2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS) | 978-1-6654-5519-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/FOCS54457.2022.00016

Testing Positive Semidefiniteness Using Linear Measurements

Deanna Needell
University of California Los Angeles
Los Angeles CA, United States
deanna@math.ucla.edu

William Swartworth
University of California Los Angeles
Los Angeles CA, United States
wswartworth@math.ucla.edu

David P. Woodruff
Carnegie Mellon University
Pittsburgh PA, United States
dwoodruf@cs.cmu.edu

***Abstract*—We study the problem of testing whether a symmetric $d \times d$ input matrix A is symmetric positive semidefinite (PSD), or is ϵ-far from the PSD cone, meaning that $\lambda_{\min}(A) \leq -\epsilon \|A\|_p$, where $\|A\|_p$ is the Schatten-p norm of A. In applications one often needs to quickly tell if an input matrix is PSD, and a small distance from the PSD cone may be tolerable. We consider two well-studied query models for measuring efficiency, namely, the matrix-vector and vector-matrix-vector query models. We first consider one-sided testers, which are testers that correctly classify any PSD input, but may fail on a non-PSD input with a tiny failure probability. Up to logarithmic factors, in the matrix-vector query model we show a tight $\widetilde{\Theta}(1/\epsilon^{p/(2p+1)})$ bound, while in the vector-matrix-vector query model we show a tight $\widetilde{\Theta}(d^{1-1/p}/\epsilon)$ bound, for every $p \geq 1$. We also show a strong separation between one-sided and two-sided testers in the vector-matrix-vector model, where a two-sided tester can fail on both PSD and non-PSD inputs with a tiny failure probability. In particular, for the important case of the Frobenius norm, we show that any one-sided tester requires $\widetilde{\Omega}(\sqrt{d}/\epsilon)$ queries. However we introduce a bilinear sketch for two-sided testing from which we construct a Frobenius norm tester achieving the optimal $\widetilde{O}(1/\epsilon^2)$ queries. We also give a number of additional separations between adaptive and non-adaptive testers. Our techniques have implications beyond testing, providing new methods to approximate the spectrum of a matrix with Frobenius norm error using dimensionality reduction in a way that preserves the signs of eigenvalues.**

I. Introduction

A real-valued matrix $A \in \mathbb{R}^{n \times n}$ is said to be Positive Semi-Definite (PSD) if it defines a non-negative quadratic form, namely, if $x^T Ax \geq 0$ for all x. If A is symmetric, the setting on which we focus, this is equivalent to the eigenvalues of A being non-negative. Multiple works [1]–[3] have studied the problem of testing whether a real matrix is PSD, or is far from being PSD, and this testing problem has numerous applications, including to faster algorithms for linear systems and linear algebra problems, detecting the existence of community structure, ascertaining local convexity, and differential equations; we refer the reader to [3] and the references therein.

We study this testing problem under two fundamental query models. In the matrix-vector model, one is given implicit access to a matrix A and may query A by choosing a vector v and receiving the vector Av. In the vector-matrix-vector model one chooses a pair of vectors (v, w) and queries the bilinear form associated to A. In other words the value of the query is $v^T Aw$. In both models, multiple, adaptively-chosen queries can be made, and the goal is to minimize the number of queries to solve a certain task. These models are standard computational models in the numerical linear algebra community, see, e.g., [2] where PSD testing was studied in the matrix-vector query model. These models were recently formalized in the theoretical computer science community in [4], [5], though similar models have been studied in numerous fields, such as the number of measurements in compressed sensing, or the sketching dimension of a streaming algorithm. The matrix-vector query and vector-matrix-vector query models are particularly relevant when the input matrix A is not given explicitly.

A natural situation occurs when A is presented implicitly as a the Hessian of a function $f : \mathbb{R}^d \to \mathbb{R}^d$ at a point x_0, where f could be the loss function of a neural network for example. One might want to quickly distinguish between a proposed optimum of f truly being a minimum, or being a saddle point with a direction of steep downward curvature. Our query model is quite natural in this context. A Hessian-vector product is efficient to compute using automatic differentiation techniques. A vector-matrix-vector product corresponds to a single second derivative computation, $D^2 f(v, w)$. This can be approximated using 4 function queries by the finite difference approximation $D^2 f(v, w) \approx \frac{f(x_0+hv+hw)-f(x_0+hv)-f(x_0+hw)+f(x_0)}{h^2}$, where h is small.

While there are numerically stable methods for computing the spectrum of a symmetric matrix, and thus determining if it is PSD, these methods can be prohibitively slow for very large matrices, and require a large number of matrix-vector or vector-matrix-vector products. Our goal is to obtain significantly more efficient algorithms in these models, and we approach this problem from a property testing perspective. In particular, we focus on the following version of the PSD-testing problem. In what follows, $\|A\|_p = (\sum_{i=1}^n \sigma_i^p)^{1/p}$ is the Schatten-p norm of A, where the σ_i are the singular values of A.

Definition 1. *For $p \in [1, \infty]$, an (ϵ, ℓ_p)-tester is an algorithm that makes either matrix-vector or vector-matrix-vector queries to a real symmetric matrix A, and outputs True with*

Authors DN and WS were partially supported by NSF DMS #2011140 and NSF DMS #2108479. DW was supported by NSF CCF #1815840 and Office of Naval Research Grant N00014-18-1-2562.

978-1-6654-5519-0/22/$31.00 ©2022 IEEE
DOI 10.1109/FOCS54457.2022.00016

at least $2/3$ probability if A is PSD, and outputs False with $2/3$ probability if A is $\epsilon \|A\|_p$-far in spectral distance from the PSD cone, or equivalently, if the minimum eigenvalue $\lambda_{\min}(A) \leq -\epsilon \|A\|_p$. If the tester is guaranteed to output True on all PSD inputs (even if the input is generated by an adversary with access to the random coins of the tester), then the tester has one-sided error. Otherwise it has two-sided error. When ϵ is clear from the context we will often drop the ϵ and simply refer to an ℓ_p-tester.

Our work fits more broadly into the growing body of work on property testing for linear algebra problems, see, for example [3], [6], [7]. However, a key difference is that we focus on matrix-vector and vector-matrix-vector query models, which might be more appropriate than the model in the above works which charges a cost of 1 for reading a single entry. Indeed, such models need to make the assumption that the entries of the input are bounded by a constant or slow-growing function of n, as otherwise strong impossibility results hold. This can severely limit the applicability of such algorithms to real-life matrices that do not have bounded entries; indeed, even a graph Laplacian matrix with a single degree that is large would not fit into the above models. In contrast, we use the matrix-vector and vector-matrix-vector models, which are ideally suited for modern machines such as graphics processing units and when the input matrix cannot fit into RAM, and are standard models in scientific computing, see, e.g., [8].

While we focus on vector-matrix-vector queries, our results shed light on several other natural settings. Many of our results are in fact tight for general linear measurements which vectorize the input matrix and apply adaptively chosen linear forms to it. For long enough streams, the best known single or multi-pass algorithms for any problem in the turnstile streaming model form a sketch using general linear measurements, and with additional restrictions, it can be shown that the optimal multi-pass streaming algorithm just adaptively chooses general linear measurements [9], [10]. Therefore, it is plausible that many of our vector-matrix-vector algorithms give tight single pass streaming bounds, given that vector-matrix-vector queries are a special case of general linear measurements, and that many our lower bounds are tight even for general linear measurements.

Moreover our vector-matrix-vector algorithms lead to efficient communication protocols for deciding whether a distributed sum of matrices is PSD, provided that exact vector-matrix-vector products may be communicated. While we expect our methods to be stable under small perturbations (i.e., when the vector-matrix-vector products are slightly inexact), we leave the full communication complexity analysis to future work.

A. Our Contributions

We study PSD-testing in the matrix-vector and vector-matrix-vector models. In particular, given a real symmetric matrix A, and $p \in [1, \infty]$, we are interested in deciding between (i) A is PSD and (ii) A has an eigenvalue less than $-\epsilon \|A\|_p$.

Vector-matrix-vector queries	
Adaptive, one-sided ℓ_p	$\widetilde{\Theta}(\frac{1}{\epsilon} d^{1-1/p})$
Non-adaptive, one-sided ℓ_p	$\widetilde{\Theta}(\frac{1}{\epsilon^2} d^{2-2/p})$
Adaptive, two-sided ℓ_2	$\widetilde{\Theta}(\frac{1}{\epsilon^2})^*$
Non-adaptive, two-sided ℓ_2	$\widetilde{\Theta}(\frac{1}{\epsilon^4})^*$
Adaptive, two-sided ℓ_p, $2 \leq p < \infty$	$\widetilde{\Theta}(\frac{1}{\epsilon^2} d^{1-2/p})^*$
Matrix-vector queries	
Adaptive one-sided ℓ_p	$\widetilde{O}((1/\epsilon)^{p/(2p+1)} \log d)$, $\Omega((1/\epsilon)^{p/(2p+1)})$
Adaptive one-sided ℓ_1	$\widetilde{\Theta}((1/\epsilon)^{1/3})$
Non-adaptive one-sided ℓ_p	$\Theta(\frac{1}{\epsilon} d^{1-1/p})$

TABLE I
* INDICATES THAT THE LOWER BOUND HOLDS FOR GENERAL LINEAR MEASUREMENTS.

a) Tight Bounds for One-sided Testers: We make particular note of the distinction between one-sided and two-sided testers. In some settings one is interested in a tester that produces one-sided error. When such a tester outputs False, it must be able to produce a proof that A is not PSD. The simplest such proof is a witness vector v such that $v^T A v < 0$, and indeed we observe that in the matrix-vector model, any one-sided tester can produce such a v when it outputs False. This may be a desirable feature if one wishes to apply these techniques to saddle point detection for example: given a point that is not a local minimum, it would be useful to produce a descent direction so that optimization may continue. In the vector-matrix-vector model the situation is somewhat more complicated in general, but all of our one-sided testers produce a witness vector whenever they output False.

We provide *optimal bounds* for one-sided testers for both matrix-vector and vector-matrix-vector models. The bounds below are stated for constant probability algorithms. Here $\widetilde{O}(f) = f \cdot \text{poly}(\log f)$.

1) In the matrix-vector query model, we show that up to a factor of $\log d$, $\widetilde{\Theta}(1/\epsilon^{p/(2p+1)})$ queries are necessary and sufficient for an ℓ_p-tester for any $p \geq 1$. In the $p = 1$ case, we note that the $\log d$ factor may be removed.
2) In the vector-matrix-vector query model, we show that $\widetilde{\Theta}(d^{1-1/p}/\epsilon)$ queries are necessary and sufficient for an ℓ_p-tester for any $p \geq 1$. Note that when $p = 1$ we obtain a very efficient $\widetilde{O}(1/\epsilon)$-query algorithm. In particular, our tester for $p = 1$ has query complexity independent of the matrix dimensions, and we show a sharp phase transition for $p > 1$, showing in some sense that $p = 1$ is the largest value of p possible for one-sided queries.

The matrix-vector query complexity is very different than the vector-matrix-vector query complexity, as the query complexity is $\text{poly}(1/\epsilon)$ for any $p \geq 1$, which captures the fact that each matrix-vector query response reveals more information than that of a vector-matrix-vector query, though a priori it

was not clear that such responses in the matrix-vector model could not be compressed using vector-matrix-vector queries.

b) An Optimal Bilinear Sketch for Two-Sided Testing: Our main technical contribution for two-sided testers is a bilinear sketch for PSD-testing with respect to the Frobenius norm, i.e., $p = 2$. We consider a Gaussian sketch $G^T AG$, where G has small dimension $\widetilde{O}(\frac{1}{\epsilon^2})$. By looking at the smallest eigenvalue of the sketch, we are able to distinguish between A being PSD and being ϵ-far from PSD. Notably this tester may reject even when $\lambda_{\min}(G^T AG) > 0$, which results in a two-sided error guarantee. This sketch allows us to obtain tight two-sided bounds in the vector-matrix-vector model for $p \geq 2$, both for adaptive and non-adaptive queries.

c) Separation Between One-Sided and Two-Sided Testers: Surprisingly, we show a separation between one-sided and two-sided testers in the vector-matrix-vector model. For the important case of the Frobenius norm, i.e., $p = 2$, we utilize our bilinear sketch to construct an $\widetilde{O}(1/\epsilon^2)$ query two-sided tester, whereas by our results above, any adaptive one-sided tester requires at least $\Omega(\sqrt{d}/\epsilon)$ queries.

We also show that for any $p > 2$, any possibly adaptive two-sided tester requires $d^{\Omega(1)}$ queries for constant ϵ, and thus in some sense, $p = 2$ is the largest value of p possible for two-sided queries.

d) On the Importance of Adaptivity: We also study the role of adaptivity in both matrix-vector and vector-matrix-vector models. In both the one-sided and two-sided vector-matrix-vector models we show a quadratic separation between adaptive and non-adaptive testers, which is the largest gap possible for any vector-matrix-vector problem arising from a rotationally invariant distribution [4].

In the matrix-vector model, each query reveals more information about A than in the vector-matrix-vector model, allowing for even better choices for future queries. Thus we have an even larger gap between adaptive and non-adaptive testers in this setting.

e) Spectrum Estimation: While the two-sided tester discussed above yields optimal bounds for PSD testing, it does not immediately give a way to estimate the negative eigenvalue when it exists. Via a different approach, we show how to give such an approximation with $\epsilon \|A\|_F$ additive error. In fact, we show how to approximate all of the top k eigenvalues of A using $O(k^2 \text{poly}\frac{1}{\epsilon})$ non-adaptive vector-matrix-vector queries, which may be of independent interest.

We note that this gives an $O(k^2 \text{poly}\frac{1}{\epsilon})$ space streaming algorithm for estimating the top k eigenvalues of A to within additive Frobenius error. Prior work yields a similar guarantee for the singular values [11], but cannot recover the signs of eigenvalues.

B. Our Techniques

a) Matrix-Vector Queries: For the case of adaptive matrix-vector queries, we show that Krylov iteration starting with a single random vector yields an optimal ℓ_p-tester for all p. Interestingly, our analysis is able to beat the usual Krylov matrix-vector query bound for approximating the top eigenvalue, as we modify the usual polynomial analyzed for eigenvalue estimation to implicitly implement a *deflation* step of all eigenvalues above a certain threshold. We do not need to explicitly know the values of the large eigenvalues in order to deflate them; rather, it suffices that there exists a low degree polynomial in the Krylov space that implements this deflation.

Further, we show that our technique is tight for all $p \geq 1$ by showing that any smaller number of matrix-vector products would violate a recent lower bound of [12] for approximating the smallest eigenvalue of a Wishart matrix. This lower bound applies even to two-sided testers.

b) Vector-Matrix-Vector Queries: We start by describing our result for $p = 1$. We give one of the first examples of an algorithm in the vector-matrix-vector query model that leverages adaptivity in an interesting way. Most known algorithms in this model work non-adaptively, either by applying a bilinear sketch to the matrix, or by making many independent queries in the case of Hutchinson's trace estimator [13]. Indeed, the algorithm of [11] works by computing $G^T AG$ for a Gaussian matrix G with $1/\epsilon$ columns, and arguing that all eigenvalues that are at least $\epsilon \|A\|_1$ can be estimated from the sketch. The issue with this approach is that it uses $\Omega(1/\epsilon^2)$ queries and this bound is tight for non-adaptive testers! One could improve this by running our earlier matrix-vector algorithm on top of this sketch, without ever explicitly forming the $1/\epsilon \times 1/\epsilon$ matrix $G^T AG$; however, this would only give an $O(1/\epsilon^{4/3})$ query algorithm.

To achieve our optimal $\widetilde{O}(1/\epsilon)$ complexity, our algorithm instead performs a novel twist to Oja's algorithm [14], the latter being a stochastic gradient descent (SGD) algorithm applied to optimizing the quadratic form $f(x) = x^T Ax$ over the sphere. In typical applications, the randomness of SGD arises via randomly sampling from a set of training data. In our setting, we instead artificially introduce randomness at each step, by computing the projection of the gradient onto a randomly chosen direction. This idea is implemented via the iteration

$$x^{(k+1)} = x^k - \eta(g^T Ax^k)g \text{ where } g \sim \mathcal{N}(0,1) \qquad (1)$$

for a well-chosen step size η. If f ever becomes negative before reaching the maximum number of iterations, then the algorithm outputs False, otherwise it outputs True. For $p = 1$, we show that this scheme results in an optimal tester (up to logarithmic factors). Our proof uses a second moment analysis to analyze a random walk, that is similar in style to [15], though our analysis is quite different. Whereas [15] considers an arbitrary i.i.d. stream of unbiased estimators to A (with bounded variance), our estimators are simply $gg^T A$, which do not seem to have been considered before. We leverage this special structure to obtain a better variance bound on the iterates throughout the first $\widetilde{O}(1/\epsilon)$ iterations, where each iteration can be implemented with a single vector-matrix-vector query. Our algorithm and analysis gives a new method for the fundamental problem of approximating eigenvalues.

Our result for general $p > 1$ follows by relating the Schatten-p norm to the Schatten-1 norm and invoking the

algorithm above with a different setting of ϵ. We show our method is optimal by proving an $\Omega(d^{2-2/p}/\epsilon^2)$ lower bound for non-adaptive one-sided testers, and then using a theorem in [5] which shows that adaptive one-sided testers can give at most a quadratic improvement. We note that one could instead use a recent streaming lower bound of [16] to prove this lower bound, though such a lower bound would depend on the bit complexity.

c) Two-Sided Testers.: The key technical ingredient behind all of our two-sided testers is a bilinear sketch for PSD-testing. Specifically, we show that a sketch of the form G^TAG with $G \in R^{d\times k}$ is sufficient for obtaining a two-sided tester for $p = 2$. In contrast to the $p = 1$ case, we do not simply output False when $\lambda_{\min} := \lambda_{\min}(G^TAG) < 0$ as such an algorithm would automatically be one-sided. Instead we require a criterion to detect when $\lambda_{\min}$ is suspiciously small. For this we require two results.

The first is a concentration inequality for $\lambda_{\min}(G^TAG)$ when A is PSD. We show that $\lambda_{\min} \geq \mathrm{Tr}(A) - \widetilde{O}(\sqrt{k})\,\|A\|_F$ with very good probability. This result is equivalent to bounding the smallest singular value of $A^{1/2}G$, which is a Gaussian matrix whose rows have different variances. Although many similar bounds for constant variances exist in the literature [17], [18], we were not able to find a general bound that applies when A is not a multiple of the identity. In particular, most existing bounds do not seem to give the concentration around $\mathrm{Tr}(A)$ that we require.

When A has a negative eigenvalue of $-\epsilon$, we show that $\lambda_{\min} \leq \mathrm{Tr}(A) - \epsilon O(k)$. By combining these two results, we are able to take $k = \widetilde{O}(1/\epsilon^2)$, yielding a tight bound for non-adaptive testers in the vector-matrix-vector model. In fact this bound is even tight for general linear sketches, as we show by applying the results in [19].

We also utilize this bilinear sketch to give tight bounds for adaptive vector-matrix-vector queries, and indeed for general linear measurements. By first (implicitly) applying the sketch, and then shifting by an appropriate multiple of the identity we are able to reduce to the (ϵ^2, ℓ_1)-testing problem, which as described above may solved using $\widetilde{O}(1/\epsilon^2)$ queries.

d) Spectrum Estimation: A natural approach for approximating the eigenvalues of an $n \times n$ matrix A is to first compute a sketch G^TAG or a sketch G^TAH for Gaussian matrices G and H with a small number of columns. Both of these sketches appear in [11]. As noted above, G^TAG is a useful non-adaptive sketch for spectrum approximation, but the error in approximating each eigenvalue is proportional to the Schatten-1 norm of A. One could instead try to make the error depend on the Frobenius norm $\|A\|_2$ of A by instead computing G^TAH for independent Gaussian matrices G and H, but now G^TAH is no longer symmetric and it is not clear how to extract the signs of the eigenvalues of A from G^TAH. Indeed, [11] are only able to show that the *singular values* of G^TAH are approximately the same as those of A, up to additive $\epsilon\|A\|_2$ error. We thus need a new way to *preserve sign information of eigenvalues*.

To do this, we show how to use results for providing the best PSD low rank approximation to an input matrix A, where A need not be PSD and need not even be symmetric. In particular, in [20] it was argued that if G is a Gaussian matrix with $O(k/\epsilon)$ columns, then if one sets up the optimization problem $\min_{\text{rank k PSD } Y} \|AGYG^TA^T - A\|_F^2$, then the cost will be at most $(1+\epsilon)\|A_{k,+} - A\|_F^2$, where $A_{k,+}$ is the best rank-k PSD approximation to A. By further sketching on the left and right with so-called *affine embeddings* S and T, which have $\mathrm{poly}(k/\epsilon)$ rows and columns respectively, one can reduce this problem to $\min_{\text{rank k PSD } Y} \|SAGYG^TA^TT - SAT\|_F^2$, and now SAG, G^TA^TT and SAT are all $\mathrm{poly}(k/\epsilon) \times \mathrm{poly}(k/\epsilon)$ matrices so can be computed with a $\mathrm{poly}(k/\epsilon)$ number of vector-matrix-vector products. At this point the optimal Y can be found with no additional queries and its cost can be evaluated. By subtracting this cost from $\|A\|_F^2$, we approximate $\|A_{+,i}\|_F^2$, and $\|A_{-,i}\|_F^2$ for all $i \in [k]$, which in turn allows us to produce (signed) estimates for the eigenvalues of A.

When A is PSD, we note that Theorem 1.2 in [11] is able to reproduce our spectral approximation guarantee using sketching dimension $O(\frac{k^2}{\epsilon^8})$, compared to our sketch of dimension $O(\frac{k^2}{\epsilon^{12}})$. However as mentioned above, our guarantee is stronger in that it allows for the signs of the eigenvalues to be recovered, i.e., our guarantee holds even when A is not PSD. Additionally, we are able to achieve $O(\frac{k^2}{\epsilon^8})$ using just a single round of adaptivity.

e) Lower Bounds for One-sided Testers: To prove lower bounds for one-sided non-adaptive testers, we first show that a one-sided tester must be able to produce a witness whenever it outputs False. In the matrix-vector model, the witness is a vector v with $v^TAv < 0$, and in the vector-matrix-vector model, the witness is a PSD matrix M with $\langle M, A\rangle < 0$. In both cases we show that even for the simplest non-PSD spectrum $(-\lambda, 1, \ldots, 1)$, that it takes many queries to produce a witness when λ is small. In the matrix-vector model, our approach is simply to show that the $-\lambda$ eigenvector is typically far from the span of all queried vectors, when the number of queries is small. This will imply that A is non-negative on the queried subspace, which precludes the tester from producing a witness. In the vector-matrix-vector model our approach is similar, however now the queries take the form of inner products against rank one matrices $x_ix_i^T$. We therefore need to work within the space of symmetric matrices, and this requires a more delicate argument.

C. Additional Related Work

Numerous other works have considered matrix-vector queries and vector-matrix queries, see, e.g., [4], [12], [21]–[24]. We outline a few core areas here.

a) Oja's Algorithm: Several works have considered Oja's algorithm in the context of streaming PCA, [15], [25], [26]. [15] gives a tight convergence rate for iteratively approximating the top eigenvector of a PSD matrix, given an eigengap, and [26] extends this to a gap free result for k-PCA.

b) PSD Testing: As mentioned above, PSD-testing has been investigated in the bounded entry model, where one

assumes that the entries of A are bounded by 1 [3], and one is allowed to query the entries of A. This is a restriction of the vector-matrix-vector model that we consider where only coordinate vectors may be queried. However since we consider a more general query model, we are able to give a better adaptive tester – for us $\widetilde{O}(1/\epsilon)$ vector-matrix-vector queries suffice, beating the $\Omega(1/\epsilon^2)$ lower bound given in [3] for entry queries.

Another work on PSD-testing is that of [2], who construct a PSD-tester in the matrix-vector model. They first show how to approximate a general trace function $\sum f(\lambda_i)$ for sufficiently smooth f, by using a Chebyshev polynomial construction to approximate f in the sup-norm over an interval. This allows them to construct an ℓ_∞-tester by taking f to be a smooth approximation of a shifted Heaviside function. Unfortunately this approach is limited to ℓ_∞-testers, and does not achieve the optimal bound; they require $\Omega((\log d)/\epsilon)$ matrix-vector queries compared to the $\widetilde{O}((\log d)/\sqrt{\epsilon})$ queries achieved by Krylov iteration.

c) Spectrum Estimation: The closely-related problem of spectrum estimation has been considered several times, in the context of sketching the largest k elements of the spectrum [11] discussed above, and approximating the entire spectrum from entry queries in the bounded entry model [27].

D. Notation

A symmetric matrix A is positive semi-definite (PSD) if all eigenvalues are non-negative. We use Δ_+^d to represent the PSD-cone, which is the subset of $d \times d$ symmetric matrices that are PSD.

For a matrix A we use $\|A\|_p$ to denote the Schatten p-norm, which is the ℓ_p norm of the vector of singular values of A. The Frobenius norm will play a special role in several places, so we sometimes use the notation $\|A\|_F$ to emphasize this. Additionally, $\|A\|$ without the subscript indicates operator norm (which is equivalent to $\|A\|_\infty$).

We always use d to indicate the dimension of the matrix being tested, and use $\epsilon < 1$ to indicate the parameter in Definition 1.

When applied to vectors, $\langle \cdot, \cdot \rangle$ indicates the standard inner product on $\mathbb{R}^n$. When applied to matrices, it indicates the Frobenius inner product $\langle X, Y \rangle := \mathrm{Tr}(X^T Y)$.

S^{d-1} indicates the set of all unit vectors in $\mathbb{R}^d$.

We use the notation $X^\dagger$ to indicate the Moore-Penrose pseudoinverse of X.

For a symmetric matrix $A \in \mathbb{R}^{d \times d}$ with eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_d$, we let A_k denote the matrix A with all but the top k eigenvalues zeroed out. Formally, if U is an orthogonal matrix diagonalizing A, then $A_k = U^T \operatorname{diag}(\lambda_1, \ldots, \lambda_k, 0, \ldots, 0)U$, where U is such that $\lambda_i \geq \lambda_j$ for $i < j$. We also let $A_{-k} = A - A_k$.

Throughout, we use c to indicate an absolute constant. The value of c may change between instances.

II. Vector-matrix-vector queries

A. An optimal one-sided tester.

To construct our vector-matrix-vector tester, we analyze the iteration

$$x^{(k+1)} = x^k - \eta((g^{(k)})^T A x^{(k)}) g^{(k)}, \tag{2}$$

where $g^{(k)} \sim \mathcal{N}(0, I_d)$ and $x^{(0)} \sim \mathcal{N}(0, I_d)$.

Our algorithm is essentially to run this scheme for a fixed number of iterations with with well-chosen step size η. If the value of $(x^{(k)})^T A x(k)$ ever becomes negative, then we output False, otherwise we output True. Using this approach we prove the following.

Theorem 2. *There exists a one-sided adaptive ℓ_1-tester, that makes $O(\frac{1}{\epsilon}\log^3 \frac{1}{\epsilon})$ vector-matrix-vector queries to A.*

As an immediate corollary we obtain a bound for ℓ_p-testers.

Corollary 3. *There is a one-sided adaptive ℓ_p-tester that makes $O(\frac{1}{\epsilon} d^{1-1/p} \log^3(\frac{1}{\epsilon} d^{1-1/p}))$ vector-matrix-vector queries.*

Proof. This follows from the previous result along with the bound $\|A\|_p \geq d^{1/p-1} \|A\|_1$. □

We now turn to the proof of Theorem 2. Since our iterative scheme is rotation-invariant, we assume without loss of generality that $A = \operatorname{diag}(\lambda_1, \ldots, \lambda_d)$. For now, we assume that $\|A\|_1 \leq 1$, and that the smallest eigenvalue of A is $\lambda_1 = -\epsilon$. We consider running the algorithm for N iterations. We will show that our iteration finds an x with $x^T A x < 0$ in $N = \tilde{O}(1/\epsilon)$ iterations. We will use c to denote absolute constants that we don't track, and that may vary between uses.

Our approach is to show that the first coordinate (which is associated to the $-\epsilon$ eigenvalue) grows fairly quickly with good probability. Our key lemma bounds the second moments of every coordinate simultaneously.

Lemma 4. *Suppose η and N satisfy the following list of assumptions: (1) $\eta \leq \frac{1}{4}$, (2) $\eta^2 \epsilon N \leq \frac{1}{8}$, (3) $(1+\eta^2\epsilon^2)^N \leq \frac{5}{4}$, (4) $(1+\eta\epsilon)^N \geq \frac{10}{\epsilon^2}$. Then $x_1^{(N)} \geq \frac{1}{\epsilon^2}$ with probability at least* 0.2.

Proof. Following [15] we define the matrix $B_k = \prod_{i=1}^k \left(I - \eta g^{(i)} (g^{(i)})^T A\right)$, where the $g^{(i)}$ are independent $\mathcal{N}(0, I)$ gaussians. Note that $x^{(k)} = B_k x^{(0)}$. We will show that $B_k^T e_1$ has large norm with good probability (in fact we will show that $\langle B_k^T e_1, e_1 \rangle$ is large). This will then imply that $\langle B_k x^{(0)}, e_1 \rangle$ is large with high probability, where $x^{(0)} \sim \mathcal{N}(0, I)$.

Step 1: Deriving a recurrence for the second moments. Let $y^{(k)} = B_k^T e_1$ and let $u_i^{(k)}$ be the second moment of the coordinate $y_i^{(k)}$. Note that $u_i^{(0)} = \delta_{1i}$ (where δ is the Dirac delta). To simplify the notation, we drop the superscript on the g. We compute $y_i^{(k+1)} = \left((I - \eta A g g^T) y^{(k)}\right)_i = y_i^{(k)} - \eta (Ag)_i (g_1 y_1^{(k)} + \ldots + g_d y_d^{(k)}) = y_i^{(k)} - \eta \lambda_i g_i (g_1 y_1^{(k)} + \ldots + g_d y_d^{(k)})$.

Next we observe that (after grouping terms) the coefficients of the $y_i^{(k)}$ terms are pairwise uncorrelated. Using this, along with the fact that the g_i's are independent of the $y_i^{(k)}$'s gives

$$\begin{aligned} u_i^{(k+1)} &= \mathbb{E}(1-\eta\lambda_i g_i^2)^2 u_i^{(k)} + \eta^2\lambda_i^2 \sum_{j\neq i} u_j^{(k)} \\ &= (1-2\eta\lambda_i+3\eta^2\lambda_i^2)u_i^{(k)} + \eta^2\lambda_i^2\sum_{j\neq i} u_j^{(k)} \\ &= (1-2\eta\lambda_i+2\eta^2\lambda_i^2)u_i^{(k)} + \eta^2\lambda_i^2\sum_{j=1}^{d} u_j^{(k)}. \end{aligned}$$

Let $S^{(k)} = u_1^{(k)} + \ldots + u_d^{(k)}$, and $\gamma_i = 1-2\eta\lambda_i+2\eta^2\lambda_i^2$. Then we can write the recurrence as

$$u_i^{(k+1)} = \gamma_i u_i^{(k)} + \eta^2\lambda_i^2 S^{(k)}. \tag{3}$$

Iterating this recurrence gives

$$u_i^{(k)} = \delta_{1i}\gamma_i^k + \eta^2\lambda_i^2\left(\gamma_i^{k-1}S^{(0)} + \gamma_i^{k-2}S^{(1)} + \ldots + S^{(k-1)}\right). \tag{4}$$

Step 2: Bounding $S^{(k)}$.

Summing the above equation over i allows us to write a recurrence for the $S^{(k)}$'s: $S^{(k)} = \gamma_1^k + \alpha_{k-1}S^{(0)} + \alpha_{k-2}S^{(1)} + \ldots + \alpha_0 S^{(k-1)}$, where we define $\alpha_j := \sum_{i=1}^d \eta^2\lambda_i^2\gamma_i^j$.

We split α_j into two parts, α_j^+ and α_j^- corresponding to terms in the sum where λ_i is positive or negative respectively. We now use the recurrence to bound $S^{(k)}$. First by Holder's inequality, $S^{(k)} \leq \gamma_1^k + \max(S^{(0)},\ldots,S^{(k-1)})(\alpha_0^+ + \ldots + \alpha_{k-1}^+) + (\alpha_{k-1}^- S^{(0)} + \alpha_{k-2}^- S^{(1)} + \ldots + \alpha_0^- S^{(k-1)})$.

We calculate

$$\begin{aligned} \sum_{j=0}^{k-1}\alpha_j^+ &= \sum_{j=0}^{k-1}\sum_{i:\lambda_i>0}\eta^2\lambda_i^2\gamma_i^j \\ &= \sum_{i:\lambda_i>0}\eta^2\lambda_i^2\sum_{j=0}^{k-1}\gamma_i^j \\ &= \sum_{i:\lambda_i>0}\eta^2\lambda_i^2\frac{1-\gamma_i^k}{1-\gamma_i} \\ &= \sum_{i:\lambda_i>0}\eta^2\lambda_i^2\frac{1-\gamma_i^k}{2\eta\lambda_i-2\eta^2\lambda_i^2} \\ &= \sum_{i:\lambda_i>0}\eta\lambda_i\frac{1-\gamma_i^k}{2-2\eta\lambda_i} \\ &\leq \sum_{i:\lambda_i>0}\eta\lambda_i \leq \eta, \end{aligned}$$

where we used that $\eta\lambda_i \leq 1/2$, (which is a consequence of Assumption 1), that $\gamma_i < 1$ (which holds since $\lambda_i > 0$) and that $\sum_{i:\lambda_i>0}\lambda_i \leq 1$. Since we assume that $-\epsilon$ is the smallest eigenvalue,

$$\alpha_j^- \leq \eta^2\gamma_1^j\sum_{i:\lambda_i<0}\lambda_i^2 \leq \eta^2\gamma_1^j\epsilon\sum_{i:\lambda_i<0}|\lambda_i| \leq \eta^2\gamma_1^j\epsilon.$$

Let $\tilde{S}^{(k)} = \max(S^{(0)},\ldots S^{(k)})$. Then combining our bounds gives

$$\begin{aligned} \tilde{S}^{(k)} \leq \max(S^{(k-1)}, \gamma_1^k + \eta\tilde{S}^{(k-1)} + \eta^2\epsilon(\gamma_1^{k-1}\tilde{S}^{(0)} + \\ \gamma_1^{k-2}\tilde{S}^{(1)} + \ldots + \tilde{S}^{(k-1)})). \end{aligned}$$

The next step is to use this recurrence to bound $\tilde{S}^{(k)}$. For this, define $c^{(k)}$ such that $\tilde{S}^{(k)} = c^{(k)}\gamma_1^k$. Plugging in to the above and dividing through by γ_1^k, we get that $c^{(k)}$ satisfies

$$\begin{aligned} c^{(k)} &\leq \max\left(\frac{c^{(k-1)}}{\gamma_1}, 1 + \frac{\eta}{\gamma_1}c^{(k-1)} + \frac{\eta^2\epsilon}{\gamma_1}(c^{(0)} + \ldots + c^{(k-1)})\right) \\ &\leq \max\left(c^{(k-1)}, 1 + \eta c^{(k-1)} + \eta^2\epsilon(c^{(0)} + \ldots + c^{(k-1)})\right), \end{aligned}$$

where we used the fact that $\gamma_1 \geq 1$. Now set $\tilde{c}^{(k)} = \max(c^{(0)},\ldots c^{(k)})$. By assumptions 1 and 2, $\eta + \eta^2\epsilon k \leq 1/2$. This gives

$$\begin{aligned} \tilde{c}^{(k)} &\leq \max\left(\tilde{c}^{(k-1)}, 1 + \eta\tilde{c}^{(k-1)} + \eta^2\epsilon k\tilde{c}^{(k-1)}\right) \\ &\leq \max\left(\tilde{c}^{(k-1)}, 1 + \frac{1}{2}\tilde{c}^{(k-1)}\right). \end{aligned}$$

Note that $c^{(0)} = S^{(0)} = 1$, so a straightforward induction using the above recurrence shows that $\tilde{c}^{(k)} \leq 2$ for all k. It follows that $S^{(k)} \leq 2\gamma_1^k$.

Step 3: Bounding the second moment. Plugging the bound above in to (4) gives

$$u_1^{(k)} \leq \gamma_1^k + 2k\eta^2\epsilon^2\gamma_1^{k-1} \leq \left(1 + 2k\eta^2\epsilon^2\right)\gamma_1^k.$$

Step 4: Applying Chebyshev. We focus on the first coordinate, $y_1^{(k)}$. Note that $I - \eta Agg^T$ has expectation $I - \eta A$, so a straightforward induction shows that $\mathbb{E}y_1^{(k)} = (1+\eta\epsilon)^k$.

Using the bound for the second moment of the first coordinate, we get

$$\begin{aligned} \frac{u_1^{(k)}}{\left(\mathbb{E}y_1^{(k)}\right)^2} &\leq \frac{(1+2k\eta^2\epsilon^2)\gamma_1^k}{(1+\eta\epsilon)^{2k}} && (5) \\ &= (1+2k\eta^2\epsilon^2)\left(\frac{1+2\eta\epsilon+2\eta^2\epsilon^2}{1+2\eta\epsilon+\eta^2\epsilon^2}\right)^k && (6) \\ &= (1+2k\eta^2\epsilon^2)\left(1+\frac{\eta^2\epsilon^2}{1+2\eta\epsilon+\eta^2\epsilon^2}\right)^k && (7) \\ &\leq (1+2k\eta^2\epsilon^2)(1+\eta^2\epsilon^2)^k. && (8) \end{aligned}$$

By Assumptions 2 and 4, $N\eta^2\epsilon^2 \leq 1/8$ and $(1+\eta^2\epsilon^2)^N \leq 5/4$, so we get that $u_1^{(k)} \leq 25/16\left(\mathbb{E}u_1^{(k)}\right)^2$.

Thus by Chebyshev's inequality,

$$\mathbb{P}\left(\left|y_1^{(k)} - \mathbb{E}(y_1^{(k)})\right| \geq 0.9\mathbb{E}(y_1^{(k)})\right) \leq \frac{25}{36}. \tag{9}$$

So with probability at least 0.3, $y_1^{(N)} \geq \frac{1}{10}\mathbb{E}(y_1^{(N)}) = \frac{1}{10}(1+\eta\epsilon)^N$.

Under assumption 4, $(1+\eta\epsilon)^N \geq \frac{10}{\epsilon^2}$, which means that $y_1^{(N)} \geq \frac{1}{\epsilon^2}$ with at least 0.3 probability.

Step 5: Concluding the argument. We showed that $\left\langle B_N^T e_1, e_1 \right\rangle \geq \frac{1}{\epsilon^2}$ with probability at least 0.3. In particular this implies that $\left\| B_N^T e_1 \right\| \geq \frac{1}{\epsilon^2}$. Now since $x^{(0)}$ is distributed as $\mathcal{N}(0, I)$, $\left\langle B_N x^{(0)}, e_1 \right\rangle = \left\langle x^{(0)}, B_N^T e_1 \right\rangle \sim \mathcal{N}(0, \left\| B_N^T e_1 \right\|^2)$, which is at least $\left\| B_N^T e_1 \right\|$ in magnitude with 0.67 probability. It follows that $x_1^{(N)} \geq \frac{1}{\epsilon^2}$ with probability at least 0.2. □

Let $f(x) = x^T A x$. We next understand how the value of $f(x^{(k)})$ is updated on each iteration.

Proposition 5. *For $g \sim \mathcal{N}(0, 1)$, we have $f(x^{(k)}) - f(x^{(k+1)}) = \eta (g^T A x^{(k)})^2 (2 - \eta g^T A g)$.*

Proof. Plugging in the update rule and expanding gives

$$f(x^{(k+1)}) = (x^{(k)})^T A x^{(k)} - \eta (g^T A x^{(k)})^2 (2 - \eta g^T A g),$$

from which the proposition follows. □

A consequence of this update is that the sequence $f(x^{(k)})$ is almost guaranteed to be decreasing as long as η is chosen small enough.

Proposition 6. *Assume that $\mathrm{Tr}(A) \leq 1$ and that $\eta < c$. After N iterations, $f(x^{(N)}) \leq f(x^{(0)})$ with probability at least $99/100$ provided that $\eta \leq \frac{c}{\log N + 1}$.*

Proof. We show something stronger; namely that for the first N iterations, the sequence $f(x^{(k)})$ is decreasing. By Proposition 5, $f(x^{(k+1)}) \leq f(x^{(k)})$ as long as $g^T A g \leq \frac{2}{\eta}$. The probability that this does not occur is $\Pr\left(\sum \lambda_i g_i^2 \geq \frac{2}{\eta}\right) \leq \Pr\left(\sum \lambda_i (g_i^2 - 1) \geq \frac{2}{\eta} - 1\right)$.

The $g_i^2 - 1$ terms are independent subexponential random variables. So by Bernstein's inequality (see [18] Theorem 2.8.2 for the version used here), this probability is bounded by $2\exp(-c/\eta)$ as long as η is a sufficiently small constant. Taking a union bound gives that $f(x^{(N)}) \leq f(x^{(0)})$ with probability at least $1 - 2N\exp(-c/\eta)$, which is at least $99/100$ under the conditions given. □

Theorem 7. *Suppose that $\|A\|_1 \leq 1$, $\epsilon < 1/2$, and that A has $-\epsilon$ as an eigenvalue. If we take $\eta \leq \min\left(\frac{1}{32\log(10/\epsilon^2)}, \frac{c}{\log \frac{1}{\epsilon}}\right)$, then for some $N = \Theta\left(\frac{1}{\epsilon\eta}\log\frac{1}{\epsilon}\right)$ we have $f(x^{(N)}) < 0$ with constant probability.*

Proof. Given an η as in the statement of the theorem, choose $N = \left\lceil \frac{2}{\eta\epsilon}\log\frac{10}{\epsilon^2} \right\rceil$, which satisfies the assumptions of Lemma 4. Then $x_1^{(N)} \geq \frac{1}{\epsilon^2}$ with probability at least 0.2. By proposition 6, $f(x^{(N)}) \leq f(x^{(0)}) \leq 2$ with at least 0.99 probability, using the fact that $\eta \leq \frac{c}{\log \frac{1}{\epsilon}}$ for an appropriately chosen absolute constant c, such that the hypothesis of proposition 6 holds.

If $f(x^{(N)}) < 0$, then the algorithm has already terminated. Otherwise conditioned on the events in the above paragraph, we have with constant probability that $2 - \eta (g^{(N)})^T A g^{(N)} \geq \frac{1}{2}$ and $(g^{(N)})^T A x^{(N)} \geq \left\| A x^{(N)} \right\|^2 \geq \frac{1}{\epsilon^2}\lambda_1 \geq \frac{1}{\epsilon}$. Then by Proposition 5 it follows that $f(x^{(N+1)}) \leq f(x^{(N)}) - \frac{\eta}{2\epsilon^2} \leq 2 - \frac{\eta}{2\epsilon^2} < 0$. □

We also observe that we can reduce the dimension of the problem by using a result of Andoni and Nguyen. This allows us to avoid a $\log d$ dependence.

Proposition 8. *Suppose that A satisfies $\lambda\min(A) < -\alpha \|A\|_1$, and let $G \in \mathbb{R}^{d \times m}$ have independent $\mathcal{N}(0, \frac{1}{d})$. Then we can choose $m = O(1/\alpha)$ such that $\lambda_{\min}(G^T A G) < -\alpha/2$ and $\left\| G^T A G \right\|_1 \leq 2 \|A\|_1$.*

We are now ready to give the proof of Theorem 2.

Proof. The above result applies after scaling the η given in Theorem 7 by $1/\|A\|_1$. So it suffices to choose η to be bounded above by

$$\frac{1}{\|A\|_1} \min\left(\frac{1}{32\log(10/\epsilon^2)}, \frac{c}{\log\frac{1}{\epsilon}}\right),$$

and within a constant factor of this value.

To choose an η, pick a standard normal g, and compute Ag using $1/\epsilon$ vector-matrix-vector queries. Then with constant probability, $\lambda_{\max}(A) \leq \|Ag\| \leq 2d\lambda_{\max}$. Given this, we have

$$d\|Ag\| \geq \|A\|_1 \geq \frac{\|Ag\|}{2d}, \tag{10}$$

which allows us to approximate $\|A\|_1$ to within a factor of d^2 with constant probability. Given this, one may simply try the above algorithm with an η at each of $O(\log(d^2)) = O(\log d)$ different scales, with the cost of an extra $\log d$ factor.

Finally, we may improve the $\log d$ factor to a $\log(1/\epsilon)$ factor by using Proposition 8 to sketch A, and then applying the above analysis to $G^T A G$. Note that the sketch may be used implicitly; once G is chosen, a vector-matrix-vector query to $G^T A G$ can be simulated with a single vector-matrix-vector query to A. □

B. Lower bounds

We later show a lower bound for two-sided testers which implies that the bound for ℓ_1-testers given in Theorem 2 is tight up to $\log$ factors. If we require the tester to have one-sided error, then we additionally show that the bound in Corollary 3 is tight for all p. Note that this distinction between one-sided and two-sided testers is necessary given Theorem 17.

In order to obtain these lower bounds for adaptive testers, we first show corresponding lower bounds for non-adaptive testers. A minor modification to Lemma 3.1 in [4] shows that an adaptive tester can have at most quadratic improvement over a non-adaptive tester. This allows us to obtain our adaptive lower bounds as a consequence of the non-adaptive bounds.

For non-adaptive testers with one-sided error, we have the following hard instance.

Theorem 9. *Let $\lambda > 0$ and suppose for all matrices A with spectrum $(-\lambda, 1, \ldots, 1)$ that a non-adaptive one-sided tester $\mathcal{T}$ outputs False with $2/3$ probability. Then $\mathcal{T}$ must make at least $\frac{1}{9}\left(\frac{d}{1+\lambda}\right)^2$ vector-matrix-vector queries.*

In particular, this result implies that for non-adaptive one-sided testers, a poly$(1/\epsilon)$ ℓ_p-tester can only exist for $p = 1$.

Theorem 10. *A one-sided non-adaptive ℓ_p-tester must make at least $\Omega(\frac{1}{\epsilon^2}d^{2-2/p})$ vector-matrix-vector queries.*

Proof. This follows as a corollary of Theorem 9; simply apply that result to the spectrum $(\epsilon(d-1)^{1/p}, 1\ldots, 1)$ where there are $d-1$ 1's. □

Our Theorem 10 along with a minor modification of Lemma 3.1 in [4] yields a lower bound for adaptive testers.

Theorem 11. *An adaptive one-sided ℓ_p-tester must make at least $\Omega(\frac{1}{\epsilon}d^{1-1/p})$ vector-matrix-vector queries.*

III. Adaptive matrix-vector queries

We analyze random Krylov iteration. Namely we begin with a random $g \sim \mathcal{N}(0, I_d)$ and construct the sequence of iterates $g, Ag, A^2g, \ldots A^kg$ using k adaptive matrix-vector queries. The span of these vectors is denoted $\mathcal{K}_k(g)$ and referred to as the k^{th} Krylov subspace.

Krylov iteration suggests a very simple algorithm. First compute $g, Ag, \ldots, A^{k+1}g$. If $\mathcal{K}_k(g)$ contains a vector v such that $v^TAv < 0$ then output False, otherwise output True. (Note that one can compute Av and hence v^TAv for all such v, given the $k+1$ matrix-vector queries.) We show that this simple algorithm is in fact optimal.

As a point of implementation, we note that the above condition on $\mathcal{K}_k(g)$ can be checked algorithmically. One first uses Gram-Schmidt to compute the projection Π onto $\mathcal{K}_k(g)$. The existence of a $v \in \mathcal{K}_k(g)$ with $v^TAb < 0$ is equivalent to the condition $\lambda_{\min}(\Pi A\Pi) < 0$. When A is ϵ-far from PSD, the proof below will show that in fact $\lambda_{\min}(\Pi A\Pi) < -\Omega(\epsilon)\left\|A\right\|_p$, so it suffices to estimate $\lambda_{\min}(\Pi A\Pi)$ to within $O(\epsilon)\left\|A\right\|_p$ accuracy.

Proposition 12. *For $r > 0$, $\alpha > 0$ and $\delta > 0$ there exists a polynomial p of degree $O(\frac{\sqrt{r}}{\sqrt{\alpha}}\log\frac{1}{\delta})$, such that $p(-\alpha) = 1$ and $|p(x)| \le \delta$ for all $x \in [0, r]$.*

Proof. Recall that the degree d Chebyshev polynomial T_d is bounded by 1 in absolute value on $[-1, 1]$ and satisfies

$$T_d(1+\gamma \ge 2^{d\sqrt{\gamma}-1}).$$

(See [23] for example.) The proposition follows by shifting and scaling T_d. □

Theorem 13. *Suppose that A has an eigenvalue $\lambda_{\min}$ with $\lambda_{\min} \le -\epsilon\left\|A\right\|_p$. When $p = 1$, the Krylov subspace $\mathcal{K}_k(g)$ contains a vector v with $v^TAv < 0$ for $k = O\left(\left(\frac{1}{\epsilon}\right)^{\frac{1}{3}}\log\frac{1}{\epsilon}\right)$. When $p \in (1, \infty]$, the same conclusion holds for $k = O\left(\left(\frac{1}{\epsilon}\right)^{\frac{p}{2p+1}}\log\frac{1}{\epsilon}\log d\right)$.*

Proof. Without loss of generality, assume that $\left\|A\right\|_p \le 1$. Fix a value T to be determined later, effectively corresponding to the number of top eigenvalues that we deflate. By Proposition 12 we can construct a polynomial q, such that $q(\lambda_{\min}) = 1$ and $|q(x)| \le \sqrt{\frac{\epsilon/10}{d^{1-1/p}}}$ for $x \in [0, T^{-1/p}]$ with

$$\deg(q) \le C\frac{T^{-1/(2p)}}{\sqrt{\epsilon}}\log\left(\sqrt{\frac{d^{1-1/p}}{\epsilon/10}}\right), \tag{11}$$

where C is an absolute constant.

Now set

$$p(x) = q(x)\prod_{i:\lambda_i > T^{-1/p}}\frac{\lambda_i - x}{\lambda_i - \lambda_{\min}}. \tag{12}$$

Since we assume $\left\|A\right\|_p \le 1$, there at most T terms in the product, so

$$\deg(p) \le T + C\frac{T^{-1/(2p)}}{\sqrt{\epsilon}}\log\left(\sqrt{\frac{d^{1-1/p}}{\epsilon/10}}\right). \tag{13}$$

By setting $T = \epsilon^{-p/(2p+1)}$, we get

$$\deg(p) = \begin{cases} O\left(\left(\frac{1}{\epsilon}\right)^{\frac{p}{2p+1}}\log\frac{1}{\epsilon}\right) & \text{if } p = 1 \\ O\left(\left(\frac{1}{\epsilon}\right)^{\frac{p}{2p+1}}\log\frac{1}{\epsilon}\log d\right) & \text{if } p > 1 \end{cases} \tag{14}$$

As long as k is at least $\deg(p)$, then $v = p(A)g$ lies in $\mathcal{K}_k(g)$, and

$$v^TAv = g^Tp(A)^2Ag. \tag{15}$$

By construction, $p(\lambda_{\min}) = 1$. Also for all x in $[0, T^{-1/p}]$, $|p(x)| \le |q(x)| \le \sqrt{\epsilon/10}d^{(1/p)-1}$.

Therefore the matrix $p(A)^2A$ has at least one eigenvalue less than $-\epsilon$, and the positive eigenvalues sum to at most

$$\sum_{i:\lambda_i>0}\frac{\epsilon}{10}d^{1/p-1}\lambda_i \le \frac{\epsilon}{10}, \tag{16}$$

by using Holder's inequality along with the fact that $\left\|A\right\|_p \le 1$. So with at least $2/3$ probability, $g^Tp(A)^2Ag < 0$ as desired. □

Remark 14. *While we observe that deflation of the top eigenvalues can be carried out implicitly within the Krylov space, this can also be done explicitly using block Krylov iteration, along with the guarantee given in Theorem 1 of [23].*

We showed above that we could improve upon the usual analysis of Krylov iteration in our context. We also establish a matching lower bound by utilizing the proof of Theorem 3.1 presented in [12].

Theorem 15. *A two-sided, adaptive ℓ_p-tester in the matrix-vector model must in general make at least $\Omega(\frac{1}{\epsilon^{p/(2p+1)}})$ queries.*

IV. An optimal bilinear sketch

We present an optimal bilinear sketch for PSD-testing which will also yield an optimal ℓ_2-tester in the vector-matrix-vector model.

Our sketch is very simple. We choose $G \in \mathbb{R}^{d\times k}$ to have independent $\mathcal{N}(0, 1)$ entries and take our sketch to be G^TAG. In parallel we construct estimates α and β for the trace and

Frobenius norm of A respectively, such that β is accurate to within a multiplicative error of 2, and α is accurate to with $\|A\|_F$ additive error. (Note that this may be done at the cost of increasing the sketching dimension by $O(1)$.)

If $G^T AG$ is not PSD then we automatically reject. Otherwise, we then consider the quantity

$$\gamma := \frac{\alpha - \lambda_{\min}(G^T AG)}{\beta\sqrt{k}\log k} \tag{17}$$

If γ is at most c_{psd} for some absolute constant c_{psd}, then the tester outputs False, otherwise it outputs True.

By applying concentration inequalities to establish a lower bound on $\lambda_{\min}(G^T AG)$ when A is PSD, and an upper bound when A is far from PSD, we achieve the following sketching guarantee.

Theorem 16. *There is a bilinear sketch $G^T AG$ with sketching dimension $k = O(\frac{1}{\epsilon^2}\log^2\frac{1}{\epsilon})$ that yields a two-sided ℓ_2-tester that is correct with at least 0.9 probability.*

Note that this result immediately gives a non-adaptive vector-matrix-vector tester which makes $\widetilde{O}(1/\epsilon^4)$ queries.

By shifting the bilinear sketch above by an appropriate multiple of the identity, we place ourselves in the situation of Theorem 2 and hence are also able achieve tight bounds for adaptive testers with two-sided error.

Theorem 17. *There is a two-sided adaptive ℓ_2-tester in the vector-matrix-vector model, which makes $\widetilde{O}(1/\epsilon^2)$ queries.*

As a consequence we also obtain a two-sided p-tester for all $p \geq 2$.

Corollary 18. *For $p \geq 2$, there is a two-sided adaptive ℓ_p-tester in the vector-matrix-vector model, which make $\widetilde{O}(1/\epsilon^2)d^{1-1/p}$ queries.*

Proof. Apply Theorem 17 along with the bound $\|A\|_p \geq d^{\frac{1}{p}-\frac{1}{2}}\|A\|_F$. □

A. Lower bounds for two-sided testers

Our lower bounds for two-sided testers comes from the spiked Gaussian model introduced in [19]. As before, our adaptive lower bounds will come as a consequence of the corresponding non-adaptive bounds.

Theorem 19. *A two-sided ℓ_p-tester that makes non-adaptive vector-matrix-vector queries requires at least*

- $\Omega(\frac{1}{\epsilon^{2p}})$ *queries for* $1 \leq p \leq 2$
- $\Omega(\frac{1}{\epsilon^4}d^{2-4/p})$ *queries for* $2 < p < \infty$ *as long as* d *can be taken to be* $\Omega(1/\epsilon^p)$.
- $\Omega(d^2)$ *queries for* $p = \infty$.

V. Spectrum Estimation

We make use of the following result, which is Lemma 11 of [20] specialized to our setting.

Lemma 20. *For a symmetric matrix $A \in \mathbb{R}^{d\times d}$, there is a distribution over an oblivious sketching matrix $R \in \mathbb{R}^{d\times m}$ with $m = O(\frac{k}{\epsilon})$ so that with at least 0.9 proability,*

$$\min_{Y^* \in \text{ rank } k,\text{PSD}} \left\|(AR)Y^*(AR)^T - A\right\|_F^2 \leq (1+\epsilon)\left\|A_{k,+} - A\right\|_F^2, \tag{18}$$

where $A_{k,+}$ is the optimal rank-one PSD approximation to A in Frobenius norm.

Remark 21. *In our setting one can simply take R to be Gaussian since the guarantee above must hold when A is drawn from a rotationally invariant distribution. In many situations, structured or sparse matrices are useful, but we do not need this here.*

We also recall the notion of an affine embedding [28].

Definition 22. *S is an affine embedding for matrices A and B if for all matrices X of the appropriate dimensions, we have*

$$\|S(AX - B)\|_F^2 = (1 \pm \epsilon)\|AX - B\|_F^2. \tag{19}$$

We also recall that when A is promised to have rank at most r, there is a distribution over S with $O(\epsilon^{-2}r)$ rows such that (19) holds with constant probability for any choice of A and B [28].

Lemma 23. *There is an algorithm which makes $O(\frac{k^2}{\epsilon^6}\log\frac{1}{\delta})$ vector-matrix-vector queries to A and with at least $1-\delta$ probability outputs an approximation of $\|A\|_{k,+}$, accurate to within $\epsilon\|A\|_F^2$ additive error.*

Proof. We run two subroutines in parallel.

Subroutine 1. Approximate $\|A_{k,+} - A\|_F^2$ up to $O(\epsilon)$ multiplicative error.

Our algorithm first draws affine embedding matrices S_1 and S_2 for $r = k/\epsilon$, and with ϵ distortion, each with $O(\frac{k}{\epsilon^3})$ rows. We also draw a matrix R as in Lemma 20 with $m = O(\frac{k}{\epsilon})$ columns.

We then compute S_1AR and S_2AR, each requiring $\frac{k^2}{\epsilon^4}$ vector-matrix-vector queries, and compute $S_1AS_2^T$ requiring $\frac{k^2}{\epsilon^6}$ queries.

Let Y_k be arbitrary with the appropriate dimensions (later we will optimize Y_k over rank k PSD matrices). By using the affine embedding property along with the fact that R has rank at most $\frac{k}{\epsilon}$, we have

$$\begin{aligned}
&\left\|(S_1AR)Y_k(S_2AR)^T + S_1AS_2^T\right\|_F^2 \\
&\quad= (1\pm\epsilon)\left\|ARY_k(S_2AR)^T + AS_2^T\right\|_F^2 \\
&\quad= (1\pm\epsilon)\left\|S_2ARY_kR^TA + S_2A\right\|_F^2 \\
&\quad= (1\pm 3\epsilon)\left\|ARY_kR^TA + A\right\|_F^2.
\end{aligned}$$

As a consequence of this, and the property held by R, we have

$$\min_{\mathrm{rk}(Y_k)\le k, Y_k \text{ PSD}} \left\|(S_1AR)Y_k(S_2AR)^T + S_1AS_2^T\right\|_F^2 \quad (20)$$

$$= (1 \pm 3\epsilon) \min_{Y_k} \left\|ARY_kR^TA + A\right\|_F^2 \quad (21)$$

$$= (1 \pm 7\epsilon) \left\|A_{k,+} - A\right\|_F^2. \quad (22)$$

Thus by computing the quantity in the left-hand-side above, our algorithm computes an $O(\epsilon)$ multiplicative approximation using $O(k^2/\epsilon^6)$ vector-matrix-vector queries.

Subroutine 2. Approximate $\|A\|_F^2$ up to $O(\epsilon)$ multiplicative error.

We simply apply Theorem 2.2. of [29], set $q = 2$, and note that the entries of the sketch correspond to vector-matrix-vector products. By their bound we require $O(\epsilon^{-2}\log(1/\epsilon))$ vector-matrix-vector queries.

Since $\|A_{k,+}\|_F^2 = \|A\|_F^2 - \|A_{k,+} - A\|_F^2$, we obtain an additive $O(\epsilon)\|A\|_F^2$ approximation to $\|A_{k,+}\|_F^2$ by running the two subroutines above and subtracting their results.

Finally, by repeating the above procedure $O(\log \frac{1}{\delta})$ times in parallel and taking the median of the trails, we obtain a failure probability of at most δ.

□

We note that we immediately obtain a $\mathrm{poly}(1/\epsilon)$ query ℓ_2-tester by applying Lemma 23 to approximate $A_{1,-}$. However this yields a worse ϵ dependence than Theorem 16. Perhaps more interestingly, these techniques also give a way to approximate the top k (in magnitude) eigenvalues of A while preserving their signs. We note a minor caveat. If λ_k and λ_{k+1} are very close in magnitude, but have opposite signs, then we cannot guarantee that we approximate λ_k. Therefore in the statement below, we only promise to approximate eigenvalues with magnitude at least $|\lambda_k| + 2\epsilon$.

Theorem 24. *Let $\lambda_1, \lambda_2, \ldots$ be the (signed) eigenvalues of A sorted in decreasing order of magnitude.*

There is an algorithm that makes $O(\frac{k^2}{\epsilon^{12}}\log k)$ non-adaptive vector-matrix-vector queries to A, and with probability at least 0.9, outputs $\tilde{\lambda}_1, \ldots, \tilde{\lambda}_k$ such that

(i) There exists a permutation σ on $[k]$ so that for all i with $|\lambda_i| \ge |\lambda_k| + 2\epsilon$, $|\tilde{\lambda}_{\sigma(i)} - \lambda_i| \le \epsilon\|A\|_F$

(ii) For all i, there exists j with $|\lambda_j| \ge |\lambda_k| - \epsilon$ and $|\tilde{\lambda}_i - \lambda_j| \le \epsilon\|A\|_F$

With one additional round of adaptivity the number of measurements can be reduced to $O(\frac{k^2}{\epsilon^8}\log k)$.

ACKNOWLEDGMENT

The authors would like to thank the anonymous referees for their helpful suggestions in preparing this manuscript.

REFERENCES

[1] R. Krauthgamer and O. Sasson, "Property testing of data dimensionality," in *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, January 12-14, 2003, Baltimore, Maryland, USA*. ACM/SIAM, 2003, pp. 18–27.

[2] I. Han, D. Malioutov, H. Avron, and J. Shin, "Approximating spectral sums of large-scale matrices using stochastic chebyshev approximations," *SIAM Journal on Scientific Computing*, vol. 39, no. 4, pp. A1558–A1585, 2017.

[3] A. Bakshi, N. Chepurko, and R. Jayaram, "Testing positive semi-definiteness via random submatrices," *arXiv preprint arXiv:2005.06441*, 2020.

[4] X. Sun, D. P. Woodruff, G. Yang, and J. Zhang, "Querying a matrix through matrix-vector products," *arXiv preprint arXiv:1906.05736*, 2019.

[5] C. Rashtchian, D. P. Woodruff, and H. Zhu, "Vector-matrix-vector queries for solving linear algebra, statistics, and graph problems," *arXiv preprint arXiv:2006.14015*, 2020.

[6] M.-F. Balcan, Y. Li, D. P. Woodruff, and H. Zhang, "Testing matrix rank, optimally," in *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2019, pp. 727–746.

[7] R. Bhattacharjee, C. Musco, and A. Ray, "Sublinear time eigenvalue approximation via random sampling," *CoRR*, vol. abs/2109.07647, 2021.

[8] Z. Bai, G. Fahey, and G. Golub, "Some large-scale matrix computation problems," *Journal of Computational and Applied Mathematics*, vol. 74, no. 1-2, pp. 71–89, 1996.

[9] Y. Li, H. L. Nguyen, and D. P. Woodruff, "Turnstile streaming algorithms might as well be linear sketches," in *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, D. B. Shmoys, Ed. ACM, 2014, pp. 174–183.

[10] Y. Ai, W. Hu, Y. Li, and D. P. Woodruff, "New characterizations in turnstile streams with applications," in *31st Conference on Computational Complexity (CCC 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

[11] A. Andoni and H. L. Nguyn, "Eigenvalues of a matrix in the streaming model," in *Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms*. SIAM, 2013, pp. 1729–1737.

[12] M. Braverman, E. Hazan, M. Simchowitz, and B. Woodworth, "The gradient complexity of linear regression," in *Conference on Learning Theory*. PMLR, 2020, pp. 627–647.

[13] M. F. Hutchinson, "A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines," *Communications in Statistics-Simulation and Computation*, vol. 18, no. 3, pp. 1059–1076, 1989.

[14] E. Oja, "Simplified neuron model as a principal component analyzer," *Journal of mathematical biology*, vol. 15, no. 3, pp. 267–273, 1982.

[15] P. Jain, C. Jin, S. M. Kakade, P. Netrapalli, and A. Sidford, "Streaming pca: Matching matrix bernstein and near-optimal finite sample guarantees for oja's algorithm," in *Conference on learning theory*. PMLR, 2016, pp. 1147–1164.

[16] P. Indyk, S. Narayanan, and D. P. Woodruff, "Frequency estimation with one-sided error," in *SODA*, 2022.

[17] A. E. Litvak, A. Pajor, M. Rudelson, and N. Tomczak-Jaegermann, "Smallest singular value of random matrices and geometry of random polytopes," *Advances in Mathematics*, vol. 195, no. 2, pp. 491–523, 2005.

[18] R. Vershynin, *High-dimensional probability: An introduction with applications in data science*. Cambridge university press, 2018, vol. 47.

[19] Y. Li and D. P. Woodruff, "Tight bounds for sketching the operator norm, schatten norms, and subspace embeddings," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

[20] K. L. Clarkson and D. P. Woodruff, "Low-rank psd approximation in input-sparsity time," in *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2017, pp. 2061–2072.

[21] R. A. Meyer, C. Musco, C. Musco, and D. P. Woodruff, "Hutch++: optimal stochastic trace estimation," in *Symposium on Simplicity in Algorithms (SOSA)*. SIAM, 2021, pp. 142–155.

[22] M. Simchowitz, A. El Alaoui, and B. Recht, "Tight query complexity lower bounds for pca via finite sample deformed wigner law," in *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, 2018, pp. 1249–1259.

[23] C. Musco and C. Musco, "Randomized block krylov methods for stronger and faster approximate singular value decomposition," *arXiv preprint arXiv:1504.05477*, 2015.

[24] K. Wimmer, Y. Wu, and P. Zhang, "Optimal query complexity for estimating the trace of a matrix," in *International Colloquium on Automata, Languages, and Programming*. Springer, 2014, pp. 1051–1062.

[25] O. Shamir, "Convergence of stochastic gradient descent for pca," in *International Conference on Machine Learning*. PMLR, 2016, pp. 257–265.
[26] Z. Allen-Zhu and Y. Li, "First efficient convergence for streaming k-pca: a global, gap-free, and near-optimal rate," in *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2017, pp. 487–492.
[27] R. Bhattacharjee, C. Musco, and A. Ray, "Sublinear time eigenvalue approximation via random sampling," *arXiv preprint arXiv:2109.07647*, 2021.
[28] K. L. Clarkson and D. P. Woodruff, "Low-rank approximation and regression in input sparsity time," *Journal of the ACM (JACM)*, vol. 63, no. 6, pp. 1–45, 2017.
[29] M. Meister, T. Sarlos, and D. Woodruff, "Tight dimensionality reduction for sketching low degree polynomial kernels," 2019.