UNVEILING THE SAMPLING DENSITY IN NON-UNIFORM GEOMETRIC GRAPHS

Anonymous authors
Paper under double-blind review

ABSTRACT

A powerful framework for studying graphs is to consider them as geometric graphs: nodes are randomly sampled from an underlying metric space, and any pair of nodes is connected if their distance is less than a specified neighborhood radius. Currently, the literature mostly focuses on uniform sampling and constant neighborhood radius. However, real-world graphs are likely to be better represented by a model in which the sampling density and the neighborhood radius can both vary over the latent space. For instance, in a social network communities can be modeled as densely sampled areas, and hubs as nodes with larger neighborhood radius. In this work, we first perform a rigorous mathematical analysis of this (more general) class of models, including derivations of the resulting graph shift operators. The key insight is that graph shift operators should be corrected in order to avoid potential distortions introduced by the non-uniform sampling. Then, we develop methods to estimate the unknown sampling density in a self-supervised fashion. Finally, we present exemplary applications in which the learnt density is used to 1) correct the graph shift operator and improve performance on a variety of tasks, 2) improve pooling, and 3) extract knowledge from networks. Our experimental findings support our theory and provide strong evidence for our model.

1 INTRODUCTION

Graphs are mathematical object used to represent relationships among entities. Their use is ubiquitous, ranging from social networks to recommender systems, from protein-to-protein interactions to functional brain networks. Despite their versatility, some challenges arises when signal analysis is performed on them, primarily caused by their non-euclidean nature: the lack of a well-posed definition of direction, the arbitrariness in the order of nodes, the variability of neighbourhoods in terms of size and shape. Those challenges are especially cumbersome when dealing with a set of different graphs, since the possibly different number of nodes could prevent us from using their algebraic characterization for comparisons. Therefore, new ways of thinking about graphs were developed by the community. One solution is proposed in graphon theory (see, e.g.,Lovász (2012)): every convergent sequence of graphs converges to a limit object called *graphon* and every graphon is the limit object of a convergent sequence of graphs. Hence, as done in Ruiz et al. (2020); Maskey et al. (2022), graphs are similar if they approximate the same graphon. A different approach is presented in Levie et al. (2020): graphs shift operators are thought as discretizations of operators in latent Hilbert spaces, and graphs are similar if their graph shift operators discretize the same operator. In both approaches, a particularly important special case are *geometric graphs*, also referred to as *spatial networks*: a set of points is uniformly sampled from a region of a metric space, and any two points are linked if their distance is less than a specified neighborhood radius.

In contrast to other graph representation models, geometric graphs inherit from the latent space a geometric footprint that is reflected into the structural properties of the graph itself. This can be leveraged in order to perform rigorous mathematical analysis, as done in the current work. For instance, in Section (2.2) the non-uniform geometric graph model (NuG) is introduced. Differently from geometric graphs, a NuG is generated by a non-uniform sampling, and any two points are connected if their distance is less than a specified value that can depend on the point. In a nutshell, Section (2.2) addresses the problem of graph generation knowing the latent space and the sampling density. In reality we usually face the reverse problem: for a raw graph data, just the connectivity

of the graph is known, and we would like to say something about the process that generated it. This will be the focus of Sections (2.3) to (2.5), where it is shown which latent properties can be inferred from the graph. The theoretical findings are then corroborated by numerical experiments and proof-of-concept applications in Section (3).

1.1 Our Contributions

In contrast to the standard geometric graph models (see Section (4)), whose focus is mainly on uniform sampling, in this work we allow the sampling density and the neighborhood radius to vary on the latent space; the generated graph is, therefore, referred to as non-uniform geometric graphs (NuG).

This is motivated by the assumption that real-world graphs are not uniformly sampled. For example, in location-based social networks the spatial distribution of nodes is rarely uniform, e.g. people tend to congregate around the city centres (Wang & González, 2009; Cho et al., 2011). Similarly, in a WWW network there are more pages (and thus nodes) for popular topics compared to obscure topics. In online communities such as the the LiveJournal social network, non-uniformity may also arise since the probability of befriending a particular person is inversely proportional to the number of closer people (Liben-Nowell et al., 2005; Hu et al., 2011). Similarly, different demographics (age, gender, ethnicity, ...) may join social networks at different rates. For graphs that discretize surfaces, depending on the required level of details, certain locations may be sampled more finely. In all these cases, the sampling procedure may not be accessible, least of all tuned, but determined by the environment.

The imbalance caused by the sampling procedure could affect the analysis and lead to biased results. For the Spatial Preferential Attachment model, Janssen et al. (2016) shows that incorrectly assuming uniform density consistently overestimates the node distances, and that using the (estimated) density is more accurate. By way of illustration, a practitioner interested in deriving society trends by observing social networks may want to correct such imbalance to obtain proper prediction. Therefore, it is essential to assess the sampling density, which is one of the main goals of this paper.

Justified by new formulas grounded in Monte-Carlo analysis, the graph shift operators should be corrected in order to avoid possible distortions. Such correction is intimately related to the inverse sampling density; therefore, it is essential to develop methods to approximate and/or learn the density. We first motivate theoretically the feasibility of the problem and study it on synthetic datasets. Then, we show that the correction can indeed enhance performance on real-world graphs. Learning the sampling density, via our correction formulas, can be seen as a way to learn the Laplacian itself. Hence, together with Dasoulas et al. (2021) and Sahbi (2021), this work can be listed as a theoretically grounded way to learn the Laplacians.

Even though the analysis is performed on discretizations of metric measure spaces, the approach can be generalized to other type of graphs, such as molecular graphs. In this setting, the sampling density acquires a new meaning of importance, which can be exploited to improve pooling, and to extract knowledge from the dataset.

2 Non-Uniform Geometric Models

In practical data science on graphs, typically only the graph structure is given by the data, and the practitioner has the freedom to design a corresponding graph shift operator (GSO) (for the definition of GSO see, for instance, Mateos et al. (2018); Dasoulas et al. (2021)). In our setting, we define a systematic approach to model GSOs from graphs in such a way that the GSO is guaranteed to be a discretization of some latent continuous Laplacian, called *metric-probability Laplacian*. This involves two ideas. First, we model graphs as randomly and non-uniformly sampled from a metric-probability space, where pairs of nodes are connected by an edge if they are below some threshold distance. Here, different locations in the metric space may have different corresponding distance thresholds. This guarantees that the graph approximates some *neighborhood model* of the metric space. Second, we restrict our focus to a class of metric-probability Laplacians that are uniquely determined by the neighborhood model. This allows computing a corresponding GSO directly from the graph structure and the sampling density, while guaranteeing that the GSO approximates the metric-probability Laplacian. *Non-uniform geometric graphs* embody this construction, where the

non-uniformity describes the varying radius in the neighborhood model and the non-constant sampling density. In this section we develop the above ideas rigorously, and show how to estimate the sampling density of a class of non-uniform geometric graphs, called *geometric graphs with hubs*, just by observing the graph structure and the graph signal.

2.1 Continuous Laplacians As Generators of Graph Laplacians

Throughout the current work, we suppose graphs are sampled from a metric-probability space $(\mathcal{S}, d, \mu)$, where μ is the Borel measure arising from the sigma-algebra induced by the metric d. Intuitively, the distance d gives a way to identify balls $\mathrm{B}_\alpha(x) = \{y \in \mathcal{S} : d(x,y) \leq \alpha\}$, while the measure μ gives a way to compute their volumes.

Definition 1 (Metric Probability Laplacian)**.** Let $(\mathcal{S}, d, \mu)$ be a metric-probability space with a probability measure μ; let $m \in \mathrm{L}^\infty(\mathcal{S})$[1]; let $K \in \mathrm{L}^\infty(\mathcal{S} \times \mathcal{S})$. The *metric-probability Laplacian* $\mathcal{L} = \mathcal{L}_{K,m}$ is defined to be

$$\mathcal{L} : \mathrm{L}^\infty(\mathcal{S}) \to \mathrm{L}^\infty(\mathcal{S}) \,,\ (\mathcal{L}u)(x) = \int_{\mathcal{S}} K(x,y)\, u(y)\, \mathrm{d}\mu(y) - m(x)\, u(x) \,. \tag{1}$$

The chosen name is justified by the fact that the metric-probability Laplacian can approximate the usual Laplace operator. For instance, let $\mathcal{S}$ be a n-dimensional Riemannian manifold, let $K(x,y)$ be the indicator function over small balls $\mathrm{B}_\alpha(x)$ normalized by $\mu(\mathrm{B}_\alpha(x))$, let $m(x) = 1$; the metric-probability Laplacian $\mathcal{L}_{K,m}$ approximates the Laplace-Beltrami operator (Burago et al., 2018).

As a modelling assumption, we suppose graphs are non-uniformly sampled from $(\mathcal{S}, d, \mu)$, but rather according to a different measure ν. We assume ν is a weighted measure with respect to μ, that is, there exists a density function $\rho : \mathcal{S} \to (0, \infty)$ such that $\mathrm{d}\nu(y) = \rho(y)\, \mathrm{d}\mu(y)$[2].We assume that ρ is bounded away from 0 and ∞. Hence, it is crucial to study how the metric-probability Laplacian in Equation (1) changes under a change of measure. Using a simple change of variable, it is easy to see

$$(\mathcal{L}u)(x) = \int_{\mathcal{S}} K(x,y)\, \rho^{-1}(y)\, u(y)\, \mathrm{d}\nu(y) - m(x)\, u(x).$$

In the above setting, a random graph is defined directly via its GSO, which is sampled randomly from the metric-measure Laplacian $\mathcal{L}$. Let $\mathbf{x} = \{x_i\}_{i=1}^N$ be a random independent sample from $\mathcal{S}$ according to the distribution ν. The corresponding GSO $\mathbf{L}$ is defined by

$$L_{i,j} = N^{-1} K(x_i, x_j) \rho(x_j)^{-1} - m(x_i) \,. \tag{2}$$

Given a signal $u \in \mathrm{L}^\infty(\mathcal{S})$, define the corresponding sampled graph signal $\mathbf{u} = \{u(x_i)\}_{i=1}^N$. Based on Monte-Carlo analysis, it is well known that $(\mathbf{Lu})_i$ approximates $\mathcal{L}u(x_i)$ for every $i \in \{1, \ldots, N\}$ (Hein et al., 2007; von Luxburg et al., 2008).

2.2 Non-uniform Geometric Metric-Measure Laplacian

The support of $K(x, \cdot)$ in Equation (1) identifies the set of points in $\mathcal{S}$ which contribute to the value of the metric-probability Laplacian at x, or equivalently, the continuous neighborhood of x. The support of the kernel can be decoupled from the value it attains by defining a suitable neighborhood model $\mathcal{N}(x)$, namely, a mapping that assigns a neighborhood of x to every $x \in \mathcal{S}$, and a suitable weighting function $W(x,y)$. Consequently, Equation (1) can be rewritten as

$$(\mathcal{L}u)(x) = \int_{\mathcal{N}(x)} W(x,y)\, u(y)\, \mathrm{d}\mu(y) - m(x) u(x) \,.$$

In the next definition, which we take as a modelling assumption, we suppose that the neighborhood $\mathcal{N}(x)$ is a ball centered at x.

[1] A function $g : \mathcal{S} \to \mathbb{R}$ is an element of $\mathrm{L}^\infty(\mathcal{S})$ iff. $\exists M < \infty : \mu(\{x \in \mathcal{S} : g(x) > M\}) = 0$. The norm in $\mathrm{L}^\infty(\mathcal{S})$ is the essential supremum, i.e. $\inf\{M \geq 0 : |g(x)| \leq M$ for almost every $x \in \mathcal{S}\}$.

[2] Formally, ν is absolutely continuous with respect to μ, with Radon-Nykodin derivative ρ.

Definition 2 (Neighborhood model)**.** Let $(\mathcal{S}, d, \mu)$ be a metric-probability space. Let $\alpha : \mathcal{S} \to (0, +\infty)$ be a non-negative measurable function, referred to as *neighborhood radius*. Define the neighborhood model $\mathcal{N}$ as the set-valued function that assigns to each $x \in \mathcal{S}$ the ball

$$\mathcal{N}(x) \coloneqq \{y \in \mathcal{S} \,:\, d(x,y) \leq \max\big(\alpha(x), \alpha(y)\big)\}\,.$$

The neighborhood model determines which points are considered adjacent in $\mathcal{S}$. Hence, we define the *adjacency operator* $\mathcal{A}_{\mathcal{N}}$ as the operator that integrates over neighborhoods, and the degree operator $\mathcal{D}_{\mathcal{N}}$ as the operator that computes the size of neighborhoods

$$\forall\, u \in \mathrm{L}^\infty(\mathcal{S})\,,\ (\mathcal{A}_{\mathcal{N}} u)\,(x) = \int\limits_{\mathcal{N}(x)} u(y)\,\mathrm{d}\mu(y)\,; \quad \mathcal{D}_{\mathcal{N}}(x) = \int\limits_{\mathcal{N}(x)} \mathrm{d}\mu(y)\,.$$

Finally, given a function $f \in \mathrm{L}^\infty(\mathcal{S})$, we define the multiplication operator $\mathcal{M}[f]$ by

$$\forall\, u \in \mathrm{L}^\infty(\mathcal{S})\,,\ \big(\mathcal{M}[f]u\big)\,(x) = f(x)\,u(x)\,.$$

With those ingredients at hand, we can define a mapping that assigns a metric-probability Laplacian to every metric-probability space with neighborhood model.

Definition 3 (Metric-probability Laplacian model)**.** Let $\mathcal{N}$ be a neighborhood model as in Definition (2). Let $m^{(i)} : \mathbb{R} \to \mathbb{R}$ for every $i \in \{1, \ldots, 4\}$. The *metric-probability Laplacian model* $\mathcal{L}$ is defined as

$$\mathcal{L}_{\mathcal{N}} \coloneqq \mathcal{M}\left[m^{(1)}_{\mathcal{N}}\right] \mathcal{A}_{\mathcal{N}} \mathcal{M}\left[m^{(2)}_{\mathcal{N}}\right] - \mathcal{M}\left[q_{\mathcal{N}}\right], \quad q_{\mathcal{N}} \coloneqq \mathcal{M}\left[m^{(3)}_{\mathcal{N}}\right] \mathcal{A}_{\mathcal{N}} \mathcal{M}\left[m^{(4)}_{\mathcal{N}}\right] 1,$$

where $m^{(i)}_{\mathcal{N}}(x) = m^{(i)}\left(\mathcal{D}_{\mathcal{N}}(x)\right)$ for every $i \in \{1, \ldots, 4\}$, and 1 is the function $\mathcal{S} \ni x \mapsto 1$.

The metric-probability Laplacian model naturally respects the connectivity of the generated graph; thus, it agrees with the definition of GSO. Moreover, the model gives a structured way to create customized graph Laplacians, an example of which is introduced, and studied, in Appendix (C).

2.3 Non-Uniform geometric Graph Laplacian

A non-uniform geometric GSO $\mathbf{L}_{\mathcal{N}}$ can be directly defined by sampling a metric-probability Laplacian $\mathcal{L}_{\mathcal{N}}$ according to Equation (2). However, such an approach would violate our motivating guidelines, since we would like to compute a GSO without any explicit knowledge or assumption on the underlying continuous structure. In the following, we show how to overcome the need to sample the metric-probability Laplacian $\mathcal{L}_{\mathcal{N}}$; instead, we will approximate $\mathcal{L}_{\mathcal{N}}$ by computing a GSO directly on the graph $\mathcal{G}$ from the known sample of the density. Let us introduce some preliminary notations. Given a vector $\mathbf{u} \in \mathbb{R}^N$ and a function $q : \mathbb{R} \to \mathbb{R}$, we denote by $q(\mathbf{u})$ the vector obtained applying q entry-wise, and by $\mathrm{diag}(\mathbf{u}) \in \mathbb{R}^{N \times N}$ the diagonal matrix with diagonal entries given by $\mathbf{u}$. With these notation, we are ready to give a core definition, namely, the non-uniform geometric GSO.

Definition 4 (Non-uniform geometric GSO)**.** Let $\mathcal{G} = (V, E)$ be a graph with adjacency matrix $\mathbf{A}$; let $\boldsymbol{\rho} : V \to (0, \infty)$ be a graph signal, referred to as *graph density*. The *non-uniform geometric GSO* is defined to be

$$\mathbf{L}_{\mathcal{G},\boldsymbol{\rho}} = N^{-1}\,\mathrm{diag}\left(m^{(1)}\left(N^{-1}\mathbf{d}_{\boldsymbol{\rho}}\right)\right)\mathbf{A}_{\boldsymbol{\rho}}\,\mathrm{diag}\left(m^{(2)}\left(N^{-1}\mathbf{d}_{\boldsymbol{\rho}}\right)\right) - N^{-1}\,\mathrm{diag}\left(\mathbf{q}_{\mathcal{G},\boldsymbol{\rho}}\right), \tag{3}$$

where

$$\mathbf{q}_{\mathcal{G},\boldsymbol{\rho}} \coloneqq \mathrm{diag}\left(m^{(3)}\left(N^{-1}\mathbf{d}_{\boldsymbol{\rho}}\right)\right)\mathbf{A}_{\boldsymbol{\rho}}\,\mathrm{diag}\left(m^{(4)}\left(N^{-1}\mathbf{d}_{\boldsymbol{\rho}}\right)\right)\mathbf{1},$$

$\mathbf{A}_{\boldsymbol{\rho}} = \mathbf{A}\,\mathrm{diag}(\boldsymbol{\rho})^{-1}$ is the normalized adjacency matrix, and $\mathbf{d}_{\boldsymbol{\rho}} = \mathbf{A}_{\boldsymbol{\rho}}\mathbf{1}$ is the normalized degree.

The non-uniform geometric GSO model can retrieve, as particular cases, the usual graph Laplacian models, as shown in Table (2) in Appendix (C). A question that may arise is whether the non-uniform geometric GSO converges to the metric-probability Laplacian, and in which sense. This is shown in the following proposition, that is a direct application of Monte-Carlo theory.

Proposition 1. *Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a random graph with i.i.d sample $\mathbf{x} = \{x_i\}_{i=1}^N$ from the metric-probability space $(\mathcal{S}, d, \mu)$ with neighborhood structure $\mathcal{N}$. Let $\mathbf{L}_{\mathcal{G},\boldsymbol{\rho}}$ be the non-uniform geometric GSO as in Definition (4). Let $u \in \mathrm{L}^\infty(\mathcal{V})$ and $\mathbf{u} = \{u(x_i)\}_{i=1}^N$. Then, for every $i = 1, \dots, N$,*

$$\mathbb{E}\left((\mathbf{L}_{\mathcal{G},\boldsymbol{\rho}}\mathbf{u})_i - (\mathcal{L}_{\mathcal{N}}u)(x_i)\right)^2 = \mathcal{O}(N^{-1}).$$

Similarly to Proposition (1), by means of Hoeffding's Inequality, we also prove in Appendix D a concentration of error result: with probability at least $1 - p$, it holds

$$\forall\, i \in \{1, \dots, N\}, \quad |(\mathbf{L}_{\mathcal{G},\boldsymbol{\rho}}\mathbf{u})_i - (\mathcal{L}_{\mathcal{N}}u)(x_i)| = \mathcal{O}\left(N^{-\frac{1}{2}}\sqrt{\log(1/p) + \log(N)}\right).$$

2.4 Inferring the Sampling Density

As a consequence of Proposition (1), as the number of nodes grows, the distortion introduced by the non-uniform sampling is corrected if the adjacency matrix is properly normalized as explained in Definition (4). In real-world scenarios, the true value of the sampling density is not known: for a raw graph data, the only available information is the connectivity and node features, and every other property must be inferred from the given data. As a first step towards estimating the sampling density, the following result gives an approximation of the density in terms of the continuous degree.

Lemma 1. *Let $(\mathcal{S}, d, \mu)$ be a metric-probability space; let $\mathcal{N}$ be a neighborhood model as in Definition (2); let ν be a weighted measure with respect to μ with continuous density ρ. There exists a function $c : \mathcal{S} \to \mathcal{S}$ such that $c(x) \in \mathcal{N}(x)$ and $(\rho \circ c)(x)\mu\big(\mathcal{N}(x)\big) = \nu\big(\mathcal{N}(x)\big)$.*

In light of Lemma (1), if the neighborhood radius of x is small enough, if the volumes $\mu(\mathcal{N}(x))$ are approximately constant, and if ρ does not vary too fast, the sampling density at x is roughly proportional to $\nu(\mathcal{N}(x))$, that is, the likelihood a point is drawn from $\mathcal{N}(x)$. Therefore, the sampling density can be approximated in this situation by the degree of the node x. In practice, we are interested in graphs where the volumes of the neighborhoods $\mu(\mathcal{N}(x))$ are not constant. Still, a normalization of the GSO by the degree can soften the distortion introduced by the non-uniform sampling, at least locally in areas where $\mu(\mathcal{N}(x))$ is slowly varying. This suggests that the degree of a node is a good input feature for a method that learns the sampling density.

2.5 Learning the Sampling Density for Geometric Graphs with Hubs

When designing a method to estimate the sampling density from the graph data, the degree on its own is not a sufficient input parameter. The reason is that the degree of a node has two main contributions: the sampling density and the neighborhood radius. Indeed, the degree of a node can be large either due to a large underlying neighborhood radius or due to high density samples (or both). In general, the problem of decoupling the two contribution is difficult. However, if the sampling density is slowly varying, and if the neighborhood radius is piecewise constant, the problem becomes easier. Intuitively, a slowly varying sampling density causes a slight change in the degree of adjacent nodes, while a sudden change in the degree is caused by a radius jump. This motivates our model of *geometric graphs with hubs*.

Definition 5 (Geometric Graph With Hubs)**.** A *geometric graph with hubs* is a random graph with non-uniform geometric GSO, sampled from a metric-probability space $(\mathcal{S}, d, \mu)$ with neighborhood model $\mathcal{N}$, where the sampling density ρ is Lipschitz continuous in $\mathcal{S}$ and $\mu(\mathcal{N}(x))$ is piece-wise constant.

We call this model a geometric graph with hubs since we typically assume that $\mu(\mathcal{N}(x))$ has a low value for most points $x \in \mathcal{S}$, while only a few small regions, called *hubs*, have large neighborhoods. In Section (3.1) we exhibit that geometric graphs with hubs can, indeed, model real-world graphs. To validate this, we train a graph auto-encoder on common citation networks, where the decoder is restricted to be a geometric graph with hubs. The fact that such decoder can achieve low error rates suggests that real-world graphs can often be modeled as geometric graphs with hubs.

In the following, we propose a strategy to assess the sampling density $\boldsymbol{\rho}$. As suggested by the above discussion, the local changes in the degree of the graph give us a lot of information about the local changes in the sampling density and neighborhood radius of geometric graphs with hubs. Hence, we

implement the density estimator as a message passing graph neural network (MPNN) Θ, because it performs local computations and it is equivariant to node indexing, a property that both the density and the degree satisfy. Since we are mainly interested in estimating the inverse of the sampling density, Θ takes as input the inverse of the degree and the inverse of the mean degree of the one-hop neighborhood for all nodes in the graph as two input channels.

However, it is not yet clear how to train Θ. Since in real-world scenarios the ground-truth density is not known, we train Θ in a self-supervised manner. In this context, we choose a task (link prediction, node or graph classification, ...) on a real-world graph $\mathcal{G}$ and we solve it by means of a graph neural network Ψ, referred to as *task network*. Since we want Ψ to depend on the sampling density estimator Θ, we define Ψ as a spectral graph convolution network based on the non-uniform geometric GSO $\mathbf{L}_{\mathcal{G},\Theta(\mathcal{G})}$, e.g., GCN, ChebNet or CayleyNet (Kipf & Welling, 2017; Defferrard et al., 2017; Levie et al., 2019). We, therefore, train Ψ end-to-end on the given task.

The idea behind the proposed method is that the task depends mostly on the underlying continuous model. For example, in shape classification, the label of each graph depends on the surface from which the graph is sampled, rather than the specific intricate structure of the discretization. Therefore, the task network Ψ can perform well if it learns to ignore the particular fine details of the discretization, and focus on the underlying space. The correction of the GSO via the estimated sampling density (Equation (3)) gives the network exactly such power. Therefore, we conjecture that Θ will indeed learn how to estimate the sampling density. In order to verify the previous claim, and to validate our model, we focus on link prediction on synthetic datasets (see Appendix (B)), for which the groundtruth sampling density is known. As shown in Figure (1), the MPNN Θ is able to correctly identify hubs, and correctly predict the ground-truth density in a self-supervised manner.

3 EXPERIMENTS

In the following, we verify experimentally the validity of our model and the feasibility of learning the density. The latter is first demonstrated on synthetic dataset, then applied to real-world graphs both in a transductive (node classification) and inductive (graph classification) setting. Finally, we propose proof-of-concept applications in explainability, learning GSOs, and differentiable pooling.

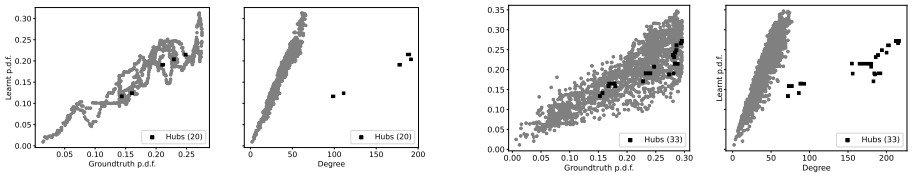
3.1 LINK PREDICTION

The auto-encoder self-supervision method proposed in Section (2.5) is first applied on synthetic datasets of geometric graphs with hubs (see for details Appendices (A.1) to (A.2)). In Figure (1) it is shown that the MPNN is able to correctly predict the value of the sampling density. The top plots shows that the density is well approximated both at hubs and non-hub nodes. Looking at the bottom plots, it is evident that the the density cannot be predicted solely from the degree.

In Figure (2) it is shown that the geometric model is able to effectively represent real-world graphs. Here, we learn an auto-encoder with three types of decoders: inner product, constant neighborhood radius, and piecewise constant neighborhood radius corresponding to a geometric-graph with hubs (see Appendix (A.3) for more details). Better performances are reached if the graph is allowed to be a geometric graph with hubs as in Definition (5). This corroborate the claims that real-world graphs can be better modeled as geometric graphs with varying neighborhood radius. In Figure (6) in Appendix (A.3), we show an example of the learnt probability of being a hub, and the learnt values of the varying radius.

3.2 NODE CLASSIFICATION

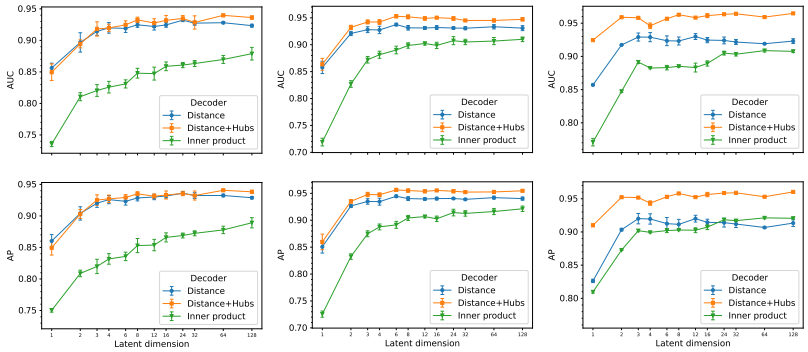
Another exemplary application is the correction of the GSO in a spectral graph convolution network for node classification tasks, as suggested by Equation (3). The details of the implementation are reported in Section (3.2). In Figure (3) we show the accuracy of the best scoring GSO when the density is ignored against the performances of the best scoring GSO when the sampling density is learnt. For Citeseer, both the best GSOs are the symmetric normalized adjacency matrix. For Cora and Pubmed, the best density-ignored GSO is still the symmetric normalized adjacency matrix, while the best density-normalized GSO is the adjacency matrix. This is, once again, a strong validation of



(a) $\mathbb{S}^1$ geometric graph with hubs.

(b) $\mathbb{D}$ geometric graph with hubs.

Figure 1: Example of learnt probability density function on (a) unit-circle geometric graph with hubs and (b) unit disk geometric graph with hubs. (Left) Groundtruth sampling density vs. learnt sampling density at the nodes. (Right) Degree vs. learnt sampling density.



(a) Citeseer

(b) Cora

(c) Pubmed

Figure 2: Test AUC and AP for link prediction on common citation networks as a function of the dimension of the latent space. Performances averaged across 10 runs on each value of the latent dimension.

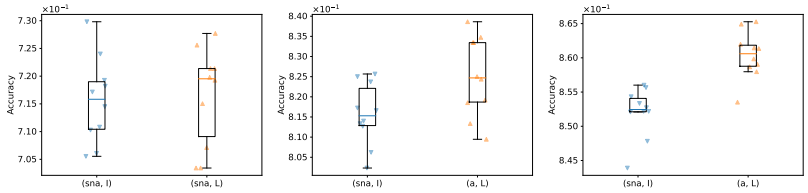
our analysis: if the sampling density is ignored, the best choice is to normalize the Laplacian by the degree to soften the distortion of non-uniform sampling.

3.3 Explainability in Graph Classification

In this experiment, we show how to use the density estimator for explainability, since one could compute which features are mostly correlated with high or low values of $\boldsymbol{\rho}$. The inverse density vector $\boldsymbol{\rho}^{-1}$ can be detached from its intrinsic meaning of sampling density and seen as a way to assign to each node a measure of importance relative to the task in hand. This new way of thinking about $\boldsymbol{\rho}^{-1}$ as importance vector, rather than sampling vector, is helpful, especially when the proposed reasoning is applied to graphs that are far from being Monte-Carlo approximations of metric-probability spaces. We applied this paradigm to the AIDS dataset introduced in Riesen & Bunke (2008), as explained in Appendix (A.2). Figure (4) shows that the classification performances are better if a quota of parameters is used to learn $\boldsymbol{\rho}$ and correct the Laplacian. This means that $\boldsymbol{\rho}$ is an important feature for the classification task, hence, it can be exploited to extract knowledge from the graph. For each element e, we define the mean importance as the sum of all values of $\boldsymbol{\rho}^{-1}$ corresponding to nodes whose label is e divided by the cardinality of e in the dataset. The normalization is useful to reduce the contribution of nodes that have many occurrences in the graphs, such as carbon, which constitute the skeleton of any organic compound. Figure (5) shows the mean average computed on the learnt $\boldsymbol{\rho}^{-1}$ used to correct the Laplacian, and on the one learnt to perform weighted pooling. In both cases, the most important elements are the same; therefore, the two methods seems to be consistent.

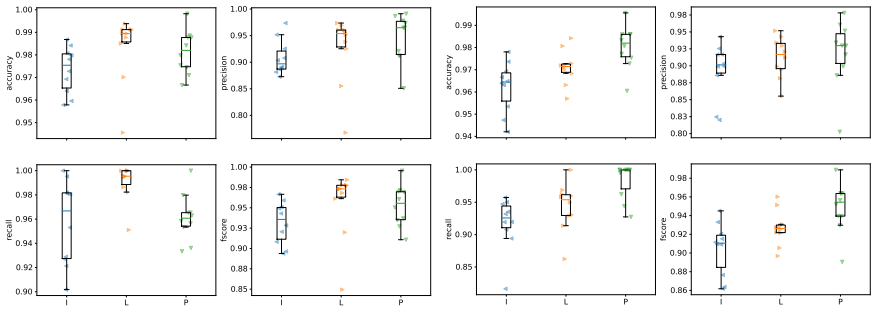
3.4 Differentiable pooling

The learnt $\boldsymbol{\rho}$ on the AIDS dataset can be used not only to correct the Laplacian, but also to perform a better pooling (Appendix (A.2) explains how this is implemented). Usually, a graph convolutional



(a) Citeseer (b) Cora (c) Pubmed

Figure 3: Test accuracy on node classification task on common citation networks. Comparison between the best scoring Laplacian when the density is ignored or learnt. Results averaged across 10 runs: each point represent the performances at each run.



(a) Combinatorial Laplacian (b) Symmetric normalized adjacency

Figure 4: Test metrics of the graph classification task on AIDS dataset, using the combinatorial Laplacian (a) and the symmetric normalized Laplacian (b), averaged over 10 runs. Comparison when the importance $\boldsymbol{\rho}$ is ignored (I), used to correct the Laplacian (L) or used for pooling (P). Each point represent the performance at each run. Learning $\boldsymbol{\rho}$ improves accuracy and reduces its variance; however, in (a) the best performances are reached when $\boldsymbol{\rho}$ is used to correct the Laplacian, in (b) when $\boldsymbol{\rho}$ is used for pooling.

neural network is followed by a global pooling layer in order to extract a representation of the whole graph. A vanilla pooling layer aggregates uniformly the contribution of all nodes. We implemented a weighted pooling layer that takes into account the importance of each node. As shown in Figure (4), the weighted pooling layer can indeed improve performances on the graph classification task.

4 Related Work

Geometric graphs have a long history, dating back to the 60s (Gilbert, 1961), and they have been extensively used to model complex spatial networks (Barthelemy, 2011). One of the very first model is the *random geometric graph* (Penrose, 2003), where the latent space is a Euclidean unit square. Various generalizations and modifications of the previous model have been proposed, such as *random rectangular graphs* (Estrada & Sheerin, 2015), *random spherical graphs* (Allen-Perkins, 2018), and *random hyperbolic graphs* (Krioukov et al., 2010) for which the underlying spaces are respectively a unit rectangle, a hypersphere and a hyperbolic disk.

Geometric graphs are particularly useful since they can be easily generalized to model properties shared by real-world networks. For instance, real-world networks are *small world* and *scale free*. The former is the property for which, despite the large number of nodes, any pair of nodes are just a few edges distant. The latter is the empirical observation that the degree follows a power law distribution: intuitively, just a little number of nodes has a lot of connections while the majority of them has small neighborhoods. These two properties are intimately related to the presence of *hubs* which are nodes with a huge amount of connections. Some random generative models specifically aim to replicate such features: the Watts-Strogatz model (Watts & Strogatz, 1998) creates small-world networks adding long range connections, but it fails to exhibit a plausible degree distribution; the Barabási-Albert model (Albert & Barabasi, 2001) is able to reproduce both features using pref-

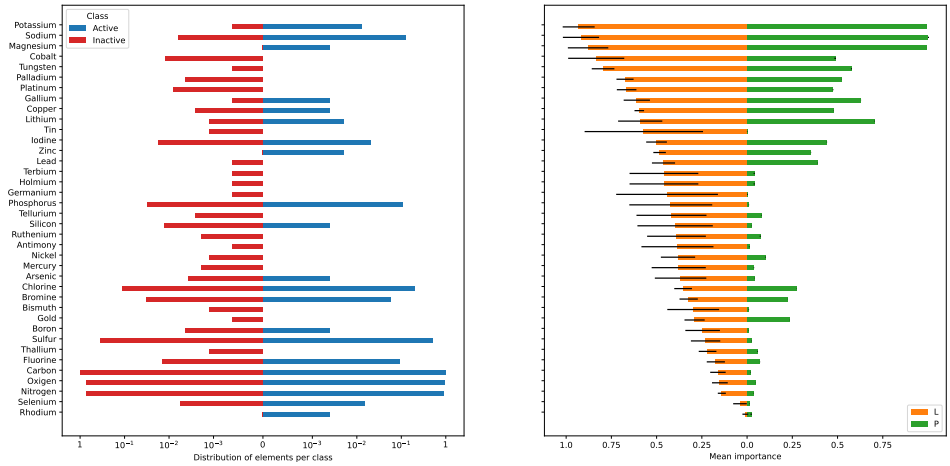


Figure 5: (Left) Distribution of chemical elements per class, i.e. proportion of compounds with that particular element in each class. (Right) mean importance of each element (right) on AIDS dataset when $\boldsymbol{\rho}^{-1}$ is used to correct the GSO (L) and when it is used for weighted pooling (P). Carbon, oxigen and nitrogen are in almost all compounds; therefore, their mean importance is little.

erential attachment. In a geometric graph, a hub can be simply modeled as a node with a larger neighborhood radius. Real-world networks also shows a community structure (*high clustering*) and a tendency for similar nodes to be connected (*assortativity*). The stochastic block model (Holland et al., 1983) is able to model the above-mentioned properties partitioning the nodes in communities and specifying the inter- and intra-community linking probabilities. A geometric graph reproduces such properties in a different fashion: assortativity is intrinsically modeled by the fact that near (hence, similar) points are connected, while communities are modeled as densely sampled areas.

Barring a few exceptions non-uniformity is rarely considered. Iyer & Thacker (2012) study a class of non-uniform random geometric graphs where the radii depend on the location. Martínez-Martínez et al. (2022) study non-uniform graphs on the plane with the density functions specified in polar coordinates. Pratt et al. (2018) consider temporal connectivity in finite networks with non-uniform measures. Janssen et al. (2016) study the behaviour of the Spatial Preferential Attachment (SPA) model with non-uniform distribution of nodes. In all these works the focus is on simple (asymptotic) topological properties of the graphs such as the average degree and the number of isolated nodes. In this work we focus on deriving the corresponding GSO and we explicitly correct for the non-uniformity.

Conclusions

In this paper we addressed the problem of learning the latent sampling density by which graphs are sampled from their underlying continuous models. We developed formulas for representing graphs in view of their connectivity structure and sampling density, namely, using non-uniform geometric GSOs. We then showcased how the density of geometric graphs with hubs can be estimated using self-supervision, and validated our approach experimentally. Last, we showed how knowing the sampling density can help with various tasks, e.g., improving spectral methods, improving pooling, and gaining knowledge from graphs. One limitation in our methodology is the difficulty in validating that real-wold graphs are indeed sampled from latent geometric spaces. While we reported experiments that support this modeling assumption, an important future direction is to develop further experiments and tools to further support our model. For instance, can we learn a density estimator on one class of graphs and transfer it to another? Can we use ground truth demographic data to validate the estimated density in social networks? We believe that future research will shed light on these questions and find new ways to exploit the sampling density for various applications.

REFERENCES

Reka Albert and Albert-Laszlo Barabasi. Statistical mechanics of complex networks. 2001. doi: 10.48550/ARXIV.COND-MAT/0106096.

Alfonso Allen-Perkins. Random spherical graphs. *Physical Review E*, 98(3):032310, September 2018. ISSN 2470-0045, 2470-0053. doi: 10.1103/PhysRevE.98.032310.

Marc Barthelemy. Spatial Networks. *Physics Reports*, 499(1-3):1–101, February 2011. ISSN 03701573. doi: 10.1016/j.physrep.2010.11.002.

Dmitri Burago, Sergei Ivanov, and Yaroslav Kurylev. Spectral stability of metric-measure Laplacians. *arXiv:1506.06781 [math]*, August 2018.

Eunjoon Cho, Seth A. Myers, and Jure Leskovec. Friendship and mobility: User movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '11*, pp. 1082, San Diego, California, USA, 2011. ACM Press. ISBN 978-1-4503-0813-7. doi: 10.1145/2020408.2020579.

Fan R. K. Chung. *Spectral Graph Theory*. Number no. 92 in Regional Conference Series in Mathematics. Published for the Conference Board of the mathematical sciences by the American Mathematical Society, Providence, R.I, 1997. ISBN 978-0-8218-0315-8.

Dragos Cvetkovic and Slobodan Simic. Towards a spectral theory of graphs based on the signless Laplacian, I. *Publications de l'Institut Mathematique*, 85(99):19–33, 2009. ISSN 0350-1302, 1820-7405. doi: 10.2298/PIM0999019C.

George Dasoulas, Johannes Lutzeyer, and Michalis Vazirgiannis. Learning Parametrised Graph Shift Operators. *arXiv:2101.10050 [cs, stat]*, April 2021.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. *arXiv:1606.09375 [cs, stat]*, February 2017.

Ernesto Estrada and Matthew Sheerin. Random Rectangular Graphs. *Physical Review E*, 91(4): 042805, April 2015. ISSN 1539-3755, 1550-2376. doi: 10.1103/PhysRevE.91.042805.

E. N. Gilbert. Random Plane Networks. *Journal of the Society for Industrial and Applied Mathematics*, 9(4):533–543, December 1961. ISSN 0368-4245, 2168-3484. doi: 10.1137/0109045.

Luca Gugelmann, Konstantinos Panagiotou, and Ueli Peter. Random Hyperbolic Graphs: Degree Sequence and Clustering. *arXiv:1205.1470 [physics]*, May 2012.

Matthias Hein, Jean-Yves Audibert, and Ulrike von Luxburg. Graph Laplacians and their convergence on random neighborhood graphs, June 2007.

Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109–137, June 1983. ISSN 03788733. doi: 10.1016/0378-8733(83) 90021-7.

Yanqing Hu, Yong Li, Zengru Di, and Ying Fan. Navigation in non-uniform density social networks, January 2011.

Srikanth K. Iyer and Debleena Thacker. Nonuniform random geometric graphs with location-dependent radii. *The Annals of Applied Probability*, 22(5), October 2012. ISSN 1050-5164. doi: 10.1214/11-AAP823.

Jeannette Janssen, Paweł Prałat, and Rory Wilson. Nonuniform Distribution of Nodes in the Spatial Preferential Attachment Model. *Internet Mathematics*, 12(1-2):121–144, March 2016. ISSN 1542-7951, 1944-9488. doi: 10.1080/15427951.2015.1110543.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, January 2017.

Thomas N. Kipf and Max Welling. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv:1609.02907 [cs, stat]*, February 2017.

Dmitri Krioukov, Fragkiskos Papadopoulos, Maksim Kitsak, Amin Vahdat, and Marián Boguñá. Hyperbolic geometry of complex networks. *Physical Review E*, 82(3):036106, September 2010. ISSN 1539-3755, 1550-2376. doi: 10.1103/PhysRevE.82.036106.

Ron Levie, Federico Monti, Xavier Bresson, and Michael M. Bronstein. CayleyNets: Graph Convolutional Neural Networks With Complex Rational Spectral Filters. *IEEE Transactions on Signal Processing*, 67(1):97–109, January 2019. ISSN 1053-587X, 1941-0476. doi: 10.1109/TSP.2018.2879624.

Ron Levie, Wei Huang, Lorenzo Bucci, Michael M. Bronstein, and Gitta Kutyniok. Transferability of Spectral Graph Convolutional Neural Networks. *arXiv:1907.12972 [cs, stat]*, March 2020.

David Liben-Nowell, Jasmine Novak, Ravi Kumar, Prabhakar Raghavan, and Andrew Tomkins. Geographic routing in social networks. *Proceedings of the National Academy of Sciences*, 102 (33):11623–11628, August 2005. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.0503018102.

László Lovász. *Large Networks and Graph Limits*, volume 60 of *Colloquium Publications*. American Mathematical Society, Providence, Rhode Island, December 2012. ISBN 978-0-8218-9085-1 978-1-4704-1583-9. doi: 10.1090/coll/060.

C. T. Martínez-Martínez, J. A. Méndez-Bermúdez, Francisco A. Rodrigues, and Ernesto Estrada. Nonuniform random graphs on the plane: A scaling study. *Physical Review E*, 105(3):034304, March 2022. ISSN 2470-0045, 2470-0053. doi: 10.1103/PhysRevE.105.034304.

Sohir Maskey, Ron Levie, and Gitta Kutyniok. Transferability of Graph Neural Networks: An Extended Graphon Approach, June 2022.

Gonzalo Mateos, Santiago Segarra, Antonio G. Marques, and Alejandro Ribeiro. Connecting the Dots: Identifying Network Structure via Graph Signal Processing, October 2018.

Mathew Penrose. *Random Geometric Graphs*. Number 5 in Oxford Studies in Probability. Oxford University Press, Oxford ; New York, 2003. ISBN 978-0-19-850626-3.

Pete Pratt, Carl P. Dettmann, and Woon Hau Chin. Temporal connectivity in finite networks with nonuniform measures. *Physical Review E*, 98(5):052310, November 2018. ISSN 2470-0045, 2470-0053. doi: 10.1103/PhysRevE.98.052310.

Kaspar Riesen and Horst Bunke. IAM Graph Database Repository for Graph Based Pattern Recognition and Machine Learning. In Niels da Vitoria Lobo, Takis Kasparis, Fabio Roli, James T. Kwok, Michael Georgiopoulos, Georgios C. Anagnostopoulos, and Marco Loog (eds.), *Structural, Syntactic, and Statistical Pattern Recognition*, volume 5342, pp. 287–297. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008. ISBN 978-3-540-89688-3 978-3-540-89689-0. doi: 10.1007/978-3-540-89689-0_33.

Luana Ruiz, Luiz F. O. Chamon, and Alejandro Ribeiro. Graphon Neural Networks and the Transferability of Graph Neural Networks. *arXiv:2006.03548 [cs, stat]*, October 2020.

Hichem Sahbi. Learning Laplacians in Chebyshev Graph Convolutional Networks. In *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pp. 2064–2075, Montreal, BC, Canada, October 2021. IEEE. ISBN 978-1-66540-191-3. doi: 10.1109/ICCVW54120.2021.00234.

Ulrike von Luxburg, Mikhail Belkin, and Olivier Bousquet. Consistency of spectral clustering. *The Annals of Statistics*, 36(2), April 2008. ISSN 0090-5364. doi: 10.1214/009053607000000640.

Pu Wang and Marta C. González. Understanding spatial connectivity of individuals with nonuniform population density. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1901):3321–3329, August 2009. ISSN 1364-503X, 1471-2962. doi: 10.1098/rsta.2009.0089.

Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic Graph CNN for Learning on Point Clouds, June 2019.

Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, June 1998. ISSN 0028-0836, 1476-4687. doi: 10.1038/30918.

A IMPLEMENTATION DETAILS

A.1 SYNTHETIC DATASET GENERATION

This section explains how to generate a synthetic dataset of geometric graphs with hubs. We first consider a metric space. For our experiments, we mainly focused on the unit-circle $\mathbb{S}^1$ and on the unit-disk $\mathbb{D}$ (see Appendix (B) for more details). Each graph is generated as follows. First, a non-uniform distribution is randomly generated. We considered an angular non-uniformity as described in Definition (6), where the number of oscillating terms, as well as the parameters $\boldsymbol{c}$, $\boldsymbol{n}$, $\boldsymbol{\mu}$, are chosen randomly. In the case of 2-dimensional spaces the radial distribution is the one shown in Table (1). According to each generated probability density function, N points $\{x_i\}_{i=1}^N$ are drawn independently. Among them, $m < N$ are chosen randomly to be hubs, and any other node whose distance from a hub is less than some $\varepsilon > 0$ is also marked as hub. We consider two parameters α, $\beta > 0$. The neighborhood radius about non-hub nodes is taken to be α, and $\alpha + \beta$ for hub nodes. Any two points are then connected if

$$d(x_i, x_j) \leq \max\{r(x_i), r(x_j)\}\,,\ r(x) = \begin{cases} \alpha & x \text{ is non-hub} \\ \alpha + \beta & x \text{ is hub} \end{cases}.$$

In practical terms, α is computed such that the resulting graph is strongly connected, hence, it differs from graph to graph; β is set to be $3\,\alpha$ and ϵ to be $\alpha/10$.

A.2 DENSITY ESTIMATION WITH SELF-SUPERVISION

Density Estimation Network In our experiments, the inverse of the sampling density, $1/\rho$, is learnt by means of an EdgeConv neural network Θ (Wang et al., 2019), in the following referred to as PNet, implementing a MLP, with $\max(\cdot)$ aggregation and $\mathrm{abs}(\cdot)$ non-linearity. The number of hidden layers, hidden channels and output channels is 3, 32 and 1, respectively. Since the degree is an approximation of the sampling density, as stated in Lemma (1), and since we are interested on computing its inverse to correct the Laplacian, the input of PNet is the inverse of the degree and the inverse of the mean degree of the one-hop neighborhood. Justified by the Monte-Carlo approximation

$$1 = \int_{\mathcal{S}} \mathrm{d}\mu(y) = \int_{\mathcal{S}} \rho(y)^{-1}\,\mathrm{d}\nu(y) \approx N^{-1} \sum_{i=1}^{N} \rho(x_i)^{-1}\,,\ x_i \sim \rho\ \forall i = 1, \ldots, N\,,$$

the output of PNet is normalized by its mean.

Self-Supervision of the PNet via Link Prediction on Synthetic Dataset To train the PNet Θ, for each graph $\mathcal{G}$, we use $\Theta(\mathcal{G})$ to define a GSO $\mathbf{L}_{\mathcal{G},\Theta(\mathcal{G})}$. Then, we define a graph autoencoder, where the encoder is implemented as a spectral graph convolution network with GSO $\mathbf{L}_{\mathcal{G},\Theta(\mathcal{G})}$. The decoder is the usual inner-product. The graph signal has chosen to be a slice of 20 random columns of the adjacency matrix. The number of hidden channels, hidden layers, output channels is respectively 32, 2, 2. For each node j, the network outputs a feature $\Theta(\mathcal{G})_j$ in $\mathbb{R}^n$. Here, $\mathbb{R}^n$ is seen as the metric space underlying the NuG. In our experiments (Section (3.1)), we choose $n = 2$. Some results are shown in Figure (1).

Node Classification Let $\mathcal{G}$ be the real-world graph. In Section (3.2), we considered $\mathcal{G}$ to be Citeseer, Cora or Pubmed. The task network Ψ is a polynomial convolutional neural network implementing a GSO $\mathbf{L}_{\mathcal{G},\Theta(\mathcal{G})}$, where Θ is the PNet; the order of the polynomial spectral filters is 1, the number of hidden channels 32 and the number of hidden layers 2. The optimizer is ADAM (Kingma & Ba, 2017) with learning rate 10^{-2}. We split the nodes in training (85%), validation (5%) and test (10%) in a stratified fashion, and apply early stopping. The performances of the method are shown in Figure (3).

Graph Classification Let $\mathcal{G}$ be the real-world graph. In Section (3.3), $\mathcal{G}$ is any compound in the AIDS dataset. The task network Ψ is a polynomial convolutional neural network implementing a GSO $\mathbf{L}_{\mathcal{G},\Theta(\mathcal{G})}$, where Θ is the PNet; the order of the spectral polynomial filters is 1, the number

of hidden channels 128 and the number of hidden layers 2. The optimizer is ADAM with learning rate 10^{-2}. We perform a stratified splitting of the graphs in training (85%), validation (5%) and test (10%), and applied early stopping. The chosen batch size is 64. The pooling layer is a global add layer.

In case of weighted pooling as in Section (3.4), the task network Ψ implements as GSO $\mathbf{L}_{\mathcal{G},\mathbf{1}}$, while Θ is used to output the weights of the pooling layer. The performance metrics of both approaches are shown in Figure (4)

A.3 Geometric graphs with Hubs auto-encoder

Here, we validate that real-world graphs can be modeled approximately as geometric graphs with hubs, as claimed in Section (3.1). We consider common citation networks, such as Citeseer, Cora and Pubmed. Let $\mathcal{G}$ be the real-world graph with N nodes, F features; let $\mathbf{X} \in \mathbb{R}^{N\times F}$ be the feature matrix. First, to learn geometric graphs without hubs, we define a graph-autoencoder with encoder Ψ as defined at the end of this subsection, and decoder implemented as follows. Let $\Psi(\mathbf{X})_i$, $\Psi(\mathbf{X})_j \in \mathbb{R}^n$ be the embeddings of node i and node j respectively; the decoder computes

$$\sigma\left(\alpha - \|\Psi(\mathbf{X})_i - \Psi(\mathbf{X})_j\|_2\right) ,$$

where $\sigma(\cdot)$ is the logistic sigmoid function, and α is the trainable radius. Such decoder is referred to as *distance-decoder*. The previous formula is justified by the fact that the condition $\|\Psi(\mathbf{X})_i - \Psi(\mathbf{X})_j\|_2 \leq \alpha$ can be rewritten using heaviside function as $\mathrm{H}(\alpha - \|\Psi(\mathbf{X})_i - \Psi(\mathbf{X})_j\|_2)$. To guarantee the model is differentiable, the heavisde function has been replaced by the logistic sigmoid.

In order to take into account the presence of hubs, a neural network Υ with same architecture as PNet, has been implemented to output the probability that each node is a hub. The corresponding decoder is defined as

$$\sigma\left(\alpha + \max\{\Upsilon(\mathcal{G})_i, \Upsilon(\mathcal{G})_j\}\beta - \|\Psi(\mathbf{X})_i - \Psi(\mathbf{X})_j\|_2\right) .$$

and referred to as *distance+hubs-decoder*. Here, both α and β are trainable. In order to guarantee that $0 \leq \Upsilon(\mathcal{G})_j \leq 1$ the network is followed by a min-max normalization.

The encoder Ψ is a polynomial spectral graph convolutional neural network implementing as GSO the symmetric normalized adjacency matrix; the order of the polynomial filters is 1, the number of hidden channels 32 and the number of hidden layers 2. In the case of distance-decoder, the loss is the cross entropy of existing and non-existing edges. In the case of distance+hubs-decoder we also add $\|\Upsilon(\mathcal{G})\|_1/N$ to the loss, as a regularization term, since in our model we suppose the number of hubs is low. The optimizer is ADAM with learning rate 10^{-2}. We split the edges in training (85%), validation (5%) and test (10%), and apply early stopping. The performances of both methods are shown in Figure (2), while examples of embeddings and learnt α and β are shown in Figure (6).

B Synthetic Datasets - a blueprint

In the following, we consider some simple latent metric spaces, and construct methods for randomly generating non-uniform samples. For each space, structural properties of the corresponding NuG are studied, such as the expected degree of a node and the expected average degree, in case the radius is fixed and the sampling is non-uniform. All proofs can be found in Appendix (D), if not otherwise stated.

Three natural metric measure spaces are the euclidean, spherical and hyperbolic spaces. If we restrict the attention to 2-dimensional spaces, a way to uniformly sample is summarized in Table (1). In all three cases, the *radial* component arises naturally from the measure of the space. A possible way to introduce non-uniformity is changing the *angular* distribution. In this way, preferential direction will be identified, leading to an anisotropic model.

Definition 6. Given a natural number $C \in \mathbb{N}$, and vectors $\boldsymbol{c} \in \mathbb{R}^C$, $\boldsymbol{n} \in \mathbb{N}^C$, $\boldsymbol{\mu} \in \mathbb{R}^C$ the function

$$\mathrm{sb}(\theta; \boldsymbol{c}, \boldsymbol{n}, \boldsymbol{\mu}) = \frac{1}{B}\sum_{i=1}^{C} c_i \cos(n_i(\theta - \mu_i)) + \frac{A}{B} ,\ A = \sum_{i=1}^{C}|c_i| ,\ B = 2\pi\left(\sum_{i=1}^{C}|c_i| + \sum_{i:n_i=0} c_i\right) ,$$

is a continuous, 2π-periodic probability density function. It will be referred to as *spectrally bounded*.

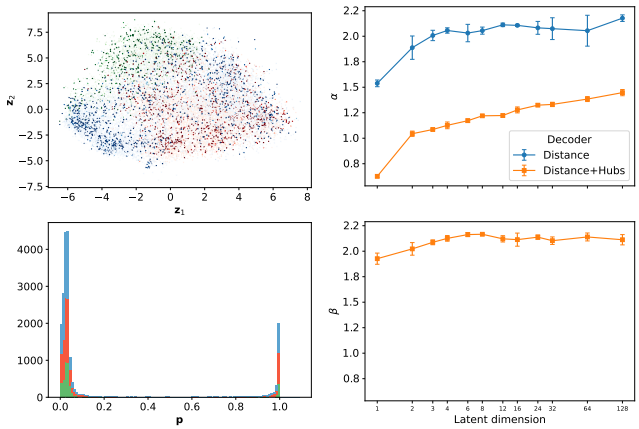


Figure 6: (Left) Embedding of Pubmed dataset in 2 dimensions when the probability of hubs $\mathbf{p}$ is learnt, and distribution of $\mathbf{p}$ in each class. The intensity of the color for node i is proportional to p_i. (Right) Learnt values of α and β on Pubmed dataset as a function of the latent dimension. Results averaged across 10 runs for each value of the latent dimension. The average probability of being a hub is 19.06%, the number of nodes with a probability of being a hub greater than 0.99 is the 10.10%.

The cosine can be replaced by a generic 2π-periodic function; the only change in the construction will be the offset and the normalization constant.

Definition 7. Given a natural number $C \in \mathbb{N}$, and the vectors $\boldsymbol{c} \in \mathbb{R}^C$, $\boldsymbol{n} \in \mathbb{N}^C$, $\boldsymbol{\mu} \in \mathbb{R}^C$, $\boldsymbol{\kappa} \in \mathbb{R}^C_{\geq 0}$, the function

$$\mathrm{mvM}(\theta; \boldsymbol{c}, \boldsymbol{n}, \boldsymbol{\mu}, \boldsymbol{\kappa}) = \frac{1}{B} \sum_{i=1}^{C} c_i \, \frac{\exp(\kappa_i \cos(n_i \, (\theta - \mu_i))}{2 \, \pi \, \mathrm{I}_0(\kappa_i)} + \frac{A}{B} \, ,$$

where

$$A = \sum_{i : c_i < 0} c_i \, \frac{\exp(\kappa_i)}{2 \, \pi \, \mathrm{I}_0(\kappa_i)} \, , \; B = \sum_{i : n_i \geq 1} c_i + \sum_{i : n_i = 0} c_i \, \frac{\exp(\kappa_i)}{\mathrm{I}_0(\kappa_i)} + \sum_{i : c_i < 0} c_i \, \frac{\exp(\kappa_i)}{\mathrm{I}_0(\kappa_i)} \, ,$$

is a continuous, 2π-periodic probability density function. It will be referred to as *multimodal von Mises*.

Both densities introduced previously can be thought as functions over the unit circle. Hence, the very first space to be studied is $\mathbb{S}^1 = \{\mathbf{x} \in \mathbb{R}^2 : \|x\| = 1\}$ equipped with geodesic distance. As shown in the next proposition, the geodesic distance can be computed in a fairly easy way.

Proposition 2. *Given two points $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{S}^1$ corresponding to the angles $x, y \in [-\pi, \pi)$, their geodesic distance is equal to*

$$d(\boldsymbol{x}, \boldsymbol{y}) = \pi - |\pi - |x - y|| \, .$$

The next proposition computes the degree of a node in a non-uniform unit circle graph.

Proposition 3. *Given a spectrally bounded probability density function as in Definition (6), the expected degree of a node θ in a unit circle geometric graph with neighborhood radius α is*

$$\deg(\theta) = \frac{2 \, N}{B} \left(\sum_{i : n_i \neq 0} \frac{c_i}{n_i} \, \cos(n_i \, (\theta - \mu_i)) \, \sin(n_i \, \alpha) + \left(\sum_{i : n_i = 0} c_i + A \right) \, \alpha \right) \, ,$$

and the expected average degree of the whole graph is

$$\mathbb{E}[\deg(\theta)] = \frac{2 \, \pi \, N \, \alpha}{B^2} \left(\sum_{i : n_i \neq 0} \sum_{j : n_i = n_j} c_i c_j \, \cos(n_i \, (\mu_i - \mu_j)) \frac{\sin(n_i \, \alpha)}{n_i \, \alpha} + 2 \left(\sum_{i : n_i = 0} c_i + A \right)^2 \right) \, .$$

As a direct consequence, in the limit of r going to zero

$$\lim_{\alpha\to 0^+} \frac{\mathbb{P}[\mathrm{B}_\alpha(\theta)]}{2\,\alpha} = \frac{1}{B}\left(\sum_{i:n_i\neq 0} c_i\,\cos(n_i\,(\theta-\mu_i))\,\left(\lim_{\alpha\to 0^+}\frac{\sin(n_i\,\alpha)}{n_i\,\alpha}\right) + \left(\sum_{i:n_i=0} c_i + A\right)\right)$$
$$= \mathrm{sb}(\theta;\boldsymbol{c},\boldsymbol{n},\boldsymbol{\mu})$$

thus, for sufficiently small α, the probability of a ball centered at θ is proportional to the density computed in θ. Moreover, the error can be computed as

$$\left|\mathrm{sb}(\theta;\boldsymbol{c},\boldsymbol{n},\boldsymbol{\mu}) - \frac{\mathbb{P}[\mathrm{B}_\alpha(\theta)]}{2\,\alpha}\right| \leq \frac{1}{6\,B}\left(\sum_{i=1}^{C} n_i^2\,c_i\right)\alpha^2\,,$$

which shows that the approximation worsen the more oscillatory terms there are. In the case of multimodal von Mises distribution, a closed formula for the probability of balls does not exists. The following proposition introduces an approximation based solely on cosine functions.

Proposition 4. *A multimodal von Mises probability density function can be approximated by a spectrally bounded one.*

The previous result, combined with Proposition (3), gives a way to approximate the expected degree of spatial networks sampled accordingly to a multimodal von Mises angular distribution. However, the computation is straightforward when $\boldsymbol{n}$ is the constant vector $n\,\mathbf{1}$, since the product of two von Mises pdfs is the kernel of a von Mises pdf

$$\frac{\exp(\boldsymbol{\kappa}_1\cos(n(\theta-\boldsymbol{\mu}_1)))}{2\,\pi\;\mathrm{I}_0(\boldsymbol{\kappa}_1)}\frac{\exp(\boldsymbol{\kappa}_2\cos(n(\theta-\boldsymbol{\mu}_2)))}{2\,\pi\;\mathrm{I}_0(\boldsymbol{\kappa}_2)}$$
$$= \frac{\exp\left(\sqrt{\boldsymbol{\kappa}_1^2+\boldsymbol{\kappa}_2^2+2\boldsymbol{\kappa}_1\boldsymbol{\kappa}_2\cos(n(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2))}\cos\left(n(\theta-\varphi)\right)\right)}{4\,\pi^2\;\mathrm{I}_0(\boldsymbol{\kappa}_1)\,\mathrm{I}_0(\boldsymbol{\kappa}_2)}\,,$$

where

$$\varphi = \arctan\left(\frac{\boldsymbol{\kappa}_1\sin(\boldsymbol{\mu}_1)+\boldsymbol{\kappa}_2\sin(\boldsymbol{\mu}_2)}{\boldsymbol{\kappa}_1\cos(\boldsymbol{\mu}_1)+\boldsymbol{\kappa}_2\cos(\boldsymbol{\mu}_2)}\right)\,.$$

The unit circle model is preparatory to the study of more complex spaces, for instance the unit disk $\mathbb{D} = \{\mathbf{x}\in\mathbb{R}^2 : \|\mathbf{x}\|\leq 1\}$ equipped with geodesic distance, as in Table (1).

Proposition 5. *Given a spectrally bounded angular distribution as in Definition (6), the degree of a node (r,θ) in a unit disk geometric graph with neighborhood radius α is*

$$\deg(r,\theta) \approx 2\,\pi\,\alpha^2\,N\,\mathrm{sb}(\theta;\boldsymbol{c},\boldsymbol{n},\boldsymbol{\mu})\,,$$

and the average degree of the whole network is

$$\mathbb{E}[\deg(r,\theta)] \approx \frac{2\,\pi^2\,\alpha^2\,N}{B^2}\left(\sum_{i:n_i\neq 0}\sum_{j:n_i=n_j} c_i c_j\;\cos(n_i\,(\mu_i-\mu_j)) + 2\left(\sum_{i:n_i=0} c_i + A\right)^2\right)\,.$$

Figure (7a) shows some examples of non-uniform sampling of the unit disk. The last example will be the hyperbolic disk $\mathbb{H} = \{\mathbf{x}\in\mathbb{R}^2 : \|\mathbf{x}\|\leq R, R\gg 1\}$, equipped with geodesic distance as in Table (1).

Proposition 6. *Given a spectrally bounded angular distribution as in Definition (6), the degree of a node (r,θ) in a hyperbolic geometric graph with neighborhood radius α is*

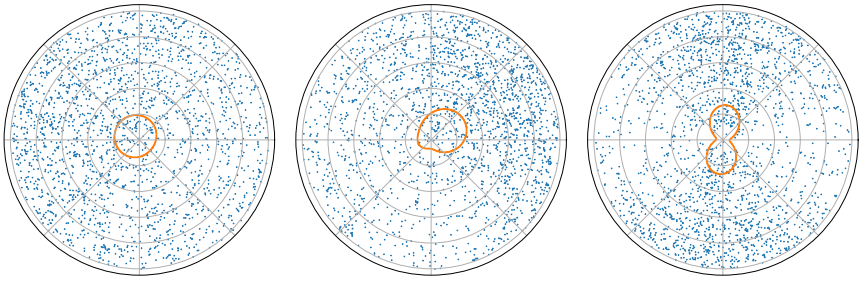
$$\deg(r,\theta) \approx 8\,N\,e^{\frac{\alpha-R-r}{2}}\;\mathrm{sb}(\theta;\boldsymbol{c},\boldsymbol{n},\boldsymbol{\mu})\,,$$

and the average degree of the whole network is $\mathcal{O}(N\,e^{\frac{\alpha-2R}{2}})$.

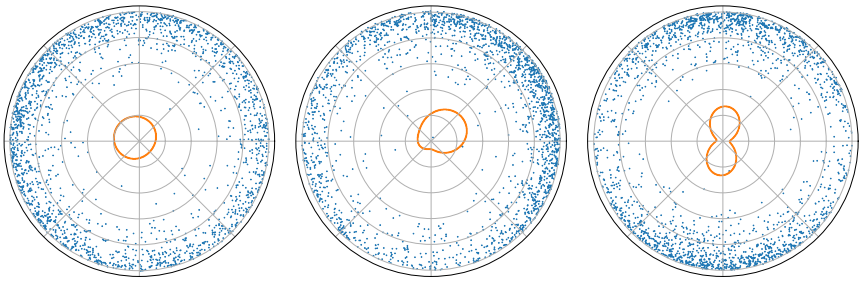
The proof can be found in Appendix (D). The computed approximation is in line with the findings of Krioukov et al. (2010), where a closed formula for the uniform case is provided when $\alpha = R$. To the best of our knowledge, this is the first work that considers $\alpha\neq R$. Examples of non-uniform sampling of the hyperbolic disk are shown in Figure (7b).

Table 1: Properties of euclidean, spherical and hyperbolic spaces of dimension 2. In the case of euclidean and hyperbolic spaces, the uniform distribution refers to a disk of radius R.

Property	Geometry	
Measure of a ball of radius α	euclidean	$\pi\,\alpha^2$
	spherical	$2\,\pi\,(1-\cos(\alpha))$
	hyperbolic	$2\,\pi\,(\cosh(\alpha)-1)$
Uniform p.d.f.	euclidean	$(2\,\pi)^{-1}\mathbb{1}_{[-\pi,\pi)}(\theta)2R^{-2}r\mathbb{1}_{[0,R)}(r)$
	spherical	$(2\,\pi)^{-1}\mathbb{1}_{[-\pi,\pi)}(\theta)2^{-1}\sin(\varphi)\mathbb{1}_{[0,\pi)}(\varphi)$
	hyperbolic	$(2\,\pi)^{-1}\mathbb{1}_{[-\pi,\pi)}(\theta)(\cosh(R)-1)^{-1}\sinh(r)\mathbb{1}_{[0,R)}(r)$
Distance in polar coordinates	euclidean	$\sqrt{r_1^2+r_2^2-2r_1r_2\cos(\theta_1-\theta_2)}$
	spherical	$\arccos\big(\cos(\phi_1)\,\cos(\phi_2)+\sin(\phi_1)\,\sin(\phi_2)\,\cos(\theta_1-\theta_2)\big)$
	hyperbolic	$\operatorname{arccosh}(\cosh(r_1)\cosh(r_2)-\sinh(r_1)\sinh(r_2)\cos(\theta_1-\theta_2))$



(a) Non-uniform sampling of euclidean disk.



(b) Non-uniform sampling of hyperbolic disk.

Figure 7: Examples of non-uniform (a) euclidean and (b) hyperbolic sampling. The orange curve represent the angular probability density function, conveniently rescaled for visibility purposes.

C Retrieving and Building GSOs

In the current section, we first show how to retrieve the usual definition of graph shift operators from Definition (4), and then how Definition (4) can be used to create novel GSOs. For simplicity, for both goals we suppose uniform sampling $\boldsymbol{\rho} = \mathbf{1}$; Equation (3) can be rewritten as

$$\begin{aligned}\mathbf{L}_{\mathcal{G},\mathbf{1}} = & N^{-1} \operatorname{diag}\left(m^{(1)}\left(N^{-1}\mathbf{d}\right)\right) \mathbf{A} \operatorname{diag}\left(m^{(2)}\left(N^{-1}\mathbf{d}\right)\right) \\ & - N^{-1} \operatorname{diag}\left(\operatorname{diag}\left(m^{(3)}\left(N^{-1}\mathbf{d}\right)\right) \mathbf{A} \operatorname{diag}\left(m^{(4)}\left(N^{-1}\mathbf{d}\right)\right) \mathbf{1}\right)\end{aligned} \tag{4}$$

where $\mathbf{A}$ is the adjacency matrix and $\mathbf{d}$ is the degree vector. Table (2) exhibit which choice of $\{m^{(i)}\}_{i=1}^{4}$ correspond to which graph Laplacian.

A question that may arise is whether the innermost $\operatorname{diag}(\cdot)$ in Equation (4) can be factored out of the outermost one. As shown in the next proposition, it is not possible in general.

Proposition 7. *Let $\mathbf{A} \in \mathbb{R}^{N\times N}$, $\mathbf{A} = \mathbf{A}^{\mathrm{T}}$; let $\mathbf{v} \in \mathbb{R}^{N}_{\geq 0}$ and $\mathbf{V} = \operatorname{diag}(\mathbf{v})$, it holds*

$$\operatorname{diag}(\mathbf{VA1}) = \mathbf{V}\ \operatorname{diag}(\mathbf{A1}) = \operatorname{diag}(\mathbf{A1})\,\mathbf{V}\,.$$

Moreover

$$\operatorname{diag}(\mathbf{AV1}) = \operatorname{diag}(\mathbf{VA1}) \iff A_{i,j} = 0\ \forall\, i,j = 1,\dots,N\ :\ v_i \neq v_j\,.$$

The proof of the statement can be found in Appendix (D). An important consequence of Proposition (7) is that the graph Laplacian

$$\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}} - \operatorname{diag}\left(\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}\mathbf{1}\right)\ ,\ \mathbf{D} = \operatorname{diag}(\mathbf{d})\,, \tag{5}$$

obtained with $m^{(i)}(x) = x^{-\frac{1}{2}}$ for every $i \in \{1,\dots,4\}$, is in general different from the symmetric normalized Laplacian, since

$$\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}} - \operatorname{diag}\left(\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}\mathbf{1}\right) \neq \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}} - \mathbf{D}^{-\frac{1}{2}}\operatorname{diag}(\mathbf{A1})\,\mathbf{D}^{-\frac{1}{2}} = \mathbf{L}_{sn}\,,$$

In light of Proposition (7), the two Laplacians are equivalent if every node is connected to nodes with same degree, i.e. if the graph is k-regular.

The difference between the two Laplacians can be better seen studying their spectrum. The next proposition introduces an upper bound on the eigenvalues of the Laplacian in Equation (5).

Proposition 8. *Let $\mathcal{G} = (\mathcal{V},\mathcal{E})$ be an undirected graph with adjacency matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$ and degree matrix $\mathbf{D} = \operatorname{diag}(\mathbf{A1})$. Let λ be an eigenvalue of the graph Laplacian*

$$\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}} - \operatorname{diag}\left(\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}\mathbf{1}\right)\ ,$$

it holds $|\lambda| \leq 2\sqrt{N}$.

The proof of the proposition can be found in Appendix (D). It is well known that the spectral radius of the symmetric normalized Laplacian is less than or equal to 2 (Chung, 1997), with equality holding for bipartite graphs. However, this is not the case for the Laplacian in Equation (5), as shown in the next example.

Example 1 (Complete Bipartite Graph)**.** Consider the complete bipartite graph with n nodes in the first part and $m \geq n$ nodes in the second part. Its adjacency and degree matrix are

$$\mathbf{A} = \begin{pmatrix} \mathbf{0}_{n\times n} & \mathbf{1}_{n\times m} \\ \mathbf{1}_{m\times n} & \mathbf{0}_{m\times n} \end{pmatrix}\ ,\ \mathbf{D} = \begin{pmatrix} m\mathbf{I}_{n\times n} & \mathbf{0}_{n\times m} \\ \mathbf{0}_{m\times n} & n\mathbf{I}_{m\times m} \end{pmatrix}\ .$$

A simple computation leads to

$$\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}} - \operatorname{diag}\left(\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}\mathbf{1}\right) = \begin{pmatrix} -m^{\frac{1}{2}}n^{-\frac{1}{2}}\mathbf{I}_{n\times n} & (nm)^{-\frac{1}{2}}\mathbf{1}_{n\times m} \\ (nm)^{-\frac{1}{2}}\mathbf{1}_{m\times n} & -n^{\frac{1}{2}}m^{-\frac{1}{2}}\mathbf{I}_{m\times m} \end{pmatrix}\ .$$

Table 2: Usual graph shift operators as metric-probability Laplacians.

Graph shift operator	$m^{(1)}(x)$	$m^{(2)}(x)$	$m^{(3)}(x)$	$m^{(4)}(x)$
Adjacency	$1(x)$	$1(x)$	$0(x)$	$0(x)$
Combinatorial	$1(x)$	$1(x)$	$1(x)$	$1(x)$
Signless (Cvetkovic & Simic, 2009)	$1(x)$	$1(x)$	$-1(x)$	$1(x)$
Random walk	x^{-1}	$1(x)$	x^{-1}	$1(x)$
Right normalized	$1(x)$	x^{-1}	x^{-1}	$1(x)$
GCN-GSO (Kipf & Welling, 2017)	$x^{-\frac{1}{2}}$	$x^{-\frac{1}{2}}$	$0(x)$	$0(x)$
Symmetric normalized	$x^{-\frac{1}{2}}$	$x^{-\frac{1}{2}}$	x^{-1}	$1(x)$
Equation (5)	$x^{-\frac{1}{2}}$	$x^{-\frac{1}{2}}$	$x^{-\frac{1}{2}}$	$x^{-\frac{1}{2}}$

It can be noted that $\mathbf{L}$ has a $\lambda_1 = 0$ eigenvalue corresponding to the constant vector. The vector $\mathbf{v}_i = -\mathbf{e}_1 + \mathbf{e}_i$, $i \in \{2, \dots, n\}$ is an eigenvector with eigenvalue $\lambda_2 = -\sqrt{m/n}$, whose multiplicity is $n-1$. Analogously, $\mathbf{v}_i = -\mathbf{e}_{n+1} + \mathbf{e}_{i+1}$, $i \in \{n+1, \dots, n+m-1\}$ is an eigenvector with eigenvalue $\lambda_3 = -\sqrt{n/m}$, whose multiplicity is $m-1$. Finally, the vector $\mathbf{v}_{n+m} = [-m/n\mathbf{1}_n^{\mathrm{T}}, \mathbf{1}_m^{\mathrm{T}}]^{\mathrm{T}}$ is eigenvector with eigenvalues $\lambda_4 = \lambda_2 + \lambda_3$. Therefore, the spectral radius of $\mathbf{L}$ is

$$|\lambda_4| = \frac{m+n}{\sqrt{m\,n}}\,.$$

In the case of a balanced graph, $n = m$ implies that the spectral radius is 2. In the case of a star graph, $n = 1$ and $|\lambda_4| = \mathcal{O}(\sqrt{m})$ as $m \to \infty$; therefore, the asymptotic in Proposition (8) is tight.

D PROOFS

Proof of Proposition (1) and concentration of error. Let $\mathbf{x} = \{x_i\}_{i=1}^N$ be an i.i.d. random sample from ρ. Let K and m be the kernel and diagonal parts corresponding to the metric-probability Laplacian $\mathcal{L}_{\mathcal{N}}$. Let $\mathbf{L}$, $\mathbf{u}$ be

$$L_{i,j} = N^{-1} K(x_i, x_j \rho^{-1})(x_j) - m(x_i)\,,\ u_i = u(x_i)\,.$$

Note that the non-uniform geometric GSO $\mathbf{L}_{\mathcal{G},\boldsymbol{\rho}}$ based on the graph $\mathcal{G}$, which is randomly sampled from $\mathcal{S}$ with neighborhood model $\mathcal{N}$ via the sample points $\mathbf{x}$, is exactly equal to $\mathbf{L}$. Conditioned on $x_i = x$, the expected value is

$$\mathbb{E}\left(\mathbf{L}\,\mathbf{u}\right)_i = N^{-1} \sum_{j=1}^{N} \mathbb{E}\left(K(x, x_k)\,\rho(x_j)^{-1} u(x_j)\right) - m(x)u(x) = \mathcal{L}_{\mathcal{N}} u(x)\,.$$

Since the random variables $\{x_j\}_{j=1}^N$ are i.i.d. to y, then also the random variables

$$\left\{K(x, x_j)\rho(x_j)^{-1}\,u(x_j)\right\}_{j=1}^N$$

are i.i.d., hence,

$$\begin{aligned}
\operatorname{var}\left(\mathbf{L}\mathbf{u}\right)_i &= \operatorname{var}\left(N^{-1}\sum_{j=1}^{N} K(x, x_j)\rho(x_j)^{-1}u(x_j) - m(x)\,u(x)\right) \\
&= N^{-1}\operatorname{var}\left(K(x,y)\rho(y)^{-1}u(y)\right) \\
&\le N^{-1}\mathbb{E}\left(\left|K(x,y)\rho(y)^{-1}u(y)\right|^2\right) \\
&= N^{-1}\int_{\mathcal{S}} \left|K(x,y)\rho(y)^{-1}u(y)\right|^2 \rho(y)\,\mathrm{d}\mu(y) \\
&\le N^{-1}\left\|K(x,\cdot)^2\rho(\cdot)^{-1}\right\|_{\mathrm{L}^\infty(\mathcal{S})} \|u\|^2_{\mathrm{L}^2(\mathcal{S})}\,,
\end{aligned}$$

which proves (1).

Next, we prove the concentration of error result. We know that there exist $a, b > 0$ such that almost everywhere $K(x, x_j)\rho(x_j)^{-1}u(x_j) \in [a, b]$, since K, ρ^{-1} and u are essentially bounded. By Hoeffding's inequality, for $t > 0$,

$$\mathbb{P}\left[|(\mathbf{Lu})_i - \mathcal{L}_{\mathcal{N}}u(x)| \geq t\right] \leq \exp\left(-\frac{2\,N\,t^2}{b-a}\right).$$

Setting

$$\frac{p}{N} = \exp\left(-\frac{2\,N\,t^2}{b-a}\right) \implies t = N^{-\frac{1}{2}}\sqrt{(b-a)\log(2\,Np^{-1})}\,,$$

we obtain that for every node there is an event with probability at least $1 - p/N$ such that

$$|(\mathbf{Lu})_i - \mathcal{L}_{K,m}u(x)| \leq N^{-\frac{1}{2}}\sqrt{(b-a)\log(2Np^{-1})}\,.$$

We then intersect all of these events to obtain an event of probability at least $1 - p$ that satisfies (2.3). □

Proof of Proposition (7). Equality (2) is trivial, since diagonal matrices commutes; equality (1) follows from

$$\begin{aligned}\operatorname{diag}(\mathbf{VA1})_{i,i} = (\mathbf{VA1})_i &= \sum_{j=1}^{n} V_{i,j}(\mathbf{A1})_j = \sum_{j=1}^{n}\sum_{k=1}^{n} V_{i,j}A_{j,k} \\ &= \sum_{k=1}^{n} V_{i,i}A_{i,k} = V_{i,i}\sum_{k=1}^{n} A_{i,k} = (\mathbf{V}\ \operatorname{diag}(\mathbf{A1}))_{i,i}\,.\end{aligned}$$

In order to prove (3), we note that $\mathbf{V}$ can be decomposed as $\mathbf{V} = \sum_{i=1}^{n} v_i\,\mathbf{e}^{(i)}\mathbf{e}^{(i)^{\mathrm{T}}}$. Therefore

$$\begin{aligned}0 &= (\operatorname{diag}(\mathbf{AV1}) - \operatorname{diag}(\mathbf{VA1}))_{k,k} = \operatorname{diag}\left(\sum_{i=1}^{n} v_i\left(\mathbf{A}\mathbf{e}^{(i)}\mathbf{e}^{(i)^{\mathrm{T}}}\mathbf{1} - \mathbf{e}^{(i)}\mathbf{e}^{(i)^{\mathrm{T}}}\mathbf{A1}\right)\right)_{k,k} \\ &= \left(\sum_{i=1}^{n} v_i\left(\mathbf{A}\mathbf{e}^{(i)} - \sum_{j=1}^{n} A_{i,j}\mathbf{e}^{(i)}\right)\right)_k = \sum_{i=1}^{n} v_i A_{k,i} - v_k\sum_{j=1}^{n} A_{k,j} \\ &= \sum_{i=1}^{n}(v_i - v_k)A_{k,i}\,,\end{aligned}$$

must hold for all values of k. Consider the indices $k_1, k_2, \ldots, k_n$ corresponding to the values $v_{k_1} \leq v_{k_2} \leq \cdots \leq v_{k_n}$, then

$$0 = \sum_{i=1}^{n}\underbrace{(v_i - v_{k_1})}_{\geq 0}A_{k_1,i}\,,$$

then $A_{k_1,i} = 0$ for each i such that $v_i > v_{k_1}$. Take the index k_2 and consider

$$0 = \sum_{i=1}^{n}\underbrace{(v_i - v_{k_2})}_{\geq 0}A_{k_2,i} = \sum_{\substack{i=1\\ i\neq k_1}}^{n}\underbrace{(v_i - v_{k_2})}_{\geq 0}A_{k_2,i} + \underbrace{(v_{k_1} - v_{k_2})A_{k_2,k_1}}_{=0}\,.$$

The second addend is 0 because v_{k_2} can be either equal to v_{k_1}, in which case the difference is null, or $v_{k_2} > v_{k_1}$, in which case from the previous step $A_{k_2,k_1} = 0$. Therefore $A_{k_2,i} = 0$ for each i such that $v_i > v_{k_2}$. By finite induction, the thesis holds when $\mathbf{A}$ has null entries in position (i, j) whenever $v_i \neq v_j$. □

Proof of Proposition (8). The eigenvalues can be characterized via the Rayleigh quotient

$$\frac{\left\langle \mathbf{u}, \left(\operatorname{diag}\left(\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}\mathbf{1}\right) - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}\right)\mathbf{u}\right\rangle}{\langle \mathbf{u}, \mathbf{u}\rangle}.$$

Using Proposition (7), and considering $\mathbf{u} = \mathbf{D}^{\frac{1}{2}}\mathbf{v}$ the previous formula can be rewritten as

$$\frac{\left\langle \mathbf{D}^{\frac{1}{2}}\mathbf{v}, \left(\operatorname{diag}\left(\mathbf{A}\mathbf{D}^{-\frac{1}{2}}\mathbf{1}\right) - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\right)\mathbf{v}\right\rangle}{\left\langle \mathbf{D}^{\frac{1}{2}}\mathbf{v}, \mathbf{D}^{\frac{1}{2}}\mathbf{v}\right\rangle} = \frac{\mathbf{v}^{\mathrm{T}}\left(\operatorname{diag}\left(\mathbf{D}^{\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}\mathbf{1}\right) - \mathbf{A}\right)\mathbf{v}}{\mathbf{v}^{\mathrm{T}}\mathbf{D}\mathbf{v}}.$$

Let $d_i = \mathbf{D}_{i,i}$ the degree of the i-th node, using the symmetry of $\mathbf{A}$, the numerator can be rewritten as

$$\begin{aligned}
&\sum_{i,j} v_i^2 \sqrt{\frac{d_i}{d_j}} A_{i,j} - \sum_{i,j} v_i A_{i,j} v_j \\
&= \frac{1}{2}\sum_{i,j} v_i^2 \sqrt{\frac{d_i}{d_j}} A_{i,j} + \frac{1}{2}\sum_{i,j} v_j^2 \sqrt{\frac{d_j}{d_i}} A_{i,j} - \sum_{i,j} v_i A_{i,j} v_j \\
&= \frac{1}{2}\left(\sum_{i,j} v_i A_{i,j}\left(\sqrt{\frac{d_i}{d_j}} v_i - v_j\right) - \sum_{i,j} v_j A_{i,j}\left(v_i - \sqrt{\frac{d_j}{d_i}} v_j\right)\right) \\
&= \frac{1}{2}\sum_{i,j}\left(\frac{v_i}{\sqrt{d_j}} - \frac{v_j}{\sqrt{d_i}}\right) A_{i,j}\left(\sqrt{d_i} v_i - \sqrt{d_j} v_j\right) \\
&= \frac{1}{2}\sum_{i,j} \frac{A_{i,j}}{\sqrt{d_i d_j}}\left(\sqrt{d_i} v_i - \sqrt{d_j} v_j\right)^2 .
\end{aligned}$$

From the last equality follows that the eigenvalues are all positive. From $(a-b)^2 \leq 2(a^2+b^2)$ follows

$$\begin{aligned}
&\leq \sum_{i,j} \frac{A_{i,j}}{\sqrt{d_i d_j}}\left(d_i v_i^2 + d_j v_j^2\right) \\
&= 2\sum_{i,j} A_{i,j}\sqrt{\frac{d_i}{d_j}} v_i^2 \\
&\leq 2\sqrt{N}\sum_i d_i v_i^2 \\
&= 2\sqrt{N}\,\mathbf{v}^{\mathrm{T}}\mathbf{D}\mathbf{v},
\end{aligned}$$

from which the thesis follows. □

Proof of Proposition (2). Consider the map

$$\varphi : [-\pi, \pi) \to \mathbb{S}^1,\ \theta \mapsto \left(\cos(\theta), \quad \sin(\theta)\right)^{\mathrm{T}},$$

and the angles $x, y, \in [-\pi, \pi)$ such that $\varphi(x) = \boldsymbol{x}$, $\varphi(y) = \boldsymbol{y}$, it holds

$$\begin{aligned}
\mathring{d}(\boldsymbol{x}, \boldsymbol{y}) &= \arccos(\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y}) = \arccos(\cos(x)\cos(y) + \sin(x)\sin(y)) \\
&= \arccos(\cos(x-y)) \\
&= x - y + 2\,k\,\pi \\
&= \begin{cases} 2\,\pi + x - y\,, & x - y \in [-2\,\pi, -\pi) \\ y - x\,, & x - y \in [-\pi, 0) \\ x - y\,, & x - y \in [0, \pi) \\ 2\,\pi + y - x\,, & x - y \in [\pi, 2\,\pi) \end{cases}
\end{aligned}$$

$$
\begin{aligned}
&= \begin{cases} 2\,\pi - |x-y|\,, & |x-y| > \pi \\ |x-y|\,, & |x-y| < \pi \end{cases} \\
&= \pi - |\pi - |x-y||\,.
\end{aligned}
$$

□

Proof of Proposition (3). The expected degree of a node θ is the probability of the ball centered at θ times the size N of the sample. The probability of a ball can be computed noting that

$$
\int_{\theta_c-\alpha}^{\theta_c+\alpha} \cos(n_i(\theta-\mu_i))\,\mathrm{d}\theta = \begin{cases} 2\,\alpha\,, & n_i = 0 \\ \dfrac{2\,\cos(n_i(\theta_c-\mu_i))\sin(n_i\,\alpha)}{n_i}\,, & \text{otherwise} \end{cases}
$$

Therefore, the average degree can be computed as

$$
\overline{d} = N \int_{-\pi}^{\pi} \mathbb{P}[\mathrm{B}_\alpha(\theta)]\,\mathrm{sb}(\theta;\boldsymbol{c},\boldsymbol{n},\boldsymbol{\mu})\,\mathrm{d}\theta\,.
$$

The inspection of $\mathrm{sb}(\theta;\boldsymbol{c},\boldsymbol{n},\boldsymbol{\mu})$ and $\mathbb{P}[\mathrm{B}_\alpha(\theta)]$ shows that the only terms surviving integration are the constant term and the product of cosines with same frequency

$$
\int_{-\pi}^{\pi} \cos(n_i(\theta-\mu_i))\cos(n_i(\theta-\mu_i))\,\mathrm{d}\theta = \begin{cases} \pi\cos(n_i(\mu_j-\mu_i)), & n_i = n_j \\ 0, & n_i \neq n_j \end{cases}
$$

from which the thesis follows. □

Proof of Proposition (4). Using Taylor expansion, it holds

$$
\begin{aligned}
\exp(\kappa_i\cos(n_i(\theta-\mu_i))) &= \sum_{m=0}^{\infty} \frac{\kappa_i^m}{m!}\cos(n_i(\theta-\mu_i))^m \\
&= 1 + \sum_{m=1}^{\infty} \frac{\kappa_i^{2m}}{(2m)!}\cos(n_i(\theta-\mu_i))^{2m} \\
&\quad + \sum_{m=1}^{\infty} \frac{\kappa_i^{2m-1}}{(2m-1)!}\cos(n_i(\theta-\mu_i))^{2m-1}\,.
\end{aligned}
$$

A first approximation can be made noting that $\cos(x)^{2m} \leq \cos(x)^2$ and $\cos(x)^{2m-1} \approx \cos(x)$ for all $m \geq 1$, obtaining

$$
\exp(\kappa_i\cos(n_i(\theta-\mu_i))) \approx 1 + (\cosh(\kappa_i)-1)\cos(n_i(\theta-\mu_i))^2 + \sinh(\kappa_i)\cos(n_i(\theta-\mu_i))\,.
$$

Such approximation deteriorates fast when κ_i increases. A more refined approximations is obtained considering the power of cosine with higher coefficient in the Taylor expansion. Using Stirling approximation of factorial, it can be shown that

$$
\frac{\kappa_i^m}{m!} \approx \frac{1}{\sqrt{2\,\pi\,m}}\left(\frac{\kappa_i\,e}{m}\right)^m\,.
$$

In order to make the computation easier, suppose κ_i is an integer; When $m = \kappa_i + 1$, it holds

$$
\frac{1}{\sqrt{2\,\pi\,(\kappa_i+1)}}\left(\frac{\kappa_i\,e}{\kappa_i+1}\right)^{\kappa_i+1} = \frac{e^{\kappa_i+1}}{\sqrt{2\,\pi\,(\kappa_i+1)}}\left(\frac{\kappa_i}{\kappa_i+1}\right)^{\kappa_i+1} < \frac{e^{\kappa_i}}{\sqrt{2\,\pi\,(\kappa_i+1)}} < \frac{e^{\kappa_i}}{\sqrt{2\,\pi\,\kappa_i}}\,,
$$

where the first inequality is justified by the fact that $(\kappa_i/(\kappa_i+1))^{\kappa_i+1}$ is an increasing sequence that tends to $1/e$. The previous formula shows that the coefficient with $m = \kappa_i + 1$ is always smaller than the coefficient with $m = \kappa_i$. The same reasoning can be applied to all the coefficients with

$m > \kappa_i$. Suppose now $\kappa_i \geq 3$, if $m \leq \kappa_i - 2$ the previous reasoning holds. A peculiarity happens when $m = \kappa_i - 1$:

$$\frac{1}{\sqrt{2\,\pi\,(\kappa_i - 1)}} \left(\frac{\kappa_i\, e}{\kappa_i - 1}\right)^{\kappa_i - 1} = \frac{e^{\kappa_i - 1}}{\sqrt{2\,\pi\,\kappa_i}} \left(\frac{\kappa_i}{\kappa_i - 1}\right)^{\kappa_i - \frac{1}{2}} > \frac{e^{\kappa_i}}{\sqrt{2\,\pi\,\kappa_i}}\,,$$

because the sequence $(\kappa_i/(\kappa_i - 1))^{\kappa_i - 1/2}$ is decreasing; therefore $m = \kappa_i - 1$ is the point of maximum, and $m = \kappa_i$ is the second largest value. Therefore, the following approximation for $\exp(\kappa_i \cos(n_i(\theta - \mu_i)))$ holds:

$$\begin{cases} 1 + (\cosh(\kappa_i) - 1)\cos(n_i(\theta - \mu_i))))^2 + \sinh(\kappa_i)\cos(n_i(\theta - \mu_i)))\,, & \kappa_i \leq 1 \\ 1 + (\cosh(\kappa_i) - 1)\cos(n_i(\theta - \mu_i))))^{\kappa_i} + \sinh(\kappa_i)\cos(n_i(\theta - \mu_i))))^{\kappa_i - 1}\,, & \kappa_i \geq 1,\ \text{even} \\ 1 + (\cosh(\kappa_i) - 1)\cos(n_i(\theta - \mu_i))))^{\kappa_i - 1} + \sinh(\kappa_i)\cos(n_i(\theta - \mu_i))))^{\kappa_i}\,, & \kappa_i \geq 1,\ \text{odd} \end{cases}$$

The thesis follows from the equality

$$\cos(n_i(\theta - \mu_i))^{\kappa_i} = \frac{1}{2^{\kappa_i}} \sum_{k=0}^{\kappa_i} \binom{\kappa_i}{k} \cos((2\,k - \kappa_i) n_i(\theta - \mu_i))\,.$$

□

Proof of Proposition (5). The domain of integration can be parametrized as $d_{\mathbb{D}}((r,\theta),(r_c,\theta_c)) \leq \alpha$, leading to

$$\theta \in \left(\theta_c - \arccos\left(\frac{r^2 + r_c^2 - \alpha^2}{2\,r\,r_c}\right), \theta_c + \arccos\left(\frac{r^2 + r_c^2 - \alpha^2}{2\,r\,r_c}\right)\right).$$

Three cases must be discussed: (1) $0 \leq r_c - \alpha \leq r_c + \alpha \leq 1$, (2) $r_c - \alpha < 0$, (3) $r_c + \alpha > 1$. In scenario (1), the ball $\mathrm{B}_\alpha(r_c,\theta_c)$ is contained in $\mathbb{D}$. The probability of the ball can be computed as

$$\begin{aligned} \mathbb{P}\left[\mathrm{B}_\alpha(r_c,\theta_c)\right] &= 2 \int_{r_c-\alpha}^{r_c+\alpha} r \int_{\theta_c-\theta_r}^{\theta_c+\theta_r} \mathrm{sb}(\theta; \boldsymbol{c}, \boldsymbol{n}, \boldsymbol{\mu})\ \mathrm{d}\theta\ \mathrm{d}r \\ &= \frac{4}{B} \sum_{i:n_i \neq 0} \frac{c_i}{n_i} \cos(n_i\,(\theta_c - \mu_i)) \int_{r_c-\alpha}^{r_c+\alpha} r\,\sin\left(n_i\,\arccos\left(\frac{r^2 + r_c^2 - \alpha^2}{2\,r\,r_c}\right)\right) \mathrm{d}r \\ &+ \frac{4}{B}\left(\sum_{i:n_i=0} c_i + A\right) \int_{r_c-\alpha}^{r_c+\alpha} r \arccos\left(\frac{r^2 + r_c^2 - \alpha^2}{2\,r\,r_c}\right) \mathrm{d}r\,, \end{aligned} \tag{6}$$

where the last equality comes from Proposition (3). For simplicity, define

$$f_{n_i-1}(r) = r\sqrt{1 - \left(\frac{r^2 + r_c^2 - \alpha^2}{2\,r\,r_c}\right)^2}\ \mathrm{U}_{n_i-1}\left(\frac{r^2 + r_c^2 - \alpha^2}{2\,r\,r_c}\right),$$

$$g(r) = r\,\arccos\left(\frac{r^2 + r_c^2 - \alpha^2}{2\,r\,r_c}\right),$$

where U_k is the k-th Chebyschev polynomial of second kind. It is worthy to note that $f_{n_i-1}(r_c + \alpha) = 0$, $f_{n_i-1}(r_c - \alpha) = 0$, $f_{n_i-1}(\alpha - r_c) = 0$ and

$$f_{n_i-1}(r_c) = \alpha\sqrt{1 - \left(\frac{\alpha}{2\,r_c}\right)^2}\ \mathrm{U}_{n_i-1}\left(1 - \frac{\alpha^2}{2\,r_c^2}\right),\ f_{n_i-1}(\alpha) = \alpha\sqrt{1 - \frac{r_c}{2\,\alpha}}\ \mathrm{U}_{n_i-1}\left(\frac{r_c}{2\,\alpha}\right),$$

while $g(r_c + \alpha) = 0$, $g(r_c - \alpha) = 0$, $g(\alpha - r_c) = (\alpha - r_c)\pi$ and

$$g(r_c) = r_c \arccos\left(1 - \frac{\alpha^2}{2\,r_c^2}\right),\ g(\alpha) = \alpha\,\arccos\left(\frac{r_c}{2\,\alpha}\right).$$

The integral in Equation (6) can be approximated by the semi-area of an ellipse having α and $f_{n_i-1}(r_c)$ (respectively $g(r_c)$) as axes

$$\int_{r_c-\alpha}^{r_c+\alpha} f_{n_i-1}(r)\,\mathrm{d}r \approx \frac{\pi}{2}\alpha f_{n_i-1}(r_c)\,, \int_{r_c-\alpha}^{r_c+\alpha} g(r)\,\mathrm{d}r \approx \frac{\pi}{2}\alpha g(r_c)\,, \tag{7}$$

that can be seen as a modified version of the Simpson' rule, since the latter would lead to a coefficient of $4/3$ instead of $\pi/2$. A comparison between the two methods is shown in Appendix (D). In scenario (2) the domain of integration contains the origin and the argument of $\arccos$ in Appendix (D) could be not well defined. The singularity can be removed decomposing the domain of integration as the union of a disk of radius $\alpha - r_c$ around the origin and the remaining. Hence

$$\begin{aligned}\mathbb{P}\left[\mathrm{B}_\alpha(r_c,\theta_c)\right] &= 2\int_0^{\alpha-r_c} r \int_{-\pi}^{\pi} \mathrm{sb}(\theta;\boldsymbol{c},\boldsymbol{n},\boldsymbol{\mu})\;\mathrm{d}\theta\;\mathrm{d}r + 2\int_{\alpha-r_c}^{\alpha+r_c} r \int_{\theta_c-\theta_r}^{\theta_c+\theta_r} \mathrm{sb}(\theta;\boldsymbol{c},\boldsymbol{n},\boldsymbol{\mu})\;\mathrm{d}\theta\;\mathrm{d}r \\ &= (\alpha-r_c)^2 + 2\int_{\alpha-r_c}^{\alpha+r_c} r \int_{\theta_c-\theta_r}^{\theta_c+\theta_r} \mathrm{sb}(\theta;\boldsymbol{c},\boldsymbol{n},\boldsymbol{\mu})\;\mathrm{d}\theta\;\mathrm{d}r\,.\end{aligned}$$

The same reasoning as before leads to the approximations

$$\int_{\alpha-r_c}^{\alpha+r_c} f_{n_i-1}(r)\,\mathrm{d}r \approx \frac{\pi}{2} r_c\, f_{n_i-1}(\alpha)\,, \int_{\alpha-r_c}^{\alpha+r_c} g(r)\,\mathrm{d}r \approx \frac{\pi}{2} r_c\, g(\alpha)\,.$$

In scenario (3) the domain of integration partially lies outside $\mathbb{D}$. Hence

$$\mathbb{P}\left[\mathrm{B}_\alpha(r_c,\theta_c)\right] = 2\int_{r_c-\alpha}^{1} r \int_{\theta_c-\theta_r}^{\theta_c+\theta_r} \mathrm{sb}(\theta;\boldsymbol{c},\boldsymbol{n},\boldsymbol{\mu})\;\mathrm{d}\theta\;\mathrm{d}r\,,$$

that can be approximated as

$$\begin{aligned}\int_{r_c-\alpha}^{1} f_{n_i-1}(r)\,\mathrm{d}r &\approx \frac{f_{n_i-1}(r_c)}{2}\left(\frac{1-r_c}{\alpha}\sqrt{\alpha^2-(1-r_c)^2} + \alpha\,\arcsin\left(\frac{1-r_c}{\alpha}\right)\right)\,, \\ \int_{r_c-\alpha}^{1} g(r)\,\mathrm{d}r &\approx \frac{g(r_c)}{2}\left(\frac{1-r_c}{\alpha}\sqrt{\alpha^2-(1-r_c)^2} + \alpha\,\arcsin\left(\frac{1-r_c}{\alpha}\right)\right)\,.\end{aligned}$$

The three scenarios can be summarized in one big formula. For simplicity, define the operator

$$\begin{aligned}\mathcal{I}[f](r_c) = &\frac{\pi}{2}\frac{\alpha+r_c+\min\{\alpha-r_c, r_c-\alpha\}}{2} f_{n_i-1}\left(\frac{\alpha+r_c+\max\{\alpha-r_c, r_c-\alpha\}}{2}\right) \\ &- \frac{\max\{r_c+\alpha-1,0\}}{r_c+\alpha-1}\frac{f_{n_i-1}(r_c)}{2}\left(\alpha\,\arccos\left(\frac{1-r_c}{\alpha}\right) - \frac{1-r_c}{\alpha}\sqrt{\alpha^2-(1-r_c)^2}\right)\,,\end{aligned}$$

that given a function f returns the ellipse approximation of the integral over balls, it holds

$$\begin{aligned}\mathbb{P}\left[\mathrm{B}_\alpha(r_c,\theta_c)\right] = &\frac{4}{B}\sum_{i:n_i\neq 0}\frac{c_i}{n_i}\cos(n_i\,(\theta_c-\mu_i))\,\mathcal{I}[f_{n_i-1}](r_c) + \frac{4}{B}\left(\sum_{i:n_i=0} c_i + A\right)\mathcal{I}[g](r_c) \\ &+ \max\{0,\alpha-r_c\}^2\,.\end{aligned}$$

from which the thesis follows. To compute the average degree of a spatial network from the unit disk, the quantity

$$\overline{d} := \int_0^1 2\,r \int_{-\pi}^{\pi} \mathbb{P}\left[\mathrm{B}_\alpha(r,\theta)\right]\;\mathrm{sb}(\theta;\boldsymbol{c},\boldsymbol{n},\boldsymbol{\mu})\,\mathrm{d}\theta\,\mathrm{d}r\,,$$

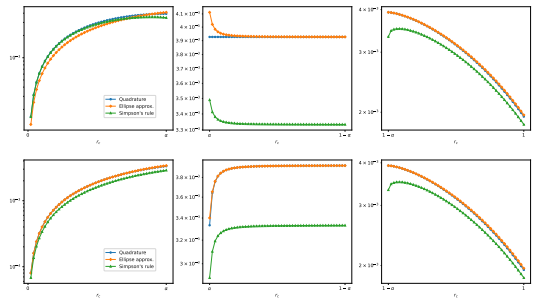


Figure 8: Approximation of $\int_a^b g(r)\,\mathrm{d}r$ (top) and $\int_a^b f_0(r)\,\mathrm{d}r$ (bottom) when $a = \alpha - r_c$ and $b = \alpha + r_c$ (left), $a = r_c - \alpha$ and $b = r_c + \alpha$ (center), and $a = \alpha + r_c$ and $b = 1$ (right) as a function of r_c, $\alpha = 0.05$.

must be computed. Using Proposition (3), the integral can be written in the form

$$\begin{aligned}\overline{d} =& \frac{8\,\pi}{B^2} \sum_{i:n_i \neq 0} \sum_{j:n_i = n_j} \frac{c_i c_j}{n_i} \cos(n_i\,(\mu_i - \mu_j)) \int_0^1 r\,\mathcal{I}[f_{n_i - 1}](r)\,\mathrm{d}r \\ &+ \frac{16\,\pi}{B^2} \left(\sum_{i:n_i = 0} c_i + A \right)^2 \int_0^1 r\,\mathcal{I}[g](r)\,\mathrm{d}r\,.\end{aligned}$$

From $\mathrm{U}_k(1) = k$ the following approximation can be derived

$$\mathcal{I}[f_{n_i-1}](r) \approx \frac{\pi}{2}\,n_i\,\alpha^2 \sqrt{1 - \left(\frac{\alpha}{2\,r}\right)^2} \sim \frac{\pi}{2}\,n_i\,\alpha^2 \left(1 - \frac{\alpha^2}{8\,r^2}\right)\,,$$

hence the integral boils down to

$$\int_0^1 r\,\mathcal{I}[f_{n_i-1}](r)\,\mathrm{d}r \approx \frac{\pi}{4} n_i \alpha^2\,.$$

The term

$$\begin{aligned}\int_0^1 r\,\mathcal{I}[g](r)\,\mathrm{d}r =& \frac{\pi\,\alpha}{24} \left(\alpha\,\sqrt{4 - \alpha^2} + 4\,\arccos\left(1 - \frac{\alpha^2}{2}\right)\right) \\ &+ \frac{\pi\,\alpha^4}{48} \left(\log\left(2 + \sqrt{4 - \alpha^2}\right) - \log(\alpha)\right) \\ &\sim \frac{\pi\,\alpha^2}{24} \left(4 + \sqrt{4 - \alpha^2}\right) \sim \frac{\pi\,\alpha^2}{24} \left(6 - \frac{\alpha^2}{4}\right) \sim \frac{\pi\,\alpha^2}{4}\,,\end{aligned}$$

from which the thesis follows. □

Proof of Proposition (6). Similarly to what has been done in Proposition (5), the domain of integration can be parametrized as

$$\theta \in (\theta_c - \theta_r, \theta_c + \theta_r)\,,\ \theta_r = \arccos\left(d_r\right)\,,\ d_r = \frac{\cosh(r)\cosh(r_c) - \cosh(\alpha)}{\sinh(r_c)\sinh(r)}\,.$$

In order to remove the singularity of the argument of arccos, the domain of integration can be decomposed as a ball containing the origin and the remaining, leading to

$$\mathbb{P}[\mathrm{B}_\alpha(r_c, \theta_c)] = \frac{1}{\cosh(R) - 1} \int_{l_1}^{u_1} \int_{-\pi}^{\pi} \sinh(r)\,\mathrm{sb}(\theta; \boldsymbol{c}, \boldsymbol{n}, \boldsymbol{\mu})\,\mathrm{d}\theta\,\mathrm{d}r$$

$$+ \frac{1}{\cosh(R)-1} \int_{l_2}^{u_2} \int_{\theta_c-\theta_r}^{\theta_c+\theta_r} \sinh(r)\, \mathrm{sb}(\theta; \boldsymbol{c}, \boldsymbol{n}, \boldsymbol{\mu})\, \mathrm{d}\theta\, \mathrm{d}r\,,$$

where $l_1 = 0$, $u_1 = \max\{\alpha - r_c, 0\}$, $l_2 = |\alpha - r_c|$ and $u_2 = \min\{r_c + \alpha, R\}$.

$$\begin{aligned}
&= \frac{\cosh(u_1)-1}{\cosh(R)-1} + \frac{1}{\cosh(R)-1} \int_{l_2}^{u_2} \sinh(r)\, \mathbb{P}\left[\mathrm{B}_{\theta_r}(\theta_c)\right] \mathrm{d}r \\
&= \frac{\cosh(u_1)-1}{\cosh(R)-1} + \frac{2\left(\sum_{i:n_i=0} c_i + A\right)}{B\,\left(\cosh(R)-1\right)} \int_{l_2}^{u_2} \sinh(r)\theta_r\, \mathrm{d}r \\
&+ \frac{2}{B} \sum_{i:n_i\neq 0} \frac{c_i}{n_i} \frac{\cos(n_i(\theta_c-\mu_i)}{\cosh(R)-1} \int_{l_2}^{u_2} \sinh(r)\sqrt{1-d_r^2}\, U_{n_i-1}(d_r)\, \mathrm{d}r\,,
\end{aligned}$$

The approximations $\theta_r \approx \sqrt{2-2\,d_r}$ and $d_r \approx 1 + 2\left(e^{-2\,r} + e^{-2\,r_c} - e^{\alpha-r_c-r} - e^{-\alpha-r_c-r}\right)$ as in Gugelmann et al. (2012) can be used to analyze the behaviours of both integrals. For large R, it holds

$$\begin{aligned}
\int_{l_2}^{u_2} \frac{\sinh(r)}{\cosh(R)-1}\theta_r\, \mathrm{d}r &\approx \int_{l_2}^{u_2} e^{r-R}\sqrt{1-d_r}\, \mathrm{d}r \\
&\approx 2\int_{l_2}^{u_2} e^{r-R}\sqrt{e^{\alpha-r_c-r} + e^{-\alpha-r_c-r} - e^{-2r} - e^{-2\,r_c}}\, \mathrm{d}r \\
&= 2\int_{l_2}^{u_2} e^{\frac{r-2\,R+\alpha-r_c}{2}}\sqrt{1+e^{-2\alpha} - e^{-r-\alpha+r_c} - e^{r-\alpha-r_c}}\, \mathrm{d}r \\
&\approx 4e^{\frac{\alpha-R-r_c}{2}}\,,
\end{aligned}$$

where the last approximation is justified by $\sqrt{1+x} = 1 + \mathcal{O}(x)$ when $|x| \leq 1$. Noting that $-1 \leq d_r \leq 1$, one can get rid of the polynomial contribution

$$\begin{aligned}
\int_{l_2}^{R} \frac{\sinh(r)}{\cosh(R)-1}\sqrt{1-d_r^2}\, U_{n_i-1}(d_r)\, \mathrm{d}r &\approx n_i \int_{l_2}^{R} e^{r-R}\sqrt{1-d_r^2}\, \mathrm{d}r \\
&= n_i \int_{l_2}^{R} e^{r-R}\sqrt{1-d_r}\sqrt{1+d_r}\, \mathrm{d}r \\
&\approx \sqrt{2}\, n_i \int_{l_2}^{R} e^{\frac{r-2R+\alpha-r_c}{2}}\sqrt{1+d_r}\, \mathrm{d}r \\
&\approx 4\, n_i e^{\frac{\alpha-R-r_c}{2}}\,.
\end{aligned}$$

Therefore, the probability of balls is approximately

$$\mathbb{P}[\mathrm{B}_\alpha(r_c, \theta_c)] \approx \frac{8\, e^{\frac{\alpha-R-r_c}{2}}}{B} \left(\sum_{i:n_i\neq 0} c_i \cos(n_i\,(\theta_c - \mu_i)) + \left(\sum_{i:n_i=0} c_i + A \right) \right)\,,$$

and the expected average degree of the network is

$$\begin{aligned}\overline{d} &= N \int_0^R \int_{-\pi}^{\pi} \mathbb{P}[\mathrm{B}_\alpha(r,\theta)]\, \mathrm{sb}(\theta; \boldsymbol{c}, \boldsymbol{n}, \boldsymbol{\mu}) \frac{\sinh(r)}{\cosh(R)-1}\, \mathrm{d}\theta\, \mathrm{d}r \\ &\approx \frac{16\,\pi\,N\,e^{\frac{\alpha-2\,R}{2}}}{B^2} \left(\sum_{i:n_i\neq 0} \sum_{j:n_i=n_j} c_i c_j \cos(n_i\,(\mu_i-\mu_j)) + 2\left(\sum_{i:n_i=0} c_i + A\right)^2 \right).\end{aligned}$$

□