A Statistical Investigation of Long Memory in Language and Music

Alec Greaves-Tunnell [1] **Zaid Harchaoui** [1]

Abstract

Representation and learning of long-range dependencies is a central challenge confronted in modern applications of machine learning to sequence data. Yet despite the prominence of this issue, the basic problem of measuring long-range dependence, either in a given data source or as represented in a trained deep model, remains largely limited to heuristic tools. We contribute a statistical framework for investigating long-range dependence in current applications of deep sequence modeling, drawing on the well-developed theory of long memory stochastic processes. This framework yields testable implications concerning the relationship between long memory in real-world data and its learned representation in a deep learning architecture, which are explored through a semiparametric framework adapted to the high-dimensional setting.

1. Introduction

Advances in the design and optimization of deep recurrent neural networks (RNNs) have lead to significant breakthroughs in the modeling of complex sequence data, including natural language and music. An omnipresent challenge in these sequence modeling tasks is to capture long-range dependencies between observations, and a great variety of model architectures have been developed with this objective explicitly in mind. However, it can be difficult to assess whether and to what extent a given RNN has learned to represent such dependencies, that is, whether it has *long memory*.

Currently, if a model's capacity to represent long-range dependence is measured at all, it is typically evaluated heuristically against some task or tasks in which success is taken an indicator of "memory" in a colloquial sense. Though undoubtedly helpful, such heuristics are rarely defined with respect to an underlying mathematical or statistical property of interest, nor do they necessarily have any correspondence to the data on which the models are subsequently trained. In this paper, we pursue a complementary approach in which long-range dependence is assessed as a quantitative and statistically accessible feature of a given data source. Consequently, the problem of evaluating long memory in RNNs can be re-framed as a comparison between a learned representation and an estimated property of the data.

The main contribution is the development and illustration of a methodology for the estimation, visualization, and hypothesis testing of long memory in RNNs, based on an approach that mathematically defines and directly estimates long-range dependence as a property of a multivariate time series. We offer extensive validation of the proposed approach and explore strategies to overcome problems with hypothesis testing for long memory in the high-dimensional regime. We report experimental results obtained on a wide-ranging collection of music and language data, confirming the (often strong) long-range dependencies that are observed by practitioners. However, we show evidence that this property is not adequately captured by a variety of RNNs trained to benchmark performance on a language dataset.[1]

Related work. Though a formal connection to long memory processes has been lacking thus far, machine learning applications to sequence modeling have long been concerned with the capture of long-range dependencies. The development of RNN models has been strongly influenced by the identification of the "vanishing gradient problem" in (Bengio et al., 1994). More complex recurrent architectures, such as long short-term memory (Hochreiter & Schmidhuber, 1997a), gated recurrent units (Cho et al., 2014), and structurally constrained recurrent networks (Mikolov et al., 2015) were designed specifically to alleviate this problem. Alternative approaches have pursued a more formal understanding of RNN computation, for example through kernel methods (Lei et al., 2017), by means of ablative strategies clarifying the computation of the RNN hidden state (Levy et al., 2018), or through a dynamical systems approach

[1] Department of Statistics, University of Washington, Seattle, USA. Correspondence to: Alec Greaves-Tunnell <alecgt@uw.edu>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019. Copyright 2019 by the author(s).

[1] Code corresponding to these experiments, including an illustrative Jupyter notebook, is available for download at `https://github.com/alecgt/RNN_long_memory`.

(Miller & Hardt, 2019). TA modern statistical perspective on nonlinear time series analysis is provided in (Douc et al., 2014).

Long-range dependence is most commonly evaluated in RNN models by test performance on a synthetic classification task. For example, the target may be the parity of a binary sequence (so-called "parity" problems), or it may be the class of a sequence whose most recent terms are replaced with white noise ("2-sequence" or "latch" problems) (Bengio et al., 1994; Bengio & Frasconi, 1994; Lin et al., 1996). A simple demonstration relatively early in RNN history by Hochreiter & Schmidhuber (1997b) showed that such tasks can often be solved quickly by random parameter search, casting doubt on their informativeness. Whereas the authors proposed a different heuristic, we seek to re-frame the problem of long memory evaluation so that it is amenable to statistical analysis.

Classical constructions of long memory processes (Mandelbrot & Van Ness, 1968; Granger & Joyeux, 1980; Hosking, 1981) laid the foundation for statistical methods to estimate long memory from time series data. See also (Moulines et al., 2008; Reisen et al., 2017) for recent works in this area. The multivariate estimator of Shimotsu (2007) is the foundation of the methodology we develop here. It is by now well understood that failure to properly account for long memory can severely diminish performance in even basic estimation (Percival & Guttorp, 1994). Furthermore, failure to model long memory has been shown to harm predictive performance, particularly in the case of multi-step forecasting (Brodsky & Hurvich, 1999).

2. Background

Long memory in stochastic processes. Long memory has a simple and intuitive definition in terms of the autocovariance sequence of a real, stationary stochastic process $X_t \in \mathbb{R}, t \in \mathbb{Z}$. The process X_t is said to have long memory if the autocovariance

$$\gamma(k) = \mathrm{Cov}(X_t, X_{t+k}), \;\; k \in \mathbb{Z}$$

satisfies

$$\gamma_X(k) \sim L_\gamma(k)|k|^{-(1-2d)} \;\text{ as }\; k \to \infty, \tag{1}$$

for some $d \in (0, 1/2)$, where $a(k) \sim b(k)$ indicates that $a(k)/b(k) \to 1$ as $k \to \infty$, and $L_\gamma(k)$ is a slowly varying function at infinity. See (Greaves-Tunnell & Harchaoui, 2019) for details on the mathematical framework. The term "long memory" is justified by the slow (hyperbolic) decay of the autocovariance sequence. As a consequence of this slow decay, the partial sums of the absolute autocovariance sequence diverge. This can be directly contrasted with the "short memory" case, in which the autocovariance sequence is absolutely summable. Moreover, we note that the parameter d allows one to quantify the memory by controlling the strength of long-range dependencies.

In the time series literature, a spectral definition of "memory" is preferred, as it unifies the long and short memory cases. A second-order stationary time series can be represented in the frequency domain by its spectral density function

$$f_X(\lambda) = \sum_{k=-\infty}^{\infty} \gamma(k) e^{ik\lambda}.$$

If X_t has a spectral density function that satisfies

$$f_X(\lambda) = L_f(\lambda)|\lambda|^{-2d} \tag{2}$$

where $L_f(\lambda)$ is slowly varying at zero, then X_t has long memory if $d \in (0, 1/2)$, short memory for $d = 0$, and "intermediate memory" or "antipersistence" if $d \in (-1/2, 0)$. The two definitions of long memory are equivalent when $L_f(\lambda)$ is quasimonotone (Beran et al., 2013).

We summarize the complementary time and frequency domain views of long memory with a simple illustration in Figure 1, which contrasts a short memory autoregressive (AR) process of order 1 with its long memory counterpart, the fractionally integrated AR process. The autocovariance series is seen to converge rapidly for the AR process, whereas it diverges for the fractionally integrated AR process. Meanwhile, Eq. (2) implies that the long memory parameter d has a geometric interpretation in the frequency domain as the slope of $\log f_X(\lambda)$ versus $-2\log(\lambda)$ as $\lambda \to 0$.

In the long memory regime, past observations can retain significant explanatory power with respect to future prediction targets, and informative forecasts are available over horizons extending well beyond that of an analogous short memory process. The contrast between AR and fractionally integrated AR processes again provides a concrete example: analyzing the optimal predictor $\hat{X}_{t+h}$ of X_{t+h} for the prediction horizon $h \geq 1$ in terms of the proportion of variance explained $R^2(h) = 1 - \mathrm{Var}(X_{t+h})^{-1}\mathrm{MSE}(h)$, it can be shown that $R(h)$ decays exponentially for the AR process but only hyperbolically in the fractionally integrated case (Beran et al., 2013).

Many common models do not have long memory. Despite the appeal and practicality of long memory for modeling complex time series, we emphasize that it is absent from nearly all common statistical models for sequence data. We offer a short list of examples; see (Greaves-Tunnell & Harchaoui, 2019) for all proofs.

- *Markov models.* If X_t is a Markov process on a finite state space $\mathcal{X}$, and $Y_t = g(X_t)$ for any function $g : \mathcal{X} \to \mathbb{R}$, then Y_t has short memory. We show that this

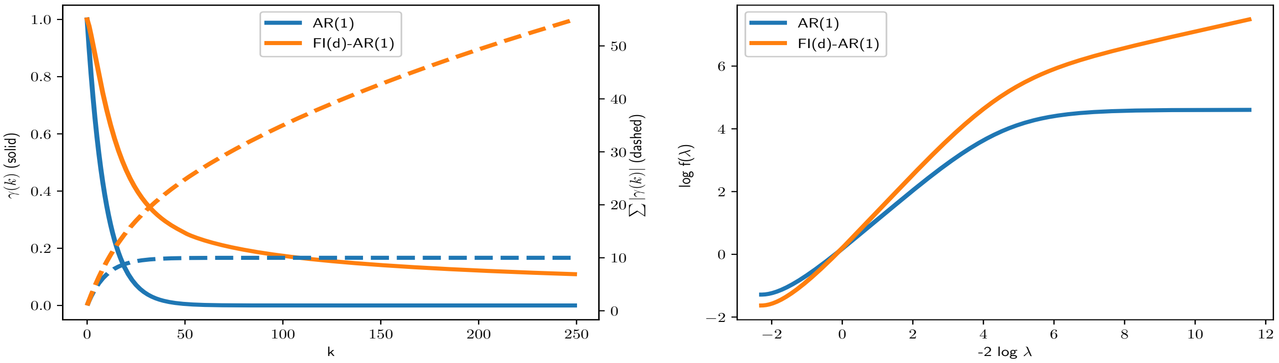

Figure 1. Time and frequency domain views of an AR(1) process (blue, $d = 0$) and its long memory counterpart obtained by fractional differencing (orange, $d = 0.25$). *Left:* Autocorrelation sequences (solid lines) of the two processes, along with their partial sums (dotted lines). *Right:* Log-log plot of the spectral density function versus frequency.

property holds even in a complex model with Markov structure, the Markov transition distribution model for high-order Markov chains (Raftery, 1985).

- *Autoregressive moving average (ARMA) models*. ARMA models, a ubiquitous tool in time series modeling, likewise have exponentially decaying autocovariances and thus short memory (Brockwell & Davis, 2013).
- *Nonlinear autoregressions*. Finally, and most importantly for our present focus, nonlinearity of the state transition function is no guarantee of long memory. We show that a class of autoregressive processes in which the state is subject to iterated nonlinear transformations still fails to achieve a slowly decaying autocovariance sequence (Lin et al., 1996; Gourieroux & Jasiak, 2005).

Semiparametric estimation of long memory. Methods for the estimation of the long memory parameter d have been developed and analyzed under increasingly broad conditions. Here, we focus on semiparametric methods, which offer consistent estimation of the long memory without the need to estimate or even specify a full parametric model. The term "semiparametric" refers to the fact that the estimation problem involves a finite-dimensional parameter of interest (the long memory vector) and an infinite-dimensional nuisance parameter (the spectral density).

Semiparametric estimation in the Fourier domain leverages the implication of Eq. (2) that

$$f_X(\lambda) \sim c_f|\lambda|^{-2d} \tag{3}$$

as $\lambda \to 0$, with c_f a nonzero constant. Estimators are constructed directly from the periodogram using only terms corresponding to frequencies near the origin. The long memory parameter d is estimated either by log-periodogram regression, which yields the Geweke-Porter-Hudak (GPH) estimator (Geweke & Porter-Hudak, 1983), or through a local Gaussian approximation, which gives the Gaussian semiparametric estimator (GSE) (Robinson, 1995). The GSE offers greater efficiency, requires weaker distributional assumptions, and can be defined for both univariate and multivariate time series; therefore it will be the main focus.

Multivariate long memory processes. Analysis of long memory in multivariate stochastic processes is a topic of more recent investigation in the time series literature. The common underlying assumption in multivariate semiparametric estimation of long memory is that the real, vector-valued process $X_t \in \mathbb{R}^p$, can be written as

$$\begin{bmatrix}(1-B)^{d_1} & & 0 \\ & \ddots & \\ 0 & & (1-B)^{d_p}\end{bmatrix}\begin{bmatrix}X_{t1} \\ \vdots \\ X_{tp}\end{bmatrix} = \begin{bmatrix}U_{t1} \\ \vdots \\ U_{tp}\end{bmatrix}, \tag{4}$$

where X_{ti} is the i^{th} component of X_t, $U_t \in \mathbb{R}^p$ is a second-order stationary process with spectral density function bounded and bounded away from zero at zero frequency, B is the backshift in time operator, and $|d_i| < 1/2$ for every $i = 1, ..., p$ (Shimotsu, 2007). The backshift operation $B^j X_t = X_{t-j}$, $j \in \mathbb{Z}$ is extended to non-integer orders via

$$(1-B)^{-d} = \sum_{k=0}^{\infty} \frac{\Gamma(d+k)}{k!\Gamma(d)} B^k,$$

and thus X_t is referred to as a *fractionally integrated* process when $d \neq 0$. Fractionally integrated processes are the most commonly used models for data with long-range dependencies, encompassing parametric classes such as the vector autoregressive fractionally integrated moving average (VARFIMA), a multivariate and long memory extension of the popular ARMA family of time series models.

If X_t is defined as in Eq. (4), then its spectral density function $f_X(\lambda)$ satisfies (Hannan, 2009)

$$f_X(\lambda) = \Phi(\lambda, d) f_U(\lambda) \Phi^*(\lambda, d),$$

where x^* denotes the complex conjugate of x, $f_U(\lambda)$ is the spectral density function of U_t at frequency λ, and

$$\Phi(\lambda, d) = \text{diag}\left((1 - e^{i\lambda})^{-d_i}\right)_{i=1,\ldots,p}.$$

Given an observed sequence $(x_1, ..., x_T) = x_{1:T}$ with discrete Fourier transform

$$y_j = \frac{1}{\sqrt{2\pi T}} \sum_{t=1}^{T} x_t e^{-i\lambda_j t}, \ \ \lambda_j = 2\pi j/T,$$

the spectral density matrix is estimated at Fourier frequency λ_j by the periodogram

$$I(\lambda_j) = y_j y_j^*.$$

Under the assumption that $f_U(\lambda) \sim G$ as $\lambda \to 0$ for some real, symmetric, positive definite $G \in \mathbb{R}^{p \times p}$, the local behavior of $f_X(\lambda)$ around the origin is governed only by d and G:

$$f(\lambda_j) \sim \Phi(\lambda, d) G \Phi^*(\lambda, d). \tag{5}$$

The Gaussian semiparametric estimator The Gaussian semiparametric estimator of d (Shimotsu, 2007) is computed from a local, frequency-domain approximation to the Gaussian likelihood based on Eq. (5). The approximation is valid under restriction of the likelihood to a range of frequencies close to the origin. Using the identity $1-e^{-i\lambda} = 2\sin(\lambda/2)e^{i(\pi-\lambda)/2}$, we have the approximation

$$\Phi(\lambda, d) \approx \text{diag}(\lambda^{-d} e^{i(\pi-\lambda)/2}) \triangleq \Lambda(d),$$

which is valid up to an error term of order $O(\lambda^2)$.

The Gaussian log-likelihood is written in the frequency domain as (Whittle, 1953)

$$\begin{aligned}\mathcal{L}_m &= \frac{1}{m} \sum_{j=1}^{m} \log\det f_X(\lambda_j) + \text{Tr}\left[f_X(\lambda_j)^{-1} y_j y_j^*\right] \\ &\approx \frac{1}{m} \sum_{j=1}^{m} \Big[\log\det \Lambda_j(d) G \Lambda_j^*(d) \\ &\qquad + \text{Tr}\left[\left(\Lambda_j(d) G \Lambda_j^*(d)\right)^{-1} I(\lambda_j)\right]\Big].\end{aligned}$$

Validity of the approximation is ensured by restriction of the sum to the first m Fourier frequencies, with $m = o(T)$.

Solving the first-order optimality condition

$$\begin{aligned}\frac{\partial \mathcal{L}_m}{\partial G} = \frac{1}{m} \sum_{j=1}^{m} \Big[&(G^T)^{-1} \\ &- \left(G^{-1} \Lambda_j(d)^{-1} I(\lambda_j) \Lambda_j^*(d)^{-1} G^{-1}\right)^T\Big] = 0\end{aligned}$$

for G yields

$$\widehat{G}(d) = \frac{1}{m} \sum_{j=1}^{m} \text{Re}\left[\Lambda_j(d)^{-1} I(\lambda_j) \Lambda_j^*(d)^{-1}\right].$$

Substitution back into the objective results in the expression

$$\mathcal{L}_m(d) = \log\det \widehat{G}(d) - 2 \sum_{i=1}^{p} d_i \sum_{j=1}^{m} \log \lambda_j, \tag{6}$$

and the Gaussian semiparametric estimator is obtained as the minimizer

$$\hat{d}_{\text{GSE}} = \text{argmin}_{d \in \Theta} \mathcal{L}_m(d), \tag{7}$$

over the feasible set $\Theta = (-1/2, 1/2)^p$.

A key result due to Shimotsu (2007) establishes that the estimator $\hat{d}_{\text{GSE}}$ is consistent and asymptotically normal under mild conditions, with

$$\sqrt{m}(\hat{d}_{\text{GSE}} - d_0) \to_d \mathcal{N}(0, \Omega^{-1}), \tag{8}$$

where

$$\Omega = 2\left[I_p + G \odot G^{-1} + \frac{\pi^2}{4}(G \odot G^{-1} - I_p)\right],$$

d_0 is the true long memory, and $\odot$ denotes the Hadamard product.

Optimization. Relatively little discussion of optimization procedures for problem in Eq. (7) is available in the time series literature. We are not aware of any proof that the objective is convex in the multivariate setting for instance.

To compute the estimator $\hat{d}_{\text{GSE}}$, we apply L-BFGS-B, a quasi-Newton algorithm that handles box constraints (Byrd et al., 1995). L-BFGS-B is an iterative algorithm requiring the gradient of the objective; see (Greaves-Tunnell & Harchaoui, 2019) for a detailed derivation of the gradient.

Bandwidth selection The choice of the bandwidth parameter m determines the tradeoff between bias and variance in the estimator: at small m the variance may be high due to few data points, while setting m too large can introduce bias by accounting for the behavior of the spectral density function away from the origin.

When it is possible to simulate from the target process, as will be the case when evaluating criteria for long memory in recurrent neural networks, the variance can be controlled by simulating long sequences and computing a dense estimate of the periodogram. Without knowledge of the shape of the spectral density function, however, it is difficult to know how to set the bandwidth to avoid bias, and thus a relatively conservative setting of $m = \sqrt{T}$ is preferred; see (Greaves-Tunnell & Harchaoui, 2019) for a detailed bias study.

3. Methods

RNN hidden state as a nonlinear model for a long memory process. The standard tool for statistical modeling of multivariate long memory processes is the vector autoregressive fractionally integrated moving average (VARFIMA) model, which represents the process $X_t \in \mathbb{R}^p$ with long memory parameter d as

$$\Phi(B)(1-B)^d X_t = \theta(B) Z_t,$$

where Z_t is a white noise process and $(1-B)^d = \mathrm{diag}((1-B)^{d_i})$, $i = 1, ..., p$ (Lobato, 1997; Sowell, 1989). Under the standard stationarity and invertibility conditions on the matrix polynomials $\Phi(B)$ and $\Theta(B)$, respectively, the process can be represented as

$$X_t = (1-B)^{-d} \Phi^{-1}(B) \Theta(B) Z_t,$$

which shows that the X_t has a composite representation in terms of linear "features" of the input sequence and an explicit fractional integration step ensuring that it satisfies the definition of multivariate long memory in Eq. (4).

We extend this view to deep network models for sequences with long range dependencies. The key difference is that RNN models are not constrained to work with a linear representation of the data, nor do they explicitly contain a step that guarantees the long memory of X_t. To evaluate long memory in an RNN model, we study the stochastic process

$$X_t = \Psi(Z_t), \tag{9}$$

where Z_t is again a white noise, and the nonlinear transformation Ψ describes the RNN transformation of inputs to the hidden state. In a typical RNN model, a decision rule is learned by linear modeling of the hidden state; this framework thus aligns with a broader theoretical characterization of deep learning as approximate linearization of complex decision boundaries in input space by means of a learned nonlinear feature representation (Bruna & Mallat, 2013; Mairal et al., 2014; Jones et al., 2019; Bietti & Mairal, 2019).

Testable criteria for RNN capture of long-range dependence. The complexity of $\Psi(\cdot)$ corresponding to even the most basic RNN sequence models precludes a fully theoretical treatment of long memory in processes described by Eq. (9). Nonetheless, this characterization suggests an approach for the statistical evaluation of long memory in RNNs, as it establishes testable criteria under which a model of the form Eq. (9) describes a process X_t with long memory. In particular, to satisfy the definition in Eq. (4) we must have

$$X_t = \Psi(Z_t) = (1-B)^{-d} \tilde{\Psi}(Z_t)$$

for some $d \neq 0$ and process $\tilde{\Psi}(Z_t)$ with bounded and nonzero spectral density at zero frequency. Semiparametric estimation of d in the frequency domain provides a means to evaluate this condition such that the results are agnostic to the behavior of $\tilde{\Psi}(Z_t)$ at higher frequencies. If $\Psi(Z_t)$ admits a representation in terms of an explicit fractional integration step, then this can be investigated in two complementary experiments:

1. **Integration of fractionally differenced input.** Define
$$\tilde{X}_t = (1-B)^d Z_t,$$
where Z_t is a standard Gaussian white noise and d is the long memory parameter corresponding to the source X_t on which the model was trained. If the sequence $\tilde{x}_{1:T}$ is drawn from $\tilde{X}_t$, then we expect to find that
$$\hat{d}_{\mathrm{GSE}}(\tilde{h}_{1:T}) \approx 0,$$
where $\tilde{h}_{1:T} = \Psi(\tilde{x}_{1:T})$ is the RNN hidden representation of the simulated input. On the other hand, nonzero long memory in the hidden state indicates a mismatch between fractional integration learned by the RNN and long memory of the data X_t.

2. **Long memory transformation of white noise.** Conversely, we expect to find that the RNN hidden representation of a white noise sequence has a nonzero long memory parameter. White noise has a constant spectrum and thus a long memory parameter equal to zero. If $\Psi(\cdot)$ performs both the feature representation and fractional integration functions that are handled separately and explicitly in the VARFIMA model, then a zero-memory input will be transformed to a nonzero-memory sequence of hidden states.

Total memory. It is common for sequence embeddings and RNN hidden layers to have hundreds of dimensions, and thus long memory estimation for these sequences naturally occurs in a high-dimensional setting. This topic is virtually unexplored in the time series literature, where multivariate studies tend to have modest dimension. Practically, this raises two main issues. First, if $p \approx m$ for dimension p and bandwidth m, then the approximation of the test statistic distribution by its asymptotic limit will be of poor quality, and the resulting test is likely to be miscalibrated. Second, it becomes difficult to interpret the long memory vector d, particularly when the coordinates of the corresponding time series are not meaningful themselves.

We resolve both issues by considering the *total memory* statistic $\bar{d}$, defined as

$$\bar{d} = 1^T \hat{d}_{\mathrm{GSE}}. \tag{10}$$

Computation of the total memory is no more complex than that of the GSE, and it has an intuitive interpretation as the coordinate-wise aggregate strength of long memory in a multivariate time series.

Asymptotic normality of the total memory estimator. The total memory is a simple linear functional of the GSE, and thus its consistency and asymptotic normality can be established by a simple argument. In particular, defining

$$\bar{d} = g(d) \triangleq 1^T \hat{d}_{\text{GSE}},$$

we see that $\nabla g(d) = 1$, so that by Eq. (8) and the delta method we have

$$\sqrt{m}(\bar{d} - \bar{d}_0) \to_d \mathcal{N}(0, 1^T \Omega^{-1} 1), \qquad (11)$$

where $\bar{d}_0$ is the true total memory of the observed process.

Visualizing and testing for long memory in high dimensions. The visual time-domain summary of long memory in Figure 1 can be extended to the multivariate setting. In this case, the autocovariance $\gamma(k) = \text{Cov}(X_t, X_{t+k})$ is matrix-valued, which for the purpose of evaluating long memory can be summarized by the scalar $\text{Tr}(|\gamma(k)|)$, where the absolute value is taken element-wise. Recall that a sufficient condition for short memory is the absolute convergence of the autocovariance series, whereas this series diverges for long memory processes.

From a testing perspective, a statistical decision rule for the presence of long memory can be derived from the asymptotic distribution of the corresponding estimator. However, when the dimension p is large and we conservatively set the bandwidth $m = \sqrt{T}$, we may have $m \approx p$ even when the observed sequence is relatively long.

The classical approach to testing for the multivariate Gaussian mean is based on the Wald statistic

$$m(\hat{d} - d_0)^T \Omega (\hat{d} - d_0),$$

which has a $\chi^2(p)$ distribution under the null hypothesis $\mathcal{H}_0 : d = d_0$.

Additional simulations in (Greaves-Tunnell & Harchaoui, 2019) demonstrate that the standard Wald test can be miscalibrated when $m \approx p$, whereas testing for long memory with the total memory statistic remains well-calibrated in this setting. These results are consistent with previous observations that the Wald test for long memory can have poor finite-sample performance even in low dimensions (Shimotsu, 2007; Hurvich & Chen, 2000).

4. Experiments

4.1. Long memory in language and music

Much of the development of deep recurrent neural networks has been motivated by the goal of finding good representations and models for text and audio data. The results in this section confirm that such data can be considered as realizations of long memory processes.[2] A full summary of results is given in Table 1, and autocovariance partial sums are plotted in Figure 2. To facilitate comparison of the estimated long memory across time series of different dimension, we report the normalized total memory $\bar{d}/p = (1^T \hat{d}_{\text{GSE}})/p$ in all tables.

In this section, we test the null hypothesis

$$\mathcal{H}_0 : \bar{d}_0 = 0$$

against the one-sided alternative of long memory,

$$\mathcal{H}_1 : \bar{d}_0 > 0.$$

We set the level of the test to be $\alpha = 0.05$ and compute the corresponding critical value c_α from the asymptotic distribution of the total memory estimator. Given an estimate of the total memory $\bar{d}(x_{1:T})$, a p-value is computed as $P(\bar{d} > \bar{d}(x_{1:T}) | \bar{d}_0 = 0)$; note that a p-value less than $\alpha = 0.05$ corresponds to rejection of the null hypothesis in favor of the long memory alternative.

Table 1. Total Memory in Natural Language and Music Data.

	Data	**Norm. total memory**	**p-value**	**Reject $\mathcal{H}_0$?**
Natural language	Penn TreeBank	0.163	$<1 \times 10^{-16}$	✓
	Facebook CBT	0.0636	$<1 \times 10^{-16}$	✓
	King James Bible	0.192	$<1 \times 10^{-16}$	✓
Music	J.S. Bach	0.0997	$<1 \times 10^{-16}$	✓
	Miles Davis	0.322	$<1 \times 10^{-16}$	✓
	Oum Kalthoum	0.343	$<1 \times 10^{-16}$	✓

Natural language data. We evaluate long memory in three different sources of English language text data: the Penn TreeBank training corpus (Marcus et al., 1993), the training set of the Children's Book Test from Facebook's bAbI tasks (Weston et al., 2016), and the King James Bible. The Penn TreeBank corpus and King James Bible are considered as single sequences, while the Children's Book Test data consists of 98 books, which are considered as separate sequences. We require that each sequence be of length at least $T = 2^{14}$, which ensures that the periodogram can be estimated with reasonable density near the origin. Finally, we use GloVe embeddings (Pennington et al., 2014) to convert each sequence of word tokens to sequence of real vectors of equal length and dimension $p = 200$.

[2] Code for all results in this section is available at `https://github.com/alecgt/RNN_long_memory`

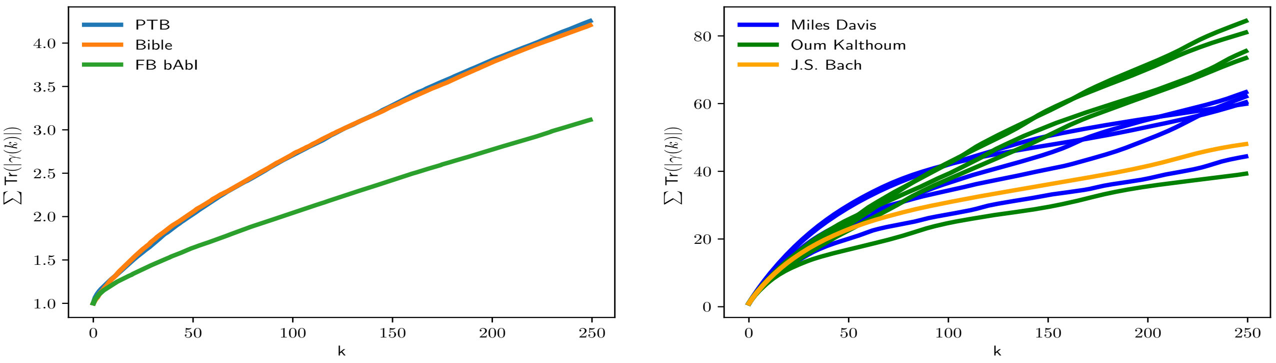

Figure 2. Partial sum of the autocovariance trace for embedded natural language and music data. *Left*: Natural language data. For clarity we include only the longest of the 98 books in the Facebook bAbI training set. *Right:* Music data. Each of the five tracks from both Miles Davis and Oum Kalthoum is plotted separately, while the Bach cello suite is treated as a single sequence.

The results show significant long memory in each of the text sources, despite their apparent differences. As might be expected, the children's book from the Facebook bAbI dataset demonstrates the weakest long-range dependencies, as is evident both from the value of the total memory statistic and the slope of the autocovariance partial sum.

Music data. Modeling and generation of music has recently gained significant visibility in the deep learning community as a challenging set of tasks involving sequence data. As in the natural language experiments, we seek to evaluate long memory in a broad selection of representative data. To this end, we select a complete Bach cello suite consisting of 6 pieces from the MusicNet dataset (Thickstun et al., 2017), the jazz recordings from Miles Davis' *Kind of Blue*, and a collection of the most popular works of famous Egyptian singer Oum Kalthoum.

For the Bach cello suite, we embed the data from its raw scalar wav file format using a reduced version of a deep convolutional model that has recently achieved near state-of-the-art prediction accuracy on the MusicNet collection of classical music (Thickstun et al., 2018).

We are not aware of a prominent deep learning model for either jazz music or vocal performances. Therefore, for the recordings of Miles Davis and Oum Kalthoum, we revert to a standard method and extract mel-frequency cepstral coefficients (MFCC) from the raw wav files at a sample rate of 32000 Hz (Logan et al., 2000). See (Greaves-Tunnell & Harchaoui, 2019) for an analysis of the impact of embedding choice on estimated long memory.

The results show that long memory appears to be even more strongly represented in music than in text. We find evidence of particularly strong long-range dependence in the recordings of Miles Davis and Oum Kalthoum, consistent with their reputation for repetition and self-reference.

Overall, while the results of this section are unlikely to surprise practitioners familiar with the modeling of language and music data, they are scientifically useful for two main reasons: first, they show that the long memory analysis is able to identify well-known instances of long-range dependence in real-world data; second, they establish quantitative criteria for the successful representation of this dependency structure by RNNs trained on such data.

4.2. Long memory analysis of language model RNNs

We now turn to the question of whether RNNs trained on one of the datasets evaluated above are able to represent the long-range dependencies that we know to be present. We evaluate the criteria for long memory on three different RNN architectures: long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997a), memory cells (Levy et al., 2018), and structurally constrained recurrent networks (SCRN) (Mikolov et al., 2015). Each network is trained on the Penn TreeBank corpus as part of a language model that includes a learned word embedding and linear decoder of the hidden states; the architecture is identical to the "small" LSTM model in (Zaremba et al., 2014), which is preferred for the tractable dimension of the hidden state. Note that the objective is not to achieve state-of-the-art results, but rather to reproduce benchmark performance in a well-known deep learning task. Finally, for comparison, we also include an untrained LSTM in the experiments; the parameters of this model are simply set by random initialization.

RNN integration of fractionally differenced input. Having estimated the long memory parameter d corresponding to the Penn TreeBank training data in the previous section, we simulate inputs $\tilde{x}_{1:T}$ with $T = 2^{16}$ from by fractional differencing of a standard Gaussian white noise and evaluate the total memory of the corresponding hidden representation $\Psi(\tilde{x}_{1:T})$ for each RNN. Results from $n = 100$

Table 2. Language Model Performance by RNN Type

Model	Test Perplexity
Zaremba et al.	114.5
LSTM	114.5
Memory cell	119.0
SCRN	124.3

trials are compiled in Table 3 (standard error of total memory estimates in parentheses). We test the null hypothesis $\mathcal{H}_0 : \bar{d} = 0$ against the one-sided alternative $\mathcal{H}_1 : \bar{d} < 0$, which corresponds to the model's failure to represent the full strength of fractional integration observed in the data.

Table 3. Residual Total Memory in RNN Representations of Fractionally Differenced Input.

Model	Norm. total memory	p-value	Reject $\mathcal{H}_0$?
LSTM (trained)	-8.36×10^{-3} (0.00475)	4.07×10^{-2}	✓
LSTM (untrained)	-6.20×10^{-2} (0.00387)	$<1 \times 10^{-16}$	✓
Memory cell	-1.18×10^{-2} (0.0539)	1.52×10^{-2}	✓
SCRN	-2.62×10^{-2} (0.0631)	3.32×10^{-5}	✓

RNN transformation of white noise. For a complementary analysis, we evaluate whether the RNNs can impart nontrivial long-range dependency structure to white noise inputs. In this case, the input sequence $z_{1:T}$ is drawn from a standard Gaussian white noise process, and we test the corresponding hidden representation $\Psi(z_{1:T})$ for nonzero total memory. As in the previous experiment, we select $T = 2^{16}$, choose the bandwidth parameter $m = \sqrt{T}$, and simulate $n = 100$ trials for each RNN. Results are detailed in Table 4. We test $\mathcal{H}_0 : \bar{d}_0 = 0$ against $\mathcal{H}_1 : \bar{d}_0 > 0$; here, the alternative corresponds to successful transformation of white noise input to long memory hidden state.

Discussion. We summarize the main experimental result as follows: there is a statistically well-defined and practically identifiable property, relevant for prediction and broadly represented in language and music data, that is not present according to two fractional integration criteria in a collection of RNNs trained to benchmark performance.

Tables 3 and 4 show that each evaluated RNN fails both criteria for representation of the long-range dependency structure of the data on which it was trained. The result holds despite a training protocol that reproduces benchmark performance, and for RNN architectures specifically engineered to alleviate the gradient issues typically implicated in the learning of long-range dependencies.

Table 4. Total Memory in RNN Representations of White Noise Input.

Model	Norm. total memory	p-value	Reject $\mathcal{H}_0$?
LSTM (trained)	-8.59×10^{-4} (0.00405)	0.583	X
LSTM (untrained)	-4.17×10^{-4} (0.00223)	0.572	X
Memory cell	-5.96×10^{-4} (0.00452)	0.552	X
SCRN	2.37×10^{-3} (0.00522)	0.324	X

5. Conclusion

We have introduced and demonstrated a framework for the evaluation of long memory in RNNs that proceeds from a well-known definition in the time series literature. Under this definition, long memory is the condition enabling meaningful autocovariance at long lags in a multivariate time series. Of course, for sufficiently complex processes, this will not fully characterize the long-range dependence structure of the data generating process. Nonetheless, it represents a practical and informative foundation upon which to develop a statistical toolkit for estimation, inference, and hypothesis testing, which goes beyond the current paradigm of heuristic checks.

The experiments investigate long memory in natural language and music data, along with the learned representations of RNNs themselves, using the total memory statistic as an interpretable quantity that avoids the challenges associated with high-dimensional testing. The results identify long memory as a broadly prevalent feature of natural language and music data, while showing evidence that benchmark recurrent neural network models designed to capture this phenomenon may in fact fail to do so. Finally, this work suggests future topics in both time series, particularly concerning long memory analysis in high dimensions, and in deep learning, as a challenge to learn long memory representations in RNNs.

Acknowledgements

This work was supported by the NIH Big Data for Genomics and Neuroscience Training Grant 8T32LM012419, NSF TRIPODS Award CCF-1740551, the program "Learning in Machines and Brains" of CIFAR, and faculty research awards.

References

Bengio, Y. and Frasconi, P. Credit assignment through time: Alternatives to backpropagation. In *Adv. NIPS*, 1994.

Bengio, Y., Simard, P., and Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994.

Beran, J., Feng, Y., Ghosh, S., and Kulik, R. *Long-Memory Processes: Probabilistic Properties and Statistical Methods*. Springer, 2013.

Bietti, A. and Mairal, J. Group invariance, stability to deformations, and complexity of deep convolutional representations. *The Journal of Machine Learning Research*, 20(1):876–924, 2019.

Brockwell, P. J. and Davis, R. A. *Time Series: Theory and Methods*. Springer, 2013.

Brodsky, J. and Hurvich, C. M. Multi-step forecasting for long-memory processes. *Journal of Forecasting*, 18(1): 59–75, 1999.

Bruna, J. and Mallat, S. Invariant scattering convolution networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1872–1886, 2013.

Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *EMNLP*, 2014.

Douc, R., Moulines, E., and Stoffer, D. *Nonlinear Time Series: Theory, Methods and Applications with R Examples*. Chapman and Hall/CRC, 2014.

Geweke, J. and Porter-Hudak, S. The estimation and application of long memory time series models. *Journal of Time Series Analysis*, 4(4):221–238, 1983.

Gourieroux, C. and Jasiak, J. Nonlinear innovations and impulse responses with application to var sensitivity. *Annales d'Economie et de Statistique*, pp. 1–31, 2005.

Granger, C. W. and Joyeux, R. An introduction to long-memory time series models and fractional differencing. *Journal of Time Series Analysis*, 1(1):15–29, 1980.

Greaves-Tunnell, A. and Harchaoui, Z. A statistical investigation of long memory in language and music. *arXiv preprint arXiv:1904.03834*, 2019.

Hannan, E. J. *Multiple Time Series*. John Wiley & Sons, 2009.

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

Hochreiter, S. and Schmidhuber, J. LSTM can solve hard long time lag problems. In *Adv. NIPS*, 1997b.

Hosking, J. R. Fractional differencing. *Biometrika*, 68(1): 165–176, 1981.

Hurvich, C. M. and Chen, W. W. An efficient taper for potentially overdifferenced long-memory time series. *Journal of Time Series Analysis*, 21(2):155–180, 2000.

Jones, C., Roulet, V., and Harchaoui, Z. Kernel-based translations of convolutional networks. *arXiv preprint arXiv:1903.08131*, 2019.

Lei, T., Jin, W., Barzilay, R., and Jaakkola, T. Deriving neural architectures from sequence and graph kernels. In *ICML*, 2017.

Levy, O., Lee, K., FitzGerald, N., and Zettlemoyer, L. Long short-term memory as a dynamically computed element-wise weighted sum. In *ACL*, 2018.

Lin, T., Horne, B. G., Tino, P., and Giles, C. L. Learning long-term dependencies in NARX recurrent neural networks. *IEEE Transactions on Neural Networks*, 7(6): 1329–1338, 1996.

Lobato, I. N. Consistency of the averaged cross-periodogram in long memory series. *Journal of Time Series Analysis*, 18(2):137–155, 1997.

Logan, B. et al. Mel frequency cepstral coefficients for music modeling. In *ISMIR*, 2000.

Mairal, J., Koniusz, P., Harchaoui, Z., and Schmid, C. Convolutional kernel networks. In *Adv. NIPS*, 2014.

Mandelbrot, B. B. and Van Ness, J. W. Fractional Brownian motions, fractional noises and applications. *SIAM Review*, 10(4):422–437, 1968.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1993.

Mikolov, T., Joulin, A., Chopra, S., Mathieu, M., and Ranzato, M. Learning longer memory in recurrent neural networks. In *ICLR*, 2015.

Miller, J. and Hardt, M. Stable recurrent models. In *ICLR*, 2019.

Moulines, E., Roueff, F., and Taqqu, M. S. A wavelet Whittle estimator of the memory parameter of a nonstationary Gaussian time series. *The Annals of Statistics*, 36(4): 1925–1956, 2008.

Pennington, J., Socher, R., and Manning, C. GloVe: Global vectors for word representation. In *EMNLP*, 2014.

Percival, D. B. and Guttorp, P. Long-memory processes, the Allan variance and wavelets. In *Wavelet Analysis and its Applications*, volume 4, pp. 325–344. Elsevier, 1994.

Raftery, A. E. A model for high-order Markov chains. *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 528–539, 1985.

Reisen, V. A., Lévy-Leduc, C., and Taqqu, M. S. An M-estimator for the long-memory parameter. *Journal of Statistical Planning and Inference*, 187:44–55, 2017.

Robinson, P. M. Gaussian semiparametric estimation of long range dependence. *The Annals of Statistics*, 23(5): 1630–1661, 1995.

Shimotsu, K. Gaussian semiparametric estimation of multivariate fractionally integrated processes. *Journal of Econometrics*, 137(2):277–310, 2007.

Sowell, F. Maximum likelihood estimation of fractionally integrated time series models. Working paper, 1989.

Thickstun, J., Harchaoui, Z., and Kakade, S. Learning features of music from scratch. In *ICLR*, 2017.

Thickstun, J., Harchaoui, Z., Foster, D. P., and Kakade, S. M. Invariances and data augmentation for supervised music transcription. In *ICASSP*, 2018.

Weston, J., Bordes, A., Chopra, S., Rush, A. M., van Merriënboer, B., Joulin, A., and Mikolov, T. Towards AI-complete question answering: A set of prerequisite toy tasks. In *ICLR*, 2016.

Whittle, P. Estimation and information in stationary time series. *Arkiv för Matematik*, 2(5):423–434, 1953.

Zaremba, W., Sutskever, I., and Vinyals, O. Recurrent neural network regularization. CoRR abs/1409.2329, 2014.