CPSC 320 Notes, Clustering Completed

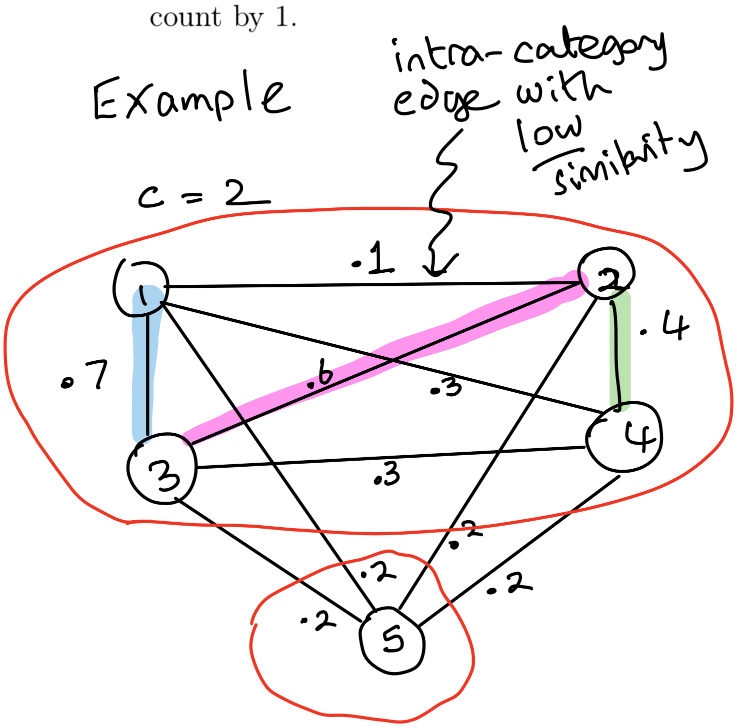
AS BEFORE: We're given a complete, weighted, undirected graph $G = (V, E)$ represented as an adjacency list, where the weights are all between 0 and 1 and represent similarities—the higher the more similar—and a desired number $1 \le k \le |V|$ of categories.

We define the similarity between two categories C_1 and C_2 to be the maximum similarity between any pair of nodes $p_1 \in C_1$ and $p_2 \in C_2$. We must produce the categorization—partition into k (non-empty) sets—that minimizes the maximum similarity between categories.

Now, we'll prove this greedy approach optimal.

1. Sort a list of the edges E in decreasing order by similarity.
2. Initialize each node as its own category.
3. Initialize the category count to $|V| = n$
4. While we have more than k categories:
   (a) Remove the highest similarity edge (u, v) from the list.
   (b) If u and v are not in the same category: Merge u's and v's categories, and reduce the category count by 1.

Example



While loop iteration	Categorization
0	$\{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}\}$
1	$\{\{1, 3\}, \{2\}, \{4\}, \{5\}\}$
2	$\{\{1, 3, 2\}, \{4\}, \{5\}\}$
3	$\{\{1, 3, 2, 4\}, \{5\}\}$

Cost : .2

CPSC 320 Notes, Clustering Completed

AS BEFORE: We're given a complete, weighted, undirected graph $G = (V, E)$ represented as an adjacency list, where the weights are all between 0 and 1 and represent similarities—the higher the more similar—and a desired number $1 \leq k \leq |V|$ of categories.

We define the similarity between two categories C_1 and C_2 to be the maximum similarity between any pair of nodes $p_1 \in C_1$ and $p_2 \in C_2$. We must produce the categorization—partition into k (non-empty) sets—that minimizes the maximum similarity between categories.

Now, we'll prove this greedy approach optimal.

1. Sort a list of the edges E in decreasing order by similarity.
2. Initialize each node as its own category.
3. Initialize the category count to $|V|$.
4. While we have more than k categories:
   (a) Remove the highest similarity edge (u, v) from the list.
   (b) If u and v are not in the same category: Merge u's and v's categories, and reduce the category count by 1.

1 Greedy is at least as good as Optimal

We'll start by noting that any solution to this problem partitions the edges into the "intra-category" edges (those that connect nodes within a category) and the "inter-category" edges (those that cross categories).

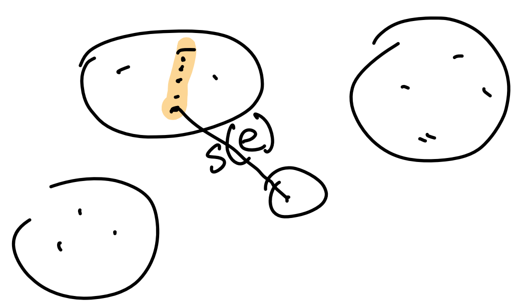
1. **Getting to know the terminology:** Imagine we're looking at a categorization produced by our algorithm in which the inter-category edge with maximum similarity is e.

   Can our greedy algorithm's solution have an intra-category edge with **lower** weight than e? Either draw an example in which this can happen, or sketch a proof that it cannot.

From worksheet question 1.1: Can our greedy algorithm's solution have an intra-category edge with lower weight than e?

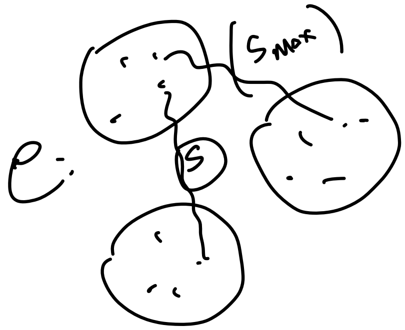
A. Yes
B. No

See example on previous page, where the



large category contains an edge with similarity .1.

2. Give a bound—indicating whether it's an upper- or lower-bound—on the maximum similarity of an arbitrary categorization $\mathcal{C}$ in terms of any one of its inter-category edge weights. That is, I tell you that $\mathcal{C}$ has an inter-category edge with weight s. How much can you tell me so far about Cost($\mathcal{C}$)?



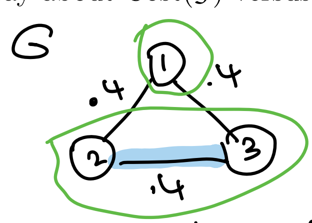
$\text{Cost}(\mathcal{C}) \geq s$

3. Let $\mathcal{G}$ be the categorization produced by our greedy algorithm, and let $\mathcal{O}$ be an optimal categorization on that instance. Let E' be the set of edges removed from the list during iterations of the While loop. With respect to the greedy solution $\mathcal{G}$, are the edges in E' inter-category? Or intra-category? Or could both types of edges be in E'?

> 1. Sort a list of the edges E in decreasing order by similarity.
> 2. Initialize each node as its own category.
> 3. Initialize the category count to $|V|$.
> 4. While we have more than k categories:
>   (a) Remove the highest similarity edge (u, v) from the list. ← (put edge into E')
>   (b) If u and v are not in the same category: Merge u's and v's categories, and reduce the category count by 1.

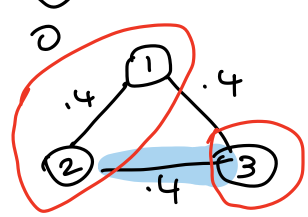
All edges in E' are intra-category.

4. Suppose that some edge $e = (p, p', s)$ of E' is inter-category in the optimal solution $\mathcal{O}$. What can we say about Cost($\mathcal{G}$) versus Cost($\mathcal{O}$)?



intracategory in G.

c=2



inter-category in O

Claim: $\text{Cost}(\mathcal{G}) \leq \text{Cost}(\mathcal{O})$.

Proof: $\text{Cost}(\mathcal{G}) \leq s \leq \text{Cost}(\mathcal{O})$

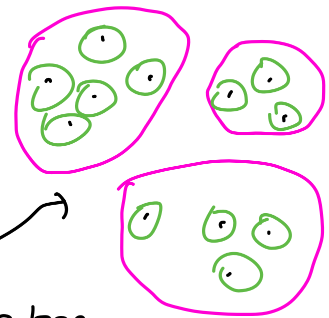
because Greedy makes all edges with similarity $\geq s$ intra-category, by part 3.

from part 2

5. Suppose that all edges of E' are intra-category not only in $\mathcal{G}$, but also in the optimal solution $\mathcal{O}$.

Claim: In this case, the clusters created at every iteration of Greedy's while loop are subsets of the clusters in $\mathcal{O}$.

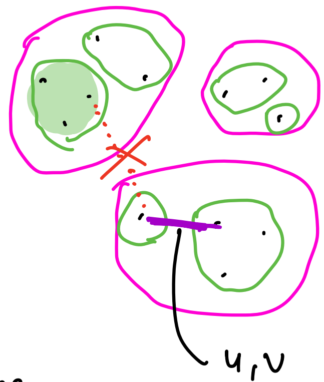
1. Sort a list of the edges E in decreasing order by similarity.
2. Initialize each node as its own category.
3. Initialize the category count to $|V|$.
4. While we have more than k categories:
   (a) Remove the highest similarity edge (u, v) from the list.
   (b) If u and v are not in the same category: Merge u's and v's categories, and reduce the category count by 1.



Proof: By induction on # iterations of while loop.

Base case: 0 iterations.

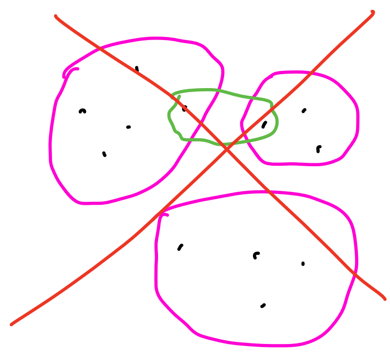
Ind. Hyp: Suppose that after i iterations, Greedy's clusters are subsets of $\mathcal{O}$'s clusters



Ind. Step: Show same is true after $i+1$ iterations.

Let $e = (u,v)$ be edge chosen by greedy at iteration $i+1$. Let C_u, C_v be the clusters containing u, v, respectively.

- If u, v happen to be in same cluster, nothing changes in iteration $i+1$, so we're done.
- If u and v are in different clusters: we know (u,v) is intra-category in $\mathcal{O}$. So C_u, C_v are subsets of the same category of $\mathcal{O}$. So the merged category $C_u \cup C_v$ is still a subset of a category of $\mathcal{O}$, and we're done.



5. Suppose that all edges of E' are intra-category not only in $\mathcal{G}$, but also in the optimal solution $\mathcal{O}$. Can there be any edges that are inter-category in $\mathcal{G}$ but intra-category in $\mathcal{O}$?

Clicker Question #3

From worksheet question 1.5: Can there be any edges that are inter-category in $\mathcal{G}$ but intra-category in $\mathcal{O}$?

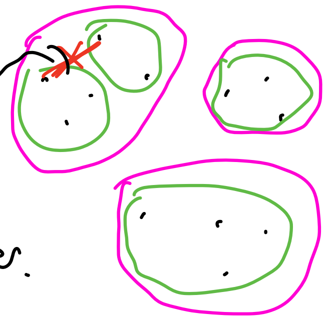
A. Yes

B. No

Hint: Keep in mind, from the claim that we just proved, that every category of Greedy is contained in a category of $\mathcal{O}$.

$c = 3$

Because if there were an intercategory edge of $\mathcal{G}$ inside a cluster of $\mathcal{O}$, then either $\mathcal{G}$ has too many clusters, or $\mathcal{O}$ has an empty cluster.



5. Suppose that all edges of E' are intra-category not only in $\mathcal{G}$, but also in the optimal solution $\mathcal{O}$. Can there be any edges that are inter-category in $\mathcal{G}$ but intra-category in $\mathcal{O}$? (Hint: imagine you have a solution produced by the greedy algorithm. Can you convert any of its inter-category edges to intra-category edges without either making some edges in E' inter-category or making your solution invalid?)

Clicker Question #3

From worksheet question 1.5: Can there be any edges that are inter-category in $\mathcal{G}$ but intra-category in $\mathcal{O}$?

A. Yes
B. No

6. Apply the progress made in parts 3 to 5 to conclude that $\mathcal{G}$ must be an optimal solution.