Faster Principal Component Regression and Stable Matrix Chebyshev Approximation

Zeyuan Allen-Zhu [*1] **Yuanzhi Li** [*2]

Abstract

We solve principal component regression (PCR), up to a multiplicative accuracy $1+\gamma$, by reducing the problem to $\widetilde{O}(\gamma^{-1})$ black-box calls of ridge regression. Therefore, our algorithm does not require any explicit construction of the top principal components, and is suitable for large-scale PCR instances. In contrast, previous result requires $\widetilde{O}(\gamma^{-2})$ such black-box calls. We obtain this result by developing a general stable recurrence formula for matrix Chebyshev polynomials, and a degree-optimal polynomial approximation to the matrix sign function. Our techniques may be of independent interests, especially when designing iterative methods.

1 Introduction

In machine learning and statistics, it is often desirable to represent a large-scale dataset in a more tractable, lower-dimensional form, without losing too much information. One of the most robust ways to achieve this goal is through *principal component projection (PCP)*:

PCP: project vectors onto the span of the top principal components of the a matrix.

It is well-known that PCP decreases noise and increases efficiency in downstream tasks. One of the main applications is *principal component regression (PCR)*:

PCR: linear regression but restricted to the subspace of top principal components.

Classical algorithms for PCP or PCR rely on a principal component analysis (PCA) solver to recover the top principal components first; with these components available, the tasks of PCP and PCR become trivial because the projection matrix can be constructed explicitly.

Unfortunately, PCA solvers demand a running time that at least linearly scales with the number of top principal components chosen for the projection. For instance, to project a vector onto the top 1000 principal components of a high-dimensional dataset, even the most efficient Krylov-based (Musco & Musco, 2015) or Lanczos-based (Allen-Zhu & Li, 2016a) methods require a running time that is proportional to $1000 \times 40 = 4 \times 10^4$ times the input matrix sparsity, if the Krylov or Lanczos method is executed for 40 iterations. This is usually computationally intractable.

1.1 Approximating PCP Without PCA

In this paper, we propose the following notion of PCP approximation. Given a data matrix $\mathbf{A} \in \mathbb{R}^{d' \times d}$ (with singular values no greater than 1) and a threshold $\lambda > 0$, we say that an algorithm solves (γ, ε)-approximate PCP if — informally speaking and up to a multiplicative $1 \pm \varepsilon$ error— it projects (see Def. 3.1 for a formal definition)

1. eigenvector ν of $\mathbf{A}^\top \mathbf{A}$ with value in $\big[\lambda(1+\gamma), 1\big]$ to ν,
2. eigenvector ν of $\mathbf{A}^\top \mathbf{A}$ with value in $\big[0, \lambda(1-\gamma)\big]$ to $\vec{0}$,
3. eigenvector ν of $\mathbf{A}^\top \mathbf{A}$ with value in $\big[\lambda(1-\gamma), \lambda(1+\gamma)\big]$ to "anywhere between $\vec{0}$ and ν."

Such a definition also extends to (γ, ε)-approximate PCR (see Def. 3.2).

It was first noticed by Frostig *et al.* (Frostig et al., 2016) that approximate PCP and PCR be solved with a running time independent of the number of principal components above threshold λ. More specifically, they reduced (γ, ε)-approximate PCP and PCR to

$$\boxed{O\big(\gamma^{-2}\log(1/\varepsilon)\big) \text{ black-box calls of any ridge regression subroutine}}$$

where each call computes $(\mathbf{A}^\top \mathbf{A} + \lambda \mathbf{I})^{-1} u$ for some vector u.[1] Our main focus of this paper is to *quadratically* improve this performance and reduce PCP and PCR to

[*]Equal contribution . Future version of this paper shall be found at `https://arxiv.org/abs/1608.04773`. [1]Microsoft Reseaerch [2]Princeton University. Correspondence to: Zeyuan Allen-Zhu <zeyuan@csail.mit.edu>, Yuanzhi Li <yuanzhil@cs.princeton.edu>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017.

[1]Ridge regression is often considered as an easy-to-solve machine learning problem: using for instance SVRG (Johnson & Zhang, 2013), one can usually solve ridge regression to an 10^{-8} accuracy with at most 40 passes of the data.

$O\big(\gamma^{-1}\log(1/\gamma\varepsilon)\big)$ black-box calls of any ridge regression subroutine

where each call again computes $(\mathbf{A}^\top\mathbf{A}+\lambda\mathbf{I})^{-1}u$.

Remark 1.1. Frostig *et al.* only showed their algorithm satisfies the properties 1 and 2 of (γ,ε)-approximation (but not the property 3), and thus their proof was only for matrix $\mathbf{A}$ with no singular value in the range $[\sqrt{\lambda(1-\gamma)},\sqrt{\lambda(1+\gamma)}]$. This is known as the *eigengap assumption*, which is rarely satisfied in practice (Musco & Musco, 2015). In this paper, we prove our result both with and without such eigengap assumption. Since our techniques also imply the algorithm of Frostig *et al.* satisfies property 3, throughout the paper, we say Frostig *et al.* solve (γ,ε)-approximate PCP and PCR.

1.2 From PCP to Polynomial Approximation

The main technique of Frostig *et al.* is to construct a *polynomial* to approximate the sign function $\mathrm{sgn}(x)\colon[-1,1]\to\{\pm1\}$:

$$\mathrm{sgn}(x):=\begin{cases}+1, & x\geq 0;\\ -1, & x<0.\end{cases}$$

In particular, given any polynomial $g(x)$ satisfying

$$\big|g(x)-\mathrm{sgn}(x)\big|\leq\varepsilon\quad\forall x\in[-1,-\gamma]\cup[\gamma,1]\ , \tag{1.1}$$

$$\big|g(x)\big|\leq 1\quad\forall x\in[-\gamma,\gamma]\ , \tag{1.2}$$

the problem of (γ,ε)-approximate PCP can be reduced to computing the matrix polynomial $g(\mathbf{S})$ for $\mathbf{S}:=(\mathbf{A}^\top\mathbf{A}+\lambda\mathbf{I})^{-1}(\mathbf{A}^\top\mathbf{A}-\lambda\mathbf{I})$ (cf. Fact 7.1). In other words,

- to project any vector $\chi\in\mathbb{R}^d$ to top principal components, we can compute $g(\mathbf{S})\chi$ instead; and
- to compute $g(\mathbf{S})\chi$, we can reduce it to ridge regression for each evaluation of $\mathbf{S}u$ for some vector u.

Remark 1.2. Since the transformation from $\mathbf{A}^\top\mathbf{A}$ to $\mathbf{S}$ is not linear, the final approximation to the PCP is a rational function (as opposed to a polynomial) over $\mathbf{A}^\top\mathbf{A}$. We restrict to polynomial choices of $g(\cdot)$ because in this way, the final rational function has all the denominators being $\mathbf{A}^\top\mathbf{A}+\lambda\mathbf{I}$, thus reduces to ridge regressions.

Remark 1.3. The transformation from $\mathbf{A}^\top\mathbf{A}$ to $\mathbf{S}$ ensures that all the eigenvalues of $\mathbf{A}^\top\mathbf{A}$ in the range $(1\pm\gamma)\lambda$ roughly map to the eigenvalues of $\mathbf{S}$ in the range $[-\gamma,\gamma]$.

Main Challenges. There are two main challenges regarding the design of polynomial $g(x)$.

- EFFICIENCY. We wish to minimize the degree $n=\deg(g(x))$ because the computation of $g(\mathbf{S})\chi$ usually requires n calls of ridge regression.
- STABILITY. We wish $g(x)$ to be stable; that is, $g(\mathbf{S})\chi$ must be given by a recursive formula where if we make ε' error in each recursion (due to error incurred from ridge regression), the final error of $g(\mathbf{S})\chi$ must be at most $\varepsilon'\times\mathsf{poly}(d)$.

Remark 1.4. Efficient routines such as SVRG (Johnson & Zhang, 2013) solve ridge regression and thus compute $\mathbf{S}u$ for any $u\in\mathbb{R}^d$, with running times only logarithmically in $1/\varepsilon'$. Therefore, by setting $\varepsilon'=\varepsilon/\mathsf{poly}(d)$, one can blow up the running time by a small factor $O(\log(d))$ in order to obtain an ε-accurate solution for $g(\mathbf{S})\chi$.

The polynomial $g(x)$ constructed by Frostig *et al.* comes from truncated Taylor expansion. It has degree $O\big(\gamma^{-2}\log(1/\varepsilon)\big)$ and is stable. This γ^{-2} dependency limits the practical performance of their proposed PCP and PCR algorithms, especially in a high accuracy regime. At the same time,

- the optimal degree for a polynomial to satisfy even only (1.1) is $\Theta\big(\gamma^{-1}\log(1/\varepsilon)\big)$ (Eremenko & Yuditskii, 2007; 2011).

Frostig *et al.* were unable to find a stable polynomial matching this optimal degree and left it as open question.[2]

1.3 Our Results and Main Ideas

We provide an efficient and stable polynomial approximation to the matrix sign function that has a near-optimal degree $O(\gamma^{-1}\log(1/\gamma\varepsilon))$. At a high level, we construct a polynomial $q(x)$ that approximately equals $\big(\frac{1+\kappa-x}{2}\big)^{-1/2}$ for some $\kappa=\Theta(\gamma^2)$; then we set $g(x):=x\cdot q(1+\kappa-2x^2)$ which approximates $\mathrm{sgn}(x)$.

To construct $q(x)$, we first note that $\big(\frac{1+\kappa-x}{2}\big)^{-1/2}$ has no singular point on $[-1,1]$ so we can apply Chebyshev approximation theory to obtain some $q(x)$ of degree $O(\gamma^{-1}\log(1/\gamma\varepsilon))$ satisfying

$$\Big|q(x)-\Big(\frac{1+\kappa-x}{2}\Big)^{-1/2}\Big|\leq\varepsilon\text{ for every }x\in[-1,1]\quad.$$

This can be shown to imply $\big|g(x)-\mathrm{sgn}(x)\big|\leq\varepsilon$ for every $x\in[-1,-\gamma]\cup[\gamma,1]$, so (1.1) is satisfied.

In order to prove (1.2) , we prove a separate lemma:[3]

$$q(x)\leq\Big(\frac{1+\kappa-x}{2}\Big)^{-1/2}\text{ for every }x\in[1,1+\kappa]\quad.$$

Note that this does not follow from standard Chebyshev theory because Chebyshev approximation guarantees are only with respect to $x\in[-1,1]$.

This proves the "EFFICIENCY" part of the main challenges discussed earlier. As for the "STABILITY" part, we prove a general theorem regarding any weighted sum of Chebyshev polynomials applied to matrices. We provide a backward recurrence algorithm and show that it is stable under noisy

[2] Using degree reduction, Frostig *et al.* found an explicit polynomial $g(x)$ of degree $O\big(\gamma^{-1}\log(1/\gamma\varepsilon)\big)$ satisfying (1.1). However, that polynomial is unstable because it is constructed monomial by monomial and has exponentially large coefficients in front of each monomial. Furthermore, it is not clear if their polynomial satisfies the (1.2).

[3] We proved a general lemma which holds for any function whose all orders of derivatives are non-negative at $x=0$.

computations. This may be of independent interest.

For interested readers, we compare our polynomial $q(x)$ with that of Frostig *et al.* in Figure 1.

1.4 Related Work

There are a few attempts to reduce the cost of PCA when solving PCR, by for instance approximating the matrix $\mathbf{A}\mathbf{P}_\lambda$ where $\mathbf{P}_\lambda$ is the PCP projection matrix (Chan & Hansen, 1990; Boutsidis & Magdon-Ismail, 2014). However, they cost a running time that linearly scales with the number of principal components above λ.

A significant number of papers have focused on the low-rank case of PCA (Musco & Musco, 2015; Allen-Zhu & Li, 2016a; 2017) and its online variant (Allen-Zhu & Li, 2016b). Unfortunately, all of these methods require a running time that scales at least linearly with respect to the number of top principal components.

More related to this paper is work on *matrix sign function*, which plays an important role in control theory and quantum chromodynamics. Several results have addressed Krylov methods for applying the sign function in the so-called Krylov subspace, without explicitly constructing any approximate polynomial (van den Eshof et al., 2002; Schilders et al., 2008). However, Krylov methods are not (γ,ε)-approximate PCP solvers, and there is no supporting stability theory behind them.[4] Other iterative methods have also been proposed, see Section 5 of textbook (Higham, 2008). For instance, Schur's method is a slow one and also requires the matrix to be explicitly given. The Newton's iteration and its numerous variants (e.g. (Nakatsukasa & Freund, 2016)) provide rational approximations to the matrix sign function as opposed to polynomial approximations. Our result and Frostig *et al.* (Frostig et al., 2016) differ from these cited works, because we have only accessed an *approximate ridge regression* oracle, so ensuring a *polynomial* approximation to the sign function and ensuring its *stability* are crucial.

Using matrix Chebyshev polynomials to approximate matrix functions is not new. Perhaps the most celebrated example is to approximate $\mathbf{S}^{-1}$ using polynomials on $\mathbf{S}$, used in the analysis of conjugate gradient (Shewchuk, 1994). Independent from this paper,[5] Han *et al.* (Han et al., 2016) used Chebyshev polynomials to approximate the trace of the matrix sign function, i.e., $\mathbf{Tr}(\mathrm{sgn}(\mathbf{S}))$, which is similar but a different problem.[6] Also, they did not study the case when the matrix-vector multiplication oracle is only approximate (like we do in this paper), or the case when $\mathbf{S}$ has eigenvalues in the range $[-\gamma,\gamma]$.

Roadmap. In Section 2, we provide notions for this paper and basics for Chebyshev polynomials. In Section 3, we formally define approximate PCP and PCR, and reduce PCR to PCP. In Section 4, we show a general lemma for Chebyshev approximations. In Section 5, we design our polynomial approximation to $\mathrm{sgn}(x)$. In Section 6, we show how to stably compute Chebyshev polynomials. In Section 7, we state our main theorems regarding PCP and PCR. In Section 8, we provide empirical evaluations.

2 Preliminaries

We denote by $\mathbb{1}[e]\in\{0,1\}$ the indicator function for event e, by $\|v\|$ or $\|v\|_2$ the Euclidean norm of a vector v, by $\mathbf{M}^\dagger$ the Moore-Penrose pseudo-inverse of a symmetric matrix $\mathbf{M}$, and by $\|\mathbf{M}\|_2$ its spectral norm. We sometimes use $\vec{v}$ to emphasize that v is a vector.

Given a symmetric $d\times d$ matrix $\mathbf{M}$ and any $f\colon\mathbb{R}\to\mathbb{R}$, $f(\mathbf{M})$ is the matrix function applied to $\mathbf{M}$, which is equal to $\mathbf{U}\mathsf{diag}\{f(D_1),\dots,f(D_d)\}\mathbf{U}^\top$ if $\mathbf{M}=\mathbf{U}\mathsf{diag}\{D_1,\dots,D_d\}\mathbf{U}^\top$ is its eigendecomposition.

Throughout the paper, matrix $\mathbf{A}$ is of dimension $d'\times d$. We denote by $\sigma_{\max}(\mathbf{A})$ the largest singular value of $\mathbf{A}$. Following the tradition of (Frostig et al., 2016) and keeping the notations light, we assume without loss of generality that $\sigma_{\max}(\mathbf{A})\le 1$. We are interested in PCP and PCR problems with an eigenvalue threshold $\lambda\in(0,1)$.

Throughout the paper, we denote by $\lambda_1\ge\cdots\ge\lambda_d\ge 0$ the eigenvalues of $\mathbf{A}^\top\mathbf{A}$, and by $\nu_1,\dots,\nu_d\in\mathbb{R}^d$ the eigenvectors of $\mathbf{A}^\top\mathbf{A}$ corresponding to $\lambda_1,\dots,\lambda_d$. We denote by $\mathbf{P}_\lambda$ the projection matrix $\mathbf{P}_\lambda:=(\nu_1,\dots,\nu_j)(\nu_1,\dots,\nu_j)^\top$ where j is the largest index satisfying $\lambda_j\ge\lambda$. In other words, $\mathbf{P}_\lambda$ is a projection matrix to the eigenvectors of $\mathbf{A}^\top\mathbf{A}$ with eigenvalues $\ge\lambda$.

Definition 2.1. *The principal component projection (PCP) of $\chi\in\mathbb{R}^d$ at threshold λ is $\xi^*=\mathbf{P}_\lambda\chi$.*

Definition 2.2. *The principal component regression (PCR) of regressand $b\in\mathbb{R}^{d'}$ at threshold λ is*

$$x^*=\arg\min_{y\in\mathbb{R}^d}\|\mathbf{A}\mathbf{P}_\lambda y-b\|_2 \quad \textit{or equivalently}$$
$$x^*=(\mathbf{A}^\top\mathbf{A})^\dagger\mathbf{P}_\lambda(\mathbf{A}^\top b)\ .$$

2.1 Ridge Regression

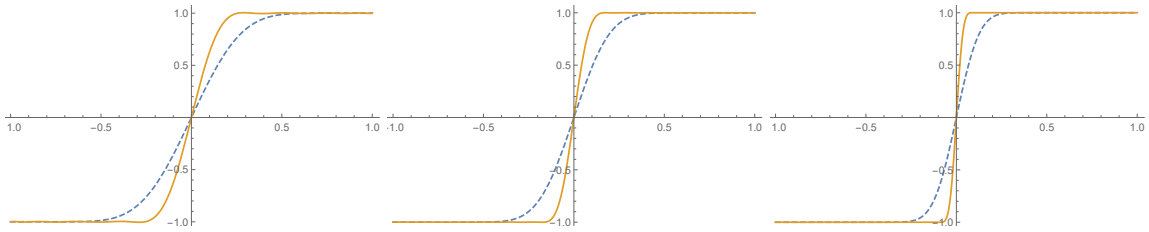
Definition 2.3. *A black-box algorithm $\mathtt{ApxRidge}(\mathcal{A},\lambda,u)$ is an ε-approximate ridge regression solver, if for every $u\in\mathbb{R}^d$, it satisfies $\|\mathtt{ApxRidge}(\mathcal{A},\lambda,u)-(\mathbf{A}^\top\mathbf{A}+\lambda\mathbf{I})^{-1}u\|\le\varepsilon\|u\|$.*

Ridge regression is equivalent to solving well-conditioned linear systems, or minimizing strongly convex and smooth objectives $f(y):=\frac{1}{2}y^\top(\mathbf{A}^\top\mathbf{A}+\lambda\mathbf{I})y-u^\top y$.

[4] We anyways have included Krylov method in our empirical evaluation section and shall discuss its performance there, see the full version of this paper.

[5] Their paper appeared online two months before us, and we became aware of their work in March 2017.

[6] In particular, their degree of the Chebyshev polynomial is $O\big(\gamma^{-1}(\log^2(1/\gamma)+\log(1/\gamma)\log(1/\varepsilon))\big)$ in the language of this paper; in contrast, we have degree $O\big(\gamma^{-1}\log(1/\gamma\varepsilon)\big)$.



(a) degree 21 (b) degree 41 (c) degree 101

Figure 1: Comparing our polynomial $g(x)$ (orange solid curve) with that of Frostig *et al.* (blue dashed curve).

Remark 2.4. There is huge literature on efficient algorithms solving ridge regression. Most notably,

(1) Conjugate gradient (Shewchuk, 1994) or accelerated gradient descent (Nesterov, 2004) gives fastest full-gradient methods;

(2) SVRG (Johnson & Zhang, 2013) and its acceleration Katyusha (Allen-Zhu, 2017) give the fastest stochastic-gradient method; and

(3) NUACDM (Allen-Zhu et al., 2016) gives the fastest coordinate-descent method.

The running time of (1) is $O(\mathsf{nnz}(\mathbf{A})\lambda^{-1/2}\log(1/\varepsilon))$ where $\mathsf{nnz}(\mathbf{A})$ is time to multiply $\mathbf{A}$ to any vector. The running times of (2) and (3) depend on structural properties of $\mathbf{A}$ and are always faster than (1).

Because the best complexity of ridge regression depends on the structural properties of $\mathbf{A}$, following Frostig *et al.*, we only compute our running time in terms of the "number of black-box calls" to a ridge regression solver.

2.2 Chebyshev Polynomials

Definition 2.5. *Chebyshev polynomials of 1st and 2nd kind are* $\{\mathcal{T}_n(x)\}_{n\geq 0}$ *and* $\{\mathcal{U}_n(x)\}_{n\geq 0}$ *where*

$$\mathcal{T}_0(x) := 1, \quad \mathcal{T}_1(x) := x, \quad \mathcal{T}_{n+1}(x) := 2x\cdot\mathcal{T}_n(x) - \mathcal{T}_{n-1}(x)$$
$$\mathcal{U}_0(x) := 1, \quad \mathcal{U}_1(x) := 2x, \quad \mathcal{U}_{n+1}(x) := 2x\cdot\mathcal{U}_n(x) - \mathcal{U}_{n-1}(x)$$

Fact 2.6 ((Trefethen, 2013))**.** *It satisfies* $\frac{d}{dx}\mathcal{T}_n(x) = n\mathcal{U}_{n-1}(x)$ *for* $n \geq 1$ *and*

$$\forall n\geq 0\colon \mathcal{T}_n(x) = \begin{cases} \cos(n\arccos(x)), & \text{if } |x|\leq 1; \\ \cosh(n\,\mathrm{arccosh}(x)), & \text{if } x\geq 1; \\ (-1)^n\cosh(n\,\mathrm{arccosh}(-x)), & \text{if } x\leq -1. \end{cases}$$

In particular, when $x \geq 1$,

$$\mathcal{T}_n(x) = \frac{1}{2}\left[\left(x-\sqrt{x^2-1}\right)^n + \left(x+\sqrt{x^2-1}\right)^n\right]$$
$$\mathcal{U}_n(x) = \frac{1}{2\sqrt{x^2-1}}\left[\left(x+\sqrt{x^2-1}\right)^{n+1} - \left(x-\sqrt{x^2-1}\right)^{n+1}\right]$$

Definition 2.7. *For function* $f(x)$ *whose domain contains* $[-1,1]$*, its degree-*n *Chebyshev truncated series and degree-*n *Chebyshev interpolation are respectively*

$$p_n(x) := \sum_{k=0}^{n} a_k\mathcal{T}_k(x) \quad \text{and} \quad q_n(x) := \sum_{k=0}^{n} c_k\mathcal{T}_k(x)\ ,$$

$$\text{where} \quad a_k := \frac{2-\mathbb{1}[k=0]}{\pi}\int_{-1}^{1}\frac{f(x)\mathcal{T}_k(x)}{\sqrt{1-x^2}}dx$$
$$c_k := \frac{2-\mathbb{1}[k=0]}{n+1}\sum_{j=0}^{n} f\left(x_j\right)\mathcal{T}_k\left(x_j\right)\ .$$

Above, $x_j := \cos\left(\frac{(j+0.5)\pi}{n+1}\right) \in [-1,1]$ *is the* j*-th Chebyshev point of order* n*.*

The following lemma is known as the aliasing formula for Chebyshev coefficients:

Lemma 2.8 (cf. Theorem 4.2 of (Trefethen, 2013))**.** *Let* f *be Lipschitz continuous on* $[-1,1]$ *and* $\{a_k\},\{c_k\}$ *be defined in Def. 2.7, then*

$$c_0 = a_0+a_{2n}+a_{4n}+\ldots \quad , \quad c_n = a_n+a_{3n}+a_{5n}+\ldots \quad , \text{ and}$$

for every $k\in\{1,2,\ldots,n-1\}$,

$$c_k = a_k+(a_{k+2n}+a_{k+4n}+\ldots)+(a_{-k+2n}+a_{-k+4n}+\ldots)$$

Definition 2.9. *For every* $\rho>0$*, let* $\mathcal{E}_\rho$ *be the ellipse* $\mathcal{E}$ *of foci* ± 1 *with major radius* $1+\rho$*. (This is also known as Bernstein ellipse with parameter* $1+\rho+\sqrt{2\rho+\rho^2}$*.)*

The following lemma is the main theory regarding Chebyshev approximation:

Lemma 2.10 (cf. Theorem 8.1 and 8.2 of (Trefethen, 2013))**.** *Suppose* $f(z)$ *is analytic on* $\mathcal{E}_\rho$ *and* $|f(z)|\leq M$ *on* $\mathcal{E}_\rho$*. Let* $p_n(x)$ *and* $q_n(x)$ *be the degree-*n *Chebyshev truncated series and Chebyshev interpolation of* $f(x)$ *on* $[-1,1]$*. Then,*

- $\max_{x\in[-1,1]}|f(x)-p_n(x)| \leq \frac{2M}{\rho+\sqrt{2\rho+\rho^2}}\left(1+\rho+\sqrt{2\rho+\rho^2}\right)^{-n}$;
- $\max_{x\in[-1,1]}|f(x)-q_n(x)| \leq \frac{4M}{\rho+\sqrt{2\rho+\rho^2}}\left(1+\rho+\sqrt{2\rho+\rho^2}\right)^{-n}$.
- $|a_0|\leq M$ *and* $|a_k|\leq 2M\left(1+\rho+\sqrt{2\rho+\rho^2}\right)^{-k}$ *for* $k\geq 1$.

3 Approximate PCP and PCR

We formalize our notions of approximation for PCP and PCR, and provide a reduction from PCR to PCP.

3.1 Our Notions of Approximation

Recall that Frostig *et al.* (Frostig et al., 2016) work only with matrices $\mathbf{A}$ that satisfy the eigengap assumption, that is, $\mathbf{A}$ has no singular value in the range

$[\sqrt{\lambda(1-\gamma)}, \sqrt{\lambda(1+\gamma)}]$. Their approximation guarantees are very straightforward:

- an output ξ is ε-approximate for PCP on vector χ if $\|\xi - \xi^*\| \le \varepsilon\|\chi\|$;
- an output x is ε-approximate for PCR with regressand b if $\|x - x^*\| \le \varepsilon\|b\|$.

Unfortunately, these notions are too strong and impossible to satisfy for matrices that do not have a large eigengap around the projection threshold λ.

In this paper we propose the following more general (but yet very meaningful) approximation notions.

Definition 3.1. *An algorithm $\mathcal{B}(\chi)$ is (γ,ε)-approximate PCP for threshold λ, if for every $\chi \in \mathbb{R}^d$*

1. $\left\|\mathbf{P}_{(1+\gamma)\lambda}\big(\mathcal{B}(\chi) - \chi\big)\right\| \le \varepsilon\|\chi\|$.
2. $\left\|(\mathbf{I} - \mathbf{P}_{(1-\gamma)\lambda})\mathcal{B}(\chi)\right\| \le \varepsilon\|\chi\|$.
3. *$\forall i$ such that $\lambda_i \in \big[(1-\gamma)\lambda, (1+\gamma)\lambda\big]$, it satisfies $|\langle \nu_i, \mathcal{B}(\chi) - \chi\rangle| \le |\langle \nu_i, \chi\rangle| + \varepsilon\|\chi\|$.*

Intuitively, the first property above states that, if projected to the eigenspace with eigenvalues above $(1+\gamma)\lambda$, then $\mathcal{B}(\chi)$ and χ are almost identical; the second property states that, if projected to the eigenspace with eigenvalues below $(1-\gamma)\lambda$, then $\mathcal{B}(\chi)$ is almost zero; and the third property states that, for each eigenvector ν_i with eigenvalue in the range $[(1-\gamma)\lambda, (1+\gamma)\lambda]$, the projection $\langle \nu_i, \mathcal{B}(\chi)\rangle$ must be between 0 and $\langle \nu_i, \chi\rangle$ (but up to an error $\varepsilon\|\chi\|$).

Naturally, $\mathbf{P}_\lambda(\chi)$ itself is a $(0,0)$-approximate PCP.

We propose the following notion for approximate PCR:

Definition 3.2. *An algorithm $\mathcal{C}(b)$ is (γ,ε)-approximate PCR for threshold λ, if for every $b \in \mathbb{R}^{d'}$*

1. $\left\|(\mathbf{I} - \mathbf{P}_{(1-\gamma)\lambda})\mathcal{C}(b)\right\| \le \varepsilon\|b\|$.
2. $\|\mathbf{A}\mathcal{C}(b) - b\| \le \|\mathbf{A}x^* - b\| + \varepsilon\|b\|$.

where $x^* = (\mathbf{A}^\top\mathbf{A})^\dagger\mathbf{P}_{(1+\gamma)\lambda}\mathbf{A}^\top b$ is the exact PCR solution for threshold $(1+\gamma)\lambda$.

The first notion states that the output $x = \mathcal{C}(b)$ has nearly no correlation with eigenvectors below threshold $(1-\gamma)\lambda$; and the second states that the regression error should be nearly optimal with respect to the exact PCR solution but at a different threshold $(1+\gamma)\lambda$.

Relationship to Frostig *et al.* Under eigengap assumption, our notions are equivalent to Frostig *et al.*:

Fact 3.3. *If $\mathbf{A}$ has no singular value in $[\sqrt{\lambda(1-\gamma)}, \sqrt{\lambda(1+\gamma)}]$, then*

- *Def. 3.1 is equivalent to $\|\mathcal{B}(\chi) - \mathbf{P}_\lambda(\chi)\| \le O(\varepsilon)\|\chi\|$.*
- *Def. 3.2 implies $\|\mathcal{C}(\chi) - x^*\| \le O(\varepsilon/\lambda)\|b\|$ and $\|\mathcal{C}(\chi) - x^*\| \le O(\varepsilon)\|b\|$ implies Def. 3.2.*

Above, $x^* = (\mathbf{A}^\top\mathbf{A})^\dagger\mathbf{P}_\lambda\mathbf{A}^\top b$ is the exact PCR solution.

3.2 Reductions from PCR to PCP

If the PCP solution $\xi = \mathbf{P}_\lambda(\mathbf{A}^\top b)$ is computed exactly, then by definition one can compute $(\mathbf{A}^\top\mathbf{A})^\dagger\xi$ which gives a solution to PCR by solving a linear system. However, as pointed by Frostig *et al.* (Frostig et al., 2016), this computation is problematic if ξ is only approximate. The following approach has been proposed to improve its accuracy by Frostig *et al.*

- "compute $p((\mathbf{A}^\top\mathbf{A} + \lambda\mathbf{I})^{-1})\xi$ where $p(x)$ is a polynomial that approximates function $\frac{x}{1-\lambda x}$."

This is a good approximation to $(\mathbf{A}^\top\mathbf{A})^\dagger\xi$ because the composition of functions $\frac{x}{1-\lambda x}$ and $\frac{1}{1+\lambda x}$ is exactly x^{-1}. Frostig *et al.* picked $p(x) = p_m(x) = \sum_{t=1}^m \lambda^{t-1}x^t$ which is a truncated Taylor series, and used the following procedure to compute $s_m \approx p_m((\mathbf{A}^\top\mathbf{A} + \lambda\mathbf{I})^{-1})\xi$:

$$s_0 = \mathcal{B}(\mathbf{A}^\top b), \quad s_1 = \texttt{ApxRidge}(\mathbf{A}, \lambda, s_0),$$
$$\forall k \ge 1\colon\ s_{k+1} = s_1 + \lambda \cdot \texttt{ApxRidge}(\mathbf{A}, \lambda, s_k)\ . \quad (3.1)$$

Above, $\mathcal{B}$ is an approximate PCP solver and `ApxRidge` is an approximate ridge regression solver. Under *eigengap assumption*, Frostig *et al.* (Frostig et al., 2016) showed

Lemma 3.4 (PCR-to-PCP)**.** *For fixed $\lambda, \gamma, \varepsilon \in (0,1)$, let $\mathbf{A}$ be a matrix whose singular values lie in $\big[0, \sqrt{(1-\gamma)\lambda}\big] \cup \big[\sqrt{(1-\gamma)\lambda}, 1\big]$. Let* `ApxRidge` *be any $O(\frac{\varepsilon}{m^2})$-approximate ridge regression solver, and let $\mathcal{B}$ be any $(\gamma, O(\frac{\varepsilon\lambda}{m^2}))$-approximate PCP solver[7]. Then, procedure (3.1) satisfies*

$$\|s_m - (\mathbf{A}^\top\mathbf{A})^\dagger\mathbf{P}_\lambda\mathbf{A}^\top b\| \le \varepsilon\|b\| \quad \text{if} \quad m = \Theta(\log(1/\varepsilon\gamma))\ .$$

Unfortunately, the above lemma *does not hold* without eigengap assumption. In this paper, we fix this issue by proving the following analogous lemma:

Lemma 3.5 (gap free PCR-to-PCP)**.** *For fixed $\lambda, \varepsilon \in (0,1)$ and $\gamma \in (0, 2/3]$, let $\mathbf{A}$ be a matrix whose singular values are no more than 1. Let* `ApxRidge` *be any $O(\frac{\varepsilon}{m^2})$-approximate ridge regression solver, and $\mathcal{B}$ be any $(\gamma, O(\frac{\varepsilon\lambda}{m^2}))$-approximate PCP solver. Then, procedure (3.1) satisfies,*

$$\left\{\begin{array}{c} \|(\mathbf{I} - \mathbf{P}_{(1-\gamma)\lambda})s_m\| \le \varepsilon\|b\| \ \text{, and} \\ \|\mathbf{A}s_m - b\| \le \|\mathbf{A}(\mathbf{A}^\top\mathbf{A})^\dagger\mathbf{P}_{(1+\gamma)\lambda}\mathbf{A}^\top b - b\| + \varepsilon\|b\| \end{array}\right\}$$

if $m = \Theta(\log(1/\varepsilon\gamma)$

Note that the conclusion of this lemma exactly corresponds to the two properties in our Def. 3.2. The proof of Lemma 3.5 is not hard, but requires a very careful case analysis by decomposing vectors b and each s_k into three components, each corresponding to eigenvalues of $\mathbf{A}^\top\mathbf{A}$ in the range $[0, (1-\gamma)\lambda]$, $[(1-\gamma)\lambda, (1+\gamma)\lambda]$ and $[(1+\gamma)\lambda, 1]$.

[7] Recall from Fact 3.3 that this requirement is equivalent to saying that $\|\mathcal{B}(\chi) - \mathbf{P}_\lambda\chi\| \le O(\frac{\varepsilon\sqrt{\lambda}}{m^2})\|\chi\|$.

We defer the details to the full version.

4 Chebyshev Approximation Outside $[-1, 1]$

Classical Chebyshev approximation theory (such as Lemma 2.10) only talks about the behaviors of $p_n(x)$ or $g_n(x)$ on interval $[-1, 1]$. However, for the purpose of this paper, we must also bound its value for $x > 1$. We prove the following general lemma in the full version, and believe it could be of independent interest: (we denote by $f^{(k)}(x)$ the k-th derivative of f at x)

Lemma 4.1. *Suppose $f(z)$ is analytic on $\mathcal{E}_\rho$ and for every $k \geq 0$, $f^{(k)}(0) \geq 0$. Then, for every $n \in \mathbb{N}$, letting $p_n(x)$ and $q_n(x)$ be be the degree-n Chebyshev truncated series and Chebyshev interpolation of $f(x)$, we have*

$$\forall y \in [0, \rho]\colon \quad 0 \leq p_n(1+y), q_n(1+y) \leq f(1+y) \ .$$

5 Our Polynomial Approximation of $\operatorname{sgn}(x)$

For fixed $\kappa \in (0, 1]$, we consider the degree-n Chebyshev interpolation $q_n(x) = \sum_{k=0}^{n} c_k \mathcal{T}_k(x)$ of the function $f(x) = \left(\frac{1+\kappa-x}{2}\right)^{-1/2}$ on $[-1, 1]$. Def. 2.7 tells us that

$$c_k := \frac{2 - \mathbb{1}[k=0]}{n+1} \sum_{j=0}^{n} \left(\sqrt{2}\cos\left(\frac{k(j+0.5)\pi}{n+1}\right)\right) \times \left(1 + \kappa - \cos\left(\frac{(j+0.5)\pi}{n+1}\right)\right)^{-1/2} .$$

Our final polynomial to approximate $\operatorname{sgn}(x)$ is therefore

$$g_n(x) = x \cdot q_n(1+\kappa-2x^2) \quad \text{and} \quad \deg(g_n(x)) = 2n+1 \ .$$

We prove the following theorem in this section:

Theorem 5.1. *For every $\alpha \in (0, 1], \varepsilon \in (0, 1/2)$, choosing $\kappa = 2\alpha^2$, our function $g_n(x) := x \cdot q_n(1 + \kappa - 2x^2)$ satisfies that as long as $n \geq \frac{1}{\sqrt{2}\alpha} \log \frac{3}{\varepsilon\alpha^2}$, then (see also Figure 1)*

- *$|g_n(x) - \operatorname{sgn}(x)| \leq \varepsilon$ for every $x \in [-1, \alpha] \cup [\alpha, 1]$.*
- *$g_n(x) \in [0, 1]$ for every $x \in [0, \alpha]$ and $g_n(x) \in [-1, 0]$ for every $x \in [-\alpha, 0]$.*

Note that our degree $n = O\big(\alpha^{-1} \log(1/\alpha\varepsilon)\big)$ is near-optimal, because the minimum degree for a polynomial to satisfy even only the first item is $\Theta\big(\alpha^{-1} \log(1/\varepsilon)\big)$ (Eremenko & Yuditskii, 2007; 2011). However, the results of (Eremenko & Yuditskii, 2007; 2011) are not constructive, and thus may not lead to stable matrix polynomials.

We prove Theorem 5.1 by first establishing two simple lemmas. The following lemma is a consequence of Lemma 2.10:

Lemma 5.2. *For every $\varepsilon \in (0, 1/2)$ and $\kappa \in (0, 1]$, if $n \geq \frac{1}{\sqrt{\kappa}} \left(\log \frac{1}{\kappa} + \log \frac{4}{\varepsilon}\right)$ then*

$$\forall x \in [-1, 1], \quad |f(x) - q_n(x)| \leq \varepsilon \ .$$

Proof of Lemma 5.2. Denoting by $f(z) = \left(\frac{1+\kappa-z}{2}\right)^{-0.5}$, we know that $f(z)$ is analytic on ellipse $\mathcal{E}_\rho$ with $\rho = \kappa/2$, and it satisfies $|f(z)| \leq \sqrt{2/\kappa}$ in $\mathcal{E}_\rho$. Applying Lemma 2.10, we know that when $n \geq \frac{1}{\sqrt{\kappa}} \left(\log \frac{1}{\kappa} + \log \frac{4}{\varepsilon}\right)$ it satisfies $|f(x) - q_n(x)| \leq \varepsilon$. □

The next lemma an immediate consequence of our Lemma 4.1 with $f(z) = \left(\frac{1+\kappa-z}{2}\right)^{-0.5}$:

Lemma 5.3. *For every $\varepsilon \in (0, 1/2), \kappa \in (0, 1], n \in \mathbb{N}$, and $x \in [0, \kappa]$, we have*

$$0 \leq q_n(1+x) \leq \left(\frac{\kappa - x}{2}\right)^{-1/2} \ .$$

Proof of Theorem 5.1. We are now ready to prove Theorem 5.1.

- When $x \in [-1, \alpha] \cup [\alpha, 1]$, it satisfies $1 + \kappa - 2x^2 \in [-1, 1]$. Therefore, applying Lemma 5.2 we have whenever $n \geq \frac{1}{\sqrt{\kappa}} \log \frac{6}{\varepsilon\kappa} = \frac{1}{\sqrt{2}\alpha} \log \frac{3}{\varepsilon\alpha^2}$ it satisfies $|f(1 + \kappa - 2x^2) - q_n(1+\kappa-2x^2)|_\infty \leq \varepsilon$. This further implies
$$|g_n(x) - \operatorname{sgn}(x)| = |x q_n(1+\kappa-2x^2) - x f(1+\kappa-2x^2)| \leq |x||f(1 + \kappa - 2x^2) - q_n(1 + \kappa - 2x^2)| \leq \varepsilon \ .$$
- When $|x| \leq \alpha$, it satisfies $1 + \kappa - 2x^2 \in [1, 1 + \kappa]$. Applying Lemma 5.3 we have for all $x \in [0, \alpha]$,
$$0 \leq g_n(x) = x \cdot q_n(1 + \kappa - 2x^2) \leq x \cdot (x^2)^{-1/2} = 1$$
and similarly for $x \in [-\alpha, 0]$ it satisfies $0 \geq g_n(x) \geq -1$. □

A Bound on Chebyshev Coefficients. We also give an upper bound to the coefficients of polynomial $q_n(x)$. Its proof can be found in the full version, and this upper bound shall be used in our final stability analysis.

Lemma 5.4 (coefficients of q_n)**.** *Let $q_n(x) = \sum_{k=0}^{n} c_k \mathcal{T}_k(x)$ be the degree-n Chebyshev interpolation of $f(x) = \left(\frac{1+\kappa-x}{2}\right)^{-1/2}$ on $[-1, 1]$. Then, for all $i \in \{0, 1, \ldots, n\}$,*

$$|c_i| \leq \frac{e\sqrt{32(i+1)}}{\kappa} \left(1 + \kappa + \sqrt{2\kappa + \kappa^2}\right)^{-i}$$

6 Stable Computation of Matrix Chebyshev Polynomials

In this section we show that any polynomial that is a weighted summation of Chebyshev polynomials with bounded coefficients, can be stably computed when applied to matrices with approximate computations. We achieve so by first generalizing Clenshaw's backward method to matrix case in Section 6.1 in order to compute a matrix variant of Chebyshev sum, and then analyze its stability in Section 6.2 with the help from Elloit's forward-backward transformation (Elliott, 1968).

Remark 6.1. We wish to point out that although Chebyshev polynomials are known to be stable under error when computed on *scalars* (Gil et al., 2007), it is not immediately clear why it holds also for matrices. Recall that Chebyshev polynomials satisfy $\mathcal{T}_{n+1}(x) = 2x\mathcal{T}_n(x) - \mathcal{T}_{n-1}(x)$. In the matrix case, we have $\mathcal{T}_{n+1}(\mathbf{M})\chi = 2\mathbf{M}\mathcal{T}_n(\mathbf{M})\chi - \mathcal{T}_{n-1}(\mathbf{M})\chi$ where $\chi \in \mathbb{R}^d$ is a vector. If we analyzed this formula coordinate by coordinate, error could blow up by a factor d per iteration.

In addition, we need to ensure that the stability theorem holds for matrices $\mathbf{M}$ with eigenvalues that can exceed 1. This is not standard because Chebyshev polynomials are typically analyzed only on domain $[-1, 1]$.

6.1 Clenshaw's Method in Matrix Form

Consider any computation of the form

$$\vec{s}_N := \sum_{k=0}^{N} \mathcal{T}_k(\mathbf{M})\vec{c}_k \in \mathbb{R}^d \tag{6.1}$$

where $\mathbf{M} \in \mathbb{R}^{d\times d}$ is symmetric and each $\vec{c}_k$ is in $\mathbb{R}^d$. (Note that for PCP and PCR purposes, we it suffices to consider $\vec{c}_k = c'_k\chi$ where $c'_k \in \mathbb{R}$ is a scalar and $\chi \in \mathbb{R}^d$ is a fixed vector for all k. However, we need to work on this more general form for our stability analysis.)

Vector s_N can be computed using the following procedure:

Lemma 6.2 (backward recurrence)**.** *$\vec{s}_N = \vec{b}_0 - \mathbf{M}\vec{b}_1$ where*

$$\vec{b}_{N+1} := \vec{0}, \quad \vec{b}_N := \vec{c}_N, \quad \textit{and}$$

$$\forall r \in \{N-1, \ldots, 0\} \colon \vec{b}_r := 2\mathbf{M}\vec{b}_{r+1} - \vec{b}_{r+2} + \vec{c}_r \in \mathbb{R}^d \ .$$

6.2 Inexact Clenshaw's Method in Matrix Form

We show that, if implemented using the backward recurrence formula, the Chebyshev sum of (6.1) can be stably computed. We define the following model to capture the error with respect to matrix-vector multiplications.

Definition 6.3 (inexact backward recurrence)**.** *Let $\mathcal{M}$ be an approximate algorithm that satisfies $\|\mathcal{M}(u) - \mathbf{M}u\|_2 \le \varepsilon\|u\|_2$ for every $u \in \mathbb{R}^d$. Then, define inexact backward recurrence to be*

$$\widehat{b}_{N+1} := 0, \quad \widehat{b}_N := \vec{c}_N, \quad \textit{and}$$

$$\forall r \in \{N-1, \ldots, 0\} \colon \widehat{b}_r := 2\mathcal{M}\big(\widehat{b}_{r+1}\big) - \widehat{b}_{r+2} + \vec{c}_r \in \mathbb{R}^d \ ,$$

and define the output as $\widehat{s}_N := \widehat{b}_0 - \mathcal{M}(\widehat{b}_1)$.

The following theorem gives an error analysis to our inexact backward recurrence. We prove it in full version, and the main idea of our proof is to convert each error vector of a recursion of the backward procedure into an error vector corresponding to some original $\vec{c}_k$.

Theorem 6.4 (stable Chebyshev sum)**.** *For every $N \in \mathbb{N}^*$, suppose the eigenvalues of $\mathbf{M}$ are in $[a, b]$ and suppose there are parameters $C_U \ge 1, C_T \ge 1, \rho \ge 1, C_c \ge 0$ satisfying $\forall k \in \{0, 1, \ldots, N\}$:*

$$\left\{\rho^k\|\vec{c}_k\| \le C_c \quad \bigwedge \quad \forall x \in [a, b] \colon \begin{array}{c} |\mathcal{T}_k(x)| \le C_T\rho^k \\ |\mathcal{U}_k(x)| \le C_U\rho^k \end{array}\right\} \ .$$

Then, if the inexact backward recurrence in Def. 6.3 is applied with $\varepsilon \le \frac{1}{4NC_U}$, we have

$$\|\widehat{s}_N - \vec{s}_N\| \le \varepsilon \cdot 2(1 + 2NC_T)NC_UC_c \ .$$

7 Algorithms and Main Theorems for PCP and PCR

We are now ready to state our main theorems for PCP and PCR. We first note a simple fact:

Fact 7.1. *$(\mathbf{P}_\lambda)\chi = \frac{\mathbf{I}+\mathrm{sgn}(\mathbf{S})}{2}$ where $\mathbf{S} := 2(\mathbf{A}^\top\mathbf{A} + \lambda\mathbf{I})^{-1}\mathbf{A}^\top\mathbf{A} - \mathbf{I} = (\mathbf{A}^\top\mathbf{A} + \lambda\mathbf{I})^{-1}(\mathbf{A}^\top\mathbf{A} - \lambda\mathbf{I})$.*

In other words, for every vector $\chi \in \mathbb{R}^d$, the exact PCP solution $\mathbf{P}_\lambda(\chi)$ is the same as computing $(\mathbf{P}_\lambda)\chi = \frac{\mathbf{I}+\mathrm{sgn}(\mathbf{S})}{2}\chi$. Thus, we can use our polynomial $g_n(x)$ introduced in Section 5 and compute $g_n(\mathbf{S})\chi \approx \mathrm{sgn}(\mathbf{S})\chi$. Finally, in order to compute $g_n(\mathbf{S})$, we need to multiply $\mathbf{S}$ to $\deg(g_n)$ vectors; whenever we do so, we call perform ridge regression once.

Since the high-level structure of our PCP algorithm is very clear, due to space limitation, we present the pseudocodes of our PCP and PCR algorithms in the full version.

7.1 Our Main Theorems

We first state our main theorem under the eigengap assumption, in order to provide a direct comparison to that of Frostig *et al.* (Frostig et al., 2016).

Theorem 7.2 (eigengap assumption)**.** *Given $\mathbf{A} \in \mathbb{R}^{d'\times d}$ and $\lambda, \gamma \in (0, 1)$, assume that the singular values of $\mathbf{A}$ are in the range $[0, \sqrt{(1-\gamma)\lambda}] \cup [\sqrt{(1+\gamma)\lambda}, 1]$. Given $\chi \in \mathbb{R}^d$ and $b \in \mathbb{R}^{d'}$, denote by*

$$\xi^* = \mathbf{P}_\lambda\chi \quad \textit{and} \quad x^* = (\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{P}_\lambda\mathbf{A}^\top b$$

the exact PCP and PCR solutions, and by `ApxRidge` *any ε'-approximate ridge regression solver. Then,*

- `QuickPCP` *outputs ξ satisfying $\|\xi^* - \xi\| \le \varepsilon\|\chi\|$ with $O\big(\gamma^{-1}\log\frac{1}{\gamma\varepsilon}\big)$ oracle calls to* `ApxRidge` *as long as $\log(1/\varepsilon') = \Theta\big(\log\frac{1}{\gamma\varepsilon}\big)$.*
- `QuickPCR` *outputs x satisfying $\|x - x^*\| \le \varepsilon\|b\|$ with $O\big(\gamma^{-1}\log\frac{1}{\gamma\lambda\varepsilon}\big)$ oracle calls to* `ApxRidge`*, as long as $\log(1/\varepsilon') = \Theta\big(\log\frac{1}{\gamma\lambda\varepsilon}\big)$.*

In contrast, the number of ridge-regression oracle calls was $\Theta(\gamma^{-2}\log\frac{1}{\gamma\varepsilon})$ for PCP and $\Theta(\gamma^{-2}\log\frac{1}{\gamma\lambda\varepsilon})$ for PCR in (Frostig et al., 2016). We include the proof of Theorem 7.2 in the full version.

We state our theorem without the eigengap assumption.

Theorem 7.3 (gap-free). *Given $\mathbf{A} \in \mathbb{R}^{d' \times d}$, $\lambda \in (0,1)$, and $\gamma \in (0, 2/3]$, assume that $\|\mathbf{A}\|_2 \leq 1$. Given $\chi \in \mathbb{R}^d$ and $b \in \mathbb{R}^{d'}$, and suppose* `ApxRidge` *is an ε'-approximate ridge regression solver, then*

- `QuickPCP` *outputs ξ that is (γ, ε)-approximate PCP with $O\big(\gamma^{-1} \log \frac{1}{\gamma\varepsilon}\big)$ oracle calls to* `ApxRidge` *as long as $\log(1/\varepsilon') = \Theta\big(\log \frac{1}{\gamma\varepsilon}\big)$.*
- `QuickPCR` *outputs x that is (γ, ε)-approximate PCR with $O\big(\gamma^{-1} \log \frac{1}{\gamma\lambda\varepsilon}\big)$ oracle calls to* `ApxRidge` *as long as $e\log(1/\varepsilon') = \Theta\big(\log \frac{1}{\gamma\lambda\varepsilon}\big)$.*

We make a final remark here regarding the practical usage of `QuickPCP` and `QuickPCR`.

Remark 7.4. Since our theory is for (γ, ε)-approximations that have two parameters, the user in principle has to feed in both γ and n where n is the degree of the polynomial approximation to the sign function. In practice, however, it is usually sufficient to obtain $(\varepsilon, \varepsilon)$-approximate PCP and PCR. Therefore, our pseudocodes allow users to set $\gamma = 0$ and thus ignore this parameter γ; in such a case, we shall use $\gamma = \log(n)/n$ which is equivalent to setting $\gamma = \Theta(\varepsilon)$ because $n = \Theta(\gamma^{-1} \log(1/\gamma\varepsilon))$.

8 Experiments

We provide empirical evaluations in the full version of this paper.

9 Conclusion

We summarize our contributions.

- We put forward approximate notions for PCP and PCR that do not rely on any eigengap assumption. Our notions reduce to standard ones under the eigengap assumption.
- We design near-optimal polynomial approximation $g(x)$ to $\mathrm{sgn}(x)$ satisfying (1.1) and (1.2).
- We develop general stable recurrence formula for matrix Chebyshev polynomials; as a corollary, our $g(x)$ can be applied to matrices in a stable manner.
- We obtain faster, provable PCA-free algorithms for PCP and PCR than known results.

References

Allen-Zhu, Zeyuan. Katyusha: The First Direct Acceleration of Stochastic Gradient Methods. In *STOC*, 2017.

Allen-Zhu, Zeyuan and Li, Yuanzhi. LazySVD: Even Faster SVD Decomposition Yet Without Agonizing Pain. In *NIPS*, 2016a.

Allen-Zhu, Zeyuan and Li, Yuanzhi. First Efficient Convergence for Streaming k-PCA: a Global, Gap-Free, and Near-Optimal Rate. *ArXiv e-prints*, abs/1607.07837, July 2016b.

Allen-Zhu, Zeyuan and Li, Yuanzhi. Doubly Accelerated Methods for Faster CCA and Generalized Eigendecomposition. In *Proceedings of the 34th International Conference on Machine Learning*, ICML '17, 2017.

Allen-Zhu, Zeyuan, Richtárik, Peter, Qu, Zheng, and Yuan, Yang. Even faster accelerated coordinate descent using non-uniform sampling. In *ICML*, 2016.

Boutsidis, Christos and Magdon-Ismail, Malik. Faster SVD-truncated regularized least-squares. In *2014 IEEE International Symposium on Information Theory*, pp. 1321–1325. IEEE, 2014.

Chan, Tony F and Hansen, Per Christian. Computing truncated singular value decomposition least squares solutions by rank revealing QR-factorizations. *SIAM Journal on Scientific and Statistical Computing*, 11(3):519–530, 1990.

Elliott, David. Error analysis of an algorithm for summing certain finite series. *Journal of the Australian Mathematical Society*, 8(02):213–221, 1968.

Eremenko, Alexandre and Yuditskii, Peter. Uniform approximation of sgn x by polynomials and entire functions. *Journal d'Analyse Mathématique*, 101(1):313–324, 2007.

Eremenko, Alexandre and Yuditskii, Peter. Polynomials of the best uniform approximation to sgn (x) on two intervals. *Journal d'Analyse Mathématique*, 114(1):285–315, 2011.

Frostig, Roy, Musco, Cameron, Musco, Christopher, and Sidford, Aaron. Principal Component Projection Without Principal Component Analysis. In *ICML*, 2016.

Gil, Amparo, Segura, Javier, and Temme, Nico M. *Numerical Methods for Special Functions*. Society for Industrial and Applied Mathematics, jan 2007. ISBN 978-0-89871-634-4. doi: 10.1137/1.9780898717822. URL `http://epubs.siam.org/doi/abs/10.1137/1.9780898717822http://epubs.siam.org/doi/book/10.1137/1.9780898717822`.

Han, Insu, Malioutov, Dmitry, Avron, Haim, and Shin, Jinwoo. Approximating the spectral sums of large-scale matrices using chebyshev approximations. *arXiv preprint arXiv:1606.00942*, 2016.

Higham, N. *Functions of Matrices*. Society for Industrial and Applied Mathematics, 2008. doi: 10.1137/1.9780898717778. URL `http://epubs.siam.org/doi/abs/10.1137/1.9780898717778`.

Johnson, Rie and Zhang, Tong. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, NIPS 2013, pp. 315–323, 2013.

Musco, Cameron and Musco, Christopher. Randomized block krylov methods for stronger and faster approximate singular value decomposition. In *NIPS*, pp. 1396–1404, 2015.

Nakatsukasa, Yuji and Freund, Roland W. Computing fundamental matrix decompositions accurately via the matrix sign function in two iterations: The power of zolotarev's functions. *SIAM Review*, 58(3):461–493, 2016.

Nesterov, Yurii. *Introductory Lectures on Convex Programming Volume: A Basic course*, volume I. Kluwer Academic Publishers, 2004. ISBN 1402075537.

Schilders, Wilhelmus H.A., Van der Vorst, Henk A., and Rommes, Joost. *Model order reduction: theory, research aspects and applications*, volume 13. Springer, 2008.

Shewchuk, Jonathan Richard. An introduction to the conjugate gradient method without the agonizing pain, 1994.

Trefethen, Lloyd N. *Approximation Theory and Approximation Practice*. SIAM, 2013.

van den Eshof, Jasper, Frommer, Andreas, Lippert, Th, Schilling, Klaus, and van der Vorst, Henk A. Numerical methods for the qcdd overlap operator. i. sign-function and error bounds. *Computer Physics Communications*, 146(2):203–224, 2002.