Tutorial: Clustering of Network data

MAP573 – Introduction to unsupervised learning

École Polytechnique - Autumn 2019

Preliminaries

Goals.

1. Basic network manipulations with `igraph`, vizualization, descriptive statistics
2. Graph partitioning: hierarchical clustering and spectral clustering algorithms
3. Analysis of the network of French political blogs

Required packages. Check that the following packages are available on your computer:

```
library(igraph)
library(sand)
library(Matrix)
library(aricode)
```

You also need `Rstudio`, LaTeX and packages for markdown:

```
library(knitr)
library(rmarkdown)
```

If(and only if!!) you have time, you can also play with

```
library(ggraph)
```

1 Introduction to **igraph**

1.1 Tutorials

Have a glance at these two tutorials

- an **igraph** tutorial for graphs manipulation.
- network analysis with **igraph** that gives an overview of the standard features of **igraph** to perform basic statistical analysis on network graphs.

2 Analysis of the French political Blogs in 2006

Load the data set and upgrade to the current version of **igraph**

```
blogosphere2006 <- upgrade_graph(fblog)
```

2.1 Graph Partitioning

It is time to find some subtle structure in the network by performing graph clustering of the nodes. A commonly seeked structure in networks is the presence of *communities*, like clusters of friends in social networks.

The two methods explored today for community detection do not assume any underlying model on the graph. The first one is based on a generalization of hierarchical clustering for graphs using the concept of *modularity* to define an appropriate dissimilarity measure between clusters. The second is the normalized spectral clustering.

2.1.1 Hierarchical clustering with modularity

Just like in usual hierarchical clustering, we need some cost function to choose which clusters are fused during construction of the hierarchy. Let $\mathcal{C} = \{C_1, \dots, C_K\}$ be a candidate partition and define $f_{ij}(\mathcal{C})$ to be the fraction of edges in the original network that connect vertices in C_i with vertices in C_j. The modularity of $\mathcal{C}$ is the value

$$\text{modularity}(\mathcal{C}) = \sum_{k=1}^{K} (f_{kk}(\mathcal{C}) - f_{kk}^{\star})^2$$

where $f_{kk}^{\star}$ is the expected value of f_{kk} under some model of random edge assignment. The `igraph::fastgreedy.community` function performs and approximated optimization of the modularity measure.

Use this function to extract a possible clustering of the nodes for the French blogosophere. Use the plot function for object with class `communities` outputing from `igraph::fastgreedy.community`. Compare this clustering with the political labels of the nodes (use for instance confusion tables with `table` or adjusted Rand-Index with `aricode::ARI`).

Explore the results offered by other clustering methods for community detection (e.g. `igraph::cluster_edge_betweenness`), or the ones obtained by a simple hierarchical clustering on dissimilarity measured on the adjacency matrix.

2.1.2 Spectral Analysis

Here is a reminder of the spectral clustering algorithm presented during course.

Algorithm 1: Spectral Clustering by Ng, Jordan and Weiss (2002)

```
Input: Adjacency matrix and number of classes $Q$
Compute the normalized graph Laplacian $\mathbf{L}$
Compute the eigen vectors of $\mathbf{L}$ associated with the $Q$ smallest eigenvalues
Define $\mathbf{U}$, the $p \times Q$ matrix that encompasses these $Q$ vectors
Define $\tilde{\mathbf{U}}$, the row-wise normalized version of $\mathbf{U}$: $\tilde{u}_{ij} = \frac{u_{ij}}{\|\mathbf{U}_i\|_2}$
Apply k-means to $(\tilde{\mathbf{U}}_i)_{i=1,\dots,p}$
Output: vector of classes $\mathbf{C} \in \mathcal{Q}^p$, such as $C_i = q$ if $i \in q$
```

1. Compute the graph Laplacian (normalized of unormalized) by hands or with the `igraph::graph.laplacian` function.
2. Compute its eigen values and represent the scree plot (eigen values by increasing order). Comment.
3. Compute its eigen vectors and represent the pairs plot between the first 10 eigen vectors with non-null eigen values. Add colors associated to the Political labels of the nodes. Comment.
4. Implement the Spectral clustering algorithm and apply it to the French Blogosphere for various numbers of clusters.
5. Compare your clustering to the political labels and to the one obtained by hierarchical clustering. Comment, make plots changing node colors, etc.
6. Redo the analysis with the absolute spectral clustering of Rohe et al (2011).

Remark: Another fancy alternative to the **igraph** package is the `Matrix::image` or `corrplot` methods on the adjacency matrix of a graph once rows and columns reordered according to the clustering to represent the clustered network.