Overview of Results Thus Far ...

Overview of Ratio Cut

Given an undirected similarity graph $G = (V, E)$ represented by weight matrix W, we would like to partition the nodes V into $k < n$ clusters in a reasonable manner. A naive method is to choose a set of edges $\{w_i\} \in E$ such that the graph separates into k distinct sub-graphs when they are removed, and pick $\{w_i\}$ such that $\sum_i w_i$ is minimized. In practice, this often results in highly imbalanced clusters. Ratio cut accounts for this by weighing the cuts by the inverse of the cluster sizes:

$$R(\{C_1, ..., C_k\}) = \sum_i^k \frac{\frac{1}{2} \sum_{r \in C_i, s \notin C_i} w_{rs}}{n_i}$$

where C_i is the set of vertices in cluster i and n_i is the number of verticies in C_i.

It can be shown[1] that this is equivalent to solving

$$\arg\min_H \operatorname{Tr}(H^\top L H)$$

where $L = T - W$, the combinatorial graph Laplacian, T is a diagonal matrix such that $T_{ii} = \sum_j W_{ij}$, and H is a special type of matrix:

$$H_{ij} = \begin{cases} n_i^{-1/2} & i \in C_j \\ 0 & \text{else} \end{cases}$$

Spectral Clustering

The optimization problem for the solution to ratio cut is a very particular type of constrained discrete optimization that is NP-hard. We can relax the constraints and only require that $H^\top H = I_k$ which leads us to the solution:

$$H^* = \begin{bmatrix} v_0 & v_1 & \cdots & v_{k-1} \end{bmatrix}$$

where v_i is the i^{th} eigenvector of L, in order of increasing eigenvalues. Then we note that H^* induces an embedding in $\mathbb{R}^{k-1}$ (since $v_0 = n^{-1/2} e$) and perform some sort of clustering method using Euclidean distances (or covariances). The heuristic is to perform k-means clustering.

It turns out this is equivalent to kernel k-means clustering:[2]

$$\arg\max_H \operatorname{Tr}(H^\top K H)$$

where H is constrained the same way as in ratio cut, K is a kernel matrix, and we set $K = L^\dagger$, the Moore-Penrose inverse of L. Note that since L is positive semi-definite, $L^\dagger$ is as well. Furthermore, if we take the eigendecomposition of $L = V \Lambda V^\top$, we have $L^\dagger = V \Lambda^\dagger V^\top$.

[1] https://arxiv.org/pdf/0711.0189.pdf
[2] http://pages.iu.edu/~mtrosset/Courses/675/notes.pdf

Kernel k-means is also NP-hard, but if the kernel matrix is positive semidefinite, then there exists a Euclidean representation for it, and Lloyd's method has been shown to be successful in solving k-means for Euclidean data. But note that treating $L^\dagger$ as a kernel matrix results in a different embedding. Instead of using the first $k-1$ eigenvalues of L (which is equivalent to using the first $k-1$ eigenvalues of $L^\dagger$), we might instead perform the full embedding in $\mathbb{R}^{n-1}$ to get the $\mathbb{R}^{n \times n-1}$ data matrix

$$X = \begin{bmatrix} \lambda_1^{-1/2} v_1 & \cdots & \lambda_{n-1}^{-1/2} v_{n-1} \end{bmatrix}$$

where v_i and λ_i are the eigenvectors and eigenvalues of L.

We can also note that this embedding is the full combinatorial Laplacian eigenmap, and the resulting squared Euclidean distance matrix is the same as the matrix of expected commute times.

Example of Non-Equivalency

Solving the relaxed version of ratio cut is almost equivalent to solving kernel k-means, with a slight difference in the embeddings. One might be tempted to say:

$$\arg\min_H \operatorname{Tr}(H^\top L H) = \arg\max_H \operatorname{Tr}(H^\top L^\dagger H)$$

However, this is not the case, as provided by a counterexample from Trosset[3].

Proposed Problems of Interest

1. Under what criteria can we equate $\arg\min_H \operatorname{Tr}(H^\top L H) = \arg\max_H \operatorname{Tr}(H^\top L^\dagger H)$? von Luxburg proposed that if the original similarity weight matrix W is positive semidefinite, this may hold. However, counterexamples can be constructed to show that a positive semidefinite (or positive definite) W results in different solutions to each problem, and it is trivial to construct an example of a W that is not positive (semi)definite yet results in the same solution to both problems.
2. Can we develop a method that is better than the heuristic spectral clustering algorithm to solve ratio cut? One proposal might be to come up with some sort of exchange algorithm. Such a method is very slow, and an example[4] showed that it's easy to get stuck at a local minimum.
3. Can we find some other kernel matrix K such that $\arg\min_H \operatorname{Tr}(H^\top L H) = \arg\max_H \operatorname{Tr}(H^\top K H)$? If such a kernel matrix can be found, then this would also solve #2.

Equivalence Criteria

Defining h_j as the j^{th} column of H (corresponding to C_j), it can be shown that $\operatorname{Tr}(H^\top L H) = \operatorname{Tr}(\Lambda V^\top H H^\top V)$, and $[V^\top H H^\top V]_{ii} = \sum_j^k (v_i^\top h_j)^2$, so the ratio cut objective can be written as $\sum_{i=1}^{n-1} \lambda_i \sum_j^k (v_i^\top h_j)^2$. Note that $\sum_j^k (v_i^\top h_j)^2 = k$.

From here, we can formulate an equivalent problem: Show under what criteria

$$\sum_i^r p_i \lambda_i \leq \sum_i^r q_i \lambda_i \iff \sum_i^r p_i / \lambda_i \geq \sum_i^r q_i / \lambda_i$$

[3] http://pages.iu.edu/~mtrosset/Courses/675/notes.pdf#page=128

[4] https://github.com/johneverettkoo/summer-research-2018/blob/master/graph-partitioning-exploration.pdf

where $p_i, q_i \geq 0$,
$\sum_i p_i = \sum_i q_i = 1$, and
$\lambda_i > 0$.

This has been shown to be always true when $r = 2$, and counterexamples can be constructed for $r > 2$.[5]

Re-formulation

For $r = 3$, the statement can be rewritten as

$$(\lambda_1 - \lambda_3)(p_1 - q_1) + (\lambda_2 - \lambda_3)(p_2 - q_2) \leq 0 \overset{?}{\iff} \lambda_2(\lambda_1 - \lambda_3)(p_1 - q_1) + \lambda_1(\lambda_2 - \lambda_3)(p_2 - q_2) \leq 0$$

Without loss of generality, we can say $\lambda_1 \leq \cdots \leq \lambda_r \implies \lambda_1 - \lambda_3 \leq 0$ and $\lambda_2 - \lambda_3 \leq 0$.

For arbitrary r, we have the statement

$$\sum_i^{r-1} (\lambda_i - \lambda_r)(p_i - q_i) \leq 0 \iff \sum_i^{r-1} (\prod_{j \neq i} \lambda_j)(\lambda_i - \lambda_r)(p_i - q_i) \leq 0$$

Some possible solutions can be ascertained, but it's difficult to relate those solutions back to what L should have to look like.

Expectations

Defining random variables X and Y such that $P(X = \lambda_i) = p_i$ and $P(Y = \lambda_i) = q_i$, we can rewrite the statement as

$$E[X] \leq E[Y] \overset{?}{\iff} E[X^{-1}] \geq E[Y^{-1}]$$

Using Jensen's inequality, we can say $E[X^{-1}] \geq (E[X])^{-1}$ and $E[Y^{-1}] \geq (E[Y])^{-1}$. We also have $E[X] \leq E[Y] \iff (E[X])^{-1} \geq (E[Y])^{-1}$. This unfortunately doesn't get us anywhere.

We can also note that $E[X^{-1}]$ is the reciprocal of the harmonic mean of X.

Kernels

Can we find some other kernel matrix K such that $\arg\min_H \operatorname{Tr}(H^\top LH) = \arg\max_H \operatorname{Tr}(H^\top KH)$? It has been shown[6] that if we define $K = \sigma I - L$, the equality holds, and if we set $\sigma \geq \lambda_{n-1}$, the largest eigenvalue of L, then K is positive semidefinite, so it is possible to embed fully. However, as σ increases, Lloyd's algorithm has more difficulty in finding the global optimum. One proposed workaround is to set σ to a smaller value and perform Lloyd's algorithm on the partial embedding.

Misc.

Ling and Strohmer[7] showed that if we change the relaxation from $H^\top H = I$ to $H^\top H = I$ and $h_{ij} \geq 0$ $i <= n$, $j \leq k$, there exists a condition under which performing k-means clustering on the embedding induced by $H^* = \arg\min_H \operatorname{Tr}(H^\top LH)$ results in the solution to ratio cut. They do not provide a solution to this optimization problem, but they note that it is a semidefinite programming problem.

[5] file:///home/johnkoo/dev/summer-research-2018/inequality.pdf
[6] http://people.bu.edu/bkulis/pubs/spectral_techreport.pdf
[7] https://arxiv.org/pdf/1806.11429.pdf

Going back to the counterexample graph from before[8], we have the following facts:

- The characteristic polynomial equation for the graph Laplacian is $\lambda(\lambda^3-(4-2\epsilon)\lambda^2+(e+7\epsilon)\lambda-4\epsilon)=0$[9]. This has one obvious root at $\lambda_0=0$, as expected, and the other roots are difficult to parse.
- It can be shown that $[H^\top LH]_{ii}=\frac{W(C_i,C_i^c)}{n_i}$, so the trace works out nicely to the ratio cut objective. This statement is true in general, not just for this particular graph.
- I could not find any such relationship for $H^\top L^\dagger H$. The following can be shown:
  - For the 1-2 cut, $[H^\top L^\dagger H]_{11}=\frac{5}{16}+\frac{9}{16\epsilon}$, and $[H^\top L^\dagger H]_{22}=\frac{5}{48}+\frac{3}{16\epsilon}$.
  - For the 2-3 cut, $[H^\top L^\dagger H]_{11}=\frac{5}{8}+\frac{1}{8\epsilon}$, and $[H^\top L^\dagger H]_{22}=\frac{5}{8}+\frac{1}{8\epsilon}$.
  - For the 3-4 cut, $[H^\top L^\dagger H]_{11}=\frac{13}{48}+\frac{1}{48\epsilon}$, and $[H^\top L^\dagger H]_{22}=\frac{13}{16}+\frac{1}{16\epsilon}$.
- $(H^\top LH)^\dagger=f(\epsilon)(H^\top L^\dagger H)$. Similarly, $(H^\top L^\dagger H)^\dagger=g(\epsilon)(H^\top LH)$. So $\arg\min_H \mathrm{Tr}(H^\top LH)=\arg\min_H g(\epsilon)\mathrm{Tr}((H^\top L^\dagger H)^\dagger)$ and $\arg\max_H \mathrm{Tr}(H^\top L^\dagger H)=f(\epsilon)\arg\max_H \mathrm{Tr}((H^\top LH)^\dagger)$.

[8] http://pages.iu.edu/~mtrosset/Courses/675/notes.pdf#page=128

[9] I believe there's an error in the textbook