Lecture 12: 10-18-18

• Note: For an [redacted] data $\{x_i\}_{i=1}^n \subset \mathbb{R}^D$, the resulting Laplacian is $n \times n$. So, to compute the eigenvectors (all) is $O(n^3)$. Even if I only want the one with second smallest eigenvalue, this is $O(n^3)$.

. If the data is, say, [redacted] $M \times N$ image, then $n = MN$, so complexity is $O(M^2N^2)$... if $M = N = 1000$, this is $\sim 10^{12}$ calculations... a lot!

. From a practical perspective, we can improve things with sparse matrices, i.e. Laplacians that are mostly 0's. This can be done with, for example, nearest neighbor searches: $W_{ij} = \begin{cases} \|x_i - x_j\|_2^2, & x_i \text{ is a } k\text{-NN of } x_j \text{ or vice versa} \\ 0, & \text{else} \end{cases}$

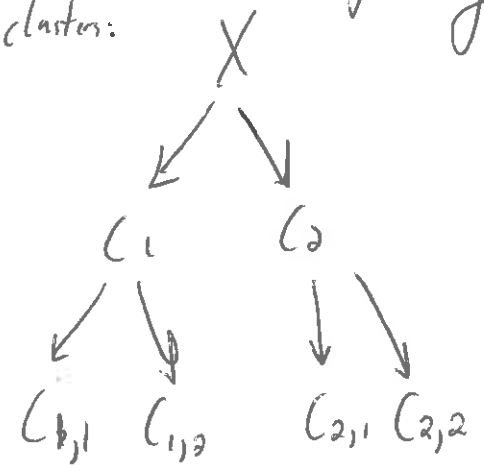
. The catch: computing a point's k NN is, a priori $O(n)$ for $k = O(1)$, again since a naive implementation requires sorting each point's distance to every other point. This brings us back to $O(n^2)$!

• Can do fast ($O(n \log(n))$ nearest neighbor searches using cover trees.

• This allows spectral graph methods to be run on very large datasets, and usually considering sparse graphs doesn't hurt us too badly.

Q: What if we want to cluster into more than two clusters?

- An approach is to recursively segment the clusters by running the segmentation algorithm again on each of the clusters:



- A perhaps better method is simply to use more eigenvectors.
- Indeed, one can note that each ϕ_k is an eigenvector of L, it is of length n. Then we can think of the map $x_i \mapsto (\phi_1(i), \phi_2(i), \ldots, \phi_n(i))$ as a dimension reduction/new coordinate system.
- Then instead of running K-means on $\{x_i\}_{i=1}^n$ we can run K-means on the transformed data $\{(\phi_1(i), \ldots, \phi_m(i))\}_{i=1}^n$ for some choice of $m \leq n$.
- This is sometimes refered to as the Laplacian eigenmaps embedding.

- A variant of spectral clustering is to: 1.) Compute L/L_{sym}/L_{rw}...
  2.) Compute the K principal eigenvectors $\{\phi_i\}_{i=1}^K$
  3.) Run K-means on $\{\phi_i\}_{i=1}^K$, with K = number eigenvectors used

↓

Ng, Jordan, Weiss, NIPS 2002.

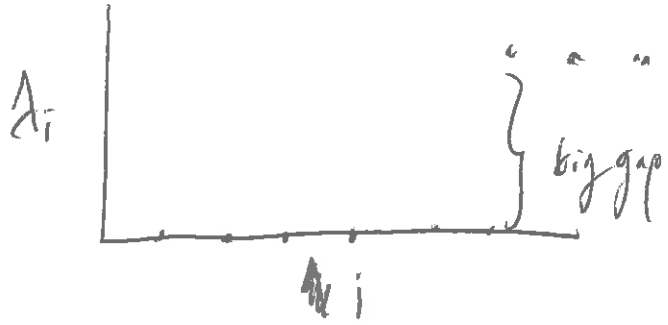
Remark: Spectral clustering allows to estimate k heuristically as

$$\hat{k} = \arg\max \lambda_{k+1} - \lambda_k,$$

the so-called eigengap heuristic.

Why does this have some reason? Well, suppose the clusters are points that are distance 0 apart and infinitly farm from each other. Then, modulo a reseorting of the columns, $W = \begin{pmatrix} \boxed{1} & & \\ & \ddots & \\ & & \boxed{1} \end{pmatrix}$ with $k = \#$ Cluster 1 blocks; zeros else.

In this case, W [illegible] is rank k. Thus, if we consider $L = D - W$, and if clusters are the same size, the eigenvalues of W will look like: $\lambda_1 = \lambda_2 = \ldots = \lambda_k = 0$, $\lambda_{k+1} > 0$:



- Unfortunately, this heuristic often fails in practice.
- Real graphs don't look like that!

Remark: $\lambda_0 = 0$, as per homework. In general, # 0 eigenvalues = # connected components in the graph.