Graph Clustering and Community Detection

Working with distance graphs

Matthieu Pierre Boyer*
École Normale Supérieure
Paris, France
matthieu.boyer@ens.fr

Martin Cuingnet*
École Normale Supérieure – Paris-Saclay
Gif-sur-Yvette, France
martin.cuingnet@ens-paris-saclay.fr

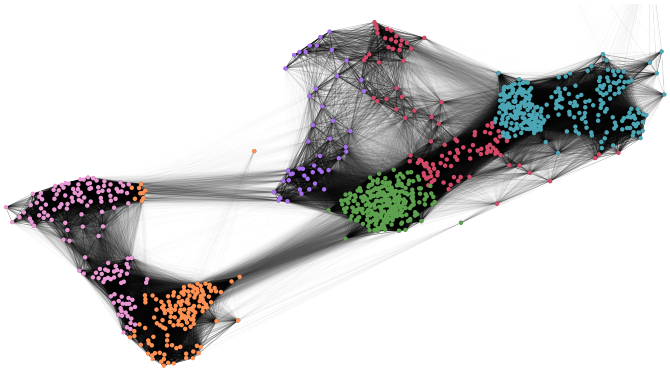


Figure 1: World cities distance graph clustered using the EM wERMG algorithm. (Q = 6)

ABSTRACT

In this report we are interested in extending *graph clustering* methods designed in [2] with a probabilistic graphical model, to weighted graphs. This will allow us to define a graphical model for clustering on manifold-like graphs, at the cost of a bit of complexity and an added free parameter.

INTRODUCTION

The problem of graph clustering, also known as community detection, concerns partitions of the vertices of a graph into groups (clusters or communities) such that vertices within the same group are more densely connected to each other than to vertices in other groups. This fundamental problem arises across diverse domains: identifying functional modules in biological networks, detecting communities in social networks, discovering related documents in information retrieval systems, and analyzing interaction patterns in complex systems.

Traditional approaches to graph clustering often rely on heuristic methods or spectral techniques. However, these approaches typically lack a probabilistic foundation that would enable principled statistical inference, model selection, and uncertainty quantification. The mixture model framework that we extend below offers a compelling alternative by providing a rigorous statistical foundation for understanding and detecting community structure in networks.

The paper by Daudin, Picard and Robin ([2]) introduces a mixture of probability distributions used to model the classical Erdös-Rényi model of random graphs, allowing for the traditional EM algorithm to do statistical inference on the parameters of the graph, as well as a better model for networks which appear most importantly in biology. In this report, we suggest a possible method of providingg the model to possibly take into account less well defined connections and especially weighted graphs.

We refer to [6] for a more detailed overview on graph clustering.

Related Work

This work was only based on[1] [2], although other people have worked on the matter. One could for example cite [3] which present a gaussian mixture model based on the extension of [2]'s bernoulli by modifying those to use beta laws, [5] which use laws in the exponential family to model weight distributions on counts of interactions. Although we will in the end present a similar solution to simplify our inference procedure, it is important to note that our model and algorithm is more general. In a similar fashion, [4] propose a gaussian mixture model for the weights and apply the EM to optimize on the parameters.

1 MIXTURE MODELS FOR CLUSTERING OF WEIGHTED GRAPHS

The base random graph model we work on is the following: Given a set V of nodes (usually $V = \{1, \dots, n\}$), we generate, at random, edges between nodes based on a probability distribution to mimic clustering behaviours that might be found in real life. The goal is to find a parametric law for edge generation which allows us to correctly generate clusters.

The *Erdös-Rényi Mixture for Graphs* model, as defined in [2], is a probabilistic graphical model which proposes distributions of edges

*Both authors contributed equally.

[1] Our goal was to expand [2] by making our own model, to consider a specific class of problems. As such, we decided to not read any literature on the subject before completing the mathematical form of the model.

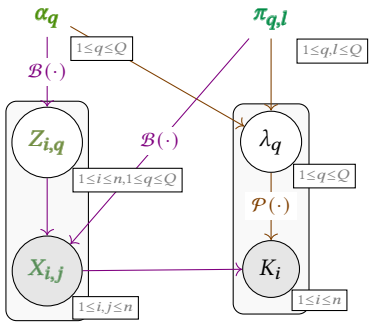


Figure 2: Graphical model for ERMG

in a graph depending on the number of clusters it exhibits, and how *distinct* those clusters are. The goal is to find a law for the *between group connectivity* and *aggregation factor* (how much is the graph composed of clusters), which reproduces the distribution of degrees in biological data. In the following paragraphs, we will re-present the notations introduced in [2], and modify those to extend their usage to *weighted* graphs in our *wERMG* model, trying to account for a better representation.

1.1 ERMG model

The ERMG model admits the graphical model representation in Figure 2 and is such that the diagram formed by violet and brown paths is commutative.

In the model by [2]:

- $Z_{i,q} \sim \mathcal{B}(\alpha_q)$ is the probability that node i is a member of cluster q;
- $X_{i,j} \sim \sum_{q,l} Z_{i,q}Z_{j,l}\mathcal{B}\left(\pi_{q,l}\right)$ is the probability that nodes i and j are connected with an edge;
- $\lambda_q = (n-1)\sum_l \alpha_l\pi_{q,l}$ is the right parameter for approximation in degree definitions;
- K_i is the degree of node i;

The edge in the graph model between K_i and λ_q comes from the Poisson approximation of the binomial law:

$$K_i \sim \mathcal{B}\left(n-1, \sum_l \alpha_l\pi_{q,l}\right) \xrightarrow[n\to\infty]{\text{law}} \mathcal{P}\left(\lambda_q\right)$$

The probability distribution for K_i is not actually defined in that sense, but is actually defined through the $X_{i,j}$, with the parameters for degrees observed being the one that need fit for actual data. This fact will be the corner stone of our extension to weighted graphs, but first, let us remind the version of the EM algorithm used for optimization of the parameters α_q and $\pi_{q,l}$ based on observed data $X_{i,j} = \mathcal{X}$:

E Step. First, we approximate that the joint distribution of the $Z_{i,q}$ is the product of conditional distributions given the other coordinates. Let $\mathcal{Z}_i = \{Z_{i,q}\}_{1\le q\le Q}$ and $\mathcal{Z}^i = \mathcal{Z} \setminus \mathcal{Z}_i$ where $\mathcal{Z} = \{Z_{i,q}\}_{i,q}$.

$$\mathbb{P}\left(Z_{i,q} \mid \mathcal{X}\right) = \prod_i \mathbb{P}\left(\mathcal{Z}_i \mid \mathcal{X}, \mathcal{Z}^i\right).$$

We then iterate until convergence to define $\mathbb{P}(Z_{i,q})$ over the fact that:

$$\begin{aligned}\widehat{\tau_{i,q}} &= \mathbb{P}\left(Z_{i,q} = 1 \,\middle|\, \mathcal{Z}^i\right) \\ &\propto \alpha_q \prod_m b\left(\sum_k Z_{km}X_{i,k}; \sum_{j\neq i} Z_{j,m}, \pi_{q,m}\right)\end{aligned}$$

where $b(C, N, \pi) = \pi^C(1-\pi)^{N-C}$ is the bernoulli likelihood, a continuous version of the binomial law. This is actually an approximation, which holds when n goes to infinity.

M Step. We modify the values of the parameters as follows to maximize the complete-data log-likelihood:

$$\widehat{\alpha_q} = \sum_i \frac{\widehat{\tau_{i,q}}}{n} \tag{M-α}$$

$$\widehat{\pi_{q,l}} = \frac{\sum_i \sum_j \widehat{\tau_{i,q}}\widehat{\tau_{j,l}}X_{i,j}}{\sum_i \sum_j \widehat{\tau_{i,q}}\widehat{\tau_{j,l}}} \tag{M-π}$$

1.2 wERMG model

The main issue with the above model is that, in real life, graphs edges model continuous similarities/proximities instead of purely binary settings. For example a network of all social media posts with edges weighted by the content similarity, or a transportation network with edges weighted by the time it takes to go from a point to another using a single mode of transportation. In turn, this means that the distribution chosen for weights of edges will actually be meaningful only when considering a setting. In the definition of the model, we choose that two points whose similarity or proximity is 0 are not connected in the graph, even when considering edge weights as a measure of distance between nodes. This is done without loss of generality up to post-composition on the weights by a bijection on $\mathbb{R}$ which sends 0 to infinity, and we can always consider weights as similarities instead of distances by the same reasoning.

In the *weighted Erdös-Rényi Mixture for Graphs* model that we propose here, we simply modify the definition of the usage of α_q and $\pi_{q,l}$:

- $Z_{i,q}$, the *clustering probability* of node i being a member of cluster q still follows a Bernoulli law of parameter α_q;
- $X_{i,j} \sim \sum_{q,l} Z_{i,q}Z_{j,l}\mathcal{L}(\pi_{q,l})$ where $\mathcal{L}(\pi_{q,l})$ is any probability distribution such that the collection of $X_{i,j}$s verifies *Lyapunov's condition*[2] is the *weight label of edge* (i, j);
- K_i becomes $\sum_j X_{i,j}$ the *weighted degree* of node i.

Note that again, $\sum_q \alpha_q = 1$.

Proposition 1. *The law of K_i converges to a normal law when n goes to infinity.*

[2] See [1] for an overview of the condition (and a presentation of ERMG's Poisson approximation in this setting).

Proof. We have the following conditional distribution for K_i:

$$\begin{aligned}(K_i \mid i \in Q) &= \sum_j \left(X_{i,j} \,\middle|\, i \in q\right) \\ &= \sum_j \sum_l \left(X_{i,j} \,\middle|\, i \in q \wedge j \in l\right) Z_{j,l} \\ &\sim \sum_j \sum_l \alpha_l \mathcal{L}\left(\pi_{q,l}\right) \\ &= \sum_l n\alpha_l \mathcal{L}\left(\pi_{q,l}\right)\end{aligned}$$

Here, we will make the same approximation as for the ERMG that the $X_{i,j}$ actually are independent when the number of vertices goes to infinity (see the next paragraph for more details). Defining the *empirical standard deviation of the model*:

$$\sigma_n = \sqrt{\sum_j \sum_l \mathbb{V}\left[\alpha_l \mathcal{L}\left(\pi_{q,l}\right)\right]} \overset{\text{def}}{=} \sigma_q \sqrt{n},$$

The above assumption is the only one we need to make, as it implies the correct definition of σ_n and σ_q without having to consider covariances. The Lyapunov central limit theorem then gives :

$$\begin{aligned}&\frac{1}{\sigma_n} \sum_j \sum_l \left(\alpha \mathcal{L}\left(\pi\right) - \mathbb{E}\left[\alpha \mathcal{L}\left(\pi\right)\right]\right) \\ &= \frac{1}{\sigma_n} \sum_j \underbrace{\left(\sum_l \alpha \mathcal{L}\left(\pi\right) - \underbrace{\sum_l \mathbb{E}\left[\alpha \mathcal{L}\left(\pi\right)\right]}_{\overset{\text{def}}{=} \mu_q} \right)}_{\text{independent of } j} \\ &= \frac{1}{\sigma_n} n \left(\sum_l \alpha \mathcal{L}\left(\pi\right) - \mu_q \right) \\ &\xrightarrow[n\to\infty]{\text{law}} \mathcal{N}(0,1)\end{aligned}$$

where $n\mu_q$ is the *empirical mean of the model* and we have used the shorthands $\pi = \pi_{q,l}$ and $\alpha = \alpha_l$.

$$\frac{\sqrt{n}}{\sigma_q}\left(K_i - n\mu_q\right) \xrightarrow[n\to+\infty]{\text{law}} \mathcal{N}(0,1)$$

which, equivalently can be rewritten as:

$$K_i \sim \mathcal{N}\left(n\mu_q, \sqrt{n}\sigma_q\right) = \mathcal{N}\left(\lambda_q\right)$$

□

This, again, should be considered as exactly true only when n is large enough for the independance of the $X_{i,j}$ to be true. Moreover, the actual parameters of the proof provide us with a way to get the result of ERMG by replacing them with parameters that arise from $\mathcal{L} = \mathcal{B}$.

Main differences with ERMG. The main difference that one should consider when comparing our wERMG model to [2]'s ERMG, is that there is no reason that $\pi_{q,l} = \pi_{l,q}$. Indeed, the graph can now be freely oriented, for example considering shortest routes from a town to another which might differ based on the train schedules.

However we still should have a condition on the probability distributions of edges between clusters:

$$\forall q, l, \mathbb{P}\left(\mathcal{L}\left(\pi_{q,l}\right) = 0 \,\middle|\, \mathcal{L}\left(\pi_{l,q}\right) \neq 0\right) = 0$$
$$\text{or equivalently } \mathbb{P}\left(\mathcal{L}\left(\pi_{q,l}\right) \neq 0 \,\middle|\, \mathcal{L}\left(\pi_{l,q}\right) \neq 0\right) = 1$$

This is saying that (almost surely), there are no one-way potential barriers, and there is (almost surely) always a way to go from i to j if there is a way to go from j to i. Really this assumption is purely an assumption to improve the realism of the model in that case, and is not needed for the model to make sense mathematically[3].

The other main difference with ERMG comes with the notion of degree and clusters. In ERMG, the clustering coefficient of i is defined as

$$C_i = \mathbb{E}_{j,k}\left[\mathbb{P}\left(X_{j,k} = 1 \,\middle|\, X_{i,j}X_{i,k} = 1\right)\right]$$

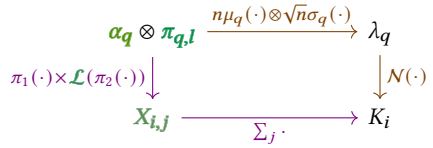
where $j \neq k$ are taken uniformly among the set of vertices.

However, in the above example considering that our graph is modelling travel times, most vertices will be connected but with much higher[4] values for vertices (stations) in different clusters (towns). This means that clustering coefficients and cluster interconnectivity cannot be defined in the same way as for ERMG: lower[5] values are present when clusters are tighter, but even if all points in q and all points in l are connected, they might be with very large distances and thus q and l might be different. To account for this, we suggest the following definition for the clustering coefficient:

$$C_i = \mathbb{E}_{w,j,k}\left[\mathbb{P}\left(X_{j,k} \leq w \,\middle|\, X_{i,j} \leq w \wedge X_{i,k} \leq w\right)\right]$$

where $j \neq k$ are taken uniformly among the set of vertices and w takes values along $\sum \alpha_l \mathcal{L}(\pi_{q,l})$ and has to indicate a connection. Note that we get back ERMG's definition when $X_{j,k}$ follows a Bernoulli law. This definition only makes sense when considering the $\mathcal{L}(\pi_{q,l})$ can be taken on the same physical scale (or compared to a common physical scale), e.g. when we have travel times or cosine similarities in $[0,1]$. Otherwise, the expectation

On the graphical side we have the representation in Figure 3, where again, the diagram formed by violet and brown paths is commutative:



with $\otimes$ the cartesian product and p_1, p_2 the associated projections[6] and μ_q and σ_q as defined in the proof of Proposition 1.

[3] Even worse, it actually adds the need for the approximation of independance on the degrees that would not be needed in a purely directed setting.
[4] We can always say, without loss of generality, that $X_{i,j} = 0$ means that direct travel is impossible when $i \neq j$ and thus distance is infinite.
[5] Again, without loss of generality, one could consider higher edge weights as *tighter* connections.
[6] Usually denoted with π, which would be more confusing than anything here.

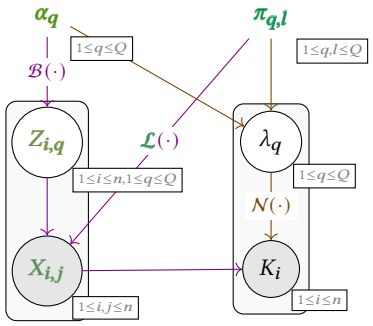


Figure 3: Graphical Model for wERMG

We now have the following version of the EM algorithm, adapted for probability distributions:

E Step. First, we make the same approximation as in ERMG that the joint distribution of the $Z_{i,q}$ is the product of conditional distributions given the other coordinates:

$$\mathbb{P}\left(Z_{i,q} \mid \mathcal{X}\right) = \prod_i \mathbb{P}\left(indiclusti \mid \mathcal{X}, \mathcal{Z}^i\right).$$

We then iterate until convergence to define $\mathbb{P}(Z_{i,q})$ over the following fact:

Proposition 2. *The posterior probabilities τ satisfies the following fixed point relation:*

$$\begin{aligned}\widehat{\tau_{i,q}} &= \mathbb{P}\left(Z_{i,q} = 1 \,\middle|\, \mathcal{X}, \widehat{\mathcal{Z}^i}\right) \\ &\propto \alpha_q \prod_{j \neq i} \prod_l \left[\mathcal{L}\left(\pi_{q,l}\right)\left(X_{i,j}\right)\right]^{Z_{j,l}} \qquad \text{(wE-}\tau\text{)}\end{aligned}$$

We actually need to approximate the variables in $\mathcal{Z}^i$ as equal to their conditional expectations. Unlike for ERMG, there is no way to simplify more the result of this computation until we actually define $\mathcal{L}$.

Proof. To better understand the formula in (wE-τ), see that $x^{Z_{j,l}}$ is equal to 1 (the multiplicative unit) if $j \notin l$ and x if $j \in l$. As such, the product in Equation wE-τ should be understood as the product for all vertices distinct from i of the probability that the edge $X_{i,j}$ as the value it has knowing that j is in cluster l. What this means is that the posterior probability of i being in cluster q is equal (up to renormalization) to the product of the posterior probabilities that i is in q, and that $X_{i,j}$ actually has the observed value knowing that j is in l. This is of course true for the posterior explaining why we have a fixed-point relationship. □

M Step. The complete data log-likelihood is written here as

$$\begin{aligned}Q(\mathcal{X}) &= \mathbb{E}[\log p(\mathcal{X}, \mathcal{Z}) \mid \mathcal{X}] \\ &= \sum_i \sum_q \tau_{i,q} \log \alpha_q \\ &\quad + \sum_{i<j} \sum_{q,l} \theta_{iqjl} \log(\mathcal{L}(\pi_{q,l})(X_{i,j})),\end{aligned}$$

where $\theta_{iqjl} = \tau_{i,q}\tau_{j,l}$. We can the derive the M step for the EM algorithm:

Proposition 3. *The following updates to the values of the parameters maximize $Q(\mathcal{X})$ subject to the normalization constraint $\sum \alpha = 1$:*

$$\widehat{\alpha_q} = \sum_i \frac{\widehat{\tau_{i,q}}}{n} \qquad \text{(wM-}\alpha\text{)}$$

$$\widehat{\pi_{q,l}} = \operatorname*{argmax}_\pi \sum_{i<j} \widehat{\theta_{iqjl}} \log \mathcal{L}(\pi)(X_{i,j}) \qquad \text{(wM-}\beta\text{)}$$

Proof. The proof is done by seeing that each of the pairs of sums above can be exchanged and that each of the returned sums depend only on one of the α_q or $\pi_{q,l}$. Maximizing Q then amounts to maximizing each of the individual sums, thus giving the result presented above. □

Again, we cannot go further in the computation for the updated value of π without defining $\mathcal{L}$ with a mathematical expression, which is why we suggest that the following proposition gives a good starting point for the models.

Proposition 4. *If $(\mathcal{L}(\pi))_\pi$ is an exponential family such that*

$$\mathcal{L}(\pi)(x) = h(x) \exp\left(\eta(\pi)T(x) - A(\eta(\pi))\right),$$

where A is the normalization function in the true parameter $\eta(\pi)$, h is a normalization function in x and $T(x)$ is the actual data transformation, then, necessarily, the maximization condition (when deriving) leads to

$$A'(\eta(\pi_{q,l})) = \frac{\sum_{i,j} \widehat{\theta_{iqjl}} T(X_{i,j})}{\sum_{i,j} \widehat{\theta_{iqjl}}}. \qquad (\pi\text{-exp-update})$$

This shape of the update state can more easily be solved when $A' \circ \eta$ is a bijection.

The exponential family of probability distributions functions chosen as examples for $\mathcal{L}$ might seem arbitrary, but it actually encompasses everything we need to prove the result for Bernoulli[7], Poisson[8], Exponential[9] and Gaussian[10] distributions, showing *en passant* the coherence of our results and model with ERMG.

Proof. To prove statement (π-exp-update), start from the definition of Q in our setting (and in particular only the terms of Q depending on $\pi_{q,l}$ as in Proposition 3)

$$\begin{aligned}Q_{q,l}(\eta) &= \sum_{i,j} \theta_{iqjl} \log \mathcal{L}(\eta)\left(X_{i,j}\right) \\ &= \sum_{i,j} \theta_{iqjl} \left(\log h\left(X_{i,j}\right) + \eta T\left(X_{i,j}\right) - A(\eta)\right) \\ &= \underbrace{\sum_{i,j} \theta_{iqjl} \log h\left(X_{i,j}\right)}_{\text{constant in } \eta} \\ &\quad + \eta \sum_{i,j} \theta_{iqjl} T\left(X_{i,j}\right) - WA(\eta)\end{aligned}$$

[7] Take $\eta(\pi) = \log \frac{\pi}{1-\pi}$ and $T = \mathrm{id}$.
[8] Take $\eta(\lambda) = \log \lambda$, $h(x) = \frac{1}{x!}$ and $T = \mathrm{id}$.
[9] Take $\eta(\lambda) = -\lambda$, h, $T = \mathrm{id}$
[10] Given variance σ, take $\eta(\mu) = \frac{\mu}{\sigma^2}$, $h(x)$ the pdf for centered normal distributions and $T = \mathrm{id}$.

where $W = \sum \theta_{iqjl}$. Differentiating with respect to η gives

$$\frac{\mathrm{d}}{\mathrm{d}\eta} Q_{q,l}(\eta) = \sum_{i,j} \theta_{iqjl} T\left(X_{i,j}\right) - W A'(\eta),$$

which, given the extremum condition on the vanishing of the derivative rewrites as

$$A'(\eta) \sum_{i,j} \theta_{iqjl} = \sum_{i,j} \theta_{iqjl} T\left(X_{i,j}\right).$$

□

2 IMPLEMENTATION AND EVALUATION OF WEIGHTED CLUSTERING MODELS

2.1 General Implementation

We implemented [2]'s method as well as our own method in Python 3.13.19, using the libraries NumPy 2.2.6, NetworkX 3.5, SciPy 1.16.3, and pandas 2.2.3. The code is available on GitHub: GitHub.

For the real-data experiments, we used the dataset from [7]. The free version of this dataset consists of approximately 48,000 large cities across the world, including their names, countries, longitudes, and latitudes. From this dataset, we built distance graphs using the Haversine formula. The distances were then normalized and inverted to obtain a weighted graph with weights between 0 and 1, where higher weights correspond to cities that are geographically closer.

Using all 48,000 data points would yield a distance graph with over 2 billion edges. This is impractical and would require excessive RAM. Since our independence approximation for the Lyapunov Central Limit Theorem 1 is empirically verified for $n \geq 500$ vertices, we considered a subset of 1,000 cities chosen randomly from the dataset.

After examining the weight distribution of the distance graph, we selected an exponential law for our wERMG model. More precisely, we defined:

$$\mathcal{L}(\pi)(x) = \frac{1}{\pi} e^{-\frac{x}{\pi}}$$

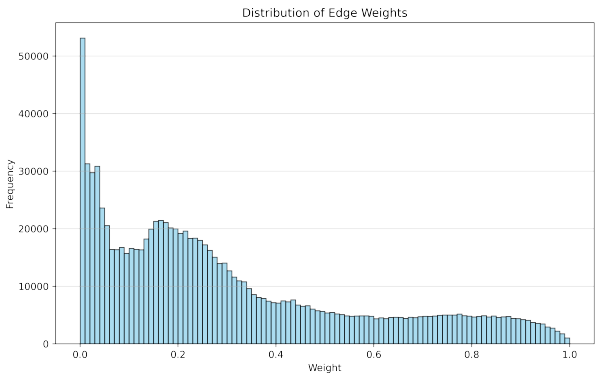


Figure 4: Weight distribution of the distance graph.

Therefore, we used this distribution as the primary distribution for all experimental results.

The wERMG EM algorithm was implemented specifically for the aforementioned exponential distribution. This choice allowed us to significantly simplify the general formulation, leading to a more optimized version of the algorithm suited for the large graphs processed.

Hence, all experiments and examples below use the exponential distribution for wERMG.

2.2 Results

2.2.1 Model comparison. We plot here an ERMG graph as well as a wERMG graph using the same parameters α and π.

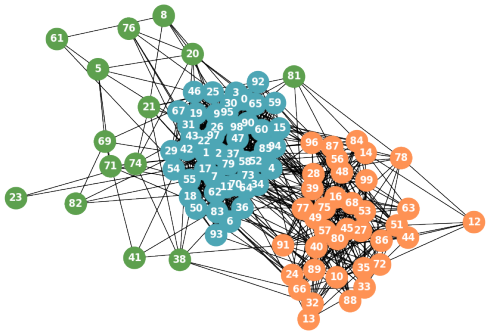


Figure 5: Synthetic ERMG graph with fixed α and π

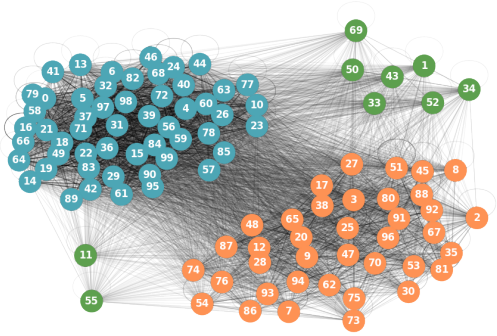


Figure 6: Synthetic wERMG graph with fixed α and π

2.2.2 Coherence of Lyapunov's central limit theorem.

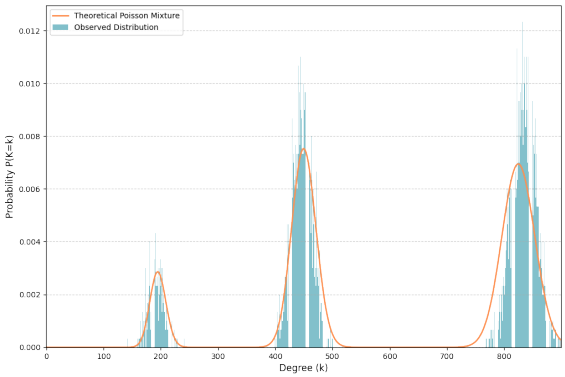


Figure 7: Degree distribution for a synthetic ERMG graph along with its Poisson approximation

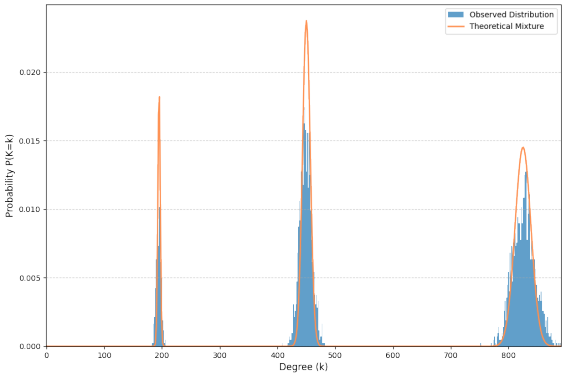


Figure 8: Degree distribution for a synthetic wERMG graph along with its normal approximation

2.2.3 *EM algorithm.*

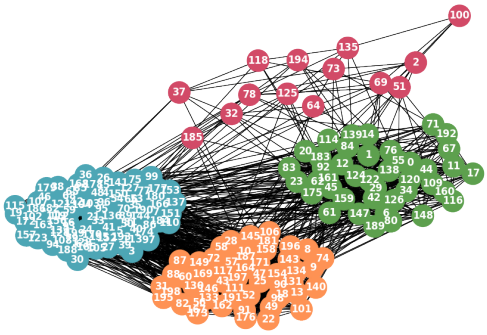


Figure 9: True ERMG graph used for testing the EM algorithm.

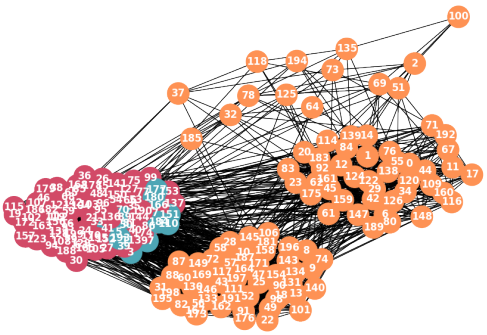


Figure 10: Clustered ERMG graph using the EM algorithm.

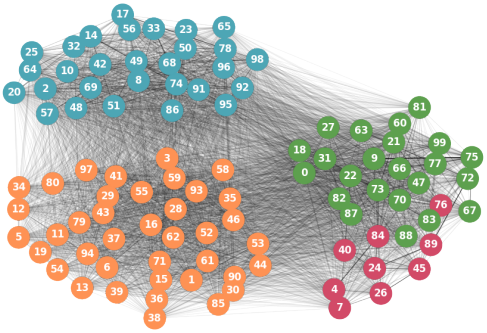


Figure 11: True wERMG graph used for testing the EM algorithm.

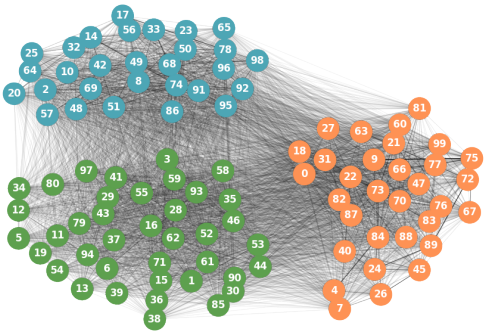


Figure 12: Clustered wERMG graph using the EM algorithm.

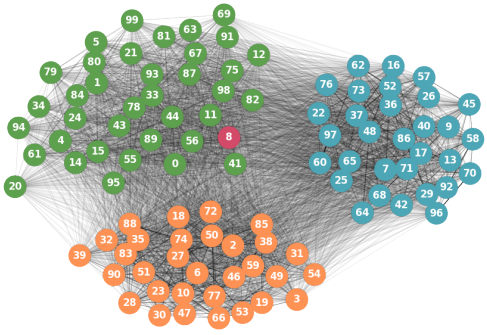


Figure 13: Reconstructed wERMG graph based on the hidden parameters estimation of the EM algorithm.

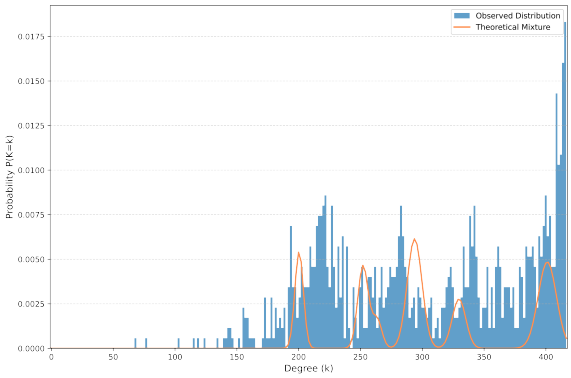


Figure 16: Degree distribution of the world cities distance graph along with its normal estimation based on the associated wERMG model obtained with the EM algorithm.

2.2.4 *Experiment on world cities.*

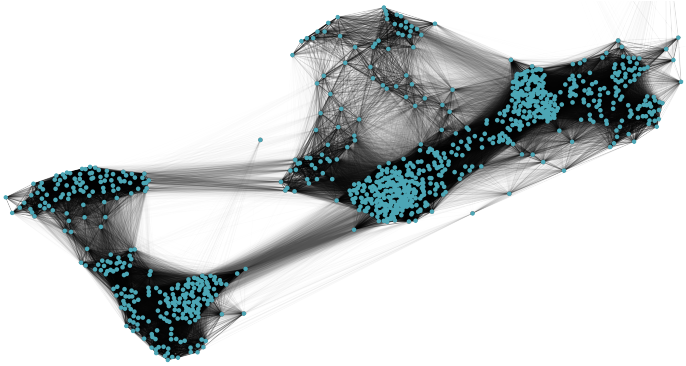


Figure 14: World cities distance graph with 1000 cities.

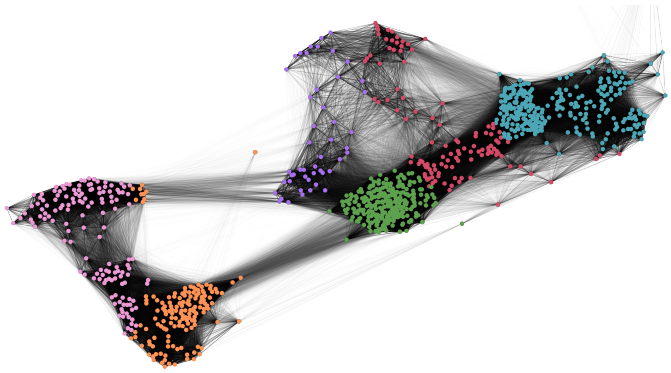


Figure 15: World cities distance graph clustered using the EM wERMG algorithm (Q = 6).

REFERENCES

[1] Alfredo Cuzzocrea, Edoardo Fadda, and Alessandro Baldo. "Lyapunov Central Limit Theorem: Theoretical Properties and Applications in Big-Data-Populated Smart City Settings". In: *Proceedings of the 2021 5th International Conference on Cloud and Big Data Computing*. ICCBDC '21. Liverpool, United Kingdom: Association for Computing Machinery, 2021, pp. 34–38. ISBN: 9781450390408. DOI: 10.1145/3481646.3481652. URL: https://doi.org/10.1145/3481646.3481652.

[2] Jean-Jacques Daudin, Franck Picard, and Stéphane Robin. *A mixture model for random graphs*. Research Report RR-5840. INRIA, 2006, p. 19. URL: https://inria.hal.science/inria-00070186.

[3] {Michael Charles} Davis et al. "Generating Realistic Labelled, Weighted Random Graphs". English. In: *Algorithms* 8.4 (Dec. 2015), pp. 1143–1174. ISSN: 1999-4893. DOI: 10.3390/a8041143.

[4] Israel Dejene Gebru et al. "EM Algorithms for Weighted-Data Clustering with Application to Audio-Visual Scene Analysis". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.12 (Dec. 2016), pp. 2402–2415. ISSN: 2160-9292. DOI: 10.1109/tpami.2016.2522425. URL: http://dx.doi.org/10.1109/TPAMI.2016.2522425.

[5] Pavel N. Krivitsky. "Exponential-family random graph models for valued networks." In: *Electronic journal of statistics* 6 (2011), pp. 1100–1128. URL: https://api.semanticscholar.org/CorpusID:4360023.

[6] Satu Elisa Schaeffer. "Graph clustering". In: *Computer Science Review* 1.1 (2007), pp. 27–64. ISSN: 1574-0137. DOI: https://doi.org/10.1016/j.cosrev.2007.05.001. URL: https://www.sciencedirect.com/science/article/pii/S1574013707000020.

[7] SimpleMaps. *World Cities Database*. Accessed: 2025-11-29. 2025. URL: https://simplemaps.com/data/world-cities.