UNIVERSITY OF SOUTHAMPTON

A Morse-theoretical clustering algorithm for annotated networks and spectral bounds for fuzzy clustering

by

Fabio Strazzeri

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the
Faculty of Social Sciences
School of Mathematical Sciences

November 2018

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF SOCIAL SCIENCES
SCHOOL OF MATHEMATICAL SCIENCES

Doctor of Philosophy

by Fabio Strazzeri

Given a set of objects X a clustering algorithm is a formal procedure that groups together objects which are similar and separates the ones which are not, thus mimicking the human ability to categorise and group together objects. Clustering algorithms have been growing for decades and clustering has become a standard data analytic technique for many fields. Standard clustering methods however fail to integrate object metadata, often readily available to the user, in the analysis.

We present in this thesis a novel clustering algorithm, called `Morse`, which integrates metadata information and Morse theory, a well-known topological theory, to reveal the "basins of attraction" induced by the metadata. The algorithm is described in its general form together with a study of its performance on the LFR benchmark model. We tested `Morse` in a real-world scenario and showed it helped to identify phenotypes of asthma based on blood gene expression profiles. We also looked at `Morse` in the axiomatic setting proposed by Kleinberg and introduce a novel axiom, Monotonic Consistency, that avoids the widely-reported problematic behaviour of Kleinberg's Consistency, and a possibility result for Monotonic Consistency given again by `Morse`. Furthermore, we extended Kleinberg's axiomatic setting to graph clustering and proved an impossibility result for Consistency, and a possibility result for Monotonic Consistency given again by `Morse`.

Lastly, we explored how a general clustering algorithm affects the structure of a graph using a graph spectral distance. In this direction, we proved two different bounds for such distance with respect a graph and its quotient graph induced by a hard partition, and generalised these results to fuzzy partitions.

Contents

Academic Thesis: Declaration of Authorship

I, Fabio Strazzeri, declare that this thesis and the work presented in it are my own and has been generated by me as the result of my own original research.

A Morse-theoretical clustering algorithm for annotated networks and spectral bounds for fuzzy clustering.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published works of others, this has always been clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself or jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

Acknowledgements

First I would like to thanks my family, Vittorio, Anna and Alice, for always been there for me.

My supervisors, Ruben, Paul, Ben and Rob, helped me a lot during these three years in Southampton, providing useful suggestions, constructive criticism and support along the way. I want to thank especially Ruben for the amount of effort he put into teaching me how to present in talks and reports my work, for the faith he had in my ability as a researcher and in my ideas. I also thank Jim Schofield, who played an essential role in the development of my thesis, collaborating with me as a friend and a colleague. It has been a pleasure working with all of them.

These three years have been an important part of my life. It would not have been possible to go through the difficulties and anxieties of a PhD life without all the friends that were with me. I'd like to thank all the colleagues in the Maths department for the fun we had together, board game nights, beer festivals (thanks Mike and Joe), the breakfast club (thanks Charles) and tournaments of Perudo. Thanks Robin, Mike, Dave, Marta, Ariana, Kiko, Ilaria, Donato, Martina P, Martina T, Fede, Liana, Yafet, Joe, Larry, Emma, Xin, Abi, James, Conrad, Tom, Lorna, Charles, Ingrid, Motiejus, Ana, Hector, Matt, Simon, Vlad, Hollis, Megan, Naomi, Holly, Simos, Ashley.

Thanks to Robin, Kiko and Mike for the mathematical discussions about Morse Theory, algorithms and a lot of algebra. It has been always interesting to talk and to discuss maths with you all. Again thanks to Robin, you are an exceptional friend, which I have been very lucky to meet.

My life has not been a sedentary one for years and there are friends that have been distant geographically but never really absent in my life. Thanks to Giovanni and Francesco to convince me that I could actually do a PhD, and to Uguale, Floris, Sara, Angelo and Bene for our reunions as to Antonella for discussing the numerous upsides (and downsides) of life after a Master. Thanks to Elisa for your support, presence and understanding. For always being available for excursions and funny evenings when I went back home, thank you Andrea L, Michele, Andrea C, Nurco, Daniele, Mauro e Stefano.

These three years have been exciting, despite the difficulties. The support and friendliness showed by so many people was of essential help in the completion of my PhD. I am so glad that I could meet so many kind people and help them as they were helping me when life seemed hard.

Chapter 1

Introduction and Literature Review

Given a set of objects X and a pairwise similarity function, that is, a real-valued function $f : X \times X \to \mathbb{R}$, a *clustering algorithm* is a formal procedure that groups together objects which are similar and separates the ones which are not, thus mimicking the human ability to categorise and group together objects [JD88]. In cluster analysis a group of objects is split up into a number of more or less homogeneous subgroups on the basis of an often subjectively chosen measure of similarity, i.e. chosen subjectively based on its ability to create *interesting* clusters, such that the similarity between objects within a subgroup is larger than the similarity between objects belonging to different subgroups [BJ81]. Methods and approaches to clustering algorithms have been growing for decades [JD88, JMF99, AR13], with clustering becoming a standard data analytic technique [Jai10] for many fields, such as machine learning, artificial intelligence, pattern recognition, textual document collection, image segmentation, genetics, biology, psychiatry or geography [Har75]. Such applications can be the study of psychological traits, as in [Cat43] where clustering techniques were essential to solve the inconsistency of results of factor analyses of personality, or genome-wide expression analysis, as in [ESBB98], where genes of known similar function are grouped together efficiently, aiding to understand the status of cellular processes and to infer functions of many genes for which information is not available.

Clustering is a subjective process so an absolute judgement regarding the quality or the efficiency of all clustering techniques is precluded [JMF99], in addition different fields may require different techniques or make different assumptions on the "interesting" clusters, the similarity function and so on. In general, cluster analysis needs a series of trials and repetitions, and to guide the user choice for the appropriate algorithm is still a problem requiring effort, as there are no universal and effective criteria [XW05]. In this direction, different indices of goodness of a clustering solution have been proposed, [Des13], but as well studies of clustering algorithm and their properties, also called in the literature axioms, have been carried out, see [Kle03, ZB12, YX14].

Although simple and useful, standard clustering methods fail to incorporate object metadata, [NC16], that is, external quantitative or qualitative information on the objects. A metadata, also called annotation, can represent physical location, age, ethnicity but as well a measure of overall importance. In community detection there have been different attempts to include this information, manly using them as an inference of the underlying communities to enhance some clustering methods as in[NC16]. However, in [PLC17] it has been proven that using metadata in such way does not always improve the clustering result but in addition it could bring "severe theoretical and practical problems" [PLC17]. In the study of networks phenomena, such as synchronising, cascading or spreading, it is well-known that only few nodes are influential [LCR+16] and different measures of such influence have been proposed: examples are betweenness, closeness or eigenvalue centrality but as well ranking functions such as PageRank.

We present here a novel clustering algorithm, called `Morse`, that incorporates metadata in the analysis, not as in community detection techniques, but instead as a measure of importance of the objects. With this approach, we do not aim to group together objects with similar metadata, as in [NC16], we instead extract the *basin of attraction* of those few important nodes in the graph, see Figure 1.2, and so we aim to group a node with its respective "attractor". `Morse` integrates the topological theory *Morse Theory* on a graph representation of the clustering problem, that is, constructing a *graph* $G = (V, E)$ where the node-set V (vertices) are the objects in X and the link-set E (edges) encodes the similarity function $f : X \times X \to \mathbb{R}$. This theory has been applied successfully both to continuous spaces [Mil63] and discrete ones [For98], and as a graph can be regarded as a discrete space we were able to develop the algorithm `Morse` using (discrete) Morse theory.

We will now in Section 1.1 review some clustering algorithms showing the different techniques employed; for a more exhaustive work we refer to [XW05, JD88, Har75], and how metadata have been employed in clustering algorithms in Section 1.2. There follows an overview of our algorithm `Morse` in Section 1.3 together with the different analysis and applications we pursued to study it.

1.1 Clustering algorithms

Different criteria and focus can lead to different taxonomy of clustering algorithms, as shown in Figure 1.1 where each leaf of the tree corresponds to a different family of algorithms.

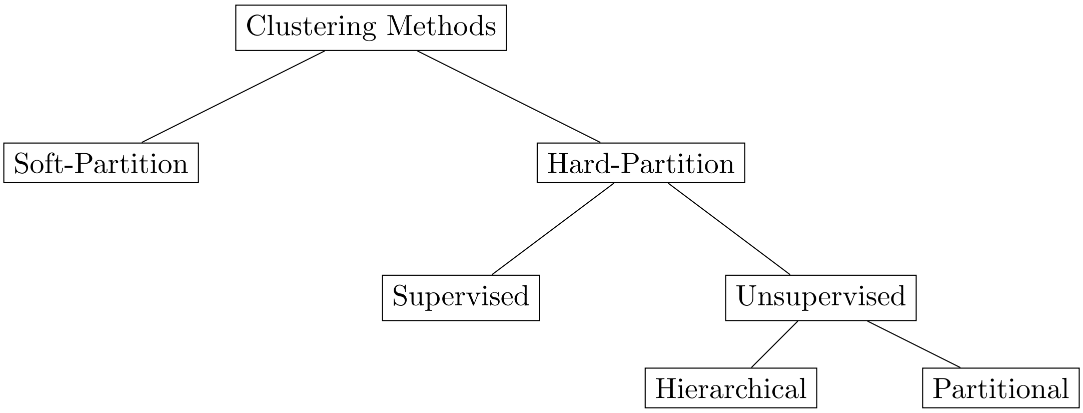


FIGURE 1.1: Tree of algorithm types, see [JD88, p. 58]

We will summarise now these different families and some of the most clustering methods.

Soft/Hard Partition: Given a set X, a hard-partition algorithm will return a partition $\mathcal{P}$ (that is, a collection of subsets of X) where each object belongs to only one cluster, that is, the clusters are pairwise disjoint. A soft-partition algorithm will return instead clusters which are not disjoint and for each pair of object and cluster a probability of belongingness is given. In the literature, some authors refer to soft-partitioning algorithms as *fuzzy* clustering algorithms.

Supervised and Unsupervised: If an algorithm takes as input only the dissimilarity measure between objects then it is called *unsupervised.* Otherwise, if any type of external information is used to set in advance, guide, or correct the algorithm then it is called *supervised.* Such information can be for example a pre-existent set of categories to which assign each object, or instead a precedent study on which we *train* the algorithm.

Hierarchical and Partitional: As result of a *partitional algorithm* we obtain a disjoint division of the set X. A *hierarchical* clustering algorithm instead returns a nested sequence of hard-partitions of X, that is, at each step of the sequence two or more cluster are always merged together.

Hierarchical algorithms construct a hierarchical structure of X according to the dissimilarity measure and it is left to the user to decide when to stop the merging process. Hierarchical algorithms that proceed from the singleton partition (where each object forms a cluster) to the trivial one (where each object is in the same cluster) are called *agglomerative.* There are others which operate in opposite direction splitting instead of merging clusters and are called *divisive*. These algorithms are often computationally expensive, as a set of m objects can be split in two subsets in $2^{m-1} - 1$ different ways, so they are not commonly used in practice; for a further discussion on them we refer to [ELLS01].

The steps of an agglomerative hierarchical algorithms are the following:

1. Start with the singleton partition and the similarity measure between objects;
2. Search for the two clusters which are more similar and merge them;
3. Update the dissimilarity measure between the new formed clusters;
4. Repeat steps 2-3 till all objects are in the same cluster.

It is clear that there are as many algorithms as ways of updating dissimilarity measures between clusters. For the *Hierarchical Single Linkage* method the dissimilarity between two clusters is determined by the less dissimilar objects in different clusters, whilst on the contrary in the *Hierarchical Complete Linkage* method it is determined by the most dissimilar ones. As shown in [LW67], it is possible to generalise these two algorithms and derive more elaborate definitions of dissimilarity between clusters, such as in *Median Linkage*,*Centroid Linkage* or *Ward's method*. Note that these are called *geometric* methods as they consider geometric centres of clusters for dissimilarity update, whilst complete and single linkage consider all pairs of points in two clusters and are so called *graph* methods. Often, hierarchical algorithms lack robustness, they are computationally expensive (complexity at least $O(n^2)$ with n number of objects) and are sensitive to noise and outliers [XW05]. In addition, after a merging is done it is irreversible, so there is no possibility for corrections and they tend to form spherical shaped clusters. Recently, different algorithms have been presented to deal with such problems and to tackle large-scale data, such as BIRCH [ZRL96], CURE [GRS98] and ROCK [GRS99]. BIRCH is able to deal with large dataset, with computational complexity of $O(n)$, and it has shown to be robust to outliers. This is achieved through a *clustering feature tree* which stores the summaries of X with respect each cluster and so captures clustering information reducing the required storage. On the other hand, CURE, developed by Guha, Rastogi and Shim, has the ability to recognise more sophisticated cluster shapes and avoid the chaining effect of which some Linkage algorithms suffer of. The crucial property of CURE is the representation of the clusters using well-scattered points, which can be even more pushed towards the cluster centroid, using a user-defined parameter, reducing the effect of outliers. The agglomerative hierarchical algorithm ROCK instead has been developed to deal with objects that possess qualitative or categorical characteristics. Both ROCK and CURE use random sampling and partition to reduce the computational complexity.

In partitional clustering objects are divided in k clusters, where k is often a parameter fixed a priori, and in principle the clustering solution given is the optimal partition based on certain criteria. However, finding this solution is usually infeasible in practice due to expensive computation, so different techniques to overcome such difficulty have been proposed over the years. The sum of the squared-error function is one of the criterion most widely used. We assume that the objects are a set of vectors in a real vector-space $\mathbb{R}^d$ and we want to divide them in a partition Γ with k clusters. The *squared error criterion* is then defined as

$$f(\Gamma, X) = \sum_{C_i \in \Gamma} \sum_{x_j \in C_i} ||x_j - \mu_i||,$$

where μ_i is the *centroid* of the cluster C_i, that is, the 'average' of the objects in C_i. The best-known algorithm of this class is the k*-means* algorithm, which steps follows.

1. Randomly assign each object to one of k clusters;
2. Compute the centroid μ_j of each cluster C_j;
3. Assign each object x_i to the nearest cluster C_j, that is, such that its centroid μ_j is the closest to x_i among all centroids;
4. Repeat step 2-3 until there is no change for each cluster.

Note that there exists other choices of initial partition, proposed by Forgy [For65], Kaufman [KR09] and MacQueen [Mac67].

The k-means method is very simple and works in a wide range of practical problems, in particular it performs very well with clusters that are compact (small distances between points) and hyperspherical (shaped as a high dimensional sphere in $\mathbb{R}^d$). There are, however, several drawbacks of this algorithm which are well-studied and many variants have been proposed to overcome them. We will now illustrate some of them. There is no universal method to identify the initial partition or the number of clusters and the convergence of the algorithm can vary a lot with different initial points. A strategy often used is to run the algorithm several times with different initial partitions as proposed in [BF98] or instead to use a global k-means algorithm where the number of clusters varies from 1 to k as in [LVV02]. A technique to deal with the estimation of k has been proposed in [BH65] with the algorithm ISODATA. In this case, the number of clusters changes by merging and splitting clusters according to some predefined thresholds, shifting the problem of identifying the number of cluster on the choice of the threshold parameters. The splitting procedure of ISODATA also deals with the obstacle of outliers, which can affect sensibly k-means, again for its tendency of creating hyperspherical clusters.

Using a probabilistic point of view we can assume that the objects are generated via a probabilistic distribution and so objects in different clusters should be generated by different probability distributions. With such assumptions, we can formulate algorithms of the *Mixture Densities-based* family. We assume that the prior probability $p(C_i)$ for C_i, with $i = 1, \dots, k$, is known, or estimated by the user, and as well the conditional probability density $p(X|C_i, \theta_i)$, with θ_i unknown. Using as weights the probability density functions $\{P(C_i)\}_i^k$, with $\sum_i P(C_i) = 1$, we can construct the *mixture probability density function* for the entire set X as

$$p(X|\Theta) = \sum_{i=1}^{k} p(C_i)p(X|\theta_i).$$

If $\Theta = \{\theta_1, \dots, \theta_k\}$ is determined, the degree of belongingness for an object x_j to a cluster C_i can be calculated using the posterior probability and the Bayes's theorem. To find then Θ we can use *Maximisation-Likelihood* estimation [DHS12] which consists of choosing the best estimate as the one that maximise the probability of generating all observations given by

$$p(X|\Theta) = \prod_{j=1}^{n} p(x_j|\Theta),$$

where we assume that each single observation x_j is independent and identically distributed with respect the probability distribution p. If we try to differentiate and find the maximum of such function we obtain a complex set of implicit equation with no easy solution, so instead we perform an iterative process using a well-known algorithm, the *Expectation-Maximisation* algorithm [MK07]. We consider each object x_j as divided in two part, the observable one x_j^g, which include all the data given by the user, and the missing x_j^m which instead will represent the belongingness to a cluster, namely a binary vector with entry $x_{i,j}^m = 1$ if $x_j \in C_i$ and 0 otherwise. The set X^g will be then the observed object set X while X^m will be the clustering solution. The log-likelihood is then defined as

$$l(\Theta) = \sum_{i=1}^{k} \sum_{j=1}^{n} x_{i,j}^m \log p(C_i)p(x_j^g|\theta_i).$$

The algorithm, after choosing a convergence criterion, will then proceed as follows.

1. Choose a starting Θ_0 and set $i = 0$
2. Compute the expectation of the log-likelihood

$$q(\Theta, \Theta_i) = E[\log p(X^g, X^m|\Theta_i)|X^g, \Theta_i]$$

3. Find Θ_{i+1} that maximise $q(\Theta, \Theta_i)$ and increase i
4. Repeat step 2-3 till the convergence criterion is satisfied.

There are few drawbacks of using this algorithm, such as sensitivity to starting Θ_0, possible convergence to local but not global optimum and slow convergence ratio [MK07].

Another possible approach to cluster analysis uses the graph-representation of a set of objects X. A graph G is a pair formed by a node-set V (vertices), a link-set $E \subset V \times V$ (edges). The similarities between objects can be incorporated through a *weight function* w, which is a real positive function on E. A graph-representation of the set of objects X is then a graph G such that $V \equiv X$ and the weight of a link e between two nodes measures how similar or not they are. If we consider a threshold δ, we can construct the graph G_δ as the graph obtained removing all links with higher value than δ. With this technique, Single Linkage hierarchical clustering algorithm is equivalent to finding the maximally connected subgraphs with different values of δ and on the other hand Complete Linkage HC to finding the maximally complete subgraphs.

Another type of techniques relies on the *Laplacian* operator of a graph. This operator is a linear operator L on the real vector space generated by the nodes of G. Let n be the number of nodes in G and $V = \{v_1, \ldots, v_n\}$, then any vector $u \in \mathbb{R}^n$ belongs to such vector space but as well determines a real-valued function $f_u : V \to \mathbb{R}$ such that $f_u(v_i) = u_i$ for $i = 1, \ldots, n$. Assuming a weight function w is given on G, the Laplacian will map a function f_u to another $g = L(f_u)$ as follows

$$g(v_i) = \sum_{j \neq i} w(i,j)[f_u(v_i) - f_u(v_j)].$$

A well-known algorithm that makes use of such operator is the *spectral clustering algorithm* [Mac67]. The Laplacian is a semi-definite positive operator [Mer94] so its eigenvalues and eigenvectors are real. Given a number k, we can select the smallest k eigenvectors of L and embed each object x_i in $\mathbb{R}^k$ using the i^{th} entry of those eigenvectors. Equivalently, we are considering k eigenfunctions of the Laplacian operator, say $f_1, \ldots, f_k$, and we assign coordinates in $\mathbb{R}^k$ to a object x_i equal to $(f_1(v_i), \ldots, f_n(v_i))$, with v_i the node in G representing x_i. The points $x_1, \ldots, x_n$ will form in $\mathbb{R}^k$ a *spatial network*, that is, a graph with nodes embedded in a metric space. Using this embedding, we can search the partition $\mathcal{P}$ with k clusters using the k-means algorithm.

Another approach consists of using both a probabilistic approach and a graph-theoretic one, as in CLICK [SS00]. Consider the binary relationship $\sim$ on X such that $x \sim y$ if and only if x, y are in the same cluster, then we define the weight of a link $e = (x_i, x_j)$ as

$$w'(x_i, x_j) = \log \frac{Prob(x_i \sim x_j | w(x_i, x_j))}{Prob(x_i \not\sim x_j | w(x_i, x_j))}.$$

CLICK also assumes that the weight function follows a Gaussian distributions with different means and variances, respectively for intra and inter clusters links. The distribution parameters can be estimated from prior knowledge or using parameter estimation methods. CLICK then recursively checks the graph and compute the clusters as subgraphs that satisfy some criterion function, singleton clusters are removed for further computation. At last, using these clusters both singleton cluster adoption and cluster merging are performed to obtain the final clustering solution.

It is worth noticing that clustering algorithms have been developed in a wealth of different scenarios, which shows the importance of clustering as technique, but on the other hand this can also cause confusion, due to different assumptions and goals. In fact, clustering algorithms developed for some specialised fields can be biased by some assumptions, as on the cluster structure or the similarity function. For example, the k-means algorithm explained before is based on the assumption that a cluster has hyper spherical structure, which may not be true in all clustering problems. A similar argument holds for the Expectation-Maximisation algorithm in which data has to fit a model set in advance. These biases naturally affect the performance of an algorithm when such assumption are not satisfied.

1.2 Clustering on annotated networks

Although simple and useful, standard clustering methods, such as the ones presented, fail to incorporate node annotation and a set X is considered usually in a graph-representation as an unadorned set of nodes and links [NC16]. Node annotation, sometimes called in the literature metadata, is often readily available and application dependent but ignored and can be crucial to provide insights and more useful clustering solutions when used as an additional input. This approach, in general, falls in the category of supervised algorithms, as the annotation given comes from an external source and it can describe properties of the nodes, such as age, ethnicity or physical location. In community detection, it is a standard practice to treat some discrete values observed on the objects as annotations representing with some uncertainty the underlying ground-truth communities [NC16]. In this type of approach, a failure to find communities well-correlated with the metadata is often seen as an algorithm failure, while it could instead come from metadata irrelevance to the structure of the graph or from differences between the aspects captured by the network and the metadata [PLC17]. With the same spirit of Wolpert in [Wol96] and [WM97], Peel, Larremore and Clauset proved a *no-free lunch theorem* [PLC17, Theorem 3] for community detection algorithms, showing that for any method f the average accuracy over all possible community detection problem is a constant independent of f. This means that, given two community detection methods f_1, f_2, if f_1 outperforms on a set of problems f_2 then there exists another set of problems in which f_2 outperforms f_1. It is impossible to get overall better performance without some cost. So, in a natural trade-off, either an algorithm performs very well on a specialised set of tasks and poorly in the rest, or it will perform fairly well but on a wide variety of tasks. In conclusion, treating metadata as ground truth does not give us better algorithms a priori. In addition, doing so we are also simultaneously testing both the metadata's relevance and the algorithm's performance with no ability to differentiate between the two. This approach then not only does not ensure better solutions but also "*induces severe theoretical and practical problems*" [PLC17].

A different way of incorporate annotations in graph cluster analysis has been proposed with *weighted clustering algorithms* [ABDBL12]. In this work, a set X is considered embedded in an euclidean space with distance d and two objects $x \neq y$ can be *duplicates*, that is, coexist in the same position, so $d(x, y) = 0$. Following the work in [Wri73], each object is considered with a mass and when duplicates are present we consider them as one object with mass equal to the sum of their masses. The effect of duplicates is then studied with the assumption that the mass is encoded by a *weight function* w *over* X, $w : X \to \mathbb{R}^+$. In practice, this assumption results in a deformation of the link weight function due to the different mass of its nodes. Different properties are then formulated to help the user to distinguish and select the appropriate algorithm for the analysis. Based on these properties, we have clustering algorithms that are *weight robust*, always ignore node weights, others that are instead always *responsive to weights*, but also some that can be both [ABDBL12]. The authors point out as well that algorithms known to perform well in practice tend to be more responsive to weights (such as k-means) while single linkage, which often performs poorly in practice, [AH81], is weight robust.

Annotations can come also from the graph structure, without any external source or assumption on presence of duplicates and node mass. In the understanding of graph structures, different measures of node importance have been proposed over the last decades. Those can be degree (weighted or not), closeness, betweenness or eigenvalue centrality [LFH10], but as well ranking functions as PageRank [PBMW99] or GeneRank [MBHG05]. It is widely accepted that several graph phenomena as cascading, spreading or synchronising are not equally affected by each node, but instead only by few influential one and to study and identify such nodes is of importance both theoretically and practically [KGH$^+$10, LZYZ11, LCR$^+$16]. One could then consider these measures as annotations of the nodes instead of an external metadata and so unravel not only influential nodes but as well their community, or basin of attraction. In our work, we developed and studied an algorithm, `Morse`, which is able to achieve such a goal when the annotations are unsupervised, that is, come from a graph analysis, or supervised, such as external metadata.

1.3 `Morse` clustering

Our algorithm is deeply rooted in a topological theory, *Discrete Morse Theory*, which is capable of inference on topological changes and properties of a discrete space, such as a graph or a mesh of a continuous space, and has been used widely in different fields: in [RWS11] for grey-scale digital images representation where a discrete Morse function is constructed to characterise the topological changes of the image and simplify its representation; in [CCL03] for molecular shape analysis where a discretisation of the Connolly's function, that controls the docking (process by which two or several molecules form a complex), is used to decompose a surface into regions of homogeneous flow and provide relations between local quantities and global ones; in [ACR+02] it has been used for path planning of robots between two points, again using discrete Morse theory to decompose a space in several cells, so that simple control strategies to achieve tasks, such as coverage, are feasible within each cell. More applications of such theory can be found in astronomy [Sou11], text mining [WDM12], study of dissipation elements [GBG+14], surface reconstruction [SKK91] and so on.

We present here (Chapter 3) an algorithm that integrates discrete Morse theory with clustering on annotated graphs. More precisely, given an annotated graph, `Morse` is able to extract a clustering solution where each cluster is given not just as a subset of objects but as a *directed rooted tree*. Such directed trees will form an annotation-ascending flow on the graph, that unravels a hierarchy of nodes in terms of their annotation and for each cluster returns its most influential node with its basin of attraction. In the same spirit of other applications of discrete Morse theory, our algorithm in fact decomposes the graph in regions which are scale-free with respect the annotation and tree-like with respect to the flow. This procedure instead of assuming that clusters represent similar annotation groups, such as in the community detection algorithms, assumes that a cluster is defined by one influential node, the peak of such hierarchy, and its area of influence.

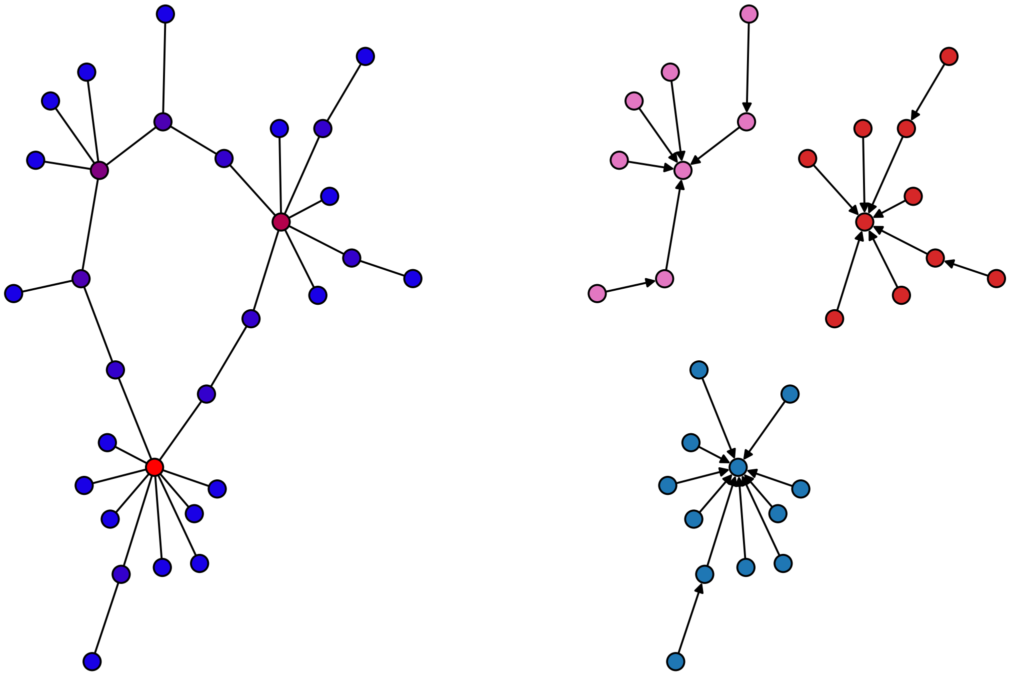


FIGURE 1.2: On the left we have a colouring based on the degree-annotation (blue for lower degree nodes, red for higher ones), while on the right we can see the `Morse` clustering based on such annotation together with a degree-ascending flow.

In general, clustering algorithms in community detection that incorporate metadata fall in the category of supervised algorithms, as the information is external; however in our algorithm this can be not the case. In fact, `Morse` is unsupervised when we incorporate as measure of importance an intrinsic node information, such as weighted degree, closeness centrality and so on, allowing it to be free from bias or from external editing. It is left to the user discretion how to use `Morse`, as a unsupervised or supervised algorithm. With both approaches, we are able to employ different (external or intrinsic) centrality measures to determine a clustering solution that decompose the graph with respect to them. In this way, we can unravel different basin of attraction depending on the definition of node influence, that is, node metadata. Clearly, also a mixture of intrinsic and external centrality measures can be used as well. The flexibility of `Morse` allows us to apply it to different scenarios, as it needs only a comparison between nodes (to determine the hierarchy structure) and a comparison between links (to determine the tree structure). In addition, `Morse` is a fast algorithm as it is local with computational complexity linear in terms of the number of links, see Figure 3.4. We present in Section 3.2 also a study we pursued to test `Morse`, where we compared its accuracy against different generative graph models, *LFR model* [LF09], *Barabasi scale-free model* [Bar09] and *Erdös Renyi model* [ER60].

After showing these different benchmarks we present in Chapter 4 an application on a Mapper-network [SMC91] of gene expression profiles of blood samples from a cohort of 498 asthmatics (comprising a range of severities from mild/moderate to severe, steroid-dependent asthma) and healthy control participants (this work has been submitted to Nature Communications). Using a mixture of intrinsic and external annotation, `Morse` enabled the identification of 9 molecular phenotypes characterised by differences in circulating immune cell populations, activity of glucocorticoid receptor signalling, and activation of Type-1 and 2 cytokine inflammatory pathways, thereby providing proof of concept for the application of `Morse` to stratify disease phenotypes according to their molecular pathology.

Developing `Morse`, we also examined it using an *axiomatic* approach for evaluation and assessment of clustering algorithms [Wri73, Kle03]. The process of developing new, faster and more adaptable algorithms has in fact been complemented by an interest in underlying principles and general desirable properties (sometimes called *axioms*) of clustering algorithms [FN71]. This axiomatic approach has been proposed in different flavours, such as on cost functions [PHB00] or on quality functions [BDA09]. A more recent interest in the axiomatic approach was sparked by Kleinberg's impossibility theorem [Kle03]. In the spirit of Arrow's impossibility theorem in social science, Kleinberg gives three natural properties then proves they cannot be simultaneously satisfied [Kle03, Theorem 2.1]. Given a clustering algorithm f and a distance d, Kleinberg's axioms are:

Scale-invariance: $f(\alpha d) = f(d)$ for $\alpha > 0$

Richness: for every partition $\mathcal{P}$ there exists a d such that $f(d) = \mathcal{P}$

Consistency: let $\mathcal{P}$ be a partition and d such that $f(d) = \mathcal{P}$. If we decrease, resp. increase, $d(i, j)$ with i, j in the same cluster, resp. different clusters, of $\mathcal{P}$, then f will return the same $\mathcal{P}$. Such deformation of d is called *$\mathcal{P}$-consistent transformation* of d.

Several authors have since criticised Kleinberg's approach, particularly the Consistency axiom [BDA09, ABDL10, CM13], and proposed alternative frameworks that circumvent the impossibility result [ZB12, CM13, CM10]. In all these cases, Kleinberg's impossibility is avoided by either restricting or extending the definition of clustering function, or shifting the axiomatic focus to clustering quality measures [BDA09, LM14, YX14], or cost functions [Kar99, PHB00]. We remained close to Kleinberg's original setting and directly address the problematic behaviour of the Consistency axiom instead replacing it by a weaker condition that we call *Monotonic Consistency*. Monotonic Consistency avoids the aforementioned problematic behaviour and, moreover, we show that an instance of `Morse` brings a possibility result: Monotonic Consistency, Scale Invariance and Richness are mutually compatible clustering axioms (Theorem 5.16). As far as we know, this is the only alternative in the literature to the Consistency axiom that is compatible with Richness and Scale Invariance without modifying the definition of clustering function. Generalising Kleinberg's axiomatic approach to graph clustering, in Section 5.4 we show that the impossibility result still holds, even when Richness is relaxed to Connected-Richness (partitions where every cluster is a connected subgraph) and a possibility result for Monotonic Consistency with the same instance of `Morse`. As the sparse case (G arbitrary) contains the complete case (G complete), Kleinberg's impossibility theorem [Kle03] is now a particular case of our graph clustering impossibility result (Theorem 5.21).

As `Morse` is a local algorithm it could be helpful also as a pre-processing step for a global algorithm, such as spectral clustering. Other authors have studied the effects of changes on structure on graphs, such as removing links, nodes or identifying them, and related them in terms of the Laplacian spectrum [HJ13a]. In particular, in [GHL15] work has been done in the direction of measuring these effects, using the *Wasserstein metric* [Dob70], when the graph is undirected and unweighted, that is, all links have weight 1. Using this distance, one could measure how far two graphs are, but as well how much a structural change can affect a graph. We continue in this direction in Chapter 6 studying how a partition influences the structure of a graph. We used the approach of [HJ13b] to define the Laplacian of a quotient graph with respect a partition, in this way we could extend the work of [GHL15] for weighted graphs and to derive two bounds (Theorem 6.17) for the distance between such quotient graph and the original one. Using this work, we could measure how a partition preserves the graph structure and how much the quotient graph, used in most hierarchical algorithms, reflects the original graph structure. We went further and extended this work for fuzzy partitions for which the results in [HJ13b] and [GHL15] can not be applied directly. From our point of view, we provide with this work an important measure for clustering quality that would apply to any type of clustering and could inform about the structural change it induces. In the future, it could be applied in the development of a `Morse`-spectral method for combining spectral clustering on a `Morse` pre-clustered data.

In Chapter 7 we present some variants of `Morse` which are still in a development process. We will describe how they can be useful to investigate the robustness of `Morse` with respect perturbation and outliers or to study the graph structure induced by the annotation, for the construction of a soft-partition `Morse` algorithm or a hierarchical agglomerative one.

I acknowledge the use of the IRIDIS High Performance Computing Facility, and associated support services at the University of Southampton, in the completion of the analyses in Section 3.2 and Chapter 4.

Each result in the present thesis has been referenced to the original source. Whenever the reference is missing, the reader should consider this thesis to be the original source.

Chapter 2

Morse Theory

Topology is the study of spaces of properties that are invariant under continuous deformation such as bending, contracting or stretching, but not cutting or gluing them. An important tool to study the *topological invariants* of a space is *Morse Theory*, which has been developed for differentiable manifolds by Milnor [Mil58] and for discrete spaces by Forman [For98]. Discrete Morse theory has been successfully applied in networks (which are graphs and so discrete spaces) in biology [PTNKT11], image processing [RWS11] and shape analysis [BGSF08]. In this chapter, we will introduce both theories recalling some of the major results. In our algorithm, we will use the applicability of discrete Morse Theory to graphs, as they are discrete spaces, to describe a particular way of constructing a Morse function on a graph. Here, we will, however, illustrate both theories in their general setting, that is, on *manifolds* in the continuous case and *simplicial complexes* for the discrete one.

2.1 Differential Morse Theory

In *Differential Morse Theory* the objects studied are differentiable manifolds, that is, n-dimensional spaces which locally resemble Euclidean n-space [Car92]. On these space we can define a *Morse function* as a real-valued function f with some mild conditions on its critical points. Using such conditions, we are able to infer topological properties of the space, such as if it is connected or orientable. Furthermore, we are able to describe a flow that detects the unique minimal decreasing paths respect to f. Before introducing Morse theory for differentiable manifolds, we need to recall some notions of differential geometry. For a more comprehensive treatment of differential Morse theory, we refer the reader to [Mil58, Mat02].

A differentiable manifold M of dimension n is a Hausdorff space covered by open sets $\{U_i\}_i$ each of which is equipped with a coordinate system or chart $\phi_i : U_i \to \mathbb{R}^n$, [DFN12]. Given two such charts $\phi : U \to \mathbb{R}^n$ and $\psi : V \to \mathbb{R}^n$, the map $\phi \circ \psi^{-1}$ is $1:1$ and differentiable where it is defined. A function $f : M \to \mathbb{R}$ is called differentiable when $f \circ \phi^{-1}$ is differentiable (from an open subset of $\mathbb{R}^n$ to $\mathbb{R}$) for every chart ϕ. For such functions we can define their *gradient*.

Definition 2.1 ([DFN12])**.** Let M be a manifold of dimension n and $f : M \to \mathbb{R}$ a function. Let φ be a *coordinate system* at a point $p \in M$ in a neighbourhood $U \subset M$ of p, that is,

$$\varphi : U \to \mathbb{R}^n, q \mapsto \varphi(q) = (x_1(q), \dots, x_n(q)),$$

where each x_i is a differentiable function $x_i : U \to \mathbb{R}$ and define the i^{th} coordinate of q in $\mathbb{R}^n$. We define the *gradient of f in $q \in U$* as the vector:

$$\nabla_f(q) = \left(\left.\frac{\partial f \circ \varphi^{-1}}{\partial x_1}\right|_{\varphi(q)}, \dots, \left.\frac{\partial f \circ \varphi^{-1}}{\partial x_n}\right|_{\varphi(q)} \right)$$

The gradient vector is the extension to manifolds of the concept of first derivative. To categorise different types of critical point of a function, we recall the following extension of second derivative.

Definition 2.2 ([DFN12])**.** Let M^n be a manifold and $f : M \to \mathbb{R}$ a function. Given a point $p \in M$ and $\varphi : U \to \mathbb{R}^n$ a coordinate system on a neighbourhood U of p we define for any point $q \in U$ the *Hessian matrix of f* as the square matrix $H_f(q)$ such that

$$(H_f(q))_{i,j} = \left.\frac{\partial^2 f \circ \varphi^{-1}}{\partial x_i \partial x_j}\right|_{\varphi(q)}, \ 1 \leq i, j \leq n.$$

As anticipated in order to define a Morse function $f : M \to \mathbb{R}$ we need to add some constrains on its critical points, in particular they will be always *non-degenerate*.

Definition 2.3 ([DFN12])**.** Let M^n be a manifold and $f : M \to \mathbb{R}$ a function, a point $p \in M$ is *critical for f* if $\nabla_f(p)$ vanishes for every coordinate system on a neighbourhood of p. A critical point p is *degenerate* if $H_f(p)$ is singular. The *signature* of $H_f(p)$ is the pair (e_+, e_-) where e_+ (resp. e_-) is the number, counted with multiplicity, of positive (resp. negative) eigenvalues of $H_f(p)$.

Given ϕ, ψ, two distinct coordinate system in a neighbourhood of p, we know that the map $\phi \circ \psi^{-1}$ is a bijective and differentiable. If the gradient of f vanishes with respect to ϕ, then it vanishes also for ψ. That is, if $\nabla_f(p)$ vanishes for some coordinate system in a neighbourhood of p, then it vanishes for every coordinate system in a neighbourhood of p. More, the eigenvalues of $H_f(p)$ do not depend from the coordinate system used, and so neither its singularity, but $H_f(p)$ itself does.

The eigenvalues of the Hessian are real as $H_f(p)$ is always a symmetric matrix and when a critical point p is non-degenerate we have that $e_+ + e_- = n$. This allow us to define a particular coordinate system in p.

Lemma 2.4 (Morse Lemma [Mil58])**.** *Let M^n be a manifold and $f : M \to \mathbb{R}$ a function at least twice continuously differentiable, if p is a non-degenerate critical point of f then $H_f(p)$ has signature $(n-i, i)$ for a natural number i and there exists U_p, open neighbourhood of p, with coordinate system $\varphi : U_p \to \mathbb{R}^n$ such that*

$$f(q) = f(p) - \sum_{j=1}^{i} x_j^2 + \sum_{j=i+1}^{n} x_j^2, \; q \in U_p, \varphi(q) = (x_1, \dots, x_n).$$

With abuse of notation we have written f instead of $f \circ \varphi^{-1}$. The number i is called the index *of p for f.*

Using Lemma 2.4 we have that a non-degenerate critical point p with index 0 is a local minimum for f and one with index n is a local maximum. We can now introduce the notion of *Morse function.*

Definition 2.5 ([Mil58])**.** Given a compact and closed manifold M^n and a function f on it, we say that f is a *Morse function* if all critical points of f are non-degenerate. In particular, every critical point has a well-defined index.

It is clear that any Morse function needs to be at least twice differentiable.

Given a function f on a manifold M we have that a point p can be non-critical, critical non-degenerate or critical degenerate. The former type is always present, otherwise we would have that f is constant and so every point is critical and degenerate as $\nabla_f \equiv 0$. The assumption on the absence of the latter type could seem a strong one, but when M is closed and compact it is not. It can be proven that, if M is closed and compact then the set of Morse functions on M is dense in the set of all at least twice differentiable functions on M, that is, given a function f there exists g, small perturbation of f such that g is a Morse function, [Mil58].

In Figure 2.1 we present two Morse function on two different spaces embedded in $\mathbb{R}^3$. The function in both cases is the height function, $f : M \subset \mathbb{R}^3 \to \mathbb{R}$ such that $f(x, y, z) = z$, which is Morse. A critical point in these spaces is a point where the tangent plane is horizontal. As we can see, the points D_1, D_2 are minima for f, so they have index 0 and as they are global minima any path starting in D_1, or D_2, will end at a point with higher value with respect to f than D_1, or D_2. The opposite happens for A_1 and A_2 as they are critical points of index 2 and in particular global maxima. For both A_1 and A_2 we have that each path starting at them will always end at a point with lower value with respect to f and so this defines a 2-dimensional space of descending directions with respect to f. The presence of two critical points of index 1 in the torus and not on the sphere is a consequence of the difference in terms of their topology, as we will explain briefly.

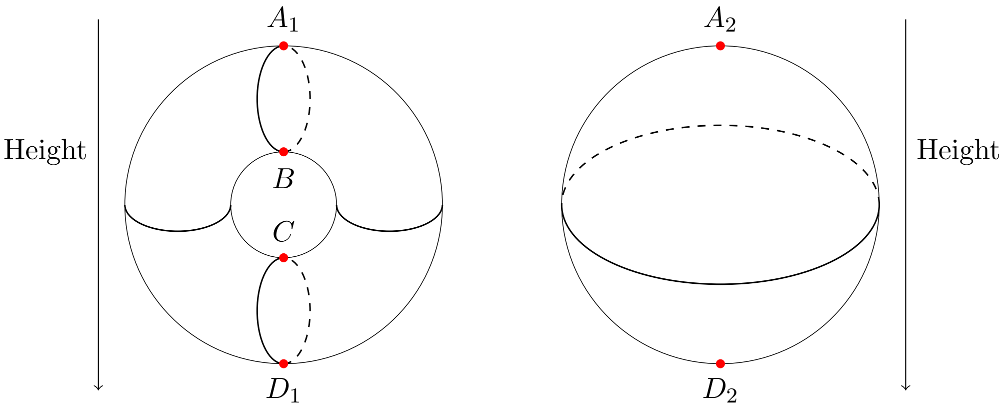


FIGURE 2.1: Morse function (vertical height) on a torus and on a sphere, and associated critical points (red). The number and type of critical points, for any Morse function, is a topological invariant of the torus.

As already anticipated when M is closed and compact we have that any non-Morse function f, at least twice continuously differentiable on M, can be transformed, after a small enough perturbation, to a Morse function. This property is a consequence of the following theorem on well-separation of the critical points of Morse functions.

Theorem 2.6 ([Mil63]). *Given M a compact and closed manifold, if f is a Morse function on M then f has a finite number of critical points and they are all isolated.*

Proof. Given a critical point p for f from Lemma 2.4 there exists a neighbourhood $\widehat{U}$ and a coordinate system $\varphi : \widehat{U} \to \mathbb{R}^n$ such that

$$f(q) = f(p) - \sum_{j=1}^{i} x_j^2 + \sum_{j=i+1}^{n} x_j^2, \text{ with } q \in \widehat{U},$$

where again with abuse of notation we wrote f instead of $f \circ \varphi^{-1}$. In $\widehat{U}$ there is only one critical point for f, which is p, so the critical points are isolated.

Let $\mathcal{C} = \{p_i\}_i$ be an infinite set of (non-degenerate) critical points. We know that M is Hausdorff, so there exists convergent non-constant sequence $\{q_0, q_1, \dots, q_n, \dots\} \subset \mathcal{C}$, let its limit point be $\widehat{q}$. For continuity and f Morse, $\widehat{q}$ is (non-degenerate) critical, so as from before it is isolated. We obtain then that a limit point of a non-constant convergent sequence in a Hausdorff space is isolated, which is absurd. □

The finiteness and separation of critical points is essential both for the study of topological invariants of M and for the definition of a descending flow on M with respect to f.

The topological invariants are studied using the *cell-decomposition* of M. That is, given a space M we decompose it as the result of gluing together *cells* of different dimension, where a cell of dimension p is the p-dimensional ball in $\mathbb{R}^{p+1}$. Note that as we are considering a topological decomposition these cells can take different shapes, as we can stretch and bend them, so a 1-cell can be the full circle with one point removed but as well a segment, similarly a 2-cell can be a full triangle. The topological spaces that can be described using such decomposition are called *CW-complexes*, for an introduction to these spaces we refer the reader to [Whi49]. The following result tells us how a Morse function $f : M \to \mathbb{R}$ gives always a precise cell-decomposition of M.

Theorem 2.7 ([Mil58])**.** *If f is a Morse function on compact and closed manifold M then M is homotopy equivalent to a CW-complex with as many p-cells as critical points of index p for f.*

Intuitively Theorem 2.6 is saying that each Morse function gives us information on the topology of M, thanks to the homotopy equivalence, such as upper bounds for the *Betti numbers* of M or its *Euler characteristic*. In Figure 2.1, for example, the presence of only one minimum in both spaces tells us that they are connected. A consequence of Theorem 2.7 are the so-called *Morse inequalities* [Mil63] which tell us that the topology of M determines the minimum possible number of critical points a Morse function can have. This can be seen in Figure 2.1 where on a sphere we could define a Morse function without any critical point of index 1, whilst this is impossible for a torus.

As anticipated given a compact and closed manifold M and f a Morse function on it we can construct a flow that detects the descending paths of f for any particle on M. Such flow is constructed using the notion of 1-*parameter group of diffeomorphisms*, also called *flow*.

Definition 2.8 ([Mil63])**.** Given a differentiable manifold M, we define a 1-*parameter group of diffeomorphisms* as:

$$\varphi : \mathbb{R} \times M \to M,$$

such that:

- For $t \in \mathbb{R}$ we have $\varphi_t = \varphi(t, \cdot) : M \to M$ is a diffeomorphism;
- For $t, s \in \mathbb{R}$ it is true that $\varphi_{s+t} = \varphi_s \circ \varphi_t$.

The flow we want to construct will be a map $\varphi : \mathbb{R} \times M \to M$ such that it is a 1-parameter group and in addition if $q = \varphi(t, p)$ with $t > 0$ then $f(q) < f(p)$. Intuitively, we are asking that for a particle at position p in M and $t > 0$ the map $\varphi_p(t) = \varphi(t, p)$ smoothly moves p decreasing its value with respect to f. To ensure such property, given $p \in M$ we consider the following vector

$$X_p = -\frac{\nabla_f(p)}{|| \nabla_f(p) ||_g},$$

assuming the tangent space, to whom X_p belongs, has a Riemannian metric g, [Car92]. Then, we define in p the following differential equation for $\varphi_p : \mathbb{R} \to M$

$$\begin{cases} \frac{d\varphi_p}{dt}\Big|_{\varphi_p(t)} = X_{\varphi_p(t)}, \\ \varphi_p(0) = p. \end{cases} \tag{2.1}$$

It is easy to notice that such equation can not have solution when p is critical, as at time 0 the equation is not defined, as it is not X_p. However, for any non-critical point p there exists a small enough positive $\epsilon(p)$ such that $\varphi_p : (-\epsilon(p), \epsilon(p)) \to M$ is well-defined and uniquely determined by the condition in $t = 0$ [Lan72]. An extension of these local solutions can be aptly constructed as none of them will ever hit any critical point. In fact, if $\varphi_t(p) = q$ and q is critical then we reach a contradiction as X_q is not defined. We can see then that φ_p will flow infinitely close for increasing value of t to a critical point and for decreasing values to another. These critical points cannot be the same as $f(\varphi_p(t)) < f(\varphi_p(s))$ when $t > s$. In particular thanks to $||X||= 1$ we have that if $q = \varphi(t,p)$ then $f(q) = f(p) - t$, whenever q is defined. We can formalise such behaviour at infinity as follows.

Let K be a compact subset of M without critical points, then by compactness for any point p there exists an interval I of $\mathbb{R}$ such that φ_p solution of Equation 2.1 is uniquely defined in I. The critical points of f are finite so as well its critical values. Let ϵ be a positive real number and consider for any c critical value of f the set $V_{c,\epsilon} = (c-\epsilon, c+\epsilon)$. If U_ϵ is the union of $f^{-1}(V_{c,\epsilon})$ among all critical values of f, then $K_\epsilon = M \setminus U_\epsilon$ is a compact set and clearly no critical point belongs to it. As previously said, we have that exists an interval I such that $\varphi : I \times K_\epsilon \to M$ is uniquely defined. We are able to extend such uniqueness to M using the following.

Lemma 2.9 ([Mil58]). *A smooth vector field on M which vanishes outside of a compact set $C \subset M$ generates a unique 1-parameter group of diffeomorphisms of M.*

Consider $L_\epsilon = K_\epsilon \cup \varphi(I \times K_\epsilon) \subset M$, a compact neighbourhood of K_ϵ without critical points, and define $\rho : M \to \mathbb{R}$ smooth function such that

- $\rho(p) = \frac{1}{||\nabla_f(p)||_g}$, if $p \in K_\epsilon$;
- $\rho(p) = 0$, if $p \notin L_\epsilon$.

Define now the vector

$$Y_p = -\rho(p) \cdot \nabla_f(p),$$

which clearly coincide with X_p inside K_ϵ and it is zero outside L_ϵ so Lemma 2.9 holds. That is, there exists $\varphi : \mathbb{R} \times M \to M$ such that when p belongs to K_ϵ its path $\varphi_p(t)$ flows downhill, respect to f, for positive value of t and when p is outside L_ϵ we have

$\varphi(t,p) = p$ for any t, as $Y_p = 0$. As ϵ can be as small as we want, this tells us that we can define the unique solution of Equation 2.1 on a compact set as close as we want to each critical point. Such flow φ leaves only the points in a $\epsilon-$neighbourhood of a critical point fixed, with ϵ arbitrary.

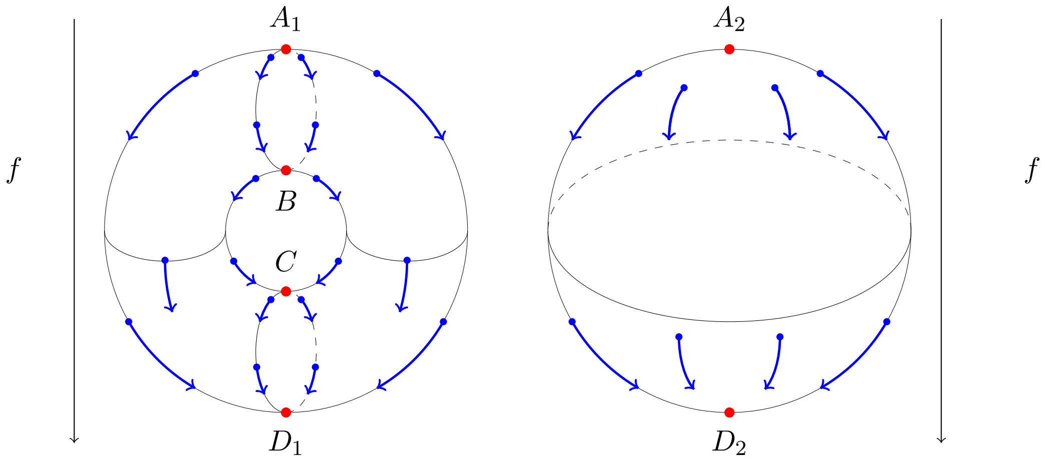


FIGURE 2.2: Morse function on a torus and on a sphere, with associated Morse flow.

2.2 Discrete Morse Theory

As shown there is a very close relationship between the topology of a manifold M and the critical points of a differentiable function $f : M \to \mathbb{R}$. We recall now its adaptation applied to any simplicial complex (or more general cell complex) as presented in [For98]. There have been other adaptations of Morse Theory on combinatorial spaces. For example, a *Morse theory of piecewise linear functions* appears in [BK87] and the very powerful *Stratified Morse Theory* was developed by Goresky and MacPherson, [GM88]. With respect these theories, Forman's Morse Theory takes a slightly different approach. Rather than choosing a suitable class of continuous functions on our (discrete) spaces to play the role of Morse functions, we will be assigning a single number to each cell of our complex, and all associated processes will be discrete. Hence, this theory is referred in the literature as *Discrete Morse Theory.* Of course, these different approaches to combinatorial Morse Theory are not distinct. One can sometimes translate results from one of these theories to another by "smoothing out" a discrete Morse function, or by carefully replacing a continuous function by a discrete set of its values. We will not focus on this possibility but instead we will show how discrete Morse theory is capable, even without introducing any continuity, to recreate on combinatorial spaces, a complete theory that captures many of the intricacies of the smooth theory, and can be used as an effective tool for a wide variety of combinatorial and topological problems.

In order to describe it we need to recall what are considered the constructing blocks of the discrete spaces we will analyse, the *simplices*.

Definition 2.10 ([For98])**.** Given a natural number k and S, a set of $k+1$ points linearly independent in an euclidean space $\mathbb{E}$, a *k-dimensional simplex* σ is the convex hull of S in $\mathbb{E}$. We call *p-face* of σ, with $p < k$, a p-dimensional simplex whose $(p+1)$ points are a subset of S.

Examples of simplices can be nodes and segments, as the elements of a graph, but as well triangles and tetrahedrons. As we can see in Figure 2.3 in a tetrahedron we have triangles (4), links (6) and nodes (4) as faces.

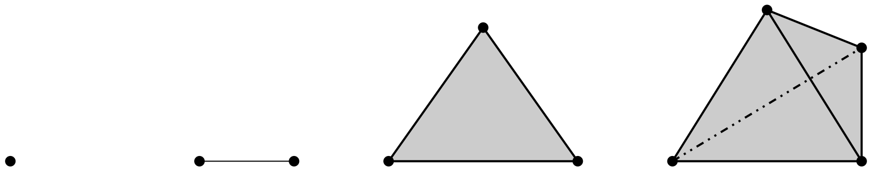


FIGURE 2.3: Examples of simplices of different dimensions

.

Note that given a finite set of simplices we can always regard them as lying in some $\mathbb{R}^n$, with n big enough. Using this property, we can consider intersection, union and other topological, or combinatorial, operations between two or more simplices as they were performed in such $\mathbb{R}^n$.

In [For98] the definition of discrete Morse theory is given in full generality for any discrete space. However as such general framework is beyond our scope, we will describe it only on a particular class of discrete spaces which are called *finite simplicial complexes*.

Definition 2.11 ([For98])**.** A *simplicial complex* K is a set of simplices such that:

- Any p-face of a k-simplex in K is also in K;
- Given two simplices $\sigma_1, \sigma_2 \in K$ their intersection is either empty or a p-face of both them.

We call K *finite simplicial complex* if the number of its simplices is finite.

An example of a simplicial complex can be the following:

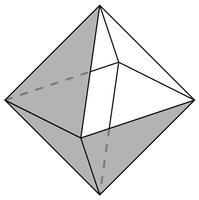


FIGURE 2.4: A 2-dimensional simplicial complex with only 3 simplices of dimension 2, coloured in grey.

A simplicial complex can be considered as a topological space with the induced topology from the $\mathbb{R}^n$ in which its simplices are lying, see [Whi49]. Note that a finite simplicial complex can be seen as the result of gluing together a finite number of simplices of different dimension but as well as a discretisation of a finite dimensional manifold as in Figure 2.4. In particular, a simple finite graph, that is a graph without multi-links or self-loops, is a finite simplicial complex and the 1-skeleton of a finite simplicial complex, that is, the subset of simplices of dimension either 1 or 0, is always a simple finite graph.

Given a finite simplicial complex we can define a *discrete function* on it, as a map that assigns to each simplex a real number, $f : K \to \mathbb{R}$. As in the differentiable case, to obtain a Morse function we require some properties on a discrete function. Those will permit, as before, to extrapolate information on the topology of K but as well to define a descending discrete flow.

Definition 2.12 ([For98])**.** Consider a finite simplicial complex K and a discrete map f on it, we call f *discrete Morse function* if for any pair of simplices $\sigma \subset \tau$ we have $f(\sigma) \leq f(\tau)$ and for any simplex σ it is true that

1. $F_1(\sigma) := \{v \mid f(v) = f(\sigma) \text{ and } v \subset \sigma\}$ has at most one element, and
2. $F_2(\sigma) := \{\tau \mid f(\tau) = f(\sigma) \text{ and } \tau \supset \sigma\}$ has at most one element.

This property tells us that each Morse function has to be compatible almost everywhere with the dimension, that is, if σ is a face of τ, so with lower dimension, then it has also lower value than $f(\tau)$, with at most one exception. From [For98], we know that this exception can be present either in $F_1(\sigma)$ or in $F_2(\sigma)$. Moreover if $\sigma \subset \tau$ and $f(\sigma) = f(\tau)$ then the dimensions of σ and τ differ by 1, see [For98]. We can now introduce the definition in this framework of *critical simplex* for a Morse function.

Definition 2.13 ([For98])**.** Given a finite simplicial complex K and a Morse function f on it, a simplex σ is called *critical simplex of f* if $F_i(\sigma) = \emptyset$ for $i = 1, 2$ with *index* equal to its dimension.

Note that we can always define a Morse function on a finite simplicial complex such that every simplex is critical, for example $f : K \to \mathbb{R}$ where $f(\sigma) = \dim(\sigma)$. These functions, even not interesting to infer topological properties of K, show that at least one Morse function always exists on any simplicial complex K.

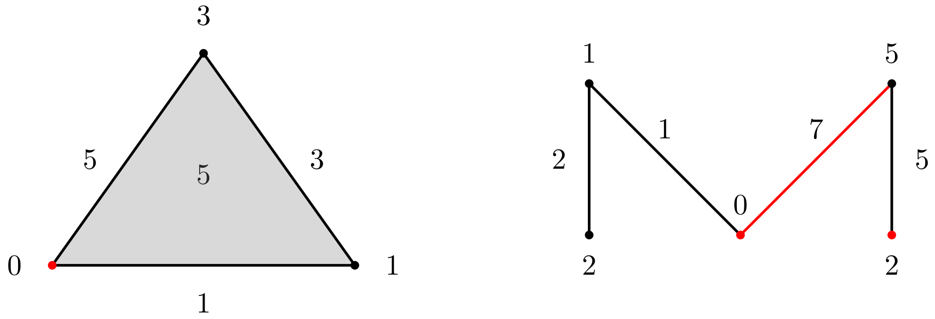


FIGURE 2.5: Morse function on two discrete spaces. Each simplex σ has at most one element in $F_1(\sigma) \cup F_2(\sigma)$. The critical simplices are drawn in red.

Given a Morse function f its properties allow us to divide K in three disjoint sets

- $A_1 = \{\sigma \in K \mid F_1(\sigma) \neq \emptyset\}$;
- $A_2 = \{\sigma \in K \mid F_2(\sigma) \neq \emptyset\}$;
- $H = \{\sigma \in K \mid F_1(\sigma) \cup F_2(\sigma) = \emptyset\}$.

Clearly H is the set of critical simplices and A_1 and A_2 are in bijection, as when $F_1(v) = \sigma$ we have $F_2(\sigma) = v$. Thanks to such pairing, we can state the discrete counterpart of Theorem 2.7.

Theorem 2.14 ([For98]). *Let f be a Morse function on a simplicial complex K then K is homotopy equivalent to a CW-complex with as many p-cells as critical simplices of index p for f.*

Again this theorem gives us a way to infer some topological invariants of K as its Betti number or Euler characteristic, and again the topology of K gives restriction on the number of critical simplices of any Morse function. In particular, as each simplex of dimension p is in fact a p-cell we have that given a simplicial complex K and a Morse function $f : K \to \mathbb{R}$, each non-critical simplex for f can be effectively removed to obtain the decomposition of Theorem 2.14, this operation is also called *collapsing*.

Given a discrete Morse function f, the induced subdivision of K in three disjoint sets gives us the tools to define a *discrete gradient vector* on K.

Definition 2.15 ([For98]). Let K be a finite simplicial complex and f a Morse function on it, we define the *discrete vector field* of f as the map $\mathcal{V} : K \to K \cup \{0\}$ with

$$\mathcal{V}(\sigma) = \begin{cases} \tau, & \text{if } F_2(\sigma) \neq \emptyset, \text{ and } \tau \in F_2(\sigma), \\ 0, & \text{otherwise.} \end{cases}$$

Note that if $\tau = \mathcal{V}(\sigma) \neq 0$ then τ has higher dimension and $f(\tau) = f(\sigma)$, in addition $\mathcal{V}^2(\sigma) = 0$ for any σ. In Figure 2.6 the discrete vector field V has been pictured with blue arrows, so that if $\mathcal{V}(\sigma) = \tau$ then an arrow is drawn with tail on σ and head on τ.

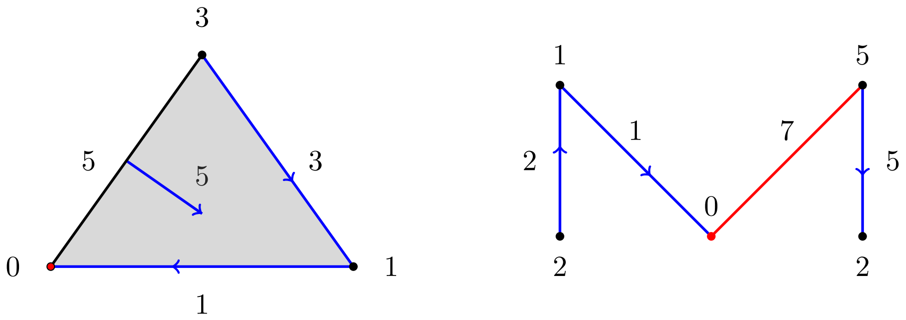


FIGURE 2.6: Morse functions on a discrete space, the discrete vector field (blue) and critical simplices (red).

In the graph on the right we can see that the operator $\mathcal{V}$ maps a non-critical vertex v to a link (v, w) with same value where w has always lower value than v. A similar behaviour can be observed in the triangle on the left, where the link e valued 5 is mapped to the triangle and each link different from e has value lower than 5. We will recall now how to construct a discrete counterpart of the Morse flow for manifolds. This flow will highlight the directions where the Morse function $f : K \to \mathbb{R}$ decreases, for example in the graph in Figure 2.6 it will map a non-critical node v to the one with lower value and connected to v with a non-critical link. In order to present it, we need to recall first the definition of *boundary map* and *chain complex* associated to K.

Definition 2.16 ([For98])**.** Given any orientation on K, for each p we define the $\mathbb{R}$-vector space $C_p(K)$ generated by all simplices of dimension p. The *chain complex* associated to K is the $\mathbb{R}$-vector space $C_*(K) = \bigoplus_p C_p(K)$, or equivalently the $\mathbb{R}$-vector space generated by the set K. A *boundary operator* on $C_*(K)$ is a linear map ∂ such that $\partial^2 = 0$ and $\partial(C_p(K)) \subset C_{p-1}(K)$ for any p, where $C_{-1}(K) = 0$.

Suppose that an indexing on the nodes of K is given so we can express any simplex σ formed by $k+1$ nodes as $v_{j_0}, v_{j_1}, \dots, v_{j_k}$ with $j_s < j_t$ for $s < t$. We can then define a boundary operator ∂ on $C_*(K)$ such that if $\sigma = [v_{j_0}, v_{j_1}, \dots, v_{j_k}]$

$$\partial(\sigma) = \sum_{t=0}^{k} (-1)^t [v_{j_0}, \dots, \widehat{v_{j_t}}, \dots, v_{j_k}], \tag{2.2}$$

where the simplex $[v_{j_0}, \dots, \widehat{v_{j_t}}, \dots, v_{j_k}]$ is the simplex generated by all nodes of σ but v_{j_t}. From now on we will assume that such boundary operator is defined on the chain complex of K.

The set K generates $C_*(K)$ as $\mathbb{R}$-vector space we can consider the inner product $<,>$ such that for any $a, b \in C_*(K)$ we have

$$< a, b >= \sum_{\sigma \in K} a_\sigma b_\sigma, \text{ for } a = \sum_{\sigma \in K} a_\sigma \sigma, \ b = \sum_{\sigma \in K} b_\sigma \sigma. \tag{2.3}$$

To extend linearly the map $\mathcal{V} : K \to K$ to $C_*(K)$ we need to assign an orientation to each $\tau = \mathcal{V}(\sigma) \neq 0$. We do so assuming that $< \partial\mathcal{V}(\sigma), \sigma >= -1$ for any $\sigma \in K$ such that $\mathcal{V}(\sigma) \neq 0$. Such assumption resembles the definition of X_p in terms of the gradient, see Equation 2.1. We are now ready to recall the definition of *discrete Morse flow* associated to a Morse function.

Definition 2.17 ([For98])**.** Given a discrete vector field $\mathcal{V}$ associated to a Morse function f, define the *discrete Morse flow* associated to f as

$$\varPhi := \mathrm{Id} + \mathcal{V}\partial + \partial\mathcal{V} : C_*(K) \to C_*(K).$$

For any p we have that $\varPhi$ restricted to $C_p(K)$ is a linear endomorphism with properties that resemble the ones we have in the differentiable framework.

Lemma 2.18 ([For98])**.** *Given $\varPhi$ the discrete flow associated to a Morse function f the following holds*

1. *$\varPhi$ is a chain operator, $\partial\varPhi = \varPhi\partial$;*
2. *For any σ critical we have that $< \varPhi(\sigma), \sigma >= 1$, whilst $< \varPhi(\sigma), \sigma >= 0$ when σ is non-critical;*
3. *If $< \varPhi(\sigma), \tau >\neq 0$ and $\sigma \neq \tau$ then $f(\tau) < f(\sigma)$;*
4. *There exists a N such that $\varPhi^N = \varPhi^{N+1}$.*

If K is a simple graph, we can see that $\varPhi$ always maps nodes to nodes, that is given a node v its image $\varPhi(v)$ is in fact a node and not a linear combination of nodes. Furthermore, for any $s, t \in \mathbb{N}$ and $s < t$ we have that $f(\varPhi^s(v)) \leq f(\varPhi^t(v))$ where the equality holds only if $\varPhi^s(v) = \varPhi^t(v)$. This can be seen as the discrete counterpart of the property we have for the Morse flow $\varphi : \mathbb{R} \times M \to M$, where $\varphi(t, p) = p$ for any ϵ-extension of φ from K_ϵ to M if and only if p is critical and otherwise $f(\varphi(t, p)) < f(p)$ when $t > 0$.

In both differential and discrete case the definition of Morse flow depends entirely on the gradient, resp. discrete vector field. Suppose f, g are two (discrete) Morse functions that differ only by a real constant c. We have that the critical points of f and g coincide but as well do their flows. The critical points can be extracted directly from the gradient ∇_f as they will be the points p where $\nabla_f(p) = 0$. Similarly, in the discrete case the critical simplices σ are such that $\sigma \in (K \setminus \mathrm{Im}(V)) \cap \mathrm{Ker}(V)$. On the other hand, it is true that two different gradients, or discrete vector fields, cannot be induced by the same Morse function, as they are uniquely determined by it. Instead of focusing the attention on Morse function, one could shift it to the gradient-like vectors, resp. discrete vector fields, and check when they can be originated from a Morse function. In general, a discrete vector field, not associated to a Morse function, on K is defined as follows.

Definition 2.19 ([For98]). Given a finite simplicial complex K, a *discrete vector field* is a map $\mathcal{V} : K \to K \cup \{0\}$ such that $\mathcal{V}^2 = 0$ and if $\tau = \mathcal{V}(\sigma) \neq 0$ then $\sigma \subset \tau$ with $\dim(\tau) = \dim(\sigma) + 1$.

We can extend again $\mathcal{V}$ to the chain complex of K as before and determine when it is induced by a Morse function f, that is $\mathcal{V} \equiv \mathcal{V}_f$.

Theorem 2.20 ([For98]). *Given a finite simplicial complex K and a discrete vector field $\mathcal{V}$ on it, let $\varPhi = \mathrm{Id} + \partial\mathcal{V} + \mathcal{V}\partial$. There exists N such that $\varPhi^N = \varPhi^{N+1}$ if and only if $\mathcal{V}$ is a discrete vector field of a Morse function f.*

When K is a simple graph the hypothesis of Theorem 2.20 is equivalent to ask that no $\varPhi$-cycle is present, that is, if v is not critical then $\varPhi^k(v) \neq v$ for any $k > 0$. We can likewise use discrete Morse theory on graphs, 1-dimensional finite simplicial complexes, with respect to a Morse function, determined by a choice of node-annotation and link-weight, to extrapolate the underlying flow. As this flow comes from a Morse function, we know that each node will flow ultimately to a critical node. If we cluster then together nodes that flow to the same critical node we obtain a partition of the graph, with the additional property that for each cluster only one critical node is present and it is the one with minimal value with respect to f. The Morse flow of f will detect then not only a decomposition of the graph in separate regions, but also for each critical node it will reveal the respective basin of attraction.

Chapter 3

A novel clustering algorithm: `Morse`

As already explained we can approach cluster analysis using a graph representation. That is, we can identify objects with nodes of a simple graph $G = (V, E)$ and connect them depending on their interaction. This interaction is also used to construct a *weight function* on the links, $w : E \to \mathbb{R}$, which could tell us how strong the connection is, how much we rely on it or also how similar two objects are. As anticipated, a graph is a simplicial complex of dimension 1 and discrete Morse theory applies. Other authors have worked on such possibility, see [KKM05, RWS11], focussing merely on the network topology. Our intention is to employ Morse theory to study the effect of a hierarchical structure of node annotation and determine which basin of attraction such structure induces.

Arbitrary link weight and node annotation functions are unlikely to be Morse functions. In fact, the node annotation function $f : V \to \mathbb{R}$ should be such that each node v has at most one link with weight equal to $f(v)$. We present here an algorithm, `Morse`, that constructs a Morse function from an annotated weighted network, detecting the *basins of attraction* induced by the annotation. Intuitively, we want to detect for each node, when possible, its best connection that in addition links it with a more influential node, so that we can construct an annotation-ascending flow which will be a Morse flow.

3.1 Method

Given a finite and simple graph G, let V be the set of its nodes and E of its links respectively. Without loss of generality, we will from now on assume that G is connected. The clustering algorithm `Morse` we will now describe is mostly dependent on how we describe the *importance* of a node and the *strength* of a link, or more precisely how we define, given two nodes (or two links), which is higher (or stronger). These comparisons are formalised by the notion of preorder.

Definition 3.1 ([Sch03])**.** Given a set S a *preorder* $\preceq$ is a binary relationship that is

Reflexive: for all a in S we have $a \preceq a$;

Transitive: if $a \preceq b$ and $b \preceq c$ then $a \preceq c$ for all a, b, c in S.

One can think of preorders as order-like relationships in which ties are allowed. Note that a preorder may not be total, that is, there could exist elements a, b for which we cannot establish if $a \preceq b$ or $b \preceq a$, so a, b are not comparable. We will write $a \prec b$ if $a \preceq b$ holds and $b \preceq a$ does not exist.

Using preorders we can now describe the notion of strength of links and importance of nodes. Assume a preorder $\preceq_E$ (resp. $\preceq_V$) is given on E (resp. on V). We say then that a link (v, w) is *stronger* than another (v, s) if $(v, w) \prec_E (v, s)$. Similarly, a node v is *higher* than another w if $v \prec_V w$. As our algorithm wants to find the best connection for each node v, when we restrict ourselves to the out-link at v

$$E_v = \{e \in E \mid \exists w \in V \ s.t. \ e = (v, w)\},$$

and assume that $\preceq_E$ is a total preorder on it. This allow us to compare every pair of links in E_v, so for any two links of the form $(v, s), (v, t)$ we can always say which is stronger or if they are equally strong. Technically, such assumption is not necessary and we can still apply `Morse` with $\preceq_E$ partial preorder also in the subsets E_v. Note that at this point, we are not including any type of information on the weight function $w : E \to \mathbb{R}$ or the annotation $f : V \to \mathbb{R}$. In this sense, the algorithm `Morse` we will describe now can be considered more as a clustering method, where different choices of preorders generate different algorithms, see Section 3.2 and Section 5.3.

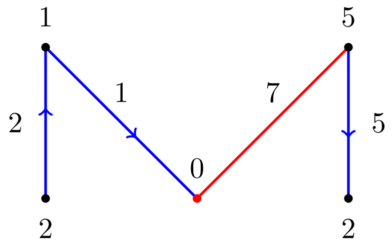
In its most general form `Morse` will proceed as follows. For each node v, we select its *maximal link* $e_v = (v, s)$, if it exists, defined as the unique maximal link in E_v, w.r.t. $\preceq_E$, that connects v with a higher node, w.r.t. $\preceq_V$. We then construct a map $\Phi : V \to V$ such that it maps every node v to the other end of its maximal link e_v or to itself when e_v does not exists. The pseudo-code of our algorithm, called `Morse`, can be summarised as follows.

```
Input: graph $G = (V, E)$, link preorder $\preceq_E$, node preorder $\preceq_V$
Output: Morse flow $\Phi\colon V \to V$

foreach $v \in V$ do
    if maximal link $e_v = (v, s)$ of $E_v$ exists then
        $\Phi(v) \leftarrow s$
    else
        $\Phi(v) \leftarrow v$
    end
end
```

Algorithm 1: Morse flow algorithm.

Consider for example G a graph with a Morse function f on it and define the link preorder $\preceq_E$ such that $e_1 \prec_E e_2$ if $f(e_2) < f(e_1)$ with $e_1, e_2 \in E$. If the node preorder $\preceq_V$ is such that $v \prec_V s$ when $f(s) < f(e) = f(v)$ with $e = (v, s) \in E$, then we can see that the flow Φ we construct with `Morse` is exactly the Morse flow of f as in Definition 2.17.



Consider now the subset of nodes $V_f = \{v \mid \Phi(v) = v\}$. For each $v \in V_f$ we can construct the subgraph $\mathcal{T}_v = (S_v, E_v)$ where

$$\begin{cases} S_v = \{w \mid \exists n \geq 0 \text{ s.t. } \Phi^n(w) = v\}\,, \\ E_v = \{(w, \Phi(w) \mid w \in S_v\} \subset E. \end{cases}$$

Note that such $\mathcal{T}_v$ will be exactly the basin of attraction of v as anticipated.

The map $\Phi : V \to V$ is an importance increasing map along the links of G and it can be proven that there exists a Morse function with flow exactly Φ.

Theorem 3.2. *Let $G = (V, E)$ be a finite simple graph with $\preceq_E, \preceq_V$, link and node preorders respectively, and associated Morse flow $\Phi\colon V \to V$. Then:*

(i) There exists a Morse function that induces Φ as it stabilises, that is, for a $N \geq 0$ we have $\Phi^N = \Phi^{N+1}$;

(ii) $\{S_v \mid v \in V_f\}$ is a partition of V;

(iii) $\mathcal{T}_v$ *is a directed (link directions given by the flow) rooted tree with root* v*;*

(iv) Within T_v*, the node* v *is the unique maximal node w.r.t.* $\preceq_V$*.*

Proof.

(i) Consider $v \in V$ and a Φ-sequence of nodes $\gamma = \{v = v_0, v_1, \dots, v_k\}$ where $v_i = \Phi(v_{i-1}) = \Phi^i(v)$. If $v \in V_f$ clearly for any i we have $v_i = v$. Suppose that $v \notin V_f$ and that for $0 < i < j$ we have $v_i = v_j \notin V_f$. We would have by definition of Φ that $v_i \prec_V v_{i+1} \prec_V \cdots \prec_V v_j = v_i$, leading to contradiction as by transitivity we have $v_i \prec_V v_i$. This implies that if γ contains a repetition, $v_i = v_j$ for some $i < j$ then $v_i \in V_f$ and so $v_i = v_{i+1} = \cdots = v_j = \cdots = v_k$. As G is finite, say $|V| = n$, we can have at most a sequence of n nodes without any repetition. Let γ be an arbitrary sequence of $n+1$ nodes, $\gamma = \{v_0, v_1, \dots, v_n\}$. We know that a repetition will appear in γ and so there exists $i < n$ such that for any $j > i$ we have $v_i = v_j \in V_f$. In conclusion, for any $v \in V$ we obtain $\Phi^n(v) \in V_f$ so $\Phi^n(v) = \Phi^{n+1}(v)$ and Theorem 2.20 applies.

(ii) As just stated for every $v \in V$ we have that $\Phi^n(v) = w \in V_f$. If in the sequence $\{\Phi^i(v)\}_{i=0}^n$ we have that $\Phi^j(v) \in V_f$ for some j then $\Phi^{j+t}(v) = \Phi^j(v)$ for any $t > 0$. In conclusion, $v \in S_w$ with w unique, that is $\{S_w\}_{w \in V_f}$ is a partition with clusters disjoint pairwise.

(iii) Since all links in $\mathcal{T}_v$ are of the form $(w, \Phi(w))$ and $\Phi^n(w) = v$, a cycle would imply that there exists a sequence that does not end in v, a contradiction. All links are directed and point towards the root v, by the above discussion.

(iv) For the definition of Φ and because $\Phi^n(w) = v$ for any $w \in \mathcal{T}_v$ we have that $w \prec_V v$ for any $w \neq v$. So v is maximal in $\mathcal{T}_v$ w.r.t. $\preceq_V$.

□

Remark 3.3. The meaning of higher node and strong connection, that is, the definition of $\preceq_V$ and $\preceq_E$, determines the structure of the `Morse` clustering solution. Those preorders can be defined separately from the annotation and the weight function or be a mixture of them. Suppose the annotation function is a centrality measure, $f : V \to \mathbb{R}$, and the weights descend from a distance-like function. We can define then

$$\begin{cases} (v,s) \preceq_E (v,t) & \text{if} \quad w(v,t) \leq w(v,s) \\ v \preceq_V s & \text{if} \quad f(v) \leq f(s). \end{cases} \tag{3.1}$$

In this way a node v will be mapped to a node s if the latter is strictly more central than the former and it is the closest one. Note that in this case if the closest node to v is a less or equally central node then $\Phi(v) = v$. Other example of preorders will be presented later on in Chapter 4 and Section 5.3.

A preorder $\preceq_V$ that is based on a hierarchical structure of the annotation function will allow our algorithm to unravel such hierarchy and construct basins of attraction encoded by a rooted tree as shown in Figure 1.2.

Given $f : [a, b] \to \mathbb{R}$ continuous function, we can draw its graph Γ_f where each point has coordinates $(x, f(x))$.

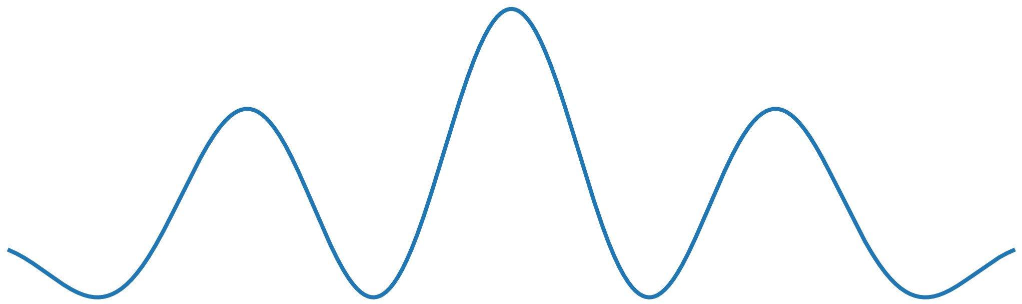


FIGURE 3.1: The graph of $f(x) = \cos(x)^2 e^{-x^2}$

Let $I = \{x_i\}_{i=1}^n$ be a uniformly distributed discrete set of real values in $[a, b]$ we can construct a *point cloud* V on the curve described by Γ_f, using as node set $\{\Gamma_f(x_i)\}_{i=1}^n$. We connect then two nodes p, q if their path-length in Γ_f, $d_{\Gamma_f}(p, q)$, is close enough to their euclidean distance, $d_{\mathcal{E}}(p, q)$, that is, $d_{\Gamma_f}(p, q) \leq \frac{5}{4} d_{\mathcal{E}}(p, q)$. We assign weight to a link (i, j) equal to such path-length from $(x_i, f(x_i))$ and $(x_j, f(x_j))$, that is equal to

$$w(i, j) = \left| \int_{x_i}^{x_j} \sqrt{1 + f'(x)^2} dx \right| .$$

As each point $v \in V$ is a point in $\mathbb{R}^2$ we can consider the annotation function $f : V \to \mathbb{R}$ such that $f(v) = y$ when $v = (x, y)$. Assume now that stronger connections have lower weight and higher nodes instead larger annotation, so exactly as in Equation 3.1. Our algorithm detects then the minimal path for a particle on Γ_f to move uphill using the points in V. Each cluster found by `Morse` will be the basin of attraction of a local maximum of f.

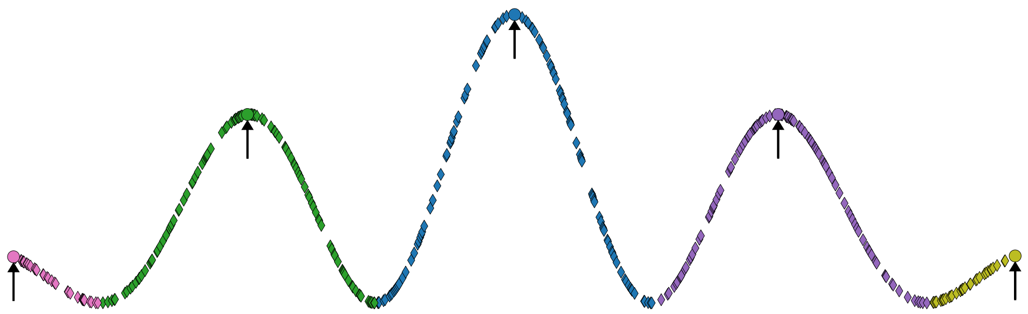


FIGURE 3.2: `Morse` clusters for an uphill flow on Γ_f coloured differently, non-critical nodes with a diamond shape and critical ones with a larger circle shape.

Reversely if we now consider more influential those node with lower y-coordinate then the clustering solution will show the basin of attraction of local minima of f.

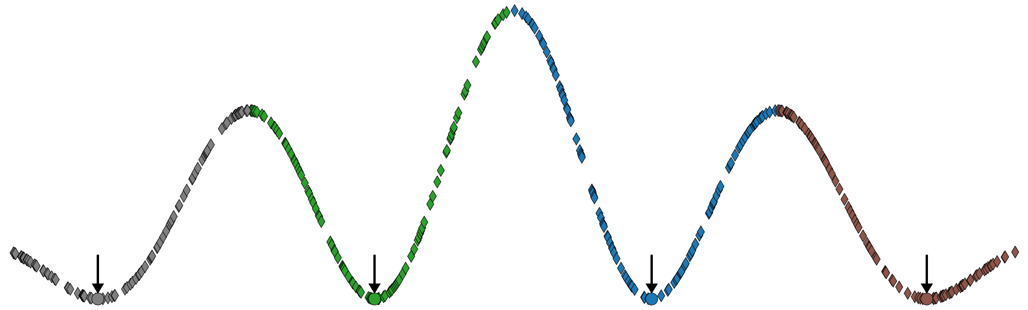


FIGURE 3.3: `Morse` clusters for an downhill flow on Γ_f coloured differently, non-critical nodes with a diamond shape and critical ones with a larger circle shape.

Computationally `Morse` is a local algorithm, because of the definition of *strongest edge*, and its order of complexity is $O(m)$ where m is the number of links in G. The linear correlation between time spent and number of links results clear from Figure 3.4, where each point $p = (x, y)$ represents an execution of `Morse` on a random graph with x links that took y seconds.

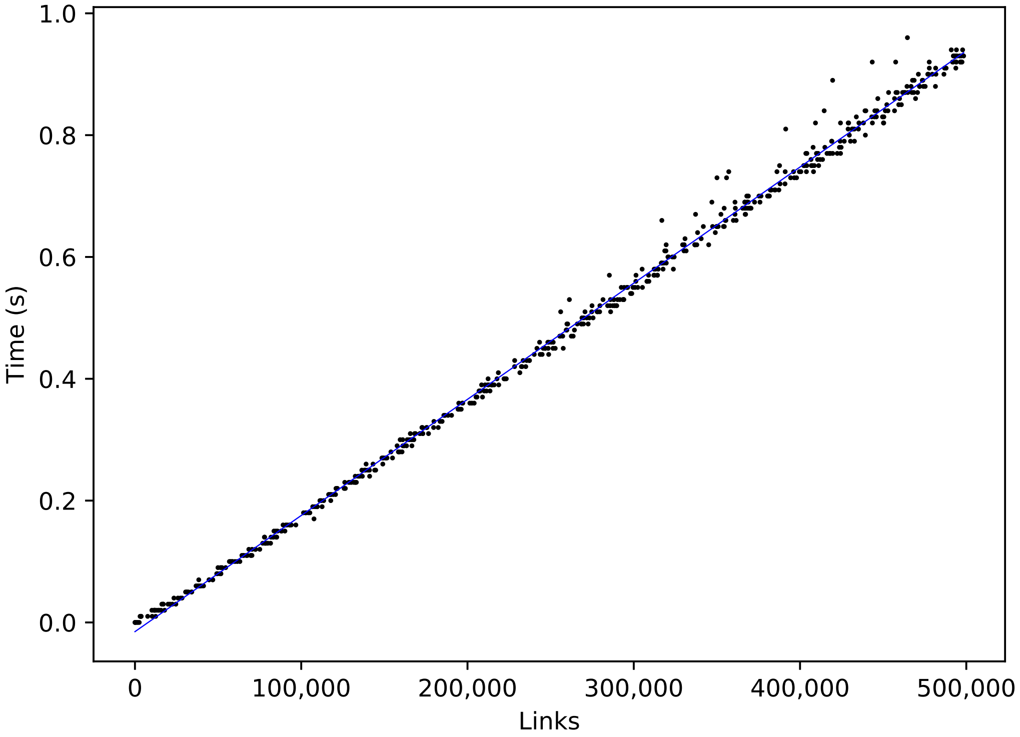


Figure 3.4: Computational complexity of `Morse` over randomly generated graphs with a linear regression blue line.

In particular we have implemented the search of each node strongest link using a link iteration approach, so at each link we update the strongest link for both its ends, in this way the computational space used by `Morse` is $O(n)$. To compute the clusters we need an additional time $O(n)$, as we compute the connected components of a forest of rooted trees (see Theorem 3.2) which has $n-k$ links, with $n = |V|$ and k the number of critical nodes, using the *Breadth-first search* [ZUS59, HT73].

In conclusion `Morse` is a fast and local algorithm capable of detecting the best route flow in an annotated graph, with the property that in each route we have an annotation-increasing flow. In other words `Morse` uses the weight function to determine the best connection available for a node v and the annotation to check if such connection is an improvement for v, that is, it connects v with a more central/influential node. This combination of weight and annotation could overcome the locality of `Morse` when the annotation represents a global measure of node importance. In such case in fact, `Morse` will search locally the maximal link for each node, but at the same time it will employ a global understanding of the graph (given by the annotation) for the choice of the best route. As explained, we do not have simply a clustering of the nodes, but a deeper understanding of each cluster structure in terms of rooted directed tree. This structure will give us not only the Morse flow for each cluster in terms of best increasing annotation route, but it will also provide a representative (the root) for each cluster. In this sense, it could be possible to analyse a clustering solution in terms of the relationships between each node in a cluster and its representative instead of considering each node the same as it happens often with quality functions for clustering, [BDA09].

3.2 Benchmarking

We performed different types of benchmarking analysis and procedure against our algorithm. The first we present here is based on the *LFR benchmark* presented in [LF09], where using fixed parameters a graph is generated together with underlying *ground-truth communities*. Those are the allegedly true clusters that a "good" clustering algorithm should recognise. In this sense the LFR benchmark is useful for studying algorithm performances for different choices of parameters. The LFR procedure generates a graph using a similar procedure to configuration model graphs [FLNU18], with an additional mixing parameter μ_t that enhances or reduces the density of communities in terms of internal links. We studied the performance of our algorithm against such benchmark using the same parameters used in [LF09], where a similar study is delivered for different clustering techniques and the *Normalised Mutual Information* is used as index of correctness [PTVF07]. In addition, thanks to our algorithm's low computational complexity, we were able to establish performances also in the case of very large LFR-generated network with 50,000 and 100,000 nodes. We also tested `Morse` against the Erdos-Renyi model [ER60] and the Scale-free model [Bar09] in order to check that when no real communities structure is present the algorithm does not deliver artificial communities.

In [LF09] a generative model for graph is proposed as a benchmark for clustering algorithms. The graphs produced are supposedly similar to those we encounter in real-world datasets, that is, scale-free in terms of degree distribution but also in terms of community size. The generative method proceed as follows. Given n nodes, we assume first that the degree distribution is determined by a discrete probability function p_d of the power-law family, so $p_d(x) \propto x^{-\tau_1}$. This means that in the network high-degree nodes will be rare and low-degree ones instead frequent. After, we define the ground-truth communities, using again a discrete probability function p_c from the power-law family distribution, $p_c(x) \propto x^{-\tau_2}$. In this case we force large communities to be scarce and instead small ones to be abundant. In both cases, as the size of a community and the degree of node are discrete values, we consider discrete probability function and we also add bounds for both minimum and maximum size/degree. Such assumptions on degree and community size distribution are deduced by results and observation of real-world datasets [Bar16]. Each node is then assigned to a community and connected using a configuration model ([FLNU18]) together with a *mixing parameter* μ_t. This parameter controls how each community is well inter-connected, as for each node v we impose that

$$d_{out}(v) \approx \mu_t d(v),$$

where $d_{out}(v)$ is its outer degree, with respect its community, and $d(v)$ instead the total degree. The performance of a clustering algorithm can be studied using different values of μ_t, as we expect high performances when μ_t is low and worse ones as we start increasing it. In particular, we have from [LFK09] that the communities should be well-defined till

$$\mu_t \leq 1 - \frac{n_{\max}}{n},$$

with $n_{\max}$ the size of the largest community [LF09].

The LFR procedure lacks of weight and annotation function so we defined them as follows. As weight, we choose a function similar to the *Jaccard similarity* function presented in [RR15]. For two connected nodes, we assign weight

$$w(p, q) = |N(p) \cap N(q)|,$$

with $N(s)$ the set of neighbours of v. As we assume that a link has always weight greater than 0, we put $w(p, q) = \epsilon$ whenever $N(p) \cap N(q) = \emptyset$, with $0 < \epsilon < 1$. This weight function is telling us that the more two nodes have common neighbours the more they are similar. We used this weight function to define as well the annotation as $f(v) = (f_1(v), f_2(v)) : V \to \mathbb{R}^2$ such that

$$\begin{array}{lcl} f_1 : V \to \mathbb{R}, & v & \mapsto & \max_{(v,s)\in E} w(v, s), \\ f_2 : V \to \mathbb{R}, & v & \mapsto & \sum_{(v,s)\in E} w(v, s). \end{array}$$

If we consider f_1 and f_2 as centrality measures, then the former tends to value more those nodes that have one or more out-links with high weight, while the latter instead values more the overall out-link weights. In order to test the importance or effect of annotation on the output of `Morse`, we tested the algorithm also giving as annotation a random indexing I on V. We choose then as preorder $\preceq_V$ the lexicographic preorder for f, that is, w is ‘higher’ than v if $f_1(v) < f_1(w)$ or if $f_1(v) = f_1(w)$ and $f_2(v) < f_2(w)$. When the annotation was a random indexing, we used as preorder the canonical order we have on $\mathbb{R}$. On the links instead as we wanted to value more links with high weight we choose as link preorder $(v, s) \preceq_E (v, t)$ if

$$\begin{cases} d(v, s) \leq d(v, t) \text{ and} \\ v \preceq_V t \preceq_V s \text{ when } d(v, t) = d(v, s). \end{cases} \tag{3.2}$$

So when we have a preorder tie we solve it, when possible, using $\preceq_V$, that is, the annotation function.

The correctness of a clustering solution is defined in [LF09] with the *Normalised Mutual Information*. Given two clusterings $\mathcal{X}$ and $\mathcal{Y}$ over a graph with n nodes, we define their mutual information as

$$I(\mathcal{X},\mathcal{Y}) = \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} \frac{\mid x \cap y \mid}{n} \log_2\left(n\frac{\mid x \cap y \mid}{\mid x \mid \mid y \mid}\right).$$

The mutual information can be seen as the information that $\mathcal{X}$ and $\mathcal{Y}$ share, that is, how much we reduce the uncertainty about one knowing the other. The drawback of this measure is that it is not 'normalised', that is, it does not assign a value of 1 when $\mathcal{X}$ and $\mathcal{Y}$ are equal and 0 when they are totally dissimilar. Different normalised versions of mutual information have been proposed to satisfy this property; for coherence we used the same as presented in [LFK09] defined as

$$NMI(\mathcal{X},\mathcal{Y}) = \frac{I(\mathcal{X},\mathcal{Y})}{H(\mathcal{X}) + H(\mathcal{Y})},$$

where H is the entropy of a clustering defined as

$$H(\mathcal{X}) = \sum_{x\in\mathcal{X}} \frac{\mid x \mid}{n} \log_2\left(\frac{\mid x \mid}{n}\right).$$

In the following pictures we can see how `Morse` performed using different annotations against the parameters proposed in [LF09].

- ◆ LFR-generated graphs with 1000 nodes and small community size bounds set to 10 and 50;
- ▼ LFR-generated graphs with 1000 nodes and community size bounds set to 20 and 100;
- ■ LFR-generated graphs with 5000 nodes and community size bounds set to 10 and 50;
- • LFR-generated graphs with 5000 nodes and community size bounds set to 20 and 100.

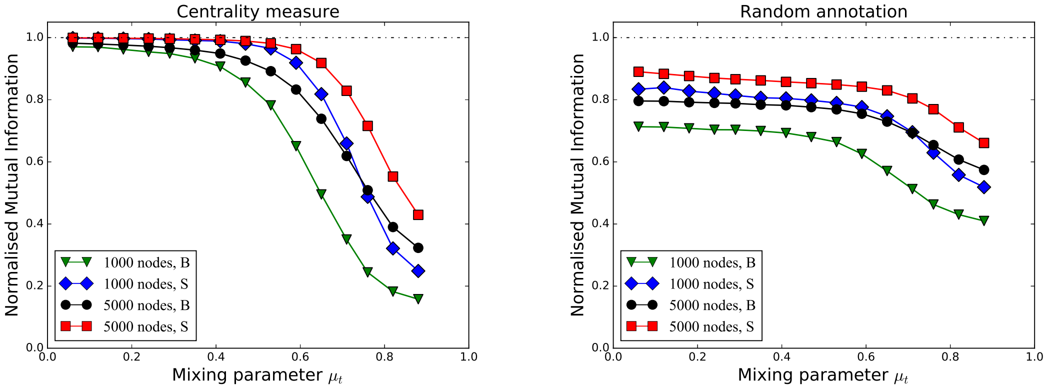


FIGURE 3.5: Tests of the `Morse` algorithm on LFR benchmark graphs with undirected and unweighted links, where each point represents an average performance over 100 realisations of the model.

As one can see the annotation allowed `Morse` to detect precisely the ground-truth communities for μ_t less than 0.5. A random annotation, or equivalently a randomly generated indexing of the nodes, gives us a steady but lowest score. However, because the indexing does not carry any type of centrality measure, its peaks will be randomly placed in the network and this will determine a very low cluster size on average, as we can see below, where on the y-axis we have the number of clusters found by the algorithm.

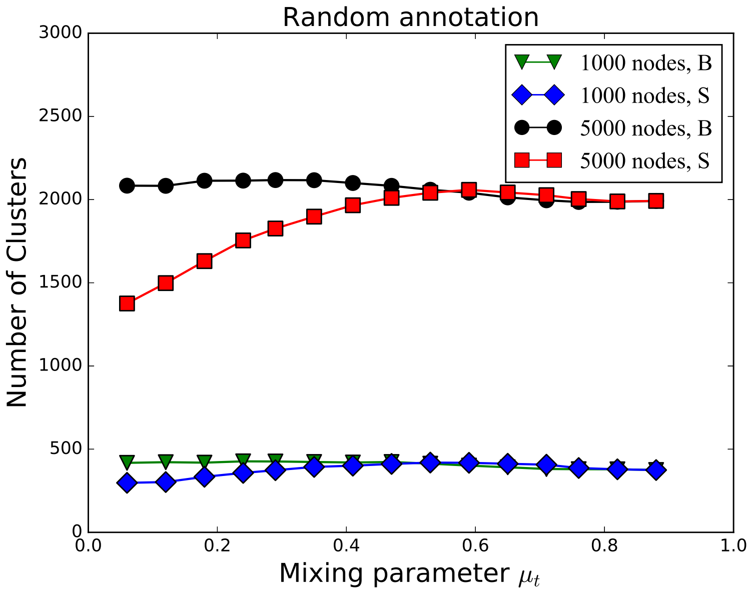


FIGURE 3.6: In the LFR models with 1000 nodes the average cluster size for `Morse` is 2 whilst when we have 5000 nodes it is at most 5 among all μ_t.

In [LF09] we can find different analysis of several clustering algorithms. `Morse` shows similar results of *Blondel et al.* [BGLL08]. That is, `Morse` performance is similar to the "best performing algorithms on the LFR undirected and unweighted benchmark"

[LF09]. In addition, as Infomap [RB08] and *Blondel et al.*, `Morse` is computationally fast and so we can carry out another set of tests on the LFR benchmark for two additional cases:

- 🟥 $n = 50{,}000$, maximum degree 200, $p_c(x) \propto x^{-1}$, with maximum community size 1000 and minimum community size 20;
- ♦ $n = 100{,}000$, maximum degree 200, $p_c(x) \propto x^{-1}$, with maximum community size 1000 and minimum community size 20;

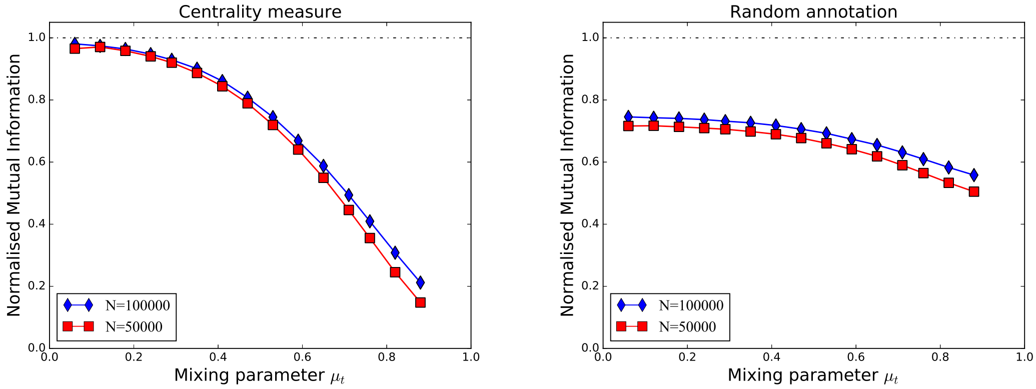


FIGURE 3.7: Tests of the `Morse` algorithm on LFR benchmark graphs with undirected and unweighted links, where each point represents an average performance over 100 realisations of the model.

As we can see the performance of `Morse` for this set of parameters is worse than before, similarly to *Blondel et al.*, see [LF09]. The choice of annotation penalises the clustering, as the increasing number of nodes produces too many peaks for the same community. Also if the performances of `Morse` for this set of parameters were similar to *Blondel et al.*, with `Morse` we can investigate also the annotation increasing flow that comes with a cluster and understand the cluster not as a simple aggregate of objects or nodes, but as a rooted tree structure. This is an additional feature that can be employed successfully to understand not only in which cluster the analysed data splits but as well what type of structure an annotation defines on it.

As in [LF09] we also carried out the analysis for the well-known Erdos-Renyi [ER60] and Scale-free [Bar09] generative models. We kept the weight and annotation functions as in the previous analysis and similarly the preorders were unchanged. Both models were constructed over a set of 1000 nodes using an average degree parameter and configuration model for the link creation. As we can see again, the random annotation creates clusters with average size 2. Using instead the centrality measure, we do not detect any community, as wanted, for low value of average degree. As soon we start increasing it, we obtain more and more peak and few artificial modules are detected.

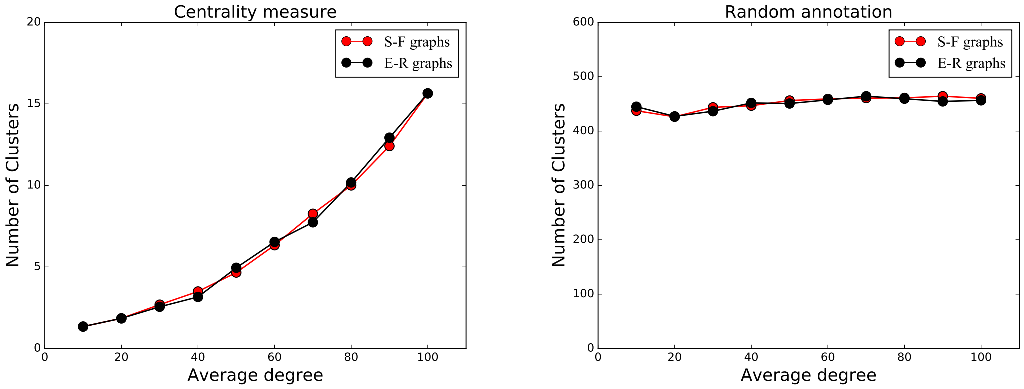


Figure 3.8: Tests of the `Morse` algorithm on LFR benchmark graphs with undirected and unweighted links, where each point represents an average performance over 100 realisations of the model.

Chapter 4

Application for an asthma study

Topological Data Analysis (TDA) is a powerful, unsupervised machine learning tool suitable for analysis of high-dimensional datasets [LSL$^+$13, NPL$^+$15, NLC11]. Application of TDA via the Mapper algorithm [SMC91] generates a TDA network, a compressed representation of high-dimensional data with major features embedded where similar data points are grouped into nodes, and nodes with common data points are connected by links. Clusters reported within TDA networks represent data points with similarity in their major features defined by *lenses* such as principal component analysis (PCA) [BH89], multidimensional scaling (MDS) [Kru64] or T-distributed Stochastic Neighbour Embedding (t-SNE) [MH08]. A lens is a function that maps the dataset into a vector, assigning to every data-point a single real number; such a function as can come from geometry, statistics, or can be user-defined. The lens determines how the data is then clustered and can bring various aspects of the data “into focus”. These clusters can, therefore, be used to describe sub-groups according to key features of a dataset, such as gene expression profiles that represent conserved mechanistic pathological pathways, thereby enabling stratification of diseases by mechanisms rather than crude clinical data. However, clusters within TDA networks have typically been delineated by eye [NLC11, LBC$^+$14, HZS$^+$15], without algorithmic reproducibility. Some studies have delineated clusters using community detection via the network representation, as implemented by the now widely used Ayasdi Python SDK. The limitation of this approach is that the algorithm only analyses connectivity between nodes without considering the density of data points clustered within nodes. This information can be regarded as an annotation on the nodes and the TDA network can be visualised by colouring (see Figure 4.2). On the TDA network we can, therefore, apply `Morse` and describe it as a 3D map of data points clustered around peaks that represent conserved profiles of major features. For example, gene expression profiles can be represented as a 3D map, where peaks within the TDA network represent patient groups with common underlying molecular pathological features of disease.

We used data from the U-BIOPRED (Unbiased BIOmarkers for the Prediction of respiratory disease outcomes) study, the largest multi-centre asthma programme to date, involving 20 academic institutions, 11 pharmaceutical companies and patient groups and charities, with the aim to improve understanding of the complex molecular mechanisms underpinning asthma and identify useful biomarkers, see [WGB+12, FMB+15, LAA+16, LDML+17]. The clinical and molecular profiles of a large cohort of severe asthmatics, mild asthmatics and control participants were measured, applying multiple 'omics technologies to identify sub-types of asthma with differences in molecular biology. The Ayasdi TDA software platform was used to simplify relationships between patients according to differential gene expression profiling of peripheral blood as measured by the Affymetrix GPL570 platform, with a *normalised correlation metric* [SRB07] and two *Neighbourhood Lenses* based on t-SNE [HR03, MH08]. Following the same procedure of [LSL+13], these two lenses map each data-point in $\mathbb{R}^2$ and given the parameters N, called *resolution*, and p, called *gain*, we determine a grid-subdivision of the region of $\mathbb{R}^2$ occupied by the data-points. Each subregion, with the shape of a square, will have the same area and an uniform overlap of p with the neighbouring subregions. Each non-empty subregion will determine a cluster, and so a node in the TDA network. In our analysis, the resolution for each Neighbourhood Lens was set to 100 and the overlap was set to 0.6. Within the two clusters identified by hierarchical clustering and previously reported [BBS+17], sub-clusters were evident in the TDA networks but were not investigated or described in the previous works. Here, we have applied Morse-clustering to further delineate these clusters and investigate the molecular phenotypes of the identified sub-types of asthma. The following work is currently submitted to Nature Communications .

4.1 Materials and Methods

U-BIOPRED is a multi-centre prospective cohort study, involving 16 clinical centres in 11 European countries. The adult cohort of the U-BIOPRED study consists of four groups [SSF+15]:

i) severe asthmatics/non-smoking ($n = 311$),

ii) severe smoking asthmatics ($n = 110$),

iii) mild/moderate non-smoking asthmatics ($n = 88$)

iv) non-smoking non-asthmatics ($n = 101$).

Blood samples were collected from 606 study participants (309 non-smoking severe asthmatics, 110 smoking severe asthmatics, 87 non-smoking mild/moderate asthmatics and 100 non-smoking non-asthmatic individuals). RNA was isolated using the PAXgene

Blood RNA kit (Qiagen, Valencia, CA) with on-column DNase treatment (Qiagen). RNA integrity was assessed using a Bioanalyzer 2100 (Agilent Technologies, Santa Clara, CA). Samples with RNA integrity number greater than 6 were processed for microarray as described (19) and hybridised onto Affymetrix HT HG-U133PM+ arrays (Affymetrix, Santa Clara, CA) using a GeneTitanR according to Affymetrix technical protocols. The microarray data are deposited in GEO under GSE69683.

The complete U-BIOPRED study comprised analyses by several 'omics platforms. Exemplary results relevant to individual TDA cluster described below were taken to show the value of the applied bioinformatics approach. After RNA and microarray quality control and exclusion of samples due to discrepancies with demographic data, the 498 samples available for analysis were randomised into training ($n = 328$) and validation sets ($n = 170$).

The transcriptomics data were clustered by topological data analysis (TDA), as in [BBS+17], using Ayasdi Platform with a norm correlation metric and two Neighbourhood lenses. Correlation was measured using normalised values for the expression of each probeset and metric norm-correlation. The space for clustering was generated using 100 bins in each dimension according to t-SNE calculated vectors and 60% overlap between neighbouring bins; this generated a network where each node represents a non-empty bin and two nodes are connected if their respective bins have non-empty overlapping, see Figure 4.1.

4.2 `Morse` clustering of high patient density regions of TDA graphs

Using the Ayasdi Platform, the magnitude of nodes was represented by a colour heatmap where the colour spectrum from blue to red represents the range from the lowest to highest levels. Our generation of a TDA graph of the blood transcriptomics data of U-BIOPRED participants was created with people as rows, clustered by similarity in their column profiles; columns here are microarray probes. The two-dimensional uncoloured TDA graph resulted from clustering people with similar transcriptomes into nodes and links between nodes with shared people. When the TDA graph was coloured by the number of rows (people) per node, regions of the TDA graph were clearly shown to have high or low densities of people, providing a numerical value when graph values were exported from Ayasdi Platform via commands in the Python SDK. The graph G exported will be then a graph where each node represents a cluster and two nodes are connected if they share at least one patient. If we imagine the magnitude of a node as an node annotation, we could apply `Morse`. Moreover, if such annotation is the height of a node in a $3D$ space we can picture it in a three dimensional space as follows.

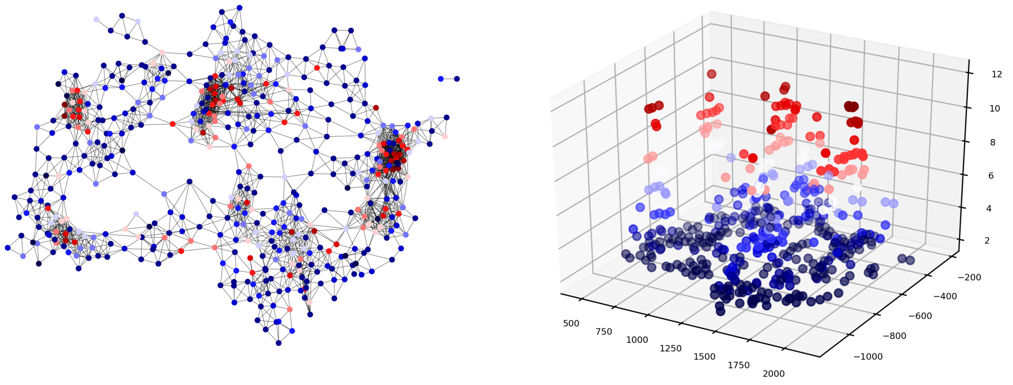


FIGURE 4.1: Mapper cluster-network coloured with a heat map based on cluster-size.

In our analysis in addition to the node magnitude we add also another annotation to determine the importance of a node. As we used the normalised correlation metric in relation to the Ayasdi lenses, we considered it as a valuable metric to compare the 'compactness' of a node. Given $p, q \in \mathbb{R}^n$, two row vectors representing two patients, and $||\cdot||_2$ the euclidean distance in $\mathbb{R}^n$, following [SRB07] we define the *normalised correlation distance* d between p, q as

$$d(p,q) = 1 - \frac{(p-\bar{p})(q-\bar{q})}{||p-\bar{p}|_2||q-\bar{q}||_2},$$

where $\bar{p}$ (resp. $\bar{q}$) is the constant vector with entries equal to the mean of the entries of vector p (resp. q).

The *correlation* of a cluster C_i is then defined as

$$Corr(C_i) = 1 - 2\frac{\sum_{p\in C_i}\sum_{q\in C_i} d(p,q)}{n_i(n_i-1)},$$

with n_i is the number of patients in C_i. The value $1 - Corr(C_i)$ can be consider as the averaged sum of all distances between any two patient row-vectors in C_i. In this way, a node C_i has higher correlation if it is more 'compact' with respect to the norm-correlation distance. We defined then the annotation function g such that for each node C_i we have

$$g(C_i) = \left(size(C_i) + \sum_{C_j \cap C_i \neq \emptyset} size(C_j) \right) \cdot Corr(C_i)$$

Using this annotation which is a mixture of intrinsic centrality measure (given by the neighbourhood of C_i) and external (given by the cluster normalised correlation), we have a deeper understanding of the network and its nodes, as we are now considering an annotation that values more large and highly intra-correlated nodes but connected to large nodes. We can see in Figure 4.2 how the heat map colouring induced by g on the

TDA network changes respect to Figure 4.1 and similarly when we consider g as node height in a $3D$ space.

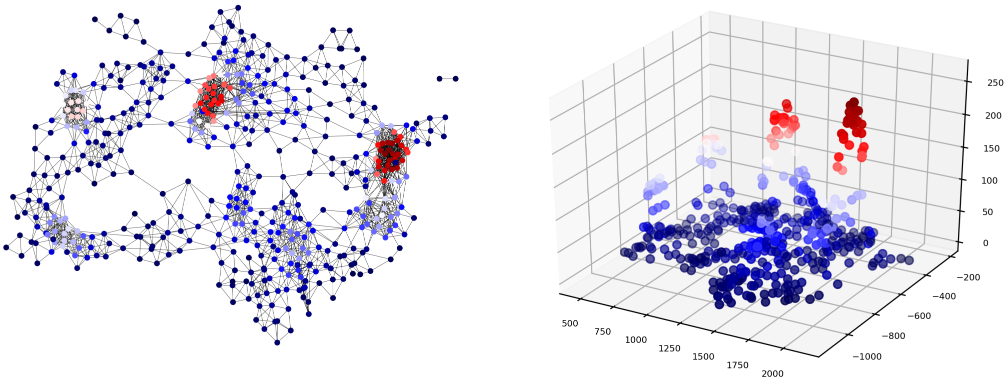


FIGURE 4.2: A different colouring and lifting using intra-cluster norm correlation.

On the graph G we defined then the annotation on a node C_i as

$$f(C_i) = (size(C_i), g(C_i)).$$

To determine which node were a peak for this annotation we choose the *lexicographic order* so that a node A was higher than node B if it was larger, in terms of people inside A, and when they had identical size the highest one was the one with larger $g(C_i)$. The use of g for the annotation allowed us to give context to where a node lies within the broader topology, effectively 'smoothing' the data, decreasing noise and allowing identification of the most prominent peaks. Analysing the TDA network of blood gene expression of 498 U-BIOPRED study participants, this `Morse` algorithm identified 9 large clusters. Seven of these were reproducibly identified in an independent validation set using ROC analysis of logistic regression classification models. That is, we trained a logistic regression classification model using a *training set* (subset of the entire patient-set) and we used this model to predict the belongingness of the remaining patients (*validation set*) to a `Morse`-cluster. For each `Morse`-cluster we then computed its reporter operating characteristic (ROC) curve created by plotting the true positive identified patients (correctly located inside the cluster) against the false positive ones (erroneously located inside the cluster). The identified clusters represented groups of patients with significant differences in the activation of pathways related to inflammation, including pathways associated with Glucocorticoid receptor (GR) signalling, Type (T)-2, T-1 and T-17 inflammatory responses.

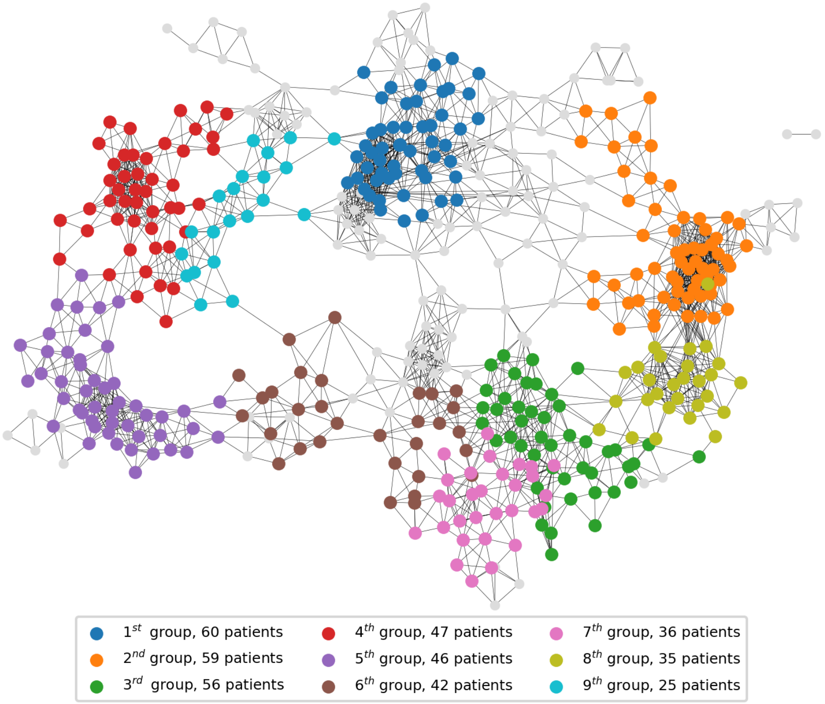


FIGURE 4.3: The 9 largest `Morse` clusters in terms of number of patients.

4.3 Results and Discussion

There is no consensus on how to test the success of unsupervised clustering algorithms. It is possible to calculate the average silhouette value for clusters [Rou87] (tightness of a cluster based on a distance metric). However, this is not appropriate for clustering in a TDA network since the lenses select key features for network construction. We therefore applied logistic regression (supervised machine learning) [KK10] to test the tightness of the clusters according to key features identified by logistic regression. A logistic regression model was trained on a pre-defined training set of ($n_{train} = 328$) and the classification accuracy tested on a validation data set ($n_{val} = 170$). Accuracy of the logistic regression reflects reproducibility in the clustering, that is, robust classification assigned by clustering results in accurate classification of validation data by an independently trained logistic regression model. The model we choose was the *Elastic Net* model [ZH05], a linear model trained on a matrix X with a convex combination of the l_1 norm ($||\cdot||_1$) and the l_2 one ($||\cdot||_2$) as prior regularisers, and the function we want to minimise is

$$\min_{w\in\mathbb{R}^k} \frac{1}{2n}||Xw-y||_2^2+\alpha\rho||w||_1+\frac{\alpha(1-\rho)}{2}||w||_2^2, \tag{4.1}$$

where k is the number of features we select to perform the logistic regression and X is a matrix $n_{train} \times k$. As we will perform a One vs Rest analysis, that is, test each Morse cluster against the rest of the data set, the y vector in the formula above will be a binary vector with i^{th} entry 1 if a patient p belongs to at least one of the nodes in the Morse cluster chosen and 0 otherwise. Elastic Net is useful when there are multiple features which are correlated with one another. Other methods as *Lasso* [Han09] or *Ridge* [HK70] are likely to pick one of these at random, while Elastic Net is likely to pick both. If we set $\rho = 1$ then we obtain the Lasso model, whilst if $\rho = 0$ the model will coincide with the Ridge model. A practical advantage of trading-off between Lasso and Ridge is that it allows Elastic Net to inherit some of Ridge's stability under rotation.

To determine k, given a `Morse`-cluster, we first ordered the features using a p-value obtained with a *t-test statistic* [Stu08]. That is, a statistical test which tells us, with a value from 0 to 1, if two samples have identical average (expected) values, in our case if for a specific feature the average value is identical for the patients inside the cluster and outside of it. We then performed the Elastic Net analysis for each `Morse`-cluster and each combination of k features (with $1 \leq k \leq 10$) chosen between the highest 25 in terms of the p-value of the t-test statistic. For each `Morse`-cluster we selected the combination which prediction of cluster belongingness was more accurate. That is, given the vector w, that satisfies Equation 4.1, we compute the predicted belongingness vector $p = X_{val}w$, where X_{val} is the features matrix corresponding to the validation set, and the accuracy of the prediction is measured using the R^2 *score* of the prediction [DS14]

$$R^2(p,t) = 1 - \frac{||p - t||_2^2}{||t - \bar{t}||_2^2},$$

where t is a binary vector with i^{th} entry 1 if a patient p belongs to at least one of the nodes in the Morse cluster chosen and 0 otherwise.

The ROC area under the curve (AUC) for the 9 clusters ranged from 0.63 to 0.97 (see Figure 4.4), suggesting very good to excellent prediction of cluster classification in the validation set based on a logistic regression model identifying predictors of the cluster in the training set. However, the $7^{th}, 8^{th}$ and 9^{th} clusters were small ($n = 36, 35, 25$, respectively), with correspondingly low representation in the training and validation sets which resulted in ROC curves whose shapes were not smooth and may have represented overfitting. While all clusters were used in subsequent analysis of features, we acknowledge the relative weakness of $7^{th}, 8^{th}$ and 9^{th} clusters compared to the other clusters.

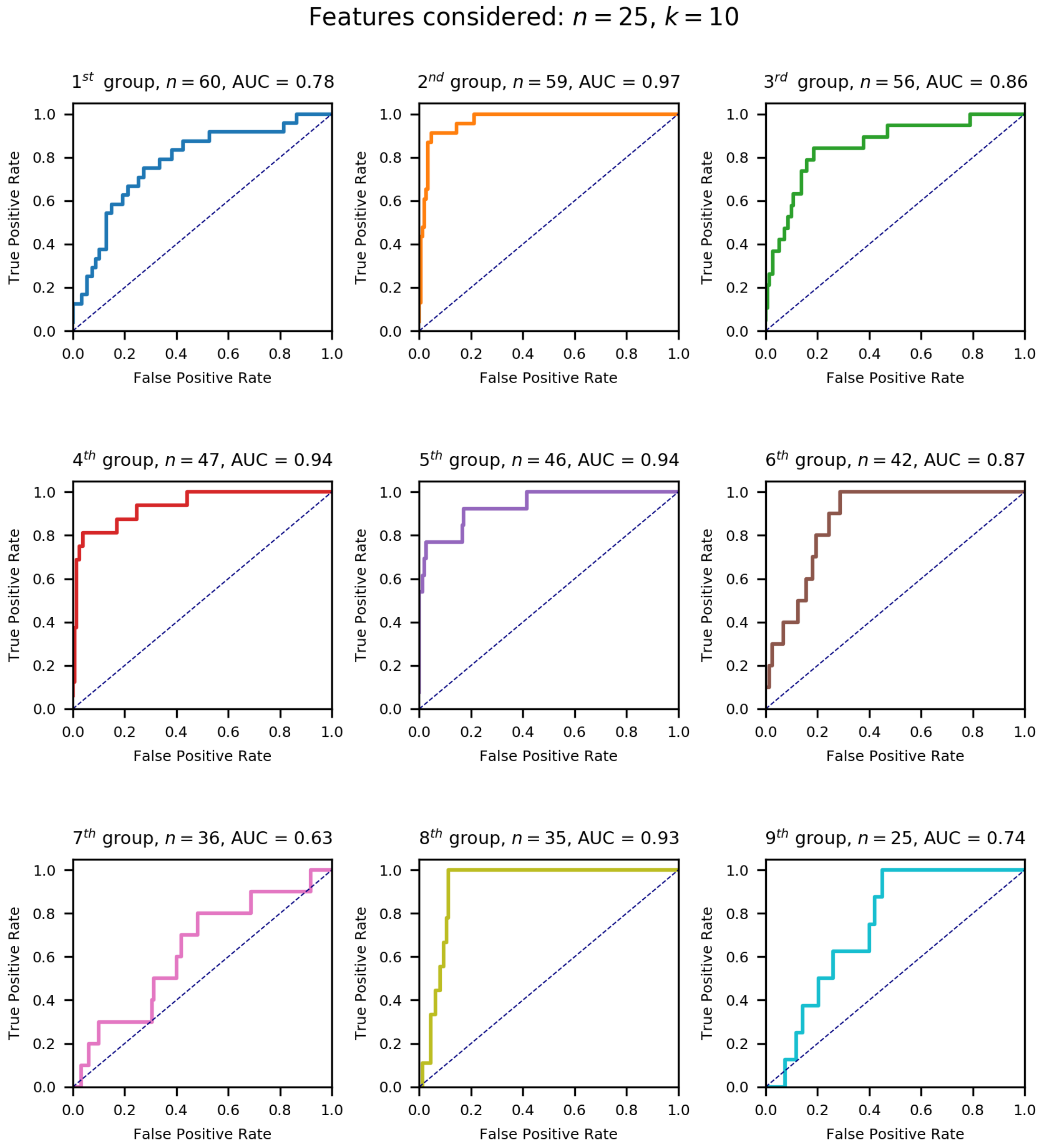


FIGURE 4.4: ROC curves of the 9 biggest clusters found with `Morse`.

In summary, this study has demonstrated the value of Morse-clustering on annotated TDA graphs by identifying groups of patients with signatures of discrete molecular pathology. We propose that Morse clustering of annotated graphs can be applied to TDA graphs of patient data to identify sub-phenotypes of disease, thereby offering new insights into disease mechanisms and stratification of patients for more targeted drug development based on molecular mechanisms.

Chapter 5

Morse Theory and an Impossibility Theorem for Graph Clustering

As clustering is an infamously ill-defined problem in the abstract [Jai10, LWG12] there has been a growing interest in underlying principles and general desirable properties (sometimes called *axioms*) of clustering algorithms [FN71]. This axiomatic approach has been proposed in different flavours.

1. **On cost functions:** In this approach the algorithm is supposed to search for a optimal partition with respect a *cost function* f. In [PHB00] given a set X of n objects, the authors consider the number k of clusters obtained to be fixed. A cost function can be seen as a function

   $$f_{n,k} : \mathcal{M}_{n,k} \times \mathcal{D} \to \mathbb{R},$$

   where $\mathcal{M}_{n,k}$ is the set of all partition of X with k clusters and $\mathcal{D}$ is the set of all dissimilarity measures, called also *distances*, on X, that is, all possible weight functions on the all-to-all connected graph G with node-set X. Two general principles one should ask to f are then

   Permutation invariance: any permutation of the nodes does not change the cost function value;

   Monotonicity: given a partition $\mathcal{P}$ increasing the value of $d(i,j)$ will decrease, resp. increase, the cost function value, if i and j are in the same cluster, resp. in different clusters of $\mathcal{P}$.

The authors also proposed other axioms concerning different aspects of cost-function, such as invariance with respect to linear transformation of d, *Scale/Shift-Invariance*, or sensibility with respect to perturbation of d, *Weak/Strong Robustness*.

2. **On clustering algorithms:** A more recent interest in the axiomatic approach was sparked by Kleinberg's impossibility theorem [Kle03]. A clustering algorithm is considered here as
$$f : \mathcal{D} \to \mathcal{M}_n,$$
where $\mathcal{M}_n$ consists of all partitions of X. In the spirit of Arrow's impossibility theorem in social science, Kleinberg gives three natural properties then proves they cannot be simultaneously satisfied [Kle03, Theorem 2.1]. Kleinberg's axioms are the following

   Scale-invariance: $f(\alpha d) = f(d)$ for $\alpha > 0$

   Richness: for every partition $\mathcal{P}$ there exists a d such that $f(d) = \mathcal{P}$

   Consistency: let $\mathcal{P}$ be a partition and d such that $f(d) = \mathcal{P}$. If we decrease, resp. increase, $d(i, j)$ with i, j in the same cluster, resp. different clusters, of $\mathcal{P}$, then f will return the same $\mathcal{P}$. Such deformation of d is called $\mathcal{P}$-*consistent transformation* of d.

3. **On quality functions:** Given a clustering solution one could ask how reliable it is. To deal with such question the notion of *quality measures* has been introduced. A quality measure is a function of the form:
$$m_X : \mathcal{M}_n \times \mathcal{D} \to \mathbb{R}.$$
It is clear that one could use a cost function as quality measure; however, such measure can be unfit as it could not compare clustering from different algorithms or instead give biased results, see [BDA09]. A set of axioms is introduced in [BDA09] with the intent to propose an axiomatic approach for quality measures and avoid in such a different framework the impossibility result of Kleinberg. The axioms proposed are as follows.

   Scale-Invariance: Given any clustering $\mathcal{P}$ obtained with d, we have
$$m_X(\mathcal{P}, d) = m_X(\mathcal{P}, \lambda d), \text{ for every } \lambda > 0;$$

   Consistency: for every clustering $\mathcal{P}$ of (X, d) and $\mathcal{P}$-consistent transformation d' of d, we have that
$$m_X(\mathcal{P}, d) \geq m_X(\mathcal{P}, d');$$

> **Richness:** for each non-trivial clustering $\mathcal{P}$ of X there exists a distance d such that $\mathcal{P} = Argmax_C\{m_X(C, d)\}$;
>
> **Isomorphism Invariance:** given a clustering $\mathcal{P}$ over (X, d) and $\mathcal{P}'$ a clustering obtained with a distance-preserving permutation of X we have
>
> $$m_X(\mathcal{P}, d) = m_X(\mathcal{P}', d).$$
>
> Then the authors also prove that in this setting it is possible to find a clustering algorithm that satisfies all those axioms.

Several authors have since criticised Kleinberg's approach, particularly the Consistency axiom [BDA09, ABDL10, CM13], and proposed alternative frameworks that circumvent the impossibility result. For instance, by restricting clustering functions to k-partitions, for a fixed k, the axioms can coexist [ZB12]; if we allow arbitrary parameters, Kleinberg's axioms are compatible when applied to a parametric family of a clustering algorithm, as discussed in [CM13]; and, by replacing partitions by dendrograms as the output of a clustering function, the authors in [CM10] show a possibility and uniqueness result satisfied by single-linkage hierarchical clustering. In all these cases, Kleinberg's impossibility is avoided by either restricting or extending the definition of clustering function, or shifting the axiomatic focus to clustering quality measures [BDA09, LM14, YX14], or cost functions [Kar99, PHB00].

We remain close to Kleinberg's original setting and directly address the problematic behaviour of the Consistency axiom instead, which we replace by a weaker condition that we call *Monotonic Consistency*, where the rate of expansion, respectively contraction, of inter-, respectively intra-, cluster distances is not arbitrary, but globally controlled by an expansive function η (Section 5.1). In essence, η controls the inter-cluster expansion, while its inverse η^{-1} controls the intra-cluster contraction. As η is a function on distances, not pairs of points, the control is global, with points at similar distances experiencing the same expansion or contraction. Without this global condition, we recover Outer or Inner Consistency, each incompatible with Scale Invariance and Richness [ABDL10]. Monotonic Consistency avoids the aforementioned problematic behaviour and, moreover, we show a possibility result: Monotonic Consistency, Scale Invariance and Richness are mutually compatible clustering axioms (Theorem 5.16). As far as we know, this is the only alternative in the literature to the Consistency axiom that is compatible with Richness and Scale Invariance without modifying the definition of clustering function.

We present three instances of `Morse` corresponding to three choices of node and link preorders, then show that each of them satisfies a pair of Kleinberg's original axioms, and that all of them satisfy Monotonic Consistency Section 5.3. In particular, one of them satisfies Monotonic Consistency, Scale Invariance and Richness, which are therefore mutually compatible clustering axioms (Theorem 5.16). Note for the scope of this chapter these instances do not consider any external or intrinsic measure of centrality, instead the annotation is just a (fixed) random indexing of the nodes. For this reason we call these instances *agnostic* as they effectively will have no additional information (intrinsic or external) from the annotations.

We present also a generalisation of Kleinberg's axiomatic approach to graph clustering (Section 5.4). A distance function d on a set X can be represented by a complete graph G with node set X and links weighted by $d(u, v) > 0$. In fact, many clustering algorithms (including `Morse`) work on this graph representation. A classical example is Single Linkage, which, in fact, only depends on a minimum spanning tree of G [GR69]. A natural generalisation of Kleinberg's setting is, therefore, the case when G is an arbitrary, rather than complete, graph. That is, we fix a graph G and consider distances supported on the link set (this is the natural setting of graph clustering [Sch07]). In Section 5.4 we consider Kleinberg's axioms in this graph clustering setting, show that the impossibility result still holds, even when Richness is relaxed to Connected-Richness (partitions where every cluster is a connected subgraph), and give a possibility result for Monotonic Consistency and the same instance of Morse clustering. As the sparse case (G arbitrary) contains the complete case (G complete), Kleinberg's impossibility theorem [Kle03] is now a particular case of our graph clustering impossibility result (Theorem 5.21).

This work is currently submitted to the Journal of Machine Learning Research.

5.1 A critique of Kleinberg's axioms

Given a set X of n objects that we want to compare, a *dissimilarity* on X is a pairwise function

$$d : X \times X \to \mathbb{R}$$

such that $d(i, j) = d(j, i) \geq 0$, and $d(i, j) = 0$ if and only if $i = j$, for all $i, j \in X$. We will adhere to the convention in the literature and refer to d from now on as a *distance*, although it may not satisfy the triangle inequality. Following [Kle03], we define a *clustering algorithm* on X as a map

$$F : \{d \text{ distance on } X\} \to \{\mathcal{P} \text{ partition of } X\}.$$

We consider here a partition $\mathcal{P} = \{X_i\}_{i=1}^k$ of X as a disjoint union $X = X_1 \cup \ldots \cup X_k$, where a cluster is any X_i of $\mathcal{P}$. Given a partition $\mathcal{P}$ of X and $x, y \in X$, we use the notation $x \sim_{\mathcal{P}} y$ if x and y belong to the same cluster of $\mathcal{P}$, and $x \not\sim_{\mathcal{P}} y$ if not.

Kleinberg [Kle03] introduced three natural properties for a clustering algorithm, then proved that they cannot be simultaneously satisfied by any clustering algorithm F. As stated earlier in the chapter introduction (Item **On clustering algorithms**), Kleinberg also showed that each pair of these properties can be simultaneously satisfied, in fact by three different versions of Single Linkage.

Our first contribution is a weakening of the Consistency property which is both very natural, and can coexist with Richness and Scale-Invariance.

To motivate our definition, we first discuss the problematic behaviour of Kleinberg's Consistency in the presence of Richness and Scale Invariance (see also [CM13, ABDL10, ZB12]). Let us recall the definition of *consistent algorithm.*

Definition 5.1 ([Kle03]). Given a set X and a partition of X, $\mathcal{P}$, let d and d' be two distances on X. Then d' is a *$\mathcal{P}$-transformation of d* if

$$\begin{cases} d'(v,u) \leq d(v,u) & \text{if } v \sim_{\mathcal{P}} u, \text{ and} \\ d'(v,u) \geq d(v,u) & \text{if } v \not\sim_{\mathcal{P}} u, \end{cases} \tag{5.1}$$

A clustering algorithm is *consistent* if $F(d) = F(d')$ whenever d' is a consistent $F(d)$-transformation of d.

Given F a consistent and scale-invariant clustering algorithm, and two different partitions $F(d_1) \neq F(d_2)$, it can be shown [Kle03, Theorem 3.1] that each partition is not the refinement of the other (a partition $\mathcal{P}$ is a *refinement* of $\mathcal{Q}$ if each cluster of $\mathcal{P}$ is contained in a cluster of $\mathcal{Q}$). In particular, given a distance d and associated partition $\mathcal{P} = F(d)$, we can never obtain a partition identical to $\mathcal{P}$ but with one, or more, of its clusters further subdivided (Figure 5.1). On the other hand, consider any distance d' satisfying

$$\begin{cases} d'(v,u) < d(v,u) & \text{if } v,u \in C_1, \\ d'(v,u) < d(v,u) & \text{if } v,u \in C_2, \\ d'(v,u) = d(v,u) & \text{otherwise,} \end{cases}$$

where C is a cluster of $\mathcal{P}$ and $\{C_1, C_2\}$ is an arbitrary partition of C. Note that any such d' is a $\mathcal{P}$-transformation of d. This means that we can arbitrarily emphasise the subcluster structure, to the point that it could be more natural to consider C_1 and C_2 as separate clusters (Figure 5.1), while Consistency implies $F(d) = F(d')$ regardless.

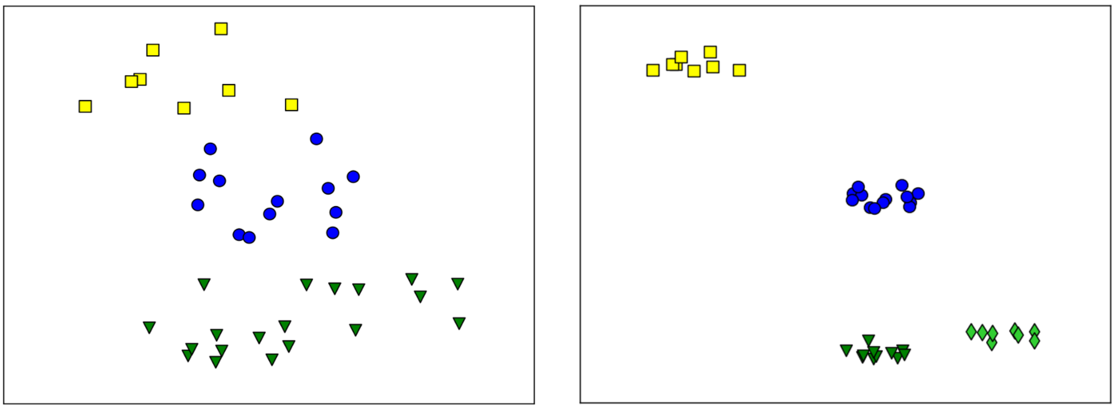


FIGURE 5.1: Problematic behaviour of the Consistency axiom. We can arbitrarily emphasise any subcluster structure without affecting the output of the clustering algorithm. This behaviour is explicitly avoided by Monotonic Consistency (Figure 5.4).

We propose a more restrictive definition of Consistency which avoids this type of behaviour. The idea is to globally fix the rate at which we can increase (decrease) the intra-cluster (inter-cluster) distances. We do this restricting to $\mathcal{P}$-transformations obtained through a particular class of functions, which we describe next.

Definition 5.2. Let X and Y be subsets of $\mathbb{R}$. We call a continuous map $\eta\colon X \to Y$ *expansive* if

$$|\eta(x) - \eta(y)| \geq |x - y| \quad \text{for all } x, y \in X. \tag{5.2}$$

By reversing the inequality, we define a *contractive* map.

Expansive maps can be defined more generally for maps between metric spaces [GH55] as maps that do not decrease distances between pairs of points, and we have added the continuity hypothesis for convenience (see Remark 5.4). We will use expansive maps to expand and contract distances with respect to a partition, namely, $d'(u,v) = \eta(d(u,v))$ if u and v belong to different clusters, and $d'(u,v) = \eta^{-1}(d(u,v))$ if they belong to the same cluster. In particular, we take $X = Y = [0,\infty)$ in the definition above, and assume $\eta(0) = 0$. The following lemma summarises some useful properties.

Lemma 5.3. *Let $\eta\colon [0,\infty) \to [0,\infty)$ be a continuous expansive map with $\eta(0) = 0$. Then:*

(i) η is strictly increasing, a bijection, and satisfies $\eta(x) \geq x$ for all x;

(ii) η^{-1} is strictly increasing, a contractive map, and satisfies $\eta^{-1}(x) \leq x$ for all x.

Proof. (i) By contradiction, if η is not strictly increasing, we can find $x > y$ with $\eta(x) \leq \eta(y)$, so that $\eta(0) = 0 \leq \eta(x) \leq \eta(y)$ and, by the Intermediate Value Theorem, we can find $z \in [0, y]$ such that $\eta(z) = \eta(x)$, a contradiction. The growth condition is immediate from the expansion property Equation 5.2 for $y = 0$,

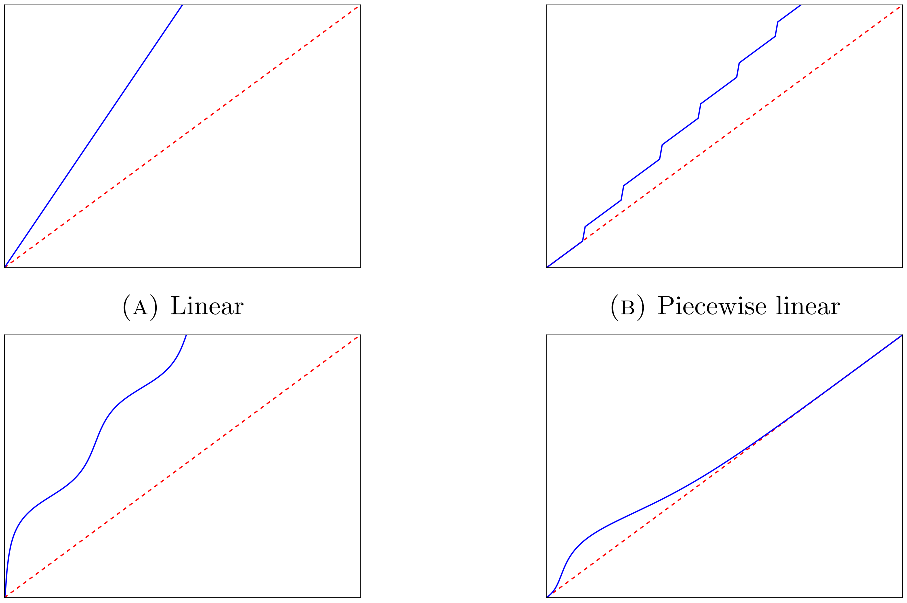
$$|\eta(x)| = \eta(x) \geq |x| = x,$$

for all $x \in [0, \infty)$. Since η is strictly increasing, it is injective. It is also surjective: The growth condition above gives $\eta(x) \to \infty$ as $x \to \infty$ and, together with $\eta(0) = 0$ and continuity, we have that η takes all values in $[0, \infty)$.

(ii) Since η is bijective, it has an inverse η^{-1}. The inverse of a (strictly) increasing function is also (strictly) increasing. To show this, and the two remaining properties, one can simply use the corresponding properties of η in (i) on $x' = \eta(x)$ and $y' = \eta(y)$. □

Example 5.1. *The following are examples of continuous expansive functions η on $[0, \infty)$ with $\eta(0) = 0$.*

1. *(Linear) $\eta(x) = \alpha x$ for $\alpha \geq 1$ (Figure 5.2(a)).*
2. *(Piecewise linear) $\eta = \eta(d_i) + \alpha_i(x - d_i)$ for $x \in [d_i, d_{i+1}]$, where $0 = d_1 < d_2 < \ldots < d_n$, $\eta(0) = 0$, and $\alpha_i \geq 1$, for all i (Figure 5.2(b)) .*
3. *(Differentiable) A differentiable function $\eta\colon [0, \infty) \to [0, \infty)$ with $\eta(0) = 0$ is expansive if and only if $\eta'(x) \geq 1$ for all x (Figure 5.2(c)).*
4. *(Graphical criterion) A continuous function $\eta\colon [0, \infty) \to [0, \infty)$ is expansive if and only if the function $\eta(x) - x$ is increasing (this follows from Remark 5.4).*



(A) Linear

(B) Piecewise linear

(C) Differentiable

(D) Counterexample

Figure 5.2: Examples of expansive functions and one counterexample (solid blue lines). At each point, the function growths at least as fast as the line $y = x$ (dashed red line).

Remark 5.4. If η is increasing, Equation 5.2 is equivalent to

$$\eta(x) - \eta(y) \geq x - y \quad \text{for all } x \geq y. \tag{5.3}$$

In fact, this equation alone implies η increasing and thus Equation 5.2. We could drop the continuity hypothesis in Definition 5.2, and define an expansive function simply by Equation 5.3. In practice, however, a monotonic transformation (Definition 5.5) can always be realised by a continuous, piecewise linear function η (Lemma 5.7).

5.2 Monotonic transformations

In Kleinberg's original Consistency axiom, arbitrary transformations that increase inter-cluster distances and decrease intra-cluster distances are allowed. To avoid an impossibility result, we restrict to transformations obtained via an expansive function η, as follows. Recall that we write $x \sim_{\mathcal{P}} y$ if x and y are in the same cluster with respect to a partition $\mathcal{P}$, and $x \not\sim_{\mathcal{P}} y$ if not.

Definition 5.5. Let d be a distance on a set X, and $\mathcal{P}$ a partition of X. A *$\mathcal{P}$-monotonic transformation of d* is any distance d' on X such that

$$\begin{cases} d(x,y) = \eta(d'(x,y)) & \text{if } x \sim_{\mathcal{P}} y, \text{ and} \\ d(x,y) = \eta^{-1}(d'(x,y)) & \text{if } x \not\sim_{\mathcal{P}} y, \end{cases} \tag{5.4}$$

for some expansive map $\eta\colon [0,\infty) \to [0,\infty)$, and all $x, y \in X$. (Note that such η necessarily satisfies $\eta(0) = 0$.)

Definition 5.6. A clustering algorithm F is *Monotonic Consistent* if $F(d') = F(d)$ whenever d' is a $F(d)$-monotonic transformation of d.

Note that, given d and $\mathcal{P}$, d' is uniquely determined by η. Since $\eta(x) \geq x$ and $\eta^{-1}(x) \leq x$ for all x (Lemma 5.3), the distance function d' increases inter-cluster distances and decreases intra-cluster distances (hence Consistency implies Monotonic Consistency). However, our allowed transformations do so globally (d' depends on distances between points, not the actual points) and monotonically (the rates at which we expand or contract distances are the inverse of one another). Finally, note that $\mathcal{P}$-monotonic transformations can be composed and this corresponds to the composition $\eta_2 \circ \eta_1$ of expansive maps.

Example 5.2. *The following are examples of $\mathcal{P}$-monotonic transformations.*

1. *(Linear) Let $\eta(x) = \alpha x$, $\alpha \geq 1$. The corresponding $\mathcal{P}$-monotonic transformation multiplies inter-cluster distances by α, and intra-cluster distances by $1/\alpha$. This*

is similar to Inner and Outer Consistency, introduced in [ABDL10], except that the expansion and contraction rates are not arbitrary, but the reciprocal of one another.

2. *(Linear step function) This is the function*

$$\eta(x) = \begin{cases} x & 0 \le x \le d_1, \\ \alpha(x - d_1) + d_1 & d_1 \le x \le d_2, \\ (x - d_2) + \alpha d_2 & d_2 \le x, \end{cases} \tag{5.5}$$

for some $0 \le d_1 < d_2$ and $\alpha > 1$. The associated $\mathcal{P}$-monotonic transformation preserves (inter- or intra-cluster) distances below d_1, scales distances between d_1 and d_2 as in Figure 5.2b, and (necessarily) translates distances above d_2, adding $\eta(d_2) = \alpha d_2$ to inter-cluster distances, and subtracting $\eta(d_2)$ to intra-cluster distances. Note that d_2 can be equal to $+\infty$ and so the third line in Equation 5.5 becomes obsolete.

3. *(Piecewise linear) This is generalises both (1) and (2): For the piecewise linear η as in Figure 5.2c, we have a rate of expansion/contraction α_i, and a translation by $\eta(d_i)$, for distances in the interval $[d_i, d_{i+1}]$ where η is linear.It can be shown that each piecewise linear function is a composition of linear step functions. Note that as we will be considering a finite number of distance values (cf. Lemma 5.7) we can assume that the set $\{\alpha_i\}_{i \ge 0}$ is finite with cardinality N. If η is a piecewise expansive function then we have that it can be described as follows*

$$\eta(x) = \begin{cases} \alpha_i(x - d_i) + \eta(d_i), & \text{for } d_i \le x \le d_{i+1}, i \le N \\ \alpha_{N+1}(x - d_{N+1}) + \eta(d_{N+1}), & \text{for } d_{N+1} \le x, \end{cases}$$

where $\alpha_i \ge 1$ for $0 \le i \le N$.

Consider the piecewise function η_1 defined as

$$\eta_1(x) = \begin{cases} \alpha_i(x - d_i) + \eta(d_i), & \text{for } d_i \le x \le d_{i+1}, i \le N - 1 \\ (x - d_N) + \eta(d_N), & \text{for } d_N \le x. \end{cases}$$

and the step function s_N

$$s_N(x) = \begin{cases} x, & \text{for } x \le d_N \\ \alpha_{N+1}(x - d_{N+1}) + d_{N+1}, & \text{for } d_N \le x. \end{cases}$$

Clearly both of them are expansive and in particular $\eta(x) = (\eta_1 \circ s_N)(x)$. By induction we can express $\eta(x)$ as composition of $\{s_i\}_{i=0}^N$ where each s_i is an expansive linear step function.

Below, we show that every $\mathcal{P}$-monotonic transformation is induced by a piecewise linear η, or, equivalently, by a finite composition of linear step functions.

Although d' is uniquely determined by η, this η is not unique, that is, different choices of η may result in the same $\mathcal{P}$-monotonic transformation d'. Indeed, any expansive η interpolating the points $(d'(x,y), d(x,y))$ with $x \sim_{\mathcal{P}} y$ and $(d(x,y), d'(x,y))$ with $x \not\sim_{\mathcal{P}} y$ necessarily gives the same $\mathcal{P}$-monotonic transformation d', by Equation 5.4. In particular, we can always assume η to be piecewise linear in Definition 5.5, and, in fact, we can determine whether such function exists directly from d', as the next result shows.

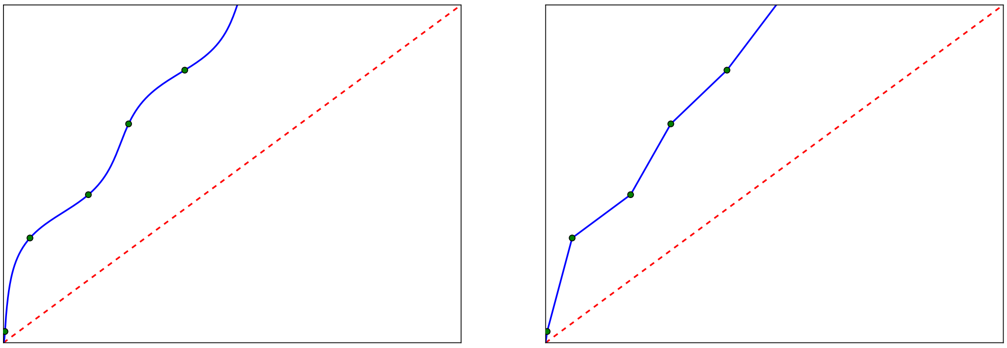


FIGURE 5.3: Expansive map (left) and linear interpolation (right) through the points in a subset S (as in Lemma 5.7). Both maps determine the same $\mathcal{P}$-monotonic transformation d' of a distance d. In the linear interpolation (right), the slope of each successive segment must be at least 1.

Lemma 5.7. *Let d and d' be distances on a finite set X and $\mathcal{P}$ a partition of X. Then d' is a $\mathcal{P}$-monotonic transformation of d if and only if a linear interpolation of the points*

$$S = \left\{ \left(d(x,y), d'(x,y)\right) \mid x \sim_{\mathcal{P}} y \right\} \; \cup \; \left\{ \left(d'(x,y), d(x,y)\right) \mid x \not\sim_{\mathcal{P}} y \right\} \subseteq \mathbb{R}^2$$

is a well-defined expansive map $\eta\colon [0,\infty) \to [0,\infty)$, that is, S, regarded as subset of $\mathbb{R}^2$, is the graph *of a expansive map η.*

Proof. Clearly, if there exists a linear interpolation η of the points in S such that it is a well-defined expansive map, then d' is a $\mathcal{P}$-monotonic transformation of d, by definition. In particular S, regarded as subset of $\mathbb{R}^2$, will be the graph of such η.

Now assume d' is a $\mathcal{P}$-monotonic transformation of d. Then we can write

$$S = \left\{ \left(d(x,y), \eta\left(d(x,y)\right)\right) \mid x \sim_{\mathcal{P}} y \right\} \; \cup \; \left\{ \left(d'(x,y), \eta\left(d'(x,y)\right)\right) \mid x \not\sim_{\mathcal{P}} y \right\},$$

where $\eta\colon [0,\infty) \to [0,\infty)$ is an expansive map. To define a linear interpolation of S we will assume that S is ordered lexicographically

$$S = \{(x_0, y_0), (x_1, y_1), \ldots, (x_N, y_N)\},$$

where $y_i = \eta(x_i)$ for $0 \leq i \leq N$ and $x_i < x_{i+1}$. We can assume the latter since η is injective: if $x_i = x_{i+1}$ then $y_i = y_{i+1}$. Consider now the linear interpolation of S consisting of segments between consecutive pairs of points (x_i, y_i) and (x_{i+1}, y_{i+1}). As every point in S is of the form $(x, \eta(x))$, we have that the slope of each segment is

$$\frac{\eta(x_{i+1}) - x_{i+1}}{\eta(x_i) - x_i} \geq 1,$$

as η is expansive, Equation 5.3. From this we have that the linear interpolation above, effectively a discretisation of η, is in fact a well-defined expansive map, of which again S is the graph. □

We now show how the problematic behaviour of Kleinberg's Consistency axiom (Figure 5.1) is avoided by Monotonic Consistency. Suppose that we have a set X and a partition $\mathcal{P} = F(d)$ with respect to a clustering algorithm F and a distance d on X. Choose a cluster C and a partition $\{C_1, C_2\}$ of C that we wish to emphasise with a distance d' which (necessarily) decreases the intra-cluster distances, but in a way that distances within each C_1 and C_2 decrease much faster than distances between C_1 and C_2, in order to achieve the behaviour depicted in Figure 5.1.

Let $u, v \in C_1$ distinct and $w \in C_2$, and call $x = d(u, v)$, $x' = d'(u, v)$, $y = d(u, w)$ and $y' = d'(u, w)$. We impose $x' \leq x$ and $y' \leq y$, and, in addition, we want to make $x - x'$ large while keeping $y - y'$ small (Figure 5.4). This is not possible if d' if a $\mathcal{P}$-monotonic transformation of d, as follows. Let η be an expansive map realising d'. Then $x = \eta(x')$ and $y = \eta(y')$. Assume first $x \leq y$. Then Equation 5.3 gives

$$\eta(y') - y' \geq \eta(x') - x' \iff y - y' \geq x - x'. \tag{5.6}$$

This implies that if we want to reduce the distances inside of a subcluster ($x - x'$ large), we need to reduce the distances between the clusters ($y - y'$) by at least the same amount. The remaining case, $x \geq y$, follows from η^{-1} being a decreasing function (Lemma 5.3),

$$x \geq y \implies x' = \eta^{-1}(x) \geq \eta^{-1}(y) = y', \tag{5.7}$$

so that we cannot decrease the intra-cluster distance x without also decreasing the inter-cluster distance y.

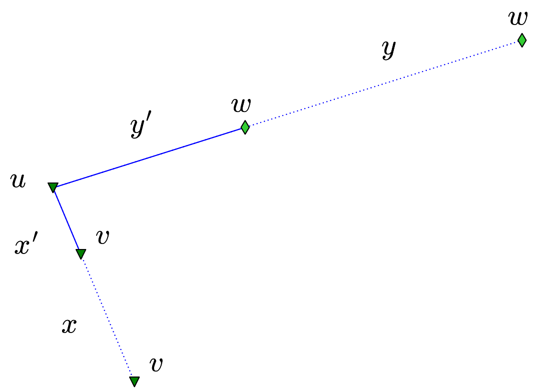


FIGURE 5.4: Avoidance of the problematic behaviour by Monotonic Consistency. A $\mathcal{P}$-monotonic transformation of d reduces the distance from u to v by $x - x'$, and the distance from u to w by $y - y'$. Then either $x' \geq y'$ (Equation 5.7), or $y - y' \geq x - x'$ (Equation 5.6). In either case, we cannot separate u and v from w within the same cluster.

We finish Section 5.1 by exploring Monotonic Consistency for Single Linkage, and for metrics.

We will show that Monotonic-Consistency, a weakening of Consistency, can be satisfied together with Richness and Scale-Invariance by a particular instance of `Morse` (Theorem 5.16). This is in contrast with Single Linkage, which, with different stopping conditions, satisfies each pair of Kleinberg's axioms [Kle03]. The instance of Single-Linkage, present in [Kle03], that satisfies Richness and Scale Invariance is *Scale-α Single Linkage* with $0 < \alpha < 1$, see [Kle03, p. 4]. This instance however does not satisfy Monotonic Consistency, as we show next.

Lemma 5.8. *Let $\alpha \in (0, 1)$. Then Scale-α Single-Linkage does not satisfy Monotonic Consistency.*

Proof. Let X be any set with at least three points, $\mathcal{P}$ any partition of X with at least two clusters, and $x, y \in X$ such that $x \not\sim_{\mathcal{P}} y$. Define d on X as follows

$$d(u,v) = \begin{cases} \dfrac{\alpha}{2}, & \text{if } u \sim_{\mathcal{P}} v, \\ 1, & \text{if } u = x, v = y, \\ \alpha, & \text{otherwise.} \end{cases}$$

Let $d_{\max} = \max_{s,t \in X} d(s,t) = 1$. If we represent (X, d) by a complete graph with node set X and links (i, j), $i \neq j$, weighted by $d(i, j) > 0$, recall that Scale-α Single-Linkage returns the connected component of the graph obtained after removing all links (i, j) with value $d(i, j) \geq \alpha \, d_{\max} = \alpha$, in this case. Consequently, Scale-α Single-Linkage applied to d returns the original partition $\mathcal{P}$.

Let d' be the $\mathcal{P}$-monotonic transformation of d given by

$$\eta(x) = \frac{x^2 + x}{\alpha}.$$

(Note that $\eta(0) = 0$ and $\eta'(x) = \frac{2x+1}{\alpha} > 1$ for all x, so η is indeed expansive.) Then

$$d'(u,v) = \begin{cases} \eta^{-1}\left(\frac{\alpha}{2}\right) = \frac{-1+\sqrt{1+2\alpha^2}}{2}, & \text{if } u \sim_{\mathcal{P}} v, \\ \eta(1) = \frac{2}{\alpha}, & \text{if } u = x, v = y, \\ \eta(\alpha) = 1 + \alpha, & \text{otherwise.} \end{cases}$$

We now have $d'_{\max} = \eta(1) = \frac{2}{\alpha}$ and thus scale$-\alpha$ Single-Linkage removes the links (i,j) with $d(i,j) \geq \alpha d'_{\max} = 2$. Since $\alpha < 1$, the only removed link is $d(x,y)$ and, since X has at least three points, the algorithm returns the trivial partition $\{X\}$, clearly not $\mathcal{P}$. □

A *metric* is a distance (in the sense of this chapter) which also satisfies the triangle inequality, $d(u,w) \leq d(u,v) + d(v,w)$ for all u, v, w. Metrics arise naturally when X is embedded in a metric space such as $\mathbb{R}^m$, and, in fact, for many clustering algorithms (for example k-means clustering), the distance function is always a metric. It is therefore natural to ask whether Monotonic Consistency is a useful property in this context, namely, whether a non-trivial (that is, η not the identity) $\mathcal{P}$-monotonic transformation of a metric can be a metric. If not, Monotonic Consistency would become an empty axiom for metrics. Of course, not every $\mathcal{P}$-monotonic transformation of a metric will be a metric, but we show below that, given a metric d and an arbitrary partition $\mathcal{P}$, we can always find $\mathcal{P}$-monotonic transformations of d which are metrics.

Given a distance d on a set X, we call a triple of points $i, j, k \in X$ *aligned* if they are distinct and $d(i,k) = d(i,j) + d(j,k)$.

Theorem 5.9. *Let X be a set, $\mathcal{P}$ a partition of X, and d a distance on X such that no triple of nodes is aligned. Then there exists a constant $c(d,\mathcal{P}) > 1$ such that, for all $s \in [1, c(d,\mathcal{P}))$, the $\mathcal{P}$-monotonic transformation of d given by $\eta(x) = sx$ is a metric. Moreover, there is an universal constant $c(d)$ independent of the partition, that is, $1 < c(d) \leq c(d,\mathcal{P})$ for all partitions $\mathcal{P}$ of X.*

Proof. Let d' be the $\mathcal{P}$-monotonic transformation of d given by $\eta(x) = sx$ for some $s \geq 1$. We will find conditions on s to guarantee that d' satisfies the triangle inequality. Let $i, j, k \in X$ distinct (if not, the triangle inequality is automatically satisfied). We want to show that

$$d(i,k) \leq d(i,j) + d(j,k) \implies d'(i,k) \leq d'(i,j) + d'(j,k).$$

Recall that

$$d'(i,j) = \begin{cases} s\,d(i,j) & \text{if } i \sim_{\mathcal{P}} j, \\ \dfrac{d(i,j)}{s} & \text{otherwise.} \end{cases}$$

If i, j and k are in the same cluster then clearly

$$\frac{d(i,k)}{s} \leq \frac{d(i,j)}{s} + \frac{d(j,k)}{s}.$$

If they are all in pairwise different clusters, then

$$s\ d(i,k) \leq s\ d(i,j) + s\ d(j,k).$$

If i and k are in the same cluster but j is not, then (recall $s \geq 1$)

$$\frac{d(i,k)}{s} \leq d(i,k) \leq s\,d(i,j) + s\,d(j,k).$$

Since i and k are interchangeable in the triangle inequality above, the only remaining case is when i and j are in the same cluster, but k is not. In this case, we want to show that

$$s\,d(i,k) \leq \frac{d(i,j)}{s} + s\,d(j,k). \tag{5.8}$$

If $d(i,k) \leq d(j,k)$ then $s\,d(i,k) \leq s\,d(j,k)$ and Equation 5.8 is automatically satisfied. If $d(i,k) > d(j,k)$, Equation 5.8 is satisfied if and only if

$$s^2\left(d(i,k) - d(j,k)\right) \leq d(i,j) \iff s \leq \sqrt{\frac{d(i,j)}{d(i,k) - d(j,k)}}\,.$$

Define

$$c(d,\mathcal{P}) = \min_{\substack{i \sim_{\mathcal{P}} j, i \not\sim_{\mathcal{P}} k \\ d(i,k) > d(j,k)}} \sqrt{\frac{d(i,j)}{d(i,k) - d(j,k)}} \quad \text{and}$$

$$c(d) = \min_{d(i,k) > d(j,k)} \sqrt{\frac{d(i,j)}{d(i,k) - d(j,k)}}\,.$$

Clearly, $c(d) \leq c(d,\mathcal{P})$ for all partitions $\mathcal{P}$. To finish the proof, note that the triangle inequality for d guarantees $c(d) \geq 1$, and $c(d) = 1$ if and only if there is an aligned triple of points. □

Defining the minimum of an empty set as infinity, we might have $c(d,\mathcal{P}) = \infty$ (or $c(d) = \infty$), meaning that the $\mathcal{P}$-monotonic transformation of d given by $\eta(x) = sx$ is a metric for any $s \geq 1$, and Theorem 5.9 still holds. Of course, this would only occur if for all i, j, k with $i \sim_{\mathcal{P}} j$ and $i \not\sim_{\mathcal{P}} k$, we have $d(i,k) = d(j,k)$.

5.3 Agnostic Morse clustering

In this section we describe the clustering algorithm `Morse` in the form of three variants: `SiR Morse`, k-`Morse` and δ-`Morse`. Each of them satisfy one pair of the original Kleinberg axioms, and all of them satisfy Monotonic Consistency. In particular, one of them (`SiR Morse`) satisfies Scale Invariance and Richness, showing that our three axioms can be simultaneously satisfied (Theorem 5.16).

Let X be a finite set and d a distance on X. If $G = (V, E)$ is the complete graph with $V \equiv X$ then d induces a link weight function $w : E \to \mathbb{R}$, that is for any $e = (v, u) \in E$ we define $w(e) = d(v, u)$. Similarly, a labelling $X = \{x_1, x_2, \ldots, x_n\}$ induces a node annotation function $f : X \to \mathbb{R}$, where $f(x_i) = i$. The function w (resp. f) induces naturally a total preorder on E (resp. V) as follows.

Example 5.3. *Let $\mathcal{G} = (V, E)$ be a complete graph.*

(a) For any weight function $w : E \to \mathbb{R}$, the relation $e \preceq_E f$ if $w(e) \leq w(f)$ is a total preorder on E.

(b) For any annotation function $f : V \to \mathbb{R}$, the relation $u \preceq_V v$ if $f(u) \leq f(v)$ is a total preorder on V.

Note that any total preorder on a set S is induced by a function $I : S \to \mathbb{N}$.

In order to show the properties of such variants, an useful result to determine when two Morse partitions, that is partitions obtained from a (discrete) Morse flow, are equal is as follows.

Lemma 5.10. *Let Φ and Φ' be Morse flows on X with associated Morse partitions $\mathcal{P}$ and $\mathcal{P}'$. If $x \sim_{\mathcal{P}} \Phi'(x)$ for all $x \in X$, then $\mathcal{P}'$ is a refinement of $\mathcal{P}$.*

Proof. Write $\mathcal{P} = \{X_1, \ldots, X_n\}$ and $\mathcal{P}' = \{X'_1, \ldots, X'_{n'}\}$. Write x_i, respectively x'_j, for the critical node in X_i, respectively X'_j, for all i, j. Choose $N \geq 1$ such that both Φ and Φ' stabilise, that is, $\Phi^N = \Phi^{N+1}$ and $(\Phi')^N = (\Phi')^{N+1}$. We need to show that, for each index j there exists an index i such that $X'_j \subseteq X_i$.

Let $x \in X'_j$ and consider the flow paths

$$p(x) = \{x, \Phi(x), \ldots, \Phi^N(x) = x_i\} \text{ and}$$
$$p'(x) = \{x, \Phi'(x), \ldots, (\Phi')^N(x) = x'_j\}$$

By definition of Morse partition, all nodes in $p(x)$ are in the same cluster X_i of $\mathcal{P}$ and all nodes in $p'(x)$ in the same cluster X_j of $\mathcal{P}'$. By hypothesis, $(\Phi')^n(x) \sim_{\mathcal{P}} (\Phi')^{n+1}(x)$ for all $n \geq 0$, so $p'(x) \subseteq X_i$. In particular $x'_j \sim_{\mathcal{P}} x_i$.

Given any other $y \in X'_j$,

$$p(y) = \{y, \Phi(y), \dots, \Phi^N(y) = x_k\} \subseteq X_k \;\text{ and}$$
$$p'(y) = \{y, \Phi'(y), \dots, (\Phi')^N(y) = x'_j\} \subseteq X'_j,$$

for a possibly different cluster X_k. Again, by hypothesis, we have $p'(y) \subseteq X_k$ and, in particular, $x'_j \sim_{\mathcal{P}} x_k$. Then $x'_j \in X_i \cap X_k \neq \emptyset$ and hence $i = k$, as distinct clusters are disjoint. Since y was arbitrary, we conclude that $X'_j \subseteq X_i$. □

Note that two partitions are equal if and only if each is the refinement of the other, or if they have the same size (number of clusters) and one is the refinement of the other.

Different choices of link and node preorders result in different instances of `Morse`. We now show three versions of Morse clustering, each satisfying a different pair of Kleinberg's axioms, and such that all of them satisfy Monotonic Consistency.

Let (X, d) be a set with a distance function, and consider the complete graph with node set X. Let us fix, once and for all, a labelling $X = \{x_1, x_2, \dots, x_n\}$, which we will use to create the node preorders (see the remarks at the end of Section 5.3 on labelling). We also assume that X has at least three points.

Our first instance of Morse clustering is called `SiR-Morse` (`Scale-invariant and Rich`), and corresponds to the choices of node and link preorders given below.

`SiR-Morse`

- $v_i \preceq_V v_j$ if $i \leq j$
- $(v, s) \preceq_E (v, t)$ if $d(v, s) \geq d(v, t)$

Note that the node preorder is a total order, and the link preorder is also locally total (at each node). The corresponding Morse flow chooses, at each node v, the link with smallest distance, if it is unique and admissible, that is, it connects v with a higher (with respect $\preceq_V$) node. On the other hand, if more than one link at v achieves the smallest distance, or if such link is not admissible, then v is critical, that is, the Morse flow fixes v, $\Phi(v) = v$.

Theorem 5.11. *`SiR Morse` is Scale-Invariant and Rich.*

Proof. (**Scale-invariance**) Scale-Invariance does not affect the node preorder, since $\preceq_V$ is independent of d. For the link preorder $\preceq_E$ we have that $d(v, s) \leq d(v, t)$ if and only if $\alpha\, d(v, s) \leq \alpha\, d(v, t)$ for all $\alpha > 0$ and so also the preorder relationships do not change under Scale-Invariance. Hence the output of `SiR Morse` for (X, d) and for $(X, \alpha d)$ are the same.

(**Richness**) Consider $V = V_1 \cup \ldots \cup V_k$ an arbitrary partition of V. Let v_i be the maximal node in V_i ($\preceq_V$ is a total order) and define a distance d as follows

$$d(v,s) = \begin{cases} 1, & \text{if } v, s \in V_i \text{ for some } i \text{, and either } v = v_i \text{ or } s = v_i, \\ 2, & \text{otherwise,} \end{cases}$$

for all $v \neq s$. If $v \in V_i$, the link to v_i is always admissible and the largest with respect to $\preceq_E$, so $\Phi(v) = v_i$ for the Morse flow, and we recover the partition $V_1 \cup \ldots \cup V_k$. □

Let $k \geq 1$ be an integer. Next we present a Morse algorithm that guarantees a partition with k clusters (Theorem 5.12), and thus it cannot be rich. However it satisfies Consistency and Scale Invariance (Theorem 5.13).

k-Morse

- $v_i \preceq_V v_j$ if $i = j$ or $i + k < j$
- $(v,s) \preceq_E (v,t)$ if

  $s \preceq_V v \preceq_V t$, or

  $d(v,s) > d(v,t)$ and $v \preceq_V t$, or

  $d(v,s) = d(v,t)$ and $s \preceq_V t$.

For this choice of node preorder, there are exactly k critical nodes, which are v_n, v_{n-1}, $\ldots$, v_{n-k+1}, and hence k clusters (see Theorem 5.12 below). The link preorder is defined such that admissible links are always greater than non-admissible ones, and admissible ones are compared using distances, with the node preorder used as tie-breaking procedure. In particular, if there are admissible links at v, the maximal admissible link at v exists and it is unique.

Theorem 5.12. *k-Morse always produces a partition with k clusters.*

Proof. If $v_i \in X$ with $i > n - k$ then there are no nodes greater than v_i with respect to $\preceq_V$ hence no admissible links at v and thus $\Phi(v_i) = v_i$ critical. On the other hand, v_i with $i \leq n - k$ cannot be critical, as there are admissible links $(v_i, v_j) \in E_{v_i}$ for all $j > i + k$, so the maximum exists and it is unique. All in all, there are exactly k critical nodes $v_n, v_{n-1}, \ldots, v_{n-k+1}$ and therefore exactly k clusters. □

Theorem 5.13. *k-Morse is Scale-Invariant and Consistent.*

Proof. (**Scale-invariance**) A distance transformation $d' = \alpha \cdot d$ for $\alpha > 0$ does not affect the k-Morse node or link preorder, hence we obtain the same partition, see Theorem 5.11.

(**Consistency**) Let d be a distance in X, $\mathcal{P}$ the partition given by k-`Morse` on (X, d), and d' a $\mathcal{P}$-transformation of d, that is,

$$d(v,u) \geq d'(v,u), \ \text{ if } v \sim_{\mathcal{P}} u, \tag{5.9}$$

$$d(v,u) \leq d'(v,u), \ \text{ otherwise.} \tag{5.10}$$

Let Φ respectively Φ' be the Morse flow corresponding to d respectively d'. The critical nodes depend on the node preorder alone, hence, as in the proof of Theorem 5.12, we have $\Phi(v_i) = v_i = \Phi'(v_i)$ for all $i > n - k$ and thus $\mathcal{P}$ and $\mathcal{P}'$ have the same number of clusters. Therefore, it suffices to show that $x \sim_{\mathcal{P}} \Phi'(x)$ for all $x \in X$, by Lemma 5.10.

Let $x \in X$. If x is critical, $\Phi(x) = \Phi'(x)$ as they have the same critical nodes, so clearly $x \sim_{\mathcal{P}} \Phi(x) = \Phi'(x)$. If x is not critical, let $s = \Phi(x)$ and $t = \Phi'(x)$. The maximality and the definition of $\preceq_E$ implies

$$d(x,s) \leq d(x,t) \text{ and } d'(x,t) \leq d'(x,s).$$

Since $\Phi(x) = s$, they are in the same cluster, $x \sim_{\mathcal{P}} s$, and thus $d'(x,s) \leq d(x,s)$, by Equation 5.9 above. All in all,

$$d'(x,t) \leq d'(x,s) \leq d(x,s) \leq d(x,t). \tag{5.11}$$

Now, if $d'(x,t) < d(x,t)$, they are necessarily in the same cluster, $x \sim_{\mathcal{P}} t$, by Equation 5.9 and Equation 5.10 above. The remaining case $d'(x,t) = d(x,t)$ implies equalities in Equation 5.11, and, by the definition of the link preorders and the maximality of (x,s) with respect to d, we have $s = t$. In both cases, $x \sim_{\mathcal{P}} t = \Phi'(x)$. □

Let $\delta > 0$. The final instance of `Morse` clustering satisfies Consistency and Richness, and is given by the following choices of preorders.

δ-`Morse`

- $v_i \preceq_V v_j$ if $i \leq j$
- $(v,s) \preceq_E (v,t)$ if

  $s = t$, or

  $d(v,t) < \min\{d(v,s), \delta\}$ and $v \preceq_V t$, or

  $d(v,s) = d(v,t) < \delta$ and $v \preceq_V s \preceq_V t$.

With this preorder, only admissible links with distance less than the threshold parameter δ are considered for the flow. Among those links, we choose the one with minimal distance, using the node preorder to resolve ties. Note that, if there are admissible links at distance less than δ, the maximum admissible link exists and it is unique.

Theorem 5.14. *$\delta-$`Morse` satisfies Consistency and Richness.*

Proof. (**Richness**) Consider an arbitrary partition $X = X_1 \cup \ldots \cup X_k$ and define the distance function

$$d(v,s) = \begin{cases} \frac{\delta}{2}, & \text{if } v,s \text{ are in the same cluster, and} \\ \delta, & \text{otherwise,} \end{cases}$$

for $v \neq s$. Let x_i be the largest node in X_i with respect to $\preceq_V$ and $v \in X_i$ arbitrary. By the definition of d and the link preorder, we have that (v, x_i) is the maximum admissible link at v. Also, x_i is critical: the maximum link at x_i is of the form (x_i, s) for $s \in X_i$, hence not admissible or, if $|X_i| = 1$, any link in E_{x_i} is maximal, hence unique (since $|X| \geq 3$). Therefore, $\delta-$Morse reproduces the partition $X_1 \cup \ldots \cup X_k$ (in fact, each cluster is a directed star with root x_i).

(**Consistency**) Let d be a distance in X, $\mathcal{P}$ the partition given by δ-`Morse` on (X,d), and d' a $\mathcal{P}$-transformation of d, that is,

$$d(v,u) \geq d'(v,u), \text{ if } v \sim_{\mathcal{P}} u, \tag{5.12}$$

$$d(v,u) \leq d'(v,u), \text{ otherwise.} \tag{5.13}$$

Let Φ respectively Φ' be the Morse flow corresponding to d respectively d'. Let $t \in X$ arbitrary, $v = \Phi(t)$ and $s = \Phi'(t)$ with $v, s \neq t$. As in the proof of Theorem 5.13, we have

$$d'(t,s) \leq d'(t,v) \leq d(t,v) \leq d(t,s).$$

Then either $d'(t,s) < d(t,s)$, and so $t \sim_{\mathcal{P}} s = \Phi'(t)$ by Equation 5.12, or $d'(t,s) = d(t,s)$, which implies, by the definition of link preorder, $v = s$, and thus $t \sim_{\mathcal{P}} s = \Phi'(t)$ too. As t was arbitrary, we conclude that $\mathcal{P}'$ is a refinement of $\mathcal{P}$, by Lemma 5.10. To prove that they are equal, it suffices to show that they have the same critical nodes (i.e. the same number of clusters), that is, $\Phi(v) = v$ if and only if $\Phi'(v) = v$.

Suppose that $\Phi(v_i) = v_i$ and $\Phi'(v_i) = v_j$, with $i \neq j$, the node preorder is strictly increasing along the flow, hence $v_i \prec_V v_j$ and $i < j$. By the definition of `Morse` clustering, $v_i \sim_{\mathcal{P}'} v_j$ hence $v_i \sim_{\mathcal{P}} v_j$, since $\mathcal{P}'$ is a refinement. However, this contradicts v_i being maximal in its $\mathcal{P}$ cluster as $i < j$.

Now suppose $\Phi'(v_i) = v_i$ and $\Phi(v_i) = v_j$, with $i \neq j$. As the link (v_i, v_j) is maximal for v_i and d then $d(v_i, v_j) < \delta$ and $j > i$. Because it is not maximal when the distance is d' we have $d'(v_i, v_j) \geq \delta$. By Equation 5.12, since $v_i \sim_{\mathcal{P}} v_j$ and d' is a $\mathcal{P}$-transformation, we have $d'(v_i, v_j) \leq d(v_i, v_j) < \delta$, which implies that v_i has at least one admissible link (as $j > i$) when d' is the distance on X. By the definition of $\preceq_E$, if v_i is critical for Φ' then either it is the node with highest index or all its out-links have distance value higher or equal than δ. The former is impossible as v_j has higher index than v_i, thanks to $\Phi(v_i) = v_j$ and the latter as well as $d'(v_i, v_j) \leq d(v_i, v_j) < \delta$. □

Our three instances of Morse clustering satisfy Monotonic Consistency: δ-`Morse` and k-`Morse` because they already satisfy Consistency, and `SiR-Morse` by the following Theorem.

Theorem 5.15. *`SiR Morse` satisfies Monotonic Consistency.*

Proof. Let d be a distance on X, $\mathcal{P}$ the output partition of `SiR Morse` on (X, d), and d' a $\mathcal{P}$-monotonic transformation of d. We want to show that `SiR Morse` produces the same partition on (X, d'). We will prove that, in fact, the associate Morse flows Φ and Φ' are identical.

Let η be a monotonic transformation realising d', that is,

$$\begin{aligned} d(u,v) &= \eta(d'(u,v)) && \text{if } u \sim_{\mathcal{P}} v, \text{ and} \\ d(u,v) &= \eta^{-1}(d'(u,v)) && \text{if } u \not\sim_{\mathcal{P}} v. \end{aligned}$$

Let $v \in X$ and consider first the case $s = \Phi(v) \neq v$. Then, by the definition of `SiR Morse` preorders,

$$d(v,s) < d(v,t) \text{ for all } t \neq v, s.$$

To prove that $\Phi'(v) = s$, we need to show that $d'(v,s) < d'(v,t)$ for all $t \neq v, s$. We have two sub-cases.

1. If $t \sim_{\mathcal{P}} v$, we have $d'(v,s) = \eta^{-1}(d(v,s))$ and $d'(v,t) = \eta^{-1}(d(v,t))$, so

$$d(v,s) < d(v,t) \text{ implies } d'(v,s) < d'(v,t),$$

   as η^{-1} is increasing (Lemma 5.3).

2. If $t \not\sim_{\mathcal{P}} v$, we have $d'(v,s) = \eta^{-1}(d(v,s))$ and $d'(v,t) = \eta(d(v,t))$, so

$$d(v,s) < d(v,t) \text{ implies } d'(v,s) \leq d(v,s) < d(v,t) \leq d'(v,t),$$

   as $\eta^{-1}(x) \leq x \leq \eta(x)$ for all x (Lemma 5.3).

In conclusion, we have $d'(v,s) < d'(v,t)$ for all $t \neq v, s$ so $\Phi'(v) = s$.

We show now that when $\Phi(v) = v$ then $\Phi'(v) = v$. Suppose, by contradiction, that $s = \Phi'(v) \neq v$. This implies $v \prec_V s$ and $d'(v,s) < d'(v,t)$ for all $t \neq v, s$. Note that, since v is critical and therefore maximal within its cluster, we have $v \not\sim_{\mathcal{P}} s$. On the other hand, $\Phi(v) = v$ means that either the unique maximal link is not admissible, or it is admissible but the maximum is not unique.

First we show that $d(v,s)$ is also a minimal distance at v (possibly not unique). Suppose, by contradiction, $d(v,t) < d(v,s)$ for some $t \neq v, s$. There are two sub-cases.

1. If $t \sim_{\mathcal{P}} v$, then we have $d'(v,t) = \eta^{-1}(d(v,t))$ and $d'(v,s) = \eta(d(v,s))$, so
$$d(v,t) < d(v,s) \text{ implies } d'(v,t) \leq d(v,t) < d(v,s) \leq d'(v,s),$$
as $\eta^{-1}(x) \leq x \leq \eta(x)$ (Lemma 5.3).
2. If $t \not\sim_{\mathcal{P}} v$, then we have $d'(v,t) = \eta(d(v,t))$ and $d'(v,s) = \eta(d(v,s))$, so
$$d(v,t) < d(v,s) \text{ implies } d'(v,t) < d'(v,s),$$
as η is increasing (Lemma 5.3).

In either case, we have $d'(v,t) < d'(v,s)$, a contradiction to the minimality of $d'(v,s)$.

Since $d(v,s)$ is a minimal distance and $v \prec_V s$, but $\Phi(v) = v \neq s$, the minimal distance (maximal link) has to be not unique. Let $d(v,t) = d(v,s)$ for some $t \neq v, s$. We have, again, two sub-cases.

1. If $t \sim_{\mathcal{P}} v$, then we have $d'(v,t) = \eta^{-1}(d(v,t))$ and $d'(v,s) = \eta(d(v,s))$, so
$$d(v,t) = d(v,s) \text{ implies } d'(v,t) \leq d(v,t) = d(v,s) \leq d'(v,s),$$
as $\eta^{-1}(x) \leq x \leq \eta(x)$ (Lemma 5.3).
2. If $t \not\sim_{\mathcal{P}} v$, then we have $d'(v,t) = \eta(d(v,t))$ and $d'(v,s) = \eta(d(v,s))$, so
$$d(v,t) = d(v,s) \text{ implies } d'(v,t) = d'(v,s),$$
as η is injective (Lemma 5.3).

This implies that $d'(v,t) \leq d'(v,s)$, so $d'(v,s)$ cannot be the unique minimal distance for d' at v, a contradiction. □

Corollary 5.16. *Scale Invariance, Richness and Monotonic Consistency are mutually compatible clustering axioms.*

We have summarised the clustering axioms satisfied by our three instances of Morse clustering in Table 5.1.

	Scale-Invariance	Richness	Consistency	Monotonic-Consistency
`SiR Morse`	✓	✓	✗	✓
k-`Morse`	✓	✗	✓	✓
δ-`Morse`	✗	✓	✓	✓

TABLE 5.1: Clustering axioms and three instances of Morse clustering

We finish this section with a few remarks on node labelling and tie-breaking. Note that our choices of node preorders depend on an arbitrary but fixed choice of node labelling $X = \{x_1, \ldots, x_n\}$. Such a choice is implicit in [Kle03], where it is used as a tie-breaking procedure for Single Linkage clustering. For `Morse` algorithm, on the other hand, this node labelling represents a choice of a node potential function and is fundamental to the algorithm, as only 'uphill' links are admissible. Nevertheless, the results in this section apply to an arbitrary, but fixed, labelling or ordering of the elements in X, and this suffices in our axiomatic setting.

5.4 An Impossibility Theorem for Graph Clustering

In this section, we consider the axiomatic approach in the context of graph clustering, that is, of distances supported on a given graph G. We prove Kleinberg's Impossibility Theorem for graph clustering, and a possibility result for Monotonic Consistency.

Definition 5.17 ([Hel06])**.** A *pseudo-distance on a set* X is a function $d : X \times X \to \mathbb{R}$ such that $d(v, u) = d(u, v) \geq 0$ and $d(v, v) = 0$ for all $v, u \in X$ (that is, we allow $d(u, v) = 0$ for $u \neq v$). A *pseudo-distance on a graph* $G = (V, E)$ is a pseudo-distance d on the node set V that is supported on the link set, that is, $d(v, u) = 0$ if and only if $(v, u) \notin E$. (Equivalently, a positive weight function on undirected links.)

Note that, for this definition to make sense, G must be loopless and undirected (we will assume this from now on). Given a graph $G = (V, E)$, we define a *graph clustering algorithm* as any function

$$F : \{d \text{ pseudo-distance on } G\} \to \{\mathcal{P} \text{ partition of } V\}. \qquad (5.14)$$

Clearly, a distance on a set X is the same as a pseudo-distance on the complete graph with node set $V = X$. Hence this so-called *sparse* setting generalises Kleinberg's setting from a complete to an arbitrary (but fixed) graph on X.

Remark 5.18. This is the natural setting in graph clustering, [Sch07], where the absence of a link is significant: Zero distances are interpreted as 'not defined' or 'not relevant', rather than the actual numerical value 0. For example, some clustering algorithms require a sparse network representation of the data as a pre-processing step [VL07].

When we consider a distance on a set X we tend to value more two nodes with low distance, as that means that they "close". In the sparse case however if two nodes have distance 0 then one could assume that they are identical. From now on we will discard this interpretation and instead consider the pseudo-distances more as weight functions, where links with weight equal to 0 are not connecting identical nodes, but instead they denote an absence of user-knowledge with respect the relationship of those nodes.

Remark 5.19. If we were to allow the support of d to vary, that is, if we consider instead clustering algorithms of the form

$$F : \{d \text{ pseudo-distance on } X\} \to \{\mathcal{P} \text{ partition of } X\},$$

then a possibility theorem holds: If F is the function returning the connected components of the graph representation of d, then F is Consistent, Rich and Scale Invariant (see [LM14]).

Kleinberg's axioms can be stated in the graph clustering setting:

- **Scale-invariance**: For any pseudo-distance d on G and $\alpha > 0$, we have $F(d) = F(\alpha \cdot d)$;
- **Richness**: Given a partition $\mathcal{P}$, there exists a pseudo-distance d on G such that $F(d) = \mathcal{P}$;
- **Consistency**: Given pseudo-distances d and d' on G with $\mathcal{P} = F(d)$, if d' is a $\mathcal{P}$-transformation of d, that is,

$$\begin{cases} d'(v,u) \leq d(v,u) & \text{if } v \sim_{\mathcal{P}} u, \text{ and} \\ d'(v,u) \geq d(v,u) & \text{if } v \not\sim_{\mathcal{P}} u, \end{cases} \tag{5.15}$$

then $F(d') = F(d)$.

(If G is a complete graph these axioms coincide with Kleinberg's for the set $X = V$.)

In the sparse setting it seems natural to restrict to *connected partitions*, that is, partitions where each cluster is a connected subgraph of G, as otherwise we would be grouping together objects which are unknown to be similar or not, in apparent contradiction with the very principle of clustering. Therefore, we define a weaker Richness axiom:

Definition 5.20 (Connected-Richness)**.** Given a graph G and a clustering algorithm F we say that F is *connected-rich* if for any connected partition $\mathcal{P}$ there exists a pseudo-distance d on G such that $F(d) = \mathcal{P}$.

Similarly, we will only consider connected graphs from now on (it seems sensible to assume $F(G) = F(G_1) \cup F(G_2)$ whenever G is the disjoint union of G_1 and G_2).

Connected-Richness is clearly equivalent to Richness in the complete case. In the sparse case, however, many graph clustering algorithms, such as Single Linkage, or `Morse`, always produce a connected partition (which seems very sensible in any case). Since clustering algorithms cannot create new links, such algorithms would not satisfy Richness in its general form on sparse graphs. Since Richness implies Connected-Richness, our impossibility result also holds for Scale-Invariance, Consistency and Richness.

Theorem 5.21 (An Impossibility Theorem for Graph Clustering)**.** *Let G be a connected graph with at least three nodes, and F a graph clustering algorithm on G. Then F cannot satisfy Scale-Invariance, Consistency and Connected-Richness.*

Before proving this theorem, we introduce some notation. Given a pseudo-distance d on $G = (V, E)$ and a partition $\mathcal{P}$ of V, let $g(\mathcal{P}, d) = (x, y)$ and $h(\mathcal{P}, d) = (p, q)$ where

$$\begin{aligned} x &= \max\{d(u,v) \mid (u,v) \in E, u \sim_{\mathcal{P}} v\}, \\ p &= \min\{d(u,v) \mid (u,v) \in E, u \sim_{\mathcal{P}} v\}, \\ y &= \min\{d(u,v) \mid (u,v) \in E, u \not\sim_{\mathcal{P}} v\}, \\ q &= \max\{d(u,v) \mid (u,v) \in E, u \not\sim_{\mathcal{P}} v\}, \end{aligned}$$

the maximal (minimal) intra (inter) cluster distances, and, if $\mathcal{P}$ is the trivial partition, that is, all nodes in only one cluster, we set $y = q = 0$. Similarly if $\mathcal{P}$ is the singleton partition, that is, each cluster has only one node, then we set $x = p = 0$.

We observe that, if d and d' are pseudo-distances on G and $\mathcal{P}$ is a partition of V, the condition $h(\mathcal{P}, d) = g(\mathcal{P}, d')$ guarantees that d' is a $\mathcal{P}$-transformation of d.

Proof. Note that, in any connected graph, we can always remove a node so that the remaining graph is connected. For example, if T is a spanning tree of G, v any node, and s the node realising the maximal (shortest path) distance from v in T, then the graph induced by $V \setminus \{s\}$ must still be connected. Since $|V| \geq 3$, we can repeat the argument on $V \setminus \{s\}$ and find $t \neq s$ such that $\mathcal{P} = \{\{s\}, X \setminus \{s\}\}$ and $\mathcal{P}' = \{\{s\}, \{t\}, X \setminus \{s, t\}\}$ are connected partitions.

Since F satisfies Connected-Richness, there exist pseudo-distances d and d' on G such that $F(d) = \mathcal{P}$ and $F(d') = \mathcal{P}'$. Let $h(\mathcal{P}, d) = (p, q)$ and $h(\mathcal{P}', d') = (p', q')$. Since F satisfies Consistency, we can assume $p < q$ and $p' < q'$. In fact if that is not the case without loss of generality we substitute d (resp. d') with a $\mathcal{P}$ (resp. $\mathcal{P}'$) transformation of d (resp. d') that satisfies such condition. Also, note that p, q and q' cannot be zero. Let d^* be the pseudo-distance on G defined by $d^*(s, v) = q$ if $v \neq s$ and $(v, s) \in E$, $d^*(v, t) = p$ if $v \neq s, t$ and $(v, t) \in E$, and $d^*(u, v) = (pp')/q'$ if $u, v \neq s, t$ and $(u, v) \in E$.

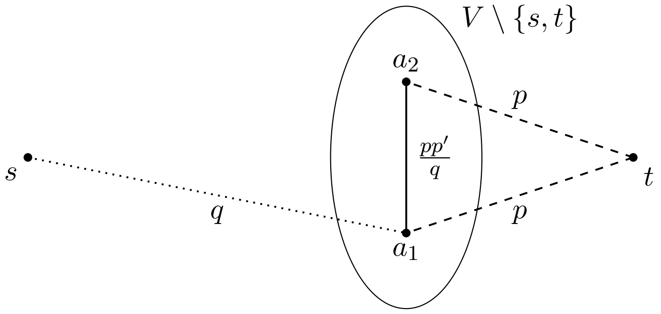


Figure 5.5: Example of G and pseudo-distance d^*.

Then $g(\mathcal{P}, d^*) = (p, q)$, since the only inter-cluster distance value is q, and the only intra-cluster distance values are p and $p(p'/q') < p$. Therefore, $g(\mathcal{P}, d^*) = h(\mathcal{P}, d)$, hence d^* is a $\mathcal{P}$-transformation of d, by the observation before the proof, and, consequently, $F(d^*) = F(d)$, by Consistency.

On the other hand, $g(\mathcal{P}', \alpha d^*) = \alpha g(\mathcal{P}', d^*)$ for any α positive constant. If we choose $\alpha = q'/p$ then we have $g(\mathcal{P}', \alpha d^*) = \alpha((pp')/q', p) = (p', q') = h(\mathcal{P}', d')$ so, by the same argument as above, αd^* is a $\mathcal{P}'$-transformation of d' and thus $F(\alpha d^*) = F(d') = \mathcal{P}'$, by Consistency. Since F satisfies Scale-Invariance, this implies $F(\alpha d^*) = F(d^*) = F(d) = \mathcal{P}$ and, therefore, $\mathcal{P} = \mathcal{P}'$, clearly a contradiction. □

Next we consider Monotonic Consistency and `Morse` clustering in the sparse setting. Clearly, we can extend Monotonic-Consistency to connected graphs by considering monotonic transformations (Definition 5.5) of pseudo-distances on a given graph.

- **Monotonic-Consistency**: Given pseudo-distances d and d' on G with $\mathcal{P} = F(d)$, if d' is a $\mathcal{P}$-monotonic transformation of d, then $F(d') = F(d)$.

The input of `Morse` is an arbitrary graph, and the output flow always induces a connected partition (Theorem 3.2). Therefore, we can consider `Morse` clustering, and hence any of its instances, as graph clustering algorithms.

The three instances of `Morse` clustering discussed in Section 5.3 satisfy the analogous axioms as in the complete case except that we need to allow the node labelling (arbitrary but prefixed in the complete case) to be part of the algorithm to satisfy Connected-Richness. This is a necessary condition: once a node labelling (or preorder) is fixed, only ‘uphill’ links are admissible, preventing certain configurations to occur (for example, u and v cannot be in the same cluster if all paths from u to v contain a node lower than both). This is not an intrinsic limitation of `Morse` clustering but reflects the fact that it is fundamentally a node-weighted clustering algorithm, that is, both distance and node preorder are part of the input data.

We can either allow the (so far arbitrary and prefixed) node labelling to be part of the algorithm, or to restrict to partitions compatible with such a committed choice. Formally, given a node preorder $\preceq_V$ on V, we say that a partition $\mathcal{P} = \{V_1, \dots, V_k\}$ of V is *compatible with* $\preceq_V$ if there is a rooted spanning tree T_i of (the subgraph induced by) V_i rooted at a node v_i such that every directed link in T_i (links directed towards the root) is admissible with respect to $\preceq_V$. Note that v_i is necessarily the maximal node in T_i with respect to the preorder, and that $\mathcal{P}$ is necessarily a connected partition.

Remark 5.22. One can show that $\mathcal{P}$ is compatible with $\preceq_V$ if and only if for every $u \sim_{\mathcal{P}} v$ there exists a path from u to v such that no node in the path is strictly less than both u and v.

Clearly, for every partition there is a choice of compatible preorder $\preceq_V$. This is also true for the `SiR` and δ-`Morse` node preorders: given a partition, there is a choice of labelling $V = \{v_1, \dots, v_n\}$ such that the preorder is compatible with the partition.

Formally, we define Morse-Richness for a `Morse` clustering algorithm F on a graph $G = (V, E)$ with a choice of node preorder $\preceq_V$ as follows.

- **Morse-Richness**: Given a partition $\mathcal{P}$ of V compatible with $\preceq_V$, there exists a pseudo-distance d on G and a node preorder such that $F(d) = \mathcal{P}$.

Morse-Richness is equivalent to Connected-Richness if we accept the node labelling as an input (mutable) of the algorithm.

Now we can show that the three instances of `Morse` clustering satisfy the analogous axioms as in Section 5.3 (see Table 5.1), including a possibility theorem for Monotonic-Consistency and `SiR Morse`.

Theorem 5.23. *Let $G = (V, E)$ be a graph, and consider `SiR Morse`, k-`Morse` and δ-`Morse` as graph clustering algorithms on G, for some fixed labelling $V = \{v_1, \dots, v_n\}$. Then:*

(i) `SiR Morse` satisfies Scale-Invariance, Morse-Richness and Monotonic Consistency.

(ii) k-`Morse` satisfies Scale-Invariance and Consistency.

(iii) δ-`Morse` satisfies Morse-Richness and Consistency.

Proof. $\boxed{\text{i}}$ The proofs of Scale Invariance and Monotonic Consistency are identical (they do not use the fact that G is a complete graph) as those in Theorem 5.11. For Morse-Richness, consider $V = V_1 \cup \ldots \cup V_k$ an arbitrary connected partition of V. For each V_i, choose a spanning tree T_i and a root v_i such that each link in T_i is admissible.

Define a pseudo-distance d on G as follows. If (s,t) is a link on T_i, then $d(s,t)$ is the maximum of the distance from s to v_i in T_i and the distance from t to v_i in T_i (by distance in a tree we simply mean the 'hop' distance). If (s,t) is an link not in any spanning tree, then $d(s,t) = |V|$.

With this choice, v_i is critical and, if $v \in V_i$, then the maximal link at v is the one connecting it to a node in T_i closer to v_i, and it is admissible. All in all, the associated tree $T_{v_i} = T_i$ and the Morse flow recovers the original partition.

$\boxed{\text{ii}}$ The proof of Scale Invariance is identical to that in Theorem 5.12. For Consistency, let d be a pseudo-distance on G, $\mathcal{P}$ the partition given by k-`Morse`, and d' a $\mathcal{P}$-transformation of d, that is,

$$\begin{cases} d(v,u) \geq d'(v,u), & \text{if } v \sim_{\mathcal{P}} u, \\ d(v,u) \leq d'(v,u), & \text{otherwise.} \end{cases}$$

Let $\varPhi$ respectively $\varPhi'$ be the Morse flow corresponding to d respectively d'. As in the proof of Theorem 5.12, for all $i > n-k$ we have that $\varPhi(v_i) = v_i = \varPhi'(v_i)$, critical.

Suppose now $\varPhi(v_i) = v_i$ for some $i \leq n-k$. Let $J = \{v_j \mid (v_i, v_j) \in E, v_i \prec_V v_j\}$, the admissible links from v_i. By the definition of the link preorder, if there are admissible links ($J \neq \emptyset$) then the maximal admissible link exists and it is unique. Since v_i is critical, we must have $J = \emptyset$. Since there are no admissible links at v_i, we also have $\varPhi'(v_i) = v_i$. All in all, $\varPhi$ and $\varPhi'$ have the same number of critical nodes and therefore $\mathcal{P}$ and $\mathcal{P}'$ have the same number of clusters (possibly more than k). The rest of the proof goes as in the proof of Theorem 5.13.

$\boxed{\text{iii}}$ The proof of Consistency is identical to that in Theorem 5.14. For Morse-Richness, consider $V = V_1 \cup \ldots \cup V_k$ an arbitrary connected partition of V, and choose a spanning tree T_i and a root v_i such that each link in T_i is admissible.

Define a pseudo-distance d on G as follows. If (s,t) is an link in some T_i, then $d(s,t) = \delta/2$, and if (s,t) is not an link in any T_i then $d(s,t) = \delta$. By definition of link preorder, v_i is critical and the maximal link at $v \in V_i \setminus \{v_i\}$ is the only link in T_i connecting v to a node closer to v_i in T_i. All in all, the tree associated to v_i by the Morse flow is T_i and hence we recover the original partition. □

Chapter 6

Laplacian

We showed that `Morse` is a fast clustering algorithm and how its application can prove its ability to discern basins of attraction with respect the annotation. In the analysis of a dataset, however, its locality could be a drawback, for example if the annotation presents too many local peaks, and so `Morse` would return more clusters and basins of attraction than expected. In such cases, `Morse` could be better used as a preprocessing step for a global algorithm, such as spectral clustering.

One of the main goals in graph theory is to deduce the principal properties and structure of a graph from its Laplacian spectrum, such as to determine when two graphs are similar or how links perturbations effect eigenvalues, [Mer94, WZ08]. The discrete graph Laplacian is the counterpart of a well-known operator in differential analysis, the *Laplacian operator*. Given a function f on a n-dimensional Euclidean space the Laplacian operator of f, written as Δf, is defined as the divergence of the gradient of f, or equivalently the sum of all the *unmixed* second partial derivatives in the Cartesian coordinates $\{x_1, \dots, x_n\}$,

$$\Delta f = \operatorname{div}(\nabla f) = \sum_{i=1}^{n} \frac{\partial^2 f}{\partial x_i^2}. \tag{6.1}$$

The differential Laplacian operator was named after Pierre-Simone de Laplace, who applied it to the study of celestial mechanics [mdL22], but it has been employed in different type of physical phenomena and equations such as heat equation, Schrödinger equation, wave equation, Poisson equation or Laplace equation [Str92]. As their name suggest, the solution of these equations has engaged leading mathematics for centuries. The mathematical problems those bring have proven to be virtually inexhaustible and a central role is played in fact by the Laplacian.

The (discrete) Laplacian L on a graph $G = (V, E)$ is defined as a map on the set of real-valued functions on V. More precisely, if f is such function then

$$L(f)(v) = \sum_{w \neq v} w(v, w)(f(v) - f(w)),$$

with $w : E \subset V \times V \to \mathbb{R}$ the weight function on the links of G. We can consider this set as the real vector-space, namely $\mathbb{R}^n$, with n the number of nodes in G, and so given an indexing on V we can write L as a matrix

$$L(i, j) = \begin{cases} -w(v_i, v_j), & \text{if } i \neq j \\ \sum_{s \neq i} w(v_i, v_s), & \text{otherwise.} \end{cases} \tag{6.2}$$

There are various results that show how the Laplacian encodes properties and structure of G. A well-known result is that the dimension of $\mathrm{Ker}(L)$ is equal to the number of connected components of G, [MACO91]. The smallest positive eigenvalue of L relates to how difficult is to split G in two disjoint graphs while the eigenvector of its largest eigenvalue has been used as a centrality measure and well-known variants of such measure are PageRank [LK14] and GeneRank [MBHG05]. In [HJ13b] the Laplacian operator has been extended to high dimensional simplicial complexes showing that it encodes as well the properties and structure of these high dimensional complexes.

The discrete Laplacian can also be seen as an approximation of the continuous Laplacian as follows. Given a function $f : \mathbb{R}^2 \to \mathbb{R}$ and using finite-difference method, we can write an approximation of $\nabla f(x, y)$ at each point $(x, y) \in \mathbb{R}^2$ as follows

$$\Delta f(x, y) \approx \frac{f(x-h, y) + f(x+h, y) + f(x, y-h) + f(x, y+h) - 4f(x, y)}{h^2},$$

with h small enough. Let G be a graph with nodes $v_1 = (x, y)$, $v_2 = (x-h, y)$, $v_3 = (x+h, y)$, $v_4 = (x, y-h)$, $v_5 = (x, y+h)$ in $\mathbb{R}^2$ and links of the form (v, w) with either v or w equal to the point (x, y). If the weight of each link is $\frac{1}{h^2}$ then the discrete Laplacian of G on $v_1 = (x, y)$ coincides with the approximation of Δf in (x, y), with sign changed

$$L(f)(v_1) = \frac{1}{h^2} \sum_{i=2}^{5} (f(v_1) - f(v_i)) \approx -\Delta f(v_1).$$

To use a prescribed partition into spectral clustering we will require a definition of Laplacian for the quotient graph such partition induces. To our knowledge, the mathematical machinery for this approach is not quite developed so we decided to investigate it, generalising the study of [GHL15] on spectral distance between graphs. In [HJ13a] the authors studied the effect on simplicial complexes (see Definition 2.11) of *collapses* and *contraction* [Whi49], that are respectively the removal of a simplex and one of its faces, and the identification of pair of simplices, on such high dimensional Laplacian. This study has been then employed for comparison of graph structures in [GHL15], where a notion of *spectral distance* between graphs is introduced. This distance can be used to measure differences between graphs but as well the effect of structural changes on a graph, this includes the changes induced by link collapses, which is effectively the identification of its nodes. Using the approach of [HJ13b], we were able to formalise the definition of Laplacian on a quotient graph induced by a partition, that is, a graph where nodes represent clusters of the partition, and so determine the distance between such quotient graph and the starting one.

After recalling some properties of the Laplacian, in Section 6.3 we present two different bounds for the spectral distance between a graph and the quotient graph induced by a partition, improving the bounds presented in [GHL15]. In Section 6.4 we show how these bounds can hold when an overlapping clustering is given and how to overcome the inapplicability of techniques presented in [HJ13a] for this case. This work gives us the possibility to control and compare different clustering solutions using a spectral distance, but as well the possibility of constructing a `Morse`-spectral clustering algorithm, in which `Morse` provides a fast but local preprocessing step and spectral clustering is used to globally study the quotient graph induced by the Morse partition.

6.1 Discrete Laplacian

We recall now some definitions and properties of the Laplacian operator, for a more through reading we refer to [Mer94, GR01, MACO91].

Definition 6.1 ([Mer94]). Let $G = (V, E)$ be a graph and $w : E \to \mathbb{R}$ a link weight function on it. The *Laplacian* of G is a linear operator $L : \mathbb{R}^n \to \mathbb{R}^n$, with $|V|= n$, defined by the matrix

$$L(i,j) = \begin{cases} -w(i,j) & \text{if } i \neq j \\ \sum_{s\neq i} w(i,s) & \text{otherwise.} \end{cases}$$

We assume from now on that w is a positive function and that $w(i,j) = 0$ only when i and j are not connected in G. We consider $w : E \to \mathbb{R}$ as we focus on simple graphs, that is, any node i is not connected to itself and has at most one link with another node. We recall the definition of *weighted degree matrix* D, that is,

$$D(i,j) = \begin{cases} \sum_{s\neq i} w(i,s), & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases}$$

The entry $D(i,i)$ is called the *degree* of node i. Note that a node will have degree equal to 0 if and only if it is disconnected from any other node. In a similar way, we can define the *adjacency matrix* A as

$$A(i,j) = \begin{cases} w(i,j) & \text{if } i \neq j \\ 0 & \text{otherwise.} \end{cases}$$

It is clear that

$$L = D - A.$$

Given a graph G and its Laplacian we can actually study its topology, as follows.

Theorem 6.2 ([MACO91])**.** *Given a graph G and a weight function w, let L be the Laplacian operator associated to them. Then L is a semi-positive definite operator with non-negative real eigenvalues and its kernel has dimension equal to the number of connected components of G.*

Without loss of generality we will assume from now on that G is connected. As the Laplacian operates separately on each connected component of G, its spectrum will be simply the union of the Laplacian operator spectra defined on each component, but the eigenvalues multiplicity may be affected. Note that an upper bound for the Laplacian eigenvalues which is valid for any graph (even only the connected ones) does not exist. A common procedure is to normalise the Laplacian as follows.

Definition 6.3 ([Mer94])**.** Given a graph G and a weight function w, the *symmetric normalised Laplacian* is a linear operator $\mathcal{L} : \mathbb{R}^n \to \mathbb{R}^n$ defined as

$$\mathcal{L}^{sym} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}},$$

with D the weighted degree matrix.

The kernels of L^{sym} and L, as linear operators, are isomorphic as D is invertible, and in addition the spectrum of L^{sym} sits inside the interval $[0, 2]$ with its eigenvalues summing up to n, [Mer94]. Another version of the normalised Laplacian is called *random walk normalised Laplacian*,

$$\mathcal{L}^{rw} = D^{-1} L.$$

Given a random walker on the graph, for $i \neq j$, each entry $\mathcal{L}^{rw}(i,j)$ is its probability to move from i to j. In general, such an operator is not symmetric, however $\mathcal{L}^{rw}$ and $\mathcal{L}^{sym}$ share the same eigenvalues as they are similar matrices. So if we want to study their spectral properties we can consider one or the other equivalently.

In [HJ13b] the Laplacian operator is defined for simplicial complexes and their chain complexes. Similarly to Chapter 2, we will assume that the boundary map ∂ satisfies Equation 2.2 and as well the inner product on $C_*(K)$ satisfies Equation 2.3. In particular, this means that in a graph we have

$$\begin{cases} \partial_1(e) & = & \pm(v-w) & \text{with } e=(v,w)\in E \\ \partial_0(v) & = & 0 & \text{with } v\in V. \end{cases}$$

Given these assumptions we can redefine the Laplacian as follows.

Definition 6.4 ([HJ13b])**.** Given a simplicial complex K and its chain complex $C_*(K)$ with boundary operator ∂, let $\partial^* : C_*(K) \to C_{*+1}(K)$ be the *adjoint operator* of δ respect to its inner product, that is, for any $c, d \in C_*(K)$ we have

$$< \partial^*(c), d >=< c, \partial(d) > \tag{6.3}$$

Then the i^{th} *Laplacian operator* L_i is defined as

$$L_i = \partial_{i+1}\partial_i^* + \partial_{i-1}^*\partial_i : C_i(K) \to C_i(K).$$

This definition assumes that K is unweighted, meaning $w(\sigma) = 1$ for any $\sigma \in K$. In general, we can define a weighted version as follows. Suppose we have $w : K \to \mathbb{R}$, a positive weight function on K, and let $W_i : C_i(K) \to C_i(K)$ be its linear extension to each $C_i(K)$, that is, the linear operator on $C_i(K)$ with matrix form such that the entry $W_i(\sigma, \tau) = w(\sigma)$ if $\tau = \sigma$ and 0 otherwise. Then the *weighted Laplacian* is

$$L_i^W = \partial_{i+1}W_{i+1}\partial_i^*W_i^{-1} + W_i^{-1}\partial_{i-1}^*W_{i-1}\partial_i.$$

Note that in the case of a graph G the weighted Laplacian on the nodes is exactly the random walk normalised Laplacian

$$L_0^W = \partial_1 W_1 \partial_0^* W_0^{-1} = \mathcal{L}^{rw},$$

with W_1 linear extension on $C_1(G)$ of a weight function on G while W_0 is the linear extension of the weighted degree on $C_0(G)$. As in the literature the Laplacian is often considered as a symmetric operator we consider here a different definition for the weighted Laplacian

$$L_i^W = W_i^{-\frac{1}{2}}\partial_{i+1}W_{i+1}\partial_i^*W_i^{-\frac{1}{2}} + W_i^{-\frac{1}{2}}\partial_{i-1}^*W_{i-1}\partial_i W_i^{-\frac{1}{2}}.$$

This Laplacian is symmetric and it has the same spectrum of the one defined in Definition 6.4 as they are similar matrices. Clearly in the case of a graph we have that

$$L_0^W = W_0^{-\frac{1}{2}}\partial_1 W_1 \partial_0^* W_0^{-\frac{1}{2}} = \mathcal{L}^{sym}.$$

After recalling here some definition of the graph Laplacian and some of its properties, both for the normalised and unnormalised version, and its generalisation to high dimensional complexes, we will move in the next section to the definition of *quotient graph* induced by a partition and how to define a Laplacian operator on it. We will show how this procedure can be formalised as in [HJ13b] when the partition is extrapolated from a Morse flow and the relationships between the Laplacian on a graph and on its quotient graph.

6.2 Quotient graph and its Laplacian

In this section we introduce the definition of quotient graph induced by a clustering on a graph G constructed using a similar procedure to [HJ13b]. Thanks to this construction and the definition of its Laplacian, we are able to define the distance between a graph and its clustering and detect how much the graph structure is preserved by the clustering.

In order to define the graph induced by a clustering we need to recall the notion of *simplicial map*.

Definition 6.5 ([HJ13a])**.** Given two simplicial complex K and Z a simplicial map φ is a map between their nodes

$$\varphi : K_0 \to Z_0,$$

such that whenever a simplex is spanned in K by $v_1, \dots, v_k$ then $\varphi(v_1), \dots, \varphi(v_k)$ span a simplex in Z.

Let G be a graph and $\mathcal{P} = \{S_i\}_{i=1}^k$ a partition of its nodes, that is, $V = S_1 \cup \dots \cup S_k$ disjoint union. We can define the *quotient graph* with respect to the partition $\mathcal{P}$ as the graph $G_\mathcal{P} = (V_\mathcal{P}, E_\mathcal{P})$ such that $V_\mathcal{P} = \mathcal{P}$. Define now φ as follows

$$\varphi : V \to \mathcal{P}, \;\; v \mapsto S_i, \;\text{ if } v \in S_i.$$

As we want φ to be a simplicial map we need to impose that

$$E_\mathcal{P} = \{(S_i, S_j) \mid \exists v \in S_i, w \in S_j \; s.t. \; (v, w) \in E\}.$$

If two connected nodes v, w are in the same cluster S_i we will have that $G_\mathcal{P}$ is not a simple graph, as the link (v, w) produces the self-loop (S_i, S_i) in $E_\mathcal{P}$. Thanks to our assumption on the boundary map, however, if e is such self-loop then $\partial_1(e) = 0$ and so we can assume $E_\mathcal{P}$ without such links as the Laplacian ignores self-loops. Note that if $\mathcal{P}$ is induced by a Morse function, that is, by a Morse flow $\varPhi$, then φ coincides exactly with $\varPhi^N$, where N is such that $\varPhi^N = \varPhi^{N+1}$. Moreover, as we can actually identify each cluster with its root in G we have that $G_\mathcal{P}$ is in fact the result of a sequence of link contractions in G, see [For98, HJ13b].

In order to define a *quotient Laplacian* on $G_{\mathcal{P}}$ we need to introduce the *quotient matrix* P relative to G and $\mathcal{P}$ defined as the $k \times n$ binary matrix, such that for any pair of node v_j and cluster S_i we have

$$P(i,j) = \begin{cases} 1, & \text{if } v_j \in S_i \\ 0, & \text{otherwise.} \end{cases}$$

Such a matrix determines the linear map $\varphi : C_0(G) \to C_0(G_{\mathcal{P}})$, an extension of the simplicial one, and we have the following well-known result about quotient graphs, see [GR01].

Lemma 6.6 ([GR01])**.** *Let G be a graph, if $\mathcal{P}$ is a partition of G with partition matrix P then the matrix $H = P \cdot P^T$ is an invertible diagonal matrix and $H(i,i) = |S_i|$.*

Proof. Let w be a vector in $\mathbb{R}^k$ then

$$w^T H w = \|P^T w\|^2,$$

where $\| \ \|$ is the euclidean norm in $\mathbb{R}^n$, so H is semi-definite positive. For contradiction, suppose H is singular; then there exists $w \neq 0$ such that $\|wP\|^2 = 0$. We can write then

$$\sum_i w_i P(i,j) = 0, \ j = 1, \ldots, n.$$

For each j there exists only one $i = i(j)$ such that $P(i,j) \neq 0$, that is, each node v_j belongs to only one cluster $S_{i(j)}$, so $\sum_i w_i P(i,j) = w_i = 0$. As each cluster S_i contains at least one node then $w = 0$.

From definition of P we can write

$$H(i,j) = \sum_k P(i,k)P(j,k),$$

and as $S_i \cap S_j = \emptyset$ if $i \neq j$, we have

$$H_{i,j} = \begin{cases} s_i = |S_i|, & \text{if } i = j \\ 0, & \text{otherwise.} \end{cases}$$

□

This lemma is equivalently saying that P induces a surjective map on $C_0(G_{\mathcal{P}})$.

When it is not possible to deduce any sufficiently strong claim about the graph's eigenvalues, it is useful to rely on a few well-known *interlacing* theorems instead of focusing on the whole spectrum of a graph. We will now recall some of these theorems as they will be useful through the entire chapter and they are usually the first, not always successful, step for studying the spectral properties of a graph. We refer the interested reader to [BR$^+$91, MACO91, GR01] for a more thorough analysis and discussion of these interlacing results.

Theorem 6.7 (Cauchy Interlacing Theorem [MACO91])**.** *Let A be a real symmetric $n \times n$ matrix with eigenvalues $\lambda_1 \leq \cdots \leq \lambda_n$, and let B be an arbitrary principal submatrix of A obtained deleting k rows and k columns of A with the same indices. If the eigenvalues of B are $\mu_1 \leq \cdots \leq \mu_{n-k}$ then*

$$\lambda_i \leq \mu_i \leq \lambda_{i+k}, \ i = 1, \dots, n-k$$

Note that if we consider a graph G the above theorem is telling us that the eigenvalues of the Laplacian on the graph G and on the graph G' obtained removing k nodes have an interlacing property. In particular, if we remove only one node the above formula becomes $\lambda_1 \leq \mu_1 \leq \lambda_2 \leq \mu_2 \leq \cdots \leq \mu_{n-1} \leq \lambda_n$, which clarifies the name. Note that Godsil [GR01] reserves the name of 'interlacing' only for this latter case and calls it generalised interlacing otherwise. A well-known theorem by Haemers [BH11] gives us a similar result in the case of quotient-graphs. Let A be $n \times n$ square matrix indexed by the set $I = \{1, \dots, n\}$ and $\mathcal{P} = \{S_i\}_i^k$ be an arbitrary partition of I, such that $I = S_i \cup \cdots \cup S_k$ disjoint union. For i, j let A_{S_i,S_j} be the submatrix of A with rows in S_i and columns in S_j, and let $b_{i,j}$ denote the average row sum of A_{S_i,S_j} then the matrix $B = (b_{i,j})$ is called the *quotient matrix* of A with respect $\mathcal{P}$ and the following holds.

Theorem 6.8 ([BH11])**.** *Let A be a real symmetric $n \times n$ matrix with eigenvalues $\lambda_1 \leq \cdots \leq \lambda_n$, and B its quotient matrix with eigenvalues $\mu_1 \leq \cdots \leq \mu_k$ then*

$$\lambda_i \leq \mu_i \leq \lambda_{i+n-k}, \ i = 1, \dots, k$$

All these theorem are a special case of a more general one which will be essential later in this chapter.

Theorem 6.9 (Theorem 2.1 [BH11])**.** *Let P be a real $n \times m$ matrix such that $P^T P = \mathrm{Id}$ and let A be a symmetric $n \times n$ matrix with eigenvalues $\lambda_1 \leq \cdots \leq \lambda_n$. Define $B = P^T AP$ and let B have eigenvalues $\mu_1 \leq \cdots \leq \mu_k$.*

(i) The eigenvalues of B interlace those of A, that is

$$\lambda_i \leq \mu_i \leq \lambda_{i+n-m}, \ i = 1, \dots, m;$$

(ii) If $\mu_i = \lambda_i$ or $\mu_i = \lambda_{i+n-m}$ for some $i = 1, \dots, m$ then B has a μ_i-eigenvector v such that Pv is μ_i-eigenvector of A;

(iii) If the interlacing is tight, that is, $\lambda_i = \mu_i$ or $\mu_i = \lambda_{n-m+i}$ for all $i = 1, \dots, k$, then $PB = AP$.

From the theorem above it is clear that if we define the Laplacian on $G_{\mathcal{P}}$ as $L_{\mathcal{P}} = PLP^T$ and similarly the normalised Laplacian as $\mathcal{L}_{\mathcal{P}} = P\mathcal{L}P^T$, we do not necessarily have any interlacing property, as $PP^T \neq \mathrm{Id}$. To obtain those properties, we will consider the quotient matrix P of a graph as the $k \times n$ real matrix such that

$$P(i,j) = \begin{cases} \frac{1}{\sqrt{s_i}} & \text{if } v_j \in S_i \\ 0 & \text{otherwise,} \end{cases}$$

with as before $s_i = |S_i|$. From Lemma 6.6 we have that P satisfies the hypothesis of Theorem 6.9. Let the Laplacian on $G_{\mathcal{P}}$ be defined as follows.

Definition 6.10. Given a weighted graph G and a partition $\mathcal{P}$ let P be the quotient matrix relative to $\mathcal{P}$, with $PP^T = \mathrm{Id}_k$, then the *weighted quotient Laplacian* on $G_{\mathcal{P}}$ is defined as

$$L_{\mathcal{P}} = PLP^T, \tag{6.4}$$

with L the weighted Laplacian of G.

As Theorem 6.9 holds for $S = P^T$ we have that L and $L_{\mathcal{P}}$ interlace, that is, if $\{\lambda_i\}_{i=1}^n$ are the eigenvalues of L and $\{\mu_i\}_{i=1}^k$ of $L_{\mathcal{P}}$ then

$$\lambda_i \leq \mu_i \leq \lambda_{n-k+i}, i = 1, \dots, k.$$

More, as P has right inverse, $\mathrm{Ker}(L_{\mathcal{P}})$ is isomorphic to $\mathrm{Ker}(L)$ and so the multiplicity of 0 is the same for both. This is explained on the other hand by the fact that the map φ is simplicial so preserves the connectivity of G and consequently the connected components of $G_{\mathcal{P}}$ and G are equal in number.

It is important to notice that the above definition of quotient Laplacian when the link weight is constant to 1 has been introduced by Haemers in [BH11] for the case of almost equitable partitions, that is, when for any cluster S_i the number of neighbours of a node $v \in S_i$ in a different cluster S_j does not depend on the choice of v in S_i. We presented it here from the point of view of simplicial maps and as an operator on the chain complex of $G_{\mathcal{P}}$ for any partition and any link weight function. Before we proceed to the case of a normalised version of Laplacian, we want to focus on the expression of the entries of $L_{\mathcal{P}}$.

Lemma 6.11. *Given a partition $\mathcal{P}$ we define the* cut *between two clusters as the weighted sum of the links between S_i and S_j, that is*

$$cut(S_i, S_j) = \sum_{p \in S_i} \sum_{q \in S_j} w(p,q).$$

The quotient Laplacian $L_{\mathcal{P}}$ can be written then as

$$L_{\mathcal{P}}(i,j) = \begin{cases} -\frac{cut(S_i,S_j)}{\sqrt{s_i s_j}}, & \text{if } i \neq j \\ \frac{\mathrm{Vol}(S_i)-cut(S_i,S_i)}{s_i}, & \text{otherwise}, \end{cases}$$

where $\mathrm{Vol}(S_i) = \sum_j cut(S_j, S_i) = \sum_{j \in S_i} D(j,j)$.

Proof. As $L_{\mathcal{P}} = PLP^T$ we have

$$L_{\mathcal{P}}(i,j) = \sum_{p,q} P(i,p)L(p,q)P(j,q).$$

From the definition of P we can then write for $i \neq j$

$$L_{\mathcal{P}}(i,j) = \sum_{p \in S_i} \sum_{q \in S_j} \frac{L(p,q)}{\sqrt{s_i}\sqrt{s_j}} = -\frac{cut(S_i,S_j)}{\sqrt{s_i s_j}},$$

as wanted. To prove the case $i = j$, we will use the fact that the vector $v = (1, \ldots, 1)$ is a 0-eigenvector of $L(G)$, so consider the vector $w = (\sqrt{s_1}, \ldots, \sqrt{s_k}) \in \mathbb{R}^k$. Then $P^T w = v$ and so $L_{\mathcal{P}} w = 0$. This implies that

$$L_{\mathcal{P}}(i,i)\sqrt{s_i} - \sum_{j \neq i} -\frac{cut(S_i,S_j)\sqrt{s_j}}{\sqrt{s_i s_j}} = 0$$

So $L_{\mathcal{P}}(i,i) = \frac{\mathrm{Vol}(S_i)-cut(S_i,S_i)}{s_i}$. □

If we consider now $G_{\mathcal{P}}$ as a simplicial complex and we define on it δ as in Equation 2.2, then we can consider the Laplacian $L_{\mathcal{P}}$ as a normalised-like, or weighted, Laplacian of $G_{\mathcal{P}}$. If we take $w(S_i,S_j) = cut(S_i,S_j)$ as defined before, then we can consider W_1 as the linear extension of w on $C_1(G_{\mathcal{P}})$. If we define now W_0 as the linear extension of the map $f : \mathcal{P} \to \mathbb{R}$ such that $f(S_i) = s_i$ then we have that

$$L_{\mathcal{P}} = W_0^{-\frac{1}{2}} \delta_0^* W_1 \delta_1 W_0^{-\frac{1}{2}}.$$

So the normalisation in this case is carried out by the size of each cluster. In the case of the *normalised quotient Laplacian* we could simply normalise as in the graph-Laplacian case. That is, define $D_{\mathcal{P}}$ as the diagonal matrix such that $D_{\mathcal{P}}(i,i) = L_{\mathcal{P}}(i,i)$ and then the normalised quotient Laplacian would be

$$\mathcal{L}_{\mathcal{P}} = D_{\mathcal{P}}^{-\frac{1}{2}} L_{\mathcal{P}} D_{\mathcal{P}}^{-\frac{1}{2}}.$$

However such operation does not bring us any interlacing property neither with L nor with $\mathcal{L}$. This is a consequence of the fact that

$$\begin{aligned}\mathcal{L}_{\mathcal{P}} &= R_1 L R_1^T, \text{ with } R_1 = D_{\mathcal{P}}^{-\frac{1}{2}} P \\ \mathcal{L}_{\mathcal{P}} &= R_2 \mathcal{L} R_2^T, \text{ with } R_2 = D_{\mathcal{P}}^{-\frac{1}{2}} P D^{\frac{1}{2}}.\end{aligned}$$

For these R_1, R_2 Theorem 6.9 does not hold, as $R_1 R_1^T$ is the diagonal matrix $D_{\mathcal{P}}^{-1}$, which is not the identity in general, while $R_2 R_2^T$ is a diagonal matrix with entries of the form $\frac{Vol(S_i)}{L_{\mathcal{P}}(i,i)}$ which is the identity if and only if each cluster S_i is completely disconnected. As already explained in Section 5.4, such possibility is not sensible.

There is, however, a normalisation matrix for $L_{\mathcal{P}}$ that gives us the interlacing properties desired.

Theorem 6.12. *Let G be a graph and $\mathcal{P} = \{S_i\}_{i=1}^k$ a partition of its nodes. Consider D the degree matrix and $D_{\mathcal{P}}$ the* average volume matrix *defined as the diagonal matrix with entries $D_{\mathcal{P}}(i,i) = \frac{\mathrm{Vol}(S_i)}{s_i}$. Then the matrix R defined as*

$$R = D_{\mathcal{P}}^{-\frac{1}{2}} P D^{\frac{1}{2}}$$

is such that $RR^T = \mathrm{Id}_k$.

Proof. Let $H = RR^T$, we know from the definition of R that

$$H = D_{\mathcal{P}}^{-\frac{1}{2}} (PDP^T) D_{\mathcal{P}}^{-\frac{1}{2}}.$$

So we have that

$$H(i,j) = \frac{\sqrt{s_i}}{\sqrt{\mathrm{Vol}(S_i, S_i)}} \sum_{\substack{p \in S_i \\ q \in S_j}} P(i,p) D(p,q) P(j,q) \frac{\sqrt{s_j}}{\sqrt{\mathrm{Vol}(S_j, S_j)}}.$$

We know that $D(p,q) \neq 0$ only when $p = q$ and $P(i,p)P(j,p) \neq 0$ only if $i = j$. This implies that

$$H(i,j) = \begin{cases} \frac{s_i}{\mathrm{Vol}(S_i,S_i)} \sum_{p \in S_i} \frac{D(p,p)}{s_i} = 1, & \text{if } i = j \\ 0, & \text{otherwise.} \end{cases}$$

□

Thanks to this property we can now define the *normalised quotient Laplacian* and thanks to Theorem 6.9 we know that an interlacing property holds for $S = R^T$.

Definition 6.13. Given G weighted graph and $\mathcal{P}$ a clustering of G, let R be as in Theorem 6.12 then the *normalised quotient Laplacian* of $G_{\mathcal{P}}$ is defined as

$$\mathcal{L}_{\mathcal{P}} = R \mathcal{L} R^T.$$

Note that unless $\mathcal{P}$ is the singleton partition the trace of $\mathcal{L}_{\mathcal{P}}$ (sum of its eigenvalues) will never be equal to k, number of clusters. This thanks to the fact that if one diagonal entry of $\mathcal{L}_{\mathcal{P}}$ is equal to 1 it means that the respective cluster is totally disconnected, which we excluded as it means that we group together objects with no connection whatsoever. Furthermore, if we want to see such Laplacian in the form of symmetric normalised Laplacian we can write $\mathcal{L}_{\mathcal{P}}$ as follows

$$\begin{aligned}
\mathcal{L}_{\mathcal{P}} &= R\mathcal{L}R^T \\
&= (D_{\mathcal{P}}^{-\frac{1}{2}}PD^{\frac{1}{2}})D^{-\frac{1}{2}}LD^{-\frac{1}{2}}(D^{\frac{1}{2}}P^TD_{\mathcal{P}}^{-\frac{1}{2}}) \\
&= D_{\mathcal{P}}^{-\frac{1}{2}}PLP^TD_{\mathcal{P}}^{-\frac{1}{2}} \\
&= D_{\mathcal{P}}^{-\frac{1}{2}}L_{\mathcal{P}}D_{\mathcal{P}}^{-\frac{1}{2}}.
\end{aligned}$$

So in this case the normalisation of $L_{\mathcal{P}}$ is carried by the matrix $\mathcal{D}_{\mathcal{P}}$.

We have here explained how we can define a (normalised) Laplacian for a quotient graph, in the case of any partition on G and any type of weight. As interlacing properties hold thanks to Theorem 6.9, we are able to investigate further the relationships between a graph and its quotient graph. In the next section using the spectral distance proposed in [GHL15] we will show how to bound the structural changes a partition induces on a weighted graph using the normalised quotient Laplacian just introduced, which has spectrum contained in the interval $[0, 2]$, as the normalised graph Laplacian.

6.3 Spectral distance and bounds

Any clustering algorithm can always provide a partition of the graph, no matter if the partition reveal effectively patterns or significant structure. We already explored in Chapter 5 how we can control or test the confidence we have on a clustering algorithm using an axiomatic approach. We consider here a different approach which focuses on measuring the quality of the clustering solution. As different approaches usually lead to different clusters, effective quantitative criteria are important to provide the users a measure of confidence for the solution obtained. Generally, these criteria are divided in three categories: *external indices*, based on some prespecified structure, that is, external information, and used to validate the clustering; *internal indices*, independently from external information examine the clustering structure, such as compactness, connectivity and so on; *relative indices*, which focus on the comparison of different clustering structures so to determine which reveals desirable characteristics [XW05]. Here we present a study of a clustering solution with respect to the changes it will induce on the graph when we consider each cluster as an unique object, that is, when we construct the associated quotient graph. In this sense we present here an internal index for clustering algorithms that employs the spectral description of a graph via the symmetric normalised Laplacian. Combining the work of [GHL15] for graph spectral distance we are able to measure approximately how a graph differs in structure and properties from its quotient graph induced by a clustering. This will allow the user to understand when and how much a clustering solution reflects the graph structure.

The section is organised as follows. First we will explain how the spectral distance between graphs is defined in [GHL15] and how the interlacing properties of the normalised quotient Laplacian can be employed to measure such distance between G and $G_{\mathcal{P}}$. After we will present the bounds of such distance in the case of a weighted graph G and its quotient graph $G_{\mathcal{P}}$, which generalises, in terms of link weight function, relations with the Laplacian spectra and simplicial map inducing the quotient graph, the bound found in [GHL15].

Consider a graph G and its normalised Laplacian $\mathcal{L}$ which is known to have spectrum contained in the interval $[0, 2]$. We can construct a *probability density function* associated to G as follows

$$\mu_G(x) = \frac{1}{n}\sum_{i=1}^{n} \delta(x - \lambda_i),$$

where $\{\lambda_i\}_{i=1}^n$ is the spectrum of $\mathcal{L}$ and δ is the Dirac delta function. Given a probability density function μ on $[0, 2]$ we denote the associated *cumulative distribution function* by $F_\mu : [0, 2] \to [0, 1]$ defined as

$$F_\mu(x) = \frac{1}{n}\sum_{i=1}^{n} H(x - \lambda_i),$$

where H is the Heaviside step function. The associated *inverse cumulative distribution function* is $F_\mu^{-1} : [0, 1] \to [0, 2]$ defined as

$$F_\mu^{-1}(x) = \inf\{t \in \mathbb{R} \mid F_\mu(x) > x\}$$

Thanks to the properties of $\mathcal{L}$ such probability density function for any graph G belongs to the space of functions

$$\mathcal{P}\left([0, 2]\right) = \left\{ f : [0, 2] \to [0, 1] \middle| \int_0^2 f(x)dx = 1 \right\},$$

which is a metric space for the following class of distances.

Definition 6.14 ([GHL15])**.** Given two probability density functions $\mu, \nu \in \mathcal{P}([0, 2])$ we define their p^{th} *Wasserstein metric* as

$$d_p^W(\mu, \nu) = \left(\int_0^1 \left| F_\mu^{-1}(x) - F_\nu^{-1}(x) \right|^p dx \right)^{\frac{1}{p}},$$

where F_μ and F_ν are the cumulative distribution functions associated to μ and ν respectively.

A way to understand the Wasserstein metric is to consider the optimal transport problem. That is, given two distribution of mass μ, ν on a space X, in our case $[0, 2]$, we want to find the best way of transforming the distribution μ into ν with a motion of masses. Suppose that a *cost function* $c : [0, 2] \times [0, 2] \to [0, \infty)$ is given so that it measures how much effort we need to spend to move a mass from x to y, and let $c(x, y) = |x - y|$. The cost of the optimal plan will then coincide with the Wasserstein metric d_1^W [Vil08].

With this in mind we recall the definition of *spectral distance* between two graphs introduced in [GHL15]

Definition 6.15 ([GHL15])**.** Given two connected graphs G_1 and G_2 we define their *spectral distance* as

$$d(G_1, G_2) := d_1^W(\mu_1, \mu_2),$$

where μ_i is the probability density function of G_i, for $i = 1, 2$.

Note that calling such comparison between graphs 'distance' is effectively an abuse of notation as it could happen that two graphs G, G' have identical spectra and so identical probability functions associated. This means that there exists pair of graphs for which $d(G, G') = 0$ with $G \neq G'$, so the comparison d between graphs is more precisely a pseudo-metric. Unlike in Section 5.4 now we consider two objects with distance equal to 0 to be identical from the user point of view. We have then the following

Theorem 6.16 ([GHL15])**.** *Given $p \in [1, \infty)$ and $\mathcal{G}$ the collection of all (possibly infinite) weighted graphs, endowed with d_p^W, the Hausdorff distance induced from $\mathcal{P}([0, 2])$, is a pseudo-metric space and*

$$diam(\mathcal{G}, d_p^W) \leq 2^{1-\frac{1}{p}}.$$

This theorem tells us that for any two graphs G_1 and G_2 we have $d_1(G_1, G_2) \leq 1$. In [GHL15] it has been proven that if G_2 is obtained from G_1 via some elementary operation, such as collapse of a link, removal of a proper subgraph, or identification of nodes, there exists a sharper bound of their distance, [GHL15, Theorem 6.3]. This bound is obtained using the interlacing properties showed in [HJ13b] when each link has weight equal to 1 and the operation of node identification does not create self-loops in G_2. We generalise now this bound for the case when the graph G_2 is the quotient graph of G_1 with respect a partition $\mathcal{P}$.

Recalling that $\mathcal{L}_\mathcal{P}$ and $\mathcal{L}$ interlace we have that

$$0 \leq \lambda_i \leq \mu_i \leq \lambda_{i+n-k} \leq 2, \text{ for } i = 1, \ldots k$$

with respectively $\{\lambda_i\}_{i=1}^n$ spectra of $\mathcal{L}$ and $\{\mu_i\}_{i=1}^k$ spectra of $\mathcal{L}_\mathcal{P}$. This implies that $\mathcal{L}_\mathcal{P}$ induces a probability density function for $G_\mathcal{P}$ that belongs to $\mathcal{P}([0,2])$. All in all we can compute the Wasserstein distance between G and $G_\mathcal{P}$. It is important to notice that given G and G', quotient graph of G, their distance can never be 0, that is, their probability density functions can not be equal. Let n be the number of nodes in G and k that of G', then if the probability functions $\mu = \mu_G$ and $\nu = \mu_{G'}$ are equal it would mean that their eigenvalues are equal, so the interlacing is tight, see Theorem 6.9. Moreover, as each Heaviside function in F_μ is weighted $\frac{1}{n}$ and $\frac{1}{k}$ in ν, we have that the multiplicity of each eigenvalue in G has to be $\frac{n}{k}$ times its multiplicity in G'. We know that the connectivity of G' is the same of G, thanks to the definition of quotient graph, that is, 0 has same multiplicity in G and G'. In conclusion we have $\frac{n}{k} = 1$, so $n = k$. If G' has the same number of nodes of G then it means that the partition inducing G' is the singleton partition and $G' = G$.

To our knowledge, the results that follow are novel and not contained either in [HJ13a], where is considered the unweighted normalised Laplacian and nor in [GHL15] (see [GHL15, Remark 6.1]).

Theorem 6.17. *Given a graph G and a partition $\mathcal{P} = \{S_1, \dots, S_k\}$ of the node set of G, then*

$$\begin{cases} d(G, G_\mathcal{P}) \leq 2 - \frac{2}{k}\sum_{i=1}^k \lambda_i - \frac{h(\mathcal{P})}{k} \\ d(G, G_\mathcal{P}) \leq \frac{2}{k}\sum_{i=1}^k \lambda_{n-k+i} - 2 - \frac{h(\mathcal{P})}{k}, \end{cases} \tag{6.5}$$

where $h(\mathcal{P}) = \sum_{i=1}^k \frac{cut(S_i, S_i)}{\mathrm{Vol}(S_i)}$.

Proof. Following the proof of [GHL15, Theorem 6.3] we consider a transportation plan $\xi : \{\lambda_i\}_{i=1}^n \times \{\mu_j\}_{j=1}^k \to \mathbb{R}$ defined as

$$\xi(\lambda_i, \mu_j) = \begin{cases} \frac{1}{n}, & \text{if } j = i \leq k \\ \left(\frac{1}{k} - \frac{1}{n}\right)\frac{1}{n-k}, & \text{if } i > k, \\ 0, & \text{otherwise.} \end{cases}$$

We have then that

$$\begin{aligned}
d(G, G_{\mathcal{P}}) &\leq \sum_{j=1}^{k}\sum_{i=1}^{n} \xi(\lambda_i, \mu_j)|\lambda_i - \mu_j| \\
&= \frac{1}{n}\sum_{j=1}^{k}(\mu_j - \lambda_j) + \left(\frac{1}{k} - \frac{1}{n}\right)\frac{1}{n-k}\sum_{j=1}^{k}\sum_{i=k+1}^{n}|\lambda_i - \mu_j| \\
&= \frac{1}{n}\sum_{j=1}^{k}(\mu_j - \lambda_j) + \frac{1}{nk}\sum_{j=1}^{k}\sum_{i=k+1}^{n}|\lambda_i - \lambda_j + \lambda_j - \mu_j| \\
&\leq \frac{1}{n}\sum_{j=1}^{k}(\mu_j - \lambda_j) + \frac{1}{nk}\sum_{j=1}^{k}\sum_{i=k+1}^{n}(\lambda_i - \lambda_j) + (\mu_j - \lambda_j) \\
&= \frac{1}{n}\sum_{j=1}^{k}(\mu_j - \lambda_j) + \frac{1}{n}\sum_{i=k+1}^{n}\lambda_i + \frac{n-k}{nk}\sum_{j=1}^{k}\mu_j - \frac{2(n-k)}{nk}\sum_{j=1}^{k}\lambda_j \\
&= 1 + \left(\frac{1}{n} + \frac{n-k}{nk}\right)\sum_{j=1}^{k}\mu_j - \left(\frac{1}{n} + \frac{2(n-k)}{nk} + \frac{1}{n}\right)\sum_{j=1}^{k}\lambda_j \\
&= 1 + \frac{1}{k}\left(\sum_{j=1}^{k}\mu_j\right) - \frac{2}{k}\left(\sum_{j=1}^{k}\lambda_j\right).
\end{aligned}$$

We used that $n = \sum_{i=1}^{n}\lambda_i$. Thanks to

$$\sum_{j=1}^{k}\mu_j = \mathrm{Tr}(\mathcal{L}_{\mathcal{P}}) = \sum_{j=1}^{k}\left(1 - \frac{cut(S_j, S_j)}{\mathrm{Vol}(S_j)}\right) = k - h(\mathcal{P}),$$

we can substitute and obtain the first bound.

The second bound can be obtained with a similar procedure. We consider a transportation plan $\xi : \{\lambda_i\}_{i=1}^{n} \times \{\mu_j\}_{j=1}^{k} \to \mathbb{R}$ defined as

$$\xi(\lambda_i, \mu_j) = \begin{cases} \frac{1}{n}, & \text{if } i = j + n - k, j \leq k \\ \frac{1}{nk}, & \text{if } i \leq n - k, \\ 0, & \text{otherwise.} \end{cases}$$

We have then that

$$\begin{aligned}
d(G,G_{\mathcal{P}}) &\leq \sum_{j=1}^{k}\sum_{i=1}^{n}\xi(\lambda_i,\mu_j)|\lambda_i-\mu_j| \\
&= \frac{1}{n}\sum_{j=1}^{k}(\lambda_{n-k+j}-\mu_j)+\frac{1}{nk}\sum_{j=1}^{k}\sum_{i=1}^{n-k}|\lambda_i-\mu_j| \\
&= \frac{1}{n}\sum_{j=1}^{k}(\lambda_{n-k+j}-\mu_j)+\frac{1}{nk}\sum_{j=1}^{k}\sum_{i=1}^{n-k}|\lambda_i-\lambda_{n-k+j}+\lambda_{n-k+j}-\mu_j| \\
&\leq \frac{1}{n}\sum_{j=1}^{k}(\lambda_{n-k+j}-\mu_j)+\frac{1}{nk}\sum_{j=1}^{k}\sum_{i=1}^{n-k}(\lambda_{n-k+j}-\lambda_i)+(\lambda_{n-k+j}-\mu_j) \\
&= \left(\frac{1}{n}+\frac{2(n-k)}{nk}+\frac{1}{n}\right)\sum_{j=1}^{k}\lambda_{n-k+j}-1-\left(\frac{1}{n}+\frac{(n-k)}{nk}\right)\sum_{j=1}^{k}\mu_j \\
&= \frac{2}{k}\sum_{j=1}^{k}\lambda_{n-k+j}-1-\frac{1}{k}\sum_{j=1}^{k}\mu_j.
\end{aligned}$$

We used again that $n=\sum_{i=1}^{n}\lambda_i$ and as before

$$\sum_{j=1}^{k}\mu_j=\mathrm{Tr}(\mathcal{L}_{\mathcal{P}})=\sum_{j=1}^{k}\left(1-\frac{cut(S_j,S_j)}{\mathrm{Vol}(S_j)}\right)=k-h(\mathcal{P}),$$

substituting we obtain the second bound. □

Note that if we take the average of these two bounds the result will be a bound for the distance, but it will not involve $\mathcal{P}$ being

$$d(G,G_{\mathcal{P}})\leq\frac{1}{k}\sum_{i=1}^{k}(\lambda_{n-k+j}-\lambda_j).$$

Moreover, as it is the average of two bounds it will be sharper than one but also more loose than the other.

The bounds proved in Theorem 6.17 for $d(G,G_{\mathcal{P}})$ are minimal over partitions with k clusters when $\mathcal{P}$ has maximum $h(\mathcal{P})$ or conversely when $k-h(\mathcal{P})$ reaches its minimum. Note that if $k=2$ then

$$2-h(\mathcal{P})=cut(S_1,S_2)\left(\frac{1}{\mathrm{Vol}(S_1)}+\frac{1}{\mathrm{Vol}(S_2)}\right),$$

which is equal to the weighted version of the Fiedler constant [Fie95].

Note that $h(\mathcal{P})\leq k$ and the equality holds when $k=1$. If $h(\mathcal{P})=k$ then each cluster is disconnected from the others so G has exactly k connected components, that is $k=1$. In such case we know, see [GHL15], that $d(G,G_{\mathcal{P}})=1$, that is the largest possible distance obtainable in $\mathcal{G}$, Theorem 6.16. This is sensible, as the graph $G_{\mathcal{P}}$ will consist of only one node, as $k = 1$, the entire structure and properties of G will become meaningless in $G_{\mathcal{P}}$. On the other hand we already excluded that $h(\mathcal{P}) = 0$ as it would imply that each cluster is totally disconnected.

In general Equation 6.5 is telling us that we can approximate the spectral distance using the k smallest (or largest) eigenvalues of L instead of calculating both spectra. Using the (normalised) quotient Laplacian presented here we can iterate a coarse-grained method based on Laplacian spectrum or instead, given a hierarchical clustering method, compute such distance at each step, using the approximation presented in Equation 6.5, to determine a merge stopping condition. In the next and last section we will extend these results to overlapping clusterings.

6.4 Generalisation to overlapping partitions

Given a simplicial map φ between two graphs we always obtain a non-overlapping partition induced by the inverse image of φ, conversely an overlapping partition can not be realised through a simplicial map. This implies that the approach used in [HJ13b, GHL15] fails when we try to apply it to overlapping partitions. To overcome such obstruction we present here a generalisation of Theorem 6.17 obtained using the notion of *lift* of a graph associated to an overlapping partition. Moreover, we will prove that the bounds in Theorem 6.17 are satisfied when the number of clusters in $\mathcal{P}$ is small enough.

To adapt the definition of quotient matrix to overlapping partitions we will assume that together with $\mathcal{P}$ a function of belongingness is given, $p : \mathcal{P} \times V \to \mathbb{R}$, so that for each node v_j and cluster S_i we have the probability, or degree, of belongingness of one to the other, $p(S_i, v_j)$. We also impose that for each node v we have $\sum_i p(S_i, v) = 1$. The quotient matrix P is so defined

$$P(i,j) = \begin{cases} p(S_i, v_j) & v_j \in S_i, \\ 0 & \text{otherwise.} \end{cases}$$

Using such definition we can prove the following lemma.

Lemma 6.18. *Let G be a graph and $\mathcal{P} = \{S_1, \ldots, S_k\}$ an overlapping partition of its node set. If for each cluster S_i there exists a node v_j such that v_j does not belong to any other cluster S_q with $q \neq i$ then the matrix $H = PP^T$ is invertible.*

Proof. We have that $wHw^T = \|wP\|^2$ for $w \in \mathbb{R}^k$ and with $\|\cdot\|$ the euclidean norm in $\mathbb{R}^n$. Suppose H is singular then there exists a $w \neq 0$ such that $\|wP\|^2 = 0$. Then

$$(wP)_t = \sum_q w_q P(q,t) = 0, \qquad \forall t = 1, \ldots, n.$$

By hypothesis for any $i = 1, \ldots, k$ there exists $j = j(i)$ such that $P(i,j) = 1$ and $P(q,j) = 0$ for $q \neq i$, as v_j belongs only to S_i. Then

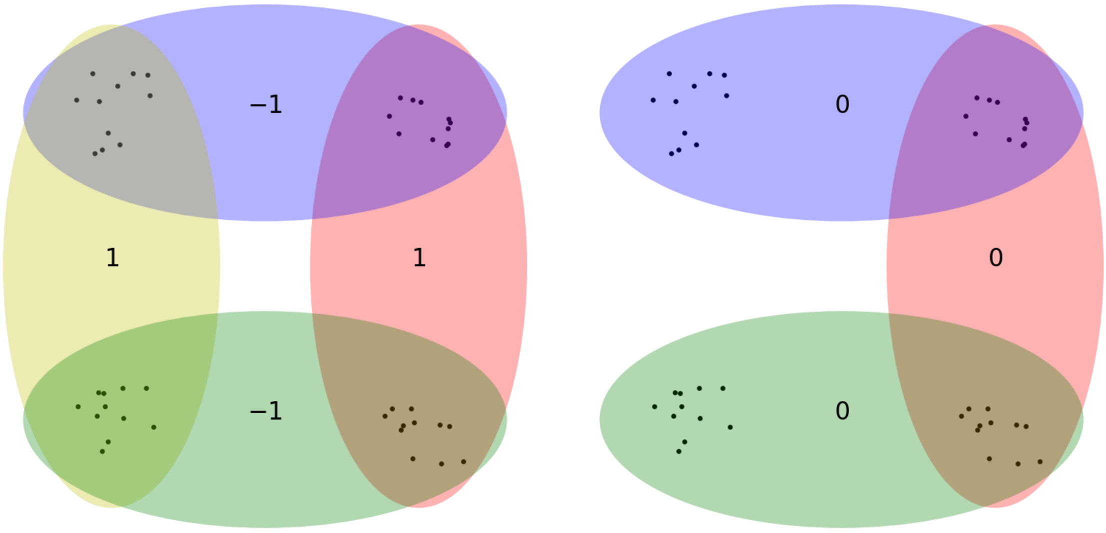
$$(wP)_j = w_i = 0,$$

so $w = 0$, which leads to contradiction. □

Note that the hypothesis in Lemma 6.18 are always satisfied if $\mathcal{P}$ is not overlapping. In the case of overlapping partition we can see the condition in Lemma 6.18 as the condition that for each $S_i \in \mathcal{P}$ we can not express it only in terms of the remain clusters, that is

$$S_i \neq \bigcup_{j \neq i} S_i \cap S_j. \tag{6.6}$$

From now on we will assume that this does not happen, so $H = PP^T$ is always invertible. As the conditions in Lemma 6.18 are sufficient and not necessary, it is possible that H is invertible and there exists $S_i \in \mathcal{P}$ such that $S_i = \bigcup_{j \neq i} S_i \cap S_j$, as shown in Figure 6.1.



(A) There exists a vector $w \neq 0$ in Ker$H$.

(B) The only vector in Ker$H$ is 0.

FIGURE 6.1: We have two overlapping clusterings with clusters denoted by the ellipses, with each node belonging with same probability to its clusters. The vector $w \in \text{Ker}(H)$ is displayed with entry w_i over the ellipse relative to cluster S_i.

If $\mathcal{P}$ is overlapping we have that H can not be diagonal as

$$H(i,j) = \sum_{k \in S_i \cap S_j} P(k,i)P(k,j) = |S_i \cap S_j|.$$

In particular the weight function we obtain on $G_{\mathcal{P}}$ gives to a link (S_i, S_j) weight equal to

$$w_{\mathcal{P}}(S_i, S_j) = \sum_{p \in S_i} \sum_{q \in S_j} P(i,p)w(p,q)P(j,q), \tag{6.7}$$

As before we need to find interlacing properties between the (normalised) Laplacian and the (normalised) quotient one. To obtain those with P can be computationally challenging, as we need to first amend P so that $H = PP^T = \mathrm{Id}_k$. For this purpose we need the diagonalisation of H, that is, $H = Q\Lambda Q^T$, with Λ diagonal and Q orthonormal. We can amend P then as

$$P_{norm} = \Lambda^{-\frac{1}{2}} Q^T P.$$

The quotient Laplacian of $G_{\mathcal{P}}$ can be defined then as

$$L_{\mathcal{P}} = P_{norm} L P_{norm}^T,$$

and similarly the normalised one. Also if these will interlace with the (normalised) Laplacian of G finding Q and Λ can be expensive in terms of computation. To avoid it we present here a different approach using the notion of *lift* of a graph G (see [GT01]) which brings similar bounds to Equation 6.5.

Definition 6.19 ([GT01])**.** Let $\mathcal{I} = \{p_i\}_{i=1}^k$ be a set of real positive numbers such that $\sum_{i=1}^k p_i = 1$ and consider a weighted graph $G = (V, E;\ w)$ where $V = \{v_1, \dots, v_n\}$ with $k < n$ and $v = v_n$. The *weighted lift* of G with respect to v and $\mathcal{I}$ is the graph $G' = (V', E'; w')$ defined as

$$\begin{cases} V' &= V \cup \{v_{n+1}, \dots, v_{n+k-1}\} \\ E' &= \{(v_i, v_j) \in E \mid i, j < n\} \cup \{(v_i, v_j) \mid n \leq j, (v_i, v_n) \in E\}. \end{cases} \tag{6.8}$$

The weight function w' is defined as

$$w'(v_s, v_t) = \begin{cases} 0, & \text{if } s \geq, t \geq n \\ p_i w(v_n, v_t), & \text{if } s = n+i-1, t < n \\ w(v_s, v_t), & \text{otherwise .} \end{cases}$$

The definition we give here of lift of a graph, is similar to the one given in [GT01] with the exception that in our case we are in the weighted case and so we require some additional condition on the link weight function. In particular the graph G' in Definition 6.19 is a *covering* of G in the sense of [GT01] when regarded as unweighted graphs. In

G' the node v_n is duplicated k times, as its connections, and with $\mathcal{I}$ we ensure that each link duplicate weights proportionally to the original one. Note that given a node v in an overlapping partition we can always consider $\mathcal{I} = \{P(v, S_i)\}_{i=1}^{k}$ and construct the associated lift. More we can iterate such construction for each node obtaining a graph where each node is duplicated exactly as many times it belongs to a cluster in $\mathcal{P}$. Consider $\mathcal{P}$ an overlapping partition and its quotient matrix P, we construct the unweighted graph $\widehat{G}$ such that

$$\begin{aligned} \widehat{V} &= \{v_{i,j} \mid v_j \in S_i\} \\ \widehat{E} &= \{(v_{p,i}, v_{q,j}) \mid v_i \in S_p, v_j \in S_q, (v_i, v_j) \in E\}. \end{aligned} \tag{6.9}$$

It is easy to see that $\widehat{G}$ is a lift of G in the sense of [GT01]. We can define in $\widehat{G}$ two non-overlapping partitions, $\mathcal{P}_n, \mathcal{P}_k$, such that $\mathcal{P}_n$ (resp. $\mathcal{P}_k$) groups together nodes in $\widehat{G}$ corresponding to the same node in G (resp. the same cluster in $\mathcal{P}$). These clusters can be written as follows

$$\begin{aligned} N_j = \{v_{i,k} \mid k = j\} &\quad \text{for} \quad \mathcal{P}_n, \\ C_i = \ \{v_{k,j} \mid k = i\} &\quad \text{for} \quad \mathcal{P}_k. \end{aligned}$$

The rest of this section is devoted to show that $\widehat{G}_{\mathcal{P}_n} \equiv G$ and $\widehat{G}_{\mathcal{P}_k} \equiv G_{\mathcal{P}}$, both as weighted graphs.

$$\begin{tikzcd} & \widehat{G} \arrow[dl] \arrow[dr] & \\ G_{\mathcal{P}} & & G \arrow[ll, dashed] \end{tikzcd}$$

FIGURE 6.2: The graphs G and $G_{\mathcal{P}}$ are both quotient graphs of $\widehat{G}$.

Thanks to this we will use Theorem 6.17 to bound $d(G, G_{\mathcal{P}})$ with $d(\widehat{G}, G) + d(\widehat{G}, G_{\mathcal{P}})$ and generalise Theorem 6.17 for overlapping partitions. As first step in this direction we will now show the spectral properties of a lift of a graph G with respect to a node v and a set $\mathcal{I}$.

Lemma 6.20. *Let G' be a weighted lift of a graph G as in Definition 6.19, consider a partition $\mathcal{P} = \{S_i\}_{i=1}^{n}$ of G' such that $S_n = \{n + i - 1 | i = 1, \dots, k\}$ when $i = n$ and $S_i = \{i\}$ when $i \neq n$. Then*

$$\mathcal{L} = R\mathcal{L}'R^T,$$

where $\mathcal{L}'$ is the normalised Laplacian of G' and R is the normalised quotient matrix associated to $\mathcal{P}$ as in Theorem 6.12.

Proof. Consider two clusters S_i, S_j in $\mathcal{P}$ with $i \neq j$, then we have that if $i, j \neq n$

$$cut(S_i, S_j) = \sum_{p \in S_i} \sum_{q \in S_j} w'(p, q) = w'(v_i, v_j) = w(v_i, v_j),$$

thanks to the definition of w'. If $i = n$ then

$$cut(S_n, S_j) = \sum_{p \in S_n} \sum_{q \in S_j} w'(p, q) = \sum_{i=1}^{k} p_i w'(v_n, v_j) = w(v_i, v_j),$$

as $\sum_{i=1}^{k} p_i = 1$. Note that $cut(S_i, S_i) = 0$ for $i < n$ as S_i consists of only one node. If $i = n$ we have as well $cut(S_n, S_n) = 0$ as each pair in S_n is disconnected in G'. Consider now the weighted degree matrix D' in G', if $i < n$ then

$$\begin{aligned} D'(i,i) &= \textstyle\sum_{j=1}^{n+k-1} w'(i,j) \\ &= \textstyle\sum_{j=1}^{n-1} w(i,j) + \sum_{j=1}^{k} p_j w(i,n) \\ &= \textstyle\sum_{j=1}^{n-1} w(i,j) + w(i,n) = D(i,i). \end{aligned}$$

Consider now $i = n + t - 1$ for $t = 1, \ldots, k$ then

$$\begin{aligned} D'(i,i) &= \textstyle\sum_{j=1}^{n+k-1} w'(i,j) \\ &= \textstyle\sum_{j=1}^{n-1} p_t w(n,j) = p_t D(i,i). \end{aligned}$$

Thanks to this we have that $\mathrm{Vol}(S_i, S_i) = D(i,i)$ for $i = 1, \ldots, n$. Then from Definition 6.13 we have that the normalised quotient Laplacian obtained with $\mathcal{P}$ coincides with $\mathcal{L}$. □

If we consider a weighted lift of a graph G and then its quotient with respect the same node, that is as in Lemma 6.20, the Laplacian of G coincides with the quotient Laplacian so obtained. Thanks to Section 6.2 we have that the normalised Laplacian of G interlaces with the normalised Laplacian of G'. As G' is a lift of G we have that the spectrum of G' is the same of G with 1 added $k-1$ times as the interlacing is tight.

Theorem 6.21. *Given a weighted graph G and its lifted graph G' defined as in Definition 6.19. If Λ, resp. Λ', is the spectrum of $\mathcal{L}$, resp. $\mathcal{L}'$, counted with multiplicity then $\Lambda \subseteq \Lambda'$ and $\Lambda' \setminus \Lambda = \{1\}_{i=1}^{k-1}$.*

Proof. Consider $\mathcal{E}$ the linearly independent set in $\mathbb{R}^n$ formed by the eigenvectors of $\mathcal{L}$ and R the normalised quotient matrix relative to G. Pick $e \in \mathcal{E}$, with eigenvalue λ and define $f = R^T e$ then

$$f_t = \begin{cases} \sqrt{p_j} e_n, & \text{if } t = n + j - 1, \\ e_t, & \text{otherwise,} \end{cases}$$

Consider first the t-entry of $\mathcal{L}'f$ for $t < n$ then

$$\begin{aligned}(\mathcal{L}'f)_t &= \textstyle\sum_{j=1}^{n+k-1} \mathcal{L}'(t,j)f_j \\ &= \textstyle\sum_{j=1}^{n-1} \mathcal{L}(t,j)e_j + \sum_{j=1}^{k} \sqrt{p_j}\mathcal{L}(t,n)\sqrt{p_j}e_n \\ &= \textstyle\sum_{j=1}^{n-1} \mathcal{L}(t,j)e_j + \left(\sum_{j=1}^{k} p_j\right) \mathcal{L}(t,n)e_n \\ &= \textstyle\sum_{j=1}^{n} \mathcal{L}(t,j)e_j = \lambda e_t = \lambda f_t.\end{aligned}$$

Consider now $t = n + i - 1$ then, using that $\mathcal{L}'(t, n + j - 1) = 0$ for $j \neq i$, we have

$$\begin{aligned}(\mathcal{L}'f)_t &= \textstyle\sum_{j=1}^{n+k-1} \mathcal{L}'(t,j)f_j \\ &= \textstyle\sum_{j=1}^{n-1} \mathcal{L}'(t,j)f_j + \mathcal{L}_1(t,t)f_t \\ &= p_i \textstyle\sum_{j=1}^{n-1} \mathcal{L}(n,j)e_j + p_i\mathcal{L}(n,n)e_n \\ &= p_i \left(\textstyle\sum_{j=1}^{n} \mathcal{L}(n,j)e_j\right) = \lambda\left(p_ie_n\right) = \lambda f_t.\end{aligned}$$

In conclusion $\mathcal{L}'f = \lambda f$ for any f of the form R^Te with $e \in \mathcal{E}$. Let $\mathcal{F}$ be the set $\{R^Te_i \mid e_i \in \mathcal{E}\}$ and suppose that it is a linearly dependent set, that is, there exists a linear combination such that

$$\sum_{i=1}^{n} a_i\left(R^Te_i\right) = 0.$$

Multiplying by R both sides we have that

$$\sum_{i=1}^{n} a_ie_i = 0.$$

So $a_i = 0$ for all $i = 1, \ldots, n$ as $\mathcal{E}$ is a linearly independent set, so also $\mathcal{F}$ is. This implies that $\Lambda \subseteq \Lambda'$.

To prove that $\Lambda' \backslash \Lambda = \{1\}_1^{k-1}$ we will construct a linearly independent set of eigenvectors $\mathcal{F}_1$ such that $\mathcal{F} \cup \mathcal{F}_1$ is a linearly independent set and if $f \in \mathcal{F}_1$ then $\mathcal{L}'f = f$. Let $\mathcal{F}_1 = \{f_i\}_{i=2}^{k}$ such that each t-entry of $f = f_i \in \mathcal{F}_1$ is defined as

$$f_t = \begin{cases} \sqrt{p_i}, & \text{if } t = n, \\ -\sqrt{p_1}, & \text{if } t = n+i-1, \\ 0, & \text{otherwise.} \end{cases}$$

First we prove that they are all 1-eigenvectors of $\mathcal{L}'$. For each $f \in \mathcal{F}_1$ the t-entry of $\mathcal{L}'f$ is

$$(\mathcal{L}'f)_t = \sqrt{p_i}\mathcal{L}'(t,n) - \sqrt{p_1}\mathcal{L}'(t,j),\ j = n+i-1.$$

- If $t < n$ then $(\mathcal{L}'f)_t = \sqrt{p_i}\sqrt{p_1}\mathcal{L}(t,n) - \sqrt{p_1}\sqrt{p_i}\mathcal{L}(t,n) = 0$.
- If $t = j$ then $(\mathcal{L}'f)_t = 0 - \sqrt{p_1} = f_j$.
- if $t = n$ then $(\mathcal{L}'f)_t = \sqrt{p_i} - 0 = f_n$.

- If $t = n + s - 1$, with $s \neq 1, i$ then $(\mathcal{L}'f)_t = 0 - 0 = 0$.

In conclusion $\mathcal{F}_1$ is formed by 1-eigenvectors of $\mathcal{L}'$. We have that clearly $\mathcal{F}_1$ is linearly independent set as otherwise there would exists a linear combination

$$\sum_{i=2}^{k} a_i f_i = 0.$$

Given t, the t-entry of f_i is non-zero only if $i = t-n+1$, and so $a_i = 0$ for all $i = 2, \dots, k$.

To conclude we need to prove that $\mathcal{F} \cup \mathcal{F}_1$ is a linearly independent set. We know that each $f \in \mathcal{F}$ is such that $f \notin \mathrm{Ker}(R)$ otherwise $Rf = 0 \in \mathcal{E}$ and so $\mathcal{E}$ is not a linearly independent set. On the other side for each $f \in \mathcal{F}_1$ we have that $Rf = 0$. Then $\mathcal{F} \perp \mathcal{F}_1$ so it suffices that both of them are linearly independent sets. □

We have then that not only G and G' share eigenvalues but also we know which are the not shared ones. Recall that given an overlapping partition $\mathcal{P}$ we can construct $\widehat{G}$ as a lift of G, iteratively lifting each node with respect to its row-vector in the quotient matrix P. Using the definition of weighted lift of a graph, the link weight function we will have on $\widehat{G}$ is

$$\widehat{w}(v_{i,s}, v_{j,t}) = P(i,s)w(s,t)P(j,t).$$

Using Theorem 6.21 we can prove that the spectrum of $\mathcal{L}$, normalised Laplacian of G, is contained in $\widehat{\mathcal{L}}$, normalised Laplacian of $\widehat{G}$, and the eigenvalues in $\widehat{\mathcal{L}} \setminus \mathcal{L}$ are all equal to 1.

Theorem 6.22. *Given a weighted graph G and an overlapping partition $\mathcal{P}$, let $\widehat{G}$ be the weighted graph as in Equation 6.9 with weight $\widehat{w}$. The normalised quotient Laplacian with respect to $\mathcal{P}_n$ is equal to the normalised graph Laplacian of G, moreover if Λ, resp. $\widehat{\Lambda}$, is the set of eigenvalues of $\mathcal{L}$, resp. $\mathcal{L}(\widehat{G})$, counted with multiplicity then $\Lambda \subseteq \widehat{\Lambda}$ and $\widehat{\Lambda} \setminus \Lambda = \{1\}_{i=1}^{d}$, where d is equal to the difference of nodes between $\widehat{G}$ and G.*

Proof. By Theorem 6.21 if we consider $\mathcal{I}_1 = \{p_j = P(j,1) \mid P(j,1) \neq 0\}$ and G_1 as the graph obtained duplicating only v_1 then we have that its normalised Laplacian $\mathcal{L}_1$ has the same eigenvalues of $\mathcal{L}$ plus a set of $k_1 - 1$ eigenvalues all equal to 1, with k_1 the number of clusters to which v_1 belongs. We can construct G_2 from G_1 duplicating now v_2, using $\mathcal{I}_2 = \{p_j = P(j,2) \mid P(j,2) \neq 0\}$. Again thanks to Theorem 6.21, we obtain $\Lambda_2 = \Lambda_1 \cup \{1\}_{i=1}^{k_2-1} = \Lambda \cup \{1\}_{i=1}^{k_1+k_2-2}$ with k_2 the number of clusters to which v_2 belongs. By induction, iterating this procedure for all nodes in G, we obtain as final result exactly $\widehat{G}$. The weight function w_n coincides with $\widehat{w}$ as at each iteration we multiplied the link weights by $P(j,i)$ for $i = 1, \dots, n$. In conclusion as $t = \sum_{i=1}^{n} k_i$, with k_i the number of clusters to which v_i belongs, is exactly the number of nodes in $\widehat{G}$ we have

$$\widehat{\Lambda} = \Lambda_n = \Lambda \cup \{1\}_{i=1}^{t-n},$$

as wanted. □

In [SG18] the case of quotient graph sharing eigenvalues is considered in the light of symmetries on a graph G. When we consider the automorphisms of a graph, that is, the maps from V to V which induce a isomorphism of G, then we obtain that those are a subgroup of the symmetric group of n elements, $\mathcal{S}_n$. The orbits of such subgroup form a non-overlapping partition $\mathcal{P}$ and the quotient graph obtained is called *regular quotient*. This graph has the property that each eigenvalue of its Laplacian is also eigenvalue of $\mathcal{L}$, possibly with less multiplicity. Theorem 6.22 does not hold in such general case as the clusters are not always inter-disconnected so the eigenvalues of $\mathcal{L}$ which are not in the spectrum of the quotient Laplacian can be different from 1.

Because $\mathcal{P}_n$ is a non-overlapping partition we can use Theorem 6.17 to find a bound for $d(G, \widehat{G})$ however thanks to Theorem 6.22 we can derive a better one.

Theorem 6.23. *Given a weighted graph G and an overlapping partition $\mathcal{P}$, let $\widehat{G}$ be the weighted graph obtained with $\mathcal{P}$ as in Equation 6.9 then*

$$d(\widehat{G}, G) \leq \alpha(G) \left(\frac{1}{n} - \frac{1}{n+d} \right),$$

where $\alpha(G)$ depends only on $\mathcal{L}$ and d is the difference between the numbers of nodes of $\widehat{G}$ and G.

Proof. To prove this bound we will use the same procedure of optimal transport plan used in Theorem 6.17. Define $\xi : \widehat{\Lambda} \times \Lambda \to \mathbb{R}_{\geq 0}$, transport plan, as follows

$$\xi(x_i, y_j) = \begin{cases} \frac{1}{n+d}, & \text{if } x_i \in \Lambda, x_i = y_j \\ \frac{1}{(n+d)n}, & \text{if } x_i \notin \Lambda \\ 0, & \text{otherwise.} \end{cases}$$

We have then that

$$d(\widehat{G}, G) \leq \sum_{x_i \in \widehat{\Lambda}} \sum_{y_j \in \Lambda} |x_i - y_j| \, \xi(x_i, y_j).$$

The only non-zero terms in the sum are the ones when $x_i \notin \Lambda$, that is, when $x_i = 1$ then

$$d(\widehat{G}, G) \leq \frac{d}{n(n+d)} \sum_{j=1}^{n} |1 - \lambda_j| = \alpha(G) \left(\frac{1}{n} - \frac{1}{n+d} \right),$$

where $\alpha(G) = \sum_{j=1}^{n} |1 - \lambda_j|$. □

We can see $\alpha(G)$ as a measure of how much the eigenvalues of $\mathcal{L}$ are away from 1. The bound just introduced does not depend on the entries of the quotient matrix P

but only on the number of duplicates it induces. If $d = 0$, or equivalently when $\mathcal{P}$ is non-overlapping, the bound is equal to 0 and $\widehat{G} \equiv G$.

We will show now the bounds with respect to $\mathcal{P}_k$. Thanks to the definition of $\widehat{G}$ and $\widehat{w}$ we have that the quotient graph obtained from $\widehat{G}$ with $\mathcal{P}_k$ coincides with $G_{\mathcal{P}}$. In fact consider a link (C_i, C_j) then we have that its weight, following Theorem 6.12, is

$$cut(C_i, C_j) = \sum_{p \in C_i} \sum_{q \in C_i} \widehat{w}(p, q) = \sum_{v_s \in C_i} \sum_{v_t \in C_i} P(i, s) w(s, t) P(j, t),$$

as we obtained in Equation 6.7. As $d = d_1^W$ is a (pseudo) metric in $\mathcal{G}$ we have

$$d(G, G_{\mathcal{P}}) \leq d(\widehat{G}, G) + d(\widehat{G}, G_{\mathcal{P}}).$$

Note again that both $\widehat{G}$, $G_{\mathcal{P}}$ and G are connected graphs, so none of the distances above can be 0 when $\mathcal{P}$ is overlapping. Using Theorem 6.23 and Theorem 6.17 we can then bound such distance as

$$d(G, G_{\mathcal{P}}) \leq \alpha(G) \left(\frac{1}{n} - \frac{1}{n+d} \right) + B,$$

where B is one of the bounds found in Equation 6.5. Note that now the eigenvalues used in Equation 6.5 are of $\widehat{G}$ and so we could have some eigenvalue equal to 1 introduced in the expression of B. This possibility does not happen if k is small enough.

Theorem 6.24. *Given a graph G and an overlapping partition $\mathcal{P} = \{S_i\}_{i=1}^k$. If $\lambda_k \leq 1$ or $\lambda_{n-k} \geq 1$ then*

$$\begin{aligned} d(G, G_{\mathcal{P}}) &\leq 2 - \frac{2}{k} \sum_{i=1}^{k} \lambda_i - \frac{h(\mathcal{P}_k)}{k}, \\ d(G, G_{\mathcal{P}}) &\leq \frac{2}{k} \sum_{i=1}^{k} \lambda_{n-i+1} - 2 - \frac{h(\mathcal{P}_k)}{k}, \end{aligned}$$

with $h(\mathcal{P}_k) = \sum_{i=1}^k \frac{cut(C_i, C_i)}{\mathrm{Vol}(C_i)}$, where

$$\begin{aligned} cut(C_i, C_i) &= \textstyle\sum_{s \in C_i} \sum_{t \in C_i} P(s, i) w(s, t) P(t, i), \\ \mathrm{Vol}(C_i) &= \textstyle\sum_{s \in C_i} P(s, i) D(s, s), \end{aligned}$$

and D is the weighted degree matrix of G.

Proof. Consider p such that $\lambda_p \leq 1$ and $\lambda_{p+1} > 1$, then we have for $i = 1, \ldots, p$

$$\begin{cases} \lambda'_i = \lambda_i \\ \lambda'_{n-p+d+i} = \lambda_{n-p+i} \end{cases}$$

If $\lambda_k \leq 1$ then the first k eigenvalues of $\mathcal{L}$ are less or equal than 1, in particular $k \leq p$ and for $i = 1, \ldots, k$

$$\lambda_i = \lambda'_i \leq \mu_i \leq \lambda'_{i+n+d-k} = \lambda_{i+n-k}.$$

That is $\mathcal{L}_{\mathcal{P}}$ and $\mathcal{L}$ interlace so the bounds proved in Theorem 6.17 hold. In this case however we will have $\sum_{i=1}^{k} \mu_i = h(\mathcal{P}_k)$. An similar argument holds when $\lambda_{n-k} \geq 1$. □

If and only if $n < 2k$ we have that Theorem 6.24 does not hold, or equivalently when $\lambda_{n-k} < 1 < \lambda_k$. In fact if $2k \leq n$ then $\lambda_k \leq \lambda_{n-k}$ and so when $1 < \lambda_k$ we get $1 < \lambda_{n-k}$ whilst when $\lambda_{n-k} < 1$ we get $\lambda_k < 1$, so Theorem 6.24 holds.

The work presented can be useful both to measure the quality, in terms of graph structure change, of a partition, overlapping or not, but as well to study the effects of a Morse function on a graph G. We know that given f, Morse function on G, its flow Φ induces a simplicial map φ on the nodes, where $\varphi = \Phi^N$. As Φ is a map on the entire chain complex $C_*(G)$ we can also examine φ as a function on the links of G. On $C_1(G)$ the map φ cannot be considered just as a map of the links, because it could map a link e to a linear combination of links. More precisely on $C_1(K)$ we have $\Phi = \mathrm{Id} + \mathcal{V}\partial$. It is then easy to see from Lemma 2.18 that if e is non-critical then $\varphi(e) = 0$. If instead e is critical then we know, again from Lemma 2.18 that $< \Phi(e), e >= 1$ and as $\Phi(e) - e \in \mathrm{Im}(\mathcal{V})$ we have that $\varphi(e) - e$ is a linear combination of non-critical links. All in all we can define the following partition of links $\mathcal{P} = \{S_e\}_e$ such that

$$S_e = \{f \in E \mid < \varphi(e), f > \neq 0\}.$$

From the consideration of before we have that if e is non-critical then $S_e = \emptyset$ as $\varphi(e) = 0$. Consider Figure 6.3 below then we have

$$\begin{cases} \varphi(e_{1,6}) &= e_{1,6} \\ \varphi(e_{3,7}) &= e_{3,7} + e_{1,3} - e_{6,7} \\ \varphi(e_{4,5}) &= e_{4,5} + e_{2,4} - e_{3,5} + e_{1,2} - e_{1,3}. \end{cases}$$

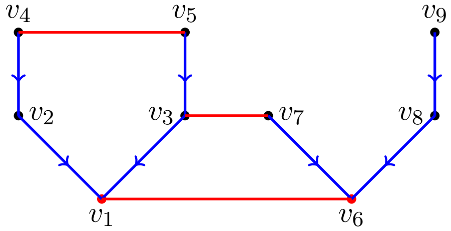


FIGURE 6.3: A Morse flow on a graph, with critical links and nodes are depicted in red, whilst non-critical ones in blue.

We have then that $e_{1,3}$ belongs to $S_{e_{3,7}}, S_{e_{4,5}}$ and the partition $\mathcal{P}$ induced by φ is overlapping with each cluster satisfying Equation 6.6. In fact $\varphi(e) - e$ is a combination of non-critical links, meaning that the only critical link f such that $< \varphi(e), f > \neq 0$ is e itself. We could combine the results proved in Theorem 6.24 and the high dimensional Laplacian introduced in [HJ13b] to study the effect of a Morse function on the graph structure, in terms of both nodes and links partition.

Chapter 7

Variants of `Morse` clustering

In this final chapter we will explain some variants of our algorithm `Morse`. Although most of them are still in their early stage of development, with yet no result or application available, we will present them here and explain their characteristics and potential. The variants proposed here can be divided in three main categories: *threshold-based* `Morse`, where we use a threshold parameter either for the nodes or the links, Section 7.1; *stochastic* `Morse`, where, instead of flow through the unique maximal link in E_v, we allow a node to flow along all the maximal and admissible links, Section 7.2; *iterative* `Morse`, which employs an iterative process either on the critical links or on the critical nodes (for the latter a hierarchical clustering algorithm is constructed), Section 7.3. All versions but the stochastic one, employ the notion of *dendogram* [JD88], which is well-known way of represents a nested sequence of clustering solutions, see Figure 7.1.

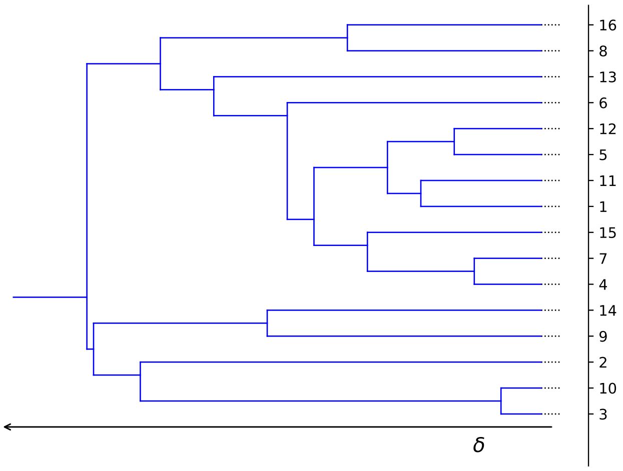


FIGURE 7.1: An example of dendogram with parameter δ. For any value of δ the corresponding vertical cut will give a partition of the nodes.

As we will see in the next sections the properties of `Morse` allow us to employ such procedure also when we have a clustering solution and not a sequence of clusterings.

7.1 Threshold-based `Morse`

Dendograms are often used for hierarchical clustering algorithms [Har75], where a sequence of nested clusters is given as solution, $\{\mathcal{P}_0, \mathcal{P}_1, \ldots, \mathcal{P}_t\}$, with $\mathcal{P}_0$ the singleton partition. As each cluster in $\mathcal{P}_{i-1}$ is subset of a cluster in $\mathcal{P}_i$ for $i = 1, \ldots, k$, we can graphically picture such sequence using a tree structure. The *dendogram tree* $\mathcal{D}$ is constructed using as nodes each cluster C of $\mathcal{P}_i$ for $i = 0, \ldots, k$, and connecting two nodes $C \in \mathcal{P}_i$ and $C' \in \mathcal{P}_{i+1}$ when $C \subset C'$. To draw $\mathcal{D}$ we assume that a certain parameter δ is monotonically increasing/decreasing through the sequence because of theoretical and practical obstructions [FP11]. Example of such δ can be the maximum/minimum of the inter/intra cluster link-weights (with the weight a distance-like function) so that an increasing δ implies that as we keep merging we lose quality in the clusters, that is clusters will contain less similar nodes. In the case of `Morse` clustering we are able to represent directly the partition, assuming some conditions on the preorders, using a dendogram. For a hierarchical algorithm in general it is assumed that at each step we have a clustering and we update the similarities between clusters, constructing so a new weighted graph where nodes are clusters and links are weighted using such similarities. However there are some hierarchical algorithms, as Single Linkage or Complete Linkage, in which the similarities are updated using pair of nodes in different clusters and so called *graph* methods [XW05]. For such algorithms the hierarchical process can be visualised directly on the original graph G. For example in the case of Single Linkage we can consider the clusters at each step as the connected components of the subgraph G_δ of G, in which each link has weight less than δ, for some δ.

A similar procedure can be carried out also for `Morse`. Given an annotated weighted graph G we will consider a iterative process in which, starting with the empty graph (no link is present), we include links depending on the link, or node, preorder. Through this procedure we will obtain a nested sequence of clusters where the final partition coincides with the one we would obtain applying `Morse` directly on G. Consider `Morse` with preorders $\preceq_V, \preceq_E$. First we will proceed including links using the preorder $\preceq_E$ and so we require that it is a total preorder on E, thus inducing a function $f_E : E \to \mathbb{N}$ where $f(e) \geq f(e')$ if $e \preceq_E e'$. This assumption is necessary otherwise we could not include iteratively links in the graph, as some of them will not be comparable and so we would not know which one gets included first. Let G_i be the graph with node set V and link set $E_i = \{e | f_E(e) \leq i\}$. It is clear that if $a < \min_e f_E(e)$ (resp. $b \geq \max_e f_E(e)$) then G_a is the empty graph (resp. $G_b \equiv G$). Let Φ_i be the Morse flow obtained on G_i. If $\Phi_i(v) = s$ with $v \neq s$ then $\Phi_j(v) = s$ for $j > i$, as follows. As v is not critical and $\Phi(v) = s$, the link (v, s) is maximal in E_i for v and so $f_E(e)$ is minimum among the out-links of v in E_i. For any link e' added later to G_j with $j > i$ it is true that $f_E(e') > i \geq f_E(e)$ so e remains maximal and unique. More as for $b \geq \max_e f_E(e)$ we have that Φ_b is the same Morse flow as obtained on G, and the set $\{\Phi_i\}_i$ induces a nested sequence of clusters. As just explained the difference between two flows Φ_i and Φ_{i+1} can be that a node v is critical for Φ_i but not for Φ_{i+1} and a node becomes not critical as its maximal link is included in G_{i+1}. As we can see in Figure 7.2 the merging process will take place only between a node and an existing rooted tree, but never between two rooted trees, as a consequence of $\preceq_E$ being total.

In the following example we used as annotation function the degree centrality measure and assigned as link weight a random real number between 0 and 1.

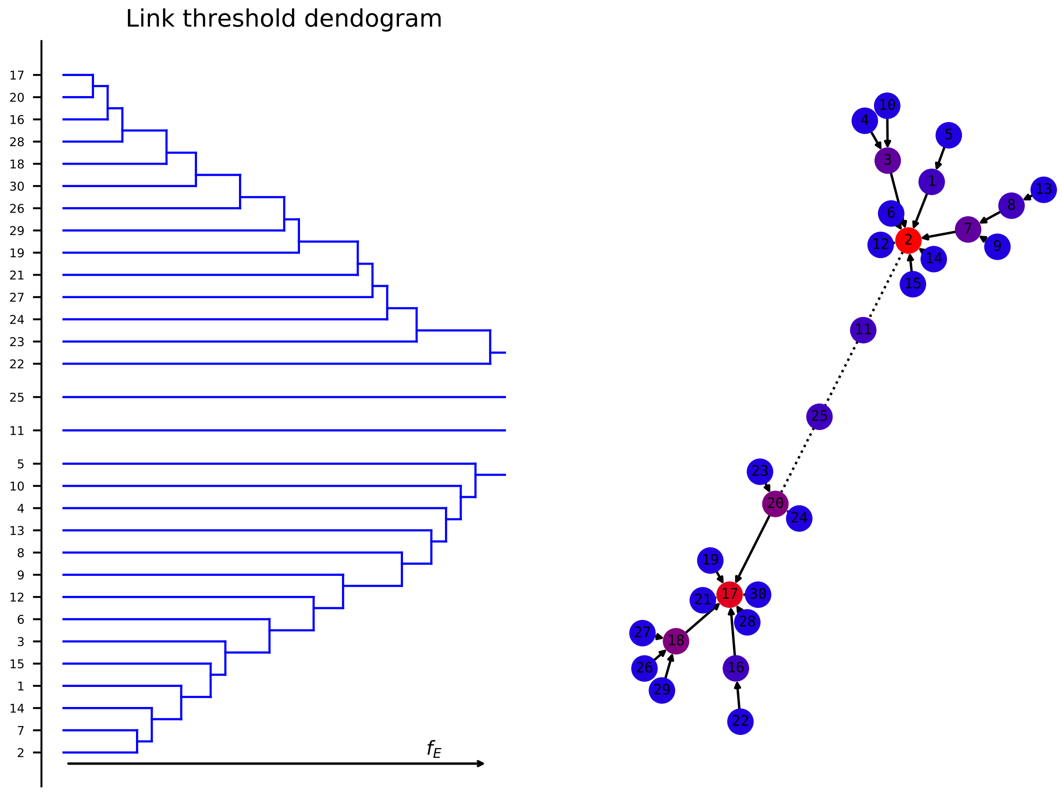


FIGURE 7.2: In the dendogram from left to right we increase the link-threshold w.r.t. f_E and obtain a nested sequence of clusters. On the right picture we can see the flow Φ on G with node colouring based on their annotation (degree) and dotted lines for links not used by Φ. The link connecting the two central nodes is their maximal link, so they will be critical.

Thanks to this approach an analysis of the stability and convergence of `Morse` can be carried out, similarly to [CM10] where a *functoriality* property that characterises *Hierarchical Single Linkage* is proven.

A similar technique can be applied using the node preorder instead, that is, considering now the function $f_V : V \to \mathbb{N}$ induced by the preorder $\preceq_V$. The process we describe now has the property of revealing the basins of attraction of a graph while adding links. As we will include links (and nodes) using the node preorder we require that $\preceq_V$ is a total preorder. In addition we will ask that $\preceq_E$ is a hill-climbing preorder, that is, if $t \prec_V v \prec_V s$ then $(v,t) \prec_E (v,s)$, so we value more links connecting v to higher nodes.

The iterative process will be as follows. Let $f_V : V \to \mathbb{N}$ be such that if $f_V(v) < f_V(s)$ then $v \prec_V s$, define the graph $G_i = (V, E_i)$ where $e = (v, u) \in E_i$ if $f_V(v), f_V(u) \geq i$ and $e \in E$. Note that for $b \geq \max_u f(u)$ (resp. $a < \min_u f_V(u)$) we have that G_b is the empty graph (resp. $G_a \equiv G$). Consider Φ_i Morse flow obtained on G_i, if $\Phi_i(v) = s$ with $v \neq s$ then $\Phi_j(v) = s$ for $j < i$. In fact, given j such that $j < i$ and $e = (v, t)$ in $E_j \setminus E_i$, we have that $f_V(t) < i \leq f_V(v), f_V(s)$ and so $(v, s) \prec_E (v, t)$, for the properties of $\preceq_E$. We obtain so a nested sequence of clusters, for decreasing values of i, as before.

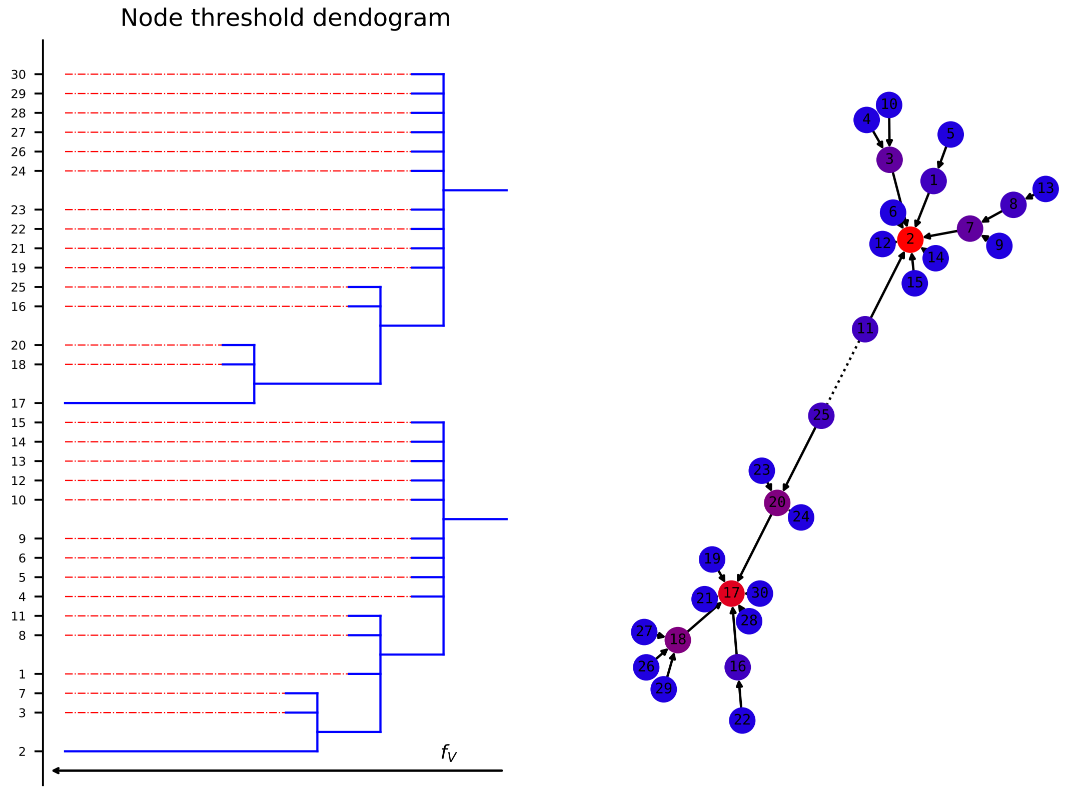


FIGURE 7.3: In the dendogram from left to right we decrease the node-threshold w.r.t. f_V and obtain a nested sequence of clusters. A red dotted line for a node v is draw till the threshold reaches $f_V(v)$. On the right picture we can see the flow Φ on G with node colouring based on their annotation (degree) and dotted lines for links not used by Φ.

As the preorder on the links is not total, the nodes in the centre of the graph (labelled 25 and 11), which were critical in Figure 7.2, do have a maximal link connecting them to a higher node.

This approach can be seen as the level-set approach used often in Morse Theory [Mil65]. Let M be a smooth manifold and f a Morse function on it, we can consider for each value c the *level-set* $M(c) = \{p \mid f(p) \geq c\}$. These level-sets create a filtering on the space and they can be used to understand the *persistence* of some topological features [ELZ00]. We can see in the picture below for example that for decreasing value of c we will obtain first three connected components, then two and only at the end one component. In particular with this approach we can always extrapolate a *Reeb graph* associated to f, [SKK91].

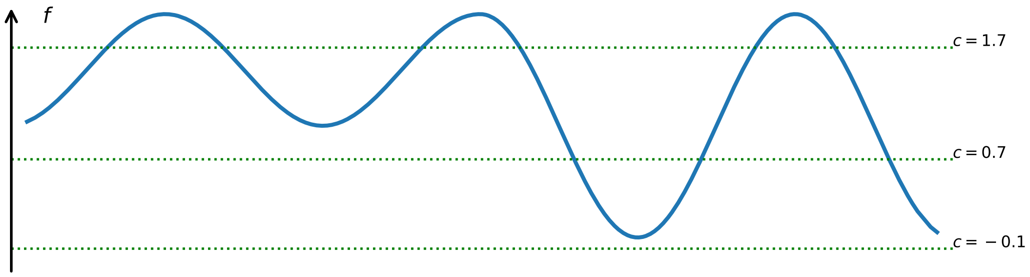


FIGURE 7.4: For different values of c we obtain different number of connected components of $M(c)$.

Note that if we don't assume that $\preceq_E$ is a hill-climbing preorder we could eventually have that a non-critical node v at step i becomes critical at step $j < i$. In fact we could have that a node s connected to v with $f_V(s) < f_V(v)$ realises the maximal link, w.r.t. $\preceq_E$, for v and so v becomes critical. This procedure is telling us that as soon as we start including connection with lower nodes, a node that before was not a local maxima, w.r.t. f_V, becomes one. On one side this is a sensible consequence as including nodes could produce new basins of attraction different from the previous ones and so a nested sequence of roots is generated. In fact given $G_i = (V_i, E_i)$, subgraph induced by all the nodes v such that $f_V(v) \geq i$, if a node v in G_i is critical for Φ_i it remains critical when we consider the Morse flow Φ_j in G_j, with $j < i$. In fact any link $e = (v, s)$ added to G_i at step j connects v with lower node s, so even if e in G_j is a maximal link for v w.r.t. $\prec_E$, it is not an admissible link, as s is lower that v w.r.t. $\prec_V$. In conclusion after a node is critical it can never become not critical, however its basin of attraction could change as we can see in Figure 7.5.

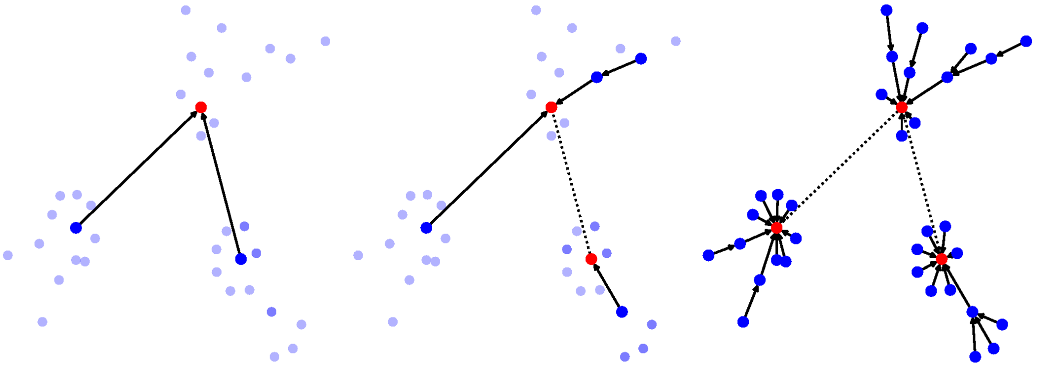


FIGURE 7.5: From right to left we can see the increasing presence of critical nodes (in red). Their change is caused by the appearance of local basins of attraction.

If the previous two versions of threshold-based `Morse` produced a dendogram which can encode the evolution or growth of the Morse flow on a network, this last version could shed light into the changes of basins of attraction in the network when we include or remove nodes with respect to their influence.

7.2 Stochastic `Morse`

We move now to a stochastic version of Morse in which we loosen the constraint on the uniqueness of the maximal link. Such constraint gave us the presence of a Morse flow and so a Morse function, if we let nodes flow through all their maximal and admissible links, it is not necessarily true that we can always recover a Morse flow and a Morse function. However, we still obtain a description of the network in terms of peaks and basins of attraction.

Consider a graph $G = (V, E)$ and two preorders $\preceq_V, \preceq_E$. We will define $\varPhi$ as a *multi-map*, that is, $\varPhi : V \to \mathcal{P}(V)$ where $\mathcal{P}(V)$ is the set of all subset of V. If a node v possesses one or more maximal link, w.r.t. $\preceq_E$, connecting it with higher nodes, w.r.t. $\preceq_V$, say $w_1, \ldots, w_k$ then we define $\varPhi(v) = \{w_1, \ldots, w_k\}$, otherwise $\varPhi(v) = \{v\}$. If we define a map $\varPhi'$ such that for each node v we have $\varPhi'(v) = w_i \in \varPhi(v)$ for some i, we will always obtain a Morse flow and a Morse function. Furthermore, for any of these choices the critical nodes will be always the same. From an intuitive point of view all the Morse flows we could define show different paths for a node v to reach a peak using the best connection(s) available.

As the critical nodes are invariant for each Morse flow $\varPhi'$ so defined, consider a node v such that $\varPhi(v) = \{v\}$ and define the cluster S_v as follows

$$S_v = \{w \mid \exists \varPhi', N \text{ s.t. } \varPhi'^N(w) = v\}.$$

The collection of these clusters, $\mathcal{P} = \{S_v\}_v$, will form in general a fuzzy clustering of G, because a node can now belong to more than one cluster, as shown in Figure 7.6.

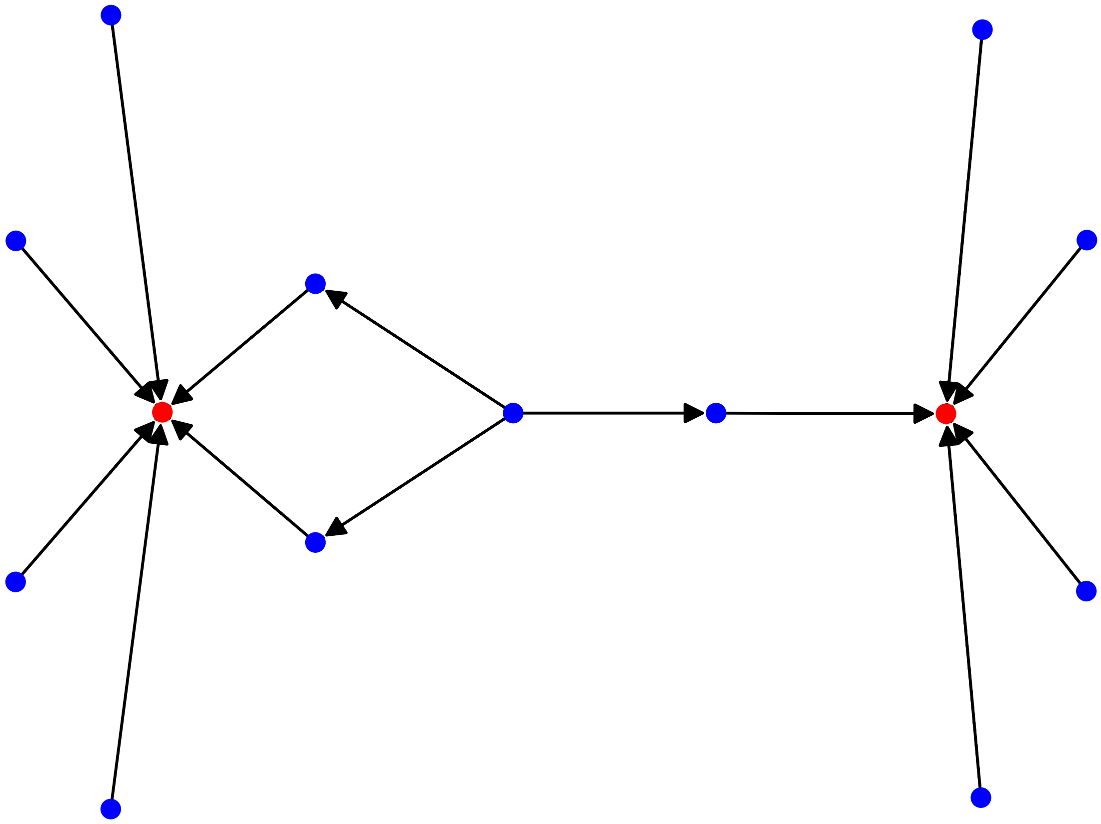


Figure 7.6: In the figure we see a non-critical node (in blue) that has three different flows that bring it to two different critical nodes (in red). Of those two will bring it to the critical node on the left and only one to the critical node on the right.

For each critical node v its cluster S_v will represent its basin of attraction, which however may not have the structure of a tree. Consider the graph $G_v = (S_v, E_v)$ where a link $e = (s, t)$ belongs to E_v if $s, t \in S_v$ and $s \in \Phi(t)$ or $t \in \Phi(s)$. It could happen that a node s flows to v using two different paths and so we would have a cycle as their end and start will coincide, see Figure 7.6.

One can consider Φ as a linear combination of Morse flows on G with same set of critical nodes. That is, define $\varphi : C_0(K) \to C_0(K)$ such that

$$\varphi(v) = \sum_{s \in \Phi(v)} s. \tag{7.1}$$

As previously explained we have that φ is stable, as each v will eventually flow to a critical node. We could consider the matrix expression of φ with respect to V, basis of $C_0(K)$, if needed after a column normalisation, as the belongingness matrix of the partition $\mathcal{P}$, see Section 6.4. This allows us determine the spectral distance of this `Morse` clustering from G. Clearly the linear map defined in Equation 7.1 can be more complex and for example also integrate the link-weight

$$\varphi_{w'}(v) = \sum_{s \in \Phi(v)} w'(s,v)\ s, \tag{7.2}$$

with $w'(s,v) = w(s,v)$ if $v \neq s$ and $w'(v,v) = 1$. Note that not every choice of w' will define a stable $\varphi_{w'}$, that is , such that exists N with $\varphi^N_{w'} = \varphi^{N+1}_{w'}$. For example if $< \varphi_{w'}(v), v > \neq 1$ when v is critical then $\varphi_{w'}$ will never be stable.

This variant could be employed to test `Morse` when we have ground-truth communities that are overlapping, as for example for the SNAP benchmark with ground-truth communities established in [LK14].

7.3 Hierarchical `Morse`

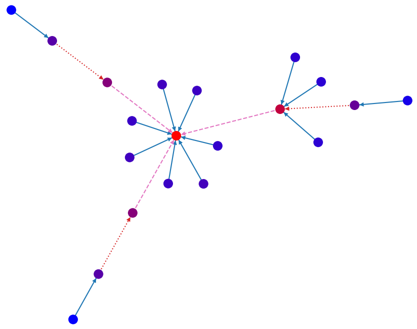
In this last section we present a hierarchical approach. More precisely we will consider instead of a threshold or a overlapping version of `Morse` an actual iteration over the critical links or nodes. We will assume that a graph G is given with weight and annotation functions, together with two preorders on nodes and links. As we will consider an iterative process we assume that the `Morse` preorders are kept invariant, however as in any other hierarchical algorithm the similarities between clusters will be updated and as we are in the annotated graph context also the annotation function.

The first hierarchical approach we present represents a layering of links of G. This process can be summarised as follows

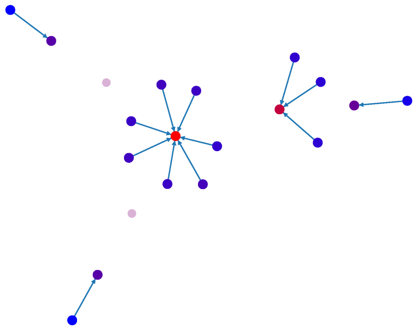
1. Start with an annotated weighted graph $G_0 = G$ and set $i = 0$;
2. Apply `Morse` to G_i obtaining the Morse flow Φ_i;
3. Define the graph G_{i+1} removing from G_i the non-critical links for Φ_i and increase i by 1;
4. Repeat steps 2-3 until no link can be further removed.

Note that at step 3 we will also update if necessary the weight and annotation function on G_{i+1}.

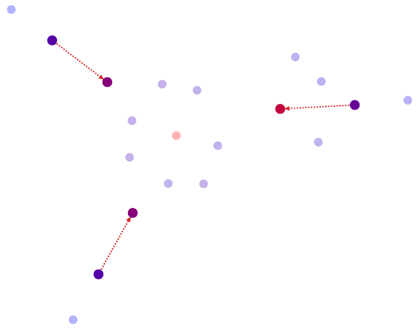
As `Morse` tries to find the best connection (maximal link) which connects to a higher node, the result of this iterative process is a sequence of best connections for each node. At each step i the link $e_i = (v, \Phi_i(v))$, if it exists, is chosen as maximal link for v, after all links better than e have been already chosen, and so removed, in the previous steps. Note that differently from Section 7.1 we could potentially have that the comparison between nodes changes when the annotation gets updated. Suppose the annotation function is a centrality measure, it is sensible then to assume that removal of links would influence such measure and so its update at step i could cause a node to be more central than another also if before it was not. In the following example we chose to keep the annotation and weight function invariant through the iterations.



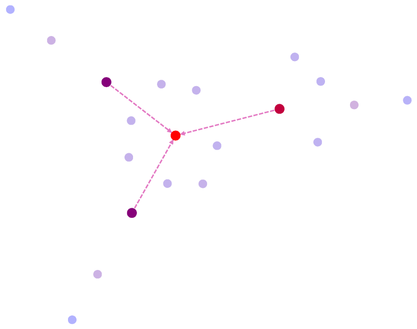
Node colouring based on the annotation, closeness centrality.



1st step with 13 links used, solid blue lines



2nd step with 3 links used, dotted red lines

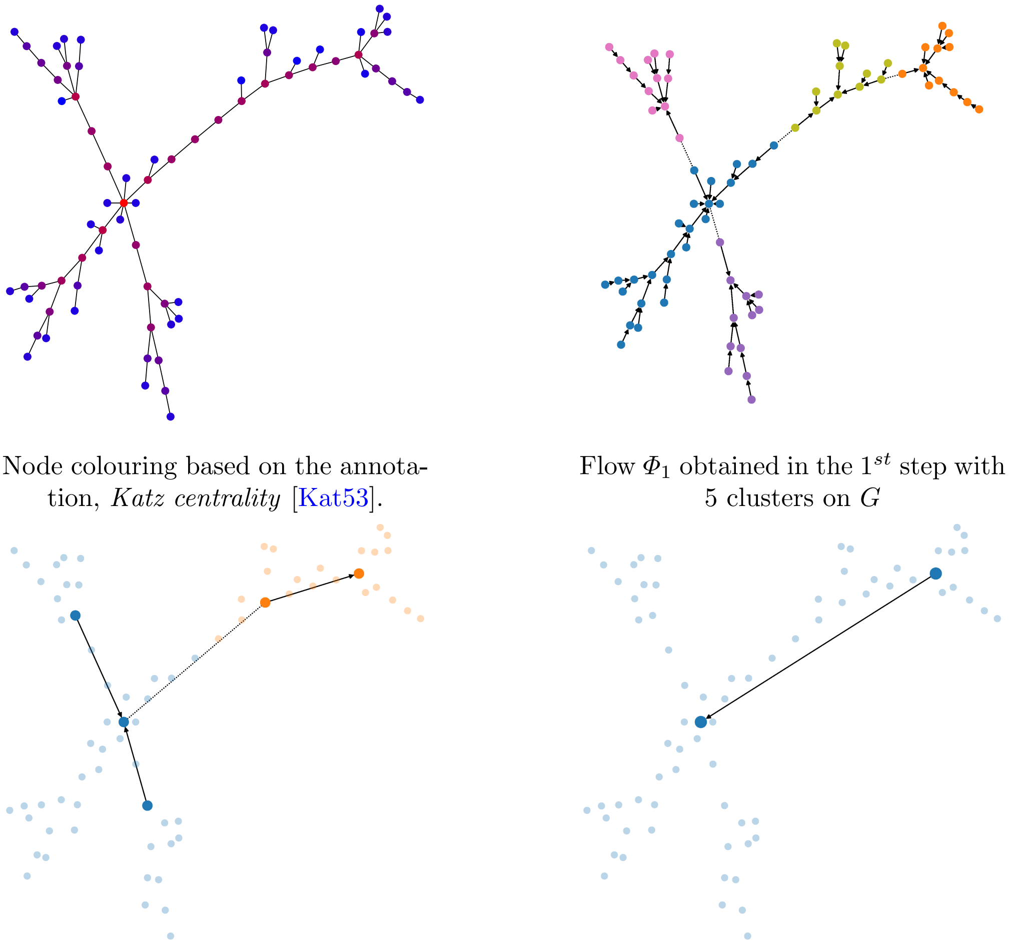


3rd and final step with 3 links used, dashed purple lines

We propose here another hierarchical algorithm which will take in account the critical nodes and will resemble most of hierarchical algorithms. Using that there exists a well-defined bijection between Morse clusters and critical nodes for a Morse flow on a graph G, we will consider the following iterative process

1. Start with an annotated weighted graph $G_0 = G$ and set $i = 0$;
2. Apply `Morse` to $G_i = (V_i, E_i)$ obtaining the Morse flow $\varPhi_i$ and partition $\mathcal{P}_i$;
3. Define the graph G_{i+1} as the quotient graph of G_i with respect the partition $\mathcal{P}_i$ and increase i by 1;
4. Repeat steps 2-3 until $\mathcal{P}_i$ is the singleton partition on G_i.

Note again that at step 3 we will also update if necessary the weight and annotation functions on G_{i+1}. In the following pictures we can see as each step induces a clustering on G, highlighted by the different colouring.



Node colouring based on the annotation, *Katz centrality* [Kat53].

Flow $\varPhi_1$ obtained in the 1^{st} step with 5 clusters on G

Flow $\varPhi_2$ obtained in the 2^{nd} step with 2 clusters on G_1

Flow $\varPhi_3$ obtained in the 3^{rd} step with 1 cluster on G_2

FIGURE 7.7

For the properties of Morse clustering, given $\varPhi_i$ and $\varPhi_j$ with $j < i$ we can identify a critical node for the former with a critical node for the latter. So each critical node at each step corresponds with a node in $G_0 = G$, as shown in Figure 7.7.

In hierarchical algorithms a stopping condition is often defined to break the merging when the quality of the clustering becomes too low [JD88], For example we can set the algorithm to stop when the maximal/minimal average inter/intra-cluster weight is higher than a certain δ. These stopping conditions would be also applicable for the hierarchical version of Morse presented. However, we introduce now an approach that uses the Morse flows defined in the hierarchical process, and so can not be employed by other hierarchical clustering algorithms. This approach would lead to Morse-related stopping conditions.

For simplicity we will focus now on the first two steps of the Morse hierarchical procedure. Consider G_1 and $G_0 = G$ with respective Morse flows $\varPhi_1$ and $\varPhi_0$. We know that $\varPhi_1$ induces a connected partition on G and each of these clusters possesses one and only one critical node for $\varPhi_1$, using the identification of critical node and clusters. In particular this tells us that it is possible to find a rooted directed spanning tree in each cluster with root such critical node. Using these trees we can define a Morse flow $\varPhi$ on G that trivially induces the same partition of $\varPhi_1$ on G. As there could be more than one of these Morse flows we measure how much $\varPhi$ coincides with $\varPhi_0$ and $\varPhi_1$. We define then the *flow extension cost* function for a flow $\varPhi$ on G with respect $\varPhi_1$ and $\varPhi_0$ as

$$Ext_0^1(\varPhi) = \left|\{v \neq \varPhi_0(v) \mid \varPhi_0(v) \neq \varPhi(v)\} \cup \{v = \varPhi_0(v) \mid \varPhi_1(v) \neq \varPhi(v)\}\right|,$$

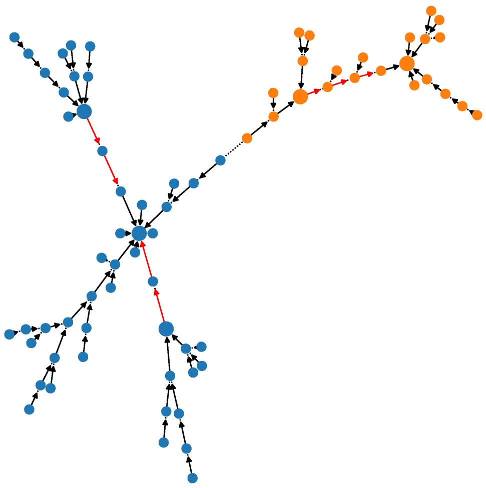
using the identification for the critical nodes of $\varPhi_0$ with the node in G_1. Generalising this argument at step i and j, we can define the flow extension cost function for a flow $\varPhi$ on G_j with respect $\varPhi_i$ and $\varPhi_j$ as

$$Ext_i^j(\varPhi) = \left|\{v \neq \varPhi_j(v) \in V_j \mid \varPhi_j(v) \neq \varPhi(v)\} \cup \{v = \varPhi_j(v) \in V_j \mid \varPhi_i(v) \neq \varPhi(v)\}\right|.$$

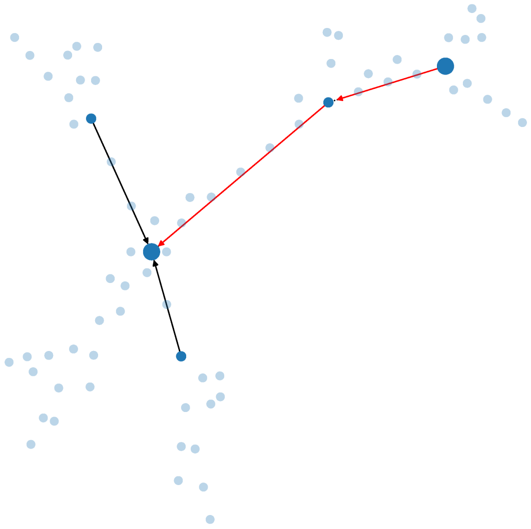
Using the identification of clusters and critical nodes, we know that the clusters $\varPhi_i$ induces on G_j are always connected and so at least one flow $\varPhi$ on G_j inducing same partition of $\varPhi_i$ always exists. Then we can define the *minimal extension* $\overline{\varPhi}_{i\to j}$ as the Morse flow on G_j that induces the same partition of $\varPhi_i$ on G_j and such that

$$\overline{\varPhi}_{i\to j} = \mathrm{Argmin}_{\varPhi}\, Ext_i^j(\varPhi).$$

If $Ext_{i,j}(\overline{\varPhi}_{i\to j}) = 0$ then we have that in each cluster induced by $\varPhi_i$ on G_j there exists a rooted directed spanning tree which directions coincides exactly with $\varPhi_i$ and $\varPhi_j$. In the figure below we can see how we can extend the Morse flow from one step to the previous.



In the 2^{nd} step we need to change 7 flow links to extend $\varPhi_1$ on G



In the 3^{rd} step we need to change 2 flow links to extend $\varPhi_2$ on G_1

This procedure can not be used by other hierarchical clustering methods, as any clustering can be induced by multiple Morse flow and so the minimal extension will lose importance. In the case of `Morse` we can instead define a stopping condition that takes in account the flow extension cost of the minimal extension for two adjacent Morse flows, Φ_i and Φ_{i+1}, or for Φ_i and Φ_0, but as well more elaborate ones that take in account also the weight function, or the preorders used.

Chapter 8

Conclusion

In this thesis I presented a clustering algorithm, `Morse` that, using a novel strategy to incorporate metadata and integrating discrete Morse theory, returns an *annotated clustering solution* revealing the basins of attraction present in the annotated network in form of rooted directed trees. `Morse` can be highly flexible, not only in terms of the preorders, but as well in terms of the annotations used, which can be intrinsic (so free of bias), external (giving a supervised algorithm) or a mixture of them. As shown in Chapter 3 for any choice of preorders `Morse` is a fast and local algorithm and its performance on the LFR benchmark is comparable to that of *Blondel et al* shown in [LF09]. More, `Morse` proved to be valuable in the identification of phenotypes of asthma based on blood gene expression profiles, when used as a post-processing step of a TDA analysis, Chapter 4.

We further studied `Morse` using two different approaches: an axiomatic approach and a Laplacian based one. The first showed that if instead of Kleinberg's Consistency axiom we consider the Monotonic Consistency axiom (see Definition 5.5) then not only does the impossibility theorem not hold, but `Morse` provides a possibility result. In addition this result holds also in the more general case of sparse graphs and graph clustering algorithms. In the Laplacian-based approach we have shown how the spectral distance introduced by [GHL15] can be used to determine the structural changes when we consider a cluster as an unique object. We explored in Chapter 6 bounds of such measure, in the case of hard partition (disjoint clusters), but as well in the case of fuzzy partition (overlapping clusters). These results could allow `Morse` to be used as a pre-processing clustering algorithm and they could help to construct a coarse-grained spectral algorithm.

In conclusion we showed how our novel approach to annotated networks, using `Morse` as a clustering algorithm, provides a way to extract a clustering where each cluster is not only an aggregation of objects, but holds a structure of basin of attraction, with the cluster-attractor being a local "maximum" for the annotation. Thanks to its flexibility and computational speed we showed that Morse can be employed successfully as a clustering algorithm, and it could in the future inspire the development of novel clustering algorithms, as those presented in Chapter 7.

There are several lines of research that can be explored starting from the work presented here, we will now summarise some of them.

Benchmark and validation

The Morse algorithm should be further tested and validated on real-world networks. For that, we need real-world examples with ground-truth communities or an index of correctness of the clustering, such as a clustering quality function. These networks could be collaboration networks, biological networks (which can be protein-protein interactions networks, 'omics networks, cancer data and so on) or power networks. Moreover, the variants presented in Chapter 7 can be further developed and applied to reveal properties of the network and its basins of attraction, such as a hierarchy of peaks/links, a multipath ascending flow or a threshold-based coarse-grained analysis of peaks and basins of attraction.

Theoretical developments

Regarding the axiomatic approach, a remaining open question whether Monotonic Consistency is satisfied by other clustering algorithms that do not satisfy also Consistency. To study and show which are the characteristics that such axiom requires or induces on an algorithm is a topic of interest. Moreover, our intuition is that the possibility theorem proved in Chapter 5 it is not an uniqueness result and so there exists clustering algorithms for which Monotonic Consistency coexists with Richness and Scale-Invariance.

As Morse works with annotations that reveals hierarchical structures, we leave for future work the possibility of integrating both external metadata (which could infer community belongingness) and intrinsic centrality measures (encoding influence in the network), so that the basin of attraction will also include an inference of community belonging. In this direction other authors have already proposed some solutions, such as [MBHG05], in which the weight function of the network is ignored, or [PBMW99] where instead the network is assumed directed. This type of integration should be studied not only in the case of quantitative metadata, as in the previous works, but as well for qualitative one, such as ethnicity, political beliefs and so on.

With respect to the Laplacian results in Chapter 6, it should be possible to generalise these results to higher dimensions, when a Morse function is present on the simplicial complex, using the Wasserstein distance to determine the effect of collapsing, but also to produce a coarse-grained spectral algorithm for graph clustering.

References

[ABDBL12] Margareta Ackerman, Shai Ben-David, Simina Brânzei, and David Loker. Weighted clustering. In *AAAI*, 2012.

[ABDL10] M. Ackerman, S. Ben-David, and D. Loker. Towards property-based classification of clustering paradigms. In J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 10–18. Curran Associates, Inc., 2010.

[ACR$^+$02] Ercan U Acar, Howie Choset, Alfred A Rizzi, Prasad N Atkar, and Douglas Hull. Morse decompositions for coverage tasks. *The International Journal of Robotics Research*, 21(4):331–344, 2002.

[AH81] J A. Hartigan. Consistency of single linkage for high-density clusters. *Journal of the American Statistical Association - J AMER STATIST ASSN*, 76:388–394, 06 1981.

[AR13] C. C. Aggarwal and C. K. Reddy. *Data Clustering: Algorithms and Applications.* CRC press, 2013.

[Bar09] Albert-László Barabási. Scale-free networks: A decade and beyond. *Science*, 325(5939):412–413, 2009.

[Bar16] Albert-László Barabási. *Network science.* Cambridge University Press, 2016.

[BBS$^+$17] Jeannette Bigler, Michael Boedigheimer, James PR Schofield, Paul J Skipp, Julie Corfield, Anthony Rowe, Ana R Sousa, Martin Timour, Lori Twehues, Xuguang Hu, et al. A severe asthma disease signature from gene expression profiling of peripheral blood from u-biopred cohorts. *American Journal of Respiratory and Critical Care Medicine*, 195(10):1311–1320, 2017.

[BDA09] S. Ben-David and M. Ackerman. Measures of clustering quality: A working set of axioms for clustering. In *Advances in Neural Information Processing Systems 21*, pages 121–128. Curran Associates, Inc., 2009.

[BF98] Paul S Bradley and Usama M Fayyad. Refining initial points for k-means clustering. In *ICML*, volume 98, pages 91–99. Citeseer, 1998.

[BGLL08] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, 2008.

[BGSF08] Silvia Biasotti, Daniela Giorgi, Michela Spagnuolo, and Bianca Falcidieno. Reeb graphs for shape analysis and applications. *Theoretical Computer Science*, 392(1-3):5–22, 2008.

[BH65] Geoffrey H Ball and David J Hall. Isodata, a novel method of data analysis and pattern classification. Technical report, Stanford Research Institute Menlo Park CA, 1965.

[BH89] Pierre Baldi and Kurt Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 2(1):53–58, 1989.

[BH11] Andries E Brouwer and Willem H Haemers. *Spectra of graphs*. Springer Science & Business Media, 2011.

[BJ81] Eric Backer and Anil K Jain. A clustering performance measure based on fuzzy set decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (1):66–75, 1981.

[BK87] U. Brehm and W. Kühnel. Combinatorial manifolds with few vertices. *Topology*, 26(4):465 – 473, 1987.

[BR^{+}91] Richard A Brualdi, Herbert John Ryser, et al. *Combinatorial matrix theory*, volume 39. Springer, 1991.

[Car92] Manfredo Perdigão do Carmo. *Riemannian geometry*. Birkhäuser, 1992.

[Cat43] Raymond B Cattell. The description of personality: Basic traits resolved into clusters. *The Journal of Abnormal and Social Psychology*, 38(4):476, 1943.

[CCL03] F. Cazals, F. Chazal, and T. Lewiner. Molecular shape analysis based upon the morse-smale complex and the connolly function. In *Proceedings of the Nineteenth Annual Symposium on Computational Geometry*, SCG '03, pages 351–360. ACM, 2003.

[CM10] G. Carlsson and F. Memoli. Characterization, stability and convergence of hierarchical clustering methods. *Journal of Machine Learning Research*, 11:1425–1470, 2010.

[CM13] J. Correa-Morris. An indication of unification for different clustering approaches. *Pattern Recognition*, 46(9):2548–2561, 2013.

[Des13] Bernard Desgraupes. Clustering indices. *University of Paris Ouest-Lab Modal'X*, vol. 1, 2013.

[DFN12] Boris A Dubrovin, Anatolij T Fomenko, and Sergeĭ Petrovich Novikov. *Modern geometry? Methods and applications: Part II: The geometry and topology of manifolds*, volume 104. Springer Science & Business Media, 2012.

[DHS12] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification.* John Wiley & Sons, 2012.

[Dob70] R. Dobrushin. Prescribing a system of random variables by conditional distributions. *Theory of Probability & Its Applications*, 15(3):458–486, 1970.

[DS14] Norman R Draper and Harry Smith. *Applied regression analysis*, volume 326. John Wiley & Sons, 2014.

[ELLS01] Brian Everitt, Sabine Landau, Morven Leese, and Daniel Stahl. *Cluster analysis.* Arnold, London, 4th edition, 2001.

[ELZ00] Herbert Edelsbrunner, David Letscher, and Afra Zomorodian. Topological persistence and simplification. In *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*, pages 454–463. IEEE, 2000.

[ER60] Paul Erdos and Alfréd Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5(1):17–60, 1960.

[ESBB98] Michael B. Eisen, Paul T. Spellman, Patrick O. Brown, and David Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863–14868, 1998.

[Fie95] Miroslav Fiedler. An estimate for the nonstochastic eigenvalues of doubly stochastic matrices. *Linear Algebra and Its Applications*, 214:133–143, 1995.

[FLNU18] Bailey K Fosdick, Daniel B Larremore, Joel Nishimura, and Johan Ugander. Configuring random graph models with fixed degree sequences. *SIAM Review*, 60(2):315–355, 2018.

[FMB+15] Louise Fleming, Clare Murray, Aruna T Bansal, Simone Hashimoto, Hans Bisgaard, Andrew Bush, Urs Frey, Gunilla Hedlin, Florian Singer, Wim M van Aalderen, et al. The burden of severe asthma in childhood and adolescence: results from the paediatric u-biopred cohorts. *European Respiratory Journal*, 46(5):1322–1333, 2015.

[FN71] L. Fisher and J. W. Van Ness. Admissible clustering procedures. *Biometrika*, 58(1):91–104, 1971.

[For65] Edward Forgey. Cluster analysis of multivariate data: Efficiency vs. interpretability of classification. *Biometrics*, 21(3):768–769, 1965.

[For98] R. Forman. Morse theory for cell complexes. *Advances in Mathematics*, 134:90–145, 1998.

[FP11] Murtagh Fionn and Contreras Pedro. Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(1):86–97, 09 2011.

[GBG$^+$14] Attila Gyulassy, Peer-Timo Bremer, R Grout, Hemanth Kolla, Jacqueline Chen, and Valerio Pascucci. Stability of dissipation elements: A case study in combustion. In *Computer Graphics Forum*, volume 33, pages 51–60. Wiley Online Library, 2014.

[GH55] W. H. Gottschalk and G. A. Hedlund. *Topological dynamics*, volume 36. American Mathematical Soc., 1955.

[GHL15] Jiao Gu, Bobo Hua, and Shiping Liu. Spectral distances on graphs. *Discrete Applied Mathematics*, 190:56–74, 2015.

[GM88] Mark Goresky and Robert MacPherson. Stratified morse theory. In *Stratified Morse Theory*, pages 3–22. Springer, 1988.

[GR69] J. C. Gower and G. J. S. Ross. Minimum spanning trees and single linkage cluster analysis. *Applied Statistics*, pages 54–64, 1969.

[GR01] Chris Godsil and Gordon Royle. Algebraic graph theory. *Springer, New York*, 2001.

[GRS98] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. Cure: an efficient clustering algorithm for large databases. In *ACM Sigmod Record*, volume 27, pages 73–84. ACM, 1998.

[GRS99] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. Rock: A robust clustering algorithm for categorical attributes. In *Data Engineering, 1999. Proceedings., 15th International Conference on*, pages 512–521. IEEE, 1999.

[GT01] Jonathan L Gross and Thomas W Tucker. *Topological graph theory.* Courier Corporation, 2001.

[Han09] Chris Hans. Bayesian lasso regression. *Biometrika*, 96(4):835–845, 2009.

[Har75] John A Hartigan. *Clustering algorithms.* John Wiley & Sons, Inc., 1975.

[Hel06] Aleksandr Yakovlevich Helemskii. *Lectures and exercises on functional analysis*, volume 233. American Mathematical Society Providence, RI, 2006.

[HJ13a] Danijela Horak and Jürgen Jost. Interlacing inequalities for eigenvalues of discrete laplace operators. *Annals of Global Analysis and Geometry*, 43(2):177–207, 2013.

[HJ13b] Danijela Horak and Jürgen Jost. Spectra of combinatorial laplace operators on simplicial complexes. *Advances in Mathematics*, 244:303–336, 2013.

[HK70] Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[HR03] Geoffrey E Hinton and Sam T Roweis. Stochastic neighbor embedding. In *Advances in neural information processing systems*, pages 857–864, 2003.

[HT73] John Hopcroft and Robert Tarjan. Algorithm 447: efficient algorithms for graph manipulation. *Communications of the ACM*, 16(6):372–378, 1973.

[HZS$^+$15] Timothy SC Hinks, Xiaoying Zhou, Karl J Staples, Borislav D Dimitrov, Alexander Manta, Tanya Petrossian, Pek Y Lum, Caroline G Smith, Jon A Ward, Peter H Howarth, et al. Innate and adaptive t cells in asthmatic patients: relationship to severity and disease mechanisms. *Journal of Allergy and Clinical Immunology*, 136(2):323–333, 2015.

[Jai10] A. K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651–666, 2010.

[JD88] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data.* Prentice-Hall, Inc., 1988.

[JMF99] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM Comput. Surv.*, 31(3):264–323, 1999.

[Kar99] N. B. Karayiannis. An axiomatic approach to soft learning vector quantization and clustering. *IEEE Transactions on Neural Networks*, 10(5):1153–1165, 1999.

[Kat53] Leo Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[KGH$^+$10] Maksim Kitsak, Lazaros Gallos, Shlomo Havlin, Fredrik Liljeros, Lev Muchnik, H Stanley, and Hernan A. Makse. Identifying influential spreaders in complex networks. *Nature Physics*, 6, 01 2010.

[KK10] David G Kleinbaum and Mitchel Klein. *Logistic regression: a self-learning text.* Springer Science & Business Media, 2010.

[KKM05] Henry King, Kevin Knudson, and Neža Mramor. Generating discrete morse functions from point data. *Experimental Mathematics*, 14(4):435–444, 2005.

[Kle03] J. Kleinberg. *An Impossibility Theorem for Clustering.* MIT Press, 2003.

[KR09] Leonard Kaufman and Peter J Rousseeuw. *Finding groups in data: an introduction to cluster analysis*, volume 344. John Wiley & Sons, 2009.

[Kru64] Joseph B Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.

[LAA$^+$16] Matthew J Loza, Ian Adcock, Charles Auffray, Kiang F Chung, Ratko Djukanovic, Peter J Sterk, Vedrana S Susulic, Elliot S Barnathan, Frederik Baribaud, and Philip E Silkoff. Longitudinally stable, clinically defined clusters of patients with asthma independently identified in the adept and u-biopred asthma studies. *Annals of the American Thoracic Society*, 13(Supplement 1):S102–S103, 2016.

[Lan72] Serge Lang. *Differential manifolds.* Springer, 1972.

[LBC$^+$14] Claudia Landi, Elena Bargagli, Alfonso Carleo, Laura Bianchi, Assunta Gagliardi, Antje Prasse, Maria G Perari, Rosa M Refini, Luca Bini, and Paola Rottoli. A system biology study of balf from patients affected by idiopathic pulmonary fibrosis (ipf) and healthy controls. *PROTEOMICS–Clinical Applications*, 8(11-12):932–950, 2014.

[LCR$^+$16] Linyuan Lü, Duanbing Chen, Xiao-Long Ren, Qian-Ming Zhang, Yi-Cheng Zhang, and Tao Zhou. Vital nodes identification in complex networks. *Physics Reports*, 650:1–63, 2016.

[LDML$^+$17] Diane Lefaudeux, Bertrand De Meulder, Matthew J Loza, Nancy Peffer, Anthony Rowe, Frédéric Baribaud, Aruna T Bansal, Rene Lutter, Ana R Sousa, Julie Corfield, et al. U-biopred clinical adult asthma clusters linked to a subset of sputum omics. *Journal of Allergy and Clinical Immunology*, 139(6):1797–1807, 2017.

[LF09] Andrea Lancichinetti and Santo Fortunato. Community detection algorithms: a comparative analysis. *Physical Review E*, 80(5):056117, 2009.

[LFH10] Andrea Landherr, Bettina Friedl, and Julia Heidemann. A critical review of centrality measures in social networks. *Business & Information Systems Engineering*, 2(6):371–385, Dec 2010.

[LFK09] Andrea Lancichinetti, Santo Fortunato, and János Kertész. Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics*, 11(3), 2009.

[LK14] Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, 2014.

[LM14] T. Van Laarhoven and E. Marchiori. Axioms for graph clustering quality functions. *Journal of Machine Learning Research*, 15:193–215, 2014.

[LSL$^+$13] Pek Y Lum, Gurjeet Singh, Alan Lehman, Tigran Ishkanov, Mikael Vejdemo-Johansson, Muthu Alagappan, John Carlsson, and Gunnar Carlsson. Extracting insights from the shape of complex data using topology. *Scientific reports*, 3:1236, 02 2013.

[LVV02] Aristidis Likas, Nikos Vlassis, and Jakob Verbeek. The global k-means clustering algorithm. *Pattern Recognition*, 36:451–461, 08 2002.

[LW67] Godfrey N Lance and William Thomas Williams. A general theory of classificatory sorting strategies: 1. hierarchical systems. *The Computer Journal*, 9(4):373–380, 1967.

[LWG12] U. Von Luxburg, R. C. Williamson, and I. Guyon. Clustering: Science or art? In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, volume 27 of *Proceedings of Machine Learning Research*, pages 65–79. PMLR, 2012.

[LZYZ11] Linyuan Lü, Yi-Cheng Zhang, Chi Ho Yeung, and Tao Zhou. Leaders in social networks, the delicious case. *PLOS ONE*, 6(6):1–9, 2011.

[Mac67] J. Macqueen. Some methods for classification and analysis of multivariate observations. In *In 5-th Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, 1967.

[MACO91] Bojan Mohar, Y Alavi, G Chartrand, and OR Oellermann. The laplacian spectrum of graphs. *Graph Theory, Combinatorics, and Applications*, 2(871-898):12, 1991.

[Mat02] Yukio Matsumoto. *An Introduction to Morse Theory*. Translations of Mathematical Monographs, 2002.

[MBHG05] Julie L. Morrison, Rainer Breitling, Desmond J. Higham, and David R. Gilbert. Generank: Using search engine technology for the analysis of microarray experiments. *BMC Bioinformatics*, 6(1):233, 2005.

[mdL22] Pierre Simon marquis de Laplace. *A treatise of celestial mechanics*. Dublin, R. Milliken, 1822.

[Mer94] Russell Merris. Laplacian matrices of graphs: a survey. *Linear Algebra and Its Applications*, 197-198:143 – 176, 1994.

[MH08] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

[Mil58] J. Milnor. *Differential Topology.* Princeton University, 1958.

[Mil63] J. Milnor. *Morse Theory.* Princeton University Press, 1963.

[Mil65] J. Milnor. *Lectures on the* h*-cobordism Theorem.* New Jersey Princeton University Press, 1965.

[MK07] Geoffrey McLachlan and Thriyambakam Krishnan. *The EM algorithm and extensions*, volume 382. John Wiley & Sons, 2007.

[NC16] M. E. J. Newman and A. Clauset. Structure and inference in annotated networks. *Nature Communications*, 7, 2016.

[NLC11] Monica Nicolau, Arnold J Levine, and Gunnar Carlsson. Topology based data analysis identifies a subgroup of breast cancers with a unique mutational profile and excellent survival. *Proceedings of the National Academy of Sciences*, page 201102826, 2011.

[NPL^{+}15] Jessica L Nielson, Jesse Paquette, Aiwen W Liu, Cristian F Guandique, C Amy Tovar, Tomoo Inoue, Karen-Amanda Irvine, John C Gensel, Jennifer Kloke, Tanya C Petrossian, et al. Topological data analysis for discovery in preclinical spinal cord injury and traumatic brain injury. *Nature Communications*, 6:8581, 2015.

[PBMW99] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[PHB00] J. Puzicha, T. Hofmann, and J. M. Buhmann. A theory of proximity based clustering: Structure detection by optimization. *Pattern Recognition*, 33(4):617–634, 2000.

[PLC17] Leto Peel, Daniel B. Larremore, and Aaron Clauset. The ground truth about metadata and community detection in networks. *Science Advances*, 3(5), 2017.

[PTNKT11] Frank J Poelwijk, Sorin Tănase-Nicola, Daniel J Kiviet, and Sander J Tans. Reciprocal sign epistasis is a necessary condition for multi-peaked fitness landscapes. *Journal of Theoretical Biology*, 272(1):141–144, 2011.

[PTVF07] WH Press, SA Teukolsky, WT Vetterling, and BP Flannery. Numerical recipes: The art of scientific computing. In *Section 14.7. 3. Conditional Entropy and Mutual Information.* Cambridge University Press, Cambridge, UK, 3rd edition, 2007.

[RB08] Martin Rosvall and Carl T Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences*, 105(4):1118–1123, 2008.

[Rou87] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[RR15] Ahmad Rawashdeh and Anca L Ralescu. Similarity measure for social networks-a brief survey. In *MAICS*, pages 153–159, 2015.

[RWS11] Vanessa Robins, Peter John Wood, and Adrian P Sheppard. Theory and algorithms for constructing discrete morse complexes from grayscale digital images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1646–1658, 2011.

[Sch03] Bernd SW Schröder. *Ordered sets.* Springer, 2003.

[Sch07] S. E. Schaeffer. Graph clustering. *Computer Science Review*, 1(1):27–64, 2007.

[SG18] Ruben J Sanchez-Garcia. Exploiting symmetry in network analysis. *arXiv preprint arXiv:1803.06915*, 2018.

[SKK91] Yoshihisa Shinagawa, Tosiyasu L. Kunii, and Yannick L. Kergosien. Surface coding based on morse theory. *IEEE Comput. Graph. Appl.*, 11(5):66–78, 1991.

[SMC91] Gurjeet Singh, Facundo Mémoli, and Gunnar Carlsson. Mapper: a topological mapping tool for point cloud data. In *Eurographics symposium on point-based graphics*, volume 102, 1991.

[Sou11] Thierry Sousbie. The persistent cosmic web and its filamentary structure–i. theory and implementation. *Monthly Notices of the Royal Astronomical Society*, 414(1):350–383, 2011.

[SRB07] Gábor J. Székely, Maria L. Rizzo, and Nail K. Bakirov. Measuring and testing dependence by correlation of distances. *The Annals of Statistics*, 35(6):2769–2794, 12 2007.

[SS00] Roded Sharan and Ron Shamir. Click: a clustering algorithm with applications to gene expression analysis. In *Proc Int Conf Intell Syst Mol Biol*, volume 8, page 16, 2000.

[SSF^{+}15] Dominick E Shaw, Ana R Sousa, Stephen J Fowler, Louise J Fleming, Graham Roberts, Julie Corfield, Ioannis Pandis, Aruna T Bansal, Elisabeth H Bel, Charles Auffray, et al. Clinical and inflammatory characteristics of the european u-biopred adult severe asthma cohort. *European Respiratory Journal*, pages ERJ–00779, 2015.

[Str92] Walter A Strauss. *Partial differential equations.* John Wiley & Sons New York, NY, USA, 1992.

[Stu08] Student. The probable error of a mean. *Biometrika*, 6(1):1–25, 1908.

[Vil08] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

[VL07] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[WDM12] Hubert Wagner, Paweł Dłotko, and Marian Mrozek. Computational topology in text mining. In *Computational Topology in Image Context*, pages 68–78. Springer, 2012.

[WGB$^+$12] Craig E Wheelock, Victoria M Goss, David Balgoma, Ben Nicholas, Joost Brandsma, Paul J Skipp, Stuart Snowden, Arnaldo D'Amico, Ildiko Horvath, Amphun Chaiboonchoe, et al. Application of 'omics technologies to biomarker discovery in inflammatory lung diseases. *European Respiratory Journal*, pages erj00788–2012, 2012.

[Whi49] John HC Whitehead. Combinatorial homotopy. i. *Bulletin of the American Mathematical Society*, 55(3):213–245, 1949.

[WM97] David H Wolpert and William G Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.

[Wol96] David H Wolpert. The lack of a priori distinctions between learning algorithms. *Neural Computation*, 8(7):1341–1390, 1996.

[Wri73] William E Wright. A formalization of cluster analysis. *Pattern Recognition*, 5(3):273–282, 1973.

[WZ08] Richard C. Wilson and Ping Zhu. A study of graph spectra for comparing graphs and trees. *Pattern Recognition*, 41(9):2833 – 2841, 2008.

[XW05] Rui Xu and Donald Wunsch. Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, 16(3):645–678, 2005.

[YX14] J. Yu and Z. Xu. Categorization axioms for clustering results. *arXiv preprint arXiv:1403.2065*, 2014.

[ZB12] R. Zadeh and S. Ben-David. A uniqueness theorem for clustering. *CoRR*, abs/1205.2600, 2012.

[ZH05] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.

[ZRL96] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Birch: an efficient data clustering method for very large databases. In *ACM Sigmod Record*, volume 25, pages 103–114. ACM, 1996.

[ZUS59] K ZUSE. Über den Plankalkül. *it-Information Technology*, 1(1-4):68–71, 1959.