

CS5275 – The Algorithm Designer's Toolkit  
(S2 AY2025/26)

Lecture 6:

Expanders – the power method

# Recap

$$\text{Cheeger inequality: } \frac{\lambda_2}{2} \leq \Phi(G) \leq \sqrt{2\lambda_2}$$

## Lemma:

- Let  $\mathbf{x}$  be any vector orthogonal to  $\mathbf{1}$ .
- There is a polynomial-time algorithm that computes a cut with conductance at most  $\sqrt{2R_N(\mathbf{x})}$ .



Setting  $\mathbf{x} = \mathbf{v}_2$  yields a cut with conductance at most  $\sqrt{2\lambda_2}$ .

# Recap

$$\text{Cheeger inequality: } \frac{\lambda_2}{2} \leq \Phi(G) \leq \sqrt{2\lambda_2}$$

## Lemma:

- Let  $\mathbf{x}$  be any vector orthogonal to  $\mathbf{1}$ .
- There is a polynomial-time algorithm that computes a cut with conductance at most  $\sqrt{2R_N(\mathbf{x})}$ .



Setting  $\mathbf{x} = \mathbf{v}_2$  yields a cut with conductance at most  $\sqrt{2\lambda_2}$ .

**Our goal:** Obtaining this in polynomial time.

From any vector  $\mathbf{x} \perp \mathbf{1}$  with  $R_N(\mathbf{x}) \leq \lambda_2 + \epsilon$ , we get a cut with conductance at most  $\sqrt{2(\lambda_2 + \epsilon)}$ .

# Setting

- Let  $\mathbf{M}$  be a real symmetric  $(n \times n)$ -matrix.
  - Eigenvalues:  $0 \leq \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$
  - Eigenvectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$  (orthonormal vectors)

Eigenvalue range for  $d$ -regular graphs:

- $\mathbf{A}$  (adjacency):  $[-d, d]$  ✗
- $\mathbf{L} = d\mathbf{I} - \mathbf{A}$  (Laplacian):  $[0, 2d]$  ✓
- $\mathbf{N} = \mathbf{I} - \frac{1}{d}\mathbf{A}$  (normalized Laplacian):  $[0, 2]$  ✓

# Setting

- Let  $M$  be a real symmetric  $(n \times n)$ -matrix.
  - Eigenvalues:  $0 \leq \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$
  - Eigenvectors  $v_1, v_2, \dots, v_n$  (orthonormal vectors)
- **First goal:** Approximating the largest eigenvector  $v_n$ .
  - Find a vector  $x$  with  $R_M(x) \geq (1 - \epsilon) \cdot \lambda_n$ .



We will later extend this to:

- The second largest eigenvector  $v_{n-1}$ .
- The smallest eigenvector  $v_1$ .
- The second smallest eigenvector  $v_2$  (our main goal).

# The power method

## Algorithm **POWER**:

Input: a matrix  $\mathbf{M}$  and a parameter  $k$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$ .

**Claim:** There is a choice of parameter  $k \in O\left(\frac{\log n}{\epsilon}\right)$  such that  $\Pr[R_{\mathbf{M}}(\mathbf{x}) \geq (1 - \epsilon) \cdot \lambda_n] \geq \frac{3}{16}$ .

## Proof sketch:

The matrix  $\mathbf{M}^k$  has:

- Eigenvalues:  $0 \leq \lambda_1^k \leq \lambda_2^k \leq \dots \leq \lambda_n^k$
- The same eigenvectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$

# The power method

## Algorithm **POWER**:

Input: a matrix  $\mathbf{M}$  and a parameter  $k$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$ .

**Claim:** There is a choice of parameter  $k \in O\left(\frac{\log n}{\epsilon}\right)$  such that  $\Pr[R_{\mathbf{M}}(\mathbf{x}) \geq (1 - \epsilon) \cdot \lambda_n] \geq \frac{3}{16}$ .


## Proof sketch:

The matrix  $\mathbf{M}^k$  has:

- Eigenvalues:  $0 \leq \lambda_1^k \leq \lambda_2^k \leq \dots \leq \lambda_n^k$
- The same eigenvectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$

Suppose  $\lambda_{i^*} \leq (1 - \epsilon) \cdot \lambda_n < \lambda_{i^*+1}$ .

Then  $\lambda_{i^*}^k \leq (1 - \epsilon)^k \cdot \lambda_n^k$

  
 $1/\text{poly}(n)$

# The power method

## Algorithm **POWER**:

Input: a matrix  $\mathbf{M}$  and a parameter  $k$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$ .

**Claim:** There is a choice of parameter  $k \in O\left(\frac{\log n}{\epsilon}\right)$  such that  $\Pr[R_{\mathbf{M}}(\mathbf{x}) \geq (1 - \epsilon) \cdot \lambda_n] \geq \frac{3}{16}$ .

## Proof sketch:

The matrix  $\mathbf{M}^k$  has:

- Eigenvalues:  $0 \leq \lambda_1^k \leq \lambda_2^k \leq \dots \leq \lambda_n^k$
- The same eigenvectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$

Suppose  $\lambda_{i^*} \leq (1 - \epsilon) \cdot \lambda_n < \lambda_{i^*+1}$ .

Then  $\lambda_{i^*}^k \leq (1 - \epsilon)^k \cdot \lambda_n^k$

$\downarrow$   
 $1/\text{poly}(n)$

$$\mathbf{x}_0 = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \dots + c_i \mathbf{v}_i + c_{i+1} \mathbf{v}_{i+1} + \dots + c_n \mathbf{v}_n$$

$$\mathbf{x} = \mathbf{M}^k \mathbf{x}_0 = \underbrace{c_1 \lambda_1^k \mathbf{v}_1 + c_2 \lambda_2^k \mathbf{v}_2 + \dots + c_{i^*} \lambda_{i^*}^k \mathbf{v}_{i^*}}_{\text{Negligibly small}} + \underbrace{c_{i^*+1} \lambda_{i^*+1}^k \mathbf{v}_{i^*+1} + \dots + c_n \lambda_n^k \mathbf{v}_n}_{R_{\mathbf{M}}(\cdot) \geq (1 - \epsilon) \cdot \lambda_n}$$

# The power method

## Algorithm POWER:

Input: a matrix  $\mathbf{M}$  and a parameter  $k$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$ .

**Claim:** There is a choice of parameter  $k \in O\left(\frac{\log n}{\epsilon}\right)$  such that  $\Pr[R_{\mathbf{M}}(\mathbf{x}) \geq (1 - \epsilon) \cdot \lambda_n] \geq \frac{3}{16}$ .

## Proof sketch:

The matrix  $\mathbf{M}^k$  has:

- Eigenvalues:  $0 \leq \lambda_1^k \leq \lambda_2^k \leq \dots \leq \lambda_n^k$
- The same eigenvectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$

Suppose  $\lambda_{i^*} \leq (1 - \epsilon) \cdot \lambda_n < \lambda_{i^*+1}$ .

Then  $\lambda_{i^*}^k \leq (1 - \epsilon)^k \cdot \lambda_n^k$

$\downarrow$   
 $1/\text{poly}(n)$

To make this argument work, we need to start with a sufficiently large  $c_n = \langle \mathbf{x}_0, \mathbf{v}_n \rangle$ .

$$\mathbf{x}_0 = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \dots + c_i \mathbf{v}_i + c_{i+1} \mathbf{v}_{i+1} + \dots + c_n \mathbf{v}_n$$

$$\mathbf{x} = \mathbf{M}^k \mathbf{x}_0 = \underbrace{c_1 \lambda_1^k \mathbf{v}_1 + c_2 \lambda_2^k \mathbf{v}_2 + \dots + c_{i^*} \lambda_{i^*}^k \mathbf{v}_{i^*}}_{\text{Negligibly small}} + \underbrace{c_{i^*+1} \lambda_{i^*+1}^k \mathbf{v}_{i^*+1} + \dots + c_n \lambda_n^k \mathbf{v}_n}_{R_{\mathbf{M}}(\cdot) \geq (1 - \epsilon) \cdot \lambda_n}$$

# The power method

**Claim:** for any unit vector  $\mathbf{v}$ ,  $\Pr \left[ \langle \mathbf{x}_0, \mathbf{v} \rangle \geq \frac{1}{2} \right] \geq \frac{3}{16}$

## Algorithm **POWER**:

Input: a matrix  $\mathbf{M}$  and a parameter  $k$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$ .

To make this argument work, we need to start with a sufficiently large  $c_n = \langle \mathbf{x}_0, \mathbf{v}_n \rangle$ .

$$\mathbf{x}_0 = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \cdots + c_i \mathbf{v}_i + c_{i+1} \mathbf{v}_{i+1} + \cdots + \mathbf{c}_n \mathbf{v}_n$$

# The power method

**Claim:** for any unit vector  $\mathbf{v}$ ,  $\Pr\left[\langle \mathbf{x}_0, \mathbf{v} \rangle \geq \frac{1}{2}\right] \geq \frac{3}{16}$

**Proof:**

- $\mathbf{x}_0 = (x_1, x_2, \dots, x_n)$
- $\mathbf{v} = (v_1, v_2, \dots, v_n)$
- $Z = \langle \mathbf{x}_0, \mathbf{v} \rangle = \sum_{i=1}^n x_i v_i$

$$\mathbb{E}[Z] = \sum_{i=1}^n \mathbb{E}[x_i] v_i = 0$$

**Algorithm POWER:**

Input: a matrix  $\mathbf{M}$  and a parameter  $k$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$ .

# The power method

**Claim:** for any unit vector  $\mathbf{v}$ ,  $\Pr \left[ \langle \mathbf{x}_0, \mathbf{v} \rangle \geq \frac{1}{2} \right] \geq \frac{3}{16}$

## Algorithm **POWER**:

Input: a matrix  $\mathbf{M}$  and a parameter  $k$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$ .

## Proof:

- $\mathbf{x}_0 = (x_1, x_2, \dots, x_n)$
- $\mathbf{v} = (v_1, v_2, \dots, v_n)$
- $Z = \langle \mathbf{x}_0, \mathbf{v} \rangle = \sum_{i=1}^n x_i v_i$

$$\mathbb{E}[Z] = \sum_{i=1}^n \mathbb{E}[x_i] v_i = 0$$

$$\mathbb{E}[Z^2] = \sum_{i=1}^n \sum_{j=1}^n \mathbb{E}[x_i x_j] v_i v_j = \sum_{i=1}^n \mathbb{E}[x_i^2] v_i^2 = \sum_{i=1}^n v_i^2 = 1$$

$$\mathbb{E}[x_i x_j] = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

$\mathbf{v}$  is a unit vector.

# The power method

**Claim:** for any unit vector  $\mathbf{v}$ ,  $\Pr \left[ \langle \mathbf{x}_0, \mathbf{v} \rangle \geq \frac{1}{2} \right] \geq \frac{3}{16}$

## Algorithm **POWER**:

Input: a matrix  $\mathbf{M}$  and a parameter  $k$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$ .

### Proof:

- $\mathbf{x}_0 = (x_1, x_2, \dots, x_n)$
- $\mathbf{v} = (v_1, v_2, \dots, v_n)$
- $Z = \langle \mathbf{x}_0, \mathbf{v} \rangle = \sum_{i=1}^n x_i v_i$

$$\mathbb{E}[Z] = \sum_{i=1}^n \mathbb{E}[x_i] v_i = 0$$

$$\mathbb{E}[Z^2] = \sum_{i=1}^n \sum_{j=1}^n \mathbb{E}[x_i x_j] v_i v_j = \sum_{i=1}^n \mathbb{E}[x_i^2] v_i^2 = \sum_{i=1}^n v_i^2 = 1$$

$$\mathbb{E}[Z^4] = \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n \sum_{\ell=1}^n \underbrace{\mathbb{E}[x_i x_j x_k x_\ell]}_{\text{bracket}} v_i v_j v_k v_\ell = 3 \sum_{i=1}^n \mathbb{E}[x_i^2] v_i^2 \sum_{j=1}^n \mathbb{E}[x_j^2] v_j^2 - 2 \sum_{i=1}^n \mathbb{E}[x_i^4] v_i^4$$

$$i \notin \{j, k, \ell\} \rightarrow \mathbb{E}[x_i x_j x_k x_\ell] = 0$$

$$j \notin \{i, k, \ell\} \rightarrow \mathbb{E}[x_i x_j x_k x_\ell] = 0$$

$$k \notin \{i, j, \ell\} \rightarrow \mathbb{E}[x_i x_j x_k x_\ell] = 0$$

$$\ell \notin \{i, j, k\} \rightarrow \mathbb{E}[x_i x_j x_k x_\ell] = 0$$

For any  $i \neq j$ ,  $\mathbb{E}[x_i^2 x_j^2]$  appears  $\binom{4}{2} = 6$  times.

For any  $i$ ,  $\mathbb{E}[x_i^4]$  appears once.

# The power method

**Claim:** for any unit vector  $\mathbf{v}$ ,  $\Pr\left[\langle \mathbf{x}_0, \mathbf{v} \rangle \geq \frac{1}{2}\right] \geq \frac{3}{16}$

## Algorithm **POWER**:

Input: a matrix  $\mathbf{M}$  and a parameter  $k$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$ .

## Proof:

- $\mathbf{x}_0 = (x_1, x_2, \dots, x_n)$
- $\mathbf{v} = (v_1, v_2, \dots, v_n)$
- $Z = \langle \mathbf{x}_0, \mathbf{v} \rangle = \sum_{i=1}^n x_i v_i$

$$\mathbb{E}[Z] = \sum_{i=1}^n \mathbb{E}[x_i] v_i = 0$$

$$\mathbb{E}[Z^2] = \sum_{i=1}^n \sum_{j=1}^n \mathbb{E}[x_i x_j] v_i v_j = \sum_{i=1}^n \mathbb{E}[x_i^2] v_i^2 = \sum_{i=1}^n v_i^2 = 1$$

$$\mathbb{E}[Z^4] = \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n \sum_{\ell=1}^n \mathbb{E}[x_i x_j x_k x_\ell] v_i v_j v_k v_\ell = 3 \underbrace{\sum_{i=1}^n \mathbb{E}[x_i^2] v_i^2}_{=1} \underbrace{\sum_{j=1}^n \mathbb{E}[x_j^2] v_j^2}_{=1} - 2 \underbrace{\sum_{i=1}^n \mathbb{E}[x_i^4] v_i^4}_{= \sum_{i=1}^n v_i^4 \geq 0} \leq 3$$

# The power method

**Claim:** for any unit vector  $\mathbf{v}$ ,  $\Pr \left[ \langle \mathbf{x}_0, \mathbf{v} \rangle \geq \frac{1}{2} \right] \geq \frac{3}{16}$

**Proof:**

- $\mathbf{x}_0 = (x_1, x_2, \dots, x_n)$
- $\mathbf{v} = (v_1, v_2, \dots, v_n)$
- $Z = \langle \mathbf{x}_0, \mathbf{v} \rangle = \sum_{i=1}^n x_i v_i$
- $\mathbb{E}[Z] = 0$
- $\mathbb{E}[Z^2] = 1$
- $\mathbb{E}[Z^4] \leq 3$
- $\text{Var}[Z^2] = \mathbb{E}[Z^4] - \mathbb{E}[Z^2]^2 < \infty$

**Algorithm POWER:**

Input: a matrix  $\mathbf{M}$  and a parameter  $k$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$ .

# The power method

**Claim:** for any unit vector  $\mathbf{v}$ ,  $\Pr\left[\langle \mathbf{x}_0, \mathbf{v} \rangle \geq \frac{1}{2}\right] \geq \frac{3}{16}$

## Algorithm **POWER**:

Input: a matrix  $\mathbf{M}$  and a parameter  $k$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$ .

### Proof:

- $\mathbf{x}_0 = (x_1, x_2, \dots, x_n)$
- $\mathbf{v} = (v_1, v_2, \dots, v_n)$
- $Z = \langle \mathbf{x}_0, \mathbf{v} \rangle = \sum_{i=1}^n x_i v_i$

- $\mathbb{E}[Z] = 0$
- $\mathbb{E}[Z^2] = 1$
- $\mathbb{E}[Z^4] \leq 3$
- $\text{Var}[Z^2] = \mathbb{E}[Z^4] - \mathbb{E}[Z^2]^2 < \infty$

$$\begin{aligned}\Pr\left[\langle \mathbf{x}_0, \mathbf{v} \rangle \geq \frac{1}{2}\right] &= \Pr\left[Z \geq \frac{1}{2}\right] \geq \Pr\left[Z^2 \geq \frac{1}{4}\right] \\ &= \Pr[Z^2 \geq \delta \mathbb{E}[Z^2]] \geq (1 - \delta)^2 \frac{\mathbb{E}[Z^2]^2}{\mathbb{E}[Z^4]} \geq \left(\frac{3}{4}\right)^2 \cdot \frac{1}{3} = \frac{3}{16}\end{aligned}$$

$$\delta = \frac{1}{4}$$

$$X = Z^2$$

**Paley–Zygmund inequality:**  $\Pr[X \geq \delta \mathbb{E}[X]] \geq (1 - \delta)^2 \frac{\mathbb{E}[X]^2}{\mathbb{E}[X^2]}$ , for  $X \geq 0$ ,  $\text{Var}[X] < \infty$ ,  $\delta \in (0, 1)$

# The power method

The matrix  $\mathbf{M}^k$  has:

- Eigenvalues:  $0 \leq \lambda_1^k \leq \lambda_2^k \leq \dots \leq \lambda_n^k$
- The same eigenvectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$

Suppose  $\lambda_{i^*} \leq (1 - \epsilon) \cdot \lambda_n < \lambda_{i^*+1}$ .

$$\mathbf{x}_0 = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \dots + c_i \mathbf{v}_i + c_{i+1} \mathbf{v}_{i+1} + \dots + c_n \mathbf{v}_n$$



Assume  $c_n = \langle \mathbf{x}_0, \mathbf{v}_n \rangle \geq \frac{1}{2}$ .

$$\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$$

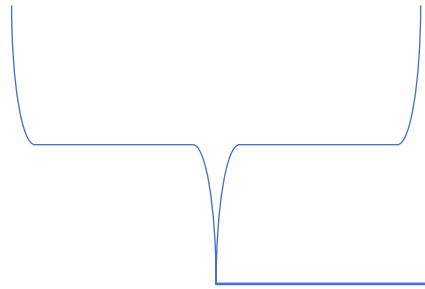
$$\begin{aligned} R_M(\mathbf{x}) &= \frac{\mathbf{x}^\top \mathbf{M} \mathbf{x}}{\mathbf{x}^\top \mathbf{x}} \\ &= \frac{\mathbf{x}_0^\top \mathbf{M}^{2k+1} \mathbf{x}_0}{\mathbf{x}_0^\top \mathbf{M}^{2k} \mathbf{x}_0} \\ &= \frac{\sum_{i=1}^n c_i^2 \lambda_i^{2k+1}}{\sum_{i=1}^n c_i^2 \lambda_i^{2k}} \end{aligned}$$

# The power method

The matrix  $\mathbf{M}^k$  has:

- Eigenvalues:  $0 \leq \lambda_1^k \leq \lambda_2^k \leq \dots \leq \lambda_n^k$
- The same eigenvectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$

Suppose  $\lambda_{i^*} \leq (1 - \epsilon) \cdot \lambda_n < \lambda_{i^*+1}$ .



$$\mathbf{x}_0 = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \dots + c_i \mathbf{v}_i + c_{i+1} \mathbf{v}_{i+1} + \dots + c_n \mathbf{v}_n$$



Assume  $c_n = \langle \mathbf{x}_0, \mathbf{v}_n \rangle \geq \frac{1}{2}$ .

$$\begin{aligned} R_{\mathbf{M}}(\mathbf{x}) &= \frac{\mathbf{x}^\top \mathbf{M} \mathbf{x}}{\mathbf{x}^\top \mathbf{x}} \\ &= \frac{\mathbf{x}_0^\top \mathbf{M}^{2k+1} \mathbf{x}_0}{\mathbf{x}_0^\top \mathbf{M}^{2k} \mathbf{x}_0} \\ &= \frac{\sum_{i=1}^n c_i^2 \lambda_i^{2k+1}}{\sum_{i=1}^n c_i^2 \lambda_i^{2k}} \\ &\geq \frac{(1 - \epsilon) \cdot \lambda_n \cdot \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}}{\sum_{i=1}^{i^*} c_i^2 \lambda_i^{2k} + \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}} \end{aligned}$$

# The power method

The matrix  $\mathbf{M}^k$  has:

- Eigenvalues:  $0 \leq \lambda_1^k \leq \lambda_2^k \leq \dots \leq \lambda_n^k$
- The same eigenvectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$

Suppose  $\lambda_{i^*} \leq (1 - \epsilon) \cdot \lambda_n < \lambda_{i^*+1}$ .

$$\sum_{i=1}^n c_i^2 = \|\mathbf{x}_0\|^2 = n$$

$$4 \cdot c_n^2 \leq 1$$

$$\begin{aligned} \sum_{i=1}^{i^*} c_i^2 \lambda_i^{2k} &\leq (1 - \epsilon)^{2k} \cdot \lambda_n^{2k} \cdot \sum_{i=1}^n c_i^2 \\ &= (1 - \epsilon)^{2k} \cdot n \cdot \lambda_n^{2k} \\ &\leq (1 - \epsilon)^{2k} \cdot 4n \cdot c_n^2 \lambda_n^{2k} \\ &\leq (1 - \epsilon)^{2k} \cdot 4n \cdot \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k} \end{aligned}$$

$$\mathbf{x}_0 = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \dots + c_i \mathbf{v}_i + c_{i+1} \mathbf{v}_{i+1} + \dots + c_n \mathbf{v}_n$$



Assume  $c_n = \langle \mathbf{x}_0, \mathbf{v}_n \rangle \geq \frac{1}{2}$ .

$$\begin{aligned} R_M(\mathbf{x}) &= \frac{\mathbf{x}^\top \mathbf{M} \mathbf{x}}{\mathbf{x}^\top \mathbf{x}} \\ &= \frac{\mathbf{x}_0^\top \mathbf{M}^{2k+1} \mathbf{x}_0}{\mathbf{x}_0^\top \mathbf{M}^{2k} \mathbf{x}_0} \\ &= \frac{\sum_{i=1}^n c_i^2 \lambda_i^{2k+1}}{\sum_{i=1}^n c_i^2 \lambda_i^{2k}} \\ &\geq \frac{(1 - \epsilon) \cdot \lambda_n \cdot \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}}{\sum_{i=1}^{i^*} c_i^2 \lambda_i^{2k} + \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}} \\ &\geq \frac{(1 - \epsilon) \cdot \lambda_n \cdot \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}}{(1 + (1 - \epsilon)^{2k} \cdot 4n) \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}} \end{aligned}$$

# The power method

The matrix  $\mathbf{M}^k$  has:

- Eigenvalues:  $0 \leq \lambda_1^k \leq \lambda_2^k \leq \dots \leq \lambda_n^k$
- The same eigenvectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$

Suppose  $\lambda_{i^*} \leq (1 - \epsilon) \cdot \lambda_n < \lambda_{i^*+1}$ .

$$\mathbf{x}_0 = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \dots + c_i \mathbf{v}_i + c_{i+1} \mathbf{v}_{i+1} + \dots + c_n \mathbf{v}_n$$



Assume  $c_n = \langle \mathbf{x}_0, \mathbf{v}_n \rangle \geq \frac{1}{2}$ .

$$\begin{aligned} R_{\mathbf{M}}(\mathbf{x}) &= \frac{\mathbf{x}^\top \mathbf{M} \mathbf{x}}{\mathbf{x}^\top \mathbf{x}} \\ &= \frac{\mathbf{x}_0^\top \mathbf{M}^{2k+1} \mathbf{x}_0}{\mathbf{x}_0^\top \mathbf{M}^{2k} \mathbf{x}_0} \\ &= \frac{\sum_{i=1}^n c_i^2 \lambda_i^{2k+1}}{\sum_{i=1}^n c_i^2 \lambda_i^{2k}} \\ &\geq \frac{(1 - \epsilon) \cdot \lambda_n \cdot \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}}{\sum_{i=1}^{i^*} c_i^2 \lambda_i^{2k} + \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}} \\ &\geq \frac{(1 - \epsilon) \cdot \lambda_n \cdot \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}}{(1 + (1 - \epsilon)^{2k} \cdot 4n) \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}} \\ &= \frac{(1 - \epsilon) \cdot \lambda_n}{1 + (1 - \epsilon)^{2k} \cdot 4n} \\ &\in \left(1 - O(\epsilon)\right) \cdot \lambda_n \end{aligned}$$

$$k \in O\left(\frac{\log n}{\epsilon}\right)$$

# The power method

$$\mathbf{x}_0 = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \cdots + c_i \mathbf{v}_i + c_{i+1} \mathbf{v}_{i+1} + \cdots + \mathbf{c}_n \mathbf{v}_n$$



Assume  $c_n = \langle \mathbf{x}_0, \mathbf{v}_n \rangle \geq \frac{1}{2}$ .

This happens with probability  $\geq \frac{3}{16}$ .

**Claim:** There is a choice of parameter  $k \in O\left(\frac{\log n}{\epsilon}\right)$  such that  $\Pr[R_M(\mathbf{x}) \geq (1 - \epsilon) \cdot \lambda_n] \geq \frac{3}{16}$ .

with a change of variable:  $\epsilon' \in \Theta(\epsilon)$

$$\begin{aligned} R_M(\mathbf{x}) &= \frac{\mathbf{x}^\top \mathbf{M} \mathbf{x}}{\mathbf{x}^\top \mathbf{x}} \\ &= \frac{\mathbf{x}_0^\top \mathbf{M}^{2k+1} \mathbf{x}_0}{\mathbf{x}_0^\top \mathbf{M}^{2k} \mathbf{x}_0} \\ &= \frac{\sum_{i=1}^n c_i^2 \lambda_i^{2k+1}}{\sum_{i=1}^n c_i^2 \lambda_i^{2k}} \\ &\geq \frac{(1 - \epsilon) \cdot \lambda_n \cdot \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}}{\sum_{i=1}^{i^*} c_i^2 \lambda_i^{2k} + \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}} \\ &\geq \frac{(1 - \epsilon) \cdot \lambda_n \cdot \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}}{(1 + (1 - \epsilon)^{2k} \cdot 4n) \sum_{i=i^*+1}^n c_i^2 \lambda_i^{2k}} \\ &= \frac{(1 - \epsilon) \cdot \lambda_n}{1 + (1 - \epsilon)^{2k} \cdot 4n} \\ &\in (1 - O(\epsilon)) \cdot \lambda_n \end{aligned}$$

# Second largest eigenvector

- Let  $\mathbf{M}$  be a real symmetric  $(n \times n)$ -matrix.
  - Eigenvalues:  $0 \leq \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$
  - Eigenvectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$  (orthonormal vectors)
- **Done:** Find a vector  $\mathbf{x}$  with  $R_{\mathbf{M}}(\mathbf{x}) \geq (1 - \epsilon) \cdot \lambda_n$ .
- **Next:** The second largest eigenvector  $\mathbf{v}_{n-1}$ .

Given  $\mathbf{v}_n$ , the goal is to find a vector  $\mathbf{x} \perp \mathbf{v}_n$  with  $R_{\mathbf{M}}(\mathbf{x}) \geq (1 - \epsilon) \cdot \lambda_{n-1}$ .

# Second largest eigenvector

## Algorithm **POWER**:

Input: a matrix  $\mathbf{M}$  and a parameter  $k$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$ .

This restricts ourselves to  $\text{span}\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{n-1}\}$ .  $\longrightarrow$

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- $\mathbf{x}'_0 = \mathbf{x}_0 - \langle \mathbf{x}_0, \mathbf{v}_n \rangle \cdot \mathbf{v}_n$
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}'_0$ .

# Second largest eigenvector

**Claim:** for any unit vector  $\mathbf{v}$ ,  $\Pr \left[ \langle \mathbf{x}_0, \mathbf{v} \rangle \geq \frac{1}{2} \right] \geq \frac{3}{16}$

$$\langle \mathbf{x}'_0, \mathbf{v}_{n-1} \rangle = \langle \mathbf{x}_0, \mathbf{v}_{n-1} \rangle$$

We still have:

$$\Pr \left[ \langle \mathbf{x}'_0, \mathbf{v}_{n-1} \rangle \geq \frac{1}{2} \right] \geq \frac{3}{16}$$

The same analysis:

There is a choice of parameter  $k \in O\left(\frac{\log n}{\epsilon}\right)$  such that  $\Pr[R_M(\mathbf{x}) \geq (1 - \epsilon) \cdot \lambda_{n-1}] \geq \frac{3}{16}$ .

**Algorithm POWER:**

Input: a matrix  $\mathbf{M}$  and a parameter  $k$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}_0$ .

- Pick  $\mathbf{x}_0 \in \{-1, 1\}^n$  uniformly at random.
- $\mathbf{x}'_0 = \mathbf{x}_0 - \langle \mathbf{x}_0, \mathbf{v}_n \rangle \cdot \mathbf{v}_n$
- Return  $\mathbf{x} = \mathbf{M}^k \mathbf{x}'_0$ .

# Approximating $\mathbf{v}_1$ and $\mathbf{v}_2$

- Let  $\mathbf{M}$  be a real symmetric  $(n \times n)$ -matrix.
  - Eigenvalues:  $0 \leq \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$
  - Eigenvectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$  (orthonormal vectors)
- Suppose we know an upper bound  $c \geq \lambda_n$
- Consider  $c\mathbf{I} - \mathbf{M}$ :
  - Same eigenvectors:  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$
  - Eigenvalues:  $c - \lambda_1 \geq c - \lambda_2 \geq \dots \geq c - \lambda_n \geq 0$

Now we can use the same method to approximate  $\mathbf{v}_1$  and  $\mathbf{v}_2$ .

# Computing a low-conductance cut

- Approximating the second largest eigenvector of  $2\mathbf{I} - \mathbf{N}$ :
  - We get a vector  $\mathbf{x} \perp \mathbf{1}$  with

$$R_{2\mathbf{I}-\mathbf{N}}(\mathbf{x}) = 2 - R_{\mathbf{N}}(\mathbf{x}) \geq (1 - \epsilon) \cdot (2 - \lambda_2)$$



$$R_{\mathbf{N}}(\mathbf{x}) \leq 2 - (1 - \epsilon) \cdot (2 - \lambda_2) \leq \lambda_2 + 2\epsilon$$



$$\epsilon' = 2\epsilon$$

We get a cut with conductance at most  $\sqrt{2(\lambda_2 + \epsilon')}$ .

# Summary

- Let  $\mathbf{M}$  be a real symmetric  $(n \times n)$ -matrix.
  - Eigenvalues:  $0 \leq \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$
  - Eigenvectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$

We showed how to approximate them efficiently.



Many applications in theory and in practice.

# Outlook

- We finished the discussion of these two topics:
  - Conductance approximation.
  - Spectral graph theory.
- **Next:** The probabilistic aspect of expanders.



This includes **random walks** and their applications.

# References

- **Main reference:**
  - Lecture 4.2 of <https://sites.google.com/site/th saranurak/teaching/Expander>
  - Chapter 9 of <https://lucatrevisan.github.io/books/expanders-2016.pdf>
- **Additional/optional reading:** The power method in distributed computing.
  - “Distributed Sparsest Cut via Eigenvalue Estimation” by Yannic Maus and Tijn de Vos  
<https://arxiv.org/pdf/2508.19898>