf{x} = \begin{bmatrix}\bar{\mathbf{x}}^T & \mathbf{y}^T\end{bmatrix}^T$. In this case, the predictor uses past values of $\mathbf{y}$ to predict the current one, i.e., the predictor is *autoregressive*. In particular, in this case, for the learning problem to be meaningful, the class of predictors $\mathcal{H}$ should be such that $h(\{\begin{bmatrix}\bar{\mathbf{x}}^T(s) & \mathbf{y}^T(s)\end{bmatrix}^T\}_{s=0}^{t})$ does not depend on the value $\mathbf{y}(t)$ of $\mathbf{y}$ at time instant t.

The training data is a single sample $\mathcal{D} = \{\mathbf{y}(s)(\omega), \mathbf{x}(s)(\omega)\}_{s=0}^{N-1}$, for some $\omega \in \Omega$, from the random variables $\{\mathbf{y}(s), \mathbf{x}(s)\}_{s=0}^{N-1}$. We are interested in the *empirical loss*

$$\hat{\mathcal{L}}_N(h) \triangleq \frac{1}{N} \sum_{t=0}^{N-1} \ell(\boldsymbol{y}(t), h(\{\mathbf{x}(s)\}_{s=0}^{t})) \tag{1}$$

and *generalisation loss*

$$\mathcal{L}(h) \triangleq \lim_{t\to\infty} \mathbb{E}[\ell(\mathbf{y}(t), h(\{\mathbf{x}(s)\}_{s=0}^{t}))] \tag{2}$$

for some loss function $\ell : \mathbb{Y} \times \mathbb{Y} \to \mathbb{R}_+$. Note that, classically generalisation loss is defined simply as $\mathbb{E}[\ell(\mathbf{y}(t), h(\{\mathbf{x}(s)\}_{s=0}^{t}))]$. However in the time-series setting, this would depend on the time t, and as such would not give the desired intuition. Indeed, in time series prediction, the predictions are updated as new data points become

available, so the relevant metric of generalisation power is the prediction error as $t \to +\infty$.

For now, we assume that the limit defining the generalisation loss $\mathcal{L}(h)$ exists for any $h \in \mathcal{H}$. Later we will define the class $\mathcal{H}$ in such a manner that this assumption holds.

With this in mind, the goal of the PAC-Bayesian framework is to analyse the generalisation gap $\Delta_N(h) \triangleq d(E_{h\sim\rho}\mathcal{L}(h), E_{h\sim\rho}\hat{\mathcal{L}}_N(h))$, where d is any convex function, and $E_{h\sim\rho}\mathcal{L}(h)$ and $E_{h\sim\rho}\hat{\mathcal{L}}_N(h)$ denote the expectation of $\mathcal{L}(h)$ and $\hat{\mathcal{L}}_N(h)$, when h is randomly sampled from a probability distribution on $\mathcal{H}$ with the density ρ.
In this paper we look at the special case of $\Delta_N(h) \triangleq E_{h\sim\rho}\mathcal{L}(h) - E_{h\sim\rho}\hat{\mathcal{L}}_N(h)$, since bounding $\Delta_N(h)$ will provide immediate bounds on $E_{h\sim\rho}\mathcal{L}(h)$. Furthermore we consider predictors h realised by dynamical systems:

$$\begin{aligned} \mathbf{s}(t+1) &= f(\mathbf{s}(t), \mathbf{x}(t)), \quad \mathbf{s}(0) = \mathbf{s}_0, \\ h(\{\mathbf{x}(s)\}_{s=0}^{t}) &= g(\mathbf{s}(t), \mathbf{x}(t)), \end{aligned} \tag{3}$$

where $\mathbf{s}(t) \in \mathbb{R}^{n_\mathbf{s}}$ is the hidden state. Note that recurrent neural networks (RNNs) represent a special case of the dynamical systems of the form of equation 3. Under suitable assumptions, which will be discussed in later assumptions section we derive the following *Catoni-like PAC-Bayesian inequality*. Informally, let $\mathcal{H}$ be a family of predictors which can be realised by a dynamical system of the form equation 3. Moreover, informally, let π be a probability density on $\mathcal{H}$ and let $\mathcal{M}_\pi$ denote the set of all probability densities for which the corresponding probability measures are absolutely continuous w.r.t. to the probability measure defined by π. Formal details will be given in assumptions section.

Theorem 1 (Informal theorem)**.** *There exist constants G_1 and G_2, which depend on the class of predictors $\mathcal{H}$, such that for all $\lambda > 0$ and for any $\delta \in (0, 1/2)$, the following holds with probability at least $1 - 2\delta$ on the data*

$$\begin{aligned} \forall \hat{\rho} \in \mathcal{M}_\pi : E_{h\sim\hat{\rho}}\mathcal{L}(h) &\leq E_{h\sim\hat{\rho}}\hat{\mathcal{L}}_N(h) \\ &+ \frac{1}{\lambda}\left[D_{\mathrm{KL}}(\hat{\rho}\|\pi) + \ln\frac{1}{\delta} + \frac{\lambda^2}{N}G_1 + \frac{\lambda}{N}G_2\right] \end{aligned} \tag{4}$$

where $D_{\mathrm{KL}}(\hat{\rho}\|\pi)$ denotes the Kullback–Leibler (KL) divergence between $\hat{\rho}$ and π.

We refer to π as the prior and to $\hat{\rho}$ as the posterior density on the space of predictors. However, there is no relation between these densities except for the condition that the probability measure of $\hat{\rho}$ has to be absolutely continuous w.r.t. the probability measure of π. The bound in equation 4 suggests that for learning we could choose a posterior $\hat{\rho}$ which minimises the right-hand side of equation 4. Then we can either randomly sample a model from that posterior or choose the model with the highest likelihood (Alquier 2021). The posterior which minimises the right-hand side of equation 4 is known as the *Gibbs posterior* (Alquier, Ridgway, and Chopin 2016), which is defined by $\rho_N(h) \triangleq Z^{-1}\pi(h)e^{-\lambda\hat{\mathcal{L}}_N(h)}$, $Z = E_{\theta\sim\pi}[e^{-\lambda\hat{\mathcal{L}}_N(h)}]$. In particular, the model which maximises $\rho_N(h)$ is the one which minimises the regularised empirical loss $\hat{\mathcal{L}}_N(h) - \frac{1}{\lambda}\ln(\pi(h))$. This allows us to use the bound for deriving learning algorithms, similarly to Catoni-like bounds, and interpret the prior π as a regularisation term added to the empirical loss functions.

Remark 1 (Asymptotic properties: $O(1/\sqrt{N})$ bound)**.** *The derived bound has similar asymptotic properties to classical Catoni-like bounds. In particular, if we choose λ to be of order $\mathcal{O}\left(\sqrt{N}\right)$, then the bound $r_N(\lambda, \delta, \hat{\rho}) \triangleq \frac{1}{\lambda}\left[D_{\mathrm{KL}}(\hat{\rho}\|\pi) + \ln\frac{1}{\delta} + \frac{\lambda^2}{N}G_1 + \frac{\lambda}{N}G_2\right]$ converges to zero with the rate $O(1/\sqrt{N})$ for each fixed δ and posterior $\hat{\rho}$. This implies existence of $N^* > 0$, where the proposed bound is non-vacuous for all $N > N^*$.*

Remark 2 (Intuition behind the constants)**.** *Intuitively, the bounds G_1 and G_2 are increasing functions of θ-mixing coefficients of the data, and certain quantities which are related to robustness of the predictors. In particular, the more independent the data is, and the more robust the predictors are, the smaller the bound is.*

Proof strategy The proof relies on PAC-Bayes inequalities obtained by applying Donsker-Varadhan change of measure, in the form of the following lemma.

Lemma 1 (Theorem 3 of Germain et al. (2016))**.** *For any measurable functions X, Y on $\mathcal{H}$, any $\delta \in (0, 1]$, and $\lambda > 0$ the following holds with probability at least $1 - \delta$*

$$\begin{aligned} \forall \hat{\rho} \in \mathcal{M}_\pi :\ & E_{h\sim\hat{\rho}}(X(h) - Y(h)) \\ &\leq \frac{1}{\lambda}\left[D_{\mathrm{KL}}(\hat{\rho}\|\pi) + \ln\frac{1}{\delta} + \Psi_\pi(\lambda, N)\right], \end{aligned} \tag{5}$$

with $\Psi_\pi(\lambda, N) = \ln E_{h\sim\pi}\mathbb{E}[e^{\lambda(X(h)-Y(h))}]$.

We first apply Lemma 1 by choosing $Y(h) = \hat{\mathcal{L}}_N(h)$ to be the empirical loss and

$$X(h) = V_N(h) \triangleq \lim_{\bar{s}\to-\infty} \frac{1}{N}\sum_{t=0}^{N-1} \ell(\mathbf{y}(t), h(\{\mathbf{x}(s)\}_{s=\bar{s}}^{t}))$$

to be the infinite horizon loss, a version of the empirical loss without the transient caused by the initial state. We bound the moment generating function $\mathbb{E}[e^{\lambda(\hat{\mathcal{L}}_N(h) - V_N(h))}]$, by assuming that the dynamical systems realising the predictors have the exponential convergence property (see Def. 2), and using (Alquier and Wintenberger 2012, Proposition 4.2) for weakly dependent processes. In this way we obtain a first PAC-Bayesian inequality in the form of equation 4, which covers the issue of choosing the initial state of the predictor.

Next, we apply Lemma 1 such that $Y(h) = V_N(h)$ is the infinite horizon empirical loss and $X(h) = \mathcal{L}(h)$ is the generalisation loss. Since $\mathbb{E}[V_N(h)] = \mathcal{L}(h)$, we can bound the moment generating function $\mathbb{E}[e^{\lambda(\mathcal{L}(h) - V_N(h))}]$ by using a generalisation of Hoefding's lemma (Alquier and Wintenberger 2012, Theorem 6.6), (Rio 2000) . We can apply this lemma since the difference between the true and predicted labels is weakly dependent. The latter is true, since the system which generates this difference (Eringis et al. 2023a, Lemma A.7) has the property that its outputs are weakly dependent.

We can apply union bound on the two obtained PAC-Bayesian inequalities, resulting in the bound of Theorem 1. In fact, we derive more general theorem, whose corollary is Theorem 1. For full details see (Eringis et al. 2023a, Appendix).

Class of Dynamical Systems

In this paper we will consider predictors realised by a special class of dynamical systems. Furthermore, later we assume that the data is an output of such a system. In this section, we define this class of systems. To this end, let us consider a generalisation of the dynamical system in equation 3:

$$S\begin{cases}\boldsymbol{s}(t+1) = f(\boldsymbol{s}(t), \boldsymbol{v}(t)), & \text{(6a)}\\ \quad \boldsymbol{y}(t) = g(\boldsymbol{s}(t), \boldsymbol{v}(t)), & \text{(6b)}\end{cases}$$

where $\boldsymbol{v} : \mathbb{T} \to \mathbb{V}$ is an arbitrary input trajectory (e.g., $\mathbf{x}(t)$ as in equation 3), $\boldsymbol{s} : \mathbb{T} \to \mathbb{S}$ is the state trajectory, $\boldsymbol{y} : \mathbb{T} \to \mathbb{Y}$ is the output trajectory, and $\mathbb{T} \subseteq \mathbb{Z}$ is an interval in $\mathbb{Z}$ which has no upper bound. That is, either $\mathbb{T}$ is the whole $\mathbb{Z}$ or $\mathbb{T}$ is an interval $[t_0, +\infty) \cap \mathbb{Z}$ for some t_0. Moreover, the sets $\mathbb{S} \subset \mathbb{R}^n$, $\mathbb{V} \subset \mathbb{R}^m$, and $\mathbb{Y} \subset \mathbb{R}^p$ are bounded. If $\mathbb{T} = [t_0, +\infty) \cap \mathbb{Z}$, then $\boldsymbol{s}$ is uniquely determined by the initial state $\boldsymbol{s}(t_0) = \boldsymbol{s}_0$ and the input $\boldsymbol{v}$, and we write $\boldsymbol{s}(t) = \boldsymbol{s}_S(\boldsymbol{s}_0, t_0, \boldsymbol{v}; t)$ to emphasise the dependence on the initial state $\boldsymbol{s}(t_0) = \boldsymbol{s}_0$ and input $\boldsymbol{v}$. Moreover, if $\boldsymbol{v}$ is defined on the whole time axis $\mathbb{Z}$, then $\boldsymbol{s}_S(\boldsymbol{s}_0, t_0, \boldsymbol{v}; t)$ will be understood as the solution corresponding to the restriction of $\boldsymbol{v}$ to $\mathbb{T}$. When S is clear from the context, we use $\boldsymbol{s}(\boldsymbol{s}_0, t_0, \boldsymbol{v}; t)$ instead of $\boldsymbol{s}_S(\boldsymbol{s}_0, t_0, \boldsymbol{v}; t)$.

Likewise, we will use $\boldsymbol{y}_S(\boldsymbol{s}_0, t_0, \boldsymbol{v}; t)$ to denote the output $\boldsymbol{y}(t)$ which corresponds to the state trajectory $\boldsymbol{s}(t) = \boldsymbol{s}_S(\boldsymbol{s}_0, t_0, \boldsymbol{v}; t)$, i.e., $\boldsymbol{y}_S(\boldsymbol{s}_0, t_0, \boldsymbol{v}; t) = g(\boldsymbol{s}_S(\boldsymbol{s}_0, t_0, \boldsymbol{v}; t), \boldsymbol{v}(t))$. As before, we drop the subscript S if it is clear from the context.

In addition, we will *identify systems of the form (6) with the tuple* $S = (\mathbb{S}, \mathbb{Y}, \mathbb{V}, f, g)$.

For systems (6) to be suitable for time-series prediction, they should be robust. In other words, they should be capable of withstanding small perturbations in inputs and initial states without leading to significant changes in the state and output trajectories over time. Without robustness, even minor numerical rounding errors, when implemented, can accumulate over time, resulting in increasingly inaccurate predictions. Note that this is not much of an issue when the prediction is a static function of a finite number of past inputs (auto-regression, systems with fully observed state), or the length of the time-series used for prediction is small. However, it becomes an issue when, as it is often the case in time-series prediction, we use an increasing number of data points in prediction, as more of them become available. Additionally, it is desirable for predictors to preserve stationarity and mixing properties of the inputs. Predictors are expected to generate one-step ahead approximations of the input process. If they cannot even preserve such simple properties, then there is little hope for them being accurate. Both of these properties can be guaranteed by requiring certain stability properties to be defined below. To this end, we recall the following

Definition 1 (UEC and steady-state state and output trajectories (Pavlov and van de Wouw 2012))**.** *The system S from equation 6 is called uniformly exponentially convergent (UEC) with constants C and $\tau \in [0, 1)$, if for each bounded $\boldsymbol{v} : \mathbb{Z} \to \mathbb{V}$, there exists a unique bounded state trajectory $\boldsymbol{s} = \boldsymbol{s}_{S,\boldsymbol{v}} : \mathbb{Z} \to \mathbb{S}$ of S, and for any initial state $\boldsymbol{s}_0$*

$$\|\boldsymbol{s}_S(\boldsymbol{s}_0, t_0, \boldsymbol{v}; t) - \boldsymbol{s}_{S,\boldsymbol{v}}(t)\|_2 \leq C\tau^{t-t_0}\|\boldsymbol{s}_0 - \boldsymbol{s}_{S,\boldsymbol{v}}(t_0)\|_2$$

We refer to $\mathbf{s}_{S,\boldsymbol{v}}$ as the steady-state state trajectory *of S associated with $\boldsymbol{v}$. We call $\mathbf{y}_{S,\boldsymbol{v}} : \mathbb{Z} \ni t \mapsto g(\mathbf{s}_{S,\boldsymbol{v}}(t), \boldsymbol{v}(t))$ the* steady-state output trajectory *associated with $\boldsymbol{v}$.*

UEC guarantees robustness to perturbations in the initial state, and existence of steady-state trajectories for every input. The latter represent the asymptotic behavior of the system as $t \to +\infty$ for the given input, and they can be viewed as the state trajectory starting at $t_0 = -\infty$, with arbitrary initial state, see (Eringis et al. 2023a, Lemma A.4). Although UEC is stronger than other stability notions, many systems used in practice have this property (Pavlov and van de Wouw 2012). Next, we define the class of systems which has all the desired properties mentioned above.

Definition 2 (Class $\mathcal{S}$ system)**.** *We will say that the system from equation 6 is a class $\mathcal{S}$ system, with associated constants $C \geq 1, \tau \in [0, 1), L_{\mathbf{v}} > 0, L_{g,\boldsymbol{s}} > 0$, and $L_{g,\mathbf{v}} > 0$ if the following holds:*

***(UEC)** S is UEC with constants C and τ*

***(Exponential robustness in inputs)** For any two bounded input trajectories $\boldsymbol{v}_1, \boldsymbol{v}_2 : \mathbb{Z} \to \mathbb{V}$,*

$$\|\boldsymbol{s}_{\boldsymbol{v}_1}(t) - \boldsymbol{s}_{\boldsymbol{v}_2}(t)\|_2 \leq L_{\boldsymbol{v}} \sum_{k=1}^{\infty} \tau^{k-1}\|\boldsymbol{v}_1(t-k) - \boldsymbol{v}_2(t-k)\|_2 \tag{7}$$

***(Lipschitz output)** The output function g has Lipschitz constants $L_{g,\boldsymbol{s}}, L_{g,\boldsymbol{v}} > 0$, i.e., $\|g(\xi_1, v_1) - g(\xi_2, v_2)\|_2 \leq L_{g,\boldsymbol{s}}\|\xi_1 - \xi_2\|_2 + L_{g,\boldsymbol{v}}\|v_1 - v_2\|_2$, for all $\xi_1, \xi_2 \in \mathbb{S}$, $v_1, v_2 \in \mathbb{V}$.*

UEC implies robustness with respect to perturbations in the initial state. Exponential robustness in inputs implies that state is robust w.r.t. perturbations in the inputs. In fact, the effect of an instantaneous perturbations in the input decays exponentially fast. The requirement that the output function is Lipschitz ensures that the output trajectories inherit the favorable robustness properties of the state trajectories.

Remark 3 (Role of constants)**.** *The smaller the constants C, τ, $L_{\boldsymbol{v}}$ and $L_{q,\boldsymbol{s}}, L_{g,\boldsymbol{v}}$ are, the more robust the system is to perturbations in inputs and states. Among these constants, τ is the most significant one, as it determines how fast the effect of perturbations will decay with time.*

In addition, outputs of $\mathcal{S}$ systems generated by mixing and stationary inputs are also mixing and stationary, see (Eringis et al. 2023a, Lemma A.4). Some remarks are in order.

Contractive systems are of class $\mathcal{S}$. By (Eringis et al. 2023a, Lemma A.6), a sufficient condition for belonging to class $\mathcal{S}$ is that f, g are Lipschitz, and f is a contraction in its first argument, i.e. $\|f(s, v) - f(s', v)\|_2 < \tau\|s - s'\|_2$, with $\tau \in (0, 1)$.

Examples of RNNs which are of class $\mathcal{S}$. In particular, for RNNs, f being contraction boils down to the activation functions being Lipschitz and to the condition $\mathrm{Lip}(\sigma_f)\|A\|_2 < 1$, see Table 1 for the corresponding constants. Note that commonly used activation functions (ReLu, tanh, sigmoid, linear, etc.,) are Lipschitz. Table 1 shows explicitly how to compute the necessary constants for RNNs.

A sufficient condition for $\mathrm{Lip}(\sigma_f)\|A\|_2 < 1$ to hold is that the absolute values of all the entries of A are smaller than $(\mathrm{Lip}(\sigma_f)n)^{-1}$. Note that many trained and used models have small weights, e.g., (Woo et al. 2021). Furthermore, weight regularisation, which tends to lower the norm of the weights, is commonly used in learning.

Systems of class $\mathcal{S}$ which are not contractions A sufficient condition for a system to be of class $\mathcal{S}$ is that f and g are Lipschitz, and that there exist a quadratic Lyapunov function, see conditions 1 and 2 of (Eringis et al. 2023a, Lemma A.6). The latter may hold even if f is not a contraction. For instance, stable linear state-space systems, i.e., systems of the form $f(\mathbf{s},\mathbf{x}) = A\mathbf{s}+B\mathbf{x}$ and $g(\mathbf{s},\mathbf{x}) = C\mathbf{s}+D\mathbf{x}$ for suitable matrices A, B, C, D such that the spectral radius of A is smaller than 1, satisfy (Eringis et al. 2023a, Lemma A.6), but they are not contractions.

For piecewise-affine functions f (e.g., RNNs with ReLu activation) the conditions of (Eringis et al. 2023a, Lemma A.6) can be checked using Linear Matrix Inequalities (LMI), see (Pavlov and van de Wouw 2012, Theorem 2).

Interconnection of systems, multi-layer RNNs Class $\mathcal{S}$ property is preserved under interconnection: series interconnection of two class $\mathcal{S}$ systems is also $\mathcal{S}$ (see (Eringis et al. 2023a, Lemma A.8)). Furthermore, the corresponding constants can be computed from those of the two systems. In particular, multilayer RNNs which can be represented as a series interconnection of several single layer RNNs will be of class $\mathcal{S}$, if each individual layer is of class $\mathcal{S}$.

Assumptions on Data and Predictors

Let us now formally state our assumptions on the data and the class of predictors.

In a nutshell, we would like the process $\mathbf{y},\mathbf{x}$ to be outputs of a class $\mathcal{S}$ system for an i.i.d. input process, and the predictors (hypotheses) to be class $\mathcal{S}$ systems. To this end, we need to consider $\mathcal{S}$ systems driven by stochastic inputs. More precisely, let $S = (\mathbb{S},\mathbb{Y},\mathbb{V},f,g)$ be a system of class $\mathcal{S}$, and let $\mathbf{v}$ be an essentially bounded stochastic process taking values in $\mathbb{V}$. Then for any initial state $\boldsymbol{s}_0$, time instances $t_0 \le t$, we can define the random variable $\mathbf{y}_S(\boldsymbol{s}_0,t_0,\mathbf{v};t) : \Omega \ni \omega \mapsto \mathbf{y}_S(\boldsymbol{s}_0,t_0,\mathbf{v}(\omega);t)$, and we refer to it as the *output process of S at time t for the input $\mathbf{v}$, initial state $\boldsymbol{s}_0$ and initial time t_0*. Likewise, for every $t \in \mathbb{Z}$, we define the random variable $\mathbf{y}_{S,\mathbf{v}}(t) : \Omega \ni \omega \mapsto \mathbf{y}_{S,\mathbf{v}(\omega)}(t)$, and we call the stochastic process formed by $\{\mathbf{y}_{S,\mathbf{v}}(t)\}_{t\in\mathbb{Z}}$ *the steady-state output process of S associated with $\mathbf{v}$*. We are now ready to state our assumptions on the data.

Assumption 1 (Data Generator)**.** *There exists a system $S_g = (\mathbb{S}_g, \mathbb{Y}\times\mathbb{X}, \mathbb{V}_g, f_g, g_g)$ of class $\mathcal{S}$ with constants $C_g, \tau_g, L_{g,\mathbf{v}}, L_{g_g,\mathbf{s}}, L_{g_g,\mathbf{v}}$, and an essentially bounded i.i.d. process $\mathbf{e}_g$ such that $\left[\mathbf{y}^T,\mathbf{x}^T\right]^T$ is the steady-state output process of S_g associated with $\mathbf{e}_g$, i.e., $\left[\mathbf{y}^T(t),\mathbf{x}^T(t)\right]^T = \mathbf{y}_{S_g,\mathbf{e}_g}(t)$ for all $t \in \mathbb{Z}$.*

Note that, we assume the existence of the generator S_g, but we do not assume the knowledge of S_g.

Assumption 1 can be viewed as a realisability assumption: we assume that the data generator is of the same type as the predictor. Another reason for Assumption of 1 is that it guarantees certain weak dependence properties of the data, which, in turn, allow us to use extensions of Hoeffding inequalities (Rio 2000) to prove PAC-Bayesian bounds.

More precisely, recall from (Alquier, Li, and Wintenberger 2013, Definition 5) the notion of the mixing coefficient $\theta^{\mathbf{q}}_{\infty,N}(1)$, $\forall\, 1 \le N \in \mathbb{Z}$, of a process $\mathbf{q}$. Following (Alquier, Li, and Wintenberger 2013) we will say that $\mathbf{q}$ is *weakly dependent with constants $B^{\mathbf{q}}$ and $\bar{\theta}^{\mathbf{q}}_{\infty}$*, if $\mathbf{q}$ is stationary, essentially bounded and for all $t \in \mathbb{Z}$, $B^{\mathbf{q}} \ge \|\mathbf{q}(t)\|_\infty$ w.p. 1, and $\bar{\theta}^{\mathbf{q}}_{\infty}(1) \ge \theta^{\mathbf{q}}_{\infty,N}(1)$ for all $N \ge 1$.

Lemma 2. *Under Assumption 1, $\mathbf{q} = [\mathbf{y},\mathbf{x}]$ is weakly dependent with the constants*

$$B^{\mathbf{q}} \triangleq 2\|\mathbf{e}_g(0)\|_\infty \left(L_{g_g,\mathbf{v}} + \frac{L_{g,\mathbf{v}}L_{g_g,\mathbf{s}}}{1-\tau_g}\right),$$

$$\bar{\theta}^{\mathbf{q}}_{\infty}(1) \triangleq 2\|\mathbf{e}_g(0)\|_\infty \frac{L_{g,\mathbf{v}}L_{g_g,\mathbf{s}}}{(1-\tau_g)^2}$$

Later on we shall see that the bound on the generalisation gap depends on the constants $B^{\mathbf{q}}$ and $\bar{\theta}^{\mathbf{q}}_{\infty}(1)$. Hence, we will assume that they are known. Intuitively, $B^{\mathbf{q}}$ is just an upper bound on the norm of the data. The constant $\bar{\theta}^{\mathbf{q}}_{\infty,N}(1)$ encodes the information on how non i.i.d. the data is. In particular, it is zero for i.i.d. data. The knowledge of these constants is often assumed in the literature, e.g, (Alquier, Li, and Wintenberger 2013; Alquier 2021).

Similarly, we assume that the predictors $\hat{\mathbf{y}}(t) = h(\{\mathbf{x}(s)\}_{s=0}^t)$, take the form of class $\mathcal{S}$ system and they are parametrised by elements of a parameter set Θ.

Assumption 2 (Parameterisation & realisation by class $\mathcal{S}$)**.** *There exists a compact set $\Theta \subseteq \mathbb{R}^{n_\theta}$, a bounded set $\mathbb{S} \subseteq \mathbb{R}^n$, a function $\Pi : \Theta \to \mathcal{H}$, and continuous functions*

$$\hat{f} : \mathbb{S}\times\mathbb{X}\times\Theta \to \mathbb{S}, \quad \hat{g} : \mathbb{S}\times\mathbb{X}\times\Theta \to \mathbb{Y}, \quad \hat{\boldsymbol{s}}_s : \Theta \to \mathbb{S}$$

such that the following holds.

- *For any $\theta \in \Theta$, consider the functions*

$$\hat{f}_\theta : \mathbb{S}\times\mathbb{X} \ni (\boldsymbol{s},\boldsymbol{v}) \mapsto \hat{f}(\boldsymbol{s},\boldsymbol{v},\theta) \in \mathbb{S},$$
$$\hat{g}_\theta : \mathbb{S}\times\mathbb{X} \ni (\boldsymbol{s},\boldsymbol{v}) \mapsto \hat{g}(\boldsymbol{s},\boldsymbol{v},\theta) \in \mathbb{Y}$$

  Then $S_\theta \triangleq (\mathbb{S},\mathbb{Y},\mathbb{X},\hat{f}_\theta,\hat{g}_\theta)$ is a system of class $\mathcal{S}$.
- *The predictor $h_\theta \triangleq \Pi(\theta)$ is such that for all $t \ge 0$, $t \in \mathbb{Z}$, $h_\theta(\{\mathbf{x}(s)\}_{s=0}^t)$ is the output of S_θ at time t for input $\mathbf{x}$, initial state $\hat{\boldsymbol{s}}_s(\theta)$, and initial time 0, i.e.,*

$$h_\theta(\{\mathbf{x}(s)\}_{s=0}^t) = \mathbf{y}_{S_\theta}(\hat{\mathbf{s}}_s(\theta),0,\mathbf{x};t).$$

This class of predictors includes RNNs (Θ is the set of weights), and most of standard classes of state-space representations used for time-series prediction and filtering. Assumption 2 guarantees existence of the generalisation loss

System	Conditions	Constants
$\mathbf{s}(t+1) = \sigma_f(A\mathbf{s}(t) + B\mathbf{x}(t) + b_1)$ $\mathbf{y}(t) = \sigma_h(C\mathbf{s}(t) + D\mathbf{x}(t) + b_2)$	σ_f σ_h are Lipschitz $\mathrm{Lip}(\sigma_f)\|A\|_2 < 1$	$C = 1, \quad \tau = \mathrm{Lip}(\sigma_f)\|A\|_2$ $L_{\mathbf{v}} = \mathrm{Lip}(\sigma_f)^{-1}\|B\|_2,$ $L_{g,\mathbf{s}} = \mathrm{Lip}(\sigma_h)\|C\|_2$ $L_{g,\mathbf{v}} = \mathrm{Lip}(\sigma_h)\|D\|_2$

Table 1: Example of Class $\mathcal{S}$ systems ($\mathrm{Lip}(\phi)$ denotes the Lipschitz constant of ϕ)

defined in equation 2. To this end, for every $\theta \in \Theta$ let us denote:

$$\begin{aligned} \hat{\mathbf{y}}_\theta(t|0) &\triangleq \mathbf{y}_{S_\theta}(\hat{\mathbf{s}}_s(\theta), 0, \mathbf{x}; t) = h_\theta(\{\mathbf{x}(s)\}_{s=0}^t), \\ \hat{\mathbf{y}}_\theta(t) &= \mathbf{y}_{S_\theta,\mathbf{x}}(t) \end{aligned} \tag{8}$$

i.e., $\hat{\mathbf{y}}_\theta(t|0)$ is just the prediction at time t generated by h_θ, and $\hat{\mathbf{y}}_\theta(t)$ is the output process $\mathbf{y}_{S_\theta,\mathbf{x}}(t)$ of the $\mathcal{S}$ system S_θ associated with $\mathbf{x}$.

From (Eringis et al. 2023a, Lemma A.4) it follows that the limit

$$\lim_{t\to\infty} \mathbb{E}[\ell(\mathbf{y}(t), h_\theta(\{\mathbf{x}(s)\}_{s=0}^t)) = \lim_{t\to\infty} \mathbb{E}[\ell(\mathbf{y}(t), \hat{\mathbf{y}}_\theta(t \mid 0))$$

exists and equals $\mathbb{E}[\ell(\mathbf{y}(t), \hat{\mathbf{y}}_\theta(t))]$ hence the generalisation loss $\mathcal{L}(h_\theta)$ is well-defined for all $\theta \in \Theta$.

Previously, we have informally taken expectation over predictors. Assumption 2 makes its possible to define the latter notion formally. Let B_Θ be the σ-algebra of Lebesque-measurable subsets of the parameter set Θ, and m denote the Lebesque measure on $\mathbb{R}^{n_\theta}$. If ρ is a probability density function on the measure space (Θ, B_θ, m), and F is a measurable and absolutely integrable function on θ, then we denote by

$$E_{\theta\sim\rho} F(\theta) \triangleq \int_{\theta\in\Theta} \rho(\theta) F(\theta) dm(\theta)$$

the expectation of F w.r.t. ρ. If $F : \mathcal{H} \to \mathbb{R}$ is a map such that $\Theta \ni \theta \mapsto F(h_\theta)$ is measurable and absolutely integrable, then with an abuse of notation we use

$$E_{h\sim\rho} F(h) \triangleq \int_{\theta\in\Theta} \rho(\theta) F(h_\theta) dm(\theta)$$

to denote the expectation of $\Theta \ni \theta \mapsto F(h_\theta)$ w.r.t. ρ. In particular, with the convention above, the expectations $E_{h\sim\rho}\hat{\mathcal{L}}_N(h)$ and $E_{h\sim\rho}\mathcal{L}(h)$ are well defined.

Assumption 3. *The loss function ℓ is Lipschitz, with global Lipschitz constant $L_\ell > 0$*

Note that Assumption 2 implies that for any parameter $\theta \in \Theta$, the output $\hat{\mathbf{y}}_\theta(t|0)$ is bounded, i.e., $\|\hat{\mathbf{y}}_\theta(t|0)\| \leq G_\theta B^{\mathfrak{q}}$, where $G_\theta \triangleq \frac{L_{g,\mathbf{s}}(\theta) L_{\mathbf{v}}(\theta)}{1-\tau(\theta)} + L_{g,\mathbf{v}}(\theta)$. Since all the processes are bounded, one can use restrictions of locally Lipschitz loss functions, e.g. square loss, to a bounded set, and they will be Lipschitz. For example, the restriction of the square error loss $\ell(\hat{\mathbf{y}}, \mathbf{y}) = \|\hat{\mathbf{y}} - \mathbf{y}\|_2^2$, is Lipschitz with a constant $L_\ell = 2B^{\mathfrak{q}} \max\{1, \sup_{\theta\in\Theta} G_\theta\}$.

Remark 4. *In contrast to standard classification, classification with soft labels also fit our framework (Hinton, Vinyals, and Dean Dec. 2014). To this end, we use the softmax loss function, $\ell(\mathbf{y}, \hat{\mathbf{y}}) = -\sum_{i=1}^K y_i \ln\left(\frac{e^{\hat{y}_i}}{\sum_{j=1}^K e^{\hat{y}_j}}\right)$, see (Eringis et al. 2023a, Lemma A.10).*

Main Result

With the definitions and assumptions above, we can state the main result formally. The informal theorem Theorem 1 is a special case of this main result. To this end, for any probability density π on (Θ, B_θ, m), denote by $\mathcal{M}_\pi$ the set of all probability densities for which the corresponding probability measures are absolutely continuous w.r.t. to the probability measure defined by π. Furthermore, for any $\theta \in \Theta$, denote by $C(\theta), L_{g,\mathbf{s}}(\theta), L_{g,\mathbf{v}}(\theta), L_{\mathbf{v}}(\theta)$, and $\tau(\theta) \in [0, 1)$ the constants of the class $\mathcal{S}$ system $\mathcal{S}_\theta$, and let $\hat{\boldsymbol{s}}_s(\theta)$ be the initial state of S_θ from which the hypothesis h_θ is generated, see Assumption 2. Let $\bar{\theta}^{\mathfrak{q}}_\infty(1)$ and $B^{\mathfrak{q}}$ be the constants of the data generating system from Lemma 2.

Theorem 2. *Under Assumptions 1,2, and 3, for any probability density π on (Θ, B_θ, m), for any $\delta \in (0, 0.5]$, and $\lambda > 0$ the following holds with probability at least $1 - 2\delta$,*

$$\begin{aligned} \forall \hat{\rho} \in \mathcal{M}_\pi : \; & E_{h\sim\hat{\rho}}\mathcal{L}(h) \leq E_{h\sim\hat{\rho}}\hat{\mathcal{L}}_N(h) \\ & + \frac{1}{\lambda}\left[D_{\mathrm{KL}}(\hat{\rho}\|\pi) + \ln\frac{1}{\delta} + \widehat{\Psi}_\pi(\lambda, N)\right], \end{aligned} \tag{9}$$

with $\widehat{\Psi}_\pi(\lambda, N) \triangleq \frac{1}{2}\left(\ln E_{\theta\sim\pi}\widehat{\Psi}_1(\theta) + \ln E_{\theta\sim\pi}\widehat{\Psi}_2(\theta)\right)$,

$$\widehat{\Psi}_1(\theta) \triangleq \exp\left\{\frac{2\lambda^2 L_\ell^2}{N}\left(B^{\mathfrak{q}}(G_\theta + H_\theta) + \bar{\theta}^{\mathfrak{q}}_\infty(1) G_\theta\right)^2\right\},$$

$$\widehat{\Psi}_2(\theta) \triangleq \exp\left\{\frac{2\lambda L_\ell C(\theta)}{N}\left(2B^{\mathfrak{q}} H_\theta + \|\hat{\boldsymbol{s}}_s(\theta)\|_2 \psi_0(\theta)\right)\right\},$$

$$G_\theta \triangleq \frac{L_{g,\mathbf{s}}(\theta) L_{\mathbf{v}}(\theta)}{1-\tau(\theta)} + L_{g,\mathbf{v}}(\theta), \; H_\theta \triangleq \frac{L_{g,\mathbf{s}}(\theta) L_{\mathbf{v}}(\theta)}{(1-\tau(\theta))^2},$$

$\psi_0(\theta) \triangleq \dfrac{L_{g,\mathbf{s}}(\theta)}{1-\tau(\theta)}$ *and* $D_{\mathrm{KL}}(\hat{\rho}\|\pi) \triangleq E_{\theta\sim\hat{\rho}} \ln\frac{\hat{\rho}(\theta)}{\pi(\theta)}$ *is the Kullback–Leibler (KL) divergence between $\hat{\rho}$ and π.*

The proof of Theorem 2 is presented in (Eringis et al. 2023a, Appendix A.5). The quantities $\widehat{\Psi}_1(\theta), \widehat{\Psi}_2(\theta)$ can be computed using Markov-Chain Monte-Carlo methods for standard architectures, e.g. RNNs, for which the relevant constants have explicit expressions (e.g., see Table 1). Moreover, $\widehat{\Psi}_i(\theta)$, $i = 1, 2$ depend on the parameterisation, which is reflected in the use of $E_{\theta\sim\pi}$ to denote their averages w.r.t. the prior π. In contrast, $\mathcal{L}(h)$ and $\hat{\mathcal{L}}_N(h)$ depend only on the hypothesis class, and we use $E_{h\sim\hat{\rho}}$ to denote their averages w.r.t. the posterior $\hat{\rho}$.

Theorem 2, provides the most general case. Under more strict assumptions, from Theorem 2 we can trivially arrive at Catoni-like bounds:

Corollary 1. *If*

$$G_1 \triangleq \sup_{\theta\in\Theta} 2L_\ell^2 \left(B^{\mathbf{q}}\left(G_\theta + H_\theta\right) + \bar{\theta}^{\mathbf{q}}_{\infty,N}(1) G_\theta\right)^2,$$
$$G_2 \triangleq \sup_{\theta\in\Theta} 2L_\ell C(\theta) \left(2B^{\mathbf{q}} H_\theta + \frac{\|\hat{\boldsymbol{s}}_s(\theta)\|_2 L_{g,\mathbf{s}}(\theta)}{1-\tau(\theta)}\right) \tag{10}$$

exist, then Theorem 1, with G_1, G_2 from equation 10 holds.

The constants G_1 and G_2 depend on the chosen parametrisation of the hypothesis class.

Discussion on the bound The bound is increasing in :

(1) the magnitude of the data (term B^q, Lemma 2),

(2) in the mixing coefficient $\bar{\theta}^{\mathbf{q}}_\infty(1)$ of the data see Lemma 2. The smaller this mixing coefficient is, i.e., the closer the data is to being i.i.d., the smaller is the bound. In particular, for i.i.d. inputs (i.e., $\bar{\theta}^{\mathbf{q}}_\infty(1) = 0$) we get back the classical PAC-Bayesian bounds (Alquier 2021).

(3) In the degree of robustness of the predictors captured by $\tau(\theta)$, and to a smaller extent by $L_{g,\mathbf{s}}(\theta), L_{\boldsymbol{v}}(\theta)$, $L_{g,\boldsymbol{v}}$, $C(\theta)$. By Remark 3, the smaller these constants are, the more robust the predictors are. That is, robustness is connected to a smaller generalisation gap.

The role of the number of time steps In our setting, due to the definition of the generalisation loss, the number of time steps for which the predictor has been run during inference does not enter the bounds. The key for achieving this was to assume that the predictors are stable dynamical systems. This is in contrast to other bounds for RNNs/dynamical systems (Koiran and Sontag 1998; Sontag 1998; Chen, Li, and Zhao 2020; Wei and Ma 2019; Akpinar, Kratzwald, and Feuerriegel 2020; Joukovsky et al. 2021) which grow with the number of time steps used for inference. The latter makes it difficult to use those bounds to characterise the generalisation loss for inference from long sequences.

Role of the depth of RNNs As it was mentioned previously, for multi-layer RNNs, the constants $\tau(\theta)$, $L_{g,\mathbf{s}}(\theta), L_{\boldsymbol{v}}(\theta)$, $L_{g,\boldsymbol{v}}$, $C(\theta)$ can be estimated based on the corresponding constants for each layer. However, these estimates grow more conservatives, as the number of layers grows, resulting in a more conservative PAC-Bayesian bound. That is, similarly to other PAC(-Bayesian) bounds, our bound may increase with the depth of RNNs, even though the depths does not enter it directly.

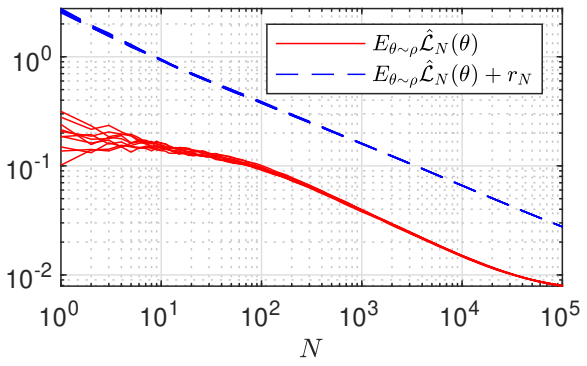

Figure 1: Theorem 2 is used to compute the results of the numerical example, evaluated on 10 different realisations of data.

Illustrative Example

In this section we shall explore a synthetic example to illustrate the proposed bound. The code for this example is available in Git repository in (Eringis 2023).We randomly chose a generator as in Assumption 1 with:

$$\mathbf{s}_g(t+1) = \text{ReLu}\left(A_g \mathbf{s}_g(t) + B_g \mathbf{e}_g(t) + b_{\mathbf{s},g}\right), \tag{11a}$$
$$\begin{bmatrix}\mathbf{y}(t)\\ \mathbf{x}(t)\end{bmatrix} = \tanh\left(C_g \mathbf{s}_g(t) + D_g \mathbf{e}_g(t) + b_{\mathbf{y},g}\right), \tag{11b}$$

with $n_{\mathbf{s}} = 2, n_{\mathbf{y}} = 1, n_{\mathbf{x}} = 1$, see numerical values of the weights $(A_g, B_g, b_{\mathbf{s},g}, C_g, D_g, b_{\mathbf{y},g})$ in (Eringis et al. 2023a, equation 213, Appendix B). Then Lemma 2 holds with $B^{\mathbf{q}} = \sqrt{2}$, $\bar{\theta}_{\infty,N}(1) = 2$, and $\|\mathbf{e}_g(t)\|_\infty \leq 1.27$

We generate data using equation 11 by sampling $\mathbf{e}_g(t)$ from a truncated Gaussian distribution. The predictors use Relu and tanh activation functions, and 2 hidden states, and all weights are parameterised including the initial state. The loss function is square loss. We employ Markov Chain Monte-Carlo sampling to compute the various expectations over the prior and posterior appearing in the bound of Theorem 2, see (Eringis et al. 2023a, Appendix B).

The prior is chosen as $\pi = \mathcal{N}(0, \sigma^2 I)$, with $\sigma^2 = 0.02$. The posterior is the Gibbs posterior, i.e. $\hat{\rho}_N(\theta) \propto \pi(\theta) e^{-\lambda_N \hat{\mathcal{L}}(\theta)}$ with $\lambda_N = \sqrt{N}$.For this particular example, the predictor output of which is always zero is the one with the maximum prediction error, which is 1. In particular, as we can see in Figure 1, the proposed bound is smaller than 1 and hence it is non-vacuous for $N \geq 9$.

Conclusion

In this paper, we have provided non-asymptotic bounds on the generalisation gap of exponentially stable dynamical systems. Under suitable conditions on hyper-parameter λ, we see that the bound on generalisation gap converges at a rate of $\mathcal{O}(1/\sqrt{N})$. The bound will converge either to 0 for a fixed posterior or to a constant involving only KL divergence, for Gibbs posterior. Furthermore the bound only depends on quantities related to the magnitude of the label and input processes, and the θ_∞ mixing coefficient of these processes. Not only does the proposed bound inform us how to design priors that yield smaller generalisation gap, but since the proposed bound only requires limited knowledge of the data generator, we could potentially apply these bounds directly for various applications.

Potential future research directions include applying our results to various architectures, which would require deriving tools to check if these architectures belong to class $\mathcal{S}$. Furthermore, designing LMIs or other tools to obtain tighter system constants, would immediately yields tighter bounds on the generalisation gap.

Acknowledgments

This work was partially supported by the IEA program of CNRS.

References

Akpinar, N.-J.; Kratzwald, B.; and Feuerriegel, S. 2020. Sample Complexity Bounds for RNNs with Application to Combinatorial Graph Problems (Student Abstract). *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(10): 13745–13746.

Alquier, P. 2021. User-friendly introduction to PAC-Bayes bounds. *arXiv:2110.11216*.

Alquier, P.; Li, X.; and Wintenberger, O. 2013. Prediction of time series by statistical learning: general losses and fast rates. *Dependence Modeling*, 1(2013): 65–93.

Alquier, P.; Ridgway, J.; and Chopin, N. 2016. On the properties of variational approximations of Gibbs posteriors. *JMLR*, 17(239): 1–41.

Alquier, P.; and Wintenberger, O. 2012. Model selection for weakly dependent time series forecasting. *Bernoulli*, 18(3): 883 – 913.

Blanke, M.; and Lelarge, M. 2023. FLEX: an Adaptive Exploration Algorithm for Nonlinear Systems. *arXiv preprint arXiv:2304.13426*.

Chen, M.; Li, X.; and Zhao, T. 2020. On Generalization Bounds of a Family of Recurrent Neural Networks. In Chiappa, S.; and Calandra, R., eds., *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, 1233–1243. PMLR.

Cohen-Karlik, E.; Menuhin-Gruman, I.; Giryes, R.; Cohen, N.; and Globerson, A. 2023. Learning Low Dimensional State Spaces with Overparameterized Recurrent Neural Nets. In *The Eleventh International Conference on Learning Representations*.

Dedecker, J.; Doukhan, P.; Lang, G.; Leon, J. R.; Louhichi, S.; and Clémentine, P. 2007. *Weak dependence: with examples and applications*. Lecture notes in statistics. New York: Springer. ISBN 978-0-387-69951-6.

Dziugaite, G. K.; and Roy, D. M. 2017. Computing Nonvacuous Generalization Bounds for Deep (Stochastic) Neural Networks with Many More Parameters than Training Data. In *UAI*. AUAI Press.

Emami, M.; Sahraee-Ardakan, M.; Pandit, P.; Rangan, S.; and Fletcher, A. K. 2021. Implicit bias of linear rnns. In *International Conference on Machine Learning*, 2982–2992. PMLR.

Eringis, D. 2023. Code Appendx for PAC-Bayes Generalisation Bounds for Dynamical Systems Including Stable RNNs. https://gitlab.com/DeividasEringis/PAC-Bayes-NL.

Eringis, D.; Leth, J.; Tan, Z.-H.; Wisniewski, R.; Esfahan, A. F.; and Petreczky, M. 2021. PAC-Bayesian theory for stochastic LTI systems. In *2021 60th IEEE Conference on Decision and Control (CDC)*, 6626–6633.

Eringis, D.; Leth, J.; Tan, Z.-H.; Wisniewski, R.; and Petreczky, M. 2022. Explicit construction of the minimum error variance estimator for stochastic LTI state-space systems. *arXiv preprint arXiv:2109.02384*.

Eringis, D.; Leth, J.; Tan, Z.-H.; Wisniewski, R.; and Petreczky, M. 2023a. PAC-Bayes Generalisation Bounds for Dynamical Systems Including Stable RNNs. arXiv:2312.09793.

Eringis, D.; Leth, J.; Tan, Z.-H.; Wisniewski, R.; and Petreczky, M. 2023b. PAC-Bayesian bounds for learning LTI-ss systems with input from empirical loss. *arXiv:2303.16816*.

Foster, D.; Sarkar, T.; and Rakhlin, A. 2020. Learning nonlinear dynamical systems from a single trajectory. In *Learning for Dynamics and Control*, 851–861. PMLR.

Germain, P.; Bach, F.; Lacoste, A.; and Lacoste-Julien, S. 2016. PAC-Bayesian Theory Meets Bayesian Inference. In *NIPS*, 1876–1884.

Goodfellow, I.; Bengio, Y.; Courville, A.; and Bengio, Y. 2016. *Deep learning*, volume 1. MIT Press.

Guedj, B. 2019. A Primer on PAC-Bayesian Learning. *arXiv preprint arXiv:1901.05353*.

Hazan, E.; Lee, H.; Singh, K.; Zhang, C.; and Zhang, Y. 2018. Spectral Filtering for General Linear Dynamical Systems. In Bengio, S.; Wallach, H.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Hinton, G.; Vinyals, O.; and Dean, J. Dec. 2014. Distilling the knowledge in a neural network. *Deep Learning and Representation Learning Workshop, NIPS, Montreal, Canada*.

Joukovsky, B.; Mukherjee, T.; Van Luong, H.; and Deligiannis, N. 2021. Generalization error bounds for deep unfolding RNNs. In de Campos, C.; and Maathuis, M. H., eds., *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161 of *Proceedings of Machine Learning Research*, 1515–1524. PMLR.

Koiran, P.; and Sontag, E. D. 1998. Vapnik-Chervonenkis dimension of recurrent neural networks. *Discrete Applied Mathematics*, 86(1): 63–79.

Lale, S.; Azizzadenesheli, K.; Hassibi, B.; and Anandkumar, A. 2020. Logarithmic regret bound in partially observable linear dynamical systems. *Advances in Neural Information Processing Systems*, 33: 20876–20888.

Li, Y.; Ildiz, M. E.; Papailiopoulos, D.; and Oymak, S. 2023. Transformers as Algorithms: Generalization and Stability in In-context Learning. In Krause, A.; Brunskill, E.; Cho, K.; Engelhardt, B.; Sabato, S.; and Scarlett, J., eds., *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 19565–19594. PMLR.

Ljung, L. 1999. *System Identification: Theory for the user (2nd Ed.)*. PTR Prentice Hall., Upper Saddle River, USA.

Mania, H.; Jordan, M. I.; and Recht, B. 2022. Active Learning for Nonlinear System Identification with Guarantees. *J. Mach. Learn. Res.*, 23: 32–1.

Oymak, S.; and Ozay, N. 2022. Revisiting Ho–Kalman-Based System Identification: Robustness and Finite-Sample Analysis. *IEEE Transactions on Automatic Control*, 67(4): 1914–1928. Bounds on markov params of SS (no code).

Pavlov, A.; and van de Wouw, N. 2012. Steady-State Analysis and Regulation of Discrete-Time Nonlinear Systems. *IEEE Transactions on Automatic Control*, 57(7): 1793–1798.

Rio, E. 2000. Inégalités de Hoeffding pour les fonctions lipschitziennes de suites dépendantes. *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics*, 330(10): 905–908.

Roy, A.; Balasubramanian, K.; and Erdogdu, M. A. 2021. On Empirical Risk Minimization with Dependent and Heavy-Tailed Data. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y.; Liang, P. S.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*, volume 34, 8913–8926. Curran Associates, Inc.

Sarkar, T.; Rakhlin, A.; and Dahleh, M. A. 2021. Finite Time LTI System Identification. *J. Mach. Learn. Res.*, 22: 26:1–26:61.

Sattar, Y.; and Oymak, S. 2022. Non-asymptotic and accurate learning of nonlinear dynamical systems. *The Journal of Machine Learning Research*, 23(1): 6248–6296.

Sattar, Y.; Oymak, S.; and Ozay, N. 2022. Finite Sample Identification of Bilinear Dynamical Systems. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, 6705–6711.

Sayedana, B.; Afshari, M.; Caines, P. E.; and Mahajan, A. 2022. Consistency and Rate of Convergence of Switched Least Squares System Identification for Autonomous Markov Jump Linear Systems. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, 6678–6685.

Shalev-Shwartz, S.; and Ben-David, S. 2014. *Understanding machine learning: From theory to algorithms*. Cambridge university press.

Shi, S.; Mazhar, O.; and De Schutter, B. 2022. Finite-sample analysis of identification of switched linear systems with arbitrary or restricted switching. *IEEE Control Systems Letters*, 7: 121–126.

Simchowitz, M. 2021. *Statistical Complexity and Regret in Linear Control*. University of California, Berkeley.

Simchowitz, M.; Boczar, R.; and Recht, B. 2019. Learning linear dynamical systems with semi-parametric least squares. In *Conference on Learning Theory*, 2714–2802. PMLR.

Simchowitz, M.; and Foster, D. 2020. Naive Exploration is Optimal for Online LQR. In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 8937–8948. PMLR.

Simchowitz, M.; Mania, H.; Tu, S.; Jordan, M. I.; and Recht, B. 2018. Learning without mixing: Towards a sharp analysis of linear system identification. In *Conference On Learning Theory*, 439–473. PMLR.

Sontag, E. D. 1998. A learning result for continuous-time recurrent neural networks. *Systems & control letters*, 34(3): 151–158.

Tsiamis, A.; and Pappas, G. J. 2019. Finite Sample Analysis of Stochastic System Identification. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, 3648–3654.

Wei, C.; and Ma, T. 2019. Data-dependent Sample Complexity of Deep Neural Networks via Lipschitz Augmentation. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Woo, Y.; Kim, D.; Jeong, J.; Ko, Y.-W.; and Lee, J.-G. 2021. Zero-Keep Filter Pruning for Energy/Power Efficient Deep Neural Networks. *Electronics*, 10(11): 1238.

Zhang, J.; Lei, Q.; and Dhillon, I. 2018. Stabilizing Gradients for Deep Neural Networks via Efficient SVD Parameterization. In Dy, J.; and Krause, A., eds., *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 5806–5814. PMLR.

Ziemann, I.; and Tu, S. 2022. Learning with little mixing. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 4626–4637. Curran Associates, Inc.

Ziemann, I. M.; Sandberg, H.; and Matni, N. 2022. Single Trajectory Nonparametric Learning of Nonlinear Dynamics. In Loh, P.-L.; and Raginsky, M., eds., *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, 3333–3364. PMLR.