Finding the Homology of Decision Boundaries with Active Learning

Weizhi Li
Arizona State University
weizhili@asu.edu

Gautam Dasarathy
Arizona State University
gautamd@asu.edu

Karthikeyan Natesan Ramamurthy
IBM Research
knatesa@us.ibm.com

Visar Berisha
Arizona State University
visar@asu.edu

Abstract

Accurately and efficiently characterizing the decision boundary of classifiers is important for problems related to model selection and meta-learning. Inspired by topological data analysis, the characterization of decision boundaries using their homology has recently emerged as a general and powerful tool. In this paper, we propose an active learning algorithm to recover the homology of decision boundaries. Our algorithm sequentially and adaptively selects which samples it requires the labels of. We theoretically analyze the proposed framework and show that the query complexity of our active learning algorithm depends naturally on the intrinsic complexity of the underlying manifold. We demonstrate the effectiveness of our framework in selecting best-performing machine learning models for datasets just using their respective homological summaries. Experiments on several standard datasets show the sample complexity improvement in recovering the homology and demonstrate the practical utility of the framework for model selection. Source code for our algorithms and experimental results is available at `https://github.com/wayne0908/Active-Learning-Homology`.

1 Introduction

A broadly known meta-learning [1] is to design an model to learn a task learning process and, with this model, a new task can be adapted to with fewer examples. [2] views meta-learning from another perspective: The complexity of the data at hand is an important insight that, if gleaned correctly from past experience, can greatly enhance the performance of a meta-learning procedure. A particularly useful characterization of data complexity is to understand the topological properties of the decision boundary; for example, by using topological data analysis (TDA) [3, 4, 5]. This scenario makes sense in settings where large corpora of labeled training data are available to recover the persistent homology of the decision boundary for use in downstream machine learning tasks [5, 6, 7, 8]. However the utility of this family of methods is limited in applications where labeled data is expensive to acquire.

In this paper, we explore the intersection of active learning and topological data analysis for the purposes of efficiently learning the persistent homology of the decision boundary in classification problems. In contrast to the standard paradigm, in active learning, the learner has access to unlabeled data and sequentially selects a set of points for an oracle to label. We propose an efficient active learning framework that adaptively select points for labeling near the decision boundary. A theoretical analysis of the algorithm results in an upper bound on the number of samples required to recover the recover the decision boundary homology. Naturally, this query complexity depends on the intrinsic complexity of the underlying manifold.

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

There have been several other studies that have explored the use of topological data analysis to characterize the decision boundary in classification problems. In [9], the authors use the persistent homology of the decision boundary to tune hyperparameters in kernel-based learning algorithms. They later extended this work and derived the conditions required to recover the homology of the decision boundary from only samples [5]. Other works have explored the use of other topological features to characterize the difficulty of classification problems [10, 11, 7]. While all previous work assumes full knowledge of data labels, only samples near the decision boundary are used to construct topological features. We directly address this problem in our work by proposing an active approach that adaptively and sequentially labels only samples near the decision boundary, thereby resulting in significantly reduced the complexity of needed labelled samples. To the best of our knowledge, this is the first work that explores the intersection of active learning and topological data analysis.

Our main contributions are as follows:

- We introduce a new algorithm for actively selecting samples to label in service of finding the persistent homology of the decision boundary. We provide theoretical conditions on the query complexity that lead to the successful recovery of the decision boundary homology.
- We evaluate the proposed algorithm for active homology estimation using synthetic data and compare its performance to a passive approach that samples data uniformly. In addition, we demonstrate the utility of our approach relative to a passive approach on a stylized model selection problem using real data.

2 Preliminaries

In this section, we define the decision boundary manifold and discuss the labeled Čech Complex [5] which we then use to estimate the homology of this manifold from labeled data. For more background and details, we direct the reader to Appendix B.

2.1 The Decision Boundary Manifold and Data

Let $\mathcal{X}$ be a Euclidean space that denotes the domain/feature space of our learning problem and let μ denote the standard Lebesgue measure on $\mathcal{X}$. We will consider the binary classification setting in this paper and let $\mathcal{Y} = \{0, 1\}$ denote the label set. Let p_{XY} denote a joint distribution on $\mathcal{X} \times \mathcal{Y}$. Of particular interest to us in this paper is the so-called *Bayes decision boundary* $\mathcal{M} = \{\mathbf{x} \in \mathcal{X} | p_{Y|X}(1|\mathbf{x}) = p_{Y|X}(0|\mathbf{x})\}$. Indeed, identifying $\mathcal{M}$ is equivalent to being able to construct the provably optimal binary classifier called the Bayes optimal predictor:

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } p_{Y|X}(1 \mid \mathbf{x}) \geq 0.5 \\ 0 & \text{otherwise} \end{cases}. \tag{1}$$

Following along the lines of [5], the premise of this paper relies on supposing that the set $\mathcal{M}$ is in fact a reasonably well-behaved manifold[1]. That is, we will make the following assumption.

Assumption 1. *The decision boundary $\mathcal{M}$ is a manifold in $\mathcal{X}$ with a condition number $1/\tau$.*

The condition number $\frac{1}{\tau}$ is an intrinsic property of $\mathcal{M}$ and encodes both the local and global curvature of the manifold. The value τ is the largest number such that the open normal bundle about $\mathcal{M}$ of radius r is embedded in $\mathbb{R}^K$ for every $r < \tau$. *E.g.*, in Figure 1, $\mathcal{M}$ is a circle in $\mathbb{R}^2$ and τ is its radius. We refer the reader to the Appendix A (or [13]) for a formal definition.

Now we will suppose that we have access to N i.i.d samples $\mathcal{D} = \{\mathbf{x}_1, ..., \mathbf{x}_N\} \subset \mathcal{X}$ drawn according to the marginal distribution p_X. Notice that in a typical passive learning setting, we assume access to the N corresponding labels as well. The goal of this paper is to demonstrate that if we are allowed to choose the labels observed in a sequential and adaptive fashion, then we may obtain far fewer than N labels while still being competitive with the traditional passive learning approaches. Based on the observed data, we define the set $\mathcal{D}^0 = \{\mathbf{x} \in \mathcal{D} : f(\mathbf{x}) = 0\}$, that is the set of all samples with Bayes optimal label of 0; similarly, we let $\mathcal{D}^1 = \{\mathbf{x} \in \mathcal{D} : f(\mathbf{x}) = 1\}$.

[1]Note that it is conceivable that the decision boundary is not strictly a manifold. While this assumption is critical to the rest of this paper, it is possible to extend the results here by following the theory in [12]. We will leave a thorough exploration of this for future work.

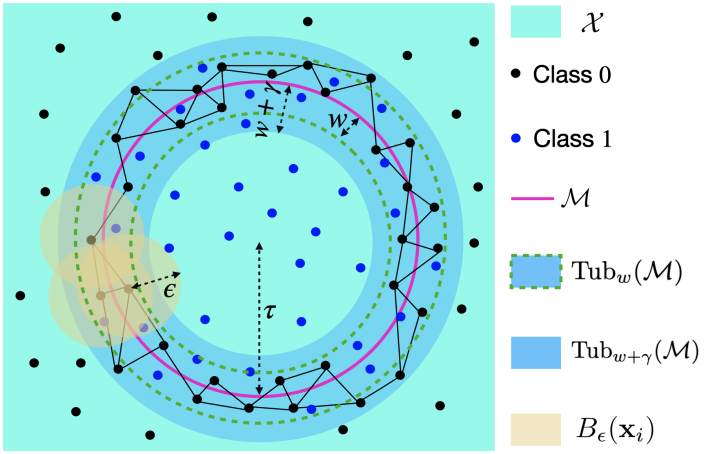


Figure 1: An example of $(\epsilon, \gamma)-$labeled Čech complex, constructed in a tubular neighborhood $\mathrm{Tub}_{w+\gamma}(\mathcal{M})$ of radius $w + \gamma$, for a manifold $\mathcal{M}$ of condition number $1/\tau$. The overlap between the two classes ($\mathfrak{D}$) is contained in $\mathrm{Tub}_w(\mathcal{M})$. The complex is constructed on samples in class 0, by placing balls of radius ϵ ($B_\epsilon(\mathbf{x}_i)$), and is "witnessed" by samples in class 1. $\mathcal{X}$ is the compact probability space for the data. Each triangle is assumed to be a $2-$simplex in the simplicial complex. Note that we keep the samples from both classes sparse for aesthetic reasons.

2.2 The Labeled Čech Complex

As outlined in Section 1, our goal is to recover the homological summaries of $\mathcal{M}$ from data. Homological summaries such as Betti numbers estimate the number of connected components and the number of holes of various dimensions that are present in $\mathcal{M}$. Since we only have a sample of data points in practice, we first construct a simplicial complex from these points that mimics the shape of $\mathcal{M}$. We can then estimate the rank of any homology group H_i of dimension i from this complex. This rank is called the Betti number β_i and informally denotes the number of holes of dimension i in the complex. The multi-scale estimation of Betti numbers, which involves gradual "thickening" of the complex, results in a persistence diagram PD_i that encodes the *birth* and *death time* of the $i-$dimensional holes in the complex. For more background we refer the reader to [14].

In [5], the authors consider the passive learning setting for estimating the homology of $\mathcal{M}$ and propose a simplicial approximation for the decision boundary called the Labeled Čech (LČ) Complex. We now provide a definition of this complex, letting $B_\epsilon(\mathbf{x})$ denote a ball of radius ϵ around $\mathbf{x}$. We refer the reader to the Appendix B or [5] for the more details.

Definition 1. *Given $\epsilon, \gamma > 0$, an (ϵ, γ)-labeled Čech complex is a simplicial complex constructed from a collection of simplices such that each simplex σ is formed on the points in a set $S \subseteq \mathcal{D}^0$ witnessed by the reference set $\mathcal{D}^1$ satisfying the following conditions:* ***(a)*** *$\bigcap_{\mathbf{x}_i \in \sigma} B_\epsilon(\mathbf{x}_i) \neq \emptyset$, where $\mathbf{x}_i \in S$ are the vertices of σ.* ***(b)*** *$\forall \mathbf{x}_i \in S \subseteq \mathcal{D}^0$, $\exists \mathbf{x}_j \in \mathcal{D}^1$ such that, $\|\mathbf{x}_i - \mathbf{x}_j\|_2 \leq \gamma$.*

The set S is used to construct the LČ complex witnessed by the reference set $\mathcal{D}^1$. This allows us to infer that each vertex of the simplices σ are within distance γ to some point in $\mathcal{D}^1$. The authors in [5] show that, under certain assumption on the manifold and the distribution, provided sufficiently many random samples (and their labels) drawn according p_{XY}, the set $U = \bigcup_{\mathbf{x}_i \in \sigma} B_\epsilon(\mathbf{x}_i)$ forms a cover of $\mathcal{M}$ and deformation retracts to $\mathcal{M}$. Moreoever, the nerve of the covering is homotopy equivalent to $\mathcal{M}$. The assumptions under which the above result holds also turns out to be critical to achieve the results of this paper, and hence we will devote the rest of this section to elaborating on these.

Before stating our assumptions, we need a few more definitions. For the distribution p_{XY}, we will let

$$\mathfrak{D} \triangleq \{\mathbf{x} \in \mathcal{X} : p_{X|Y}(\mathbf{x} \mid 1) p_{X|Y}(\mathbf{x} \mid 0) > 0\}.$$

In other words, $\mathfrak{D}$ denotes the region of the feature space where both classes overlap, i.e., both class conditional distributions $p_{X|Y}(\cdot \mid 1)$ and $p_{X|Y}(\cdot \mid 0)$ have non-zero mass. For any $r > 0$, we let $\mathrm{Tub}_r(\mathcal{M})$ denote a "tubular" neighborhood of radius r around $\mathcal{M}$ [13]. We will let $\mathrm{Tub}_w(\mathcal{M})$ denote the smallest tubular neighborhood of $\mathcal{M}$ that encloses $\mathfrak{D}$. That is

$$w \triangleq \arg\inf \{\rho > 0 : \mathrm{Tub}_\rho(\mathcal{M}) \supset \mathfrak{D}\}.$$

A stylized example that highlights the relationship between these parameters is shown in Figure 1. In this sequel, we will introduce a relevant assumption and a lemma underlying the results of this paper

(similar to those in [13, 5]). All assumptions and follow-on results are dependent on two parameters that are specific to the joint distribution: w - the amount of overlap between the distributions and τ - the global geometric property of the decision boundary manifold.

Assumption 2. $w < (\sqrt{9} - \sqrt{8})\tau$.

A principal difference between [13] and our work is that [13] has direct access to the manifold and supposes that all generated samples are contained within $\mathrm{Tub}_{(\sqrt{9}-\sqrt{8})\tau}(\mathcal{M})$; this is one of the sufficient conditions for manifold reconstruction from samples. In contrast, as in [5], we do not have direct access to the decision boundary manifold. Rather, certain properties of $\mathcal{M}$ are *inferred* from the labels. In this paper, we show that we can infer the homology of $\mathcal{M}$ with far fewer labels if we are allowed to sequentially and adaptively choose which labels to obtain.

Since the generated samples do not necessarily reside within $\mathrm{Tub}_{(\sqrt{9}-\sqrt{8})\tau}(\mathcal{M})$, it is not immediately apparent that it is possible to find an S (see Definition 1) that is entirely contained in $\mathrm{Tub}_{(\sqrt{9}-\sqrt{8})\tau}(\mathcal{M})$. However, Assumption 2 allows us to guarantee precisely this. To see this, we will first state the following lemma.

Lemma 1. *Provided $\mathcal{D}^0$ and $\mathcal{D}^1$ are both $\frac{\gamma}{2}$-dense in $\mathcal{M}$, then S is contained in $Tub_{w+\gamma}(\mathcal{M})$ and it is $\frac{\gamma}{2}$-dense[2] in $\mathcal{M}$.*

The lemma 1 is based on definition 1(b) that S is a subset of $\mathcal{D}^0$ within distance γ from $\mathcal{D}^1$ and the proof goes as follows. $\mathcal{D}^0$ and $\mathcal{D}^1$ being $\frac{\gamma}{2}$-dense in $\mathcal{M}$ implies $\|\mathbf{x}_i - \mathbf{x}_j\|_2 < \gamma$, where $\mathbf{x}_i \in \mathcal{D}^0 \bigcap B_{\frac{\gamma}{2}}(\mathbf{p})$ and $\mathbf{x}_j \in \mathcal{D}^1 \bigcap B_{\frac{\gamma}{2}}(\mathbf{p})$ for every $\mathbf{p} \in \mathcal{M}$. Therefore, as the distance from $\mathbf{x}_i \in S \subseteq \mathcal{D}^0$ to $\mathbf{x}_j \in \mathcal{D}^1$ is bounded by γ, S is also $\frac{\gamma}{2}$-dense in $\mathcal{M}$. This immediately implies S is $(w+\gamma)$-dense in $\mathcal{M}$. Given lemma 1, as our initial sampling strategy is such that both $\mathcal{D}^0$ and $\mathcal{D}^1$ are $\frac{\gamma}{2}$-dense in $\mathcal{M}$, S is contained in $Tub_{w+\gamma}(\mathcal{M})$ and $\frac{\gamma}{2}$-dense in $\mathcal{M}$. and provided we choose $\gamma < (\sqrt{9} - \sqrt{8})\tau - w$, we can guarantee that $S \subset \mathrm{Tub}_{(\sqrt{9}-\sqrt{8})\tau}(\mathcal{M})$.

With $S \subset \mathrm{Tub}_{(\sqrt{9}-\sqrt{8})\tau}(\mathcal{M})$, S being $(w+\gamma)$-dense in $\mathcal{M}$ and an appropriate ϵ properly selected, we will have the (ϵ, γ)-LČ complex homotopy equivalent to $\mathcal{M}$ per theorems in [5, 13]. We now state the following proposition.

Proposition 1. *The (ϵ, γ)-LČ complex is homotopy equivalent to $\mathcal{M}$ as long as (a) $\gamma < (\sqrt{9} - \sqrt{8})\tau - w$; (b) $\mathcal{D}^0$ and $\mathcal{D}^1$ are $\frac{\gamma}{2}$-dense in $\mathcal{M}$ and (c) $\epsilon \in \left(\frac{(w+\gamma+\tau)-\sqrt{(w+\gamma)^2+\tau^2-6\tau(w+\gamma)}}{2}, \frac{(w+\gamma+\tau)+\sqrt{(w+\gamma)^2+\tau^2-6\tau(w+\gamma)}}{2}\right)$.*

Notice that the range for ϵ in (c) above is non-trivial as long as $\gamma < (\sqrt{9} - \sqrt{8})\tau - w$. A pictorial description of relations between w, γ and τ is shown in the stylized example in Figure 1; here $\mathcal{M}$ is a circle, $\mathrm{Tub}_{w+\gamma}(\mathcal{M})$ is an annulus and the radius ϵ of the covering ball $B_\epsilon(\mathbf{x})$ is constrained by τ.

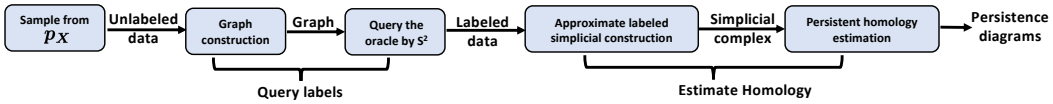


Figure 2: The proposed active learning framework for finding the homology of decision boundaries.

3 Active Learning for Finding the Homology of Decision Boundaries

As the definitions above and results from [5] make clear, constructing a useful LČ complex requires sampling both class-conditional distributions in the region around the decision boundary to a sufficient resolution. The key insight of our paper is to devise a framework based on active learning that sequentially and adaptively decides where to obtain data and therefore query-efficiently samples points near the decision boundary. In what follows, we will provide a brief description of our algorithm, and then we will establish rigorous theoretical guarantees on the query complexity of the proposed algorithm.

[2] A set W is $\frac{\gamma}{2}$-dense in $\mathcal{M}$ if for every $\mathbf{p} \in \mathcal{M}$ there exists a $\mathbf{x} \in W$ such that $\|\mathbf{p} - \mathbf{x}\|_2 < \frac{\gamma}{2}$. In other words, there exists at least one $\mathbf{x} \in W$ in $B_{\frac{\gamma}{2}}(\mathbf{p})$ for every $\mathbf{p} \in \mathcal{M}$.

3.1 The Active Learning Algorithm

A schematic diagram of the proposed active learning framework is presented in Figure 2. As illustrated, the framework starts from sampling sufficient unlabelled data from p_X. Subsequently, the framework takes as input an unlabeled dataset $\mathcal{D}$, and this dataset is used to generate an appropriate graph on the data. This graph is then used to iteratively query labels near the decision boundary. The subset of labeled samples are used to estimate the homology, resulting in the persistence diagram of the LČ complex. We briefly outline the label query and homology estimation phases below, and refer the reader to the appendices for the full details.

Label query phase: The label query phase starts with constructing a graph $G = (\mathcal{D}, E)$ from the unlabeled dataset $\mathcal{D}$. While other choices are possible, we will suppose that the graph we construct is either a k-radius nearest neighbor or a k-nearest neighbors graph[3]. After graph construction, a graph-based active learning algorithm (S^2) path [15] accepts the graph $G = (\mathcal{D}, E)$ and selects the data points whose labels it would like to see. This selection is based on the structure of the graph and all previous gathered labels. Specifically, S^2 continually queries for the label of the vertex that bisects the shortest path between any pair of oppositely-labeled vertices. The authors in [15] show that S^2 can provably query and efficiently locate the cut-set in this graph (i.e., the edges of the graph that have oppositely labeled vertices). As a result, the query phase outputs a set $\tilde{\mathcal{D}}$ associated with the labels that is near the decision boundary.

Homology estimation phase: During the homology estimation stage, we construct an approximation of the LČ complex from the query set $\tilde{\mathcal{D}}$. Specifically, we construct the locally scaled labeled Vietoris-Rips (LS-LVR) complex introduced in [5]. Sticking to the query set $\tilde{\mathcal{D}}$ as an example, there are two steps to construct the LS-LVR complex: (1) Generate an initial graph from $\tilde{\mathcal{D}}$ by creating an edge set $\tilde{E}$ as follows: $\tilde{E} = \{\{\mathbf{x}_i, \mathbf{x}_j\} | (\mathbf{x}_i, \mathbf{x}_j) \in \tilde{\mathcal{D}}^2 \wedge y_i \neq y_j \wedge \|\mathbf{x}_i - \mathbf{x}_j\| \leq \kappa\sqrt{\rho_i \rho_j}\}$. Here, κ is a scale parameter, ρ_i is the smallest radius of a sphere centered at $\mathbf{x}_i$ to enclose k-nearest opposite class neighbors and ρ_j has a similar definition. This creates a bipartite graph where every edge connects points from opposite classes; (2) Connect all 2-hop neighbors to build a simplicial complex. Varying scale parameter κ produces a filtration of the simplicial complex, which can be used to estimate persistence diagrams that quantify the persistent homology.

3.2 Query Complexity of the Active Learning Algorithm

Let $G = (\mathcal{D}, E)$ denote a k-radius neighbor graph constructed from the dataset $\mathcal{D}$. This allows us to define the cut-set $C = \{(\mathbf{x}_i, \mathbf{x}_j) | y_i \neq y_j \wedge (\mathbf{x}_i, \mathbf{x}_j) \in E\}$ and cut-boundary $\partial C = \{\mathbf{x} \in V : \exists e \in C \text{ with } \mathbf{x} \in e\}$. We begin by sketching a structural lemma about the graph G and refer the reader to the Appendix D for a full statement and proof.

Lemma 2. *Suppose $\mathcal{D}^0$ and $\mathcal{D}^1$ are $\frac{\gamma}{2}$-dense in $\mathcal{M}$, then the graph $G = (\mathcal{D}, E)$ constructed from $\mathcal{D}$ is such that $\mathcal{D}^0 \bigcap \partial C$ and $\mathcal{D}^1 \bigcap \partial C$ are both $\frac{\gamma}{2}$-dense in $\mathcal{M}$ and $\partial C \subseteq \mathrm{Tub}_{w+\gamma}(\mathcal{M})$ for $k = \gamma$.*

$\mathcal{D}^0$ and $\mathcal{D}^1$ being $\frac{\gamma}{2}$-dense in $\mathcal{M}$ indicates the longest distance between $\mathbf{x}_i \in \mathfrak{D}^0 \bigcap B_{\frac{\gamma}{2}}(\mathbf{p})$ and $\mathbf{x}_j \in \mathfrak{D}^0 \bigcap B_{\frac{\gamma}{2}}(\mathbf{p})$ for $\mathbf{p} \in \mathcal{M}$ is γ. Therefore, letting $k = \gamma$ as Lemma 2 suggested results in both $\mathcal{D}^0 \bigcap \partial C$ and $\mathcal{D}^1 \bigcap \partial C$ being $\frac{\gamma}{2}$-dense in $\mathcal{M}$. Similar to Lemma 1, constructing a graph with a γ radius inevitably results in a subset of points in ∂C leaking out of $\mathrm{Tub}_w(\mathcal{M})$ and we formally have $\partial C \subseteq \mathrm{Tub}_{w+\gamma}(\mathcal{M})$. The key intuition behind our approach is that S^2 is naturally turned to focusing the labels acquired within $\mathrm{Tub}_{w+\gamma}(\mathcal{M})$. As we show below, this is done in a remarkably query efficient manner, and furthermore, when we obtain labeled data via querying we can construct an LČ complex; this allows us to find the homology of the manifold $\mathcal{M}$. We next need some structural assumptions about the manifold $\mathcal{M}$.

Assumption 3. ***(a)*** $\inf_{\mathbf{x} \in \mathcal{M}} \mu_{\mathcal{X}|y}(B_{\gamma/4}(\mathbf{x})) > \rho^y_{\gamma/4}, y \in \{0, 1\}$. ***(b)*** $\sup_{\mathbf{x} \in \mathcal{M}} \mu_{\mathcal{X}}(B_{(w+\gamma)}(\mathbf{x})) < h_{(w+\gamma)}$. ***(c)*** $\mu_{\mathcal{X}}(\mathrm{Tub}_{w+\gamma}(\mathcal{M})) \leq N_{w+\gamma} h_{w+\gamma}$.

Assumption 3(a) ensures sufficient mass in both classes such that $\mathcal{D}^0$ and $\mathcal{D}^1$ are $\frac{\gamma}{2}$-dense in $\mathcal{M}$. Assumption 3(b)(c) upper-bounds the measure of $\mathrm{Tub}_{w+\gamma}(\mathcal{M})$. Recall that $G = (\mathcal{D}, E)$ in Lemma 2

[3]The k-radius nearest neighbor graph connects all pairs of vertices that are a distance of at most k away, and the k-nearest neighbor graph connects a vertex to its k nearest neighbors

is a labeled graph; we further write β to denote the proportion of the smallest connected component with all the examples identically labeled. We lay out our main theorem as follows

Theorem 1. *Let $N_{w+\gamma}$ be the covering number of the manifold $\mathcal{M}$. Under Assumptions 1, 2 and 3, for any $\delta > 0$, we have that the (ϵ, γ)-LČ complex estimated by our framework is homotopy equivalent to $\mathcal{M}$ with probability at least $1 - \delta$ provided*

$$|\tilde{\mathcal{D}}| > \frac{\log\left\{1/\left[\beta\left(1-\sqrt{1-\delta}\right)\right]\right\}}{\log\left[1/(1-\beta)\right]} + |\mathcal{D}|N_{w+\gamma}h_{w+\gamma}(\lceil\log_2|\mathcal{D}|\rceil + 1) \tag{2}$$

where

$$|\mathcal{D}| > \max\left\{\frac{1}{P(y=0)\rho^0_{\gamma/4}}\left[\log\left(2N_{\gamma/4}\right) + \log\left(\frac{1}{(1-\sqrt{1-\delta})}\right)\right], \right. \\ \left. \frac{1}{P(y=1)\rho^1_{\gamma/4}}\left[\log\left(2N_{\gamma/4}\right) + \log\left(\frac{1}{(1-\sqrt{1-\delta})}\right)\right]\right\} . \tag{3}$$

Remark 1. *Theorem 1 demonstrates that our active learning framework has a query complexity of $\mathcal{O}(NN_{w+\gamma}h_{w+\gamma}log_2N)$. That is, after $\mathcal{O}(NN_{w+\gamma}h_{w+\gamma}log_2N)$ queries at most, a $(\epsilon, \gamma) - L\check{C}$ complex constructed from the queried examples will be homotopy equivalent to $\mathcal{M}$ with high probability. Notice that the intrinsic complexity of the manifold naturally plays a significant role, and the more complex the manifold the more significant gains the active learning framework has over its passive counterpart (cf. Eq. 3). In the Appendix E, we also provide a simple and concrete example that numerically shows the improvement in query complexity associated with our proposed framework relative to its passive counterpart.*

Remark 2. *The results of Theorem 1 can be improved by carrying out a more intricate analysis of the active learning algorithm as in [15]. Indeed, one may also replace the S^2 algorithm in our framework with a different graph-based active learning algorithm seamlessly to leverage the properties of that algorithm for active homology estimation of decision boundaries. These, and the relaxation of Assumption 2 , are promising directions for future work.*

Remark 3. *Parameters w and τ are intrinsic properties of p_{xy} and $\mathcal{M}$ and these properties are fixed to a classification problem. Variables γ and ϵ are algorithm variables and they are bounded as stated in Proposition 1.*

We provide a complete proof of Theorem 1 in the Appendix D. However, we will provide some intuition about the operation of our algorithm, and hence to the proof of the theorem here.

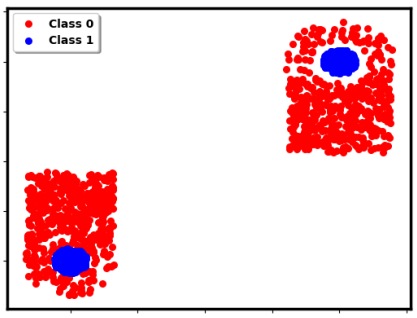


Figure 3: Visualization of the synthetic data.

The S^2 algorithm is split into two phases: uniform querying of labels and querying via path bisection. The uniform querying serves to finding a path connecting vertices of opposite labels. The path bisection phase queries at the mid-point of the shortest path that connects oppositely labeled vertices in the underlying graph. As the authors in [15] show, this endows S^2 with the ability to quickly narrow in on the cut-boundary ∂C. The uniform querying phase accounts for the first term in Eq. 2, which guarantees that there are sufficient paths to identify ∂C completely. During the path bisection phase, we take $(\lceil\log_2|\mathcal{D}|\rceil + 1)$ queries at most (this may be tightened using the techniques in [15]) to find the end point of the cut-edge inside a path; this needs to be done at most $|\partial C|$ to complete the querying phase. Next, with the Assumption 2 and the Lemma 2, it is guaranteed that $\partial C \subseteq \mathrm{Tub}_{w+\gamma}(\mathcal{M}) \subset \mathrm{Tub}_{(\sqrt{9}-\sqrt{8})\tau}(\mathcal{M})$ with γ properly selected following Proposition 1. Therefore, we may use the measure $N_{w+\gamma}h_{w+\gamma}$ from Assumption 3(b)(c) to upper-bound $|\partial C|$ which results in the second term of Eq. 2. This naturally ties the query complexity to the manifold complexity via $N_{w+\gamma}$ and $\mathrm{Tub}_{w+\gamma}(\mathcal{M})$. Eq. 3 comes from the necessary condition for the LČ complex being homotopy equivalent to $\mathcal{M}$, following the lines of [5].

4 Experimental Results

We compare the homological properties estimated from our active learning algorithm to a passive learning approach on both synthetic data and real data.

In the experiments we use the characteristics of homology group of dimension 1 (β_1, PD_1). We chose to use dimension 1 since [5] shows that this provides the best topological summaries for applications related to model selection. We have the performance evaluation for using the characteristics of homology group of dimension 0 (β_0, PD_0) presented in the Appendix F.

Using the synthetic data, we study the query complexity of active learning by examining the homological summaries β_1 and PD_1. For real data, we estimate PD_1 of the Banknote, MNIST and CIFAR10 and then utilize PD_1 to do model selection from several families of classifiers.

4.1 Experiments on Synthetic Data

The synthetic data in Figure 3 has decision boundaries that are homeomorphic to two disjoint circles. This dataset has 2000 examples. Clearly from Figure 3, β_1 of the decision boundary is two.

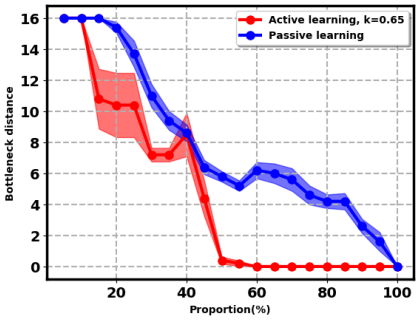


Figure 4: Bottleneck distance from ground-truth PD_1 by the passive learning and active learning.

Per the first step of our active learning algorithm, we construct a k-radius NN graph with $k = 0.65$. The scale parameter is set assuming we have full knowledge of the decision boundary manifold. Subsequently, we use S^2 to query the labels of examples on the created graph. After the label query phase, we construct the LS-LVR complex with the queried samples and compute β_1 and PD_1 using the python wrapper of the Ripser package [16, 17]. For the passive learning baseline, we uniformly query the examples with all other aspects of the experiment remaining identical to the active case. We also compute β_1 and PD_1 from the complete dataset and consider them as the "ground-truth" homology summaries. We evaluate the similarities between the estimated homology summaries and the ground-truth homology summaries to show the effectiveness of our active learning framework.

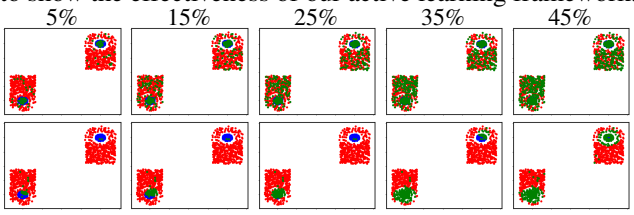


Figure 5: Visualization of the query process by passive learning (top row) and our active learning framework (bottom row) for different percentages of data labelling. More examples (highlighted by green) near the decision boundaries are selected to query in the proposed framework.

We compare the bottleneck distance [18, 19] between the ground-truth and estimated values of PD_1 for different percentages of data labelling. These results are shown on Figure 4. As is clear from the figure, the bottleneck distance for our active learning framework decreases faster than the passive learning approach and perfectly recovers the homology with only 50% of data. A visualization of the query process is shown on Figure 5. As expected, the active learning framework selects more examples to query near the decision. Please refer to the Appendix F to evaluate the performance of the active learning framework for different k-radius nearest neighbor graphs and β_1 recovery.

4.2 Experiments on Real Data

To demonstrate the effectiveness of our active learning framework on real data, we consider the classifier selection problem discussed in [5]. A bank of pretrained classifiers is accessible in the marketplace and customers select a proper one without changing the hyperparameters of the selected classifier. We consider two selection strategies as follow. One is topologically-based where the classifier with the smallest bottleneck distance from PD_1 of queried data is selected. The other one is to ensemble the topologically-selected classifier and the classifier selected based on the validation error of the queried data. We ensemble these two classifiers by averaging the output probabilities.

Banknote	KNN	SVM	Neural network	Decision tree
Passive	0.1072±0.0000	0.3753±0.0005	0.4316±0.0000	0.1997±0.0000
Active^1	0.0783±0.0014	0.3231±0.0012	0.4316±0.0000	0.1901±0.0004
Active^2	0.1017±0.0001	0.3431±0.0012	0.3730±0.0138	0.1744±0.0026
Active^3	**0.0346±0.0013**	**0.0836±0.0133**	**0.1058±0.0265**	**0.1613±0.0004**
Passive (ens)	0.0176±0.0000	**0.0259±0.0000**	0.0068±0.0000	0.0741±0.0000
Active^1 (ens)	0.0173±0.0000	**0.0259±0.0000**	**0.0039±0.0000**	**0.0731±0.0000**
Active^2 (ens)	**0.0149±0.0000**	**0.0259±0.0000**	0.0134±0.0001	**0.0731±0.0000**
Active^3 (ens)	**0.0149±0.0000**	**0.0259±0.0000**	0.0072±0.0000	0.0770±0.0000
MNIST	KNN	SVM	Neural network	Decision tree
Passive	0.0129±0.0000	**0.0141±0.0000**	0.0202±0.0000	**0.0332±0.0000**
Active^1	0.0128±0.0000	0.0161±0.0001	**0.0150±0.0000**	0.0388±0.0001
Active^2	0.0122±0.0000	0.0162±0.0001	0.0177±0.0000	**0.0332±0.0000**
Active^3	**0.0104±0.0000**	0.0156±0.0001	0.0388±0.0020	**0.0332±0.0000**
Passive (ens)	0.0119 ±0.0000	0.0124±0.0000	**0.0104±0.0000**	0.0290±0.0000
Active^1 (ens)	0.0123±0.0000	**0.0119±0.0000**	**0.0104±0.0000**	0.0284±0.0000
Active^2 (ens)	0.0108±0.0000	**0.0119±0.0000**	0.0125±0.0000	0.0284±0.0000
Active^3 (ens)	**0.0104±0.0000**	**0.0119±0.0000**	0.0127±0.0000	**0.0274±0.0000**
CIFAR10	KNN	SVM	Neural network	Decision tree
Passive	**0.3065±0.0002**	0.4683±0.0000	0.3185±0.0000	**0.3625±0.0000**
Active^1	0.3201±0.0000	0.4591±0.0005	**0.3058±0.0006**	**0.3625±0.0000**
Active^2	0.3095±0.0001	**0.4007±0.0038**	**0.3058±0.0006**	**0.3625±0.0000**
Active^3	0.3109±0.0001	0.4464 ±0.0005	0.3185±0.0000	**0.3625 ±0.0000**
Passive (ens)	0.2987 ±0.0001	0.2698±0.0000	0.2651±0.0001	**0.3137±0.0002**
Active^1 (ens)	0.2911 ±0.0001	0.2797±0.0000	**0.2558±0.0000**	0.3146±0.0000
Active^2 (ens)	0.2987±0.0001	0.2864±0.0003	0.2649±0.0001	0.3214±0.0005
Active^3 (ens)	**0.2935±0.0000**	**0.2665±0.0000**	0.2615±0.0001	0.3221±0.0004

Table 1: Average test error rates(five trials) on Banknote, MNIST and CIFAR10 for the model selected with 15% unlabelled pool data. Passive/Active stands for the non-ensemble classifiers selected by the $\mathbf{PD}_1$ homological similarities. Passive/Active (ens) stands for the classifiers ensembled from two classifiers: one is selected by the $\mathbf{PD}_1$ homological similarities and the other one is selected by the validation error. The subscript 1, 2 and 3 of the active learning indicates the used 3NN, 5NN and 7NN graphs. Best performance in the non-ensemble and ensemble cases are boldfaced.

We split the data to a training set, a test set and an unlabeled data pool. The training set is used to generate four different banks of classifiers: k-NN with k ranging from 1 to 29, SVM with polynomial kernel function degree ranging from 1 to 14, decision tree with maximum depth ranging from 1 to 27, and neural networks with the number of layers ranging from 1 to 6. The test set is used to evaluate the test error of each classifier. The unlabelled data pool is used to evaluate our active learning algorithm via selective querying.

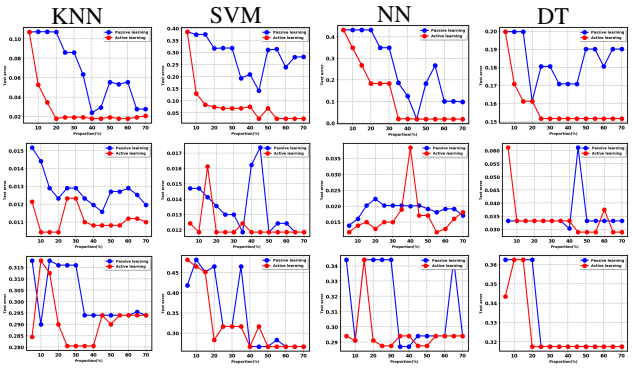


Figure 6: Test errors as a function of proportions of queried data on banknote (top), MNIST (middle) and CIFAR10 (bottom) in the model selection (non-ensemble) for different classifier families.

We use the proposed active learning framework to estimate the homological properties of the queried data: constructing a k-nearest neighbors graph, query examples by S^2 and computing the PD_1 with the queried examples. We set k =3, 5, and 7. For passive learning, we keep all the operations the same as the active learning framework except the queried examples are collected by uniform random sampling. To compute the PD of the decision boundary of the classifier, we simply use the test set input and the classifier output. Having estimated the homological summaries from the queried data and the classifiers, we compute the bottleneck distance between the PD_1 of the queried data and the classifiers. For the non-ensemble method, we simply select the classifier having the smallest bottleneck distance as a topologically-selected classifier. For the ensemble method, we further include

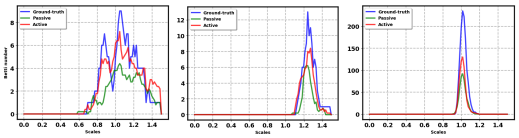
an additional classifier selected based on the validation error computed from the queried data and ensemble it with the topologically-selected classifier.

We implement the above procedure and evaluate on Banknote [20], MNIST [21] and CIFAR10 [22] datasets. Banknote contains 1372 instances in two classes with four input features for a binary classification task. We randomly sample 100 examples to construct the training set. Given the small size of the Banknote dataset, we use the remaining data as both the test set and unlabelled data pool. Although the test set and the data pool are not rigorously split in the Banknote case, it is still a fair comparison since the performance difference is only subject to the querying strategy. For the MNIST and the CIFAR10 datasets, we create (1) a 1 vs. 8 classification task from MNIST and (2) an automobile vs. ship classification task from CIFAR10. We randomly sample the data to create a training set with a sample size of 200, a test set with a sample size of 2000, and an unlabelled data pool with a sample size of 2000.

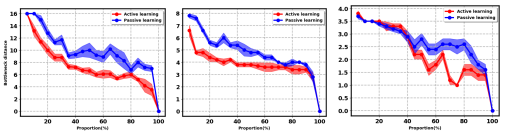
Table 3 shows the test error on banknote, MNIST and CIFAR10 for the classifier selected by querying 15% of the unlabelled data pool. We observe that the classifiers selected by our proposed active learning framework generally has a lower test error rate than the passive learning, especially in an ensemble classifier selection framework. As the experimental set-ups are identical (except for the querying strategy) we attribute the performance improvement to the proposed active learning framework used during model selection. Figure 6 indicates the performance of the non-ensemble classifiers selected by the homological similarities at the cost of the different proportions of the data pool. As expected, the proposed active learning framework achieves the best model selection faster than the passive learning for all classifier families. Note that the selection performance may be unstable with an increasing number of the queries since the active learning algorithm exhausts informative examples rapidly and begins to query noisy examples. In summary, Table 3 and Figure 6 indicate that the advantage of active learning in finding good homology summaries is also useful for model selection; this is evidenced by the lower error rates for the active learning approach relative to the passive learning approach.

4.3 Analysis of Homological Properties of Real Data

We present the homological properties estimated by passive learning and the proposed active learning framework. Similar to the experiments with the synthetic dataset, we access the complete unlabelled data pool and their labels to compute β_1 and PD_1 and use them as the ground-truth β_1 and PD_1. We then query the unlabelled data pool and estimate β_1 and PD_1 from the queried data. As we observe in the Figure 7(a), β_1 estimated by our active learning algorithm has a more similar trend to the ground-truth β_1 in all three real datasets. Furthermore, CIFAR10 has a significantly higher β_1 than MNIST and Banknote datasets indicating more complex decision boundaries. This is consistent with the Table 3 which shows that the error rates for the CIFAR10 binary classification tasks is higher than the other two datasets. Figure 7(b) shows the bottleneck distance between the estimated PD_1 and the ground-truth PD_1 for different proportions of labelled data. We observe that the active learning algorithm maintains a smaller bottleneck distance at early stages of querying. Such benefits gradually diminish as more of the data is labelled.



(a) β_1 estimation comparisons at the cost of 50% labelled data.



(b) Bottleneck distance for different proportions of labelled data.

Figure 7: Recovery of homological properties for the banknote (left), MNIST (middle) and CIFAR10 (right) using the active and passive learning approaches.

5 Conclusions

We propose an active learning algorithm to find the homology of decision boundaries. We theoretically analyze the query complexity of the proposed algorithm and prove the sufficient conditions to recover the homology of decision boundaries in the active learning setting. The extensive experiments on synthetic and real datasets with the application on model selection corroborate our theoretical results.

Broader Impact

The proposed approach, although has strong algorithmic and theoretical merits, has potential real-world application as we demonstrated.

One of the key uses of this approach is to create efficient summaries of decision boundaries of datasets [23] and models. Such summaries can be quite useful in applications like AI model marketplaces [24], where data and models can be securely matched without revealing too much information about each other. This is helpful in scenarios where the data is private and models are proprietary or sensitive.

A downside of being able to compute homology of decision boundaries with few examples is that malicious users may be able to learn about the key geometric / topological properties of the models with fewer examples than they would use otherwise. While this in itself may be benign, combined with other methods, they may be able to design better adversarial attacks on this model for instance. Ways of mitigating it in sensitive scenarios include ensuring that users do not issue too many queries of examples close to the boundary successively, since this may be revealing of malicious intent.

Acknowledgements

This work is funded in part by the Office of Naval Research under grant N00014-17-1-2826 and the National Science Foundation under grants OAC-1934766, CNS-2003081, and CCF-2007688.

Appendices

In Appendix A, we present the sufficient conditions for estimating the homology of a manifold. In Appendix B, by assuming the decision boundaries is a manifold, we present the sample complexity result to generate a simplical complex homotopy equivalent to the decision boundary manifold. This is the passive learning result that we extend to the active learning case. In Appendix C, we present the shortest shortest (S^2) path algorithm [15]. As a graph-based active learning algorithm for nonparametric classification, S^2 is used in the label query of our proposed active learning framework (see Figure 2 in the main document). After these three preliminary sections, in Appendix D, we provide a complete proof of Theorem 1 in the main content. In Appendix E, we provide numerical comparisons of the query complexity between the passive learning and our active learning algorithm for finding the homology of decision boundaries. Lastly, we present our complete experimental results in Appendix F.

A Sufficient Conditions for Finding the Homology of a Manifold

In [13], the authors show how to *learn* the homology of a manifold from samples. Specifically, [13] assumes the samples are generated from a constrained domain, and with these generated samples, [13] provide the sufficient conditions to learn the manifold the samples lie on or nearby. The sufficient conditions are used to find the homology of decision boundaries in both passive learning setting [5] and active learning setting of our work. Hence we describe those conditions in this section. To begin with, several assumptions need to be made.

Assumption A.1. *The manifold $\mathcal{M}$ has a condition number $1/\tau$.*

The quantity τ associated with $\mathcal{M}$ encodes both the local and global curvature of the manifold and is linked with the intrinsic complexity of the manifold $\mathcal{M}$: If $\mathcal{M}$ consists of several components, then τ bounds the separation between them. For example, as shown in Figure A.1, if $\mathcal{M}$ is a sphere, then τ is the radius of the sphere.

In the setting of [13], one has access to points that are near $\mathcal{M}$. Let $\mathcal{D} = (\mathbf{x}_1, ..., \mathbf{x}_N)$ denote a set of these sample points drawn from a feature space/domain $\mathcal{X}$. Furthermore, we define $\mu_{\mathcal{X}}$ as a standard Lebesgue measure on $\mathcal{X}$ and we write $Tub_r(\mathcal{M})$ to denote a tubular neighborhood with radius r around $\mathcal{M}$. As the goal is to learn $\mathcal{M}$ from samples, the domain $\mathcal{X}$ has to be *close* to $\mathcal{M}$ in some sense. To formalize this, the authors in [13] make the following assumption.

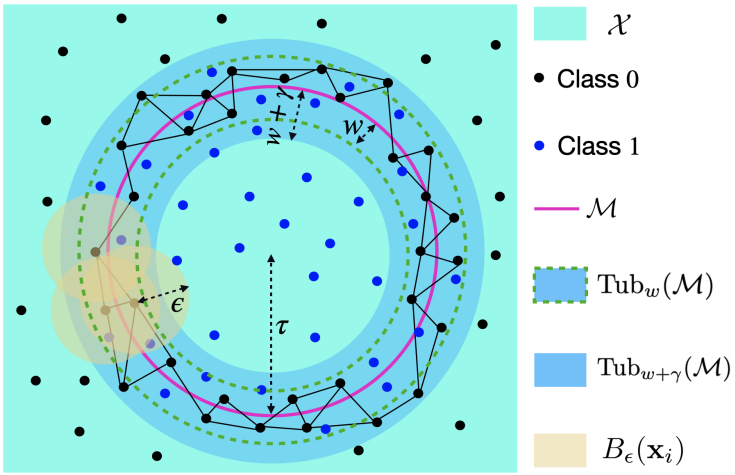


Figure A.1: An example of $(\epsilon, \gamma)-$labeled Čech complex, constructed in a tubular neighborhood $\mathrm{Tub}_{w+\gamma}(\mathcal{M})$ of radius $w+\gamma$, for a manifold $\mathcal{M}$ of condition number $1/\tau$. The overlap between the two classes is contained in $\mathrm{Tub}_w(\mathcal{M})$. The complex is constructed on samples in class 0, by placing balls of radius ϵ ($B_\epsilon(\mathbf{x}_i)$), and is "witnessed" by samples in class 1. $\mathcal{X}$ is the compact probability space for the data. Each triangle is assumed to be a $2-$simplex in the simplicial complex. Note that we keep the samples from both classes sparse for aesthetic reasons.

Assumption A.2. *The domain $\mathcal{X}$ is contained in a $\mathrm{Tub}_r(\mathcal{M})$ for $r < (\sqrt{9} - \sqrt{8})\tau$*

Assumption A.2 specifies that the generated samples are within at-most a distance of r to $\mathcal{M}$. In addition, we also define a notion of density on $\mathcal{M}$.

Definition A.1. *The set $\mathcal{D}$ is said to be $r-$dense in $\mathcal{M}$ if for every $\mathbf{p} \in \mathcal{M}$ there exists some $\mathbf{x} \in \mathcal{D}$ such that $\|\mathbf{p} - \mathbf{x}\|_2 < r$.*

This ensures a sufficient mass of samples are generated near $\mathcal{M}$. Typically, for learning a manifold $\mathcal{M}$ and then finding the homology of $\mathcal{M}$, one need to construct ϵ-balls centered at the points of $\mathcal{D}$ such that the union $U = \bigcup_{\mathbf{x}\in\mathcal{D}} B_\epsilon(\mathbf{x})$ deformation retracts to $\mathcal{M}$. [13] presents the sufficient conditions for having $U = \bigcup_{\mathbf{x}\in\mathcal{D}} B_\epsilon(\mathbf{x})$ be homotopy equivalent to $\mathcal{M}$:

Theorem A.1. *$\mathcal{M}$ is a deformation retract of $U = \bigcup_{\mathbf{x}\in\mathcal{D}} B_\epsilon(\mathbf{x})$ if **(a)** $r \leq (\sqrt{9} - \sqrt{8})\tau$, **(b)** $\epsilon \in \left(\frac{(r+\tau)-\sqrt{r^2+\tau^2-6\tau r}}{2}, \frac{(r+\tau)+\sqrt{r^2+\tau^2+6\tau r}}{2}\right)$, and **(c)** $\mathcal{D}$ is r-dense in $\mathcal{M}$.*

The proof of Theorem A.1 is under proposition 7.1 in [13]. Clearly, conditions (a) and (b) establish the relationship between ϵ, r and τ. Condition (c) ensures there are sufficient samples covering $\mathcal{M}$.

Remark A.1. *Theorem A.1 states, with appropriately constructed $B_\epsilon(\mathbf{x})$, one can find a $U = \bigcup_{\mathbf{x}\in\mathcal{D}} B_\epsilon(\mathbf{x})$ where $\mathcal{M}$ is a deformation retract. This naturally turns directly accessing the homolicial features of $\mathcal{M}$ to finding the homology of the auxiliary U. A practical way to do the above is construct the simplicial complex of the cover of U and then compute the homological properties from the constructed simplicial complex.*

One instructive way to think about the difference between the current paper and Niyogi *et al.*[13] is to think of the *sampling mechanism* of the data. The sampling mechanism in [13] essentially generates points on or very close to the manifold (depending on the amount of noise - see Assumption A.1). In our setting, however, we suppose that the feature data is supported on a much larger space $\mathcal{X}$, and there is no direct access to a manifold sampling mechanism. Instead, we have access to a *labeling oracle* which indirectly clues to us the location of the manifold. We therefore cannot make an assumption as strong as Assumption A.1, and we will relax this in Appendix B.

B The Manifold of Decision Boundaries

A recent line of work [9, 4] has emerged to find the homology of decision boundaries. The decision boundaries are simply taken as a manifold, and in this way, one can capture the holomoloical features of the decision boundaries by finding the homology of the related manifold. In the sequel, by referring to [5], we first describe the setting of a classification problem, then present a special simplicial complex called labeled Čech (LČ) complex, and finally present the sample complexity result in the passive learning setting through the LČ complex.

We consider a binary classification problem such that $\mathcal{D}$ and labels $y \in \{0, 1\}$ are drawn from joint distribution p_{XY}. For the distribution p_{XY}, we will let

$$\mathfrak{D} = \{\mathbf{x} \in \mathcal{X} : p_{X|Y}(\mathbf{x} \mid 1)p_{X|Y}(\mathbf{x} \mid 0) > 0\}.$$

In other words, $\mathfrak{D}$ denotes the region of the feature space where both classes overlap, i.e., both class conditional distributions $p_{X|Y}(\cdot \mid 1)$ and $p_{X|Y}(\cdot \mid 0)$ have non-zero mass. Similar to the notation $\mu_{\mathcal{X}}$, we write $\mu_{\mathcal{X}|y}$ to denote the measure for class y on $\mathcal{X}$. **From here on, we reuse the notation $\mathcal{M}$ to denote the manifold of the decision boundaries.** Specially, we define $\mathcal{M} = \{\mathbf{x} \in \mathcal{X} | p_{Y|X}(1|\mathbf{x}) = p_{Y|X}(0|\mathbf{x})\}$. The optimal decision boundaries is given by the classification function

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } p_{Y|X}(1 \mid \mathbf{x}) \geq 0.5 \\ 0 & \text{otherwise} \end{cases}.$$

Based on the observed data, we define the set $\mathcal{D}^0 = \{\mathbf{x} \in \mathcal{D} : f(\mathbf{x}) = 0\}$, that is the set of all samples with a Bayes optimal label of 0; similarly, we let $\mathcal{D}^1 = \{\mathbf{x} \in \mathcal{D} : f(\mathbf{x}) = 1\}$. Furthermore, we write $\mathrm{Tub}_w(\mathcal{M})$ to denote the smallest tubular neighborhood enclosing $\mathfrak{D}$.

Definition B.1. *Given $\epsilon, \gamma > 0$, an (ϵ, γ)-labeled Čech complex is a simplicial complex constructed from a collection of simplices such that each simplex σ is formed on the points in a set $S \subseteq \mathcal{D}^0$ witnessed by the reference set $\mathcal{D}^1$ satisfying the following conditions:* ***(a)*** *$\bigcap_{\mathbf{x}_i \in \sigma} B_\epsilon(\mathbf{x}_i) \neq \emptyset$, where $\mathbf{x}_i \in S$ are the vertices of σ.* ***(b)*** *$\forall \mathbf{x}_i \in S \subseteq \mathcal{D}^0$, $\exists \mathbf{x}_j \in \mathcal{D}^1$ such that, $\|\mathbf{x}_i - \mathbf{x}_j\|_2 \leq \gamma$.*

Definition B.1(a) ensures the LČ complex is constructed following the definitions of a typical Čech complex, that is, a set of ϵ ball centered at points of σ has a non-empty intersection [25]. Definition B.1(b) states that a subset S of $\mathcal{D}^0$ is selected such that S is at-most γ distance to $\mathcal{D}^1$. The authors in [5] show that, under certain assumption on the manifold and the distribution, provided that sufficiently many random samples (and their labels) are drawn according p_{XY} then $U = \bigcup_{\mathbf{x}_i \in \sigma} B_\epsilon(\mathbf{x}_i)$ is homotopy equivalent to $\mathcal{M}$. Therefore one could estimate the homologial features of the decision boundaries through the LČ complex. Herein, we introduce the assumptions and the lemmas before getting to the sample complexity result for the passive learning. These assumptions are standard in deriving theoretical results with respect to a manifold as explained in [13]. Hence they are also critical to the main theoretical results of this paper.

Note that the principal difference between [13] and our work is that [13] assumes all generated samples are contained within $\mathrm{Tub}_{(\sqrt{9}-\sqrt{8})\tau}(\mathcal{M})$ (See Assumption A.2); this is one of the sufficient conditions for manifold reconstruction from samples. In contrast, [5] considers a more practical scenario for classification where samples can be generated anywhere following the joint distribution p_{XY}. Our approach more closely follows [5], but with an important distinction. We sample from the marginal distribution p_X without knowledge of the underlying labels and use active learning to discover the labels. Nevertheless, our work and [5] share the assumptions and the lemmas introduced in the sequel. We also emphasize that, as we would show in the end, the follow-on results are dependent on two parameters that are specific to the joint distribution: w - the amount of overlap between the distributions and τ - the global geometric property of the decision boundary manifold.

Assumption B.1. $w < (\sqrt{9} - \sqrt{8})\tau$.

Since the generated samples do not necessarily reside within $\mathrm{Tub}_{(\sqrt{9}-\sqrt{8})\tau}(\mathcal{M})$, it is not immediately apparent that it is possible to find an S (see Definition B.1) that is entirely contained in $\mathrm{Tub}_{(\sqrt{9}-\sqrt{8})\tau}(\mathcal{M})$. However, Assumption B.1 allows us to guarantee precisely this. To see this, we will first state the following lemma.

Lemma B.1. *Provided $\mathcal{D}^0$ and $\mathcal{D}^1$ are both $\frac{\gamma}{2}$-dense in $\mathcal{M}$, then S is contained in $Tub_{w+\gamma}(\mathcal{M})$ and it is $\frac{\gamma}{2}$-dense in $\mathcal{M}$.*

Assumption B.1 is imposed on p_{XY} therefore there exists a S contained in $\mathrm{Tub}_{(\sqrt{9}-\sqrt{8})\tau}(\mathcal{M})$ with a proper γ. In fact, S can be partitioned to $S_{\mathfrak{D}} = S \bigcap \mathfrak{D}$ and $S \backslash S_{\mathfrak{D}}$. Under Assumption B.1, we immediately have $S_{\mathfrak{D}} \subseteq \mathrm{Tub}_{w}(\mathcal{M}) \subset \mathrm{Tub}_{(\sqrt{9}-\sqrt{8})\tau}(\mathcal{M})$. $S \backslash S_{\mathfrak{D}}$ is actually an excess from $\mathfrak{D}$ caused by identifying the necessary points $\mathbf{x} \in \mathcal{D}^0$ in the construction of LČ complex (see (b) in Definition B.1), and the exceeding extent is controlled by γ. With $\gamma < (\sqrt{9} - \sqrt{8})\tau - w$, we immediately have $S \subset \mathrm{Tub}_{(\sqrt{9}-\sqrt{8})\tau}(\mathcal{M})$. Besides, as the distance from $\mathbf{x}_i \in S \subseteq \mathfrak{D}^0$ to $\mathbf{x}_j \in \mathcal{D}^1$ is bounded by γ (see definition B.1(b)), S is also $\frac{\gamma}{2}$-dense in $\mathcal{M}$ therefore implying $(w+\gamma)$-dense in $\mathcal{M}$.

In fact, having $S \subset \mathrm{Tub}_{(\sqrt{9}-\sqrt{8})\tau}(\mathcal{M})$, S being $(w+\gamma)$-dense in $\mathcal{M}$ and ϵ properly selected are the sufficient conditions (see Theorem A.1) to construct a $U = \bigcup_{\mathbf{x}_i \in \sigma} B_\epsilon(\mathbf{x}_i)$ homotopy equivalent to $\mathcal{M}$. Remembering that the LČ complex is the cover of U, we have the following proposition

Proposition B.1. *(ϵ, γ)-LČ complex is homotopy equivalent to $\mathcal{M}$ as long as (a) $\gamma < (\sqrt{9} - \sqrt{8})\tau - w$; (b) $\mathcal{D}^0$ and $\mathcal{D}^1$ are $\frac{\gamma}{2}$-dense in $\mathcal{M}$ and (c) $\epsilon \in (\frac{(w+\gamma+\tau)-\sqrt{(w+\gamma)^2+\tau^2-6\tau(w+\gamma)}}{2}, \frac{(w+\gamma+\tau)+\sqrt{(w+\gamma)^2+\tau^2-6\tau(w+\gamma)}}{2})$.*

A pictorial description of relations between w, γ and τ is in Figure A.1 in the main content. In the stylized example in Figure A.1, $\mathcal{M}$ is a circle, $\mathrm{Tub}_{w+\gamma}(\mathcal{M})$ is an annulus and the radius ϵ of the covering ball $B_\epsilon(\mathbf{x})$ is constrained by τ.

In [5], the authors derive a sample complexity result in a passive learning setting such that a LČ complex is homotopy equivalent to $\mathcal{M}$. The heart of the derivation is figuring out how many samples are needed to have $\mathcal{D}^0$ and $\mathcal{D}^1$ both $\frac{\gamma}{2}$-dense in $\mathcal{M}$. We require an additional assumption:

Assumption B.2. $\inf_{\mathbf{x} \in \mathcal{M}} \mu_{\mathcal{X}|y}(B_{\gamma/4}(\mathbf{x})) > \rho^y_{\gamma/4}, y \in \{0, 1\}$.

The Assumption B.2 ensures sufficient mass in both classes such that $\mathcal{D}^0$ and $\mathcal{D}^1$ are $\frac{\gamma}{2}$-dense in $\mathcal{M}$. Given the Proposition B.1, this leads to the following theorem presented in Theorem 3 in [5].

Theorem B.1. *Let $N_{\gamma/4}$ be the covering number of the manifold $\mathcal{M}$. Under Assumptions A.1 B.1 and B.2, for any $\delta > 0$, we have that the (ϵ, γ)-LČ complex constructed from $\mathcal{D}_0$ and $\mathcal{D}_1$ is homotopy equivalent to $\mathcal{M}$ with probability at least $1 - \delta$ provided*

$$|\mathcal{D}| > \max\left\{ \frac{1}{P(y=0)\rho^0_{\gamma/4}} \left[\log\left(2N_{\gamma/4}\right) + \log\left(\frac{1}{(\delta)}\right) \right], \frac{1}{P(y=1)\rho^1_{\gamma/4}} \left[\log\left(2N_{\gamma/4}\right) + \log\left(\frac{1}{(\delta)}\right) \right] \right\} \tag{B.1}$$

The complete proof is elaborated in [5].

Remark B.1. *Considering the selection of γ needs to follow the Proposition B.1(b) as a function of τ and w, therefore given a probability factor δ, the sample complexity is dependent on w and τ.*

C Shortest Shortest Path Algorithm

We use an algorithm called shortest shortest (S^2) path [15] to query labels in the proposed framework. As a graph-based active learning algorithm for binary classification, S^2 has a properties of efficiently revealing the vertices near the cut-edges of a graph. Therefore, we apply S^2 to the label query stage of our proposed active learning framework (See Figure 2). The details of S^2 are presented in Algorithm 1. Repetitively using the defined notations, we let $G = (\mathcal{D}, E)$ denote a graph constructed from $\mathcal{D}$. This allows us to define the cut-set $C = \{(\mathbf{x}_i, \mathbf{x}_j) | y_i \neq y_j \wedge (\mathbf{x}_i, \mathbf{x}_j) \in E\}$ and cut-boundary $\partial C = \{\mathbf{x} \in \mathcal{D} : \exists e \in C \text{ with } \mathbf{x} \in e\}$. S^2 functions to efficiently identify ∂C by querying labels based on the structure of G. The query process is split to a uniform sampling phase and a path bisection phase. The uniform sampling serves to find a path connecting vertices of opposite labels. This corresponds to line 3 in Algorithm 1. The path bisection phase queries the mid-point of the shortest path that connects oppositely labeled vertices in the underlying graph. This corresponds to line 10 in Algorithm 1. The BUDGET in Algorithm 1 represents the cardinality of the query set $\tilde{\mathcal{D}}$. S^2 is designed in a consideration of running limited expensive number of label queries on vertices, as

Algorithm 1: S^2: Shortest Shortest Path

Input: Graph $G = (\mathcal{D}, E)$, BUDGET$\leq N$

$\tilde{\mathcal{D}} \leftarrow \emptyset$
while 1 **do**
  $\mathbf{x} \leftarrow$ Randomly chosen unlabeled vertex
  do
    Add $(\mathbf{x}, f(\mathbf{x}))$ to $\tilde{\mathcal{D}}$
    Remove all the found cut-edges of G
    if $|\tilde{\mathcal{D}}|$ = BUDGET **then**
      Return $\tilde{\mathcal{D}}$
    end if
  while $x \leftarrow$MSSP(G, $\tilde{\mathcal{D}}$) exists
end while

Procedure 1: MSSP: mid-point of the shortest shortest path

Input: $G = (\mathcal{D}, E)$, $\tilde{\mathcal{D}} \subseteq \mathcal{D}$

for each $\mathbf{x}_i, \mathbf{x}_j \in \tilde{\mathcal{D}}$ such that $f(\mathbf{x}_i) \neq f(\mathbf{x}_j)$
  $P_{ij} \leftarrow$ shortest path between $\mathbf{x}_i$ and $\mathbf{x}_j$ in G
  $\mathcal{L}_{ij} \leftarrow$ length of P_{ij} (∞ if no path exists)
end for
$i^*, j^* \leftarrow argmin_{\mathbf{x}_i, \mathbf{x}_j \in \tilde{\mathcal{D}}: f(\mathbf{x}_i) \neq f(\mathbf{x}_j)} \mathcal{L}_{ij}$
if (i^*, j^*) exists **then**
  Return mid-point of $P_{i^* j^*}$
else
  Return $\emptyset$
end if

a result, efficiently finding the vertices of cut-edges with the limited budget. Based on Theorem 1 in [15], we provide a simplified query complexity result for recovering ∂C:

Theorem C.1. *Suppose a graph $G = (\mathcal{D}, E)$ with a binary function $f : \mathcal{D} \to \{0, 1\}$ partitioning the graph G into two components identically labeled. Let β denote the proportion of the smallest components. Then for any $\delta > 0$, S^2 will recover C with probability at least 1 - δ if the complexity of queries is at least*

$$\frac{log(1/(\beta\delta))}{log(1/(1-\beta))} + |\partial C|(\lceil log_2|\mathcal{D}|\rceil + 1) \tag{C.1}$$

The query complexity is simplified by upper-bounding the path length with $|\mathcal{D}|$ and the complete proof can be found in [15].

D The Query Complexity Proof for the Proposed Active Learning Algorithm

The proof of Theorem 1 in our main content comprises of upper-bounding the query complexity to identify our target ∂C and upper-bounding the sample complexity of having $\mathcal{D}^0$ and $\mathcal{D}^1$ both $\frac{\gamma}{2}$-dense in $\mathcal{M}$. **We recap Theorem 1 in the main content as the Theorem D.1 in as follows and provide the complete proof.** Before we move to the proof, we make one last assumption,

Assumption D.1. ***(a)*** $\sup_{\mathbf{x} \in \mathcal{M}} \mu_{\mathcal{X}}(B_{(w+\gamma)}(\mathbf{x})) < h_{(w+\gamma)}$. ***(b)*** $\mu_{\mathcal{X}}(\mathrm{Tub}_{w+\gamma}(\mathcal{M})) \leq N_{w+\gamma} h_{w+\gamma}$.

Assumption D.1 upper-bounds the measure of $\mathrm{Tub}_{w+\gamma}(\mathcal{M})$. Besides, for the convenience of the theorem derivation, we use k-radius neighbor paradigm to construct $G = (D, E)$. We also need the following lemma,

Lemma D.1. *Suppose $\mathcal{D}^0$ and $\mathcal{D}^1$ are $\frac{\gamma}{2}$-dense in $\mathcal{M}$, then the graph $G = (\mathcal{D}, E)$ constructed from $\mathcal{D}$ is such that $\mathcal{D}^0 \bigcap \partial C$ and $\mathcal{D}^1 \bigcap \partial C$ are both $\frac{\gamma}{2}$-dense in $\mathcal{M}$ and $\partial C \subseteq \mathrm{Tub}_{w+\gamma}(\mathcal{M})$ for $k = \gamma$.*

$\mathcal{D}^0$ and $\mathcal{D}^1$ being $\frac{\gamma}{2}$-dense in $\mathcal{M}$ indicates the longest distance between $\mathbf{x}_i \in \mathcal{D}^0 \bigcap B_{\frac{\gamma}{2}}(\mathbf{p})$ and $\mathbf{x}_j \in \mathcal{D}^1 \bigcap B_{\frac{\gamma}{2}}(\mathbf{p})$ for $\mathbf{p} \in \mathcal{M}$ is γ. Therefore, letting $k = \gamma$ as Lemma D.1 suggests will result in $\mathcal{D}^0 \bigcap \partial C$ and $\mathcal{D}^1 \bigcap \partial C$ both being $\frac{\gamma}{2}$-dense in $\mathcal{M}$. Similar to Lemma B.1, constructing a graph with a γ radius inevitably results in a subset of points of ∂C leaking out of $\mathrm{Tub}_w(\mathcal{M})$ and we formally have $\partial C \subseteq \mathrm{Tub}_{w+\gamma}(\mathcal{M})$. Proposition B.1 states the proper choice of γ. With the above introduced, one can see the key intuition behind our approach is turning S^2 to focus on labels of points falling within $\mathrm{Tub}_{w+\gamma}(\mathcal{M})$. As we show below, this is done in a remarkably query efficient manner; when the labeled data is obtained we can construct an LČ complex and this allows us to find the homology of the manifold $\mathcal{M}$.

Theorem D.1. *Let $N_{w+\gamma}$ be the covering number of the manifold $\mathcal{M}$. Under Assumptions A.1 B.1 B.2 and D.1, for any $\delta > 0$, we have that the (ϵ, γ)-LČ complex estimated by our framework is homotopy equivalent to $\mathcal{M}$ with probability at least $1 - \delta$ provided*

$$|\tilde{\mathcal{D}}| > \frac{\log\left\{1/\left[\beta\left(1-\sqrt{1-\delta}\right)\right]\right\}}{\log\left[1/(1-\beta)\right]} + |\mathcal{D}|N_{w+\gamma}h_{w+\gamma}(\lceil\log_2|\mathcal{D}|\rceil + 1) \tag{D.1}$$

where

$$|\mathcal{D}| > \max\left\{\frac{1}{P(y=0)\rho^0_{\gamma/4}}\left[\log\left(2N_{\gamma/4}\right) + \log\left(\frac{1}{(1-\sqrt{1-\delta})}\right)\right], \right. \\ \left. \frac{1}{P(y=1)\rho^1_{\gamma/4}}\left[\log\left(2N_{\gamma/4}\right) + \log\left(\frac{1}{(1-\sqrt{1-\delta})}\right)\right]\right\} . \tag{D.2}$$

Proof. Let E_a denote an event that $\mathcal{D}^0$ and $\mathcal{D}^1$ are both $\frac{\gamma}{2}$-dense in $\mathcal{M}$. Let E_b denote an event that the LČ complex constructed from the query set $\tilde{\mathcal{D}}$ is homotopy equivalent to $\mathcal{M}$. Clearly E_b never happens if E_a does not happen due to not satisfying the condition (b) in Proposition B.1, and this results the conditional probability $P(E_b|\overline{E_a}) = 0$. Now, we expand the probability of E_b as follow:

$$\begin{aligned} P(E_b) &= P(E_b|E_a)P(E_a) + P(E_b|\overline{E_a})P(\overline{E_a}) \\ &= P(E_b|E_a)P(E_a) \end{aligned} \tag{D.3}$$

We first prove a query complexity result for the event $E_b|E_a$, i.e., how likely the event $E_b|E_a$ would happen with a certain amount of queries. Similarly, Eq. B.1 already provides a sample complexity result on the occurrence of the event E_a. With the Eq. D.3, we unify both complexity results and derive the theorem. We now consider $P(E_b|E_a)$. The query set $\tilde{\mathcal{D}}$ requires that $\tilde{\mathcal{D}}^0$ and $\tilde{\mathcal{D}}^1$ are $\frac{\gamma}{2}$-dense in $\mathcal{M}$ for E_b to happen. This can be surely achieved by constructing appropriate k-radius near neighbor/k nearest neighbor graph $G = (\mathcal{D}, E)$ stated by Lemma D.1, provided $\partial C \subseteq \tilde{\mathcal{D}}$. Hypothetically if a proper k for construction of G is selected, the event E_b becomes completely identifying the examples in ∂C through label querying. Theorem C.1 upper-bounds the query complexity of finding ∂C with a probably correct result. As the Assumption D.1 holds, we further upper-bound $|\partial C|$ by $|\partial C| \leq |\mathcal{D}|N_{w+\gamma}h_{w+\gamma}$. This gives us the query complexity of having $E_b|E_a$ happen:

$$|\tilde{\mathcal{D}}| > \frac{log(1/(\beta\alpha))}{log(1/(1-\beta))} + |\mathcal{D}|N_{w+\gamma}h_{w+\gamma}|log_2|\mathcal{D}| \tag{D.4}$$

with the probability at least $1-\alpha$. We now turn to $P(E_a)$. The occurrence of E_a is a dual implication of the LČ complex constructed from $\mathcal{D}$ being homotopy to $\mathcal{M}$. Therefore, reusing the result in Theorem B.1 (Eq. B.1), we get if

$$|\mathcal{D}| > max\left(\frac{1}{P(y=0)\rho^0_{\gamma/4}}\left(log\left(2N_{\gamma/4}\right) + log\left(\frac{1}{\eta}\right)\right), \frac{1}{P(y=1)\rho^1_{\gamma/4}}\left(log\left(2N_{\gamma/4}\right) + log\left(\frac{1}{\eta}\right)\right)\right) \tag{D.5}$$

then E_a happens with the probability at least $1-\eta$.

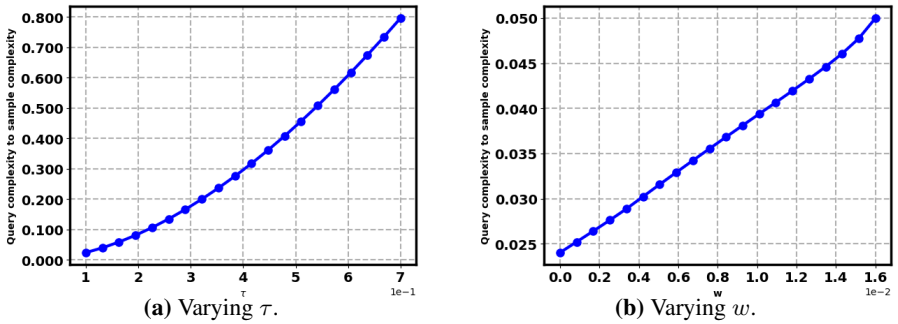
Picking $1-\alpha = \sqrt{1-\delta}$ and $1-\eta = \sqrt{1-\delta}$, we can unify the sample complexity results of $E_b|E_a$ and E_a to E_b and complete the proof. □

E Numerical Comparison for the Active Learning and Passive Learning Algorithms

As Eq. D.1 and Eq. B.1 directly provide the upper-bound query/sample complexity results of the active learning and passive learning methods, we can numerically compare the two methods. Herein, we provide a description of the evaluation; for implementation details we refer the reader to our code.

We created a stylized example illustrated in Figure A.1. We assume the feature space/domain $\mathcal{X}$ is a square area. In the domain $\mathcal{X}$, we draw samples generated from p_{XY} with a circular decision boundary of radius τ. We further use w to denote the radius of a smallest $\mathrm{Tub}_r(\mathcal{M})$ to enclose the overlap $\mathfrak{D}$ between two classes. Both τ and w are intrinsic properties of $\mathcal{M}$ and p_{XY}. Given these two properties, we set $\gamma = (\sqrt{9} - \sqrt{8})\tau - w - 10^{-5}$ to satisfy Proposition B.1(b). We make several additional assumptions regarding the problem in order to conduct the numerical experiments. Let us suppose that $\mathcal{X}$ is a square of 5×5 units. We assume $\mathcal{D}^0$ and $\mathcal{D}^1$ are uniformly distributed in the subspace $\mathcal{X}_0 \subseteq \mathcal{X}$ and subspace $\mathcal{X}_1 \subseteq \mathcal{X}$. Let $\mathcal{X}_0 \bigcap \mathcal{X}_1 = \mathrm{Tub}_w(\mathcal{M})$ and $\mathcal{X}_0 \bigcup \mathcal{X}_1 = \mathcal{X}$. Class-conditional distributions $p_{X|0}$ and $p_{X|1}$ are both uniform density functions in $\mathcal{X}_0$ and $\mathcal{X}_1$ such that class 0 and 1 completely overlap in $\mathrm{Tub}_w(\mathcal{M})$. Furthermore, we have $Area(\mathcal{X}_0) = Area(\mathcal{X}) - \pi(\tau - w)^2 = 25 - \pi(\tau - w)^2$ and $Area(\mathcal{X}_1) = \pi(\tau + w)^2$. Having the uniform probability density $d_0 = \frac{1}{Area(\mathcal{X}_0)}$ for class 0 and $d_1 = \frac{1}{Area(\mathcal{X}_1)}$ for class 1, we can easily compute the actual values of $h_{(w+\gamma)} = \mu_{\mathcal{X}}(B_{w+\gamma}(\mathbf{x}))$, $\rho^0_{\gamma/4} = \mu_{\mathcal{X}|0}(B_{\gamma/4}(\mathbf{x}))$ and $\rho^1_{\gamma/4} = \mu_{\mathcal{X}|1}(B_{\gamma/4}(\mathbf{x}))$ in Eq. B.1 and Eq. D.1 by simple algebra operations. $N_{\gamma/4}$ in Eq. B.1 indicates the cover number of $\mathcal{M}$ realized by $\frac{\gamma}{4}-$ balls. We simulate $N_{\gamma/4}$ by covering $\mathcal{M}$ with least number of $B_{\gamma/4}(\mathbf{x})$ on $\mathcal{M}$. The same operations can be applied to obtain $N_{w+\gamma}$ in Eq. D.1. β in Eq. D.1 indicates the proportion of the smallest component with the datapoints identically labelled in $G = (\mathcal{D}, E)$. For G constructed by the datapoints in our created stylized example, there are only two such components thus each component contains all the datapoints from class 0 or 1. Therefore, β is same as the mixture probability where $\beta = P(y = 1)$. We set $P(y = 1) = \frac{\pi\tau^2}{25}$ such that the probability accessing $\mathcal{M}$ by samples generated from p_{XY} increases with τ.

We compare the sample complexity results by fixing w and varying τ or fixing τ and varying w. For the case of fixing w, we vary τ from 0.1 to 0.7 and set $\delta = 0.1$ and $w = 10^{-10}$. For the case of fixing τ, on the other hand, we vary w from 10^{-10} to 1.75×10^{-2} and fix $\delta = 0.1$ and $\tau = 0.1$. Having w, τ and δ, we quantify other variables in Eq. D.1 and Eq. B.1 with the method described above and therefore acquire the query complexity for the active learning and the sample complexity for the passive learning. We calculate the ratio of the query complexity to the sample complexity and the results are shown in Figure E.1. As expected, the proposed active learning algorithm has a significant complexity gain compared to the passive learning case, especially for smaller values of τ and w.



(a) Varying τ. **(b)** Varying w.

Figure E.1: The ratio of query complexity to sample complexity by varying τ or w.

F Complete Experimental Results

We provide comprehensive performance results evaluated from using the characteristics of homology group of dimension 0 (β_0, PD_1) and dimension 1 (β_1, PD_1).

F.1 Experimental Results on Synthetic Data

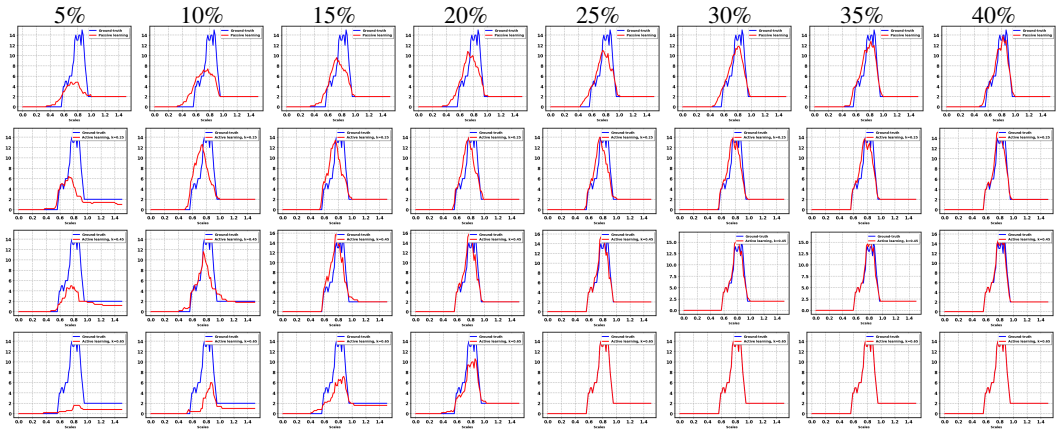


Figure F.1: β_0 estimated at the cost of different proportion of unlabelled data pool by the passive learning (first row) and active leanring methods with 0.25 (second), 0.45 (third) and 0.65 (forth) radius near neighbors graphs.

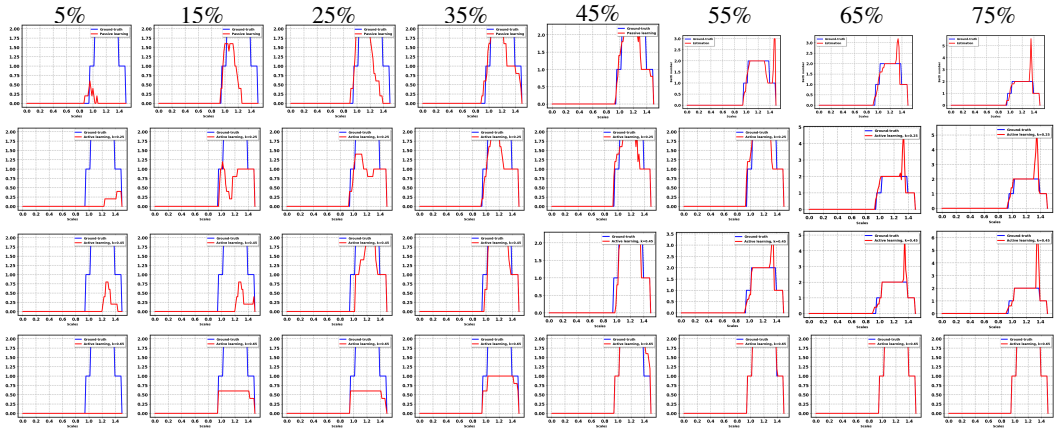
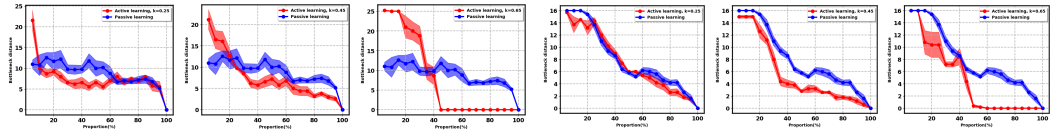


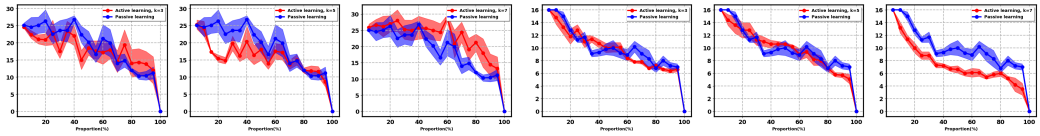
Figure F.2: β_1 estimated at the cost of different proportion of unlabelled data pool by the passive learning (first row) and active leanring methods with 0.25 (second), 0.45 (third) and 0.65 (forth) radius near neighbors graphs.



(a) Bottleneck distance from the ground-truth PD_0 **(b)** Bottleneck distance from the ground-truth PD_1

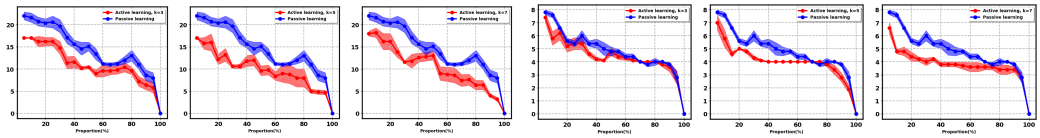
Figure F.3: Bottleneck distance from the ground-truths PD_0 and PD_1 by the passive learning and active learning on synthetic data. k indicates the values used in the k-radius near neighbor graphs for the proposed active learning algorithm.

F.2 Experimental Results on Real Data



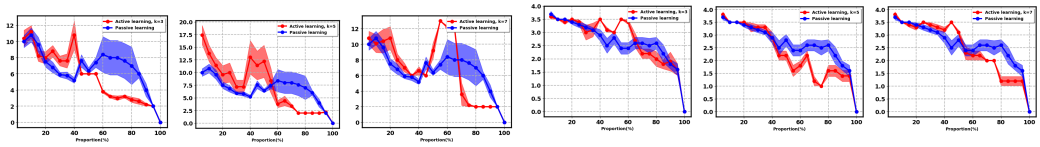
(a) Bottleneck distance from the ground-truth PD_0 **(b)** Bottleneck distance from the ground-truth PD_1

Figure F.4: Bottleneck distance from the ground-truths PD_0 and PD_1 by the passive learning and active learning on **Banknote**. k indicates the values used in the k-nearest neighbor graphs for the proposed active learning algorithm.



(a) Bottleneck distance from the ground-truth PD_0 **(b)** Bottleneck distance from the ground-truth PD_1

Figure F.5: Bottleneck distance from the ground-truths PD_0 and PD_1 by the passive learning and active learning on **MNIST**. k indicates the values used in the k-nearest neighbor graphs for the proposed active learning algorithm.



(a) Bottleneck distance from the ground-truth PD_0 **(b)** Bottleneck distance from the ground-truth PD_1

Figure F.6: Bottleneck distance from the ground-truths PD_0 and PD_1 by the passive learning and active learning on **CIFAR10**. k indicates the values used in the k-nearest neighbor graphs for the proposed active learning algorithm.

Banknote	KNN	SVM	Neural network	Decision tree
Passive	0.0508±0.0018	**0.1673±0.0077**	0.0223±0.0000	0.1491±0.0038
Active^1	0.0635±0.0012	0.3031±0.0011	**0.0157±0.0000**	0.1613±0.0004
Active^2	0.0689±0.0014	**0.1112±0.0109**	0.0676±0.0033	**0.0984±0.0026**
Active^3	**0.0193±0.0000**	0.1673±0.0077	0.0464±0.0021	0.1648±0.0024
Passive (ens)	0.0176 ±0.0000	**0.0259±0.0000**	0.0068 0.0000	0.0736±0.0000
Active^1 (ens)	0.0173±0.0000	**0.0259±0.0000**	**0.0039±0.0000**	**0.0731±0.0000**
Active^2 (ens)	**0.0149±0.0000**	**0.0259±0.0000**	0.0134±0.0001	**0.0731±0.0000**
Active^3 (ens)	**0.0149±0.0000**	**0.0259±0.0000**	0.0072±0.0000	0.0770±0.0000
MNIST	KNN	SVM	Neural network	Decision tree
Passive	0.0150±0.0000	0.0303±0.0000	0.1053±0.0000	0.0420±0.0003
Active^1	**0.0124±0.0000**	**0.0255±0.0001**	**0.0302±0.0004**	**0.0382±0.0001**
Active^2	0.0144±0.0000	0.0272±0.0001	0.0428±0.0012	0.0444±0.0002
Active^3	0.0138±0.0000	0.0303±0.0001	0.0506±0.0010	0.0448±0.0002
Passive (ens)	0.0127 ±0.0000	0.0124±0.0000	0.0137 0.0000	**0.0274±0.0000**
Active^1 (ens)	**0.0106±0.0000**	**0.0119±0.0000**	**0.0116±0.0000**	0.0284±0.0000
Active^2 (ens)	0.0121±0.0000	**0.0119±0.0000**	0.0138±0.0000	0.0284±0.0000
Active^3 (ens)	0.0110±0.0000	**0.0119±0.0000**	0.0138±0.0000	**0.0274±0.0000**
CIFAR10	KNN	SVM	Neural network	Decision tree
Passive	0.3049±0.0004	0.4309±0.0010	**0.2796±0.0009**	**0.3120±0.0000**
Active^1	0.3072±0.0004	**0.4055±0.0000**	0.2872±0.0006	**0.3120±0.0000**
Active^2	**0.2813±0.0000**	0.4182±0.0006	0.3042±0.0006	**0.3120±0.0000**
Active^3	**0.2813±0.0000**	0.4594±0.0008	0.3042±0.0006	**0.3120±0.0000**
Passive (ens)	0.2941±0.0002	**0.2698±0.0000**	0.2590 0.0001	**0.3074±0.0000**
Active^1 (ens)	0.2832±0.0000	0.2797±0.0000	**0.2558±0.0000**	**0.3074±0.0000**
Active^2 (ens)	**0.2813±0.0000**	0.2797±0.0000	**0.2558±0.0000**	0.3120±0.0000
Active^3 (ens)	**0.2813±0.0000**	0.2864±0.0003	0.2649±0.0001	0.3097±0.0000

Table 2: Average test error rates(five trials) on Banknote, MNIST and CIFAR10 for the model selected with 15% unlabelled pool data. Passive/Active stands for the non-ensemble classifiers selected by the $\mathbf{PD}_0$ homological similarities. Passive/Active (ens) stands for the classifiers ensembled from two classifiers: one is selected by the $\mathbf{PD}_0$ homological similarities and the other one is selected by the validation error. The subscript 1, 2 and 3 of the active learning indicates the used 3NN, 5NN and 7NN graphs. Best performance in the non-ensemble and ensemble cases are boldfaced.

Banknote	KNN	SVM	Neural network	Decision tree
Passive	0.1072±0.0000	0.3753±0.0005	0.4316±0.0000	0.1997±0.0000
Active^1	0.0783±0.0014	0.3231±0.0012	0.4316±0.0000	0.1901±0.0004
Active^2	0.1017±0.0001	0.3431±0.0012	0.3730±0.0138	0.1744±0.0026
Active^3	**0.0346±0.0013**	**0.0836±0.0133**	**0.1058±0.0265**	**0.1613±0.0004**
Passive (ens)	0.0176±0.0000	**0.0259±0.0000**	0.0068±0.0000	0.0741±0.0000
Active^1 (ens)	0.0173±0.0000	**0.0259±0.0000**	**0.0039±0.0000**	**0.0731±0.0000**
Active^2 (ens)	**0.0149±0.0000**	**0.0259±0.0000**	0.0134±0.0001	**0.0731±0.0000**
Active^3 (ens)	**0.0149±0.0000**	**0.0259±0.0000**	0.0072±0.0000	0.0770±0.0000
MNIST	KNN	SVM	Neural network	Decision tree
Passive	0.0129±0.0000	**0.0141±0.0000**	0.0202±0.0000	**0.0332±0.0000**
Active^1	0.0128±0.0000	0.0161±0.0001	**0.0150±0.0000**	0.0388±0.0001
Active^2	0.0122±0.0000	0.0162±0.0001	0.0177±0.0000	**0.0332±0.0000**
Active^3	**0.0104±0.0000**	0.0156±0.0001	0.0388±0.0020	**0.0332±0.0000**
Passive (ens)	0.0119 ±0.0000	0.0124±0.0000	**0.0104±0.0000**	0.0290±0.0000
Active^1 (ens)	0.0123±0.0000	**0.0119±0.0000**	**0.0104±0.0000**	0.0284±0.0000
Active^2 (ens)	0.0108±0.0000	**0.0119±0.0000**	0.0125±0.0000	0.0284±0.0000
Active^3 (ens)	**0.0104±0.0000**	**0.0119±0.0000**	0.0127±0.0000	**0.0274±0.0000**
CIFAR10	KNN	SVM	Neural network	Decision tree
Passive	**0.3065±0.0002**	0.4683±0.0000	0.3185±0.0000	**0.3625±0.0000**
Active^1	0.3201±0.0000	0.4591±0.0005	**0.3058±0.0006**	**0.3625±0.0000**
Active^2	0.3095±0.0001	**0.4007±0.0038**	**0.3058±0.0006**	**0.3625±0.0000**
Active^3	0.3109±0.0001	0.4464 ±0.0005	0.3185±0.0000	**0.3625 ±0.0000**
Passive (ens)	0.2987 ±0.0001	0.2698±0.0000	0.2651±0.0001	**0.3137±0.0002**
Active^1 (ens)	0.2911 ±0.0001	0.2797±0.0000	**0.2558±0.0000**	0.3146±0.0000
Active^2 (ens)	0.2987±0.0001	0.2864±0.0003	0.2649±0.0001	0.3214±0.0005
Active^3 (ens)	**0.2935±0.0000**	**0.2665±0.0000**	0.2615±0.0001	0.3221±0.0004

Table 3: Average test error rates(five trials) on Banknote, MNIST and CIFAR10 for the model selected with 15% unlabelled pool data. Passive/Active stands for the non-ensemble classifiers selected by the $\mathbf{PD}_1$ homological similarities. Passive/Active (ens) stands for the classifiers ensembled from two classifiers: one is selected by the $\mathbf{PD}_1$ homological similarities and the other one is selected by the validation error. The subscript 1, 2 and 3 of the active learning indicates the used 3NN, 5NN and 7NN graphs. Best performance in the non-ensemble and ensemble cases are boldfaced.

References

[1] J. Schmidhuber, "Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook," Ph.D. dissertation, Technische Universität München, 1987.

[2] L. Morán-Fernández, V. Bolón-Canedo, and A. Alonso-Betanzos, "Can classification performance be predicted by complexity measures? a study using microarray data," *Knowledge and Information Systems*, vol. 51, no. 3, pp. 1067–1090, 2017.

[3] G. Kusano, Y. Hiraoka, and K. Fukumizu, "Persistence weighted gaussian kernel for topological data analysis," in *International Conference on Machine Learning*, 2016, pp. 2004–2013.

[4] C. Chen, X. Ni, Q. Bai, and Y. Wang, "A topological regularizer for classifiers via persistent homology," *arXiv preprint arXiv:1806.10714*, 2018.

[5] K. N. Ramamurthy, K. Varshney, and K. Mody, "Topological data analysis of decision boundaries with application to model selection," vol. 97, pp. 5351–5360, 09–15 Jun 2019. [Online]. Available: http://proceedings.mlr.press/v97/ramamurthy19a.html

[6] B. Rieck, C. Bock, and K. Borgwardt, "A persistent weisfeiler-lehman procedure for graph classification," in *International Conference on Machine Learning*, 2019, pp. 5448–5458.

[7] W. H. Guss and R. Salakhutdinov, "On characterizing the capacity of neural networks using algebraic topology," *arXiv preprint arXiv:1802.04443*, 2018.

[8] B. Rieck, M. Togninalli, C. Bock, M. Moor, M. Horn, T. Gumbsch, and K. Borgwardt, "Neural persistence: A complexity measure for deep neural networks using algebraic topology," *arXiv preprint arXiv:1812.09764*, 2018.

[9] K. R. Varshney and K. N. Ramamurthy, "Persistent topology of decision boundaries," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 3931–3935.

[10] T. K. Ho, M. Basu, and M. H. C. Law, "Measures of geometrical complexity in classification problems," in *Data complexity in pattern recognition*. Springer, 2006, pp. 1–23.

[11] C. Hofer, R. Kwitt, M. Niethammer, and A. Uhl, "Deep learning with topological signatures," in *Advances in Neural Information Processing Systems*, 2017, pp. 1634–1644.

[12] J. Kim, J. Shin, F. Chazal, A. Rinaldo, and L. Wasserman, "Homotopy reconstruction via the cech complex and the vietoris-rips complex," in *The 36th International Symposium on Computational Geometry (SoCG 2020)*, 2020.

[13] P. Niyogi, S. Smale, and S. Weinberger, "Finding the homology of submanifolds with high confidence from random samples," *Discrete & Computational Geometry*, vol. 39, no. 1-3, pp. 419–441, 2008.

[14] H. Edelsbrunner and J. Harer, "Persistent homology - a survey," *Contemporary mathematics*, vol. 453, pp. 257–282, 2008.

[15] G. Dasarathy, R. Nowak, and X. Zhu, "S2: An efficient graph based active learning algorithm with application to nonparametric classification," in *Conference on Learning Theory*, 2015, pp. 503–522.

[16] U. Bauer, "Ripser: efficient computation of vietoris-rips persistence barcodes," Aug. 2019, preprint.

[17] N. Saul and C. Tralie, "Scikit-tda: Topological data analysis for python," 2019. [Online]. Available: https://doi.org/10.5281/zenodo.2533369

[18] A. Efrat, A. Itai, and M. J. Katz, "Geometry helps in bottleneck matching and related problems," *Algorithmica*, vol. 31, no. 1, pp. 1–28, 2001.

[19] M. Kerber, D. Morozov, and A. Nigmetov, "Geometry helps to compare persistence diagrams," *Journal of Experimental Algorithmics (JEA)*, vol. 22, pp. 1–20, 2017.

[20] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[21] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[22] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.

[23] H. A. Edwards and A. J. Storkey, "Towards a neural statistician," *ArXiv*, vol. abs/1606.02185, 2016.

[24] A. Bridgwater, "Enough Training, Let's Get Down To The AI Supermarket," *Forbes*, Sep 2018. [Online]. Available: https://www.forbes.com/sites/adrianbridgwater/2018/09/18/enough-training-lets-get-down-to-the-ai-supermarket/#5f13cdbc10c3

[25] R. W. Ghrist, *Elementary applied topology*. Createspace Seattle, 2014, vol. 1.