Topology Final Project

Topological Subtitle

Josh Meadows, Alex Richards, Thomas Smale, David Coles

Mathematics
California State University, Chico
United States

28 April 2021

Table of Contents

Introduction

This paragraph begins the introduction to our final project for MATH 399 Topological Data Analysis.

One of the most pressing matters on any college campus is climate change. Climate change is an issue that affects everyone in this world. As humans have evolved scientists believe that human actions are harming the planet. This is causing more extreme weather that is disrupting many natural habitats. As a result there may be enormous loss of both animal and human life. As students of a California State University, we have grown up in this beautiful state and come to appreciate the ground underneath us. We are concerned that human induced changes to the climate may result in our beautiful state turning to a dry desert. Working together and combining our backgrounds in math and computer science that we have learned at Chico State, we are going to apply groundbreaking methods to study our climate. One of the most beautiful regions in California is Lake Tahoe. It offers year round outdoor fun and our state depends on it for many resources. Since it is such a special place, people care about it and would hate to see it be destroyed. At 6,000 feet of elevation in the winter time it is cold and snowy, but warms up in the summertime with warm California blue skies. This diverse climate gives us many different aspects to analyze. The data comes from the National Oceanic and Atmospheric Administration, a federal government organization that documents climate all over the country. Government workers have been collecting daily data about the weather at its station in Tahoe City since 1903. This gives us over a 100 years of reliable data to analyze. These data points are available in a csv file that is about 43000 lines long. The data is multi dimensional as it has fields such as temperature, amount of precipitation, and 15 different weather types. We will combine all of these dimensions into a point cloud to study its shape. We will then analyze the data using topological data analysis to study its qualitative features. Topological data analysis is a great way to deal with multi dimensional data that traditional methods struggle to extract value from. We will also use traditional methods such as scatter plots or line plots to visualize the data. This is to help with our understanding of the data set and confirm our conclusions. However, we expect that using topological data analysis will provide us with information about the data not seen in the traditional graphs. Using data analysis we will be able to withdraw trends about the data like if the temperatures are increasing or the amount of yearly snow is decreasing. From this we will learn more about how the climate has been changing in Tahoe, which may be a reflection of a lot of California.

—DAVID— What are the problems to be solved? What are the mechanisms developed to solve the problems

The amount of data being collected in the world is a massive value that is continuously increasing. Data can consist of our network traffic, social media accounts, and grocery receipts. Processing all of this data is an extreme challenge and those who can make sense of it are rewarded. The market size for data science is increasing as companies look to gain a competitive edge by utilizing data to make better decisions. Data analysis is no easy feat as the size of the data, noisiness, dimensions, and incompleteness cause challenges. There are many different ways to analyze data but in the last 15 years topological data analysis has been recognized as useful in dealing with high dimensional complex data. Topological data analysis is a combination of algebraic topology, computational geometry, computer science, and more. It measures the qualitative features of data by computing the persistent homology which utilizes algebraic topology.

We are hoping to leverage a software package called dionysus. Dionysus is written by a student who studied under Gunnar Carlson, Edelsburg at Duke, and is currently at Lawrence Berkeley Laboratory. It is written in c++ with a python interface. C++ will give the library its speed, while python will provide a interface that is friendly to work with.

Project

Vietoris-Rips Complexes

We will be using Vietoris-Rips Complexes to topologically analyze our data.

Algorithm

To start, we decided to calculate the yearly average of the maximum and minimum temperatures. In short, this was because computation time for all points was overwhelming, but this is discussed more in the Evaluation section. When calculating the yearly average temperatures, we excluded any dates that had minimum or maximum temperatures missing for simplicity. Here is our final implementation to create our yearly averages:

```
with open('tahoe_city.csv', newline='') as csvfile:
    #Use DictReader so list doesn't contain header row
    climatereader = csv.DictReader(csvfile)
    average_max = 0
    average_min = 0
    # keep track of number of dates in year to take average
    num_dates_in_year = 0

    for i,row in enumerate(climatereader):
        if row['TMIN'] != "" and row['TMAX'] != "":
            # Another date counted for the current year
            num_dates_in_year += 1
            average_min += string_to_float(row['TMIN'])
            average_max += string_to_float(row['TMAX'])

            # Get the year of the current row we are on. Date format is
            # YYYY-MM-DD, so we split by "-" and get the 0th element(the year)
            row_year = row['DATE'].split("-")[0]
            # If we are starting a new year
            if row_year != cur_year :
                cur_year = row_year
                yearly_maxtemps.append(average_max / num_dates_in_year)
                yearly_mintemps.append(average_min / num_dates_in_year)
                num_dates_in_year = 0
                average_min = 0
                average_max = 0
```

The next task at hand was to create a point cloud from the data we had. It is important that our cloud had the correct dimensions that Dionysus expected. Because we had multiple one dimensional arrays, we had to combine them to make one large multidimensional array.

There are many ways that one might do this, but we chose to use a convenient numpy function:

```
point_cloud = np.vstack((yearly_mintemps,yearly_maxtemps)).T
```

At this point, we have set up the point cloud, and are able to use Dionysus in order to calculate the Vietoris-Rips Complex from the data:

```
f = d.fill_rips(point_cloud, 3, 2)
```

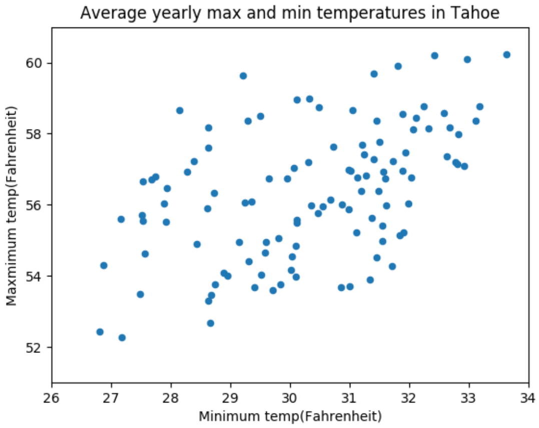
where 3 is the maximum number of dimensions, and 2 is the maximum radius of the balls. This function returns a filtration, which we then can use to get our persistence diagrams and barcodes:

```
p = d.homology_persistence(f)
dgms = d.init_diagrams(p,f)
d.plot.plot_diagram(dgms[0], show=True)
d.plot.plot_bars(dgms[0],show=True)
```

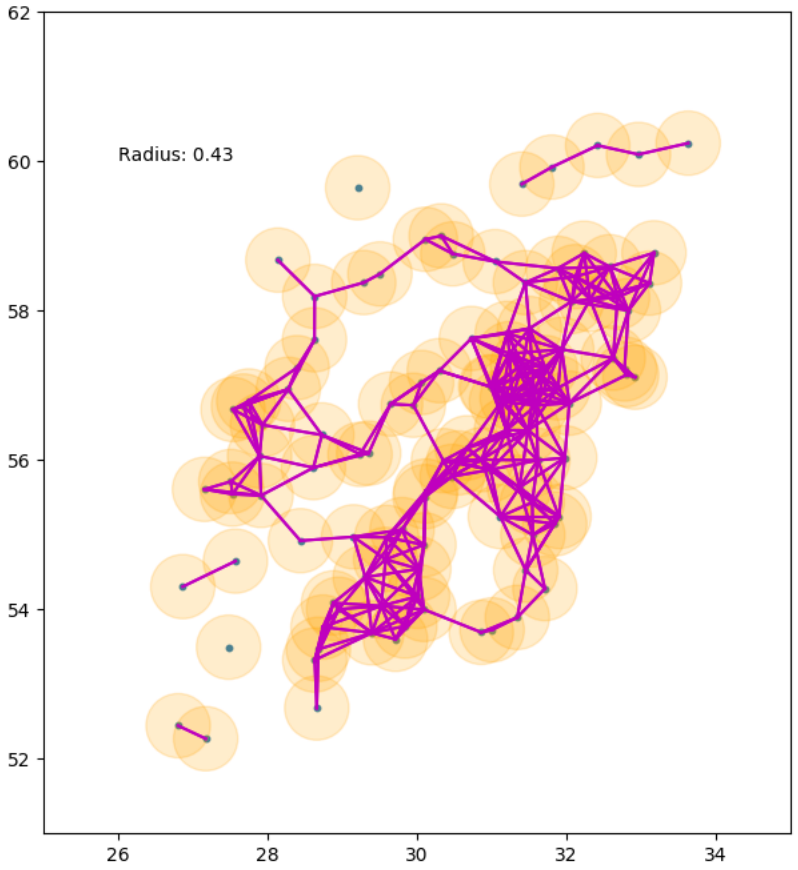
where the index of dgms is the Betti number.

There were many different directions we could have went with this data set. We settled on having the minimum yearly average temperatures on the x-axis, and maximum yearly average temperatures on the y-axis. Sticking to two dimensions allows us to better visualize the data and verify that Dionysus is creating barcodes correctly.

Here is a graph of the data:



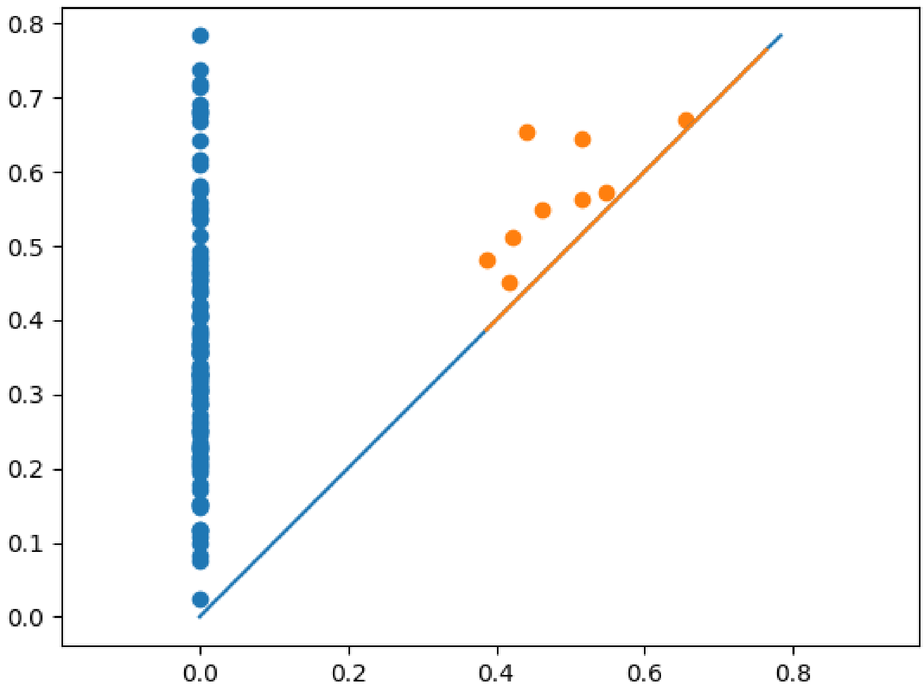
And here is an example Vietoris Rips Complex with a radius of 0.43:



The image was created using matplotlib by scattering the points, plotting circles with centers at each point, and then plotting lines if any two circles intersect. Two circles intersect if the distance between the two points is less than two times the radius. In the image, we can see that there are two prominent holes, which we expect to see in the barcodes and persistence diagrams for betti 1.

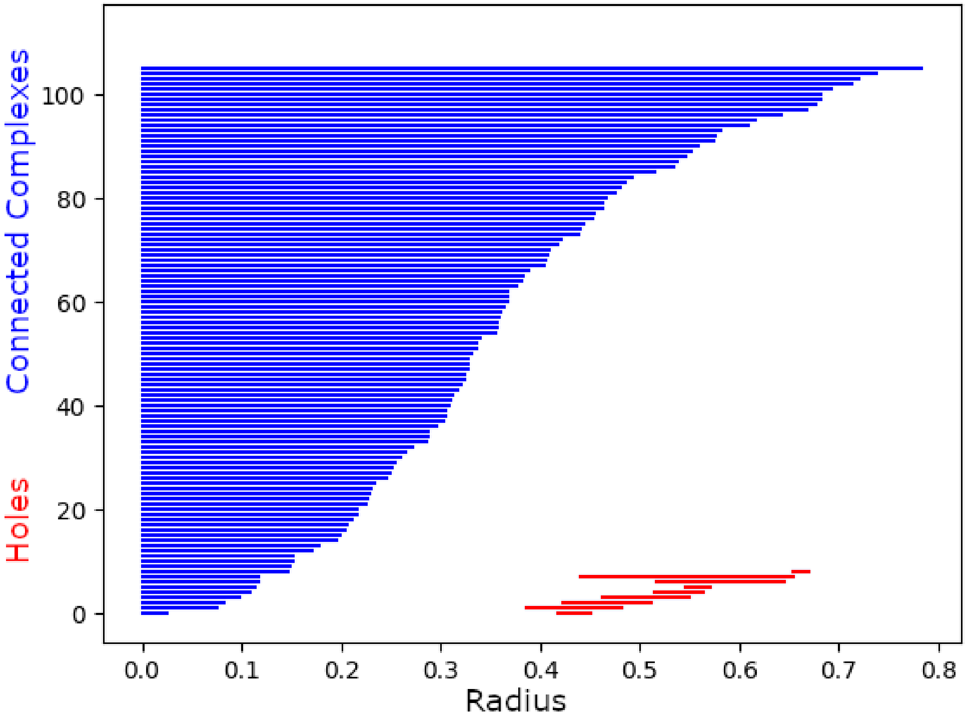
Now that we have a visualization of the Viertoris-Rips Complex, we can analyze the persistence diagrams and barcodes to see if the holes we observe above are reflected in the diagrams.

Here is the persistence diagram obtained from Dionysus:



Alex, do you want to analyze this part?

Here is the barcode obtained from Dionysus:



Alex, do you want to analyze this part?

Conclusion

References

Evaluation

We encountered several challenges during this project. To start, documentation for Dionysus is fairly sparse, and unfinished. It was unclear what format Dionysus expected the point cloud to be in, the syntax of the filtration returned by the Vietoris-Rips function, and what to do with the filtration once we had it. Another struggle we encountered was the amount of computation time that calculating the Vietoris-Rips Complex took. At first, it was very hard to pinpoint the issue, because the program would never come to completion. Even after cutting our data size in half, it would still run for an indefinite time, with one notable run taking over 2 hours before it was force-quit. In an effort to see if the program would ever complete, we used around 100 points, which would take at most a minute to two minutes. We decided this was a range that we felt comfortable with, and decided to instead take the yearly averages of the data, in order to decrease the size of the point cloud.

There were also several elements of the project that went particularly well. We were able to plot the data with ease, and even made an animation of the Vietoris-Rips complex with an increasing radius. This helped us visualize the data and try and make sense of our diagrams and barcodes.

The four of us spent at least 20-30 hours on this project. A good amount of time was spent trying to work with our csv file and Dionysus, but the main portion of our time went to examining our results and trying to find meaning from them.