Matrix representation of a graph:

- Adjacency matrix of graph $G = (V,E)$

$A \in \mathbb{R}_{0,1}^{V \times V}$

$A(u,v) = \begin{cases} 1 & \text{if } (u,v) \in E \\ 0 & \text{otherwise} \end{cases}$

Can matrix-theoretic notions help shed light on the graph & its properties/structure?

Representations can be a powerful tool to work on an object, especially from a computational point of view.

(Recall FFT algo: coefficient representation $\leftrightarrow$ evaluation)

Spectral graph theory: Eigenvalues of matrix encode valuable information about the graph.

- Useful for structural analysis
- Algorithmically powerful, since spectra of matrices can be computed efficiently

( Full course on Spectral Graph Theory offered regularly by Prof. Gary Miller including this semester )
Interlude on eigenvalues

A = real symmetric \( n \times n \) matrix

\[ A(i,j) = A(j,i) \]

(Note: Adj matrix of an undirected graph is symmetric)

Defn \( \lambda \in \mathbb{R} \) is said to be an eigenvalue of a \( n \times n \) matrix \( M \) if \( \exists \vec{x} \in \mathbb{R}^n \), s.t. \( M \vec{x} = \lambda \vec{x} \)

Such an \( \vec{x} \) is called an eigenvector.

\[
\begin{bmatrix}
\lambda \\
M
\end{bmatrix} = 
\begin{bmatrix}
\lambda \\
\vec{x}
\end{bmatrix}
\]

Standard Fact: Let \( A \) be an \( n \times n \) real symmetric matrix

1. Then \( A \) has \( n \) real eigenvalues (including repetitions): \( \lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \ldots \geq \lambda_n \)

2. There exist \( n \) eigenvectors \( v_1, v_2, \ldots, v_n \) s.t. \( A v_i = \lambda_i v_i \) for \( i = 1, 2, \ldots, n \), and

(i) The vectors \( \{v_i\} \) span \( \mathbb{R}^n \)

(ii) Eigenvectors corresponding to different eigenvalues are orthogonal.
\[
\begin{align*}
\text{Pf. of } \forall i, j \neq j: \\
Av_i &= \lambda_i v_i \\
Av_j &= \lambda_j v_j \\
\langle v_i, Av_i \rangle &= \langle v_i, (Av_i) \rangle = \langle v_i, \lambda_i v_i \rangle \\
\langle v_j, Av_i \rangle &= \langle v_j, (Av_i) \rangle = \langle v_j, \lambda_j v_i \rangle \\
\Rightarrow \lambda_i \langle v_j, v_i \rangle &= \lambda_j \langle v_j, v_i \rangle \\
\Rightarrow v_i \cdot v_i = 0 \Rightarrow v_i \text{ and } v_i \text{ are orthogonal}
\end{align*}
\]

**Cor.** Every vector \( v \in (\mathbb{R}^n) \) can be written as

\[
v = a_1 v_1 + a_2 v_2 + \ldots + a_n v_n
\]
(As a linear combination of \( v_i \)s).

**Lem.:** Matrix \( A \) has eigenvalues \( \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \) Then

\[
\lambda_1 = \max_{\langle x, x \rangle = 1} \frac{x^TAx}{x^Tx} \quad \text{(Rayleigh quotient)}
\]

**Pf.** Take \( x = v_1 \),

\[
\langle v_1, Av_1 \rangle = \lambda_1 \langle v_1, v_1 \rangle \\
\Rightarrow v_1^T(Av_1) = \lambda_1 (v_1^Tv_1)
\]

To prove \( \max \langle x, Ax \rangle \leq \lambda_1 \), take any \( x \),

\[
x = \sum a_i v_i \Rightarrow \frac{x^TAx}{x^Tx} = \frac{\sum a_i^2 \lambda_i}{\sum a_i^2} \leq \lambda_1.
\]
Ok back to graph theory!

Eigenvalues of adj. matrix $A$ infor about graph $G$ properties.

Theorem: Let $G$ be connected. Then $\lambda_2 < 0$.

Let $G$ is connected. Then $\lambda_n = -\lambda_1$ iff $G$ is bipartite.

Examples of graph spectra:

$K_n$ (complete graph on $n$ vertices):

$\lambda_1 = n-1$

$\lambda_2 = \lambda_3 = \ldots = \lambda_n = -1$

$(4, -1, -1, -1, -1, -1)$

$K_5$

$C_3$

$C_4$

$C_5$

$C_6$

Paths:

$P_2$

$P_3$

$P_5$
$d$-regular graph := a graph where all vertex degrees = $d$.

All cycles are 2-regular.

3-regular gh

(Peterson graph)

Fact: For a $d$-regular graph, largest eigenvalue of its adj. matrix equals $d$.

Proof: $A \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \rightarrow$ each row has exactly $d$ 1's

$\Rightarrow d$ is an eigenvalue

Let $x$ be eigenvector with eigenvalue $\lambda_1$.

Let $u$ be s.t. $x(u)$ max. coordinate of $x$.

$\Rightarrow x = \begin{pmatrix} -1 \\ 2 \\ 1 \\ 4 \end{pmatrix}$

$u \rightarrow A \begin{pmatrix} x(u) \\ 1 \\ 1 \\ 1 \end{pmatrix} = \lambda_1 \begin{pmatrix} x(u) \\ 1 \\ 1 \\ 1 \end{pmatrix}$

$\Rightarrow (Ax)(u) = \sum_{(u,v) \in E} x(v) \leq d x(u)$. 

$\Rightarrow \lambda_1 \leq d$. 

Cu. DEE%
Theorem: For a d-regular graph $G$ whose adjacency matrix has eigenvalues $d = \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$, $G$ is connected if and only if $\lambda_2 < d$.

Proof: 1) $G$ is not connected $\implies \lambda_2 = d$.

Both $x$ & $y$ are eigenvectors with eigenvalue $d$. Plus they are linearly independent. (In fact orthogonal.)

Thus there are at least 2 eigenvalues equal to $d$. $\implies \lambda_2 \geq d$ $\implies \lambda_2 = d$

2) $\lambda_2 = d \implies G$ is disconnected

Call (5 vectors)

$\lambda_2 = d \implies \exists \vec{x} \in \mathbb{R}^n, \vec{y}, \vec{z}, \vec{t}$ such that $\vec{x} - \vec{z} - \vec{y} - \vec{t} = 0$, $\vec{x} \cdot \vec{z} = 0$, $\sum_{u \in V} x(u) = 0$. 

\[ (\vec{x} = 0) \]
Suppose $G$ is connected.

We'll prove all entries of $\overline{z}$ have to be equal, which contradicts $\sum x(u) = 0 \quad \forall u \in V$.

Let $x(v)$ be max value in vector $\overline{z}$.

$$x(v) = \max_{u \in V} x(u)$$

$$d \cdot x(v) = (A \overline{z})(v) = \sum_{w \sim v} x(w) \leq d \cdot x(v)$$

Only way equality holds is if $x(w) = x(v)$ for all nbrs $w$ of $v$.

Continuing this argument, because we assured $G$ to be connected, we eventually reach every vertex $u \in V$, and show $x(w) = x(v)$.

[Diagram of a tree structure]

More generally, (in a regular graph) $t$ eigenvalues equal to $d$.

There are easier ways to check connectivity (of course, but this spectral)
perspective allows one to define more quantitative aspects of connectivity.

\( \lambda_2 \) is much smaller than \( d \) \( \implies \) \( G \) is very well connected?

No bottlenecks

\( \lambda_2 \) is an (expanding graph)

very few edges cross two big halves of graph

"Sparse cut"

If \( \lambda_2 \approx d \) \( \implies \) Is there a sparse cut?

Yes! Further such a cut can be found using the second eigenvector \( v_2 \) corresponding to \( \lambda_2 \).

"Spectral partitioning algorithm"

- very popular heuristic
- very useful in divide and conquer algo.