Clustering Techniques

A Thesis

submitted to
Indian Institute of Science Education and Research Pune
in partial fulfillment of the requirements for the
BS-MS Dual Degree Programme

by

Aysha Basheer

Indian Institute of Science Education and Research Pune
Dr. Homi Bhabha Road,
Pashan, Pune 411008, INDIA.

December, 2020

Supervisor: Uttara Naik Nimbalkar

Certificate

This is to certify that this dissertation entitled Clustering Techniques towards the partial fulfilment of the BS-MS dual degree programme at the Indian Institute of Science Education and Research, Pune represents study/work carried out by Aysha Basheerat Indian Institute of Science Education and Research under the supervision of Uttara Naik Nimbalkar, Professor, Department of Mathematics , during the academic year 2019-2020.

U. Naik-Nimbalkar

Uttara Naik Nimbalkar

Committee:

Uttara Naik Nimbalkar U. Naik-Nimbalkar

Prof. Pranay Goel

The world is one big data problem.

Declaration

I hereby declare that the matter embodied in the report entitled Clustering Techniques are the results of the work carried out by me at the Department of Mathematics, Indian Institute of Science Education and Research, Pune, under the supervision of Uttara Naik Nimbalkar and the same has not been submitted elsewhere for any other degree.

Aysha Basheer

Acknowledgments

First and foremost, I would like to express my sincere gratitude towards my supervisor Prof. Uttara Naik - Nimbalkar for the continuous support, for her patience, and immense knowledge. Thank you for believing in me and always being there for me with your guidance, encouragement and for inspiring me to pursue research in applied mathematics. I am deeply grateful to her for helping me build the foundations and follow my interest throughout the last few years.

Besides my advisor, I would like to thank Prof. Pranay Goel for his precious suggestions and insightful comments, which give me great help in improving my work.

A special big thanks to Adish Assain for helping me to question my work, leading up to a better thesis, stimulating discussions and the constant moral support.

Finally, I must express my very profound gratitude to my family for providing me with unfailing support and continuous encouragement throughout my years of study and through the course of the project and writing this thesis. This accomplishment would not have been possible without them.

Abstract

Clustering is one of the most widely researched areas in unsupervised learning, where the main aim is to find structures in unlabelled data sets. This is done by partitioning data set into smaller groups or clusters so that the data points in the cluster have more common features among themselves compared to those in other clusters. There are plenty of different types of clustering techniques starting from the classical to the more recent ones based on the topological and geometrical methods. It has wide application across various fields.

Different types of hierarchical, partitioning and density-based clustering algorithms are studied along with topological data analysis based clustering using persistent homology. The real data sets contain both numerical and categorical variables, which makes it difficult to cluster. Different approaches and few techniques for clustering mixed data sets are discussed.

The objective is to study all these techniques and their limitations complemented by two real-life application in business and physical science fields.

Contents

Chapter 1

Introduction

Clustering is one of the most popular and widely used techniques in various disciplines such as statistics, biology, social science, Image segmentation, software engineering to identify natural groups and structures in a data set. It is a method of unsupervised learning, of identifying similar clusters within the data and are grouped into different groups. The similarity or dissimilarity between two objects in a data set is usually measured as the distance between the vectors representing the objects. These objects are multidimensional variables, also called attributes or features. In this thesis, we refer to these variables as attributes. There are hundreds of different types of clustering techniques starting from the classical to the more recent ones based on the topological and geometrical methods. The popular clustering techniques are hierarchical clustering, partitioning clustering methods like k-means and partition around medoids, density clustering methods like DBSCAN, mode clustering and level set clusters.

Algebraic topology is a branch of mathematics that deals with using abstract algebra to study topological spaces. Topological Data Analysis is a recent and most popular research area which uses tools from topology and statistics to study and analyse structures in the data sets. The approach here is, given a point cloud in a metric space and assuming data is derived from a manifold, algebraic topology is used to measure the persistence of homology groups and classifies the point cloud geometrically. ToMATo is Topological Mode Analysis Tool used to cluster data using persistent homology.

Most of the clustering techniques are applicable either on numerical or categorical data

sets, but not applicable to mixed data sets. To cluster a mixed type data set is still a challenge. There are many approaches where algorithms used for clustering single data type is modified and used on mixed-type data sets. However, they all suffer mainly from loss of information. KAMILA is a recent approach which addresses the problems faced by other algorithms and produces more stable clusters.

This project is aimed at studying different clustering algorithms, as well as the implementation of the same on real-world data sets. The techniques studied are applied to an air quality data set to understand the healthy/less polluted places before and during COVID-19 lockdown in India across different cities. The algorithms studied are tried on a mixed data set; Online Shoppers Intention to analyse the behaviour of online shoppers.

Chapter 2

Clustering

Clustering is the process of grouping unsupervised data into smaller homogeneous groups or clusters. Without any prior information about the classes, clustering methods identify different classes from the data set. The points within the cluster are similar to each other while they are dissimilar to points in the other clusters. The properties and similarity measure of the clusters varies between applications. Veenman et al. 2003 [8] defined cluster as

Definition 2.0.1. *Given a data set* $\mathbb{X} = x_1, x_2, ..., x_N$*, where* x_i *is the vector of attributes in* p *dimensional metric space, and* N *is the number of objects in* $\mathbb{X}$*. Then* $\mathbb{C} = \{C_1, C_2, ..., C_M\}$ *is the set of* M *partitions of* $\mathbb{X}$ *and satisfies the following properties*

- $\mathbb{X} = \bigcup_{i=1}^{M} C_i$
- $C_i \neq \phi, 1 \leq i \leq M$
- $C_i \bigcap C_j = \phi, i \neq j, 1 \leq i, j \leq M$

2.1 Similarity and Dissimilarity Measures

Clustering algorithms group data points into different clusters based on the similarity or dissimilarity between them. The measure of dissimilarity or similarity of one point with

respect to another is the basic tool for clustering. Similarity measures are used to describe quantitatively how similar two data points are or how similar two clusters are. The similarity measure is high for points within the cluster. The greater the dissimilarity measure, the more dissimilar are the two data points, or the two clusters [2]. The most commonly used dissimilarity measure is the distance measure.

Definition 2.1.1. *Let x_i, x_j be any two points in R^d and $d_{ij} = d(x_i, x_j)$, i, j = 1, 2,...,n. The dissimilarity must satisfy the following properties.*

1. $d(x_i, x_j) \geq 0$ *[Non-negativity]*
2. $d(x_i, x_j) = 0 \iff x_i = x_j$ *[Reflexivity]*
3. $d(x_i, x_j) = d(x_j, x_i)$ *[Symmetry]*
4. $d(x_i, x_j) \leq d(x_i, x_k) + d(x_k, x_j)$ *[Triangle Inequality]*

The dissimilarity is defined by various distance metrics. Some of them are

- **Euclidean** - The most widely used metric for continuous data, which is the straight line distance between two points.

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{d} (x_{ik} - x_{jk})^2}$$

- **Manhattan**- This is also used for the continous data, here distance between two points is the sum of differences of cartesian coordinates.

$$d(x_i, x_j) = \sum_{k=1}^{d} |x_{ik} - x_{jk}|$$

- **Gower** - This is used when the data set contains both numerical and categorical attributes.

$$d(x_i, x_j) = \sqrt{\frac{1}{\sum_{k=1}^{d} w(x_{ik}, x_{jk})} \sum_{k=1}^{d} w(x_{ik}, x_{jk}) d_k^2(x_{ik}, x_{jk})}$$

$d_k^2(x_{ik}, x_{jk})$ is the squared distance and $w(x_{ik}, x_{jk})$ is either 0 or 1 depending on whether or not a comparison is valid for kth attribute. They are defined differently for different types of attributes [2].

- For ordinal and continuous attributes,

$$d_k(x_{ik}, x_{jk}) = \frac{|x_{ik} - x_{jk}|}{R_k}$$

  where R_k is the range of the kth attribute.

- For quantitative attributes,

$$d_k(x_{ik}, x_{jk}) = |x_{ik} - x_{jk}|$$

  and $w(x_{ik}, x_{jk}) = 0$ if there is a missing value at the kth attribute; otherwise $w(x_{ik}, x_{jk}) = 1$.

- For binary attributes, $d_k(x_{ik}, x_{jk}) = 0$ if both x_i and x_j have the kth attributes present or absent; otherwise $d_k(x_{ik}, x_{jk}) = 1$. $w(x_{ik}, x_{jk}) = 0$ if both data points x_i and x_j have the kth attribute absent; otherwise $w(x_{ik}, x_{jk}) = 1$.

- For nominal or categorical attributes, $d_k(x_{ik}, x_{jk}) = 0$ if both $x_i = x_j$; otherwise $d_k(x_{ik}, x_{jk}) = 1$. $w(x_{ik}, x_{jk}) = 0$ if there is a missing value at the kth attribute; otherwise $w(x_{ik}, x_{jk}) = 1$.

- **Hamming** - This metric is used for comparing two binary data strings, that is number of points at which two binary data differs. It is the subset of Gower distance metric and we use the same equation of Gower for computing Hamming distance.

2.2 Hierarchical Clustering

As the name suggests hierarchical clustering forms a nested sequence from the linkage between data points, which forms the hierarchy of clusters. Hierarchical clustering are of two types - **agglomerative** and **divisive**.

The agglomerative algorithm also called bottom-up methods starts with each point being its own cluster and then closest clusters are successively merged until single cluster remains.

The divisive algorithm also called top-down method is the reverse of agglomerative method, it begins with the whole set as a single cluster and then divides into smaller clusters.

Given data set $\mathbb{D} = x_1, ..., x_n$, distance metric is used to calculate the dissimilarity between the data points. $\mathcal{D} = (d_{ij})$ be the matrix of dissimilarities between the n data points and $d_{ij} = d(x_i, x_j), i, j = 1, ..., n$. In the case of agglomerative method, each data point is considered as a cluster, and the smallest dissimilarity in $\mathcal{D}$ is merged to form a cluster, say I and J are two clusters merged to form IJ cluster. Then dissimilarity between IJ cluster and other clusters K $\neq$ IJ are computed. This dissimilarities depend upon the linkage method used. The linkage method is needed to compute the distance between the clusters and to merge them. The different types of linkage methods are :

- Single linkage - This method calculates the shortest distance between the datapoints in each cluster and merges two clusters whose distance between them is the smallest.

$$d_{IJ,K} = min(d_{I,K}, d_{J,K})$$

- Complete linkage - This method calculates the largest distance between the data points in each cluster and merges two clusters with largest distance between them.

$$d_{IJ,K} = max(d_{I,K}, d_{J,K})$$

- Average linkage - This method calculates the average distance between each datapoint in a cluster to every datapoint in the other cluster.

$$d_{IJ,K} = \frac{\sum_{i \epsilon IJ} \sum_{k \epsilon K} d_{ik}}{(N_{IJ} N_K)}$$

  where N_{IJ} and N_K are the number of items in IJ and K clusters respectively.

This is repeated until all the items are merged into a single cluster. In the end, we obtain a hierarchical tree diagram, that is dendrogram. Dendrogram has nodes indicating clusters or the point at which clusters combine and lines connected to the nodes indicating the clusters which are nested into one another. The height is the vertical distance between nodes, the difference in heights indicates how close the points are. The dendrogram is cut horizontally to get different clusters. Based on the dendrogram, we calculate the number of

clusters. It is a challenging problem to decide where to cut the dendrogram, i.e., to decide the number of clusters. There is no definite method, most cases it is context-dependent and from a theoretical point of view. One method is to plot the number of clusters against the dissimilarity and choose the cluster number corresponding to the knee of the plot.

Advantages

- Of all the clustering algorithms, this is one of the most simple to understand and easiest to implement.
- No prior information about the number of clusters required.

Disadvantages

- Once decided to merge the clusters, it cannot be undone.
- The real world applications are mostly high- dimensional, which makes it difficult to visualize and extract the clusters from the dendrogram.
- With mixed data type, it is difficult to compute the distance matrix.
- Not sensitive to noise and outliers.

2.3 Partitioning clustering

Data points are grouped into predetermined k number of clusters by splitting or merging them and iteratively reassigning them into better suitable groups till it reaches optima. Unlike hierarchical clustering, there is no hierarchical relationship between the clusters, that is data points can change between the cluster during the process.The two popular methods are **K-means** and **PAM**.

2.3.1 K-Means

This method begins by either assigning data points to predetermined K clusters and then calculating the centroids or by assigning the data points to the cluster with the nearest randomly chosen centroid and then recomputing the centroid as new items are added and lost until an optimum is reached. This happens iteratively and stops when (Sum of Squared Error) ESS can't be reduced further, that is the sum of the squared distance between the data points and the cluster's centroid is at minimum. This is the most popular and simplest unsupervised clustering algorithm.

Let $x_1, x_2, ...$ be the data points and k be the number of clusters required. The k centers $c_i \in C, i = 1...k$ are randomly chosen and distance $dist(c_i, x)$ between each data point from the data set and cluster center is calculated, 2.1. Then each data point is assigned to the nearest center based on the distance that is,

$$\underset{c_i \in C}{argmin} dist(c_i, x)$$

After all the data points are assigned to each cluster, the average of all the data points in the cluster is calculated to get the mean. If $\mathbb{C}_i$ is the set of all datapoints in the ith cluster then new centroid is

$$c_i = \frac{1}{|\mathbb{C}_i|} \sum_{x_i \in \mathbb{C}_i} x_i$$

This process is repeated until there is no change in the cluster mean.

K means clustering algorithm gives us k cluster centers whose ESS is minimum. The k is subjective and depends on the similarity measure used and the the parameters, hence it's difficult to decide the value of k. Elbow method and gap statistic gives us an idea to choose the value of k. Elbow method calculates the within sum of square error (WSS) for different values of k and the WSS is plotted against k. The bend in the plot indicates the appropriate number of clusters. Gap statistic is the statistical testing method where within intra cluster variation is calculated for different values of k, W_k for the given data set and W_{kr} null reference data set with random distribution. Then gap statistic is the deviation of W_k from the expected value of W_{kr} under null hypothesis. The estimate k will be the one which maximizes the gap statistic [16].

2.3.2 Partitioning Around Medoids (PAM) or K Medoid Clustering

This is similar to the k-means clustering, instead of centroids, this algorithm begins by searching for k- medoids, where medoid is a point in the cluster which has minimum dissimilarity with other points in the cluster and then selected points are interchanged with unselected to improve the quality of the cluster. For large data sets, the final results may vary after each run because it starts by selecting random points as medoids.

Advantages

- Fast, simple and easier to understand and imply.
- Generalizes to clusters of different shapes and sizes, such as elliptical clusters.

Disadvantages

- It is difficult to handle noisy data or outliers. The outliers instead of ignored they get their own clusters.
- Predicting k in prior is not easy.
- The clusters depend upon initial values. Different results are obtained when initial partitions or centroid values are changed.
- Fails for non linear data.

2.4 Density clustering

Density-based clustering is a method to group similar data points by identifying regions of high density and separating different groups by regions of low density which contains noise points. This algorithm unlike partitioning cluster is not biased towards hyper spherical or convex shaped clusters and forms clusters of arbitrary shape. This doesn't require the

number of clusters as input. The most popular method is Density-Based Spatial Clustering of Applications with Noise **(DBSCAN)**.

2.4.1 DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

DBSCAN is one of the most widely used density algorithm which can find the natural clusters in the data space without any prior knowledge of the clusters present in the data set [17]. It is an high performance algorithm which can find clusters of arbitraty shape and identify the noise points. It starts with an arbitrary point x and finds all the point within the given radius. The clusters resulting from this algorithm are the high density regions in the data space. The algorithm requires two parameters, ϵ and N_{min}, where ϵ is the radius and N_{min} is the number of minimum points, i.e, the neighbourhood of radius ϵ must contain N_{min} to form a cluster. This algorithm doesn't require the number of clusters as input. The idea of DBSCAN can be explained using the following definitions [12],

Definition 2.4.1. ***Epsilon neighbourhood of a point*** *- Let p be a point in the data set $\mathbb{X}$, the ϵ neighborhood of a point p, denoted by $N_{(\epsilon)}p$, is defined as $N_{(\epsilon)}p = \{q \in \mathbb{X} \mid d(p,q) \leq \epsilon\}$, where d is the distance measure and $\epsilon \geq 0$.*

Density reachablity and density connectivity are the main concepts used for performing DBSCAN.

Definition 2.4.2. ***Directly density reachable*** *- p is directly density reachable from q w.r.t ϵ and N_{min}, if*
a) $p \in N_{(\epsilon)}q$
b) $|N_{(\epsilon)}q| \geq N_{min}$.

Definition 2.4.3. ***Density reachable*** *- p is density reachable from q with respect to ϵ and N_{min} if there is a chain of points p_1, ... p_n, $p_1 = q$, $p_n = p$ such that p_{i+1} is directly density-reachable from p_i.*

Definition 2.4.4. ***Density Connectivity*** *- p and q are density connected if there exists r which has sufficient number of points in its neighborhood and both the points p and q are within the ϵ distance.*

Ester et al. [12] defined cluster as

Definition 2.4.5. *A **cluster** C is a non-empty subset of data set $\mathbb{X}$ satisfying the following conditions*

1. *Maximality - $\forall\ p \in C$ and if q is density-reachable from p, then $q \in C$.*
2. *Connectivity - $\forall\ p,\ q \in C$, then p is density-connected to q.*

Three types of points can be distinguished in a cluster,

The **core points** are the points in the inside of the cluster which has more than N_{min} points within the ϵ radius.

The **border points** are the points on the border of the cluster which are significantly less in number than the core points.

The **Noise** is a set of points in data set $\mathbb{X}$, which doesn't belong to any cluster C_i.

The algorithm begins by visiting any point p and extracting every density reachable points with respect to ϵ and N_{min}, forming an ϵ neighbourhood. If p is a core point, then it forms a cluster. If p is a border point, then no points are density reachable from p and then next point is visited [12]. This repeats until all points are visited. The points which are not in the clusters are considered as noise points.

To apply this algorithm on the data set we need to decide on the ϵ and N_{min} values. The N_{min} should be at least one point plus the dimension of the data set. The points within the cluster are close to each other so they will have small k-nearest neighbour (KNN) distance and the noise points will have largest distance. Plotting the k-nearest neighbour distance (i.e., the distance to the kth nearest neighbor) in descending order, the threshold point is where the KNN distance curve bends. This can be used as optimal ϵ value.

Advantages

- No prior information about cluster numbers required.
- Noise points are identified.
- Can handle clusters of different shapes and sizes.

Disadvantages

- Fails for high dimensional and varying density clusters.
- Sensitive to parameters, epsilon and minimum points.

2.5 Kernel Density Estimation

Kernel density estimation (KDE) is a non parametric density estimation method.The density-based algorithm requires to estimate the underlying probability density from the sample data.

Definition 2.5.1. *Let $\mathbb{X}$ be the d-dimensional Euclidean space, R^d. A function $K : \mathbb{X} \to R$ is said to be a kernel if it satisfies the following properties,*

a. K is non negative; $K(x) \geq 0$.

b. $\int K(x)dx = 1$..

c. $\int xK(x)dx = 0$.

d. $\int xK(x)dx < \infty$

Definition 2.5.2. *Given a set of data $x_1, \ldots, x_n \subset R^d$, kernel K and a positive number h, called the bandwidth or smoothing parameter, the kernel density estimator is defined as*

$$\hat{f}(x) = \frac{1}{nh^d} \sum_{i=1}^{N} K(\frac{x - x_i}{h}) \tag{2.1}$$

Given the bin width, this function places symmetrical humps or kernels over each data point, see Figure 2.1, and the distance from a reference point is calculated. Sum of the individual kernels gives us the density estimate for the distribution [18]. The most commonly used kernel is Gaussian Kernel,

$$K(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$$

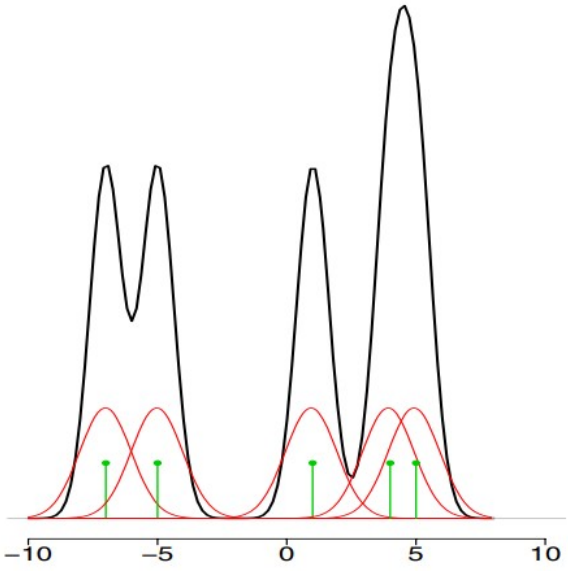

Figure 2.1: A kernel density estimator $\hat{f}_h(x)$. At each point x, $\hat{f}_h(x)$ is the average of the kernels centered over the data points x_i. The data points are indicated by short vertical bars. Source : Wasserman, 2006 [24]

On the d-dimensional space the kernel is defined as

$$K(x_1, x_2, ..., x_d) = \prod_{i=1}^{d} K(x_i)$$

To obtain a good estimate we need to adjust the bandwidth h accordingly. The value of h which minimizes the integrated square error (ISE) is optimum. Large value of h gives smoother estimates while smaller values gives rough estimates. Some of the properties of KDE are given below.

Theorem 2.5.1. *Assume that f is continuous at x and that $h \to 0$ and $nh \to \infty$ as $n \to \infty$. Then $\hat{f}_h(x) \to^P f(x)$*

As the sample size tends to infinity, the density estimate converges to it's true value in probability, then $\hat{f}_h$ is said to be a *consistent estimator* of f.

Theorem 2.5.2. *Under the assumption in Theorem 2.5.1, the bias and variance of the KDE at x are*

$$Bias[\hat{f}_h(x)] = \frac{1}{2}\mu_2(K)f''(x)h^2 + O(h^2) \tag{2.2}$$

$$Variance[\hat{f}_h(x)] = \frac{\int K^2(x)dx}{nh} f(x) + O(nh)^{-1} \tag{2.3}$$

Corollary 2.5.1. *Under the definition 2.5.1, the MSE (Mean Squared Error) of the KDE at x is*

$$MSE[\hat{f}_h(x)] = \frac{(\mu_2)^2(K)}{4}(f''(x))^2 h^4 + \frac{\int K^2(x)dx}{nh} f(x) + O(h^4 + (nh)^{-1}) \tag{2.4}$$

If f is square integrable, then performance of $\hat{f}_h$ at $x \in R^d$ is measured by MSE,

$$MSE(x) = E_f\{\hat{f}_h(x) - f(x)\}^2$$

If MSE(x) $\to 0$ for all $x \in R^d$ as $n \to \infty$, then $\hat{f}_h$ is said to be *pointwise consistent estimator of f in quadratic mean.*

KDE can be extended to estimate multivariate densities f in R^d based on the same principle 2.5.1. Given $\mathbb{X}_i, \mathbb{X}_2, ..., \mathbb{X}_n$ in R^d, the KDE of f at $x \in R^d$ is

$$\hat{f}_H = \frac{1}{n|H|^{1/2}} \sum_{i=1}^{n} K(H^{-1/2}(x - \mathbb{X}_i)) \tag{2.5}$$

and the where K is the multivariate kernel and H is the bandwidth matrix; fxf symmetric and positive definite matrix. The most commonly used multivariate kernel is normal kernel,

$$K(x) = \frac{1}{\sqrt{2\pi}^f} e^{-1/2} x'x$$

2.6 Level Set Clusters

The level set clustering is a non parametric density clustering method.

Definition 2.6.1. *Let $x_1, \ldots, x_n$ be a random sample from a distribution $\mathbb{F}$ with density f ; $x_i \in \mathbb{X} \subset \mathbb{R}^d$. $\forall\ t \geq 0$, the upper level set is,*

$$L_t = \{x : f(x) \geq t\}$$

To define the density of level set, kernel density estimate (KDE) is used. Kernel density

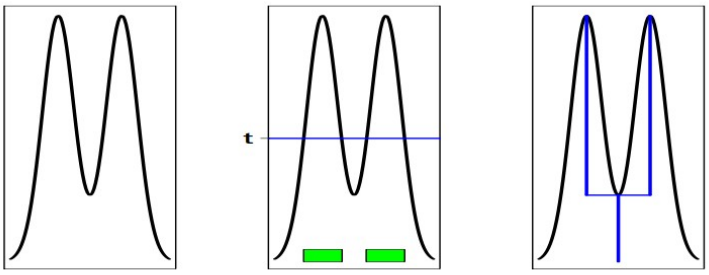

Figure 2.2: Left: A density function f. Middle: density clusters corresponding to $L_t = \{x : f(x) \geq t\}$. Right: the density tree corresponding to f is shown under the density. The leaves of the tree correspond to modes. The branches correspond to connected components of the level sets. Source : Wasserman, 2016 [1].

.

estimate doesn't care about the specific shape of the level set [19]. Given kernel K on $\mathbb{R}^d$ and bandwidth h > 0, such that h $\to 0$ as n $\to \infty$ [19], $\hat{f}_h$ be the KDE. Then the estimate of L_t is $\hat{L}_t = \{x : \hat{f}_h(x) \geq t\}$.

A high-density cluster is a maximal connected component of $\hat{L}_t$ for any t and the level set tree τ is simply the set of all such clusters. $\mathbb{C} = \bigcup_{t \geq 0} C_t$, where C_t is the density level clusters at level t. To find the connected components of the $\hat{L}_t$, Let $I_t : \{i : \hat{f}_h(x_i) > t\}$. Now create a graph whose nodes corresponds to $(x_i : i \in I_t)$ and put an edge between x_i and x_j if $\|x_i - x_j\| < \epsilon$ where $\epsilon > 0$ is a tuning parameter. Then connected components C_1, C_2, ... of the graph estimate clusters at level t, Figure 2.2.

2.7 Mode Clustering

This is another non-parametric density clustering method. In this method, we start by estimating the density function and then finding the modes of the estimator and then clusters are defined as the basins of attraction of these modes. The kernel density estimator is used to find the density estimator and mean shift algorithm finds the modes and basins of attraction [4].

Let $x_1, \ldots, x_n$ be a random sample with density $\hat{f}$; $x_i \in \mathbb{X} \subset R^d$. Let's assume $\hat{f}$ has

compact support $K \subset R^d$ and has k local maxima M = $m_1, \ldots, m_k$ and is a morse function.

Definition 2.7.1. *A function $f : M \to R$ is a Morse function if its critical points are non-degenerate, i.e., the Hessian of f at each critical point is non-singular.*

A point m is called local mode if there exists an open neighbourhood N of x such that f(x)>f(y) for all y element of N and x not equal to y. Since f is assumed to be morse, the m is a local mode if and only if $g(m) = (0, ..., 0)^T$ and $\lambda(H(m)) < 0$, where λ is the largest eigen value of H(m).

According to Morse theory, integral curves never intersects except at the critical points. The integral curve through x is a path $\pi_x : R \to R^d$ such that $\pi_x(0) = x$ and

$$\pi^{'}(t) = \nabla f(\pi_x(t))$$

. Now let's define the destination of the integral curve beginning at x as

$$dest(x) = \lim_{t \to \infty} \pi_x(t)$$

. Following the steepest ascent path we will reach a mode (true except for Lebesgue measure 0), then $dest(x) = m_j$. The basin of attraction of m_J is defined by

$$C_j = \{x : dest(x) = m_j\}, j = 1, ..., k$$

. The set $C_1, ..., C_k$ are called the population clusters. Now we use the mean shift algorithm described below to find the modes $\hat{M} = \{\hat{m}_1, ..., \hat{m}_k\}$. The estimated basins of attractions are

$$\hat{C}_j = \{x \in R^d : \hat{dest}(x) = \hat{m}_j\}, j = 1, ..., \hat{k}$$

and the clusters are $X_j = \{X_i : X_i \in \hat{C}_j\} = \{X_i : \hat{dest}(X_i) = \hat{m}_j\}$.

2.7.1 Mean Shift Algorithm

The mean shift algorithm is a non parametric method to estimate the density gradient, seeks a mode or local maximum of density of a given distribution. It begins by choosing a search window, and computing the mean of the data in the search window. Then the search window

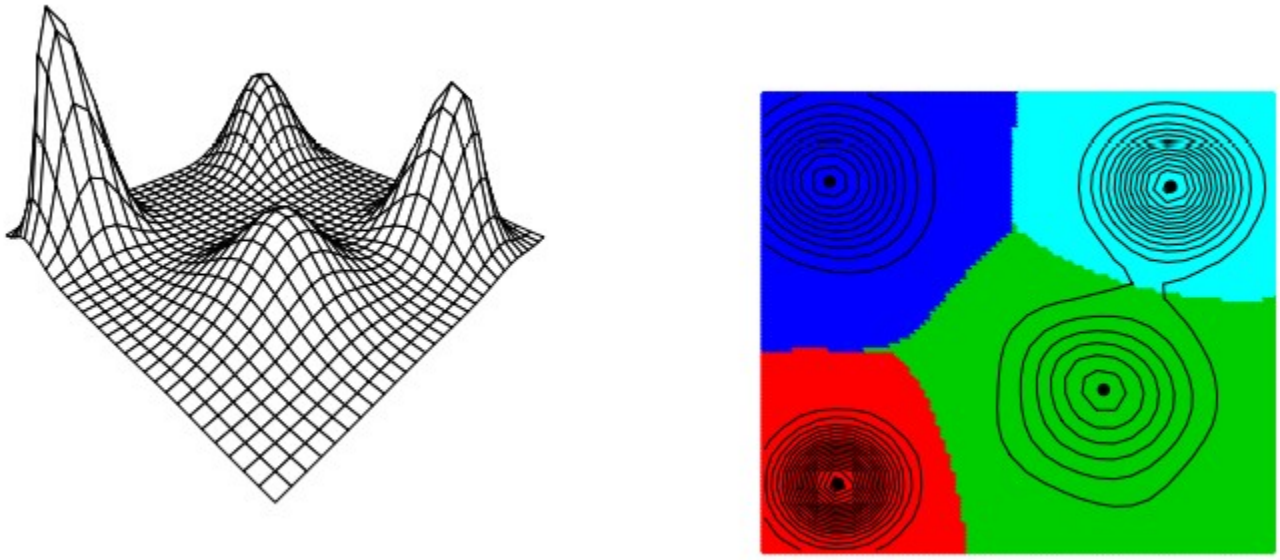

Figure 2.3: Left: a density with four modes. Right: the partition (basins of attraction) of the space induced by the modes. These are the population clusters.[1]

is centered at the new mean location and repeated until convergence. Comaniciu et al, [25] explained the theory and approach behind Mean Shift algorithm in detail.

Advantages

- Insensitive to initialization and outliers.
- Uses a specific kernel and models complex clusters having non convex shapes.

Disadvantages

- Only one parameter bandwidth is required, but it is difficult to choose the bandwidth value.
- It doesn't works well in higher dimensions, KDEs breaks down.
- It is difficult to determine meaningful and non-meaningful modes.

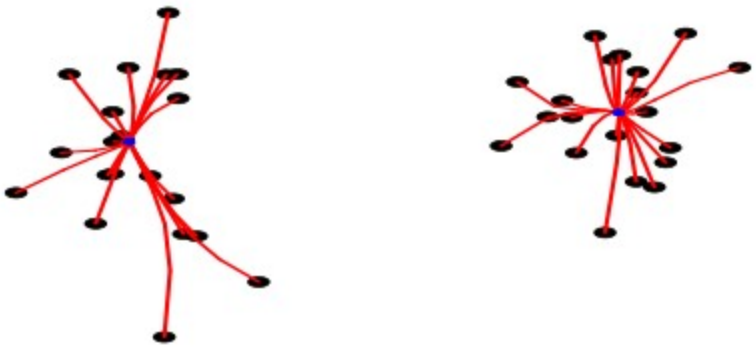

Figure 2.4: The mean shift algorithm. The data are represented by the black dots. The modes of the density estimate are the two blue dots. The red curves show the mean shift paths; each data point moves along its path towards a mode as we iterate the algorithm.[1]

- Fails when clusters overlap.

Chapter 3

Topological Data Analysis

3.1 Introduction

Topological data analysis is basically set of statistical, mathematical and algorithmic methods to analyse and find the topological and geometrical structures in a data set. This helps us to analyze and understand high dimensional and complex data sets. This method was first introduced in 2000 by Edelsbrunner, Letcher and Zomorodian [21]. TDA has lots of applications including clustering algorithms, image processing, neuroscience, shape segmentation etc.

Persistent homology is a topological data analysis method used to analyse the qualitative (topological) features of a data set which leads to a persistent diagram. The connected components of a topological space is detected by the homology and persistence homology assigns birth and death values to measure the features. By qualitative features it indicates clusters, cycles, flares etc in a data set.

3.2 Persistent Homology

Given a data set that lies in a metric space with a distance measure, algebraic topology computes the characteristics of the data such as connected components or existence of holes

by associating vector spaces or simply by counting them or by associating complex algebraic structures.

Homology associates vector space $H_i(X)$ to space X. $H_0(X)$ counts the number of path components in X, $H_1(X)$ counts the number of holes and $H_2(X)$ counts the number of voids. These numbers are called Betti numbers and represented by β_0, β_1, ... For arbitrary topological spaces it's hard to capture the homology, thus simplicial complexes are used to estimate the homology which can be computed algorithmatically [21].

Definition 3.2.1. ***Simplicial Complex*** *Let X be a discrete set. An abstract simplicial complex is a collection C of finite subsets of X such that if $\sigma \in \mathcal{C}$ then $\tau \in \mathcal{C}$ for all $\tau \subseteq \sigma$. If $|\sigma| = k + 1$ then σ is called a k-simplex.*

Simplicial complexes are the sets composing points, edges, triangles and higher dimensional polytopes. There are many types of simplicial complexes and the choice of them depends on the nature of data, computational cost etc. The most widely used one is Vietoris Rips Complex.

Definition 3.2.2. ***Čech complex*** *Let P be a finite set of points in $R^n, and B_x(r)$ be a ball with center $x \in \mathbb{R}^n$ and radius $r \in \mathbb{R}$, the the Čech complex of P and r is*

$$\mathcal{C}(r) := \{\sigma \subseteq P | \bigcap_{x \in \sigma} B_x(r) \neq \phi\} \tag{3.1}$$

Definition 3.2.3. ***Vietoris–Rips complex*** *Given a scale parameter r and a finite set of points P, the Vietoris–Rips complex is defined as the simplicial complex that contains all subsets whose diameter is at most r:*

$$\mathcal{V}(r) := \{\sigma \subseteq P | diam\sigma \leq r\} \tag{3.2}$$

Let $\mathcal{U}$ be a cover of X i.e., a collection of subsets of X such that the union of the subsets is X. The k-simplices of the Čech complex are the non-empty intersections of k+1 sets in the cover $\mathcal{U}$ [21]. The nerve of a collection of sets is defined as,

Definition 3.2.4. *Let $\mathcal{U} = \{U_i\}_{i \in I}$ be a non-empty collection of sets. The nerve of $\mathcal{U}$ is the simplicial complex with set of vertices given by I and k-simplices given by $i_0, \ldots, i_k$ if and only if $\bigcap_{j=0}^{k} U_{i_j} \neq \phi$.*

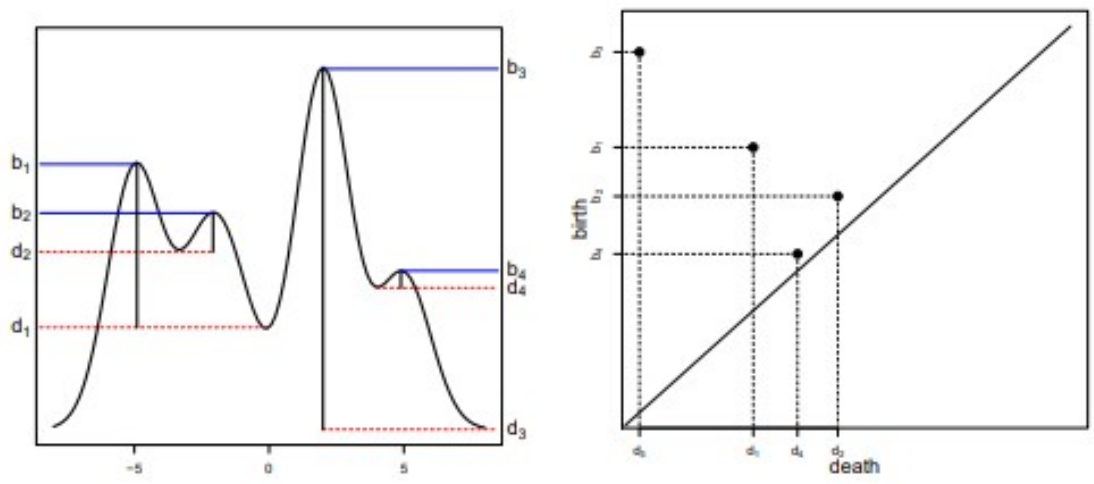

Figure 3.1: The birth and death time of modes (left) and the persistence diagram (right). Source : Wasserman, 2016 [1]

Theorem 3.2.1 (Nerve Theorem)**.** *Let* $\mathcal{U} = \{U_i\}_{i \in I}$ *be an open cover of a topological space* X *by open sets such that intersection of any sub collection of* U_i*'s is either empty or contractible. Then* X *and the nerve of* $\mathcal{U}$ *are homotopicaly equivalent.*

Let $X_i, ..., X_n$ be the data points. $B(x, \epsilon)$ represent ball with a radius ϵ centred at x. $B(X_1, \epsilon), B(X_2, \epsilon), ..., B(X_n, \epsilon)$ represents the set of balls around each data point. As the value of ϵ increases, the topological features also changes. When $\epsilon = 0$, there will be no connected components, as the value of ϵ increases the connected components are formed and merged until one component is left, at a particular value of ϵ, a hole will appear (birth) and the hole will disappear (death) at a larger ϵ value. The birth and death point of each feature is recorded as a bar in a barcode plot and a persistence diagram is made with each feature as a point on the diagram with coordinates representing birth and death Figure 3.1.

3.3 Persistence Based Density Clustering

The mode clustering algorithm defined in section 2.7 estimates the KDE, finds the modes and the basins of attractions by using Mean Shift Algorithm [2.7.1]. Now, we can use persistence homology to detect and merge unstable modes [22]. The prominence of a peak is the height difference between the height of the peak and level at which the basin of attraction meets it's

parent peak. The persistence computes the prominence of the density peaks and a hierarchy of the peaks based on it are created.

Let $t = sup_x f(x)$, where f is the density function and L_t be the upper level set, $L_t = \{x : f(x) \geq t\}$. As the t changes from [-$\infty, +\infty$], when new modes are formed, new connected components of L_t are born and then died by merging with the other connected component. From this we can say each mode has a lifetime and this lifetime can be plotted as points in the plane with x-coordinate representing birth and y-coordinate representing the death time. This plot is called the persistence diagram (PD) of f. The modes which are far from the diagonal ($y = x$) are the stronger modes and they are the level sets with more lifetime and the small modes have short lifetime level sets. Any mode whose point PD is farther from the diagonal is considered as significant mode [1]. The clusters are obtained by merging the peaks of prominence less than a given thresholding parameter τ into its parent peak in the persistence hierarchy.

3.4 ToMATo

Topological Mode Analysis Tool (ToMATo) is the clustering algorithm using TDA. ToMATo relies on three parameters; the neighbourhood graph G, the density estimator $\hat{f}$ and the merging parameter τ. The popular neighbourhood graphs are δ rips graph and k nearest neighbour (KNN) graph. The algorithm begins by taking G and a non-negative τ. Let $\hat{f}(i)$ be the estimated density value of each vertex i of G at that point. The first step is mode seeking where i with highest $\hat{f}$ compared to its neighbours are selected as the peak of $\hat{f}$. The next step is merging the peaks with prominence less than τ. The output is the collection of merged clusters. The article by Chazal et al [22] describes the theory and applications of ToMATo in detail.

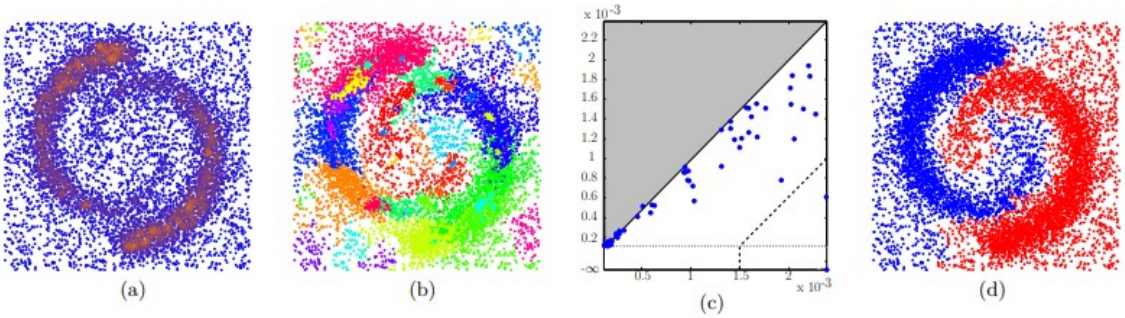

Figure 3.2: ToMATo approch : (a) estimation of the underlying density function f at the data points; (b) result mode seeking step; (c) approximate PD; (d) final result obtained after merging the clusters of non-prominent peaks. Source : Chazal, et al 2013 [22]

Chapter 4

Clustering Mixed Type Data Set

4.1 Introduction

The clustering is defined as grouping objects into different clusters based on the similarity between the data points. Most of the clustering algorithms studied are based on euclidean distance which are primarily used on data sets with all real valued attributes. The classical clustering algorithms like partitioning clustering works by taking the average of distance between the data points which can't be applied for categorical attributes. Typically, most real data sets involve both real valued and categorical attributes which can't be clustered using classical clustering algorithms. Clustering mixed data is still a challenge. To balance the contribution from continuous and categorical variables is one of the main challenge. Most of the clustering algorithms suffer from information loss due to discretization, parametric assumptions or the choice of weights of continuous versus categorical variables.

There are several clustering algorithms available for mixed data type. Most of them are modifications of the clustering algorithms used for non mixed type data. Gower's distance metric can be used as similarity metric for hierarchical clustering, but for large data sets it becomes meaningless.

4.2 Different Approaches

The techniques attempted to cluster mixed data type tries to use the same method for single data type set. One method is by changing the categorical variables to dummy codes and then applying the clustering techniques suitable for numerical data types. For a multi valued categorical variable the dummy codes will be 0 & c indicating absence and presence of the particular value in the variable for each of the values. A large c will emphasize the categorical variables more whereas small c will emphasize the continuous variable more. Taking c = 1 is not a very good approach in the case of mixed type data clustering.

Another approach is by using a distance metric applicable for mixed data like Gower's distance. And then clustering methods which is based on distance is used to cluster. From 2.1 we know each variable need to be assigned a weight, the choice of weights will again like dummy coding, cause either over or under emphasizing the variables. Hence the cluster output we get won't be accurate.

There are many other approaches for clustering mixed data type using single data type algorithms, but they are not relevant here. Moving on, stabler, effective and recent mixed data type clustering techniques are discussed.

4.3 W K Prototype Algorithm

This algorithm proposed by Huang (1998) is an extension of partitioning algorithms used for either numerical or categorical data types. This method integrates k modes and k means algorithms for mixed data sets. A new similarity measure is defined and the cluster centers are means for numerical variables and modes for categorical variables. The distance is defined as $d^e + \gamma d^h$ [14], where d^e and d^h represents the euclidean distance of the numeric attributes and the hamming distance of the categorical attributes respectively. The γ is a weight to balance both the attributes. The weights are inversely proportional to sum of within cluster distances [15]. The k prototype algorithm process same as k means algorithm except k mode is used for categorical attributes. The hamming distance is defined for bi-valued categorical variables, hence won't be accurate in case of multi valued categorical variables.

4.4 KAMILA

The above mentioned approaches and algorithms fails in handling the contribution of continuous and categorical variables, ths problems are more or less adressed by a recent technique KAy-means for MIxed LArge data sets (KAMILA) proposed by Foss et. al. (2016). This algorithm combines Gaussian Multinomial mixture model [23] and K- means algorithm. This method works by estimating density from the data and balances the continuous and categorical variable's contribution without assigning weights.

Hunt et al [23] gives insight into the theory and mathematics behind KAMILA in detail.

4.5 Clustering Mixed-Type Data Using Persistent Homology

Let $\mathbb{X} = x_1, x_2, x_3, ..., x_D$ interdependent random variable, f is the probability function and G = (ν,ϵ) is the graph, where ν is the node-set with each element representing a random variable and ϵ is the edge, representing the dependency relation between the variables.

A value assignment to all random variables $x = (x_1, ..., x_D)$ is called a configuration. The potential function $f : x \to R$ assigns each configuration a real value. And this is inversely proportional to the log of the probability distribution.

We are focusing tree-structured graphical model, T = (ν,ϵ). For this, we can factorise the probability function into products as

$$p(x) = \prod_{(i,j)\in\epsilon} \left(\frac{p(x_i), p(x_j)}{p(x_i, x_j)} \right) \prod_{k\in\nu} p(x_k)$$

and compute the mutual information

$$MI_{ij} = \int_{x_i,x_j} p(x_i, x_j) log(\frac{p(x_i, x_j)}{p(x_i)p(x_j)}) dx_i dx_j$$

. Let I_D and I_C be index sets of random variables of discrete and continuous domain

respectively. Within the discrete domain, we use hamming distance $dist_H(x, x^{'})$ and L2 distance $dist_{L2}(x, x^{'})$ within the continuous domain.

$$N_\delta^d(x) = \{x^{'} \mid dist_d(x, x^{'}) \leq \delta \Lambda dist_c(x, x^{'}) = 0\}$$

is the discrete neighbourhood of x with radius $\delta > 0$, hamming distance not greater than δ and zero euclidean distance. Similarly, the continuous neighbourhood of x with radius $\epsilon > 0$ is

$$N_c(x) = \{x^{'} \mid dist_d(x, x^{'}) = 0 \Lambda dist_c(x, x^{'}) \geq \epsilon\}$$

Now we define a mode, which is the local maxima in both neighbourhoods. A point $x \in \chi$ is a mode if and only if there exist positive numbers ϵ and δ such that
$p(x) \geq p(x^{'})$ for any $x^{'} \in N_c(x)$
$p(x) \geq p(x^{'})$ for any $x^{'} \in N_d(x)$

We estimate all pairwise mutual information and then compute tree (ν, ϵ). Next, we use the Mode seeking algorithm for finding modes.

Mode Seeking Algorithm

The section 2.7 describes mode clustering in detail. It is an iterative algorithm, starts with an initial data point x, we have a kernel function (2.1) which estimates the probability in the neighbourhood until convergence. The final position is the mode of convergence. At each step, we first consider discrete variables in the discrete neighbourhood until no better elements exist. Then we update the continuous variables using gradient descent until the gradient of f at continuous dimensions becomes zero.

Merging clusters using topological persistence

The modes provide the clusters. We find the relative height, that is the distance between the height of the peak(mode) and level at which basin of attraction meets another higher mode. $\{x^t = x \in \mathbb{X} \mid f(x) \geq t\}$ is the super level set, the probability density is greater or equal to t. Each mode assigns to the birth of the new connected component in the super level set and when a component created by higher mode meets this component, it merges to form new

connected component. The density value of creating a mode is called birth time and the density value of the point at the saddle (where two components meet) is called death time. The difference between birth and death times is called persistence. Persistence measures the saliency of modes.

Chapter 5

Comparison of Air Pollution Levels

5.1 Introduction

According to IQAir's report, 21 out of the 30 most polluted cities in the world are in India. The significant causes of air pollution are emissions from the factories and exhaust from the vehicle. Due to COVID-19 pandemic, strict nationwide restrictions were placed initially by the government, which lead to a reduction in public movement and vehicular traffic. Since work-from-home became the new normal, offices no longer had to function fully. A nationwide lockdown was imposed from March 24, 2020, till May 1. All of this had a severe impact on air quality during the first few weeks when the lockdown was enforced strictly.

Different clustering techniques studied are used to compare the air pollution level in different cities of India before and during the lockdown. In order to compare 7 major air pollutants, PM_{10}, $PM_{2.5}$, O_3, SO_2, NO_2, NO and CO are considered.

5.2 Objective

The study is an effort to use different clustering algorithms for comparison. The objective is to compare and analyse the changes in air quality before and during COVID-19 lockdown in India over eight different cities.

5.3 Dataset

For comparing the air quality levels, seven most frequently used pollutants for analysing the air quality is taken. The contaminants are Ozone, NO, NO_2, SO_2, CO, PM_{10} and $PM_{2.5}$. The data has been collected from Central Pollution Control Board (CPCB) website. The daily level (averaged over 24 hr period time) of each pollutant in mg/m^3 for 70 days before and during COVID-19 lockdown are used. The pollutant level data from January 1, 2020 to March 23, 2020 (before lockdown) and from March 24, 2020 to May 1, 2020 (after lockdown) are collected and analysed. The average of the each pollutant level 70 days before and after COVID-19 lockdown is used to do different clustering methods. The eight different stations are,

1. Mumbai (CST)
2. Nashik
3. Kochi (Vytilla)
4. Delhi (JNU stop)
5. Guwahati
6. Chandigarh
7. Patiala
8. Bangalore (SB road)

5.4 Method

Various clustering algorithms namely DBSCAN, K means, PAM, hierarchical clustering and ToMATo discussed in this thesis have been applied to compare the significance of different air pollutants in different cities. These algorithms group the cities into different classes/ clusters, and we analyse the nature of different clusters.

Additionally, clustering is applied to Vytilla station over 78 days before and during lockdown to observe the changes in air quality due to lockdown. Rand Index is a similarity

Result						
	Prelockdown			Postlockdown		
Method	Cluster A	Cluster B	Noise	Cluster A	Cluster B	Noise
DBSCAN	2,6,7,8	1,4,5	3	2,5,6,7,8	1,3	4
AGNES(S)	2,3,6,7,8	1,4,5		2,4,5,6,7,8	1,3	
AGNES(A)	2,3,6,7,8	1,4,5		2,4,5,6,7,8	1,3	
AGNES(C)	2,3,6,7,8	1,4,5		2,4,5,6,7,8	1,3	
DIANA	2,3,6,7,8	1,4,5		2,4,5,6,7,8	1,3	
K MEANS	2,3,6,7,8	1,4,5		2,4,5,6,7,8	1,3	
PAM	2,3,6,7,8	1,4,5		2,4,5,6,7,8	1,3	
ToMATo	2,3,6,7,8	1,4,5		2,4,5,6,7,8	1,3	

Table 5.1: The clustering output of different stations.

	DBSCAN	K Means	Hierarchical	ToMATo
DBSCAN	1	0.9058682	1	0.4391144
K Means	0.9058682	1	0.896138	0.01304527
Hierarchical	1	0.896138	1	0.03702071
ToMATo	0.4391144	0.01304527	0.03702071	1

Table 5.2: Rand Index values between different clustering method results for Vytilla station.

measure between two different types of clusters of the data set. The Rand index ranges between 0 and 1 where 0 indicates that two data clusters do not agree on any pair of points and 1 indicating that the data clusters are exactly the same. In this application, this is used to compare the results of clustering on Vytilla data set.

5.5 Results and Discussion

Clustering the data of eight different stations yielded a solution with 2 clusters. The output clusters of pre lockdown and post lockdown gave the same results for different techniques.

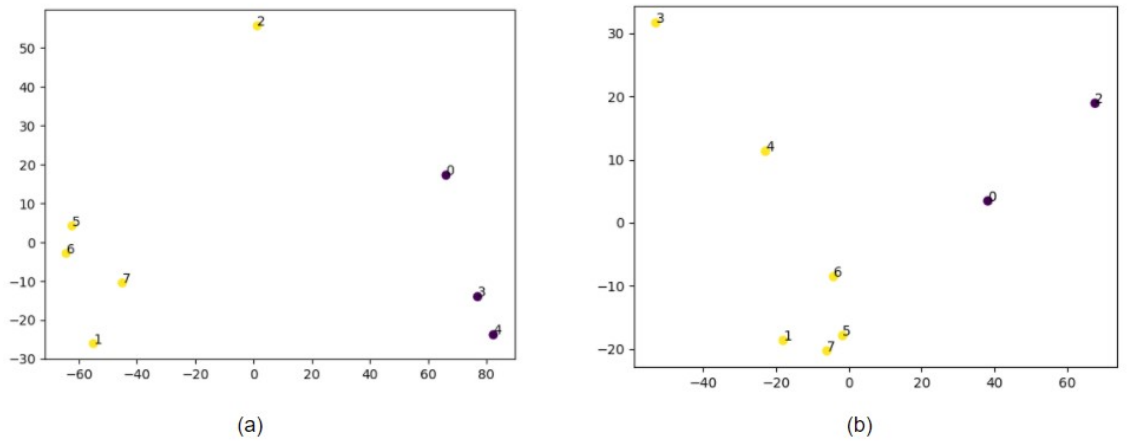

Figure 5.1: Plot of ToMATo clustering :(a) Pre-COVID19 and (b) Post-COVID19 Lockdown

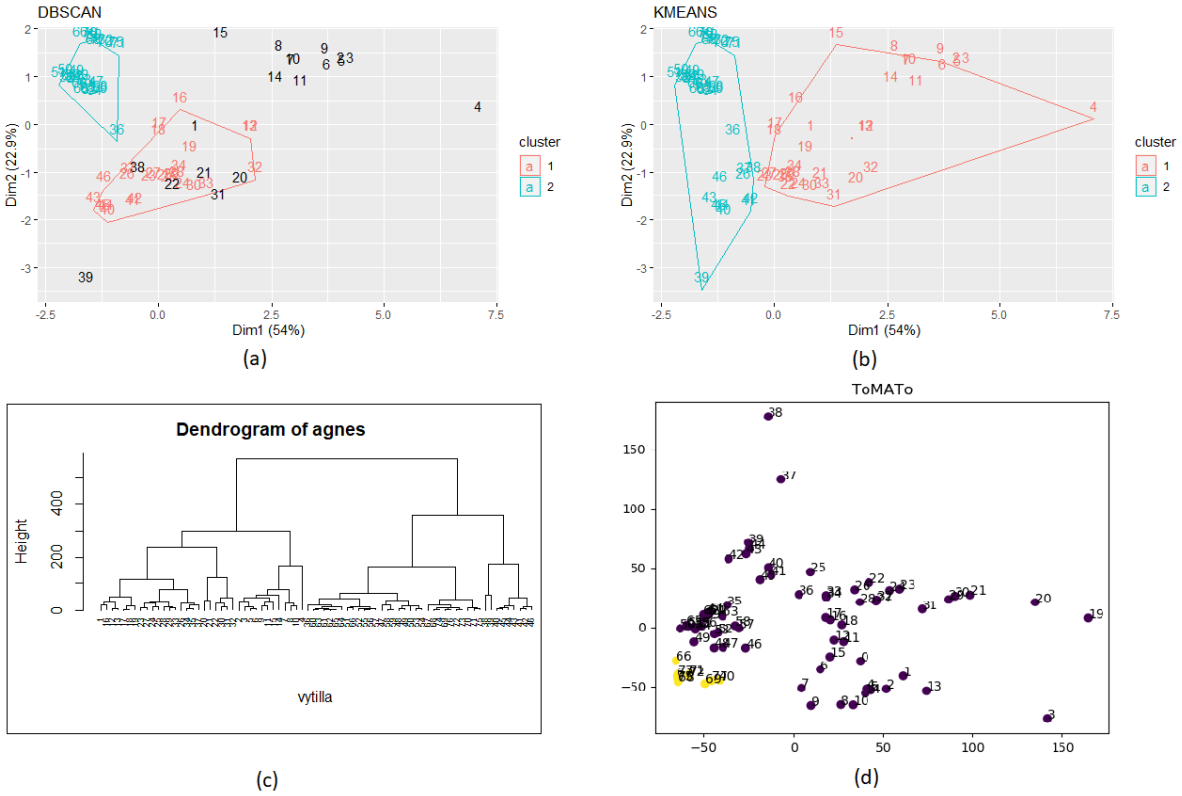

Figure 5.2: The clustering outputs of Vytilla city. (a) DBSCAN (b) K-Means (c) Hierarchical clustering and (d) ToMATo

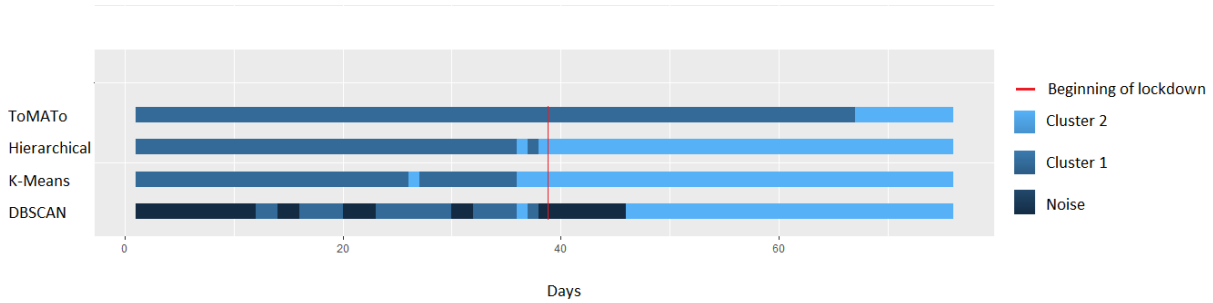

Figure 5.3: Comparison of ToMATo, hierarchical, K-Means and DBSCAN clustering results. The colour corresponding to each day represents the cluster to which it is assigned to.

Table 5.1 summarizes the clustering result.

- Healthier and non healthier clusters were decided by comparing the cluster centers/medoids of both clusters with CPCB prescribed standards.
- Clusters obtained post-lockdown are "healthier" than pre-lockdown clusters.
- Although lockdown made station 3 healthier, impact of lockdown wasn't strong when compared to other stations which classified it into relatively unhealthy.

Figure 5.2 shows the different clustering plots of Vytilla and Table 5.2 gives the Rand Index. Figure 5.3 shows us that the observation before and during lockdown are grouped into different clusters. The 5.3 and Rand Index gives us a similar output. The Rand Index is high for DBSCAN, kmeans and hierarchical clustering, which means the clusters resulting from these three methods are similar. There is a significant impact of lockdown on Vytilla's air quality.

Chapter 6

The Study of Online Shopper's Behaviour to Seek the Pattern

6.1 Introduction

Online shopping has grown exponentially taking a remarkable share of retail market in last two decades. Strategically marketing offers and promotions are necessary to attract more customers and to encourage sales in online shopping platforms. Even though online shopping sites observe high traffic number, only a small fraction of users actually complete transaction and contribute to revenue generation. In order to make experience of purchasing a product enjoyable for customer and to increase successful transaction rate, marketers can carefully target advertisements to a more relevant crowd. We use data set downloaded from https://archive.ics.uci.edu/ml/datasets/Online+Shoppers+Purchasing+Intention+Dataset. The source of this data set is Sakar et al. Clustering algorithms are applied to understand intent of each user from their analytics data. Gower distance metric is used to calculate the dissimilarity between the observations, as the data set consists of numerical and categorical variables.

6.2 Data set

The data set consists of mixed covariates with 12330 observations of 18 attributes. The data set was formed so that each session would belong to a different user in a 1-year period to avoid any tendency to a specific campaign, special day, user profile, or period [9]. The data consists of 10 numerical and 8 categorical variables. The features are Administration, Administration duration, Information, Information duration, Product related duration, Bounce rate, Exit rate, Page value, Special day, Operating system, Browser, Region, Traffic type, Visitor type, Weekend and Revenue.

The numerical features Administration, Administration duration, Information, Information duration, Product related duration represents the number of three different pages visitors visited and the time duration they spent in seconds. Bounce rate and exit rate indicates the percentage of visitors entered and then left, and visitors exiting the site after visiting number of pages respectively. Page value indicates the average value of pages visited before completing the transaction. Special day indicates whether the visitors visited site on days near to special days.

The categorical attributes operating system and browser indicates the operating system of the device and the web browser respectively, from which visitor visits the site. There are 8 and 13 different operating system and browsers respectively. The Region indicates the location of the visitor. Traffic type indicates how the visitor ended up at the site. New visitor, existing visitor and other visitor are the three different visitor types. Weekend is Boolean value indicating whether the visitor visited on weekend or not. Revenue is a class label representing whether the visitor ended up buying or not.

6.3 Objective

The aim is to drop the class label Revenue and use different clustering methods on the remaining attributes to check whether the group (classes) match those given by it.

DBSCAN	K Means	W K Prototype	ToMATo	KAMILA
0.7331659	0.7068107	0.8483791	0.0011	0.8920143

Table 6.1: Rand Index values compared with Revenue class label

6.4 Methods.

Since the data set is large and complex, it is difficult to get informative output by using hierarchical methods. As the data set consists mixed variables, using Euclidean distance metric to find the dissimilarity is not possible. Also partitioning methods and DBSCAN can't be used directly.

- So one approach would be to use Gower's distance to calculate the dissimilarity and then perform clustering using k means and dbscan. Gower's distance (or similarity) first computes distances between pairs of variables of data set and then combines those distances to a single value per record-pair [11]. Gower's similarity measure is scaled to fall between 0 and 1, 0 corresponds to identical points and 1 corresponds to maximally dissimilar points. To assign weights for Gower, feature importance is done using random forest and the variables are ranked according to their importance. The weights are assigned according to the rank.
- ToMATo
- W K prototype
- KAMILA

6.5 Results and Discussion

The clusters obatined were compared with the Revenue class label. From table 6.1, the KAMILA algorithm gave more matching results with Revenue label compared with other techniques applied using Gower metric. Two clusters were obtained, "I" and "II". Cluster I contains more than 85 percent of the total data points. From the Figure 6.1, the customers who spent more time on administrative pages are more likely to make purchase. Results

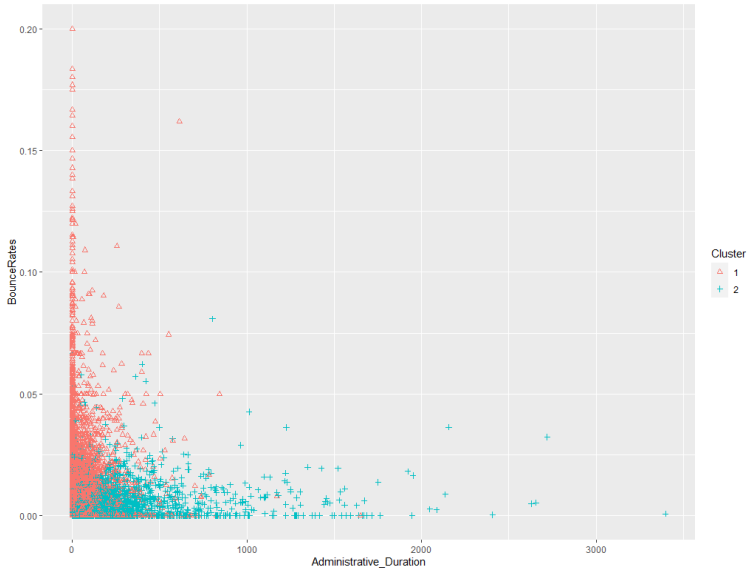

Figure 6.1: Clustering plot : administrative duration against bounce rate

shows that the cluster "II" with very few data points is the cluster which ends up purchasing or contributing to the revenue. The data points of cluster "II" were more engaging in consumer culture, hence relevent e-commerce websites can ad target them instead of data points in cluster "I".

Chapter 7

Conclusion

In this thesis, we learned the theory behind various clustering algorithms and their limitations are seen. We also explored the mathematics behind the most popular Topological Data Analysis tool persistent homology and saw how it could be integrated with unsupervised learning like clustering. The clustering results cannot be generalised; it depends upon the data set and our interpretations. The classical algorithms like hierarchical, k-means and pam group the data set into different clusters even if there is no group structure, while DBSCAN identifies the noise points and finds clusters accordingly.

The studied methods are applied to two different types of data sets, numerical and mixed type. The clustering outputs of Air Quality data set (numerical) were similar. To cluster the Online shoppers' intention (mixed) data set, we attempted two ways, one by using Gower metric and applying the techniques designed for numerical data sets and another way by using two new recently published algorithms for mixed-type data set. The latter method gave more better results than the former method.

Bibliography

[1] Wasserman, L. *Topological data analysis. Annual review of statistics and its application,* 2018.

[2] Gan, G., Ma, C., & Wu, J. *Data clustering : Theory, Algorithms, and Applications,* 2007.

[3] Izenman, A.J. *Modern multivariate statistical techniques, Regression, classification and manifold Learning,* 2006.

[4] Chen, Y., Genovese, C.R., & Wasserman, L. *A comprehensive approach to mode clustering,* 2015.

[5] Kim, J., Chen, Y., Balakrishnan, S., Rinaldo, A., & Wasserman, L. *Statistical inference for cluster trees,* 2017.

[6] Hartigan, J. *Clustering Algorithms.* John Wiley & Sons, 1975.

[7] Fukunaga, K., Hostetler, L.D. *The estimation of the gradient of a density function, with applications in pattern recognition,* 1975.

[8] Veenman, C.J., Reinders, M.J.T., & Backer, E. . *A Cellular Coevolutionary Algorithm for Image Segmentation,* 2003.

[9] Sakar C.O, Polat S.O, Katircioglu M, Kastro Y. *Real-time prediction of online shoppers' purchasing intention using multilayer perceptron and LSTM recurrent neural networks.* Neural Comput. Appl. 2019.

[10] Baati, K.& Mohsil, M. *Real-Time Prediction of Online Shoppers' Purchasing Intention Using Random Forest,* Artificial Intelligence Applications and Innovations. 2020.

[11] Gower. J.C. *A General Coefficient of Similarity and Some of Its Properties,* 1971.

[12] Ester, M., Kriegel, H.P., Sander, J., Xu, X. *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise,* 1996.

[13] Hahsler, M., Piekenbrock, M. Doran, D. *dbscan: Fast Density-Based Clustering with R,* vol.-91, no.-1, pages-1-30, 2019.

[14] Z. Huang, *Extensions to the k-means algorithm for clustering large data sets with categorical values,* Data Min. Knowl.Discov., vol. 2, no. 3, pp.283–304, 1998.

[15] Ahmad, A. Khan, S *Survey of State-of-the-Art Mixed Data Clustering Algorithms,* 2019.

[16] Thibshirani, R., Walther, G. Hastle, T., *Estimating the number of clusters in a data set via the gap statistic,* 2001.

[17] Tran, T. N., Wehrens, R., Buydens, L. M. C., *KNN-kernel density-based clustering for high-dimensional multivariate data,* Computational Statistics and Data Analysis, 51, pp. 513-525, 2006.

[18] Silverman, B. W., *Density estimation for statistics and data analysis,* 1986.

[19] Cadre, B., *Kernel estimation of density level sets,* 2006.

[20] Cheng, Y., *Mean Shift, Mode Seeking and Clustering,* 1995.

[21] Otter, N., Porter, M., A., Tillman, U., Grindrod, P. Harrington, H., A., *A Roadmap For The Computation Of Persistent Homology,* 2017.

[22] Chazal, F., Oudot, S., Skraba, P. Guibas, L., J., *Persistence Based Clustering in Riemannian Manifolds,* 2013.

[23] Hunt, L., Jorgensen, M., *Clustering mixed data*, WIREs Data Mining and Knowledge Discovery, 1, 352–361, 2011.

[24] Wasserman, L., *All of Nonparametric Statistics,* 2006.

[25] Comaniciu, D. Meer, P., *Mean Shift Analysis and Applications,* 1999.