SIAM Review
Vol. 63, No. 1, pp. 67–99

Persistent Homology of Geospatial Data: A Case Study with Voting*

Michelle Feng†
Mason A. Porter‡

Abstract. A crucial step in the analysis of persistent homology is the transformation of data into an appropriate topological object (which, in our case, is a simplicial complex). Software packages for computing persistent homology typically construct Vietoris–Rips or other distance-based simplicial complexes on point clouds because they are relatively easy to compute. We investigate alternative methods of constructing simplicial complexes and the effects of making associated choices during simplicial-complex construction on the output of persistent-homology algorithms. We present two new methods for constructing simplicial complexes from two-dimensional geospatial data (such as maps). We apply these methods to a California precinct-level voting data set, and we thereby demonstrate that our new constructions can capture geometric characteristics that are missed by distance-based constructions. Our new constructions can thus yield more interpretable persistence modules and barcodes for geospatial data. In particular, they are able to distinguish short-persistence features that occur only for a narrow range of distance scales (e.g., voting patterns in densely populated cities) from short-persistence noise by incorporating information about other spatial relationships between regions.

Key words. persistent homology, topological data analysis, voting data, geospatial data

AMS subject classifications. Primary, 55N31; Secondary, 55-04, 55U10, 62R40, 91D20

DOI. 10.1137/19M1241519

Contents

*Received by the editors January 29, 2019; accepted for publication (in revised form) March 25, 2020; published electronically February 4, 2021.
https://doi.org/10.1137/19M1241519

†Department of Mathematics, University of California, Los Angeles, CA 90095 USA. Current address: Department of Computing + Mathematical Sciences, California Institute of Technology, Pasadena, CA 91125 USA (mfeng@caltech.edu, https://directory.caltech.edu/personnel/mfeng).
‡Department of Mathematics, University of California, Los Angeles, CA 90095 USA (mason@math.ucla.edu, https://www.math.ucla.edu/~mason/).

1. Introduction. Historically, the study of algebraic topology has been concerned with classifying topological spaces using algebraic invariants that describe their global properties [12, 23]. More recently, however, ideas from algebraic topology have also been applied to data sets as a way of examining the "shape" of data [12, 16, 19, 36]. One way to classify topological spaces is to distinguish them based on their number and types of holes. For example, a circle is distinct from a disk; we distinguish them based on the hole in the center of the circle. For two-dimensional (2D) geospatial[1] data, we can interpret holes as concrete geographical features like lakes or deserts. Previous applications of topological data analysis (TDA) in which space plays an important role include studies of the geography of country development [4], the spread of social [48] and biological [30] contagions, communication patterns in cities [3], voting in the "Brexit referendum" [46], continuum disk percolation [45], granular materials [37], flow networks in biological transport [40], and migration networks [25].

To identify holes in a data set, we need to assign a topological structure to the data and compute its homology groups. The homology of a topological space X is a set of

[1]Following the conventions of the demography community, we use the term "geospatial data" to refer to information about entities on or near Earth's surface that one can locate using some coordinate system (in our case, using latitude and longitude). In this paper, we use the term "geospatial" interchangeably with "geographical" (or "geographic"), in contrast to more general spatial data, which need not be based on geographical location. Additionally, it is common for 2D geospatial data to represent geographical objects (such as rivers) that are not inherently 2D. We are concerned with the dimension of the data itself, rather than with the dimensions of the underlying geographical features.

topological invariants that are represented by homology groups $\{H_k(X)\}_{k\in\mathbb{N}}$, where $H_k(X)$ describes k-dimensional holes in X. If X is a network, the dimension of $H_0(X)$ records the number of connected components and the dimension of $H_1(X)$ records the number of cycles. Because of the 2D nature of our focal data, we mostly restrict ourselves to computing these homology groups. Homology groups are particularly useful as topological invariants because of the existence of efficient combinatorial algorithms for computing the homology groups of simplicial and cellular complexes [36], as other topological invariants (such as homotopy) are less computationally tractable.

Persistent homology (PH) is the most common method of TDA for computing holes in data. Point clouds have an inherent 0-dimensional (0D) structure, and they thus have few interesting topological properties when considered simply as a finite collection of points. However, by turning a point cloud into a higher-dimensional simplicial complex, we can gain more information about its "shape." In PH, we take a point cloud and turn it into a series of simplicial complexes at different scales; we then compute the homology of each of these complexes and track homological features across scales. Intuitively, consider looking at a point cloud and filling in the areas between points that are close to each other to obtain a manifold. Changing our notion of what it means to be "close to each other" results in a collection of different manifolds, each of which approximates our original point cloud. Most simply, we can take "close" to mean within some Euclidean distance, and we can then progressively increase this distance. Imagine squinting at a point cloud until it blurs and takes on some shape; the harder you squint, the more the edges of the shape blur and expand. This approach is particularly useful because of its ability to encode geometric information using a scaling parameter. Although topological invariants are useful because of their mathematically rigorous meaning, our intuition about what it means for a data set to have a certain shape includes many concepts that cannot be captured up to homeomorphism. Consider the classical example of a coffee cup and a donut: their homology groups are indistinguishable, yet we may still be interested in identifying differences between them.

Persistent homology has been used in a large variety of problems in numerous disciplines [36]. Applications of PH have included studies in protein compressibility [50, 51], DNA structure [18], computer vision [13], granular and particulate systems [28, 37], a wealth of different topics in neuroscience [14, 22], and more. One aspect of PH that makes it very appealing is its robustness to noise: because one examines data at multiple scales simultaneously, conventional wisdom suggests that features that persist over a variety of scales should be the result of a signal (rather than of noise). However, in some data sets (as in the case of geographical data sets), several distance scales are represented in a single point cloud, making it difficult to find persistent features. More generally, for both spatial and nonspatial data, features with short persistence can convey important signals, as illustrated in [47] in an application to neuroscience. In these situations, it can be difficult to distinguish between (1) features that are real but appear only at specific scales and (2) noise.

Data sets that have interesting features at multiple scales are a particularly poor fit for constructions that use distances to turn point clouds into complexes. However, distance-based constructions—especially the Vietoris–Rips (VR) construction—are the most common choice for constructing simplicial complexes from point clouds because of their relatively fast computation times [36, 53]. Much of the recent literature on methods for constructing simplicial complexes has focused either on finding faster ways to build VR complexes or on building approximations to a VR complex using

less data and thereby reducing computation time [36]. Computing VR complexes or other distance-based complexes has been very effective for many applications [36], but distance-based complexes can sometimes lead to considerable difficulty in interpreting results, especially in applications where scaling is a major factor. To mitigate the effect of scaling, we propose the construction of simplicial complexes that are based on the network or contiguity properties (when they are available) of a data set, as this allows an interpretation of persistence that does not rely on a distance scale and which is thus easier to interpret for geographical data.

Our paper proceeds as follows. In section 2, we discuss our data set of votes in the 2016 presential election and give background information about the methods of PH. In section 3, we discuss several methods for the construction of simplicial complexes, including traditional distance-based constructions (VR and alpha complexes) and two new constructions that are based on the contiguity of geographical data. We also discuss the geometric differences between these methods and our intuition about how those differences affect our analysis. In section 4, we give some computational results that support our intuition and provide guidelines for when each construction is appropriate. In section 5, we discuss future directions for the computation of PH on 2D data. In the appendices, we give further background on simplicial homology and additional details about some of our computations and results.

2. Background.

2.1. Voting Data. Throughout our paper, we use data from the *LA Times* California 2016 Election Precinct Maps project [44]. This data set, which was compiled by the *Los Angeles Times* Data Visualization Team after the November 2016 elections, has precinct-level results for every statewide race in California. Specifically, it encompasses results for the presidential race, California's senatorial race, and 17 statewide propositions. The data covers all of California's 24626 precincts (which are organized into 58 counties); for each one, it includes the number of votes for each choice in each race, along with an associated SHAPEFILE and other metadata. While processing the raw data, QGIS [38] encountered an error with four counties—Butte, Santa Clara, Siskiyou, and Ventura—so we examine only the other 54 counties. We generate precinct maps for each county and classify precincts in the presidential race based on the margin of victory for each candidate. Precinct boundaries are very complicated, vary across elections, and may be "split" across political districts during redistricting. We aggregate precincts at the county level, as precincts are organized neatly at the county level and federal election results are tabulated at the precinct level. We show voting maps of two California counties in Figure 1. These voting maps include all of the precincts of a county, regardless of which candidate they favored, but our PH computations use voting maps that include only precincts that voted for the same candidate.

We are especially interested in examining the phenomenon in which a region (e.g., one precinct) votes differently from the areas that surround it (e.g., "an island of red voters in a sea of blue," or vice versa). We refer to these regions as voting "islands." Understanding this phenomenon gives one way of quantifying the voting patterns of counties: some counties have rather uniform voting patterns, whereas others may have clusters of communities that vote differently from their neighbors, potentially signaling the presence of urban areas, demographically distinct neighborhoods, or gerrymandering.

This application is particularly appropriate for analysis using PH because we can interpret regions with outlying voting preferences as holes. Additionally, computing

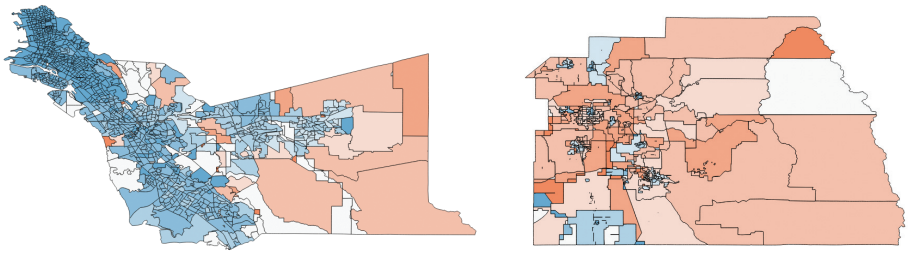


Fig. 1 *Voting maps of (left) Alameda County and (right) Tulare County. Red precincts voted predominantly for Donald Trump, and blue ones voted predominantly for Hillary Clinton. Darker shading in a precinct indicates a stronger majority for the winning candidate, so Trump won dark red precincts by a large margin and Clinton won dark blue precincts by a large margin. We use the color white for precincts with an equal number of votes for the two candidates.*

the homologies of these counties allows us to classify them based on their topological features. We can consider a county as a point cloud, where each precinct is a point with some additional data that is assigned to it (specifically, voting preferences and the geographical space that it occupies). For the remainder of this paper, we consider only votes for the candidates Hillary Clinton and Donald Trump in the presidential election. We use "red" to indicate a voting preference for Trump and "blue" to indicate a preference for Clinton, with darker colors signifying stronger voting preferences. We consider the voting preferences of each precinct.

2.2. Persistent Homology. We now give a more rigorous discussion of some of the intuitive descriptions of PH from section 1. Suppose that we have experimental data X_{observed}, from which we construct a sequence $X_0 \subseteq X_1 \subseteq \cdots \subseteq X_l$ of simplicial complexes of dimension d. In section 3, we will discuss several methods to construct such a sequence. We require that the sequence $\{X_i\}$ is increasing, such that it forms a filtered simplicial complex (which we sometimes call simply a "filtration"), and we call each X_i a subcomplex. The filtered simplicial complex, along with inclusion maps between subcomplexes and chain and boundary maps of each subcomplex, is called a "persistence complex." We examine the homology of each subcomplex, and we note that the inclusion map $X_i \hookrightarrow X_j$ induces a map $f_{i,j} : H_m(X_i) \to H_m(X_j)$ and that, by functoriality,

$$f_{k,j} \circ f_{i,k} = f_{i,j}\,. \tag{2.1}$$

DEFINITION 2.1. *Let $X_0 \subseteq X_1 \subseteq \cdots \subseteq X_l = X$ be a filtered simplicial complex. The* mth persistent homology *of X is the pair*

$$\left(\{H_m(X_i)\}_{0\le i\le l}\,, \{f_{i,j}\}_{0\le i\le j\le l}\right)\,,$$

where $f_{i,j} : H_m(X_i) \to H_m(X_j)$ for all $i \le j$ and m smaller than the dimension[2] of X are the maps that are induced by the action of the homology functor on the inclusion maps $X_i \hookrightarrow X_j$. We refer to the collection of all mth persistent homologies as the persistent homology (PH) *of X.*

The PH of a filtered simplicial complex encodes information about the maps between each subcomplex, thereby giving more information than the homologies of

[2]Note that m is not necessarily smaller than the dimension of X, as shown in [1], but this is a convenient simplification for many applications (including ours).

the individual subcomplexes. Each homology group with field coefficients $H_m(X_i)$ is a vector space whose generators correspond to holes in X_i, and the maps $f_{i,j}$ allow us to track these generators from $H_m(X_i)$ to $H_m(X_j)$. By picking a convenient basis for $H_m(X_i)$, which we are able to do by the Fundamental Theorem of Persistent Homology [54], we can construct a well-defined and unique collection of disjoint half-open intervals such that each generator $x \in H_p(X_i)$ corresponds to an interval $[b_x, d_x)$, with X_{b_x} denoting the subcomplex in which the generator (and its associated hole) first appears and X_{d_x} denoting the subcomplex in which the generator dies. More precisely, we say that $x \in H_p(X_{b_x})$, with $x \neq 0$, is born in X_{b_x} if it is not in the image of f_{b_x-1,b_x}; it dies in $H_p(X_{d_x})$ if $d_x > b_x$ is the smallest index for which $f_{b_x,d_x}(x) = 0$. If $f_{b_x,j}(x) \neq 0$ for all $j \in \{b_x + 1, \ldots, l\}$, then x lives forever and we associate the interval $[b_x, \infty)$ to it. For a more in-depth discussion of PH and other homological concepts, see Appendix A; for further material, see [19, 23, 36, 54].

The collection of half-open intervals is known as the "barcode" [19] of X, and we use it to visualize the mth PH. Generators with longer associated half-open intervals are more persistent. In general, one uses the persistence of features to distinguish signal from noise, but recent work indicates that persistence is not always readily interpretable in a meaningful way [10, 24, 47]. In our computations, we find that using traditional distance-based constructions on the *LA Times* voting data yields ambiguous results about the persistence of features. However, by constructing a persistence complex in an appropriate way (see sections 3.2 and 3.3), we obtain barcodes for each county in which persistence is a useful property for separating genuine features from noise.

We also draw attention to the distinction between the dimension of an embedding and the dimension of a topological object that we are studying, as we will be referencing both in the remainder of this paper. When we refer to the dimension of a simplicial complex, we mean the dimension of the highest-dimensional simplex in the simplicial complex. Similarly, the dimension of a homological feature refers to the dimension of the homology group in which it lives; that is, a feature in the mth PH has dimension m. By contrast, when we refer to 2D data sets, we mean that the data is embedded in a 2D ambient space. Therefore, a point set in 2D is a 0D object that lives in a 2D space. A feature in H_0 is a 0D object that we can visualize as a point that lives in a 2D space. A feature in H_1 is a one-dimensional (1D) object that we can visualize as a loop that lives in a 2D space. A precinct is a 2D object that is embedded in a 2D space using its latitude and longitude. We refer to data sets and geographical maps based on the dimensions of the spaces in which they live, and we refer to simplicial complexes, homology groups, and homological features based on the dimensions of the objects themselves.

3. Methods for Constructing Filtered Simplicial Complexes. In this section, we describe the various methods that we use for constructing simplicial complexes from the voting data. The geographical data comes in the form of SHAPEFILEs; it is a collection of polygons, rather than a point cloud. Although we do include computations based on existing constructions, which use point-cloud data, we also leverage the additional information inherent in the polygon form of geographical maps to suggest two new constructions that are better suited to our application.[3] In section 3.4, we explain why one should expect these new approaches to yield better results for

[3]In our work, we are concerned with space in a way that is conceptually different from the concerns of geographers. Our primary concern is the existence of holes and how they develop across a filtration. We do not track their precise locations and boundaries. It would be necessary to go beyond our topological notion of shape to do so.

geospatial data. We perform computations using both these new constructions and two traditional ones, and we compare their performance in section 4.

3.1. Distance-Based Constructions. We begin by reviewing common methods for constructing filtered simplicial complexes from point clouds. One of the most prevalent constructions is the Vietoris–Rips (VR) complex, which one constructs using the pairwise distances[4] between points in a point cloud [16, 49].

Let X be a data set[5] in the form of a point cloud. Given a real number $\epsilon \geq 0$, we define the VR complex $\mathrm{VR}_\epsilon(X)$ as follows:

$$\mathrm{VR}_\epsilon(X) = \{\sigma \subseteq X : \ \forall\, x, y \in \sigma\,,\ d(x,y) \leq \epsilon\}\,.$$

In this construction, we produce a "thickening" of a point cloud by replacing its points with balls of radius ϵ. If there are n points in X, the maximal possible VR complex is the $(n-1)$-simplex that consists of all points in X along with all of its subsimplices. By taking a collection $\{\epsilon_i\}$, with $0 = \epsilon_0 < \epsilon_1 < \epsilon_2 < \cdots < \epsilon_l$, and considering

$$X = \mathrm{VR}_{\epsilon_0}(X) \subseteq \mathrm{VR}_{\epsilon_1}(X) \subseteq \cdots \subseteq \mathrm{VR}_{\epsilon_l}(X)\,,$$

we obtain a filtered simplicial complex whose PH we can compute. It is straightforward to construct a VR complex because we only need to compute pairwise distances. Additionally, there are various fast algorithms for constructing it [53]. Unfortunately, for large point clouds, the worst-case VR complex has $2^n - 1$ simplices and dimension $n-1$. The largest county in our data set is Los Angeles County, which has almost 5000 precincts, resulting in a worst-case VR complex with about $2^{5000} - 1$ simplices. This is very problematic.

The large number of precincts in several counties makes it intractable to compute VR complexes for these counties. For county–candidate combinations[6] with at least 151 precincts, we instead compute alpha complexes. The alpha complex [17], which we denote by $\mathrm{A}_\epsilon(X)$, also relies on a distance parameter and is defined as follows. Let $\epsilon > 0$, and let $X_\epsilon = \bigcup_{x \in X} B(x, \epsilon)$. Additionally, let $(V_x)_{x \in X}$ be the Voronoi diagram of X. Consider the intersection $V_x \cap B(x, \epsilon)$ for each $x \in X$, and note that the collection of these sets covers X_ϵ. The alpha complex is

$$\mathrm{A}_\epsilon(X) = \left\{ \sigma \subseteq X : \ \forall\, x_i \in \sigma\,,\ \bigcap_i (V_{x_i} \cap B(x_i, \epsilon)) \neq \emptyset \right\}.$$

Because of the restriction of the ϵ-balls to the Voronoi diagram, the alpha complex restricts the dimension of the space in which X is embedded. In our case, because our data is embedded in $\mathbb{R}^2$, the alpha complex of a county has 2D simplices (i.e., faces) as its highest-dimensional simplices.

The two constructions above both require the input data to be in the form of a point cloud. Each precinct has an associated centroid, which we calculate according to (latitude, longitude) coordinates using a built-in feature of QGIS [38]. We di-

[4] In this paper, our measures of distance are metrics in the mathematical sense.

[5] In TDA, it is common to overload notation by denoting a data set by X, which one also uses to denote a simplicial complex. One can alternatively denote a data set by X_{observed}, as we did in section 2.2.

[6] When we construct a simplicial complex for a county, we use only the precincts in it in which the majority voted for a specified candidate (either Hillary Clinton or Donald Trump). We consider one candidate at a time so that precincts with a majority that voted for the other candidate never enter the simplicial complex. This allows us to detect regions as topological features for all of our choices of filtration parameter (which is ϵ for a VR complex).

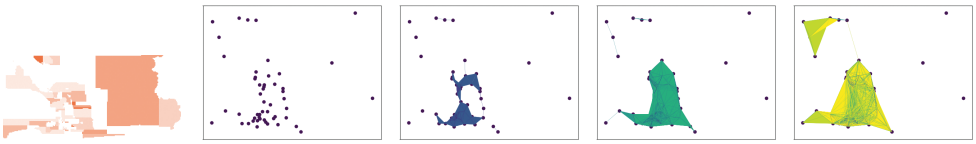


Fig. 2 *Illustration of a VR complex on the* LA Times *voting data.* (a) *The red precincts (in which more people voted for Donald Trump than for Hillary Clinton) of Imperial County in* 2016. *In panels* (b)–(e)*, we show the VR complex that approximates the county, with each successive image showing the VR complex as we increase* ϵ*. Observe that the contiguous region in the east of the county is not captured by this complex and that the western region includes a large number of* 1*-simplices and* 2*-simplices (see Appendix* A *for the definition of* k*-simplex), despite the fact that the county has relatively few precincts. Both phenomena occur because the eastern precincts are much larger, so their centroids are much farther apart than the small (but not necessarily contiguous) precincts in the west.*

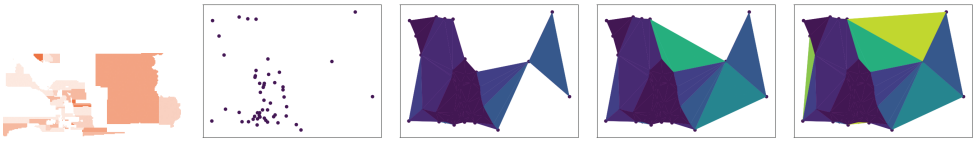


Fig. 3 *Illustration of an alpha complex on the* LA Times *voting data.* (a) *The red precincts (in which more people voted for Donald Trump than for Hillary Clinton) of Imperial County in* 2016. *In panels* (b)–(e)*, we show the alpha complex that approximates the county, with each successive image showing the complex as we increase* ϵ*. Observe that the filtered simplicial complex has much larger* 2*-simplices than those that we obtain for a VR complex (see Figure* 2*) and that (unlike in Figure* 2*) once the western region is covered by* 2*-simplices (which, as one can see in panel (c), occurs fairly early in the filtration), new* 2*-simplices do not arise as we increase* ϵ*. However, similar to what we observed in the VR complex, the resulting simplicial complex yields a simply-connected region in the west; this does not accurately reflect the underlying geographical map.*

rectly compute the Euclidean distance between the (latitude, longitude) coordinates of precinct centroids. Therefore, we do not make a choice of map projection.

It is common to employ VR complexes because it is relatively easy to construct them, they are intuitively appealing, they have important theoretical guarantees from the Nerve Theorem [27], and (perhaps most importantly) they have been implemented widely in existing software packages for computing PH. In general, ϵ-ball thickenings are a natural way to approach the problem of approximating a space from which one has only a sample of points. Points that are close to each other should be much more likely to be connected in the space than points that are far apart from each other, and thickenings also capture the intuition of blurring an image by squinting at it until the points start to merge. However, for our purposes, the point clouds that we construct do not bear a strong visual resemblance to the geographical maps from which we construct them, and the locations of holes in these maps are independent of distance. In Figures 2 and 3, we show visualizations of VR and alpha complexes for Imperial County. Note that we consider only the red precincts (or, alternatively, only the blue precincts); we make this simplification both to decrease computational complexity and to preserve an intuitive notion of closeness in voting patterns. In these visualizations, observe that the simplicial complexes do not visually resemble the underlying geographical maps and that they also appear to have rather different

topological properties. To address these issues, we propose two novel constructions of filtered simplicial complexes in the next two subsections.

3.2. Adjacency Complexes. Our first new type of construction of a filtered simplicial complex is based on the notion of a network adjacency. Consider a network whose vertices are precincts and whose edges are determined by "queen adjacency." We use the definition of queen adjacency from Geographic Information Systems (GIS); two precincts are queen adjacent if they touch at any two points, including corners. (This is reminiscent of the movement of queens in chess, but it is not quite the same.) This is distinct from "rook adjacency," in which two precincts are adjacent if they share a boundary. Intuitively, we can view such a network as one in which any path in it corresponds to an ability to physically walk from one precinct to another in a contiguous fashion. Some precincts are not simply connected or may even have multiple connected components. In section 4, we discuss the effects of such features on our results.

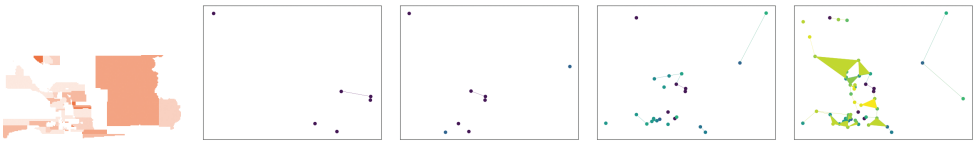
By considering different levels of voting preferences for Donald Trump or Hillary Clinton, we construct a nested sequence of networks. We define a value

$$\delta_{b,r}(p) = \frac{|V_b(p) - V_r(p)|}{|V_b(p) + V_r(p)|}, \tag{3.1}$$

where $V_b(p)$ is the number of blue (i.e., Clinton) votes in a precinct p and $V_r(p)$ is the number of red (i.e., Trump) votes in that precinct. For example, for a given county, consider all of its precincts that have a majority who voted for Hillary Clinton in 2016. For our first network, we consider only those precincts for which $\delta_{b,r}(p) \geq .95$. For the next network in the sequence, we take all precincts with $\delta_{b,r}(p) \geq .90$. We continue decreasing the strength of voting preference until we consider all precincts in which Clinton won, along with all of their adjacencies. At this stage, we stop and construct a filtered simplicial complex of 1D simplicial complexes. To incorporate faces, we add a 2-simplex between any three vertices that are all pairwise adjacent. This gives a 2D filtered simplicial complex, on which we can perform PH computations.

Using network adjacencies allows us to retain spatial information about our precincts that we lose when we consider only a point cloud of precinct centroids. In our application to voting data, our adjacency construction captures a notion of contiguity that is missing from the existing distance-based constructions. In Figure 4, we show an example of a filtered simplicial complex, which we construct using adjacencies, that approximates Imperial County. It has better contiguity properties than the VR and alpha complexes that we showed in Figures 2 and 3. However, this adjacency approach still requires us to associate a single point to each precinct polygon, rather than considering the entire area that it covers. This suggests another possible construction (based on level sets) of filtered simplicial complexes. We describe it in section 3.3.

Although we have framed our discussion of adjacency complexes in terms of map adjacencies and voting preferences, we can use any type of network adjacency to build such a complex. To construct an adjacency complex, we require some type of network (i.e., some type of relationship that gives adjacencies) and some function from the network to $\mathbb{R}$. In the present paper, we use voting preference to give us a function from the vertices of a network to $\mathbb{R}$. One can also consider functions from the edges of a network to $\mathbb{R}$, and one can then construct a filtration by adding edges (rather than vertices) at each step. Related constructions that explicitly build filtrations based on function values were explored in [7, 11, 26, 40, 41].



(a) (b) (c) (d) (e)

Fig. 4 *Illustration of an adjacency complex on the* LA Times *voting data.* (a) *The red precincts (in which more people voted for Donald Trump than for Hillary Clinton) of Imperial County in* 2016. *In panels* (b)–(e)*, we show an associated adjacency complex that approximates the county; we order the panels based on decreasing strength of preference for Trump. In panel* (b)*, we show the precincts with the strongest preference for Trump along with the adjacencies between them. As we move across the panels from left to right, we add more* 0*-simplices as we incorporate precincts with progressively weaker preferences for Trump. We color the simplices based on their strength of preference, with the darkest simplices having the strongest preference and the lightest ones having the weakest. For visual clarity, we scale panels* (b)–(e) *to use an entire rectangle. In panel* (e)*, we observe that the eastern region is simply connected and that the western region has many* 1*-simplices. However, the western region is not covered by* 2*-simplices, so it is not simply connected. Although the depicted filtered simplicial complex does not seem to visually resemble the geographical map in Figure* 4(a)*, its topological properties do appear to be similar.*

3.3. Level-Set Complexes. The second new method that we introduce is one that leverages the manifold nature of our data. For the previous methods (namely, the VR, alpha, and adjacency complexes), we were forced to make choices in how to assign precincts to points. For the VR and alpha constructions (i.e., the distance-based methods), we also had to make a choice of embedding into Euclidean space. We now introduce a complex that is based on level sets. To construct a level-set complex, we use polygon SHAPEFILEs as input and evolve them using the level-set method for the motion of interfaces. In this section, we give an overview of the filtered simplicial complex that we generate using the level-set method. The level-set method was introduced in [35]; we give an intuitive explanation of it in this section, and further details are available in [34].

Let M denote the 2D manifold that consists of the collection of all of a county's precincts that voted for the same candidate (regardless of the strength of the majority). We construct a sequence

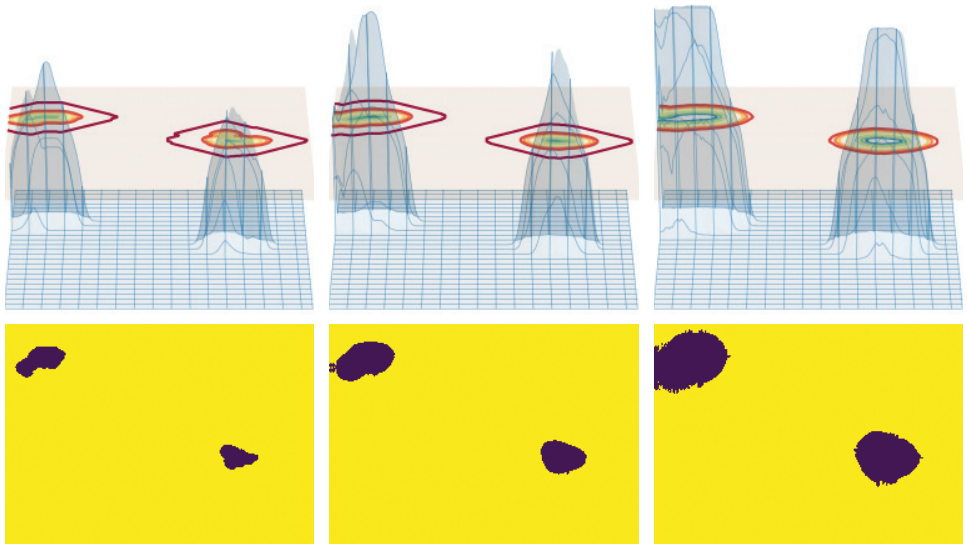
$$M = M_0 \subseteq M_1 \subseteq \cdots \subseteq M_l$$

of manifolds by considering the boundary Γ of M and performing front propagation on it so that the boundary expands outward, resulting in a larger manifold. We use the level-set method to efficiently solve the front-propagation problem. To do this, we evolve a function $\phi(\vec{x}, t) : \mathbb{R}^2 \times \mathbb{R} \to \mathbb{R}$ according to the level-set equation

$$\frac{\partial \phi}{\partial t} = v\,|\nabla \phi|\ , \tag{3.2}$$

where v is velocity.

By assigning the initial condition $\phi(\vec{x}, 0)$ to be the signed distance function from $\vec{x}$ to Γ, we see that the 0-level set of $\phi(\vec{x}, 0)$ is precisely the set of points that lie on Γ. When we evolve ϕ according to (3.2) up to time T, the resulting 0-level set of $\phi(\vec{x}, T)$ gives Γ_T, the expansion of Γ that results from movement normal to the boundary at velocity v. In this paper, we use $v = 1$.



(a) $T = 0$ (b) $T = 4$ (c) $T = 10$

Fig. 5 *Evolution of (top row) a level set on red precincts (in which more people voted for Donald Trump than for Hillary Clinton) in San Mateo County, with corresponding (bottom row) contour plots of ϕ, which satisfies the level-set equation* (3.2)*. As the time T increases, the graph of ϕ translates upward, so the* 0*-superlevel set expands. (The clipping of minimum and maximum values, which we do for computational efficiency, leads to flat areas at the minimum and maximum values of ϕ.)*

Fig. 6 *Contour plots of ϕ for the evolution of a level set on blue precincts (in which more people voted for Hillary Clinton than for Donald Trump) in San Mateo County.*

Intuitively, in terms of our geographical map, we can visualize the graph of $\phi(\vec{x}, 0)$ as a mountain (or multiple mountains, if there is more than one connected component), with the boundary of the map at sea level, the interior of the map above sea level, and the complement of the map below sea level. The set M_0 is the set of points $\vec{x}$ that are at or above sea level. As we evolve ϕ, we move the entire mountain upward, which increases the amount of land that is above sea level. The new region that is at or above sea level is our expanded manifold M_T. In Figure 5, we show the evolution of the 0-superlevel set (i.e., all points $\vec{x}$ such that $\phi(\vec{x}, t) \geq 0$) as T increases. We also show the graph of ϕ to help visualize the corresponding evolution of the level-set equation (3.2).

In Figure 6, we show a sequence of manifolds that we obtain by evolving a level set on blue precincts in San Mateo County. The original geographical map has holes

of various sizes, and the amount of time that it takes for a given hole to disappear is longer for larger holes.

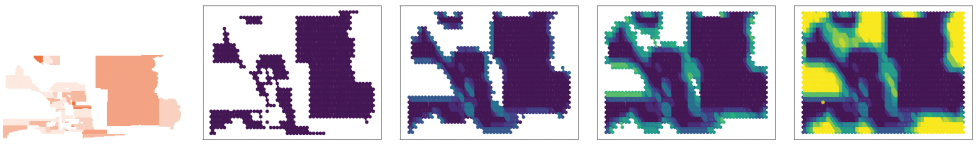
To turn this sequence into a filtered simplicial complex, we choose a triangulation of the plane and impose each M_T over this triangulation in the following manner. In our triangulation, (1) every fifth pixel is a vertex and (2) each vertex is connected to its four neighbors in the cardinal directions, as well as to its northwest and southeast neighbors. Other triangulation choices are also viable, but ours is computationally convenient (because it limits the number of vertices) and is easy to visualize. If all vertices of a 2-simplex lie within M_T, we add that simplex and all of its subsimplices to the corresponding simplicial complex X_T. This yields the filtered simplicial complex

$$X_0 \subseteq X_1 \subseteq \cdots \subseteq X_l \,.$$

We evolve until a time T that is sufficiently large for all holes to close. (The geographical maps are in a bounded subset of $\mathbb{R}^2$, so such a time is guaranteed to exist.) For more implementation details, see Appendix B.2.

The greatest strength of our level-set approach to constructing a filtered simplicial complex is that it gives an explicit triangulation of a geographical map that does not depend on how we assign precincts to points. The simplicial complexes that we build using the level-set method thus embed nicely into the plane, and they more closely resemble the underlying geographical maps from which we start than the complexes from the other methods that we examined. Moreover, persistence is nicely interpretable for the level-set approach. Any hole that exists in the geographical map also exists in the initial simplicial complex (as long as the hole is not finer than one's triangulation of $\mathbb{R}^2$), so every hole is a feature that is born at time 0. The persistence of the feature indicates the distance scales on which it exists. We can thereby distinguish between genuine short-persistence features and short-persistence noise from the evolution, because short-persistence noise does not appear until later times in the level-set evolution. (An example of this occurs in Figure 6, where a bay on the eastern side of the map is not a closed loop in the leftmost image, but it is closed in the next image because the opening of the bay is smaller than the bay itself.) Furthermore, although the level-set complex still suffers from the sensitivity to scale of other distance-based constructions, it does not require us to make a scaling choice, as is necessary for existing distance-based constructions. Both very large and very small holes are captured immediately, because the connectedness of a simplex does not rely on the distance between precinct centroids. In Figure 7, we show a level-set simplicial complex for the voting map of red precincts of Imperial County.

3.4. Comparing the Simplicial-Complex Constructions. In the previous subsections, we briefly discussed some of the ideas that we intend to capture with the different constructions of filtered simplicial complexes. We now give more detail about why these ideas are particularly useful for applications to geospatial data. PH on point clouds is based largely on the idea that the distance between points indicates something meaningful about the similarity or connectness of their associated regions. Under this assumption, points that are close together have a fundamentally different relationship to each other than points that are far apart. Consequently, features that occur at small distance scales should not represent the same patterns as features that occur at large scales. However, in our case (and in other applications to geospatial data), we observe two problems: (1) voting islands occur at a variety of distance scales; and (2) physical distance does not correspond to geographical connectedness. More generally, spatial applications for which the information of interest is not en-



(a) (b) (c) (d) (e)

Fig. 7 *Illustration of a level-set complex on the* LA Times *voting data.* (a) *The red precincts of Imperial County in* 2016. *In panels* (b)–(e)*, we show the level-set complex that is associated with the voting map of red precincts of Imperial County. We order it according to the number of time steps in the level-set evolution. Observe in panel* (b) *that the complex immediately resembles the original voting map and that small holes fill in faster than large ones. Given enough time steps, the level set will evolve to cover the entire bounding box that we show in the figure.*

coded in distances may suffer from both issues. We refer to the first issue as "scaling" and to the second issue as "contiguity."

In sections 3.4.1 and 3.4.2, we discuss why existing PH constructions struggle with scaling and contiguity, which of our methods address them, and how they do so. In Table 1, we summarize the methods and their performance. One potential solution to the problem of physical distance being unrepresentative of geographical connectedness is to replace it with some other distance and to perform PH using the new distance as the filtration parameter. Unfortunately, this is an undesirable solution for many applications to spatial data. Although the Euclidean distance between points in a data set may not encode the features of interest, the embeddedness of the data into space is often relevant. Changing the notion of distance may remove important information from an embedding, or it may force one to make a choice about how to combine multiple notions of distance. By contrast, our new methods for PH allow us to incorporate the spatial embedding of data without reducing that embedding to a set of pairwise distances between points, while also potentially avoiding the scaling and contiguity issues that arise from distance-based constructions.

Table 1 *Comparison of various methods of constructing filtered simplicial complexes based on whether they address scaling and contiguity problems.*

Issue	VR	Alpha	Adjacency	Level-set
Scaling	✗	✗	✓	✗
Contiguity	✗	✗	✓	✓

3.4.1. Scaling. When associating precincts to point clouds, the physical distance between precincts is based mostly on the extent to which the area is urban or rural. Accordingly, distance constructions result in very few persistent features. In rural areas, the sparse connections between adjacent precincts can cause one to miss voting islands because of missing edges. Additionally, the dense connectivity of urban areas at large scales can cause one to miss small voting islands because 2-simplices are automatically filled in. We thus see that many meaningful features (e.g., a single red island in an urban community) are not persistent. Even worse, the most persistent features give information about whether there are densely populated areas that surround relatively sparsely populated ones, but they give little meaningful information about the underlying political inclinations of the populations in those regions. These

results counter the conventional wisdom about PH that the strongest signals should come from the most persistent features and that short-persistence features are likely to be the result of noise.

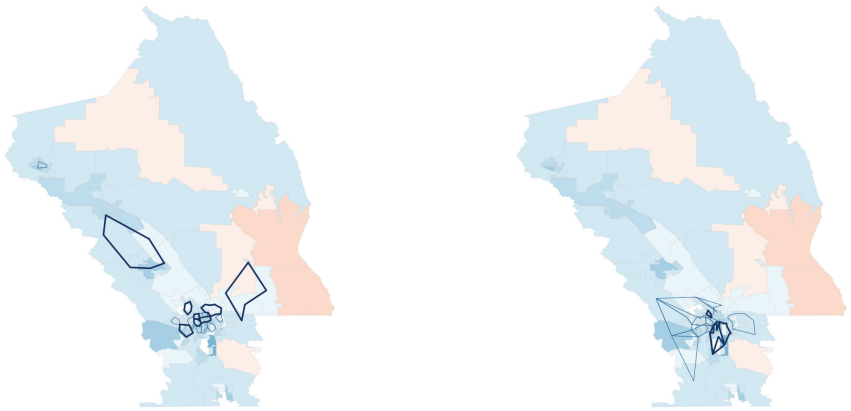
This leaves us with two possibilities: either (1) we evaluate the features that result from PH using criteria that do not depend solely on examining the most persistent features; or (2) we must find other ways of constructing filtered simplicial complexes, such that persistence becomes a meaningful quantity to compute for the problem of interest. There exists work on the former approach [2, 8, 9, 29, 39, 52], and our work complements this prior research by adopting the latter approach. In our adjacency construction, by letting the filtration parameter be the strength of voting preference rather than distance, we are able to interpret persistence as a measure of the difference between the preferences of the population in a "hole" and the preferences of the populations in the areas that surround it. That is, more persistent features represent holes with voting results that are very different from those of their neighboring regions. Consequently, the most persistent features are exactly the most meaningful ones, as they indicate which regions are the most extreme political islands (i.e., with voting preferences that are most different from the areas that surround them).

3.4.2. Contiguity. For our PH computations to be meaningful, we want the simplicial complexes that we build to approximate our data as closely as possible. For the VR and alpha constructions, we assumed that precincts (i.e., points) are connected to each other as long as their centroids are close enough. In practice, whether or not two precincts are adjacent has little to do with the distance between them. In urban areas, precincts that are very close to each other may have other precincts sandwiched between them, such that they are not connected. In rural areas, by contrast, precincts whose centroids are very far apart from each other may in fact be contiguous. Both the adjacency and level-set constructions address this issue.

In our adjacency construction, we define the adjacency matrix of a network based on whether or not two precincts share a border. As a result, all of the 1-simplices in our filtered simplicial complex come directly from physical contiguity. In the level-set construction, because our input data comes in the form of a manifold, both the 1-simplices and the 2-simplices reflect the physical contiguity of the original geographical maps. Both constructions allow us to build simplicial complexes that seem to approximate the data better than traditional distance-based constructions. See our illustration in Figure 8. It may be possible to improve a distance-based construction by using the minimum distance between points in a precinct, rather than the distance between centroids (or between other representative points). However, the computation of these minimum distances takes sufficiently long that we do not expect it to be a practical solution.

4. Computational Results. In this section, we summarize our computational results. For the construction of the VR and alpha complexes, we use the PYTHON package GUDHI [21, 32, 42, 43]. For the computation of the PHs and their generators, we modify PHAT [6], which is a `C++` package for the fast computation of barcodes. We implement the adjacency and level-set constructions by adapting the fast incremental VR algorithm of [53]. For details about our implementation and links to code, see Appendix B.

4.1. Sizes and Computation Times. The construction of simplicial complexes can be very slow, as one must check all possible simplices to determine whether they are present. The number of simplices grows as n^d, where n is the number of vertices



(a) Vietoris–Rips complex (b) Adjacency complex

Fig. 8 *Napa County, with the generators of features in H_1 marked as cycles in dark blue. We refer to this type of visualization, which we discuss in section* 4.2*, as a "feature map." In* (a)*, the VR complex has at least one "loop" in the eastern part of Napa County that is not contiguous, because it is composed of several small precincts whose union is not connected. In* (b)*, the adjacency construction captures several loops, each of which has generators whose union forms a contiguous region.*

and d is the maximum simplex dimension that one is considering. Consequently, methods that build smaller simplicial complexes tend to be faster. In Table 2, we compare the number of simplices in the simplicial complexes that we construct using the various methods. To keep computation times tractable, we compute VR complexes only for counties with at most 150 precincts that voted for a certain candidate. If 151 or more precincts voted for the same candidate, we instead compute alpha complexes.

From Table 2, we see that the adjacency and level-set complexes do not scale in size as rapidly as the VR complexes. This arises from how we construct these complexes. In adjacency complexes, the number of neighbors tends to be almost constant for any number of precincts, as there are practical bounds on the number of precincts that can border another precinct. The beneficent scaling of the level-set complexes with respect to the number of precincts arises from our specific choices of how we construct them. Because we take each vertex of a simplicial complex to be a point on a triangular grid, it has at most six neighboring vertices (one for each of its cardinal directions, as well as one to its upper left and one to its lower right), and it can thus be a member of at most six 2-simplices. One can make different choices of triangular grids—in our case, we simply added a northwest/southeast diagonal to each square in a square grid—and the number of neighbors is $O(1)$, as long as the grid is composed of triangles that have roughly the same size and shape (as is true for many grids). However, even when the number of precincts is rather small, a level-set complex can still be rather large. Even when there are relatively few precincts, if those precincts constitute a large enough portion of a voting map, they will include many grid points and hence many vertices. In practice, we obtain a relatively large number of the possible 2-simplices in our level-set complexes because our voting maps have large contiguous regions.

In Table 3, we compare the computation times for the construction of simplicial complexes and computation of PH for several counties. We include a range of counties

Table 2 *Sizes (i.e., number of simplices) of the filtered simplicial complexes. We first partition each county into precincts that voted for Clinton (C) and precincts that voted for Trump (T). We do not consider precincts that did not favor one of the two candidates. We then compute VR (or alpha), adjacency, and level-set complexes for each of these sets of precincts. (We compute VR complexes for counties with at most* 150 *precincts that voted for a given candidate and alpha complexes for counties with* 151 *or more such precincts.)*

County	# Precincts	VR		Alpha		Adjacency		Level-set	
		C	T	C	T	C	T	C	T
Alameda	1156	–	1967	5843	–	5755	70	3327	3578
Alpine	5	2	1	–	–	11	1	11962	1505
Amador	30	3	884	–	–	2	168	46	3979
Calaveras	29	8	641	–	–	6	92	1897	5195
Colusa	17	19	74	–	–	10	46	1665	5329
Contra Costa	711	–	3551	3561	–	3240	126	4135	3215
Del Norte	18	5	204	–	–	4	61	3584	6385
El Dorado	196	2397	89301	–	–	136	1123	782	4965
Fresno	592	–	–	1825	1431	1540	1192	2031	4788
Glenn	34	8	1152	–	–	4	156	329	5247
Humboldt	127	45998	680	–	–	504	119	15211	7323
Imperial	179	32496	6320	–	–	313	129	4375	6223
Inyo	25	33	216	–	–	14	51	4169	2242
Kern	642	–	–	1125	2119	928	2083	1429	5033
Kings	183	6305	69786	–	–	155	599	4849	7338
Lake	70	2279	779	–	–	99	73	4468	11275
Lassen	51	1	5920	–	–	1	250	193	11439
Los Angeles	4988	–	–	26551	1747	27705	1067	8587	6686
Madera	67	927	1947	–	–	103	132	925	5139
Marin	182	–	3	1037	–	1074	3	7893	621
Mariposa	25	5	401	–	–	7	91	2241	4485
Mendocino	250	–	692	1115	–	946	51	11901	1400
Merced	268	139832	54664	–	–	546	435	2213	6999
Modoc	21	0	399	–	–	0	94	0	7995
Mono	12	41	5	–	–	35	4	2499	3452
Monterey	467	–	13887	2297	–	1059	135	3597	4370
Napa	170	170093	56	–	–	858	15	10414	4968
Nevada	82	2569	2242	–	–	230	201	2946	2495
Orange	1668	–	–	5391	3811	4373	2632	5719	6513
Placer	363	5085	–	–	1685	141	1902	1210	3354
Plumas	30	8	618	–	–	6	102	723	6609
Riverside	1126	–	–	2291	2833	1602	2081	2231	2617
Sacramento	1267	–	–	2935	1275	15893	3459	4263	6748
San Benito	54	1804	276	–	–	152	67	699	6357
San Bernardino	2654	–	–	6206	4953	3658	2465	1700	6487
San Diego	2111	–	–	8007	3329	7480	2977	4680	7447
San Francisco	599	–	0	3499	–	3728	0	6826	0
San Joaquin	500	–	–	1659	1091	1490	902	7115	13419
San Luis Obispo	161	24600	14301	–	–	307	351	1319	4321
San Mateo	467	–	8	2573	–	2457	4	13865	782
Santa Barbara	250	–	11950	971	–	835	287	3488	6542
Santa Cruz	267	–	28	1307	–	1301	7	4737	295
Shasta	121	3	75177	–	–	2	745	941	5973
Sierra	22	3	233	–	–	2	57	417	3677
Solano	258	125438	13096	–	–	727	338	4589	5891
Sonoma	491	–	886	2355	–	2204	32	6031	899
Stanislaus	218	45984	51289	–	–	420	493	2536	6219
Sutter	52	62	3558	–	–	23	266	588	10689
Tehama	46	0	4261	–	–	0	241	0	5007
Trinity	25	25	243	–	–	12	60	5485	10344
Tulare	250	13096	–	–	921	235	1032	2242	7763
Tuolomne	68	18	10605	–	–	6	334	3380	3997
Yolo	129	49597	486	–	–	559	70	5089	4597
Yuba	46	5	3422	–	–	3	199	1909	8521

Table 3 *Computation times of selected county–candidate pairs, where we show the fastest method for each example in bold. We show computation times for building filtered simplicial complexes in the "Complex" columns and sums of the computation times for producing the H_0 and H_1 barcodes in the "PH" columns. We include results for several large counties (i.e., ones with many precincts) to show that our methods are substantially faster than computing VR complexes. For small counties, such as Imperial and Tulare, the improvement in computation time is less noticeable. Computing level-set complexes is not substantially faster for small counties than for large counties, as the number of simplices in a level-set complex is based on the resolution of the geographical map, rather than on the number of precincts.*

County	VR		Alpha		Adjacency		Level-set	
	Complex	PH	Complex	PH	Complex	PH	Complex	PH
El Dorado (T)	182.361 s	0.783 s	–	–	**0.090 s**	**0.008 s**	5.623 s	0.011 s
Imperial (C)	20.680 s	0.154 s	–	–	**0.0137 s**	0.009 s	9.29 s	**0.007 s**
Los Angeles (C)	–	–	15.479 s	0.065 s	39.264 s	0.069 s	**9.963 s**	**0.045 s**
Merced (C)	488.823 s	0.669 s	–	–	**0.0217 s**	**0.009 s**	6.677 s	0.025 s
Napa (C)	654.803 s	0.980 s	–	–	**0.048 s**	**0.010 s**	8.309 s	0.042 s
San Bernardino (C)	–	–	1.765 s	0.032 s	**0.691 s**	0.030 s	4.385 s	**0.019 s**
Tulare (T)	–	–	**0.0515 s**	0.016 s	0.129 s	0.015 s	5.180 s	**0.006 s**

to compare the speed of each method for both large and small counties. For a complete table of all computation times for building simplicial complexes, see Appendix D. From Table 3, we see that our constructions of the adjacency and level-set complexes are significantly faster than the construction of VR complexes, even for relatively small counties like El Dorado (which has only 196 precincts). This is especially striking in light of the fact that we have not optimized our implementations of our new methods to make them as fast as possible. For the level-set complexes, it is possible to make the computations much faster using existing implementations of level-set dynamics [20]. One can also leverage the wealth of research on level-set methods to evolve between manifolds in other ways, potentially leading to further methodological developments.

We also see that our computations are only slightly slower than or have similar computation times to those for the construction of alpha complexes. These speed gains are due largely to the significantly smaller number of simplices that we need for our new constructions of filtered simplicial complexes. In 2D geospatial applications, the number of simplices is smaller than for other applications because of constraints from our starting geographical maps. In other applications, one does not typically benefit from such a built-in limitation in numbers. (For example, networks in general do not satisfy the property that the degrees of the vertices are roughly constant for any total number of vertices [33].) However, the analysis of other spatial systems (e.g., granular materials, transportation networks, and various examples in biology) will also benefit from these ideas.

4.2. Barcodes and Feature Maps. In this section, we illustrate the differences between the results of the various methods for constructing filtered simplicial complexes. We generate two types of visualizations for our PH results. The first takes the form of barcodes (for both H_0 and H_1), where we display each feature as a bar whose length corresponds to its persistence. The second is a map visualization (see Figure 8 for examples), where we mark the location of each feature that we find by computing PH by drawing a cycle that passes through all of its generators. We call this visualization a "feature map," and we use the term "generator precincts" for the generators of a homological feature (see Figure 8). These generators are not necessarily unique, and we select our generators by using a standard PH algorithm (specifically, by using

the row-reduced boundary matrix) [54]. Although the nonuniqueness of generators is a potential concern, in our study, any set of generators results in some group of precincts that surround a voting island. We color each cycle according to the political party of the associated candidate. For example, if we find a blue hole in a sea of red, we draw a red cycle. To help illustrate the various interpretations of persistence, we highlight "long-persistence" features in H_1. Specifically, if an element $x \in H_1$ has the persistence interval $[\text{birth}(x), \text{death}(x))$, we compute

$$l = \text{death}(x) - \text{birth}(x)\,. \tag{4.1}$$

If $l \geq \text{int}(0.75 \max_{y \in H_1}[\text{death}(y) - \text{birth}(y)])$ (where the floor function $\text{int}(z)$ denotes the integer part of z), we consider x to be a long-persistence feature. We color long-persistence features in dark red or dark blue, depending on the political party of the candidate, and we color other features in lighter shades of red or blue. We also color long-persistence features with darker bars in the barcodes. We discuss results for two counties in this section, and we give additional examples in Appendix C.

4.2.1. Example 1: Red Precincts in Tulare County. We compare the barcodes and feature maps that we obtain by computing the PHs of the alpha, adjacency, and level-set complexes that we generate from red precincts (i.e., those with a majority who voted for Donald Trump) in Tulare County (see Figure 9). Tulare County is relatively small, with only 250 precincts. The county is predominantly rural, although it has a few small urban areas toward its western side. Tulare is a strongly Republican county, and only a very small proportion of its precincts voted blue (i.e., for Hillary Clinton) in the 2016 election. In the voting map of Tulare in Figure 9, we observe several islands of blue voters that we hope to be able to detect using PH. To detect these blue islands, we consider the topological structure of the simplicial complexes that we construct using the voting map of red precincts; we seek to find holes in these complexes. In Figure 10, we show the results of the three different constructions.

For the alpha complex, the H_1 barcodes indicate that most features do not have long persistence. The loops that surround the blue holes are light red, indicating that they are not long-persistence features. The single long-persistence feature corresponds to a loop in the northwest part of the voting map; it connects three precincts whose union is disconnected, and it does not surround any blue areas. It thus exhibits both the scaling and the contiguity problems that we discussed in section 3.4. The spacing between these three precincts is such that the pairwise distances between them are similar, but these distances are larger than the precincts themselves, causing them to form a loop even though none of them is adjacent to any of the other precincts on the map. Because this loop corresponds to the only long-persistence bar in the barcode, it is difficult to use persistence to distinguish fake loops like this one from real loops in the western region of the map. Overall, the alpha complex does detect some voting islands, but it misses a few of them that are located slightly southeast of the central area; it also detects many features that are not real.

In contrast to our observations with the alpha complex, generator precincts in the adjacency complex mostly form contiguous loops. Because of our construction, edges cannot occur between the centroids of precincts that are not adjacent to each other. However, the resulting feature map does have a few generator precincts that appear to be disconnected from their neighboring generator precincts, largely because the precincts themselves have complicated shapes. For example, some of the precincts are not simply connected and others have multiple connected components. Some work in mathematical gerrymandering has focused on tackling some of these issues by

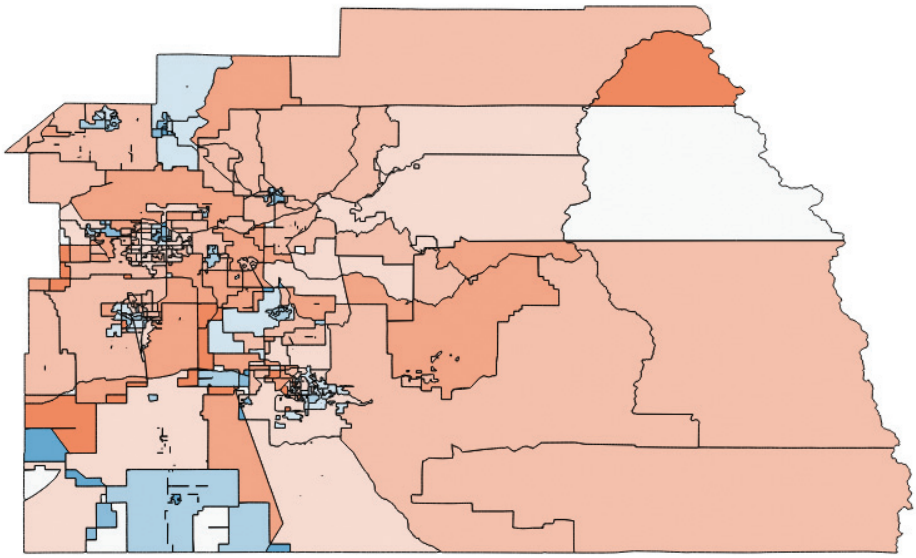
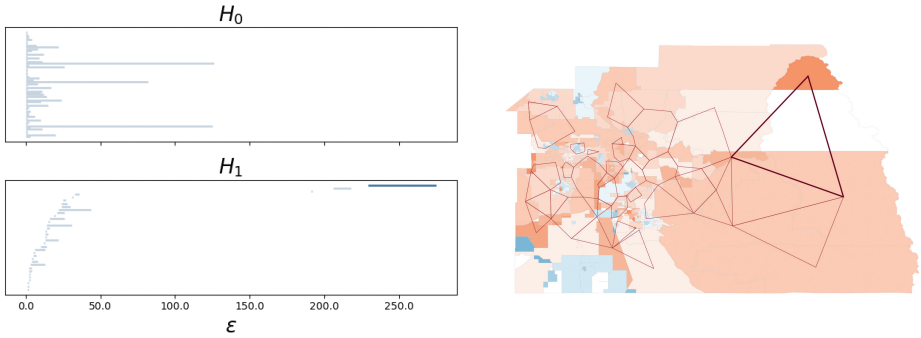


Fig. 9 *Tulare County, which we color based on voting results in the* 2016 *presidental election. Red precincts have a majority who voted for Trump, and blue precincts have a majority who voted for Clinton. Darker colors indicate stronger majorities.*

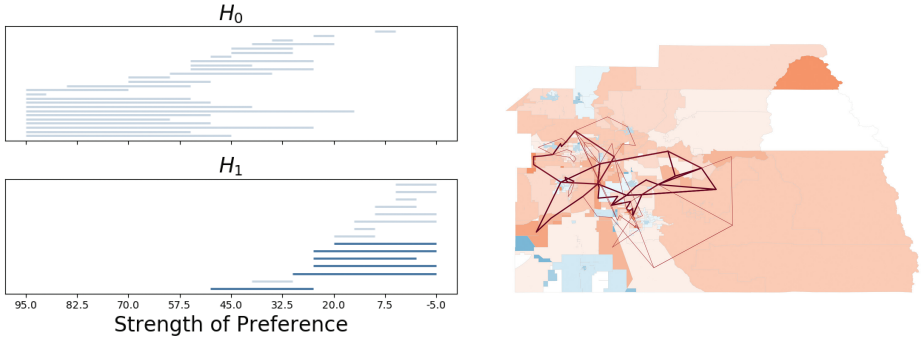
quantifying the idea that electoral districts ought to be "compact" [5, 15]. However, for the most part, the generator precincts surround blue and light red holes in the voting map. Additionally, there are fewer bars in the H_1 barcode in the adjacency complex than in the alpha complex, and more of the bars in the adjacency complex correspond to long-persistence features. The longest bar corresponds to the large hole in the map's center that includes both blue and light red precincts. Although these light red precincts do eventually join the filtered simplicial complex, the blue precincts in the center ensure that this hole never closes. Keeping in mind that the generators of a feature are not necessarily unique, the particular algorithm that we use to compute PH selects the group of darker red precincts that surround that light red area. We also observe several small light red holes (which correspond to the bars in the barcode that are born early) and several blue holes (which correspond predominantly to the bars in the barcode that are born late). The adjacency complex is able to locate most of the blue areas of the voting map,[7] and it has little noise. All of the aforementioned long-persistence features are genuine features, and we therefore see that we do a better job of distinguishing signal from noise for Tulare County with the adjacency complex than with the alpha complex.

Finally, we examine the barcodes and feature map in the level-set complex that we construct using the red precincts of Tulare County. The H_1 barcode has several features—some that have long persistence and some that do not—that start at time 0, and there is also one feature that starts at a much later step of the filtration. The bars that start at time 0 correspond to some of the holes in the western area of the voting map. We detect only six of these holes, as some of them occur on size scales that are too small for us to capture in our level-set complex because of our choice of

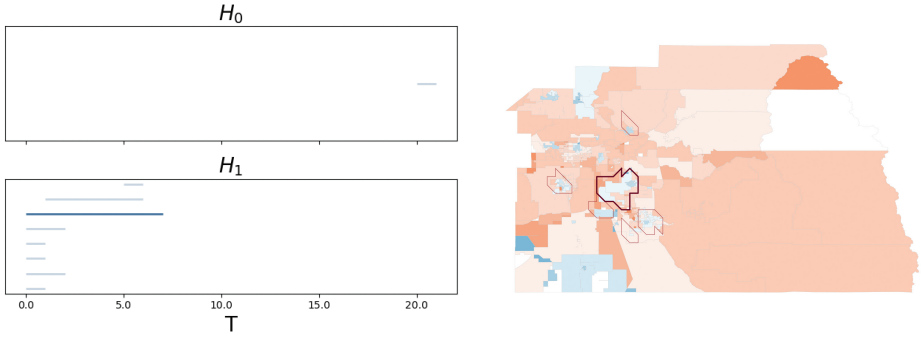
[7]The exceptions are a few areas near the edges of the county. There is no hope of detecting several of these as holes, because they lie on the county's borders and thus cannot be surrounded.



(a) Alpha complex



(b) Adjacency complex



(c) Level-set complex

Fig. 10 *Barcodes and feature maps for red precincts in Tulare County. We mark long-persistence features using darker loops with thicker widths. In the barcode of an adjacency complex, a bar that extends to* -5.0 *indicates a feature that lives past* 0.0*. In a level-set complex, the bars that correspond to loops start at* $T = 0.0$*.*

grid resolution in its construction. We also observe that the persistence of a bar is positively correlated with the size of its associated hole. The single long-persistence feature corresponds to the largest blue hole. Overall, the level-set complex captures most of the blue areas in the voting map and avoids most of the noise, although it does fail to detect some of the smaller regions.

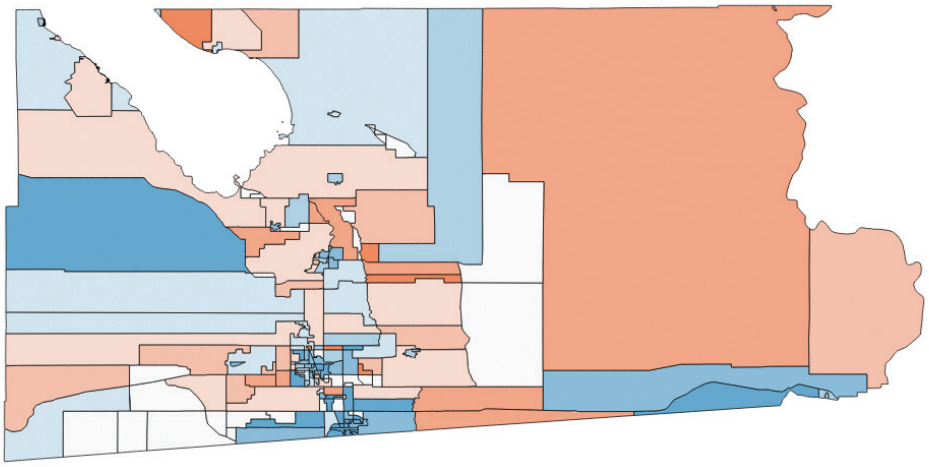


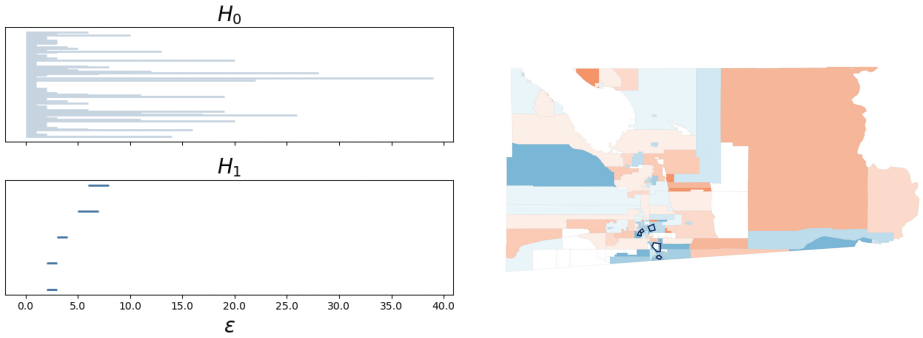
Fig. 11 *Imperial County, which we color based on presidential voting. Red precincts have a majority who voted for Trump, and blue precincts have a majority who voted for Clinton. Darker colors indicate stronger majorities.*

4.2.2. Example 2: Blue Precincts in Imperial County. We now construct VR, adjacency, and level-set complexes using Imperial County's blue precincts, which we show in the map in Figure 11. We show the barcodes and feature maps of these simplical complexes in Figure 12. For visualizations of the various simplicial complexes that we built from Imperial County's red precincts, see Figures 2–4 and 7 in section 3. In contrast to Tulare County, it is not immediately evident where there may be holes in the voting map of Imperial County. There do seem to be a few very small red precincts that are surrounded by blue precincts, so we hope to be able to capture some of those. Overall, however, we expect to observe relatively few features.

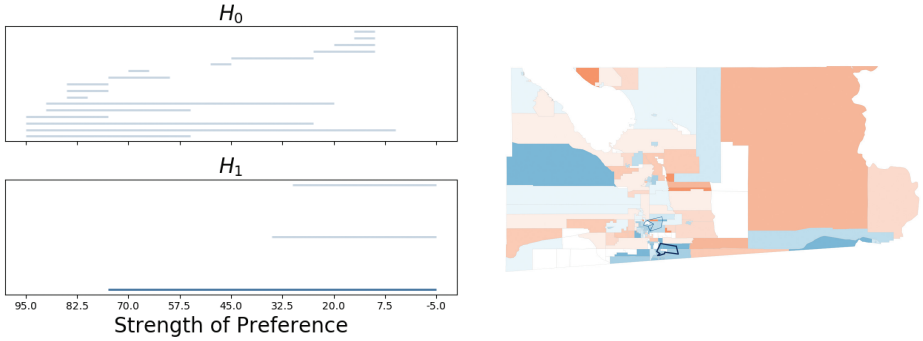
Examining the results from the various constructions, we observe that the VR complex picks up some noise and that only one of the features appears to surround a hole. Instead of finding voting islands, it finds several areas in which the blue precincts are tightly clustered, but they do not seem to surround any red precincts. Furthermore, all of the features have similar persistences and they are all categorized as long-persistence features. Because so many of the precincts in Imperial County are small, it is unsurprising that all of the features have similar persistences, so it is difficult to distinguish signal from noise. Moreover, as we will see, our findings from the adjacency complex and the level-set complex imply that the VR complex is not picking up any real holes in the voting map.

The adjacency complex picks up one long-persistence feature and two other features. On inspection, these appear to be small white or light blue holes that are surrounded by darker blue districts. All three of the holes appear to be around either white precincts or red precincts, and the single long-persistence feature is generated by relatively dark blue precincts. The long-persistence feature also seems to be the only feature that corresponds to a feature from the VR construction.

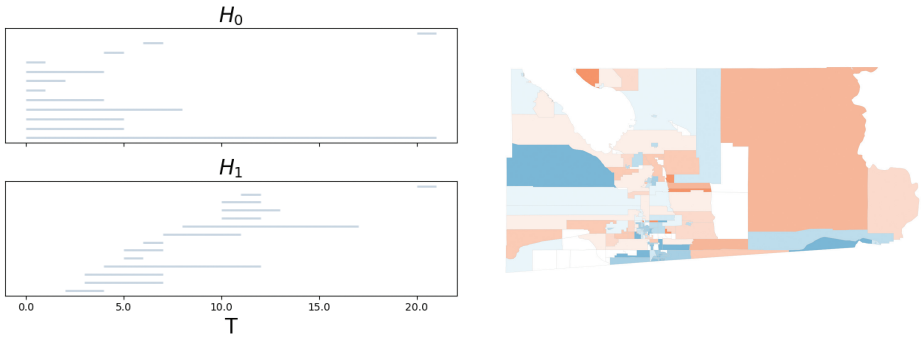
In contrast to the adjacency and VR complexes, which include very few features, the level-set complex picks up a large number of 1D features, but none of them starts at time 0. This occurs because the separate connected components eventually combine as the level set evolves to yield a larger number of holes than the number that exist in the voting map. This illustrates one of the problems with level-set complexes:



(a) VR complex



(b) Adjacency complex



(c) Level-set complex

Fig. 12 *Barcodes and feature maps for blue precincts in Imperial County. The VR complex results in several false "features"; the adjacency complex detects two white holes and one red hole; and the level-set complex does not detect any holes, because there do not exist sufficiently large white or red holes.*

as time passes, a level-set complex tends to become progressively more connected, which can create some false features when the simplicial complex starts with many connected components. However, if one considers only those features that exist at time 0, one can distinguish between genuine and false features. Most of the counties have relatively homogeneous voting patterns, with small voting islands, so few of the California counties yield these false features in practice. Additionally, including only features that begin at time 0 results in reasonable feature maps.

4.3. Comparison of Our Results to "Ground Truth." We conclude our analysis with some discussion of the accuracy with which we are able to use long-persistence features to identify genuine voting patterns in the California counties. In Table 4, we show the proportion of long-persistence features that indicate an actual hole, as determined by human eyes. We highlight the most successful method for each county in bold. We see that our adjacency and level-set constructions outperform the VR and alpha constructions. This indicates that our methods are less likely than the traditional distance-based approaches to detect noise as significant features in these examples.

5. Conclusions. Analyzing persistent homology in geospatial data can often lead to results that are difficult to interpret because of the heterogeneity of distance scales in such data. A particularly difficult aspect of barcodes is that bars with similar lengths may represent either signal or noise, in stark contrast to the conventional wisdom that the features that persist the longest also carry the most meaningful information about a data set. The difficulty in identifying interesting features from a barcode can make PH a challenging tool to apply effectively, even in applications in which topological holes seem like something that is appropriate to compute to gain insights into a problem. Therefore, it is extremely important to further explore the issue of signal versus noise in PH, especially for multiscale problems. In the present paper, we introduced two new methods for constructing a filtered simplicial complex that approximates a geographical map and we discussed the effects that different types of complexes have on the resulting PH. Our constructions attempt to address the difficulties of applying topological data analysis (TDA) to data that is not well-represented by traditional point clouds. In our application to voting data, our adjacency complex allowed us to incorporate data about relationships other than distance between points, while preserving the embedding of geographical maps in space and avoiding the need to make specific choices of distance transformations for different counties. Our level-set complex allowed us to compute, in a way that is inexpensive relative to other PH computations, complexes that are very similar in intuition to traditional VR complexes without having to start from a point cloud.

Both the adjacency and level-set complexes do a better job than traditional distance-based complexes of encoding information about the contiguity of voting maps, thereby making it possible to interpret differences in the distance scales of features. An adjacency complex does this by ignoring distance entirely in its construction. In a level-set complex, the persistence of the features that we detect encodes the distance scales of those features, but with fewer concerns than in VR or alpha complexes about noise due to precinct sizes. Consequently, the barcodes of the adjacency and level-set complexes are more interpretable than those of traditional PH constructions for our geospatial data, allowing us to better understand the topology of voting patterns in counties from the barcodes alone. In future work, it is worth considering adjustments to our constructions that improve their ability to detect voting islands. For example, one may wish to apply a scaling based on voting preference (as in our adjacency construction) to a geographical map instead of to precinct vertices to obtain a sublevel-set filtration. Such an approach may help leverage the voting-strength interpretation of our adjacency construction while also enjoying the easily interpretable visual contiguity of our level-set construction.

Although we have tailored our methods to yield improvements for the particular problem of detecting voting patterns from SHAPEFILE data, one can use an adjacency construction on data sets with a network structure and the level-set construction is appropriate for any type of 2D manifold data (and one can extend it to higher

Table 4 *Proportion of long-persistence features that identify a real voting-map feature in our simplicial complexes. For each county, we show the value from the method(s) with the largest proportion in bold. In general, both our adjacency construction and our level-set construction perform very well, whereas we obtain mixed results with the VR and alpha complexes. A "–" symbol signifies either that we do not compute the associated simplicial complex or that there are no features.*

County	VR		Alpha		Adjacency		Level-set	
	C	T	C	T	C	T	C	T
Alameda	–	0.00	**1.00**	–	**1.00**	–	**1.00**	–
Alpine	–	–	–	–	–	–	–	–
Amador	–	**1.00**	–	–	–	–	–	–
Calaveras	–	**1.00**	–	–	–	–	–	**1.00**
Colusa	–	**1.00**	–	–	–	–	–	**1.00**
Contra Costa	–	0.00	0.00	–	**1.00**	–	**1.00**	**1.00**
Del Norte	–	0.00	–	–	–	**1.00**	–	0.00
El Dorado	0.00	**1.00**	–	–	**1.00**	**1.00**	–	**1.00**
Fresno	–	–	0.00	0.00	0.67	0.00	–	**1.00**
Glenn	–	0.00	–	–	–	0.00	–	**1.00**
Humboldt	0.00	0.00	–	–	0.50	–	**1.00**	**1.00**
Imperial	0.20	**1.00**	–	–	**1.00**	**1.00**	–	**1.00**
Inyo	–	0.00	–	–	–	**1.00**	–	–
Kern	–	–	0.00	**1.00**	**1.00**	**1.00**	–	**1.00**
Kings	0.00	0.00	–	–	**1.00**	0.67	–	0.87
Lake	**1.00**	0.00	–	–	–	–	**1.00**	–
Lassen	–	**1.00**	–	–	–	–	**1.00**	**1.00**
Los Angeles	–	–	0.00	0.00	–	–	**1.00**	–
Madera	**1.00**	**1.00**	–	–	**1.00**	**1.00**	–	**1.00**
Marin	–	–	**1.00**	–	**1.00**	–	**1.00**	–
Mariposa	–	**1.00**	–	–	–	–	–	–
Mendocino	–	0.00	**1.00**	–	**1.00**	–	**1.00**	–
Merced	0.11	**1.00**	–	–	0.50	**1.00**	–	**1.00**
Modoc	–	0.00	–	–	–	–	–	–
Mono	0.00	–	–	–	–	–	–	–
Monterey	–	0.00	0.00	–	**1.00**	0.00	**1.00**	**1.00**
Napa	0.25	0.00	–	–	**1.00**	–	0.75	–
Nevada	0.00	**1.00**	–	–	**1.00**	**1.00**	**1.00**	**1.00**
Orange	–	–	0.00	0.00	0.00	0.50	**1.00**	**1.00**
Placer	0.50	–	–	0.00	–	**1.00**	**1.00**	**1.00**
Plumas	–	**1.00**	–	–	–	**1.00**	–	**1.00**
Riverside	–	–	0.00	0.33	**1.00**	**1.00**	**1.00**	**1.00**
Sacramento	–	–	0.00	0.00	0.00	**1.00**	**1.00**	**1.00**
San Benito	**1.00**	0.00	–	–	**1.00**	–	–	**1.00**
San Bernardino	–	–	0.00	0.00	–	0.75	–	**1.00**
San Diego	–	–	0.00	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
San Francisco	–	–	0.00	–	**1.00**	–	**1.00**	–
San Joaquin	–	–	0.00	0.00	0.75	**1.00**	**1.00**	**1.00**
San Luis Obispo	0.00	0.14	–	–	**1.00**	**1.00**	–	**1.00**
San Mateo	–	–	**1.00**	–	**1.00**	–	**1.00**	–
Santa Barbara	–	**1.00**	0.00	–	0.67	**1.00**	–	**1.00**
Santa Cruz	–	–	**1.00**	–	0.00	–	**1.00**	–
Shasta	–	0.00	–	–	–	**1.00**	–	–
Sierra	–	–	–	–	–	–	–	–
Solano	0.00	0.00	–	–	**1.00**	**1.00**	**1.00**	**1.00**
Sonoma	–	0.00	0.00	–	**1.00**	–	**1.00**	–
Stanislaus	0.00	0.00	–	–	**1.00**	**1.00**	–	**1.00**
Sutter	–	0.00	–	–	–	**1.00**	–	**1.00**
Tehama	–	0.00	–	–	–	**1.00**	–	–
Trinity	–	0.00	–	–	–	0.00	**1.00**	**1.00**
Tulare	0.00	–	–	0.00	–	**1.00**	–	**1.00**
Tuolomne	–	0.00	–	0.00	–	**1.00**	–	**1.00**
Yolo	0.00	–	–	–	**1.00**	**1.00**	0.00	0.00
Yuba	–	0.00	–	–	–	**1.00**	–	**1.00**

dimensions with some programming adjustments, although computations take longer). More generally, given the ubiquity of 2D spatial data, the insights that we highlighted in our application to voting data are relevant for a broad range of problems in the study of transportation networks, spatial demography, granular materials, biological structures, and many other topics.

Appendix A. Simplicial Homology. In this appendix, we discuss the formalism of simplicial homology, which we discussed at an intuitive level in the main text. There are many different homology theories in algebraic topology. We give context for our particular choice of simplicial homology. For an explanation of the differences between simplicial homology and other common homology theories, see [23].

We begin by defining some of the basic building blocks of simplicial homology.

DEFINITION A.1. *A* k-simplex *is a* k*-dimensional polytope that is the convex hull of its* $k+1$ *vertices.*

DEFINITION A.2. *An* orientation *of a* k*-simplex is an ordering of the vertices, which we write as* $(v_0, \dots, v_k)$*, with the rule that two orderings define the same orientation if and only if they differ by an even permutation.*

DEFINITION A.3. *An* m-face *is the convex hull of a subset of cardinality* $m+1$ *of a* k*-simplex, with* $m < k$ *and the orientation preserved. A* face *refers to an* m*-face of any dimension* m*.*

DEFINITION A.4. *A simplex* A *is a* coface *of a simplex* B *if* B *is a face of* A*.*

DEFINITION A.5. *A* simplicial complex S *is a set of simplices that satisfies the following conditions:*

1. *every face of a simplex in* S *is also in* S*;*
2. *the intersection of any two simplices* $\sigma_1, \sigma_2 \in S$ *is a face of both* σ_1 *and* σ_2*.*

Our definition of simplicial complex makes no use of orientation. However, in our discussion of simplicial homology, we will see that orientation of simplices is very important.

DEFINITION A.6. *Let* S *be a simplicial complex. A* simplicial k-chain *is a finite formal sum*

$$\sum_{i=1}^{N} c_i \sigma_i \,,$$

where σ_i *is an oriented* k*-simplex and each* $c_i \in F$ *for some field* F*.*

We denote the group of k-chains on S by C_k. (With a consistent choice of orientation, we can also consider this as the free Abelian group on the basis of k-simplices in S.)

DEFINITION A.7. *Let* $\sigma = (v_0, \dots, v_k)$ *be an oriented* k*-simplex. The* boundary operator

$$\delta_k : C_k \to C_{k-1}$$

is the homomorphism defined by

$$\delta_k(\sigma) = \sum_{i=0}^{k} (-1)^i (v_0, \dots, \hat{v_i}, \dots, v_k) \,,$$

where $(v_0, \dots, \hat{v_i}, \dots, v_k)$ *is the oriented* $(k-1)$*-simplex that we obtain by deleting the* i*th vertex of* σ*.*

Elements of $Z_k = \ker \delta_k$ *are called* cycles*, and elements of* $B_k = \operatorname{im} \delta_{k+1}$ *are called* boundaries*.*

One can show by direct computation that $\delta^2 = 0$, so the groups (C_k, δ_k) form a chain complex. See [23] for a discussion of chain complexes.

DEFINITION A.8. *The* kth homology group H_k *of* S *over* F *is the quotient group*

$$H_k(S; F) = Z_k / B_k \, .$$

Note that $H_k(S; F)$ is nontrivial precisely when there are k-cycles on the simplicial complex S that are not boundaries; this occurs when there are k-dimensional holes. For example, a cycle between three points gives a 1-cycle,[8] and it is also a boundary precisely when the triangle with vertices at those three points is in the simplicial complex S.

In our application (and in many applications of TDA), we compute homology groups over the field $\mathbb{F}_2$. Crucially, $1 = -1 \in \mathbb{F}_2$, so we do not need to consider the orientation of our simplicial complexes.

The final definition that we introduce is that of a simplicial map.

DEFINITION A.9. *Let* S *and* T *be simplicial complexes. A* simplicial map $f : S \to T$ *is a function from the vertex set of* S *to the vertex set of* T *that preserves simplices.*

A simplicial map $f : S \to T$ also induces a homomorphism $f_* : H_k(S) \to H_k(T)$ for each nonnegative integer k. The homomorphism f_* is associated with a chain map from the k-chain complex of S to the k-chain complex of T. This chain map is

$$(v_0, \ldots, v_k) \mapsto (f(v_0), \ldots, f(v_k)) \, ,$$

where $(f(v_0), \ldots, f(v_k)) = 0$ if two or more of $f(v_0), \ldots, f(v_k)$ are not distinct.

This construction gives a functor from simplicial complexes to Abelian groups; this is essential to the theory of PH that we discussed in section 2.2.

Appendix B. Algorithms and Implementations. In this appendix, we discuss the algorithms that we developed to construct our simplicial complexes. All implementations that we discuss in this section are available at https://github.com/mhcfeng/precinct. For the computation of VR and alpha complexes, we use built-in functionality of the software package GUDHI [31]. For the adjacency and level-set constructions, we implement (in PYTHON) the incremental VR algorithm that is described in [53]. This algorithm adds one vertex at a time to a simplicial complex, and it then checks all possible cofaces of that vertex; it adds them if all other vertices of a coface are already part of the simplicial complex. To use this algorithm, we need to do some preprocessing, which we discuss in the next two subsections.

B.1. Adjacency Complex. The incremental VR algorithm that we use requires the following items as input: a list of vertices, a list of neighbors for each vertex, and some method of ordering the vertices to determine whether or not a neighbor is a "lower neighbor" (i.e., a neighbor that appears prior to the vertex in the ordering). Specifically, the ordering of the vertices must respect the entry times of those vertices. To determine the neighboring precincts for each precinct, we wrote code in QGIS that checks for queen adjacency. (Recall from the main text that two precincts are queen adjacent if they touch each other at any point, including corners.)

We then sort precincts by strength of preference for a particular candidate, as the precincts enter a filtered simplicial complex in order from strongest preference to weakest preference. Once we set this ordering, we compare a precinct to its neighbors to determine whether its neighbors are already in the simplicial complex. It is then straightforward to apply the incremental VR algorithm.

[8] This notion of "cycle" is different from the one in network analysis [33].

B.2. Level-Set Complex. Constructing a level-set complex requires several steps. First, we rasterize our SHAPEFILEs to obtain geographical maps in image format of all precincts in a county that voted for the same candidate. We denote this image data by X, and we constrain these images to have dimension no greater than 250×250. We then define a function $\phi(X, 0)$, where $\phi(x, 0)$ gives the distance from a point $x \in \mathbb{R}^2$ to the boundary, such that the boundary is the 0-level set of $\phi(X, 0)$. We then implement a level-set method with motion according to normal forces [34] to generate the evolved geographical map $\phi(X, T)$ at each time T. To convert ϕ to a simplicial complex S, we implement Algorithm B.1, which takes the following items as input: $\phi(X, T)$, a list V of vertices that are already in the simplicial complex S, a list $\tilde{t}$ of entry times for all vertices that are already in S, and the current time T.

Algorithm B.1 Generate ordered vertices from ϕ.

Given ϕ, V, $\tilde{t}$, T
$V' = \{v : v \notin V\,;\, \phi(v, T) < 0\,;\, \text{row}(v) = 0 \pmod 5\,, \text{col}(v) = 0 \pmod 5\}$
for $v \in V'$ **do**
  $V = V \bigcup \{v\}$
  $\tilde{t}(v) = T$
end for
return V, $\tilde{t}$

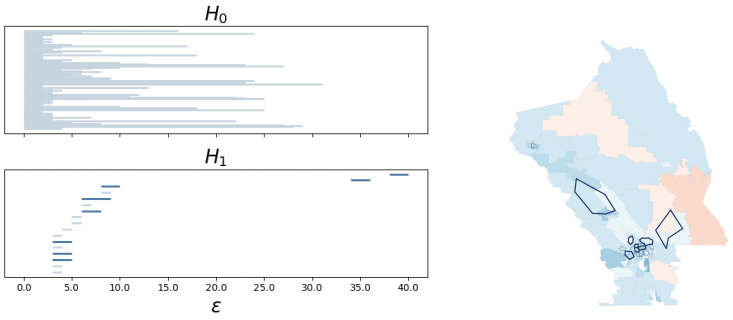
As vertices, we use only pixels that are in rows and columns that are multiples of 5 (see Algorithm B.1). This prevents us from having more than 50×50 potential vertices, which would significantly increase computation time. It also reduces the amount of noise in the barcodes, because holes must be sufficiently large in diameter for us to detect them. Once we have a list of vertices and their entry times, we generate 1-simplices using Algorithm B.2.

Algorithm B.2 Generate level-set adjacencies.

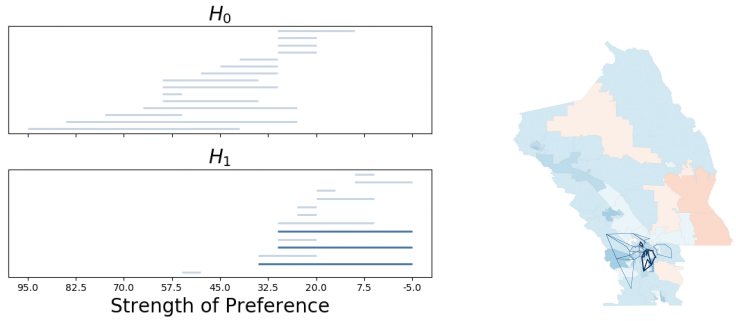
Given V, height h of image, width w of image
for $v \in V$ **do**
  Set $N(v)$ to the set of six possible neighbors of v. (These are the four cardinal neighbors, along with the northwest and southeast diagonal neighbors. We limit ourselves to six neighbors because this yields a convenient triangulation and connecting to all eight neighbors would result in nonplanarity.)
  $N(v) = N(v) \bigcap V$
end for
return N

Once we have generated the 1-simplices, we use the entry times $\tilde{t}$ from Algorithm B.1 to determine whether or not a neighbor of a given vertex is a lower neighbor in the incremental VR algorithm.

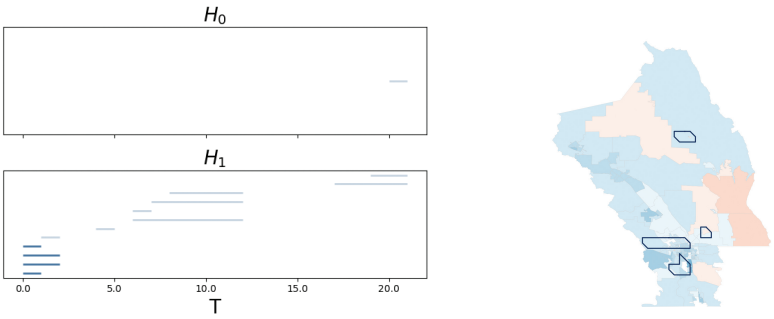
Appendix C. Additional Examples. In Figures 13 and 14, we show barcodes and feature maps for Napa County and Los Angeles County. These examples further illustrate some of the problems with barcode interpretability that we discussed in section 3.4.



(a) VR complex

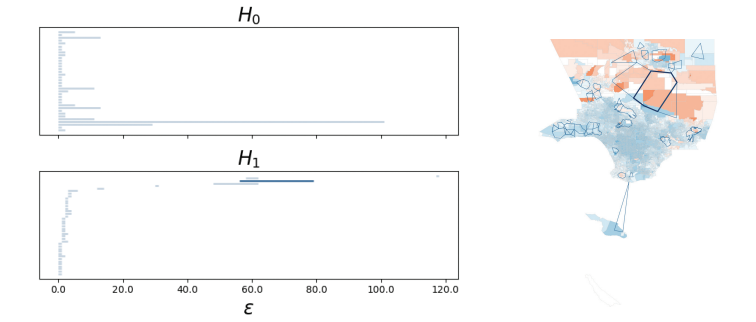


(b) Adjacency complex

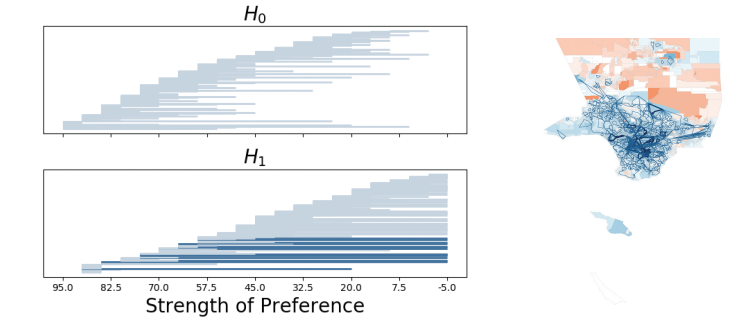


(c) Level-set complex

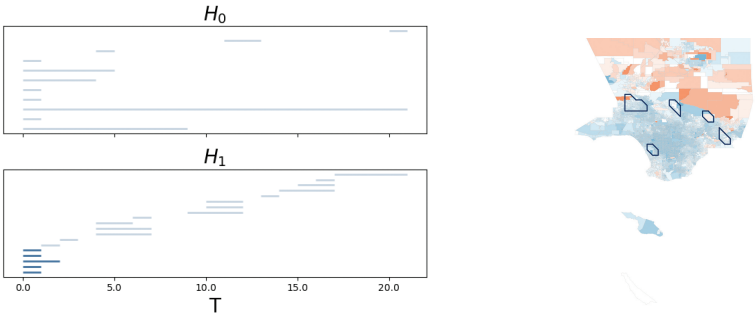
Fig. 13 *Barcodes and feature maps for blue precincts in Napa County. There are several long-persistence bars in the H_1 barcode of the VR complex. Some of these correspond to real holes in the densely populated areas in the southern region of the county, but others correspond to contiguous blue regions without holes, making it difficult to distinguish signal from noise. By contrast, the H_1 barcode of the adjacency complex has three long-persistence features, all of which correspond to light blue or white holes. Similarly, the H_1 barcode of the level-set complex has four features that start at time 0 and correspond to visible white or red holes. There is a red hole in the eastern part of the county that is detected by the alpha and level-set complexes, but not by the adjacency complex. This is due to the shape of the blue precinct, which wraps partially around a red precinct such that it covers precisely enough grid points in the level-set complex to register as a hole, despite not actually fully surrounding the red precinct. In practice, this occurs rarely in our voting data, but it does give an example of a potential problem with the level-set complex.*



(a) Alpha complex



(b) Adjacency complex



(c) Level-set complex

Fig. 14 *Barcodes and feature maps for blue precincts in Los Angeles County. We again observe many featues that do not have long persistence in the H_1 barcode of the alpha complex. This arises from the fact that the southern part of the county has a much higher population density than the northern part. There is also a single long-persistence feature; its generators are adjacent to red precincts that are not surrounded by blue precincts. The large number of precincts in this county makes it difficult to interpret many of the cycles in the highly populated precincts by eye. The alpha complex includes several cycles that traverse red swaths of the county that do not appear to be holes, whereas this does not occur in either the adjacency complex or the level-set complex.*

Appendix D. Complete Table of Computation Times of Simplicial Complexes. In Table 5, we give the computation times for the constructions of all computed simplicial complexes.

Table 5 *Computation times for the constructions of our simplicial complexes. (We give results to three significant digits, so values with fewer visible digits have the appropriate number of 0s appended to them.)*

County	VR		Alpha		Adjacency		Level-set	
	C	T	C	T	C	T	C	T
Alameda	–	0.191 s	0.742 s	–	1.62 s	0.0019 s	4.97 s	4.76 s
Alpine	0.00169 s	0.015 s	–	–	0.00174 s	0.000727 s	12.3 s	15.5 s
Amador	0.000706 s	0.0323 s	–	–	0.00281 s	0.00591 s	5.18 s	5.24 s
Calaveras	0.00117 s	0.0172 s	–	–	0.000872 s	0.00248 s	9.66 s	7.41 s
Colusa	0.00097 s	0.00251 s	–	–	0.00184 s	0.00175 s	4.96 s	6.31 s
Contra Costa	–	0.593 s	0.468 s	–	0.619 s	0.0033 s	4.81 s	5.12 s
Del Norte	0.0011 s	0.0187 s	–	–	0.00265 s	0.0039 s	13.1 s	10.6 s
El Dorado	0.302 s	182 s	–	–	0.00363 s	0.0905 s	5.46 s	5.62 s
Fresno	–	–	0.143 s	0.0952 s	0.123 s	0.102 s	7.73 s	8.54 s
Glenn	0.00116 s	0.0433 s	–	–	0.000836 s	0.00421 s	5.45 s	5.3 s
Humboldt	43.7 s	0.0214 s	–	–	0.0309 s	0.00644 s	10.1 s	10.6 s
Imperial	20.7 s	0.756 s	–	–	0.0137 s	0.00291 s	9.29 s	6.2 s
Inyo	0.00102 s	0.00329 s	–	–	0.00112 s	0.00215 s	7.32 s	8.02 s
Kern	–	–	0.0737 s	0.221 s	0.109 s	0.388 s	3.34 s	4.26 s
Kings	0.93 s	108 s	–	–	0.104 s	0.0847 s	10.5 s	17.4 s
Lake	0.131 s	0.0264 s	–	–	0.00672 s	0.00407 s	10.7 s	11.7 s
Lassen	0.00195 s	0.81 s	0.000417 s	0.0234 s	0.00343 s	0.0108 s	10.9 s	11.9 s
Los Angeles	–	–	15.5 s	0.133 s	39.3 s	0.0602 s	9.96 s	12.9 s
Madera	0.0344 s	0.13 s	–	–	0.0046 s	0.00399 s	5.24 s	6.03 s
Marin	–	0.00196 s	0.0705 s	–	0.063 s	0.000784 s	8.46 s	7.56 s
Mariposa	0.0012 s	0.0155 s	–	–	0.00323 s	0.00215 s	5.32 s	5.62 s
Mendocino	–	0.0317 s	0.0857 s	–	0.0571 s	0.00148 s	10.3 s	9.52 s
Merced	489 s	59.4 s	–	–	0.0217 s	0.0154 s	6.68 s	7.18 s
Modoc	1.91×10^{-6} s	0.0112 s	–	–	2.15×10^{-6} s	0.00271 s	3.81 s	4.37 s
Mono	0.00116 s	0.00194 s	–	–	0.00152 s	0.000919 s	5.8 s	5.78 s
Monterey	–	4.23 s	0.272 s	–	0.0766 s	0.00302 s	5.49 s	5.6 s
Napa	655 s	0.00569 s	–	–	0.0478 s	0.00115 s	8.31 s	8.47 s
Nevada	0.168 s	0.134 s	–	–	0.00751 s	0.00543 s	3.24 s	3.11 s
Orange	–	–	0.844 s	0.693 s	1.1 s	0.613 s	8.1 s	8.47 s
Placer	0.736 s	–	–	0.184 s	0.0172 s	0.553 s	3.1 s	3.35 s
Plumas	0.00138 s	0.0269 s	–	–	0.00109 s	0.00401 s	4.65 s	5.52 s
Riverside	–	–	0.263 s	0.422 s	0.483 s	0.554 s	2.21 s	1.99 s
Sacramento	–	–	0.516 s	0.0841 s	12.3 s	0.606 s	8.48 s	9.5 s
San Benito	0.1 s	0.00899 s	–	–	0.00662 s	0.00339 s	6.14 s	6.79 s
San Bernardino	–	–	1.77 s	0.833 s	0.691 s	0.476 s	4.39 s	5.25 s
San Diego	–	–	1.63 s	0.492 s	3.13 s	0.416 s	6.11 s	6.87 s
San Francisco	–	0 s	0.353 s	–	0.707 s	1.19×10^{-6} s	5.99 s	5.13 s
San Joaquin	–	–	0.0857 s	0.0487 s	0.108 s	0.052 s	8.22 s	13.2 s
San Luis Obispo	14.4 s	4.45 s	–	–	0.016 s	0.0115 s	3.61 s	3.88 s
San Mateo	–	0.00359 s	0.266 s	–	0.45 s	0.00442 s	8.28 s	5.84 s
Santa Barbara	–	3.35 s	0.0551 s	–	0.0612 s	0.0101 s	6.35 s	7.36 s
Santa Cruz	–	0.0017 s	0.0625 s	–	0.207 s	0.00239 s	5.7 s	10.4 s
Shasta	0.00118 s	120 s	–	–	0.00526 s	0.0426 s	3.9 s	4.47 s
Sierra	0.00342 s	0.00752 s	–	–	0.0046 s	0.00311 s	2.86 s	3.3 s
Solano	401 s	3.59 s	–	–	0.0438 s	0.013 s	6.24 s	5.86 s
Sonoma	–	0.0299 s	0.25 s	–	0.516 s	0.0175 s	6.26 s	5.46 s
Stanislaus	46.1 s	52.1 s	–	–	0.0173 s	0.0325 s	8.25 s	9.41 s
Sutter	0.00187 s	0.244 s	–	–	0.00494 s	0.0193 s	8.75 s	9.79 s
Tehama	1.91×10^{-6} s	0.372 s	–	–	3.1×10^{-6} s	0.0601 s	2.32 s	2.78 s
Trinity	0.000981 s	0.0038 s	–	–	0.00276 s	0.0225 s	8.44 s	8.86 s
Tulare	3.57 s	–	–	0.0515 s	0.0562 s	0.129 s	4.81 s	5.18 s
Tuolomne	0.000928 s	2.15 s	–	–	0.00328 s	0.0117 s	4.76 s	4.5 s
Yolo	51.4 s	0.0142 s	–	–	0.0635 s	0.00449 s	5.92 s	6.1 s
Yuba	0.00109 s	0.266 s	–	–	0.00988 s	0.0096 s	7.97 s	8.72 s

Acknowledgments. We thank Moon Duchin, Joshua Gensler, Mike Hill, Stan Osher, Nina Otter, Bernadette Stolz, Bao Wang, and two anonymous referees for helpful comments. We also thank Emilia Alvarez, Eion Blanchard, Austin Eide, Patrick Girardet, Everett Meike, Dmitriy Morozov, Justin Solomon, Courtney Thatcher, Jim Thatcher, and Maia Woluchem for insightful discussions.

REFERENCES

[1] M. Adamaszek and H. Adams, *The Vietoris–Rips complexes of a circle*, Pacific J. Math., 290 (2017), pp. 1–40. (Cited on p. 71)

[2] H. Adams, T. Emerson, M. Kirby, R. Neville, C. Peterson, P. Shipman, S. Chepushtanova, E. Hanson, F. Motta, and L. Ziegelmeier, *Persistence images: A stable vector representation of persistent homology*, J. Mach. Learn. Res., 18 (2017), pp. 218–252. (Cited on p. 80)

[3] P. Bajardi, M. Delfino, A. Panisson, G. Petri, and M. Tizzoni, *Unveiling patterns of international communities in a global city using mobile phone data*, European Phys. J. Data Sci., 4 (2015), art. 3. (Cited on p. 68)

[4] A. Banman and L. Ziegelmeier, *Mind the gap: A study in global development through persistent homology*, in Research in Computational Topology, E. W. Chambers, B. T. Fasy, and L. Ziegelmeier, eds., Springer International Publishing, Cham, Switzerland, 2018, pp. 125–144. (Cited on p. 68)

[5] R. Barnes and J. Solomon, *Gerrymandering and Compactness: Implementation Flexibility and Abuse*, preprint, https://arxiv.org/abs/1803.02857, 2018. (Cited on p. 85)

[6] U. Bauer, M. Kerber, J. Reininghaus, and H. Wagner, Phat—*Persistent homology algorithms toolbox*, in Mathematical Software—ICMS 2014, H. Hong and C. Yap, eds., Springer-Verlag, Heidelberg, Germany, 2014, pp. 137–143. (Cited on p. 80)

[7] P. Bendich, H. Edelsbrunner, D. Morozov, and A. Patel, *Homology and robustness of level and interlevel sets*, Homology Homotopy Appl., 15 (2013), pp. 51–72. (Cited on p. 75)

[8] O. Bobrowski, S. Mukherjee, and J. E. Taylor, *Topological consistency via kernel estimation*, Bernoulli, 23 (2017), pp. 288–328. (Cited on p. 80)

[9] P. Bubenik, *Statistical topological data analysis using persistence landscapes*, J. Mach. Learn. Res., 16 (2015), pp. 77–102. (Cited on p. 80)

[10] P. Bubenik, M. Hull, D. Patel, and B. Whittle, *Persistent homology detects curvature*, Inverse Problems, 36 (2020), art. 025008. (Cited on p. 72)

[11] H. M. Byrne, H. A. Harrington, R. Muschel, G. Reinert, B. J. Stolz, and U. Tillmann, *Topology characterises tumour vasculature*, Math. Today, 55 (2019), pp. 206–210. (Cited on p. 75)

[12] G. Carlsson, *Topological methods for data modelling*, Nat. Rev. Phys., 2 (2020), pp. 697–708 (Cited on p. 68)

[13] G. Carlsson, T. Ishkhanov, V. de Silva, and A. Zomorodian, *On the local behavior of spaces of natural images*, Internat. J. Comput. Vision, 76 (2008), pp. 1–12. (Cited on p. 69)

[14] C. Curto, *What can topology tell us about the neural code?*, Bull. Amer. Math. Soc., 54 (2017), pp. 63–78. (Cited on p. 69)

[15] M. Duchin and B. E. Tenner, *Discrete Geometry for Electoral Geography*, preprint, https://arxiv.org/abs/1808.05860, 2018. (Cited on p. 85)

[16] H. Edelsbrunner and J. Harer, *Computational Topology: An Introduction*, AMS, Providence, RI, 2010. (Cited on pp. 68, 73)

[17] H. Edelsbrunner, D. Kirkpatrick, and R. Seidel, *On the shape of a set of points in the plane*, IEEE Trans. Inform. Theory, 29 (1983), pp. 551–559. (Cited on p. 73)

[18] K. Emmett, B. Schweinhart, and R. Rabadan, *Multiscale topology of chromatin folding*, in Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (Formerly BIONETICS), BICT '15, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2016, pp. 177–180. (Cited on p. 69)

[19] R. Ghrist, *Barcodes: The persistent topology of data*, Bull. Amer. Math. Soc., 45 (2008), pp. 61–75. (Cited on pp. 68, 72)

[20] F. Gibou, R. Fedkiw, and S. Osher, *A review of level-set methods and some recent applications*, J. Comput. Phys., 353 (2018), pp. 82–109. (Cited on p. 83)

[21] The Gudhi Project, Gudhi *User and Reference Manual*, Version 3.0.0, Gudhi Editorial Board, 2015, https://gudhi.inria.fr/doc/3.0.0/. (Cited on p. 80)
[22] C. Giusti, R. Ghrist, and D. S. Bassett, *Two's company, three (or more) is a simplex*, J. Comput. Neurosci., 41 (2016), pp. 1–14. (Cited on p. 69)
[23] A. Hatcher, *Algebraic Topology*, Cambridge University Press, Cambridge, UK, 2002. (Cited on pp. 68, 72, 91, 92)
[24] D. P. Humphreys, M. R. McGuirl, M. Miyagi, and A. J. Blumberg, *Fast estimation of recombination rates using topological data analysis*, Genetics, 211 (2019), pp. 1191–1204. (Cited on p. 72)
[25] P. S. P. Ignacio and I. K. Darcy, *Tracing patterns and shapes in remittance and migration networks via persistent homology*, European Phys. J. Data Sci., 8 (2019), art. 1. (Cited on p. 68)
[26] L. Kanari, P. Dłotko, M. Scolamiero, R. Levi, J. C. Shillcock, K. Hess, and H. Markram, *A topological representation of branching neuronal morphologies*, Neuroinform., 16 (2018), pp. 3–13. (Cited on p. 75)
[27] M. Kerber and R. Sharathkumar, *Approximate Čech Complex in Low and High Dimensions*, preprint, https://arxiv.org/abs/1307.3272, 2013. (Cited on p. 74)
[28] M. Kramár, A. Goullet, L. Kondic, and K. Mischaikow, *Quantifying force networks in particulate systems*, Phys. D, 283 (2014), pp. 37–55. (Cited on p. 69)
[29] R. Kwitt, S. Huber, M. Niethammer, W. Lin, and U. Bauer, *Statistical topological data analysis—A kernel perspective*, in Advances in Neural Information Processing Systems 28, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, eds., Curran Associates, 2015, pp. 3070–3078. (Cited on p. 80)
[30] D. Lo and B. Park, *Modeling the spread of the Zika virus using topological data analysis*, PLoS ONE, 13 (2018), art. e0192120. (Cited on p. 68)
[31] C. Maria, *Filtered complexes*, in Gudhi User and Reference Manual, Version 3.0.0, Gudhi Editorial Board, 2015, https://gudhi.inria.fr/doc/3.0.0/group__simplex__tree.html. (Cited on p. 92)
[32] C. Maria, P. Dłotko, V. Rouvreau, and M. Glisse, *Rips complex*, in Gudhi User and Reference Manual, Version 3.0.0, Gudhi Editorial Board, 2016, https://gudhi.inria.fr/doc/3.0.0/group__rips__complex.html. (Cited on p. 80)
[33] M. E. J. Newman, *Networks*, 2nd ed., Oxford University Press, Oxford, UK, 2018. (Cited on pp. 83, 92)
[34] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, Appl. Math. Sci. 153, Springer-Verlag, Heidelberg, Germany, 2003. (Cited on pp. 76, 93)
[35] S. Osher and J. A. Sethian, *Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations*, J. Comput. Phys., 79 (1988), pp. 12–49. (Cited on p. 76)
[36] N. Otter, M. A. Porter, U. Tillmann, P. Grindrod, and H. A. Harrington, *A roadmap for the computation of persistent homology*, European Phys. J. Data Sci., 6 (2017), art. 17. (Cited on pp. 68, 69, 70, 72)
[37] L. Papadopoulos, M. A. Porter, K. E. Daniels, and D. S. Bassett, *Network analysis of particles and grains*, J. Complex Networks, 6 (2018), pp. 485–565. (Cited on pp. 68, 69)
[38] QGIS Association, *QGIS* 2.18.17: *A Free and Open Source Geographic Information System*, 2016, http://www.qgis.org. (Cited on pp. 70, 73)
[39] J. Reininghaus, S. Huber, U. Bauer, and R. Kwitt, *A stable multi-scale kernel for topological machine learning*, in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 4741–4748. (Cited on p. 80)
[40] J. W. Rocks, A. J. Liu, and E. Katifori, *The Topological Basis of Function in Flow Networks*, preprint, https://arxiv.org/abs/1901.00822, 2019. (Cited on pp. 68, 75)
[41] H. Ronellenfitsch, J. Lasser, D. C. Daly, and E. Katifori, *Topological phenotypes constitute a new dimension in the phenotypic space of leaf venation networks*, PLOS Comput. Biol., 11 (2015), art. e1004680. (Cited on p. 75)
[42] V. Rouvreau, *Alpha complex*, in Gudhi User and Reference Manual, Version 3.0.0, Gudhi Editorial Board, 2015, https://gudhi.inria.fr/doc/3.0.0/group__alpha__complex.html. (Cited on p. 80)
[43] V. Rouvreau, *Python interface*, in Gudhi User and Reference Manual, Version 3.0.0, Gudhi Editorial Board, 2016, https://gudhi.inria.fr/python/3.0.0. (Cited on p. 80)
[44] J. Schleuss, J. Fox, and P. Krishnakumar, *California* 2016 *Election Precinct Maps*, https://github.com/datadesk/california-2016-election-precinct-maps, 2016. (See https://www.latimes.com/projects/la-pol-ca-california-neighborhood-election-results/ for the associated newspaper article.) (Cited on p. 70)

[45] L. Speidel, H. A. Harrington, S. J. Chapman, and M. A. Porter, *Topological data analysis of continuum percolation with disks*, Phys. Rev. E, 98 (2018), art. 012318. (Cited on p. 68)
[46] B. J. Stolz, H. A. Harrington, and M. A. Porter, *The Topological "Shape" of Brexit*, preprint, https://arxiv.org/abs/1610.00752, 2016. (Cited on p. 68)
[47] B. J. Stolz, H. A. Harrington, and M. A. Porter, *Persistent homology of time-dependent functional networks constructed from coupled time series*, Chaos, 27 (2017), art. 047410. (Cited on pp. 69, 72)
[48] D. Taylor, F. Klimm, H. A. Harrington, M. Kramár, K. Mischaikow, M. A. Porter, and P. J. Mucha, *Topological data analysis of contagion maps for examining spreading processes on networks*, Nature Commun., 6 (2015), art. 7723. (Cited on p. 68)
[49] L. Vietoris, *Über den höheren zusammenhang kompakter räume und eine klasse von zusammenhangstreuen abbildungen*, Math. Ann., 97 (1927), pp. 454–472. (Cited on p. 73)
[50] K. Xia and G.-W. Wei, *Persistent homology analysis of protein structure, flexibility, and folding*, Internat. J. Numer. Methods Biomed. Engrg., 30 (2014), pp. 814–844. (Cited on p. 69)
[51] W. Zhou and H. Yan, *Alpha shape and Delaunay triangulation in studies of protein-related interactions*, Briefings Bioinform., 15 (2014), pp. 54–64. (Cited on p. 69)
[52] X. Zhu, A. Vartanian, M. Bansal, D. Nguyen, and L. Brandl, *Stochastic multiresolution persistent homology kernel*, in Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI '16, AAAI Press, 2016, pp. 2449–2455. (Cited on p. 80)
[53] A. Zomorodian, *Fast construction of the Vietoris–Rips complex*, Computers & Graphics, 34 (2010), pp. 263–271. (Cited on pp. 69, 73, 80, 92)
[54] A. Zomorodian and G. Carlsson, *Computing persistent homology*, Discrete Comput. Geom., 33 (2005), pp. 249–274. (Cited on pp. 72, 84)