Topological Data Analysis on Weight Space

Exposition and Interpretation of the Topology of Neural Networks

Haiyu Zhang

March 21, 2024

Data have shape
shape has meaning
meaning brings value.

Motivations

- Filters of CNNs have the same size of local patches. Can we try to put similar analysis on the space of weight vectors?
- We know that neurons in the primary visual cortex and the weight vectors and the filters in CNNs are reflecting responses or functions on the space of patches. There is a conjecture that weight vectors or filters are also distributed in similar structures and can be regarded as a function on patches via inner product constructions to better extract features.

Techniques

- **Density Filtration**

Threshholding $\mathcal{M}$

Define $\mathcal{M}[T] \subseteq \mathcal{M}$ by

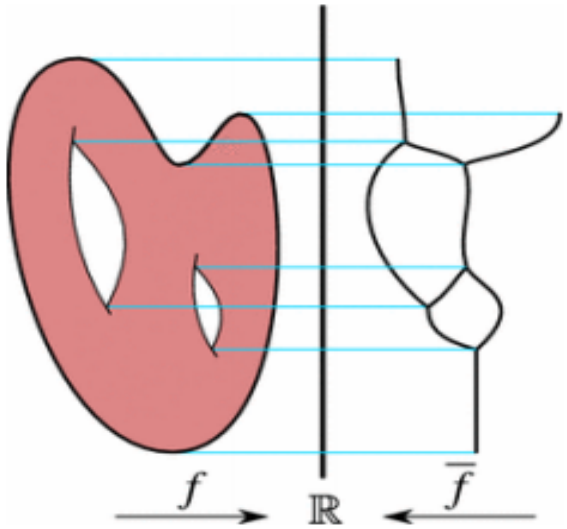
$$\mathcal{M}[T] = \{x | x \text{ is in } T\text{-th percentile of densest points}\}$$

Core Set of M

- **Mapper**

the Reeb Graph

Let $f : X \to \mathbb{R}$ be a continuous map, where X is a topological space. Given $x, x' \in X$, we define an equivalence by $x \simeq x'$ if (a) $f(x) = f(x')$ and (b) x and x' belong to the same connected component of $f^{-1}(f(x))$. The Reeb graph $R(f)$ is defined to be the quotient $X/\simeq$. It is equipped with a map to the real line that sends an equivalence class to the value of f on any of its members. *(Reeb 1946)*

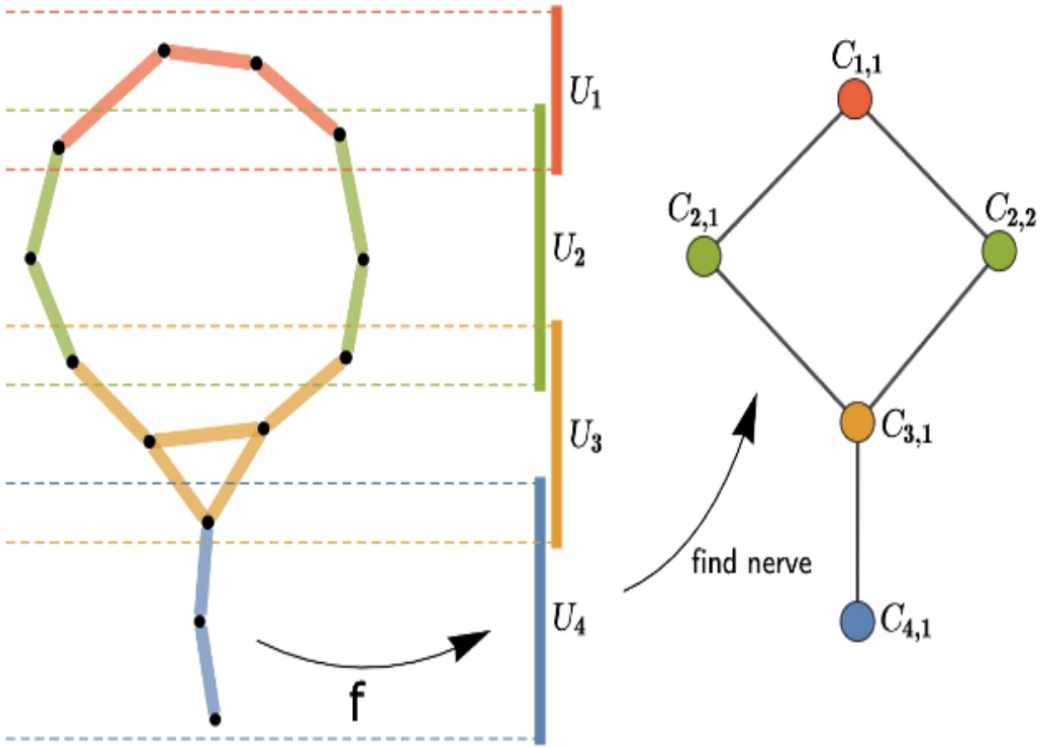


Mapper

Mapper is a combination of dimensionality reduction, clustering and graph networks techniques used to get higher level understanding of the structure of data.
Given a point cloud X, the following are the specific steps:

- **Dimensionality Reduction:** Use a filter function f called lens to map X into a lower-dimensional space.
- **Cover of Projected Space:** Construct a cover $(U_i)_{i\in I}$ of the projected space which satisfies that overlapping intervals have constant length.
- **Clustering:** For each interval U_i, cluster the points in $f^{-1}(U_i)$ into sets $S_{i,1}, \ldots, S_{i,k_i}$. Note: we do not cluster the entire set X but only the subsets $f^{-1}(U_i)$.
- **Graph Construction:** Construct the graph whose vertices are the cluster sets and if two clusters have some points in common, an edge between them exists. (i.e. We construct the nerve of the covering of X by all the clusters.)

Example



Parameters

There are two new parameters in the construction:

- The **resolution** is the number of open sets in the range.
- The **gain** is the amount of overlap of these intervals.

Roughly speaking, the resolution controls the number of nodes in the output and the "size" of feature you can pick out, while the gain controls the number of edges and the "tightness" of the graph.

Lens

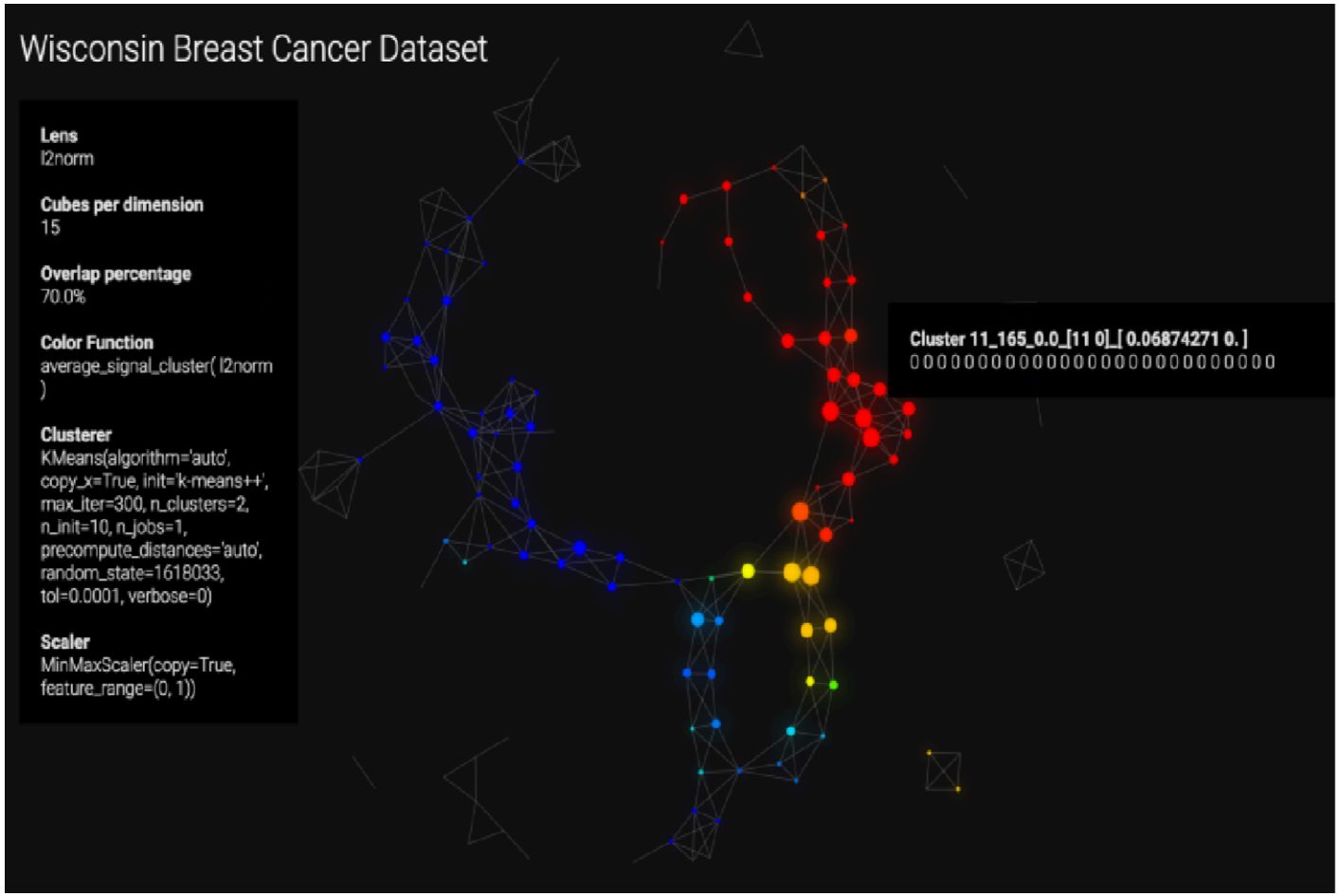
A Non Exhaustive Table of Lenses

Statistics	Geometry	Machine Learning	Data Driven
Mean/Max/Min	Centrality	PCA/SVD	Individual features
Variance	Curvature	Autoencoders	
n-Moment	Harmonic Cycles	Isomap/MDS/TSNE	
Density	...	SVM Distance from Hyperplane	
...		Error/Debugging Info	
		...	

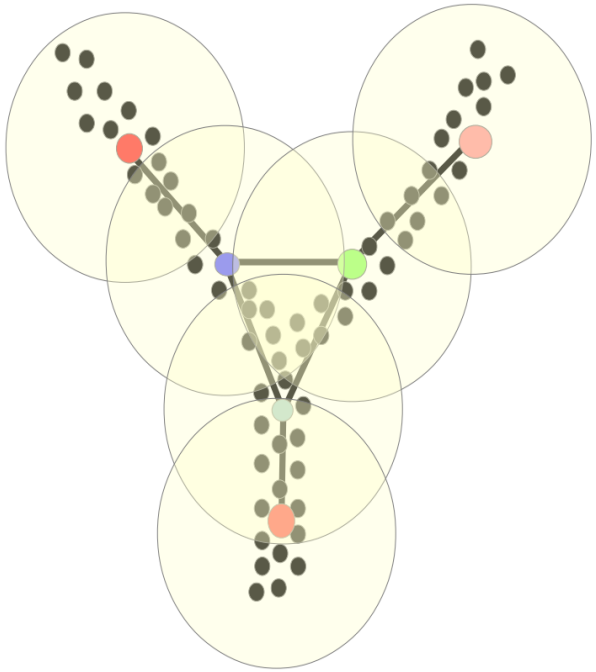
Strengths

• There is less projection loss since the clustering happens in the original space.
• It offers a compressed graph representations.
• It can help select features that best discriminate data. We could find interesting connected components with different lenses.
• The mapper visualization can be used also for model explainability. It induces "geometric query" which push the data from geometrical world into statistical world.

Application



Ball Mapper



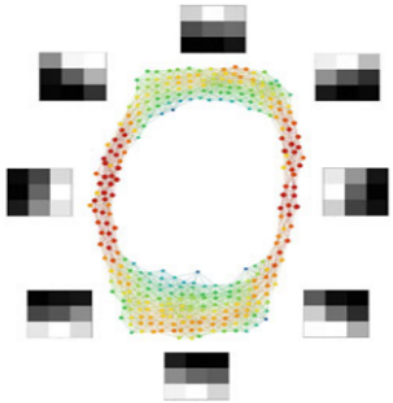
Take one dimensional nerve of that cover (an abstract graph whose vertices correspond to $B(n, \epsilon)$, and edges to nonempty intersections of balls)

Weight Space

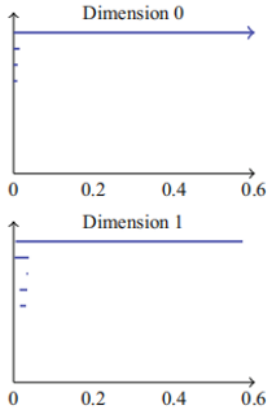
- Fix an architecture and train CNNs for several times.
- Obtain a space of weight vectors (after mean-centering and normalization) and apply TDA on it.

The First Layer of CNN with MNIST

- Train 100 CNNs of type M(64,32,64) for 40,000 batch iterations with a batch size of 128 to a test accuracy of about 99.0%.
- These 100 trained CNNs give us 64 Ã 100 = 6400 9-dimensional points (first layer spatial filters) which we mean-center and normalize.
- Process the data with density filtration and use Mapper algorithm to set up a simplicial complex and analyze it by persistent homology.

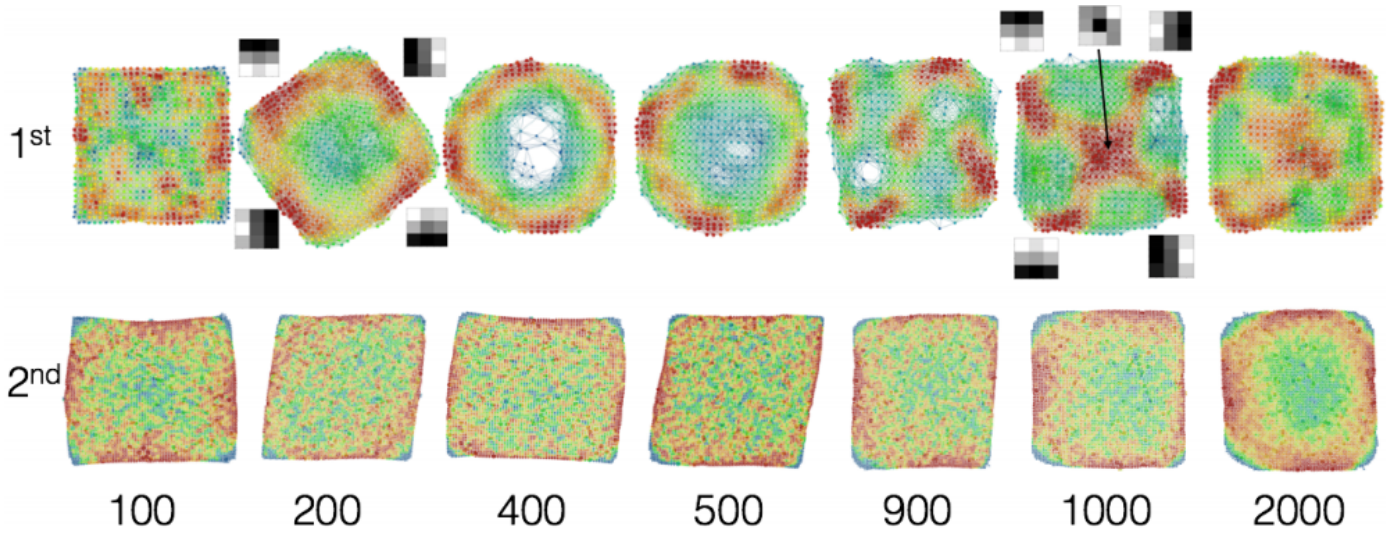


First Layer



Barcode

Learning Process of CNN with CIFAR10



Note: the color represents the density, increasing from blue to red.

- It showed that the spaces of spatial filters learn simple global structures. (Not only for the first layer, but occurs at least up to layers at depth 13.)
- Also demonstrated the change of the simple structures over the course of training.

Pretrain the CNN with Simple Topological Structures

- Faster Speed

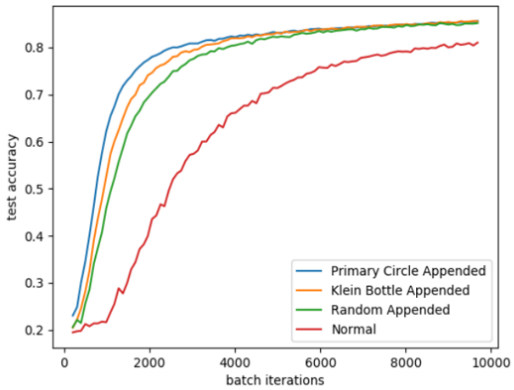
Preprocess each input image with a set of fixed 3 × 3 weights whose inner product with each 3 × 3 patch of the input image was appended to the central pixel value of the patch.

Three different sets of proprocessing weights:

(1) 64 weights from the idealized primary circle

(2) 64 weights from the idealized extension to the three-circle structure, i.e. the Klein bottle

(3) a random gaussian



- Better Generalization

Train a network of type M(64, 32, 64) on MNIST under three different circumstances:

(i) Fix the first convolutional layer to a perfect discretization of the primary circle.

(ii) Fix the first convolutional layer to a random gaussian.

(iii) Train the network as in regular circumstances with nothing fixed.

Then test on 26,032 images of SVHN:

Test accuracies of the three circumstances above:

(i) 28 % (ii) 12 % (iii) 11 %.

Correlation between Topological Structure of Weight Space and the Generality of CNNs

Topological information may serve as a measure of generality.

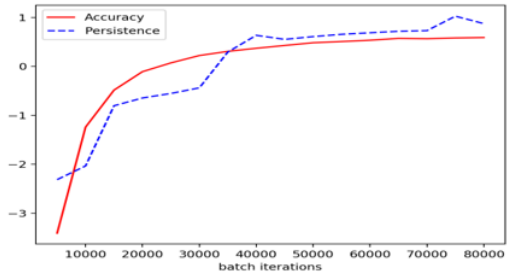


Figure 12: MNIST: Test accuracy and Persistence

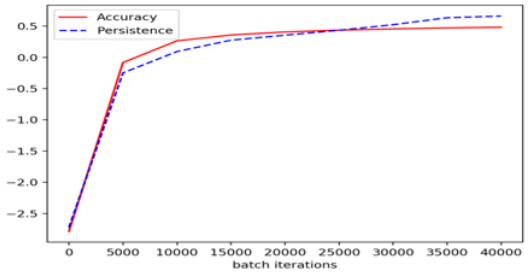
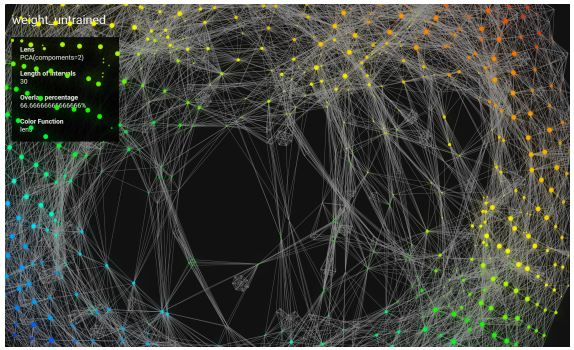
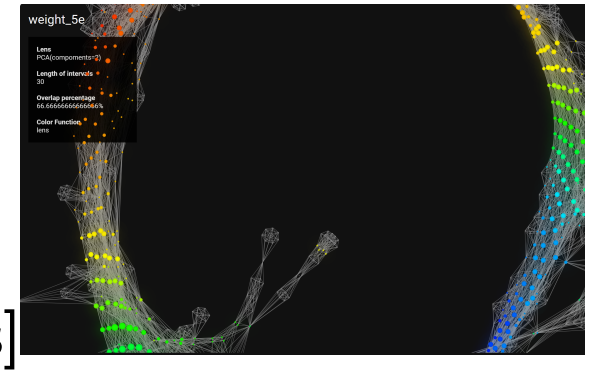


Figure 13: SVHN: Test accuracy and Persistence

- It shows a measure of the strength (or simplicity) of a topological feature and how it correlates with test accuracy on unseen test data.
- It indicates the connection between the existence of simple topological models of the learned weight spaces and the ability to generalize across data sets.

Experiments

[untrained] 

[5 epochs] 

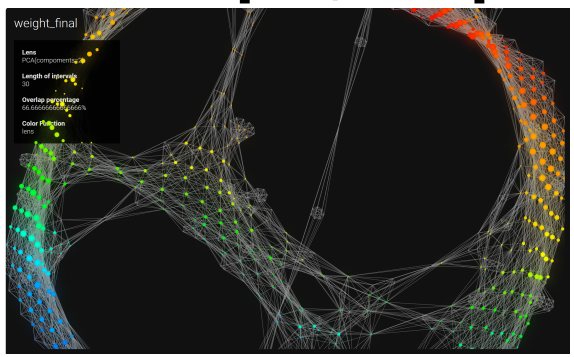
[final] 

Figure: Weight Space of first layer

Persistence Diagram of final weights