The Shape of Digits

A Bayesian Topological Data Analytic Approach to Classification of Handwritten Digits

Thomas Reinke Theophilus A. Bediako Daniel Lim

August 16, 2025

Table of contents

1 Abstract 2

2 Introduction 2

3 Background and Related Work 2
3.1 MNIST . . . 2
3.2 Related Work . . . 3

4 Methodology 3
4.1 Traditional Machine Learning . . . 3
4.2 Our Bayes TDA Methodology . . . 5
4.3 TDA & ML . . . 7

5 Experiments 7
5.1 Data Summary . . . 7
5.2 NN dropout . . . 8
5.3 NN Ridge . . . 8
5.4 NN Lasso . . . 8
5.5 Multinomial Logistic . . . 8
5.6 Bayes TDA . . . 8
5.7 TDA & ML . . . 8

6 Discussion and Analysis 8

7 Conclusion 8

8 References 9

1 Abstract

This paper ...

2 Introduction

The motivation for this work was to compare and examine how dimension reduction via topological data analysis can be used with machine learning and classification models. Algebraic topologist, Gunnar Carlsson, has a quote, "Data has shape, shape has meaning, and meaning brings value." The work in this paper follows this idea, that if there is inherent structure present in data, it can be exploited to aid in modeling.

The Modified National Institute of Standards and Technology database is a set of handwritten digits that is frequently used to train and test image processing models. Our goal is to classify handwritten digits to their correct numeric label. We compared model performance primarily on accuracy, and secondly on the number of features or predictors. First, we want to build an accurate model that can discriminate between digits. Similar to the idea of parsimony in model selection, if two models have comparable accuracy, we will favor the model trained on fewer features. We also consider the ability of TDA-based models to assess the importance of individual predictors, providing insights into which features most influence MNIST classification

3 Background and Related Work

3.1 MNIST

The development of the MNIST dataset stems from early efforts in optical character recognition (OCR), a field of automating the processing of handwritten information like postal codes and census forms. Its lineage can be traced back to the late 1980s with the creation of the USPS database, a collection of 16×16 grayscale images of handwritten zip codes used to train the LeNet neural network. During the same period, the National Institute of Standards and Technology (NIST) was developing its own Special Databases for OCR research, sourcing images from census forms and high-school student samples. One such collection, SD-7, released in 1992, would later form a core part of the MNIST test set.

Challenges in generalizing models across these different datasets, highlighted during a 1992 NIST/Census Bureau competition, revealed biases within the original NIST data. To address these issues, the MNIST database was created in 1994, providing a cleaner, more standardized benchmark for the machine learning community. The dataset led to the development of several modern variations. These include EMNIST (2017), which extended the character set to include letters; QMNIST (2019), which restored the complete original test set; and Fashion MNIST (2017), a drop-in replacement featuring images of clothing items.

3.2 Related Work

Topological Data Analysis is a technique for extracting structural features from datasets, proving effective in classification tasks across various domains. Work in this area includes that of Nicolau et al., who successfully used a topology-based approach to identify a distinct subgroup of breast cancers with excellent survival rates, demonstrating the method's ability to uncover patterns invisible to other methods (Nicolau, Levine, and Carlsson 2011). Researchers have developed more generalized TDA-based classification methods, applying them to problems involving multiple measurements and other complex data structures (Riihimäki et al. 2019; Kindelan et al. 2021).

Many traditional machine learning models have been successfully applied to MNIST (Yeboah 2025), but it has also been a subject of interest for topological methods. Garin and Tauzin provide a "Topological 'Reading' Lesson" by applying TDA specifically to classify MNIST digits, showing that topological features can serve as effective predictors (Garin and Tauzin 2019).

We try an integration of a Bayesian framework to the TDA methodology. This is inspired by the Bayesian framework for persistent homology by Maroulas et al. (Maroulas, Nasrin, and Oballe 2020). Here, we can quantify uncertainty in topological features, in an attempt to increase the predictive power of TDA-based classification models.

4 Methodology

4.1 Traditional Machine Learning

In order to evaluate the value added by TDA-based methods, it is essential to benchmark them against well-established approaches. We consider a range of traditional machine learning (ML) methods, including neural networks with different regularization schemes and classical multinomial logistic regression. Neural networks can overfit, especially on high-dimensional data such as images. Regularization techniques, like dropout, ridge, and lasso help reduce overfitting, and ensure that the network captures meaningful patterns rather than noise.

4.1.1 Neural Networks

The neural network framework considered in this project is the feedforward neural network. It is one of the most widely used architectures for supervised learning. In a feedforward network, information flows in a single direction—from the input layer, through one or more hidden layers, to the output layer—without cycles or feedback connections. The input layer consists of neurons corresponding to the features of the data. One or more hidden layers are placed between the input and output layers, enabling the network to learn complex and non-linear patterns. Finally, the output layer produces the classification results, with the number of neurons equal to the number of digit classes.

Dropout

Dropout is a regularization method that randomly turns off a fraction of neurons during training. This helps the network avoid relying too much on any single neuron and encourages it to learn more general patterns.

Ridge

In NN ridge regularization, a penalty proportional to the square of the weights is added to the loss function. This shrinks large weights while keeping most parameters nonzero.

Lasso

Lasso regularization penalizes the absolute values of weights, encouraging sparsity in the network. The lasso NN forces many weights towards zero, effectively performing feature selection among pixel intensities.

Multinomial Logistic Regression as ML

Multinomial logistic regression, also known as softmax regression, provides a linear baseline for multi-class classification. It is implemented as a single dense layer with softmax activation(James et al. 2013). Table 1 provides a summary of the model architecture.

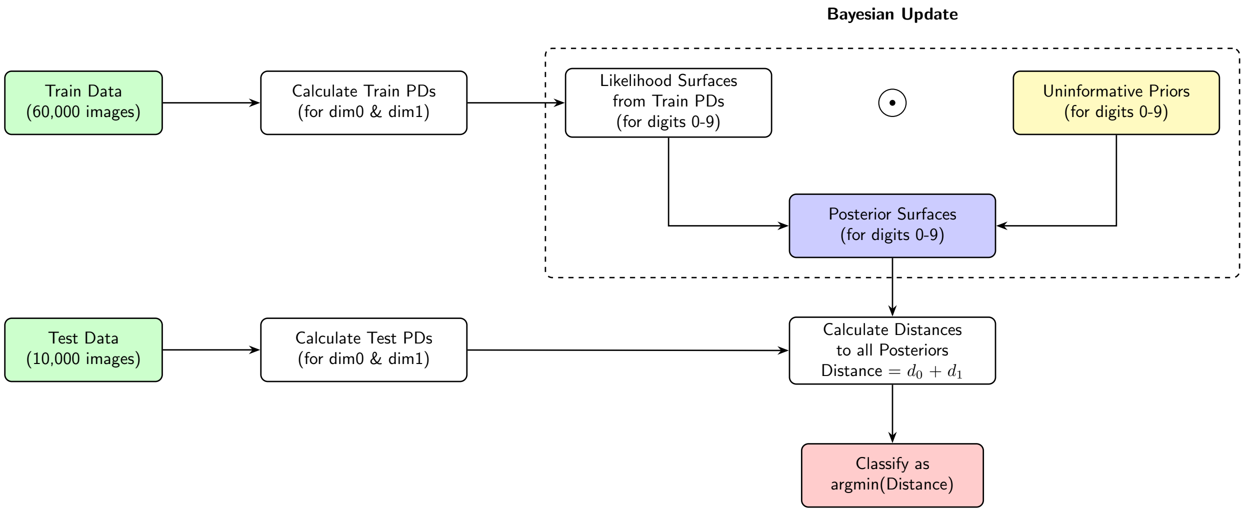
4.1.2 Table 1. Model architectures

Model	Hidden Layers (units)	Activation Functions	Regularization	Output Layer
Multinomial	None	–	None	10
Dropout NN	256, 128	ReLU (hidden), Softmax (out)	Dropout (0.4, 0.3)	10
Ridge NN (L2)	256, 128	ReLU (hidden), Softmax (out)	L2 penalty ($\lambda = 0.01$)	10
Lasso NN (L1)	256, 128	ReLU (hidden), Softmax (out)	L1 penalty ($\lambda = 0.01$)	10

All models were trained under identical conditions—30 epochs, batch size of 128, and a validation split of 0.2. This uniform training setup allows for a fair comparison of traditional machine learning methods.

4.2 Our Bayes TDA Methodology

Our methodology can be summarized in this flowchart, where the 'Bayesian update' comes from A Bayesian framework for persistent homology.

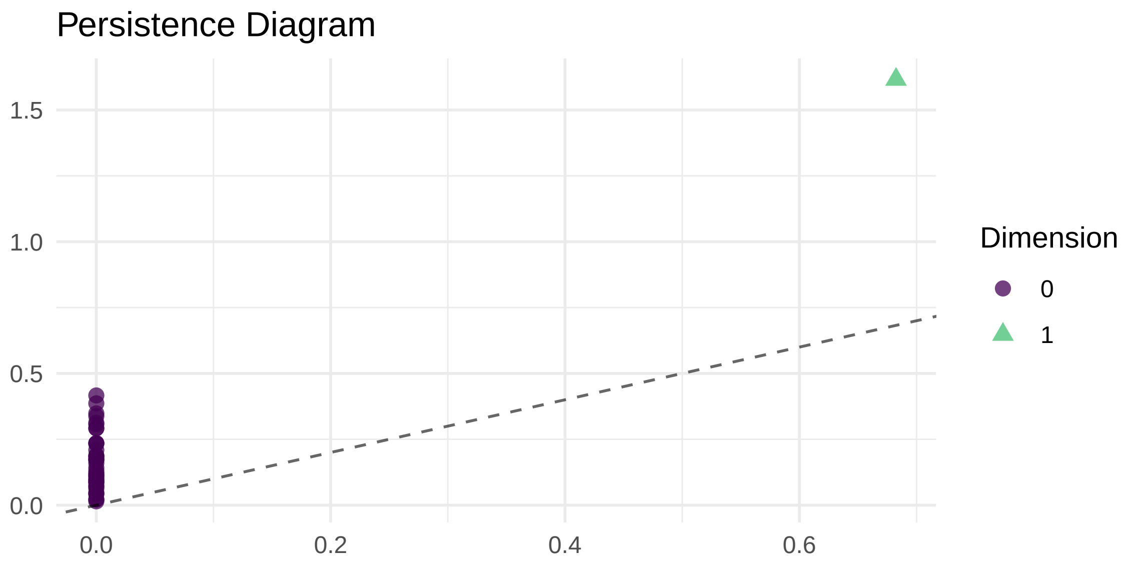

Bayesian Update
Train Data (60,000 images)
Calculate Train PDs (for dim0 & dim1)
Likelihood Surfaces from Train PDs (for digits 0-9)
⊙
Uninformative Priors (for digits 0-9)
Posterior Surfaces (for digits 0-9)
Test Data (10,000 images)
Calculate Test PDs (for dim0 & dim1)
Calculate Distances to all Posteriors Distance = $d_0 + d_1$
Classify as argmin(Distance)

4.2.1 Cubical Complexes & Persistent Homology

For our methodology, we need to get from an image to a persistence diagram. A 2-dimension image is a map $\mathcal{I} : I \subseteq \mathbb{Z}^2 \to \mathbb{R}$. An element $v \in I$ is a pixel, and has value $\mathcal{I}(v)$, which is the intensity. We can binarize an image by the map $\mathcal{B} : I \subseteq \mathbb{Z}^2 \to \{0, 1\}$. With data from a point cloud, we typically build simplicial complexes, but with our data from an image, we will build a cubical complex. Pixels are represented by a $d-cube$, including its faces. With the image represented by a cubical complex K, we build a filtration, which is a sequence of nested subcomplexes, using the image's grayscale values. We do this with a series of sublevel sets:

$$K_i := \{\sigma \in K | I(\sigma) \leq i\}$$

Essentially if a pixel has an intensity less than i, the cube representing it is included in the corresponding complex. After applying persistent homology to this filtration, the birth and death 'times' of topological features are tracked across the intensity levels. The persistence diagram D, is a multiset, (b, d, k), where each point is a homological feature with dimension k, born at intensity b, and dies at intensity d. Persistence is the length of time a feature lasts, $d - b$. In our case, we can only consider $k = 0$, connected components, and $k = 1$, loops/one dimension holes. An example persistence diagram is show below of 40 points sampled from a circle.



From this process we are able to go from an image to a cubical complex to a filtration to a persistence diagram.

4.2.2 Marked Poisson Point Process

A Poisson Point Process Π allows us to model a collection of random points $\{x_1, \ldots, x_n\}$ in a space $\mathbb{X}$, with an intensity measure Λ. The number of points N, is a random variable and follows a Poisson distribution, with mean $\mu = \Lambda(\mathbb{X})$. For a region $A \subseteq X$, $\Lambda(A) = \mathbb{E}[|\Pi \cap A|]$. The mark of each point, m, comes from a space $\mathbb{M}$, in our case, the marks will be the dimension k. We first have our set of locations $\{x_i\}$, then for each x_i, a mark m_i is drawn conditionally & independently from a kernel $\ell(x_i, \cdot)$.

4.2.3 Bayes Update & Guassian representation

Now we wish to incorporate a Bayesian framework to these persistence diagrams represented as point processes. Here we use the tilted representation of the diagram, so instead of (b, d), we use $(b, d - b)$ for a persistence diagram D.

The first part we model is the latent or 'true' underlying persistence diagram D_x; we model it by the intensity $\lambda_{D_x}(x)$. D_x is decomposed into two independent parts. D_{XO} are the points that can be observed with probability $\alpha(x)$: $\alpha(x)\lambda_{\mathcal{D}_X}(x)$. The second part is for points that are missed or not observed: $(1 - \alpha(x))\lambda_{\mathcal{D}_X}(x)$.

The second part we model is the observed persistence diagram D_Y, also two components. D_{YO} are the points generated from D_{XO}, which forms the pair $(\mathcal{D}_{XO}, \mathcal{D}_{YO})$. The connection is via the kernel $\ell(y|x)$. It gives the probability density of observing $y \in \mathcal{D}_{YO}$ given a latent point $x \in \mathcal{D}_{XO}$. The other component is that the points arise from noise: $\lambda_{\mathcal{D}_{YS}}(y)$.

Now we bring the two parts together. The posterior intensity $\lambda_{\mathcal{D}_X|D_{Y^{1:m}}}(x)$ for the latent PD $\mathcal{D}_X$, given m independent observed PDs $D_{Y^1}, \dots, D_{Y^m}$. Let $D_{Y^{1:m}} = \cup_{i=1}^m D_{Y^i}$. The posterior intensity is:

$$\lambda_{\mathcal{D}_X|D_{Y^{1:m}}}(x) = \underbrace{(1-\alpha(x))\lambda_{\mathcal{D}_X}(x)}_{\text{Prior Vanished Part}} + \underbrace{\frac{1}{m}\alpha(x)\sum_{i=1}^{m}\sum_{y\in D_{Y^i}} \frac{\ell(y|x)\lambda_{\mathcal{D}_X}(x)}{\lambda_{\mathcal{D}_{Y_S}}(y) + \int_{\mathbb{W}} \ell(y|u)\alpha(u)\lambda_{\mathcal{D}_X}(u)du}}_{\text{Update from Observed Points } y} \quad \text{a.s.}$$

Without going into details, Maroulas achieves this is computationally using Gaussian mixtures for λ_{D_x}, $\ell(y|x)$, and $\lambda_{\mathcal{D}_{Y_S}}(y)$

4.2.4 Classification

Now that we are able to a posterior persistence diagram, we can do this for each digit 0-9. We can then calculate the Wasserstein distance[1] from the test persistence diagrams to the posterior diagrams for each dimension. Then we sum the distances for each dimension and classify the test image as the class associated with the minimum distance.

4.3 TDA & ML

4.3.1 Filtering

5 Experiments

5.1 Data Summary

- **Data Composition**:
- **Training set**: 60,000 images (50% SD-3, 50% SD-7).
- **Test set**: 10,000 images (originally 60,000, later reduced to 10k).
- **Image Processing**:
- Resized to **28x28 pixels** with anti-aliasing.
- Normalized to center of mass alignment.

[1] Also known as Kantorovich-Rubinstein metric, it is a distance function between probability distributions on a metric space $\mathbb{M}$. $W_q(X,Y) = \left(\inf_{\eta:X\to Y}\sum_{x\in X}\|x-\eta(x)\|_\infty^p\right)^{1/p}$. See more here

5.1.1 EDA

We examined the MNIST training dataset in terms of label distribution, pixel intensity, and overall structure. Label distribution is balanced, with approximately 6,000 examples per digit. We can expect our classifiers not to biased toward any particular class. Pixel intensity analysis reveals that most pixels are zero, reflecting the black background of the images. We also use t-Distributed Stochastic Neighbor Embedding(t-SNE) as developed by (Krijthe 2015) to provide visualization of the 784-dimensional pixel space in two dimensions. The Figure shows well-separated clusters for each digit, suggesting that the classes are distinguishable and that supervised models can capture these patterns effectively.

5.2 NN dropout

5.3 NN Ridge

5.4 NN Lasso

5.5 Multinomial Logistic

5.6 Bayes TDA

5.7 TDA & ML

6 Discussion and Analysis

Comparing the models

7 Conclusion

8 References

Garin, Adélie, and Guillaume Tauzin. 2019. “A Topological ”Reading” Lesson: Classification of MNIST Using TDA.” *CoRR* abs/1910.08345. http://arxiv.org/abs/1910.08345.

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2013. *An Introduction to Statistical Learning: With Applications in r*. Springer. https://faculty.marshall.usc.edu/gareth-james/ISL/.

Kindelan, Rolando, José Frías, Mauricio Cerda, and Nancy Hitschfeld. 2021. “Classification Based on Topological Data Analysis.” *CoRR* abs/2102.03709. https://arxiv.org/abs/2102.03709.

Krijthe, Jesse H. 2015. *Rtsne: T-Distributed Stochastic Neighbor Embedding Using Barnes-Hut Implementation*. https://github.com/jkrijthe/Rtsne.

Maroulas, Vasileios, Farzana Nasrin, and Christopher Oballe. 2020. “A Bayesian Framework for Persistent Homology.” *SIAM J. Math. Data Sci.* 2 (1): 48–74.

Nicolau, Monica, Arnold J. Levine, and Gunnar Carlsson. 2011. “Topology Based Data Analysis Identifies a Subgroup of Breast Cancers with a Unique Mutational Profile and Excellent Survival.” *Proceedings of the National Academy of Sciences* 108 (17): 7265–70. https://doi.org/10.1073/pnas.1102826108.

Riihimäki, Henri, Wojciech Chachólski, Jakob Theorell, Jan Hillert, and Ryan Ramanujam. 2019. “A Topological Data Analysis Based Classification Method for Multiple Measurements.” *CoRR* abs/1904.02971. http://arxiv.org/abs/1904.02971.

Yeboah, Felix. 2025. “Classification and Evaluation of Machine Learning Algorithms on the MNIST Dataset.”