

For task 1, you are asked to run the CELF algorithm on the attached mystery graph. Each vertex in the graph has a “class” attribute, which takes one of the values (a, b, c or d).

You need to modify the CELF algorithm to take into consideration vertex classes. In other words, sorting the vertices according to influence will have a composite criterion of both number of activation and class of activated nodes. For example:

If v1 can activate 5 neighbors, all of which belong to class “a”, vertex v2 is more influential if it can activate the same count of neighbors (5) if these neighbors belong to 2 classes or more.

Implementation tips

- You may use the below method to load the pickle format attached into a graph:

```
import igraph  
  
Graph.Read_Pickle(fname="MysteryGraph")
```
- You may reference the CELF and Independent Cascade model implementations available from https://hautahi.com/im_greedyself
- Use probability of influence $p = 0.3$ and Monte Carlo repetitions = 500
- Attempt number of seeds = 1, 2, 4

Submission

Submit a well-formatted notebook with your code and output.

Based on the ID of the seed nodes, try to infer the underlying structure of the graph.

Bonus

Plot the graph and the seed nodes to validate your findings about its underlying structure.