SPECTRAL CLUSTERING
The RANDOM WALK Method

- Random walk on a graph

- Consider a random walk on an undirected graph

- Spectral clustering can be interpreted as trying to find a partition of the graph such that the random walk stays long within the same cluster and seldom jumps between clusters.
This is consistent with the \texttt{GRAPHit} cut initialization we discussed.

A partition with a low cut will have the property that the random walk does not have many opportunities to jump between clusters.

Defining the \texttt{RANDOM WALK}

The probability of jumping from vertex \(i\) to vertex \(j\) is 
\[
P_{ij} = \frac{w_{ij}}{\delta_i} \quad \text{where} \quad w_{ij} \quad \text{the weight of edge} \quad \{i,j\}
\]
and \(\delta_i = \text{degree of vertex} \ i\)

The transition matrix of the random walk
\[ P = \{ P_{ij} \}_{i,j=1,2,...} \]
defined as \[ P = D^{-1} W \]
where \( D \) is the degree matrix
\[ D = \begin{bmatrix} d_1 & & & \\ & \ddots & & \\ & & d_n & \\ 0 & & & 0 \end{bmatrix} \]
\[ D^{-1} = \begin{bmatrix} \frac{1}{d_1} & & & \\ & \ddots & & \\ & & \frac{1}{d_n} & \\ 0 & & & 0 \end{bmatrix} \]
\( W \) is the graph matrix

If the graph \( G \) is connected and not bipartite, the \( \text{RANDOM WALK} \) has a unique stationary distribution
\[ \pi = (\pi_1, ..., \pi_n)^T \]
which is given by

$$\bar{T}_i = \frac{d_i}{\text{vol}(A)}$$

- A beautiful relation between the normalized Laplacian $L_{zw}$ and the Random Walk matrix $L_{zw} = I - P$

As a consequence if

$\lambda$ is an eigenvalue of $L_{zw}$

with eigenvector $v$

iff

$1 - \lambda$ is an eigenvalue of $P$ with eigenvector $v$

conclusion: The smallest eigenvalue of $P$ and
the smallest eigenvalues of $L$ can be used to describe cluster properties of the graph.

**RANDOM WALKS & Ncut**

- A formal equivalence between Ncut and $P$:

**PROPOSITION**

Let $G$ be a connected graph and non-bipartite. Assume that we run the random walk $(X_t)_{t \geq 0}$.
Starting with $X_0$ in the stationary distribution $\pi$. For disjoint subsets $A, B \subset V$ denote

$$P(A \mid B) = P(X_1 \in B \mid X_0 \in A)$$

Then:

$$N_{cut}(A, \overline{A}) = P(\overline{A} \mid A) + P(A \mid \overline{A})$$

This property tells us that when minimizing $N_{cut}$, we actually look for a cut through the graph such that the random walk
Seldom transitions from \( A \) to \( \bar{A} \)
or vice-versa

The Commute Distance

A tight connection between random walks & graph Laplacians can be made via the commute distance on the graph.

The commute distance is also known as resistance distance in electronics graph theory.
The commute distance $c(i,j)$ between vertices $i$ and $j$ is the expected time of it takes the random walk to travel from vertex $i$ to vertex $j$ and back.

- The commuting distance has some nice properties which makes it particularly appealing for machine learning.

- As opposed to the shortest path distance on a graph,
the commute distance between two vertices decreases if there are many different short ways to get from vertex $i$ to vertex $j$.

- So instead of just looking for one shortest path, the commute distance looks at the set of short paths.

- Points which are connected by a short path and lie in the same cluster of the graph are much closer to each other than points
which are connected by a short path but lie in different clusters.

- **Remarkably**, the commute distance on a graph can be computed by the pseudoinverse or Moore-Penrose inverse $L^+$ of the graph Laplacian.

- The above shows that
  \[ V \] can be considered as a Euclidean distance function
on the vertices of the graph. This means that we can construct an embedding which maps the vertices $y_i$ of the graph on points $z_i \in \mathbb{R}^n$ such that the Euclidean distance between the points $z_i$ coincide with the commute distance on the graph.

Avoiding this way the "curse of dimensionality."