Topological exploration of artificial neuronal network dynamics

Jean-Baptiste Bardin, Gard Spreemann, and Kathryn Hess

Laboratory for Topology and Neuroscience, Brain Mind Institute, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland

September 23, 2018

Abstract

One of the paramount challenges in neuroscience is to understand the dynamics of individual neurons and how they give rise to network dynamics when interconnected. Historically, researchers have resorted to graph theory, statistics, and statistical mechanics to describe the spatiotemporal structure of such network dynamics. Our novel approach employs tools from algebraic topology to characterize the global properties of network structure and dynamics.

We propose a method based on persistent homology to automatically classify network dynamics using topological features of spaces built from various spike-train distances. We investigate the efficacy of our method by simulating activity in three small artificial neural networks with different sets of parameters, giving rise to dynamics that can be classified into four regimes. We then compute three measures of spike train similarity and use persistent homology to extract topological features that are fundamentally different from those used in traditional methods. Our results show that a machine learning classifier trained on these features can accurately predict the regime of the network it was trained on and also generalize to other networks that were not presented during training. Moreover, we demonstrate that using features extracted from multiple spike-train

distances systematically improves the performance of our method.

1 Introduction

A major objective in neuroscience is to understand how populations of interconnected neurons perform computations and process information. It is believed that the dynamics of a neuronal network are indicative of the computations it can perform. Its dynamics are affected by how the neurons are physically connected and by the activity history of the neurons. Understanding this spatiotemporal organization of network dynamics is essential for developing a comprehensive view of brain information processing mechanisms, the *functional connectome*. Two neurons can be considered "functionally connected" if their dynamics are similar or if one appears highly likely to spike causally after the other. The same notion of functional connectivity can be considered also on a macroscopic level, where one can study the causal relationships between brain regions. The notion can also be formalized for similarly structured systems from outside of neuroscience. Techniques like the one we present in this paper thus have broad applicability.

Furthermore, it is well known that certain neuronal systems can play multiple roles, characterized by different patterns of activity. For example, neurons in the thalamus have tonic or phasic behavior depending on the afferent signals and neuromodulators they receive or on different phases of the sleep cycle [38]. Another example is the hippocampus, which plays a role both in memory and in navigation. Researchers have also observed distinct rhythms in EEG recordings in awake and in sleeping rats [5].

An understanding of network dynamics is also of medical importance, as many neurological disorders are characterized by abnormal global or local activity. During epileptic seizures, for instance, EEG recordings show an increase in the amplitude of neural oscillations [13]. In Alzheimer's disease, one observes a shift in the power spectrum toward lower frequencies and a decrease in the coherence of fast rhythms [20].

Partly because of the clinical importance of neural dynamics, various methods have already been developed to automatically detect abnormal regimes, for example those related to epileptic seizures [2, 36, 21]. The best ones rely on artificial neural networks. Here we propose a novel approach using techniques from topological data analysis, a part of applied mathematics.

Traditionally, neuroscientists have analyzed functional networks using pairwise neuron statistics and graph theory. Such methods often neglect certain global structures that may be present in the dynamics. The analysis of network dynamics using alternative methods from topological data analysis has recently enjoyed success [34, 9, 11, 15, 8, 35]. These methods provide information about connectedness, adjacency, and global structure such as holes (of various dimension) in a dataset[1]. In particular, persistent homology detects holes or cavities and quantifies how robust these are with respect to a threshold variable related to the dynamics of the system.

Several interesting properties of neuronal network structure and function have been revealed through these recent developments. For example, persistent homology has been applied to detect and characterize changes in the functional connectome in disorders such as autism and attention deficit disorder [25], in the Parkinson mouse model [19], and after injection of a hallucinogenic substance [32]. It has also been employed to describe brain function during different tasks, such as multimodal and unimodal speech recognition [22]. Moreover, the homology of a digital reconstruction of the rat cortical microcircuit revealed that the brain substructure is non-random, that it is substantially different from various other null-models, and that its activity tends to concentrate in areas with greater local organization [33]. In the same article, homology was shown to distinguish the dynamics arising from different stimuli injected into the digital reconstruction. It is also interesting to note that the mammalian brain seems to encode topological information, as there are strong indications that the place cells [30] in the hippocampus build a topological, rather than geometric, representation of the animal's surroundings [10]. The latter article also shows how such a map can be encoded by a spiking network of neurons.

To this day, the few articles in which persistent homology has been applied to *in vivo* [15, 34] or synthetic [35] spike data have all used spike train (Pearson) correlation as the distance between neurons. The use of correlations requires one to make specific assumptions about neural coding that may not be reasonable or relevant in all research areas. There exists a wide variety of spike train distances and similarity metrics, and the restrictiveness of their assumptions and the kind of information they encode can vary significantly. As we demonstrate in this paper, the appropriate notion of spike train distance to use depends on context, and it can also be beneficial to combine

---

[1]The dataset is turned into a mathematical *space* in a manner detailed in section 4.2. It is in the context of such a space that the notion of holes arises.

several of them.

In this paper, we simulate activity in an artificial network of neurons to generate spiking data and build weighted graphs based on various spike train similarities. These graphs are then transformed into topological spaces, which are analyzed using persistent homology. Finally, we extract simple features from topological invariants and use them to train a classifier for predicting the global network dynamics. These topological features are fundamentally different from those that might arise from graph-theoretic or other classical methods, as they take into account relations within triplets and not just pairs of neurons. Our results show that it is possible to perfectly predict network regimes from a few features extracted from the persistent homology. The trained classifier also predicts with high accuracy the dynamics of networks other than that on which it was trained. Finally, our results illustrate the importance of employing several spike train similarities, as the best performance was achieved using a combination of them.

2   Results

In this section, we summarize our work, namely the simulation of network dynamics, the processing of spike trains, the topological analysis and feature selection, and the classification method, before presenting our results. A more detailed explanation of the method can be found in section 4.

2.1   Simulation of a downscaled Brunel network

We consider simulated activity in the Brunel network [4], a simple and well-studied *in silico* model network of sparsely connected excitatory and inhibitory neurons. For computational reasons, we use a downscaled version of the network as described below.

2.1.1   The Brunel network

The Brunel network consists of two homogeneous sub-populations of excitatory (here indexed by E) and inhibitory (indexed by I) neurons modelled by a current-based leaky integrate-and-fire (LIF) model.

Each of the N neurons has a membrane potential $V_{\mathrm{m}}(t)$ whose dynamics are described by a differential equation. Once the membrane potential reaches a threshold value V_θ, the neuron

4

sends a spike through its synapses to its post-synaptic neurons, and its potential resets to a value V_{r}. The synapses are δ-current synapses, i.e., after a delay D, each pre-synaptic spike induces a positive (respectively, negative) jump in the membrane potential of the post-synaptic neurons if the pre-synaptic neuron is excitatory (respectively, inhibitory). The excitatory sub-population is four times larger than the inhibitory one ($N_{\mathrm{E}} = 4N_{\mathrm{I}}$) in accordance with cortical estimations [28], but their synapses are relatively weaker. Formally, if the excitatory synapses induce an increase of membrane potential of J, the inhibitory ones will induce a decrease of $-gJ$ for some $g > 1$. Every neuron receives K inputs coming from a fixed proportion P of the neurons in each sub-population, i.e., $K = P(N_{\mathrm{E}} + N_{\mathrm{I}})$. Furthermore, each neuron receives $C_E = PN_{\mathrm{E}}$ external excitatory inputs from an independent Poisson population (of size $C_E N$) with fixed rate ν_{ext}. The relative synaptic efficiency (g) and the external population rate (ν_{ext}) are the free parameters with respect to which we study network dynamics, once we have fixed the other model parameters, in particular J, P, and D. We adopt the convention of Brunel's original article [4] and express ν_{ext} as a multiple of the minimal external rate necessary to trigger spiking in the neurons without recurrent connections, denoted ν_{θ}.

Because computing persistent homology is expensive for large and dense spaces, which tend to arise from large and dense networks, the number N of neurons was reduced from 12,500 in [4] to 2,500. Such a downscaling of the network while N/K is kept constant will result in an increase in the correlation between the neurons [37], more salient oscillations in the network dynamics [18], and potentially a loss in the diversity of network dynamics. To prevent these undesirable effects, a correction to the synaptic strength J was applied, and the external population was modified according to [18]. Specifically, the synaptic strength J was adjusted to keep JK constant, and the rate of the external population ν_{ext} was increased. An external inhibitory population with appropriate rate was also introduced to preserve the mean and variance of the external inputs. The external rate correction is relevant only when neurons are expected to show irregular firing, i.e., in the regimes where inhibition dominates ($g > 4$).

We generated three versions of the Brunel network to validate our method across different networks.

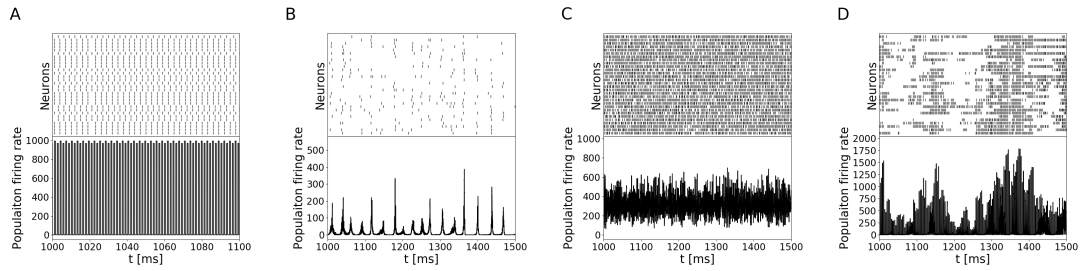**Version 1:** Relatively sparse connectivity ($P = 10\%$) and fast synaptic transmission ($D = 1.5$

Figure 1. The four regimes produced by the downscaled Brunel network. Each panel shows the spiking activity (raster plot) of 30 randomly selected neurons (top) and the population rate (bottom). **(A)** A 100 ms segment of simulation in the synchronous regular regime (SR, $g = 2, \nu_{\text{ext}}/\nu_\theta = 3$, version 1). **(B)** A 500 ms segment of simulation in the synchronous irregular regime (SI, $g = 5, \nu_{\text{ext}}/\nu_\theta = 1$, version 1). **(C)** A 500 ms piece of simulation in the asynchronous irregular regime (AI, $g = 5, \nu_{\text{ext}}/\nu_\theta = 2$, version 1). **(D)** A 500 ms piece of simulation in the alternative regime (Alt, $g = 4, \nu_{\text{ext}}/\nu_\theta = 2$, version 3).

ms, $J = 0.1$ mV).

Version 2: Denser connectivity ($P = 40\%$), fast and strong synaptic transmission ($D = 1.5$ ms, $J = 0.2$ mV).

Version 3: Denser connectivity ($P = 40\%$), slow and strong synaptic transmission ($D = 3.0$ ms, $J = 0.2$ mV).

A complete description of the final model and parameter sets, following the formalism in the field [29], is presented in Tables 2 and 3.

2.1.2 Simulations performed

Each network was simulated for 20 seconds of biological time with 28 different values of the pairs of free parameters g and $\nu_{\text{ext}}/\nu_\theta$. These pairs form a rectangular grid in the parameter space, with g taking values from 2 to 8 and $\nu_{\text{ext}}/\nu_\theta$ taking values from 1 to 4. Since the network is connected according to a random model, each simulation was repeated 10 times with different network instantiations, resulting in a total of 280 simulations for each network version. We recorded the spiking times of all neurons, as well as the overall population firing rate, for all the simulations.

Four distinct activity regimes, shown in Figure 1, were identified by inspecting the simulations:

SR: a regime characterized by synchronized neurons behaving as high-frequency oscillators, clustered in a few groups, similar to the synchronous regular regime (Figure 1A),
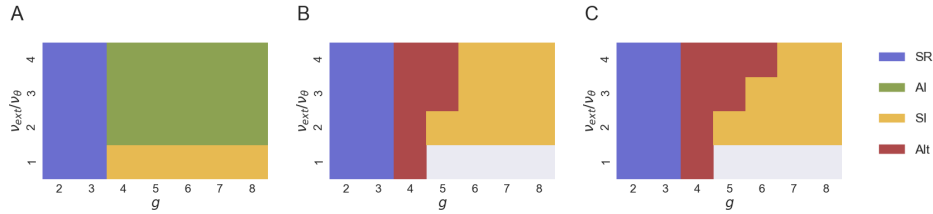
6

Figure 2. Diagrams of the different regimes for each version of the Brunel network (**(A)**, **(B)** and **(C)** corresponding to versions 1, 2, and 3 of the network, respectively) in the parameter space $(\nu_{\text{ext}}/\nu_\theta, g)$. The white areas indicate where no neurons fired and no data is available.

SI: a regime characterized by a slow oscillatory global pattern and synchronous irregular firing of individual neurons (Figure 1B),

AI: a regime characterized by asynchronous irregular firing of individual neurons (Figure 1C),

Alt: a regime characterized by neurons alternating between periods of silence and periods of rapid firing (Figure 1D).

Note that the Alt regime is not present in the full-size network. This is not an issue for us, however, since our goal is to discriminate between different regimes, not to understand the Brunel network per se.

For each of the three networks, we visually identified the network regime for every pair of parameters $(g, \nu_{\text{ext}}/\nu_\theta)$. The result is shown in Figure 2. The simulations in which none of the neurons fired were removed from the analysis (40 simulations for versions 2 and 3). Note that the first network (version 1) does not exhibit the Alt regime, while versions 2 and 3 do not exhibit the AI regime. This issue is addressed in section 2.4.

2.2 Spike train similarities

We used three different measures of spike train similarity to compare the recorded neuron activity in the networks.

One is the widely used Pearson correlation. It is often employed in analyzing spiking data because it has been shown to encode particular information that is not present in the firing rate alone. For example, in the auditory cortex of the marmoset, Pearson correlation encodes the purity of sounds [12]. It can also be used to infer connectivity or extract information about

network function [6]. However, it is tied to the correlation population coding hypothesis [31], and thus may not be relevant to the problem at hand. We therefore also employed two complementary measures: SPIKE-synchronicity [23] and SPIKE-distance [24]. Both are exploratory measures relying on an adaptive time window to detect cofired spikes and involve a pairwise similarity measure of spike trains. Conceptually, the size of the window depends on the local firing rate of the two neurons under consideration. If one of the neurons has a high local firing rate, then the time window will be short, while if both neurons have low local firing rates, the time window will be longer. SPIKE-synchronicity is the fraction of cofired spikes according to this adaptive window, while SPIKE-distance is the average over time of a dissimilarity profile between spike trains. See equations (7)–(13) in [24] for details.

We computed Pearson correlations by time-binning the spike trains with a 2 ms time window and a binning error correction, as described in section 4. The SPIKE measures were computed using the Python package PySpike [27].

2.3 Persistent homology

In *topology*, a branch of mathematics, one works with very general objects called *spaces*. Spaces have a notion of "nearness", but in general lack more geometric structure such distances or angles, as well as familiar algebraic structure. The unaccustomed reader may still gain intuition about what is meant by a space by thinking of geometric objects. One should still keep in mind that the spaces we consider may be defined entirely intrinsically without any reference to some ambient Euclidean space.

Algebraic topology, then, concerns itself with describing a space X in terms of algebraic invariants that capture global properties of the space. Examples include the Betti numbers, which can be thought of as the numbers of components and of n-dimensional unfilled cavities in the space. As these notions can be defined intrinsically, i.e., without any reference to how the space is embedded in Euclidean space, they are useful in analyzing spaces arising from abstract data where no such embedding can be constructed in a principled way. We consider here only the zeroth Betti number $b_0(X)$, which is the number of connected components, and the first Betti number $b_1(X)$, which is the number of one-dimensional unfilled loops. In the special case

of graphs — which is *not* the case we consider — these are precisely the number of connected components and the number of cycles, respectively.

The spaces we study here will be built from spiking data, and we are interested in how these algebraic invariants change as a function of a spike train similarity threshold. We therefore build a *filtration*, a multi-scale sequence of spaces depending on a threshold, and compute *persistent homology*, a multi-scale invariant that captures Betti numbers (which are then often also referred to as *Betti curves* to reflect their scale-/threshold-dependent nature). For details, see section 4. More background information can be found in the survey [14] and its bibliography.

As is common in topological data analysis, we follow the convention that edges with low weights are to be considered "most important" and enter first in the filtration (see section 4 for the definition). The correlation and SPIKE-synchronicity values were transformed through the function $x \mapsto 1 - x$ so that they range from 0 to 1 with 0 being the value assigned to a pair of identical spike trains (i.e., we work with dissimilarity measures). We compute persistent homology in dimensions 0 and 1.

2.4 Classifying network dynamics

We used the output of persistent homology of the space built from pairwise spike train similarities as an input feature for machine learning in order to discern information about the global network dynamics.

From every filtration, four simple features were extracted:

- from the Betti-0 curve, the area under the curve and the filtration value at which it starts to decrease,

- from the Betti-1 curve, the global maximum and the area under the curve.

As a result, a total of 12 features were extracted from each simulation (four features per filtration, three filtrations from the three spike train similarity measures). For some simulations in the SR regime, all the pairwise similarities attained the maximal value, resulting in a space with no topological features and a constantly zero Betti-1 curve. The filtration value at which the curve starts to decrease was defined to be 0 in this case.
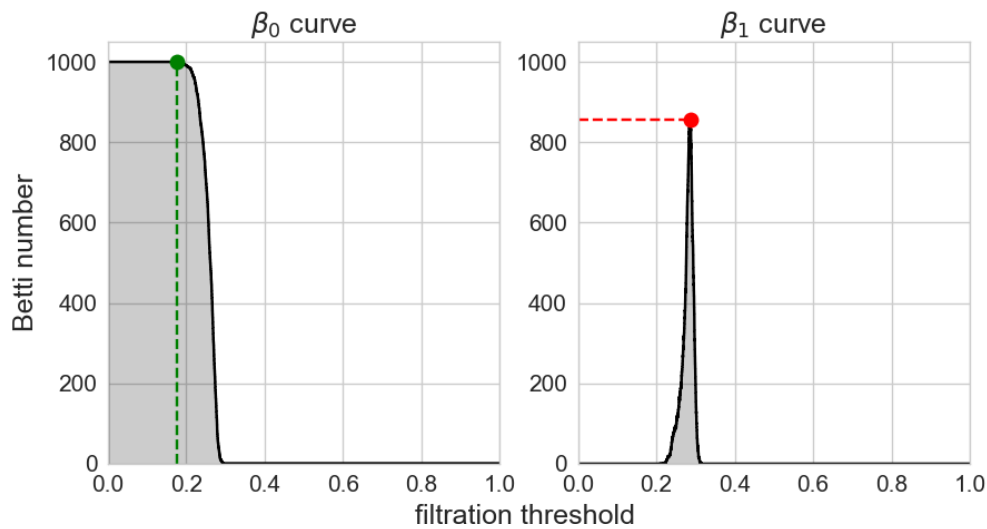
9

Figure 3. Example of three of the features extracted from a filtration: the filtration threshold at which the Betti-0 curve (left) starts to decrease, the global maximum (red) of the Betti-1 curve, and the area under each curve (grey area)

Before doing any classification, potentially good features were selected by plotting all the features against each other for the samples coming from network version 1. Six features were selected by visual inspection because they were deemed to produce non-overlapping clusters. These features are the area under the Betti-0 curve for the three similarity measures, the area under the Betti-1 curve for correlation and SPIKE-synchronicity, and the maximum of the Betti-1 curve for the SPIKE-distance. These features were among the ones with the highest mutual information in the three networks, although their ranking varied between the networks. Section 4.4, in particular Figure 7, provides further details of the feature selection process.

Four different training sets were used for classification, three of which were composed of randomly selected samples (90%) coming from a specific network version, while the last set was the union of the three other sets. For each training set, an L^2-regularized support vector machine (SVM) classifier was trained to identify the different regimes. The classifier was composed of four sub-classifiers, each of which had to distinguish one particular regime from the others. The final decision was computed in a one-vs-rest manner. The regularizing hyperparameter was selected with a 10-fold cross-validation.

When assessing the performance of the classifier for network version k using multi-class accuracy, we validated it on three test sets: one composed of the 10% of the samples from

10

version k that had not been used for training and two containing all the valid samples from one of the other network versions. A sample was considered valid if it was labeled with a regime that was present in the training sets. For example, when the classifier trained on version 1 was tested on version 2, the samples labeled Alt were ignored, since no version 1 networks exhibit the Alt behavior. The performance accuracy and the numbers of valid samples are reported in Table 1. The trained classifiers all achieved perfect accuracy (100%) on the network version they were trained on, indicating that the topological features extracted are sufficient to perfectly discriminate the regimes of the training network. Moreover, they also generalized well to other versions, with 94.26% accuracy on average, suggesting that the topological features extracted are consistent across the three network versions.

Additionally, we combined all the samples from the three networks and used a 90%-10% training-testing sample repartition, attaining perfect classification ("All versions" row in Table 1) of the four regimes. This provides the complementary information that the Alt and AI regimes are also distinguishable from one another, since none of the network versions can exhibit both regimes.

	Testing set			
Training set	Ver. 1	Ver. 2	Ver. 3	All ver.
Version 1	100% (28)	86.67% (180)	91.18% (170)	89.68% (378)
Version 2	97.69% (130)	100% (24)	93.33% (240)	95.18% (394)
Version 3	99.23% (130)	99.17% (240)	100% (24)	99.23% (394)
All versions	100% (28)	100% (24)	100% (24)	100% (76)

Table 1. Classification accuracy for each pair of training and testing sets. The training sets were comprised of 90% of the samples coming from a specific network (version 1, 2 or 3), or 90% of all samples (all versions). The testing sets contained the remaining samples from the network version that are not used for training. i.e., the remaining 10%. The number of samples in every testing set is reported in parentheses.

Finally, we checked whether the persistent homology-derived features provide complementary information when based on different similarity measures, and thus whether it can be advantageous to use several of them together for classification. The same classification experiments were repeated using either all the computed features or only the features coming from one of the similarity measures. The classifier accuracies were compared with those from the previous computations (Figure 4). Although the accuracies obtained using features coming from a single similarity measure were satisfactory (on average 79.10%, 92.35%, and 80.73% accuracy for correlation,
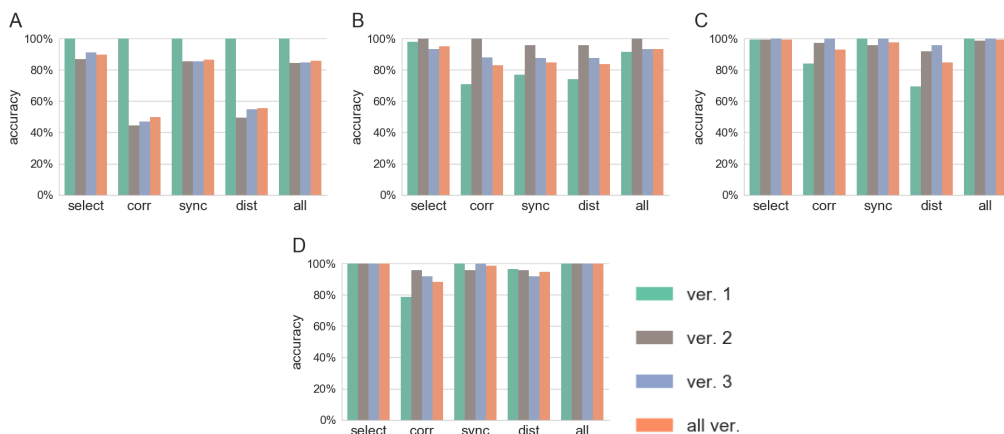
11

Figure 4. Testing accuracy of the classifier trained on samples from version 1 (**A**), version 2 (**B**), version 3 (**C**) and all the samples (**D**). In each panel, the classification accuracy for the test samples from each network and all the samples together is reported for five sets of features. "Select" designates the features we visually selected. "Corr", "sync" and "dist" designate the features extracted using the correlation, SPIKE-synchronicity, and SPIKE-distance, respectively. "All" designates the set of all the features.

SPIKE-synchronicity, and SPIKE-distance, respectively), better performance was consistently attained by a combination of measures. Moreover, selection of potentially good features yields the best results (94.94% and 96.63% average accuracy with the "all" set and "select" set, respectively).

3 Discussion

In this paper we analyzed the dynamics of spiking networks of neurons using features derived from persistent homology. We generated three versions of a simple artificial network of LIF neurons (a downscaled version of the Brunel network) by modifying connectivity density, synaptic delay, and synaptic strength. Activity in the networks was then simulated with 28 pairs of the free parameters (external population rate and relative synaptic efficiency values). Across all the simulations, four regimes of activity were observed based on the pattern of the global population firing rate and the individual neuron spiking times.

For each simulation, we computed three pairwise spike train similarity measures: Pearson correlation, SPIKE-synchronicity, and SPIKE-distance. We computed the persistent homology of the flag complex of the weighted graph coming from each similarity measure and extracted simple features from the zeroth and first Betti curves. The interesting features were selected by

visual inspection of the sample distribution. Finally, an SVM classifier was trained to identify the dynamics regimes of the simulations.

Our experiments showed that it is possible to perfectly predict the dynamic regime in simulations coming from the network trained on, and from other networks with a high degree of accuracy, as long as some samples of the regimes in question were available during training.

We also illustrated the importance of using and combining several similarity measures. Indeed, SPIKE-synchronicity carries more information, and does so more consistently across the network versions, than the other two measures, but the best accuracies were consistently obtained when an ensemble of features selected by visual inspection was used. Moreover, if one were to automatically select the features based on a score, we showed that the mutual information between features and the regime label is a good indicator to consider.

We tested our method in the context of a simple network. It would be interesting to test it also with more complex networks, with neurons and synapses modeled in greater detail. Topological features can also be extracted from other types of neural data, such as the population firing rate or neuron voltage traces. We consider the examination of how the topological methods perform in classifying such data as interesting future work.

Here we have illustrated just one concrete use of topological data analysis (TDA) in the study of network dynamics, but the class of methods should be applicable to a wide variety of systems from within and without neuroscience. To the best of our knowledge, there have been no previous attempts at applying TDA to automatic detection of regimes in spiking neural networks, since they are usually identified analytically [4, 18] and can often be discriminated visually. However, a topological approach to this task may be interesting in recordings of real data, such as EEG or fMRI. One might, for example, investigate the feasibility of solving a more subtle task, such as automatic detection of movement intention or seizure detection in epileptic patients.

Although great progress has been made in neuroscience since the first recording of a neuron activity in 1928 [1], a unified model of the brain across its different scales is still lacking, and many hard challenges have barely been attempted. Recent work has highlighted how TDA could help shed new light on both brain structure and dynamic and are promising advances towards a more comprehensive understanding of the brain. The method we have outlined in this paper takes a novel view of one challenge, the automated classification of neuronal dynamics, by considering

features that are topological in nature. We believe that including such features will be of great help in the understanding of both structural and dynamical aspects of neuronal networks and other similarly structured systems.

4   Methods

We give here the full details of our computations, and expand on the topological constructions involved in the analysis.

4.1   Network simulations

A complete specification of the three simulated networks following formalism and notation commonly used in the field [29] can be found in Tables 2 and 3.

All the networks were simulated with 28 pairs of parameter values for the relative strength between inhibition and excitation g (integer values from 2 to 8) and the external population rate $\nu_{\mathrm{ext}}/\nu_\theta$ (integer values from 1 to 4). The systems were simulated 10 times for each parameter pair, for a total of 280 simulations per network. The simulations were performed with the Brian2 simulator [16], with a time step of 0.01 ms and a total biological time of 20 seconds. Because of the downscaling of the network, the synaptic transmission J was increased compared to that used in [4] in order to keep $C_E J$ constant, and an external inhibitory population was introduced [18] when the spiking of neurons was expected to be irregular [4]. This external population was modeled by a Poisson process with rate

$$\nu_{\mathrm{bal}} = \frac{\sigma_{\mathrm{i}}^2 - \sigma_{\mathrm{loc}}^2 - \tau_m \nu_{\mathrm{ext}} J_{\mathrm{ext}}^2}{\tau_m J_{\mathrm{ext}}^2 (1 + g^2)} \tag{1}$$

as in [18]. Here, σ_{i} is the variance of input in the original network and σ_{loc} is the variance due to local input from recurrent connections in the downscaled network. The variances can be approximated as

$$\sigma_{\mathrm{i}}^2 = C_E^* J^{*2} \tau_m \left(\nu_{\mathrm{ext}} + \nu_0 \left(1 + g^2 \gamma \right) \right) \tag{2}$$

$$\sigma_{\mathrm{loc}}^2 = C_E J^2 \tau_m \nu_0 \left(1 + g^2 \gamma \right). \tag{3}$$

14

The parameters for the original network are labelled by an asterisk and differ from their counterpart in the downscaled network by a scaling factor α such that $C_E = C_E^*/\alpha$ and $J = \alpha J^*$. From equations (2), (3) and (1), we obtain

$$\nu_{\text{bal}} = \frac{(1/\alpha - \alpha)\,\nu_{\text{ext}} + (1/\alpha - 1)\,\nu_0\left(1 + g^2\gamma\right)}{\tau_m J_{\text{ext}}^2(1 + g^2)}. \tag{4}$$

Here ν_0 is the stationary frequency of the original population and can be approximated by [4]

$$\nu_0 = \frac{\nu_{\text{ext}} - \nu_\theta}{g\gamma - 1}. \tag{5}$$

4.2   Topological framework

In *algebraic topology*, a well-established field of mathematics, one studies *topological spaces* by turning them into well-behaved *algebraic invariants* and deducing properties of the spaces from those of the algebraic objects. We shall not define any of these concepts precisely here, but will instead give relevant examples of both. See for example [17] for an introductory textbook that includes all the details with full precision.

A space will in our context mean a kind of object that is built from certain geometric pieces by specific rules that reflect the data of the dynamics (or structure) of systems of neurons. These objects are "high-dimensional" in the sense that they can be thought of as inhabiting $\mathbb{R}^n$ for (possibly) very large n, but the way that they do so is of no relevance to the algebraic invariants we employ here. Moreover, the Euclidean coordinatization of the space in general contains no information about the underlying system, so it is therefore to our benefit that the topological methods ignore it.

A *simplicial complex* can be thought of as a higher-dimensional analog of a graph, with its constituent building blocks referred to as *simplices*. In addition to comprising vertices (0-dimensional pieces, or 0-simplices) and edges (1-dimensional pieces, 1-simplices), a simplicial complex may have filled triangles (2-simplices), filled tetrahedra (3-simplices), and so on. Just as for graphs, the way these simplices fit together is subject to some rules. First, a p-simplex is identified uniquely by a set of $p + 1$ vertices. We technically impose a global ordering on all

A	Model Summary
Populations	Four: excitatory, inhibitory, excitatory external input, inhibitory external input
Topology	-
Connectivity	Random convergent connections with probability P and fixed in-degree of $C_E = PN_E$ and $C_I = PN_I$
Neuron model	Leaky integrate-and-fire, fixed voltage threshold, fixed absolute refractory time (voltage clamp)
Channel models	-
Synapse model	δ-current inputs
Plasticity	-
Input	Independent fixed-rate Poisson spike trains to all neurons
Measurements	Spike timing, population firing rate.

B	Populations	
Name	**Elements**	**Size**
E	Iaf neuron	N_E
I	Iaf neuron	$N_I = \gamma N_E = 0.25 N_E$
E_{ext}	Poisson generator	$C_E(N_E + N_I)$
I_{ext}	Poisson generator	$C_I(N_E + N_I)$

C	Connectivity		
Name	**Source**	**Target**	**Pattern**
EE	E	E	Random convergent $C_E \to 1$, weight J, delay D
IE	I	E	Random convergent $C_E \to 1$, weight J, delay D
EI	E	I	Random convergent $C_I \to 1$, weight $-gJ$, delay D
II	I	I	Random convergent $C_I \to 1$, weight $-gJ$, delay D
Ext_{exc}	E_{ext}	$E \cup I$	Non-overlapping $C_E \to 1$, weight J_{ext}, delay D
Ext_{inh}	I_{ext}	$E \cup I$	Non-overlapping $C_I \to 1$, weight J_{ext}, delay D

D	Neuron and Synapse Model
Name	- (Brian2)
Type	leaky integrate-and-fire, δ-synapse
Membrane potential	$\tau_m \dot{V}_m(t) = -V_m(t) + RI(t)$ if not refractory ($t > t^* + \tau_{rp}$) $V_m(t) = V_r$ while refractory ($t^* < t \leq t^* + \tau_{rp}$) $I(t) = \frac{\tau_m}{R} \sum_{\tilde{t}} \omega \delta(t - (\tilde{t} + D))$
Spiking	if $V(t-) < \theta \wedge V(t+) \geq \theta$ 1. set $t^* = t$ 2. emit spike with time-stamp t^*

E	Input	
Type	**Description**	

Poisson generators	Fixed rate ν, $C_E + C_I$ generators per neuron, each generator projects to one neuron:
	     if excitation dominates ($g \leq 4$): E_{ext} rate $= \nu_{\text{ext}}$, I_{ext} rate $= 0$
	     if inhibition dominates ($g > 4$): E_{ext} rate $= \nu_{\text{ext}} + \nu_{\text{bal}}$, I_{ext} rate $= \nu_{\text{bal}}/g$	

F	Measurement
Spiking times of all neurons, population firing rate	

Table 2. Description of the neuronal network following the formalism of [29], part 1/2.

G	Network parameters			
		Network configuration		
Parameters	Common	ver. 1	ver. 2	ver. 3
Number of excitatory neurons N_E	2000	-	-	-
Number of inhibitory neurons N_I	500	-	-	-
Excitatory synapses per neuron C_E	-	200	800	800
Inhibitory synapses per neuron C_I	-	50	200	200
H	Neurons parameters			
		Network configuration		
Parameters	Common	ver. 1	ver. 2	ver. 3
Membrane time constant τ_m/ms	20	-	-	-
Refractory period τ_{rp}/ms	2	-	-	-
Firing threshold V_θ/mV	20	-	-	-
Resting potential V_{rest}/mV	0	-	-	-
Reset potential V_r/mV	10	-	-	-
Post-synaptic potential (PSP) amplitude from reccurent connections J/mV	-	0.5	1.0	1.0
PSP amplitude from external connections J_{ext}/mV:				
if excitation dominates ($g \leq 4$)	-	0.5	1.0	1.0
if inhibition dominates ($g > 4$)	-	0.1	0.2	0.2
Synaptic delay D/ms	-	1.5	1.5	3

Table 3. Description of the neuronal network following [29], part 2/2.

the vertices in the complex and write simplices as tuples respecting this ordering. Second, if a p-simplex $\sigma = (v_0, v_1, \ldots, v_p)$ is present in a simplicial complex K, then its $p+1$ *boundary* $(p-1)$-simplices

$$(v_1, v_2, v_3, \ldots, v_p), (v_0, v_2, v_3, \ldots, v_p), (v_0, v_1, v_2, \ldots, v_p), \ldots, (v_0, v_1, v_2, \ldots, v_{p-1})$$

are all also required to be present in K. Note that the definition of a boundary and the associated rule are entirely combinatorial, although they do have a geometric interpretation if we think of simplices as geometric objects, as illustrated in Figure 5.
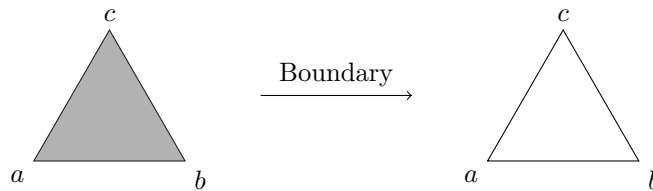

Figure 5. Left: A simplicial complex consisting of a single 2-simplex $((a, b, c))$, three 1-simplices $((a, b), (b, c)$ and $(a, c))$ and three 0-simplices $((a), (b)$ and $(c))$. Right: The boundary of (a, b, c) consists of the three 1-simplices and three 0-simplices that one would expect geometrically, giving geometric meaning to a purely combinatorially defined concept.

17

Simplicial complexes can encode data and domain-specific knowledge in different ways depending on the availability of structure and the type of information that is sought. In this work, we take as input a choice of spike train similarity and a spike train for each of n neurons considered. From this, we build a simplicial complex as follows.

Let G be the complete graph on n vertices (representing the neurons). The edge between two neurons i and j is assigned a weight $w(i,j)$ equal to the dissimilarity of the corresponding spike trains. A simplicial complex K is then formed by adding in every possible 2-simplex. As a space, this is not very interesting, as there is simply a filled triangle between every triple of neurons (one says that the space is contractible). The crucial part is that each 2-simplex is given a weight equal to the maximum of the weights given to its boundary edges, i.e.,

$$w(i,j,k) = \max\{w(i,j), w(i,k), w(j,k)\}.$$

We then consider a *filtration* of K, enabling us to study a sequence of thresholded versions of K. At the start of the filtration, the filtration consists only of the vertices of K. Then, as the threshold increases, 1- and 2-simplices from K appear if their weight is below the threshold, so as to include into the filtration pieces stemming from ever more dissimilar spike trains. See Figure 6 for an illustration.

The construction above is applicable to simplices of dimension higher than 2, so even though we stop at dimension 2 in our analysis, the following description employs generic dimensions p in order to simplify notation and give the bigger picture.

A basic algebraic invariant that we track as the dissimilarity threshold increases is the *Betti numbers*, giving rise to *Betti curves* for the filtration as a whole. The Betti numbers of a simplicial complex can be defined formally as the dimensions of the *homology* vector spaces of the complex, as we now sketch. Define $C_p(K)$ to be the collection of formal binary sums of the p-simplices in K. This makes $C_p(K)$ a vector space over the binary numbers, and allows us to view the boundary as an algebraic operation encoded in a linear map $\partial_p : C_p(K) \to C_{p-1}(K)$ given by

$$\partial_p(v_0, v_1, \ldots, v_p) = (v_1, v_2, \ldots, v_p) + (v_0, v_2, \ldots, v_p) + (v_0, v_1, \ldots, v_p) + \cdots + (v_0, v_1, \ldots, v_{p-1}).$$

One checks that the application of two consecutive boundary maps always gives zero, i.e., that $\partial_{p-1}(\partial_p(\sigma)) = 0$ for every p-simplex σ. This algebraic statement reflects the geometric fact that the boundary of a boundary is empty. A general collection of p-simplices — a sum in $C_p(K)$ — that has zero boundary is called a *p-cycle*. Figure 6 shows several examples.

It turns out that p-cycles that are not the boundary of collections of $(p + 1)$-simplices correspond geometrically to holes ($p > 0$) or connected components ($p = 0$) in the simplicial complex. *Persistent homology*, a widely employed construction in topological data analysis, tracks such holes/components as they appear and disappear across a filtration. The record of the "life" and "death" of such *topological features* provides valuable information about the filtration and thus about the underlying data. We do not use all of the data recorded in persistent homology, but instead just keep track of the *number* of holes in each dimension as a function of the filtration[2]. These integer-valued functions, called *Betti curves*, are the features we use for machine learning. An example of a Betti curve is given in Figure 6. We emphasize that the features captured by persistent homology may be much more global in nature than in this small example.[3]

4.3 Machine learning

Before doing any machine learning, the features selected were standardized. If a feature was not computable because there were no corresponding Betti curves, its value was set to 0.

For each version of the network, four training sets were formed, one containing 90% of the samples from a specific network version and a fourth one containing 90% of all the samples, stratified so that its distribution of the samples was representative of all the samples. One classifier per training set was trained and tested against four test sets: one for each network version, using the valid samples not in the training set, and the fourth one containing all the valid samples not used during training.

Support Vector Machine methods [7] using a radial basis function kernel with L^2-regularization were applied to classify the samples into the four different regimes. The multi-class classification was achieved by training four sub-classifiers with a one-vs-rest decision function. The regularization

---

[2]The reason for reducing the data of persistent homology to Betti curves is in part that the algebraic invariant produced by the former lacks many desirable properties.

[3]As an example, the reader is invited to build a torus as a simplicial complex, filter it from one side to another, and observe what features are captured by persistent homology in dimensions 0, 1, and 2.
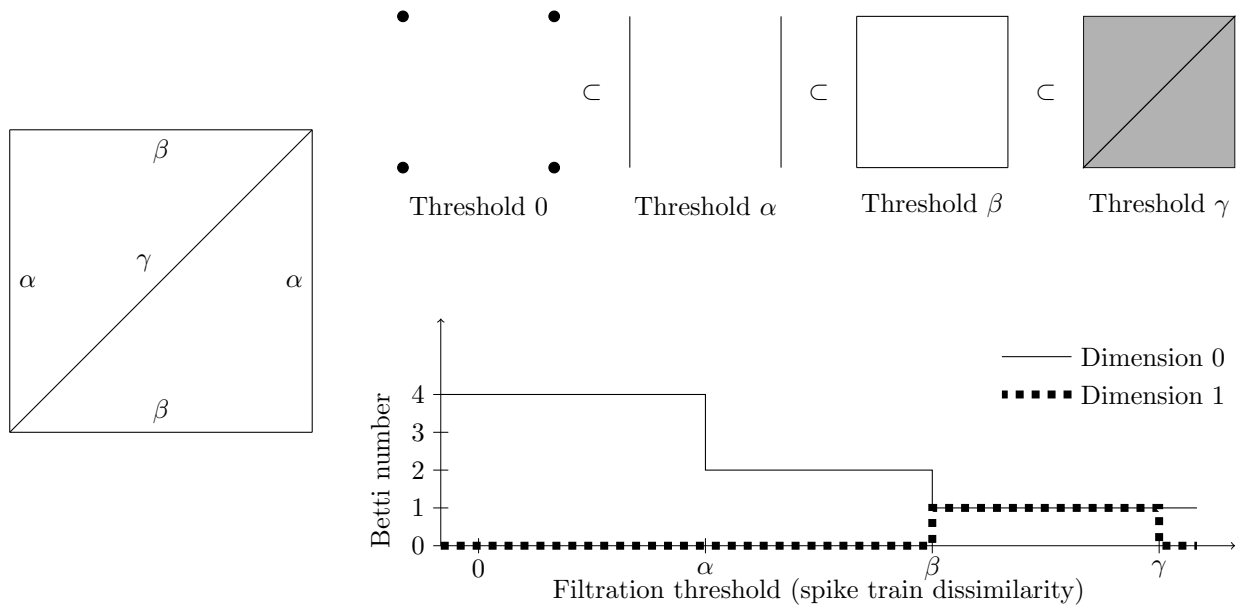
Figure 6. Left: A weighted graph G on four vertices/neurons. Assume that $0 < \alpha < \beta < \gamma$. Top: The filtered simplicial complex K built from G. The 0-simplices are drawn in a different way at threshold 0 to make them more visible. Bottom: The Betti curves of K in dimensions 0 and 1.

parameter was found by accuracy optimization thanks to 10-fold cross-validation.

The performance of the classifiers was assessed using an accuracy score.

4.4 Mutual Information

In section 2.4, we mentioned that mutual information between the features we selected by visual inspection and the regime labels was relatively high, suggesting that one could use the mutual information score to automatically select features when visual inspection would be time consuming or violate a need for automation or independence from human input. The mutual information between each feature and the labels for the three datasets is presented in Figure 7, where one can observe that some features, such as the area under the Betti-0 curve for correlation and SPIKE-distance, have a consistent mutual information score across the three datasets. This suggests that they are important features that allow the classifier to correctly sample from other datasets. Moreover, the area under the curve (AUC) features tend to have a higher score than the peak amplitude of the Betti curve. This is perhaps natural since the former includes information from all of the filtration, while the latter includes only one single aspect of it.
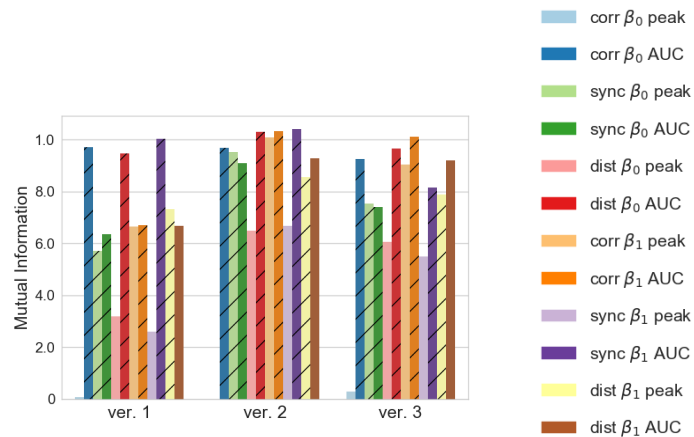
Figure 7. Mutual information between each feature and the label for the training sets obtained from the simulations of the different network versions. Features that were selected by visual inspection are represented with a hashed bar. The *peak* and *AUC* labels designate the peak amplitude and the area under the Betti curve features, respectively.

Acknowledgments

References

[1] E. D. Adrian. The basis of sensation. *British Medical Journal*, 1(4857):287, 1928.

[2] A. Alkan, E. Koklukaya, and A. Subasi. Automatic seizure detection in EEG using logistic regression and artificial neural network. *Journal of Neuroscience Methods*, 148(2):167–176, 2005.

[3] U. Bauer. Ripser. https://github.com/Ripser/ripser, 2016.

[4] N. Brunel. Dynamics of networks of randomly connected excitatory and inhibitory spiking neurons. *Journal of Physiology Paris*, 94(5-6):445–463, 2000.

[5] G. Buzsáki, L. S. Chen, and F. H. Gage. *Spatial organization of physiological activity in the hippocampal region: relevance to memory formation.* Elsevier, 1990.

[6] M. R. Cohen and A. Kohn. Measuring and interpreting neuronal correlations. *Nature Neuroscience*, 14(7):811–819, 2011.

[7] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, sep 1995.

[8] C. Curto. What can topology tell us about the neural code? *Bulletin of the American Mathematical Society*, 54:63–78, 2017.

[9] C. Curto and V. Itskov. Cell groups reveal structure of stimulus space. *PLoS Computational Biology*, 4(10):e1000205, 2008.

[10] Y. Dabaghian, V. L. Brandt, and L. M. Frank. Reconceiving the hippocampal map as a topological template. *eLife*, 3, 2014.

[11] Y. Dabaghian, F. Mémoli, L. Frank, and G. Carlsson. A topological paradigm for hippocampal spatial map formation using persistent homology. *PLoS Computational Biology*, 8(8), 2012.

[12] R. C. DeCharms and M. M. Merzenich. Primary cortical representation of sounds by the coordination of action-potential timing. *Nature*, 381 6583:610–613, 1996.

[13] R. S. Fisher, W. Van Emde Boas, W. Blume, C. Elger, P. Genton, P. Lee, and J. Engel. Epileptic seizures and epilepsy: definitions proposed by the international league against epilepsy (ilae) and the international bureau for epilepsy (IBE). *Epilepsia*, 46(4):470–472, 2005.

[14] R. Ghrist. Barcodes: the persistent topology of data. *Bulletin of the American Mathematical Society*, 45(1):61–75, 2008.

[15] C. Giusti, E. Pastalkova, C. Curto, and V. Itskov. Clique topology reveals intrinsic geometric structure in neural correlations. *Proceedings of the National Academy of Sciences*, 2015.

[16] D. Goodman and R. Brette. The brian simulator. *Frontiers in Neuroscience*, 3:26, 2009.

[17] A. Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.

[18] M. Helias, T. Tetzlaff, and M. Diesmann. Echoes in correlated neural systems. *New Journal of Physics*, 15, 2013.

[19] H. J. Im, J. Hahm, H. Kang, H. Choi, H. Lee, D. W. Hwang, E. E. Kim, J. K. Chung, and D. S. Lee. Disrupted brain metabolic connectivity in a 6-ohda-induced mouse model of parkinson's disease examined using persistent homology-based analysis. *Scientific Reports*, 6(September):1–11, 2016.

[20] J. Jeong. EEG dynamics in patients with Alzheimer's disease. *Clinical Neurophysiology*, 115(7):1490–1505, 2004.

[21] Y. U. Khan and J. Gotman. Wavelet based automatic seizure detection in intracerebral electroencephalogram. *Clinical Neurophysiology*, 114(5):898–908, 2003.

[22] H. Kim, J. Hahm, H. Lee, E. Kang, H. Kang, and D. S. Lee. Brain networks engaged in audiovisual integration during speech perception revealed by persistent homology-based network filtration. *Brain Connectivity*, 5(4):245–58, 2015.

[23] T. Kreuz, D. Chicharro, C. Houghton, R. G. Andrzejak, and F. Mormann. Monitoring spike train synchrony. *Journal of Neurophysiology*, 109(5):1457–72, 2013.

[24] T. Kreuz, M. Mulansky, and N. Bozanic. Spiky: a graphical user interface for monitoring spike train synchrony. *Journal of Neurophysiology*, 113(9):3432–3445, 2015.

[25] H. Lee, M. K. Chung, H. Kang, B.-N. Kim, and D. S. Lee. Persistent network homology from the perspective of dendrograms. *IEEE Transactions in Medical Imaging*, 12(2006):2381–2381, 2011.

[26] H. Markram. The Blue Brain Project. *Nature Reviews Neuroscience*, 7(2):153, 2006.

[27] M. Mulansky and T. Kreuz. PySpike — a python library for analyzing spike train synchrony. *SoftwareX*, 5:183–189, 2016.

[28] C. R. Noback, D. A. Ruggiero, and N. L. Demarest, Robert J. Strominger. *The Human Nervous System: Structure and Function*. Humana Press, 6th edition, 2007.

[29] E. Nordlie, M. O. Gewaltig, and H. E. Plesser. Towards reproducible descriptions of neuronal network models. *PLoS Computational Biology*, 5(8), 2009.

[30] J. O'Keefe and J. Dostrovsky. The hippocampus as a spatial map. preliminary evidence from unit activity in the freely-moving rat. *Brain Research*, 34(1):171–175, 1971.

[31] S. Panzeri, J. H. Macke, J. Gross, and C. Kayser. Neural population coding: combining insights from microscopic and mass signals. *Trends in Cognitive Sciences*, 19(3):162–172, 2015.

[32] G. Petri, P. Expert, F. Turkheimer, R. Carhart-Harris, D. Nutt, P. J. Hellyer, and F. Vaccarino. Complex brain networks: graph theoretical analysis of structural and functional systems. *Journal of the Royal Society Interface*, 2014.

[33] M. W. Reimann, M. Nolte, M. Scolamiero, K. Turner, R. Perin, G. Chindemi, P. Dłotko, R. Levi, K. Hess, and H. Markram. Cliques of neurons bound into cavities provide a missing link between structure and function. *Frontiers in Computational Neuroscience*, 11:48, 2017.

[34] G. Singh, F. Memoli, T. Ishkhanov, G. Sapiro, G. Carlsson, and D. L. Ringach. Topological analysis of population activity in visual cortex. *Journal of Vision*, 8(8):1–18, 2008.

[35] G. Spreemann, B. Dunn, M. B. Botnan, and N. A. Baas. Using persistent homology to reveal hidden covariates in systems governed by the kinetic ising model. *Phys. Rev. E*, 97:032313, Mar 2018.

[36] A. T. Tzallas, M. G. Tsipouras, and D. I. Fotiadis. Automatic seizure detection based on time-frequency analysis and artificial neural networks. *Computational Intelligence and Neuroscience*, 2007, 2007.

[37] S. J. van Albada, M. Helias, and M. Diesmann. Scalability of asynchronous networks is limited by one-to-one mapping between effective connectivity and correlations. *PLoS Computational Biology*, 2014.

[38] T. G. Weyand, M. Boudreaux, W. Guido, M. Tsanov, E. Chah, N. Wright, S. D. Vann, R. Reilly, T. Jonathan, J. P. Aggleton, S. M. O. Mara, H. J. Alitto, B. D. M. Iv, D. L.

Rathbun, and W. M. Usrey. Burst and tonic response modes in thalamic neurons during sleep and wakefulness. *Journal of Neurophysiology*, 85(3):1107–1118, 2001.