HiPoNet: A Multi-View Simplicial Complex Network for High Dimensional Point-Cloud and Single-Cell data

Siddharth Viswanath[*1] **Hiren Madhu**[*1] **Dhananjay Bhaskar**[1] **Jake Kovalic**[1]

David R. Johnson[2] **Christopher Tape**[3] **Ian Adelstein**[1] **Rex Ying**[1]

Michael Perlmutter[2] **Smita Krishnaswamy**[1]

[1] Yale University [2] Boise State University [3] University College London

[*]Equal Contribution Correspondence: smita.krishnaswamy@yale.edu

Abstract

In this paper, we propose HiPoNet, an end-to-end differentiable neural network for regression, classification, and representation learning on high-dimensional point clouds. Our work is motivated by single-cell data which can have very high-dimensionality – exceeding the capabilities of existing methods for point clouds which are mostly tailored for 3D data. Moreover, modern single-cell and spatial experiments now yield entire cohorts of datasets (i.e., one data set for every patient), necessitating models that can process large, high-dimensional point-clouds at scale. Most current approaches build a single nearest-neighbor graph, discarding important geometric and topological information. In contrast, HiPoNet models the point-cloud as a set of higher-order simplicial complexes, with each particular complex being created using a reweighting of features. This method thus generates multiple constructs corresponding to different views of high-dimensional data, which in biology offers the possibility of disentangling distinct cellular processes. It then employs simplicial wavelet transforms to extract multiscale features, capturing both local and global topology from each view. We show that geometric and topological information is preserved in this framework both theoretically and empirically. We showcase the utility of HiPoNet on point-cloud level tasks, involving classification and regression of entire point-clouds in data cohorts. Experimentally, we find that HiPoNet outperforms other point-cloud and graph-based models on single-cell data. We also apply HiPoNet to spatial transcriptomics datasets using spatial coordinates as one of the views. Overall, HiPoNet offers a robust and scalable solution for high-dimensional data analysis.

1 Introduction

High-dimensional point clouds, i.e., sets of points $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^n \subseteq \mathbb{R}^d$ now arise in many fields—most prominently single-cell analysis [55, 45, 12, 34, 1, 31], where using modern technologies such as mass cytometry or scRNA-seq, large cohorts of patients can now be measured producing several high-dimensional data. Further, technologies like perturb-seq enable the possibility of studying single-cell data under many conditions leading to comparable cohorts of datasets. Thus, machine learning techniques that once reasoned about data points and classified data points, now have to reason about collections of datasets. This serves as the motivation for HiPoNet illustrated in Figure 1.

Generally, methods like UMAP [36] or PHATE [38] reduce point clouds defined by single-cell data into an individual graph learned from all features (often in the tens of thousands). However, this single

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

graph may not specifically organize cells according to processes defined by subsets of dimensions. For instance, cells may be in a specific phase of the cell cycle which would be revealed by an organization based on cell cycle genes, or at a certain point in a differentiation process which would be revealed by stem cell genes. To address these limitations, HiPoNet utilizes multiple reweighted feature views. Additionally, existing neural network methods for point cloud data, such as PointNet [43] and its variants [44, 65, 58, 33] are primarily designed for 3D point clouds and rely on spatially localized features[1]. Hence, they are not well equipped to handle high-dimensional point clouds or disentangle processes from high-dimensional data. Therefore, there is a growing need to develop neural networks that can efficiently handle these diverse high-dimensional point clouds, ingest them seamlessly, encode the cellular processes, and perform various downstream machine learning tasks on them.

In HiPoNet, we use multiple views of the data, each of which is learned using a feature reweighting vector, thus potentially detangling and making implicit biological processes explicit. Furthermore, rather than modeling the data as a graph, we model it as a simplicial complex that captures higher-order relationships between cells, which HiPoNet analyzes using multiscale wavelets. Overall, we gain a rich representation that disentangles processes by capturing hierarchical relationships and separating overlapping biological processes, while preserving geometric and topological information of the underlying point clouds. Preserving this structure is important because it reflects the intrinsic geometry and topology of the data, providing a better interpretation of complex biological processes, and hence improving downstream analysis.

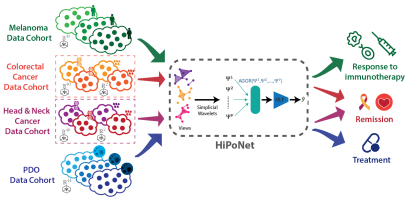

Figure 1: The HiPoNet pipeline.

To capture the global structure of the point clouds, we use a wavelet-based multiscale message aggregation scheme rather than the message passing operations utilized in most common graph and simplicial neural networks. This choice is based on the observation that such networks struggle to capture long-range dependencies and multiscale relationships as well as work on the geometric scattering transform [21, 41, 19, 20, 29, 59, 8, 62, 51] showing that wavelets may help remedy this issue. Although newer models like graph transformers [16] can capture long range dependencies, they often essentially ignore the graph geometry. When applied on these point clouds, they only work on a single graph and are do not preserve the intrinsic geometric structure of the point cloud. However, using diffusion wavelets, we are able to show theoretically that we capture the underlying geometry of the point cloud including the 0-homology (connected components) and curvature. Empirically we show that we can predict persistence homology directly from our simplicial neural network, without explicitly computing topological signatures or reducing the data to those features. These properties that are preserved by HiPoNet are essential in its ability to perform classification tasks, such as response to immunotherapy or drug treatment response. The key features of HiPoNet include:

- **Learning multiple views the the data:** We introduce a learnable scaling mechanism to reweight each feature via a weighting vector, which we then use to learn a graph to organize the data by the processes represented by the particular view.
- **Higher-order constructions:** We learn simplicial complexes from the data views. Our neural networks utilize these complexes and retain geometric and topological information including volume, curvature, connected components, and holes in the data.
- **Multiscale Message Aggregation:** Rather than simple message passing schemes that smooth data, we use multiscale simplicial wavelets to aggregate features over the simplicial complex and construct simplicial scattering coefficients. This allows us to extract both local and global information from the data without oversmoothing.

We demonstrate the utility of HiPoNet on three diverse single-cell *data cohorts*, each comprising ensembles of datasets collected from multiple patients or conditions. The first involves melanoma [42] patients undergoing immunotherapy, where the task is to predict treatment response from multiplex ion beam imaging (MIBI) with 61K cells from 54 patients. The second features patient-derived organoids [45] (PDOs) from 12 colorectal cancer patients, grown under various treatments and co-culture conditions; comprising of 1.8M cells, and the task is to identify the applied treatment. The third cohort uses spatial transcriptomics data from around 500 cancer biopsies [61], capturing 40

[1]We discuss these techniques more in Appendix B .

protein markers per cell through CODEX technology along with spatial arrangements, comprising of about 558K cells across three data cohorts. These datasets highlight the scalability of HiPoNet for complex, high-dimensional biological point clouds, where complexity arises from the number of datasets rather than the number of points per dataset

2 Background

In this section, we present the mathematical foundations needed for our method. We model point clouds $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ as simplicial complexes in order to capture higher-order geometric and topological structures. We will first introduce simplicial complexes and their algebraic representations, including boundary matrices and Laplacians. We then describe wavelet transforms and diffusion processes for extracting multi-scale topological features, and finally discuss random walks as efficient approximations of diffusion for scalable computation.

Given a finite set $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, a k-simplex, denoted as $\sigma_k = \{\mathbf{x}_{\sigma_k^0}, \mathbf{x}_{\sigma_k^1}, \ldots, \mathbf{x}_{\sigma_k^k}\}$, is a subset of $\mathcal{X}$ containing $k+1$ points. If τ_{k-1} is a subset of σ_k that contains k points, we say that τ_{k-1} is a face of σ_k. A simplicial complex $\mathcal{S}$ is a finite collection of simplices that satisfies the principle of inclusion [5, 49]. That is, we have $\tau_{k-1} \in \mathcal{S}$ whenever if τ_{k-1} is a face of σ_k for some $\sigma_k \in \mathcal{S}$. The order of a simplicial complex, K, is determined by the highest-order simplex contained within $\mathcal{S}$. Therefore a K-th order simplicial complex $\mathcal{S}$ contains simplices of orders $k = 0, 1, \ldots, K$. Observe that when $K = 1$, a simplicial complex is essentially equivalent to a graph, where the 0-simplices and 1-simplices correspond to the vertices and edges.

Features of simplices across all orders are denoted as $\mathbf{X} = \{\mathbf{X}_0, \mathbf{X}_1, \ldots, \mathbf{X}_K\}$, with $\mathbf{X}_k \in \mathbb{R}^{N_k \times D_k}$, where N_k is the number of k simplices and D_k is the feature dimension of k-simplices. We note that we can consider both unoriented simplices or oriented simplices, where the orientation is defined via a fixed ordering on the vertices, i.e., 0-complexes. (For details, please see [48] and the references within.) Except when otherwise specified, our theory and algorithms will apply to either case.

In a simplicial complex $\mathcal{S}$, a k-simplex $\sigma_k \in \mathcal{S}$ can have four types of neighbors:

- **Boundary adjacent neighbors or faces:** These are the $(k-1)$-simplices τ_{k-1} that are faces of σ_k, denoted as $\mathcal{N}_{\mathcal{B}}(\sigma_k) = \{\tau_{k-1} : \tau_{k-1} \subset \sigma_k\}$.
- **Coboundary adjacent neighbors or co-faces:** These are all $(k+1)$-simplices τ_{k+1} which have σ_k as a face, denoted as $\mathcal{N}_{\mathcal{C}}(\sigma_k) = \{\tau_{k+1} : \sigma_k \subset \tau_{k+1}\}$.
- **Lower adjacent neighbors:** These are simplices τ_k of the same order as σ_k that share a common face, ρ_{k-1}, represented as $\mathcal{N}_{\mathcal{L}}(\sigma_k) = \{\tau_k : \rho_{k-1} = \sigma_k \cap \tau_k,\ |\rho_{k-1}| = k-1\}$.
- **Upper adjacent neighbors:** These are simplices, τ_k of the same order as σ_k that are contained in a common $(k+1)$-simplex $\rho_{k+1} \in \mathcal{S}$, given by $\mathcal{N}_{\mathcal{U}}(\sigma_k) = \{\tau_k : \rho_{k+1} = \sigma_k \cup \tau_k,\ |\rho_{k+1}| = k+1\}$.

We let $\mathcal{N}_1(\sigma_k) = \cup\{\sigma_k, \mathcal{N}_{\mathcal{B}}(\sigma_k), \mathcal{N}_{\mathcal{C}}(\sigma_k), \mathcal{N}_{\mathcal{L}}(\sigma_k), \mathcal{N}_{\mathcal{U}}(\sigma_k)\}$ denote the 1-hop neighborhood of a simplex σ_k, and for $m \geq 2$, we define the m-hop neighborhood of a simplex σ_k is recursively as $\mathcal{N}_m(\sigma_k) = \cup_{\tau \in \mathcal{N}_{m-1}(\sigma_k)} \mathcal{N}_1(\tau)$ for $m \geq 2$.

The relationships between k-simplices and their faces are encoded in the boundary matrices, denoted as $\mathbf{B}_k \in \mathbb{R}^{N_{k-1} \times N_k}$, where N_k is the number of simplices in $\mathcal{S}$ of order k. The (i,j)-th entry of $\mathbf{B}_k$ is non-zero if the i-th $(k-1)$-simplex is a face of the j-th k-simplex. For oriented simplices, these entries $\mathbf{B}_k$ take values ± 1, reflecting their relative orientation. For unoriented simplices, all of the non-zero entries of $\mathbf{B}_k$ are equal to 1. We let $\mathbf{B} = \{\mathbf{B}_k\}_{k=1}^K$, denote the set of all boundary matrices.

We note that the boundary matrices are defined in terms of only boundary adjacent the and co-boundary adjacent neighbors. However, they may also be used to construct the Laplacians which encode information about lower and upper adjacent neighbors. Specifically, we define the lower and upper Laplcians by $\mathbf{L}_k^{\ell} = \mathbf{B}_k^{\top}\mathbf{B}_k$ and $\mathbf{L}_k^{u} = \mathbf{B}_{k+1}\mathbf{B}_{k+1}^{\top}$. We then define the k-Hodge Laplacian by:

$$\Delta_k = \mathbf{L}_k^{\ell} + \mathbf{L}_k^{u}, \tag{1}$$

(with $\mathbf{L}_0^{\ell} = \mathbf{L}_K^{u} = 0$). We note that for oriented simplices, Δ_0 is merely the standard graph Laplacian.

2.1 Heat Diffusion and random walks on Simplicial Complexes

The heat equation on k-simplices [27] is given by:

$$\frac{\partial u_k(\sigma_k, t)}{\partial t} = -\Delta_k u_k(\sigma_k, t). \tag{2}$$

Analogous to classical notions of heat diffusion, it describes how the distribution of heat propagates over the simplicial complex over time. The use of Δ_k ensures that the diffusion process respects the higher-order connectivity of the complex, encoding its topological structure.

The intuition behind much of our methods is to use the heat equation to uncover the topological structure of our point clouds and simplicial complexes, analogous to manifold learning algorithms such as Diffusion Maps [14]. However, in our actual algorithm, we will use random walk matrices, which we interpret as computationally efficient proxies for heat diffusion (similar to [14] which uses a diffusion matrix as a discrete approximation of the heat kernel on the underlying data manifold). Analogous to heat equation, random walks allow information to spread, i.e., diffuse, over the simplicial complex. However, they allow for efficient implementation via sparse matrix-vector multiplications.

We note that simplicial random walks extend the concept of traditional random walks from graphs to simplicial complexes, enabling a richer representation of the data which is able to encode higher-order features. Unlike standard graph-based random walks, which model transitions between vertices, simplicial random walks capture transitions between k-simplices, thereby incorporating higher-order interactions. Formally, the simplicial random walk is represented by the transition matrix $\mathbf{P}_k$, which describes the probability of transitioning between k-simplices. For unoriented simplices, we define:

$$\mathbf{P}_k = \Delta_k \mathbf{D}_k^{-1}, \tag{3}$$

where $\mathbf{D}_k = \mathrm{diag}(\Delta_k \mathbf{1})$ is a diagonal matrix that normalizes the transition probabilities, ensuring that the columns of $\mathbf{P}_k$ sum to one. In the case of oriented simplices, defining $\mathbf{P}_k$ is non-trivial since the entries of Δ_k may be either positive or negative. Here, for the sake of simplicity, we will define $\mathbf{P}_k$ the same way as in the unoriented case, but with first taking the entrywise absolute values of Δ_k, i.e., $\mathbf{P}_k = |\Delta_k| \mathbf{D}_k^{-1}$, $\mathbf{D}_k = \mathrm{diag}(|\Delta_k| \mathbf{1})$. Alternatively, in the case $k = 0$ or 1, one could define $\mathbf{P}_k$ in terms of a suitably normalized version of the Hodge Laplacian. (For details, please see [50].)

3 Methodology

We now describe the proposed method, HiPoNet, a High-dimensional POint cloud neural NETwork designed to learn expressive representations of an input point set $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\} \in \mathbb{R}^d$. As summarized in Algorithm 1, our framework comprises three primary steps:

1. **Multi-view creation via learnable feature weighting:** We learn multiple sets of weights $\alpha^{(v)}$, $1 \leq v \leq V$, to highlight different feature dimensions to create reweighted point clouds $\tilde{\mathcal{X}}^{(v)}$, which are effectively different views of the data.
2. **Simplicial complex-based point-cloud modeling:** From each $\tilde{\mathcal{X}}^{(v)}$, we construct a Vietoris-Rips complex $\mathcal{S}^{(v)}$ [57]. This approach allows us to capture not only pairwise relationships but also higher-order interactions within the data.
3. **Multi-scale Feature Extraction via Wavelets:** For each simplicial complex $\mathcal{S}^{(v)}$, we use wavelets to construct simplicial scattering coefficients which form a multi-scale embeddings $\Psi^{(v)}$, that capture both local and global relationships. These embeddings are then concatenated into a single representation Φ and are passed to a multilayer perceptron to yield the final predictions $\hat{\mathbf{y}}$.

By combining learnable multi-view feature weighting, simplicial complex construction, and simplicial wavelet transforms, HiPoNet leverages multiple perspectives of the underlying point cloud, at multiple scales to deliver robust representations for downstream classification and prediction.

Multi-view creation via learnable feature weighting: In many scenarios, such as scRNA-seq, feature vectors can encode critical properties, such as gene expression levels. However, not all dimensions (genes) contribute equally to downstream tasks, and irrelevant or noisy features can

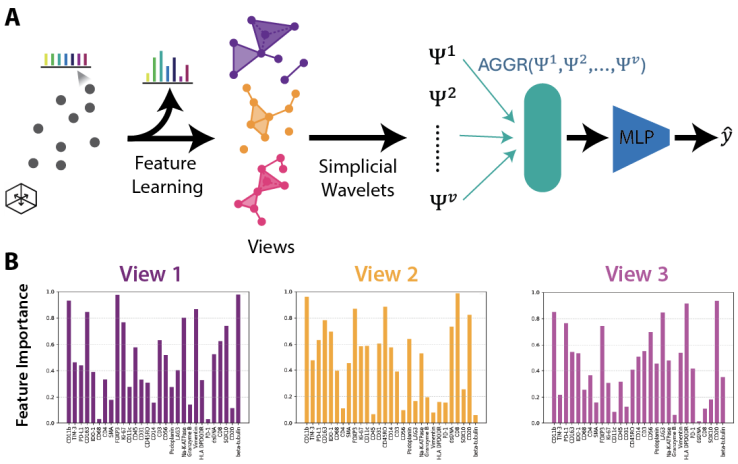

Figure 2: (A) The HiPoNet architecture. (B) Feature weight visualized across three learned views

Algorithm 1: HiPoNet

Input : Point cloud $\mathcal{X}$, kernel bandwidth σ, number of views V.
Learnable Parameters: learnable weights $\alpha^{(v)}$, MLP $\mathbf{z}$.

Output : Predictions $\hat{\mathbf{y}}$.

for $v = 1$ **to** V **do**

- $\tilde{\mathbf{x}}_i^{(v)} = \alpha^{(v)} \odot \mathbf{x}_i$ for $1 \le i \le n$;
- $\tilde{\mathcal{X}}^{(v)} = \{\tilde{\mathbf{x}}_i^{(v)}\}_{i=1}^n$;
- $\mathcal{S}^{(v)} = \text{Vietoris-Rips}(\tilde{\mathcal{X}}^{(v)}, \epsilon)$;
- Compute boundary matrices $\mathbf{B}^{(v)}$ from simplices in $\mathcal{S}^{(v)}$;
- Set $\mathbf{X}_0^{(v)} = \tilde{\mathcal{X}}^{(v)}$ `// Vertex features` ;
- **for** $k = 0$ **to** $K - 1$ **do**
  - $\mathbf{X}_{k+1}^{(v)} = (\mathbf{B}_k^{(v)})^\top \mathbf{X}_k^{(v)}$ `// Features for higher-order simplices`
- **for** $k = 0$ **to** K **do**
  - Compute transition matrix: $\mathbf{P}_k^{(v)} = \Delta_k^{(v)} \mathbf{D}_k^{-1}$;
  - **for** $j = 1$ **to** J **do**
    - $\Psi_k^{(v,j)} \mathbf{X}_k^{(v)} = \left(\left(\mathbf{P}_k^{(v)}\right)^{2^j} - \left(\mathbf{P}_k^{(v)}\right)^{2^{j-1}} \right) \mathbf{X}_k^{(v)}$;
  - Compute Scattering Coefficients:
  - $S_k^v[j]\mathbf{X}_k^{(v)} = |\Psi_k^{(v,j)} \mathbf{X}_k^{(v)}|$ and $S_k^v[j,j']\mathbf{X}_k^{(v)} = |\Psi_k^{(v,j')}|\Psi_k^{(v,j)} \mathbf{X}_k^{(v)}||$
- Aggregate features across simplex orders: $|S(\mathbf{B}^{(v)}, \mathbf{X}^{(v)})| = \left\{ |S_0^{(v)} \mathbf{X}_0^{(v)}|, \ldots, |S_K^{(v)} \mathbf{X}_K^{(v)}| \right\}$;

Final Aggregation and Prediction:

$\Phi = \text{Aggr}\{S(\mathbf{B}^{(1)}, \mathbf{X}^{(1)}), \ldots, S(\mathbf{B}^{(V)}, \mathbf{X}^{(V)})\}$;

$\hat{\mathbf{y}} = \mathbf{z}(\Phi)$;

return $\hat{\mathbf{y}}$

impair model performance, especially in high-dimensional settings. To address this, we introduce a learnable weighting mechanism that dynamically adjusts the weight of each feature dimension. Concretely, we define V distinct weight vectors $\alpha^{(v)}$, $1 \le v \le V$, each as a parameter of the model. For the v-th view, each point $\mathbf{x}_i$ is rescaled to

$$\tilde{\mathbf{x}}_i^{(v)} = \alpha^{(v)} \odot \mathbf{x}_i = [\alpha_1^{(v)} x_{i1},\ \alpha_2^{(v)} x_{i2}, \ldots, \alpha_d^{(v)} x_{id}],$$

where $\odot$ denotes the Hadamard (elementwise) product. We let $\tilde{\mathcal{X}}^{(v)} = \{\tilde{\mathbf{x}}_1^{(v)}, \tilde{\mathbf{x}}_2^{(v)}, \ldots, \tilde{\mathbf{x}}_n^{(v)}\}$ denote the reweighted point cloud. The reweighting operations amplify the more informative dimensions for the task at hand while downplaying less relevant features, effectively reducing the need for manual feature engineering. In the context of single-cell data, where the feature space can be vast,

such automated reweighting may be crucial for isolating key gene expressions that drive improved downstream performance. The learned weights can also be used for other biological insights, such as which gene markers are important for cancer diagnosis as illustrated in Figure 2B.

Simplicial learning from piont cloud data For each reweighted set $\tilde{\mathcal{X}}^{(v)}$, we create a simplicial complex $\mathcal{S}^{(v)}$. To do this, we first define kernelized distances between two points $\tilde{\mathbf{x}}_i^{(v)}$ and $\tilde{\mathbf{x}}_j^{(v)}$ as:

$$d_{i,j}^{(v)} = \exp\left(\frac{-\|\tilde{\mathbf{x}}_i^{(v)} - \tilde{\mathbf{x}}_j^{(v)}\|_2^2}{2\sigma^2}\right).$$

We then fix a scale parameter, ϵ, and construct the Vietoris-Rips Complex $\mathcal{S}^{(v)}$, a simplicial complex defined by the rule that $\sigma_k = \{\mathbf{X}_{i_0}^{(v)}, \mathbf{X}_{i_1}^{(v)}, \ldots, \mathbf{X}_{i_k}^{(v)}\}$ is an element of $\mathcal{S}^{(v)}$ if $d_{i_p,i_q}^{(v)} \leq \epsilon$ for all $1 \leq p, q \leq k$. After constructing the simpicial complex $\mathcal{S}^{(v)}$, we construct the boundary operators $\mathbf{B}^{(v)}$. We then define feature matrices by $\mathbf{X}_0^{(v)} = \tilde{\mathcal{X}}^{(v)}$ (i.e., the i-th row of $\mathbf{X}_0^{(v)}$ is $\tilde{\mathbf{x}}_i^{(v)}$) and then $\mathbf{X}_{k+1}^{(v)} = (\mathbf{B}_k^{(v)})^T \mathbf{X}_k^{(v)}$. In our code, we use a custom implementation of the Vietoris-Rips filtration that enables us to make simplicial construction differentiable, making HiPoNet end-to-end trainable.

We repeat this process V times, and construct $\mathbf{B}^{(v)}$ for each view $1 \leq v \leq V$. By constructing an ensemble of simplicial complexes $\mathcal{S}^{(1)}, \mathcal{S}^{(2)}, \ldots, \mathcal{S}^{(V)}$, our method integrates V distinct perspectives of the original high-dimensional point cloud. Constructing a simplicial complex $\mathcal{S}^{(v)}$ for each reweighted set $\tilde{\mathcal{X}}^{(v)}$ allows the model to disentangle and analyze distinct subspaces or biological processes captured by each view. For instance, each complex may focus on a different cellular process, capturing unique local and global structures that is overlooked when relying on a single representation. Consequently, subsequent learning stages benefit from a richer set of structural cues, enhancing the overall representation quality for downstream tasks.

Multi-scale feature extraction via simplicial wavelets After the simplicial complexes have been constructed, we can leverage geometric deep learning methods to compute representations for the point cloud. Although message passing neural networks (MPNNs) for simplicial complexes [6, 9, 23] have been introduced, they often suffer from oversmoothing effects [4], where representations become indistinguishable after multiple layers. In addition, oversquashing [2] is another constraint of such message-passing frameworks. As the receptive field of each node grows, large amounts of information from distant nodes are compressed or "squashed" into a fixed-size vector, leading to loss of important information between the nodes. This limits the ability of MPNNs to model long-range dependencies.

In order to address these shortcomings, we employ the simplicial wavelet transform [35, 46, 47] (SWT). The SWT is a multi-scale feature extractor for processing a signal $\mathbf{X}$ defined on the simplices. These wavelets are based on diffusion wavelets introduced in [13] which utilize random walk matrices at different time scales. In SWT, the random walk matrices are calculated first as per Eq 3. Then, features are iteratively aggregated over all the neighborhoods and stored for each scale j. Finally, a difference between different scales is taken to extract multiscale features.

The random walk diffusion matrix plays a crucial role in the SWT. For each view v, we interpret each $\left(\mathbf{P}_k^{(v)}\right)^j \mathbf{X}_k^{(v)}$ as diffused node features at scale j. Our wavelets, defined below, can be thought of as measuring the differences between these diffused features at different scales. Alternatively, from a signal processing perspective, these wavelets can be interpretted as a band-pass filters, with each j highlighting different frequency bands within the signal.

Formally, the simplicial wavelet transform at scale j, $0 \leq j \leq J$, is defined as the difference between diffusion at consecutive scales:

$$\Psi_k^{(v,j)} \mathbf{X}_k^{(v)} = \left(\left(\mathbf{P}_k^{(v)}\right)^j - \left(\mathbf{P}_k^{(v)}\right)^{j-1}\right) \mathbf{X}_k^{(v)}. \tag{4}$$

After applying SWT for multiple scales j, we apply a non-linearity, which we choose to be the absolute value $|\cdot|$, inspired by the geometric scattering transform [22, 19, 66, 41], a multilayered feedforward network constructed using diffusion wavelets on graphs. We then repeat this process with second wavelet, at a different scale $j' > j$, and then another absolute value. This yields first- and second-order *simplicial scattering coefficients* of the form $S_k^v[j]\mathbf{X}_k^{(v)} = |\Psi_k^{(v,j)} \mathbf{X}_k^{(v)}|$ and $S_k^v[j, j']\mathbf{X}_k^{(v)} =$

$|\Psi_k^{(v,j')}|\Psi_k^{(v,j)}\mathbf{X}_k^{(v)}||$. We then denote the collection of all first- and second-order scattering coefficients for each view by $|S(\mathbf{B}^{(v)}, \mathbf{X}^{(v)})| = \{S_k^v[j]\mathbf{X}_k^{(v)}\}_{\substack{0\le j\le J,\\ 1\le k\le K}} \cup \{S_k^v[j,j']\mathbf{X}_k^{(v)}\}_{\substack{0\le j\le j'\le J,\\ 1\le k\le K}}$. After extracting features for each view v, we aggregate the features over views as:

$$\Phi = \text{Aggr}\{S(\mathbf{B}^{(1)}, \mathbf{X}^{(1)}), S(\mathbf{B}^{(2)}, \mathbf{X}^{(2)}), \ldots, S(\mathbf{B}^{(V)}, \mathbf{X}^{(V)})\}.$$

Φ is then processed by a multilayer perceptron $\mathbf{z}$ to generate our final predictions, $\hat{\mathbf{y}} = \mathbf{z}(\Phi)$.

4 Theoretical results

In this section, we provide theoretical motivation for our model. Specifically, we first show that heat diffusion on simplices captures the 0-homology of the point cloud. Then, we show that simplicial complexes can be expanded into a graph, and the heat equation on the simplicial complex agrees with the heat equation on the equivalent graph. Then, we show that the diffusion operators can capture geodesic distances. Proofs, as well as more detailed theorem statements, are provided in the appendix.

Heat Diffusion and Connectivity. A simplicial complex is said to be connected if any two simplices can be linked by a sequence of simplices that share common faces. More formally, if for any two simplices σ and τ, there exists $m \geq 0$ such that $\tau \in \mathcal{N}_m(\sigma)$. If a simplicial complex is not connected, it can be decomposed into a union of disjoint connected components, each of which is a maximal connected subcomplex.

The following theorem demonstrates that the solution to the heat equation on a simplicial complex remains confined to the connected components where the initial condition is supported. This reflects the intuitive idea that heat cannot diffuse across disconnected regions of the complex. It also aligns with the concept of 0-homology in algebraic topology since the connected components of a simplicial complex correspond to the generators of its 0-homology group. Thus, the heat equation's confinement to connected components ensures that diffusion dynamics respect the 0-homology structure of the simplicial complex. The proof of Theorem 4.1 is available in Appendix C.

Theorem 4.1. *The heat equations, Eq 2, respect the* 0*-homology structure of the simplicial complex.*

Proof Sketch. The heat equation is governed by Δ_k, which acts locally by coupling neighbors. This ensures that heat diffusion cannot propagate between connected components. □

Heat Diffusion on Simplicial Graphs. In this section, we demonstrate that the solution to the heat equation on a simplicial complex $\mathcal{S}$ agrees with the solution to the heat equation on an associated simplicial graph, i.e., a graph where the vertices represent simplices of all orders and edges encode upper and lower adjacencies. This construction allows us to equate heat diffusion on the simplicial complex as heat diffusion on an associated graph, providing a simplified framework for analysis. The proof of Theorem 4.3 is in Appendix D.

Definition 4.2 (Simplicial Graph)**.** Let $\mathcal{S}$ be a simplicial complex. The associated simplicial graph $\mathcal{G}(\mathcal{S})$ is a graph whose vertices are given by $V(\mathcal{G}) = \mathcal{S}$ and whose edge set $E(\mathcal{G})$ consists of pairs of simplices which are either upper or lower adjacent, as defined in Section 2, that is,

$$E(\mathcal{G}) = \{\{\sigma, \sigma'\} \mid \sigma, \sigma' \in V(\mathcal{G}),\ \sigma' \in \mathcal{N}_{\mathcal{L}}(\sigma) \cup \mathcal{N}_{\mathcal{U}}(\sigma)\}.$$

Theorem 4.3. *The heat equation on a simplicial complex $\mathcal{S}$ agrees with the heat equation on the associated simplicial graph $\mathcal{G}(\mathcal{S})$, defined in terms of $\Delta_{\mathcal{G}} = B(\mathcal{G}(\mathcal{S}))B(\mathcal{G}(\mathcal{S}))^T$, where $B(\mathcal{G}(\mathcal{S}))$ is the incidence matrix of $\mathcal{G}(\mathcal{S})$.*

Proof Sketch. The graph Laplacian $\Delta_{\mathcal{G}}$ on the simplicial graph can be represented as a block diagonal matrix with the Hodge Laplacians Δ_k as the diagonal blocks. □

Heat Solution Captures Geometry. Most of our experiments focus on single-cell data which are known to satisfy the manifold hypotheses [28, 24, 17, 55, 1]. That is, the data points $\mathbf{x}_i$ lie upon some low-dimensional manifold. More specifically, we interpret each of the transformed point clouds $\tilde{\mathcal{X}}^{(v)}$ as corresponding to different submanifolds of some global cellular manifold. The following theorem establishes that the diffusion operator on the simplicial complex can approximate geodesic distances on these underlying manifolds. The proof of Theorem 4.4 is in Appendix E.

Theorem 4.4. *Assume that a transformed point cloud $\tilde{\mathcal{X}}^{(v)}$ lies upon a Riemannian manifold $\mathcal{M}$. Then, the 0-th order Hodge Laplacian Δ_0 on the associated oriented simplicial complex can approximate geodesic distances on the underlying manifold.*

Proof Sketch. The result follows from Varadhan's formula [53], which shows that distances may be computed via the heat kernel, as well as Theorem 3 of [15], which analyzes the convergence of the graph heat kernel to the manifold heat kernel as well as Theorem 4.3 which relates the graph heat equation to the simplicial complex heat equation. □

This theorem formalizes the link between diffusion processes, e.g., the heat equation, and the geometric structure of the data manifold. Notably, computation of geodesic distances on manifolds is computationally expensive and often infeasible in high dimensions. By using diffusion-based approximations, we can efficiently estimate geodesic distances with discrete operators. Additionally, we note that there is a long literature (dating back to at least [13]) relating the Laplacian and the diffusion operator to the geometry of the data manifold, including quantities such as curvature [7]. Most relevant to our work are the results which relate the asymptotic expansion of the trace of the heat kernel to geometric quantities like dimension, volume, and total scalar curvature (see for instance [37]). Indeed, following immediately from Theorems 4.3 and 4.4 we have:

Corollary 4.5. *The equivalence between the heat kernel on $\mathcal{S}$ and $\mathcal{G}(\mathcal{S})$ implies the equivalence of dimension, volume, and total scalar curvature on the respective underlying data manifolds.*

The proof of Corollary 4.5 is available in Appendix Section F. We use Weyl's law [60] and the eigenvalue comparison theorem [11] to prove the equivalence.

5 Empirical Results

We compare the performance of HiPoNet with KNN-GNNs (i.e., GNNs on KNN graphs) as well as PointNet++ and it's variants are described in Appendix H.1. The KNN-based graph neural networks (GCN [30], SAGE [25], GAT [54], GIN [63] and Graph Transformer [16], TopoGNN [26]) are presented in the first block, followed by point cloud and topology oriented models (DGCNN [58], PointNet++ [44], PointTransformer [65], TopoAE [39], GCNN [40], HGCNN [18]), and finally the proposed HiPoNet. We present the mean and standard deviation over 5 seeds[2]. A bolded score indicates the highest overall performance, whereas an underlined value identifies the second-best performance. Unless otherwise stated, we use unoriented simplices in our experiments. Details on the data cohorts considered in our experiments are provided in Appendix G. Hyperparameters are described in Appendix H. We provide a discussion of computational complexity in Appendix I.Discussion of limitations and societal impact is provided in Appendix Sections L and K, respectively.

Topology and Geometry Prediction. We evaluate HiPoNet on the task of predicting persistence features of the point clouds, which provides a topological summary of the dataset. The ground truth for persistence features was calculated by calculating using persistence diagrams. We note that the purpose of these experiments is not to develop a new method of computing persistence features. Instead, it is to show that HiPoNet has the expressivity needed in order to compute such features, indicating that it is a viable method for tasks which require a network to capture the underlying topology of the data.

The results in Table 1 indicate that HiPoNet outperforms baseline methods, achieving the lowest MSE across both datasets. Notably, KNN-based models perform competitively, with KNN-SAGE and KNN-GIN showing relatively lower errors compared to other GNN-based approaches. However, point-cloud-based models (DGCNN, PointNet++, and PointTransformer) exhibit significantly higher error values, suggesting that they struggle to capture the necessary

Model	Melanoma	PDO
KNN-GCN	1.0683 ± 0.002	1.0454 ± 0.018
KNN-SAGE	$\underline{0.734 \pm 0.031}$	1.0338 ± 0.010
KNN-GAT	1.101 ± 0.028	$\underline{1.068 \pm 0.008}$
KNN-GIN	0.850 ± 0.061	1.2605 ± 0.006
KNN-GraphTransformer	1.274 ± 0.038	1.3754 ± 0.006
DGCNN	28.412 ± 0.001	1353.0262 ± 1.12
PointNet++	28.418 ± 0.001	28.417 ± 0.005
PointTransformer	28.422 ± 0.014	28.41 ± 0.003
HiPoNet	$\mathbf{0.633 \pm 0.043}$	$\mathbf{0.4046 \pm 0.006}$

Table 1: MSE on prediction of persistence features.

[2]The code is available at `https://github.com/KrishnaswamyLab/HiPoNet`.

Data cohort	Nuumber of cells	Total # Datasets	Task	Data Modality
Melanoma patient samples	61K	54	Response to Immunotherapy	MIBI
Patient-derived organoids (PDOs)	1.8M	1625	Treatment Administered	Mass Cytometry
Charville	196K	196	Outcome to chemotherapy	CODEX
	196K	196	Cancer recurrence	
UPMC	308K	308	Outcome to chemotherapy	
	308K	308	Cancer recurrence	
DFCI	54K	54	Outcome to chemotherapy	

Table 2: Information about single-cell data

topological features for persistence feature prediction on these data sets, perhaps because they were designed with 3D point clouds in mind, rather than high-dimensional point clouds.

Single-cell data classification. Next we assess the performance of HiPoNet on classifiying point clouds from single-cell data cohorts, described in Table 2. (Detailed descriptions of this data is provided in the Appendix G.) As mentioned in the introduction, we emphasize that each data cohort is a measurement of thousands cells on a large cohort of patients and the label is prediction of outcome (of disease or treatment). As we can see in Table 4, HiPoNet outperforms all the graph-based, topological, and point cloud methods in Melanoma. On the PDO data, it is second to TopoGNN. However, HiPoNet is much more consistent than TopoGNN having a standard deviation of 0.94% in comparison to TopoGNNs standard deviation of 16.15%.

Application to spatial transcriptomics data. In addition to the experiments on single-cell data, we further demonstrate an application of HiPoNet in analyzing spatial transcriptomics data. Unlike previous experiments where multiple views were constructed in the same manner, here we construct two views by different methods: one capturing spatial proximity of cells and the other encoding gene expression similarity. Table 3 compares the predictive performance of various models on spatial transcriptomics data drawn from four different cohorts: DFCI, Charville, UPMC, and Melanoma. DFCI and Melanoma include one task-outcome prediction-while Charville and UPMC includes two tasks—either outcome prediction or recurrence prediction. Notably, HiPoNet achieves the top results in most settings, outperforming KNN-based graph neural networks (GCN, SAGE, GAT, GIN), PointNet++, and the KNN-GraphTransformer.

Data	DFCI	Charville		UPMC		Melanoma
Task	Outcome	Outcome	Recurrence	Outcome	Recurrence	Response
KNN-GCN	0.597 ± 0.049	0.547 ± 0.010	0.642 ± 0.056	$\mathbf{0.668 \pm 0.032}$	0.5 ± 0.0	0.606 ± 0.13
KNN-SAGE	0.82 ± 0.040	0.618 ± 0.021	0.581 ± 0.016	0.631 ± 0.013	0.5 ± 0.0	0.572 ± 0.05
KNN-GAT	0.557 ± 0.047	0.675 ± 0.051	0.530 ± 0.032	0.647 ± 0.029	0.5 ± 0.0	0.567 ± 0.05
KNN-GIN	0.700 ± 0.048	0.609 ± 0.020	0.624 ± 0.045	0.663 ± 0.015	0.514 ± 0.02	0.567 ± 0.06
KNN-GraphTransformer	0.668 ± 0.051	0.578 ± 0.040	0.5 ± 0.0	0.629 ± 0.00	0.5 ± 0.0	0.528 ± 0.04
PointNet++	0.491 ± 0.057	0.451 ± 0.078	0.499 ± 0.001	0.506 ± 0.01	0.495 ± 0.00	0.503 ± 0.00
HiPoNet	$\mathbf{0.916 \pm 0.03}$	$\mathbf{0.681 \pm 0.012}$	$\mathbf{0.681 \pm 0.01}$	0.665 ± 0.01	$\mathbf{0.6044 \pm 0.0}$	$\mathbf{0.732 \pm 0.01}$

Table 3: AUC ROC on Spatial Transcriptomics Classification

Intepretability of Learned Features. HiPoNet offers high degree of interpretability enabled by its multi-view and learnable feature weighting architecture that learns meaningful representations from high-dimensional point cloud data. As seen in the three bar plots in Figure 2(B), each plot corresponds to a different learned view or representation of the 30 protein expression markers in the melanoma single-cell classification task, with the y-axis including the learning importance of each marker in that specific view. Biologically meaningful markers like CD11b, CD118, and FOXP3 have consistently high importance across the different views, aligning with their roles in immune regulation within the tumor microenvironment and promoting immune evasion in tumors.

Model	**Melanoma**	**PDO**
KNN-GCN	72.72 ± 5.45	53.66 ± 0.70
KNN-SAGE	76.36 ± 6.03	55.89 ± 0.83
KNN-GAT	61.81 ± 4.45	52.56 ± 10.06
KNN-GIN	85.45 ± 3.63	57.02 ± 1.20
KNN-GraphTransformer	80.00 ± 8.90	39.46 ± 2.78
TopoGNN	88.18 ± 8.19	$\mathbf{79.90 \pm 16.15}$
HGNN	80.00 ± 7.6	38.76 ± 0.87
DGCNN	63.33 ± 12.47	40.00 ± 4.36
PointNet++	45.00 ± 22.42	45.24 ± 2.08
PointTransformer	79.99 ± 6.24	30.00 ± 4.70
TopoAE	52.73 ± 11.35	15.71 ± 0.87
GCNN	63.63 ± 0.0	29.34 ± 1.03
HiPoNet	$\mathbf{90.90 \pm 4.92}$	77.38 ± 0.94

Table 4: Accuracies on classification tasks.

Mitigation of Oversmoothing. To empirically examine whether our modeling choice mitigates oversmoothing, we conducted a comparative analysis of Dirichlet energy across various methods. As established by [10], Dirichlet energy provides a principled measure of the expressiveness of node representations, where lower values indicate oversmoothed embeddings, and higher values reflect

greater feature variation across graph neighborhoods. We computed the Dirichlet energy of node representations produced by different models.

As seen in Table 5, results indicate that our model, HiPoNet, exhibits substantially higher Dirichlet energy than traditional message-passing networks. This supports the claim that HiPoNet mitigates oversmoothing and retains richer, more expressive node-level information. Notably, the GCN and diffusion-based models exhibit particularly low energy, consistent with oversmoothed representations. Thus, this empirical evidence reinforces the motivation for using wavelet transforms, which allow HiPoNet to capture multi-scale variation and higher-frequency components in the signal that are typically suppressed in standard GNN architectures.

Model	Dirichlet Energy
GCN	1.669 ± 0.13
GAT	2.694 ± 0.49
Graph Transformer	3.811 ± 0.80
Diffusion-based GNN	0.584 ± 0.00
GWT	15.807 ± 0.00
HiPoNet (ours)	$\mathbf{21.033 \pm 6.25}$

Table 5: Dirichlet Energy across different models.

Ablations. We provide ablations in Appendix J. Table A4 shows that increasing the number of views improves performance, with four views yielding the best results. In Table A5, we observe that removing multi-view learning or simplicial modeling leads to the largest performance drops, confirming their critical importance. Table A6 presents a sensitivity analysis of the Vietoris–Rips threshold, showing that the optimal threshold varies by dataset and significantly affects accuracy. In Table A7, we analyze the effect of kernel bandwidth, finding that careful tuning is required to avoid oversmoothing. Finally, Table A8 demonstrates that while 1st order simplices (essentially graphs) are sufficient for some data sets, incorporating higher-order simplices improves performance for others, particularly in predicting clinical outcomes.

6 Conclusion

In this work, we introduced HiPoNet, a novel neural network architecture designed for high-dimensional point cloud datasets. By leveraging multiple views, higher-order constructs, and multi-scale wavelet transforms, HiPoNet effectively captures complex geometric and topological structures inherent in biological data. Our experiments demonstrate that HiPoNet learns distinct, meaningful representations tailored to each dataset and significantly improves downstream analysis across diverse single-cell and spatial transcriptomics cohorts. These results highlight the importance of integrating higher-order and multi-scale information to overcome the limitations of existing point cloud and graph-based models.

7 Acknowledgements

D.B. received funding from the Yale - Boehringer Ingelheim Biomedical Data Science Fellowship and the Kavli Institute for Neuroscience Postdoctoral Fellowship. M.P. acknowledges funding from The National Science Foundation under grant number OIA-2242769. S.K. is funded in part by the NIH (NIGMSR01GM135929, R01GM130847), NSF CAREER award IIS-2047856, NSF IIS-2403317, NSF DMS-2327211 and NSF CISE-2403317. S.K is also funded by the Sloan Fellowship FG-2021-15883, the Novo Nordisk grant GR112933. S.K. and M.P. also acknowledge funding from NSF DMS-2327211.

References

[1] Constantin Ahlmann-Eltze and Wolfgang Huber. Analysis of multi-condition single-cell data with latent embedding multivariate regression. *Nature Genetics*, Jan 2025. ISSN 1546-1718. doi: 10.1038/s41588-024-01996-0. URL `https://doi.org/10.1038/s41588-024-01996-0`.

[2] Uri Alon and Eran Yahav. On the bottleneck of graph neural networks and its practical implications. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=i80OPhOCVH2`.

[3] Michael Angelo, Sean C Bendall, Rachel Finck, Matthew B Hale, Chuck Hitzman, Alexander D Borowsky, Richard M Levenson, John B Lowe, Scot D Liu, Shuchun Zhao, et al. Multiplexed ion beam imaging of human breast tumors. *Nature medicine*, 20(4):436–442, 2014.

[4] Muhammet Balcilar, Guillaume Renton, Pierre Héroux, Benoit Gaüzère, Sébastien Adam, and Paul Honeine. Analyzing the expressive power of graph neural networks in a spectral perspective. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=-qh0M9XWxnv.

[5] Sergio Barbarossa and Stefania Sardellitti. Topological signal processing over simplicial complexes. *IEEE Transactions on Signal Processing*, 68:2992–3007, 2020. ISSN 1941-0476. doi: 10.1109/tsp.2020.2981920. URL http://dx.doi.org/10.1109/TSP.2020.2981920.

[6] Claudio Battiloro, Lucia Testa, Lorenzo Giusti, Stefania Sardellitti, Paolo Di Lorenzo, and Sergio Barbarossa. Generalized simplicial attention neural networks, 2024. URL https://arxiv.org/abs/2309.02138.

[7] Dhananjay Bhaskar, Kincaid MacDonald, Oluwadamilola Fasina, Dawson Thomas, Bastian Rieck, Ian Adelstein, and Smita Krishnaswamy. Diffusion curvature for estimating local curvature in high dimensional data, 2022. URL https://arxiv.org/abs/2206.03977.

[8] Bernhard G Bodmann and Íris Emilsdóttir. A scattering transform for graphs based on heat semigroups, with an application for the detection of anomalies in positive time series with underlying periodicities. *Sampling Theory, Signal Processing, and Data Analysis*, 22(2):17, 2024.

[9] Cristian Bodnar, Fabrizio Frasca, Yu Guang Wang, Nina Otter, Guido Montúfar, Pietro Liò, and Michael M. Bronstein. Weisfeiler and lehman go topological: Message passing simplicial networks. *CoRR*, abs/2103.03212, 2021. URL https://arxiv.org/abs/2103.03212.

[10] Chen Cai and Yusu Wang. A note on over-smoothing for graph neural networks, 2020. URL https://arxiv.org/abs/2006.13318.

[11] Shiu-Yuen Cheng. Eigenvalue comparison theorems and its geometric applications. *Mathematische Zeitschrift*, 1975. URL https://doi.org/10.1007/BF01214381.

[12] Joyce Chew, Holly Steach, Siddharth Viswanath, Hau-Tieng Wu, Matthew Hirn, Deanna Needell, Matthew D. Vesely, Smita Krishnaswamy, and Michael Perlmutter. The manifold scattering transform for high-dimensional point cloud data. In *Proceedings of Topological, Algebraic, and Geometric Learning Workshops 2022*, volume 196 of *Proceedings of Machine Learning Research*, pages 67–78. PMLR, 25 Feb–22 Jul 2022. URL https://proceedings.mlr.press/v196/chew22a.html.

[13] R. Coifman and M. Maggioni. Diffusion wavelets. *Appl. Comp. Harm. Anal.*, 21(1):53–94, 2006.

[14] Ronald R Coifman and Stéphane Lafon. Diffusion maps. *Applied and computational harmonic analysis*, 21(1):5–30, 2006.

[15] David B Dunson, Hau-Tieng Wu, and Nan Wu. Spectral convergence of graph laplacian and heat kernel reconstruction in l^∞ from random samples. *Applied and Computational Harmonic Analysis*, 55:282–336, 2021.

[16] Vijay Prakash Dwivedi and Xavier Bresson. A generalization of transformer networks to graphs. *CoRR*, abs/2012.09699, 2020. URL https://arxiv.org/abs/2012.09699.

[17] Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. Testing the manifold hypothesis. *Journal of the American Mathematical Society*, 29(4):983–1049, 2016.

[18] Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. Hypergraph neural networks, 2019. URL https://arxiv.org/abs/1809.09401.

[19] Fernando Gama, Alejandro Ribeiro, and Joan Bruna. Diffusion scattering transforms on graphs. *arXiv preprint arXiv:1806.08829*, 2018.

[20] Fernando Gama, Alejandro Ribeiro, and Joan Bruna. Stability of graph scattering transforms. *Advances in Neural Information Processing Systems*, 32, 2019.

[21] Feng Gao, Guy Wolf, and Matthew Hirn. Geometric scattering for graph data analysis. In *International Conference on Machine Learning*, pages 2122–2131. PMLR, 2019.

[22] Feng Gao, Guy Wolf, and Matthew Hirn. Geometric scattering for graph data analysis. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2122–2131. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/gao19e.html.

[23] L. Giusti, C. Battiloro, P. Di Lorenzo, S. Sardellitti, and S. Barbarossa. Simplicial attention neural networks, 2022. URL https://arxiv.org/abs/2203.07485.

[24] A. N. Gorban and I. Y. Tyukin. Blessing of dimensionality: mathematical foundations of the statistical physics of data. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 376(2118):20170237, 2018. doi: 10.1098/rsta.2017.0237. URL https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2017.0237.

[25] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1024–1034, 2017. URL https://arxiv.org/abs/1706.02216.

[26] Max Horn, Edward De Brouwer, Michael Moor, Yves Moreau, Bastian Rieck, and Karsten Borgwardt. Topological graph neural networks, 2022. URL https://arxiv.org/abs/2102.07835.

[27] Bobo Hua and Xin Luo. Davies-gaffney-grigor'yan lemma on simplicial complexes, 2017. URL https://arxiv.org/abs/1702.00529.

[28] Guillaume Huguet, Alexander Tong, Edward De Brouwer, Yanlei Zhang, Guy Wolf, Ian Adelstein, and Smita Krishnaswamy. A heat diffusion perspective on geodesic preserving dimensionality reduction. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=HNd4qTJxkW.

[29] Yuanhong Jiang, Dongmian Zou, Xiaoqun Zhang, and Yu Guang Wang. Limiting over-smoothing and over-squashing of graph message passing by deep scattering transforms. *arXiv preprint arXiv:2407.06988*, 2024.

[30] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017. URL https://arxiv.org/abs/1609.02907.

[31] Charlotte Kröger, Sophie Müller, Jacqueline Leidner, Theresa Kröber, Stefanie Warnat-Herresthal, Jannis Bastian Spintge, Timo Zajac, Anna Neubauer, Aleksej Frolov, Caterina Carraro, Silka Dawn Freiesleben, Slawek Altenstein, Boris Rauchmann, Ingo Kilimann, Marie Coenjaerts, Annika Spottke, Oliver Peters, Josef Priller, Robert Perneczky, Stefan Teipel, Emrah Düzel, Frank Jessen, Simone Puccio, Anna C. Aschenbrenner, Joachim L. Schultze, Tal Pecht, Marc D. Beyer, Lorenzo Bonaguro, and DELCODE Study Group. Unveiling the power of high-dimensional cytometry data with cycondor. *Nature Communications*, 15(1):10702, Dec 2024. ISSN 2041-1723. doi: 10.1038/s41467-024-55179-w. URL https://doi.org/10.1038/s41467-024-55179-w.

[32] Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101, 2017. URL http://arxiv.org/abs/1711.05101.

[33] Xu Ma, Can Qin, Haoxuan You, Haoxi Ran, and Yun Fu. Rethinking network design and local geometry in point cloud: A simple residual mlp framework, 2022. URL https://arxiv.org/abs/2202.07123.

[34] Kincaid MacDonald, Dhananjay Bhaskar, Guy Thampakkul, Nhi Nguyen, Joia Zhang, Michael Perlmutter, Ian Adelstein, and Smita Krishnaswamy. A flow artist for high-dimensional cellular data, 2023. URL https://arxiv.org/abs/2308.00176.

[35] Hiren Madhu, Sravanthi Gurugubelli, and Sundeep Prabhakar Chepuri. Unsupervised parameter-free simplicial representation learning with scattering transforms. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 34145–34160. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/madhu24a.html.

[36] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2020. URL https://arxiv.org/abs/1802.03426.

[37] Ion Mihai. Chapter 6 complex differential geometry. In Franki J.E. Dillen and Leopold C.A. Verstraelen, editors, *Handbook of Differential Geometry*, volume 2 of *Handbook of Differential Geometry*, pages 383–435. North-Holland, 2006. doi: https://doi.org/10.1016/S1874-5741(06)80009-9. URL https://www.sciencedirect.com/science/article/pii/S1874574106800099.

[38] Kevin R Moon, David Van Dijk, Zheng Wang, Scott Gigante, Daniel B Burkhardt, William S Chen, Kristina Yim, Antonia van den Elzen, Matthew J Hirn, Ronald R Coifman, et al. Visualizing structure and transitions in high-dimensional biological data. *Nature biotechnology*, 37 (12):1482–1492, 2019.

[39] Michael Moor, Max Horn, Bastian Rieck, and Karsten Borgwardt. Topological autoencoders. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 7045–7054. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/moor20a.html.

[40] Mathilde Papillon, Guillermo Bernárdez, Claudio Battiloro, and Nina Miolane. Topotune : A framework for generalized combinatorial complex neural networks, 2025. URL https://arxiv.org/abs/2410.06530.

[41] Michael Perlmutter, Alexander Tong, Feng Gao, Guy Wolf, and Matthew Hirn. Understanding graph neural networks with generalized geometric scattering transforms. *SIAM Journal on Mathematics of Data Science*, 5(4):873–898, 2023.

[42] Jason Ptacek, Matthew Vesely, David Rimm, Monirath Hav, Murat Aksoy, Ailey Crow, and Jessica Finn. 52 characterization of the tumor microenvironment in melanoma using multiplexed ion beam imaging (mibi). *Journal for ImmunoTherapy of Cancer*, 9(Suppl 2):A59–A59, 2021. doi: 10.1136/jitc-2021-SITC2021.052. URL https://jitc.bmj.com/content/9/Suppl_2/A59.

[43] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation, 2017. URL https://arxiv.org/abs/1612.00593.

[44] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *CoRR*, abs/1706.02413, 2017. URL http://arxiv.org/abs/1706.02413.

[45] María Ramos Zapatero, Alexander Tong, James W. Opzoomer, Rhianna O'Sullivan, Ferran Cardoso Rodriguez, Jahangir Sufi, Petra Vlckova, Callum Nattress, Xiao Qin, Jeroen Claus, Daniel Hochhauser, Smita Krishnaswamy, and Christopher J. Tape. Trellis tree-based analysis reveals stromal regulation of patient-derived organoid drug responses. *Cell*, 186(25):5606–5619.e24, 2023. ISSN 0092-8674. doi: https://doi.org/10.1016/j.cell.2023.11.005. URL https://www.sciencedirect.com/science/article/pii/S0092867423012205.

[46] Naoki Saito, Stefan C Schonsheck, and Eugene Shvarts. Multiscale hodge scattering networks for data analysis. *arXiv preprint arXiv:2311.10270*, 2023.

[47] Naoki Saito, Stefan C Schonsheck, and Eugene Shvarts. Multiscale transforms for signals on simplicial complexes. *Sampling Theory, Signal Processing, and Data Analysis*, 22(1):2, 2024.

[48] Anton Savostianov, Francesco Tudisco, and Nicola Guglielmi. Cholesky-like preconditioner for hodge laplacians via heavy collapsible subcomplex. *SIAM Journal on Matrix Analysis and Applications*, 45(4):1827–1849, 2024.

[49] Michael T. Schaub, Austin R. Benson, Paul Horn, Gabor Lippner, and Ali Jadbabaie. Random walks on simplicial complexes and the normalized hodge laplacian. *CoRR*, abs/1807.05044, 2018. URL http://arxiv.org/abs/1807.05044.

[50] Michael T Schaub, Austin R Benson, Paul Horn, Gabor Lippner, and Ali Jadbabaie. Random walks on simplicial complexes and the normalized hodge 1-laplacian. *SIAM Review*, 62(2): 353–391, 2020.

[51] Alexander Tong, Frederik Wenkel, Dhananjay Bhaskar, Kincaid Macdonald, Jackson Grady, Michael Perlmutter, Smita Krishnaswamy, and Guy Wolf. Learnable filters for geometric scattering modules. *arXiv preprint arXiv:2208.07458*, 2022.

[52] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html.

[53] S. R. S. Varadhan. On the behavior of the fundamental solution of the heat equation with variable coefficients. *Communications on Pure and Applied Mathematics*, 20(2):431–455, 1967. doi: https://doi.org/10.1002/cpa.3160200210. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.3160200210.

[54] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations (ICLR)*, 2018. URL https://arxiv.org/abs/1710.10903.

[55] Aarthi Venkat, Dhananjay Bhaskar, and Smita Krishnaswamy. Multiscale geometric and topological analyses for characterizing and predicting immune responses from single cell data. *Trends in Immunology*, 44(7):551–563, 2023. ISSN 1471-4906. doi: https://doi.org/10.1016/j.it.2023.05.003. URL https://www.sciencedirect.com/science/article/pii/S1471490623000844. Special issue: Systems immunology II.

[56] Matthew Vesely and David Johnson. Processed multiplexed ion beam imaging (mibi) of tumor microenvironments of 54 melanoma samples following immunotherapy, 2025. URL https://data.mendeley.com/datasets/79y7bht7tf/1.

[57] Leopold Vietoris. Über den höheren zusammenhang kompakter räume und eine klasse von zusammenhangstreuen abbildungen. *Mathematische Annalen*, 1927. URL https://doi.org/10.1007/BF01447877.

[58] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds, 2019. URL https://arxiv.org/abs/1801.07829.

[59] Frederik Wenkel, Yimeng Min, Matthew Hirn, Michael Perlmutter, and Guy Wolf. Overcoming oversmoothness in graph convolutional networks via hybrid scattering networks. *arXiv preprint arXiv:2201.08932*, 2022.

[60] Hermann Weyl. Nachrichten von der gesellschaft der wissenschaften zu göttingen. *Mathematisch-Physikalische Klasse*, 1911.

[61] Zhenqin Wu, Alexandro E. Trevino, Eric Wu, Kyle Swanson, Honesty J. Kim, H. Blaize D'Angio, Ryan Preska, Gregory W. Charville, Piero D. Dalerba, Ann Marie Egloff, Ravindra Uppaluri, Umamaheswar Duvvuri, Aaron T. Mayer, and James Zou. Space-gm: geometric deep learning of disease-associated microenvironments from multiplex spatial protein profiles. *bioRxiv*, 2022. doi: 10.1101/2022.05.12.491707. URL https://www.biorxiv.org/content/early/2022/05/13/2022.05.12.491707.

[62] Charles Xu, Laney Goldman, Valentina Guo, Benjamin Hollander-Bodie, Maedee Trank-Greene, Ian Adelstein, Edward De Brouwer, Rex Ying, Smita Krishnaswamy, and Michael Perlmutter. BLIS-Net: Classifying and analyzing signals on graphs. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 4537–4545. PMLR, 02–04 May 2024. URL https://proceedings.mlr.press/v238/xu24c.html.

[63] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations (ICLR)*, 2019. URL https://arxiv.org/abs/1810.00826.

[64] Yao Yu Yeo, Precious Cramer, Addison Deisher, Yunhao Bai, Bokai Zhu, Wan-Jin Yeo, Margaret A Shipp, Scott J Rodig, and Sizun Jiang. A hitchhiker's guide to high-dimensional tissue imaging with multiplexed ion beam imaging. *Methods in cell biology*, 186:213, 2024.

[65] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. Point transformer, 2021. URL https://arxiv.org/abs/2012.09164.

[66] Dongmian Zou and Gilad Lerman. Graph convolutional neural networks via scattering. *Applied and Computational Harmonic Analysis*, 49(3):1046–1074, 2020.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the key contributions—multi-view learning, simplicial modeling, and wavelet-based aggregation—which are directly supported by the theoretical results and experiments.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: A dedicated Limitations section is provided in Appendix Section J. It discusses strong assumptions such as the manifold hypothesis (Appendix Section E), sensitivity to hyperparameters like the kernel bandwidth and VR threshold (Tables A6 and A7), and the computational complexity of the method (Appendix Section I).

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: All theoretical results, including Theorem 4.4 and Theorem 4.3, are accompanied by explicit assumptions and full proofs in the Appendix (Sections E and D). Proof sketches are also included in the main paper to provide intuition.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: All experimental settings, including hyperparameters, model configurations, dataset details, and evaluation metrics, are fully described in Section H and Appendix J. Datasets used are publicly available (e.g., MIBI [42], PDO [45], and Spatial Transcriptomics [61]). Code is provided at `https://github.com/KrishnaswamyLab/HiPoNet` to facilitate full reproducibility.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code is available at `https://github.com/KrishnaswamyLab/HiPoNet`. The repository includes detailed instructions for setting up the environment, running experiments, and reproducing all main results. All datasets used are publicly available and referenced in Section G.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: All training and test details, including data splits, hyperparameters, optimizer settings, and how they were selected, are provided in Appendix H. The optimizer used is AdamW with a constant learning rate of 10^{-4} and weight decay of 10^{-4}. All datasets and experimental configurations are clearly described, and additional reproduction details are provided in the released code repository.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: All experiments report mean values along with standard deviation over 5 independent runs with different random seeds. These results are reported in all relevant tables (e.g., Table 4, Table 1) and clearly indicate the variability due to random initialization and data splits. We specify that the reported error bars are standard deviations, as noted in the experimental setup in Appendix H.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The experimental setup (Appendix H) specifies that all experiments were performed on a single NVIDIA A100 GPU with 40GB memory. Each experiment took approximately 2–3 hours of training time. Total compute usage for all experiments, including ablations and hyperparameter tuning, is estimated at approximately 400 GPU hours. Preliminary experiments that did not make it into the final results required an additional 100 GPU hours.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

Justification: The research adheres to the NeurIPS Code of Ethics. All datasets used are publicly available and anonymized, ensuring no privacy violations. The proposed methods aim to advance scientific understanding without harmful societal impacts. Potential limitations and risks have been transparently discussed in Appendix L.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The paper discusses the potential for positive societal impact in advancing our understanding of tissue microenvironments, which could aid biomedical research and precision medicine. Possible negative impacts, such as risks of overinterpretation of spatial patterns without proper validation and privacy concerns with patient data, are acknowledged in Appendix K, along with suggested mitigation strategies.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper does not introduce models or datasets with high risk for misuse.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets used in this work are properly cited in the main text and Appendix.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: This paper does not introduce any new datasets or other assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve any crowdsourcing experiments or research involving human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This work does not involve any new data collection from human subjects; the datasets used were publicly available, and necessary IRB approvals were obtained by the original data providers.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: LLM is used only for writing, editing, or formatting purposes.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Notations

Table A1: Notations used throughout the paper.

Notation	Description
$\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$	Original point cloud with n data points
d	Input feature dimension (e.g., number of genes or markers)
V	Number of distinct views
$\alpha^{(v)} \in \mathbb{R}^d$	Learnable weight vector for view v
$\tilde{\mathbf{x}}_i^{(v)}$	Reweighted feature vector for point i in view v
$\tilde{\mathcal{X}}^{(v)}$	Reweighted point cloud for view v
$d_{ij}^{(v)}$	Gaussian Kernel distance between points i and j in view v
ϵ	Scale parameter for Vietoris-Rips complex construction
$\mathcal{S}^{(v)}$	Simplicial complex for view v
K	Maximum simplex order in the complex
σ_k	A k-simplex in the complex
N_k	Number of k-simplices
$\mathbf{X}_k^{(v)} \in \mathbb{R}^{N_k \times D_k}$	Feature matrix for k-simplices in view v
$\mathbf{B}_k^{(v)}$	Boundary matrix for order-k simplices in view v
$\Delta_k^{(v)}$	Hodge Laplacian for k-simplices in view v
$\mathbf{P}_k^{(v)}$	Simplicial random walk matrix for k-simplices in view v
J	Number of diffusion scales used in the scattering transform
$\Psi_k^{(v,j)}$	Simplicial wavelet at scale j for order-k simplices in view v
$S_k^v[j]\mathbf{X}_k^{(v)}$	First-order scattering coefficients
$S_k^v[j,j']\mathbf{X}_k^{(v)}$	Second-order scattering coefficients
$S(\mathbf{B}^{(v)}, \mathbf{X}^{(v)})$	Collection of all scattering features for view v
Φ	Aggregated feature representation across all views
$\mathbf{z}$	Multilayer perceptron used for classification
$\hat{\mathbf{y}}$	Final model prediction

B Related Works

The vast majority of work on point cloud-based learning has focused on three dimensional problems, such as mapping and interpreting sensor readings in computer vision or localization tasks [43, 44, 65, 33, 58]. Early methods such as PointNet [43] and its successor PointNet++ [44] introduced permutation-invariant models with the ability to extract local and global features from 3D point clouds. Newer methods such as Dynamic Graph Convolutional Neural Network (DGCNN) [58] leverage dynamic graph representations to better extract rich features from input point clouds, while PointMLP [33] and Point Transformer [65] use standard Multilayer Perceptrons and local self-attention mechanism as the building blocks for better performance in classification and regression tasks. We are primarily interested in biological domains which are much higher dimensional, ranging from dozens in proteomics datasets to thousands in transcriptomic datasets, we require a new approach that is not limited by architectural decisions and is computationally efficient when working with high-dimensional data. Despite the advancements in these existing methods, they are fundamentally designed for 3D point clouds and struggle with performance and scalability in a high-dimensional setting as they rely on spatial heuristics, because local neighborhoods become less meaningful in high-dimension. As opposed to this, HiPoNet is able to scale to high-dimensional data while also preserving the geometry of the dataset.

Single-cell analysis has greatly benefited from graph-based methods that learns global structure from high-dimensional data. These methods typically rely on a *single* graph construct to infer relationships between cellular states. Methods such as UMAP [36] and t-SNE [52] perform non-linear dimensionality reduction by constructing a neighborhood graph and embedding cells into a low-dimensional space. On the other hand, PHATE [38] is a dimensionality reduction method that captures both global and local nonlinear structure but only constructs a *single* graph from the data. While these methods have been useful in understanding various biological processes, they may fail to organize cells based on processes governed by subsets of dimensions with just a single connectivity structure. In contrast to this, HiPoNet has the ability to model distinct cellular processes by leveraging its multi-view framework, preserving important geometric properties.

C Proof of Theorem 4.1

The following Theorem is a more detailed version of Theorem 4.1, which states that the heat equation respects with the 0-homology of the underlying point cloud.

Theorem C.1 (Connected Components on Simplicial Complexes)**.** *Let $\mathcal{S}$ be a simplicial complex of order K with m connected components, and let Σ_k be the set of k-simplices ($k \leq K$). We partition $\Sigma_k = \Sigma_{k,1} \sqcup \Sigma_{k,2} \sqcup \ldots \sqcup \Sigma_{k,m}$, where each $\Sigma_{k,i}$ corresponds to the k-simplices in a single connected component. Let S be a subset of k-simplices, and assume that S is the union of several connected components: $S = \Sigma_{k,i_1} \sqcup \ldots \sqcup \Sigma_{k,i_k}$. Assume the initial condition, $u_k(\cdot, 0)$, of the diffusion equation has support contained in S. Then, for any k-th order simplex $\sigma_k \notin S$ and for all $t > 0$, we have:*

$$u_k(\sigma_k, t) = 0. \tag{5}$$

Proof. The Hodge-Laplacian Δ_k acts locally, meaning its i, j-th entry will be zero unless the i-th k complex is a neighbor of the j-th k-complex (which can happen either via $i = j$ or via upper/lower adjacent neighbors). As a result, if the initial condition $u(\cdot, 0)$ has support contained in S, the solution $u(\sigma_k, t)$ must remain confined to S for all $t > 0$. To formalize this, define a function $\tilde{u}(\sigma_k, t)$ as follows:

$$\tilde{u}_k(\sigma_k, t) = \begin{cases} u_k(\sigma_k, t), & \text{if } \sigma \in S, \\ 0, & \text{if } \sigma \notin S. \end{cases}$$

We will show that $\tilde{u}_k(\sigma_k, t)$ satisfies the same heat equation as $u(\sigma_k, t)$.

First, consider a simplex $\sigma_k \in S$. By definition, $\tilde{u}_k(\sigma_k, t) = u(\sigma_k, t)$ for all $t \geq 0$, and since $u_k(\sigma_k, t)$ satisfies the heat equation, it follows that:

$$\frac{\partial \tilde{u}(\sigma_k, t)}{\partial t} = \frac{\partial u(\sigma_k, t)}{\partial t} = -\Delta_k u_k(\sigma_k, t) = -\Delta_k \tilde{u}_k(\sigma_k, t),$$

where in the final equality we used the fact that Δ_k acts locally.

Next, consider a simplex $\sigma_k \notin S$. By definition, $\tilde{u}_k(\sigma_k, t) = 0$ for all $t \geq 0$, which implies $\frac{\partial \tilde{u}_k(\sigma_k, t)}{\partial t} = 0$. Furthermore, since Δ_k acts locally and σ_k is not in S, we have also: $\Delta_k \tilde{u}_k(\sigma_k, t) = 0$ (since neighbors must be in the same connected component). Thus, $\frac{\partial \tilde{u}_k(\sigma_k, t)}{\partial t} = \Delta_k \tilde{u}_k(\sigma_k, t) = 0$, and so $\tilde{u}_k(\sigma_k, t)$ satisfies the heat equation.

Lastly, since $\tilde{u}_k(\sigma_k, t)$ satisfies the heat equation for all simplices $\sigma_k \in \mathcal{S}$ and coincides with the initial condition $u(\cdot, 0)$ on S, it follows from the uniqueness of solutions to the heat equation that $\tilde{u}_k(\sigma_k, t) = u_k(\sigma_k, t)$ for all $t > 0$. Therefore, for any simplex $\sigma_k \notin S$ and for all $t > 0$, we have: $u_k(\sigma_k, t) = \tilde{u}_k(\sigma_k, t) = 0$. □

D Proof of Theorem 4.3

Below, we will state Theorem D.1, which is a more detailed statement of Theorem 4.3. First, we will introduce some notation and preliminaries.

We will assume that the vertices of $\mathcal{G}(\mathcal{S})$, (i.e., the simplices $\sigma \in \mathcal{S}$) are ordered in a manner consistent with the size of the simplices, that is, all of the k simplices come before all of the $k + 1$ simplices in our ordering. Let N_k denote the number of simplicial complexes of order k and let

$N = \sum_{k=0}^{K} N_k$ denote the cardinality of $\mathcal{S}$, i.e., the total number of simplices. For each $t \geq 0$, let $u_{\mathcal{G}}(\cdot, t) \in \mathbb{R}^N$ denote the solution to the graph heat equation:

$$\frac{\partial u_{\mathcal{G}}(\sigma, t)}{\partial t} = -\Delta_{\mathcal{G}} u_{\mathcal{G}}(\sigma, t), \tag{6}$$

where $\Delta_{\mathcal{G}} = B(\mathcal{G}(\mathcal{S}))B(\mathcal{G}(\mathcal{S}))^T$ is the graph Laplacian of $\mathcal{G}(\mathcal{S})$, and $B(\mathcal{G}(\mathcal{S}))$ is the incidence matrix of $\mathcal{G}(\mathcal{S})$. (If $\mathcal{S}$ is an oriented simplex, then $B(\mathcal{G}(\mathcal{S}))$ is the signed incidence matrix, otherwise it is the unsigned incidence matrix.) Let u_k denote the solution to the heat equation associated to Δ_K in Eqn. 2 and, for $t \geq 0$ define the solution to the simplicial complex heat equation, $u_{\mathcal{S}}(\cdot, t) \in \mathbb{R}^N$ by

$$u_{\mathcal{S}}(\cdot, t) = [u_0(\cdot, t), \ldots, \ldots, u_K((\cdot, t)] \tag{7}$$

(i.e., the denote the concatenation of all of the u_k.)

Theorem D.1 (Agreement of Heat Equation Solutions)**.** *Let $\mathcal{S}$ be a simplicial complex, and let $\mathcal{G}(\mathcal{S})$ be its associated simplicial graph as defined in Definition 4.2. Let $u_{\mathcal{S}}(\sigma, t)$ denote the solution to the heat equation on $\mathcal{S}$. (See Eqn 7 and also Eqn. 2.) Similarly, let $u_{\mathcal{G}}(\sigma, t)$ denote the solution to the heat equation on $\mathcal{G}(\mathcal{S})$.*

Assume that the initial conditions $u_{\mathcal{S}}(\cdot, 0)$ and $u_{\mathcal{G}}(\cdot, 0)$ are consistent, i.e.,

$$u_{\mathcal{S}}(\sigma, 0) = u_{\mathcal{G}}(\sigma, 0) \quad \textit{for all } \sigma \in V(\mathcal{G}).$$

Then, for all $t > 0$ and for all simplices $\sigma \in \mathcal{S}$, the solutions agree, i.e., we have

$$u_{\mathcal{S}}(\sigma, t) = u_{\mathcal{G}}(\sigma, t).$$

Proof. Since the edge set of $\mathcal{G}(\mathcal{S})$ is defined in terms of upper and lower adjacent neighbors, $\Delta_{\mathcal{G}}$ can be written in block diagonal form as

$$\Delta_{\mathcal{G}} = \begin{bmatrix} \Delta_0 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Delta_1 & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \Delta_2 & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \Delta_{K-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \Delta_K \end{bmatrix}. \tag{8}$$

This allows us to see that $u_{\mathcal{S}}$ also satisfies the graph equation equation since

$$\begin{aligned} -\Delta_{\mathcal{G}}\, u_{\mathcal{S}}(\cdot, t) &= \begin{bmatrix} -\Delta_0 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\Delta_1 & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\Delta_2 & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & -\Delta_{K-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & -\Delta_K \end{bmatrix} u_{\mathcal{S}}(\cdot, t) \\ &= \begin{bmatrix} -\Delta_0 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\Delta_1 & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\Delta_2 & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & -\Delta_{K-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & -\Delta_K \end{bmatrix} \begin{bmatrix} u_0(\cdot, t) \\ u_1(\cdot, t) \\ u_2(\cdot, t) \\ \vdots \\ u_K(\cdot, t) \end{bmatrix} \\ &= \begin{bmatrix} -\Delta_0\, u_0(\cdot, t) \\ -\Delta_1\, u_1(\cdot, t) \\ -\Delta_2\, u_2(\cdot, t) \\ \vdots \\ -\Delta_K\, u_K(\cdot, t) \end{bmatrix} = \begin{bmatrix} \frac{\partial u_0(\cdot, t)}{\partial t} \\ \frac{\partial u_1(\cdot, t)}{\partial t} \\ \frac{\partial u_2(\cdot, t)}{\partial t} \\ \vdots \\ \frac{\partial u_K(\cdot, t)}{\partial t} \end{bmatrix} = \frac{\partial}{\partial_t} u_{\mathcal{S}}(\cdot, t). \end{aligned}$$

Therefore, $u_{\mathcal{G}} = u_{\mathcal{S}}$ by uniqueness of solutions.

□

E The proof of Theorem 4.4

The following is a more detailed statement of Theorem 4.4, which states that the diffusion on simplices captures the geodesic distances.

Theorem E.1 (Capturing Geodesic Distance)**.** *Assume that a transformed point cloud $\tilde{\mathcal{X}}^{(v)}$ lies upon a Riemannian manifold $\mathcal{M}$. Let $\mathcal{S}$ be an oriented simplicial complex constructed from $\tilde{\mathcal{X}}^{(v)}$, and let $\mathcal{G}(\mathcal{S})$ be its associated simplicial graph. Denote the 0-th order heat kernel on $\mathcal{S}$ by $H_0^t(\sigma, \tau)$, where σ, τ represent simplices of any order in $\mathcal{S}$. So that $H_0^t u_0(\cdot, 0)$ solves the heat equation with initial condition $u_0(\cdot, 0)$. Then, we may approximate the geodesic distance $d_{\mathcal{M}}(\sigma, \tau)$ on the manifold $\mathcal{M}$ from H_0^t.*

Proof. Let h_t denote the heat kernel on the underlying manifold. Varadhan's formula [53] shows that

$$\lim_{t \to 0} -4t \log h_t(\sigma, \tau) = d(\sigma, \tau)^2$$

Theorem 3 of [15] shows that the graph heat kernel H_0^t uniformly converges to the manifold heat kernel h_t and so the result follows from Theorem D.1 which establishes the equivalence of the graph heat equation and the heat equation on simplicial complexes. □

F Proof of Corollary 4.5

Corollary 5. *The equivalence between the heat kernel on $\mathcal{S}$ and $\mathcal{G}(\mathcal{S})$ implies the equivalence of dimension, volume, and total scalar curvature on the respective underlying data manifolds.*

Proof. Let h_t denote the heat kernel on the manifold $\mathcal{M}$. For any two points $x, y \in \mathcal{M}$. The eigenvalues λ_i and corresponding eigenfunctions ϕ_i of the Laplace-Beltrami operator determine the manifold heat kernel via the spectral expansion:

$$h_t(x, y) = \sum_{i=0}^{\infty} e^{-\lambda_i t} \phi_i(x) \phi_i(y)$$

The eigenvalues are influenced by the volume and curvature of the manifold via Weyl's law [60] and the eigenvalue comparison theorems [11]. Thus, as in the proof of Theorem 4.4, the result follows from Theorem 3 of [15]. □

G Data Cohorts

- **Melanoma**: We use data originally collected by Ptacek et al. [42] and made available in Vesely and Johnson [56], which consists of 54 melanoma patients, who underwent checkpoint blockade immunotherapy. The data contains approximately 489 to 1784 cells from the tumor micro-environments of each patient, resulting in a total of 11,862 cells. Each cell is characterized by 29 protein markers measured by Multiplexed Ion Beam Imaging (MIBI) [64, 3]. This data can be modeled as 54 point clouds in a 29-dimensional space, with sample sizes ranging from 489 to 1784 points per point cloud. The learning task is binary classification of whether the patient experienced recurrence or not (including stable disease as a 'non-recurrence'), predicted from the protein expression levels. We note that MIBI is able to measure the spatial locations of the cells as well as their protein counts. However, in our experiments, we do not use the spatial locations of MIBI. We make this choice so that, with this data set, we can assess the ability of methods to learn purely from the high-dimensional point clouds corresponding to protein counts.
- **Patient-derived Organoids (PDOs)**: We next consider data from Ramos Zapatero et al. [45] featuring patient-derived organoids (PDOs) from 12 different patients with colorectal cancer. Patient-derived organoids (PDOs) are cell cultures grown from patient tumor samples. Each culture, representing a PDO sample, is characterized by 44 proteins and contains approximately 1137 cells. Collectively, we consider 1,678 such cultures, each of which is viewed as a 44-dimensional point cloud, resulting in a total of 2 million cells. The goal is to infer treatment administered to the PDO based on cellular states, serving as a synthetic task for our model.

Data	Task	K
Melanoma	Classification	1
	Persistence prediction	1
PDO	Classification	1
	Persistence prediction	1
DFCI	Outcome	1
Charville	Outcome	2
	Recurrence	1
UPMC	Outcome	2
	Recurrence	1

Table A2: Order of simplicial complex

Model	Accuracy
GCN	41.67 ± 25.00
GIN	38.89 ± 7.86
Graph Transformer	50.00 ± 16.67
HiPoNet	63.60 ± 0.00

Table A3: Comparison of model accuracy on Spatial Data.

- **Spatial Transcriptomics (ST)**: We next consider Spatial Transcriptomics data generated using a 40-plex CODEX (CO-Detection by Indexing) immunofluorescent imaging workflow from Wu et al. [61], capturing 40 protein markers per cell across datasets from UPMC, DFCI, and Charville, comprising 500 patients, each of whom either had head-and-neck cancer or colorectal cancer. Here we construct views in two different manners:
  - *Spatial View:* Cellular coordinates describing how cells are arranged in two-dimensional tissue sections.
  - *Gene View:* Dozens of markers or transcripts measured per cell, detailing complex biological states.

  The overarching goal is to predict outcome of chemotherapy on cancer patients. Another goal is to predict cancer relaspe of recovered patients. This task particularly leverages HiPoNet's ability to fuse spatial context with transcriptional signals for clinically relevant insights in cancer research.

H Experimental Setup

The parameters of HiPoNet are optimized using the AdamW optimizer [32] with a constant learning rate of 10^{-4} and a weight decay of 10^{-4}. We train the model for 100 epochs on every dataset for each model. We record the best metric (accuracy in classification tasks; mean squared error in regression tasks) achieved on the test set in each training run. We compare model performances on each dataset and task by the mean and standard deviation of their resulting five metric scores. For the spatial transcriptomics data, we use the folds from Wu et al. [61]. In Table A2, we describe order of the simplicial complex that we have used for each data set. Typically, our setup necessitates approximately two to three hours of training time for each dataset on a single NVIDIA A100 GPU.

H.1 Baselines

We compare the performance of HiPoNet on the classification tasks outlined in Section 5 to two groups of alternative methods. First, to demonstrate the expressive power of our multiview graph embeddings over traditional graph neural network techniques, we construct K-nearest neighbor graphs of the input point cloud data and attempt to learn a classifier using several popular graph neural network architectures. These include Graph Convolutional Networks (GCN) [30], GraphSAGE [25], Graph Attention (GAT) Networks [54], Graph Isomorphism Networks (GIN) [63], and Graph Transformer Networks (GTNs) [16]. We discuss their performance on our high-dimensional tasks in Section 5.

Num. of Views	Melanoma	PDO
1	27.27 ± 12.85	59.80 ± 1.99
2	$\underline{54.54 \pm 11.13}$	$\underline{61.10 \pm 0.17}$
4	$\mathbf{90.90 \pm 4.92}$	$\mathbf{77.38 \pm 0.94}$

Table A4: Accuracies (mean $\pm$ standard deviation) from ablation on the number of graphs. Best result is bolded and second best is underlined.

Model	Melanoma	PDO
HiPoNet	$\mathbf{90.90 \pm 4.92}$	$\mathbf{77.38 \pm 0.94}$
w/o multi-view	27.27 ± 12.85	59.80 ± 1.99
w/o structure	46.08 ± 8.38	14.09 ± 1.49
w/o reweighing	56.36 ± 7.60	48.30 ± 3.99
w/o wavelets	70.90 ± 14.93	41.53 ± 1.84

Table A5: Ablation over components.

Second, we compare our method directly state-of-the art, bespoke point cloud learning methods. While DGCNN [58], PointNet++ [44], and PointTransformer [65] all perform very well on the three-dimensional problems they were designed to tackle, as shown in Table 4, they struggle to successfully classify high-dimensional point clouds.

I Computational Complexity

The computational complexity of one step of diffusion process, performed via sparse multiplication, has the complexity of $\mathcal{O}(\Sigma_{k=1}^{K} D_k N_k)$, where D_k is a dimension of the features of k-simplices. To construct the wavelets, we need to perform J steps of diffusion for V views. Therefore, the cost of the SWT is $\mathcal{O}(VJ(\Sigma_{k=1}^{K} D_k N_k))$. For the second order scattering coefficients there are $\mathcal{O}(J^2)$ different choices of j and j'. Thus the cost is $\mathcal{O}(VJ^2(\Sigma_{k=1}^{K} D_k N_k))$ The complexity of reweighing through Hadamard product is $\mathcal{O}(D_0 N_0)$. The complexity of VR-filtration is $\mathcal{O}(N_0^2)$. So, the total computational complexity of HiPoNet is $\mathcal{O}(N_0^2 + VJ^2(\Sigma_{k=1}^{K} D_k N_k))$.

J Ablation

In Table A4, we present an ablation study of HiPoNet showing the impact of varying number of views. The results indicate that four views is most effective for capturing the topology of the point clouds.

To assess the contribution of each component—multi-view learning, simplicial complex modeling, and wavelet transform—we conducted an ablation study on the **Melanoma** and **PDO** datasets. As shown in Table A5, removing multi-view learning causes the largest performance drop (63.6% on Melanoma, and 17.6% on PDO), highlighting its central role. Additionally, we see that not using the structure of the data, but instead passing the point cloud features directly into an MLP classifier (without forming a simplicial complex), leads to severe degradation in performance on both tasks, with a 63% drop on PDO and 44.8% drop on Melanoma. Finally, we see that removing the wavelet transform (i.e., using diffusion only) lowers performance across both datasets (with a 20% drop on Melanoma and a 25.85% drop on PDO). These results confirm that each module plays a crucial role, with multi-view learning and structural modeling being especially vital.

To assess the sensitivity to the threshold, we conducted a sensitivity analysis on the Vietoris–Rips threshold, which governs the construction of higher-order simplices, using the single-cell classification datasets. As shown in Table A6, performance varies with the threshold. On **Melanoma**, the best result is achieved at 0.50, while on **PDO**, a lower threshold of 0.15 yields the highest accuracy, indicating that the optimal choice can be dataset-specific. To select the appropriate threshold, we tune it until reaching a critical point—beyond which the number of edges grows rapidly. These critical points are used in our analysis. Notably, we found that PDO has a critical point at 0.15, which leads to improved performance compared to what was reported in the main text. These results suggest that

Threshold for V-Rips ϵ	Melanoma	PDO
0.15	68.18 ± 9.42	$\mathbf{77.38 \pm 0.94}$
0.25	65.45 ± 21.70	60.44 ± 1.61
0.50	$\mathbf{90.90 \pm 4.92}$	63.10 ± 0.00
0.75	77.27 ± 15.21	68.92 ± 0.94
1.00	61.81 ± 13.48	64.96 ± 0.85
1.25	61.81 ± 7.60	62.00 ± 0.51

Table A6: Sensitivity analysis with respect to the Vietoris–Rips threshold.

Kernel Bandwidth σ	Melanoma	PDO
0.25	61.81 ± 7.60	60.44 ± 1.21
0.50	61.81 ± 11.85	63.61 ± 1.21
0.75	65.45 ± 7.60	$\mathbf{77.38} \pm 0.94$
1.00	$\mathbf{90.90} \pm 4.92$	62.44 ± 1.46
1.25	79.09 ± 16.51	62.00 ± 0.51

Table A7: Sensitivity analysis of kernel bandwidth (%).

the model is sensitive to this hyperparameter, and careful selection of the threshold is important for achieving strong performance.

To assess the effect of kernel bandwidth, we performed a sensitivity analysis by varying the bandwidth used during graph construction. As shown in Table A7, performance on the Melanoma dataset initially improves with increasing bandwidth and peaks at 1.00 (90.90 ± 4.92), after which it slightly declines. On the PDO dataset, however, the best performance is observed at 0.75 (77.38 ± 0.94), suggesting that overly large bandwidths may oversmooth the neighborhood structure in some domains. These results indicate that kernel bandwidth is a sensitive hyperparameter and should be tuned based on dataset characteristics. We will incorporate this analysis and its implications into the revised manuscript.

Table A8 presents results across four tasks—PDO, Charville (Outcome and Recurrence tasks), and UPMC (Recurrence task)—with $K = 1, 2, 3$ representing the maximum dimension of simplices used during complex construction. We observe that for *PDO*, performance peaks at $K = 1$, suggesting that lower-order interactions are sufficient. In contrast, for outcome prediction on *Charville* and *UPMC*, higher-order structures ($K = 2$) improve performance, particularly for the UPMC dataset where AUC increases from 0.538 to 0.6044. Notably, performance saturates or slightly drops at $K = 3$, indicating diminishing returns from overly complex topology.

K Broader Impacts

This work introduces HiPoNet, a neural network for high-dimensional point cloud analysis with applications in biological data such as single-cell and spatial transcriptomics. Positive societal impacts include advancing our understanding of complex biological systems, enabling discoveries in disease mechanisms, and contributing to the development of more effective treatments through improved modeling of gene and cell interactions. Potential negative impacts include the risk of over-reliance on model predictions in critical biological or clinical decision-making without sufficient experimental validation. Additionally, applying HiPoNet to sensitive biomedical datasets raises privacy concerns if appropriate safeguards are not maintained. We recommend that HiPoNet be used primarily for hypothesis generation rather than direct clinical application, with findings validated through rigorous experimental studies. Responsible data handling practices and collaboration with domain experts are essential to mitigate these risks.

L Limitations

While HiPoNet advances the modeling of high-dimensional point clouds, it has several limitations. First, although HiPoNet is relatively efficient, training on very large datasets still requires substantial compute resources, especially for larger values of K. Second, the current model has been primarily validated on biological datasets; its generalization to other high-dimensional domains remains to be

Dataset / Task (Metric)	$K = 1$	$K = 2$	$K = 3$
PDO (Accuracy)	$\mathbf{77.38 \pm 0.94}$	72.30 ± 0.18	70.76 ± 0.71
Charville (Outcome AUC-ROC)	0.55 ± 0.001	$\mathbf{0.598 \pm 0.12}$	0.539 ± 0.05
Charville (Recurrence AUC-ROC)	$\mathbf{0.681 \pm 0.01}$	$\mathbf{0.681 \pm 0.01}$	$\mathbf{0.681 \pm 0.01}$
UPMC (Recurrence AUC-ROC)	0.538 ± 0.03	$\mathbf{0.6044 \pm 0.0}$	0.583 ± 0.01

Table A8: Performance across datasets and tasks for varying K.

explored. Finally, HiPoNet focuses on predictions rather than inferring causal relationships, limiting its direct applicability for causal inference tasks.