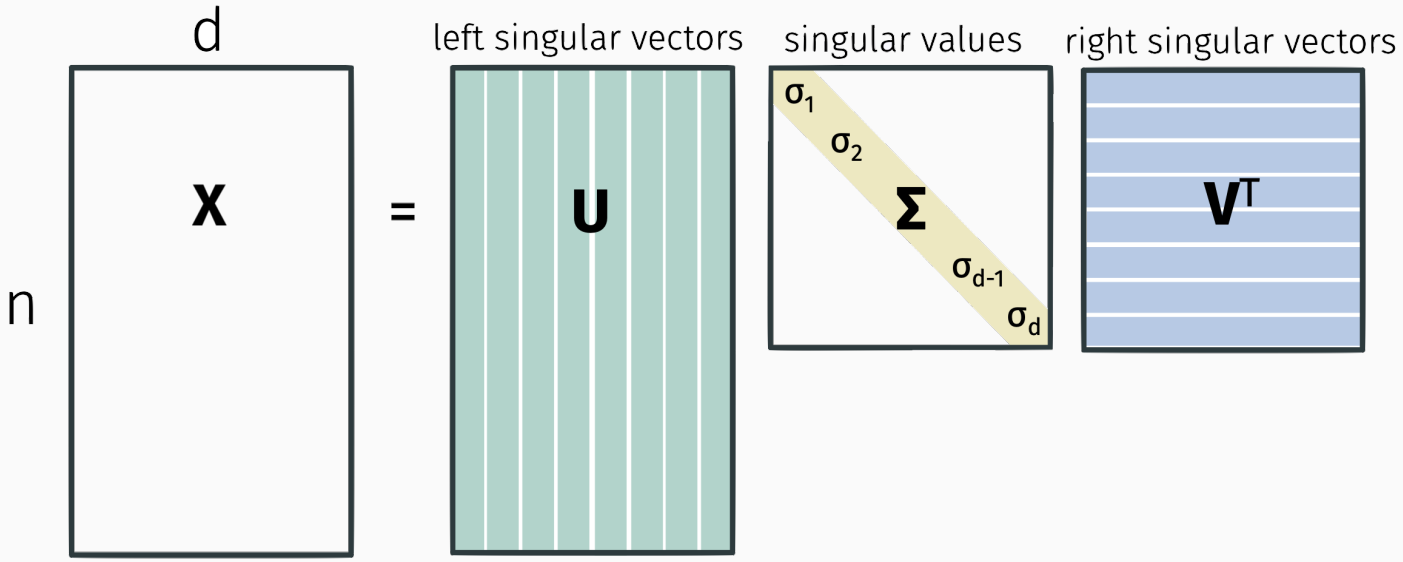
CS-GY 6763: Lecture 11
Power Method, Krylov Subspace Methods, Spectral Graph Partitioning

NYU, Prof. Ainesh Bakshi

Singular Value Decomposition

One of the most fundamental results in linear algebra.

Any matrix $\mathbf{X}$ can be written:

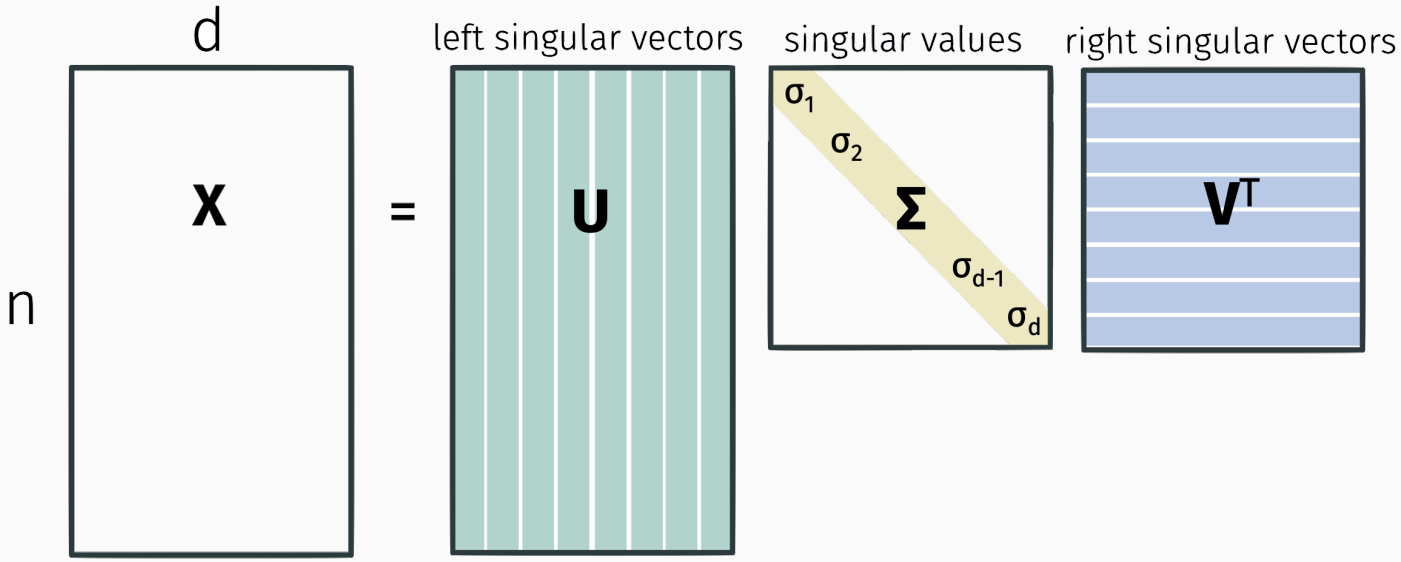


Where $\mathbf{U}^T\mathbf{U} = \mathbf{I}$, $\mathbf{V}^T\mathbf{V} = \mathbf{I}$, and $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_d \geq 0$.

Singular Value Decomposition

One of the most fundamental results in linear algebra.

Any matrix $\mathbf{X}$ can be written:

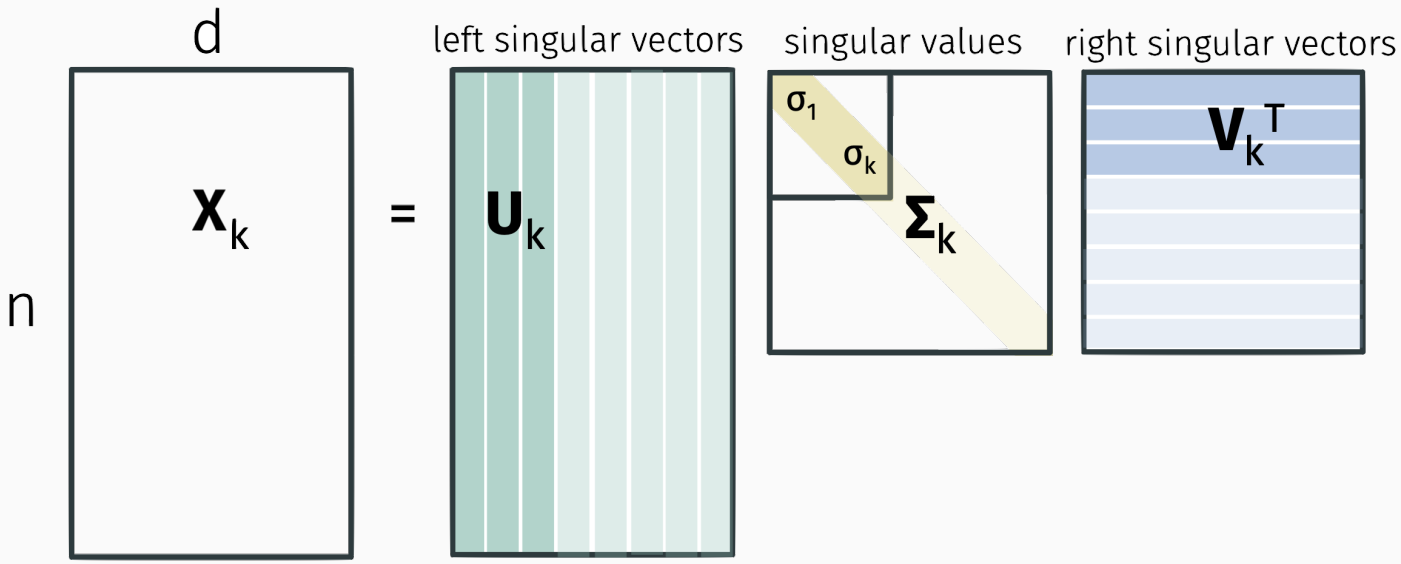


Where $\mathbf{U}^T\mathbf{U} = \mathbf{I}$, $\mathbf{V}^T\mathbf{V} = \mathbf{I}$, and $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_d \geq 0$.

Singular values are unique. Factors are not. E.g. would still get a valid SVD by multiplying both i^{th} column of $\mathbf{V}$ and $\mathbf{U}$ by -1.

Partial SVD

Key result: Can find the best low-rank approximation from the singular value decomposition.



$$\mathbf{U}_k = \underset{\text{orthogonal } \mathbf{Z} \in \mathbb{R}^{d \times k}}{\arg\min} \|\mathbf{X} - \mathbf{Z}\mathbf{Z}^T\mathbf{X}\|_F^2$$

$$\mathbf{V}_k = \underset{\text{orthogonal } \mathbf{W} \in \mathbb{R}^{d \times k}}{\arg\min} \|\mathbf{X} - \mathbf{X}\mathbf{W}\mathbf{W}^T\|_F^2$$

Computing the SVD

Suffices to compute right singular vectors $\mathbf{V}$:

- Compute $\mathbf{X}^T\mathbf{X}$.

Computing the SVD

Suffices to compute right singular vectors $\mathbf{V}$:

- Compute $\mathbf{X}^T\mathbf{X}$.
- Find eigendecomposition $\mathbf{V}\mathbf{\Lambda}\mathbf{V}^T = \mathbf{X}^T\mathbf{X}$ using e.g. QR algorithm.

Computing the SVD

Suffices to compute right singular vectors $\mathbf{V}$:

- Compute $\mathbf{X}^T\mathbf{X}$.
- Find eigendecomposition $\mathbf{V}\mathbf{\Lambda}\mathbf{V}^T = \mathbf{X}^T\mathbf{X}$ using e.g. QR algorithm.
- Compute $\mathbf{L} = \mathbf{X}\mathbf{V}$.
- Observe $\mathbf{L}_i = \mathbf{X}\mathbf{V}_i = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\mathbf{V}_i = \mathbf{U}\mathbf{\Sigma}\mathbf{e}_i = \sigma_i\mathbf{U}_i$

Computing the SVD

Suffices to compute right singular vectors $\mathbf{V}$:

- Compute $\mathbf{X}^T\mathbf{X}$.
- Find eigendecomposition $\mathbf{V}\mathbf{\Lambda}\mathbf{V}^T = \mathbf{X}^T\mathbf{X}$ using e.g. QR algorithm.
- Compute $\mathbf{L} = \mathbf{X}\mathbf{V}$.
- Observe $\mathbf{L}_i = \mathbf{X}\mathbf{V}_i = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\mathbf{V}_i = \mathbf{U}\mathbf{\Sigma}\mathbf{e}_i = \sigma_i\mathbf{U}_i$
- Set $\sigma_i = \|\mathbf{L}_i\|_2$ and $\mathbf{U}_i = \mathbf{L}_i/\|\mathbf{L}_i\|_2$.

Computing the SVD

Suffices to compute right singular vectors $\mathbf{V}$:

- Compute $\mathbf{X}^T\mathbf{X}$.
- Find eigendecomposition $\mathbf{V}\mathbf{\Lambda}\mathbf{V}^T = \mathbf{X}^T\mathbf{X}$ using e.g. QR algorithm.
- Compute $\mathbf{L} = \mathbf{X}\mathbf{V}$.
- Observe $\mathbf{L}_i = \mathbf{X}\mathbf{V}_i = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\mathbf{V}_i = \mathbf{U}\mathbf{\Sigma}\mathbf{e}_i = \sigma_i\mathbf{U}_i$
- Set $\sigma_i = \|\mathbf{L}_i\|_2$ and $\mathbf{U}_i = \mathbf{L}_i/\|\mathbf{L}_i\|_2$.

Total runtime $\approx O(nd^2 + d^3)$

Computing the SVD (Faster)

How to go faster?

- Compute approximate solution.

Computing the SVD (Faster)

How to go faster?

- Compute approximate solution.

- Only compute top k singular vectors/values (instead of entire SVD).

Computing the SVD (Faster)

How to go faster?

- Compute approximate solution.
- Only compute top k singular vectors/values (instead of entire SVD).
- Iterative algorithms achieve runtime $\approx O(ndk)$ vs. $O(nd^2)$ time.
  - **Krylov subspace methods** like the Lanczos method are most commonly used in practice.
  - **Power method** is the simplest Krylov subspace method, and still works very well.

Power Method

- **Today:** $k = 1$. **Goal:** Find $\mathbf{z} \approx \mathbf{v}_1$.
- **Input:** $\mathbf{X} \in \mathbb{R}^{n \times d}$ with SVD $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$.

Algorithm:

- Choose $\mathbf{z}^{(0)} \sim \mathcal{N}(0, \mathbf{I})$, normalize: $\mathbf{z}^{(0)} \leftarrow \mathbf{z}^{(0)} / \|\mathbf{z}^{(0)}\|_2$

Power Method

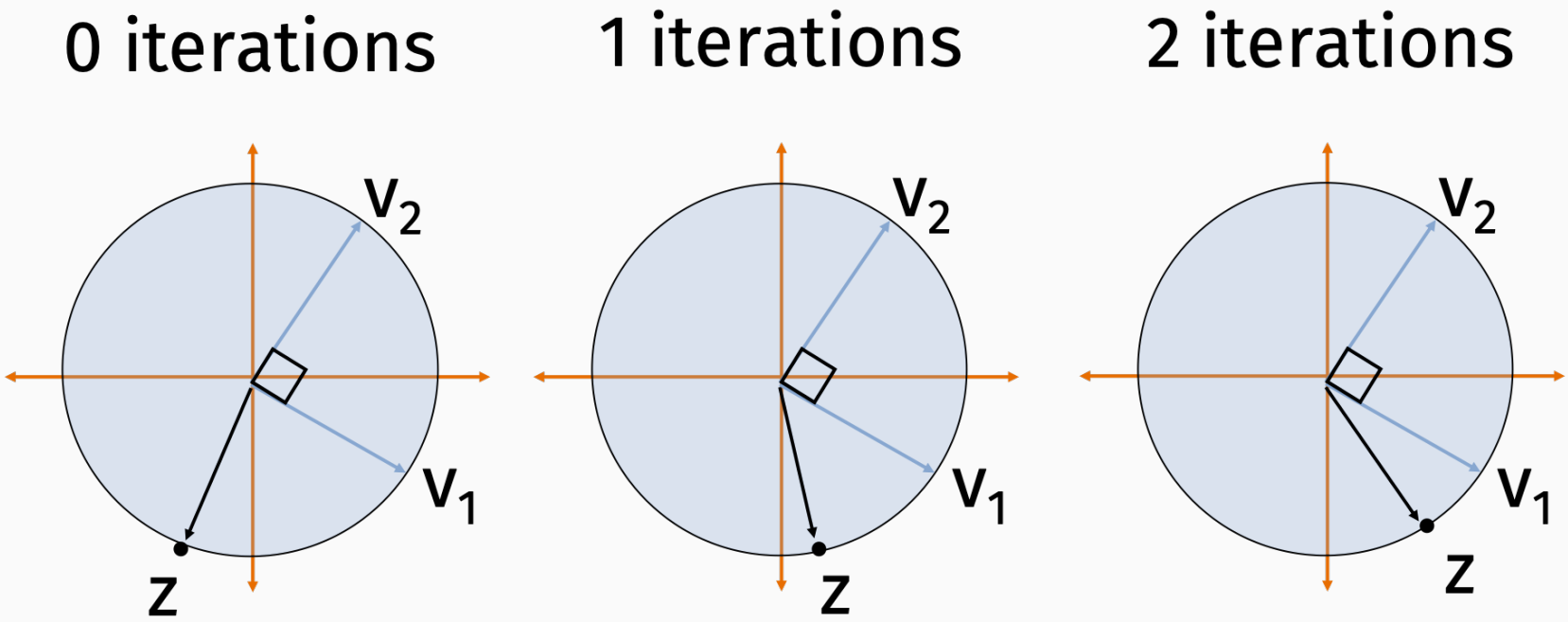
- **Today:** $k = 1$. **Goal:** Find $\mathbf{z} \approx \mathbf{v}_1$.
- **Input:** $\mathbf{X} \in \mathbb{R}^{n \times d}$ with SVD $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$.

Algorithm:

- Choose $\mathbf{z}^{(0)} \sim \mathcal{N}(0, \mathbf{I})$, normalize: $\mathbf{z}^{(0)} \leftarrow \mathbf{z}^{(0)} / \|\mathbf{z}^{(0)}\|_2$
- For $i = 1, \ldots, T$:
  - $\mathbf{z}^{(i)} = \mathbf{X}^T(\mathbf{X}\mathbf{z}^{(i-1)})$
  - $\mathbf{z}^{(i)} \leftarrow \mathbf{z}^{(i)} / \|\mathbf{z}^{(i)}\|_2$
- Return $\mathbf{z}^{(T)}$

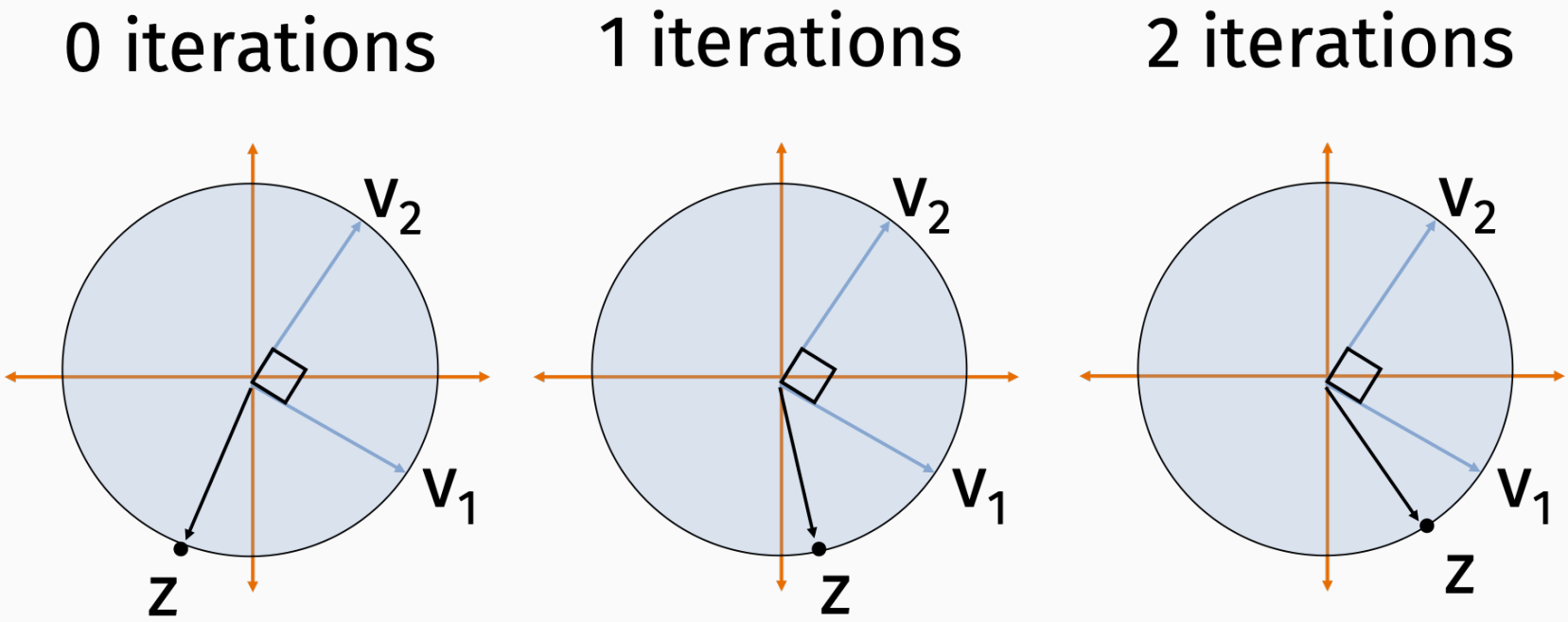
Running Time per Iteration: $O(\text{nnz}(\mathbf{X}))$, where $\text{nnz}(\mathbf{X})$ is the number of non-zero entries in $\mathbf{X}$.

Power Method Intuition



- A randomly intitionalized vector $\mathbf{z}$ gets increasingly correlated with the top singular vector $\mathbf{v}_1$.

Power Method Intuition



- A randomly intitionalized vector $\mathbf{z}$ gets increasingly correlated with the top singular vector $\mathbf{v}_1$.
- The number of iterations depend on the gap between the largest and second largest singular value

Power Method Formal Convergence

Theorem (Basic Power Method Convergence):

- Let $\gamma = \frac{\sigma_1 - \sigma_2}{\sigma_1}$ capture the gap between the first and second largest singular values.

Power Method Formal Convergence

Theorem (Basic Power Method Convergence):

- Let $\gamma = \frac{\sigma_1 - \sigma_2}{\sigma_1}$ capture the gap between the first and second largest singular values.
- Initialize with a random Gaussian vector.

Power Method Formal Convergence

Theorem (Basic Power Method Convergence):

- Let $\gamma = \frac{\sigma_1 - \sigma_2}{\sigma_1}$ capture the gap between the first and second largest singular values.
- Initialize with a random Gaussian vector.
- After $T = O\left(\frac{\log d/\epsilon}{\gamma}\right)$ steps, with high probability:

$$\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2 \leq \epsilon \quad \text{or} \quad \|\mathbf{v}_1 - (-\mathbf{z}^{(T)})\|_2 \leq \epsilon.$$

$$\textbf{Total Runtime: } O\left(nd \cdot \frac{\log d/\epsilon}{\gamma}\right)$$

- Power Method slows down when there is a small gap between σ_1 and σ_2

Power Method Formal Convergence

Theorem (Basic Power Method Convergence):

- Let $\gamma = \frac{\sigma_1 - \sigma_2}{\sigma_1}$ capture the gap between the first and second largest singular values.
- Initialize with a random Gaussian vector.
- After $T = O\left(\frac{\log d/\epsilon}{\gamma}\right)$ steps, with high probability:

$$\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2 \leq \epsilon \quad \text{or} \quad \|\mathbf{v}_1 - (-\mathbf{z}^{(T)})\|_2 \leq \epsilon.$$

$$\textbf{Total Runtime: } O\left(nd \cdot \frac{\log d/\epsilon}{\gamma}\right)$$

- Power Method slows down when there is a small gap between σ_1 and σ_2
- Matrices in practice have singular value gaps

One Step Analysis of Power Method

Write $\mathbf{z}^{(i)}$ in the right singular vector basis:

$$\mathbf{z}^{(0)} = c_1^{(0)}\mathbf{v}_1 + c_2^{(0)}\mathbf{v}_2 + \ldots + c_d^{(0)}\mathbf{v}_d \quad \text{where} \quad c_\ell^{(0)} = \langle \mathbf{z}^{(0)}, \mathbf{v}_\ell \rangle$$

One Step Analysis of Power Method

Write $\mathbf{z}^{(i)}$ in the right singular vector basis:

$$\mathbf{z}^{(0)} = c_1^{(0)}\mathbf{v}_1 + c_2^{(0)}\mathbf{v}_2 + \ldots + c_d^{(0)}\mathbf{v}_d \quad \text{where} \quad c_\ell^{(0)} = \langle \mathbf{z}^{(0)}, \mathbf{v}_\ell \rangle$$

$$\mathbf{z}^{(1)} = c_1^{(1)}\mathbf{v}_1 + c_2^{(1)}\mathbf{v}_2 + \ldots + c_d^{(1)}\mathbf{v}_d$$

One Step Analysis of Power Method

Write $\mathbf{z}^{(i)}$ in the right singular vector basis:

$$\mathbf{z}^{(0)} = c_1^{(0)}\mathbf{v}_1 + c_2^{(0)}\mathbf{v}_2 + \ldots + c_d^{(0)}\mathbf{v}_d \quad \text{where} \quad c_\ell^{(0)} = \langle \mathbf{z}^{(0)}, \mathbf{v}_\ell \rangle$$

$$\mathbf{z}^{(1)} = c_1^{(1)}\mathbf{v}_1 + c_2^{(1)}\mathbf{v}_2 + \ldots + c_d^{(1)}\mathbf{v}_d$$

$$\vdots$$

One Step Analysis of Power Method

Write $\mathbf{z}^{(i)}$ in the right singular vector basis:

$$\mathbf{z}^{(0)} = c_1^{(0)}\mathbf{v}_1 + c_2^{(0)}\mathbf{v}_2 + \ldots + c_d^{(0)}\mathbf{v}_d \quad \text{where} \quad c_\ell^{(0)} = \langle \mathbf{z}^{(0)}, \mathbf{v}_\ell \rangle$$

$$\mathbf{z}^{(1)} = c_1^{(1)}\mathbf{v}_1 + c_2^{(1)}\mathbf{v}_2 + \ldots + c_d^{(1)}\mathbf{v}_d$$

$$\vdots$$

$$\mathbf{z}^{(i)} = c_1^{(i)}\mathbf{v}_1 + c_2^{(i)}\mathbf{v}_2 + \ldots + c_d^{(i)}\mathbf{v}_d$$

One Step Analysis of Power Method

Write $\mathbf{z}^{(i)}$ in the right singular vector basis:

$$\mathbf{z}^{(0)} = c_1^{(0)}\mathbf{v}_1 + c_2^{(0)}\mathbf{v}_2 + \ldots + c_d^{(0)}\mathbf{v}_d \quad \text{where} \quad c_\ell^{(0)} = \langle \mathbf{z}^{(0)}, \mathbf{v}_\ell \rangle$$

$$\mathbf{z}^{(1)} = c_1^{(1)}\mathbf{v}_1 + c_2^{(1)}\mathbf{v}_2 + \ldots + c_d^{(1)}\mathbf{v}_d$$

$$\vdots$$

$$\mathbf{z}^{(i)} = c_1^{(i)}\mathbf{v}_1 + c_2^{(i)}\mathbf{v}_2 + \ldots + c_d^{(i)}\mathbf{v}_d$$

One Step Analysis of Power Method

Write $\mathbf{z}^{(i)}$ in the right singular vector basis:

$$\mathbf{z}^{(0)} = c_1^{(0)}\mathbf{v}_1 + c_2^{(0)}\mathbf{v}_2 + \ldots + c_d^{(0)}\mathbf{v}_d \quad \text{where} \quad c_\ell^{(0)} = \langle \mathbf{z}^{(0)}, \mathbf{v}_\ell \rangle$$

$$\mathbf{z}^{(1)} = c_1^{(1)}\mathbf{v}_1 + c_2^{(1)}\mathbf{v}_2 + \ldots + c_d^{(1)}\mathbf{v}_d$$

$$\vdots$$

$$\mathbf{z}^{(i)} = c_1^{(i)}\mathbf{v}_1 + c_2^{(i)}\mathbf{v}_2 + \ldots + c_d^{(i)}\mathbf{v}_d$$

Note: $[c_1^{(i)}, \ldots, c_d^{(i)}] = \mathbf{c}^{(i)} = \mathbf{V}^\mathsf{T}\mathbf{z}^{(i)}$.

One Step Analysis of Power Method

Write $\mathbf{z}^{(i)}$ in the right singular vector basis:

$$\mathbf{z}^{(0)} = c_1^{(0)}\mathbf{v}_1 + c_2^{(0)}\mathbf{v}_2 + \ldots + c_d^{(0)}\mathbf{v}_d \quad \text{where} \quad c_\ell^{(0)} = \langle \mathbf{z}^{(0)}, \mathbf{v}_\ell \rangle$$

$$\mathbf{z}^{(1)} = c_1^{(1)}\mathbf{v}_1 + c_2^{(1)}\mathbf{v}_2 + \ldots + c_d^{(1)}\mathbf{v}_d$$

$$\vdots$$

$$\mathbf{z}^{(i)} = c_1^{(i)}\mathbf{v}_1 + c_2^{(i)}\mathbf{v}_2 + \ldots + c_d^{(i)}\mathbf{v}_d$$

Note: $[c_1^{(i)}, \ldots, c_d^{(i)}] = \mathbf{c}^{(i)} = \mathbf{V}^\mathbf{T}\mathbf{z}^{(i)}$.

Also: Since $\mathbf{V}^T$ has orthogonormal rows and $\|\mathbf{z}^{(i)}\|_2 = 1$,

$$\|\mathbf{c}^{(i)}\|_2^2 = \|\mathbf{V}^T\mathbf{z}^{(i)}\|_2^2 = \|\mathbf{z}^{(i)}\|_2 = 1.$$

One Step Analysis of Power Method

Claim: After update $\mathbf{z}^{(i)} = \frac{1}{n_i}\mathbf{X}^T\mathbf{X}\mathbf{z}^{(i-1)}$, where

$$\mathbf{z}^{(i)} = \frac{1}{n_i}\left[c_1^{(i-1)}\sigma_1^2 \cdot \mathbf{v}_1 + c_2^{(i-1)}\sigma_2^2 \cdot \mathbf{v}_2 + \ldots + c_d^{(i-1)}\sigma_d^2 \cdot \mathbf{v}_d\right]$$

One Step Analysis of Power Method

Claim: After update $\mathbf{z}^{(i)} = \frac{1}{n_i}\mathbf{X}^T\mathbf{X}\mathbf{z}^{(i-1)}$, where

$$\mathbf{z}^{(i)} = \frac{1}{n_i}\left[c_1^{(i-1)}\sigma_1^2 \cdot \mathbf{v}_1 + c_2^{(i-1)}\sigma_2^2 \cdot \mathbf{v}_2 + \ldots + c_d^{(i-1)}\sigma_d^2 \cdot \mathbf{v}_d\right]$$

Interpretation: The coefficient of $\mathbf{v}_1$ is increasing faster than the coefficents of the other singular vectors.

One Step Analysis of Power Method

Claim: After update $\mathbf{z}^{(i)} = \frac{1}{n_i}\mathbf{X}^T\mathbf{X}\mathbf{z}^{(i-1)}$, where

$$\mathbf{z}^{(i)} = \frac{1}{n_i}\left[c_1^{(i-1)}\sigma_1^2 \cdot \mathbf{v}_1 + c_2^{(i-1)}\sigma_2^2 \cdot \mathbf{v}_2 + \ldots + c_d^{(i-1)}\sigma_d^2 \cdot \mathbf{v}_d\right]$$

Interpretation: The coefficient of $\mathbf{v}_1$ is increasing faster than the coefficents of the other singular vectors.

Proof: Recall $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ and so

$$\mathbf{X}^T\mathbf{X}\mathbf{z}^{(i-1)} = \mathbf{V}\mathbf{\Sigma}\mathbf{U}^T\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\mathbf{z}^{(i-1)} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T\mathbf{z}^{(i-1)}$$

One Step Analysis of Power Method

Claim: After update $\mathbf{z}^{(i)} = \frac{1}{n_i}\mathbf{X}^T\mathbf{X}\mathbf{z}^{(i-1)}$, where

$$\mathbf{z}^{(i)} = \frac{1}{n_i}\left[c_1^{(i-1)}\sigma_1^2 \cdot \mathbf{v}_1 + c_2^{(i-1)}\sigma_2^2 \cdot \mathbf{v}_2 + \ldots + c_d^{(i-1)}\sigma_d^2 \cdot \mathbf{v}_d\right]$$

Interpretation: The coefficient of $\mathbf{v}_1$ is increasing faster than the coefficents of the other singular vectors.

Proof: Recall $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ and so

$$\mathbf{X}^T\mathbf{X}\mathbf{z}^{(i-1)} = \mathbf{V}\mathbf{\Sigma}\mathbf{U}^T\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\mathbf{z}^{(i-1)} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T\mathbf{z}^{(i-1)}$$

$$= \mathbf{V}\mathbf{\Sigma}^2\mathbf{c}^{(i-1)} = \mathbf{V}\begin{pmatrix}\sigma_1^2 c_1^{(i-1)} \\ \\ \end{pmatrix}$$

One Step Analysis of Power Method

Claim: After update $\mathbf{z}^{(i)} = \frac{1}{n_i}\mathbf{X}^T\mathbf{X}\mathbf{z}^{(i-1)}$, where

$$\mathbf{z}^{(i)} = \frac{1}{n_i}\left[c_1^{(i-1)}\sigma_1^2 \cdot \mathbf{v}_1 + c_2^{(i-1)}\sigma_2^2 \cdot \mathbf{v}_2 + \ldots + c_d^{(i-1)}\sigma_d^2 \cdot \mathbf{v}_d\right]$$

Interpretation: The coefficient of $\mathbf{v}_1$ is increasing faster than the coefficents of the other singular vectors.

Proof: Recall $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ and so

$$\mathbf{X}^T\mathbf{X}\mathbf{z}^{(i-1)} = \mathbf{V}\mathbf{\Sigma}\mathbf{U}^T\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\mathbf{z}^{(i-1)} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T\mathbf{z}^{(i-1)}$$

$$= \mathbf{V}\mathbf{\Sigma}^2\mathbf{c}^{(i-1)} = \mathbf{V}\begin{pmatrix} \sigma_1^2 c_1^{(i-1)} \\ \vdots \\ \end{pmatrix}$$

One Step Analysis of Power Method

Claim: After update $\mathbf{z}^{(i)} = \frac{1}{n_i}\mathbf{X}^T\mathbf{X}\mathbf{z}^{(i-1)}$, where

$$\mathbf{z}^{(i)} = \frac{1}{n_i}\left[c_1^{(i-1)}\sigma_1^2 \cdot \mathbf{v}_1 + c_2^{(i-1)}\sigma_2^2 \cdot \mathbf{v}_2 + \ldots + c_d^{(i-1)}\sigma_d^2 \cdot \mathbf{v}_d\right]$$

Interpretation: The coefficient of $\mathbf{v}_1$ is increasing faster than the coefficents of the other singular vectors.

Proof: Recall $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ and so

$$\mathbf{X}^T\mathbf{X}\mathbf{z}^{(i-1)} = \mathbf{V}\mathbf{\Sigma}\mathbf{U}^T\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\mathbf{z}^{(i-1)} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T\mathbf{z}^{(i-1)}$$

$$= \mathbf{V}\mathbf{\Sigma}^2\mathbf{c}^{(i-1)} = \mathbf{V}\begin{pmatrix}\sigma_1^2 c_1^{(i-1)} \\ \vdots \\ \sigma_d^2 c_d^{(i-1)}\end{pmatrix}$$

One Step Analysis of Power Method

Claim: After update $\mathbf{z}^{(i)} = \frac{1}{n_i}\mathbf{X}^T\mathbf{X}\mathbf{z}^{(i-1)}$, where

$$\mathbf{z}^{(i)} = \frac{1}{n_i}\left[c_1^{(i-1)}\sigma_1^2 \cdot \mathbf{v}_1 + c_2^{(i-1)}\sigma_2^2 \cdot \mathbf{v}_2 + \ldots + c_d^{(i-1)}\sigma_d^2 \cdot \mathbf{v}_d\right]$$

Interpretation: The coefficient of $\mathbf{v}_1$ is increasing faster than the coefficents of the other singular vectors.

Proof: Recall $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ and so

$$\begin{aligned}\mathbf{X}^T\mathbf{X}\mathbf{z}^{(i-1)} &= \mathbf{V}\mathbf{\Sigma}\mathbf{U}^T\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\mathbf{z}^{(i-1)} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T\mathbf{z}^{(i-1)} \\ &= \mathbf{V}\mathbf{\Sigma}^2\mathbf{c}^{(i-1)} = \mathbf{V}\begin{pmatrix}\sigma_1^2 c_1^{(i-1)} \\ \vdots \\ \sigma_d^2 c_d^{(i-1)}\end{pmatrix} \\ &= \sigma_1^2 c_1^{(i-1)}\mathbf{v}_1 + \sigma_2^2 c_2^{(i-1)}\mathbf{v}_2 + \ldots + \sigma_d^2 c_d^{(i-1)}\mathbf{v}_d\end{aligned}$$

Multi-Step Analysis of Power Method

Claim: After T updates:

$$\mathbf{z}^{(T)} = \frac{1}{\prod_{i=1}^{T} n_i} \left[c_1^{(0)} \sigma_1^{2T} \cdot \mathbf{v}_1 + c_2^{(0)} \sigma_2^{2T} \cdot \mathbf{v}_2 + \ldots + c_d^{(0)} \sigma_d^{2T} \cdot \mathbf{v}_d \right]$$

Multi-Step Analysis of Power Method

Claim: After T updates:

$$\mathbf{z}^{(T)} = \frac{1}{\prod_{i=1}^{T} n_i} \left[c_1^{(0)} \sigma_1^{2T} \cdot \mathbf{v}_1 + c_2^{(0)} \sigma_2^{2T} \cdot \mathbf{v}_2 + \ldots + c_d^{(0)} \sigma_d^{2T} \cdot \mathbf{v}_d \right]$$

Proof: We initialize to some random Gaussian vector $\mathbf{z}^{(0)}$ and write it in the $\mathbf{V}$ basis as follows:

$$\mathbf{z}^{(0)} = c_1^{(0)} \mathbf{v}_1 + c_2^{(0)} \mathbf{v}_2 + \ldots + c_d^{(0)} \mathbf{v}_d$$

Multi-Step Analysis of Power Method

Claim: After T updates:

$$\mathbf{z}^{(T)} = \frac{1}{\prod_{i=1}^{T} n_i} \left[c_1^{(0)} \sigma_1^{2T} \cdot \mathbf{v}_1 + c_2^{(0)} \sigma_2^{2T} \cdot \mathbf{v}_2 + \ldots + c_d^{(0)} \sigma_d^{2T} \cdot \mathbf{v}_d \right]$$

Proof: We initialize to some random Gaussian vector $\mathbf{z}^{(0)}$ and write it in the $\mathbf{V}$ basis as follows:

$$\mathbf{z}^{(0)} = c_1^{(0)} \mathbf{v}_1 + c_2^{(0)} \mathbf{v}_2 + \ldots + c_d^{(0)} \mathbf{v}_d$$

Recall, the one step analysis says:

$$\mathbf{z}^{(i)} = \frac{1}{n_i} \left[c_1^{(i-1)} \sigma_1^2 \cdot \mathbf{v}_1 + c_2^{(i-1)} \sigma_2^2 \cdot \mathbf{v}_2 + \ldots + c_d^{(i-1)} \sigma_d^2 \cdot \mathbf{v}_d \right]$$

Multi-Step Analysis of Power Method

Claim: After T updates:

$$\mathbf{z}^{(T)} = \frac{1}{\prod_{i=1}^{T} n_i} \left[c_1^{(0)} \sigma_1^{2T} \cdot \mathbf{v}_1 + c_2^{(0)} \sigma_2^{2T} \cdot \mathbf{v}_2 + \ldots + c_d^{(0)} \sigma_d^{2T} \cdot \mathbf{v}_d \right]$$

Proof: We initialize to some random Gaussian vector $\mathbf{z}^{(0)}$ and write it in the $\mathbf{V}$ basis as follows:

$$\mathbf{z}^{(0)} = c_1^{(0)} \mathbf{v}_1 + c_2^{(0)} \mathbf{v}_2 + \ldots + c_d^{(0)} \mathbf{v}_d$$

Recall, the one step analysis says:

$$\mathbf{z}^{(i)} = \frac{1}{n_i} \left[c_1^{(i-1)} \sigma_1^2 \cdot \mathbf{v}_1 + c_2^{(i-1)} \sigma_2^2 \cdot \mathbf{v}_2 + \ldots + c_d^{(i-1)} \sigma_d^2 \cdot \mathbf{v}_d \right]$$

Therefore,

$$\mathbf{z}^{(1)} = \frac{1}{n_1} \left(c_1^{(0)} \sigma_1^2 \mathbf{v}_1 + c_2^{(0)} \sigma_2^2 \mathbf{v}_2 + \ldots + c_d^{(0)} \sigma_d^2 \mathbf{v}_d \right)$$

Multi-Step Analysis of Power Method

Claim: After T updates:

$$\mathbf{z}^{(T)} = \frac{1}{\prod_{i=1}^{T} n_i}\left[c_1^{(0)}\sigma_1^{2T}\cdot\mathbf{v}_1 + c_2^{(0)}\sigma_2^{2T}\cdot\mathbf{v}_2 + \ldots + c_d^{(0)}\sigma_d^{2T}\cdot\mathbf{v}_d\right]$$

Proof: We initialize to some random Gaussian vector $\mathbf{z}^{(0)}$ and write it in the $\mathbf{V}$ basis as follows:

$$\mathbf{z}^{(0)} = c_1^{(0)}\mathbf{v}_1 + c_2^{(0)}\mathbf{v}_2 + \ldots + c_d^{(0)}\mathbf{v}_d$$

Recall, the one step analysis says:

$$\mathbf{z}^{(i)} = \frac{1}{n_i}\left[c_1^{(i-1)}\sigma_1^{2}\cdot\mathbf{v}_1 + c_2^{(i-1)}\sigma_2^{2}\cdot\mathbf{v}_2 + \ldots + c_d^{(i-1)}\sigma_d^{2}\cdot\mathbf{v}_d\right]$$

Therefore,

$$\mathbf{z}^{(1)} = \frac{1}{n_1}\left(c_1^{(0)}\sigma_1^{2}\mathbf{v}_1 + c_2^{(0)}\sigma_2^{2}\mathbf{v}_2 + \ldots + c_d^{(0)}\sigma_d^{2}\mathbf{v}_d\right)$$

$$\mathbf{z}^{(2)} = \frac{1}{n_1 n_2}\left(c_1^{(0)}\sigma_1^{4}\mathbf{v}_1 + c_2^{(0)}\sigma_2^{4}\mathbf{v}_2 + \ldots + c_d^{(0)}\sigma_d^{4}\mathbf{v}_d\right)$$

Multi-Step Analysis of Power Method

Claim: After T updates:

$$\mathbf{z}^{(T)} = \frac{1}{\prod_{i=1}^{T} n_i} \left[c_1^{(0)} \sigma_1^{2T} \cdot \mathbf{v}_1 + c_2^{(0)} \sigma_2^{2T} \cdot \mathbf{v}_2 + \ldots + c_d^{(0)} \sigma_d^{2T} \cdot \mathbf{v}_d \right]$$

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Goal: Show that $\alpha_j \ll \alpha_1$ for all $j \neq 1$.

Power Method Formal Convergence

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Claim: If $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$ then we will have that $\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2^2 \leq \epsilon$.

Power Method Formal Convergence

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Claim: If $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$ then we will have that $\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2^2 \leq \epsilon$.

Proof:

- Since $\mathbf{z}^{(T)}$ is a unit vector, $\sum_{j=1}^{d} \alpha_j^2 = 1$.

Power Method Formal Convergence

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Claim: If $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$ then we will have that $\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2^2 \leq \epsilon$.

Proof:

- Since $\mathbf{z}^{(T)}$ is a unit vector, $\sum_{j=1}^{d} \alpha_j^2 = 1$.
- Then,

$$1 = \alpha_1^2 + \sum_{j=2}^{d} \alpha_j^2 \leq \alpha_1^2 + d \cdot \alpha_1^2(\epsilon/d) \leq \alpha_1^2 + \frac{\epsilon}{2}$$

Power Method Formal Convergence

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Claim: If $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$ then we will have that $\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2^2 \leq \epsilon$.

Proof:

- Since $\mathbf{z}^{(T)}$ is a unit vector, $\sum_{j=1}^{d} \alpha_j^2 = 1$.
- Then,

$$1 = \alpha_1^2 + \sum_{j=2}^{d} \alpha_j^2 \leq \alpha_1^2 + d \cdot \alpha_1^2(\epsilon/d) \leq \alpha_1^2 + \frac{\epsilon}{2}$$

$$\implies \alpha_1^2 \geq 1 - \frac{\epsilon}{2} \quad \implies \quad |\alpha_1| \geq 1 - \frac{\epsilon}{2}$$

Power Method Formal Convergence

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Claim: If $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$ then we will have that $\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2^2 \leq \epsilon$.

Proof:

- Since $\mathbf{z}^{(T)}$ is a unit vector, $\sum_{j=1}^{d} \alpha_j^2 = 1$.
- Then,

$$1 = \alpha_1^2 + \sum_{j=2}^{d} \alpha_j^2 \leq \alpha_1^2 + d \cdot \alpha_1^2(\epsilon/d) \leq \alpha_1^2 + \frac{\epsilon}{2}$$

$$\implies \alpha_1^2 \geq 1 - \frac{\epsilon}{2} \quad \implies \quad |\alpha_1| \geq 1 - \frac{\epsilon}{2}$$

Power Method Formal Convergence

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Claim: If $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$ then we will have that $\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2^2 \leq \epsilon$.

Proof:

- Since $\mathbf{z}^{(T)}$ is a unit vector, $\sum_{j=1}^{d} \alpha_j^2 = 1$.
- Then,

$$1 = \alpha_1^2 + \sum_{j=2}^{d} \alpha_j^2 \leq \alpha_1^2 + d \cdot \alpha_1^2(\epsilon/d) \leq \alpha_1^2 + \frac{\epsilon}{2}$$

- Finally,

$$\implies \alpha_1^2 \geq 1 - \frac{\epsilon}{2} \quad \implies \quad |\alpha_1| \geq 1 - \frac{\epsilon}{2}$$

$$\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2^2 = \|\mathbf{v}_1\|_2^2 + \|\mathbf{z}^{(T)}\|_2^2 - 2\langle \mathbf{v}_1, \mathbf{z}^{(T)} \rangle$$

Power Method Formal Convergence

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Claim: If $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$ then we will have that $\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2^2 \leq \epsilon$.

Proof:

- Since $\mathbf{z}^{(T)}$ is a unit vector, $\sum_{j=1}^{d} \alpha_j^2 = 1$.
- Then,

$$1 = \alpha_1^2 + \sum_{j=2}^{d} \alpha_j^2 \leq \alpha_1^2 + d \cdot \alpha_1^2(\epsilon/d) \leq \alpha_1^2 + \frac{\epsilon}{2}$$

- Finally,

$$\implies \alpha_1^2 \geq 1 - \frac{\epsilon}{2} \quad \implies \quad |\alpha_1| \geq 1 - \frac{\epsilon}{2}$$

$$\begin{aligned} \|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2^2 &= \|\mathbf{v}_1\|_2^2 + \|\mathbf{z}^{(T)}\|_2^2 - 2\langle \mathbf{v}_1, \mathbf{z}^{(T)} \rangle \\ &= 2 - 2\alpha_1 \leq 2 - 2 + \epsilon \end{aligned}$$

Power Method Formal Convergence

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Claim: Assume the starting coefficient on the first eigenvector is not too small: $|c_1^{(0)}| \geq 1/\sqrt{d}$. Then, for $T = O(\log(d/\epsilon)/\gamma)$, $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$.

Power Method Formal Convergence

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Claim: Assume the starting coefficient on the first eigenvector is not too small: $|c_1^{(0)}| \geq 1/\sqrt{d}$. Then, for $T = O(\log(d/\epsilon)/\gamma)$, $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$.

Proof:

- Recall, $\gamma = \frac{\sigma_1 - \sigma_2}{\sigma_1}$.

Power Method Formal Convergence

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Claim: Assume the starting coefficient on the first eigenvector is not too small: $|c_1^{(0)}| \geq 1/\sqrt{d}$. Then, for $T = O(\log(d/\epsilon)/\gamma)$, $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$.

Proof:

- Recall, $\gamma = \frac{\sigma_1 - \sigma_2}{\sigma_1}$.
- Then,

$$\frac{|\alpha_j|}{|\alpha_1|} = \frac{\sigma_j^{2T}}{\sigma_1^{2T}} \cdot \frac{|c_j^{(0)}|}{|c_1^{(0)}|} \leq \frac{\sigma_2^{2T}}{\sigma_1^{2T}} \cdot \frac{|c_j^{(0)}|}{|c_1^{(0)}|} \leq \frac{\sigma_2^{2T}}{\sigma_1^{2T}} \cdot \sqrt{d} = (1-\gamma)^{2T} \sqrt{d}$$

Power Method Formal Convergence

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Claim: Assume the starting coefficient on the first eigenvector is not too small: $|c_1^{(0)}| \geq 1/\sqrt{d}$. Then, for $T = O(\log(d/\epsilon)/\gamma)$, $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$.

Proof:

- Recall, $\gamma = \frac{\sigma_1 - \sigma_2}{\sigma_1}$.
- Then,

$$\frac{|\alpha_j|}{|\alpha_1|} = \frac{\sigma_j^{2T}}{\sigma_1^{2T}} \cdot \frac{|c_j^{(0)}|}{|c_1^{(0)}|} \leq \frac{\sigma_2^{2T}}{\sigma_1^{2T}} \cdot \frac{|c_j^{(0)}|}{|c_1^{(0)}|} \leq \frac{\sigma_2^{2T}}{\sigma_1^{2T}} \cdot \sqrt{d} = (1-\gamma)^{2T} \sqrt{d}$$

Power Method Formal Convergence

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Claim: Assume the starting coefficient on the first eigenvector is not too small: $|c_1^{(0)}| \geq 1/\sqrt{d}$. Then, for $T = O(\log(d/\epsilon)/\gamma)$, $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$.

Proof:

- Recall, $\gamma = \frac{\sigma_1 - \sigma_2}{\sigma_1}$.
- Then,

$$\frac{|\alpha_j|}{|\alpha_1|} = \frac{\sigma_j^{2T}}{\sigma_1^{2T}} \cdot \frac{|c_j^{(0)}|}{|c_1^{(0)}|} \leq \frac{\sigma_2^{2T}}{\sigma_1^{2T}} \cdot \frac{|c_j^{(0)}|}{|c_1^{(0)}|} \leq \frac{\sigma_2^{2T}}{\sigma_1^{2T}} \cdot \sqrt{d} = (1-\gamma)^{2T} \sqrt{d}$$

- Finally,

$$(1-\gamma)^{2T} \sqrt{d} = \left((1-\gamma)^{\frac{1}{\gamma}}\right)^{2\gamma T} \sqrt{d} = (1/e)^{2\gamma T} \sqrt{d}$$

Power Method Formal Convergence

Let the j-th coefficent be $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$.

Claim: Assume the starting coefficient on the first eigenvector is not too small: $|c_1^{(0)}| \geq 1/\sqrt{d}$. Then, for $T = O(\log(d/\epsilon)/\gamma)$, $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$.

Proof:

- Recall, $\gamma = \frac{\sigma_1 - \sigma_2}{\sigma_1}$.
- Then,

$$\frac{|\alpha_j|}{|\alpha_1|} = \frac{\sigma_j^{2T}}{\sigma_1^{2T}} \cdot \frac{|c_j^{(0)}|}{|c_1^{(0)}|} \leq \frac{\sigma_2^{2T}}{\sigma_1^{2T}} \cdot \frac{|c_j^{(0)}|}{|c_1^{(0)}|} \leq \frac{\sigma_2^{2T}}{\sigma_1^{2T}} \cdot \sqrt{d} = (1-\gamma)^{2T} \sqrt{d}$$

- Finally,

$$(1-\gamma)^{2T} \sqrt{d} = \left((1-\gamma)^{\frac{1}{\gamma}}\right)^{2\gamma T} \sqrt{d} = (1/e)^{2\gamma T} \sqrt{d}$$
$$\leq \left(\frac{\epsilon}{d}\right)^4 \sqrt{d} \ll \sqrt{\frac{\epsilon}{2d}}$$

Starting Coefficient Analysis

Need to prove: Starting coefficient on first eigenvector is not too small. I.e., with probability 99/100, $\left|c_1^{(0)}\right| \geq \Omega\left(\frac{1}{\sqrt{d}}\right)$.

Starting Coefficient Analysis

Need to prove: Starting coefficient on first eigenvector is not too small. I.e., with probability 99/100, $\left|c_1^{(0)}\right| \geq \Omega\left(\frac{1}{\sqrt{d}}\right)$.

Proof:

- Recall, the starting vector $\mathbf{z}^{(0)} \sim \mathcal{N}(0, I)$. Therefore

Starting Coefficient Analysis

Need to prove: Starting coefficient on first eigenvector is not too small. I.e., with probability 99/100, $\left|c_1^{(0)}\right| \geq \Omega\left(\frac{1}{\sqrt{d}}\right)$.

Proof:

- Recall, the starting vector $\mathbf{z}^{(0)} \sim \mathcal{N}(0, I)$. Therefore

$$\mathbf{c}^{(0)} = \frac{\mathbf{V}^T\mathbf{z}^{(0)}}{\|\mathbf{V}^T\mathbf{z}^{(0)}\|_2} = \frac{\mathbf{V}^T g}{\|\mathbf{V}^T g\|_2} = \frac{\mathbf{g}'}{\|\mathbf{g}'\|_2},$$

  where $\mathbf{g}' \sim \mathcal{N}(0, I)$. WHY?

Starting Coefficient Analysis

Need to prove: Starting coefficient on first eigenvector is not too small. I.e., with probability 99/100, $\left|c_1^{(0)}\right| \geq \Omega\left(\frac{1}{\sqrt{d}}\right)$.

Proof:

- Recall, the starting vector $\mathbf{z}^{(0)} \sim \mathcal{N}(0, I)$. Therefore

$$\mathbf{c}^{(0)} = \frac{\mathbf{V}^T\mathbf{z}^{(0)}}{\|\mathbf{V}^T\mathbf{z}^{(0)}\|_2} = \frac{\mathbf{V}^T g}{\|\mathbf{V}^T g\|_2} = \frac{\mathbf{g}'}{\|\mathbf{g}'\|_2},$$

  where $\mathbf{g}' \sim \mathcal{N}(0, I)$. WHY?

- Rotational invariance of Gaussians

Starting Coefficient Analysis

Need to prove: Starting coefficient on first eigenvector is not too small. I.e., with probability 99/100, $\left|c_1^{(0)}\right| \geq \Omega\left(\frac{1}{\sqrt{d}}\right)$.

Proof:

- Recall, the starting vector $\mathbf{z}^{(0)} \sim \mathcal{N}(0, I)$. Therefore

$$\mathbf{c}^{(0)} = \frac{\mathbf{V}^T \mathbf{z}^{(0)}}{\|\mathbf{V}^T \mathbf{z}^{(0)}\|_2} = \frac{\mathbf{V}^T g}{\|\mathbf{V}^T g\|_2} = \frac{\mathbf{g}'}{\|\mathbf{g}'\|_2},$$

  where $\mathbf{g}' \sim \mathcal{N}(0, I)$. WHY?
- Rotational invariance of Gaussians
- Therefore, it suffices to show that for $\mathbf{g}' \sim \mathcal{N}(0, I)$, the first coordinate of $\frac{\mathbf{g}'}{\|\mathbf{g}'\|_2}$ is $\Omega(1/\sqrt{d})$.

Starting Coefficient Analysis

Claim: With probability 0.99, first coordinate of $\frac{\mathbf{g}}{\|\mathbf{g}\|_2}$ is at least $c \cdot \frac{1}{\sqrt{d}}$.

Starting Coefficient Analysis

Claim: With probability 0.99, first coordinate of $\frac{\mathbf{g}}{\|\mathbf{g}\|_2}$ is at least $c \cdot \frac{1}{\sqrt{d}}$.

Part 1: With super high probability, $\|\mathbf{g}\|_2^2 \leq 2d$.

Starting Coefficient Analysis

Claim: With probability 0.99, first coordinate of $\frac{\mathbf{g}}{\|\mathbf{g}\|_2}$ is at least $c \cdot \frac{1}{\sqrt{d}}$.

Part 1: With super high probability, $\|\mathbf{g}\|_2^2 \leq 2d$.

Proof:

$$\mathbb{E}[\|g\|_2^2] = \mathbb{E}[\sum_{i=1}^{d} g_i^2] = \sum_{i=1}^{d} \mathbb{E}[g_i^2] = d$$

Starting Coefficient Analysis

Claim: With probability 0.99, first coordinate of $\frac{\mathbf{g}}{\|\mathbf{g}\|_2}$ is at least $c \cdot \frac{1}{\sqrt{d}}$.

Part 1: With super high probability, $\|\mathbf{g}\|_2^2 \leq 2d$.

Proof:

$$\mathbb{E}[\|g\|_2^2] = \mathbb{E}[\sum_{i=1}^{d} g_i^2] = \sum_{i=1}^{d} \mathbb{E}[g_i^2] = d$$

Starting Coefficient Analysis

Claim: With probability 0.99, first coordinate of $\frac{\mathbf{g}}{\|\mathbf{g}\|_2}$ is at least $c \cdot \frac{1}{\sqrt{d}}$.

Part 1: With super high probability, $\|\mathbf{g}\|_2^2 \leq 2d$.

Proof:

$$\mathbb{E}[\|g\|_2^2] = \mathbb{E}[\sum_{i=1}^{d} g_i^2] = \sum_{i=1}^{d} \mathbb{E}[g_i^2] = d$$

Chi-Squared Concentration: If $X \sim \chi_d^2$, then for every $\delta > 0$,

$$\Pr[X \geq (1+\delta)d] \leq \exp\left(-\frac{d}{2}(\delta - \ln(1+\delta))\right).$$

Starting Coefficient Analysis

Claim: With probability 0.99, first coordinate of $\frac{\mathbf{g}}{\|\mathbf{g}\|_2}$ is at least $c \cdot \frac{1}{\sqrt{d}}$.

Part 1: With super high probability, $\|\mathbf{g}\|_2^2 \leq 2d$.

Proof:

$$\mathbb{E}[\|g\|_2^2] = \mathbb{E}[\sum_{i=1}^{d} g_i^2] = \sum_{i=1}^{d} \mathbb{E}[g_i^2] = d$$

Chi-Squared Concentration: If $X \sim \chi_d^2$, then for every $\delta > 0$,

$$\Pr[X \geq (1+\delta)d] \leq \exp\left(-\frac{d}{2}(\delta - \ln(1+\delta))\right).$$

Application to Gaussian norm: If $g \sim \mathcal{N}(0, I_d)$, then $\|g\|_2^2 \sim \chi_d^2$.
Setting $\delta = 1$ gives

$$\Pr\left(\|g\|_2^2 \geq 2d\right) \leq \exp\left(-\frac{1-\ln 2}{2}\, d\right) \leq \exp\left(-d\right)$$

Starting Coefficient Analysis

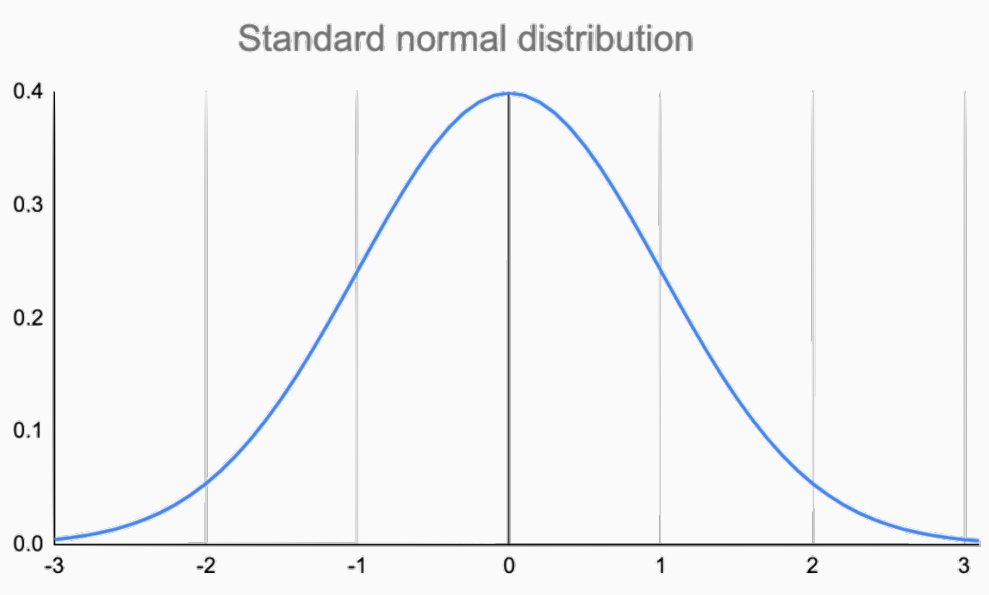
Claim: With probability 0.99, first coordinate of $\frac{\mathbf{g}}{\|\mathbf{g}\|_2}$ is at least $c \cdot \frac{1}{\sqrt{d}}$.

Part 2: With probability $1 - O(\alpha)$, $|g_1| \geq \alpha$.

Starting Coefficient Analysis

Claim: With probability 0.99, first coordinate of $\frac{\mathbf{g}}{\|\mathbf{g}\|_2}$ is at least $c \cdot \frac{1}{\sqrt{d}}$.

Part 2: With probability $1 - O(\alpha)$, $|g_1| \geq \alpha$.



Starting Coefficient Analysis

Claim: With probability 0.99, first coordinate of $\frac{\mathbf{g}}{\|\mathbf{g}\|_2}$ is at least $c \cdot \frac{1}{\sqrt{d}}$.

Part 2: With probability $1 - O(\alpha)$, $|g_1| \geq \alpha$.

Proof:

- Recall the PDF of a Gaussian random variable $x \sim \mathcal{N}(0, 1)$ is

$$p(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \leq \frac{1}{2\pi} \leq 0.4$$

- To calculate the probability that $|x| \leq \alpha$, we just integrate the PDF

$$\Pr[|x| \leq \alpha] = \int_{-\alpha}^{\alpha} p(x) \leq \int_{-\alpha}^{\alpha} 0.4 = 0.8\alpha$$

Putting It Together

- **One-step analysis:** Each update scales the j-th coefficient by σ_j^2:

$$\mathbf{z}^{(i)} \propto c_1^{(0)} \sigma_1^{2i} \cdot \mathbf{v}_1 + c_2^{(0)} \sigma_2^{2i} \cdot \mathbf{v}_2 + \ldots + c_d^{(0)} \sigma_d^{2i} \cdot \mathbf{v}_d.$$

Putting It Together

- **One-step analysis:** Each update scales the j-th coefficient by σ_j^2:

$$\mathbf{z}^{(i)} \propto c_1^{(0)} \sigma_1^{2i} \cdot \mathbf{v}_1 + c_2^{(0)} \sigma_2^{2i} \cdot \mathbf{v}_2 + \ldots + c_d^{(0)} \sigma_d^{2i} \cdot \mathbf{v}_d.$$

- **Starting coefficient:** With high probability, $|c_1^{(0)}| \geq \Omega\left(\frac{1}{\sqrt{d}}\right)$ for a random Gaussian init.

Putting It Together

- **One-step analysis:** Each update scales the j-th coefficient by σ_j^2:

$$\mathbf{z}^{(i)} \propto c_1^{(0)} \sigma_1^{2i} \cdot \mathbf{v}_1 + c_2^{(0)} \sigma_2^{2i} \cdot \mathbf{v}_2 + \ldots + c_d^{(0)} \sigma_d^{2i} \cdot \mathbf{v}_d.$$

- **Starting coefficient:** With high probability, $|c_1^{(0)}| \geq \Omega\left(\frac{1}{\sqrt{d}}\right)$ for a random Gaussian init.
- **Ratio bound:** After $T = O\left(\frac{\log d/\epsilon}{\gamma}\right)$ steps, $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$ for all $j \geq 2$.

Putting It Together

- **One-step analysis:** Each update scales the j-th coefficient by σ_j^2:

$$\mathbf{z}^{(i)} \propto c_1^{(0)} \sigma_1^{2i} \cdot \mathbf{v}_1 + c_2^{(0)} \sigma_2^{2i} \cdot \mathbf{v}_2 + \ldots + c_d^{(0)} \sigma_d^{2i} \cdot \mathbf{v}_d.$$

- **Starting coefficient:** With high probability, $|c_1^{(0)}| \geq \Omega\left(\frac{1}{\sqrt{d}}\right)$ for a random Gaussian init.
- **Ratio bound:** After $T = O\left(\frac{\log d/\epsilon}{\gamma}\right)$ steps, $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$ for all $j \geq 2$.
- **Convergence:** If $\left|\frac{\alpha_j}{\alpha_1}\right| \leq \sqrt{\frac{\epsilon}{2d}}$ for all $j \neq 1$, $\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2 \leq \epsilon$ with high probability.

Power Method Formal Convergence

Theorem (Power Method Convergence, $k = 1$):

- Let $\gamma = \frac{\sigma_1 - \sigma_2}{\sigma_1}$ capture the gap between the first and second largest singular values.
- Initialize with a random Gaussian vector.
- After $T = O\left(\frac{\log d/\epsilon}{\gamma}\right)$ steps, with high probability:

$$\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2 \leq \epsilon \quad \text{or} \quad \|\mathbf{v}_1 + \mathbf{z}^{(T)}\|_2 \leq \epsilon.$$

Power Method Formal Convergence

Theorem (Power Method Convergence, $k = 1$):

- Let $\gamma = \frac{\sigma_1 - \sigma_2}{\sigma_1}$ capture the gap between the first and second largest singular values.
- Initialize with a random Gaussian vector.
- After $T = O\left(\frac{\log d/\epsilon}{\gamma}\right)$ steps, with high probability:

$$\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2 \leq \epsilon \quad \text{or} \quad \|\mathbf{v}_1 + \mathbf{z}^{(T)}\|_2 \leq \epsilon.$$

Note: The method won't converge if γ is very small (e.g., $\gamma = 0$):

$$\mathbf{z}^{(T)} = \frac{1}{\prod_{i=1}^{T} n_i}\left[c_1^{(0)}\sigma_1^{2T} \cdot \mathbf{v}_1 + c_2^{(0)}\sigma_2^{2T} \cdot \mathbf{v}_2 + \ldots + c_d^{(0)}\sigma_d^{2T} \cdot \mathbf{v}_d\right]$$

Power Method for Low-Rank Approximation

Theorem: After $T = O\left(\frac{\log d/\epsilon}{\gamma}\right)$ steps, with high probability:

$$\|\mathbf{X} - \mathbf{X}\mathbf{z}^{(T)}(\mathbf{z}^{(T)})^T\|_F^2 \leq \|\mathbf{X} - \mathbf{X}\mathbf{v}_1\mathbf{v}_1^T\|_F^2 + \epsilon\sigma_1^2.$$

Proof:

- $\|\mathbf{X} - \mathbf{X}\mathbf{z}^{(T)}(\mathbf{z}^{(T)})^T\|_F^2 = \|\mathbf{X}\|_F^2 - \|\mathbf{X}\mathbf{z}^{(T)}\|_2^2$

Power Method for Low-Rank Approximation

Theorem: After $T = O\left(\frac{\log d/\epsilon}{\gamma}\right)$ steps, with high probability:

$$\|\mathbf{X} - \mathbf{X}\mathbf{z}^{(T)}(\mathbf{z}^{(T)})^T\|_F^2 \leq \|\mathbf{X} - \mathbf{X}\mathbf{v}_1\mathbf{v}_1^T\|_F^2 + \epsilon\sigma_1^2.$$

Proof:

- $\|\mathbf{X} - \mathbf{X}\mathbf{z}^{(T)}(\mathbf{z}^{(T)})^T\|_F^2 = \|\mathbf{X}\|_F^2 - \|\mathbf{X}\mathbf{z}^{(T)}\|_2^2$
- Write $\mathbf{z}^{(T)} = \sum_i \zeta_i \mathbf{v}_i$, so $\|\mathbf{X}\mathbf{z}^{(T)}\|_2^2 = \sum_i \sigma_i^2\zeta_i^2 \geq \sigma_1^2\zeta_1^2$

Power Method for Low-Rank Approximation

Theorem: After $T = O\left(\frac{\log d/\epsilon}{\gamma}\right)$ steps, with high probability:

$$\|\mathbf{X} - \mathbf{X}\mathbf{z}^{(T)}(\mathbf{z}^{(T)})^T\|_F^2 \leq \|\mathbf{X} - \mathbf{X}\mathbf{v}_1\mathbf{v}_1^T\|_F^2 + \epsilon\sigma_1^2.$$

Proof:

- $\|\mathbf{X} - \mathbf{X}\mathbf{z}^{(T)}(\mathbf{z}^{(T)})^T\|_F^2 = \|\mathbf{X}\|_F^2 - \|\mathbf{X}\mathbf{z}^{(T)}\|_2^2$
- Write $\mathbf{z}^{(T)} = \sum_i \zeta_i \mathbf{v}_i$, so $\|\mathbf{X}\mathbf{z}^{(T)}\|_2^2 = \sum_i \sigma_i^2\zeta_i^2 \geq \sigma_1^2\zeta_1^2$
- From convergence $\|\mathbf{z}^{(T)} - \mathbf{v}_1\|_2 \leq \epsilon$:

$$\epsilon^2 \geq \|\mathbf{z}^{(T)} - \mathbf{v}_1\|_2^2 = (1-\zeta_1)^2 + \sum_{i=2}^d \zeta_i^2 \geq (1-\zeta_1)^2$$

Power Method for Low-Rank Approximation

Theorem: After $T = O\left(\frac{\log d/\epsilon}{\gamma}\right)$ steps, with high probability:

$$\|\mathbf{X} - \mathbf{X}\mathbf{z}^{(T)}(\mathbf{z}^{(T)})^T\|_F^2 \leq \|\mathbf{X} - \mathbf{X}\mathbf{v}_1\mathbf{v}_1^T\|_F^2 + \epsilon\sigma_1^2.$$

Proof:

- $\|\mathbf{X} - \mathbf{X}\mathbf{z}^{(T)}(\mathbf{z}^{(T)})^T\|_F^2 = \|\mathbf{X}\|_F^2 - \|\mathbf{X}\mathbf{z}^{(T)}\|_2^2$
- Write $\mathbf{z}^{(T)} = \sum_i \zeta_i \mathbf{v}_i$, so $\|\mathbf{X}\mathbf{z}^{(T)}\|_2^2 = \sum_i \sigma_i^2\zeta_i^2 \geq \sigma_1^2\zeta_1^2$
- From convergence $\|\mathbf{z}^{(T)} - \mathbf{v}_1\|_2 \leq \epsilon$:

$$\epsilon^2 \geq \|\mathbf{z}^{(T)} - \mathbf{v}_1\|_2^2 = (1-\zeta_1)^2 + \sum_{i=2}^d \zeta_i^2 \geq (1-\zeta_1)^2$$

- Therefore $\zeta_1 \geq (1-\epsilon)$ and:

$$\|\mathbf{X} - \mathbf{X}\mathbf{z}^{(T)}(\mathbf{z}^{(T)})^T\|_F^2 \leq \|\mathbf{X}\|_F^2 - \sigma_1^2(1-\epsilon)^2 \leq \sum_{i=2}^d \sigma_i^2 + \epsilon\sigma_1^2.$$

Souped-Up Power Method for LRA

Theorem: After $T = O\left(\frac{\log d/\epsilon}{\gamma}\right)$ steps, with high probability:

$$\|\mathbf{X} - \mathbf{X}\mathbf{z}^{(T)}(\mathbf{z}^{(T)})^T\|_F^2 \leq (1+\epsilon)\|\mathbf{X} - \mathbf{X}\mathbf{v}_1\mathbf{v}_1^T\|_F^2.$$

Power Method – No Gap Dependence

Theorem (Gapless Power Method Convergence):

- Initialize with a random Gaussian vector.

Power Method – No Gap Dependence

Theorem (Gapless Power Method Convergence):

- Initialize with a random Gaussian vector.
- After $T = O\left(\frac{\log d/\epsilon}{\epsilon}\right)$ steps, with high probability:

$$\|\mathbf{X} - \mathbf{X}\mathbf{z}\mathbf{z}^T\|_F^2 \leq (1+\epsilon)\|\mathbf{X} - \mathbf{X}\mathbf{v}_1\mathbf{v}_1^T\|_F^2.$$

Power Method – No Gap Dependence

Theorem (Gapless Power Method Convergence):

- Initialize with a random Gaussian vector.
- After $T = O\left(\frac{\log d/\epsilon}{\epsilon}\right)$ steps, with high probability:

$$\|\mathbf{X} - \mathbf{X}\mathbf{z}\mathbf{z}^T\|_F^2 \leq (1+\epsilon)\|\mathbf{X} - \mathbf{X}\mathbf{v}_1\mathbf{v}_1^T\|_F^2.$$

Intuition: For a good low-rank approximation, we don't need to converge exactly to $\mathbf{v}_1$ when $\sigma_1 \approx \sigma_2$. Any linear combination of $\mathbf{v}_1$ and $\mathbf{v}_2$ suffices.

Generalizations to Larger k

Block Power Method (aka Simultaneous / Subspace / Orthogonal Iteration):

- Choose $\mathbf{G} \in \mathbb{R}^{d \times k}$ randomly. Set $\mathbf{Z}^{(0)} = \text{orth}(\mathbf{G})$.

Generalizations to Larger k

Block Power Method (aka Simultaneous / Subspace / Orthogonal Iteration):

- Choose $\mathbf{G} \in \mathbb{R}^{d \times k}$ randomly. Set $\mathbf{Z}^{(0)} = \text{orth}(\mathbf{G})$.
- For $i = 1, \ldots, T$:
  - $\mathbf{Z}^{(i)} = \mathbf{X}^T(\mathbf{X}\mathbf{Z}^{(i-1)})$
  - $\mathbf{Z}^{(i)} = \text{orth}(\mathbf{Z}^{(i)})$
- Return $\mathbf{Z}^{(T)}$

Generalizations to Larger k

Block Power Method (aka Simultaneous / Subspace / Orthogonal Iteration):

- Choose $\mathbf{G} \in \mathbb{R}^{d \times k}$ randomly. Set $\mathbf{Z}^{(0)} = \text{orth}(\mathbf{G})$.
- For $i = 1, \ldots, T$:
  - $\mathbf{Z}^{(i)} = \mathbf{X}^T(\mathbf{X}\mathbf{Z}^{(i-1)})$
  - $\mathbf{Z}^{(i)} = \text{orth}(\mathbf{Z}^{(i)})$
- Return $\mathbf{Z}^{(T)}$

Guarantee: After $T = O\left(\frac{\log d/\epsilon}{\epsilon}\right)$ iterations:

$$\|\mathbf{X} - \mathbf{X}\mathbf{Z}^{(T)}(\mathbf{Z}^{(T)})^T\|_F^2 \leq (1+\epsilon)\|\mathbf{X} - \mathbf{X}\mathbf{V}_k\mathbf{V}_k^T\|_F^2.$$

Runtime: $O(\text{nnz}(\mathbf{X}) \cdot k \cdot T)$.

Krylov Methods

Possible to "accelerate" these methods.

Convergence Guarantee: $T = O\left(\frac{\log d/\epsilon}{\sqrt{\epsilon}}\right)$ iterations to obtain a nearly optimal low-rank approximation:

$$\|\mathbf{X} - \mathbf{X}\mathbf{Z}\mathbf{Z}^T\|_F^2 \leq (1+\epsilon)\|\mathbf{X} - \mathbf{X}\mathbf{V_k}\mathbf{V_k}^T\|_F^2.$$

Krylov Subspace Methods

For a normalizing constant c, power method returns:

$$\mathbf{z}^{(q)} = c \cdot \left(\mathbf{X}^T\mathbf{X}\right)^q \cdot \mathbf{g}$$

Along the way we computed:

$$\mathcal{K}_q = \left[\mathbf{g}, \left(\mathbf{X}^T\mathbf{X}\right) \cdot \mathbf{g}, \left(\mathbf{X}^T\mathbf{X}\right)^2 \cdot \mathbf{g}, \ldots, \left(\mathbf{X}^T\mathbf{X}\right)^q \cdot \mathbf{g}\right]$$

$\mathcal{K}$ is called the Krylov subspace of degree q.

Krylov Subspace Methods

For a normalizing constant c, power method returns:

$$\mathbf{z}^{(q)} = c \cdot \left(\mathbf{X}^T\mathbf{X}\right)^q \cdot \mathbf{g}$$

Along the way we computed:

$$\mathcal{K}_q = \left[\mathbf{g}, \left(\mathbf{X}^T\mathbf{X}\right) \cdot \mathbf{g}, \left(\mathbf{X}^T\mathbf{X}\right)^2 \cdot \mathbf{g}, \ldots, \left(\mathbf{X}^T\mathbf{X}\right)^q \cdot \mathbf{g}\right]$$

$\mathcal{K}$ is called the Krylov subspace of degree q.

Idea behind Krylov methods: Don't throw away everything before $\left(\mathbf{X}^T\mathbf{X}\right)^q \cdot \mathbf{g}$.

Krylov Subspace Methods

Want to find $\mathbf{v}$, which minimizes $\|\mathbf{X} - \mathbf{X}\mathbf{v}\mathbf{v}^T\|_F^2$.

Lanczos method:

- Let $\mathbf{Q} \in \mathbb{R}^{d \times k}$ be an orthonormal span for the vectors in $\mathcal{K}$.
- Solve $\min_{\mathbf{v}=\mathbf{Q}\mathbf{w}} \|\mathbf{X} - \mathbf{X}\mathbf{v}\mathbf{v}^T\|_F^2$.

Krylov Subspace Methods

Want to find $\mathbf{v}$, which minimizes $\|\mathbf{X} - \mathbf{X}\mathbf{v}\mathbf{v}^T\|_F^2$.

Lanczos method:

- Let $\mathbf{Q} \in \mathbb{R}^{d \times k}$ be an orthonormal span for the vectors in $\mathcal{K}$.
- Solve $\min_{\mathbf{v}=\mathbf{Q}\mathbf{w}} \|\mathbf{X} - \mathbf{X}\mathbf{v}\mathbf{v}^T\|_F^2$.
  - Find best vector in the Krylov subspace, instead of just using last vector.

Krylov Subspace Methods

Want to find $\mathbf{v}$, which minimizes $\|\mathbf{X} - \mathbf{X}\mathbf{v}\mathbf{v}^T\|_F^2$.

Lanczos method:

- Let $\mathbf{Q} \in \mathbb{R}^{d \times k}$ be an orthonormal span for the vectors in $\mathcal{K}$.
- Solve $\min_{\mathbf{v}=\mathbf{Q}\mathbf{w}} \|\mathbf{X} - \mathbf{X}\mathbf{v}\mathbf{v}^T\|_F^2$.
  - Find best vector in the Krylov subspace, instead of just using last vector.
  - Can be done in $O\left(ndk + dk^2\right)$ time.

Krylov Subspace Methods

Want to find $\mathbf{v}$, which minimizes $\|\mathbf{X} - \mathbf{X}\mathbf{v}\mathbf{v}^T\|_F^2$.

Lanczos method:

- Let $\mathbf{Q} \in \mathbb{R}^{d \times k}$ be an orthonormal span for the vectors in $\mathcal{K}$.
- Solve $\min_{\mathbf{v}=\mathbf{Q}\mathbf{w}} \|\mathbf{X} - \mathbf{X}\mathbf{v}\mathbf{v}^T\|_F^2$.
  - Find best vector in the Krylov subspace, instead of just using last vector.
  - Can be done in $O\left(ndk + dk^2\right)$ time.
  - What you're using when you run `svds` or `eigs` in MATLAB or Python.

Krylov Subspace Methods

What vectors lie in the Krylov Subspace?

- Recall

$$\mathcal{K}_q = \left[\mathbf{g}, \left(\mathbf{X}^T\mathbf{X}\right)\cdot\mathbf{g}, \left(\mathbf{X}^T\mathbf{X}\right)^2\cdot\mathbf{g}, \ldots, \left(\mathbf{X}^T\mathbf{X}\right)^q\cdot\mathbf{g}\right]$$

- Then any vector in the span can be written as $\mathcal{K}_q z$:

$$\mathcal{K}_q z = \left(z_0 I + z_1\left(\mathbf{X}^T\mathbf{X}\right) + z_2\left(\mathbf{X}^T\mathbf{X}\right)^2 + \ldots + z_q\left(\mathbf{X}^T\mathbf{X}\right)^q\right) g$$

- This is some matrix valued polynomial of degree q multipied by g:

$$\left(\mathbf{X}^T\mathbf{X}\right)^q = \mathbf{V}^T\mathbf{\Sigma}^{2q}\mathbf{V}$$

Lanczos Method Analysis

- For a degree T polynomial p, let $\mathbf{v}_p = \frac{p(\mathbf{X}^T\mathbf{X})\mathbf{g}}{\|p(\mathbf{X}^T\mathbf{X})\mathbf{g}\|_2}$.

Lanczos Method Analysis

- For a degree T polynomial p, let $\mathbf{v}_p = \frac{p(\mathbf{X}^T\mathbf{X})\mathbf{g}}{\|p(\mathbf{X}^T\mathbf{X})\mathbf{g}\|_2}$.
- We always have that $\mathbf{v}_p \in \mathcal{K}_T$, the Krylov subspace constructed with T iterations.

Lanczos Method Analysis

- For a degree T polynomial p, let $\mathbf{v}_p = \frac{p(\mathbf{X}^T\mathbf{X})\mathbf{g}}{\|p(\mathbf{X}^T\mathbf{X})\mathbf{g}\|_2}$.
- We always have that $\mathbf{v}_p \in \mathcal{K}_T$, the Krylov subspace constructed with T iterations.
- Power method computes the polynomial $p(x) = x^T$ and outputs the vector $\mathbf{v}_{x^T}$.

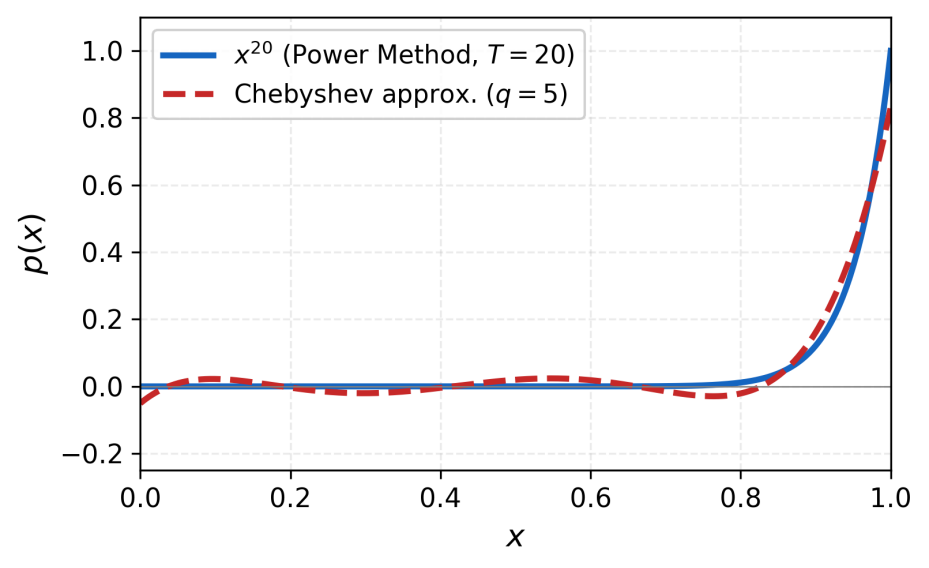
Lanczos Method Analysis

- For a degree T polynomial p, let $\mathbf{v}_p = \frac{p(\mathbf{X}^T\mathbf{X})\mathbf{g}}{\|p(\mathbf{X}^T\mathbf{X})\mathbf{g}\|_2}$.
- We always have that $\mathbf{v}_p \in \mathcal{K}_T$, the Krylov subspace constructed with T iterations.
- Power method computes the polynomial $p(x) = x^T$ and outputs the vector $\mathbf{v}_{x^T}$.
- Lanczos method can compute the best polynomial of degree T, denoted by p^*, and returns $\mathbf{v}_{p^*}$:

$$p^* = \underset{\text{degree } T \ p}{\arg\min} \|\mathbf{X} - \mathbf{X}\mathbf{v}_p\mathbf{v}_p^T\|_F^2.$$

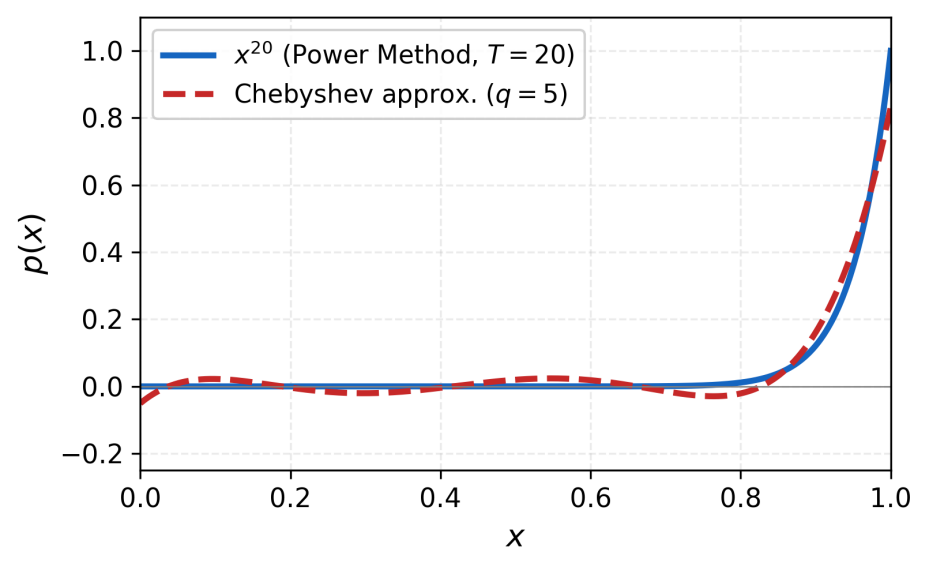
Lanczos Method Analysis

Theorem: There is a $q = O\left(\sqrt{T \log \frac{1}{\Delta}}\right)$ degree polynomial $\hat{p}$ approximating $\mathbf{x}^T$ up to error Δ on $[0, 1]$.



Lanczos Method Analysis

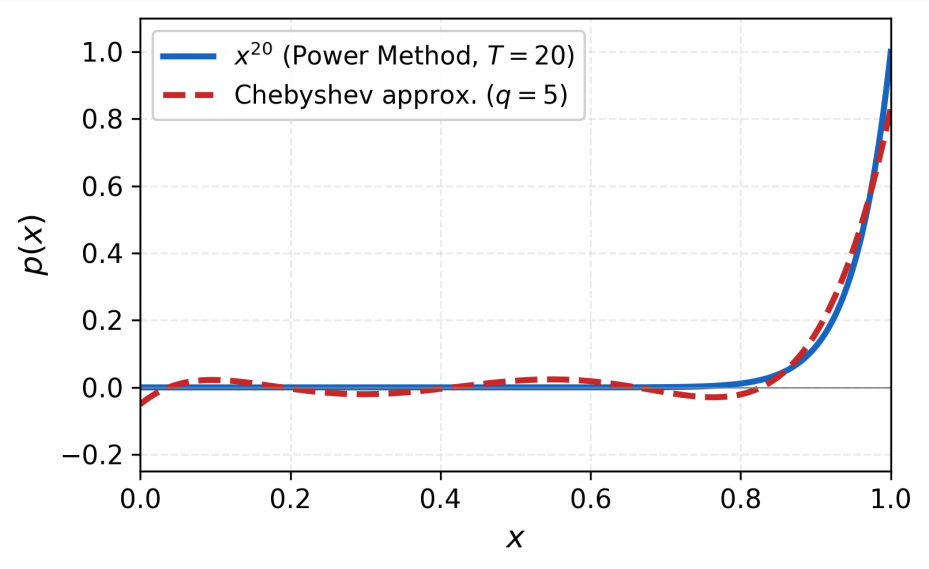
Theorem: There is a $q = O\left(\sqrt{T \log \frac{1}{\Delta}}\right)$ degree polynomial $\hat{p}$ approximating $\mathbf{x}^T$ up to error Δ on $[0, 1]$.



- Lanczos outputs the best polynomial so it is at least as good as the above polynomial
- Reduces the number of iterations from $1/\epsilon$ to $1/\sqrt{\epsilon}$!

Lanczos Method Analysis

Theorem: There is a $q = O\left(\sqrt{T \log \frac{1}{\Delta}}\right)$ degree polynomial $\hat{p}$ approximating $\mathbf{x}^T$ up to error Δ on $[0, 1]$.



- Lanczos outputs the best polynomial so it is at least as good as the above polynomial
- Reduces the number of iterations from $1/\epsilon$ to $1/\sqrt{\epsilon}$!
- Running time: $O\left(\frac{\log(d/\epsilon)}{\sqrt{\epsilon}} \cdot \text{nnz}(\mathbf{X})\right)$

Generalizations to Larger k

- Block Krylov methods
- Let $\mathbf{G} \in \mathbb{R}^{d \times k}$ be a random Gaussian matrix.
- $\mathcal{K}_q = \left[\mathbf{G}, (\mathbf{X}^T\mathbf{X}) \cdot \mathbf{G}, (\mathbf{X}^T\mathbf{X})^2 \cdot \mathbf{G}, \ldots, (\mathbf{X}^T\mathbf{X})^q \cdot \mathbf{G}\right]$

Runtime: $O\left(\text{nnz}(\mathbf{X}) \cdot k \cdot \frac{\log d/\epsilon}{\sqrt{\epsilon}}\right)$ to obtain a $(1+\epsilon)$-approximate low-rank approximation.

Generalizations to Larger k

- Block Krylov methods
- Let $\mathbf{G} \in \mathbb{R}^{d \times k}$ be a random Gaussian matrix.
- $\mathcal{K}_q = \left[\mathbf{G}, (\mathbf{X}^T\mathbf{X}) \cdot \mathbf{G}, (\mathbf{X}^T\mathbf{X})^2 \cdot \mathbf{G}, \ldots, (\mathbf{X}^T\mathbf{X})^q \cdot \mathbf{G}\right]$

Runtime: $O\left(\text{nnz}(\mathbf{X}) \cdot k \cdot \frac{\log d/\epsilon}{\sqrt{\epsilon}}\right)$ to obtain a $(1+\epsilon)$-approximate low-rank approximation.

Theorem [B., Clarkson, Woodruff '22]: Find a vector $\mathbf{z}$ such that

$$\|\mathbf{X} - \mathbf{X}\mathbf{z}\mathbf{z}^T\|_F^2 \leq (1+\epsilon)\|\mathbf{X} - \mathbf{X}\mathbf{v}_1\mathbf{v}_1^T\|_F^2$$

- **Iterations:** $O\left(\frac{\log(d/\epsilon)}{\epsilon^{1/3}}\right)$ and **Runtime:** $O\left(\text{nnz}(\mathbf{X}) \cdot \frac{\log d/\epsilon}{\epsilon^{1/3}}\right)$

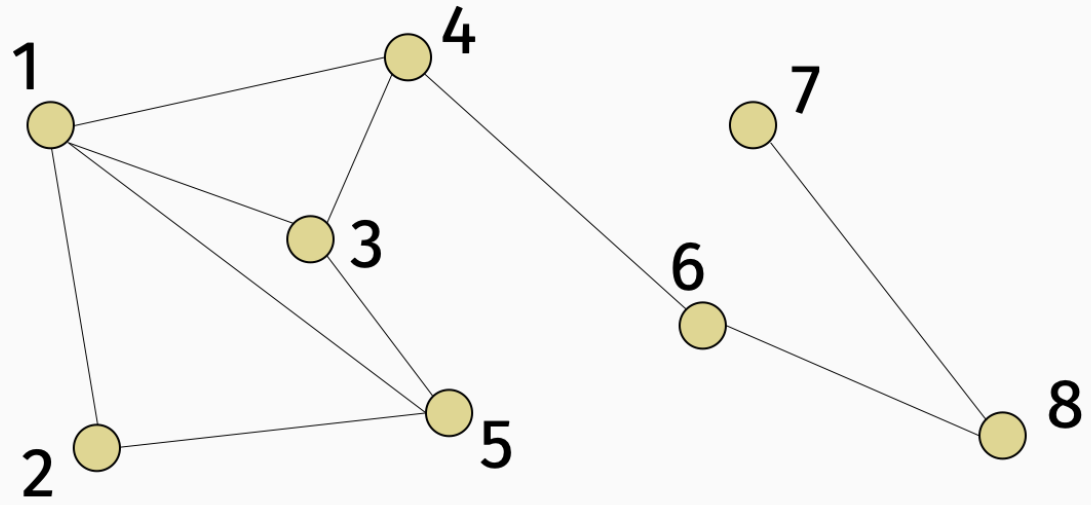
break

Spectral Graph Theory

- **Main idea:** Understand graph data by constructing natural matrix representations, and studying that matrix's spectrum (eigenvalues/eigenvectors).

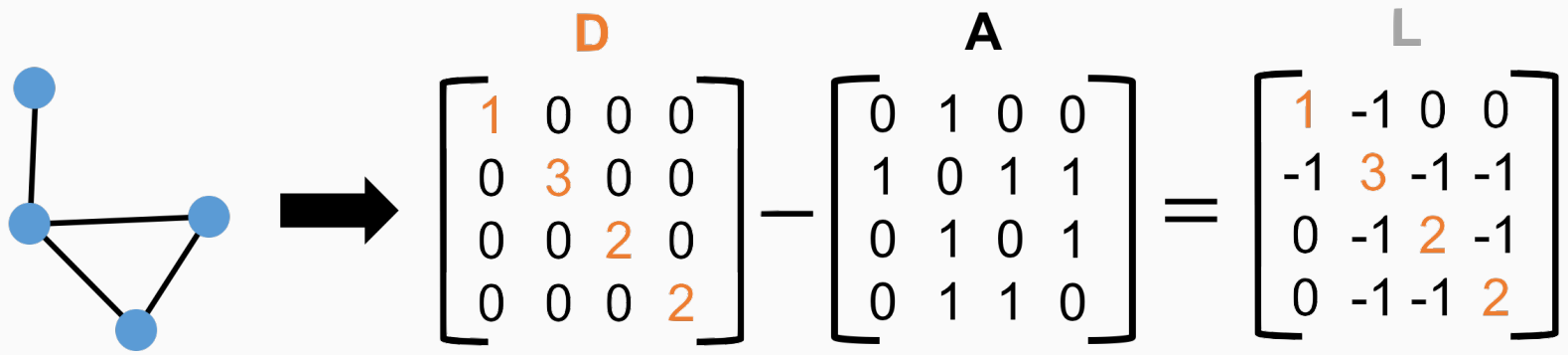
Spectral Graph Theory

- **Main idea:** Understand graph data by constructing natural matrix representations, and studying that matrix's spectrum (eigenvalues/eigenvectors).
- For now assume $G = (V, E)$ is an undirected, unweighted graph with n nodes.



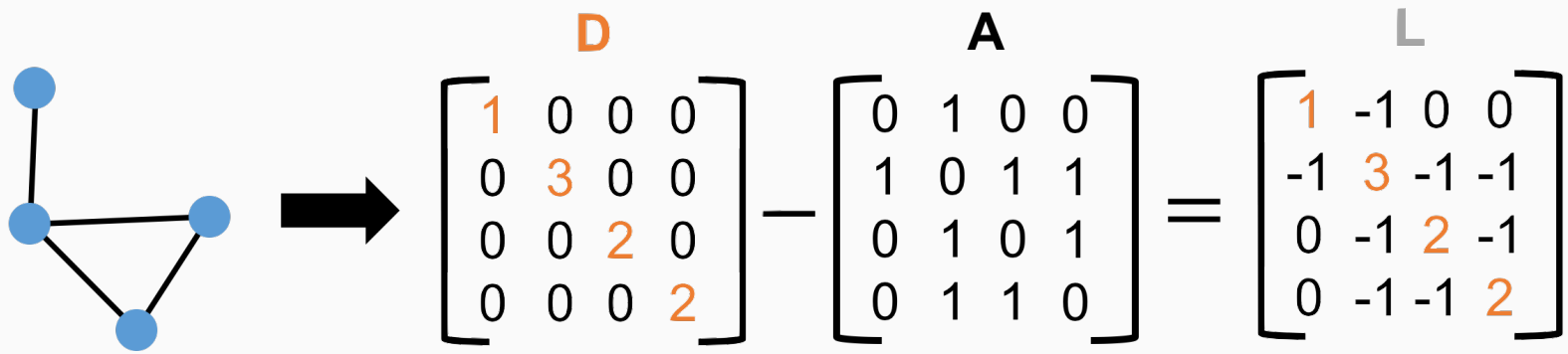
Matrix Representations of Graphs

Two most common representations: $n \times n$ adjacency matrix $\mathbf{A}$ and graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{A}$ where $\mathbf{D}$ is the diagonal degree matrix.



Matrix Representations of Graphs

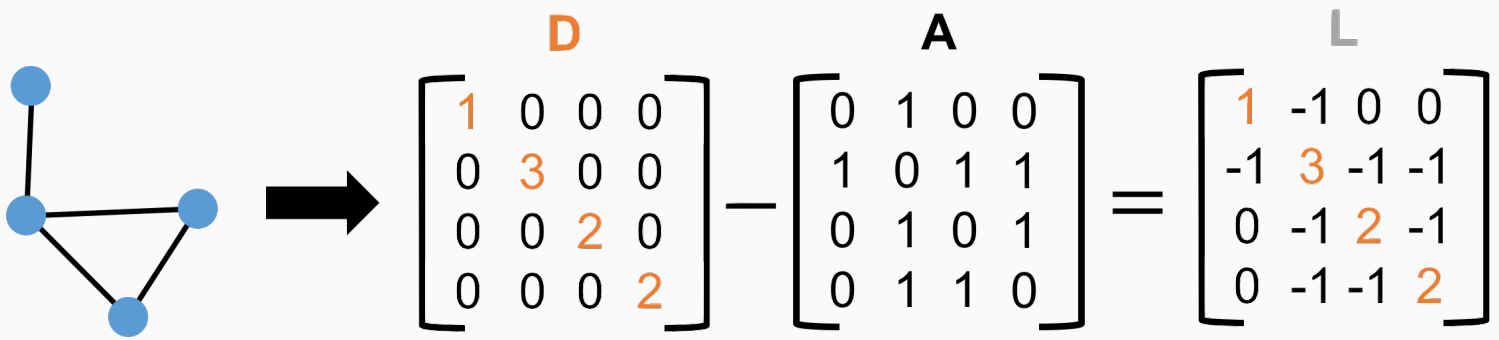
Two most common representations: $n \times n$ adjacency matrix $\mathbf{A}$ and graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{A}$ where $\mathbf{D}$ is the diagonal degree matrix.



Also common to look at normalized versions of both of these:

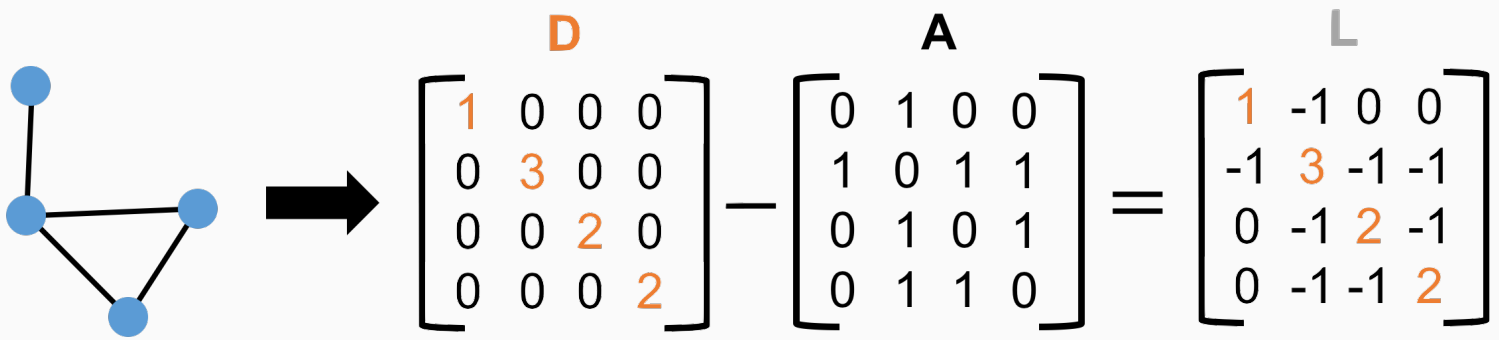
$$\bar{\mathbf{A}} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2} \quad \text{and} \quad \bar{\mathbf{L}} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$$

The Laplacian View



Edge-Vertex Incidence Matrix: B has a row for every edge in G. The row for edge (i, j) has a $+1$ at position i, a -1 at position j, and zeros elsewhere.

The Laplacian View



Edge-Vertex Incidence Matrix: $\mathbf{B}$ has a row for every edge in G. The row for edge (i, j) has a $+1$ at position i, a -1 at position j, and zeros elsewhere.

Fact: $\mathbf{L} = \mathbf{B}^T\mathbf{B}$ where B is the "edge-vertex incidence" matrix.

The Laplacian View

Example: For the graph above, $\mathbf{B}$ is a 4×4 matrix (4 edges, 4 vertices):

$$\mathbf{B} = \begin{array}{c} \\ (1,2) \\ (2,3) \\ (2,4) \\ (3,4) \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{pmatrix} +1 & -1 & 0 & 0 \\ 0 & +1 & -1 & 0 \\ 0 & +1 & 0 & -1 \\ 0 & 0 & +1 & -1 \end{pmatrix} \end{array}$$

The Laplacian View

Example: For the graph above, $\mathbf{B}$ is a 4×4 matrix (4 edges, 4 vertices):

$$\mathbf{B} = \begin{array}{c} \\ (1,2) \\ (2,3) \\ (2,4) \\ (3,4) \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{pmatrix} +1 & -1 & 0 & 0 \\ 0 & +1 & -1 & 0 \\ 0 & +1 & 0 & -1 \\ 0 & 0 & +1 & -1 \end{pmatrix} \end{array}$$

$$\mathbf{B}^T\mathbf{B} = \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 3 & -1 & -1 \\ 0 & -1 & 2 & -1 \\ 0 & -1 & -1 & 2 \end{pmatrix} = \mathbf{L}$$

The Laplacian View

Conclusions from $\mathbf{L} = \mathbf{B}^T\mathbf{B}$

- $\mathbf{L}$ is positive semidefinite: $\mathbf{x}^T\mathbf{L}\mathbf{x} \geq 0$ for all $\mathbf{x}$.

The Laplacian View

Conclusions from $\mathbf{L} = \mathbf{B}^T\mathbf{B}$

- $\mathbf{L}$ is positive semidefinite: $\mathbf{x}^T\mathbf{L}\mathbf{x} \geq 0$ for all $\mathbf{x}$.

- $\mathbf{L} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T$ where $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ is $\mathbf{B}$'s SVD. Columns of $\mathbf{V}$ are eigenvectors of $\mathbf{L}$.

The Laplacian View

Conclusions from $\mathbf{L} = \mathbf{B}^T\mathbf{B}$

- $\mathbf{L}$ is positive semidefinite: $\mathbf{x}^T\mathbf{L}\mathbf{x} \geq 0$ for all $\mathbf{x}$.
- $\mathbf{L} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T$ where $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ is $\mathbf{B}$'s SVD. Columns of $\mathbf{V}$ are eigenvectors of $\mathbf{L}$.
- For any vector $\mathbf{x} \in \mathbb{R}^n$,

$$\mathbf{x}^T\mathbf{L}\mathbf{x} = \sum_{(i,j)\in E} (\mathbf{x}(i) - \mathbf{x}(j))^2.$$

The Laplacian View

Conclusions from $\mathbf{L} = \mathbf{B}^T\mathbf{B}$

- $\mathbf{L}$ is positive semidefinite: $\mathbf{x}^T\mathbf{L}\mathbf{x} \geq 0$ for all $\mathbf{x}$.

- $\mathbf{L} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T$ where $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ is $\mathbf{B}$'s SVD. Columns of $\mathbf{V}$ are eigenvectors of $\mathbf{L}$.

- For any vector $\mathbf{x} \in \mathbb{R}^n$,

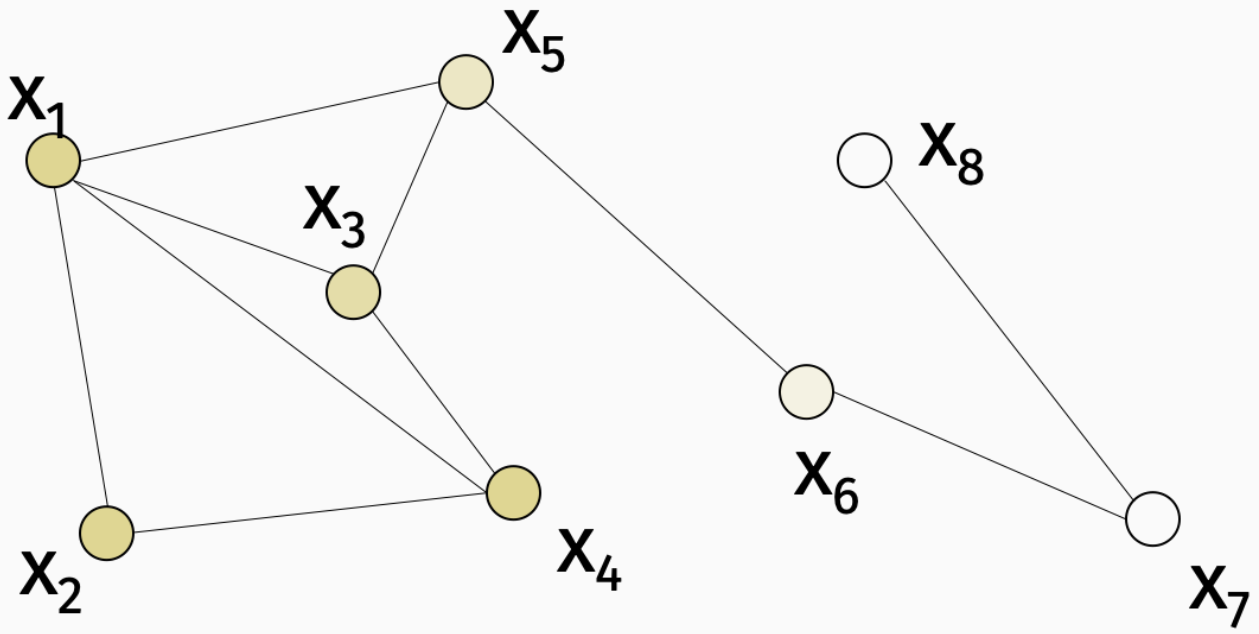
$$\mathbf{x}^T\mathbf{L}\mathbf{x} = \sum_{(i,j)\in E} (\mathbf{x}(i) - \mathbf{x}(j))^2.$$

Proof:

$$\mathbf{x}^T\mathbf{L}\mathbf{x} = \mathbf{x}^T\mathbf{B}^T\mathbf{B}\mathbf{x} = \|\mathbf{B}\mathbf{x}\|_2^2 = \sum_{\ell=1}^{m} (\mathbf{B}\mathbf{x})_\ell = \sum_{\ell=1}^{m} (\mathbf{B}_\ell^T\mathbf{x})^2$$

The Laplacian View

$\mathbf{x}^T L\mathbf{x} = \sum_{(i,j)\in E}(\mathbf{x}(i) - \mathbf{x}(j))^2$. So $\mathbf{x}^T L\mathbf{x}$ is small if $\mathbf{x}$ is a "smooth" function with respect to the graph.



Smallest Laplacian Eigenvector

Courant–Fischer min-max principle

Let $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ be the eigenvectors of $\mathbf{L}$.

$$\mathbf{v}_n = \underset{\|\mathbf{v}\|=1}{\arg\min}\, \mathbf{v}^T \mathbf{L} \mathbf{v}$$

Smallest Laplacian Eigenvector

Courant–Fischer min-max principle

Let $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ be the eigenvectors of $\mathbf{L}$.

$$\mathbf{v}_n = \arg\min_{\|\mathbf{v}\|=1} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

$$\mathbf{v}_{n-1} = \arg\min_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_n} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

Smallest Laplacian Eigenvector

Courant–Fischer min-max principle

Let $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ be the eigenvectors of $\mathbf{L}$.

$$\mathbf{v}_n = \arg\min_{\|\mathbf{v}\|=1} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

$$\mathbf{v}_{n-1} = \arg\min_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_n} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

$$\mathbf{v}_{n-2} = \arg\min_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_n, \mathbf{v}_{n-1}} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

Smallest Laplacian Eigenvector

Courant–Fischer min-max principle

Let $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ be the eigenvectors of $\mathbf{L}$.

$$\mathbf{v}_n = \arg\min_{\|\mathbf{v}\|=1} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

$$\mathbf{v}_{n-1} = \arg\min_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_n} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

$$\mathbf{v}_{n-2} = \arg\min_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_n, \mathbf{v}_{n-1}} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

$$\vdots$$

Smallest Laplacian Eigenvector

Courant–Fischer min-max principle

Let $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ be the eigenvectors of $\mathbf{L}$.

$$\begin{aligned}
\mathbf{v}_n &= \arg\min_{\|\mathbf{v}\|=1} \mathbf{v}^T \mathbf{L} \mathbf{v} \\
\mathbf{v}_{n-1} &= \arg\min_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_n} \mathbf{v}^T \mathbf{L} \mathbf{v} \\
\mathbf{v}_{n-2} &= \arg\min_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_n, \mathbf{v}_{n-1}} \mathbf{v}^T \mathbf{L} \mathbf{v} \\
&\vdots \\
\mathbf{v}_1 &= \arg\min_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_n, \ldots, \mathbf{v}_2} \mathbf{v}^T \mathbf{L} \mathbf{v}
\end{aligned}$$

Largest Laplacian Eigenvector

Courant–Fischer min-max principle

Let $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ be the eigenvectors of $\mathbf{L}$.

$$\mathbf{v}_1 = \underset{\|\mathbf{v}\|=1}{\arg\max}\, \mathbf{v}^T \mathbf{L} \mathbf{v}$$

Largest Laplacian Eigenvector

Courant–Fischer min-max principle

Let $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ be the eigenvectors of $\mathbf{L}$.

$$\mathbf{v}_1 = \arg\max_{\|\mathbf{v}\|=1} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

$$\mathbf{v}_2 = \arg\max_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_1} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

Largest Laplacian Eigenvector

Courant–Fischer min-max principle

Let $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ be the eigenvectors of $\mathbf{L}$.

$$\mathbf{v}_1 = \arg\max_{\|\mathbf{v}\|=1} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

$$\mathbf{v}_2 = \arg\max_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_1} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

$$\mathbf{v}_3 = \arg\max_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_1, \mathbf{v}_2} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

Largest Laplacian Eigenvector

Courant–Fischer min-max principle

Let $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ be the eigenvectors of $\mathbf{L}$.

$$\mathbf{v}_1 = \arg\max_{\|\mathbf{v}\|=1} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

$$\mathbf{v}_2 = \arg\max_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_1} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

$$\mathbf{v}_3 = \arg\max_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_1, \mathbf{v}_2} \mathbf{v}^T \mathbf{L} \mathbf{v}$$

$$\vdots$$

Largest Laplacian Eigenvector

Courant–Fischer min-max principle

Let $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ be the eigenvectors of $\mathbf{L}$.

$$
\begin{aligned}
\mathbf{v}_1 &= \arg\max_{\|\mathbf{v}\|=1} \mathbf{v}^T \mathbf{L} \mathbf{v} \\
\mathbf{v}_2 &= \arg\max_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_1} \mathbf{v}^T \mathbf{L} \mathbf{v} \\
\mathbf{v}_3 &= \arg\max_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_1, \mathbf{v}_2} \mathbf{v}^T \mathbf{L} \mathbf{v} \\
&\vdots \\
\mathbf{v}_n &= \arg\max_{\|\mathbf{v}\|=1, \mathbf{v} \perp \mathbf{v}_1, \ldots, \mathbf{v}_{n-1}} \mathbf{v}^T \mathbf{L} \mathbf{v}
\end{aligned}
$$

The Laplacian View

Another conclusion from $\mathbf{L} = \mathbf{B}^T\mathbf{B}$:

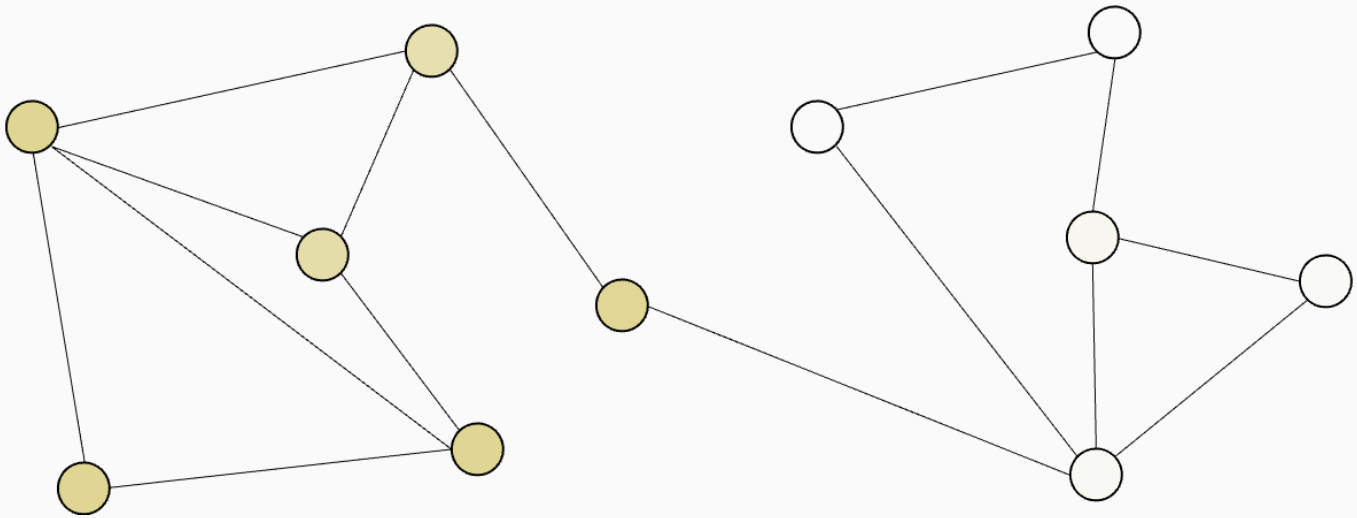
For a cut indicator vector $\mathbf{c} \in \{-1, 1\}^n$ with $\mathbf{c}(i) = -1$ for $i \in S$ and $\mathbf{c}(i) = 1$ for $i \in T = V \setminus S$:

The Laplacian View

Another conclusion from $\mathbf{L} = \mathbf{B}^T\mathbf{B}$:

For a cut indicator vector $\mathbf{c} \in \{-1, 1\}^n$ with $\mathbf{c}(i) = -1$ for $i \in S$ and $\mathbf{c}(i) = 1$ for $i \in T = V \setminus S$:

$$\mathbf{c}^T L\mathbf{c} = \sum_{(i,j)\in E} (\mathbf{c}(i) - \mathbf{c}(j))^2 = 4 \cdot \mathrm{cut}(S, T). \tag{1}$$



Spectral Graph Partitioning

- Introduce NP-hard graph partitioning prob. important in:
  - Understanding social networks.
  - Unsupervised machine learning (spectral clustering).
  - Graph visualization.
  - Mesh partitioning.

Spectral Graph Partitioning

- Introduce NP-hard graph partitioning prob. important in:
  - Understanding social networks.
  - Unsupervised machine learning (spectral clustering).
  - Graph visualization.
  - Mesh partitioning.
- See how this problem can be solved heuristically using Laplacian eigenvectors.

Spectral Graph Partitioning

- Introduce NP-hard graph partitioning prob. important in:
  - Understanding social networks.
  - Unsupervised machine learning (spectral clustering).
  - Graph visualization.
  - Mesh partitioning.
- See how this problem can be solved heuristically using Laplacian eigenvectors.
- Give an "average case" analysis of the method for a common random graph model.

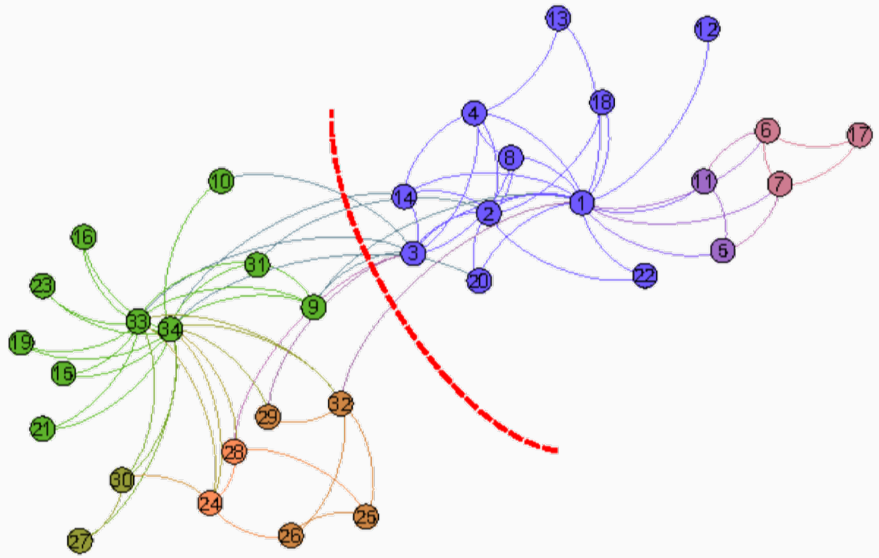
Spectral Graph Partitioning

- Introduce NP-hard graph partitioning prob. important in:
  - Understanding social networks.
  - Unsupervised machine learning (spectral clustering).
  - Graph visualization.
  - Mesh partitioning.
- See how this problem can be solved heuristically using Laplacian eigenvectors.
- Give an "average case" analysis of the method for a common random graph model.
- Use two tools: matrix concentration and eigenvector perturbation bounds.

Balanced Cut

Goal: Given a graph $G = (V, E)$, partition nodes along a cut that:

- Has few crossing edges: $|\{(u, v) \in E : u \in S, v \in T\}|$ is small.
- Separates large partitions: $|S|, |T|$ are not too small.



(a) Zachary Karate Club Graph

Example application: Understanding community structure in social networks.

Social Networks in the 1970s

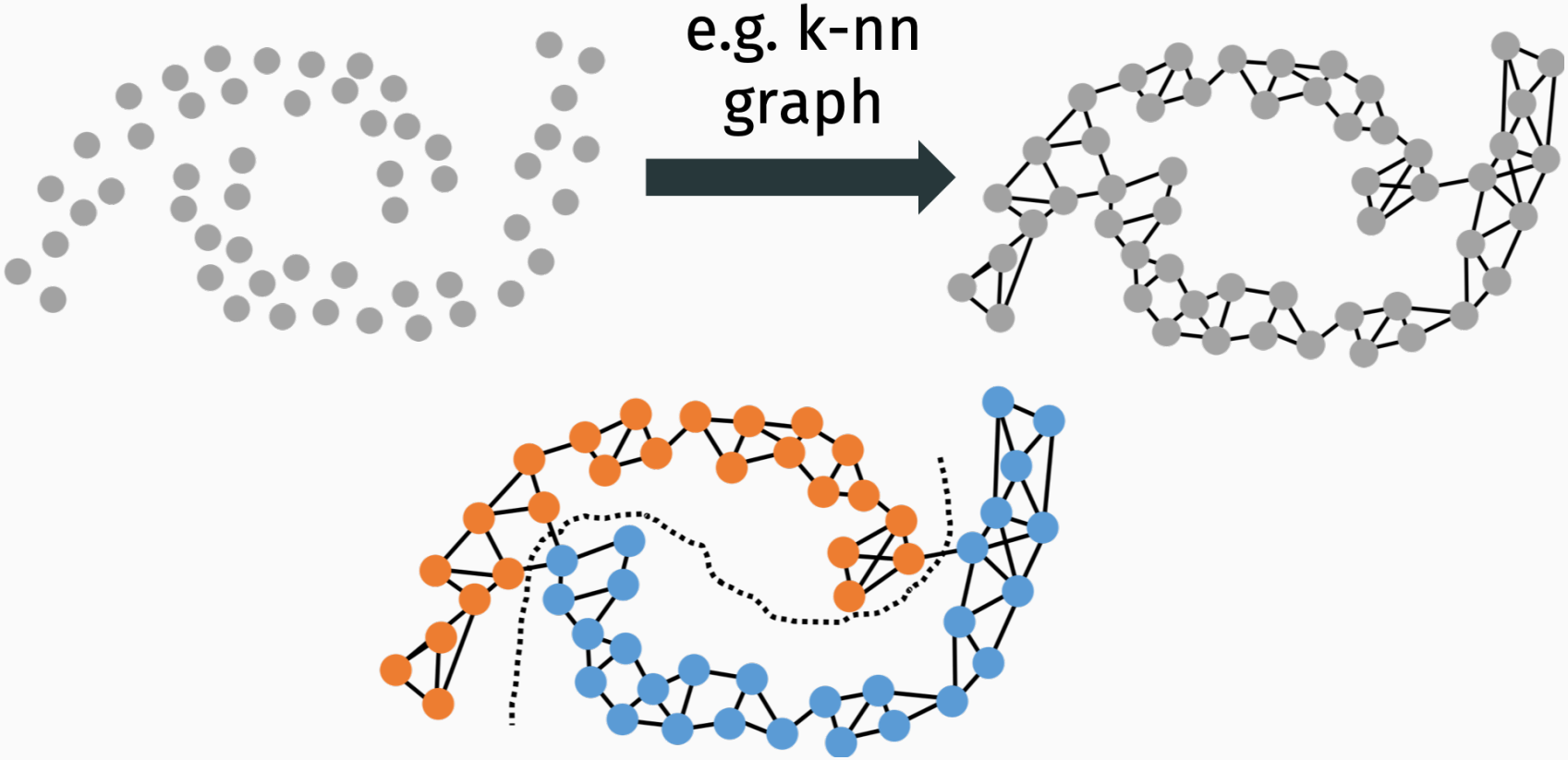
Wayne W. Zachary (1977). An Information Flow Model for Conflict and Fission in Small Groups.

"At the beginning of the study there was an incipient conflict between the club president, John A., and Mr. Hi over the price of karate lessons. Mr. Hi, who wished to raise prices, claimed the authority to set his own lesson fees, since he was the instructor. John A., who wished to stabilize prices, claimed the authority to set the lesson fees since he was the club's chief administrator. As time passed the entire club became divided over this issue, and the conflict became translated into ideological terms by most club members."

Zachary constructed a social network by hand and used a minimum cut algorithm to correctly predict who sided with who in the conflict. Beautiful paper – definitely worth checking out!

Spectral Clustering

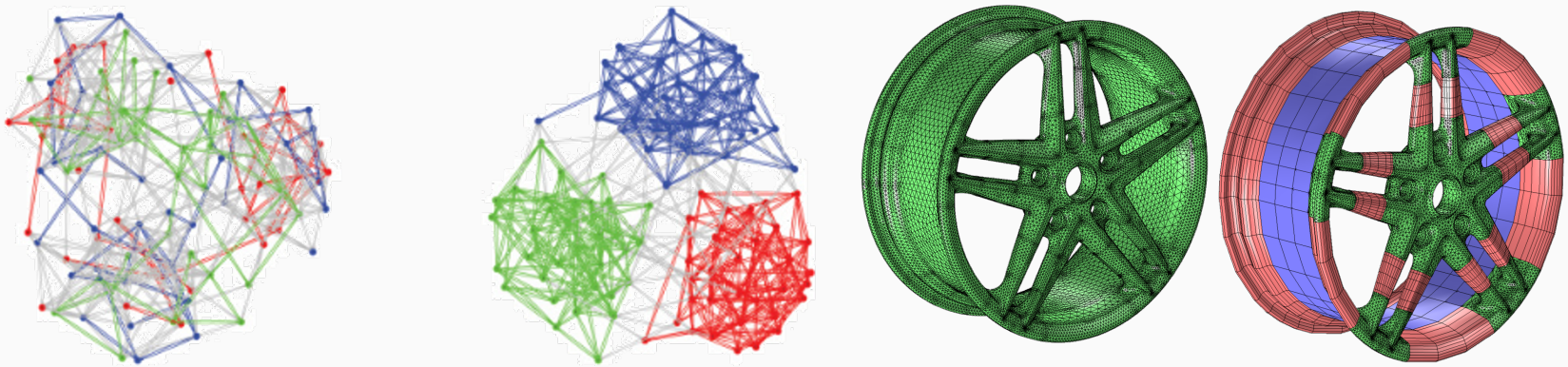
Idea: Construct synthetic graph for data that is hard to cluster.



Spectral Clustering, Laplacian Eigenmaps, Locally linear embedding, Isomap, etc.

Tons of Other Applications!

Balanced cut algorithms are also use in distributing data in graph databases, for partitioning finite element meshes in scientific computing (e.g., that arise when solving differential equations), and more.



Lots of good software packages (e.g. METIS).