Rapid evaluation of the spectral signal detection threshold and Stieltjes transform

William Leeb*

Abstract

Accurate detection of signal components is a frequently-encountered challenge in statistical applications with low signal-to-noise ratio. This problem is particularly challenging in settings with heteroscedastic noise. In certain signal-plus-noise models of data, such as the classical spiked covariance model and its variants, there are closed formulas for the spectral signal detection threshold (the largest sample eigenvalue attributable solely to noise) for isotropic noise in the limit of infinitely large data matrices. However, more general noise models currently lack provably fast and accurate methods for numerically evaluating the threshold.

In this work, we introduce a rapid algorithm for evaluating the spectral signal detection threshold in the limit of infinitely large data matrices. We consider noise matrices with a separable variance profile (whose variance matrix is rank one), as these arise often in applications. The solution is based on nested applications of Newton's method. We also devise a new algorithm for evaluating the Stieltjes transform of the spectral distribution at real values exceeding the threshold. The Stieltjes transform on this domain is known to be a key quantity in parameter estimation for spectral denoising methods. The correctness of both algorithms is proven from a detailed analysis of the master equations characterizing the Stieltjes transform, and their performance is demonstrated in numerical experiments.

1 Introduction

Random matrix theory is an increasingly popular tool for deriving methods in statistical signal processing applications. Recent applications include signal denoising [41], [18], [29], [33], [31], covariance estimation [16], [31], filter correction in signal and image processing [4], [14], multiple-input multiple-output (MIMO) wireless communication [8], [5], and factor analysis [15], [13], to name just a few.

Part of the appeal of random matrix theory is that for many classes of random matrices, certain random quantities of statistical significance converge to well-defined limits as the matrix size grows infinitely large. By way of introduction, we illustrate this phenemonon in the *spiked covariance model*, introduced by Johnstone in [24]. Here, the user observes l iid random vectors $Y_1, \dots, Y_l$ in $\mathbb{R}^k$ of the form $Y_j = X_j + \varepsilon_j$, where X_j (the signal component) is a random vector with covariance $\mathbf{\Sigma}$ of rank $r \ll \min(k, l)$ and eigenvalues $\ell_1 > \cdots > \ell_r > 0$, and ε_j (the noise component) is a zero mean Gaussian vector with covariance $\mathbf{I}_k$. Typically, the user wishes to estimate quantities related to the signal covariance $\mathbf{\Sigma}$, such as its rank, eigenvectors, and eigenvalues, from the signal-plus-noise observations $Y_1, \dots, Y_l$.

It is well-known that when k and l are both large, there emerge simple relationships between the sample covariance $\widehat{\mathbf{\Sigma}} = \frac{1}{l}\sum_{j=1}^{l}(Y_j - \overline{Y})(Y_j - \overline{Y})^T$ (where $\overline{Y} = \frac{1}{l}\sum_{j=1}^{l} Y_j$ denotes the sample mean) and the signal covariance $\mathbf{\Sigma}$. For example, the largest r eigenvalues of $\widehat{\mathbf{\Sigma}}$ converge almost surely to the following limits as $k, l \to \infty$ and $k/l \to \gamma > 0$:

$$\lambda_j = \begin{cases} (\ell_j + 1)\left(1 + \frac{\gamma}{\ell_j}\right), & \text{if } \ell_j > \sqrt{\gamma} \\ (1 + \sqrt{\gamma})^2, & \text{if } \ell_j \leq \sqrt{\gamma}. \end{cases} . \tag{1}$$

Furthermore, the inner products $\langle \mathbf{u}_j, \hat{\mathbf{u}}_m \rangle$ between the top r eigenvectors $\mathbf{u}_1, \dots, \mathbf{u}_r$ of $\mathbf{\Sigma}$ and the top r eigenvectors $\hat{\mathbf{u}}_1, \dots, \hat{\mathbf{u}}_r$ of $\widehat{\mathbf{\Sigma}}$ converge almost surely to the following limits:

$$c_{jm} = \begin{cases} \sqrt{\frac{\ell_j^2 - \gamma}{\ell^2 + \gamma \ell_j}} & \text{if } j = m \text{ and } \ell_j > \sqrt{\gamma} \\ 0, & \text{otherwise} \end{cases} . \tag{2}$$

Proofs of these relationships may be found in the seminal work [38], while extensions to more general noise models can be found in [3]. These relationships have been used to devise estimators of $\mathbf{\Sigma}$, as in the work [16], as well as to predict the signal vectors $X_1, \dots, X_l$ themselves, as in [41], [18], [29], [30].

*School of Mathematics, University of Minnesota, Twin Cities. Minneapolis, MN, USA.

Of particular significance for the present work is the value $(1+\sqrt{\gamma})^2$. This is the limiting value of the operator norm of the sample covariance $\frac{1}{l}\sum_{j=1}^{l}\varepsilon_j\varepsilon_j^T$ of the noise component alone as $k,l\to\infty$ and $k/l\to\gamma$; see [19]. More generally, for any sequence of random noise matrices whose operator norms have an almost sure asymptotic limit as the dimensions grow to infinity, we refer to that limiting value as the *spectral signal detection threshold*, or *SSDT* for short. As is apparent for white noise from the formula (1), observed eigenvalues exceeding the SSDT may be attributed to the signal component, and used to extract information about the signal. Eigenvalues below the SSDT are not as useful for recovering information on the signal (though see Remark 4 below). In typical applications of the spiked covariance model and its generalizations to signal denoising, only the eigenvalues of $\widehat{\mathbf{\Sigma}}$ exceeding the SSDT are used for denoising [18], [16], [30], [33], [14], [31], [4]. For this reason, evaluation of the SSDT is a critical task in statistical signal processing.

While the SSDT and related quantities, such as the relations between the eigenvalues of $\mathbf{\Sigma}$ and $\widehat{\mathbf{\Sigma}}$, have simple formulas in the case of white noise, more general noise models pose greater challenges. In this paper, we study a fairly broad family of noise matrices, namely those with a *separable variance profile*: every entry of the noise matrix has a potentially different variance, but the matrix of all the variances is rank one. This type of random matrix arises naturally in a number of applications, such as signal denoising with variable-strength heteroscedastic noise and the Kronecker model of multiple-input multiple-output transmission. We show that even when there are not closed formulas for the quantities of interest in this model, they may nevertheless be evaluated rapidly and to machine precision via provably correct algorithms.

1.1 This paper's contributions

This paper introduces fast, scalable, and numerically stable algorithms for two related problems arising in statistical signal processing applications. The first is to evaluate the spectral signal detection threshold (SSDT) for noise matrices with a separable variance profile. As explained above, the SSDT is the asymptotic value of the operator norm of a random matrix representing the noise component in a signal-plus-noise observation model as the dimensions of the matrix grow to infinity.

The second problem we address is evaluating the Stieltjes transform (also known as the Cauchy-Stieltjes transform) of a certain probability measure associated with the noise matrix, known as the limiting spectral distribution (LSD). The Stieltjes transform is a function of central interest in random matrix theory and its applications; we recall its precise definition in Section 2.1. As we will review in Section 2.3, the Stieltjes transform, evaluated at real values exceeding the SSDT, may be used to estimate certain model parameters and perform signal denoising. Like the SSDT itself, in isotropic noise the Stieltjes transform has a closed form, while more general variance profiles pose greater challenges which we address in the current paper.

The algorithms we present scale linearly with the number of problem parameters, and are provably accurate to machine precision. The solution and analysis of each problem makes use of the master equations characterizing the Stieltjes transform, which will be reviewed in Section 2.2. Through a detailed analysis, presented in Section 3, we prove that the desired values may be computed by the use of Newton's root-finding algorithm. For the SSDT, several intermediate quantities used in each iteration of Newton's method are themselves computable by Newton's method, as we will show. Consequently, all parameters we compute are either available analytically, or can be provably computed to full precision by a rapidly converging iterative scheme. For matrices of size p-by-n, the asymptotic costs of the algorithms scale like $O(p+n)$.

1.2 Relation to prior work

The signal-plus-noise matrix models of the kind we consider have been previously studied in the statistical literature, in works such as [24], [3], [33], [14], [31], [18], [38], [30], [16], [22], [20], [49], [21]. Estimating the number of signal terms in such principal components analysis and factor models is a well-studied problem in statistics and statistical signal processing applications [15], [13], [6], [23], [25], [37], [26]. Detection in the low SNR regime constitutes a distinct though conceptually relevant class of problems in signal processing [44], [46], [47], [48]. Noise matrices with separable variance profiles have also been studied in the context of the Kronecker product model for MIMO wireless communication [8], [5] and factor analysis with applications to economics [35]. We also note that such random matrices are a special case of "algebraic" random matrices, introduced in [40].

The solution to both of the problems we study rests on a known characterization of the Stieltjes transform of the LSD as the solution to a set of certain non-linear equations; these have appeared in [39] and [9]. We present a new, detailed analysis of these equations. As a consequence of our analysis, we show that evaluating the Stieltjes transform may be done by a straightforward application of Newton's root-finding algorithm, whereas the SSDT is computable by several nested applications of Newton's agorithm, appropriately initialized.

Previous works have considered the problem of evaluating the LSD, its Stieltjes transform, and the boundary of its support, for different classes of random noise matrices. The paper [12] proposes a scheme for evaluating the Stieltjes transform at complex values outside the support of the LSD; this method is based on a non-linear equation satisfied by the Stieltjes transform [42], [43], [32]. The method in [7] is devoted to extending the approach to mixture models. The papers [27], [28] are concerned with solving the inverse problem of recovering the population distribution from the observed

spectrum; this problem is also taken up in [17]. The paper [15] contains a method for finding the boundary of a certain family of LSDs based on root-finding, although the use of Newton's method is not employed or analyzed. To the best of our knowledge, the problem of evaluating the SSDT and Stieltjes transform for random matrices with a separable variance profile has not been addressed in prior work; and for special cases that have been studied, such as in [15], the prior work does not contain fast algorithms with the guarantees presented here.

1.3 Outline

The remainder of the paper is outlined as follows. In Section 2, we precisely state the problems we solve in this paper, and review the mathematical and numerical material that we will be using. In Section 3, we derive the core mathematical theory on which our algorithms rest, namely a detailed analysis of the master equations characterizing the Stieltjes transform of the LSD. In Section 4, we provide explicit descriptions of the numerical algorithms for finding the SSDT and evaluating the Stieltjes transform. In Section 5, we present the results of several illustrative numerical experiments demonstrating the performance of our algorithms.

2 Preliminaries

2.1 Setting and problem formulation

We suppose we have a k-by-l random matrix of the form $\mathbf{N} = \mathbf{A}^{1/2}\mathbf{G}\mathbf{B}^{1/2}$, where $\mathbf{G}$ is a random matrix with iid entries of mean zero and variance l^{-1}, and $\mathbf{A}$ and $\mathbf{B}$ are positive-definite matrices of sizes k-by-k and l-by-l, respectively. We define the *empirical spectral distribution (ESD)* μ_k to be the distribution of eigenvalues $\lambda_1, \ldots, \lambda_k$ of the matrix $\mathbf{N}\mathbf{N}^T$:

$$d\mu_k(t) = \frac{1}{k}\sum_{i=1}^{k} \delta_{\lambda_i}(t). \tag{3}$$

As is typical in high-dimensional problems, we work in the asymptotic regime where k and l both grow to infinity. More precisely, we suppose that $l = l(k)$ grows with k, and that the limit

$$\gamma = \lim_{k\to\infty} \frac{k}{l(k)} \tag{4}$$

is well-defined, positive, and finite. Furthermore, as k and l grow, we suppose that the eigenvalue spectra of $\mathbf{A}$ and $\mathbf{B}$ have well-defined asymptotic distributions ν and $\underline{\nu}$, respectively (in terms of weak convergence), with compact support. Under suitable conditions on $\mathbf{A}$ and $\mathbf{B}$, the sequence of measures μ_k will almost surely converge weakly to a compactly supported measure μ, called the *limiting spectral distribution (LSD)*. We also denote by $\underline{\mu}$ the limiting spectral distribution of the eigenvalues of $\mathbf{N}^T\mathbf{N}$.

We define the *spectral signal detection threshold (SSDT)* as follows:

$$\lambda^* = \arg\max\{\lambda > 0 : \mu([\lambda, \infty)) > 0\}. \tag{5}$$

This is the right endpoint of the LSD μ's support, or equivalently the asymptotic operator norm of the matrix $\mathbf{N}\mathbf{N}^T$. As we will explain in Section 2.3, in a signal-plus-noise model $\mathbf{Y} = \mathbf{X} + \mathbf{N}$ where $\mathbf{X}$ is a low-rank signal matrix, eigenvalues exceeding λ^* are attributable to the signal $\mathbf{X}$.

Next, we define the *Stieltjes transform* of μ:

$$s(\lambda) = \int_{\mathbb{R}} \frac{1}{t-\lambda} d\mu(t), \tag{6}$$

which has derivative equal to

$$s'(\lambda) = \int_{\mathbb{R}} \frac{1}{(t-\lambda)^2} d\mu(t). \tag{7}$$

Similarly, the Stieltjes transform of $\underline{\mu}$ is given by

$$\underline{s}(\lambda) = \int_{\mathbb{R}} \frac{1}{t-\lambda} d\underline{\mu}(t), \tag{8}$$

with derivative equal to

$$\underline{s}'(\lambda) = \int_{\mathbb{R}} \frac{1}{(t-\lambda)^2} d\underline{\mu}(t). \tag{9}$$

We call $\underline{s}(\lambda)$ the *associated Stieltjes transform* of μ. The functions s and $\underline{s}$ are defined for all complex λ outside the supports of μ and $\underline{\mu}$, respectively. However, as we will explain in Section 2.3, in the present work we are only interested in evaluating $s(\lambda)$, $\underline{s}(\lambda)$, and their derivatives for real $\lambda > \lambda^*$.

In this paper, we consider the setting where ν and $\underline{\nu}$ are discrete distributions of the form

$$d\nu = \sum_{i=1}^{p} \omega_i \delta_{a_i} \tag{10}$$

and

$$d\underline{\nu} = \sum_{j=1}^{n} \pi_j \delta_{b_j}. \tag{11}$$

Here, $a_1, \ldots, a_p$ and $b_1, \ldots, b_n$ are positive numbers, and the weights ω_i and π_j satisfy $\sum_{i=1}^{p} \omega_i = \sum_{j=1}^{n} \pi_j = 1$, $\omega_i > 0$, and $\pi_j > 0$. For example, if $k/2$ eigenvalues of $\mathbf{A}$ are 1, and the remaining $k/2$ are equal to 2, then $p = 2$, $a_1 = 1$, $a_2 = 2$, and $\omega_1 = \omega_2 = 1/2$.

With this background and notation, we can concisely state the two problems we solve in this paper:

Problem 1. Given $a_1, \ldots, a_p$ with corresponding weights $\omega_1, \ldots, \omega_p$; and $b_1, \ldots, b_n$ with corresponding weights $\pi_1, \ldots, \pi_n$; evaluate the SSDT λ^*.

Problem 2. Given $a_1, \ldots, a_p$ with corresponding weights $\omega_1, \ldots, \omega_p$; and $b_1, \ldots, b_n$ with corresponding weights $\pi_1, \ldots, \pi_n$; and a value $\lambda > \lambda^*$; evaluate $s(\lambda)$, $s'(\lambda)$, $\underline{s}(\lambda)$, and $\underline{s}'(\lambda)$.

Algorithms solving Problems 1 and 2 to machine precision, with asymptotic cost $O(p + n)$, are presented in Section 4.

Remark 1. *In the problem statements, the parameters p, n, $a_1, \ldots, a_p$, and $b_1, \ldots, b_n$ are user-supplied inputs. In many statistical applications, these parameters are assumed to be known, as in the works [15], [31], [12], [22], [20], [21].*

Remark 2. *The assumption that $d\nu$ and $d\underline{\nu}$ are discrete measures is quite common in the literature on random matrix theory and its applications [15], [31], [12], [22], [20], [21]. It is likely that the methods in the present work can be generalized to more general measures, so long as certain linear functionals of the measures can be efficiently computed (e.g. by numerical quadrature). Pursuing this in detail lies outside the scope of the current work, however.*

2.2 Properties of the Stieltjes transform

The Stieltjes and associated Stieltjes transforms of the LSD μ are defined by equations (6) and (8), respectively. We refer the reader to the standard references [1], [45] for a detailed treatment of Stieltjes transforms of probability measures, particularly random spectral measures.

Lemma 1. *The Stieltjes transforms $s(\lambda)$ and $\underline{s}(\lambda)$ satisfy the following relations:*

$$\underline{s}(\lambda) = \gamma s(\lambda) + \frac{\gamma - 1}{\lambda} \tag{12}$$

and

$$s(\lambda) = \frac{1}{\gamma}\underline{s}(\lambda) + \left(\frac{1}{\gamma} - 1\right)\frac{1}{\lambda}. \tag{13}$$

Lemma 2. *The derivatives $s'(\lambda)$ and $\underline{s}'(\lambda)$ satisfy the following relations:*

$$\underline{s}'(\lambda) = \gamma s'(\lambda) + \frac{1 - \gamma}{\lambda^2} \tag{14}$$

and

$$s'(\lambda) = \frac{1}{\gamma}\underline{s}'(\lambda) + \left(1 - \frac{1}{\gamma}\right)\frac{1}{\lambda^2}. \tag{15}$$

Lemma 2 follows from Lemma 1, which in turn follows immediately from the relation:

$$d\underline{\mu}(t) = \gamma d\mu(t) + (1 - \gamma)\delta_0(t). \tag{16}$$

Remark 3. *Lemmas 1 and 2 show that $\underline{s}(\lambda)$ and $\underline{s}'(\lambda)$ may be easily evaluated once $s(\lambda)$ and $s'(\lambda)$ are computed.*

The next result is central to our subsequent analysis.

Theorem 1. *The Stieltjes transform $s(\lambda)$ for the LSD satisfies the following* master equations*:*

$$s(\lambda) = \int_{\mathbb{R}} \frac{1}{aG(e(\lambda)) - \lambda} d\nu(a), \tag{17}$$

where

$$G(e) = \int_{\mathbb{R}} \frac{b}{1 + \gamma b e} d\underline{\nu}(b) \tag{18}$$

and $e(\lambda)$ is a function that satisfies the equation

$$e(\lambda) = \int_{\mathbb{R}} \frac{a}{aG(e(\lambda)) - \lambda} d\nu(a). \tag{19}$$

For a proof of Theorem 1, see, for instance, the paper [39]. The paper [9] presents a slightly modified form of these equations, with a detailed analysis showing that $e(\lambda)$ is smooth for real λ outside the support of μ.

We assume that ν and $\underline{\nu}$ are discrete measures of the form

$$d\nu = \sum_{i=1}^{p} \omega_i \delta_{a_i}, \tag{20}$$

and

$$d\underline{\nu} = \sum_{j=1}^{n} \pi_j \delta_{b_j}, \tag{21}$$

where $\sum_{i=1}^{p} \omega_i = \sum_{j=1}^{n} \pi_j = 1$, $\omega_i > 0$, and $\pi_j > 0$. The master equations therefore become:

$$s(\lambda) = \sum_{i=1}^{p} \frac{\omega_i}{a_i G(e(\lambda)) - \lambda} \tag{22}$$

where $e(\lambda)$ is a function satisfying

$$e(\lambda) = \sum_{i=1}^{p} \frac{a_i \omega_i}{a_i G(e(\lambda)) - \lambda} \tag{23}$$

and the function G is defined by:

$$G(e) = \sum_{j=1}^{n} \frac{b_j \pi_j}{1 + \gamma b_j e}. \tag{24}$$

We note that in the case where $\mathbf{B} = \mathbf{I}_n$ (the singly-weighted case), the master equations take on a simpler form; see, for instance, [43], [42].

2.3 Spiked models and the D-transform

In a spiked matrix model, we observe a signal-plus-noise matrix $\mathbf{Y}$ of the form

$$\mathbf{Y} = \mathbf{X} + \mathbf{N}, \tag{25}$$

where $\mathbf{X} = \sum_{m=1}^{r} \theta_m \mathbf{u}_m \mathbf{v}_m^T$ is a rank $r \ll \min\{k, l\}$ signal matrix, and $\mathbf{N}$ is a noise matrix. The D-transform is defined as follows:

$$D(\lambda) = \lambda s(\lambda) \underline{s}(\lambda), \tag{26}$$

where $s(\lambda)$ and $\underline{s}(\lambda)$ are, respectively, the Stieltjes transform and associated Stieltjes transform of the LSD of $\mathbf{N}\mathbf{N}^T$. The D-transform was introduced in [3], though as a function of $\sqrt{\lambda}$ rather than λ.

Denoting by $\lambda_1, \ldots, \lambda_r$ the top r eigenvalues of $\mathbf{Y}\mathbf{Y}^T$, and $\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_r$, $\hat{\mathbf{v}}_1, \ldots, \hat{\mathbf{v}}_r$ the corresponding left and right singular vectors of $\mathbf{Y}$, respectively, it is shown in [3] that the D-transform defines a mapping between $\lambda_1, \ldots, \lambda_r$ and the eigenvalues $\theta_1^2, \ldots, \theta_r^2$ of the signal matrix $\mathbf{X}\mathbf{X}^T$, which holds almost surely in the limit $k, l \to \infty$:

$$\theta_m^2 = \lim_{k,l \to \infty} \frac{1}{D(\lambda_m)}. \tag{27}$$

This is satisfied for sufficiently large eigenvalues θ_m^2 of $\mathbf{X}\mathbf{X}^T$, namely those for which $\theta_m^2 > 1/D(\lambda^*)$. Phrased differently, an observed eigenvalue λ_m of $\mathbf{Y}\mathbf{Y}^T$ satisfying $\lambda_m > \lambda^*$ may be attributed to the presence of signal.

Furthermore, the asymptotic cosines $\langle \mathbf{u}_m, \hat{\mathbf{u}}_m \rangle$ and $\langle \mathbf{v}_m, \hat{\mathbf{v}}_m \rangle$, $1 \leq m \leq r$, can also be evaluated using the Stieltjes transform. It is shown that, almost surely,

$$\lim_{k,l \to \infty} |\langle \mathbf{u}_m, \hat{\mathbf{u}}_m \rangle|^2 = \frac{s(\lambda_m) D(\lambda_m)}{D'(\lambda_m)} \tag{28}$$

and

$$\lim_{k,l \to \infty} |\langle \mathbf{v}_m, \hat{\mathbf{v}}_m \rangle|^2 = \frac{\underline{s}(\lambda_m) D(\lambda_m)}{D'(\lambda_m)} \tag{29}$$

Consequently, evaluating $s(\lambda_m)$ and $s'(\lambda_m)$ provides a method for estimating the angles between the population singular vectors $\mathbf{u}_m$, $\mathbf{v}_m$ and the observed singular vectors $\hat{\mathbf{u}}_m$, $\hat{\mathbf{v}}_m$. These relationships have been employed to derive optimal methods of singular value shrinkage [33].

Remark 4. *While eigenvalues of $\mathbf{Y}\mathbf{Y}^T$ exceeding the SSDT λ^* indicate the presence of signal, the converse statement – if all eigenvalues are less than λ^*, then there is no signal – is subtler. For example, the work [36] studies the detection of signals in white noise and shows that the joint distribution of all eigenvalues of $\mathbf{Y}\mathbf{Y}^T$ changes even when the signal eigenvalues do not exceed the SSDT. However, the resulting statistical test does not accurately distinguish between the presence or absence of signal with overwhelming probability. Regardless, in tasks such as covariance and signal estimation in the spiked model, only those eigenvalues exceeding the SSDT are typically used for covariance estimation and signal denoising [18], [16], [30], [33], [14], [31], [4].*

Remark 5. *Separable variance profiles arise naturally when the columns of $\mathbf{Y}$ are independent random vectors in $\mathbb{R}^p$ of the form*

$$Y_j = X_j + b_j^{1/2} \mathbf{A}^{1/2} G_j, \quad 1 \leq j \leq n, \tag{30}$$

where the X_j are signal vectors constrained to an r-dimensional subspace of $\mathbb{R}^p$; $\mathbf{A}$ is a positive definite matrix; and $b_1, \ldots, b_n$ are specified weights. In this model, each signal vector X_j is observed in the presence of heteroscedastic noise (with covariance $\mathbf{A}$) of variable strength b_j. The noise matrix alone is distributed like $\mathbf{N} = \mathbf{A}^{1/2}\mathbf{G}\mathbf{B}^{1/2}$, where $\mathbf{B} = diag(b_1, \ldots, b_n)$. Models like these are considered in, for example, [30], [11]. When $\mathbf{A} = \mathbf{I}_p$, this model is considered in [22], [20], [21]; when $\mathbf{B} = \mathbf{I}_n$, this model is considered in [49], [31].

Proposition 1. *For $\lambda > \lambda^*$, $D(\lambda)$ is decreasing and convex.*

Proof. Write $D(\lambda) = \varphi(\lambda)\underline{\varphi}(\lambda)$, where $\varphi(\lambda) = \sqrt{\lambda} s(\lambda)$ and $\underline{\varphi}(\lambda) = \sqrt{\lambda}\underline{s}(\lambda)$. Using $D'(\lambda) = \varphi(\lambda)\underline{\varphi}'(\lambda) + \varphi'(\lambda)\underline{\varphi}(\lambda)$ and $D''(\lambda) = \varphi(\lambda)\underline{\varphi}''(\lambda) + \varphi''(\lambda)\underline{\varphi}(\lambda) + 2\varphi'(\lambda)\underline{\varphi}'(\lambda)$, and $\varphi(\lambda) < 0$ and $\underline{\varphi}(\lambda) < 0$ for $\lambda > \lambda^*$, it is enough to show that $\varphi(\lambda)$ $\underline{\varphi}(\lambda)$ are increasing and concave. But this follows immediately from the identities

$$\frac{d}{d\lambda} \frac{\sqrt{\lambda}}{t - \lambda} = \frac{1}{2} \frac{\lambda + t}{\sqrt{\lambda}(t - \lambda)^2} > 0 \tag{31}$$

and

$$\frac{d^2}{d\lambda^2} \frac{\sqrt{\lambda}}{t - \lambda} = \frac{1}{4} \frac{6\lambda t + 3\lambda^2 - t^2}{\lambda^{3/2}(t - \lambda)^3} < 0, \tag{32}$$

and the definitions (6) and (8) of $s(\lambda)$ and $\underline{s}(\lambda)$. □

2.4 Newton's root-finding algorithm

Newton's method is a classical technique for finding the roots of a function of one real variable. We briefly review the method here; the reader may consult any standard reference on optimization or numerical analysis, such as [34], [10], for additional details. We are given a smooth function $f(x)$, where $x \in [a, b]$, and we suppose $f(a) < 0$ and $f(b) > 0$. We also suppose that $f'(x) > 0$ and $f''(x) < 0$ for all $x \in (a, b)$; that is, f is a strictly increasing and concave. Our goal is to compute x^*, the unique root of f in (a, b).

To find x^*, Newton's method initializes $a < x_0 < x^*$, and defines a sequence of updates recursively as follows: given an estimate x_k, the next value x_{k+1} is defined by

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}. \tag{33}$$

Geometrically, x_{k+1} is the root of the line tangent to the graph of f at the point $(x_k, f(x_k))$. Because f is concave, it is easy to see that $x_k < x_{k+1} \leq x^*$.

Because each x_k is obtained from a linear approximation to f, the errors decay quadratically in the vicinity of x^*; that is

$$|x_{k+1} - x^*| \leq C|x_k - x^*|^2 \tag{34}$$

when $|x_k - x^*|$ is sufficiently small.

Remark 6. *Quadratic convergence is what makes Newton's method an especially attractive algorithm when it is applicable – that is, when both $f(x)$ and $f'(x)$ may be accurately evaluated, and f exhibits the conditions specified above. In practical terms, quadratic convergence means that the number of accurately computed digits of x^* approximately doubles after each iteration of the algorithm, until machine precision is reached.*

3 Mathematical apparatus

In this section, we analyze the master equations (22) – (24). Our results will provide the necessary tools to devise algorithms for computing the SSDT λ^* and evaluating $s(\lambda)$ and $s'(\lambda)$ for $\lambda > \lambda^*$. We define

$$a^* = \max_{1 \leq i \leq p} a_i, \tag{35}$$

and

$$b^* = \max_{1 \leq j \leq n} b_j. \tag{36}$$

We define the function $F(\lambda, e)$ by:

$$F(\lambda, e) = e - \sum_{i=1}^{p} \frac{a_i \omega_i}{a_i G(e) - \lambda}. \tag{37}$$

Then for each $\lambda > \lambda^*$, $e(\lambda)$ satisfies $F(\lambda, e(\lambda)) = 0$. When we treat λ as a fixed parameter and e as a variable, we will use the notation $F_\lambda(e) = F(\lambda, e)$.

3.1 Range and monotonicity of $e(\lambda)$ when $\lambda > \lambda^*$

We first state a result on the range of $e(\lambda)$ for $\lambda > \lambda^*$. The function $G(e)$ approaches 0 as $e \to \infty$, and grows to $+\infty$ as $e \to (-1/\gamma b^*)^+$, and is strictly decreasing on the interval

$$J \equiv \left\{ e : e > \frac{-1}{\gamma b^*} \right\}. \tag{38}$$

For any $\lambda > 0$, we define the interval I_λ by

$$I_\lambda \equiv \left\{ e \in J : G(e) < \frac{\lambda}{a^*} \right\}. \tag{39}$$

Proposition 2. *When $\lambda > \lambda^*$, $e(\lambda)$ is contained in the interval $I_\lambda \cap (-\infty, 0)$.*

We will develop the proof in several steps.

Lemma 3. *Let $\epsilon > 0$. Then the function $G(e(\lambda))$ is bounded for all $\lambda > \lambda^* + \epsilon$.*

Proof. We show that the range of $e(\lambda)$ cannot approach any of the singularities of G, which lie at the values $-1/(\gamma b_j)$. Define

$$H(\lambda) = \frac{G(e(\lambda))}{\lambda}. \tag{40}$$

Then from (22)

$$\lambda s(\lambda) = \sum_{i=1}^{p} \frac{\omega_i}{a_i H(\lambda) - 1}, \tag{41}$$

and consequently

$$1 + \lambda s(\lambda) = \sum_{i=1}^{p} \frac{\omega_i}{a_i H(\lambda) - 1} + \sum_{i=1}^{p} \omega_i \frac{a_i H(\lambda) - 1}{a_i H(\lambda) - 1} = \sum_{i=1}^{p} \frac{a_i H(\lambda)\omega_i}{a_i H(\lambda) - 1} = G(e(\lambda))e(\lambda). \tag{42}$$

Since $\lambda s(\lambda)$ is bounded for $\lambda > \lambda^* + \epsilon$, this tells us that $G(e(\lambda))$ must stay bounded so long as $e(\lambda)$ is bounded away from 0; in particular, $e(\lambda)$ cannot be made arbitrarily close to any of the singularities $-1/(\gamma b_j)$. □

Corollary 1. *$e(\lambda) \to 0^-$ as $\lambda \to \infty$.*

Proof. Because $G(e(\lambda))$ is bounded for large λ, the result follows from (23). □

Corollary 2. *For any $\lambda > \lambda^*$, $e(\lambda) \in J$; that is,*

$$e(\lambda) > \frac{-1}{\gamma b^*}. \tag{43}$$

Proof. This follows from the continuity of $e(\lambda)$ for $\lambda > \lambda^*$, and the facts that it never passes through $-1/(\gamma b_j)$ and approaches 0 at large λ. □

Corollary 3. *For all $\lambda > \lambda^*$, $G(e(\lambda)) > 0$ and $e(\lambda) < 0$.*

Proof. The positivity of $G(e(\lambda))$ follows immediately from Corollary 2. The negativity of $e(\lambda)$ then follows from (42), and the fact that $1 + \lambda s(\lambda) < 0$. □

Lemma 4. *For all $\lambda > \lambda^*$,*

$$\frac{G(e(\lambda))}{\lambda} < \frac{1}{a^*}. \tag{44}$$

Proof. We have $G(e(\lambda)) \neq \lambda/a_i$, from (22). Suppose for some $\lambda > \lambda^*$, we had

$$\frac{G(e(\lambda))}{\lambda} > \frac{1}{a_i}. \tag{45}$$

The inequality must then remain true for all sufficiently large λ, since $G(e(\lambda))/\lambda$ is continuous and does not pass through $1/a_i$. However, the left side converges to 0 as $\lambda \to \infty$, since $G(e(\lambda))$ is bounded for large λ; a contradiction. This completes the proof. □

We have shown that for all $\lambda > \lambda^*$, $e(\lambda)$ lies in the interval defined by the inequalities

$$G(e) \leq \frac{\lambda}{a^*}, \qquad e > \frac{-1}{\gamma b^*}, \qquad e < 0. \tag{46}$$

This completes the proof of Proposition 2.

Next we prove that $e(\lambda)$ is increasing:

Proposition 3. *The function $e(\lambda)$ is increasing for $\lambda > \lambda^*$.*

Proof. We have:

$$(\partial_\lambda F)(\lambda, e) = -\sum_{i=1}^{p} \frac{a_i \omega_i}{(a_i G(e) - \lambda)^2} < 0. \tag{47}$$

Since $F(\lambda, e(\lambda)) = 0$, we have:

$$0 = \frac{\partial}{\partial \lambda}\{F(\lambda, e(\lambda))\} = (\partial_\lambda F)(\lambda, e(\lambda)) + e'(\lambda)(\partial_e F)(\lambda, e(\lambda)), \tag{48}$$

and so

$$e'(\lambda)(\partial_e F)(\lambda, e(\lambda)) = \sum_{i=1}^{p} \frac{a_i \omega_i}{(a_i G(e) - \lambda)^2} > 0. \tag{49}$$

Consequently, $e'(\lambda)$ can never be 0. Furthermore,

$$(\partial_e F)(\lambda, e) = 1 + G'(e(\lambda)) \sum_{i=1}^{p} \left(\frac{a_i}{a_i G(e(\lambda)) - \lambda} \right)^2 \omega_i \tag{50}$$

which converges to 1 as $\lambda \to \infty$ (note that $e(\lambda)$ stays bounded away from singularities of $G(e)$ and also $G'(e)$, which have the same singularities). So $e'(\lambda) > 0$ for sufficiently large λ, and hence, since it is continuous and cannot pass through 0, $e'(\lambda) > 0$ for all $\lambda > \lambda^*$. □

3.2 Behavior of $F_\lambda(e)$

In this section we characterize the behavior of $F_\lambda(e) = F(\lambda, e)$ (viewed as a function of e) on the interval I_λ. Specifically, we show the following. For any $\lambda > 0$, $F_\lambda(e)$ is a strictly convex function that approaches $+\infty$ as e approaches either end of I_λ. Furthermore, when $\lambda > \lambda^*$, the minimum value of $F_\lambda(e)$ is less than zero, and $F_\lambda(0) > 0$; consequently, there are exactly two roots of $F_\lambda(e)$, both contained in the interval $I_\lambda \cap (-\infty, 0)$. We show that $e(\lambda)$ is always equal to the largest root, namely the one at which $F'_\lambda(e) > 0$.

Lemma 5. *For $\lambda > 0$, the function $F_\lambda(e)$ is strictly convex on I_λ; that is, $(\partial^2_{ee} F)(\lambda, e) > 0$.*

Proof. We have:

$$(\partial_e F)(\lambda, e) = 1 + G'(e(\lambda)) \sum_{i=1}^{p} \left(\frac{a_i}{a_i G(e(\lambda)) - \lambda} \right)^2 \omega_i \tag{51}$$

and

$$(\partial^2_{ee} F)(\lambda, e) = G''(e) \sum_{i=1}^{p} \left(\frac{a_i}{a_i G(e) - \lambda} \right)^2 \omega_i - 2G'(e)^2 \sum_{i=1}^{p} \left(\frac{a_i}{a_i G(e) - \lambda} \right)^3 \omega_i. \tag{52}$$

Now whenever $e \in I_\lambda$, we have $e > -1/\gamma b^* \geq -1/\gamma b_j$ for all $1 \leq j \leq n$, and $G(e) < \lambda/a^* \leq \lambda/a_i$ for all $1 \leq i \leq p$; consequently,

$$G''(e) = 2\gamma^2 \sum_{j=1}^{n} \left(\frac{b_j}{1 + \gamma b_j e} \right)^3 \pi_j > 0 \tag{53}$$

and

$$\sum_{i=1}^{p} \left(\frac{a_i}{a_i G(e) - \lambda} \right)^3 \omega_i < 0. \tag{54}$$

Consequently, $(\partial^2_{ee} F)(\lambda, e) > 0$ for all $e \in I_\lambda$, i.e. the function $F_\lambda(e)$ is convex. □

Lemma 6. *The function $F_\lambda(e)$ diverges to $+\infty$ as $e \to +\infty$ and as e approaches the left endpoint of I_λ from the right.*

Proof. This is immediate from the definition of $F(\lambda, e)$. □

Lemma 7. *For each $\lambda > \lambda^*$, $F(\lambda, 0) > 0$.*

Proof. We have:

$$F(\lambda, 0) = -\sum_{i=1}^{p} \frac{a_i \omega_i}{a_i G(0) - \lambda}. \tag{55}$$

Since $G(e(\lambda)) < \lambda/a_i$ and $G(e)$ is decreasing on I_λ, and $e(\lambda) < 0$, we also have $G(0) < G(e(\lambda)) < \lambda/a_i$; consequently, $F(\lambda, 0) > 0$. □

Proposition 4. *For $\lambda > \lambda^*$, $(\partial_e F)(\lambda, e(\lambda)) > 0$.*

Proof. This follows from (49) and $e'(\lambda) > 0$. □

We have shown that $F_\lambda(e)$ has two roots in the interval $I_\lambda \cap (-\infty, 0)$ whenever $\lambda > \lambda^*$. Proposition 4 identifies $e(\lambda)$ as the root that is closest to zero, or equivalently, the root where the derivative of F_λ is positive. This characterization will be used in Section 4.2 to devise the algorithm for computing $e(\lambda)$, and consequently $s(\lambda)$.

3.3 The minimum of $F_\lambda(e)$

We will let $t(\lambda)$ denote the minimum of $F_\lambda(e)$ on J_λ; that is, $t(\lambda)$ is the unique value on J_λ that satisfies

$$(\partial_e F)(\lambda, t(\lambda)) = 0. \tag{56}$$

We define the function $Q(\lambda)$ for $\lambda > 0$ by:

$$Q(\lambda) = F(\lambda, t(\lambda)). \tag{57}$$

For any $\lambda > 0$, we define the function $R_\lambda(e)$ for $e \in I_\lambda$ by:

$$R_\lambda(e) = (\partial_e F)(\lambda, e) = F'_\lambda(e). \tag{58}$$

Note that by definition, $R_\lambda(t(\lambda)) = 0$ for all $\lambda = 0$.

We will show that Q is decreasing and convex and R_λ is increasing and concave.

Lemma 8. *$Q(\lambda)$ is a decreasing function of $\lambda > 0$.*

Proof. The derivative of Q may be computed as follows:

$$\begin{aligned} Q'(\lambda) &= \partial_\lambda \{F(\lambda, t(\lambda))\} \\ &= (\partial_\lambda F)(\lambda, t(\lambda)) + (\partial_e F)(\lambda, t(\lambda)) t'(\lambda) \\ &= (\partial_\lambda F)(\lambda, t(\lambda)) \\ &= -\sum_{i=1}^{p} \frac{a_i \omega_i}{(a_i G(t(\lambda)) - \lambda)^2}, \end{aligned} \tag{59}$$

which is negative. □

Proposition 5. *$Q(\lambda)$ is a convex function of $\lambda > 0$.*

Proof. We first compute the derivative of $t(\lambda)$. We have

$$0 = (\partial_e F)(\lambda, t(\lambda)). \tag{60}$$

Differentiating with respect to λ, we find

$$0 = (\partial^2_{\lambda e} F)(\lambda, t(\lambda)) + (\partial^2_{ee} F)(\lambda, t(\lambda)) t'(\lambda), \tag{61}$$

and so

$$\begin{aligned} t'(\lambda) &= \frac{-(\partial^2_{\lambda e} F)(\lambda, t(\lambda))}{(\partial^2_{ee} F)(\lambda, t(\lambda))} \\ &= \frac{-2G'(t(\lambda)) \sum_{i=1}^{p} \frac{a_i^2 \omega_i}{(a_i G(t(\lambda)) - \lambda)^3}}{G''(t(\lambda)) \sum_{i=1}^{p} \frac{a_i^2 \omega_i}{(a_i G(t(\lambda)) - \lambda)^2} - 2G'(e)^2 \sum_{i=1}^{p} \frac{a_i^3 \omega_i}{(a_i G(t(\lambda)) - \lambda)^3}}. \end{aligned} \tag{62}$$

Now the second derivative of Q is given by

$$Q''(\lambda) = 2 \sum_{i=1}^{p} \frac{a_i (a_i G'(t(\lambda)) t'(\lambda) - 1) \omega_i}{(a_i G(t(\lambda)) - \lambda)^3}. \tag{63}$$

We will show that $Q''(\lambda) > 0$ for all λ. In fact, we will show the stronger result that each summand is positive, or equivalently, recalling that $a^* = \max_{1 \le i \le p} a_i$,

$$a^* G'(t(\lambda)) t'(\lambda) \le 1. \tag{64}$$

To prove this, we observe that

$$\begin{aligned}
a^*G'(t(\lambda))t'(\lambda) &= \frac{-2G'(t(\lambda))^2\sum_{i=1}^p \frac{a_i^3\omega_i}{(a_iG(t(\lambda))-\lambda)^3}\frac{a^*}{a_i}}{G''(t(\lambda))\sum_{i=1}^p \frac{a_i^2\omega_i}{(a_iG(t(\lambda))-\lambda)^2} - 2G'(t(\lambda))^2\sum_{i=1}^p \frac{a_i^3\omega_i}{(a_iG(t(\lambda))-\lambda)^3}} \\
&= \frac{-2G'(t(\lambda))^2\sum_{i=1}^p \frac{a_i^3\omega_i}{(a_iG(t(\lambda))-\lambda)^3}\frac{a^*}{a_i}}{G''(t(\lambda))\sum_{i=1}^p \frac{a_i^3\omega_i}{(a_iG(t(\lambda))-\lambda)^3}\frac{a_iG(t(\lambda))-\lambda}{a_i} - 2G'(t(\lambda))^2\sum_{i=1}^p \frac{a_i^3\omega_i}{(a_iG(t(\lambda))-\lambda)^3}} \\
&= \frac{-2G'(t(\lambda))^2\sum_{i=1}^p \frac{a_i^3\omega_i}{(a_iG(t(\lambda))-\lambda)^3}\frac{a^*}{a_i}}{\sum_{i=1}^p \frac{a_i^3\omega_i}{(a_iG(t(\lambda))-\lambda)^3}\left(G''(t(\lambda))\frac{a_iG(t(\lambda))-\lambda}{a_i} - 2G'(t(\lambda))^2\right)} \\
&= \frac{\sum_{i=1}^p \frac{a_i^3\omega_i}{(\lambda-a_iG(t(\lambda)))^3}\frac{a^*}{a_i}}{\sum_{i=1}^p \frac{a_i^3\omega_i}{(\lambda-a_iG(t(\lambda)))^3}\left(1 - \frac{G''(t(\lambda))}{2G'(t(\lambda))^2}\frac{a_iG(t(\lambda))-\lambda}{a_i}\right)}.
\end{aligned} \tag{65}$$

To show that this is less than 1, it is enough to show that

$$\frac{a^*}{a_i} \le 1 - \frac{G''(t(\lambda))}{2G'(t(\lambda))^2}\frac{a_iG(t(\lambda))-\lambda}{a_i} = 1 - \frac{G''(t(\lambda))}{2G'(t(\lambda))^2}\left(G(t(\lambda)) - \frac{\lambda}{a_i}\right), \tag{66}$$

or equivalently

$$1 \le \frac{a_i}{a^*} + \frac{G''(t(\lambda))}{2G'(t(\lambda))^2}\left(\frac{\lambda}{a^*} - \frac{a_i}{a^*}G(t(\lambda))\right), \tag{67}$$

We will show that this inequality holds for any value of a_i between 0 and a^*. If $a_i = a^*$, then the right side becomes

$$1 + \frac{G''(t(\lambda))}{2G'(t(\lambda))^2}\left(\frac{\lambda}{a^*} - G(t(\lambda))\right), \tag{68}$$

and since the term

$$\frac{G''(t(\lambda))}{2G'(t(\lambda))^2}\left(\frac{\lambda}{a^*} - G(t(\lambda))\right) \tag{69}$$

is positive (because G is convex, $t(\lambda) \in J_\lambda$, and $a^*G(e) < \lambda$ on J_λ), the desired inequality is satisfied.

On the other hand, if $a_i = 0$, the right hand side becomes

$$\frac{G''(t(\lambda))}{2G'(t(\lambda))^2}\frac{\lambda}{a^*} > \frac{G''(t(\lambda))G(\lambda)}{2G'(t(\lambda))^2} \tag{70}$$

where the inequality holds since $t(\lambda) \in J_\lambda$, and $a^*G(e) < \lambda$ for all $e \in J_\lambda$. By the Cauchy-Schwarz inequality,

$$\begin{aligned}
|G'(\lambda)| = \gamma\sum_{j=1}^n \left(\frac{b_j}{1+\gamma b_j e}\right)^2\pi_j &= \gamma\sum_{j=1}^n \left(\frac{b_j}{1+\gamma b_j e}\right)^{3/2}\left(\frac{b_j}{1+\gamma b_j e}\right)^{1/2}\pi_j \\
&\le \left[\gamma^2\sum_{j=1}^n \left(\frac{b_j}{1+\gamma b_j e}\right)^3\pi_j\right]^{1/2}\left[\sum_{j=1}^n \frac{b_j}{1+\gamma b_j e}\pi_j\right]^{1/2} \\
&= \sqrt{\frac{G''(e)G(e)}{2}},
\end{aligned} \tag{71}$$

or in other words,

$$\frac{G''(e)G(e)}{2G'(e)^2} \ge 1. \tag{72}$$

This gives the desired result. □

Proposition 6. *For all $\lambda > 0$, $R_\lambda(e)$ is an increasing and concave function of $e \in I_\lambda$.*

Proof. The first derivative of R_λ is:

$$R'_\lambda(e) = (\partial^2_{ee}F)(\lambda, e), \tag{73}$$

which as we've seen is positive. The second derivative of R_λ is

$$\begin{aligned} R''_\lambda(e) =& (\partial^3_{eee}F)(\lambda, e) \\ =& G^{(3)}(e)\sum_{i=1}^{p}\left(\frac{a_i}{a_iG(e)-\lambda}\right)^2\omega_i - 6G''(e)G'(e)\sum_{i=1}^{p}\left(\frac{a_i}{a_iG(e)-\lambda}\right)^3\omega_i \\ &+ 6G'(e)^3\sum_{i=1}^{p}\left(\frac{a_i}{a_iG(e)-\lambda}\right)^4\omega_i, \end{aligned} \tag{74}$$

which is always negative, since $G^{(3)}(e) < 0$, $G''(e) > 0$, $G'(e) < 0$, and $a_iG(e) < \lambda$ for all $e \in I_\lambda$ and $1 \le i \le p$. □

4 Algorithms

In this section, we describe the algorithms for computing the SSDT λ^* and for evaluating the Stieltjes transform $s(\lambda)$ and its derivative $s'(\lambda)$ at values $\lambda > \lambda^*$. By Lemmas 1 and 2, $\underline{s}(\lambda)$ and $\underline{s}'(\lambda)$ may be easily found from $s(\lambda)$ and $s'(\lambda)$.

4.1 Computation of the boundary λ^*

In this section, we derive an algorithm for the computation of λ^*. First, we observe that when $\lambda > \lambda^*$, then as we have shown there are real roots of $F_\lambda(e)$ to the left and to the right of $t(\lambda)$; in particular, $F(\lambda, t(\lambda)) < 0$. On the other hand, if $\lambda < \lambda^*$, the function $F_\lambda(e)$ cannot have a real root, since that would imply that the Stieltjes transform is real inside the support of μ. Consequently, $F(\lambda, t(\lambda)) > 0$. It follows that the SSDT λ^* is the value at which $F(\lambda, t(\lambda)) = 0$; that is, λ^*, is the unique root of Q on $(0, \infty)$.

From Lemma 8 and Proposition 5, $Q(\lambda)$ is decreasing and convex. With an efficient procedure for evaluating $Q(\lambda)$ and $Q'(\lambda)$, we can therefore use Newton's algorithm to find its root if we initialize the algorithm to the left of the root. In Section 4.1.2, we detail how to evaluate $Q(\lambda)$ and $Q'(\lambda)$. As a preliminary step, we will need to compute the left endpoint of I_λ; we do this in 4.1.1. The resulting algorithm for evaluating λ^* is summarized in Algorithm 3.

Algorithm 1 Computation of the left endpoint of I_λ.

1: **Input:** Precision $\epsilon > 0$; parameter $\lambda > 0$
2: **Initialize:** $e > -1/(\gamma b^*)$
3: **Bisection:** $e \leftarrow (e - 1/(\gamma b^*))/2$, until $T_\lambda(e) > 0$
4: **Newton:** $e \leftarrow e - T_\lambda(e)/T'_\lambda(e)$, until $|T_\lambda(e)| < \epsilon$
5: **Output:** $e^*_\lambda = e$

Algorithm 2 Evaluation of $t(\lambda)$, $Q(\lambda)$ and $Q'(\lambda)$.

1: **Input:** Precision $\epsilon > 0$; parameter $\lambda > 0$
2: **Endpoint:** Compute e^*_λ using Algorithm 1
3: **Initialize:** $e > e^*_\lambda$
4: **Bisection:** $e \leftarrow (e + e^*_\lambda)/2$, until $R_\lambda(e) < 0$
5: **Newton:** $e \leftarrow e - R_\lambda(e)/R'_\lambda(e)$, until $|R_\lambda(e)| < \epsilon$
6: **Output:** $t(\lambda) = e$, $Q(\lambda) = F(\lambda, e)$, $Q'(\lambda) = (\partial_\lambda F)(\lambda, e)$

4.1.1 Computation of the left endpoint of I_λ, $\lambda > 0$

We recall the definition of the interval I_λ:

$$I_\lambda = \left\{ e \in J : G(e) < \frac{\lambda}{a^*} \right\}, \tag{75}$$

Algorithm 3 Evaluation of λ^*.

1: **Input:** Precision $\epsilon > 0$
2: **Initialize:** $\lambda > 0$
3: **Bisection:** $\lambda \leftarrow \lambda/2$, until $Q(\lambda) < 0$
4: **Newton:** $\lambda \leftarrow \lambda - Q(\lambda)/Q'(\lambda)$, until $|Q(\lambda)| < \epsilon$
5: **Output:** $\lambda^* = \lambda$

where J is the interval $J = \{e : e > -1/(\gamma b^*)\}$. Let us denote by e_λ^* the left endpoint of I_λ. Since $G(e)$ is a decreasing function of e on J, e_λ^* is the unique root of

$$T_\lambda(e) = G(e) - \frac{\lambda}{a^*} \tag{76}$$

on J. Since T_λ is a decreasing, convex function of e on J, we may find its root using Newton's algorithm, initialized to the left of the root. Such an initial value e_0 can be found by starting with any value e in J, and performing bisection with $-1/(\gamma b^*)$, the left endpoint of J, until we arrive at a value where $T_\lambda(e_0) > 0$. Newton's algorithm is then performed with initial value e_0. We summarize the procedure in Algorithm 1.

4.1.2 Evaluation of $t(\lambda)$, $Q(\lambda)$, and $Q'(\lambda)$

Next, we show how to evaluate the functions $t(\lambda)$, $Q(\lambda)$, and $Q'(\lambda)$. $t(\lambda)$ is defined as the root of $R_\lambda(e)$ on I_λ. Since $R_\lambda(e)$ is an increasing, concave function, we may find its root using the Newton algorithm initialized to the left of the root, i.e. the region where $R_\lambda(e) < 0$. Such an initial value may be found by taking a starting point e to the right of e_λ^*, and performing bisection on e and e_λ^* until we arrive at a value e_0 with $R_\lambda(e_0) < 0$. We can then perform Newton's algorithm on R_λ, initialized at e_0. The procedure is summarized in Algorithm 2.

4.2 Evaluation of $s(\lambda)$ and $s'(\lambda)$, $\lambda > \lambda^*$

In this section, we present an algorithm for evaluating the Stieltjes transform $s(\lambda)$ of μ, and its derivative $s'(\lambda)$, when $\lambda > \lambda^*$. This immediately provides a method for evaluating $\underline{s}(\lambda)$ and $\underline{s}'(\lambda)$, and the D-transform $D(\lambda)$ defined in Section 2.3.

As we showed in Section 3.2, the function $F_\lambda(e)$ is convex on I_λ and has two roots, both of which are negative. The root closest to 0, i.e. the rightmost root, is $e(\lambda)$. Since $F_\lambda(0) > 0$ and $F_\lambda(e)$ is convex, this tells us that Newton's method, initialized at $e_0 = 0$, will converge to $e(\lambda)$. For brevity, we introduce the function $W(\lambda, e)$ defined by:

$$W(\lambda, e) = \sum_{i=1}^{p} \frac{\omega_i}{a_i G(e) - \lambda}. \tag{77}$$

With this notation, we have:

$$s(\lambda) = W(\lambda, e(\lambda)) \tag{78}$$

and

$$s'(\lambda) = (\partial_\lambda W)(\lambda, e(\lambda)) + (\partial_e W)(\lambda, e(\lambda))e'(\lambda). \tag{79}$$

Note that from (48), we can evaluate $e'(\lambda)$:

$$e'(\lambda) = \frac{-(\partial_\lambda F)(\lambda, e(\lambda))}{(\partial_e F)(\lambda, e(\lambda))}. \tag{80}$$

The method for evaluating $s(\lambda)$ and $s'(\lambda)$ is summarized in Algorithm 4.

Algorithm 4 Evaluation of $s(\lambda)$ and $s'(\lambda)$.

1: **Input:** Precision $\epsilon > 0$; parameter $\lambda > \lambda^*$
2: **Initialize:** $e = 0$
3: **Newton:** $e \leftarrow e - F_\lambda(e)/F'_\lambda(e)$, until $|F_\lambda(e)| < \epsilon$
4: **Output:** $s(\lambda) = W(\lambda, e)$, $s'(\lambda) = (\partial_\lambda W)(\lambda, e) - (\partial_e W)(\lambda, e)(\partial_\lambda F)(\lambda, e)/(\partial_e F)(\lambda, e)$

5 Numerical results

In this section, we report on several experiments illustrating the behavior of the algorithms from Section 4. For the timings reported in Sections 5.2 and 5.5, we used an implementation written in MATLAB 2019b and run on a Dell Precision 5540 with 62.5 GB of RAM and an Intel Core i9 CPU. The MATLAB code is available at the following URL: `https://github.com/wleeb/MPBoundary`.

5.1 Convergence

Algorithms 1 – 4 are all versions of the Newton root-finding method. In this section, we illustrate the quadratic convergence of these methods, as predicted from the theory reviewed in Section 2.4. In each experiment, we used parameters $p = 512$, $n = 1024$, and $\gamma = 1/2$. We generated the values $a_1, \ldots, a_p$ and $b_1, \ldots, b_n$ randomly from a Unif$(0, 1)$ distribution, and assigned random probabilities ω_i and π_j by drawing values from Unif$(0, 1)$ and normalizing to sum to 1. For experiments in which a value λ is specified, we also choose it at random.

In Tables 1 – 4, the first column displays the iteration number m, starting from the initial value and going until the root has been reached. The second column displays the function value at the m^{th} iterate; the algorithm terminates when the function reaches machine precision ϵ. We work in double precision, so $\epsilon \approx 10^{-16}$. The third column displays the relative error in the root itself, defined by:

$$\text{err}(x_m, x) = \frac{x_m - x}{x}. \tag{81}$$

Table 1 shows the results for Algorithm 1, which computes the left endpoint e^*_λ of I_λ. Table 2 shows the results for Algorithm 2, which evaluates $t(\lambda)$. Table 3 shows the results for Algorithm 3, which computes the boundary λ^*. Table 4 shows the results for Algorithm 4, which evaluates $e(\lambda)$.

Remark 7. *For each algorithm, we observe quadratic convergence close to the root, as expected. That is, on each iteration close to termination the number of correct digits roughly doubles, and the size of the objective roughly squares, until machine precision is reached.*

5.2 Scalability

In the next experiment, we compute timings for the computation of λ^* and the evaluation of $s(\lambda)$. For increasing values of n, we set $p = n/2$ ($\gamma = 1/2$). We generated the values $a_1, \ldots, a_p$ and $b_1, \ldots, b_n$ randomly from a Unif$(1, 2)$ distribution, and assigned them random probabilities ω_i and π_j by drawing values from Unif$(0, 1)$ and normalizing to sum to 1.

For each n, we then record the time in seconds required to compute λ^*, and the time in seconds required to compute $s(\lambda)$ on a grid of 100 equispaced values of λ between $\lambda^* + 1$ and $\lambda^* + 10$. The reported timings are averaged over five runs of the experiment, and are displayed in Table 5. It is apparent that the running times scale approximately linearly with n, as we would expect. In addition to linearly scaling with n, the magnitudes of the timings are quite encouraging; for example, it takes only about 4 seconds to compute λ^* when n is over two million and p is over one million.

5.3 Finite sample accuracy for λ^*

The master equations (22) – (24) from which we compute λ^* are asymptotic and deterministic, holding almost surely in the limit as $k, l \to \infty$. In this experiment, we assess the finite sample accuracy of estimating the SSDT λ^* from the operator norm of the random matrix $\mathbf{N}\mathbf{N}^T$. We consider how the estimate improves as k and l grow. We use a model where $p = n = 2$, and both ν and $\underline{\nu}$ have point masses at 2 and 3, each with weight $1/2$; and set $\gamma = 1/2$. We generate matrices of size k-by-l, where k grows and $l = k/\gamma = 2k$.

For each value of k, we draw such a k-by-l random matrix $\mathbf{N}$ and compute the operator norm of $\mathbf{N}\mathbf{N}^T$. We compare this to the value of λ^* computed using Algorithm 3. In Table 6, we show the mean absolute error, averaged over $M = 40000$ runs for each value of k. More precisely, if $\hat{\lambda}^*_j$ is the operator norm of $\mathbf{N}\mathbf{N}^T$ from trial $j = 1, \ldots, M$, we record

$$\text{mean absolute error} = \frac{1}{M} \sum_{j=1}^{M} \frac{|\lambda^* - \hat{\lambda}^*_j|}{\lambda^*}. \tag{82}$$

We also record the average bias, defined as

$$\text{mean bias} = \frac{1}{M} \sum_{j=1}^{M} \frac{\lambda^* - \hat{\lambda}^*_j}{\lambda^*}. \tag{83}$$

In Figure 1, we plot the log error against $\log_2(k)$. The plot demonstrates a linear dependence. The slope of the line is estimated to be approximately -0.68.

Remark 8. *In [24], it is shown that for white noise the expected fluctations of the top eigenvalue of* $\mathbf{N}\mathbf{N}^T$ *are of the order* $k^{-2/3}$*, which implies the log-log plot would have a slope of* $-2/3$*. The observed slope of approximately* -0.68 *is a close match to this value.*

Remark 9. *For all values of* k*, the bias is positive. In other words, the estimated values* $\hat{\lambda}_k^*$ *tend to underestimate* λ^**.*

5.4 Finite sample accuracy of spiked model parameters

In this experiment, we test the finite sample accuracy of parameter estimation in the spiked random matrix model. We consider k-by-l random matrices of the form $\mathbf{Y} = \mathbf{X} + \mathbf{N}$, where $\mathbf{X} = \theta\mathbf{u}\mathbf{v}^T$ is a rank 1 "signal" matrix with uniformly random singular vectors $\mathbf{u}$ and $\mathbf{v}$, and $\mathbf{N} = \mathbf{A}^{1/2}\mathbf{G}\mathbf{B}^{1/2}$ is a random Gaussian noise matrix with separable variance profile. We will denote by $\hat{\lambda}$

As we reviewed in Section 2.3, the top eigenvalue of $\mathbf{Y}\mathbf{Y}^T$ converges almost surely to a value λ satisfying $\theta^2 = 1/D(\lambda)$, in the limit $k/l \to \gamma$. Furthermore, if $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$ are the top singular vectors of $\mathbf{Y}$, then the absolute inner products $|\langle \mathbf{u}, \hat{\mathbf{u}} \rangle|$ and $|\langle \mathbf{v}, \hat{\mathbf{v}} \rangle|$ converge almost surely to $c \equiv s(\lambda_m)D(\lambda_m)/D'(\lambda_m)$ and $\underline{c} \equiv \underline{s}(\lambda_m)D(\lambda_m)/D'(\lambda_m)$, respectively. We remind the reader that we define the D-transform by $D(\lambda) = \lambda s(\lambda)\underline{s}(\lambda)$.

We test the accuracy of these formulas for finite and increasing values of k and l. Using Algorithm 4 for evaluating $s(\lambda)$ and $s'(\lambda)$, we can easily evaluate $D(\lambda)$. Proposition 1 shows that for a specified parameter θ, Newton's root-finding algorithm may be used to solve for the asymptotic $\lambda = 1/D^{-1}(\theta^2)$. The asymptotic values c and $\underline{c}$ are then evaluated from their respective formulas.

We compare these asymptotic values to the top eigenvalue of $\mathbf{N}\mathbf{N}^T$ and the cosines between the singular vectors of $\mathbf{X}$ and $\mathbf{Y}$, for randomly generated data. We again use a model where $p = n = 2$, and both ν and $\underline{\nu}$ have point masses at 2 and 3, each with weight 1/2; and set $\gamma = 1/2$. We generate matrices of size k-by-l, where k grows and $l = k/\gamma = 2k$. We generate a signal matrix $\mathbf{X} = \theta\mathbf{u}\mathbf{v}^T$ where $\mathbf{u}$ and $\mathbf{v}$ are uniformly random unit vectors in $\mathbb{R}^k$ and $\mathbb{R}^l$, respectively; and $\theta = \sqrt{1/D(\lambda^*) + 20}$, ensuring a detectable signal.

For each value of k, we draw such a k-by-l random matrix $\mathbf{Y} = \mathbf{X} + \mathbf{N}$ and compute the operator norm of $\mathbf{Y}\mathbf{Y}^T$. We compare this to the asymptotic value of λ. We also compare the values $|\langle \mathbf{u}, \hat{\mathbf{u}} \rangle|$ and $|\langle \mathbf{v}, \hat{\mathbf{v}} \rangle|$ to c and $\underline{c}$, respectively. In Table 7, we show the mean absolute errors of these estimates, averaged over $M = 40000$ runs for each value of k. More precisely, if $\hat{\lambda}_j$ is the operator norm of $\mathbf{Y}\mathbf{Y}^T$ from trial $j = 1, \ldots, M$, and $\hat{c}_j$ and $\hat{\underline{c}}_j$ are the left and right cosines, respectively, we record the mean absolute errors:

$$\frac{1}{M}\sum_{j=1}^{M}\frac{|\lambda - \hat{\lambda}_j|}{\lambda}, \quad \frac{1}{M}\sum_{j=1}^{M}\frac{|c - \hat{c}_j|}{c}, \quad \frac{1}{M}\sum_{j=1}^{M}\frac{|\underline{c} - \hat{\underline{c}}_j|}{\underline{c}}. \tag{84}$$

In Figure 2, we plot the log errors against $\log_2(k)$. The plots all demonstrate a linear dependence. The slope of each line is estimated to be approximately -0.50.

Remark 10. *In [3], it is shown that the expected fluctations of the top eigenvalue of* $\mathbf{Y}\mathbf{Y}^T$ *are of the order* $k^{-1/2}$*, which implies the log-log plot would have a slope of* $-1/2$*. In [2], it is shown that in the case of white noise the expected fluctations of the cosines are also of order* $k^{-1/2}$*, also leading to a slope of* $-1/2$*. Our observed slopes closely match these values.*

5.5 One-sided weights

While Algorithm 3 computes the SSDT for an arbitrary separable variance profile, a special case of this problem that arises in certain applications is when $\mathbf{B} = \mathbf{I}_n$; that is, the variance profile has one-sided weights. In [15], it is shown that the SSDT may be evaluated by finding the minimizer v^* of the function

$$z(v) = \frac{-1}{v} + \gamma\sum_{i=1}^{p}\frac{a_i\omega_i}{1 + a_i v} \tag{85}$$

on the interval $(-1/a^*, 0)$, and then setting $\lambda_* = z(v^*)$. The minimizer v^* is the unique root of the monotonic function

$$z'(v) = \frac{1}{v^2} - \gamma\sum_{i=1}^{p}\frac{a_i^2\omega_i}{(1 + a_i v)^2} \tag{86}$$

on the interval $(-1/a^*, 0)$.

When viable, this approach has obvious advantages over Algorithm 3, namely that it finds the root of a function which can be evaluated in closed form rather than by nested applications of Newton's method. Properly applied, it should be substantially faster than Algorithm 3. However, the paper [15] does not analyze the behavior of the function (86), beyond observing that it is monotonic.

We compare the method from [15] for evaluating λ_* based on the minimizer of $z(v)$ to Algorithm 3. To find the root of $z'(v)$, we employ bisection until the error is approximately square root of machine epsilon, and then use Newton's method to achieve full accuracy. In our experiment, we set $\gamma = 1/2$, and for each value of p we generate the a_i's uniformly randomly on $[0, 1]$ and take uniform weights $\omega_i = 1/p$. For each value of p, we solve the problem using each method for 10 runs, and average the timings of all the runs; the timings are presented in Table 8.

The results of the experiment are extremely encouraging. They suggest that for one-sided weights, working with the function (86) yields a substantially faster computation of the SSDT than the general master equations (22) – (24). However, a more detailed analysis of the function (86) must be carried out to justify such an approach and show that a fast algorithm is viable for general a_i and ω_i; this is beyond the scope of the current work.

6 Conclusion

We have introduced an algorithm for rapidly computing the spectral signal detection threshold λ^* in signal-plus-noise random matrix models. We have considered a class of random noise matrices with separable variance structure, which arise often in applications. Our algorithm is based on an implicit characterization of λ^* derived from the master equations for the Stieltjes transform $s(\lambda)$. Several nested applications of Newton's method are applied to evaluate λ^* and auxiliary parameters. Fast algorithms are also introduced to evaluate $s(\lambda)$ and $s'(\lambda)$, the Stieltjes transform and its derivative, at real values $\lambda > \lambda^*$. We have demonstrated the rapid convergence of these methods and their linear scaling in numerical tests. We have also shown the effects of random fluctations for increasing values of p and n.

Acknowledgements

I acknowledge support from NSF BIGDATA award IIS 1837992 and BSF award 2018230. I thank Edgar Dobriban for helpful discussions and for pointing out the method from [15]. I also thank the reviewers for their helpful comments on the manuscript.

References

[1] Zhidong Bai and Jack W. Silverstein. *Spectral Analysis of Large Dimensional Random Matrices*. Springer Series in Statistics. Springer, 2009.

[2] Zhigang Bao, Xiucai Ding, and Ke Wang. Singular vector and singular subspace distribution for the matrix denoising model. *The Annals of Statistics*, 49(1):370–392, 2021.

[3] Florent Benaych-Georges and Raj Rao Nadakuditi. The singular values and vectors of low rank perturbations of large rectangular random matrices. *Journal of Multivariate Analysis*, 111:120–135, 2012.

[4] Tejal Bhamre, Teng Zhang, and Amit Singer. Denoising and covariance estimation of single particle cryo-EM images. *Journal of Structural Biology*, 195(1):72–81, 2016.

[5] Ezio Biglieri, Robert Calderbank, Anthony Constantinides, Andrea Goldsmith, Arogyaswami Paulraj, and H. Vincent Poor. *MIMO Wireless Communications*. Cambridge University Press, 2007.

[6] Andreas Buja and Nermin Eyuboglu. Remarks on parallel analysis. *Multivariate Behavioral Research*, 27(4):509–540, 1992.

[7] Lucilio Cordero-Grande. MIXANDMIX: numerical techniques for the computation of empirical spectral distributions of population mixtures. *Computational Statistics & Data Analysis*, 141:1–11, 2020.

[8] Romain Couillet and Merouane Debbah. *Random Matrix Methods for Wireless Communications*. Cambridge University Press, 2011.

[9] Romain Couillet and Walid Hachem. Analysis of the limiting spectral measure of large random matrices of the separable covariance type. *Random Matrices: Theory and Applications*, 3(4), 2014.

[10] Germund Dahlquist and Åke Björck. *Numerical Methods*. Prentice Hall, Inc., 1974.

[11] Xiucai Ding and Fan Yang. Spiked separable covariance matrices and principal components. *The Annals of Statistics*, 49(2):1113–1138, 2021.

[12] Edgar Dobriban. Efficient computation of limit spectra of sample covariance matrices. *Random Matrices: Theory and Applications*, 4(4), 2015.

[13] Edgar Dobriban. Permutation methods for factor analysis and PCA. *The Annals of Statistics*, 48(5):2824–2847, 2020.

[14] Edgar Dobriban, William Leeb, and Amit Singer. Optimal prediction in the linearly transformed spiked model. *Annals of Statistics*, 48(1):491–513, 2020.

[15] Edgar Dobriban and Art B. Owen. Deterministic parallel analysis: an improved method for selecting factors and principal components. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2018.

[16] David L. Donoho, Matan Gavish, and Iain M Johnstone. Optimal shrinkage of eigenvalues in the spiked covariance model. *Annals of Statistics*, 46(6), 2018.

[17] Noureddine El Karoui. Spectrum estimation for large dimensional covariance matrices using random matrix theory. *The Annals of Statistics*, 36(6):2757–2790, 2008.

[18] Matan Gavish and David L. Donoho. Optimal shrinkage of singular values. *IEEE Transactions on Information Theory*, 63(4):2137–2152, 2017.

[19] Stuart Geman. A limit theorem for the norm of random matrices. *The Annals of Probability*, 8(2):252–261, 1980.

[20] David Hong, Laura Balzano, and Jeffrey A. Fessler. Towards a theoretical analysis of PCA for heteroscedastic data. In *54th Annual Allerton Conference on Communication, Control, and Computing*, pages 496–503. IEEE, 2016.

[21] David Hong, Laura Balzano, and Jeffrey A. Fessler. Asymptotic performance of PCA for high-dimensional heteroscedastic data. *Journal of Multivariate Analysis*, 167:435–452, 2018.

[22] David Hong, Laura Balzano, and Jeffrey A. Fessler. Optimally Weighted PCA for High-Dimensional Heteroscedastic Data. *arXiv preprint arXiv:1810.12862*, 2018.

[23] John L. Horn. A rationale and test for the number of factors in factor analysis. *Psychometrika*, 30(2):179–185, 1965.

[24] Iain M Johnstone. On the distribution of the largest eigenvalue in principal components analysis. *Annals of Statistics*, 29(2):295–327, 2001.

[25] Shira Kritchman and Boaz Nadler. Determining the number of components in a factor model from limited noisy data. *Chemometrics and Intelligent Laboratory Systems*, 94(1):19–32, 2008.

[26] Shira Kritchman and Boaz Nadler. Non-parametric detection of the number of signals: Hypothesis testing and random matrix theory. *IEEE Transactions on Signal Processing*, 57(10):3930–3941, 2009.

[27] Olivier Ledoit and Michael Wolf. Spectrum estimation: A unified framework for covariance matrix estimation and PCA in large dimensions. *Journal of Multivariate Analysis*, 139:360–384, 2015.

[28] Olivier Ledoit and Michael Wolf. Numerical implementation of the QuEST function. *Computational Statistics & Data Analysis*, 115:199–223, 2017.

[29] William Leeb. Optimal singular value shrinkage for operator norm loss. *arXiv preprint arXiv:2005.11807*, 2020.

[30] William Leeb. Matrix denoising for weighted loss functions and heterogeneous signals. *SIAM Journal on Mathematics of Data Science*, Accepted, 2021.

[31] William Leeb and Elad Romanov. Optimal spectral shrinkage and PCA with heteroscedastic noise. *IEEE Transactions on Information Theory*, 67(5):3009–3037, 2021.

[32] Vladimir A Marchenko and Leonid A Pastur. Distribution of eigenvalues for some sets of random matrices. *Mat. Sb.*, 114(4):507–536, 1967.

[33] Raj Rao Nadakuditi. OptShrink: An algorithm for improved low-rank signal matrix denoising by optimal, data-driven singular value shrinkage. *IEEE Transactions on Information Theory*, 60(5):3002–3018, 2014.

[34] Yurii Nesterov. *Lectures on Convex Optimization*, volume 137 of *Springer Optimization and Its Applications*. Springer, 2nd edition, 2018.

[35] Alexei Onatski. Determining the number of factors from empirical distribution of eigenvalues. *The Review of Economics and Statistics*, 92(4):1004–1016, 2010.

[36] Alexei Onatski, Marcelo J. Moreira, and Marc Hallin. Asymptotic power of sphericity tests for high-dimensional data. *The Annals of Statistics*, 41(3):1204–1231, 2013.

[37] Damien Passemier and Jian-Feng Yao. On determining the number of spikes in a high-dimensional spiked population model. *Random Matrices: Theory and Applications*, 1(01):1150002, 2012.

[38] Debashis Paul. Asymptotics of sample eigenstructure for a large dimensional spiked covariance model. *Statistica Sinica*, 17(4):1617–1642, 2007.

[39] Debashis Paul and Jack W. Silverstein. No eigenvalues outside the support of the limiting empirical spectral distribution of a separable covariance matrix. *Journal of Multivariate Analysis*, 100:37–57, 2009.

[40] N. Raj Rao and Alan Edelman. The polynomial method for random matrices. *Foundations of Computational Mathematics*, 8:649–702, 2008.

[41] Andrey A. Shabalin and Andrew B. Nobel. Reconstruction of a low-rank matrix in the presence of Gaussian noise. *Journal of Multivariate Analysis*, 118:67–76, 2013.

[42] Jack W. Silverstein. Strong convergence of the empirical distribution of eigenvalues of large dimensional random matrices. *Journal of Multivariate Analysis*, 55:331–339, 1995.

[43] Jack W. Silverstein and Zhidong Bai. On the empirical distribution of eigenvalues of a class of large dimensional random matrices. *Journal of Multivariate Analysis*, 54:175–192, 1995.

[44] Rahul Tandra and Anant Sahai. SNR walls for signal detection. *IEEE Journal of Selected Topics in Signal Processing*, 2(1):4–17, 2008.

[45] Terence Tao. *Topics in Random Matrix Theory*. American Mathematical Society, 2012.

[46] Tevfik Yücek and Hüseyin Arslan. A survey of spectrum sensing algorithms for cognitive radio applications. *IEEE Communications Surveys & Tutorials*, 11(1):116–130, 2009.

[47] Yonghong Zeng and Ying-Chang Liang. Covariance based signal detections for cognitive radio. In *2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, pages 202–207. IEEE, 2007.

[48] Yonghong Zeng and Ying-Chang Liang. Eigenvalue-based spectrum sensing algorithms for cognitive radio. *IEEE Transactions on Communications*, 57(6):1784–1793, 2009.

[49] Anru Zhang, T. Tony Cai, and Yihong Wu. Heteroskedastic PCA: Algorithm, optimality, and applications. *arXiv preprint arXiv:1810.08316*, 2018.

Table 1: Error after each iterate in evaluation of e^*_λ.

m	$T_\lambda(e_m)$	$\mathrm{err}(e_m, e^*_\lambda)$
1	2.08e-01	4.26e-02
2	4.34e-02	1.11e-02
3	2.55e-03	6.90e-04
4	9.66e-06	2.63e-06
5	1.40e-10	3.80e-11
6	-2.75e-16	-1.23e-16

Table 2: Error after each iterate in evaluation of $t(\lambda)$.

m	$R_\lambda(e_m)$	$\mathrm{err}(e_m, t(\lambda))$
1	-3.77e-01	3.71e-02
2	-6.89e-02	8.34e-03
3	-3.40e-03	4.34e-04
4	-9.24e-06	1.18e-06
5	-6.84e-11	8.73e-12
6	-5.55e-15	7.85e-16

Table 3: Error after each iterate in evaluation of λ^*.

m	$Q(\lambda_m)$	$\mathrm{err}(\lambda_m, \lambda^*)$
1	1.25e+00	-3.10e-01
2	3.10e-01	-1.03e-01
3	3.12e-02	-1.15e-02
4	3.92e-04	-1.46e-04
5	6.34e-08	-2.37e-08
6	2.00e-15	-2.11e-16

Table 4: Error after each iterate in evaluation of $e(\lambda)$.

m	$F_\lambda(e_m)$	$\mathrm{err}(e_m, e(\lambda))$
1	4.71e-01	-1.00e+00
2	8.86e-03	-1.93e-02
3	6.42e-06	-1.40e-05
4	3.43e-12	-7.50e-12
5	4.44e-16	-1.10e-15

Table 5: Timings in seconds for evaluating λ^* and $s(\lambda)$ at 100 values of λ.

$\log_2(n)$	Timing, λ_*	Timing, $s(\lambda)$
18	4.30e-01	1.93e-01
19	8.67e-01	4.27e-01
20	1.89e+00	1.66e+00
21	3.95e+00	4.20e+00
22	8.41e+00	1.02e+01
23	1.70e+01	2.09e+01
24	3.43e+01	4.19e+01

Table 6: Errors and bias in estimating λ^*.

$\log_2(k)$	Error	Bias
5	8.73e-02	7.72e-02
6	5.41e-02	4.75e-02
7	3.39e-02	2.93e-02
8	2.12e-02	1.82e-02
9	1.32e-02	1.13e-02
10	8.20e-03	6.97e-03
11	5.12e-03	4.32e-03

Table 7: Errors in estimating λ, c, and $\underline{c}$.

$\log_2(k)$	sing. val.	left cos.	right cos.
5	7.68e-02	1.80e-02	2.26e-02
6	5.43e-02	1.27e-02	1.59e-02
7	3.85e-02	9.10e-03	1.13e-02
8	2.74e-02	6.38e-03	7.98e-03
9	1.92e-02	4.53e-03	5.61e-03
10	1.36e-02	3.20e-03	3.99e-03
11	9.58e-03	2.25e-03	2.81e-03

Table 8: Timings in seconds of Algorithm 3 and the solution via minimizing $z(v)$.

$\log_2(p)$	Alg. 3	Min. $z(v)$
18	2.67e-01	2.38e-02
19	5.29e-01	4.63e-02
20	1.14e+00	1.30e-01
21	2.43e+00	3.20e-01
22	5.37e+00	6.94e-01
23	1.09e+01	1.38e+00
24	2.18e+01	2.73e+00

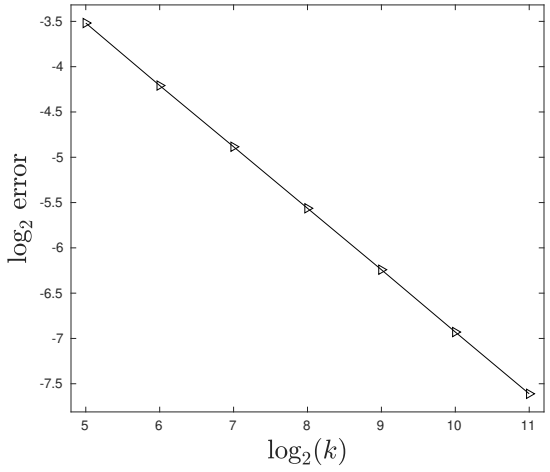

Figure 1: Log errors for estimating λ^*.

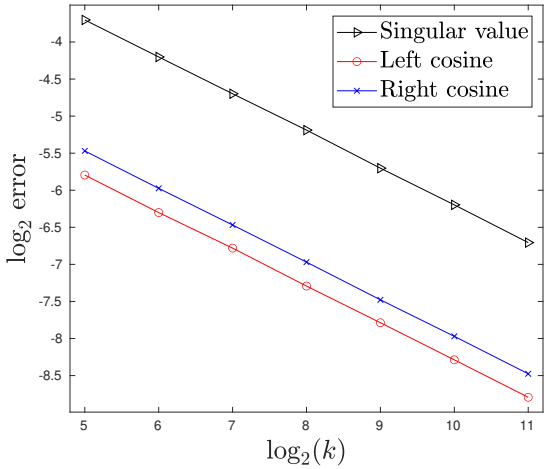

Figure 2: Log errors for estimating λ, c, and $\underline{c}$.