Predicting monthly crime rates with Topological Data Analysis

William Dahl
Computer Science
University at Albany, SUNY
Albany, NY
wdahl@albany.edu

Maxwell McNeil
Computer Science
University at Albany, SUNY
Albany, NY
mmcneil@albany.edu

Nikhil Mathews
Computer Science
University at Albany, SUNY
Albany, NY
nnikhiltittymathews@albany.edu

***Abstract*—Topological Data Analysis (TDA) is the analysis of a data set by employing the techniques used in topology. Persistent Homolgy is a method used in TDA to calculate the topological features of a given data set. This paper presents an analysis on algorithms used to perform Persistent Homology and applies the algorithm to the analysis of crime data from the Boston area. The research project will address the algorithmic complexity of an algorithm used to perform Persistent Homolgy and how it can be used to predict areas of high crime rate by time. Related works include Topological Data Analysis: Giving Data Shape by David Allen [5] where Allen provides a brief overview of an algorithm called MAPPER and also analyzes two data sets, using statistical techniques and TDA; including to analyze crime per capita in the Boston area. TDA has seen an increase in usages over the years as it presents a viable option for the analysis of complex and messy data, a historically difficult topic. In this paper I will discuss the methodology used in implementing the algorithm, detail the experiments being performed, and discuss the results from the experiments.**

I. Introduction

Topological Data Analysis (TDA) is the use of topology in data analysis. It aims to help with the analysis of complex and noisy data, something that has historically been extremely difficult. TDA acts as a general framework to analyze this complex and messy data, independent of any chosen metric through the use of dimensionality reduction. TDA's general aim is to study the shape of the data set given. This is where topology is used by employing algebraic topology to perform an in depth study into the shape of the data. The main method to perform this analysis is persistent homology [2].

Persistent Homolgy is used to compute the topological features of the given data. This method employs the reasoning that the more persistent a feature is in the daat then the more likely it is to be a true feature instead of just the result of noise, the parameters, or specific data sample used. Persistent Homolgy is performed by representing the data as a simplicial complex and then performing a filtration on the data through the use of a distance function [3].

For this specific project we plan to apply the methods used in performing Persistent Homolgy on data consisting of crimes committed in Boston, Massachutess. We Also intend to analyze the runtime and space complexity of the methods used

to calculate the topological features of the data. We then intend to use the calculated features as a feature within a regression model to predict crimes by area and time [4]. We hope that by being able to identify more persistent features within the topology of the data we will be able to more accurately analyze the crime data and provide better predictions than without using the topology of the data.

Topological Data Analysis is a relatively new application for the area of pure math known as topology. By analyzing the topology of data we are able to extract more information from it than what is implicitly provided in the data itself. By being able to extract more information, we can perform an analysis on more areas within the data and get a more in depth understanding of the data that would have been missed otherwise. TDA is an area that could improve analysis methods and further the development of new technologies and improve the performance of existing methods in computing such as machine learning. By deepening our understanding of TDA methods we can further our understanding of the data being used to solve countless problems in society.

II. Related Work

In "Topological Data Analysis: Giving Data Shape" by Dylan Allen [5]; Allen provides a brief overview of an algorithm called MAPPER. He also analyzes two data sets, using statistical techniques and TDA. In the first dataset he uses TDA to provide a summary where areas with high crime rates are noticeably separate from low crime rates. In the second data set he uses TDA to correctly diagnose benign and malignant tumors in subsets of patients with 100% accuracy. He also notes a subset of patients that seem to be related, but differ in a given attribute; which changes the model's diagnosis.

The MAPPER algorithm outputs a graph consisting of nodes and edges that acts as a visual summary of the topology of the data being analyzed. The MAPPER algorithm has 4 parameters. A filtering function that maps the data frame to a set of real numbers. A clustering technique that will be used to perform the grouping of the data. A Number of intervals that define the number of subsets to create. A percent overlap that defines the amount of overlap between each subset.

Allen used the MAPPER algorithm and data about Boston's housing market to determine the per capita crime rates using several variables including per capita crime rate per town, proportion of residential land zoned for lots over 25,000 sq.ft., and proportion of non-retail business acres per town. TDA and the MAPPER algorithm was used to filter by different variables being tested from the housing data and compared to the crime rates per town to find if there was any correlation between any of the variables and crime rates. Allen found there to be a high correlation between crime rate per town and the full-value property-tax rate per $10,000.

In "Analyzing collective motion with machine learning and topology" by Dhananjay Bhaskar et. all; they use topological data analysis and machine learning to study a seminal model of collective motion in biology. Their goal was to compare the performance of using time series order parameters as a feature vector for their machine learning algorithm, which is traditionally used in the analysis of collective motion, to the performance of using a feature vector representing the topology of the data computed through the use of persistent homology. They applied machine learning techniques to the two different types of input. First, the time series of ordered parameters of the simulation data. Second, the measures based on the topology of the simulation data over multiple scales. Bhaskar et. all found that for both the inputs, the topological approach outperformed the one that is based on time series order parameters.

Bhaskar et. all used the measure of the topology of the simulation data obtained from performing persistent homology as a feature to perform both unsupervised learning through the use of k-means clustering as well as supervised learning through the use of an SVM. They calculated the Persistent homology of the simulation data by creating a data cloud of the agents positions within the simulation and creating a simplicial complex by using Vietoris-Rips (VR) complex. After the simplicial complex is built they can then measure the topology of the data by calculating its betti numbers. To perform persistent homology, Bhaskar et. all used

the Risper [1] package to calculate a sequence of birth and death rates for the various features within the data. Next, a codenisty measure is used to help filter out any outliers that may be the result of some noise in the data. The end result is a betti curve at the given time for the data. To calculate the persistent homology of the data as it changes over time, a betti curve is computed for each time interval *t*. Each betti curve is represented by a vector which are concatenated together to form a matrix. This matrix is used as an input feature in the machine learning algorithm.

Our goal is to perform an analysis on crime data in the Boston area using the same technique used by Dylan Allen in his paper "Topological Data Analysis: Giving Data Shape". However, instead of using the MAPPER Algorithm to compute a visualization of the data, we will instead take the methods used by Bhaskar et. all in "Analyzing collective motion with machine learning and topology" to perform persistent homology on boston crime data and use the betti curve as a feature in a machine learning algorithm to predict the project number of crimes that will occur in a specific district of Boston over the span of one month.

III. Methodology

Persistent homology computes a data structure known as the persistence diagram to summarize the space of stable topological features. The most commonly used scheme for generating persistence diagrams is the Vietoris Rips filtration (VR) since it is easily defined for any point cloud. For this project we will be using the Vietoris Rips filtration algorithm to compute our persistence diagram.

The Vietoris Rips Filtration Algorithm results in a vietoris rips complex, a type of simplicial complex. A simplicial complex is a set K of finite sets such that if $\sigma \in K$ and $\tau \subseteq \sigma$, then $\tau \in K$. A simplex is maximal if it has no proper coface in K. The Vietoris-Rips complex (VR complex) $V_\varrho(S)$ of S at scale ϱ is$V_\varrho(S) = \{\sigma \subseteq S | d(u, v) \leq \varrho, \forall u \neq v \in \sigma\}$ [7]. That is the Vietoris-Rips complex we are computing is the sub-complex of S where the euclidean distance between all the vertices in the complex is no more than a specified metric. We first compute the VR complex for some maximal metric. We then iteratively lower the metric to extract the VR complex at the smallest measure. We select the minimum VR complex by specifying a weight function in which the weight is calculated for the VR complex s by recursively selecting a subcomplex t within s and computing its weight. The base case of the recursive algorithm is when t consists of just two vertices, in which the weight is the euclidean distance between the two vertices. Otherwise, the weight of t is 0. We then set the weight of the subcomplex s to be the maximum weight of all subcomplexes t. We now sort the simplices according to their weights, extracting the VR complex for any measure less than the maximal measure first computed as a prefix of this ordering. The resulting complex is the filtration of the VR complex S.

We will be implementing our algorithm in python. We will compute a persistence diagram for a data array X. If X is not a distance matrix, it will be converted to a distance matrix using the chosen metric.

Our parameters include:

1. a numpy array of either data or a distance matrix.
2. The maximum homology dimension to be computed.
3. The maximum distances considered when constructing the filtration.
4. The coefficients to compute the homology
5. An indicator that X is a distance matrix or not.
6. An indicator if cocycles should be computed.
7. The metric to use when calculating distance between instances in a feature array.
8. The number of points to subsample.

A dictionary holding all of the results of the computation is returned.

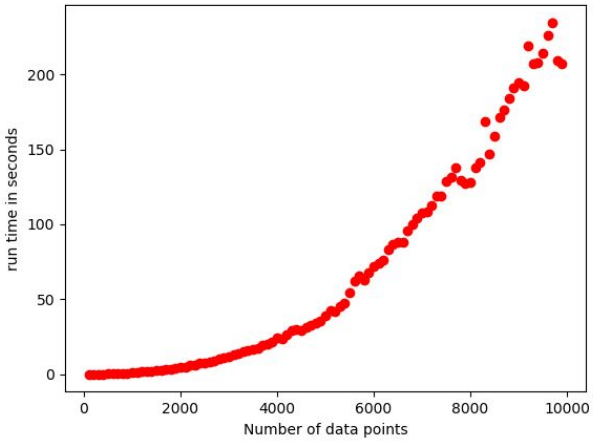
The values within the dictionary include:

1. A list of numpy arrays that represent the persistence diagrams for each dimension computed.
2. A list of numpy arrays that represent the cocycles of each dimension computed.

3. The number of edges added during the computation
4. The distance matrix used in the computation.
5. Index into the original point cloud of the points used as a subsample
6. Covering radius of the subsampled points.

For each point p we will need to check the distance to every other point p. This is p^2 calculations. We need to repeat this step for each discrete-distance that we wish to examine, we will call this ds. Thus we can say that the run time will be $O(dsp^2)$.

For synth experiments we gen-erate varying number of random data points and then run Victoria-Rips on them.



From this we can see the exponential increase as we increase the number of points p. This follows our theoretical expectation of the algorithm.

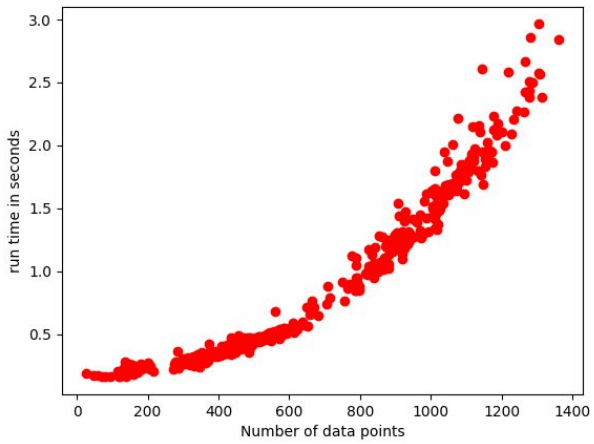
IV. Experiment

Our goal is to construct a regression model using the precedence graph of the number of crimes that occured in each district in Boston over the span of one month as an additional feature in the regression. Using This regression we aim to predict the projected numbers of crimes that will happen in a given month within a given district. We will compare the accuracy of our regression model using the computed precedence graph as the input vs a regression model using just the district and a regression model using both.

To begin our Experiment we start by calculating the vietoris rips vector and the corresponding precedence diagrams over each district in the Boston area. We create a table to hold the top 2 dimensional scores computed by the vietoris rips for the crime data. The crime data is read into a dataframe using pandas. Next, we sort the crimes in the order of their occurence. and write the districts out to a csv to be used in our regression model. We then split the data into a training set and a testing set.

We use the latitude, longitude, district, and the date of occurrence for each crime in our training data. We put the training data into a pandas data from and index the data frame on the occurrence date. We then loop through the data frame of the training data. We grouped the training data by the months that each crime occurred into another data frame containing the number of crimes that occured in each district in that month. We perform the vietoris rips algorithm on the latitude and longitude of each district and retrieve the persistence diagrams for the number of crimes in that district for the current month. Next, we extract the first two dimensional values from each of the persistence diagrams and take the highest value and store them into a table. This table contains the highest two dimensions from each prestance graph for each district for each month within the training data. The number of the crimes that had happened in the district for the current month is equal to the number of entries within the district data frame for the current month. The final resulting data frame contains the districts, number of crimes, and the 1st and 2nd dimensional values for that district in that month. The training data is then written out to a csv file to be used in our regression model.

The same process used from preparing the training data to be used in our regression model is also applied to the testing data so that it can be used to test our regression model. The testing data frame is also saved to a csv.



The above figure is the running time complexity for our vietoris rips as applied to the Boston crime data. The Time complexity on the crime data is similar to that of our base case measurement for the algorithm time complexity. Both the base measurement shown before and the time complexity measurement on the actual crime data is $O(dsp^2)$.

To begin constructing our regression model we start by defining the districts we would like to perform our predictions on. We treat the districts as labels and thus encode them into integers to be used in our regression model. The encoded labels are then turned into a sparse matrix to serve as a kernel for the regression model and stored as a data frame.

The training data and testing data are both read from the respective csv file and loaded into a data frame. The data frames are then indexed in the districts and join the kernel matrix to each data frame.

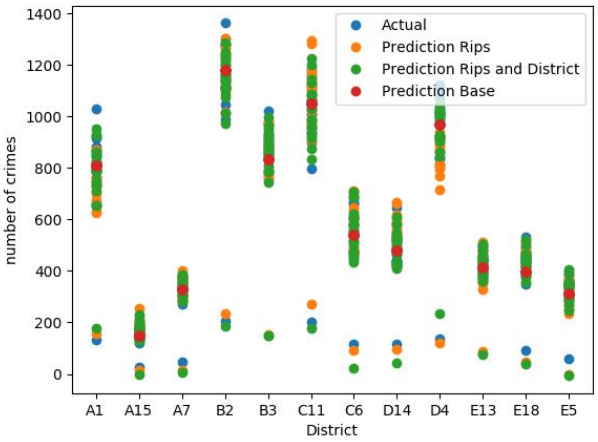
The regression is modeled as a function *f(x, y)* where x is the district and y in the number of crimes committed in the district over the span of one month. The *(x, y)* training and testing sets are extracted from the respective data frame and saved as vectors of values.

We create three separate *(x, y)* training and testing sets each, (x_{dr}, y), (x_d, y) *and* (x_r, y). x_{dr} uses the top 2 dimensional values computed from the precedence graph and the district, x_d is only the districts, and x_r uses only the topology of the data.

We then fit three separate regression lines $f(x_{dr}, y)$, $f(x_d, y)$ and $f(x_r, y)$. $f(x_{dr}, y)$ is fit using the training data with the dimensional values computed from the precedence graph and the district. $f(x_d, y)$ is fitted with the training data of just the districts. $f(x_r, y)$ is fitted using just the diminsal values computed from the precedence graph. Using these regression lines a number of crimes for each district from the testing data is predicted. $f(x_{dr}, y)$ is used to predict the number of crimes given a district and its respective dimensional values, $f(x_d, y)$ is used to predict the number of crimes from just the district, and $f(x_r, y)$ is used to predict the number of crimes using only the topology from the data.

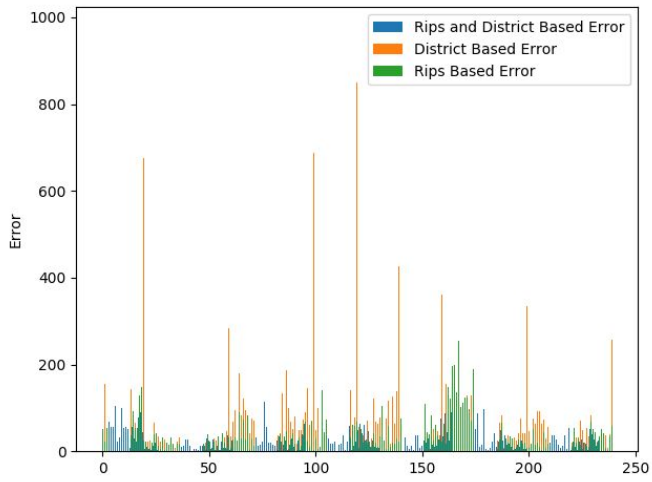
V. Results

To analyze the performance of the three regression models we analyze the mean square error between the three models.



The above figure displays the predicted number of crimes for each district over the span of one month. From the graph we can see that each district displays a relatively tight grouping for the number of crimes each month. We can see that there is a grouping around the number of crimes for each district. Both the predictions using the regression with the dimensional values and the actual number of crimes for the districts are more spread out than the predicted value from the district only regression. This is because performing a regression using only the districts results in the prediction being the mean value of the number of crimes in each district observed in the training data. This means that no matter the month, the predicted number of crimes will be the same for each district. However, thanks to the dimensional values from the precedence

graph, the regression using the dimensional values is able to make a more accurate prediction that is not simply just the mean number of crimes for the district.



The figure above shows the error for each prediction made. As we can see, the error for prediction made using the regressions with the dimensional values is much lower than the regression using just the districts. The error for each prediction is calculated by taking the absolute value of the difference between the predicted number of crimes and the actual number of crimes. The average error for the prediction using just the dimensional values is 43.409, the error for the district only regression is 79.092, and the error for the prediction using both the topology of the data and the district is 26.335. This shows that the prediction using the regression model with the dimensional values and the district is about 4 times more accurate than the regression model using just the districts and about 2 times more accurate than the regression model using only the topology of the data. Even when only using the topology of the data and giving the model no information on the district we can see nearly 2 times more accurate prediction when only giving the model a district. Clearly the information gleaned from the use of TDA is even more useful in constructing the model than using the info from the data directly. Combining the two serves to only strengthen the model even further.

VI. Conclusion

Topological Data Analysis (TDA) is the use of topological principles to analyze data. TDA can be used in conjunction with machine learning to try and make machine learning models more accurate by using the topological features of the data within the models. One of the TDA methods employed with machine learning is persistent homology. Persentance Homology is the measure of the life and death rates for topological features in the data over time. This method makes for a great way to analyze the change of data over time and use it to predict future events.

TDA has been used to provide a summary where areas with high crime rates are noticeably separate from low crime rates and to correctly diagnose benign and malignant tumors in subsets of patients with 100% accuracy. Topological Data Analysis and machine learning have been used to study a seminal model of collective motion in biology with the goal to compare the performance of using time series order parameters as a feature vector for their machine learning algorithm, which is traditionally used in the analysis of collective motion, to the performance of using a feature vector representing the topology of the data computed through the use of persistent homology in which TDA performed the best.

To compute the percentage homology of the data we used the viteros rips algorithm. This algorithm takes in our data and returns a sequence of persenace homology matrics up to the desired dimensions. Using the persistent homology computed by the vietoris rips algorithm we were able to create a regression that was 4 times more accurate than a regression not using the topology of the data. A regression model without the topology of the data resulted in just predicted the average number of crimes for the district each month. However, we know from the data that the number of crimes changes from month to month for each district. By using the topology of the data we were able to create a regression model that was able to distinguish between the different months for each district and provide more accurate estimates for the number of crimes in the given month for the district.

By analyzing the topology of the crime data in Boston, we were able to achieve a regression model for predicting the projected number of crimes in a coming month for a district with an average margin of error of about 26 crimes. Without the topological

data, the predicted number of crimes is just an average and serves no benefit for informing police forces on when and where to focus more manpower and resources. Using the topology of the crime data we can better inform police departments of one period of high crimes and low crime rates at a district level.

VII. References

[1] Tralie, Christopher, et al. "Ripser.py: A Lean Persistent Homology Library for Python." Journal of Open Source Software, 13 Sept. 2018, joss.theoj.org/papers/10.21105/joss.00925.

[2] En.wikipedia.org. (2020). Topological data analysis.

[3] En.wikipedia.org. (2020). Persistent Homology.

[4] Data.boston.gov. (2020). Crime Incident Reports (August 2015 – to Date) (Source: New System) – Analyze Boston

[5] Allen, Dylan, "Topological Data Analysis: Giving Data Shape" (2017).Mathematics, Engineering and Computer Science Undergraduate Theses 1. https://scholars.carroll.edu/mathengcompsci_theses/1

[6] Bhaskar, Dhananjay, et al. "Analyzing Collective Motion with Machine Learning and Topology." *AIP Publishing*, AIP Publishing LLC, 1 Jan. 1970, aip.scitation.org/doi/10.1063/1.5125493#fragmentNav_0.

[7] Afra Zomorodian, "Fast Construction of the Vietoris-Rips Complex" http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.210.426&rep=rep1&type=pdf