Variability in higher order structure of noise added to weighted networks

Ann S. Blevins^{1,2}, Jason Z. Kim ^{1,2} & Dani S. Bassett ¹✉

The complex behavior of many real-world systems depends on a network of both strong and weak edges. Distinguishing between true weak edges and low-weight edges caused by noise is a common problem in data analysis, and solutions tend to either remove noise or study noise in the absence of data. In this work, we instead study how noise and data coexist, by examining the structure of noisy, weak edges that have been synthetically added to model networks. We find that the structure of low-weight, noisy edges varies according to the topology of the model network to which it is added, that at least three qualitative classes of noise structure emerge, and that these noisy edges can be used to classify the model networks. Our results demonstrate that noise does not present as a monolithic nuisance, but rather as a nuanced, topology-dependent, and even useful entity in characterizing higher-order network interactions.

¹University of Pennsylvania, Philadelphia, USA. ²These authors contributed equally: Ann S. Blevins and Jason Z. Kim. ✉email: dsb@seas.upenn.edu

In weighted network analyses, edges that correspond to noise are seen as a nuisance. This view derives from the fact that they hamper our capacity to distinguish between weak edges that correspond to significant system features and weak edges that are false positives or noise. Previous research in neuroscience^{1–3}, social networks^{4,5}, and molecular biology⁶ has demonstrated the importance of weak, real edges, to the function of a network. However, weighted networks collected from data are also often plagued by noise either in the form of imprecise edge weights or spurious connections (Fig. 1a). Unfortunately, such noise often affects the signal-to-noise ratio of weak edges more than strong edges, thus complicating our understanding of weak edge network structure.

How does one deal with this undesirable noise in estimates of network structure in real systems? Current methods often attempt to remove noisy edges via thresholding^{7–11} by taking into account edge weight, density, group similarity, or network measures⁷. However, thresholding presents a challenge as one may remove too many real edges or keep too many spurious edges¹². Alternatively, many theoretical studies focus on structural properties of noisy edges independent of and isolated from data; most commonly, efforts in this space study random graphs. One can predict properties of random networks^{13–16}, which can be useful in distinguishing real networks from random graphs. However, noisy edges in real networks do not exist in isolation, but are intertwined with real edges. Consequently, neither thresholding nor studying noise in isolation is without flaws when distinguishing the structure of real weak edges from that of noisy weak edges (Fig. 1b, c). Instead of working against noise, can we understand its role in higher-order network structure to mitigate, and potentially even take advantage of, noisy edges?

Here we address the above questions by welcoming noise into our experiments and descriptions of network structure (Fig. 1d, e). Specifically, in this work we ask what (if anything) can noise added to a real network tell us about the underlying real network structure? If the added noise has the same structure regardless of the topology of the real network, then the answer is nothing. The advantage of this scenario is that we could potentially identify an appropriate threshold for our data fairly easily and in a data-driven manner. Instead, if the structure of the added noise varies

based on the topology of the real network, then the noise may carry information about the real weighted network. Indeed, given that a binary graph is defined by a list of either its edges or its non-edges, it is possible that the noise filling the empty edges of a real weighted network holds information related to the topology of the real edges.

To explore this possibility, we ask whether the higher order structure of noise varies across model network topologies in the controlled context of adding noise to model networks. We tested twelve model networks that spanned different node strength distributions, reliance on distances, and more, many of which are commonly used models of real systems. Using the network models tested, we identify at least three common profiles of added noise structure as assessed by persistent homology, that correspond to the topologies of the model networks to which the noise was added. We find that enough information exists within the structure of added noise alone to reasonably distinguish between network models, and that this information stems both from features persisting from the model network and those formed by noisy edges. Furthermore, we provide rules that generate the different patterns of added noise topology. Finally, we remark on the ability of noise to create misleading structure within weak network edges and discuss the implications for data analysis.

Results

Experimental setup. To better understand the effect of noise added to networks, we precisely measure the changes in network structure that arise from noisy, weak edges. We begin with a weighted, completely connected model network generated from a specific set of rules, which has a predictable structure (Fig. 1d, step 1, navy). Next, we create a space for weak, noisy edges by thresholding this model network at a chosen edge density ρ_T to keep only those strongest ρ_T fraction of edges (Fig. 1d, step 2, rose). We then create the added noise network (Fig. 1d, step 3, gold) by assigning a random weight to any edge not included in the ρ_T -thresholded weighted network. We ensure that all edges in the ρ_T -thresholded network have weight >1 , and then we assign weights to the added noise edges from the uniform distribution on $(0, 1)$. Finally, we combine the ρ_T -thresholded network and the added noise network to yield a combined weighted network

Fig. 1 Our experiments query the structure of noise added to model networks. **a** A schematic of real data from a structural brain network. Such data are often assumed to consist of strong and likely real edges (**b**), along with weak edges (**c**) that could arise simply due to noise in the system or in the measurements. Edge density denoted by ρ . **d** In our experiment, we begin with a model network (step 1, navy, network and adjacency matrix), threshold the model network to a specific edge density ρ_T (step 2, rose, network and adjacency matrix), fill the thresholded edges with weaker, noisy edges (step 3, gold, network and adjacency matrix) to form a combined network (step 4, rose and gold, network and adjacency matrix). **e** Then we will measure the structure of the noisy edges (gold).

(rose and gold), which now has a precise cutoff below which all edge weights are randomly chosen. Said another way, if we begin with an empty network and add edges to this graph in order of decreasing edge weights from the combined weighted network, the first $|E|\rho_T$ edges will come from modeled edges while the latter $1 - |E|\rho_T$ edges will come from noisy edges only. Then, if we measure network structure along this expanded version of the combination network (Fig. 1d, step 4), we will be able to distinguish between structure attributed to the model edges from that created by the added noise.

Mathematical framework. Real data is often characterized by three features: weighted relationships between nodes, higher order interactions, and topological constraints such as wiring distance. Given these features, we use persistent homology to study weighted network structure. Persistent homology^{17–19} records the longevity of topological cavities that form and collapse throughout the graph filtration

$$G_0 \subseteq G_1 \cdots \subseteq G_{|E|}.$$

This filtration is the formalization of the expanded view of a weighted network discussed in Fig. 1, in which G_i is the binary graph containing the i strongest edges in the weighted network (see Fig. 2a, see “Methods” section^{20–23}). This persistent homology approach has been previously used to identify differences in cognition across individuals based on resting state functional connectivity²⁴, to find percolation properties of porous materials²⁵, to differentiate neuron morphologies²⁶, and to understand many other real-world systems^{27–29}. We note that in our setup, only the rank order of edges induced by the original edge weights are preserved, so that the specific edge weights or their generating distribution does not affect the outcome. For this work, we compute the persistent homology in several dimensions: dimension 1 (gaps surrounded by edges), 2 (voids surrounded by

filled triangles), 3 (voids surrounded by collections of tetrahedra), and 4 (a higher dimensional analog). A filtration of graphs can be translated into a filtration of higher-order complexes called simplicial complexes on which we can compute persistent homology, by assigning a clique of $k + 1$ nodes to a k -simplex (Fig. 2a, see “Methods” section, Supplementary Methods). Importantly, we could have chosen any quantitative descriptor of graph structure for the following experiments, which makes the experimental setup described in Fig. 1 applicable to many data analysis scenarios.

The persistent homology of a weighted network is a collection of half-open intervals $[b_i, d_i]$ called the barcode that denote the birth b_i and death d_i of the i th persistent cavity in dimension k . We visualize this output as either the barcodes itself or a Betti curve summarization (Fig. 2a, bottom row). In the barcode visualization, each bar corresponds to a persistent cavity and extends from the bar’s birth to its death. In the Betti curve plot, $\beta_k(\rho)$ counts the number of persistent cavities alive at edge density ρ . Unlike many other graph metrics, persistent homology incorporates the strong and weak interactions in a holistic manner³⁰, which both generates a unique perspective on network structure similarity^{20,23} and allows us to more precisely understand the interplay between edges corresponding to real data and edges added randomly.

As motivating examples, we present the persistent homology of two random systems^{31,32} and summarize their persistent homology using Betti curves as shown in Fig. 2. In the first system, we create a noisy network by assigning to every edge a weight sampled uniformly at random from (0, 1). For this, the independent and identically distributed (IID) noise model, we observe the characteristic increase of the Betti curve peaks with increasing dimension^{31,33} (Fig. 2b). Next, we consider the Betti curves of another well-studied model: the random geometric complex³², a common model for embedded and physical systems.

Fig. 2 Through the lens of persistent homology, we imagine three possibilities for the structure of noise added to model networks. **a** From a weighted network (shown as an adjacency (adj.) matrix), we construct a graph filtration in the top row by adding edges to an empty graph one at a time in order of decreasing edge weight. From this filtration in the top row we create a sequence of clique complexes (simplices indicated by blue shaded regions) in the middle row on which we compute the persistent homology. We show the barcode (horizontal lines) and Betti curve $\beta_k(\rho)$ for the resulting persistent homology in the bottom row. **b, c** An example adjacency matrix and Betti curves of the independent and identically distributed (IID) noise (**b**) and random geometric (**c**) network models. Solid lines indicate the average over 500 replicates and shaded areas correspond to ± 1 standard deviation. **d** Stylized possibilities for the structure of added noise as perceived by the Betti curves: a reversion to IID noise Betti curves (left), a condensed collection of all four Betti curves following an IID noise pattern (middle), or a Betti curve pattern completely unlike that of IID noise (right).

In this second model system, we choose points uniformly at random from the unit cube and weight edges as the inverse Euclidean distance between each pair of points (see “Methods” section). Note that the random geometric Betti curves (Fig. 2c) are an order of magnitude smaller than those of the IID noise model, and that the peaks decrease with increasing dimension.

Combining persistent homology with the experimental approach set up in Fig. 1, we imagine three possibilities for the impact of added noise on the Betti curves of model networks, illustrated with stylized Betti curves in Fig. 2d–f. First, after ρ_T the Betti curves could quickly revert to the expected IID noise pattern at that edge density (Fig. 2d). Then we would see that the noise section of the Betti curves looks similar to a copy-and-pasted version of that same section of the IID noise Betti curves. Second, we might imagine that if the weighted network model has one densely connected community, there will be so much empty space that randomly adding edges will create a smaller version of the IID noise network (Fig. 2e). We might expect this scenario to produce the entire sequence (non-zero β_k for $k = 1, \dots, 4$) of increasing Betti curves condensed after ρ_T . Third, perhaps neither of the above occurs, and instead the added noise section of the filtration may show no resemblance to the IID noise Betti curves (Fig. 2f). Which of these three possibilities actually occurs?

Noise structure varies across model networks. We test which of the three scenarios described above occurs for 12 graph models (including IID noise) and 17 values of $\rho_T \in [0.1, 0.9]$. We chose the following graph models to span distinct generative rules, topological characteristics, and for their use in data analysis and systems modeling: IID noise, assortative, core periphery, cosine geometric, disassortative, (discrete) uniform configuration, dot product, geometric configuration, random geometric, ring lattice, root mean squared deviation (RMSD), and squared Euclidean (see “Methods” section, Supplementary Methods, and Supplementary Note 1). Among these model networks, we observe examples of each of the three possible scenarios (Fig. 3a). Specifically, we observe that the structure of the added noise network atop the assortative model with $\rho_T = 0.5$ is qualitatively similar to that of the IID noise model. The dot product model supports added noise that does not follow any expected pattern, and instead has a decreasing pattern of peaks with increasing dimension. Finally, we observe that the added noise atop the random geometric model produces all four dimensions of persistent homology in an increasing pattern of peaks after $\rho_T = 0.5$. We include results for all models in Supplementary Note 2.

Repeating the experiment across all 17 values of ρ_T (Fig. 3b), we observe three qualitative classes of added noise structure (Fig. 3c) within the non-IID noise models. We refer to the three qualitative classes as the random reversion, coned, and random condensed classes. First, the Betti curves of the added noise networks on the assortative, core periphery, and to a lesser extent the disassortative models mirror those of the IID noise model but scaled and sometimes slightly shifted; these models comprise the random reversion class, named for the return of the Betti curves to those of the random IID noise model. Second, noise added to the configuration and the dot product models generate Betti curve peaks that decrease with increasing dimension and dramatically shift rightwards as ρ_T increases. We will refer to this second set of models as the coned class²⁰. Third, the distance-based models (random geometric, cosine geometric, ring lattice, squared Euclidean, and RMSD) constitute the random condensed class. Here, the added noise produces an increasingly compressed collection of IID noise-like Betti curves in all four dimensions as ρ_T increases. In summary, we found that the structure of noise added to model weighted networks varies across network models

and values of ρ_T . Importantly, although we here identify three profiles of noise, we stress that additional profiles are possible and even likely to exist.

Noise added to networks can distinguish many network models. We demonstrated that the structure of noise varies based on the model to which it is added. But does the structure of added noise vary enough that it can accurately classify the network models? In the binary case, there is as much information about a binary graph from knowing all the edges that exist as there is from knowing which edges do not exist. Our experiment could be interpreted as a weighted extension of this idea, in which we know weights of model network edges, and then all the open space (edges with weight 0 after thresholding at ρ_T) are filled by random weights. The randomly-weighted edges and the model-weighted edges together form a complete (weighted) graph. Therefore, despite the fact that edge weights are chosen at random for the added noise, we expect that at ρ_T near 0.5 the added noise persistent homology will be enough to classify the model networks.

To test these ideas, we classify the networks using a Gaussian mixture model. We use three features derived from the persistent homology in dimension k , for $k = 1, \dots, 4$ (12 features total). These three features quantify the total amount of persistent homology (sum of persistent lifetimes), and the lifetime sum weighted by either the birth or death time of persistent cavities^{20,34} (see “Methods” section). We can subset the barcode based on the desired classification (for example, using only the barcode from the added noise portion of the filtration), and we perform the classification procedure for each value of ρ_T ; see confusion matrices for $\rho_T = 0.5$ in Fig. 4 and Supplementary Note 3 for all results including classification accuracy. Note that no persistent features created from the model edges live past $\rho = 0.9$, and thus we can reasonably assume that the results for the $\rho_T = 0.9$ classification will be the same as if we had classified on the original model without any thresholding.

We find that at $\rho_T = 0.5$ the classification performs similarly when using features from the added noise portion of the filtration (Fig. 4a, accuracy ≈ 0.658) over the model network portion (Fig. 4b, accuracy ≈ 0.722). For particular network models such as the RMSD and dot product models, the classification using added noise features outperformed the classification with the model features. Other models such as the squared Euclidean and uniform configuration models were better classified using model weights than the added noise portion of the barcode. Finally, when we classify using the entirety of the barcodes (Fig. 4c), we find that the prediction accuracy is at least as good or improved over both the added noise and model weights section for eight of the twelve models. Indeed, using all barcodes results in the highest accuracy for $0.2 \leq \rho_T \leq 0.6$ (Supplementary Fig. 10). This general accuracy improvement suggests that the added noise and model portions of the barcode contain non-overlapping information.

Surprisingly, at $\rho_T = 0.5$ we also find a comparably good classification when using features from a graph that contains only the noisy edges, with all model edges set to 0. As we have previously discussed, the added noise edges on any weighted network is itself a weighted graph. The above classification experiment used the added noise structure that was dependent—that is—existed atop a weighted network, such that the added noise existed within the context of the model network. That approach leaves open the question of whether this added noise graph as a standalone weighted network (independent of any model edges already added) contains the same amount of information as the added noise network (noise in the context of

Fig. 3 Betti curves of added noise vary based on the underlying network model. a Examples of network models that support varying structure of added noise (IID independent and identically distributed). Solid lines indicate mean Betti curve and shaded region corresponds to ± 1 standard deviation. See Supplementary Fig. 4 for all models. Threshold edge density value ρ_T indicated by dashed lines. **b** Repeating the thresholding process across values of ρ_T , we create a summarized visualization of the Betti curves. **c** Mean Betti curves for each of 11 tested network models (IID noise not included). Curve opacity indicates ρ_T , and color indicates dimension as shown in the legend. For the first model in each column Betti curve peaks at $\rho_T = 0.25$ are marked with a dot to emphasize the scaling and shifting of Betti curves.

the network model)? To address this question, we extracted the added noise network from each graph model for all values of ρ_T and computed their persistent homology. We show the Betti curves across values of ρ_T for these isolated noise networks in Supplementary Fig. 5. We find that classifying the topological features from the added noise independent of model edges (noise isolated) does a fine job (accuracy ≈ 0.719 , see Supplementary Fig. 10) in comparison to the added noise in-context features (Fig. 4d). Indeed, for 10 of the 12 models the classification matches or outperforms that using the added noise in context, and for three models it outperforms the classification using model

weights only. These results indicate that randomly adding edges drawn from a specific organization (here, the complement of the model network at ρ_T) contains an impressive amount of distinguishing power.

Determining the source of information contained within added noise topology. Given the ability of the added noise persistent homology features to reasonably classify the underlying network models, we next aim to clarify more precisely the reason for this result. We can label persistent cavities (bars in the barcode) that exist during the added noise section as one of two types: the first a

Fig. 4 Classification based on the added noise persistent homology can distinguish between network models. Confusion matrices showing the results of the classifications based on **a** added noise barcodes, **b** the barcode from model weights only, **c** the entire barcode, and **d** barcodes generated from the isolated added noise networks. Example barcodes above confusion matrices describe the portion of the barcode used for that classification experiment, and any portion not used. Matrix entries show the number of corresponding true models that were classified as the corresponding predicted model, out of 250 (see “Methods” section). ρ indicates edge density and ρ_T indicates threshold edge density.

noise-exclusive persistent cavity whose birth and death time are both within the added noise portion of the filtration ($b, d > \rho_T$), and a crossover cavity that was born within the model section ($b < \rho_T$) but dies within the added noise section ($d > \rho_T$, see “Methods” section). We expect that, for those graph models that have them, the crossover bars should hold the majority of the classification information since they are formed with model weight edges. After classification on these slices of the barcodes, we find that indeed the crossover bars can be used to classify the five models that consistently produce crossover bars nearly perfectly (Supplementary Fig. 15). The classification using the noise-exclusive portion (Supplementary Fig. 16) is nearly identical to that of using the added noise portion of the barcode (Fig. 4a) for all models. Together our results suggest that using only those persistent features generated within the added noise section of the barcode are sufficient for moderate accuracy in classifying this set of model networks.

Finally, we ask how much information is contained in simply the binary network at ρ_T by comparing the persistent homology from the added noise to that from a randomized model weights experiment. In this experiment, G_0 is empty, the binary graph at ρ_T is the same model network at ρ_T as before, but the ordering of model edges added has been randomized (Supplementary Fig. 6) through randomizing model edge weights. We find that the classification run on the persistent homology features of these randomized model weights (Supplementary Figs. 6 and 10) performs similarly to that of the added noise features when $\rho_T = 0.5$. In addition to the classification results using added noise or isolated noise, these findings strengthen the intuition that the binary network at ρ_T constrains the possible persistent homology outcomes generated by randomly adding edges.

In summary, our classification experiments suggest that the distinguishing information of the added noise structure comes from both crossover and noise exclusive bars. The randomized model network experiment could be viewed as a step-wise random graph process in which exactly one interior graph G_i , $0 < i < |E|$ is fixed. Therefore results from the randomized model networks experiments additionally suggest that even having one graph G_i predetermined between G_0 and $G_{|E|}$ can greatly alter the topology of the filtration between G_0 , G_i and G_i , $G_{|E|}$.

Drivers of noise profiles. Next, we ask how one might obtain each of the three main noise profiles observed in Fig. 3a, c to understand and predict such structure in real data and pursue theoretical inquiries into how such structure forms. The random reversion profile, in which the added noise section ($\rho > \rho_T$) is similar to an IID noise Betti curve copied at that edge density (Fig. 3a), can be replicated using matrix blocks that are themselves created through a random process³⁵. The assortative models shows this trend for the largest range of ρ_T , since it has four blocks of highly weighted edges and the rest are weak but randomly weighted edges (Supplementary Fig. 1). The core-periphery model shows the same pattern but for a smaller range of ρ_T because it has seven blocks of highly weighted edges, and thus its natural break between the high and low weighted edges occurs at a larger edge density than for the assortative model. The disassortative model only has four low-weight blocks, and we observe that at very few values of ρ_T it supports added noise that also produces this similar Betti curve pattern.

For the random condensed and coned profiles, we first investigated if the propensity to fill triangles contributes to the added noise profile. Both the distance-based and coned network models have a strong tendency to form triangles, either by the influence of the triangle inequality in the former or the large clique size in the latter²⁰. We create a random model network weighted so that at each step in the filtration, the next edge added completes an open triangle (three nodes connected by two edges) with probability p and connects a randomly selected pair of non-adjacent nodes with probability $1 - p$ (or if no open triangles exist). If the new edge will complete a triangle, the open triangle is chosen with probability weighted by the product of the open triangle edge weights (see “Methods” section). Hence, it is more likely for the new edge to form a triangle with two edges that were added early in the filtration than late in the filtration. We call this model the weighted probabilistic triangle model (Fig. 5a). In Fig. 5b, c we show the resulting Betti curves across values of ρ_T for $p = 0.7$ and 1. We observe that the parameter p allows us to interpolate between the IID noise network ($p = 0$) and a model that supports an added noise profile of the coned type (decreasing Betti peaks, $0.85 < p \leq 1$, see Supplementary Fig. 7). Taken to the extreme, we create a weighted clique graph in which at each step

Fig. 5 Cliques and triangles support different patterns of added noise Betti curves. **a** The weighted probabilistic triangle model proceeds by adding edges either at random or to complete an open triangle. The likelihood of completing a specific open triangle is based on the weight of its edges. **b** Betti curves at varying values of threshold density ρ_T for the weighted probability triangle with probability (w.p) $p = 0.7$ and $p = 1$, with dots marking curves corresponding to $\rho_T = 0.25$. Color of the curves correspond to colors in the legend above the panels. **c** Average (solid lines) and $\pm$ standard deviation (filled area) for Betti curves of the weighted probabilistic triangle model with probability $p = 1$, $\rho_T = 0.5$ (right). **d** Betti curves at varying values of ρ_T for the clique graph. **e** The random m -clique model creates a weighted network by selecting m nodes at random, increasing all edge weights between these nodes by 1, and repeating the process. **f** Betti curves across all values of ρ_T and **g** for $\rho_T = 0.35$ for the m -clique model with $m = 25$. Legends the same as for panels **b**, **c**, respectively.

in the filtration, the newest edge contributes to building one growing clique. This weighted clique model shows an added noise profile similar to the weighted probabilistic triangle model with $p > 0.85$ (Fig. 5d). Indeed, discussed in Supplementary Note 4, upper and lower bounds on the Betti numbers for this weighted clique network can be derived in a similar fashion to those from the IID noise graph.

What additional process or constraint underlies the distance-based network models that is not captured by the above weighted probabilistic triangle model? Although distance-based networks do fill triangles quickly, the alternative to completing a triangle in an embedded network is far from adding a random edge anywhere in the network, as is the case in the above weighted probabilistic triangle model. Instead, embedded networks often form multiple pockets of clustered nodes that eventually connect. We aimed to capture this process at a basic, non-embedded level, by creating a model that constructs a weighted network by adding pockets of densely connected nodes to the network; we call the model the m -clique model (Fig. 5e). In this model, we choose a random set of m nodes, increase all edge weights between these m nodes by 1, and repeat this process until a desired network density is reached (see “Methods” section). We record the persistent homology of these models and their added noise for varying values of ρ_T and m in Supplementary Fig. 8, and we show the $m = 25$ case in Fig. 5f, g. We find that for parameter values near $m = 25$, the added noise Betti peaks show an increasing pattern similar to that seen with the distance-based models. We note that the random m -clique model does not fully capture the extent to which the Betti curve peaks shift rightwards with increasing ρ_T in the distance-based models, suggesting that adding random m -cliques alone is not sufficient to completely recreate the observed phenomenon.

In sum, through the above generative graph models we have determined processes by which we can drive the structure of added noise towards any of the three observed profiles.

Misinterpretations caused by added noise. We close with a more realistic example of added noise on networks, and highlight situations in which added noise may erroneously suggest non-existent network features. Above we considered the situation in which there exists a sharp, binary distinction between the model network and added noise sections of the filtration. The resulting Betti curves often show an obvious point at which the trends change drastically. Though nicer to study numerically, this situation is unlikely in real data. It is more likely that as we move along the filtration, the proportion of noisy edges increases until at some point the last real edge has been added and all the later (weaker) edges are noise.

We examine this overlapping noise scenario by creating network models as before, but instead of switching from only model edges to only noise edges at ρ_T (Fig. 6a, left), we now set an increasing noise interval $[\rho_a, \rho_b]$ (Fig. 6a). For edges added at densities $\rho < \rho_a$, model network edges are added in the usual ordering. If $\rho_a \leq \rho \leq \rho_b$, then with probability $p_p = \frac{1}{\rho_b - \rho_a}(\rho - \rho_a)$ we choose the next edge in the filtration at random and with probability $1 - p_p$ we choose the next edge based on the ordered model edges. For $\rho > \rho_b$, all further edges are chosen at random.

We show all Betti curves generated by this process in Supplementary Notes 2, 3, and highlight a few interesting results in Fig. 6b, c. First, because we have an expectation for edge density intervals with non-zero persistent homology, persistent cavities that are born exceptionally late can be considered significant³⁶. Following this concept, both the quantitative and

Fig. 6 Noise overlapping real networks can complicate interpretations of network structure. **a** Previous experiments used a sharp edge density threshold ρ_T to separate model from noise edges when filtering on edge density ρ , but a more realistic scenario is to have an increasing likelihood of noisy edges over some interval $[\rho_a, \rho_b]$. Betti curves from the uniform configuration model with noise interval $[0.3, 0.5]$ and the geometric configuration model with noise interval $[0.1, 0.3]$. **c** Betti curves for the assortative model with no noise added, squared Euclidean model with noise interval $[0.2, 0.4]$, root mean squared deviation (RMSD) model with noise interval $[0.2, 0.4]$, and random geometric model with noise interval $[0.3, 0.5]$. See panel **b** for legend.

qualitative evaluations of, for example the uniform configuration persistent homology output shown in Fig. 6b, could suggest interesting or significant features exist after $\rho = 0.2$. Similar inferences might be drawn from the Betti curves of the geometric configuration model (Fig. 6b). However, we would be wrong to assign significance to the late born features in these plots. All of these late-born persistent cavities are generated by added noise.

Second, consider the Betti curve plots in Fig. 6c. Given the double peak of β_1 and β_2 ²⁰, one could reasonably conclude that all four networks have a considerable modular structure. However, again we would be wrong. Three of the four are distance-based models while only the assortative model (run without noise) actually contains distinct modules (Fig. 6c). The second peaks seen in the Betti curves generated by the distance-based models are solely produced by added noise, suggesting that qualitative interpretations of Betti curves arising from noisy data may be misleading.

Together, our results reveal the importance of a careful eye when interpreting the structure of weak edges in complex systems. Particularly, noise in the weak edges can falsely appear as significant structural features.

Discussion

In this work we investigated the structure of random edges added to pre-existing model network edges. We determined that the existing model structure dictates the topology of its added noise, and consequently that the structure of the added noise alone carries distinguishing information about the network model. We then identified generative processes for creating the three main patterns of added noise and finally highlighted consequences of variable noise structure for the analysis of weighted graphs.

The results and concepts within this paper can be used to learn about myriad real systems. Firstly, the embedded graph models chosen for analyses have been used in conjunction with persistent homology to understand physical systems such as the brain³⁶, filamentous networks³⁷, granular networks^{38,39}, and protein interaction networks⁴⁰. Similarly, the non-embedded models chosen are common in neuroscience^{41,42} and in biology⁴³. Secondly, the large spread of ρ_T values allowed us to study models

that mirrored sparse networks (low ρ_T) alongside the question of how added noise impacts dense networks (high ρ_T). Thirdly, we learn from the classification results on the added noise network that the weak edges within networks can offer unexpected perspectives on the strong edges. Specifically in this case, studying the persistent homology of the weak edges in the coned class revealed incredible structure whereas persistent homology of the strong edges showed no persistent homology. This result reinforces the point that any descriptor offers one particular lens through which we see the objects, and any lens has areas that are in and out of focus. As we see above, studying the weak edges can add more information to any particular perspective.

Importantly, though we use persistent homology as our graph descriptor and edge weight as the filtration parameter, we stress that any number of graph descriptors and filtration parameters could have been used in their place, making the experimental process relevant to a wider range of data analysis techniques than topological data analysis (TDA) alone. For example, if one is interested in the efficiency of networks, one could repeat the experiments to understand how added noise impacts network structure through the lens of network efficiency. Alternatively, one could use modularity in order to distinguish real community structure within weak edges from that created from noisy edges. If one would prefer to stay within TDA, one could expand the experiments by using a descriptor that adheres to a stability requirement^{44–48}. Though in this work we create a filtration from edge weights, if one has an ordering of the edges, perhaps via significance thresholding^{9,49}, again the entire process (Fig. 1) would still apply. We hope that the flexibility of the experimental processes used in this paper will aid researchers from a variety of fields in understanding how noise affects real data.

The increasing use of weighted networks in applied sciences from molecular biology⁵⁰ to transportation⁵¹ suggests that weak edges and topological variability⁵² will continue to be studied in the future. Another major consequence of our results is that given the variable structure of added noise atop networks, care must be taken when analyzing networks with possible noise contamination. Though here we used persistent homology to query network structure, we expect that added noise structure as viewed by many graph metrics will vary based on real network topology, as seen

previously in the binary case^{12,53}. Consequently, we suggest that one considers the structure of noise dependent on their system as perceived by their structural measure of choice before assigning value to features formed by weak edges. On the other hand, our experiments suggest an avenue for improving the detection of a threshold (or range of densities) that separate data edges from noise edges^{54–56}. If one knows the expected structure of added noise on their system, then one could use this information to determine at what threshold in a new dataset the noise begins.

Though motivated by problems in data analysis, our work also lays the foundation for additional interesting theoretical questions. First, one could interpret the experiments performed in this paper as querying the change of network structure caused by combining or shifting between two network models⁵⁷. Here, we combined one network model with IID noise, but one could repeat these experiments with any pair of graph models. For example, how does the structure of a ring lattice combined with a modular network compare to that of a ring lattice combined with a random geometric model? Such questions could be helpful for understanding systems such as the brain that naturally switch between states^{58,59}.

Second, an alternative way to interpret the patterns of added noise structure is that the model edges force or restrict the noise into a particular shape based on the topology of the empty space left by the arrangement of model edges. Following this line of thought, the random reversion class could be seen as having a deferential structure, in that the added noise was quickly able to revert to its natural architecture, and the other two model classes could be interpreted as having a forceful structure that dominates the ability of the added noise to revert. Indeed, we observed that even with only 15% of edges added, the geometric configuration model has influenced the structure of the 85% of randomly added edges, so that its added noise does not follow the IID noise pattern. We leave the question of how exactly the topology of the model edges dictate (or do not dictate) the topology of the added noise for future work. Additionally, one might ask if real-world sparse networks^{60,61} have a dominant structure that protects their architecture against random fluctuations.

Third, given a sparse network, one can use randomly added edges and the crossover bar concept to help determine geometrical properties of the network's topological cavities⁶². Specifically, since edges arrive at random, generating a distribution of death times for each topological cavity would suggest a cavity geometry that is more or less susceptible to randomly dying. One would expect cavities with large minimal generators to be unlikely to die via random edge addition, whereas cavities with multiple small minimal generators (for example a narrow tube) would likely die quickly from randomly adding edges.

Finally, our experiments suggest interesting directions for future work in cognitive science and neuroscience. Studying the added noise is equivalent to studying the empty space left by a network, making the above analyses particularly interesting for systems in which the sparsity of edges is a feature. For example, in one's brain network the structure of both the present edges and the empty space changes over development⁶³. Specifically, network edges are pruned as a person ages and learns⁶⁴, which in turn alters the structure of the non-edges in their network⁶⁵. Is the structure of the empty space in brain networks more or less important than the structure of those edges that exist? Finally, in cognitive science one could interpret learning as beginning with a noisy knowledge network in which some edges incorrectly connect disconnected concepts. Learning would then proceed by a rewiring that network to a final, correct knowledge network^{66,67}. Studying this phenomenon is effectively the reverse

of the experiments presented here, in which one begins with a noisy network and ends with an expected model network.

Conclusion

In conclusion, our work shows that the persistent homology of noise added to networks varies based on the real network topology. Additionally, we find generative network rules that produce networks supporting differing structures of noise. Finally, our results offer a reason to examine how the structure of added noise to a real network may influence the structural measure in question, in order to make real features present in weak edges clearly distinguishable from features created by noise alone.

Methods

Computations were performed in Julia, with the exception of the classification experiments which were performed in MATLAB. We use the Eirene software⁶⁸ for all persistent homology computations.

Weighted network models. We chose weighted network models that show a variety of real-world properties, including the structural feature of modularity, and the physical feature of weights that decreases with distance. For all models the final graph contained $N = 70$ nodes, and 500 replicates were created. We chose these values for nodes and replicates to balance topological richness, reliable estimates of variance, and computation time. We note that we do not expect a change in node number to severely affect results, based on previous persistent homology research on network models^{20,21,23,32,69}. If a model yielded non-unique graph weights, random noise was added such that all edge weights would be unique and that the ranking of edges with unique edge weights would remain the same. See the section of the Supplementary Information entitled "Model network generation details" and Supplementary Figs. 1, 2, and 3 for more details. We separate the descriptions of the models into two sections: distance-agnostic graph models and distance-based graph models.

Distance-agnostic graph models. The following models are created without any notion of a formal distance between nodes.

- **Assortative.** The assortative model was constructed following an implementation of the weighted stochastic block model (WSBM)³⁵ in which four high-weight blocks were positioned along the diagonal.
- **Core periphery.** This model was also constructed with the WSBM³⁵ approach but high-weight blocks were positioned along the top and left edge of the adjacency matrix to form a core and periphery. One fourth of nodes formed the core, and the rest formed the periphery.
- **Disassortative.** The inverse of the assortative model, in which nodes within a community connect strongly to nodes outside of their community.
- **Discrete uniform configuration.** A weighted configuration model⁷⁰ with the node strengths drawn from the discrete uniform distribution.
- **Dot product.** Here we chose N points at random in $\mathbb{R}^{\text{dim}}$ (here $\text{dim} = 3$). We then weighted edges between two nodes as the dot product of the associated vectors.
- **Geometric configuration.** A weighted configuration model⁷⁰ with node strengths drawn from a geometric distribution.
- **IID noise.** All edge weights were chosen at random from the uniform distribution on $(0, 1)$.

Distance-based graph models. The following models are created by first choosing N points in $\mathbb{R}^{\text{dim}}$, then calculating a distance $d(\vec{u}, \vec{v})$ between every pair of points using the definitions below, and finally taking the reciprocal of that distance as the edge weight between two nodes. By this process, two nodes that are close together, as determined by the distance metric, will parent an edge with a large weight, whereas nodes that are far apart will parent edges with small weights. For simplicity, we chose $\text{dim} = 3$ for all models.

- **Cosine geometric.** Given nodes v, u with associated vectors $\vec{v}, \vec{u} \in \mathbb{R}^{\text{dim}}$, respectively, the cosine distance is

$$d(\vec{v}, \vec{u}) = 1 - \frac{\vec{v} \cdot \vec{u}}{\|\vec{v}\| \|\vec{u}\|}.$$

- **Random geometric (Euclidean distance).** Given nodes v, u with associated vectors $\vec{v}, \vec{u} \in \mathbb{R}^{\text{dim}}$, respectively,

$$d(\vec{v}, \vec{u}) = \sqrt{\sum_{i=1}^{\text{dim}} (\vec{v}_i - \vec{u}_i)^2}.$$

- **Ring lattice.** We labeled N vertices as $1, \dots, N$, and connected nodes in one large ring. We assigned the edge weight between nodes i and j as the inverse hop distance along this ring, and assumed that hopping is allowed only between neighboring nodes⁷¹.

- Root mean squared deviation (RMSD). Given nodes v, u with associated vectors $\vec{v}, \vec{u} \in \mathbb{R}^{\text{dim}}$, respectively,

$$d(\vec{v}, \vec{u}) = \sqrt{\frac{1}{\text{dim}} \sum_{i=1}^{\text{dim}} (\vec{v}_i - \vec{u}_i)^2}.$$

- Squared Euclidean distance. Given nodes v, u with associated vectors $\vec{v}, \vec{u} \in \mathbb{R}^{\text{dim}}$, respectively,

$$d(\vec{v}, \vec{u}) = \sum_{i=1}^{\text{dim}} (\vec{v}_i - \vec{u}_i)^2.$$

Models to reproduce noise profiles. We created three models with the intention of finding simple rules that would give rise to a particular added noise persistent homology pattern. First, the weighted probabilistic triangle graph takes one parameter p that controls triangle formation. The goal of this model is to form a weighted network such that when we expand to the filtration, at each step in the filtration the new edge has probability p of forming a triangle. Beginning with N nodes and 0 edges, with probability p we either add an edge that will create a new triangle, or we add an edge at random. If we are to add a new triangle, we check to see if there are any open triangles in the graph—that is where two edges connect three nodes—and if there are no open triangles, we add an edge at random. If there are open triangles, we pick one open triangle to fill with probability proportional to the product of the two edge weights of the open triangle edges. The new edge is assigned a weight lower than any previously added edge.

The second and third models are complementary to the first. In the second model, the random m -clique networks take one parameter m that controls the size of cliques to be added. Beginning with an empty network we randomly choose m nodes and add a value of 1 to each edge weight connecting the m nodes. This process repeats until no more than 12% of edges were empty. In the third model, we create a weighted clique model in which throughout the filtration, each new edge contributes to forming one growing clique. For example, once three edges have been added the graph is a 3-clique, the first six edges will form a 4-clique, the first ten edges added will form a 5-clique, and so on.

Persistent homology. Persistent homology^{19,72} measures the birth and death of persistent cavities that arise and evolve throughout a sequence of simplicial complexes in which simplices may be added at each step (a filtered simplicial complex). Here, we form this sequence from a weighted network by creating a graph filtration

$$G_0 \subseteq G_1 \subseteq \dots \subseteq G_{|E|}, \quad (1)$$

where G_i is the binary graph containing edges with the i highest weights in the weighted network, and then taking the clique complex of each G_i ^{21–23,31}. We compute the persistent homology using the Eirene⁶⁸ package in Julia. See the Supplementary Methods in the Appendix for more details.

Derived values. Following ref. ³⁴, we use the three barcode summaries listed below. Intuitively, each returns a description of the amount of persistent homology in each dimension, but they vary by their weighting of each bar $[b_i, d_i]$ of the barcode.

- Betti bar, $\bar{\beta}_k$. Let M be the total number of persistent cavities in dimension k . Then

$$\bar{\beta}_k = \sum_{i=1}^M (d_i - b_i).$$

The $\bar{\beta}_k$ value sums the lifetimes of all bars in dimension k .

- Mu bar, $\bar{\mu}_k$. Let M be the total number of persistent cavities in dimension k . Then

$$\bar{\mu}_k = \sum_{i=1}^M b_i(d_i - b_i).$$

The $\bar{\mu}_k$ value scales each bar's lifetime by the birth time and then sums these weighted lifetimes.

- Nu bar, $\bar{\nu}_k$. Let M be the total number of persistent cavities in dimension k and L the number of edges in the complete graph. Then

$$\bar{\nu}_k = \sum_{i=1}^M (L - d_i)(d_i - b_i).$$

The $\bar{\nu}_k$ scales each bar's lifetime based on the death time of that bar, and then sums the scaled lifetimes.

Classification. We seek a simple and generative method to classify our networks to ensure flexibility in incorporating new network models or data that may be generated from a different underlying distribution. As such, we model the distribution of features for each network model using a multivariate Gaussian model, and collect these models into a Gaussian mixture model⁷³. For prediction, we use features from a held-out test set of network features, and assign a predicted label

based on the class that generates the highest posterior probability. These predicted labels are then used to generate the confusion matrices of the main text.

Generative Gaussian mixture model. For each network instantiation i belonging to network model j , we collect a 12-dimensional vector $\mathbf{x}_i^j$ of features $\bar{\beta}_k$, $\bar{\mu}_k$, and $\bar{\nu}_k$ for $k = 1, 2, 3, 4$. Then, we compute the mean μ_j and covariance Σ_j of the features for the $N_{\text{train}} = 250$ networks belonging to model j , to yield the Gaussian probability distribution of network model j as

$$G_j(\mathbf{x}|\mu_j, \Sigma_j) = \frac{1}{\sqrt{2\pi|\Sigma_j|}} e^{-\frac{1}{2}(\mathbf{x}-\mu_j)^T \Sigma_j^{-1}(\mathbf{x}-\mu_j)}.$$

Because some features are all 0 at some thresholds, we bias the covariance matrix by adding 0.01I to ensure Σ_j is positive definite. Next, we collect these models together into the following equally weighted Gaussian mixture:

$$p(\mathbf{x}) = \sum_{j=1}^J \frac{1}{J} G_j(\mathbf{x}|\mu_j, \Sigma_j),$$

where $J = 12$ is the total number of network models we used. Finally, for a test set of $N_{\text{test}} = 250$ for each model class j (for a total of $JN_{\text{test}} = 3000$ network instantiations), we computed the posterior probability for each of the mixture components, and assigned a label to each network corresponding to the class of its maximum posterior probability. While the confusion matrices represent one selection of training and testing sets, we measure the distribution of performance by taking the total classification accuracy for each feature set across 100 random and disjoint sets of training and testing sets (see Supplementary Fig. 10).

Features. Below we detail the inputs to each of the classification experiments performed in this work. For each, we take a specific subset of barcodes and use them to compute $\bar{\beta}_k$, $\bar{\mu}_k$, and $\bar{\nu}_k$. Recall that the barcode in dimension k is a set of pairs (b_i, d_i) corresponding to the birth and death time of persistent cavity i , respectively.

- Added noise (in context). We computed persistent homology on the entire graph sequence from $\rho = 0$ to $\rho = 1$, where at $\rho = \rho_T$ edge additions became random. To understand the information contained in the added noise portion of the barcode, we replaced any $b_i, d_i < \rho_T$ with $\frac{\rho_T|E|+1}{|E|}$ (the density at which the first noisy edge is added). Any cavity with $b_i = d_i$ will contribute 0 to each of the three summary statistics. Any bar with $b_i < \rho_T$ and $d_i > \rho_T$ will effectively be shortened so that the persistent cavity is born at $\frac{\rho_T|E|+1}{|E|}$.
- Model weights before ρ_T . From the barcodes computed using the full filtration, we replaced any $b_i, d_i > \rho_T$ with ρ_T . Any bar with $b_i < \rho_T$ and $d_i > \rho_T$ will effectively be shortened so that it dies at ρ_T .
- All barcodes. Here we kept the barcodes as they were originally calculated.
- Added noise (isolated). Persistent homology was calculated on the added noise graph alone so we used these barcodes for the classification.
- Crossover bars. We filtered the added noise barcode to only include bars with original birth $b_i < \rho_T$ and original death $d_i > \rho_T$. Because we begin with the added noise portion of the barcode, the smallest b_i possible is $\frac{\rho_T|E|+1}{|E|}$.
- Noise exclusive bars. We filtered the barcode to only include bars with $b_i > \rho_T$ and $d_i > \rho_T$.
- Randomized model edge weights. First, we randomized the ordering of the model edges by randomizing their weights. We then computed the persistent homology on this randomized model, and finally we replaced every $b_i, d_i > \rho_T$ with ρ_T , so that any bar with $b_i < \rho_T$ and $d_i > \rho_T$ was effectively shortened so that it would die at ρ_T .
- Overlapping noise. Here we kept all barcodes as they were originally calculated in those weighted networks generated with overlapping noise (see the section entitled “Misinterpretations caused by added noise”).

Citation diversity statement. Recent work in several fields of science has identified a bias in citation practices such that papers from women and other minority scholars are under-cited relative to the number of such papers in the field^{74–78}. Here, we sought to proactively consider choosing references that reflect the diversity of the field in thought, form of contribution, gender, race, ethnicity, and other factors. First, we obtained the predicted gender of the first and last author of each reference by using databases that store the probability of a first name being carried by a woman^{78,79}. By this measure (and excluding self-citations to the first and last authors of our current paper), our references contain 13.1% woman(first)/woman(last), 9.2% man/woman, 23.7% woman/man, and 53.9% man/man. This method is limited in that a) names, pronouns, and social media profiles used to construct the databases may not, in every case, be indicative of gender identity and b) it cannot account for intersex, non-binary, or transgender people.

Data availability

All data can be generated using the open code hosted at https://github.com/asizemore/Noise_and_TDA.

Code availability

All code to generate data and perform analyses can be found at https://github.com/asizemore/Noise_and_TDA. Interactive visualizations are hosted at https://asizemore.github.io/noise_and_tda_supplement/.

Received: 8 January 2021; Accepted: 6 September 2021;

Published online: 02 November 2021

References

1. Santarnecchi, E., Galli, G., Polizzotto, N. R., Rossi, A. & Rossi, S. Efficiency of weak brain connections support general cognitive functioning. *Hum. Brain Mapp.* **35**, 4566–4582 (2014).
2. Goulas, A., Schaefer, A. & Margulies, D. S. The strength of weak connections in the macaque cortico-cortical network. *Brain Struct. Funct.* **220**, 2939–2951 (2015).
3. Bassett, D. S., Nelson, B. G., Mueller, B. A., Camchong, J. & Lim, K. O. Altered resting state complexity in schizophrenia. *NeuroImage* **59**, 2196–2207 (2012).
4. Granovetter, M. S. The strength of weak ties. *Am. J. Sociol.* **78**, 1360–1380 (1973).
5. Friedkin, N. A test of structural features of granovetter's strength of weak ties theory. *Soc. Netw.* **2**, 411–422 (1980).
6. Ma, X. & Gao, L. Discovering protein complexes in protein interaction networks via exploring the weak ties effect. *BMC Syst. Biol.* **6**, S6 (2012).
7. Bordier, C., Nicolini, C. & Bifone, A. Graph analysis and modularity of brain functional connectivity networks: searching for the optimal threshold. *Front. Neurosci.* **11**, 441 (2017).
8. de Reus, M. A. & van den Heuvel, M. P. Estimating false positives and negatives in brain networks. *Neuroimage* **70**, 402–409 (2013).
9. Serrano, M. Á., Boguná, M. & Vespignani, A. Extracting the multiscale backbone of complex weighted networks. *Proc. Natl Acad. Sci.* **106**, 6483–6488 (2009).
10. Alexander-Bloch, A. F. et al. Disrupted modularity and local connectivity of brain functional networks in childhood-onset schizophrenia. *Front. Syst. Neurosci.* **4**, 147 (2010).
11. van Wijk, B. C. M., Stam, C. J. & Daffertshofer, A. Comparing brain networks of different size and connectivity density using graph theory. *PLoS ONE* **5**, e13701 (2010).
12. Zalesky, A. et al. Connectome sensitivity or specificity: which is more important? *NeuroImage* **142**, 407–420 (2016).
13. Wigner, E. P. On the distribution of the roots of certain symmetric matrices. *Ann. Math.* **67**, 325–327 (1958).
14. Ramezanpour, A., Karimipour, V. & Mashaghi, A. Generating correlated networks from uncorrelated ones. *Phys. Rev. E* **67**, 046107 (2003).
15. Erdős, P. & Rényi, A. On random graphs, i. *Publ. Math. Debr.* **6**, 290–297 (1959).
16. Chung, F. & Peng, X. Decomposition of random graphs into complete bipartite graphs. *SIAM J. Discret. Math.* **30**, 296–310 (2016).
17. Carlsson, G. Topology and data. *Bull. Am. Math. Soc.* **46**, 255–308 (2009).
18. Zomorodian, A. & Carlsson, G. Computing persistent homology. *Discret. Comput. Geom.* **33**, 249–274 (2005).
19. Otter, N., Porter, M. A., Tillmann, U., Grindrod, P. & Harrington, H. A. A roadmap for the computation of persistent homology. *EPJ Data Sci.* **6**, 17 (2017).
20. Sizemore, A., Giusti, C. & Bassett, D. S. Classification of weighted networks through mesoscale homological features. *J. Complex Netw.* **5**, 245–273 (2016).
21. Giusti, C., Pastalkova, E., Curto, C. & Itskov, V. Clique topology reveals intrinsic geometric structure in neural correlations. *Proc. Natl Acad. Sci.* **112**, 13455–13460 (2015).
22. Petri, G., Scolamiero, M., Donato, I. & Vaccarino, F. Topological strata of weighted complex networks. *PLoS ONE* **8**, e66506 (2013).
23. Horak, D., Maletić, S. & Rajković, M. Persistent homology of complex networks. *J. Stat. Mech.* **2009**, P03034 (2009).
24. Anderson, K. L., Anderson, J. S., Palande, S. & Wang, B. Topological data analysis of functional mri connectivity in time and space domains. In *International Workshop on Connectomics in Neuroimaging* 67–77 (Springer, 2018).
25. Robins, V., Saadatfar, M., Delgado-Friedrichs, O. & Sheppard, A. P. Percolating length scales from topological persistence analysis of micro-ct images of porous materials. *Water Resour. Res.* **52**, 315–329 (2016).
26. Kanari, L., Garin, A. & Hess, K. From trees to barcodes and back again: theoretical and statistical perspectives. <https://arxiv.org/2010.11620> (2020).
27. Curto, C. What can topology tell us about the neural code? *Bull. Am. Math. Soc.* **54**, 63–78 (2017).
28. Hess, K. *Topological Data Analysis* 277–305 (Springer, 2020).
29. Patania, A., Petri, G. & Vaccarino, F. The shape of collaborations. *EPJ Data Sci.* **6**, 18 (2017).
30. Sizemore, A. E., Phillips-Cremins, J. E., Ghrist, R. & Bassett, D. S. The importance of the whole: topological data analysis for the network neuroscientist. *Netw. Neurosci.* **3**, 656–673 (2019).
31. Kahle, M. Topology of random clique complexes. *Discret. Math.* **309**, 1658–1671 (2009).
32. Kahle, M. Random geometric complexes. *Discret. Comput. Geom.* **45**, 553–573 (2011).
33. Kahle, M. Sharp vanishing thresholds for cohomology of random flag simplicial complexes. *Ann. Math.* **179**, 1085–1107 (2012).
34. Adcock, A., Carlsson, E. & Carlsson, G. The ring of algebraic functions on persistence bar codes. Preprint <http://comptop.stanford.edu/u/preprints/multitwo> (2012).
35. Aicher, C., Jacobs, A. Z. & Clauset, A. Adapting the stochastic block model to edge-weighted networks. <https://arxiv.org/1305.5782> (2013).
36. Sizemore, A. E. et al. Cliques and cavities in the human connectome. *J. Computational Neurosci.* **44**, 115–145 (2018).
37. Ciocanel, M.-V., Juenemann, R., Dawes, A. T. & McKinley, S. A. Topological data analysis approaches to uncovering the timing of ring structure onset in filamentous networks. <https://arxiv.org/1910.05850> (2019).
38. Papadopoulos, L., Porter, M. A., Daniels, K. E. & Bassett, D. S. Network analysis of particles and grains. *J. Complex Netw.* **6**, 485–565 (2018).
39. Lim, M. X. & Behringer, R. P. Topology of force networks in granular media under impact. *EPL* **120**, 44003 (2018).
40. Higham, D. J., Rašajski, M. & Pržulj, N. Fitting a geometric graph to a protein–protein interaction network. *Bioinformatics* **24**, 1093–1099 (2008).
41. Faskowitz, J., Yan, X., Zuo, X.-N. & Sporns, O. Weighted stochastic block models of the human connectome across the life span. *Sci. Rep.* **8**, 1–16 (2018).
42. Faskowitz, J. & Sporns, O. Mapping the community structure of the rat cerebral cortex with weighted stochastic block modeling. *Brain Struct. Funct.* **225**, 71–84 (2020).
43. Grindrod, P. & Kibble, M. Review of uses of network and graph theory concepts within proteomics. *Expert Rev. Proteom.* **1**, 229–238 (2004).
44. Cohen-Steiner, D., Edelsbrunner, H. & Harer, J. Stability of persistence diagrams. *Discret. Comput. Geom.* **37**, 103–120 (2007).
45. Adams, H. et al. Persistence images: a stable vector representation of persistent homology. *J. Mach. Learn. Res.* **18**, 218–252 (2017).
46. Bubenik, P. Statistical topological data analysis using persistence landscapes. *J. Mach. Learn. Res.* **16**, 77–102 (2015).
47. Chung, Y.-M. & Lawson, A. Persistence curves: a canonical framework for summarizing persistence diagrams. <https://arxiv.org/1904.07768> (2019).
48. Atienza, N., González-Díaz, R. & Soriano-Trigueros, M. On the stability of persistent entropy and new summary functions for tda. <https://arxiv.org/1803.08304> (2018).
49. Kojaku, S. & Masuda, N. Constructing networks by filtering correlation matrices: a null model approach. *Proc. R. Soc. A* **475**, 20190578 (2019).
50. Fuller, T., Langfelder, P., Presson, A. & Horvath, S. *Handbook of Statistical Bioinformatics*, 369–388 (Springer, 2011).
51. Xing, Y., Lu, J. & Chen, S. Weighted complex network analysis of shanghai rail transit system. *Discrete Dyn. Nat. Soc.* <https://doi.org/10.1155/2016/1290138> (2016).
52. Yan, L., Wang, Y., Munch, E., Gasparovic, E. & Wang, B. A structural average of labeled merge trees for uncertainty visualization. *IEEE Trans. Vis. Comput. Graph.* **26**, 832–842 (2019).
53. van den Heuvel, M. P. et al. Proportional thresholding in resting-state fmri functional connectivity networks and consequences for patient-control connectome studies: Issues and recommendations. *Neuroimage* **152**, 437–449 (2017).
54. Wang, B. et al. Network enhancement as a general method to denoise weighted biological networks. *Nat. Commun.* **9**, 1–8 (2018).
55. Zeng, A. & Cimini, G. Removing spurious interactions in complex networks. *Phys. Rev. E* **85**, 036101 (2012).
56. Zhou, F., Mahler, S. & Toivonen, H. *Bisociative Knowledge Discovery*, 179–198 (Springer, 2012).
57. You, J., Ying, R., Ren, X., Hamilton, W. L. & Leskovec, J. Graphrnn: generating realistic graphs with deep auto-regressive models. <https://arxiv.org/1802.08773> (2018).
58. Bansal, K. et al. Cognitive chimera states in human brain networks. *Sci. Adv.* **5**, eaau8535 (2019).
59. Cornblath, E. J. et al. Temporal sequences of brain activity at rest are constrained by white matter structure and modulated by cognitive demands. *Commun. Biol.* **3**, 1–12 (2020).
60. Bassett, D. S., Brown, J. A., Deshpande, V., Carlson, J. M. & Grafton, S. T. Conserved and variable architecture of human white matter connectivity. *NeuroImage* **54**, 1262–1279 (2011).
61. Naik, C., Caron, F. & Rousseau, J. Sparse networks with core-periphery structure. <https://arxiv.org/1910.09679> (2019).

62. Obayashi, I. Volume-optimal cycle: tightest representative cycle of a generator in persistent homology. *SIAM J. Appl. Algebra Geom.* **2**, 508–534 (2018).
63. Dennis, E. L. et al. Development of brain structural connectivity between ages 12 and 30: a 4-tesla diffusion imaging study in 439 adolescents and adults. *Neuroimage* **64**, 671–684 (2013).
64. Stephan, A. H., Barres, B. A. & Stevens, B. The complement system: an unexpected role in synaptic pruning during development and disease. *Annu. Rev. Neurosci.* **35**, 369–389 (2012).
65. Morgan, S. E., White, S. R., Bullmore, E. T. & Vértes, P. E. A network neuroscience approach to typical and atypical brain development. *Biol. Psychiatry* **3**, 754–766 (2018).
66. Lynn, C. W., Kahn, A. E., Nyema, N. & Bassett, D. S. Abstract representations of events arise from mental errors in learning and memory. *Nat. Commun.* **11**, 1–12 (2020).
67. Melo, A. & Paulheim, H. Detection of relation assertion errors in knowledge graphs. In *Proceedings of the Knowledge Capture Conference* 1–8 (Association for Computing Machinery, ACM Digital Library, 2017).
68. Henselman, G. & Ghrist, R. Matroid filtrations and computational persistent homology. <https://arxiv.org/1606.00199> (2016).
69. Petri, G., Sciamiero, M., Donato, I. & Vaccarino, F. Networks and cycles: a persistent homology approach to complex networks. In *Proceedings of the European Conference on Complex Systems 2012*, 93–99 (Springer, 2013).
70. Newman, M. E. The structure and function of complex networks. *SIAM Rev.* **45**, 167–256 (2003).
71. Watts, D. J. & Strogatz, S. H. Collective dynamics of ‘small-world’ networks. *Nature* **393**, 440–442 (1998).
72. Edelsbrunner, H. & Harer, J. Persistent homology—a survey. *Contemp. Math.* **453**, 257–282 (2008).
73. Reynolds, D. A. Gaussian mixture models. *Encycl. Biom.* **741**, 659–663 (2009).
74. Mitchell, S. M., Lange, S. & Brus, H. Gendered citation patterns in international relations journals. *Int. Stud. Perspect.* **14**, 485–492 (2013).
75. Maliniak, D., Powers, R. & Walter, B. F. The gender citation gap in international relations. *Int. Organ.* **67**, 889–922 (2013).
76. Caplar, N., Tacchella, S. & Birrer, S. Quantitative evaluation of gender bias in astronomical publications from citation counts. *Nat. Astron.* **1**, 0141 (2017).
77. Dion, M. L., Sumner, J. L. & Mitchell, S. M. Gendered citation patterns across political science and social science methodology fields. *Political Anal.* **26**, 312–327 (2018).
78. Dworkin, J. D. et al. The extent and drivers of gender imbalance in neuroscience reference lists. <https://arxiv.org/2001.01002> (2020).
79. Zhou, D. et al. Gender diversity statement and code notebook v1.1. <https://github.com/dalejn/cleanBib> (2020).

Acknowledgements

The authors especially thank Dr. Erin Teich, Dr. Linden Parkes, Darrick Lee, Dr. Jakob Hansen, Zoe Cooperband, and Dr. Lia Papadopolous for their helpful comments and insightful feedback. This work was funded by the Army Research Office through contract

number W911NF-16-1-0474. D.S.B. and A.S.B. also acknowledge additional support from the John D. and Catherine T. MacArthur Foundation, the Alfred P. Sloan Foundation, the ISI Foundation, the Paul Allen Foundation, the Army Research Laboratory (W911NF-10-2-0022), the Army Research Office (Bassett-W911NF-14-1-0679, DCIST-W911NF-17-2-0181), and the National Science Foundation (NSF PHY-1554488, BCS-1631550, and IIS-1926757). The content is solely the responsibility of the authors and does not necessarily represent the official views of any of the funding agencies.

Author contributions

A.S.B. and J.Z.K. devised the experiments, performed data analysis, and crafted initial draft. D.S.B. secured funding for the work, provided feedback on the direction of the research, and edited the manuscript. J.Z.K. and D.S.B. managed all steps required for the inclusion of the Citation Diversity Statement. All authors revised the manuscript and approved the final version.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s42005-021-00725-x>.

Correspondence and requests for materials should be addressed to Dani S. Bassett.

Peer review information *Communications Physics* thanks the anonymous reviewers for their contribution to the peer review of this work.

Reprints and permission information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2021