CSE 392: Matrix and Tensor Algorithms for Data Spring 2024

Lecture 13 — 02/28/2024

Instructor: Shashanka Ubaru *Scribe: Shourya Pandey*

1 Recap

In the last lecture, we introduced iterative methods, which predate sketching-based methods, for low rank approximation of a matrix. Recall the *Power Method* for computing the top singular vector of a matrix:

Algorithm 1: Power Method

Data: $\boldsymbol{A} \in \mathbb{R}^{n \times d}, q \in \mathbb{N}$

$z_0 \sim \mathcal{N}(0, \boldsymbol{I}_{d \times d})$

$z_0 \leftarrow \frac{z_0}{\|z_0\|_2}$

for $\ell = 1, 2, \ldots, q$ **do**

  $z_\ell \leftarrow \boldsymbol{A}^\top (\boldsymbol{A} z_{\ell-1})$

  $z_\ell \leftarrow \frac{z_\ell}{\|z_\ell\|_2}$

return z_q

The following theorems record the guarantee of the power method in the gapped and gapless cases.

Theorem 1 (Power Method, Gapped)**.** *Let $\boldsymbol{A} \in \mathbb{R}^{n \times d}$ be a matrix with singular values $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_{\min(n,d)}$ and top singular vector v_1, and let $\gamma := \frac{\sigma_1 - \sigma_2}{\sigma_1}$. Then, for any $\epsilon, \delta \in (0,1)$ with $\delta = \exp(-O(d))$, the Power Method (Algorithm 1) with $q = O\left(\frac{\log(d/\epsilon) + \log(1/\delta)}{\gamma}\right)$ satisfies*

$$\|v_1 - z_q\|_2 \leq \epsilon$$

with probability at least $1 - \delta$. Moreover, the algorithm runs in time $O\left(\mathrm{nnz}\left(\boldsymbol{A}\right) \frac{\log(d/\epsilon) + \log(1/\delta)}{\gamma}\right)$.

Theorem 2 (Power Method, Gapless)**.** *Let $\boldsymbol{A} \in \mathbb{R}^{n \times d}$ be a matrix with singular values $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_{\min(n,d)}$ and top singular vector v_1, and let $\gamma := \frac{\sigma_1 - \sigma_2}{\sigma_1}$. Then, for any $\epsilon, \delta \in (0,1)$ with $\delta = \exp(-O(d))$, the Power Method (Algorithm 1) with $q = O\left(\frac{\log(d/\epsilon) + \log(1/\delta)}{\epsilon}\right)$ satisfies*

$$\|\boldsymbol{A} - \boldsymbol{A} z_q z_q^\top\|_F^2 \leq (1 + \epsilon)\|\boldsymbol{A} - \boldsymbol{A} v_1 v_1^\top\|_F^2$$

with probability at least $1 - \delta$. Moreover, the algorithm runs in time $O\left(\mathrm{nnz}\left(\boldsymbol{A}\right) \frac{\log(d/\epsilon) + \log(1/\delta)}{\epsilon}\right)$.

Note that either of these guarantees implies

$$\|\boldsymbol{A} z_q\|_2^2 \geq (1 - \epsilon)^2 \sigma_1^2.$$

In the gapped case, we can closely align the vector z_q with the top singular vector v_1, while in the gapless case the flexibility of the power method extends to aligning z_q with the eigenspace of right eigenvectors with sufficiently large singular values. Finally, we saw a natural extension of the Power Method, called the *Block Power Method*, for computing the top k singular vectors of $\boldsymbol{A}$.

2 Krylov Subspaces

To motivate the definition of Krylov subspaces, consider a linear regression problem of the form

$$\min_{x\in\mathbb{R}^d} F(x) := \frac{1}{2}\|\boldsymbol{C}x - b\|_2^2 = \min_{x\in\mathbb{R}^d}\left(\frac{1}{2}x^\top \boldsymbol{A}x - x^\top v + \frac{1}{2}\|b\|_2^2\right),$$

where $\boldsymbol{C} \in \mathbb{R}^{n\times d}$, $b \in \mathbb{R}^n$, $\boldsymbol{A} = \boldsymbol{C}^\top\boldsymbol{C}$, and $v = \boldsymbol{A}^\top b$. This is a convex optimization problem, because

$$\nabla^2 F(x) = \boldsymbol{C}^\top\boldsymbol{C} \succeq \boldsymbol{0}.$$

A natural approach is to use a descent algorithm. the gradient is given by $\nabla F(x) = \boldsymbol{A}x - v$, so we initialize the gradient descent method with x_0 equal to a multiple of v, then after q descent steps the point x_q is in the span of the vectors $v, \boldsymbol{A}v, \dots, \boldsymbol{A}^q v$. We call

$$\boldsymbol{K}_q(\boldsymbol{A}, v) := \mathrm{Span}\,(v, \boldsymbol{A}v, \dots, \boldsymbol{A}^q v)$$

the *Krylov subspace* of dimension q generated by $(\boldsymbol{A}, v)$. We will also use the notation $\boldsymbol{K}_q$ for the Krylov subspace in case the matrix $\boldsymbol{A}$ and the vector v is clear from the context. An equivalent definition is

$$\boldsymbol{K}_q(\boldsymbol{A}, v) := \{p(\boldsymbol{A})v \mid p \text{ is a polynomial of degree at most } q\}.$$

Suppose $\boldsymbol{A}$ has full rank. Then, the optimal solution to the linear regression problem is $x = \boldsymbol{A}^{-1}v$. Krylov subspace methods try to avoid the $O(nd^{\omega-1} + d^\omega)$ cost of matrix multiplication ($\boldsymbol{A} = \boldsymbol{C}^\top\boldsymbol{C}$) and matrix inversion by approximating $\boldsymbol{A}^{-1}$ using polynomials in $\boldsymbol{A}$.

Remark 1. The definition of $\boldsymbol{K}_q$ immediately implies that $\boldsymbol{K}_{q'} \subseteq \boldsymbol{K}_q$ for $q' \leq q$. Moreover, $\boldsymbol{K}_q \subseteq \mathbb{R}_d$, which has dimension d. This implies the existence of an index q_1 such that

$$\boldsymbol{K}_0 \subsetneq \boldsymbol{K}_1 \subsetneq \cdots \subsetneq \boldsymbol{K}_{q_1} = \boldsymbol{K}_{q_1+1}.$$

It can be shown that $\boldsymbol{K}_{q_1} = \boldsymbol{K}_q$ for all $q \geq q_1$. Consider the minimal polynomial p of degree $1 \leq r \leq d$ such that $p(\boldsymbol{A}) = \boldsymbol{0}$. Then, $\boldsymbol{A}^r$ is expressible as a linear combination of the matrices $\boldsymbol{I}, \boldsymbol{A}, \dots, \boldsymbol{A}^{r-1}$., which implies that $\boldsymbol{K}_r = \boldsymbol{K}_{r-1}$. Therefore, $q_1 \leq r - 1$. A partial converse is also true: there exists a vector z_1 such that q_1 achieves the value $r - 1$.

3 Lanczos Algorithm

Reconsider the problem of finding the top eigenvector of a symmetric matrix $\boldsymbol{A}$. The Krylov iteration methods introduced for linear regression apply more generally via a strategy known as *Lanczos algorithm* or *Lanczos iteration*. The Lanczos algorithm takes a symmetric matrix $\boldsymbol{A}$ and finds a matrix $\boldsymbol{Z}_q$ which is an orthonormal basis of a certain Krylov subspace $\boldsymbol{K}\,(\boldsymbol{A}, z_1)$, and such that $\boldsymbol{T}_q := \boldsymbol{Z}_q^\top \boldsymbol{A}\boldsymbol{Z}_q$ is a tridiagonal matrix. While the eigenvectors and eigenvalues are not apparent from the tridiagonal form, computing $\boldsymbol{T}_q$ is already a significant step towards it.

Algorithm 2: Lanczos Algorithm

Data: $\boldsymbol{A} \in \mathbb{R}^{n\times d}, q \in \mathbb{N}$

$z_0 \leftarrow \mathbf{0}$, $\beta_1 \leftarrow 0$

Choose a starting vector $z_1 \in \mathbb{R}^d$ with unit norm.

$z_0 \sim \mathcal{N}(0, \boldsymbol{I}_{d\times d})$

for $\ell = 1, 2, \ldots, q-1$ **do**

  $y_\ell \leftarrow \boldsymbol{A} z_\ell - \beta_\ell z_{\ell-1}$

  $\alpha_\ell \leftarrow \langle y_\ell, z_\ell \rangle$

  $y_\ell \leftarrow y_\ell - \alpha_\ell z_\ell$

  $\beta_{\ell+1} \leftarrow \|y_\ell\|_2$. If $\beta_{\ell+1} = 0$ then exit the loop.

  $z_{\ell+1} \leftarrow \frac{y_\ell}{\beta_{\ell+1}}$

$Z_q \leftarrow [z_1 \; z_2 \; \ldots \; z_q]$

return Z_q

The matrix $\boldsymbol{T}_q = \boldsymbol{Z}_q^\top \boldsymbol{A} \boldsymbol{Z}_q$ is called the *Rayleigh Ritz-projection* and is given by

$$\boldsymbol{T}_q = \begin{bmatrix} \alpha_1 & \beta_2 & & & & \\ \beta_2 & \alpha_2 & \beta_3 & & & \\ & \beta_3 & \alpha_3 & \beta_4 & & \\ & & \cdot & \cdot & \cdot & \\ & & & \cdot & \cdot & \cdot \\ & & & & \beta_q & \alpha_q \end{bmatrix}$$

If u is a top eigenvector estimate of $\boldsymbol{T}_q$, then $\boldsymbol{Z}_q u$ is the estimate of the eigenvector of $\boldsymbol{A}$.

Theorem 3 (Lanczos Algorithm, Gapped)**.** *Let $\gamma := \frac{\lambda_1 - \lambda_2}{\lambda_1}$ be the gap between the largest eigenvalue, λ_1, and the second largest eigenvalue, λ_2, of $\boldsymbol{A} \in \mathbb{S}^{d\times d}_{\succeq \mathbf{0}}$, and let v_1 be the top eigenvector of $\boldsymbol{A}$* [1]*. Let $\epsilon, \delta \in (0,1)$ with $\delta = \exp(-O(d))$. If the Lanczos's algorithm (Algorithm 2) is initialized with a normalized random Gaussian vector with $q = O\left(\frac{\log(d/\epsilon)+\log(1/\delta)}{\sqrt{\gamma}}\right)$, and u is the top eigenvector of $\boldsymbol{T}_q = \boldsymbol{Z}_q^\top \boldsymbol{A} \boldsymbol{Z}_q$, then the vector $w = \boldsymbol{Z}_q u$ satisfies*

$$\|\boldsymbol{A} - \boldsymbol{A} w w^\top\|_F^2 \le (1+\epsilon)\|\boldsymbol{A} - \boldsymbol{A} v_1 v_1^\top\|_F^2 \tag{1}$$

with probability at least $1-\delta$. Moreover, the algorithm takes time $O\left(\mathrm{nnz}\left(\boldsymbol{A}\right) \frac{\log(d/\epsilon)+\log(1/\delta)}{\sqrt{\gamma}}\right)$ [2].

Proof. First, assuming $\boldsymbol{Z}_q$ has full rank, we claim that the amongst all vectors that span the Krylov subspace $\boldsymbol{K}_q\left(\boldsymbol{A}, z_1\right)$ (which is also the span of the columns of $\boldsymbol{Z}_q$), the vector $w = \boldsymbol{Z}_q u$ minimizes $\|\boldsymbol{A} - \boldsymbol{A} w w^\top\|_F^2$. Any vector in the span of $\boldsymbol{Z}_q$ is of the form $y = \boldsymbol{Z}_q x$ for some $x \in \mathbb{R}^q$. Now,

$$\begin{aligned} \|\boldsymbol{A} - \boldsymbol{A} y y^\top\|_F^2 &= \|\boldsymbol{A} - \boldsymbol{A}\boldsymbol{Z}_q x x^\top \boldsymbol{Z}_q^\top\|_F^2 \\ &= \mathrm{Tr}\left(\boldsymbol{A}^\top\boldsymbol{A} - \boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{Z}_q x x^\top \boldsymbol{Z}_q^\top - \boldsymbol{Z}_q x x^\top \boldsymbol{Z}_q^\top \boldsymbol{A}^\top\boldsymbol{A} + \boldsymbol{Z}_q x x^\top \boldsymbol{Z}_q \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{Z}_q x x^\top \boldsymbol{Z}_q^\top\right) \\ &= \mathrm{Tr}\left(\boldsymbol{A}^\top\boldsymbol{A} - 2x^\top \boldsymbol{Z}_q^\top \boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{Z}_q x + \left(x^\top \boldsymbol{Z}_q^\top \boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{Z}_q x\right)\left(x^\top \boldsymbol{Z}_q^\top \boldsymbol{Z}_q x\right)\right) \\ &= \mathrm{Tr}\left(\boldsymbol{A}^\top\boldsymbol{A}\right) - 2x^\top \boldsymbol{Z}_q^\top \boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{Z}_q x + \left(x^\top \boldsymbol{Z}_q^\top \boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{Z}_q x\right)\|x\|_2^2. \end{aligned}$$

[1] Variants of the Lanczos algorithm work for non-symmetric matrices too, such as Arnoldi's iterations.

[2] The time taken to compute the top eigenvector u of $\boldsymbol{T}_q$ is $O(q^3)$ and can be made as small as $O(q \log q)$ via the Fast Multipole Method [1] for tridiagonal matrices.

The second equality used $\|\boldsymbol{X}\|_F^2 = \mathrm{Tr}\left(\boldsymbol{X}^\top \boldsymbol{X}\right)$, the third equality used the cyclic property of trace, and the fourth equality used $\boldsymbol{Z}_q^\top \boldsymbol{Z}_q = \boldsymbol{I}$. From this, it is clear that this problem admits a global minimum x, and this x satisfies

$$\begin{aligned} &\nabla\left(\mathrm{Tr}\left(\boldsymbol{A}^\top \boldsymbol{A}\right) - 2x^\top \boldsymbol{Z}_q^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{Z}_q x + \left(x^\top \boldsymbol{Z}_q^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{Z}_q x\right)\|x\|_2^2\right) = 0 \\ \Longrightarrow\ &\left(2 - \|x\|_2^2\right)\boldsymbol{Z}_q^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{Z}_q x = \|\boldsymbol{A}\boldsymbol{Z}_q x\|_2^2 x. \end{aligned} \tag{2}$$

Left multiplying $x^\top$ yields

$$2\left(1 - \|x\|_2^2\right)\|\boldsymbol{A}\boldsymbol{Z}_q x\|_2^2 = 0.$$

If $\boldsymbol{A}\boldsymbol{Z}_q x = 0$, then $\|\boldsymbol{A} - \boldsymbol{A}ww^\top\|_F^2 = \|\boldsymbol{A}\|_F^2$. Otherwise, $\|x\|_2 = 1$. Plugging into equation (2),

$$\boldsymbol{Z}_q^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{Z}_q x = \|\boldsymbol{A}\boldsymbol{Z}_q x\|_2^2 x,$$

which means x is a (unit) eigenvector of $\boldsymbol{Z}_q^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{Z}_q$ with a non-zero eigenvalue. Since $\boldsymbol{Z}_q$ is orthonormal, $y = \boldsymbol{Z}_q x$ is also a unit vector, which means $yy^\top$ is a rank-1 projection matrix. Therefore, the problem is equivalent to maximizing

$$\|\boldsymbol{A}yy^\top\|_F^2 = \|\boldsymbol{A}y\|_2^2 = \|\boldsymbol{A}\boldsymbol{Z}_q x\|_2^2,$$

which is achieved when x is the top eigenvector of $\boldsymbol{Z}_q^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{Z}_q$. Since $\boldsymbol{Z}_q$ is full rank and orthonormal, and $\boldsymbol{A}$ is symmetric, $\boldsymbol{Z}_q^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{Z}_q = \left(\boldsymbol{Z}_q^\top \boldsymbol{A}\boldsymbol{Z}_q\right)^2$ and $\boldsymbol{Z}_q^\top \boldsymbol{A}\boldsymbol{Z}_q$ is symmetric. Therefore, x is also the top eigenvector of $\boldsymbol{Z}_q^\top \boldsymbol{A}\boldsymbol{Z}_q = \boldsymbol{T}_q$, i.e. $x = u$.

Next, we show that if $q = O\left(\frac{\log(d/\epsilon)+\log(1/\delta)}{\sqrt{\gamma}}\right)$, then there exists a unit vector y in the span of $\boldsymbol{Z}_q$ such that $|\langle v_1, y\rangle| \geq 1 - \epsilon$.

With some work, it can be shown that $\boldsymbol{Z}_q$ is indeed an orthonormal basis of the Krylov subspace $\boldsymbol{K}_q(\boldsymbol{A}, z_1)$; for a full proof, see [8]. Therefore, for any polynomial p_q of degree at most q there exists an x such that $\boldsymbol{Z}_q x = p_q(\boldsymbol{A})z_1$. Suppose we show that there is a good approximate top eigenvector in the Krylov subspace, that is, there is a polynomial p_q such that $p_q(\boldsymbol{A})z_1$ is an approximate top eigenvector of $\boldsymbol{A}$. Then, from our previous claim about $w = \boldsymbol{Z}_q u$, the vector w is also an approximate top eigenvector of $\boldsymbol{A}$. Note crucially that we only need to show the existence of such a polynomial, do not need to explicitly compute it.

To this end, let $z_1 = \sum_{i=1}^d \mu_i v_i$, where $v_1, v_2, \ldots, v_d$ are the eigenvectors of $\boldsymbol{A}$ corresponding to eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \geq 0$. Then,

$$p_q(\boldsymbol{A})z_1 = \left(\sum_{i=1}^d p_q(\lambda_i)\, v_i v_i^\top\right)\left(\sum_{i=1}^d \mu_i v_i\right) = \sum_{i=1}^d \mu_i p_q(\lambda_i)\, v_i.$$

The goal is to find p_q such that $p_q(\lambda_1)$ is large, and $p_q(t)$ is small for any $0 \leq t \leq \lambda_2 \leq (1-\gamma)\lambda_1$. The following lemma (see Lemma 5 in [5]) on polynomial approximations is helpful:

Lemma 1. *Let $\epsilon', \gamma \in (0,1)$. Then, there exists a polynomial p of degree at most $O\left(\frac{1}{\sqrt{\gamma}}\log\frac{1}{\epsilon'}\right)$ such that $p(1) = 1$ and $|p(t)| \leq \epsilon'$ for all $0 \leq t \leq 1-\gamma$.*

We will also require the following high probability bound on $|\mu_1|$:

Lemma 2. *Let $g \sim \mathcal{N}(0, \boldsymbol{I}_{d\times d})$ and $\delta \in (0,1)$. Then, with probability at least $1-\delta$,*

$$\left| \frac{\|g\|_2^2}{d} - 1 \right| = O\left(\frac{\log(2/\delta)}{d} + \sqrt{\frac{\log(2/\delta)}{d}} \right).$$

Moreover, with probability at least $1-\delta$, $|g_1| = \Omega(\delta)$.

For a proof of Lemma 2 and more concentration inequalities for (sub)-Gaussians, see [9]. Note that our assumption that $\delta = \exp(-O(d))$ implies $\frac{\log(2/\delta)}{d} = O(1)$, so for all sufficiently large $\delta = \exp(-O(d))$, $\|g\|_2 \le 2\sqrt{d}$ with probability at least $1 - \frac{\delta}{2}$. Therefore, the values μ_i satisfies with probability at least $1-\delta$,

$$|\mu_1| = |\langle z_1, v_1\rangle| = \left| \frac{\langle g_1, v_1 \rangle}{\|g\|_2} \right| \ge C \frac{\delta}{\sqrt{d}}$$

for a sufficiently small universal constant C and $|\mu_i| \le 1$ for $i \ge 2$. Here, we used the fact that $|g_1| = |\langle g, e_1\rangle|$ and $|\langle g, v_1 \rangle|$ are identically distributed.

Let $\epsilon' \le \frac{C\delta\sqrt{\epsilon}}{d}$ and $p_q(t) := \hat{p}_q\left(\frac{t}{\lambda_1}\right)$ where $\hat{p}_q$ is the polynomial promised by Lemma 1. Then, $p_q(\lambda_1) = 1$ and $|p_q(\lambda_i)| \le \epsilon'$ for all $2 \le i \le d$. Letting $\rho_i := \mu_i p_q(\lambda_i)$,

$$\frac{|\rho_i|}{|\rho_1|} = \frac{|\mu_i|\epsilon'}{|\mu_1|} \le \frac{\epsilon'\sqrt{d}}{C\delta} \le \sqrt{\frac{\epsilon}{d}}.$$

This implies that for $q = O\left(\frac{1}{\sqrt{\gamma}} \log \frac{1}{\epsilon'}\right) = O\left(\frac{\log(d/\epsilon)+\log(1/\delta)}{\sqrt{\gamma}}\right)$,

$$\frac{|\langle p_q(\boldsymbol{A})z_1, v_1\rangle|^2}{\||p_q(\boldsymbol{A})z_1|\|_2^2} = \frac{\rho_1^2}{\rho_1^2 + \rho_2^2 + \cdots + \rho_d^2} \ge \frac{\rho_1^2}{\rho_1^2 + (d-1)\frac{\epsilon}{d}\rho_1^2} \ge \frac{1}{1+\epsilon} \ge 1 - \epsilon.$$

It follows that the polynomial $\frac{p_q(\boldsymbol{A})z_1}{\|p_q(\boldsymbol{A})z_1\|_2}$, and therefore the vector w, satisfies equation (1). ■

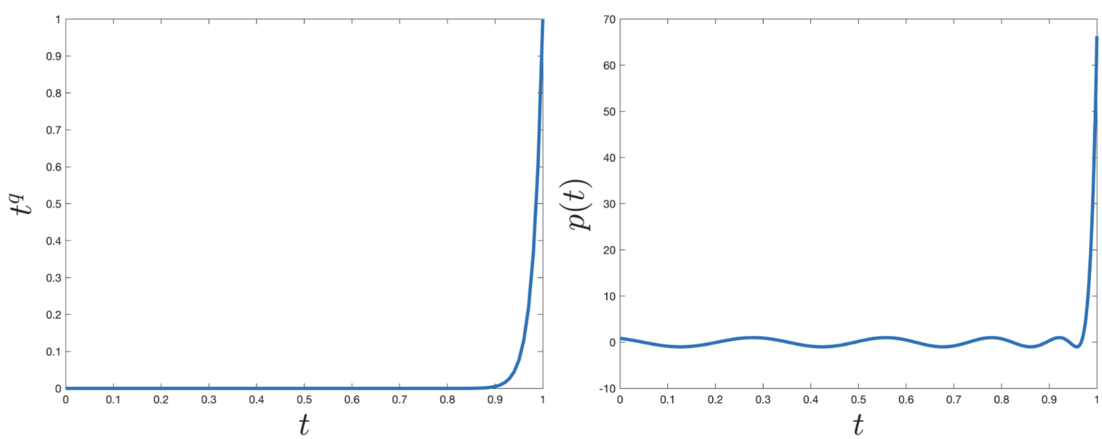


Figure 1: Comparison between t^q and $p(t)$, where p is the (unscaled) polynomial guaranteed by 1.

Remark 2. Note the $\sqrt{\gamma}$ improvement in the the runtime of the Lanczos iteration over the Power iteration 1. The main step achieving this improvement is the polynomial p_q used in 1. The Power Method applies the same technique using the polynomial $f(t) = t^q$. However, the dependence of q on γ is worse:

$$(1-\gamma)^t \leq \epsilon' \implies t = \Omega\left(\frac{1}{\gamma}\log\frac{1}{\epsilon'}\right).$$

It turns out that t^q can be approximated with a polynomial of degree roughly $\sqrt{q}$. See [4, 6, 2] for more details.

3.1 Block Krylov Methods

In the previous lecture, we saw the Block Power Method for computing the top k singular vectors of $\boldsymbol{A}$. We can similarly extend the Lanczos algorithm to the *Block Lanczos Algorithm*, which leads to a similar quadratic improvement in the number of iterations the algorithm.

Algorithm 3: Block Lanczos Algorithm

Data: $\boldsymbol{A} \in \mathbb{R}^{n\times d}, q \in \mathbb{N}, k \in \mathbb{N}$

Choose a random Gaussian matrix $\boldsymbol{S} \in \mathbb{R}^{d\times k}$

$\boldsymbol{K} \leftarrow [\boldsymbol{S}, \boldsymbol{AS}, \ldots, \boldsymbol{A}^{q-1}\boldsymbol{S}]$

$\boldsymbol{Z} \leftarrow \text{orth}(\boldsymbol{K})$, an orthogonal basis of $\boldsymbol{K}$

$\boldsymbol{T} \leftarrow \boldsymbol{Z}_q^\top \boldsymbol{A} \boldsymbol{Z}_q$

$\tilde{\boldsymbol{U}}_k \leftarrow$ top k eigenvectors of $\boldsymbol{T}$

return $\boldsymbol{Z}_q \tilde{\boldsymbol{U}}_k$

Theorem 4 (Block Lanczos Algorithm)**.** *Let $\boldsymbol{V}_k$ be the top k eigenspace of $\boldsymbol{A} \in \mathbb{S}^{d\times d}_{\succeq \boldsymbol{0}}$. Let $\epsilon, \delta \in (0,1)$ with $\delta = e^{-O(d)}$. If the Block Lanczos's algorithm (Algorithm 3) is initialized with $q = O\left(\frac{\log(d/\epsilon)+\log(1/\delta)}{\sqrt{\epsilon}}\right)$, then the output $\boldsymbol{Z} := \boldsymbol{Z}_q\tilde{\boldsymbol{U}}_k$ satisfies*

$$\|\boldsymbol{A} - \boldsymbol{A}\boldsymbol{Z}\boldsymbol{Z}^\top\|_F^2 \leq (1+\epsilon)\|\boldsymbol{A} - \boldsymbol{A}\boldsymbol{V}_k\boldsymbol{V}_k^\top\|_F^2 \tag{3}$$

with probability at least $1-\delta$. Moreover, the algorithm takes time $O\left(\text{nnz}(\boldsymbol{A})\,\frac{\log(d/\epsilon)+\log(1/\delta)}{\sqrt{\gamma}}k\right)$.

4 Linear System Solvers

We pick up from our motivation of Krylov subspaces to solve linear systems. Given a nonsingular matrix $\boldsymbol{A} \in \mathbb{R}^{d\times d}$ and a vector $b \in \mathbb{R}^d$, solve the system

$$\boldsymbol{A}x = b. \tag{4}$$

When the matrix $\boldsymbol{A}$ does not enjoy a particular structure, iterative methods are one of the most popular ways to finding an approximate solution. The idea is to solve for x via updates of the form

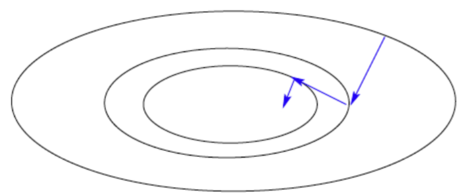
$$x_{\ell+1} \leftarrow x_\ell + \alpha r_\ell$$

for some scalar α and direction vectors r_ℓ which depend on the initial vector x_0. One such Krylov subspace method is MINRES (Minimum Residual Method). The idea is to pick x_ℓ to be the vector in the (shifted) Krylov subspace $x_0 + \mathrm{Span}\left(r_0, \boldsymbol{A}r_0, \ldots, \boldsymbol{A}^{\ell-1}r_0\right)$ (where $r_0 = b - \boldsymbol{A}x_0$) which minimizes $\|b - \boldsymbol{A}x_\ell\|_2$:

$$x_\ell \leftarrow \underset{x \in x_0 + \mathrm{Span}\left(r_0, \boldsymbol{A}r_0, \ldots, \boldsymbol{A}^{\ell-1}r_0\right)}{\arg\min} \|b - \boldsymbol{A}x\|_2$$

This is equivalent [7, 3] to moving along the direction of steepest descent:

$$r_\ell \leftarrow b - \boldsymbol{A}x_\ell$$
$$\alpha \leftarrow \frac{\langle r_\ell, r_\ell \rangle}{\langle \boldsymbol{A}r_l, r_l \rangle}$$
$$x_{\ell+1} \leftarrow x_\ell + \alpha r_\ell$$



Here, we highlight the Lanczos method for solving linear systems for symmetric matrices (a similar method exists for non-symmetric matrices via Arnoldi's iterations), which can be viewed as a repeated projection onto the Krylov subspace $\boldsymbol{K}_q(\boldsymbol{A}, b - \boldsymbol{A}x_0)$ and equivalent to the steepest descent method for solving linear systems.

Algorithm 4: Lanczos Algorithm for Linear Systems

Data: $\boldsymbol{A} \in \mathbb{R}^{n \times d}, b \in \mathbb{R}^d, x_0 \in \mathbb{R}^d, q \in \mathbb{N}$

```
r_0 ← b − A x_0, β_1 ← ‖r_0‖, r_0 ← r_0/β_1
for ℓ = 1, 2, ..., q do
    y_ℓ ← A z_ℓ − β_ℓ z_{ℓ−1}
    α_ℓ ← ⟨y_ℓ, z_ℓ⟩
    y_ℓ ← y_ℓ − α_ℓ z_ℓ
    β_{ℓ+1} ← ‖y_ℓ‖_2. If β_{ℓ+1} = 0 then exit the loop.
    z_{ℓ+1} ← y_ℓ / β_{ℓ+1}
Z_q ← [z_1 z_2 ... z_q], T_q ← tridiag(β_j, α_j, β_{j+1})
x_q ← x_0 + Z_q T_q^{−1} (β_1 e_1)
return x_q
```

4.1 Conjugate Gradient Method

The conjugate gradient (CG) method is a popular variant of the Lanczos algorithm for linear systerm, when the matrix $\boldsymbol{A}$ is positive semidefinite. In exact arithmetic, the Lanczos algorithm and the conjugate gradient method are identical.

If the matrix $\boldsymbol{A}$ is well-conditioned with condition number κ, then the CG method guarantees:

$$\|x^* - x_q\|_{\boldsymbol{A}} \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^q \|x^* - x_0\|_{\boldsymbol{A}}$$

Algorithm 5: Conjugate Gradient Method

Data: $\boldsymbol{A} \in \mathbb{S}_{\succeq \mathbf{0}}^{d \times d}, b \in \mathbb{R}^d, x_0 \in \mathbb{R}^d, q \in \mathbb{N}$

$r_0 \leftarrow b - \boldsymbol{A}x_0, p_0 \leftarrow r_0$

while *the algorithm has not converged,* **do**

  $\alpha_\ell = \langle r_\ell, r_\ell \rangle \,/\, \langle \boldsymbol{A}p_\ell, p_\ell \rangle$

  $x_{\ell+1} \leftarrow x_\ell + \alpha_\ell p_\ell$

  $r_{\ell+1} \leftarrow r_\ell - \alpha_\ell \boldsymbol{A} p_\ell$

  $\beta_\ell \leftarrow \langle r_{\ell+1}, r_{\ell+1} \rangle \,/\, \langle r_\ell, r_\ell \rangle$

  $p_{\ell+1} \leftarrow r_{\ell+1} + \beta_\ell p_\ell$

References

[1] Ed S Coakley and Vladimir Rokhlin. A fast divide-and-conquer algorithm for computing the spectra of real symmetric tridiagonal matrices. *Applied and Computational Harmonic Analysis*, 34(3):379–414, 2013.

[2] Petros Drineas, Ilse CF Ipsen, Eugenia-Maria Kontopoulou, and Malik Magdon-Ismail. Structural convergence results for approximation of dominant subspaces from block krylov spaces. *SIAM Journal on Matrix Analysis and Applications*, 39(2):567–586, 2018.

[3] Carl T Kelley. *Iterative methods for optimization*. SIAM, 1999.

[4] John C Mason and David C Handscomb. *Chebyshev polynomials*. Chapman and Hall/CRC, 2002.

[5] Cameron Musco and Christopher Musco. Randomized block krylov methods for stronger and faster approximate singular value decomposition. *Advances in neural information processing systems*, 28, 2015.

[6] Christopher Musco. Singular value decomposition and krylov subspace methods. https://www.chrismusco.com/amlds2023/notes/lecture11.html.

[7] Yousef Saad. *Iterative methods for sparse linear systems*. SIAM, 2003.

[8] Lloyd N Trefethen and David Bau. *Numerical linear algebra*. SIAM, 2022.

[9] Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.