Analysis of stochastic Lanczos quadrature for spectrum approximation

Tyler Chen [1] **Thomas Trogdon** [1] **Shashanka Ubaru** [2]

Abstract

The cumulative empirical spectral measure (CESM) $\Phi[\mathbf{A}] : \mathbb{R} \to [0,1]$ of a $n \times n$ symmetric matrix $\mathbf{A}$ is defined as the fraction of eigenvalues of $\mathbf{A}$ less than a given threshold, i.e., $\Phi[\mathbf{A}](x) := \sum_{i=1}^{n} \frac{1}{n} \mathbb{1}[\lambda_i[\mathbf{A}] \leq x]$. Spectral sums $\operatorname{tr}(f[\mathbf{A}])$ can be computed as the Riemann–Stieltjes integral of f against $\Phi[\mathbf{A}]$, so the task of estimating CESM arises frequently in a number of applications, including machine learning. We present an error analysis for stochastic Lanczos quadrature (SLQ). We show that SLQ obtains an approximation to the CESM within a Wasserstein distance of $t\ |\lambda_{\max}[\mathbf{A}] - \lambda_{\min}[\mathbf{A}]|$ with probability at least $1 - \eta$, by applying the Lanczos algorithm for $\lceil 12t^{-1} + \frac{1}{2} \rceil$ iterations to $\lceil 4(n+2)^{-1} t^{-2} \ln(2n\eta^{-1}) \rceil$ vectors sampled independently and uniformly from the unit sphere. We additionally provide (matrix-dependent) a posteriori error bounds for the Wasserstein and Kolmogorov–Smirnov distances between the output of this algorithm and the true CESM. The quality of our bounds is demonstrated using numerical experiments.

1. Introduction

Given an $n \times n$ symmetric matrix $\mathbf{A}$, the cumulative empirical spectral measure (CESM) $\Phi[\mathbf{A}] : \mathbb{R} \to [0,1]$ gives the fraction of eigenvalues less than a given threshold. That is,

$$\Phi[\mathbf{A}](x) := \sum_{i=1}^{n} \frac{1}{n} \mathbb{1}[\lambda_i[\mathbf{A}] \leq x],$$

where $\mathbb{1}[\,\cdot \leq x] : \mathbb{R} \to \{0,1\}$ is the indicator function defined by $\mathbb{1}[s \leq x] = 1$ if $s \leq x$ and $\mathbb{1}[s \leq x] = 0$ if $s > x$. The CESM contains all information about spectrum of $\mathbf{A}$ and can therefore be used to compute any quantity depending on just the spectrum. Conversely, computing the CESM requires exact knowledge of all the eigenvalues of $\mathbf{A}$, which are expensive to compute.

For many applications, however, it suffices to provide a coarse estimate of the CESM. In machine learning, approximate CESMs have found use in facilitating backpropagation through implicit likelihoods (Ramesh & LeCun, 2018) as well as for for studying properties of Hessians during neural network training (Ghorbani et al., 2019; Papyan, 2019; Yao et al., 2020). This provides insight into differences between various training approaches and/or network architectures.

In data science more broadly, approximate CESMs have become a popular approach for exploring properties of graphs and networks as well as for approximating fundamental quantities such as matrix norms, log-determinants, number of eigenvalues in an interval, etc. (Avron, 2010; Napoli et al., 2016; Ubaru et al., 2017b; Han et al., 2017; Xi et al., 2018; Musco et al., 2019; Dong et al., 2019). Approximate CESMs have also long been used in computational physics and chemistry to study various observables (Ducastelle & Cyrot-Lackmann, 1970; Haydock et al., 1975; Wheeler & Blumstein, 1972) and remain widely used in these fields today (Weiße et al., 2006; Covaci et al., 2010; Sbierski et al., 2017; Schnack et al., 2020).

Finally, and as a result of their general usefulness, approximate CESMs have become the first stage of a range of algorithms for fundamental linear algebraic tasks including methods for computing matrix functions (Fan et al., 2019) and state of the art parallel eigensolvers (Polizzi, 2009; Li et al., 2019).

In this paper, we consider a well-known algorithm for computing an approximate CESM, which we refer to as stochastic Lanczos quadrature (SLQ). The algorithm described in this paper and closely related methods have been used for decades to estimate spectral densities (Haydock et al., 1972; 1975; Lambin & Gaspard, 1982; Benoit et al., 1992), and like the Lanczos algorithm (Lanczos, 1950) on which they are based, they remain highly relevant today (Lin et al., 2016; Ghorbani et al., 2019; Papyan, 2019).

[1]Department of Applied Mathematics, University of Washington, Seattle, Washington, USA [2]IBM T.J. Watson Research Center, Yorktown Heights, New York, USA. Correspondence to: Tyler Chen <chentyl@uw.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

1.1. Stochastic Lanczos Quadrature

Using the standard definition of a matrix function,[1] for a symmetric matrix $\mathbf{A}$, we denote by $\mathbb{1}[\mathbf{A} \leq x]$ the matrix with the same eigenvectors as $\mathbf{A}$, but whose eigenvalues are 0 or 1 depending on whether the corresponding eigenvalue of $\mathbf{A}$ is below or above x; that is, $\mathbb{1}[\mathbf{A} \leq x]$ is the orthogonal projector onto the eigenspace associated to eigenvalues $\lambda_i[\mathbf{A}]$ such that $\lambda_i[\mathbf{A}] \leq x$. Thus, $\Phi[\mathbf{A}](x) = n^{-1} \operatorname{tr}(\mathbb{1}[\mathbf{A} \leq x])$.

With this definition in place, we define the weighted CESM,

$$\Psi[\mathbf{A}, \mathbf{v}](x) := \mathbf{v}^{\mathsf{T}} \mathbb{1}[\mathbf{A} \leq x] \mathbf{v}.$$

If $\mathbf{v} \sim \mathcal{U}(\mathcal{S}^{n-1})$, where $\mathcal{U}(\mathcal{S}^{n-1})$ is the uniform distribution on the unit sphere, then the weighted CESM has the desirable properties (i) that it is an unbiased estimator for $\Phi[\mathbf{A}](x)$, and (ii) that it defines a cumulative probability distribution function; i.e. $\mathbb{E}[\Psi[\mathbf{A}, \mathbf{v}](x)] = \Phi[\mathbf{A}](x)$ and $\Psi[\mathbf{A}, \mathbf{v}] : \mathbb{R} \to [0, 1]$ is weakly increasing, right continuous, and

$$\lim_{x \to -\infty} \Psi[\mathbf{A}, \mathbf{v}](x) = 0, \qquad \lim_{x \to \infty} \Psi[\mathbf{A}, \mathbf{v}](x) = 1.$$

Next, we consider the degree k Gaussian quadrature rule $[\Psi[\mathbf{A}, \mathbf{v}]]_k^{\mathrm{gq}}$ for $\Psi[\mathbf{A}, \mathbf{v}]$. In general, a Gaussian quadrature rule for a distribution function can be computed using the Stieltjes procedure, which for distributions of the form $\Psi[\mathbf{A}, \mathbf{v}]$, is equivalent to the Lanczos algorithm (Gautschi, 2004; Golub & Meurant, 2009). Specifically, if $[\mathbf{T}]_{:k,:k}$ is the symmetric tridiagonal matrix obtained by running Lanczos on $\mathbf{A}$ and $\mathbf{v}$ for k steps, then

$$[\Psi[\mathbf{A}, \mathbf{v}]]_k^{\mathrm{gq}} = \Psi([\mathbf{T}]_{:k,:k}, \hat{\mathbf{e}})$$

where $\hat{\mathbf{e}} = [1, 0, \ldots, 0]^{\mathsf{T}}$.

By repeating this process over multiple samples and averaging, we arrive at SLQ, outlined in Algorithm 1.

Algorithm 1 Stochastic Lanczos Quadrature

input $\mathbf{A}$, n_{v}, k
  for $i = 1, 2, \ldots, n_{\mathrm{v}}$ **do**
    Sample $\mathbf{v}_i \sim \mathcal{U}(\mathcal{S}^{n-1})$ (and define $\Psi_i = \Psi(\mathbf{A}, \mathbf{v}_i)$)
    Run Lanczos on $\mathbf{A}, \mathbf{v}_i$ for k steps to compute $[\mathbf{T}_i]_{:k,:k}$
    Define $[\Psi_i]_k^{\mathrm{gq}} = \Psi[[\mathbf{T}_i]_{:k,:k}, \hat{\mathbf{e}}]$
  end for
return $\langle [\Psi_i]_k^{\mathrm{gq}} \rangle := \frac{1}{n_{\mathrm{v}}} \sum_{i=1}^{n_{\mathrm{v}}} [\Psi_i]_k^{\mathrm{gq}}$

SLQ is computationally efficient. In particular, all samples can be computed in parallel on separate machines or on a single machine using blocked linear algebra routines. Moreover, the algorithm is matrix free in that we only require a method to compute the map $\mathbf{v} \mapsto \mathbf{A}\mathbf{v}$, the cost of which we denote by T_{mv}. This is particularly important for large dense matrices, where the $O(n^2)$ storage required to keep every entry of $\mathbf{A}$ may be intractable. In many cases, such as with Hessians from the training of neural networks, matrix-vector products can be computed implicitly using only $O(n)$ storage (Pearlmutter, 1994). Similarly, if $\mathbf{A}$ is sparse or structured, this map may be evaluated faster than the $O(n^2)$ computation cost for an arbitrary matrix vector product; e.g. $O(\mathrm{nnz}[\mathbf{A}])$ for a sparse matrix or $O(n \ln n)$ for a circulant matrix.

1.2. Discussion of results

Our first main result is a runtime guarantee for SLQ. In particular, we show that if $n_{\mathrm{v}} > 4(n+2)^{-1} t^{-2} \ln(2n\eta^{-1})$ and $k > 12t^{-1} + \frac{1}{2}$, then

$$\mathbb{P}\big[d_{\mathrm{W}}(\Phi[\mathbf{A}], \langle [\Psi_i]_k^{\mathrm{gq}} \rangle) > t I[\mathbf{A}]\big] < \eta,$$

where $I[\mathbf{A}] := |\lambda_{\max}[\mathbf{A}] - \lambda_{\min}[\mathbf{A}]|$ and $d_{\mathrm{W}}(\cdot, \cdot)$ denotes the Wasserstein distance between two distribution functions as in Definition 3 below. This implies that as $n \to \infty$, for $t \gg n^{-1/2}$, SLQ has a runtime of $O(T_{\mathrm{mv}} t^{-1} \log(t^{-2}\eta^{-1}))$. This bound is nearly tight in the sense that for any $t \in (0, 1)$, there exists a matrix of size $\lceil (4t)^{-1} \rceil$ such that at least $(8t)^{-1}$ matrix vector products are required to obtain an output with Wasserstein distance less than $t I[\mathbf{A}]$.

The second main result is an a posteriori upper bound for Wasserstein and Kolmogorov–Smirnov (KS) distances, which take into account spectrum dependent features such as clustered or isolated eigenvalues.

Finally, in proving these results, we show that if $n_{\mathrm{v}} > (n+2)^{-1} t^{-2} \ln(2\eta^{-1})$ then, for any $x \in \mathbb{R}$,

$$\mathbb{P}\left[\left|\Phi[\mathbf{A}](x) - \left(\frac{1}{n_{\mathrm{v}}} \sum_{i=1}^{n_{\mathrm{v}}} \mathbf{v}_i^{\mathsf{T}} \mathbb{1}[\mathbf{A} \leq x] \mathbf{v}_i\right)\right| > t\right] < \eta. \tag{1}$$

This is applicable to the analysis of a range of algorithms beyond SLQ.

1.3. Related work

A variety of algorithms for approximating the CESM have been developed; see (Fischer, 2011; Weiße et al., 2006; Lin et al., 2016; Adams et al., 2018; Cohen-Steiner et al., 2018) and the references therein for a more complete overview. By far, the most popular algorithms are the kernel polynomial method (KPM) (Weiße et al., 2006) and SLQ. The two algorithms differ primarily in how they approximate the weighted CESM $\Psi[\mathbf{A}, \mathbf{v}]$, and as a result, our analysis of

[1] For a symmetric matrix $\mathbf{A}$ with eigenvalue decomposition $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\mathsf{T}}$ and scalar function $f : \mathbb{R} \to \mathbb{R}$, the matrix function $f[\mathbf{A}]$ is defined by $f[\mathbf{A}] := \mathbf{U} f[\mathbf{\Lambda}] \mathbf{U}^{\mathsf{T}}$, where $f[\mathbf{\Lambda}]$ is the diagonal matrix with f applied entrywise to the diagonal entries of the matrix $\mathbf{\Lambda}$.

the estimator $\Psi[\mathbf{A}, \mathbf{v}]$ for $\Phi[\mathbf{A}]$ applies to the KPM as well. The main advantages of SLQ are that it is adaptive to the spectrum of $\mathbf{A}$ (i.e. it automatically detects features such as gaps in the spectrum, isolated or clustered eigenvalues, etc.) and that it produces an output which is a probability distribution function. However, in doing so, SLQ requires the computation of inner products, which may be costly, especially in a distributed computing environment (Lin et al., 2016). Moreover, the behavior of the Lanczos method is not always straightforward in finite precision.

We remark that the KPM, both with exact and inexact matrix-vector products, has recently been analyzed (Braverman et al., 2021). The analysis for KPM with exact matrix-vector products yields similar rates to our analysis for SLQ, but the sample complexity is given in terms of unspecified universal constants and has a polynomially worse dependence on the accuracy parameter t. We believe this is an artifact of analysis not an inherent property of the KPM, but to be certain a full side-by-side comparison of the algorithms is needed.

Tail bounds similar to (1) have been derived in several contexts. First, while explicit constants are not specified, the same $n^{-1}t^{-2}\ln(\eta^{-1})$ dependence for n_{v} is implied by (Deift & Trogdon, 2020, Lemma 4.5). Second, if the $\mathbf{v}_i$ are replaced by *unnormalized* Gaussian vectors (with mean zero and variance n^{-1}), then well known bounds for trace estimation (Avron & Toledo, 2011; Roosta-Khorasani & Ascher, 2014) yield similar rates in terms of explicit constants. However, the weighed CESM corresponding to an unnormalized sample will not be a probability distribution function.

Finally, the algorithm studied in this paper is closely related to an algorithm, also commonly referred to as SLQ (but called gSLQ in this paper to avoid confusion), for approximating spectral sums $\mathrm{tr}(f[\mathbf{A}]) = \sum_{i=1}^{n} f(\lambda_i[\mathbf{A}])$ (Bai et al., 1996; Bai & Golub, 1996). Indeed, the SLQ studied here is a special case of gSLQ, with $f(s) = \mathbb{1}[s \leq x]$. However, SLQ is in fact equivalent to gSLQ the sense that the gSLQ approximation to $\mathrm{tr}(f[\mathbf{A}])$ can be obtained by computing a Riemann–Stieltjes integral of f against the output of Algorithm 1. More generally, matrix function trace estimation is closely related to CESM estimation due to the fact that

$$\mathrm{tr}(f[\mathbf{A}]) = n \int f(s)\mathrm{d}\Phi(s).$$

In (Ubaru et al., 2017a), the convergence of gSLQ when f is smooth or analytic is studied. As a result, this analysis is not immediately applicable to CESM estimation itself, since $\mathbb{1}[\cdot \leq x]$ is discontinuous. One possibility is to solve a relaxed problem where $\mathbb{1}[\cdot \leq x]$ is replaced with a smoothed approximation such as a shifted hyperbolic tangent or the CDF of any continuous random variable with small enough variance. This results in an approximation to the CESM equivalent to the convolution of the CESM with a smoothing kernel (Ubaru et al., 2017a; Han et al., 2017; Ghorbani et al., 2019).

If the CESM can be reasonably smoothed, then such an approach works well. However, it is often the case that the "variance" of the smoothing kernel, in order to preserve certain aspects of the CESM, such as large jumps due to clustered eigenvalues, has to be very small. In such cases, Gaussian quadrature bounds for smooth functions are often useless, necessitating bounds such as the ones presented in this paper; see Supplement C for a detailed discussion.

1.4. Preliminaries

Matrices are denoted by bold uppercase letters and vectors are denoted by bold lowercase letters. The first canonical unit vector $[1, 0, \ldots, 0]^{\mathsf{T}}$, of size determined by context, is denoted $\hat{\mathbf{e}}$. The set of all eigenvalues of a $d \times d$ symmetric matrix $\mathbf{B}$ is denoted $\Lambda[\mathbf{B}]$, and the individual eigenvalues are $\lambda_{\min}[\mathbf{B}] = \lambda_d[\mathbf{B}] \leq \cdots \leq \lambda_1[\mathbf{B}] = \lambda_{\max}[\mathbf{B}]$. Unless otherwise stated, $\mathbf{A}$ is $n \times n$ symmetric matrix.

We denote the i-th entry of a vector $\mathbf{v}$ by $[\mathbf{v}]_i$ and the submatrix consisting of rows r to r' and columns c to c' by $[\mathbf{B}]_{r:r',c:c'}$. If any of these indices are equal to 1 or n, they may be omitted. If $r = r'$ or $c = c'$, then we will simply write this index once. Thus, $[\mathbf{B}]_{:,:2}$ denotes the first two columns of $\mathbf{B}$, and $[\mathbf{B}]_{3,:}$ denotes the third row of $\mathbf{B}$.

For some positive integer n_{v} and a set of values $\{x_i\}_{i=1}^{n_{\mathrm{v}}}$, the sample average $\frac{1}{n_{\mathrm{v}}}\sum_{i=1}^{n_{\mathrm{v}}} x_i$ is denoted $\langle x_i \rangle$.

Definition 1. *Let μ and ν be two probability distribution functions. We say the moments of μ and ν are equal up to degree $k-1$ if for all polynomials p of degree $< k$,*

$$\int p(x)\mathrm{d}\mu(x) = \int p(x)\mathrm{d}\nu(x).$$

We also have the standard definition of Kolmogorov-Smirnov and Wasserstein distances.

Definition 2. *Let μ and ν be two probability distribution functions. The Kolmogorov–Smirnov distance between μ and ν, denoted $d_{\mathrm{KS}}(\mu, \nu)$, is defined by*

$$d_{\mathrm{KS}}(\mu, \nu) := \sup_x |\mu(x) - \nu(x)|.$$

Definition 3. *Let μ and ν be two probability distribution functions. The Wasserstein (earth mover) distance between μ and ν, denoted $d_{\mathrm{W}}(\mu, \nu)$, is defined by*

$$d_{\mathrm{W}}(\mu, \nu) := \int |\mu(x) - \nu(x)|\mathrm{d}x.$$

2. The Lanczos algorithm

The primary computational cost of SLQ is due to the Lanczos algorithm (Lanczos, 1950). The Lanczos algorithm is typically viewed as a procedure for constructing an orthonormal basis $[\mathbf{Q}]_{:,:k} := [\mathbf{q}_1, \dots, \mathbf{q}_k]$ for the Krylov subspace

$$\mathcal{K}_k(\mathbf{A}, \mathbf{v}) = \operatorname{span}(\mathbf{v}, \mathbf{A}\mathbf{v}, \dots, \mathbf{A}^{k-1}\mathbf{v}).$$

This can be done by a Gram–Schmidt-like process, where $\mathbf{A}\mathbf{q}_k$ is orthogonalized against previous basis vectors $\{\mathbf{q}_j\}_{j=1}^k$, which results in a factorization

$$\mathbf{A}[\mathbf{Q}]_{:,:k} = [\mathbf{Q}]_{:,:k}[\mathbf{T}]_{:k,:k} + \beta_k \mathbf{q}_{k+1}\mathbf{e}_k^\mathsf{T}$$

where $[\mathbf{T}]_{:k,:k}$ is upper-Hessenberg. However, since $[\mathbf{T}]_{:k,:k} = [\mathbf{Q}]_{:,:k}^\mathsf{T}\mathbf{A}[\mathbf{Q}]_{:,:k}$ is symmetric, then $[\mathbf{T}]_{:k,:k}$ is actually tridiagonal. Thus, $\mathbf{A}\mathbf{q}_k$ will be orthogonal to $\mathbf{q}_j$, $j < k-1$, so we only need to orthogonalize against $\mathbf{q}_k$ and $\mathbf{q}_{k-1}$ in each iteration. As a result, the runtime of Algorithm 2 is $O(k(T_{\text{mv}} + n))$ and the required storage is $O(n)$.

Algorithm 2 Lanczos

input $\mathbf{A}$,$\mathbf{v}$,k
  $\mathbf{q}_0 = \mathbf{0}$, $\beta_{-1} = 0$, $\mathbf{q}_1 = \mathbf{v}/\|\mathbf{v}\|$
  for $i = 0, 1, \dots, k-1$ **do**
    $\tilde{\mathbf{q}}_{i+1} = \mathbf{A}\mathbf{q}_i - \beta_{i-1}\mathbf{q}_{i-1}$
    $\alpha_i = \langle \tilde{\mathbf{q}}_{i+1}, \mathbf{q}_i \rangle$
    $\tilde{\mathbf{q}}_{i+1} = \tilde{\mathbf{q}}_{i+1} - \alpha_i \mathbf{q}_i$
    if 'reorthogonlization' **then** orthogonalize $\tilde{\mathbf{q}}_{i+1}$ against $\{\mathbf{q}_j\}_{j=1}^{i-1}$ **end if**
    $\beta_i = \|\mathbf{q}_{i+1}\|$
    $\mathbf{q}_{i+1} = \tilde{\mathbf{q}}_{i+1}/\beta_i$
  end for
return $[\mathbf{T}]_{:k,:k}$ (diagonal $[\alpha_1, \dots, \alpha_k]$ and the sub and super diagonal $[\beta_1, \dots, \beta_{k-1}]$)

Remark 1. If Algorithm 2 is run for $k = n$ iterations on any right hand side with nonzero projection onto each eigenvector, the tridiagonal matrix $[\mathbf{T}]_{:n,:n}$ produced will have the same eigenvalues as $\mathbf{A}$. Thus the CESM can be computed deterministically in time $O(nT_{\text{mv}} + n^2)$.

Remark 2 (Gu & Eisenstat, 1995). The eigenvalues and first components of eigenvectors of a real symmetric tridiagonal matrix of size $k \times k$ can be computed in $O(k^2)$ operations.

Remark 3. Without reorthogonalization in Algorithm 2, the runtime of SLQ Algorithm 1 is $O(n_\text{v} k(T_{\text{mv}} + n))$ and the required storage is $O(n)$. With reorthogonalization, the runtime is $O(n_\text{v} k(T_{\text{mv}} + nk))$ and the required storage is $O(nk)$.

Remark 4. In exact arithmetic, the reorthogonalization step of Algorithm 2 is unnecessary as $\tilde{\mathbf{q}}_{i+1}$ is already orthogonal to $\{\mathbf{q}_j\}_{j=1}^{i-1}$. However, in finite precision arithmetic this orthogonality may be lost. Our bounds, as well as the bounds for gSLQ (Ubaru et al., 2017a), are derived based on exact arithmetic theory, so it must be assumed that $[\mathbf{T}]_{:k,:k}$ is computed using some implementation which produces an output close to the exact arithmetic output. The easiest way to ensure this is with full reorthogonalization, although other more advanced schemes have been considered.

For the task of computing CESMs, some practitioners (Papyan, 2019) have noted the algorithm still works without reorthogonalization. In Supplement D, we provide an overview of existing analysis on the Lanczos algorithm in finite precision (Paige, 1976; 1980; Greenbaum, 1989) which provides a high level explanation as to why SLQ still works without reorthogonalization.

3. Results

We now state the main results.

Theorem 1. *Given $0 < \eta < 1$ and $t > 0$, set $n_\text{v} > 4(n+2)^{-1}t^{-2}\ln(2n\eta^{-1})$ and $k > 12t^{-1} + \frac{1}{2}$. Then, Algorithm 1 will output an estimate $\langle [\Psi_i]_k^{\text{gq}} \rangle$ satisfying,*

$$\mathbb{P}\big[d_\text{W}(\Phi[\mathbf{A}], \langle [\Psi_i]_k^{\text{gq}} \rangle) > tI[\mathbf{A}]\big] < \eta.$$

where $I[\mathbf{A}] := |\lambda_{\max}[\mathbf{A}] - \lambda_{\min}[\mathbf{A}]|$.

Theorem 1 is essentially a direct consequence of following theorem of the average of weighted CESMs and a straightforward bound on the Wasserstein distances of distribution functions with matching moments.

Theorem 2. *Given a positive integer n_v, suppose $\{\mathbf{v}_i\}_{i=1}^{n_\text{v}} \overset{\text{iid}}{\sim} \mathcal{U}(\mathcal{S}^{n-1})$ and define $\Psi_i = \Psi(\mathbf{A}, \mathbf{v}_i)$. Then, for all $x \in \mathbb{R}$ and $t > 0$,*

$$\mathbb{P}\big[\,|\Phi[\mathbf{A}](x) - \langle \Psi_i(x) \rangle| > t\big] \le 2\exp\left(-n_\text{v}(n+2)t^2\right)$$
$$\mathbb{P}\big[d_{\text{KS}}(\Phi[\mathbf{A}], \langle \Psi_i \rangle) > t\big] \le 2n\exp\left(-n_\text{v}(n+2)t^2\right).$$

Proposition 1. *Suppose μ and ν are two probability distribution functions constant on the complement of $[a, b]$ whose moments are equal up to degree s. Then,*

$$d_\text{W}(\mu, \nu) \le 2(b-a)(1 + \pi^2/2)s^{-1} < 12(b-a)s^{-1}.$$

We also provide a posteriori error guarantees which may be of practical use.

Theorem 3. *Let $\{[d_i]_j\}_{j=1}^k$ and $\{[\theta_i]_j\}_{j=1}^k$ be the squares of the first component of eigenvectors and the eigenvalues respectively of $[\mathbf{T}_i]_{:k,:k}$ from Algorithm 1. Then*

$$d_{\text{KS}}(\langle \Psi_i \rangle, \langle [\Psi_i]_k^{\text{gq}} \rangle) \le \left\langle \max_{j=1,\dots,k} [d_i]_j \right\rangle,$$

$$d_\text{W}(\langle \Psi_i \rangle, \langle [\Psi_i]_k^{\text{gq}} \rangle) \le \left\langle \sum_{j=0}^{n} \max\{[d_i]_j, [d_i]_{j+1}\}([\theta_i]_{j+1} - [\theta_i]_j) \right\rangle$$

where, and for notational convenience, we have defined $[\theta_i]_0 = a$, $[\theta_i]_{n+1} = b$, *and* $[d_i]_0 = [d_i]_{n+1} = 0$ *for some choice of* a, b *such that* $a \leq \lambda_{\min}[\mathbf{A}]$ *and* $b \geq \lambda_{\max}[\mathbf{A}]$.

Note that the Wasserstein distance bounds requires knowledge of points a, b such that $a \leq \lambda_{\min}[\mathbf{A}]$ and $b \geq \lambda_{\max}[\mathbf{A}]$. Such bounds can be computed rigorously, both a priori (Kuczyński & Woźniakowski, 1992) or a posteriori (Parlett et al., 1982). In practice, $\lambda_{\min}([\mathbf{T}_i]_{:k,:k}) \to \lambda_{\min}[\mathbf{A}]$ and $\lambda_{\max}([\mathbf{T}_i]_{:k,:k}) \to \lambda_{\max}[\mathbf{A}]$ rapidly, so the $j = 0$ and $j = k$ terms can be omitted with negligible effect.

As noted in Remark 1, the exact CESM can be computed with n matrix vector products. However, we also have the following lower bound for a specific class of matrices.

Theorem 4. *For any* $t \in (0, 1)$, *there exists a matrix* $\mathbf{A}$ *of size* $\lceil (4t)^{-1} \rceil$ *such that if Algorithm 1 uses fewer than* $(8t)^{-1}$ *matrix vector products, then Algorithm 1 will output an estimate* $\langle [\Psi_i]_k^{\text{gq}} \rangle$ *satisfying,*

$$d_{\text{W}}(\Phi[\mathbf{A}], \langle [\Psi_i]_k^{\text{gq}} \rangle) > tI[\mathbf{A}].$$

While Theorems 3 and 4 involve random variables, the results hold surely and therefore with probability one.

4. Analysis and proofs

For notational convenience, we denote $\Phi[\mathbf{A}]$ by Φ in proofs.

4.1. Weighted CESM

We start with analyzing the weighted CESM. Note that this analysis is applicable to many algorithms for spectrum approximation, including the KPM.

Lemma 1. *Suppose* $\mathbf{v} \sim \mathcal{U}(\mathcal{S}^{n-1})$ *and define* $m = n\Phi[\mathbf{A}](x)$. *Then,*

$$\Psi[\mathbf{A}, \mathbf{v}](x) \sim \text{Beta}\left(\frac{m}{2}, \frac{n-m}{2}\right).$$

Proof. Let $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_n]$, where $\mathbf{u}_i$ is the i-th normalized eigenvector of $\mathbf{A}$. Since $\mathbf{U}$ is orthogonal, by the invariance of $\mathcal{U}(\mathcal{S}^{n-1})$ under orthogonal transforms, we have that $\mathbf{U}^{\mathsf{T}}\mathbf{v} \sim \mathcal{U}(\mathcal{S}^{n-1})$.

We may therefore assume $\mathbf{U}^{\mathsf{T}}\mathbf{v} \overset{\text{d}}{=} \mathbf{x}/\|\mathbf{x}\|$, where $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Recall that the i-th weight of $\Psi[\mathbf{A}, \mathbf{v}]$ is given by $w_i = (\mathbf{v}^{\mathsf{T}}\mathbf{u}_i)^2$. Thus, the w_i have joint distribution given by,

$$w_i \overset{\text{d}}{=} \left(\frac{[\mathbf{x}]_i}{\|\mathbf{x}\|}\right)^2 = \frac{([\mathbf{x}]_i)^2}{([\mathbf{x}]_1)^2 + \cdots + ([\mathbf{x}]_n)^2},$$

for $i = 1, \ldots, n$.

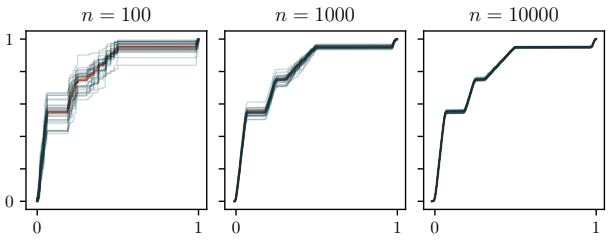

Figure 1. Concentration of 30 independent samples of the weighted CESM $\Psi[\mathbf{A}, \mathbf{v}]$ (——) about the CESM $\Phi[\mathbf{A}]$ (——) for matrices of different sizes constructed with qualitatively similar spectrums. *Remarks*: (i) the light lines are samples of a random variable with expectation given by the dark line, (ii) samples of this random variable define cumulative probability densities, and (iii) this random variable concentrates exponentially about the CESM as n increases.

Recall $m = n\Phi[\mathbf{A}](x)$. Then,

$$\Psi[\mathbf{A}, \mathbf{v}](x) = \sum_{j=1}^{m} w_j \overset{\text{d}}{=} \frac{([\mathbf{x}]_1)^2 + \cdots + ([\mathbf{x}]_m)^2}{([\mathbf{x}]_1)^2 + \cdots + ([\mathbf{x}]_n)^2}.$$

It is well known that for independent chi-square random variables $Y \sim \chi^2_\alpha$ and $Z \sim \chi^2_\beta$ (see, for example, (Johnson et al., 1994, Section 25.2)),

$$\frac{Y}{Y+Z} \sim \text{Beta}\left(\frac{\alpha}{2}, \frac{\beta}{2}\right).$$

Thus, since $([\mathbf{x}]_1)^2 + \cdots + ([\mathbf{x}]_m)^2$ and $([\mathbf{x}]_{m+1})^2 + \cdots + ([\mathbf{x}]_n)^2$ are independent chi-square random variables with m and $n - m$ degrees of freedom respectively, $\Psi[\mathbf{A}, \mathbf{v}](x)$ is a beta random variable with parameters $m/2$ and $(n - m)/2$. □

As seen in Figure 1, $\Psi[\mathbf{A}, \mathbf{v}](x)$ concentrates about its mean $\Phi[\mathbf{A}](x)$ as n increases. To understand this more precisely, we introduce the following definition and its consequences.

Definition 4. *A random variable* X *is* σ^2*-sub-Gaussian if*

$$\mathbb{E}\left[\exp(\lambda(X - \mathbb{E}[X]))\right] \leq \exp\left(\frac{\lambda^2\sigma^2}{2}\right), \ \forall \lambda \in \mathbb{R}.$$

Lemma 2. *Suppose* X *is* σ^2*-sub-Gaussian. Let* $X_1, \ldots, X_{n_{\text{v}}}$ *be iid samples of* X. *Then for all* $t \geq 0$,

$$\mathbb{P}\left[|\langle X_i \rangle - \mathbb{E}[X]| > t\right] \leq 2\exp\left(-\frac{n_{\text{v}}}{2\sigma^2}t^2\right).$$

Theorem 5 (Marchal & Arbel, 2017, Theorem 1)**.** *Suppose* $X \sim \text{Beta}(\alpha, \beta)$. *Then,* $\mathbb{E}[X] = \alpha/(\alpha + \beta)$, *and* X *is* $(4(\alpha + \beta + 1))^{-1}$*-sub-Gaussian. If* $\alpha = \beta$, *then there is no smaller* σ^2 *such that* X *is* σ^2*-sub-Gaussian.*

With these results in place, the proof of Theorem 2 is straightforward.

Proof of Theorem 2. First note that the maximums exist because Φ and $\langle \Psi_i \rangle$ are right continuous and piecewise constant except at $\{\lambda_i[\mathbf{A}]\}_{i=1}^n$.

For any x, let $m = m(x) = n\Phi(x)$. Using Lemmas 1 and 2 and Theorem 5 we have that for any x,

$$\mathbb{P}\left[\, |\Phi(x) - \langle \Psi_i(x) \rangle| > t \,\right] \leq 2\exp\left(-\frac{n_{\mathrm{v}}}{2(4(\frac{m}{2} + \frac{n-m}{2} + 1))^{-1}} t^2 \right).$$

We also have

$$\sup_{x \in \mathbb{R}} |\Phi(x) - \langle \Psi_i(x) \rangle| = \max_{i=1,\ldots,n-1} |\Phi(\lambda_i[\mathbf{A}]) - \langle \Psi_i(\lambda_i[\mathbf{A}]) \rangle|.$$

The second result follows by applying a union bound to the events that the maximum is attained at $\lambda_i[\mathbf{A}]$ for each $i = 1, \ldots, n$. □

4.2. Gaussian quadrature

We now shift our attention to the approximation of the weighted CESM by a Gaussian quadrature rule.

Definition 5. *Let μ be a distribution function with finite moments up to degree $2k-1$. The k-point Gaussian quadrature rule for μ, is the distribution*

$$\nu(x) = \sum_{j=1}^{k} d_j \mathbb{1}[\theta_j \leq x]$$

corresponding to nodes $\{\theta_j\}_{j=1}^k$ and weights $\{d_j\}_{j=1}^k$ such that the moments of μ and ν are equal up to degree $2k-1$. We denote such a distribution function by $[\mu]_k^{\mathrm{gq}}$.

This definition implies the total mass of a Gaussian quadrature rule must agree with the original distribution, which in the context of computing approximations to the weighted CESM, means that the SLQ approximation remains a probability distribution function. This property is not retained by other approaches to approximating the weighted CESM such as the KPM. More generally, Proposition 1 asserts that the Wasserstein distance decays inversely with the number of matching moments.

Since, Proposition 1 holds uniformly for all probability distribution functions constant on the complement of $[a, b]$, we also recall an a posteriori characterization of the closeness of distribution functions with matching moments due to (Karlin & Shapley, 1972) but known implicitly far earlier (Stieltjes, 1918). Before stating this theorem, we introduce a definition and a resulting lemma.

Definition 6. *A function γ has a sign change at x if there exists $x' < x$ such that $\gamma(x') \neq 0$ and $x = \inf\{t > x' : \gamma(t)\gamma(x') < 0\}$.*

Lemma 3. *Suppose γ is weakly increasing on an interval (a, b). Then γ and has a sign change at x if and only if there exists $x' < x$ such that $\gamma(x') < 0$, $\gamma(y) \leq 0$ for all $y \in (a, x)$ and $\gamma(y) > 0$ for all $y \in (x, b)$.*

Theorem 6 (Karlin & Shapley, 1972, Theorem 22.1)**.** *Suppose μ and ν are two probability distribution functions constant on the complement of $[a, b]$ whose moments are equal up to degree s. Define $\gamma : [a, b] \to [0, 1]$ by $\gamma(x) = \mu(x) - \nu(x)$. Then γ is identically zero or changes sign at least s times.*

Note that for a probability distribution function, $[\mu]_k^{\mathrm{gq}}$ is piecewise constant with k points of discontinuity. Using the fact that $[\mu]_k^{\mathrm{gq}}$ and μ share moments up to degree $2k-1$ along with Theorem 6, we immediatley obtain the following bounds on $[\mu]_k^{\mathrm{gq}}$ (proved in Supplement A for completeness).

Corollary 1. *Suppose μ is a probability distribution function constant on the complement of $[a, b]$. Let $\{\theta_j\}_{j=1}^k$ and $\{d_j\}_{j=1}^k$ respectively be the nodes and weights of the Guassian quadrature rule $[\mu]_k^{\mathrm{gq}}$. Define $[\mu]_k^{\downarrow}$ and $[\mu]_k^{\uparrow}$ by*

$$[\mu]_k^{\downarrow}(x) := \sum_{j=1}^{k-1} d_j \mathbb{1}\left[\theta_{j+1} \leq x\right],$$

$$[\mu]_k^{\uparrow}(x) := d_1 + \sum_{j=2}^{k} d_j \mathbb{1}\left[\theta_{j-1} \leq x\right].$$

Then, for all $x \in [a, b]$,

$$[\mu]_k^{\downarrow}(x) \leq \mu(x) \leq [\mu]_k^{\uparrow}(x).$$

In turn, Corollary 1 implies bounds on the Wasserstein and Kolmogorov–Smirnov distances between μ and $[\mu]_k^{\mathrm{gq}}$.

Corollary 2. *Suppose μ is a probability distribution function constant on the complement of $[a, b]$. Let $\{\theta_j\}_{j=1}^k$ and $\{d_j\}_{j=1}^k$ respectively be the nodes and weights of the Guassian quadrature rule $[\mu]_k^{\mathrm{gq}}$. Then*

$$d_{\mathrm{KS}}(\mu, [\mu]_k^{\mathrm{gq}}) \leq \max_{j=1,\ldots,k} d_j$$

$$d_{\mathrm{W}}(\mu, [\mu]_k^{\mathrm{gq}}) \leq \sum_{j=0}^{k} \max\{d_j, d_{j+1}\}(\theta_{j+1} - \theta_j)$$

where we define $\theta_0 = a$, $\theta_{k+1} = b$, and $d_0 = d_{k+1} = 0$.

Finally, we note the classical result that Lanczos algorithm computes a Gaussian quadrature rule for $\Psi[\mathbf{A}, \mathbf{v}]$ (Gautschi, 2004; Golub & Meurant, 2009).

Proposition 2. *Let $[\mathbf{T}]_{:k,:k}$ be the output of Algorithm 2 run on $\mathbf{A}$, $\mathbf{v}$ for k steps. Then the eigenvalues of $[\mathbf{T}]_{:k,:k}$ and the square of the first components of the eigenvectors of $[\mathbf{T}]_{:k,:k}$ form a degree k Gaussian quadrature rule for μ. That is, $[\Psi[\mathbf{A}, \mathbf{v}]]_k^{\mathrm{gq}} = \Psi[[\mathbf{T}]_{:k,:k}, \hat{\mathbf{e}}]$.*

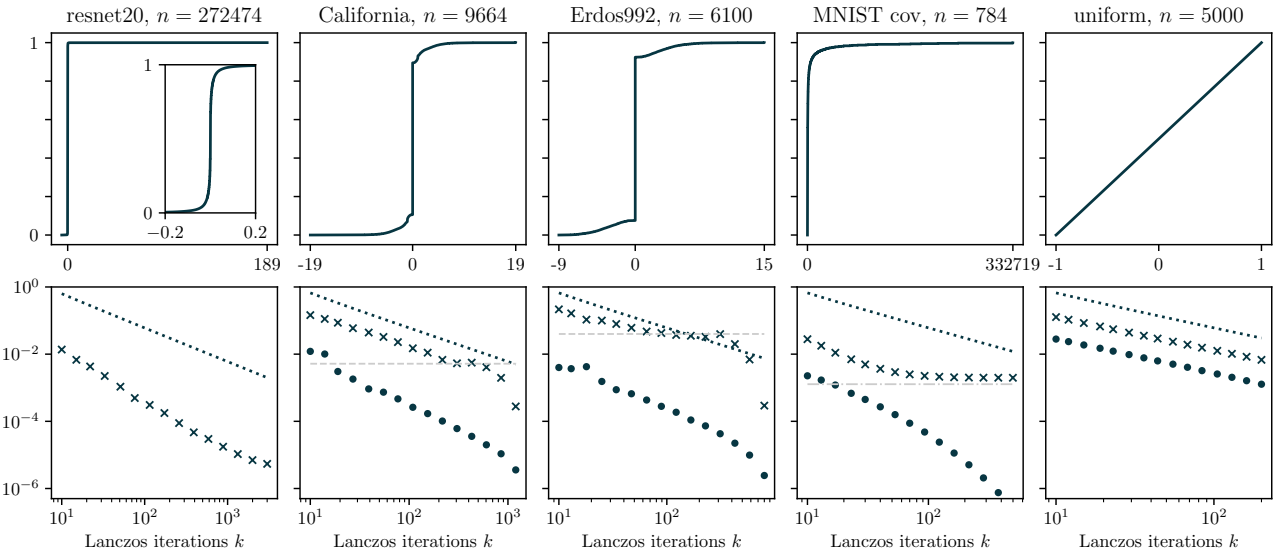

Figure 2. Top: distribution functions *Bottom*: Wasserstein error and error bounds. All problems are scaled so that $I[\mathbf{A}] = 1$ for easier comparison. From left to right, $n_{\mathrm{v}} = 2, 6, 9, 68, 11$ chosen so that n_{v} is roughly of size $O(n^{-1})$. *Legend*: $d_{\mathrm{W}}(\Phi[\mathbf{A}], \langle[\Psi_i]_k^{\mathrm{gq}}\rangle$ (•), bound $\langle\sum_{j=0}^{n} \max\{[d_i]_j, [d_i]_{j+1}\}([\theta_i]_{j+1} - \theta_i]_j)\rangle$ (×), bound $12I[\mathbf{A}](2k-1)^{-1}$ (·····), $(\Phi(d^-) - \Phi(c))\,|d-c|$ described in (2) (– – –), $I[\mathbf{A}]n^{-1}$ (—·—).

Since k is typically much smaller than n, and since $[\mathbf{T}]_{:k,:k}$ is tridiagonal, then the exact weighted CESM $\Psi[[\mathbf{T}]_{:k,:k}, \hat{\mathbf{e}}]$ can be computed directly. This allows for the efficient computation of Gaussian quadrature rules for $\Psi[\mathbf{A}, \mathbf{v}]$.

4.3. Remaining proofs

Proof of Theorem 1. Note that for any probability distribution functions μ and ν constant on the complement of $[a, b]$,

$$d_{\mathrm{W}}(\mu, \nu) \leq (b-a) d_{\mathrm{KS}}(\mu, \nu).$$

For $i = 1, \ldots, n_{\mathrm{v}}$, define Ψ_i as in Algorithm 1. Then, using Theorem 2,

$$\mathbb{P}[d_{\mathrm{KS}}(\Phi, \langle\Psi_i\rangle) > t/2] \leq 2n \exp(-(n+2)n_{\mathrm{v}}t^2/4),$$

so since $\langle\Psi_i\rangle$ and Φ are constant on the complement of $[\lambda_{\min}[\mathbf{A}], \lambda_{\max}[\mathbf{A}]]$,

$$\mathbb{P}[d_{\mathrm{W}}(\Phi, \langle\Psi_i\rangle) > tI[\mathbf{A}]/2] \leq 2n \exp\left(-n_{\mathrm{v}}(n+2)t^2/4\right).$$

By Proposition 1 and the definition of Gaussian quadrature rule we have, with probability one,

$$d_{\mathrm{W}}([\Psi_i]_k^{\mathrm{gq}}, \Psi_i) < 12I[\mathbf{A}](2k-1)^{-1}$$

for $i = 1, \ldots, n_{\mathrm{v}}$. Thus, by the triangle inequality, again with probability one,

$$d_{\mathrm{W}}(\langle[\Psi_i]_k^{\mathrm{gq}}\rangle, \langle\Psi_i\rangle) < 12I[\mathbf{A}](2k-1)^{-1}.$$

Finally, we apply the triangle inequality to obtain,

$$d_{\mathrm{W}}(\Phi, \langle[\Psi_i]_k^{\mathrm{gq}}\rangle) \leq d_{\mathrm{W}}(\Phi, \langle\Psi_i\rangle) + d_{\mathrm{W}}(\langle\Psi_i\rangle, \langle[\Psi_i]_k^{\mathrm{gq}}\rangle)$$

Setting $n_{\mathrm{v}} > 4(n+2)^{-1}t^{-2}\log(2n\eta^{-1})$ and $k > 12t^{-1} + \frac{1}{2}$ ensures the sum of the two terms is at most $I[\mathbf{A}]t$ with probability at least $1 - \eta$. □

Proof of Theorem 3. This is a direct consequence of Corollary 2 and the triangle inequality. □

Proof of Theorem 4. Let $\Upsilon(x) = x$ on $[0, 1]$ be the probability distribution function for a uniform density on $[0, 1]$.

First, for any K, non-negative weights $\{d_i\}_{i=1}^K$ summing to one and ordered points $\{\theta_i\}_{i=1}^K$ in $[0, 1]$, define $\{D_i\}_{i=0}^K$ by $D_i = \sum_{j=1}^{i} d_i$ (where $D_0 = 0$) and consider the functions

$$\varphi(x) = \sum_{i=1}^{K} d_i \mathbb{1}[\theta_i \leq x],$$

$$\tilde{\varphi}(y) = \theta_1 + \sum_{i=1}^{K-1} (\theta_{i+1} - \theta_i)\mathbb{1}[D_i \leq y].$$

Note that,

$$d_{\mathrm{W}}(\Upsilon, \varphi) = \int_0^1 |\varphi(x) - x| \mathrm{d}x = \int_0^1 |\tilde{\varphi}(y) - y| \mathrm{d}y.$$

Next, define

$$\begin{aligned}\mathcal{C} =&\{\psi : \psi(0) = 0, \psi(1) = 1,\\ &\psi'(y) = 1 \,\forall y \in (D_i, D_{i+1}), i = 0, \dots, K\}\end{aligned}$$

and observe that, because the contribution on each subinterval is independent of other subintervals,

$$\begin{aligned}d_{\mathrm{W}}(\Upsilon, \varphi) &\geq \min_{\psi\in\mathcal{C}} \int_0^1 |\tilde{\varphi}(y) - \psi(y)| \mathrm{d}y\\ &= \min_{\psi\in\mathcal{C}} \sum_{i=0}^{K-1} \int_{D_i}^{D_{i+1}} |\tilde{\varphi}(y) - \psi(y)| \mathrm{d}y\\ &= \sum_{i=0}^{K-1} \min_{\psi\in\mathcal{C}} \int_{D_i}^{D_{i+1}} |\tilde{\varphi}(y) - \psi(y)| \mathrm{d}y\\ &= \sum_{i=0}^{K-1} \left(\frac{D_{i+1} - D_i}{2}\right)^2.\end{aligned}$$

Thus, by the Cauchy–Schwarz inequality,

$$\begin{aligned}\frac{1}{4} = \left(\sum_{i=0}^{K-1} \frac{d_i}{2}\right)^2 &\leq \left(\sum_{i=0}^{K-1} 1^2\right)\left(\sum_{i=0}^{K-1} \left(\frac{D_{i+1} - D_i}{2}\right)^2\right)\\ &= K \sum_{i=0}^{K-1} \left(\frac{D_{i+1} - D_i}{2}\right)^2\end{aligned}$$

so, since $y \mapsto y \in \mathcal{C}$, we have that

$$\begin{aligned}d_{\mathrm{W}}(\Upsilon, \varphi) &= \int_0^1 |\tilde{\varphi}(y) - y| \mathrm{d}y\\ &\geq \min_{\psi\in\mathcal{C}} \int_0^1 |\tilde{\varphi}(y) - \psi(y)| \mathrm{d}y \geq \frac{1}{4K}.\end{aligned}$$

We now construct a matrix whose CESM has small Wasserstein distance to Υ. Let $n = \lceil (4t)^{-1} \rceil$ and define a matrix $\mathbf{A}$ with eigenvalues $\{(2n)^{-1} + kn^{-1} : k = 0, 1, \dots n-1\}$. By the above argument, noting the the Cauchy–Schwarz inequality is an equality of all terms in the sum are equal, it is clear that,

$$d_{\mathrm{W}}(\Upsilon, \Phi) = \frac{1}{4n} < t.$$

Now, note that $\langle [\Psi_i]_k^{\mathrm{gq}} \rangle$ is of the form of φ with $K = n_{\mathrm{v}} k$. Thus, for any n_{v}, k such that $n_{\mathrm{v}} k < (8t)^{-1}$, with probability one,

$$d_{\mathrm{W}}(\Upsilon, \langle [\Psi_i]_k^{\mathrm{gq}} \rangle) \geq \frac{1}{4 n_{\mathrm{v}} k} > 2t.$$

Then, using the triangle inequality, again with probability one,

$$d_{\mathrm{W}}(\Phi, \langle [\Psi_i]_k^{\mathrm{gq}} \rangle) \geq d_{\mathrm{W}}(\Upsilon, \langle [\Psi_i]_k^{\mathrm{gq}} \rangle) - d_{\mathrm{W}}(\Upsilon, \Phi) > t.$$

Since $n_{\mathrm{v}} k$ is the number of matrix vector products required by Algorithm 1, and $I(\mathbf{A}) < 1$, the result holds. □

Note that this proof constructs two distribution functions with matching moments up to degree k whose Wasserstein distance is $\Omega(k^{-1})$. This immediately implies that if the output of an algorithm used to approximate distribution functions depends only on the first k moments, there exist inputs on which the output error has Wasserstein distance $\Omega(k^{-1})$.

5. Numerical verification and discussion

We demonstrate the effectiveness of our bounds on several test problems. The convergence of $\langle \Psi_i \rangle$ to $\Phi[\mathbf{A}]$ is straightforward, so we focus on the convergence of the Gaussian quadrature rules $[\Psi_i]_k^{\mathrm{gq}}$ of Ψ_i.

Here, "resnet20" is a Hessian for the ResNet20 network (He et al., 2016) trained on the Cifar-10 dataset. To apply the Lanczos algorithm to this example, we use a slightly modified version of PyHessian (Yao et al., 2020). The "California" and "Erdos992" examples are graph adjacency matrices from the sparse matrix suite (Davis & Hu, 2011), the "MNIST cov" example is the covariance matrix of the MNIST training data, and "uniform" is a synthetic problem with 5000 eigenvalues evenly spaced between -1 and 1.

Our first example studies the global convergence of $\langle [\Psi_i]_k^{\mathrm{gq}} \rangle$ to $\langle \Psi_i \rangle$ as the number of Lanczos iterations k increases. Specifically, we consider the upper bounds $d_{\mathrm{W}}(\langle \Psi_i \rangle, \langle [\Psi_i]_k^{\mathrm{gq}} \rangle) \leq 12 I[\mathbf{A}](2k-1)^{-1}$ from the proof of Theorem 1 and the bound $d_{\mathrm{W}}(\langle \Psi_i \rangle, \langle [\Psi_i]_k^{\mathrm{gq}} \rangle) \leq \langle \sum_{j=0}^{k} \max\{[d_i]_j, [d_i]_{j+1}\}([\theta_i]_{j+1} - [\theta_i]_j) \rangle$ from Theorem 3. These bounds, with the true extreme eigenvalues of $\mathbf{A}$ used for $[\theta_i]_0$ and $[\theta_i]_{k+1}$ (except for on the "resnet20" example where these terms are omitted), are shown illustrated in Figure 2 for several test problems. Qualitatively, we observe several types of behavior in both the true Wasserstein distance and the bounds.

First, when k is small relative to n, the convergence rate is similar to $O(k^{-1})$. This behavior is especially visible on the "uniform" example, where the true CESM is relatively "smooth" and aligns with the intuition behind our lower bound Theorem 4. On the other hand, as observed on the "MNIST cov" example, when k becomes sufficiently large, the convergence accelerates past $O(k^{-1})$. This is also unsurprising since when $k = n$, $[\Psi_i]_k^{\mathrm{gq}} = \Psi_i$.

Second, we observe the Wasserstein distance bound from Theorem 3 sometimes stagnates. There are two causes for this. The first cause of stagnation, observed on the "MNIST cov" example, is due to the fact that the bound from Theorem 3 will never be smaller than $I[\mathbf{A}]n^{-1}$. The second source of stagnation, due to many tightly clustered eigenvalues, is observed in the "California" and "Erdos992"

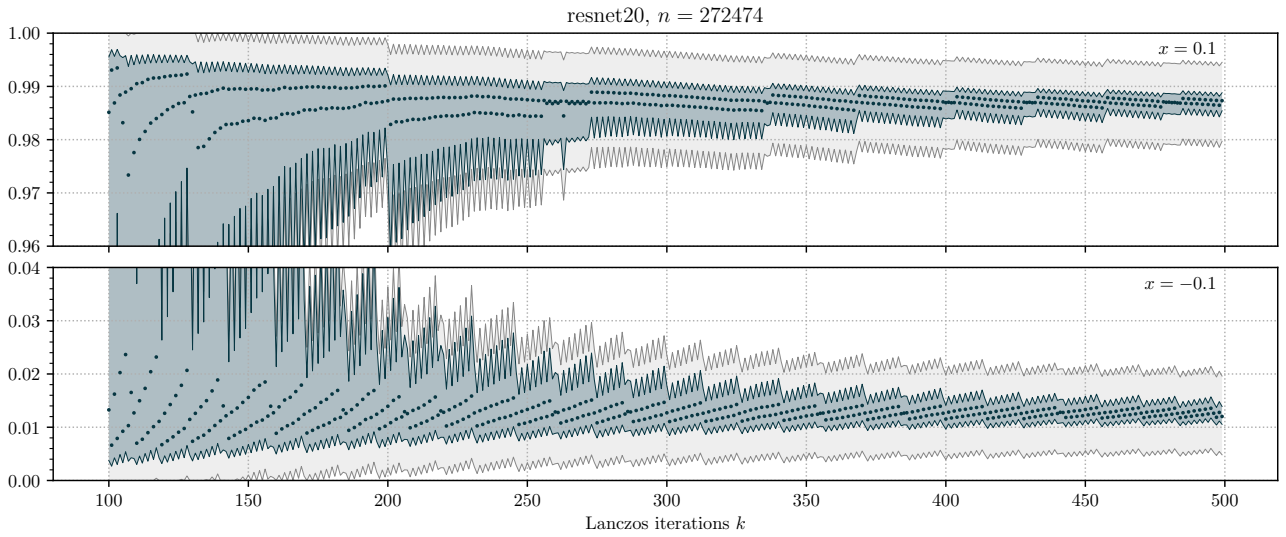

Figure 3. Bounds for $\Phi[\mathbf{A}](x)$ with $\eta = 0.01$. The average weighted CESM $\langle \Psi_i(x) \rangle = \langle \Psi(\mathbf{A}, \mathbf{v}_i)(x) \rangle$ is bounded between the averaged lower and upper bounds $\langle [\Psi_i]_k^{\downarrow} \rangle$ and $\langle [\Psi_i]_k^{\uparrow} \rangle$ (▭) from Corollary 1. By Theorem 2, the CESM $\Phi[\mathbf{A}](x)$ is within t of the average weighted CESM with probability at least $1 - \eta$, and therefore lies betwen $\langle [\Psi_i]_k^{\downarrow}(x) \rangle - t$ and $\langle [\Psi_i]_k^{\uparrow}(x) \rangle + t$ (▭) with this same probability. The output of Algorithm 1 $\langle [\Psi_i]_k^{\text{gq}} \rangle$ (•) is shown for reference.

examples. To understand this stagnation, suppose there are many eigenvalues clustered near x; i.e. in the interval $(x - \epsilon, x + \epsilon)$ for some small ϵ. Let,

$$c := \max_{\lambda \in \Lambda[\mathbf{A}]} \{\lambda < x - \epsilon\}, \quad d := \min_{\lambda \in \Lambda[\mathbf{A}]} \{\lambda > x + \epsilon\}.$$

so that there are no eigenvalues in the $(c, d) \setminus (x - \epsilon, x + \epsilon)$. Then, if $|d - c| \gg \epsilon$ and k is not too large, the Gaussian quadrature rule will have only one node in (c, d) located very near to x. As a result, the bound will stagnate near

$$(\Phi(d^-) - \Phi(c))\, |d - c|. \tag{2}$$

If the cluster of eigenvalues are all exactly equal so ϵ can be taken to be zero, this stagnation will persist for all k. If they are not, then eventually the Gaussian quadrature rule will place more nodes in this interval and the bound will recover, as observed in both examples. In Supplement B.1, we discuss a heuristic approach to address this stagnation.

Our second example studies bounds for $\Phi[\mathbf{A}](x)$ for a fixed value of x. These can be used to obtain upper and lower bounds for the number of eigenvalues in an interval. Specifically, we consider the lower and upper bounds $\langle [\Psi_i]_k^{\downarrow}(x) \rangle - t$ and $\langle [\Psi_i]_k^{\uparrow}(x) \rangle + t$, which provide probabilistic upper and lower bounds according to Theorem 2 and Corollary 2. In Figure 3, for fixed values of x, we plot the bounds as a function of the number of Lanczos iterations k for the "resnet20" example and as a function of x. Together, these plots imply that with probability $99/100$, roughly 94-99% of the eigenvalues are in the interval $[-0.1, 0.1]$.

6. Outlook

The analysis in this paper gives rigorous bounds on the accuracy of SLQ for spectrum approximation. These bounds are suited to the parameter ranges encountered in practice and demonstrate that SLQ is a viable method for spectrum approximation in many applications. As a result, we hope our analysis will allow practitioners to obtain more precise and theoretically justifiable insights about their applications without the need for heuristics.

More broadly, SLQ and KPM fall into a larger class of spectrum approximation algorithms which approximate iid samples of the weighted CESM using information from Krylov subspaces. However, the exact relationship between these algorithms has not been fully described, making the tradeoffs between the algorithms murky at best. In order to shed light on the tradeoffs between these algorithms, a comprehensive treatment providing a *unified* perspective is needed.

References

Adams, R. P., Pennington, J., Johnson, M. J., Smith, J., Ovadia, Y., Patton, B., and Saunderson, J. Estimating the spectral density of large implicit matrices, 2018.

Avron, H. Counting triangles in large graphs using randomized matrix trace estimation. In *Proceedings of KDD-LDMTA*, 2010.

Avron, H. and Toledo, S. Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix. *Journal of the ACM*, 58(2):1–34, April 2011. doi: 10.1145/1944345.1944349.

Bai, Z. and Golub, G. Bounds for the trace of the inverse and the determinant of symmetric positive definite matrices. *Annals of Numerical Mathematic*, 4:29–38, 4 1996.

Bai, Z., Fahey, G., and Golub, G. Some large-scale matrix computation problems. *Journal of Computational and Applied Mathematics*, 74(1-2):71–89, November 1996. doi: 10.1016/0377-0427(96)00018-0.

Benoit, C., Royer, E., and Poussigue, G. The spectral moments method. *Journal of Physics: Condensed Matter*, 4 (12):3125–3152, March 1992. doi: 10.1088/0953-8984/4/12/010.

Braverman, V., Krishnan, A., and Musco, C. Linear and sublinear time spectral density estimation, 2021.

Cohen-Steiner, D., Kong, W., Sohler, C., and Valiant, G. Approximating the spectrum of a graph. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, July 2018. doi: 10.1145/3219819.3220119.

Covaci, L., Peeters, F. M., and Berciu, M. Efficient numerical approach to inhomogeneous superconductivity: The chebyshev-bogoliubov–de gennes method. *Physical Review Letters*, 105(16), October 2010. doi: 10.1103/physrevlett.105.167006.

Davis, T. A. and Hu, Y. The university of florida sparse matrix collection. *ACM Transactions on Mathematical Software*, 38(1):1–25, November 2011. doi: 10.1145/2049662.2049663.

Deift, P. and Trogdon, T. The conjugate gradient algorithm on well-conditioned wishart matrices is almost deterministic. *Quarterly of Applied Mathematics*, pp. 1, July 2020. doi: 10.1090/qam/1574.

Dong, K., Benson, A. R., and Bindel, D. Network density of states. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, July 2019. doi: 10.1145/3292500.3330891.

Ducastelle, F. and Cyrot-Lackmann, F. Moments developments and their application to the electronic charge distribution of d bands. *Journal of Physics and Chemistry of Solids*, 31(6):1295–1306, June 1970. doi: 10.1016/0022-3697(70)90134-4.

Fan, L., Shuman, D. I., Ubaru, S., and Saad, Y. Spectrum-adapted polynomial approximation for matrix functions. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, May 2019. doi: 10.1109/icassp.2019.8683179.

Fischer, B. *Polynomial Based Iteration Methods for Symmetric Linear Systems*. Society for Industrial and Applied Mathematics, January 2011. doi: 10.1137/1.9781611971927.

Folland, G. B. *Real analysis: modern techniques and their applications*. Pure and applied mathematics. Wiley, 2nd ed edition, 1999. ISBN 9780471317166.

Garza-Vargas, J. and Kulkarni, A. The lanczos algorithm under few iterations: Concentration and location of the output. *SIAM Journal on Matrix Analysis and Applications*, 41(3):1312–1346, January 2020. doi: 10.1137/19m1275231.

Gautschi, W. *Orthogonal polynomials: computation and approximation*. Numerical mathematics and scientific computation. Oxford University Press, 2004. ISBN 9780198506720.

Ghorbani, B., Krishnan, S., and Xiao, Y. An investigation into neural net optimization via hessian eigenvalue density. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2232–2241. PMLR, 09–15 Jun 2019.

Golub, G. H. and Meurant, G. *Matrices, moments and quadrature with applications*, volume 30. Princeton University Press, 2009.

Greenbaum, A. Behavior of slightly perturbed Lanczos and conjugate-gradient recurrences. *Linear Algebra and its Applications*, 113:7 – 63, 1989. doi: https://doi.org/10.1016/0024-3795(89)90285-1.

Greenbaum, A. *Iterative Methods for Solving Linear Systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1997.

Gu, M. and Eisenstat, S. C. A divide-and-conquer algorithm for the symmetric tridiagonal eigenproblem. *SIAM Journal on Matrix Analysis and Applications*, 16(1):172–191, 1995. doi: 10.1137/S0895479892241287.

Han, I., Malioutov, D., Avron, H., and Shin, J. Approximating spectral sums of large-scale matrices using stochastic chebyshev approximations. *SIAM Journal on Scientific Computing*, 39(4):A1558–A1585, January 2017. doi: 10.1137/16m1078148.

Han, W. and Atkinson, K. E. *Theoretical Numerical Analysis*. Springer New York, 2009. doi: 10.1007/978-1-4419-0458-4.

Haydock, R., Heine, V., and Kelly, M. J. Electronic structure based on the local atomic environment for tight-binding bands. *Journal of Physics C: Solid State Physics*, 5(20): 2845–2858, October 1972. doi: 10.1088/0022-3719/5/20/004.

Haydock, R., Heine, V., and Kelly, M. J. Electronic structure based on the local atomic environment for tight-binding bands. II. *Journal of Physics C: Solid State Physics*, 8 (16):2591–2605, August 1975. doi: 10.1088/0022-3719/8/16/011.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2016. doi: 10.1109/cvpr.2016.90.

Johnson, N. L., Kotz, S., and Balakrishnan, N. *Continuous univariate distributions*. Wiley series in probability and mathematical statistics. Wiley, 2nd ed edition, 1994. ISBN 9780471584957.

Karlin, S. and Shapley, L. S. *Geometry of moment spaces*. Memoirs of the American Mathematical Society. American Math. Soc, 3. printing edition, 1972. ISBN 9780821812129.

Kuczyński, J. and Woźniakowski, H. Estimating the largest eigenvalue by the power and lanczos algorithms with a random start. *SIAM Journal on Matrix Analysis and Applications*, 13(4):1094–1122, October 1992. doi: 10.1137/0613066.

Kuijlaars, A. B. J. Which eigenvalues are found by the Lanczos method? *SIAM Journal on Matrix Analysis and Applications*, 22(1):306–321, 2000. doi: 10.1137/S089547989935527X.

Lambin, P. and Gaspard, J. P. Continued-fraction technique for tight-binding systems. a generalized-moments method. *Physical Review B*, 26(8):4356–4368, October 1982. doi: 10.1103/physrevb.26.4356.

Lanczos, C. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. 1950.

Li, R., Xi, Y., Erlandson, L., and Saad, Y. The eigenvalues slicing library (EVSL): Algorithms, implementation, and software. *SIAM Journal on Scientific Computing*, 41(4): C393–C415, January 2019. doi: 10.1137/18m1170935.

Lin, L., Saad, Y., and Yang, C. Approximating spectral densities of large matrices. *SIAM Review*, 58(1):34–65, January 2016. doi: 10.1137/130934283.

Marchal, O. and Arbel, J. On the sub-gaussianity of the beta and dirichlet distributions. *Electronic Communications in Probability*, 22(0), 2017. ISSN 1083-589X. doi: 10.1214/17-ecp92.

Meurant, G. and Strakoš, Z. The lanczos and conjugate gradient algorithms in finite precision arithmetic. *Acta Numerica*, 15:471–542, 2006.

Musco, C., Netrapalli, P., Sidford, A., Ubaru, S., and Woodruff, D. P. Spectrum approximation beyond fast matrix multiplication: Algorithms and hardness, 2019.

Napoli, E. D., Polizzi, E., and Saad, Y. Efficient estimation of eigenvalue counts in an interval. *Numerical Linear Algebra with Applications*, 23(4):674–692, March 2016. doi: 10.1002/nla.2048.

Paige, C. C. Error Analysis of the Lanczos Algorithm for Tridiagonalizing a Symmetric Matrix. *IMA Journal of Applied Mathematics*, 18(3):341–349, 12 1976. doi: 10.1093/imamat/18.3.341.

Paige, C. C. Accuracy and effectiveness of the Lanczos algorithm for the symmetric eigenproblem. *Linear Algebra and its Applications*, 34:235 – 258, 1980. doi: 10.1016/0024-3795(80)90167-6.

Papyan, V. The full spectrum of deepnet hessians at scale: Dynamics with sgd training and sample size, 2019.

Parlet, B. N. and Scott, D. S. C. The Lanczos algorithm with selective orthogonalization. *Mathematics of Computation*, 33(145):217–238, January 1979. doi: 10.1090/s0025-5718-1979-0514820-3.

Parlett, B. N., Simon, H., and Stringer, L. M. On estimating the largest eigenvalue with the lanczos algorithm. *Mathematics of Computation*, 38(157):153–153, January 1982. doi: 10.1090/s0025-5718-1982-0637293-9.

Pearlmutter, B. A. Fast exact multiplication by the hessian. *Neural Computation*, 6(1):147–160, January 1994. doi: 10.1162/neco.1994.6.1.147.

Polizzi, E. Density-matrix-based algorithm for solving eigenvalue problems. *Physical Review B*, 79(11), March 2009. doi: 10.1103/physrevb.79.115112.

Ramesh, A. and LeCun, Y. Backpropagation for implicit spectral densities, 2018.

Roosta-Khorasani, F. and Ascher, U. Improved bounds on sample size for implicit matrix trace estimators. *Foundations of Computational Mathematics*, 15(5):1187–1212, September 2014. doi: 10.1007/s10208-014-9220-1.

Sbierski, B., Trescher, M., Bergholtz, E. J., and Brouwer, P. W. Disordered double weyl node: Comparison of transport and density of states calculations. *Physical Review B*, 95(11), March 2017. doi: 10.1103/physrevb.95.115104.

Schnack, J., Richter, J., and Steinigeweg, R. Accuracy of the finite-temperature lanczos method compared to simple typicality-based estimates. *Physical Review Research*, 2(1), February 2020. doi: 10.1103/physrevresearch.2.013186.

Stieltjes, T. J. Sur certaines ingalits dues M. P. Tchebychef. *Oeuvres completes*, 2:586–593, 1918.

Ubaru, S., Chen, J., and Saad, Y. Fast estimation of $\mathrm{tr}(f(A))$ via stochastic lanczos quadrature. *SIAM Journal on Matrix Analysis and Applications*, 38(4):1075–1099, January 2017a. doi: 10.1137/16m1104974.

Ubaru, S., Saad, Y., and Seghouane, A.-K. Fast estimation of approximate matrix ranks using spectral densities. *Neural Computation*, 29(5):1317–1351, May 2017b. doi: 10.1162/neco_a_00951.

Weiße, A., Wellein, G., Alvermann, A., and Fehske, H. The kernel polynomial method. *Reviews of Modern Physics*, 78(1):275–306, March 2006. doi: 10.1103/revmodphys.78.275.

Wheeler, J. C. and Blumstein, C. Modified moments for harmonic solids. *Physical Review B*, 6(12):4380–4382, December 1972. doi: 10.1103/physrevb.6.4380.

Xi, Y., Li, R., and Saad, Y. Fast computation of spectral densities for generalized eigenvalue problems. *SIAM Journal on Scientific Computing*, 40(4):A2749–A2773, January 2018. doi: 10.1137/17m1135542.

Yao, Z., Gholami, A., Keutzer, K., and Mahoney, M. Pyhessian: Neural networks through the lens of the hessian, 2020.