Topological Data Analysis and Deep Learning

Anjan Dwaraknath (anjandn@stanford.edu)

Introduction

Topological data analysis is a field which has many promising applications in data science, but in this project, we aim to combine it with deep learning, to demonstrate some of the potential of the combination, using the simple MNIST data set as an example.

Problems with CNN

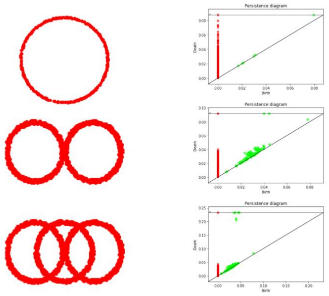
Simple convolutional neural networks have been known to do very well on the MNIST dataset, but this success is plagued with many weaknesses.

1. **Adversarial examples** – It has been shown that it is very easy to come up with adversarial examples that look like one of the classes but is misclassified as another.
2. **Overfitting** – It has been shown that a CNN sometimes uses only a few pixels to determine what the class is.

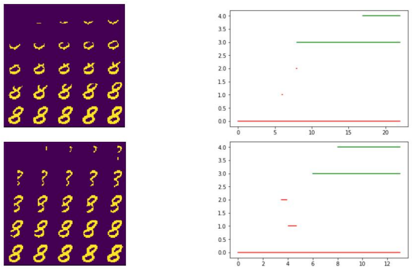
We aim to solve some of the problems above, by using features from persistence homology which try to represent the topological information of the digits.

Persistence Homology

Persistence homology can be used to extract the dominant topological features of a point cloud. The following examples will illustrate the idea.



Topological features for MNIST



We sweep across eight directions in total, to capture the way the topological features are built for each orientation.

Featurization

We featurize by computing a matrix of polynomials for each topological feature and summing across all the topological features available, thus giving us a finite representation.

$$F(m,n) = \sum_{i=1}^{N_f} (b_i + d_i)^m (d_i - b_i)^n$$

Models

We featurize with m and n up to 5. Combining all the eight directions, we get a total of **400 features.**

We use a **9-layer fully connected neural network** as the model architecture.

We first tried with just 2 sweep directions, but the performance was poor. We then switched to 8 directions which gave a significant improvement.

For the best model we changed what the sweeping represented topologically, it now scans slices of the digit.

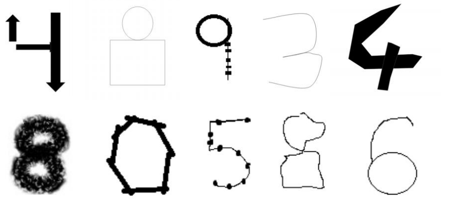
Results

Model	Training accuracy	Test accuracy
2-sweep-9layer	46%	45%
8-sweep-9layer	94%	82%
zigzag-9layer	99%	94%

There were many other models trained with smaller architectures that did not have significant performance.

Qualitative results

We can test how our model generalizes to very different kind of inputs. A small data set was manually created for this purpose.



In this data set with very different line widths and styles, we obtained an accuracy of 70%, which is surprising as the model was only trained on MNIST.

Future work

Following are some of the directions to develop this project further

1. Try different featurizations of topological features.
2. Combine with CNN architecture to get a model that has advantages of both.