

Chordless cycle filtrations for dimensionality detection in complex networks via topological data analysis

Aina Ferrà Marcús¹, Robert Jankowski^{2,3,4}, Meritxell Vila-Miñana⁵,
Carles Casacuberta¹, M. Ángeles Serrano^{2,3,6*}

¹Departament de Matemàtiques i Informàtica, Universitat de Barcelona,
Gran Via de les Corts Catalanes 585, 08007 Barcelona, Spain

²Departament de Física de la Matèria Condensada, Universitat de Barcelona,
Martí i Franquès 1, 08028 Barcelona, Spain

³Universitat de Barcelona Institute of Complex Systems (UBICS),
Universitat de Barcelona, Barcelona, Spain

⁴Faculty of Electrical Engineering, Mathematics and Computer Science,
Delft University of Technology, 2628 CD, Delft, Netherlands

⁵Center for Complex Networks and Systems Research, Luddy School of Informatics,
Computing, and Engineering, Indiana University, Bloomington, IN, USA

⁶ICREA, Passeig Lluís Companys 23, 08010 Barcelona, Spain

*Corresponding author. E-mail: marian.serrano@ub.edu

Many complex networks, ranging from social to biological systems, exhibit structural patterns consistent with an underlying hyperbolic geometry. Revealing the dimensionality of this latent space can disentangle the structural complexity of communities, impact efficient network navigation, and fundamentally shape connectivity and system behavior. We introduce a topological data analysis weighting scheme for graphs based on chordless cycles to estimate network dimensionality in a data-driven way. We further show that the resulting descriptors can effectively estimate network dimensionality using a neural network architecture trained on a synthetic graph database constructed for this purpose, which requires no retraining to transfer effectively to real-world networks. Thus, by combining cycle-aware filtrations, algebraic topology, and machine learning, our approach provides a robust and effective method for uncovering the hidden geometry of complex networks and guiding accurate modeling and low-dimensional embedding.

1 Introduction

The rapid growth of data has created significant challenges in science and technology. Large datasets from fields such as biology (e.g., gene expression and protein interactions), social networks (e.g., agent behavior), and physics (e.g., cosmological simulations) often contain rich structural information hidden in their complex nature. Capturing this information requires tools capable of detecting patterns in data. Topological Data Analysis (TDA) [1, 2, 3, 4] is one such approach that uses ideas from topology to identify features that persist across scales.

In TDA, scales are usually defined through filtrations, a systematic way to build a sequence of simplicial complexes that encode the geometric and topological structure of a dataset across multiple levels of resolution. Central to the analysis of such simplicial complexes is persistent homology [2, 3], a main TDA computational method that tracks the emergence, persistence, and disappearance of topological features—such as connected components or loops—in a filtered simplicial complex as the filtration parameter evolves. In this context, graphs can be treated as simplicial complexes and filtered by assigning weights to their nodes and/or edges.

The success of persistent homology depends critically on the choice of a filtration and topological features of interest, and the selection of these depends on the problem being addressed. Not every filtration or feature descriptor is suitable for every problem, underscoring the importance of designing suitable TDA pipelines to achieve meaningful results. The triad filtration-feature-framework (FFF) embodies an intertwined relationship that forms the foundation for effectively leveraging persistent homology. However, filtration schemes based on a variety of quantifiers, such as Forman–Ricci curvature [5] or betweenness centrality [6], are sometimes applied indiscriminately to data without considering the specifics of the selected topological features and the nature or framework of the problem at hand. This work emphasizes the importance of aligning the FFF triad by addressing dimensionality detection in complex networks, thereby bridging the gap between complex network theory and TDA. Simplicial complexes associated with complex networks have emerged as powerful representations in network science [7, 8, 9, 10, 11], offering insights into higher-order structures beyond pairwise interactions. Moreover, persistent homology of filtered simplicial complexes has been applied to the study of complex networks in general [12, 13, 14, 15, 16, 17] and to neuroscience in particular [18, 19, 20, 21].

The problem addressed, the detection of the effective dimensionality of the space in which a complex phenomenon unfolds, is a recurring theme across the sciences. In statistical physics, the spatial dimension strongly constrains scaling laws and often determines the universality class of critical, extended systems in which multiple length scales are relevant. Even in the Ising model—the simplest widely used model of collective behavior and phase transitions, in which binary spins tend to align with their neighbors—dimensionality has a strong impact on critical behavior [22, 23]. Diffusion processes are also often studied through random walks on D -dimensional lattices, where the dimension controls return probabilities, exploration rates, and typical first-passage times [24, 25, 26]. In other areas of physics, such as string-theory settings, extra dimensions beyond the observed three are needed, but remain inaccessible because they

are tightly compactified or because observable dynamics are confined to a lower-dimensional subset [27]. A similar idea appears in computer science: although data and interactions may live in a formally high-dimensional space, relevant configurations typically occupy a much smaller effective subspace, with important implications for both structure and dynamics [28, 29]. Dimensionality is also crucial in biology: transport and diffusion of molecules, encounter rates, and search processes depend sensitively on effective dimension [30, 31], and the structure of regulatory, neuronal, and ecological networks can enforce low-dimensional or genuinely multiscale dynamics, with different dimensions governing local versus global behavior [32]. Thus, the effective dimensionality of complex phenomena has direct implications for structural properties, dynamical regimes, and the mechanisms that dominate robustness, controllability, and criticality in complex systems.

In network science, several non-equivalent definitions of dimensionality that probe different aspects of structure and dynamics have been proposed in the literature. Topological shortest paths have been used to define chemical or fractal (box-counting) dimension [33, 34, 35, 36, 37], including networks with explicit geometrical embedding [38]; related to this, metric dimension is defined as the smallest number of nodes required to identify all other nodes uniquely based on shortest path distances [39]; correlation dimension is computed from network trajectories of random walkers [40]; and spectral dimension depends on the spectral properties of the graph Laplacian [41, 42]. The problem has also been addressed within the framework of network geometry [43]. A model-driven approach [44] leverages the geometric $\mathbb{S}^D/\mathbb{H}^{D+1}$ model [45, 46], which reproduces the observed connectivity of real networks, to reveal their intrinsic dimensionality in a latent hyperbolic space, where nodes are more likely to be connected if they are closer to each other. The real network’s specific dimension is determined by projecting the frequencies of chordless cycles of varying lengths onto the background model’s statistical configuration space. This technique revealed ultra-low-dimensional structures in real networks previously masked by apparent high-dimensionality [44]. This includes, for instance, tissue-specific biomolecular networks being extremely low-dimensional, brain connectomes being close to the three dimensions of their anatomical embedding, and social networks and the Internet, requiring slightly higher dimensionality. Within the same modeling framework, an alternative method for determining network dimensionality uses an embedding technique, named D -Mercator [47], to produce multidimensional maps of real networks in $(D+1)$ -hyperbolic space whose dimensionality strongly shapes network properties, including community diversity [46, 48]. The maps are used to estimate intrinsic dimensionality based on navigability and community structure, yielding results consistent with the statistical configuration space approach.

Despite this broad relevance, the concept of network dimensionality remains largely unexplored within TDA. In this work, we propose a method for dimensionality detection in complex networks based on TDA, which complements a trilogy in combination with the configuration-space and embedding methods used in [44] and [47], respectively. The dimensionality detection task, crucial for understanding the intrinsic geometry of data, showcases how tailored filtrations and feature selection can enhance the ability of persistent homology to analyze and interpret complex data for specific purposes. Specifically, we introduce a chordless cycle filtra-

tion scheme and use it to compute extended persistence of cycles, as the topological descriptor that best captures the distribution of cycles in synthetic and real networks to predict their dimensionality.

Moreover, we propose a data-driven approach to estimate network dimensionality by training a neural network on nearly 800 000 synthetic networks. A multilayer perceptron accurately estimates dimensionality and transfers effectively to real-world networks, generalizing and adapting to new data without retraining. Ablation experiments demonstrate that TDA features play an important role in this task, even when combined with average cycle densities and degree-related graph features.

2 Results

Our approach to estimating network dimensionality using persistent homology is twofold. In [44], it has been shown that measuring the intrinsic dimensionality of a complex network is possible by computing profiles of structural properties that are sensitive to dimensionality. These properties are densities of chordless cycles of sizes three (triangle), four (squares), and five (pentagons), but the method required generating ensembles of synthetic networks for each candidate network. Here, we instead focus on persistence summaries using a filtration based on the density of cycles and show that these descriptors reliably detect latent dimensionality. We then leverage the descriptors to develop a supervised machine-learning model, trained on a large database of synthetic graphs with known dimensions and controlled properties, to predict the intrinsic dimension of an input network.

2.1 Persistence descriptors for dimensionality estimation

Many real networks share universal properties, such as sparsity, heavy-tailed degree distributions, the small-world effect, high clustering coefficients, and self-similarity. These properties can be captured by a simple geometric framework [49] using the $\mathbb{S}^1/\mathbb{H}^2$ model [45, 50], which combines a popularity coordinate—controlling node degrees—with a similarity coordinate that represents all other attributes influencing network connectivity. This model can be generalized to a D -dimensional similarity space, yielding the $\mathbb{S}^D/\mathbb{H}^{D+1}$ model [45, 46]; see Section 4 for more details. These models have been used to determine the dimensionality of real networks by analyzing their cycle profiles [44]. Here, however, we propose to measure their persistence descriptors instead.

The goal of persistent homology is to compute topological features of a space equipped with a filtration. In the context of graphs, one can define filtrations by endowing either nodes or edges with suitable weightings. A common choice is the degree filtration on nodes; however, it has been shown that the degree filtration is less expressive for some graph learning tasks than other, motif-based, filtrations [1]. In this work, we propose an edge weighting scheme based on densities of chordless cycles. A chordless cycle is defined as a closed edge path in which no

two non-consecutive nodes are connected by an edge.

Our goal is to determine an optimal value of D for the $\mathbb{S}^D/\mathbb{H}^{D+1}$ model corresponding to a real network, in which γ and β —the parameters of the model controlling the scale-free degree distribution and the clustering coefficient, respectively— are usually unknown. In our data-driven method, optimality is defined in the same way as in [44], by picking the dimension of a closest point in the configuration space of synthetic surrogates generated from the given real network, using D -dimensional geometric randomization (D-GR), as described in Section 4.1, with different values of γ , β , and D . The D-GR model, originally proposed for $D = 1$ [51], works on the observed sequence of node degrees and rewires the network to maximize the likelihood that the new topology is generated by the $\mathbb{S}^D/\mathbb{H}^{D+1}$ model.

Thus, for each complex network G_0 , whose dimensionality is to be estimated, we generated an ensemble of synthetic surrogates G_1, \dots, G_n using the \mathbb{S}^D model with a range of values of D and a range of values of the clustering coefficient β until $\beta = 6D$. Densities of edge triangles, chordless squares, and chordless pentagons were then computed as specified in Section 4.2 for each surrogate in the ensemble and for the target network. Therefore, each network G yields three weighted graphs (G, w_t) , (G, w_s) and (G, w_p) , where the edge weights w_t, w_s, w_p are the densities of triangles, squares, and pentagons, respectively. The mean value of w_t over all the edges of a graph is denoted by C_t , and similarly with squares and pentagons.

Topological features of graphs equipped with a filtration given by edge weights were computed using persistent homology, a tool from algebraic topology used to describe shape characteristics from many kinds of data (Section 4.2). In this work, persistence refers specifically to the evolution of cycles along the values of a given filtration. However, cycles in graphs have infinite persistence, since there are no higher-dimensional simplices to eventually fill them. We therefore computed extended persistence of cycles, defined as the difference between the largest and smallest weights among the simplices forming a closed path (see Fig. 1). Since we focus on graphs equipped with edge weightings only, we introduce in this article a technique for replacing a given edge-weighted graph with a larger, topologically equivalent graph, that carries weights on nodes and on the original edges in a way that is consistent with both sublevel and superlevel filtrations. This is illustrated in Fig. 1 and explained with more detail in Section 4.2.

Total extended persistence was used as a topological descriptor, resulting in a feature vector (TP_t, TP_s, TP_p) for each network. A representation of the configuration space is shown in Fig. 2, where each point corresponds to a surrogate graph. Points are coloured by the corresponding dimension. The target network from which surrogates were generated has $D = 1$ and is marked with a black cross.

Plausibility of an association between total extended persistence of cycles in a graph and its embedding dimensionality is illustrated in Figs. S1, S2, and S3, which show that dimensionality is inversely associated with total extended persistence values. A negative correlation is more evident in the case of squares and pentagons, with respective Pearson coefficient values of $\rho_s = -0.3140$ and $\rho_p = -0.2433$, versus $\rho_t = -0.0522$ for triangles, calculated using a sample of size 50 000 from the SYNNET database (Section 2.2).

In order to estimate the dimensionality of the target network, a k -nearest neighbors clas-

sifier (kNN) was used. The classifier identifies the k surrogates closest to the target network in the surrogate configuration space (TP_t, TP_s, TP_p) by minimizing Euclidean distance. The value of k was not fixed, but it was determined within each ensemble of surrogates, by finding the value of k with highest accuracy in classifying each surrogate in the ensemble. Specifically, the inferred dimension D^* of the target network G_0 maximizes the weighted frequency $f(D) = \sum_{i=1}^k \omega_i \delta_{D_i, D}$, where the normalized weights are inversely proportional to the distance between the real network and the i -th surrogate G_i in the (TP_t, TP_s, TP_p) space, and $\delta_{D_i, D}$ is the Kronecker delta function [44]. A schematic pipeline summary of the suggested methodology is shown in Fig. 3a.

The closeness of the kNN approximation was measured with a congruency index reflecting the fidelity of the surrogates with respect to the target network. The index is defined as the ratio $\rho = d_0/\bar{d}$, where d_0 is the distance between the point in configuration space corresponding to the target network and its closest neighbor, and \bar{d} is the average of the distances between each surrogate point and its closest neighbor. Hence, the value of ρ is large when the point cloud is clustered while the target network falls far away from the clusters. For comparability reasons, it is convenient to provide values of $1/\rho$, as in Table 1, since, in most cases, $1/\rho$ takes values between 0 and 1, with values closer to one indicating higher fidelity between the target network and its surrogates.

As an evaluation step, the performance of the chordless cycle density filtration was tested on synthetic target networks generated using the \mathbb{S}^D model for specific values of D (from 1 to 7), with different degree heterogeneities by varying the exponent ($\gamma = 2.7$ and $\gamma = 3.5$), and two values of inverse temperature ($\beta = 2.5 D$, corresponding to the high clustering regime, and $\beta = 1.5 D$, corresponding to the low clustering regime). For each combination (D, γ, β) , ten synthetic target networks were generated, and the above method was applied to each of them to infer a dimensionality D^* . Inferred dimensionalities were compared with the original dimension D from which the target synthetic network was generated. Confusion matrices to visualize the performance of the inference method are shown in Fig. 3b. Our predictions generate some small confusion with contiguous values of D to the diagonal, especially for low values of β and γ , but this is expected, due to the high heterogeneity of the degree distribution.

Estimated dimension values for selected real-world target networks using TDA are shown in Table 1. The resulting values were compared with those obtained by implementing the method described in [44], using the configuration space of mean cycle densities (C_t, C_s, C_p) , which have been recalculated using the D-GR procedure. Inverse values of the congruency index ρ are shown. As additional information, 2D projections of the configuration space (TP_t, TP_s, TP_p) for the selected real-world networks are displayed in Figs. S1 and S2.

2.2 Dimensionality estimation using neural networks

The method described in the previous section and the approach from [44] rely on a large set of surrogate networks. For each new real network, one needs to generate a set of synthetic networks, compute their properties, and use a classifier to detect the dimension. As a consequence,

when a new network is considered, the entire pipeline must be repeated. In this section, we propose an alternative based on a neural network that, once trained, can estimate dimensions directly, even for very large networks, generalizing and adapting to new data without retraining.

Neural networks excel for such tasks, since our aim is to approximate an unknown function yielding dimensionality values from a collection of predictors, including mean chordless cycle densities and/or persistence of corresponding filtrations. For this purpose, we created a database of synthetic complex networks called SYNNET, which we use to train a neural network, named DIMNN, to estimate the dimensionality of real-world networks. The main advantage of this method is that training in our database is performed only once and independently of specific target networks. This avoids the need to generate surrogates for each case under study. In total, we produced 792 000 synthetic networks generated from the S^D model (see Section 4.3 for more details). An 80%-20% training-validation split was used.

For each synthetic network in the database, chordless cycle densities were computed as in Section 4.2 and averaged over all edges, as well as total persistence values obtained from the corresponding filtrations, and, additionally, the first moment and the normalized second moment of the degree distribution (i.e., the expected square divided by the square of the expected value). The normalized second moment is sensitive to degree fluctuations and so to very high degree nodes: for highly heterogeneous networks, the ratio becomes large, while for homogeneous networks the ratio is close to 1. Hence, the normalized second moment tends to correlate with γ . Likewise, the average density of triangles C_t serves as an approximation of the inverse temperature β .

Other relevant descriptors of complex networks that we integrated in our feature vectors are minimum and maximum degree, and average neighbor degree, which is denoted by $\langle k_{nn} \rangle$ (see Section 4.2). This is a measure used to quantify degree-degree correlations in a graph, that is, how the degree of a node relates to the degrees of its neighbors.

Feature vectors consisting of number of nodes, average degree, normalized second moment of the degree distribution, minimum and maximum degrees, average neighbor degree, mean chordless cycle densities C_t, C_s, C_p , and total persistences TP_t, TP_s, TP_p of the corresponding filtrations were fed into a residual multilayer perceptron (see Section 4.3 for details). Our pipeline is shown in Fig. 4a. The highest classification accuracy obtained with DIMNN for estimating the dimensionality in the range $D = 1$ to $D = 10$ was 83.00% on the validation set with the full feature vector. Confusion matrices are shown in Fig. 4b.

We also performed an ablation study, selecting subsets of the feature vector for both training and predictions. In Fig. 5, we show how the performance changes as a set of features is added at a time. The lowest accuracy is obtained when we use only the vector of mean chordless cycle densities with no other added features. Once degree-related properties are appended to the feature vector —e.g., minimum, maximum, and average degrees—, the accuracy increases, and incorporating topological information improves the accuracy further. In fact, combining total persistence with mean chordless cycle densities maximizes the performance. In Table S2, we show mean validation accuracies as well as the number of epochs and total training times, and a corresponding heatmap is provided in Fig. S6.

Next, we show the results of applying the trained DIMNN to detect the dimensionality of real-world complex networks. We analyzed the 10 networks from the previous section and predicted their dimensions. Table 1 compares the inferred dimension across three different methods. One can observe that the predictions do not vary significantly, by only one or two dimensions in most cases, which highlights the robustness of the trained neural network. We incorporated another dataset of real complex networks [52]. First, we filtered out bipartite and temporal networks, and restricted to networks with topological properties aligned with the training dataset (Fig. S8). Ranges of SYNNET database features are shown in Table S1. In total, we collected 53 new networks, whose properties are summarized in Tables S4 and S5.

In Fig. 6a, we group the dataset by domain, showing that the Biological and Transportation categories together account for more than half of all networks. We then apply our trained neural network to predict the intrinsic dimensions of these real-world networks (see Fig. 6b). Biological networks span a wide range of dimensions (from 1 through 6), whereas all networks in the Software domain lie at dimension 1. A closer look at subdomains (Fig. S9) reveals that web graphs—an Informational subdomain—also have dimension 1, while language networks (also Informational) appear at dimensions 5 and 9.

Subsequently, to enhance confidence in the predictions made by DIMNN, a complementary neural network model was trained. The base architecture was the same as for DIMNN, but the loss function was changed to mean squared error (MSE), with the goal of transforming a classification task into a regression task. Thus, contrary to DIMNN, the output range of the regressor was not restricted to the interval $[1, 10]$. Since regression yields continuous values, each prediction was rounded to the nearest integer. We use the term pseudo-accuracy to denote the accuracy obtained by the regressor after rounding its predictions to integers. Moreover, residual connections were removed.

The regressor model exhibits behavior very consistent with that of DIMNN on synthetic networks (Fig. S7). The percentage of agreement between the two models on the validation set is 79.33% if total persistences are combined with mean chordless cycle densities, and reaches 88.86% when degree-related graph features are added to the training process. The discrepancy between the two models is defined as the absolute value of the difference between their predicted dimensions, computed only from those networks in the validation set on which the two models do not agree. The median discrepancy value was found to be 1, with a mean of 1.05 and a standard deviation of 0.2543 when a full vector of descriptors was fed into the model. Additional details are given in Table S3.

When applied to the real-world network dataset described in Tables S4 and S5, the percentage of agreement between the two models was 64.15% over 53 networks with full feature vectors in the models, with a median discrepancy of 1, a mean of 2.95, and a standard deviation of 3.2032, indicating the occurrence of a few larger discrepancies.

In addition, we validated the explanatory power of the predicted dimension by generating surrogate networks from hyperbolic embeddings obtained with D -Mercator on real networks. Each network was embedded in the dimension predicted by DIMNN, and the inferred node positions were used to generate surrogate networks. We then compared multiple topological

properties to those of the target networks (Figs. S10–S13). Across the tested networks, the surrogates reproduce the degree distribution, clustering spectrum, and $k_{nn}(k)$, and they typically match algebraic connectivity and average shortest-path length, whereas modularity shows weaker agreement, likely reflecting sensitivity to the chosen community detection method. We also compare bond-percolation curves, finding close agreement in the decay of the giant connected component and the inferred percolation threshold (Fig. S14).

3 Discussion

The main contribution of this work is to use densities of cycles (triangles, chordless squares, and chordless pentagons) in a complex network as weights on its edges, and treat each of these three weightings as a filtering function on the underlying graph. Persistence descriptors are computed from the corresponding filtrations and used for estimating the dimensionality of the given network in latent space. The expressiveness of edge-based filtrations defined via densities of chordless cycles is further examined in [53], where they are compared with other motif-based filtrations in graph isomorphism detection and property prediction tasks.

Although topological invariants of graphs play a central role in the study of complex networks, methods from topological data analysis have seen limited adoption in this context [1, 54]. To the best of our knowledge, extended persistent homology of graphs has not previously been applied to dimensionality estimation in complex networks. Ordinary (as opposed to extended) persistent homology is not well-suited for this purpose, since cycles become permanent in graphs. Using Vietoris–Rips simplicial complexes from graph distances is not suitable in our work either, due to the small-world property, and clique complexes of graphs are not optimal either, since triangles become invisible in clique complexes. The extended persistence or lifetime of a cycle in a weighted graph is the difference between the largest weight and the smallest weight of the edges and nodes that form the given cycle. The total lifetime of a set of linearly independent cycles carries relevant information about the geometric structure of the graph.

Other persistence descriptors could be used for analytical purposes. We found persistence of connected components along the filtrations to be less expressive than its cycle-based counterpart. We also remark that the expressivity of cycle persistence is influenced by certain network parameters, such as inverse temperature (β). Indeed, in the point cloud shown in Fig. 2, low values of β tend to separate network dimensions less prominently.

Our results show that total extended persistence computed by means of chordless cycle filtrations yields dimensionality estimates which are comparable with those obtained in previous work [44]; see Table 1. Moreover, total persistence improves the accuracy of a neural network classifier if added to feature vectors containing averages of chordless cycle densities and other graph features such as average degree, as shown in Fig. 5 and in Table S2. Our conclusion is that TDA improves dimensionality estimation using a neural network, even with average cycle densities and degree-related graph descriptors.

In fact, estimating latent network dimensionality without generating surrogates is another

major contribution in this article. For this purpose, a universal database of synthetic graphs has been generated from a uniformly distributed range of parameter values. Real-world network dimensionality estimates can then be obtained by running a neural network model trained on our database. This approach considerably reduces the computation time of the estimations and the amount of memory required for this task. It is far more ambitious than creating an ensemble of surrogates for each given real network under study, as done in the first part of our work.

Fig. 5 highlights the importance of the normalized second moment $\langle k^2 \rangle / \langle k \rangle^2$ in the accuracy of a neural network classifier for latent dimension. This could be due to the fact that the normalized second moment correlates with the exponent γ , and knowing β and γ is crucial for dimensionality estimation. An approximation of β is provided by the average density of triangles C_t , which is also correlated with the total persistence value TP_t .

The agreement between two independently trained neural network models, each optimized with a different loss function, is a strong indicator of robustness and generalization. When models trained with distinct objective functions (in our case, cross-entropy for classification and mean squared error for regression) converge to similar predictions on previously unseen data, it is unlikely that their performance is due to overfitting specific patterns or uninformative noise in the training set. Rather, such an agreement suggests that both models are approximating the same underlying relationship between the input features (topological and density-based descriptors) and the latent embedding dimension. It is remarkable that a neural network trained on synthetic networks yields results closely aligned with earlier work on the latent dimension of real-world networks. Moreover, the discrepancies found between the dimensionality predictions of the two models may serve to detect potential out-of-distribution real networks with respect to the D-GR model in the study dataset, which therefore deserve further examination.

Our framework primarily targets geometric networks, i.e., those in the regime $\beta > D$. Recent work, however, suggests that some networks may instead lie in a weakly geometric region [55, 56]. Extending our approach to this regime and characterizing the interplay between dimensionality and geometricity is a natural direction for future work.

Subsequent research could focus on optimizing the accuracy of a neural network for dimensionality classification of synthetic networks and on improving the performance of latent dimension estimates for real-world networks. On the one hand, this could be achieved by enriching the training database with a wider range of complex network shapes. On the other hand, the neural network architecture could be optimized for the intended tasks. This article only provides a proof of concept for the benefits of a cycle-based filtration in TDA and the feasibility of our deep-learning assisted dimensionality estimation method. While our architecture (Fig. S5) has demonstrated performance consistent with prior work [44] and has achieved accuracy above 83% on synthetic networks, improving its overall effectiveness remains a challenge.

A further avenue concerns the distribution shift between the synthetic networks used for training and the real networks used for evaluation. In machine-learning terms, this setting can be viewed as a domain adaptation problem, where a predictor is trained on a source domain (synthetic graphs) and deployed on a related but distinct target domain (real graphs) [57, 58]. While our current pipeline does not include an explicit adaptation mechanism, a natural ex-

tension would be to augment the model with an adversarial domain discriminator that learns to distinguish synthetic from real networks, while the feature extractor is trained to produce domain-invariant representations, potentially improving transfer and robustness under distribution shift [59].

4 Methods

4.1 Multidimensional geometric soft configuration model

The $\mathbb{S}^D/\mathbb{H}^{D+1}$ model. In the \mathbb{S}^D model [45], a node i is assigned two hidden variables: a hidden degree κ_i , quantifying its importance or popularity, and a position \mathbf{v}_i in a D -dimensional similarity space, represented as a D -sphere. The probability of connection between any pair of nodes i and j follows a gravity law, in which similar nodes are angularly closer and, thus, probably connected [49]. Specifically, nodes i and j are connected with probability

$$p_{ij} = \frac{1}{1 + \left(\frac{R\Delta\theta_{ij}}{(\mu\kappa_i\kappa_j)^{1/D}} \right)^\beta} \quad (1)$$

where D is the dimension of the model, β controls the level of clustering of the network and the coupling of the network with the underlying metric space, μ controls the average degree, and $\Delta\theta_{ij}$ is the angular distance between nodes i and j , which are assigned positions \mathbf{v}_i and \mathbf{v}_j on the D -sphere. The radius R of the sphere is set such that the density of N nodes is 1 (without loss of generality). This yields $R = [\Gamma(\frac{D+1}{2}) N / (2\pi)^{\frac{D+1}{2}}]^{1/D}$, where Γ is the gamma function [49]. For $\beta < D$, networks are unclustered in the infinite-size limit, whereas for $\beta > D$ networks exhibit finite clustering in the thermodynamic limit. Finally, the parameter μ controls the average degree of the network and is defined as

$$\mu = \frac{\beta \Gamma(\frac{D}{2}) \sin(\frac{D\pi}{\beta})}{2\pi^{1+\frac{D}{2}} \langle k \rangle}. \quad (2)$$

The \mathbb{S}^D model is isomorphic to the purely geometric \mathbb{H}^{D+1} model [46] in $(D+1)$ -hyperbolic space by mapping the hidden degree into the radial coordinates as

$$r_i = \hat{R} - \frac{2}{D} \ln \frac{\kappa_i}{\kappa_0}, \quad \text{with } \hat{R} = 2 \ln \left(\frac{2R}{(\mu\kappa_0^2)^{1/D}} \right). \quad (3)$$

A network generation procedure following the $\mathbb{S}^D/\mathbb{H}^{D+1}$ model is described in Algorithm 1 in Supplementary Information, where a cutoff κ_c is calculated to prevent excessive fluctuations in the largest expected degrees when $\gamma < 3$; see [60].

Microcanonical formulation of \mathbb{S}^D model. A microcanonical version of the \mathbb{S}^1 model was first proposed in [51], where it was called geometric randomization model (GR). The GR operates on the sequence of observed node degrees and assigns node positions at random in the similarity space. The network is rewired to maximize the likelihood that the new topology is generated by the \mathbb{S}^1 model while preserving the observed degrees and, thus, the total number of edges.

In this work, we extended GR to higher-dimensional similarity spaces, which we call D-GR. We used D-GR to generate synthetic networks for dimensionality estimation when investigating the relationship between total persistence in homological dimension 1 computed from three types of chordless cycles filtrations and the inferred dimension (see Section 2.1). We want to highlight the main difference with respect to the previous approach [44]. In [44], the authors infer the set of degrees κ from a given network and generate synthetic networks with Eq. (1) given the parameters β and D . Although in [44] the degree distribution of the generated surrogates is very similar to that of the input network, the D-GR procedure is more constrained and maintains the exact degree values of the input network.

In the D-GR model, we assign to each node i a random position in the $D + 1$ dimensional Euclidean space $\mathbf{v}_i \in \mathbb{R}^{D+1}$ with $\|\mathbf{v}_i\| = R$. The nodes are uniformly distributed on the D -sphere using Marsaglia’s algorithm [61]. The rewiring procedure is carried out with the Metropolis–Hastings algorithm, aimed at finding the adjacency matrix that maximizes the likelihood function

$$\mathcal{L} = \prod_{i < j} p_{ij}^{a_{ij}} [1 - p_{ij}]^{1 - a_{ij}} \quad (4)$$

where p_{ij} comes from Eq. (1) and a_{ij} are elements of the adjacency matrix. The method proceeds as described in Algorithm 2 in Supplementary Information.

As shown in [51], the probability of swapping links between nodes i and j and between nodes l and m is given by

$$\frac{\mathcal{L}_n}{\mathcal{L}_c} = \left(\frac{\Delta\theta_{ij} \Delta\theta_{lm}}{\Delta\theta_{il} \Delta\theta_{jm}} \right)^\beta. \quad (5)$$

Notice that Eq. (5) does not depend on the dimension D .

To validate our approach, we generated synthetic networks with known dimension using the \mathbb{S}^D model (see Algorithm 1 in the Supplementary Information), with the following parameters: the number of nodes (N) was set to 500 and the average degree $\langle k \rangle$ to 10; dimension (D) ranging from 1 to 7; the inverse temperature parameter (β) taking values $1.5 D$ and $2.5 D$; and the power-law exponent (γ) taking values 2.7 and 3.5. For each set of parameters, we generated 10 network realizations, thus obtaining 280 synthetic networks. For each of these, we produced a set of surrogates using the D-GR method (Algorithm 2 in the Supplementary Information), scanning over different values: dimension (D) ranging from 1 to 7 and rescaled inverse temperature β/D ranging from 1.2 to 3.0 with steps of 0.1. The geometric randomization was repeated 10 times, yielding 1 330 surrogates per synthetic network. At the end of this procedure, we obtained a total of 372 400 networks, that were used in the confusion matrices in Fig. 3.

To infer the dimension of real networks, we proceeded in a similar fashion. We applied the D-GR method with the same set of parameters described above, yielding a total of 1 330 surrogates for each real network. Inferred dimensions of real networks are described in Table 1.

It is worth noting some limitations of this method. The acceptance probability depends on the parameter β . For very large values of β , the acceptance ratio becomes binary, i.e., moves that increase the likelihood are almost always accepted, and those that decrease it are almost always rejected, and the likelihood plateau cannot be reached. Thus, the algorithm is restricted to moderate values of β and corresponding values of the clustering coefficient. Techniques such as simulated annealing or parallel tempering can restore good mixing in the large β regime. In this work, however, we restrict our attention to real networks whose topology is well-captured by a moderate value of β .

Moreover, even though the $\mathbb{S}^D/\mathbb{H}^{D+1}$ model captures a wide range of topological network properties, some real networks may lie outside the range of values and are located further away from the surrogate networks in the persistence (TP_t, TP_s, TP_p) and mean cycle density (C_t, C_s, C_p) configuration spaces.

4.2 Graph features

Densities of chordless cycles. Let $G = (V, E)$ be a graph and $e_{ij} = \{v_i, v_j\}$ be an edge between nodes v_i and v_j in E , with degrees $k_i > 1$ and $k_j > 1$, respectively. The density of triangles corresponding to the edge e_{ij} , also called edge clustering coefficient [62], is the number $\#\Delta_{ij}$ of edge triangles in G containing e_{ij} divided by the maximum possible number of triangles in G containing e_{ij} given the degrees k_i and k_j , that is,

$$C_t(e_{ij}) = \frac{\#\Delta_{ij}}{\min(k_i, k_j) - 1}. \quad (6)$$

An edge cycle is chordless if there is no edge between its nodes except those that form the cycle. Density of squares, denoted $C_s(e_{ij})$, is defined by dividing the number of chordless edge squares in G containing e_{ij} by the maximum possible number of such squares given the degrees k_i and k_j and the existing triangles through e_{ij} in G . To define the density of pentagons $C_p(e_{ij})$ per edge e_{ij} , we count the number of chordless pentagons containing e_{ij} and normalize it by the maximum possible number of such pentagons, assuming known the degrees of v_i and v_j and the degrees of their respective neighbors. One could also define similar densities per node. However, in [62], the authors found the definitions relative to edges to be more stable with respect to degree heterogeneity than those relative to nodes.

Regarding the computational cost of computing chordless cycles, the time complexities for computing densities of triangles, squares, and pentagons are $O(N\langle k \rangle^2)$, $O(N\langle k \rangle^3)$, and $O(N\langle k \rangle^4)$, respectively, where $\langle k \rangle$ is the average degree and $N = \#E$ is the number of edges. In sparse graphs, $\langle k \rangle \ll N$. Although the computational cost for large graphs may be high, the computations can be easily parallelized or implemented on a GPU using CUDA [63].

Extended persistent homology. In this work, we use persistent homology to extract information from a graph G equipped with a filtration $\{G_t\}$, where t is a real-valued parameter. Persistence refers to the evolution of cycles along the values of the given filtration.

Homology of graphs is a special case of simplicial homology of simplicial complexes [64]. In the case of graphs, homology is determined by two numbers, called Betti numbers, namely the number β_0 of connected components and the cardinality β_1 of a maximal set of linearly independent cycles. Cycles are finite formal sums $z = \sum \lambda_i e_i$ of oriented edges with coefficients $\lambda_i = \pm 1$ such that $\sum \lambda_i \partial e_i = 0$, where, for an edge $e \in E$ from v_0 to v_1 , we denote $\partial e = v_1 - v_0$. Thus, cycles are algebraic representations of closed edge paths in the given graph.

In a filtered simplicial complex $\{K_t\}$, the birth of a cycle z of any dimension is the t -value at which z appears in K_t , and the death of z is the t -value at which z becomes the boundary of a higher chain. Hence, in the case of a filtered graph, the birth of a cycle z is the t -value at which z is formed, and the death value is infinite, since a graph does not contain higher simplices.

To avoid infinite persistence values, we use extended persistence, as in [65, 66]. For a graph $G = (V, E)$ equipped with a node weighting $w_V: V \rightarrow \mathbb{R}$, the sublevel filtration $\{G_t\}$ is defined as $G_t = (V_t, E_t)$, where $V_t = \{v \in V \mid w_V(v) \leq t\}$ and E_t is the subset of E spanned by V_t , and the superlevel filtration $\{G^t\}$ is defined as $G^t = (V^t, E^t)$, where $V^t = \{v \in V \mid w_V(v) \geq t\}$ and E^t is the subset of E spanned by V^t . Then the extended persistence of a cycle z in G is defined as $|d - b|$, where b is the birth value of z in the sublevel filtration $\{G_t\}$ and d is the birth value of z in the superlevel filtration $\{G^t\}$. Note, however, that no edge weighting can be defined on G compatibly with w_V which is consistent with both $\{G_t\}$ and $\{G^t\}$, in general.

In our work, chordless cycle filtrations are defined by means of edge weightings, not node weightings. Although it is perfectly possible to exchange the roles of nodes and edges in the definitions of sublevel and superlevel filtrations, extended persistence calculations have been carried out using the Python library Gudhi [67], which only provides software for computing extended persistence of graphs with a weighting on their nodes (see Section 2.1 of [66]). Therefore, we introduce a method to replace a graph $G = (V, E)$ equipped with an edge weighting $w_E: E \rightarrow \mathbb{R}$ by a topologically equivalent graph $G' = (V', E')$ in which $E \subseteq E'$ and endowed with a node weighting $w_{V'}: V' \rightarrow \mathbb{R}$ yielding the same extended persistence diagram as G .

To achieve this, we split each node $v_i \in V$ of degree k_i into k_i distinct nodes $v_{i,j} \in V'$, where j ranges over the subindices of the neighbors of v_i . Each node $v_{i,j}$ is assigned weight $w_{V'}(v_{i,j}) = w_E(e_{ij})$ where $e_{ij} = \{v_i, v_j\}$. The set E' contains horizontal edges $\{v_{i,j}, v_{j,i}\}$ for all i, j , joining nodes with the same weight, and vertical edges, connecting each string of nodes $v_{i,j}$ with a fixed i value, sequentially from smallest weight to largest, as in Fig. 1. Hence, the set of horizontal edges in E' is in bijective correspondence with E , and the weighting $w_{V'}$ is consistent with w_E on all edges of E , in both the sublevel and superlevel filtrations of G' .

We call G' a degree-splitting subdivision of G . An example is shown in Fig. 1, and a corresponding sequence of sublevel graphs and superlevel graphs explaining the choice of generating cycles for extended persistence is depicted in Fig. S4.

The graph G' is then fed into Gudhi [67] as a node-weighted graph, and total persistence $\sum |d_i - b_i|$ of cycles in the corresponding extended persistence diagram is recorded (relative

cycles provided by the Gudhi software, if any, are discarded). The graph G' has the same Betti numbers as G at the same thresholds, since G is obtained from G' by contracting vertical edges, which does not change homology.

Degree-related graph features. The following descriptors from complex network theory are calculated to train the models used in Section 2.2. For each graph $G = (V, E)$, we consider the average degree of its nodes, $\langle k \rangle = (1/\#V) \sum_{v \in V} \text{deg}(v)$. The minimal degree k_{\min} and maximal degree k_{\max} are also considered. The average neighbor degree of G is the average of the mean neighbor degree of its nodes:

$$\langle k_{\text{nn}} \rangle = \frac{1}{\#V} \sum_{v \in V} \left(\frac{1}{\text{deg}(v)} \sum_{u \sim v} \text{deg}(u) \right).$$

The normalized second moment of G is defined as the quotient $\langle k^2 \rangle / \langle k \rangle^2$, where we denote $\langle k^2 \rangle = (1/\#V) \sum_{v \in V} \text{deg}(v)^2$. Its significance is due to the fact that it correlates inversely with the power-law exponent γ in scale-free networks [68].

4.3 Database and neural network architecture

For better understanding and comparison of the results, we used the same real-world networks as in [44]. These are undirected networks with fewer than 100 000 nodes from very different domains. Tables S4 and S5 include a detailed description of the data as reported in [44].

SYNNET dataset for inferring dimension using neural networks. Deep networks require large amounts of data to avoid overfitting and to learn robust features, due to their large number of parameters. In fact, neural networks generally require large datasets to achieve high performance [69, 70]. However, with the \mathbb{S}^D model, we tackle this challenge by generating plenty of synthetic graphs with known dimensions to train neural networks.

We prepared a dataset of 792 000 synthetic networks generated using the \mathbb{S}^D model by means of the method described in Algorithm 1 in the Supplementary Information with the following input parameters: dimension (D) from 1 to 10; number of nodes (N) with values 200, 400, 750, 1000, 2500; power-law exponent (γ) with values 2.2, 2.4, 3.0, 4.0, 5.0; average degree ($\langle k \rangle$) with values 4, 8, 12, 25; and rescaled inverse temperature (β/D) with values 1.2, 1.4, 1.6, 1.8, 2.0, 2.2, 2.5, 2.8, 3.0, 3.5, 4.0, 5.0. For each set of parameters, 66 network realizations were made. The dataset covers a wide range of network properties—in particular, $\langle k_{\text{nn}} \rangle \in [4.4, 300.8]$, $C_t \in [0.073, 0.891]$, $C_s \in [0.00505, 0.24193]$, $C_p = [0.00010, 0.02165]$.

The database contains a set of feature vectors for each network, aiming for its use in a multilayer perceptron (MLP), which is described in Section 4.3. The following descriptors were computed, resulting in a 12-dimensional feature vector: N , $\langle k \rangle$, $\langle k^2 \rangle / \langle k \rangle^2$, k_{\min} , k_{\max} , $\langle k_{\text{nn}} \rangle$, C_t , C_s , C_p , TP_t , TP_s , TP_p , as described in Section 4.2. Once a MLP is trained, the same features are computed for real networks in order to predict their dimensions.

It is worth mentioning that some real networks could have very distinct topological properties that are not included in our dataset. Hence, neural networks might be prone to misclassifying them. We can overcome this issue by extending the range of parameters used to generate the dataset. In the released code, we provide a check for the out-of-distribution parameters.

Neural network DIMNN model. For the classification task, a deep multilayer perceptron (MLP) architecture combined with the techniques of residual networks (ResNets) was used, as in [71]. Deeper neural networks are capable of capturing highly non-linear relationships in data, but training very deep architectures often leads to optimization difficulties such as vanishing gradients. ResNets address these issues through the introduction of skip (residual) connections, which add the input of a layer to its output. Such mappings facilitate gradient flow and stabilize training, enabling the construction of deeper and more expressive models [72].

The proposed architecture integrates residual connections into a fully connected MLP, allowing the model to benefit from the depth of representation while maintaining stable training dynamics. The network comprises 21 hidden layers with ReLU activations, and the configuration depicted in Fig. S5. The output layer uses a softmax activation. A dropout regularization of 50% is applied throughout the network to mitigate overfitting and enhance generalization. The training process was carried out using AdamW as optimizer, with a learning rate of 0.0005 and incorporating early stopping to prevent overfitting. This architecture benefits from the expressive power of deep learning while incorporating the robustness of residual learning, making it convenient for the proposed classification task.

Data Availability

The real network datasets used in this study are available from the sources referenced in the manuscript and the Supplementary Information. The SYNNET dataset of 792 000 synthetic networks generated with the \mathbb{S}^D model used for neural network training, along with the neural network checkpoints, is available on Zenodo [73].

Code Availability

The open-source code generated in this work, along with the code to reproduce the figures, is available on GitHub at <https://github.com/networkgeometry/detecting-dimensionality-TDA-DimNN>.

References

- [1] Rubén Ballester and Bastian Rieck. On the expressivity of persistent homology in graph learning. In *The Third Learning on Graphs Conference*, (2024).

- [2] Gunnar Carlsson and Mikael Vejdemo-Johansson. *Topological data analysis with applications*. Cambridge University Press, (2021).
- [3] Herbert Edelsbrunner and John L. Harer. *Computational Topology: An Introduction*. American Mathematical Society, (2022).
- [4] Larry Wasserman. Topological data analysis. *Annual Review of Statistics and its Application* **5**, 501–532 (2018).
- [5] Robin Forman. Bochner’s method for cell complexes and combinatorial Ricci curvature. *Discrete & Computational Geometry* **29**, 323–374 (2003).
- [6] Linton C. Freeman. A set of measures of centrality based on betweenness. *Sociometry* **40**(10), 35–41 (1977).
- [7] Ginestra Bianconi. *Higher-Order Networks*. Cambridge University Press, (2021).
- [8] Omer Bobrowski and Dmitri Krioukov. Random simplicial complexes: Models and phenomena. In *Higher-Order Systems*, 59–96. Springer (2022).
- [9] Daniel Hernández Serrano, Juan Hernández Serrano, and Darío Sánchez Gómez. Simplicial degree in complex networks. Applications of topological data analysis to network science. *Chaos, Solitons and Fractals* **137**, 109839, Aug (2020).
- [10] Vsevolod Salnikov, Daniele Cassese, and Renaud Lambiotte. Simplicial complexes and complex systems. *European Journal of Physics* **40**(1), 014001 (2018).
- [11] Joaquín J. Torres and Ginestra Bianconi. Simplicial complexes: Higher-order spectral dimension and dynamics. *Journal of Physics: Complexity* **1**(1), 015002 (2020).
- [12] Danijela Horak, Slobodan Maletić, and Milan Rajković. Persistent homology of complex networks. *J. Stat. Mech.* **2009**, P03034 (2009).
- [13] Dane Taylor et al. Topological data analysis of contagion maps for examining spreading processes on networks. *Nature Communications* **6**(1), 7723 (2015).
- [14] Harish Kannan, Emil Saucan, Indraava Roy, and Areejit Samal. Persistent homology of unweighted complex networks via discrete Morse theory. *Scientific Reports* **9**(13817) (2019).
- [15] Audun Myers, Elizabeth Munch, and Firas A. Khasawneh. Persistent homology of complex networks for dynamic state detection. *Physical Review E* **100**(022314) (2019).
- [16] Marco Guerra, Alessandro De Gregorio, Ulderico Fugacci, Giovanni Petri, and Francesco Vaccarino. Homological scaffold via minimal homology bases. *Scientific Reports* **11**(5355) (2021).

- [17] Bukyoung Jhun. Topological analysis of the latent geometry of a complex network. *Chaos* **32**(013116) (2022).
- [18] G. Petri, P. Expert, F. Turkheimer, R. Carhart-Harris, D. Nutt, P. J. Hellyer, and F. Vaccarino. Homological scaffolds of brain functional networks. *Journal of the Royal Society Interface* **11**(101) (2014).
- [19] Chad Giusti, Eva Pastalkova, Carina Curto, and Vladimir Itsko. Clique topology reveals intrinsic geometric structure in neural correlations. *PNAS* **112**(44), 13455–13460 (2015).
- [20] Ann E. Sizemore, Jennifer E. Phillips-Cremins, Robert Ghrist, and Danielle S. Bassett. The importance of the whole: Topological data analysis for the network neuroscientist. *Network Neuroscience* **3**(3), 656–673 (2019).
- [21] Soumya Das, D. Vijay Anand, and Moo K. Chung. Topological data analysis of human brain networks through order statistics. *PLoS One* **18**(3), e0276419 (2023).
- [22] Ernst Ising. Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift für Physik* **31**, 253–258 (1925).
- [23] Lars Onsager. Crystal statistics. I. A two-dimensional model with an order-disorder transition. *Physical Review* **65**(3-4), 117 (1944).
- [24] Sidney Redner. *A Guide to First-Passage Processes*. Cambridge University Press, (2001).
- [25] Aryeh Dvoretzky and Paul Erdős. Some problems on random walk in space. In *Proc. Second Berkeley Symp. Math. Statist. Probab*, 353–367, (1951).
- [26] Stéphane Condamin, Olivier Bénichou, Vincent Tejedor, Raphaël Voituriez, and Joseph Klafter. First-passage times in complex scale-invariant media. *Nature* **450**(7166), 77–80 (2007).
- [27] Joseph Polchinski. *String Theory*. Cambridge Monographs on Mathematical Physics. Cambridge University Press, (1998).
- [28] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science* **290**(5500), 2319–2323 (2000).
- [29] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation* **15**(6), 1373–1396 (2003).
- [30] Edward A. Codling, Michael J. Plank, and Simon Benhamou. Random walk models in biology. *Journal of The Royal Society Interface* **5**(25), 813–834 (2008).
- [31] Peter H. von Hippel and Otto G. Berg. Facilitated target location in biological systems. *Journal of Biological Chemistry* **264**(2), 675–678 (1989).

- [32] Robert Peach, Alexis Arnaudon, and Mauricio Barahona. Relative, local and global dimension in complex networks. *Nature Communications* **13**(1), 3088 (2022).
- [33] Víctor M Eguíluz, Emilio Hernández-García, Oreste Piro, and Konstantin Klemm. Effective dimensions and percolation in hierarchically structured scale-free networks. *Physical Review E* **68**(5), 055102 (2003).
- [34] Chaoming Song, Lazaros K. Gallos, Shlomo Havlin, and Hernán A. Makse. How to calculate the fractal dimension of a complex network: the box covering algorithm. *Journal of Statistical Mechanics: Theory and Experiment* **2007**(03), P03006–P03006 (2007).
- [35] O. Shanker. Defining dimension of a complex network. *Modern Physics Letters B* **21**(06), 321–326 (2007).
- [36] Daijun Wei, Bo Wei, Yong Hu, Haixin Zhang, and Yong Deng. A new information dimension of complex networks. *Physics Letters A* **378**(16-17), 1091–1094 (2014).
- [37] J. S. Kim, K.-I. Goh, B. Kahng, and D. Kim. A box-covering algorithm for fractal scaling in scale-free networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science* **17**(2) (2007).
- [38] Li Daqing, Kosmas Kosmidis, Armin Bunde, and Shlomo Havlin. Dimension of spatially embedded networks. *Nature Physics* **7**(6), 481–484 (2011).
- [39] Richard C. Tillquist, Rafael M. Frongillo, and Manuel E. Lladser. Getting the lay of the land in discrete space: A survey of metric dimension and its applications. *SIAM Review* **65**(4), 919–962 (2023).
- [40] Lucas Lacasa and Jesús Gómez-Gardenes. Correlation dimension of complex networks. *Physical Review Letters* **110**(16), 168703 (2013).
- [41] Raffaella Burioni and Davide Cassi. Universal properties of spectral dimension. *Physical Review Letters* **76**, 1091–1093, Feb (1996).
- [42] Konstantin Avrachenkov, Laura Cottatellucci, and Mounia Hamidouche. Eigenvalues and spectral dimension of random geometric graphs in thermodynamic regime. In *International Conference on Complex Networks and Their Applications*, 965–975. Springer, (2019).
- [43] Marián Boguñá, Ivan Bonamassa, Manlio De Domenico, Shlomo Havlin, Dmitri Krioukov, and M. Ángeles Serrano. Network geometry. *Nature Reviews Physics* **3**(2), 114–135 (2021).
- [44] P. Almagro, M. Boguñá, and M. Á. Serrano. Detecting the ultra low dimensionality of real networks. *Nature Communications* **13**, 6096, Oct (2022).

- [45] M. Ángeles Serrano, Dmitri Krioukov, and Marián Boguñá. Self-similarity of complex networks and hidden metric spaces. *Physical Review Letters* **100**(7), 078701 (2008).
- [46] Gabriel Budel, Maksim Kitsak, Rodrigo Aldecoa, Konstantin Zuev, and Dmitri Krioukov. Random hyperbolic graphs in $d+1$ dimensions. *Physical Review E* **109**(5), 054131 (2024).
- [47] Robert Jankowski, Antoine Allard, Marián Boguñá, and M. Ángeles Serrano. The D -Mercator method for the multidimensional hyperbolic embedding of real networks. *Nature Communications* **14**(1), 7585 (2023).
- [48] Blaise Désy, Patrick Desrosiers, and Antoine Allard. Dimension matters when modeling network communities in hyperbolic spaces. *PNAS Nexus* **2**(5), pgad136 (2023).
- [49] M. Á. Serrano and M. Boguñá. *The Shortest Path to Network Geometry*. Cambridge University Press, Cambridge, UK, (2021).
- [50] D. Krioukov, F. Papadopoulos, M. Kitsak, A. Vahdat, and M. Boguñá. Hyperbolic geometry of complex networks. *Physical Review E – Statistical, Nonlinear, and Soft Matter Physics* **82**(9), Sep (2010).
- [51] Michele Starnini, Elisenda Ortiz, and M. Ángeles Serrano. Geometric randomization of real networks with prescribed degree sequence. *New Journal of Physics* **21**(5), 053039 (2019).
- [52] Amir Ghasemian, Homa Hosseinmardi, and Aaron Clauset. Evaluating overfit and underfit in models of network community structure. *IEEE Transactions on Knowledge and Data Engineering* **32**(9), 1722–1735 (2019).
- [53] Meritxell Vila-Miñana, Robert Jankowski, Aina Ferrà Marcús, Rubén Ballester, M. Ángeles Serrano, and Carles Casacuberta. Motif-based filtrations for persistent homology: A framework for graph isomorphism and property prediction. *arXiv preprint arXiv:2604.15265* (2026).
- [54] M. E. Aktas, E. Akbas, and A. El Fatmaoui. Persistence homology of networks: Methods and applications. *Applied Network Science* **4**(61) (2019).
- [55] Jasper van der Kolk, M. Ángeles Serrano, and Marián Boguñá. An anomalous topological phase transition in spatial random graphs. *Communications Physics* **5**(1), 245 (2022).
- [56] Jasper van der Kolk, M. Ángeles Serrano, and Marián Boguñá. Random graphs and real networks with weak geometric coupling. *Physical Review Research* **6**(1), 013337 (2024).
- [57] Ali Farahani, Sajad Voghoei, Khaled Rasheed, and Hamid R. Arabnia. A brief review of domain adaptation. In *Advances in Data Science and Information Engineering: Proceedings from ICDATA 2020 and IKE 2020*, 877–894. Springer (2021).

- [58] Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. *Neurocomputing* **312**, 135–153 (2018).
- [59] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 7167–7176, (2017).
- [60] Marián Boguñá, Romualdo Pastor-Satorras, and Alessandro Vespignani. Cut-offs and finite size effects in scale-free networks. *The European Physical Journal B* **38**, 205–209 (2004).
- [61] George Marsaglia. Choosing a point from the surface of a sphere. *The Annals of Mathematical Statistics* **43**(2), 645–646 (1972).
- [62] M. Ángeles Serrano and Marián Boguñá. Clustering in complex networks. I. General formalism. *Physical Review E – Statistical, Nonlinear, and Soft Matter Physics* **74**(5), 056114 (2006).
- [63] John Nickolls, Ian Buck, Michael Garland, and Kevin Skadron. Scalable parallel programming with CUDA. In *ACM SIGGRAPH 2008 Classes, SIGGRAPH '08* (Association for Computing Machinery, New York, NY, USA, 2008).
- [64] Allen Hatcher. *Algebraic Topology*. Cambridge University Press, (2002).
- [65] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Extending persistence using Poincaré and Lefschetz duality. *Foundations of Computational Mathematics* **9**(1), 79–103 (2009).
- [66] M. Carrière, F. Chazal, Y. Ike, T. Lacombe, M. Royer, and Y. Umeda. PersLay: A neural network layer for persistence diagrams and new graph topological signatures. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 108 of *PMLR*, (2020).
- [67] The GUDHI Project. *GUDHI User and Reference Manual*. GUDHI Editorial Board, (2015).
- [68] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics* **74**(47) (2002).
- [69] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 1097–1105, (2012).
- [70] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature* **521**(7553), 436–444 (2015).

- [71] Hugo Touvron et al. ResMLP: Feedforward networks for image classification with data-efficient training. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **45**(4), 5314–5321 (2023).
- [72] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778, (2016).
- [73] Aina Ferrà Marcús, Robert Jankowski, Meritxell Vila-Miñana, Carles Casacuberta, and M. Ángeles Serrano. Chordless cycle filtrations for dimensionality detection in complex networks via topological data analysis, , April (2026). Dataset. doi: 10.5281/zenodo.19470037. Available at <https://doi.org/10.5281/zenodo.19470037>.

Acknowledgments

A.F.M. was supported by MCIN/AEI under grant PRE2020-094372; A.F.M. and C.C. were partially funded through MCIN/AEI grants PID2020-117971GB-22 and PID2022-136436NB-I00, as well as AGAUR grant 2021 SGR 00697; R.J. acknowledges support from the fellowship FI-SDUR funded by Generalitat de Catalunya; M.A.S. acknowledges support from grant no. TED2021-129791B-I00 funded by MCIN/AEI/10.13039/501100011033 and by *European Union NextGenerationEU/PRTR*, and grant no. PID2022-137505NB-C22 funded by MCIN/AEI/10.13039/501100011033 and by ERDF’s *A way of making Europe*.

Author Contributions Statement

M.A.S., M.V.M. and A.F.M. designed research; M.A.S. and R.J. contributed network science methods; C.C., A.F.M. and M.V.M. contributed persistent homology methods; A.F.M. and R.J. performed the computations and numerical simulations; A.F.M. generated the DIMNN code; R.J. generated the SYNNET dataset. All authors conducted research, analyzed the results, discussed implications, and contributed to the writing of the manuscript.

Competing Interests Statement

The authors declare no competing interests.

Tables

Table 1: **Inferred dimensionality of real-world networks.** Comparison of the inferred dimension between three different methods: (1) Mean densities of chordless cycles; (2) TDA: Persistent homology from chordless cycle density filtrations; (3) DIMNN: A neural network trained on a database of synthetic networks (introduced in Section 2.2). The $1/\rho$ values are inverses of the congruency indices; higher values correspond to a closer match between the real network and its surrogate models.

Network	Domain	Densities		TDA		DIMNN
		dim	$1/\rho$	dim	$1/\rho$	dim
Human2-C	Biological (Connectome)	3	0.07	1	0.21	2
Human-M	Biological (Metabolic)	3	0.19	4	0.46	3
Cargoships	Economic (Trade)	3	0.10	2	0.09	5
Bible-CO	Informational (Language)	4	0.05	1	0.04	5
Jazz-CA	Social (Collaboration)	2	0.06	2	0.41	2
EUEmail	Social (Communication)	2	0.09	1	0.28	2
Friends-ON	Social (Communication)	6	0.06	6	0.19	7
Friends-OFF	Social (Offline)	8	0.72	8	0.28	10

Figures

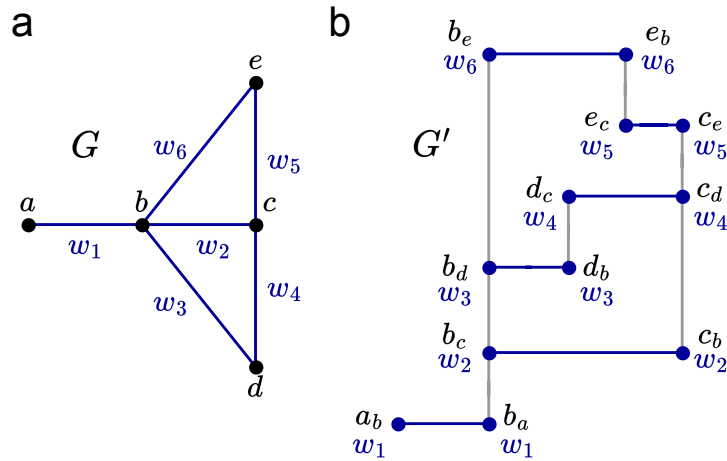


Figure 1: **Extended persistence for an edge-weighted graph filtration.** Topologically equivalent graphs equipped with (a) an edge weighting and (b) a node weighting, with the same extended persistence barcode. Assuming that $w_1 < \dots < w_6$, there are two generating cycles with birth-death coordinates (w_4, w_2) and (w_6, w_3) , yielding a total extended persistence of $|w_2 - w_4| + |w_3 - w_6|$. See Section 4.2 for details.

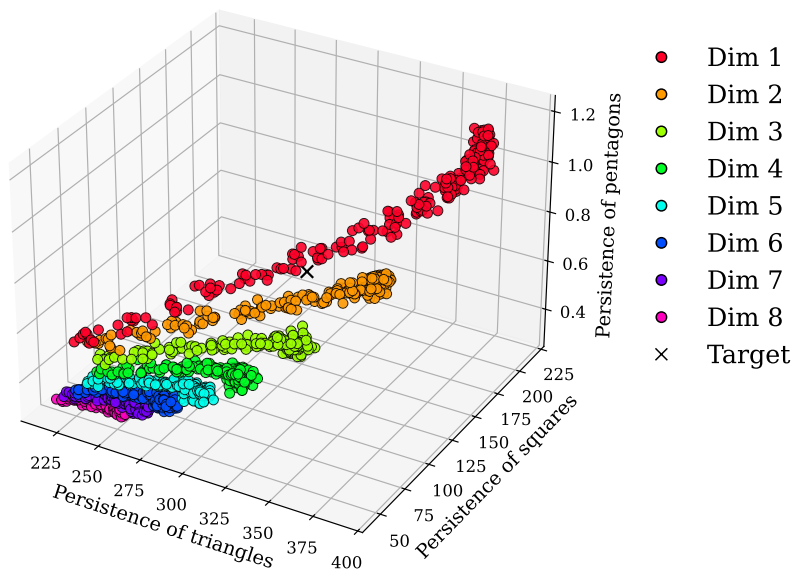


Figure 2: **Total persistence values obtained from the human connectome.** 3D view of a point cloud representing an ensemble of 1330 surrogates of the Human2 network (connectome of the human brain, including one hemisphere) in the configuration space of total persistence computed from three chordless cycle densities (triangles, squares, and pentagons). Points are coloured by dimension. The target network is marked with a black cross.

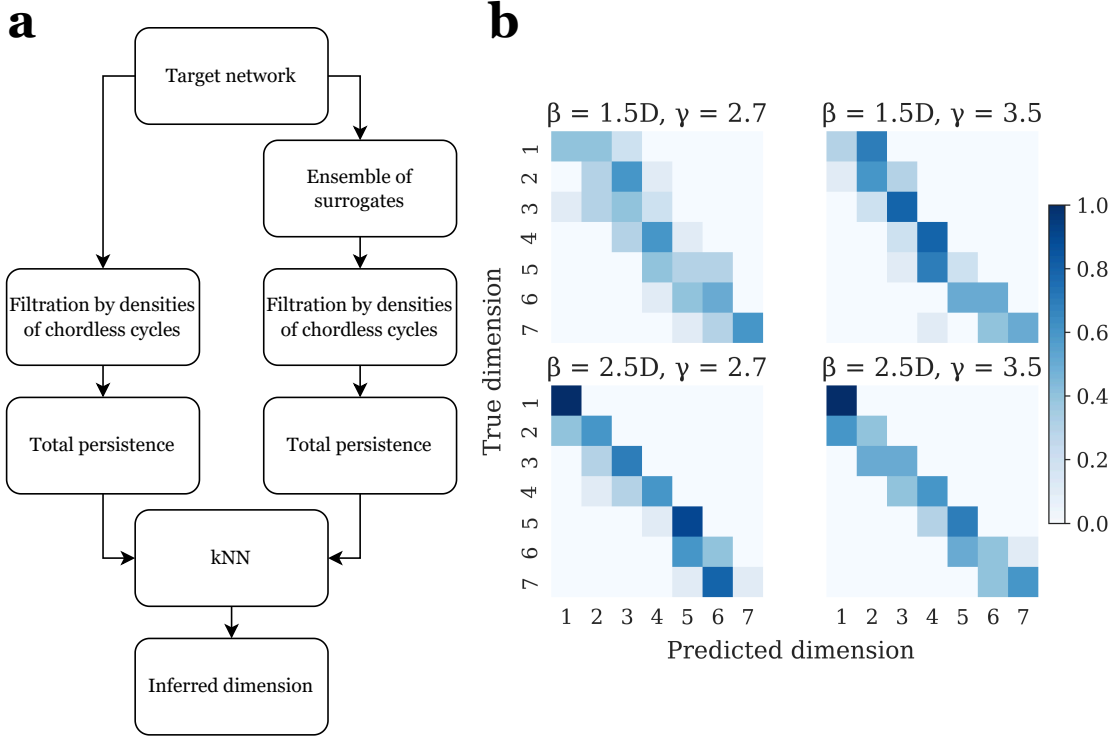


Figure 3: **Dimensionality estimation with persistence descriptors.** (a) Pipeline of our first method and (b) confusion matrices. Given a target network, we generate an ensemble of surrogates and, for each of them (including the given network), we compute densities of chordless cycles. We equip the graphs with these weightings and perform a topological analysis using extended persistence. For each network, we compute the total persistence of cycles and use a kNN classifier in the configuration space to infer a dimensionality D^* for the real network. The upper two confusion matrices show results obtained with synthetic networks for $\beta = 1.5D$ and two different values $\gamma = 2.7$ and $\gamma = 3.5$. This choice of β corresponds to the small-world phase even if $\gamma > 3$. The lower two confusion matrices show results for $\beta = 2.5D$; in this case, networks with $\gamma > 3$ are large worlds. In each matrix, the rows correspond to the true dimension D , and the columns correspond to the inferred dimension D^* . Thus, the j -th box in the i -th row shows the fraction of i -dimensional networks that were classified as j -dimensional. The darker the color, the greater the number. Each matrix is evaluated with 70 synthetic networks.

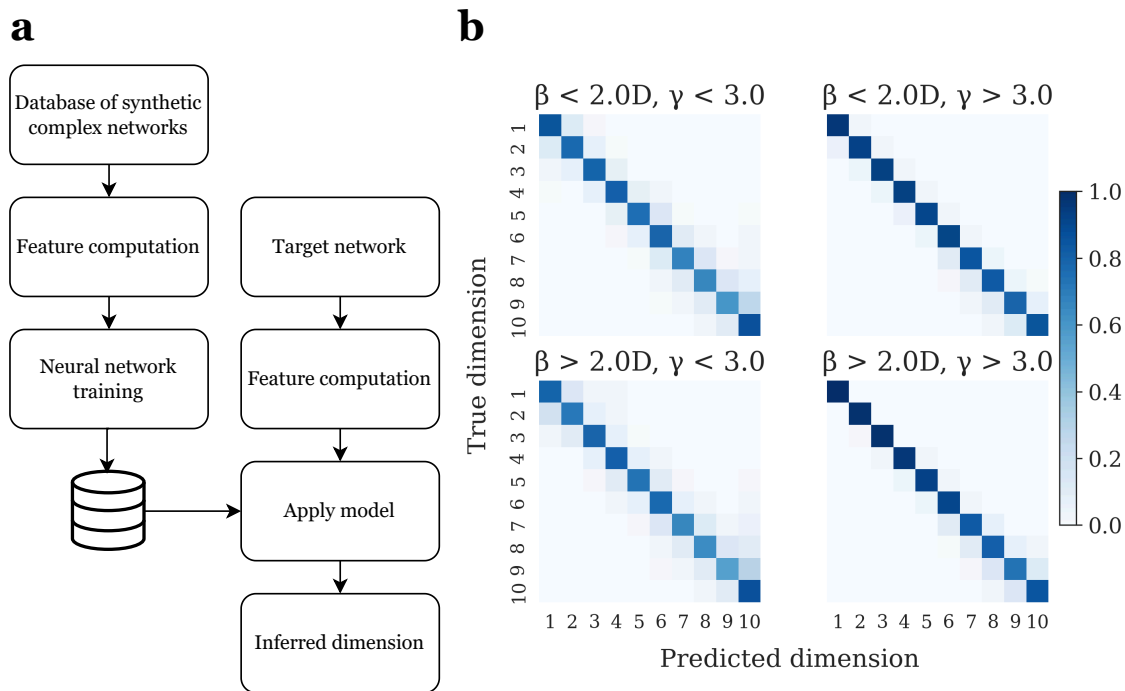


Figure 4: **Dimensionality estimation using DIMNN.** (a) Pipeline of method and (b) confusion matrices. A multilayer perceptron is trained on a database of synthetic complex networks, with features coming from filtrations by densities of chordless cycles concatenated with averages of chordless cycle densities and degree-related graph features. The parameters of the trained model are stored and the model is applied to target networks. Confusion matrices show results for the four combinations of $\beta < 2D$ versus $\beta > 2D$, and $\gamma < 3$ versus $\gamma > 3$, from a range of values $1.2D$ to $5.0D$ for β and 2.2 to 5.0 for γ . In each matrix, the rows correspond to the true dimension D and the columns correspond to the inferred dimension D^* .

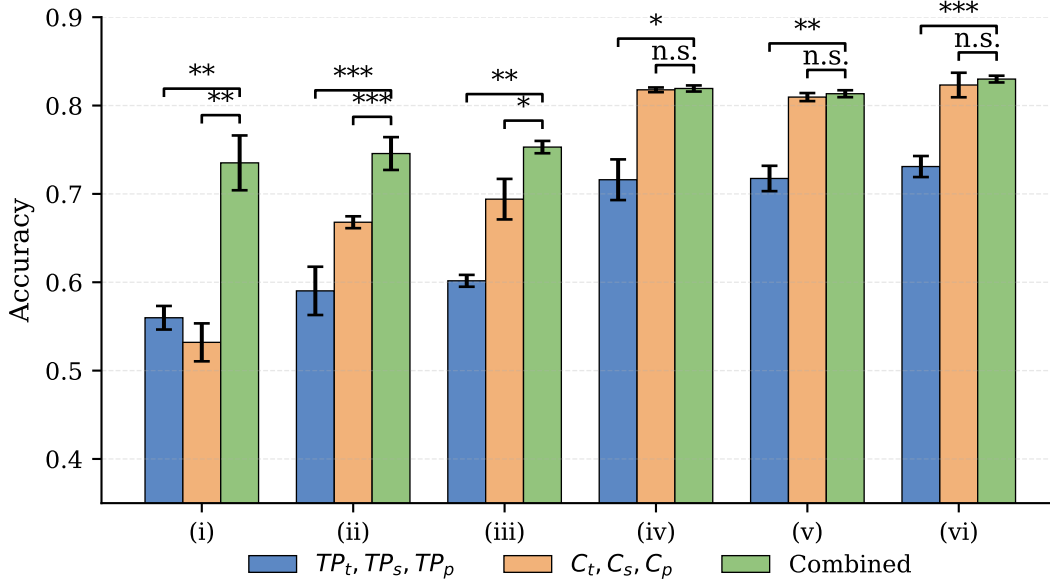


Figure 5: **Validation accuracies of DIMNN using the SYNNET database.** Blue color: Accuracies obtained using total persistences TP_t, TP_s, TP_p , plus cumulative features; Orange color: Accuracies using average cycle densities C_t, C_s, C_p , plus cumulative features; Green color: Accuracies obtained combining total persistences and mean cycle densities plus cumulative features. Successive columns correspond to incorporating one after the other the following additional features into the model: (i) no added features; (ii) number of nodes N ; (iii) number of nodes and average degree $\langle k \rangle$; (iv) number of nodes, average degree, and normalized second moment $\langle k^2 \rangle / \langle k \rangle^2$; (v) number of nodes, average degree, normalized second moment, minimum degree k_{\min} and maximum degree k_{\max} ; (vi) number of nodes, average degree, normalized second moment, minimum degree, maximum degree, and mean average neighbor degree $\langle k_{nn} \rangle$. Results are averaged over five realizations of DIMNN, and error bars indicate standard deviation. Horizontal brackets indicate paired comparisons between the combined model and each baseline; significance corresponds to paired t -tests with $p < 0.05$ (*), $p < 0.01$ (**), and $p < 0.001$ (***), while “n.s.” indicates a non-significant difference.

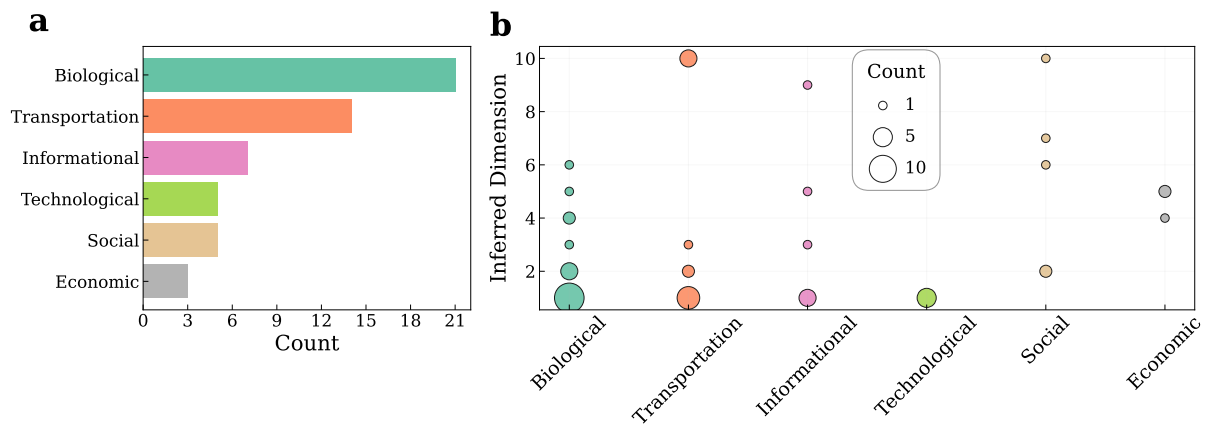


Figure 6: **Dimensionality estimation of real networks with DIMNN.** (a) Number of networks for each network domain. (b) Inferred dimension of real networks using DIMNN, grouped by network domain. The marker size indicates the number of networks.