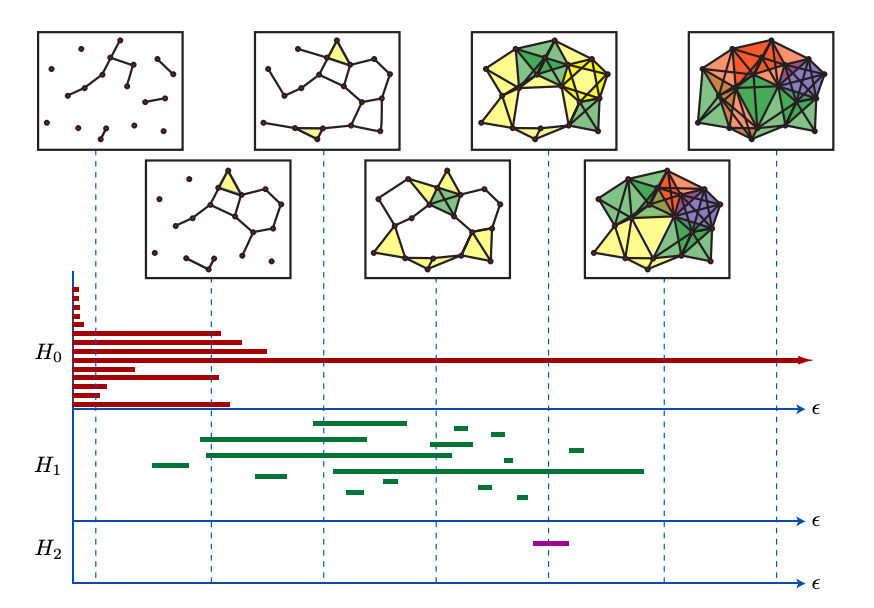
USING PERSISTENT HOMOLOGY AND BETTI NUMBERS TO CLASSFIY HANDWRITTEN APLHABETS

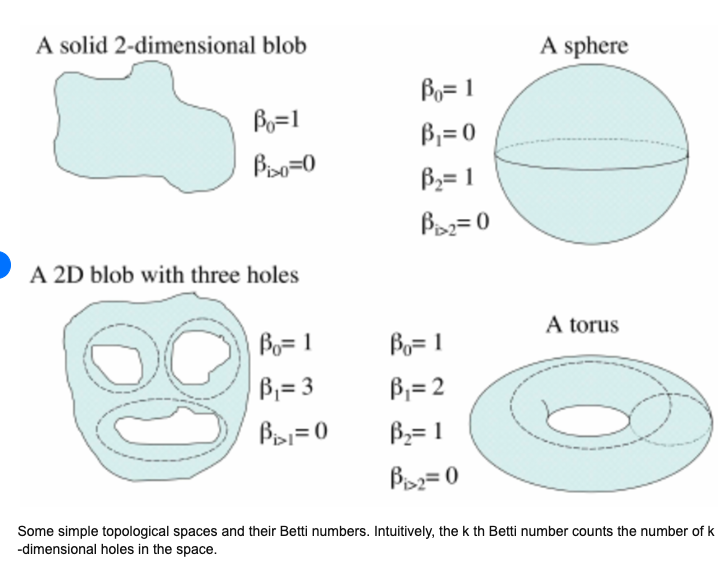
INTRODUCTION TO PERSISTENT HOMOLOGY:

* In classic Machine Learning Algorithms, we produce the projections of the data points on 2-dimension or 3-dimension and then plot those projections on the respective coordinates.
* But in Topological Data Analysis, we represent the data by nodes and edges where nodes are a group of similar data points, if these nodes have any common data point then we connect those nodes with an edge.
* This representation is known as a Topological Network. We use methods like persistent homology for measuring these topological networks.
* Persistent homology is a method in applied mathematics for computing topological features of a space at different spatial resolutions.
* To find persistent homology of space first we need to represent the space as simplicial complex.
* A distance function underlying space corresponds to a filtration of the simplicial complex.

CALCULATING PERSISTENT HOMOLGY FOR A GIVEN DATA SET:

* For any given data set, we consider the given data as a point cloud, a point cloud in general is defined as finite set of points in some Euclidean space or any metric space
* Using Persistent homology, we can calculate Persistent diagrams which in general are referred as Persistent barcodes.
* A barcode is a graphical representation of point cloud as a collection of horizontal line segments in a plane whose horizontal axis corresponds to parameter and vertical axis represents ordering of homology generators.
* Figure below shows generating barcodes from point cloud.
* 
* H0, H1 and H2 represents 0-dim ,1-dim and 2-dim homology for the given high dimensional data , the line in the barcode which continues till infinity is the one which persists.

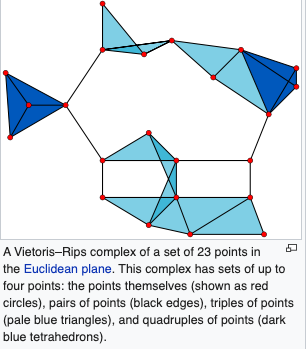
BETTI NUMBERS:

* In topology Betti numbers are used to distinguish topological spaces based on connectivity of n-dim simplicial complexes.
* For most finite complexes Betti numbers starts from 0 and from some point onward Betti numbers vanish above the dimension of a space.
* The Nth Betti number represents the rank of nth homology group, which tells us maximum number of cuts that must be made before forming 0-cycles ,1-cycles etc.
* In general, kth Betti number tells us the number of k-dim holes on a topological space.
* b0, b1, b2 represents number of connected components, circular holes and voids/cavities in shapes respectively.
* 
* Betti Numbers are used to differentiate between topological shapes using the concept of holes and voids present in the shape.

RIPS COMPLEXES:

* In Topology Vietoris RIPS complex is an abstract simplicial complex that can be defined from any metric space M and distance d by forming a simplex for every

finite set of points that has at most diameter d.



METHODS USED TO CLASSFIY HANDWRITTEN ALPHABETS USING ABOVE MENTIONED TOPLOGICAL CONCEPTS:

TASK: CALSSIFY THE ALPHABETS USING A MATRIX OF 0’S AND 1’S DATA FROM MANAULLY LABELLED DATA AND PREDICT A UNKNOWN ALPHABET WITH ADDED NOISE.

DATA COLLECTION:

* The data for this project is manually labelled (marked 0’s and 1’s) from the images.
* Marked data for all 26 alphabets and consolidated in a single CSV file.

SOFTWARE USED:

* Python 3, Jupyter Notebook, SCIKIT-TDA RIPSER CODE FROM GITHUB.
* Implemented the whole Project based on the idea of Ripser module from SCIKIT-TDA.
* Made use of Lower Star image filtrations to display the alphabets.

ALGORITHM STEPS:

1. Developed multiple scanning techniques namely right-to-left, left-to-right, up-don, down-up and middle-out scanning.
2. Used the above-mentioned scanning techniques as filtrations to classify the data.
3. The data is read as point cloud and then used the filtration methods to generate Betti numbers for 0-dim and 1-dim.
4. Using the Betti numbers generated, created a classification vector which has weights of the corresponding signatures (DataVec).
5. Then compared the unknown letter with the existing DataVec from the step5, using the minimum distance metric.

TESTS:

* Added Noise to the data collected and also randomly generated noise data.
* Tested the Nosie data on all the letters and compared the classification rate.

Higher the classification rate (close to 1), better is the prediction of unknown letter.