### Analysis 7 Community Detection Instructions

### Construct your solution using Lastname-Analysis-7-Community-Detection.Rmd

(You will do some of the work in Gephi, but include screen dumps in the R Markdown document.)

### Final version due in Laulima the evening of Monday April 11th.

### 75 points

### Preliminary Comments

Community detection methods are one of the most important methods taught this semester.

This assignment on community detection has two parts:

* Continuing our analysis of the Primary School data from Analysis 6, done mostly in R but ending up in Gephi. (If you did not do Analysis 6, you can find the solution in our class Google folder for parts you will need.)
* Examining the community structure of a large graph, Comic Hero co-appearances, in Gephi, which I hope you will find to be fun.

The purpose of this assignment is to give you practice computing and comparing different community detection methods, and interpreting results. In this interpretation, we will examine node level metrics and attributes as well as community structure.

### Primary School Contact Network (46 pts)

In last week's Analysis 6 Assortativity assignment, I asked you to filter low count and duration contacts, and remove Teachers from the network so we could focus on students’ nontrivial contact with each other. There are some changes for this assignment:

* Now we want to study community structure, including the roles of teachers. Therefore we will leave the teachers in.
* In Analysis 6 we filtered low count and duration contacts to remove the influence of spurious contacts. A revised filtering policy will be applied here.
* Since Teachers are now included, the numbers differ: you should have 242 vertices, 2759 edges, max count = 278, and max duration = 15280
* We will make a Grade attribute just like in Analysis 6. To unclutter the visualization we will also shorten labels: “Unknown” becomes “U” and “Teachers” becomes “T”.

The good news is that I have already done all of this for you so that everyone will start with the same filtered network.

### 1. Prepare the Primary School network with Teachers (0 pts)

Since this only a minor variation on the last assignment, and we want to be sure everyone has the same starting point, I will provide the code to prepare an igraph called PSF as follows (so there are no points for this section):

* Reading sp\_data\_school\_day\_1\_2\_multigraph.graphml constructed by downloading two days of .gexf data from the web site, combining it in Gephi, and exporting as .graphml (the first step of Analysis 6).
* Deleting edges for trivial or spurious interactions, as discussed in the document.
* Making a new vertex attribute called $grade and give it appropriate values by conversion of the 'classname' attribute, as we did before. Note that we now have a grade “T” for Teachers, since teachers have classname “Teachers”. (Something to think about: if we wanted to compute assortativity on grade, how would we handle this value?)
* Shortening attributes for briefer visualization in Gephi: gender “Unknown” becomes “U” and classname “Teachers” becomes “T”.

### 2. Computing and Comparing Partitions (22 pts)

Compute partitions using the InfoMap and Louvain community detection methods, and compare them to each other and two node attributes, Classname and Grade. All of these partition the network. We ask: how well does each partitioning capture the connectivity of the nodes? How similar are the partitions to each other?

**(a) Computing Community Structure Weighted by Duration (4)**

Compute cluster\_louvain and cluster\_infomap partitions of both graphs, using E(PSF)$duration as the weights and saving the resulting community objects for use below. Display the number of clusters and the sizes of the clusters for each method (see ?membership documentation).

**(b) Comparison by Modularity and Assortativity (8)**

Prepare a table of the following values. An R template will be provided using two tibbles that are row-bound together. (4)

| metric | Classname | Grade | InfoMap | Louvain |
| --- | --- | --- | --- | --- |
| Modularity |  |  |  |  |
| Assortativity |  |  |  |  |

Discuss the results:

**Explain rank ordering of Assortativity and Modularity:** Assortativity is Modularity scaled such that 1 is the maximum value it can have. We might expect that this means that assortativity is always larger, yet we see that this is not the case for Louvain and InfoMap. Explain this. (There are two reasons, one having to do with how we computed the partitions, and the other, which requires further thought, having to do with how assortativity is normalized.) (2)

**Discuss how well the partitions reflect network connectivity:** Which of the partitions best account for the connectivity of the network? Consider unweighted and weighted connectivity. *Note: this is a different discussion than the discussion in (c) below. Here we discuss how well the partitions match the network; below we discuss how well the partitions match each other.* (2)

**(c) Comparison by Normalized Mutual Information (10)**

Compare the partitions to each other using Normalized Mutual Information, displaying the results in a table of this form. (It is your choice of how to make the table, whether using the strategy above or something else.) (8)

| partition | Classname | Grade | InfoMap | Louvain |
| --- | --- | --- | --- | --- |
| Classname |  |  |  |  |
| Grade |  |  |  |  |
| InfoMap |  |  |  |  |
| Louvain |  |  |  |  |

Discuss the results: Which of the partition methods give similar results, and why? (We investigate this further in question 3.) (2)

*Comment:* Here we use assortativity coefficients essentially to test various hypotheses about what is responsible for the connectivity (link structure) of the network: degree, gender, grade, classname, or something else? It is likely that more than one of these factors contribute to the connectivity, but the above tools only let us test them individually. It would be nice to have a method by which we can test them simultaneously, and model their relative contributions. This is exactly what Exponential Random Graph Models attempt to do. Due to the complexity of ERGMs, I no longer cover them in this introductory course.

### 3. Examining Results in Gephi (24 pts)

**(a) Prepare and write out results for Gephi. (2)** Assign community membership to vertices under attributes $louvain and $infomap. Save the existing edge weights (which are 1 or 2, the number of days each pair had contacts) under a new edge attribute ‘days’. Copy the ‘duration’ attribute to “weight” so that Gephi will visualize edges sized by duration. Check the results by showing the 'summary'. Now write the graph out to a graphml file for inspection in Gephi.

**(b) Spatialized Visualization in Gephi. (2)** Read your annotated PSF into Gephi and label the workspace. Size the nodes by degree and give it a nice layout. (I used Force Atlas 2 with LinLog and Prevent Overlap on: adjust Gravity to 2 or 3 for appropriate compactness. Important: Set Edge Weight Influence to 0; otherwise the large duration weights will compress the layout.) Set the displayed labels to be classname (from which we can infer grade) and gender, (e.g., “5B-F”); turn off Label.

**(c) Visualization of Partitions. (8)** Then color the network using the same color palette to the extent possible applied to the Classname, Grade, InfoMap, and Louvain partitions. (For the Classname and Grade colorings, I suggest you make Teachers (T) yellow so they are easily identified.) Make and include screen dumps of these four colorings as specified in the .Rmd. (Don’t try to make PDF: it is too much hassle copying grade and duration together into Label to get it labeled, and the edge thickness is not displayed well.) Also include the partition color keys by taking a screen snap of the Appearance panel in Gephi. You should present the visualizations in an order that makes it easy to see similarities.

Comment: We don’t need to compute Gephi Modularity. I verified that with edge weights copied as above it gives identical results as cluster\_louvain.

Include the resulting Gephi project in your uploaded analysis folder.

**(d) Compare partitioning methods. (6)** Using the tables you computed above (modularity and NMI) and the visualizations, compare the partitions and discuss how their differences and similarities make sense. Specifically:

* How does igraph’s cluster\_louvain compare to igraph’s cluster\_infomap? Describe and explain any differences.
* How do both of these compare to the Classname Grade partitions? Describe and explain differences. (You should see a correspondence.)

These explanations can be purely in terms of the partitioning algorithms and network structure: part (d) will include actor behavior.

**(e) Draw conclusions about behavior in PSF.** **(6)** Examining the community partitions, and also examining ego node degree and connectivity across partitions, try to draw conclusions about student and teacher behavior that would be of interest to educators and school supervisors. For example:

* Observations about Collective Behavior: How do students associate in general? Which grade levels mix across classes and at which grade levels are classes more distinct?
* Observations about Salient Individuals or Pairs: Are there any individuals who have unusual patterns of connectivity? Identify them and describe what is unusual about them.

Include supporting images for full credit.

Reminder: Identifying the network results is not sufficient: translate it into hypotheses about the behavior of real people.

### Interpreting Partitions as "Communities" in a Large Network (29 pts)

In this exercise you'll examine the community structure of a large network in Gephi, and also identify individuals playing special roles beyond the community partitions. We will use **Hero-Coappearance-Network.graphml** (|V| = 6439, |E| = 167103, weighted undirected). **DO NOT USE comic-hero-network.graphml!!!** That was a bipartite graph you used on a previous assignment. Hero-Coappearance-Network.graphml is its projection onto Hero actors. Weights indicate the number of issues in which the heroes co-appeared.

Basic knowledge about comic book heroes may be needed to make sense of the results: use Google or Wikipedia as needed if you don't have that background.

Note: the errors we found in comic-hero-network.graphml have been corrected before making Hero-Coappearance-Network.graphml.

### 4. Prepare the Network (4 pts)

We are going to use both Louvain and Infomap partitioning, so we start in igraph.

Load the network into igraph and prepare it as follows.

* Compute the weighted Louvain and InfoMap partitionings of this graph and save the community objects for use below and in question 5.
* Annotate vertices with degree, strength and page rank. (These will be global vertex attributes; in Gephi we'll compute them within each partition.)
* Annotate vertices with their Louvain and InfoMap partitions.
* Show the result with summary and check that attributes are present.
* Now write the network out as a new .graphml to be read into Gephi.

### 5. Finding Crossing Vertices (8 pts)

Before we go to Gephi, find the top 20 crossing vertices, those that are on many edges that cross partitions. Print the crossing vertices and the number of edges via which they connect to others outside their partition. Do this for both Louvain and Infomap.

*Hints:* Use 'crossing' to find the crossing edges, 'ends' to get the vertices on the ends, then make a frequency table using $id and sort by frequency in decreasing order. Use head to show the top 20.

### 6. Interpreting Communities in Gephi (17 pts)

**(a) Load the annotated network in Gephi.** (0 points but see Note)

Visualization of the full graph that aids you in interpretation in part (b) is optional and not turned in. If you want to use a global visualization, I suggest the following:

* Do an OpenOrd layout as it works on large graphs.
* Size the graph nodes by one of the centrality metrics.
* Follow up with Force Atlas 2, Lin Log mode, Scaling 1, Gravity 10, and turn on Prevent Overlap once it is stable.
* Turn on the node labels
* Color the graph nodes by their Infomap and Louvain partitions. (You may limit colors to the top 8-10 largest partitions, as we'll focus on these.)

Please save your Gephi project when this is done and include it in your folder.

Note: Once you find partitions of interest, you are required to show something from Gephi: either redo the layout of the partition with the filter on, or take a screen selection of the top ranked nodes in Data Laboratory, or both. In either case, include the image (.png, no PDF required) in your report below for credit.

**(b) Analyze 3 of the partitions.** **(15)**

Using Filters and Data Laboratory, explore several of the partitions under each of Louvain and Infomap to find ones that make sense to you. If you are not an expert on the comic hero world I suggest you look at the larger partitions, as you will probably recognize the main characters. If you are an expert you might be able to interpret smaller partitions.

It is your choice whether to use Louvain or Infomap! I just want you to explore a little. Louvain has larger partitions that gather together many related characters, but they can also be less focused than Infomap. Then pick 3 partitions of those you examined to discuss.

*Suggested Method:* Under Filters/Attributes/Partition you will find a filter for each partitioning.

* Select a single partition and turn on the filter.
* Compute Degree and PageRank with weights *within* the partition.
* Optionally ForceAtlas 2 with Prevent Overlap *briefly* (a second or two) to pull the partition together and get the nodes off of each other.
* Examine the visualization to see the structure of the partition.
* Go to Data Laboratory, sort by the within-partition PageRank, and observe the highest ranked nodes.
* Based on both what you saw in the visualization and Data Laboratory, interpret how the nodes make sense as a "community" in terms of the domain being modeled. Why do these characters belong together thematically? (If you are not familiar with the heroes franchises, you may need to google names.) Try to make sense of not just the first few nodes, but what most of the higher ranked nodes have in common.
* Also notice whether any of the top nodes in the partition are also crossing vertices as we computed in igraph. You can also identify crossing vertices directly in Gephi by comparing the 'degree' that was computed in the global graph in igraph to the 'Degree' that was computed in Gephi with the partition filter applied. The difference degree - Degree should be the number of external links.

*Note:* sometimes when you switch between Filters and Statistics the partition filter list of checkboxes is lost and you have to remove and reinstate it.

After you explore several partitions, **choose 3 and write up your interpretation of what the community is, supported with evidence**, which may be images of either visualizations of the partitions or the first 10-12 lines in your Data Laboratory sorted by the centrality metric you found to be most informative.

**(c) Discuss the implications of crossing vertices. (2)** You should have found that some of the top nodes in these large partitions also have many links to other partitions. Interpret them.

* Thinking back to the bipartite graph from which this projection was derived, gIve a plausible explanation of what crossing edges/vertices mean in terms of the role these comic heroes play in a collection of comic issues.
* Identify a limitation of community detection by partitioning that the existence of crossing edges for important characters illustrates.