Effective use of clustering analysis to stabilize agricultural crops in India

Dissertation submitted in part fulfilment of the requirements

for the degree of

[*MSc Data Analytics*]

at Dublin Business School

ENUGU MADAN MOHAN REDDY

10516674

DECLARATION

I, **ENUGU MADAN MOHAN REDDY**, declare that this research. thesis entitled "Effective use of clustering analysis to stabilize agricultural crops in India: With Specific Reference to Dublin Business School" was carried out by me for the degree of MSc Data Analytics under the guidance and supervision of Mina Ghahremanzamaneh, Dublin Business School, Dublin. The interpretations put forth are based on my reading and understanding of the original texts and they are not published anywhere in the form of books, monographs or articles. In addition, I have referenced correctly all literature and sources used in this work and this work is fully compliant with the Dublin Business School's academic honesty policy.

Signed: ENUGU MADAN MOHAN REDDY

Date: 20-05-2020

ACKNOWLEDGEMENTS

I wish to sincerely thank all those who have contributed in one way or another to this study. Words can only inadequately express my deep gratitude to my guide, Mina Ghahremanzamaneh, for her meticulous care, kindness and generosity. Her fruitful comments and insightful suggestions have been a crucial formative influence on the present study. She has supported me in every possible way since the beginning of my research. Her critical and careful reading of my writing has saved me from a lot of errors. Without her guidance and encouragement, my research would have never come out in the present form. Furthermore, it has been a memorable and enjoyable experience for me to work with her.

I wish to express my sincere gratitude to Dr. Shahram Azizi Sazi who built the foundations of my work through his “Research Methods” modules, for his guidance, encouragement, and gracious support throughout the course of my work and for his expertise in the field that motivated me to work in this area. Grateful acknowledgments are also due to Marta Silva Postgraduate Business Programme Coordinator who was always helpful and super-fast to clarify and resolve any query.

Special thanks to my mother who motivated me to pursue Master’s degree and who always supported me through prays, financial and moral support, especially during my illness and difficulties. Finally, I sincerely acknowledge the courtesy of the authorities of libraries: Dublin Business School for their cooperation needed by permitting me access data and relevant materials while carrying out the present research.

ABSTRACT

With the growth of developing database systems and computer implementations, the volume of data that the person collects are rapidly increasing. Faced with the incredibly large volume of data, we slowly walk into an awkward circumstance of "good info, bad intelligence." To address this issue, data mining (Data Mining) rises up. we research the clustering analysis means and methods that manage data partitioning or grouping, which is a significant area of data mining. Secondly, focused on the understanding of the theoretical foundations of the clustering analysis, firstly, analysing the main algorithms of partitioning methods in detail. Second, comparing performance of multiple clustering algorithms from scalability, cluster structure, sensitivity to "noise," and responsiveness to data input sequence, large dimension, and efficiency of algorithms. The data was taken from the Open Government Data Platform India. The crop includes 124 types, a few required related crops to Cashew nut, Coconut, Coffee, Paddy, Tobacco, Wheat, and several more. The dataset contains 2,46,091 records with seven variables corresponding to state, district, year, season, crop, area, production and state-wise crop production sample retrieved. The sample data used for clustering analysis are states with 2 parameters production and area. Results of clusters that utilize 3 clusters: low production cluster, normal production cluster, high production cluster.

Table of Content

Table of Figures

List of Tables

CHAPTER 1 – INTRODUCTION

1.1 Introduction

Chapter 1 deals with the conceptual background for the topic. The segment further discusses what the project would include, the description of the issue and the goals of the analysis, why the work is required and the overall structure of this thesis.

1.1.1 Background

Today, India ranks second in the world in farm production. Agriculture is demographically the main economic field and plays a major role in India's overall socio-economic structure. Agriculture is a special sector crop development that relies on a broad variety of environmental and economic factors. Some of the factors that rely on agriculture are soil, climate, cultivation, irrigation, fertilizers, temperature, rainfall, harvesting, pesticide weeds, and other factors. The accurate estimate of crop production and risk allows industries which include agricultural products as raw materials, cattle, milk, animal feed, chemicals, poultry, fertilizers, pesticides, seeds, and paper to make supply chain activities, such as production planning.

There are two aspects that allow farmers and the government to make decisions, namely:

a) It allows farmers to maintain track of the past crop yield record by predicting a decline in risk management.
b) It allows the government to create crop insurance policies and policies for the process of the supply chain.

1.1.2 Introduction to Data Mining

The overall purpose of the data mining approach is to collect information from a data set and render it a simple and comprehensible format for future usage. Data mining also covers elements of database and data administration, pre-processing of results, model and inference considerations, fascinating indicators, difficulty considerations, post-processing of structures identified, visualization and online updating. Data mining is the study level of the "knowledge discovery in databases" process or KDD. thesis.

India is a country based on agriculture. Agriculture is a sector that became the back bone in the development of the Indian economy. India offers several varieties of crops which have different cropping season. The data has been taken from the “The research data was taken from Open Government Data Platform India” considering the attributes: "State", "District", "Year", "Season", "Crop", "Area", "Production”.

1.1.3 Definition of the Term Data Mining and Knowledge Discovery

Simple definition of data mining in marketing is extraction of previously unknown, overstated and adequate information from large data storage and their use for key business decisions to help them is carried out, tactical and strategic marketing strategies are formulated and their performance measured. There are diverse fields in which data mining can be applied effectively, such as organic industry, economics, physics, medicine, genetics. Data mining is commonly applicable in all fields where certain regularities, relations, and laws are to be extracted from large data.

It can be said that 'Data Mining' is in data finding standards. Data Mining technology is closely related to data storage and intertwined with the database management

system. Data mining involves discovering vast amounts of previously unknown data, and then using them in critical business decision-making processes. The key phrase here is 'unknown datum,' meaning that the datum is buried in large quantities of operational data which, if analysed, provide relevant information to decision-makers in the organizations.

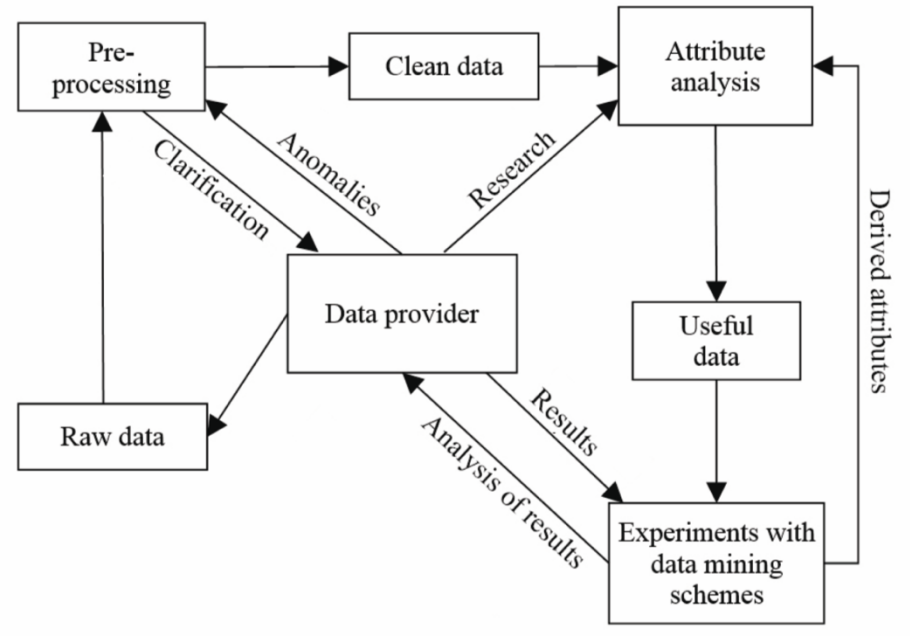


Figure 1.1: Process model for a machine learning (data flow diagram)

Clockwise circulates in the Figure data flow model. Raw data are presented as a single table provided by the data mining algorithms included in r studio – an open source framework that provides a set of data analysis tools and algorithms and predictive modelling. The table is then converted into an attribute / value table that contains header details dependent on the data category of attributes. This involves data cleaning, working with missing values, finding false values, etc., after this data is analysed by data mining algorithms, it is converted into a type that generates a readable, accurate data model. Any or two versions of the clean data are already being analysed through data mining schemes. It is now known which output data classes

are new and reliable enough or interesting enough to justify further research and which are standard knowledge in this field. ICT systems are gradually being introduced in agricultural enterprises to satisfy the needs of agronomists and executives in their day-to-day decision-making activities. Data mining techniques may be very useful for managing human limits, such as subjectivity or fatigue-induced mistakes, and for offering feedback on decision-making processes.

The purpose of data mining is that the discovery of associations, trends and models that provide predictions and decision-making processes for agro-technical initiatives and management or selling decisions. Such models could also be considered as predictive models. Such frameworks should be implemented as decision-making models in organizational information systems, minimizing subjectivity yet as decision making time. In turn, the usage of data technology in agriculture facilitates a comprehensive management of agricultural expertise and a reliable sharing of data between consumers and providers of agricultural services. Wide utilization of information technology (IT) allows the removal of the manual activities of extracting data from charts or filling in an exceedingly detailed questionnaire, extracting data directly from electronic records, transfer to a secure electronic system of agricultural reporting that decreases the price of agricultural products. Returning information by computers can help us make better decisions to stop human error. If there's a large volume of knowledge to be categorized, human decision-making is commonly bad. Data processing is that the method of identifying and processing information with the assistance of computers. this may be also defined because the extraction of previously unknown potentially useful information from an oversized number of (unstructured) data (Milovic, 2011). Due to this strategy, it's possible to predict patterns or customer activity and thus guarantee the firm's business success. It's accomplished by

examining data from various viewpoints and identifying similarities and relationships in apparently unrelated knowledge. If the expertise has been identified and introduced to the customer, measurement methods may be enhanced, processing may be further 'refined,' new data may be collected or further converted, or new data sources may be incorporated with a view to acquiring specific details of corresponding results (Zaïane, 1999). Within the process of data mining, previously hidden phenomena and patterns are discovered on the idea of past knowledge and this evidence is transformed into significant market strategies (boirefillergroup.com, 2010). When evaluating the integrity of data, there are two key problems (Yang and Wu, 2006): the way to build effective algorithms for comparing the contents of two versions of data (before and after evaluation). This task involves the creation of effective algorithms and data structures to see the standard of data in an exceedingly dataset. The way to develop algorithms for evaluating the results of such shifts in data on the statistical development of human trends acquired by general groups of data mining algorithms. Algorithms are built here that calculate the influence of changes in data values on the statistical significance of patterns found, although it should not be feasible to determine a universal measure for all data processing algorithms.

1.1.4 Data Attributes and Quality

Data is how the computer objects and their properties are stored.

- The attribute is the property or attributes of an entity. For proof, this. Hair colour, air humidity, etc.
- The attribute collection identifies an entity. The object is often referred to as a record of an instance or individual.

Different categories of attributes or types of data:

Nominal Attribute:

Nominal Attributes only have adequate attributes to differentiate between one entity and another. Like the Student Roll No., the Person's Age.

Ordinal Attribute:

The ordinal attribute value contains necessary knowledge for the items to be ordered. Including ranks, ages, height, etc.

Binary attributes:

0 and 1. Where 0 is the lack of any features and 1 is the presence of some features.

Interval:

The gap in values is important for the properties of the period. These characteristics shall be measured on units of equivalent scale. Like dates, temperature, etc.

Ratio:

The variations and ratios are important for Ratio. For example, Age, weight, height.

Data Quality: How are we pre-processing the data?

Some features serve as a determination factor for data consistency, such as incompleteness and inaccurate records, which are typical features of a large database in the real world. Factors considered for the estimation of data consistency are:

Accuracy:

There are several potential explanations for inaccurate or wrong results. i .e. I say. Incorrect values of property that may be human or machine mistakes.

Completeness:

missing records that exist for certain purposes, features of interest such as consumer details for transactions and purchase records may not always be accessible.

Consistency:

inaccurate data can often arise from differences in naming convention or data codes or from incompatible input field type. Duplicate tuples do need to clean up the data.

Timeliness:

This often affects the accuracy of the results. At the end of the month, most sales representatives are unable to release their sales records on time. There are often a variety of changes and modifications that will be rendered at the end of the month. Data contained in the database is incomplete for a span after each one month.

Believability:

it shows how often users trust the data.

Interpretability:

this represents how convenient it is for users to comprehend the details.

1.1.5 Data Pre-processing in Data Mining

If several basic principles and procedures for data mining have been checked, the next move is to query the data to be used. Unfortunately, real-world systems are heavily affected by harmful influences such as noise, Missing values, unreliable and excessive data and huge sizes in both dimensions, examples and features.

Low-quality data would then contribute to low-quality data mining results. In this segment, we will define the general categorization in which a group of pre-processing techniques may be divided.

For this reason, a variety of subsections will be provided according to the form and range of techniques that relate to each group.

1.1.5.1 Data Preparation

We refer to data preparation as a collection of techniques that properly prepare data to serve as input for a particular Data Mining algorithm. It is important to note that we prefer data planning notation to the arrangement of data pre-processing sections, which is a complicated nomenclature used in previous texts as a whole range of processes that execute data pre-processing tasks.

This is not wrong and we value this nomenclature, but we tend to differentiate specifically between data planning and data reduction, considering the significance that the latter collection of approaches has gained in recent years and some of the simple differentiations that can be made from this understanding.

The preparation of data is typically a necessary phase. This transforms previously worthless data into new data that suits the data mining method. First of all, if data is not prepared, the Data Mining algorithm increasing not be able to work, or it may disclose errors during its runtime. The algorithm should function in the best of situations, but the findings given do not make sense or will not be accepted as accurate knowledge. So, what are the essential issues that need to be addressed in the preparation of data? Here, we have a collection of questions followed by the

appropriate answers to every form of method that belongs to the data preparation family of techniques:

- How am I supposed to clean up the data? — Data Cleaning
- How can I have reliable data? — Data Transformation.
- How do I add and change the data? — Data Integration.
- How can I unify the data and scale it? — Data Normalization.
- How do I cope with missing data? — Missing Data Imputation
- How can I track and regulate noise? — Noise Identification.

1.1.5.2 Data Cleansing

Data cleansing requires operations that fix bad data, filter any incorrect data out of the data collection, and eliminate unwanted data information. It is a general term that incorporates or overlaps with certain well-known methods in data preparation. Some data-cleaning activities include the identification of discrepancies and dirty Data. The latter activities are more linked to the interpretation of the initial data and usually involve a human examination. The data may have a number of irrelevant and incomplete elements. Data cleaning is performed to tackle this portion. It includes the treatment of missing data, noise data, etc.

(a). Missing Data:

This condition happens when any data are missing from the data. Some of them are:

Ignore the tuples: This method is only acceptable when the dataset we have is very big and many values are lacking inside the tuple.

Missing Values: There are a number of ways to do this task. You may opt to manually fill in the missed values by the mean factor or by the most probable value.

(b). Noisy Data: Noisy data is an irrelevant data that cannot be processed by computers. It may be produced due to defective data processing, data entry errors, etc. It can be performed in the following way:

Binning Method: This system operates on sorted data in order to smooth it out. The entire data is separated into pieces of similar scale, and then various approaches are used to accomplish the task. Each segmented entity is treated separately. You should substitute all data in a segment with its mean or boundary values to complete the task.

Regression: Here data may be rendered smooth by applying it to a regression function. The regression used can be linear (with one independent variable) or several (with multiple independent variables).

Clustering: This method collects related data in a cluster. Outliers can be undetected or fall outside clusters.

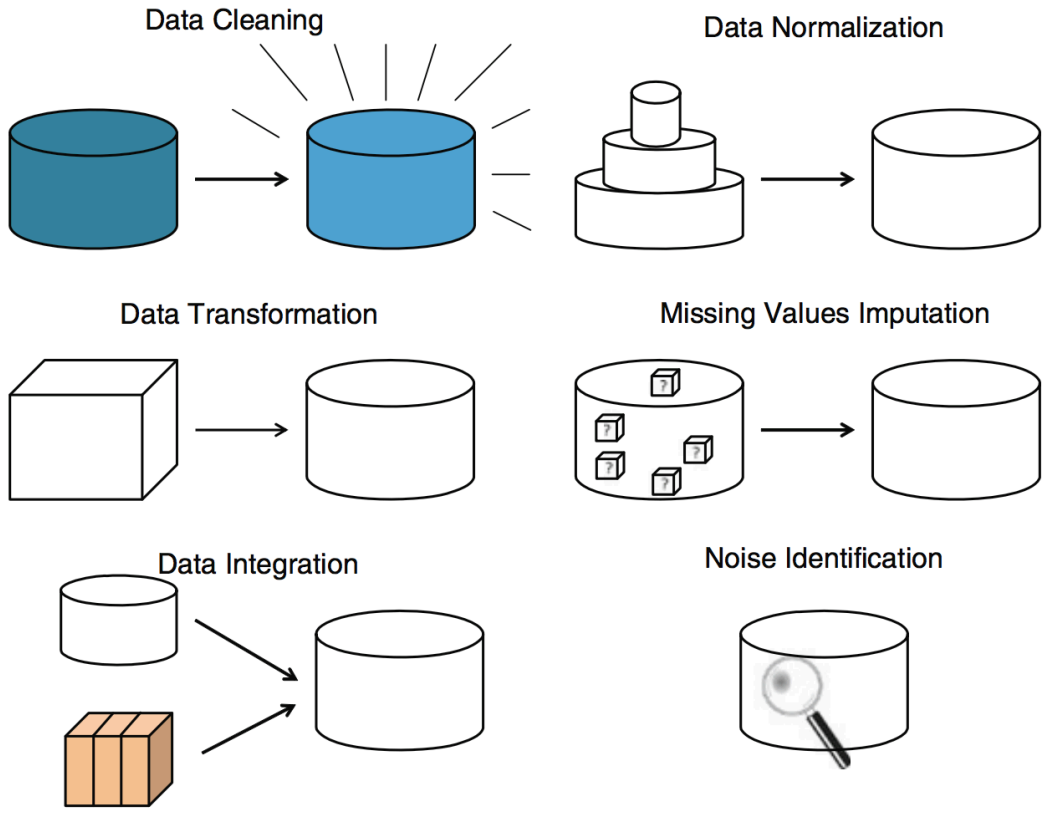


Figure 1.2: Data Mining Processes

1.1.5.3 Data transformation

In this pre-processing stage, the data is transformed or combined in such a way that the effects of the mining process can be implemented or rendered more effective. Subtasks within the processing of data include the smoothing, planning, aggregating or summarizing of data, normalization, discretion and generalization of data. Most of them may be divided as individual tasks as the transformation of data, such as data cleaning, is referred to as general data pre-processing class of techniques. The activities that involve human monitoring and are more data-dependent are classical data processing strategies, such as report generation, new attributes that combine

existing ones, and generalization of concepts, particularly in categorical attributes, such as replacing complete dates in the database with year numbers only.

1.1.5.4 Data Integration

This consists in the combining of data from various data sources. This method must be deliberately carried out in order to eliminate redundancies and inconsistencies in the resulting data collection. Typical operations conducted during data integration involve the identification and unification of variables and domains, the study of attribute similarity, the duplication of tuples and the detection of conflicts in data values of different source

1.1.5.5 Data Normalization

The unit of measuring used can have an influence on the data review. Both characteristics should be represented in the same units of measurement, using a common scale or set. Normalizing the data is meant to assign all characteristics equal weight and is particularly useful in statistical learning methods.

1.1.5.6 Missing Data Imputation

It is a data-cleaning method, where the purpose is to fill in variables containing multiple virtual storage with some intuitive data. In certain instances, it is easier to add a fair approximation of the required data value than to leave it blank.

1.1.5.7 Noise Identification

Included as a data-cleaning step and also known as data-transforming smoothing, its key purpose is to identify spontaneous errors or variances in the measured element. Notice that we apply to noise detection instead of noise elimination, which is more relevant to the Instance Selection task of data reduction. If a noisy example has been found, we should implement a correction-based process that might involve some kind of underlying operation.

1.1.5.8 Data reduction

Data reduction consists of a series of techniques which, in one way or another, obtain a reduced representation of the original data. To us, the difference between data processing strategies is those that are required to fit the data as input for the Data Mining mission.

As we discussed earlier, this implies that if the planning of the data is not carried out correctly, the Data Mining algorithms will not be performed or will show the incorrect results after training. In the case of data reduction, the data generated usually maintains the basic nature and quality of the initial data, but the volume of data is reduced. Therefore, at a glance, it can be called an optional move. This claim can, though, be contradictory. While the validity of the data is preserved, it is well recognized that every algorithm has a time complexity that relies on a variety of parameters. In Data Mining, each of these parameters is in some way directly proportional to the scale of the input database. If the size approaches the limit, the limit being very reliant on the form of Data Mining algorithms, the running of the algorithm may be prohibitive, and the task of data reduction is as important as the preparing of the results.

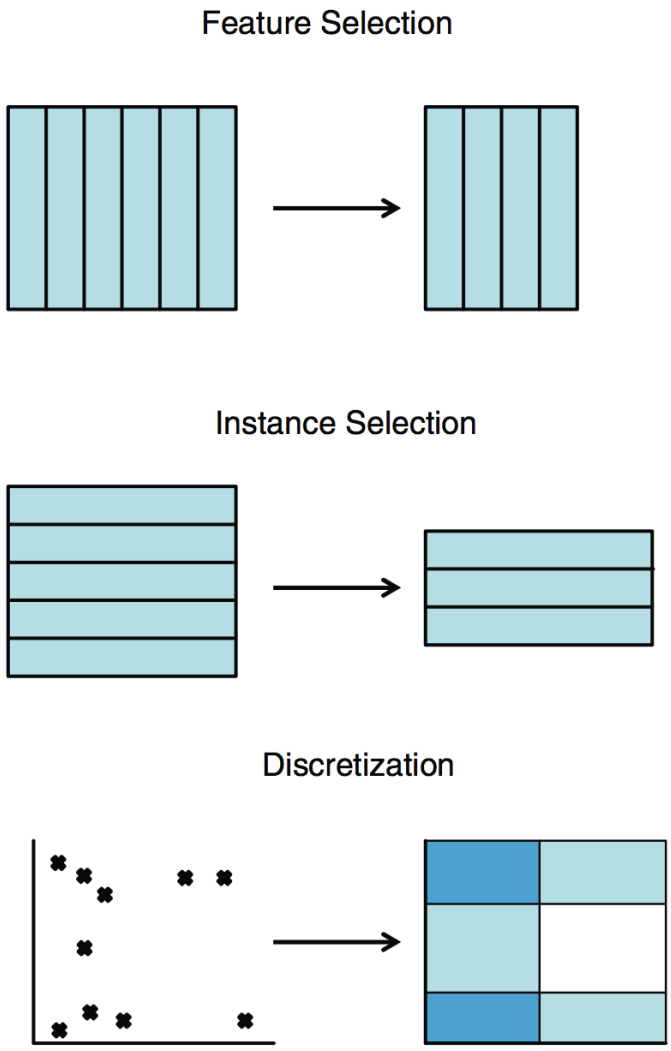


Figure 1.3: Data reduction

- As regard to other considerations, such as the reduction of complexity and the enhancement of the quality of the models generated, the role of data reduction is once again crucial. As stated earlier, what are the fundamental problems that need to be resolved in terms of data reduction? Finally, we have a set of questions linked to the right response for each form of task that is part of the data reduction techniques.
- How would I reduce the dimensionality of data?—Feature Selection (FS)

- How would I remove redundant and/or conflictive examples?—Instance Selection (IS)
- How would I simplify the domain of an attribute?—Discretization
- How would I fill in gaps in data?—Feature Extraction and/or Instance Generation

Feature Selection

It Reduces the data collection by removing unnecessary or redundant features. The object of Feature Selection is to find the minimal collection of attributes, such as the corresponding probability distribution of the data output attributes, as similar as possible to the original distribution obtained for all attributes. This facilitates the understanding of the received sequence, and increases the learning process speed.

Instance Selection

This consists of choosing a subset of the overall usable data to accomplish the original aim of the Data Mining program as if all the data had been used. This is a class of targeted approaches that, utilizing certain rules and/or heuristics, allow a very sophisticated list of the best possible subset of examples from the original results. Random collection of samples is generally referred to as sampling and is used in a very significant number of data mining models for internal validation and for preventing over-fitting.

Discretization

This method converts quantitative data into qualitative data, that is, numerical attributes into abstract or nominal attributes with a finite number of intervals, and obtains a non-overlapping division of a continuous domain. An relation is then formed

between every interval and the discrete numeric value is then established. If the discretization is carried out, the data may be viewed as partial data during any Data Mining process. It should be remembered that discretization is essentially a hybrid pre-processing strategy containing both data planning and data reduction activities. Some reports provide flexibility in the data transformation framework, while other reports call the data reduction process. In reality, discretization can be interpreted as a data reduction tool as it converts data from a huge range of numeric values to a significantly reduced subset of discrete values. Our opinion is to use it more in the data reduction phase, while we do comply with the other pattern. The reasoning behind this is that modern discretization schemes aim to reduce the amount of discrete intervals as much as possible while maintaining the efficiency of the more Data Mining process. In certain words, a straightforward discretization of any form of data is always quite fast, provided that the data is appropriate for a certain algorithm with a clear map between continuous and categorical values. The main challenge, though, is to achieve a successful reduction without losing the accuracy of the results, and most of the attempt made by researchers follows this pattern.

Feature Extraction/Instance Generation

Extends both the Feature Selection and Instance Selection by enabling the adjustment of the internal values that represent each example or attribute. In the extraction feature, apart from the removal activity of attributes, subsets of attributes can be combined or can lead to the development of artificial substitute attributes. As far as generation of instance is concerned, the process is identical in the form of examples. It allows for the development or modification of artificial alternative models that may best reflect decision-making boundaries in supervised learning.

1.1.5.9 Challenges of Data Mining

Data Mining and knowledge discovery are rapidly developing vital technologies for enterprises and researchers in many areas. Data Mining is evolving into a well-established and trusted discipline, and several issues still remaining challenges to be solved.

Most of these challenges are listed below.

Security and Social Challenges:

Decision-making processes are carried out by the exchange of data, requiring substantial security. Private information regarding people and confidential information is gathered for customer profiles, consumer activity pattern awareness. Illegal exposure to information and the confidentiality of information are now an important issue.

User Interface:

knowledge explored using data mining techniques is only valuable if it is insightful and, above all, user-friendly. From a clear understanding of data visualization, mining findings can be eased and help to truly understand their conditions. To achieve effective visualization, a lot of work is performed on broad data sets that view and modify mined information.

Mining based on Abstraction Level:

Data Mining process needs to be collective as it enables users to concentrate on discovering similarities, addressing and refining data mining requests based on the results returned.

Integrating Background Knowledge:

Previous knowledge can be used to convey discovered patterns to direct exploration processes and to convey discovered patterns.

Mining Methodology Challenges:

These issues apply to data mining methods and their limits. Mining approaches that cause problems are:

- ❖ complexity of mining methods
- ❖ diversity of available data
- ❖ domain dimensionality
- ❖ Control noise management and handling in data, etc.

Different methods can be applied differently on the basis of analysis of data. There are several algorithms that need noise-free data. Most data sets involve exceptions, incorrect or missing details that contributes to a complexity of the analysis process and compromises the quality of the results in certain situations.

Complex Data:

Real-world data is heterogeneous and can involve multimedia data including images, audio and video, dynamic data, time data, spatial data, time series, natural language text, etc. It is challenging to manage these various forms of data and to collect the required information. New methods and methodologies are being developed to collect the relevant information.

Complex data types:

The database can involve complex data components, graphical data objects, spatial data, and temporal data. Mining all such forms of data is not feasible to do with one device.

Mining from Various Sources:

Data is obtained from specific network sites. The data base can be of various kinds based on how it is processed as structured , semi-structured or unstructured.

Performance:

The performance of the data mining method depends on the reliability of the algorithms and techniques used. Algorithms and strategies developed are not up to the mark contributing to an effect on the performance of the data mining process.

Performance and Scalability of Algorithms:

The data mining method must be effective and scalable in order to retrieve information from the large number of data in the database.

Development of Mining Algorithms:

Factors such as the immense scale of the database, entire data flow and the complexity of data mining approaches motivate the development of parallel and distributed data mining algorithms.

1.2 Motivation

These days, data are being collected and employed in almost every field. Data that are collected from individuals or organizations are getting used for several things. a number of the information are getting used in observing personalized advertisement

over the web, some are getting used in viewing probability or predictions about important events, some are getting used to achieve profit in business. However, in our country, there's one sector that's not utilizing the advancement of knowledge science and this sector is agriculture. To be more exact, whether or not a researcher wants to try some research adds this field it's very hard to seek out any useful data to figure with. the smallest amount of knowledge that will be found is not in any usable format and requires plenty of processing before it is employed in any quite data science application. Although, the chance of applications of knowledge science during this sector is extremely promising on a date only a few good types of research are worn out in this sector from the angle of our country. Taking this into consideration, it worked as the main motivation behind the research. I desired to figure and contribute to the present field which has not been utilizing the resources that modern computer and data science needs to provide. We wanted to develop a system at an initial state which will predict the agricultural outcome of a rustic with the right computer file. Correct and useful data are hard to seek out during this regard. Most of the information is not yet digitized. Therefore, we required to figure during this sector for creating a decent and useful data set so any researcher who would require to figure for this within the future would have decent data to figure with. This can make the method of contribution during this area rather more effective and would support and drive the authority to gather more relative and useful data regarding this sector. This was my motivation behind putting effort during this sector that has been empty the blessing of recent data science and analysis. Successful implementation of the proposed model will enable us to predict the longer-term outcome of agricultural production which can tell the authority if there is going to

be any kind of scarcity of any crops. If so, then the authority, during this case, the govt can take advanced steps to beat this issue by suitable means.

1.3 Introduction to Clustering

Clustering is a way of defining related user classes in a data set. Entities in each category are much more like elements of the aggregate than those of the alternative classes. Clustering is the practice of isolating the group by focusing information on specific categories with the end aim that the focus of information in related groups is gradually like other information in groups similar to that in different groups. In simple terms, the goal is to separate clusters of comparable qualities and to assign them into categories. Clustering has a vast number of uses spread through different spaces. Its aim is to partition a set of patterns into disjoint clusters, specified patterns within the same cluster are similar, however, patterns belonging to 3 different clusters are dissimilar. The clustering method can be used as a technique for classifying text documents with common content and text themes. During the clustering of documents, a set of documents that have not been classified as their class is clustered according to the characteristics of the terms that each document will perform the clustering method.

The absolute most prevalent uses of grouping are

1. Market segmentation
2. Social network analysis
3. Search result grouping
4. Medical imaging
5. Image segmentation

6. Anomaly detection

Clustering is an unsupervised solution to machine learning; nevertheless, it may be used to improve the precision of machine learning equations by clustering knowledge into comparable classes. The researcher wanted to try and do research about the clustering of crops by considering the years, where the information used came from the Open Government Data Platform India. It is expected that the results of this study can facilitate the government in calculating agricultural products in each region to grasp which year produces crops that are many, medium and few. These results may be an input for the government to strive continuously in stabilizing crops commodities in India.

1.3.1 Formal Definition of Clustering

The clustering problem is described as the problem of classifying 'n' objects in 'C' clusters without any prior knowledge. Let the range of 'n' points be represented by the range of 'S' and the 'C' clusters be represented by V1,V2,...,VC.

$$V_i \neq \emptyset \text{ for } i = 1, 2, \ldots, C,$$
$$V_i \bigcap V_j = \emptyset \text{ for } i = 1, 2, \ldots, C \text{ and } i \neq j$$
$$\text{and } \cup_{i=1}^{C} V_i = S \tag{1.1}$$

1.3.2 Basic Concepts of Clustering

The data clustering problem can be formulated as follows: provided the dataset D that contains n objects x1,x2, ... ,xn (data points, records, instances, patterns, observations, items) and each data point is in d-dimensional space, i.e. each data point has d-dimensions (attributes, features, variables, components). It can be represented as a matrix

$$D = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nd} \end{bmatrix}$$

Figure 1.4: Matrix Form of Clustering

Data clustering is based on the similarity or dissimilarity (distance) of measurements between data points. Such measures also allow the study of clusters important. The high efficiency of clustering is to achieve high intra-cluster similarity and low inter-cluster similarity as seen in Figure. In fact, as the definition of dissimilarity (distance) is used, the following statement becomes: the high quality of clustering is to obtain low intra-cluster dissimilarity and high inter-cluster dissimilarity.

1.3.2.1 Measures of Distance in Data Mining

Distances are typically used to calculate the similarity or dissimilarity between two data points. Clustering consists of combining other items that are identical to each other, and may be used to determine if two items are related or separate in their

properties. In the Data Mining context, the calculation of resemblance is a distance with dimensions representing the features of the object. It implies that if the difference between the two data points is small, there is a strong degree of resemblance between the items and vice versa. Similarity is subjective and relies strongly on the meaning and implementation. For example, the similarities between vegetables may be calculated by their taste, size, colour, etc. Many clustering methods use distance measurements to determine similarities or variations between pairs of items, the most common distance measurements used are:

Euclidean Distance:

Euclidean distance is called a standard measure for geometry problems. It can simply be defined as the ordinary distance between two lines. It is one of the most widely used algorithms in cluster analysis. One of the algorithms that will use this function will be K-mean. Mathematically, it measures the sum of the squared variations between the two objects coordinates.

$$d(\mathbf{p},\mathbf{q}) = d(\mathbf{q},\mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$$
$$= \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

(1.2)

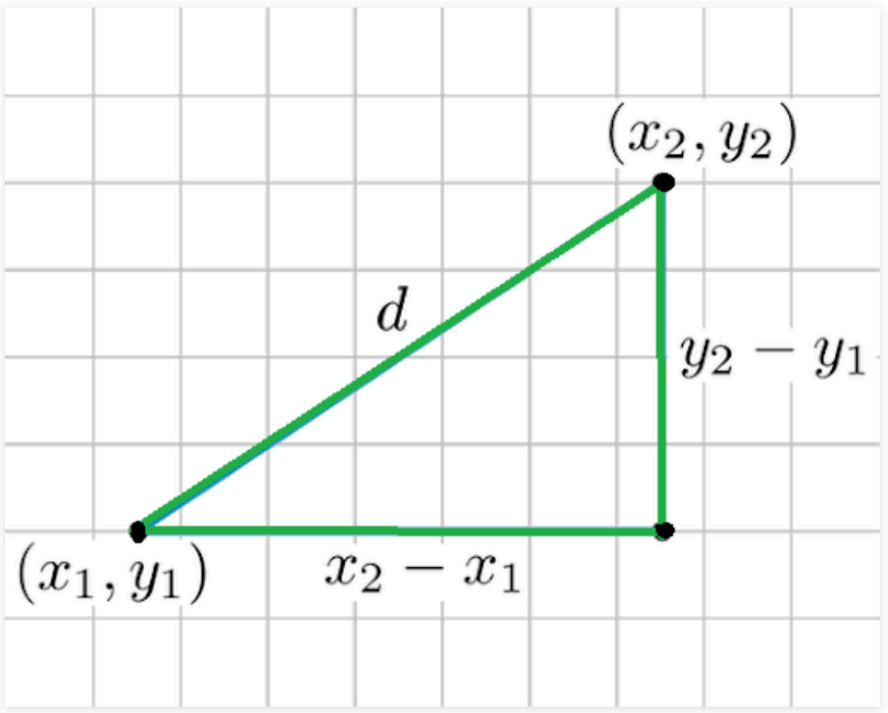


Figure 1.5: Euclidean Distance

Manhattan Distance:

This defines the total difference between the coordinate pairs. Suppose we have two points P and Q to evaluate the distance between these lines, we actually have to measure the perpendicular distance between the lines X-Axis and the points Y-Axis. In a plane with a coordinate of P (x1, y1) and a coordinate of Q (x2, y2). Manhattan distance between P and Q = |x1 – x2| + |y1 – y2|

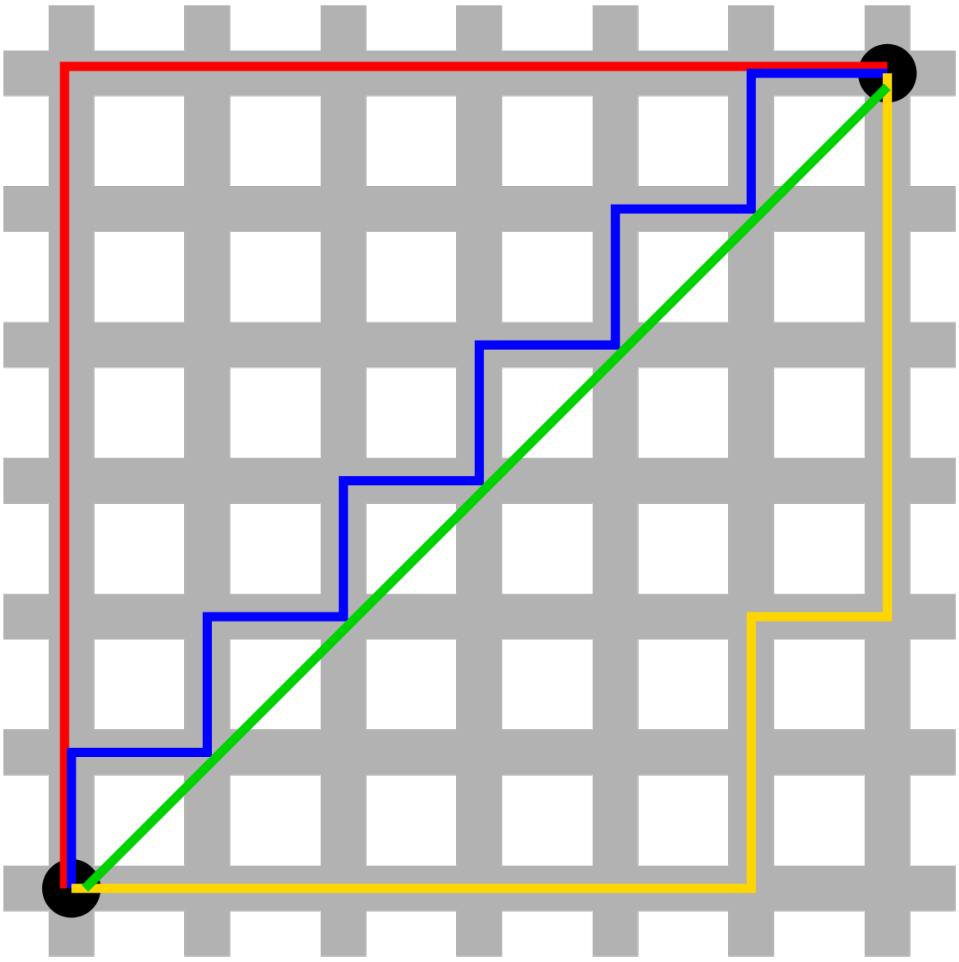


Figure 1.6: Manhattan Distance

Here, the total distance of the Red Line provides the Manhattan distance between the both points.

Jaccard Index:

The Jaccard distance measures the resemblance of the two data objects as the intersection of certain objects separated by the data items union.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \tag{1.3}$$

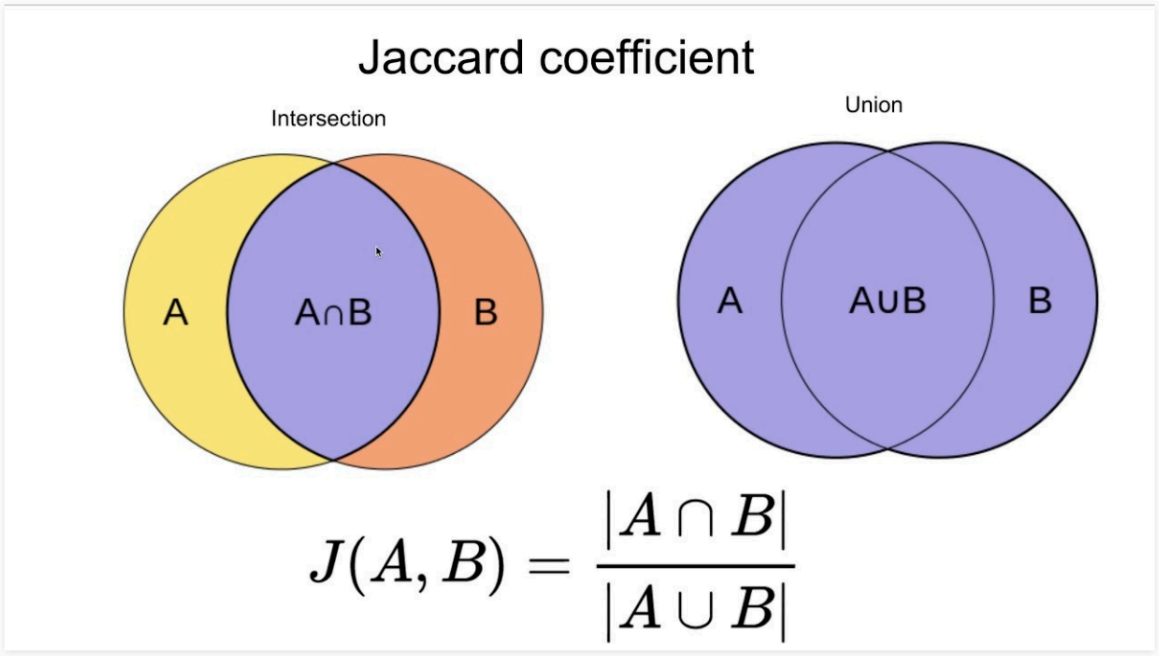


Figure 1.7: Jaccard Index

Minkowski distance:

It is the simplified version of the Euclidean and Manhattan distance measurements.

The point is defined in the N-dimensional space as,

(x1, x2, ..., xN)

Consider two points P1 and P2:

P1: (X1, X2, ..., XN)

P2: (Y1, Y2, ..., YN)

The Minkowski range between P1 and P2 is then given as:

$$\sqrt[p]{(x1-y1)^p+(x2-y2)^p+\ldots+(xN-yN)^p}$$

(1.4)

When p = 2, the distance of Minkowski is the same as the distance of Euclidean.

When p = 1, the distance from Minkowski is the same as the distance from Manhattan.

Cosine Index:

Cosine distance measurement for clustering determines the angle cosine between the two vectors in the following formula.

$$\mathrm{sim}(A,B)=\cos(\theta)=\frac{A\cdot B}{\|A\|B\|}$$

(1.5)

Here (theta) provides the angle between the two vectors and the n-dimensional vectors are A, B.

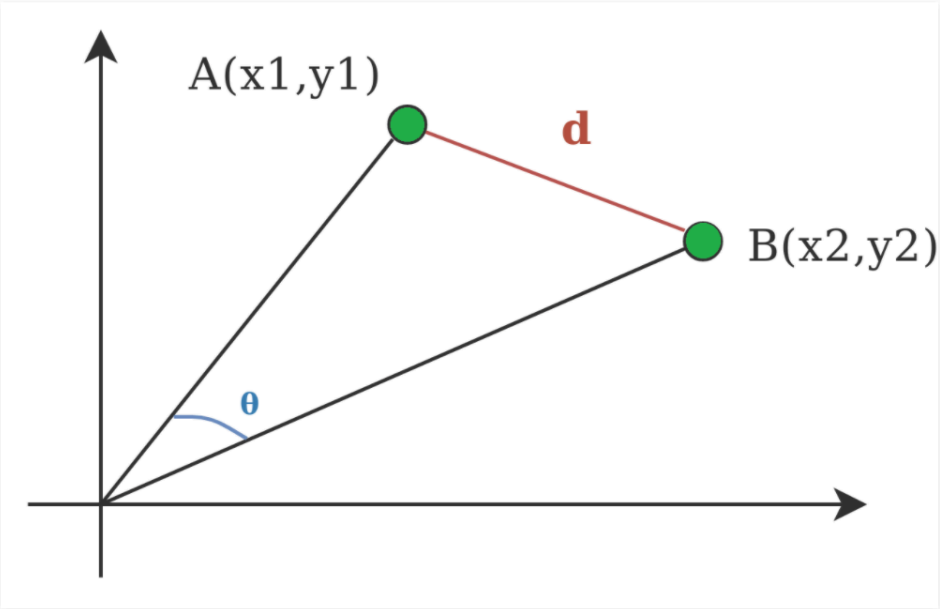


Figure 1.8: Cosine Distance

1.3.3 Importance of Clustering

Data clustering is one of the main tasks of data mining and pattern recognition.

Moreover, it can be used in many applications such as:

1. Data compression
2. Image analysis
3. Bioinformatics
4. Academics
5. Search engines
6. Wireless sensor networks
7. Intrusion detection

8. Business planning

1.4 Application of data mining technology in agriculture

The modern age has brought significant changes and information technologies in numerous areas of human activities have found wide application thus also in agriculture. development and introduction of recent information technologies that enable global networking, give agriculture the label of ‘IT agriculture’.

Agricultural information systems contain massive amounts of data including information about crops, customers, markets. With the utilization of data mining methods, useful patterns of knowledge will be found during this information, which can be used for further research and report evaluation. the important question is how to classify a large amount of data.

Automatic classification is completed based on similarities present within the data. this type of classification is just useful if the conclusion acquired is appropriate for the agronomist or the tip customer. the problem of predicting crop production is solved with data processing techniques. It should be considered that the sensor data are available for a few past, during which appropriate crop production was recorded. All this information creates a group of information which will be used for learning ways of classifying future crop production because new sensor data are available. There are different techniques of information mining which will be used for this purpose.

Agricultural crop prediction may be a very significant problem of agricultural organizations. Each agriculturist wants as soon as possible to understand what proportion yield to expect. Attempts to resolve this problem initiate to the time when first farmers started cultivating soil progressing to acquire harvest. Over the years,

agricultural crop prediction was conducted supported the farmer's experience of certain agricultural cultures and crops. However, this information may be acquired with the utilization of contemporary technologies, like GPS. Today acquiring an outsized amount of sensor data is comparatively quick, so agriculturists not only reap crops but also ever-larger amounts of knowledge.

1.5 Research Problem Definition & Research Purpose

Agriculture is considered the primary and foremost religion practiced in India from ancient times. Traditional farmers are planting the crops in their own land and they've been adapted to their needs. The natural crops are also grown and have been used by other species including humans, animals and birds.

Agriculture was among the users of data mining techniques to derive practical information from large data sets. Data mining may be implemented to forecast crop production to resolve the problem of evaluating and forecasting the crop data collection for agriculture. Big data mining in agriculture often provides a great opportunity to get a comprehensive view of their development.

The work focuses on 2 types of clustering approaches which can be used in precision agriculture. The collection of optimum cluster numbers plays a significant role in the study of clusters. To pick the optimal number of clusters, techniques such as the Elbow method and the Bayesian information criterion (BIC) procedure are employed. The research emphasizes cluster structure dependent on the K-mean cluster and the Gaussian Mixture Model clustering for agriculture crop data.

Consequently, the aim of this is not only to establish successful and efficient models for classifying crop data in agriculture, but also to evaluate the model analysis

in the sense of agriculture and identify useful clustering algorithms that can be used effectively for crop datasets in agriculture.

1.6 Research Questions & Research Objectives

Based on section 1.6, the research questions are defined as:

- How can data mining techniques be effectively applied to classification of agriculture crop dataset?
- How can data mining techniques be effectively applied to predict agriculture crop production?
- Does classification analysis offer a viable solution to know least crop production states in India?
- What are the opportunities and challenges in the application of data mining in Agriculture?

The research objectives are therefore defined as

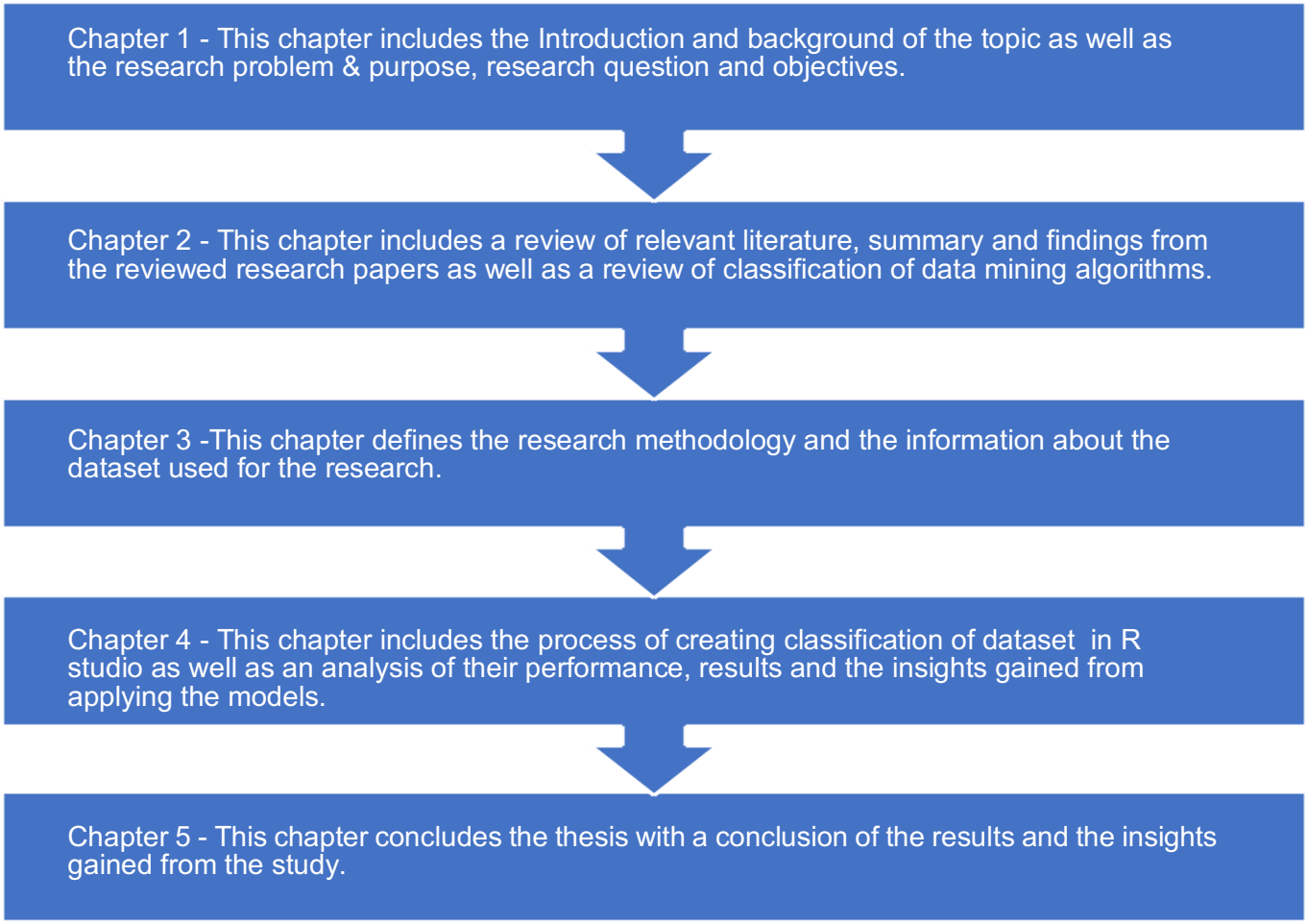
⇒ To use a large agriculture dataset and apply data mining algorithms to identify which states are in least crop production from this dataset.

⇒ Identify the best performing data mining techniques and algorithms.

⇒ To establish the opportunities and challenges that are present in the application of data mining in the agriculture.

1.7 Thesis Roadmap/Structure

This section defines the roadmap/structure of the thesis. The different chapters along with a brief explanation of the content of these chapters are illustrated in the figure shown below:


Chapter 1 - This chapter includes the Introduction and background of the topic as well as the research problem & purpose, research question and objectives.
Chapter 2 - This chapter includes a review of relevant literature, summary and findings from the reviewed research papers as well as a review of classification of data mining algorithms.
Chapter 3 -This chapter defines the research methodology and the information about the dataset used for the research.
Chapter 4 - This chapter includes the process of creating classification of dataset in R studio as well as an analysis of their performance, results and the insights gained from applying the models.
Chapter 5 - This chapter concludes the thesis with a conclusion of the results and the insights gained from the study.

CHAPTER 2 - Literature Review

2.1 Literature Review - Introduction

The study of the related literature is an integral part of any analysis. The literature review can be described as an objective and relevant summary of published literature in a particular field of study. It covers the work that has been carried out in the related area and provides the researcher with the information that can be used for further study and/or to recognize a research gap.

The literature review for this study discusses the findings from a collection of articles on Clustering analysis. The algorithms and methodologies used by the researchers have also been identified.

2.2 Algorithms

To implement our research on predicting annual Production for any particular state, different algorithms were required. Therefore, we've got implemented 2 clustering algorithms so as to predict the end result. K means clustering are used because not all algorithms were suitable for all the employment cases and also, we wanted show a comparison between different algorithms. In our research work, we've got used k means clustering and Gaussian Mixture Model.

2.3 Data Mining

Data mining is a action that uses a range of data analysis techniques and tools to find hidden relationships and patterns. The essential approach in data mining is to summarize the data and to extract valid and previously unknown helpful information.

2.4 Cluster Analysis

The main purpose of cluster analysis is to separate observations into a set of groups. A successful outcome of cluster analysis results in a number of clusters where results inside a cluster are as similar as possible, whereas differences within clusters are as large as possible. The cluster analysis will then evaluate the number of groups as well as the membership of the category observations. To evaluate the membership of the group, most clustering methods use a measure of similarity between observations. Similarity is generally represented by the difference between observations in the n- dimensional space of the variables. Cluster analysis is also a popular technique, partially because it tends to introduce a scientific component to the publication as a complicated statistical method. Paper readers using cluster analysis can be very aware of the challenges – cluster analysis should be used as a "exploratory data analysis tool" to better understand the multivariate nature of the data set.

There are also clustering methods that are not based on distance measurements, such as model-based clustering. These techniques typically identify clusters by optimizing the maximum likelihood function. The implicit assumption is that the data points representing the single clusters are usually multivariate distributed, and the algorithm aims to approximate the parameters from both the normal distribution and membership of each observation to each cluster.

2.5 Clustering Observations or Types

One of the key issues in cluster research is the existence of a number of various clustering strategies. Observations may be grouped into groups (clusters). When only one (of a few possible) cluster(s) is allocated to each observation, this is called "partitioning." Partitioning may result in a predefined (user-defined) number of clusters. It is also possible to create a hierarchy of partitions i.e. divide observations into 1 to n clusters (n = number of observations). It is termed hierarchical clustering. Hierarchical clustering also offers n cluster solutions, and on the based on these solutions, the consumer has to determine which result is more acceptable.

(a) Hierarchical Methods

The input of most hierarchical clustering algorithms is a distance matrix (distance between observations). Commonly used agglomerative strategies begin with single object clusters (each measurement is a cluster on its own) and expand clusters stepwise. The computationally more efficient reverse process starts from a common cluster with all outcomes and divides the classes step by step. This is named divisive clustering. At the beginning of an agglomerative algorithm, every observation produces its own class, contributing to a specific cluster of items. The number of clusters is decreased by one by combining (linking) the most connected groups at every algorithm stage.

The resemblance of the merged category, the new class, can be measured in all other classes, and the next two most comparable categories can be related, and so on. At the conclusion of the cycle, there is just one cluster remaining, with all the observations. A variety of different methods are needed to connect two classes, the best popular being the Wards method, average linkage, complete linkage and single

linkage. Ward's method is very successful at producing clusters that are more or less homogeneous and geochemically distinct from other clusters, opposed to the other technique, the benefit of the wards system being that it uses the variance methodology analysis to assess the variation between clusters to execute CA. Since cluster solutions develop tree-like (starting with the roots and finishing with the trunk) the results are also seen in a graphic called a dendrogram. Horizontal lines represent the association between two points or pairs, and thus the vertical axis shows the equivalent height or resemblance as a calculation of size. Objects are arranged in such a manner that the roots of the tree do not overlap. Linking two classes at a high altitude implies a heavy dissimilarity (and vice versa). A basic cluster arrangement would then be proposed if the observations are connected at a very low height and the distinct clusters are connected at a considerably higher size (the long roots of the tree). The dendrogram does not have cluster assignments of its own, so the number of clusters to be created will be selected by the user. This versatility is one of the subjective aspects of CA, because the user is able to achieve a certain target result by cutting the dendrogram at the height (phenonline) corresponding to the observable amount of clusters, which allows the clusters to be assigned to the artefacts. Visual analysis of a dendrogram is also helpful in having an initial understanding of the number of clusters to be created by a partitioning method.

(b) partitioning method

Unlike hierarchical clustering methods, partitioning methods allow it possible to pre-determine the amount of resulting clusters. As described above, since little is known about the results, it is progressively important to first carry out hierarchical clustering. The other option is to split the data into specific cluster numbers and to analyse the

results for regionalized data using a more subjective yet still reasonable approach to the assessment, by visually analysing the position of the corresponding clusters on a map. Often, this exploratory method will reveal interesting data structures. The K-means algorithm is a very common partitioning algorithm. This seeks to minimize the total squared distance between the results and their unit centres or centroids. Beginning with the initial k cluster centroids (e.g. random initialization of k observations), the algorithm assigns the measurements to the nearest centroid (e.g. Euclidean distances) recomputes the cluster centres and iteratively redirects the data points to the nearest centroid. Several algorithms exist for this purpose; the Hartigan and MacQueen algorithms are the most popular. There are also several improvements to the k-means algorithm. Manhattan distances are used for k-medians, and the centroids are the medians for every cluster. Hard competitive learning operates by randomly taking an inference from the data and shifting the closest centre to that point. Martinetz et al also implemented "neural gas", which is comparable to complicated competitive study, except in addition to the nearest centroid, the second closest centroid is often moved at iterations. Kaufmann and Rousseeuw suggested many clustering methods to be introduced in a number of software packages. The outcome of all such algorithms depends on the initial cluster centres, which are commonly used a random selection of k of the observations.

2.6 K-Means

K-means clustering involves the separation of a dataset into a variety of groups in such a manner that related objects fall or contribute to the same classes. The K-means method uses an iterative strategy to cluster the database. K-mean is that the foremost well-liked partitioning method of Clustering. It was firstly proposed by MacQueen in 1967. The K-Means formula

is effective in producing clusters for many practical applications. however, the computational complexity of the initial K-Means algorithm is extremely high, particularly for big datasets. The K-Means algorithm is a partition clustering technique that separates data into K groups.

Cluster's main objective is to group the item which is similar in one cluster and separate objects which are dissimilar by assigning them to similar clusters. One of the most common methods of clustering is algorithm K- clusters. It classifies items into predefined number of clusters offered by the user (provided clusters with K). The idea is to select random cluster centres, one per cluster. Some centres are chosen to be as far from each other as possible.

In this algorithm Euclidean measurement of distance between two multidimensional data points is used.

The K-Means approach attempts to reduce the amount of square distances between the cluster centre and all points.

- K - Number of desired clusters
- Output: A set of K clusters.

2.6.1Centroid initialization

After you have selected how many groups you want to partition the data into, there are a few options for selecting the initial centroid values. Take a random point from your data collection and label it a day. However, it is important to note that the k-cluster ends up in an estimated solution that converges to a local optimum- likely that might continue with a bad choice of centroids. The common approach to this is to run

a clustering algorithm several times and to choose the initial values that you can find for the best cluster performance (measured by the minimum average distance to the centroids-usually using a cluster sum of squares). Nevertheless, you should determine the amount of random initializations to be performed for the K-means cluster model in the sci-kit learning cycle using the n initialization parameter.

2.6.2 Assigning data points to a cluster

After you have picked the initial centroids, the next step is to allocate each data point to a cluster. Mathematically speaking, you 're doing this:

$$S_i^{(t)} = \left\{ x_p : \left\| x_p - m_i^{(t)} \right\|^2 \leq \left\| x_p - m_j^{(t)} \right\|^2 \forall j, 1 \leq j \leq k \right\}$$

(2.1)

Where clusters are represented as sets S1, ..., Si and m reflect the centroid value (which is the mean for all subsequent steps after the first step).

Set i contains all data points (x) where the distance from the data point (xp) to the mean of i (mi) is smaller than the distance from the data point to the mean of all other centroids (mj).

2.6.3 Calculate new centroid values

After every data point is allotted to a cluster, reassign the centroid value for every cluster to be the average of all the data points within the cluster.

$$m_i^{(t+1)} = \frac{1}{\left|S_i^{(t)}\right|} \sum_{x_j \in S_i^{(t)}} x_j$$

(2.2)

2.6.4 Reassign data points to new clusters

Follow a common method for initially distributing data points to clusters, but with new centroid values. In addition, measure new centroid values and reassign data points until you move to a cluster segmentation that prevents changes. Dealing with a large data collection and you don't want many data iterations.

$$S_i^{(t+1)} = \left\{ x_p : \left\| x_p - m_i^{(t+1)} \right\|^2 \leq \left\| x_p - m_j^{(t+1)} \right\|^2 \forall\, j,\ 1 \leq j \leq k \right\}$$

(2.3)

2.6.5 Elbow Method

To use the k-Means method we must find the optimum number of clusters k for the given dataset. The so-called "elbow method" can be used in some cases (as in the following) to calculate a close-optimal number of clusters k. When the elbow method is inefficient, a better result can be given by the silhouette method.

In the following we assign the maximum number of clusters arbitrarily to the squared root of the dataset size. This implies that if the data is distributed equally and a partition is constructed with up to this maximum number of clusters, then the number of data points within each cluster would be equivalent to the number of clusters, which is assumed to be large enough.

The following graph shows the WCSS curve in blue, the "extremes" line between the start and end points of the WCSS curve in green, and the orthogonal line in red.

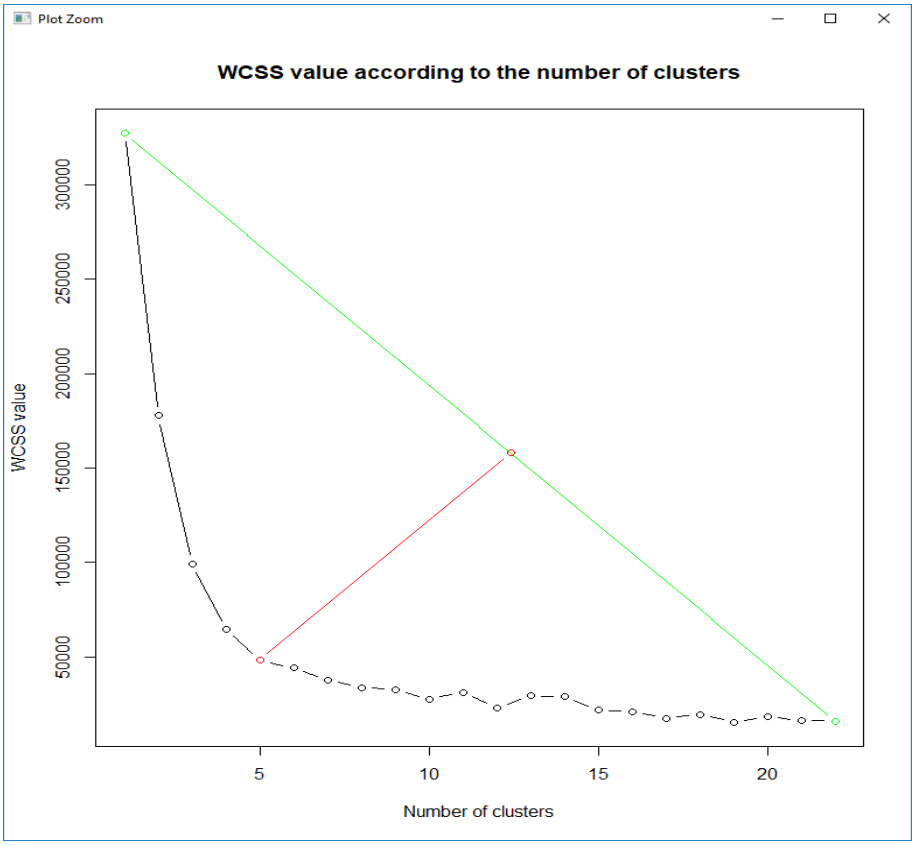


Figure 2.1: Elbow Method

The optimal number of clusters in the observations is given by elbow Point as Five which is similar to the initial number Four of groups.

2.7 Normal or Gaussian Distribution

In real life, several datasets may be based on Gaussian Distribution (Univariate or Multivariate). It is also very normal and logical to conclude that clusters come from various Gaussian distributions. Or, in other terms, an effort is made to model the dataset as a mixture of many Gaussian distributions. That is the main concept behind this model.

The probability density function of the Gaussian distribution is given in one dimension by

$$G(X|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2} \quad (2.5)$$

where mu and sigma and are respectively mean and variance of the distribution. For Multivariate (let's assume d-variate) Gaussian Distribution, the likelihood density function is defined

$$G(X|\mu,\Sigma) = \frac{1}{\sqrt{(2\pi)|\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(X-\mu)^T \boldsymbol{\Sigma}^{-1}(X-\mu)\right) \quad (2.6)$$

Here mu is a d dimensional vector denoting the mean of the distribution and **Σ** is the d X d covariance matrix.

2.8 Gaussian Mixture Model

Gaussian Mixture Models (GMMs) are more flexible than K-Means. We assume that the data points are Gaussian distributed for GMMs, which is less prohibitive than saying that they are round when utilizing the standard. Thus, two parameters are used to define the structure of the clusters: the mean and the standard deviation. The GMM model is designed to support mixed membership. GMM makes the mixed membership of cluster points is another consequence of its covariance structure. In k, the point belongs to one and only one cluster, whereas in GMM, the point belongs to each cluster to a different degree. The degree is focused on the probability of the point being formed from the normal distribution of each cluster (multivariate), with the cluster

centre as the mean distribution and the cluster covariance as its covariance. Mixed membership can be more appropriate (e.g. news articles can belong to multiple category clusters) or not (e.g. organisms may belong to just one species) based on the task.

Suppose there are K clusters (it is assumed, for the sake of convenience, that the number of clusters is identified and that it is K). And the mu and the sigma are therefore calculated for increasing k. Had there been just one distribution, they would have been calculated using the highest likelihood method. But since there are K such clusters, the probability density is defined as the linear function of the densities of all of such K distributions, i.e.

$$p(X) = \sum_{k=1}^{K} \pi_k G(X|\mu_k, \Sigma_k)$$

(2.7)

We assume that each of n ii d p dimensional observations, x1, x2, ..., xn, comes

from a Gaussian mixture distribution with a probability density function.

$$f(xj)=\sum i=1g\pi ifi(xj;\theta i)$$

(2.8)

where *g* is the number of components π_i is the prior probability for component *i* with ∑gi=1πi=1, $\theta_i = \{\mu_i, V_i\}$ is the set of the mean and covariance matrix

parameters for cluster *i*, and f_i is a multivariate Normal density (with a component-specific mean μ_i and covariance matrix V_i).

$$\mathrm{fi}(\mathrm{x};\theta\mathrm{i})=1(2\pi)\mathrm{p}/2|\mathrm{Vi}|1/2\exp(-12(\mathrm{x}-\mu\mathrm{i})'\mathrm{V}-1\mathrm{i}(\mathrm{x}-\mu\mathrm{i}))$$

(2.9)

Since each component refers to a cluster, we must apply to the component and the cluster in an exchangeable manner. The primary goal here is to estimate the cluster-specific precision matrices Wi=V−1i, though identifying the clusters is often of interest either as a direct or side product.

Given the data, the log-likelihood is

$$\mathrm{logL}(\Theta)=\sum\mathrm{j}=1\mathrm{nlog}(\sum\mathrm{i}=1\mathrm{g}\pi\mathrm{ifi}(\mathrm{xj};\theta\mathrm{i})),$$

(2.10)

where $\Theta = \{(\pi_i, \theta_i) : i = 1, 2, \ldots, g\}$ denotes the set of all unknown parameters. The Expectation-Maximization (EM) formula is also used to measure the ultimate likelihood. For high- results, the use of the maximum penalized likelihood estimator on the basis of a penalized log- is always advantageous.

$$\log L_P(\Theta) = \log L(\Theta) - p_\lambda(\Theta)$$

(2.11)

where $p_\lambda(\Theta)$ is to be specified as a penalty on all or a subset of the parameters. Various penalties have been proposed to achieve better performance in different contexts.

2.8.1 Bayesian information criterion (BIC)

This criterion offers us an estimation of how effective the GMM really is at forecasting the data we have. The lower the BIC is, the stronger the model to literally predict the data that we have, and the true, unknown, distribution by extension. This strategy penalizes equations of huge numbers of groups, in order to prevent overfitting.

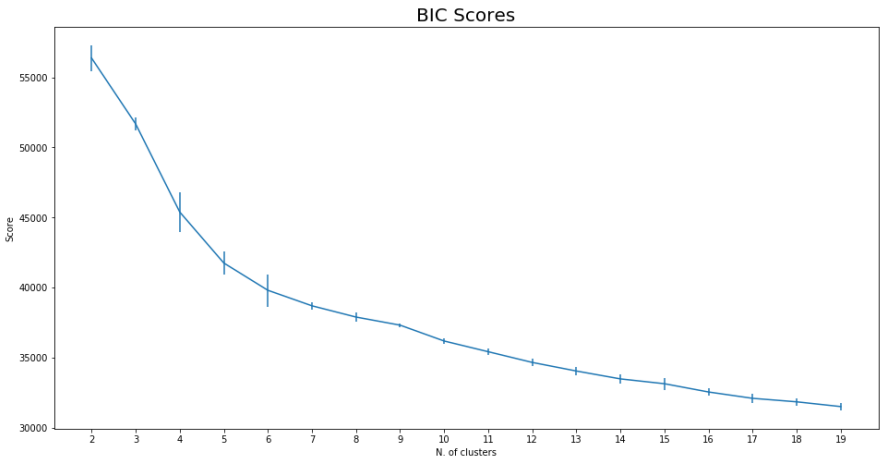


Figure 2.2: Bayesian information criterion

The larger the number of clusters, the better the model will be according to this criterion. Which implies we are not saved from overfit by the penalty BIC requirements which gives complex models. Or, the ranking fails in more prosaic terms.

But we should remember two things before we yell and throw out the technique. Firstly, the curve is very smooth and monotonous. The second is that, in various sections, the curve meets specific slopes. Starting from these two results, there is a significant temptation to test whether the slope of the BIC curve varies.

Technically, we will measure the BIC scores gradient curve. The definition of gradient is intuitively simple: if two successive points are of the same value, their gradient is zero. When they have separate values, their gradient will be negative, whether there is a lower value in the second point, or positive else. The amplitude of the gradient shows us how much the two values are different.

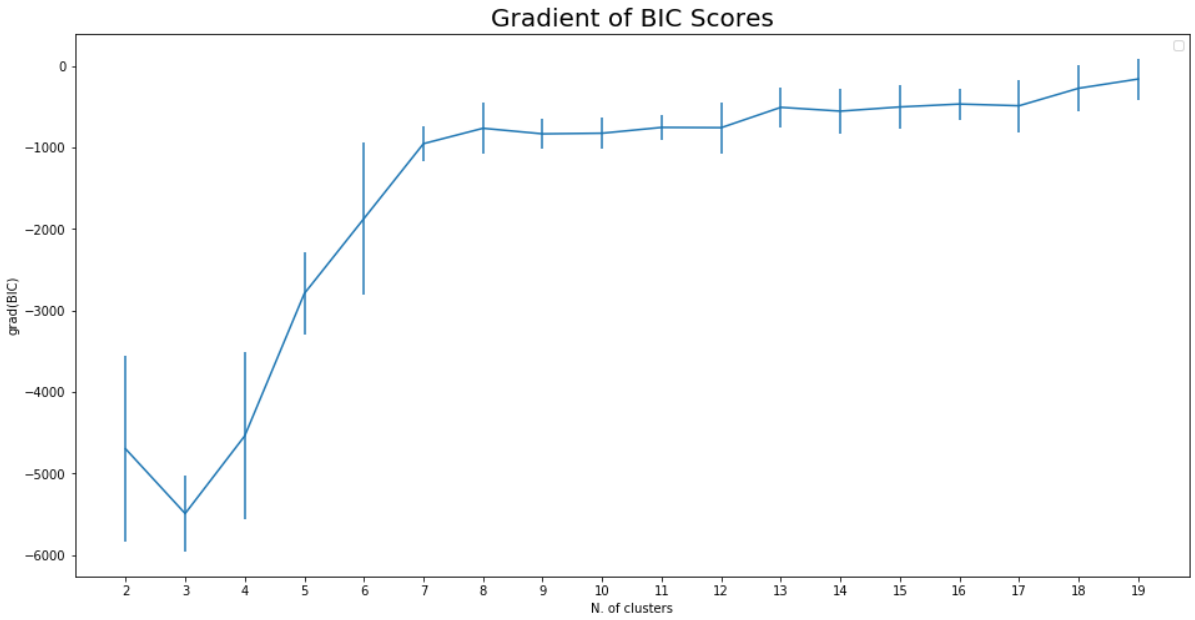


Figure 2.3: Gradient of BIC scores

As expected, there are negative values in all gradients. But we still see more obviously that the gradient is almost constant beginning with a cluster size of seven, i.e. the initial feature has a gentler decrease, i.e. there is no great advantage in raising the number of clusters. This method in brief, implies that we use six clusters.

2.8.2 The idea of BIC as regularization

BIC and AIC (Akaike Knowledge Criterion) are used for the component selection method as the regularization methods of linear regression.

- ❖ BIC / AIC is used for linear regression model regularisation.

Here for BIC the principle is expressed in a related way. In theory, it is also possible to model highly complicated clusters of data as a superimposition of a huge number of Gaussian datasets. There is no restriction on how many Gaussians to use for this purpose.

- ❖ The BIC method penalizes a large number of Gaussians, that is to say an overly complex model.

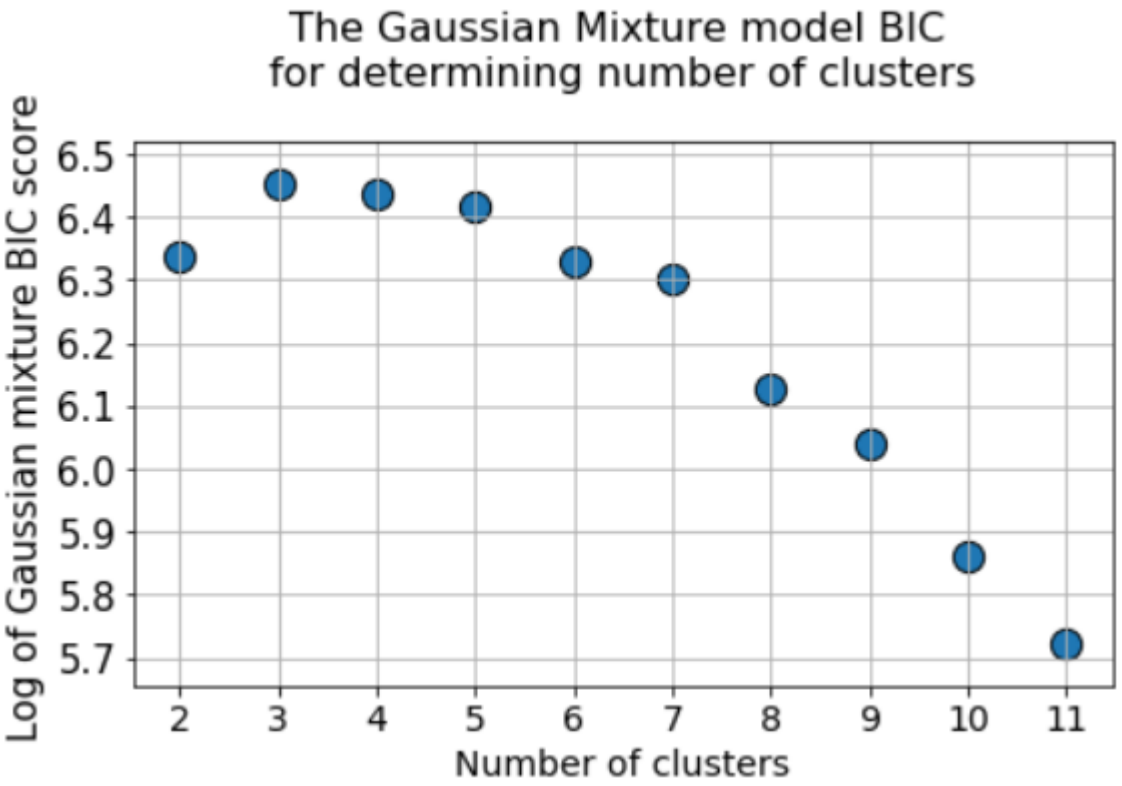


Figure 2.4: Determining number of Clusters (BIC)

CHAPTER 3 – RESEARCH METHODOLOGY

3.1 Research Process and Methodology

The design of crops clustering system by years can be found. The design includes selection of input documents, initial processes, document database and pattern discovery with k means clustering algorithm and Gaussian Mixture Modeling and the output forms of the system.

3.2 Research Strategy

The study approach identifies the goals and sets out a straightforward method for how the researcher wants to respond to the research question / goals. There are usually two types of data in the study project – principal and secondary. Primary data applies to data obtained by the researcher himself for a particular reason. Secondary data applies to data obtained by someone other than a researcher for a particular reason, but data provided by a researcher for a specific intent. It may be defined as a combination of primary and secondary data because this data has been gathered by someone other than this researcher, but the objective of this study stays close to that for which it was originally gathered for – data mining and data analytics.

3.3 Data Collection

The data used are agricultural crops based on province (1997-2015) originating from Open Government Data Platform India. The dataset contains 2, 46,091 records with seven variables corresponding to State, District, Year, Season, Crop, Area, Production and retrieved sample state-wise crop production. The sample data used

for k means clustering and Gaussian Mixture Modeling are states with parameter annual production. Clusters results using 3 clusters: low production cluster, normal production cluster, high production cluster. The study will also be used for predicting the clustered output by using predictive analytics to see the accuracy of the algorithm used with the research topic.

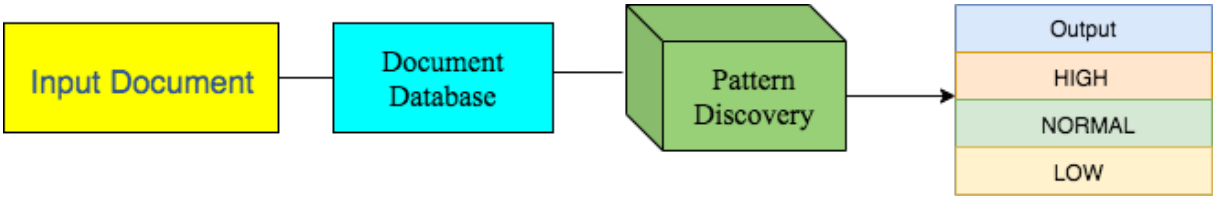


Figure 3.1: Research Process and Methodology

The description of figure:

Input Document

Input from the system is an important part of data processing related to crops. The file the used is .csv.

Initial Process

The initial process includes the filtering stage. At the filtering stage, all data that has passed the filtering process here in after referred to as term.

Document Database

For the memory efficiency and accuracy of the pattern discovery, then the threshold of the number of term will be maintained in the matrix. Only those terms that

have the strongest potential to characterize the group will be the material in the next stage of the pattern discovery.

Pattern Discovery

Cluster analysis is the most appropriate technique for obtaining large groups of existing research topics. The k means clustering algorithm was then used in the study to facilitate the group's analysis.

Output

The result of clustering from k means clustering stage is predetermined clusters.

3.3.1 Dataset Information

Variables	Description
Year	The data from the year 1997 till 2015
state	overall states in India
District	overall District of the states in India
Season	Kharif, Whole Year, Rabi
Crop	Plants like Paddy, Rice, Wheat, Cotton, Sugarcane, etc.,
Area	agriculture plants region in acres
Production	Specified year of production in Metric Tons

Table 3.1: Dataset Information

The columns of the dataset are shown in the table below:

The description of table:

State

The agriculture crop data set was collected of overall states in India.

District

The agriculture crop data set was been collected of overall District of the states in India.

Year

In the agriculture crop data set the years are mentioned from 1997 till 2015 in India.

Season

The fourth attribute in the dataset is season attribute and in the season attribute there are 3 Variables Kharif, Whole Year, Rabi.

Crop

The fifth attribute in the dataset is crop attribute and the information of crops like Paddy, Rice, Wheat, Cotton, Sugarcane, etc., are shown in the dataset.

Area

The target variable which has to be classified is the area attribute and the information of the area of agriculture plants region are shown in acres.

Production

The target variable which has to be classified is the attribute production of the Crop in the Specified year which is shown in Metric Tons.

3.3.2 Data pre-processing

Data pre-processing can be defined as the task of transforming raw data to a format that is readable and easily understood. This may involve data integration if the data is not in the correct format for analysis, data cleansing if the records are inaccurate. It also includes the task of checking for any data that is missing.

Prior to the processing of data on agricultural crops (1997-2015) by province, an average of 1 assessment attribute based production (1997-2015) was obtained. The processing of the data was done using r studio.

3.4 Exploratory Data Analysis of the Dataset in Tableau

This section presents an exploratory analysis of the dataset through data visualization in Tableau.

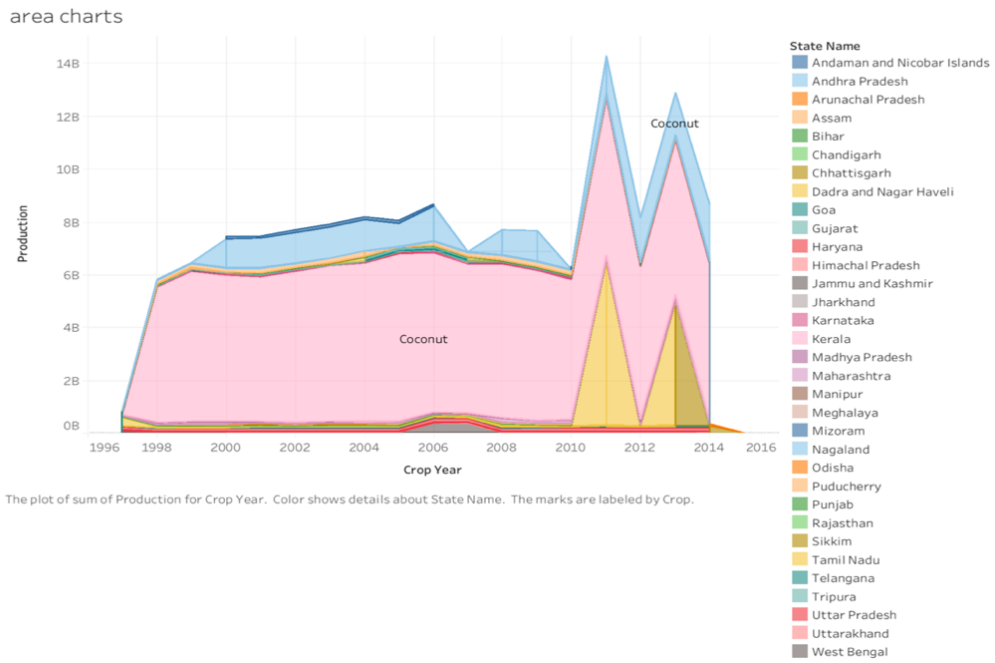


Figure 3.2: Area Chart Plot

The graph shows the changes in the total production from 1996 to 2016. The coconut crop production comparison appears to be the largest crop produced between 1996

and 2016. Kerala is the largest coconut-producing state in India compared to other countries.

Coconut crop production increased steadily between 1996 and 2016 in the state of Kerala, and Andhra Pradesh appears to be the second largest coconut-producing state in the blue colour of the graph. In conclusion, we can say that the total amount of production has increased and decreased, and there are only three countries with the highest production of coconut, and those are Kerala, Andhra Pradesh and Tamil Nadu.

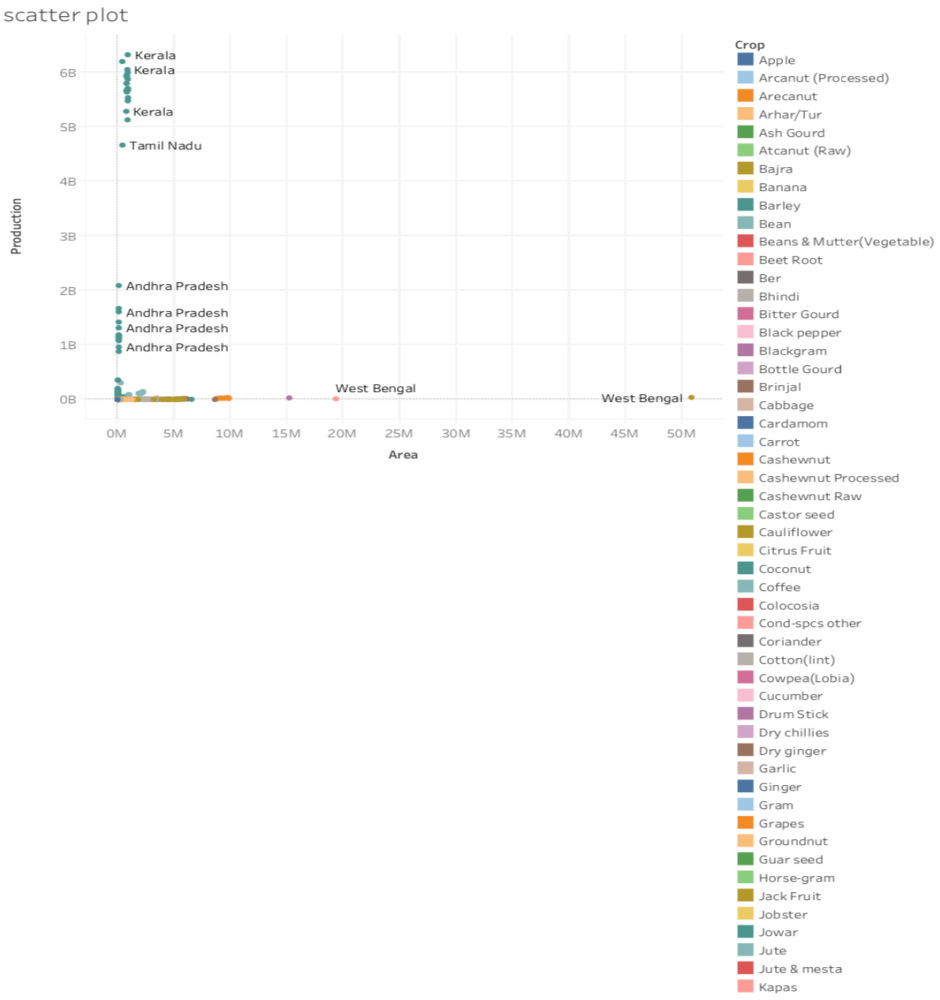


Figure 3.3: Scatter plot

A scatterplot shows the relationship between two data attributes. In the graph, we can see that the oil seed crop in West Bengal took the highest area of approximately 50,808,100 land in 1997 for the production of 38,657,300 crops. On the other hand, Kerala, Assam, Tamil Nadu and Andhra Pradesh took less land and produced a coconut crop.

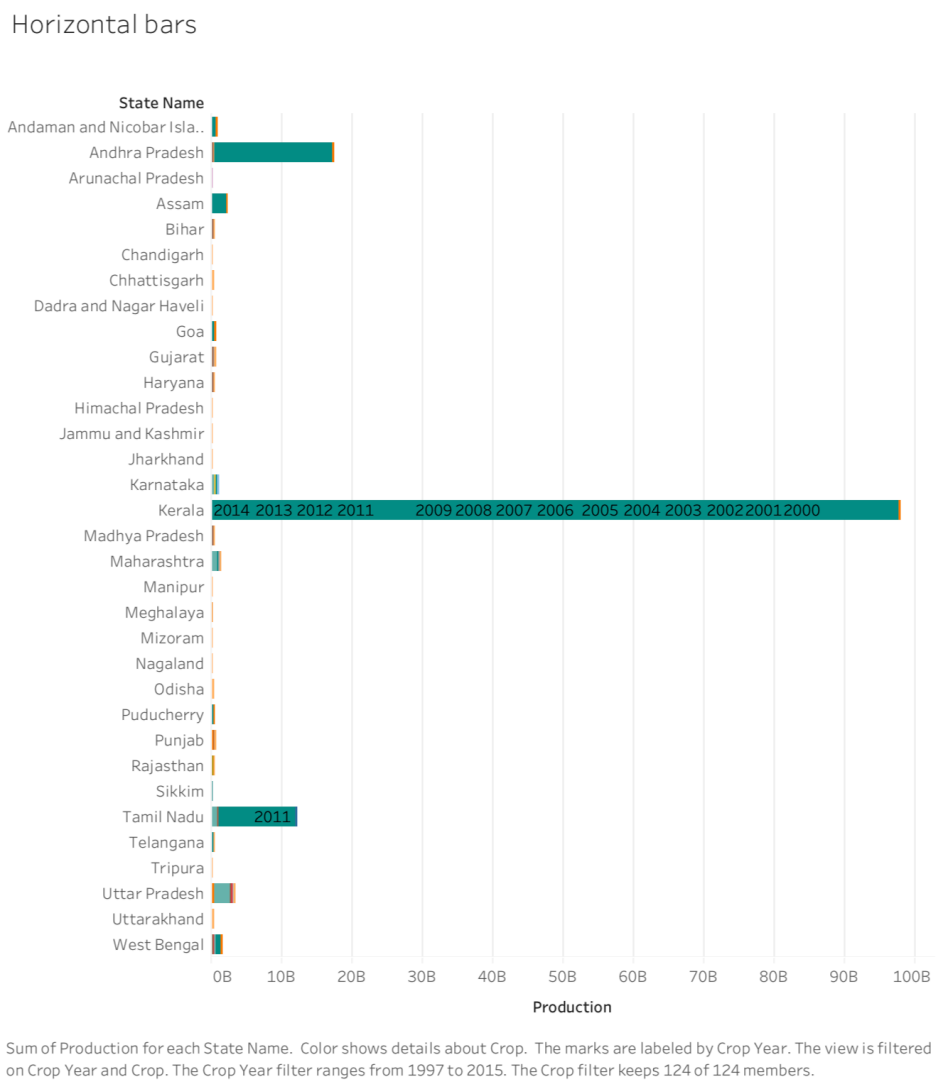


Figure 3.4: Horizontal Bar Plot

The bar chart shows the amount of crop produced in India.

Among the states, the coconut crop has the highest production in the state of Kerala from 1996 to 2015, as shown in the bar graph. The second largest state of coconut production is Andhra Pradesh. Moreover, Tamil Nadu appears to be the third largest state of coconut production.

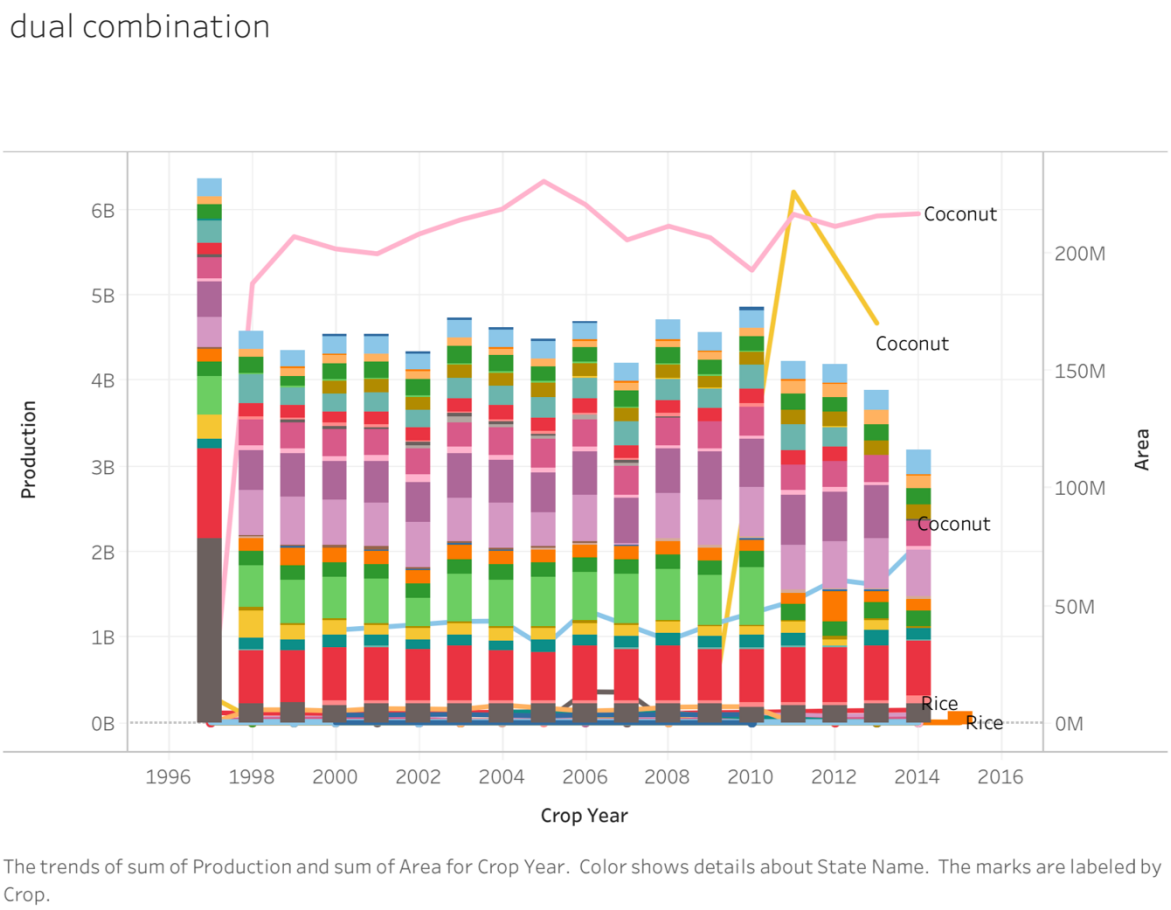


Figure 3.5: Dual Combination Plot

The graph compares the amount of production between the states of India between 1996 and 2016. As per crop, wise coconut appears to be the largest crop produced between 1996 and 2016 compared to other crops, and Kerala is the first largest coconut crop to be produced in bar graph. The second largest coconut crop producing state in India is Andhra

Pradesh. In line graph also the coconut crop is shown in pink colour line in Kerala state and Andhra Pradesh has shown in yellow colour. In the graph there are three axis shown those are production, area and crop year.

3.5 Data Analytics Using R

This section covers the choice and application of tool that were used for performing the data mining tasks for this thesis.

R is a free open- environment, a user interface design for statistical computing and data visualization. R programming is widely used by statisticians for emerging statistical software packages and data analysis. It is a wide range of packages and built-in features that support linear modelling, non-linear modelling, classification, clustering and more.

It is therefore implemented in a vast area of the financial sector, health care, etc. R programming also offers a wide range of statistical techniques for data visualization. Data scientists can use R to carry out a complicated analysis of sample observations, when distinguishing a significant correlation or cluster within the data, to place a finding in the product through enterprise scale tools.

CHAPTER 4 – IMPLEMENTATION, ANALYSIS AND RESULTS

4.1 Introduction

This chapter details the process of building and implementing machine learning models in R. The quality of the models will be assessed on the basis of measures such as rand-index or accuracy.

4.2 Implementation functions in r studio

4.2 .1 K means

Step 1

⇒ Prepare and load the dataset

- yield.data <- read.csv("~/Downloads/crop_production copy (1).csv")
- data <- na.omit(yield.data)
- dataset <- data.matrix(yield.data)
- dataset
- smple <- dataset[sample(nrow(dataset)),]
- sample_short <- smple[, c(6,7)]
- sample_matrix <- data.matrix(sample_short)

⇒ View the data as we have selected the rows

- View(smple_matrix)

⇒ removing the null values from the dataset with the below function

- sample <- na.omit(smple_matrix)

Step 2

⇒ Now, let's determine the number of clusters.

- wss <- (nrow(sample)-1)*sum(apply(sample,2,var))
- for (i in 1:5) wss[i]<-sum(kmeans(sample,centers=i)$withinss)
- plot(wss, type="b", xlab="Number of Clusters", ylab="Within Sum of Squares")

- It gives the elbow plot as follows.

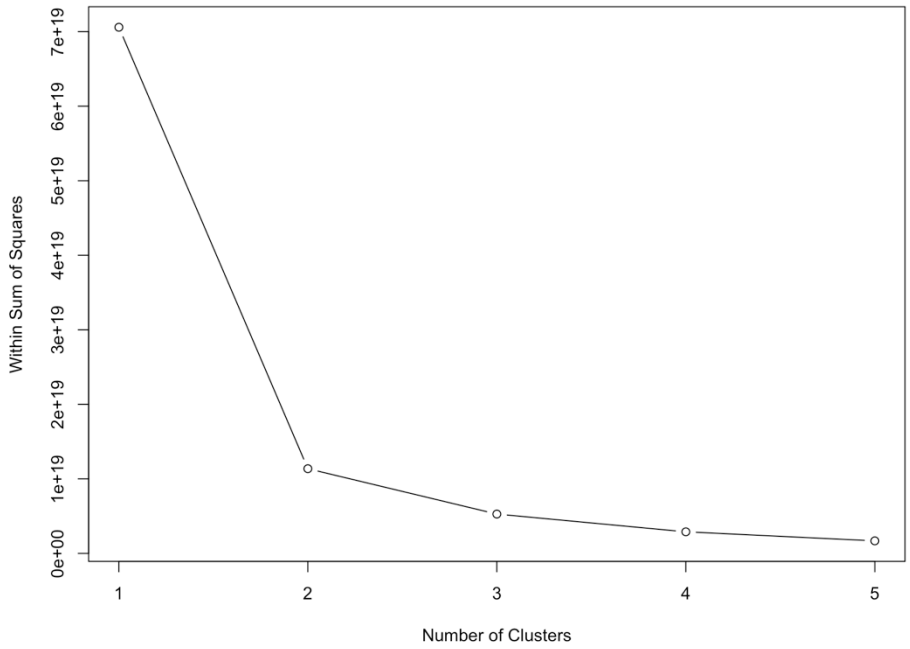


Figure 4.1 : Elbow Plot

❖ Here in the above plot the curve shows that for the data 3 clusters must be considered.

Step 3

- Now check the k-means with 3 clusters with the sample as data.

```
> cl
K-means clustering with 3 clusters of sizes 242133, 99, 129

Cluster means:
      Area  Production
1 12117.01    111275.7
2 88786.49 716899915.2
3 47974.77 335344955.4

Clustering vector:
   [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  [38] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  [75] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [112] 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [149] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [186] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [223] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [260] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [297] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [334] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [371] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [408] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [445] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [482] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [519] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [556] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [593] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [630] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [667] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [704] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [741] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [778] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [815] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [852] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [889] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [926] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [963] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
[1000] 1
 [ reached getOption("max.print") -- omitted 241361 entries ]

Within cluster sum of squares by cluster:
[1] 1.231794e+18 2.792852e+18 1.253165e+18
 (between_SS / total_SS =  92.5 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

Figure 4.2 : K means results with 3 clusters

⇒ Here, total_SS is the sum of squared distances of each data point to the global sample mean.

⇒ Whereas between_SS is the sum of squared distances of the cluster centroids to the global mean.

⇒ Here, 92.5 % is a measure of the total variance in the data set.

⇒ The goal of k-means is to maximize the between-group dispersion(between_SS).

⇒ So, higher the percentage value, better is the model.

⇒ You can plot the graph and cluster centroid using the following command.

- plot(sample, col =(cl$cluster) , main="k-means result with 3 clusters", pch=1, cex=1, las=1)
- points(cl$centers, col = "black", pch = 17, cex = 2)

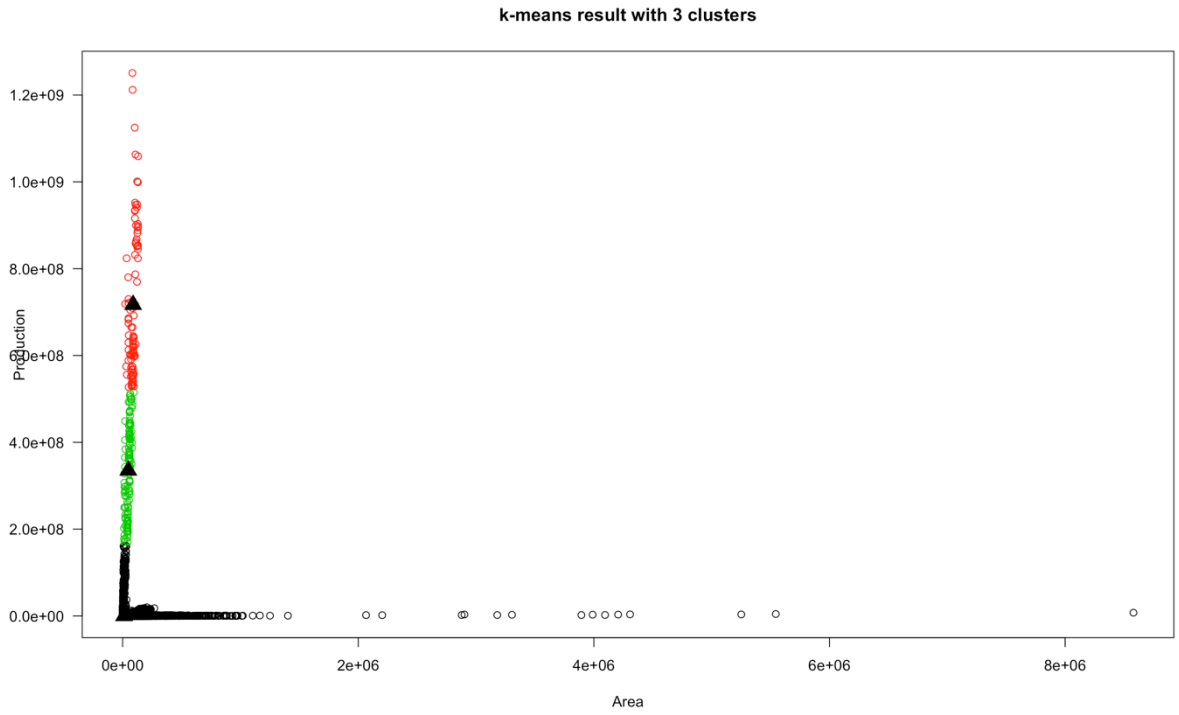


Figure 4.3 : Scatter plot with 3 clusters

⇒ For a more in-depth look at the cluster means,

```
> cl
K-means clustering with 3 clusters of sizes 242133, 99, 129

Cluster means:
      Area  Production
1 12117.01    111275.7
2 88786.49 716899915.2
3 47974.77 335344955.4
```

Figure 4.4 : Clusters Mean of 3 clusters

- ❖ With the above centres values of attributes area and production, we can see that cluster 1 has less production and area values, cluster 3 has normal production and area values and cluster 2 has high production and area values.

Step 4

➡ To know that which states having the largest and lower production and area in agriculture crop in the 3 clusters.

⇒ First select the two attributes from the dataset.

- ind <- sample(nrow(yield.data),c(6,7))
- Second add TRUE / FALSE index to the dataset
- yield.data[["crop"]] <- TRUE
- yield.data[["crop"]][ind] <- FALSE
- crop <- yield.data[yield.data[["crop"]]==TRUE,]
- crop.test <- yield.data[yield.data[["crop"]]==FALSE,]
- crop.net <- data.matrix(crop)

⇒ Remove the null variables in the crop dataset.

- crop.na <- na.omit(crop)

⇒ Load the library flexclust and separate the k means clusters to the crop dataset.

- library("flexclust")

⇒ create K-Means cluster of 3 groups based on columns

- km_net = kcca(crop.na[, c(6,7)], k=3, kccaFamily("kmeans"))

⇒ Individually check the groups for the attributes state name, area and production.

- g1 <- crop[clusters(km_net) == 1, c(1,6,7)]

- g2 <- crop[clusters(km_net) == 2, c(1,6,7)]
- g3 <- crop[clusters(km_net) == 3, c(1,6,7)]

⇒ Now check the group 1 to see which states are there.

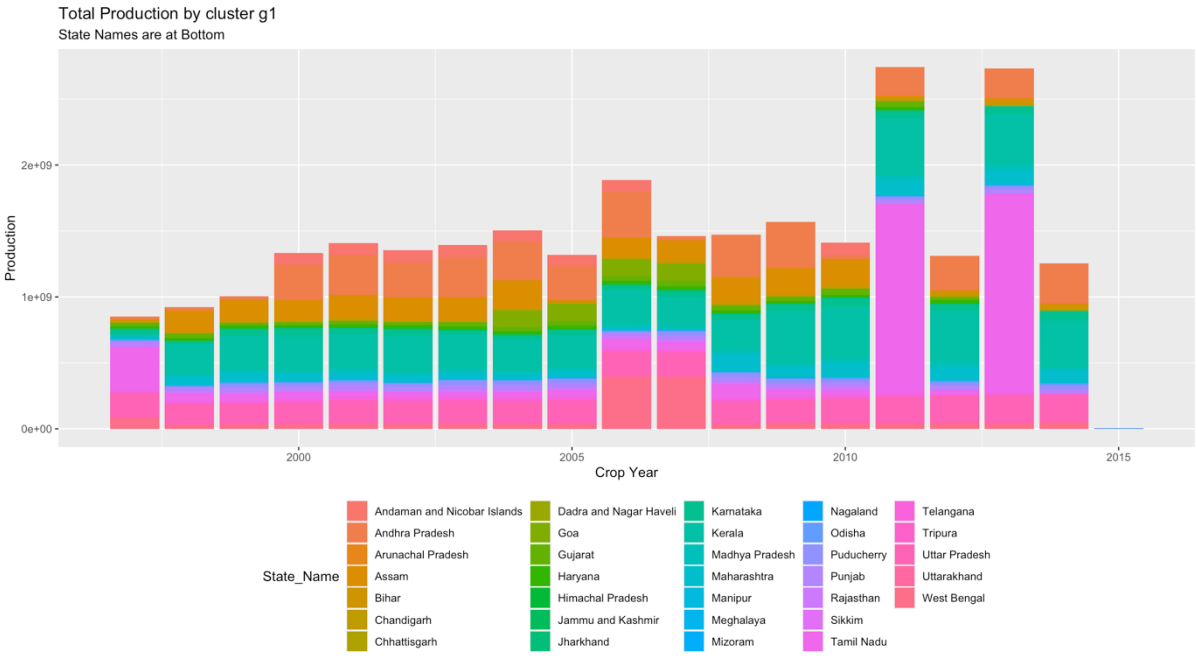


Figure 4.5 : Total Production in Group 1

- We can see Uttar Pradesh, Madhya Pradesh, Karnataka, Bihar, Assam, Odisha and others having the lowest production values because they exists in the group 1.

⇒ Now check the group 2 to see which states are there.

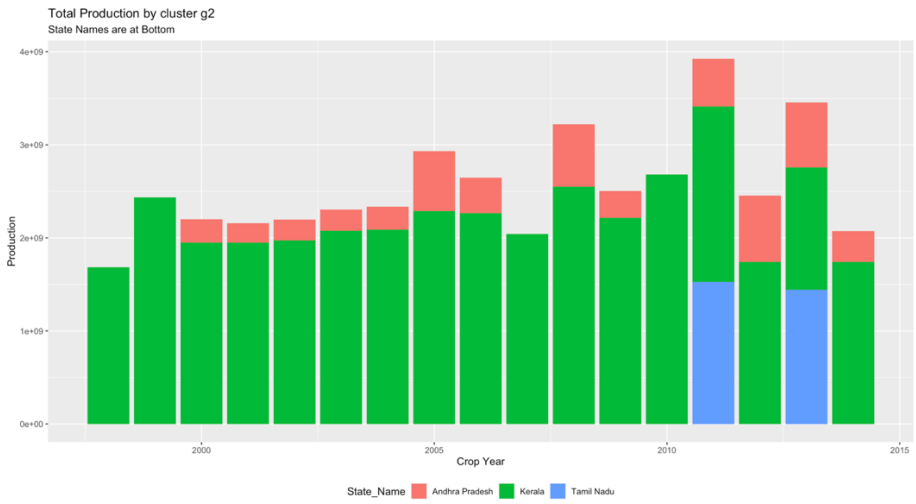


Figure 4.6 : Total Production in Group 2

- We can see Kerala, Tamil Nadu, Andhra Pradesh is maintaining the normal production in between the years 1997 till 2015 because they exists in the group 2.

  ⇒ Now check the group 3 to see which states are there.

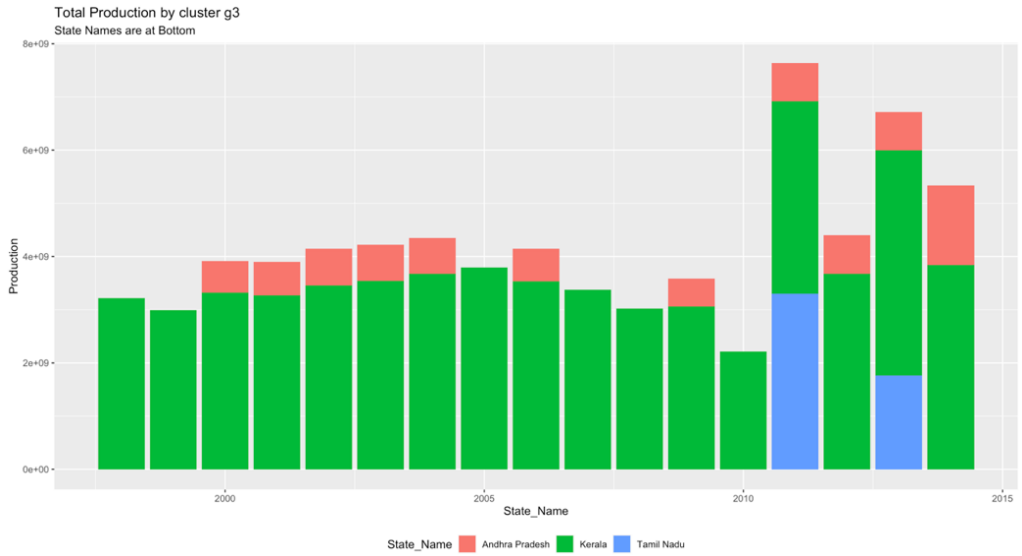


Figure 4.7 : Total Production in Group 3

- We can see Tamil Nadu, Kerala, Andhra Pradesh is maintaining the high production in between the years 1997 till 2015 because they exists in the group 3.

  ⇒ which states are in the top and least production groups

- top <- which.max(c(mean(g1$Production),mean(g2$Production), mean(g3$Production)))
- bottom <- which.min(c(mean(g1$Production),mean(g2$Production), mean(g3$Production)))
- print(paste("production Group is g", top, sep=""))
- print(paste("Least production Group is g", bottom, sep=""))

```
> top <- which.max(c(max(g1$Production),max(g2$Production), max(g3$Production)))
> bottom <- which.min(c(max(g1$Production),max(g2$Production), max(g3$Production)))
> print(paste("production Group is g", top, sep=""))
[1] "production Group is g3"
> print(paste("Least production Group is g", bottom, sep=""))
[1] "Least production Group is g1"
```

Figure 4.8: Top and Bottom in 3 Groups

Step 5

- predicting the clustered output without the production attribute using the package “library(clue)”
- apply the k means algorithm by removing the production attribute

```
train <- scale(dataset[,-7])
training<-kmeans(train, 3, iter.max = 10, nstart = 1)
```

⇒ install the “clue” package and then apply prediction on the k means result(training).

```
install.packages("clue")
library(clue)
```

⇒ predict classes for training data

```
cl_predict <- cl_predict(training)
cl_predict
summary(cl_predict)
```

```
> summary(cl_predict)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   1.000   2.000   1.993   3.000   3.000
```

***Figure* 4.9 : Summary of predicted clusters**

- For understanding compare the results of the k means and predicted k means

```
> summary(cl$cluster)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   1.000   1.000   1.001   1.000   3.000
```

***Figure* 4.10 : Summary of actual clusters**

Step 6

⇒ Predict which states is in the top crop production group

- top.pred <- print(paste("Predict which states is in the top crop production group, Group", top, sep=""), row.names = FALSE)
- pred.top <- print(data[cl_predict == top,], row.names = FALSE)
- summary(pred.top)

```
> summary(pred.top)
          State_Name       District_Name     Crop_Year            Season
 Madhya Pradesh: 8033   DINDIGUL  :  375   Min.   :1997   Autumn     :    0
 Tamil Nadu    : 7936   ERODE     :  341   1st Qu.:2002   Kharif     :    0
 Karnataka     : 6918   SALEM     :  338   Median :2005   Rabi       :    0
 Odisha        : 6642   TUMKUR    :  336   Mean   :2005   Summer     :10406
 Assam         : 5684   HASSAN    :  321   3rd Qu.:2009   Whole Year :56089
 Bihar         : 5671   COIMBATORE:  320   Max.   :2015   Winter     : 6044
 (Other)       :31655   (Other)   :70508
          Crop            Area            Production
 Sugarcane    : 6647   Min.   :     0.2   Min.   :0.000e+00
 Onion        : 4814   1st Qu.:    51.0   1st Qu.:7.600e+01
 Potato       : 4736   Median :   292.0   Median :7.040e+02
 Rice         : 4615   Mean   :  6521.4   Mean   :1.859e+06
 Dry chillies: 4156   3rd Qu.:  1912.5   3rd Qu.:7.976e+03
 Turmeric     : 3645   Max.   :491499.0   Max.   :1.251e+09
 (Other)      :43926
```

Figure 4.11 : Summary of predicted top cluster

- library(ggplot2)
- g<-ggplot(data = pred.top) + aes(x=pred.top$Crop_Year, y = Production, fill=State_Name) + geom_bar(stat = "identity") + xlab("State_Name") + ylab("Production") + ggtitle("predicted top crop production states")
- g + theme(legend.position="bottom", legend.box = "horizontal") + labs(subtitle="State Names are at Bottom")

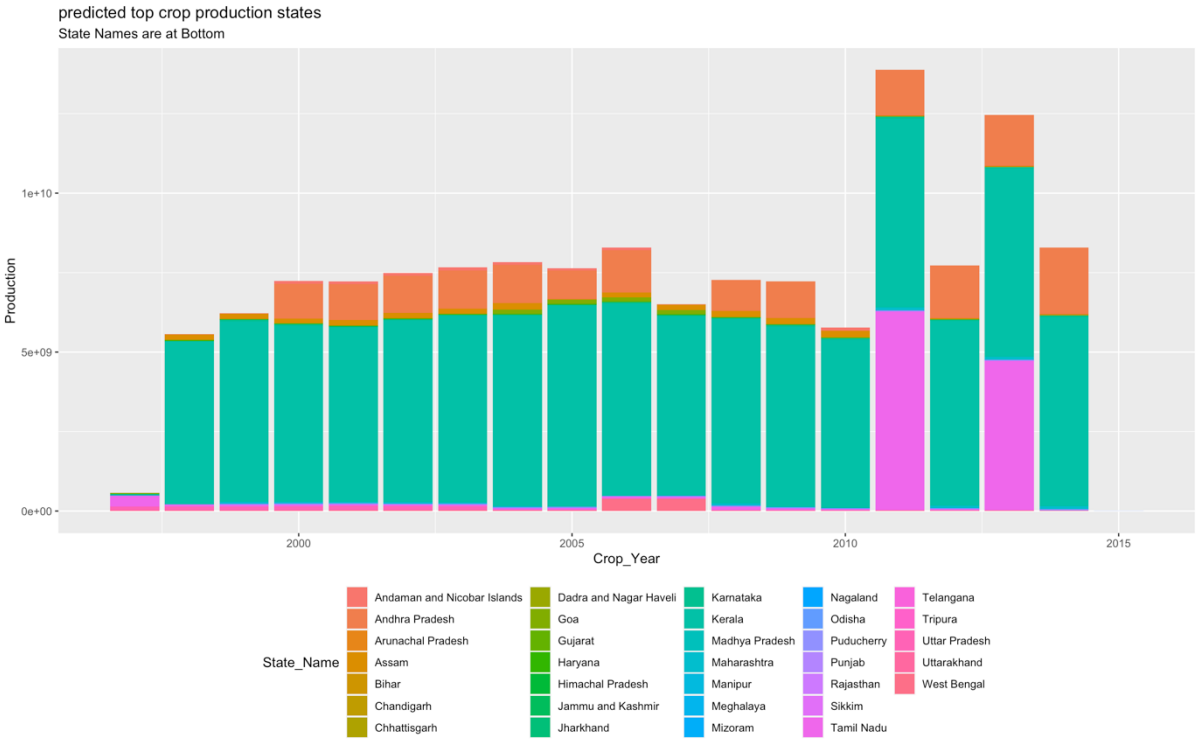


***Figure* 4.12 : Predicted Top Crop Production States**

⇒ Predict which states is in the least crop production group

- bottom.pred <- print(paste("Predict which states is in the least crop production group, Group", bottom, sep=""), row.names = FALSE)
- pred.bottom <- print(data[cl_predict == bottom,], row.names = FALSE)
- summary(pred.bottom)

```
> summary(pred.bottom)
          State_Name           District_Name     Crop_Year
 Uttar Pradesh :15080   BIJAPUR        :  451   Min.   :1997
 Karnataka     : 8913   VISAKHAPATANAM:  434   1st Qu.:2001
 Madhya Pradesh: 8814   KURNOOL        :  431   Median :2006
 Bihar         : 7931   KADAPA         :  426   Mean   :2006
 Maharashtra   : 6000   BELLARY        :  417   3rd Qu.:2010
 Assam         : 5347   CHITTOOR       :  412   Max.   :2015
 (Other)       :44369   (Other)        :93883
        Season                       Crop           Area
 Autumn     : 1693   Maize            :10773   Min.   :     0.5
 Kharif     :56141   Moong(Green Gram): 7513   1st Qu.:    98.0
 Rabi       :37689   Arhar/Tur        : 7191   Median :   686.0
 Summer     :  931   Gram             : 7039   Mean   :  8881.8
 Whole Year :    0   Groundnut        : 6878   3rd Qu.:  4395.8
 Winter     :    0   Jowar            : 6495   Max.   :652067.0
                     (Other)          :50565
   Production
 Min.   :      0
 1st Qu.:     81
 Median :    611
 Mean   :  13052
 3rd Qu.:   4593
 Max.   :2589591
```

Figure 4.13 : Summary of predicted Bottom cluster

- g<-ggplot(data = pred.bottom) + aes(x=pred.bottom$Crop_Year, y = Production, fill=State_Name) + geom_bar(stat = "identity") + xlab("Crop_Year") + ylab("Production") + ggtitle("predicted low crop production states")
- g + theme(legend.position="bottom", legend.box = "horizontal") + labs(subtitle="State Names are at Bottom")

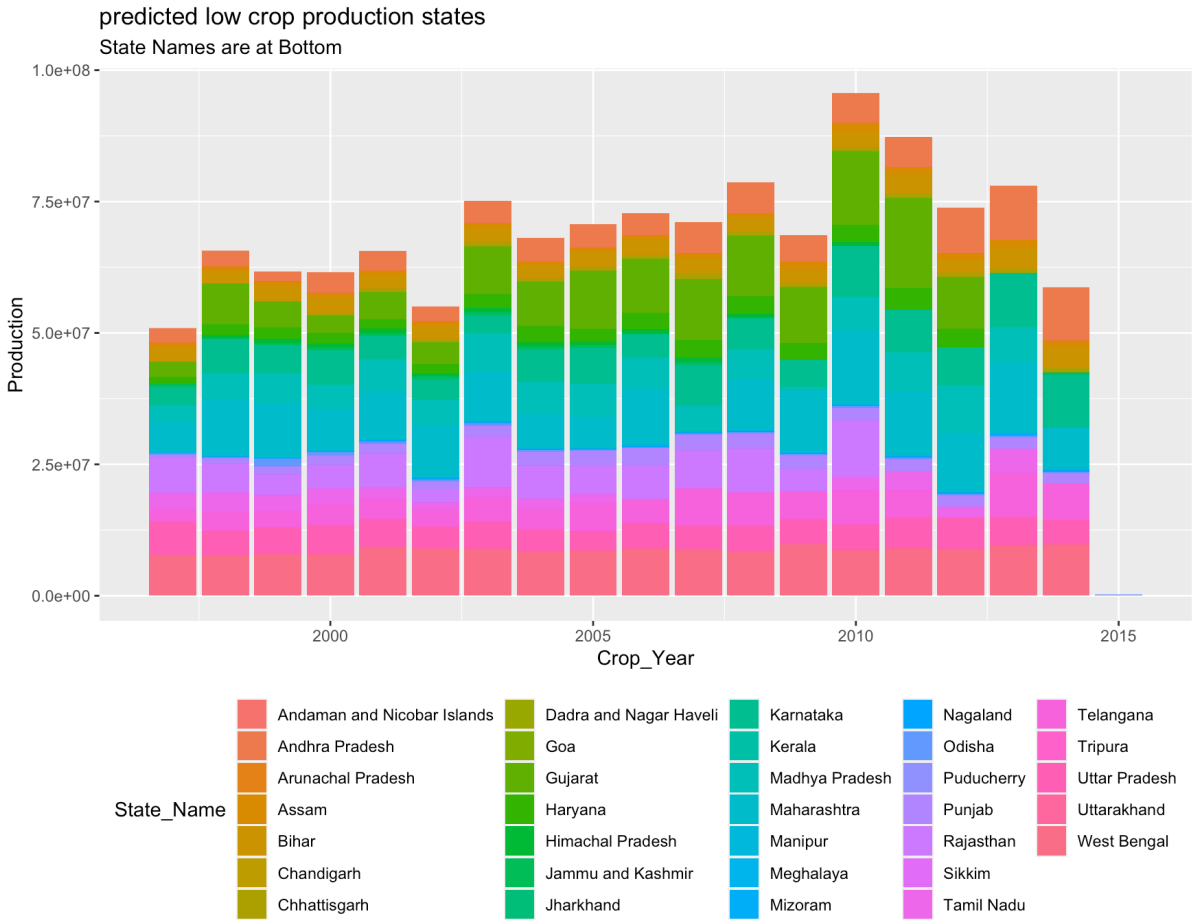


***Figure* 4.14 : Predicted Bottom Crop Production States**

⇒ Evaluating the predicted clustered output

- Assuming that true labels are available, then one could use the external validation methods(rand index, adjusted rand index, Jaccard index, Fowlkes Mallows index) to validate the output clusters.

◈ res = external_validation(cl_predict,data$Production,

method = "adjusted_rand_index", summary_stats = T)

```
> res = external_validation(cl_predict,data$Production,
+
+                           method = "adjusted_rand_index", summary_stats = T)

---------------------------------------
purity                          : 0.5696
entropy                         : 7.6381
normalized mutual information   : 0.0551
variation of information        : 13.2903
normalized var. of information  : 0.9717
---------------------------------------
specificity                     : 0.9983
sensitivity                     : 0.0021
precision                       : 0.3946
recall                          : 0.0021
F-measure                       : 0.0042
---------------------------------------
accuracy OR rand-index          : 0.66
adjusted-rand-index             : 6e-04
jaccard-index                   : 0.0021
fowlkes-mallows-index           : 0.0288
mirkin-metric                   : 19970296278
---------------------------------------
```

F*igure* 4.15 : External validation for Predicted clusters

- we can check the accuracy of the predicted clusters in the above figure
    - ➡ accuracy : 0.66 (66%)

4.2 .2 Gaussian Mixture Modeling

Step 1

⇒ Prepare and load the dataset

- yield.data <- read.csv("~/Downloads/crop_production copy (1).csv")
- data <- na.omit(yield.data)
- dataset <- data.matrix(data)
- dataset

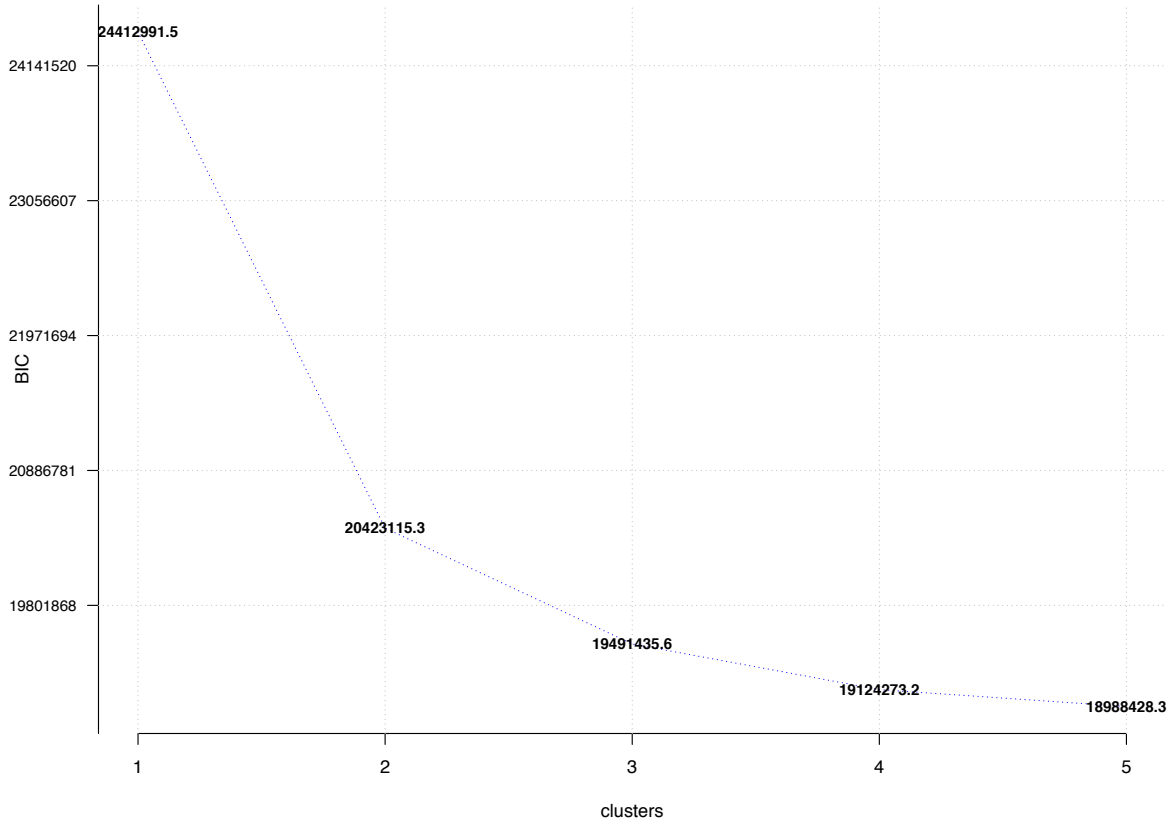
```
> yield.data <- read.csv("~/Downloads/crop_production copy (1).csv")
> data <- na.omit(yield.data)
> dataset <- data.matrix(data)
> dataset
      State_Name District_Name Crop_Year Season Crop       Area   Production
1              1           428      2000      2    3    1254.00 2.000000e+03
2              1           428      2000      2   76       2.00 1.000000e+00
3              1           428      2000      2   99     102.00 3.210000e+02
4              1           428      2000      5    8     176.00 6.410000e+02
5              1           428      2000      5   23     720.00 1.650000e+02
6              1           428      2000      5   29   18168.00 6.510000e+07
7              1           428      2000      5   39      36.00 1.000000e+02
8              1           428      2000      5  110       1.00 2.000000e+00
9              1           428      2000      5  112       5.00 1.500000e+01
10             1           428      2000      5  113      40.00 1.690000e+02
11             1           428      2001      2    3    1254.00 2.061000e+03
12             1           428      2001      2   76       2.00 1.000000e+00
13             1           428      2001      2   99      83.00 3.000000e+02
14             1           428      2001      5   23     719.00 1.920000e+02
15             1           428      2001      5   29   18190.00 6.443000e+07
16             1           428      2001      5   39      46.00 1.000000e+02
17             1           428      2001      5  110       1.00 1.000000e+00
18             1           428      2001      5  112      11.00 3.300000e+01
19             1           428      2002      2   99     189.20 5.108400e+02
20             1           428      2002      5    3    1258.00 2.083000e+03
21             1           428      2002      5    8     213.00 1.278000e+03
22             1           428      2002      5   16      63.00 1.350000e+01
23             1           428      2002      5   23     719.00 2.080000e+02
24             1           428      2002      5   29   18240.00 6.749000e+07
25             1           428      2002      5   38     413.00 2.880000e+01
26             1           428      2002      5   39      47.30 1.330000e+02
27             1           428      2002      5  110       5.00 4.000000e+01
28             1           428      2003      2   99      52.00 9.017000e+01
29             1           428      2003      5    3    1261.00 1.525000e+03
```

F*igure* 4.16 : Pre-processed Dataset

Step 2

⇒ Now, let's determine the number of clusters.

- opt_gmm = Optimal_Clusters_GMM(dataset, max_clusters = 5, criterion = "BIC", dist_mode = "maha_dist", seed_mode = "random_subset", km_iter = 10, em_iter = 10, var_floor = 1e-10, plot_data = T)



F*igure* 4.17 : Bayesian information criterion

- In addition to the previous mentioned functions, the Optimal Clusters GMM can be utilized to estimate the number of clusters of the data using either the AIC (Akaike information) or the BIC (Bayesian information) criterion.
- From the above figure with the BIC values mentioned we are considering the 3 clusters for the dataset.

Step 3

⇒ Before applying the classification algorithm selecting the 2 attributes which has to be classified.

```
smple <- dataset[sample(nrow(dataset)),]
sample_short <- smple[, c(6,7)]
sample_matrix <- data.matrix(sample_short)
```

⇒ Removing the null values from the dataset with the below function

```
sample <- na.omit(sample_matrix)
sample <- data.frame(sample)
```

⇒ install the “mclust” package and then apply classification function on the 2 attributes.

```
install.packages("mclust")
library(mclust)
```

⇒ Fit Gaussian Mixture Model

```
k <- 3 # no. of clusters
model <- Mclust(sample, G=3)
model$classification
```

```
> model$classification
   [1] 2 1 2 2 3 2 2 2 2 1 1 1 2 1 1 2 1 1 1 2 2 2 2 1 1 2 2 2 2 1 2 1 1 1 2 2 1
  [38] 2 1 1 2 2 1 1 1 3 2 1 2 2 1 1 3 1 2 1 1 2 1 1 1 1 1 2 3 1 3 1 2 1 1 2 1 2
  [75] 2 2 2 2 2 2 3 1 2 2 3 1 2 1 1 1 1 2 1 1 1 2 2 1 1 2 3 2 3 2 1 1 1 2 3 1 1
 [112] 2 2 2 1 1 1 1 2 1 2 1 2 3 2 1 2 2 2 2 2 2 2 3 1 2 3 3 2 2 2 1 1 2 1 3 2 2
 [149] 1 1 1 1 2 1 1 2 2 2 3 2 2 2 2 1 2 2 2 1 2 2 1 1 1 2 2 2 2 2 1 2 2 1 1 2 1
 [186] 1 1 1 1 1 1 2 3 2 2 2 3 2 2 2 1 1 3 2 1 1 2 1 1 2 2 2 1 1 2 2 2 2 1 2 2 2
 [223] 3 3 2 1 2 1 2 2 1 2 1 1 2 2 1 1 2 1 2 1 2 2 1 1 1 1 3 2 1 2 1 3 3 1 3 1 2
 [260] 2 2 2 1 2 2 3 2 2 2 3 2 2 1 1 2 2 2 2 2 3 1 1 1 3 2 1 1 3 1 2 3 1 2 1 1 1
 [297] 1 1 2 1 3 2 2 2 2 2 1 1 1 2 2 2 2 2 1 2 1 1 2 2 2 2 2 1 1 2 1 1 3 1 1 2 2
 [334] 2 2 2 2 1 1 1 2 2 2 1 3 1 1 1 2 1 2 1 2 3 1 2 2 2 1 3 1 2 1 1 1 2 1 2 2 2
 [371] 3 1 2 2 1 2 1 2 2 1 2 2 2 2 2 3 2 2 2 1 2 2 1 2 1 1 2 2 2 1 3 2 2 1 1 1 2
 [408] 1 1 2 1 2 1 1 2 3 2 2 1 2 1 1 3 2 2 2 2 1 3 2 2 2 1 1 1 2 2 1 1 1 2 2 2 2
 [445] 3 2 2 1 2 2 2 3 2 2 3 2 1 1 2 1 2 1 2 2 2 2 1 2 2 1 2 1 1 1 3 2 2 2 1 2 2 1
 [482] 3 1 2 2 2 1 2 2 1 2 2 1 1 2 2 1 2 2 2 2 1 1 2 3 1 3 2 2 1 2 2 1 2 2 2 1 1 1
 [519] 3 1 2 1 2 2 1 2 2 1 1 1 2 2 2 2 1 2 2 2 1 2 1 1 3 2 1 2 2 1 2 2 2 1 2 2 3
 [556] 2 1 3 1 1 2 1 1 2 1 1 1 3 2 2 1 1 1 2 2 1 1 2 2 1 3 2 1 1 2 2 1 2 2 2 2 3
 [593] 1 1 1 2 2 2 2 1 1 2 2 2 3 2 2 2 2 1 2 2 1 2 1 2 2 1 2 2 2 2 1 2 1 2 2 1 2
 [630] 3 1 1 1 1 2 2 2 1 1 2 2 1 1 1 1 1 1 2 2 2 2 1 2 1 1 2 2 2 2 2 2 2 1 2 2 1 2
 [667] 1 2 3 1 2 1 2 3 1 2 1 2 2 2 2 2 2 2 2 1 1 1 1 2 2 2 1 1 2 1 2 2 2 1 1 2 1
 [704] 2 2 1 2 2 2 1 1 2 2 1 1 2 3 1 2 1 1 2 2 1 2 1 3 2 2 2 2 1 1 2 1 1 1 2 1 2
 [741] 1 3 2 1 2 2 2 2 2 1 2 1 3 1 2 2 1 2 2 2 1 3 1 3 1 1 2 2 1 1 2 2 1 2 2 2 2 1
 [778] 2 2 1 3 1 3 2 2 3 1 2 1 3 2 1 3 1 1 2 1 2 1 2 3 2 2 3 2 1 3 2 3 2 1 2 1 1
 [815] 1 2 2 2 1 2 1 1 2 2 1 1 3 2 2 1 1 1 1 2 1 2 1 1 1 1 3 2 1 2 3 1 2 1 2 1 2
 [852] 1 2 1 2 1 2 2 1 2 2 1 1 2 3 2 3 2 2 2 1 1 2 1 1 2 2 2 2 2 1 2 1 1 1 1 2 1
 [889] 1 2 2 2 2 2 2 2 2 3 2 2 3 2 1 2 3 2 1 2 3 2 2 1 2 2 2 2 2 3 2 1 1 1 1 1 1
 [926] 2 1 2 2 1 3 1 2 3 3 2 3 3 2 2 2 2 1 1 2 2 2 1 2 1 3 1 2 2 1 1 1 2 2 2 1 1
 [963] 2 2 1 1 1 2 1 1 1 2 1 2 1 1 2 1 2 1 1 2 1 1 2 2 2 3 2 1 2 1 3 3 2 2 3 2 2
[1000] 2
 [ reached getOption("max.print") -- omitted 241361 entries ]
```

Figure 4.18 : Gaussian Mixture Modeling Classification

⇒ Adding the individual attributes to the each of the classification clusters

⇒ There are 3 clusters for which we have added the attributes “state name”, “crop year”, “season”, “area”, “production”.

⇒ As mentioned we wanted to know which of the above mentioned attributes exists in the individual clusters.

- g1 <- data[model$classification == 1, c(1,3,4,6,7)]
- g2 <- data[model$classification == 2, c(1,3,4,6,7)]
- g3 <- data[model$classification == 3, c(1,3,4,6,7)]

⇒ check that which states are in cluster g1

```
> summary(g1)
         State_Name      Crop_Year             Season          Area
 Uttar Pradesh :13160   Min.   :1997   Autumn      : 1941   Min.   :      0
 Madhya Pradesh: 8860   1st Qu.:2002   Kharif      :37085   1st Qu.:     85
 Karnataka     : 8206   Median :2006   Rabi        :26133   Median :    600
 Bihar         : 7305   Mean   :2006   Summer      : 5767   Mean   :  12120
 Assam         : 5733   3rd Qu.:2010   Whole Year  :22034   3rd Qu.:   4462
 Odisha        : 5331   Max.   :2015   Winter      : 2353   Max.   :4307200
 (Other)       :46718
   Production
 Min.   :0.000e+00
 1st Qu.:8.700e+01
 Median :7.200e+02
 Mean   :5.993e+05
 3rd Qu.:7.000e+03
 Max.   :1.001e+09
```

***Figure* 4.19 : Summary of Group 1**

- g1.prod.year <- aggregate(g1$Production ~g1$State_Name+g1$Crop_Year, FUN = sum)
- names(g1.prod.year) <- c("State_Name","Year","Production")

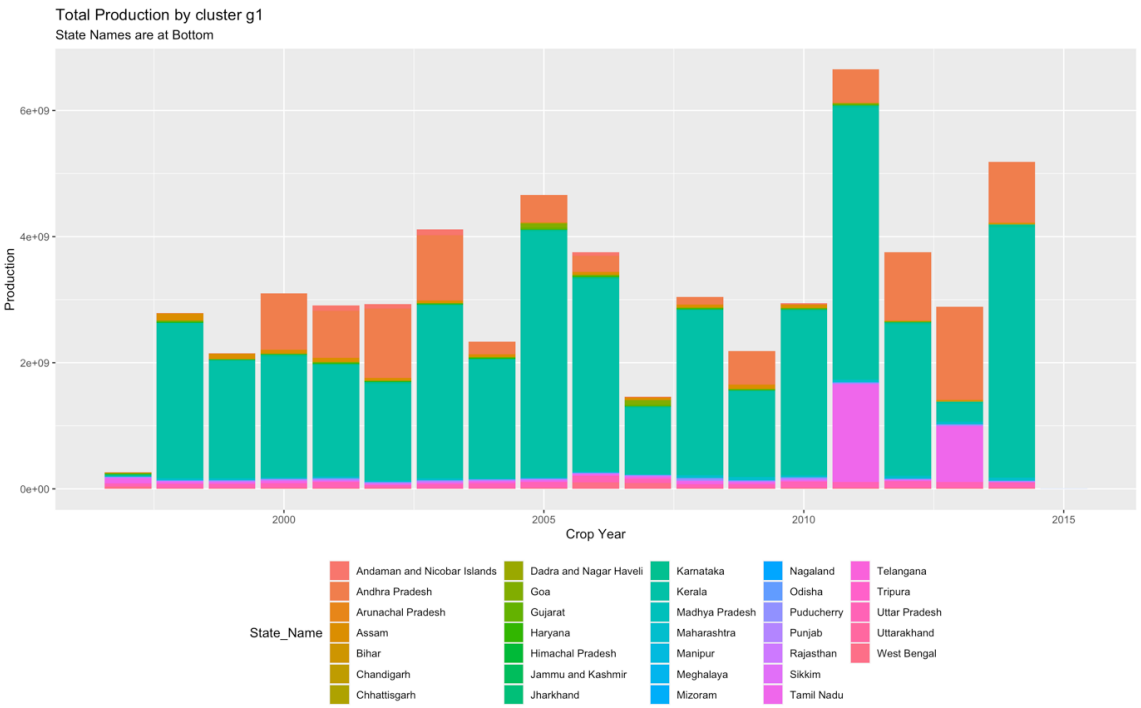


***Figure* 4.20 : Total Production in Group 1**

- As seen in the plot, we can see that almost all states exists in the g1 cluster, but the highest output is maintained by Andaman and Nicobar Islands, Andhra Pradesh, Arunachal Pradesh, Assam etc.

  ⇒ Now check that which states are in cluster g2

```
> summary(g2)
          State_Name      Crop_Year             Season           Area
 Uttar Pradesh :17147   Min.   :1997   Autumn     : 2573   Min.   :      0
 Madhya Pradesh:11733   1st Qu.:2002   Kharif     :48904   1st Qu.:     88
 Karnataka     :10999   Median :2006   Rabi       :34280   Median :    608
 Bihar         : 9922   Mean   :2006   Summer     : 7730   Mean   :  12182
 Assam         : 7572   3rd Qu.:2010   Whole Year :29018   3rd Qu.:   4628
 Odisha        : 7041   Max.   :2015   Winter     : 3154   Max.   :5544000
 (Other)       :61245
   Production
 Min.   :0.000e+00
 1st Qu.:8.800e+01
 Median :7.370e+02
 Mean   :5.461e+05
 3rd Qu.:7.094e+03
 Max.   :1.251e+09
```

Figure 4.21 : Summary of Group 2

- g2.prod.year<-aggregate(g2$Production~ g2$State_Name+g2$Crop_Year, FUN = sum)
- names(g2.prod.year) <- c("State_Name","Year","Production")
- g<-ggplot(data = g2.prod.year) + aes(x=Year, y = Production, fill=State_Name) + geom_bar(stat = "identity") + xlab("Crop Year") + ylab("Production") + ggtitle("Total Production by cluster g2")
- g + theme(legend.position="bottom", legend.box = "horizontal") + labs(subtitle="State Names are at Bottom")

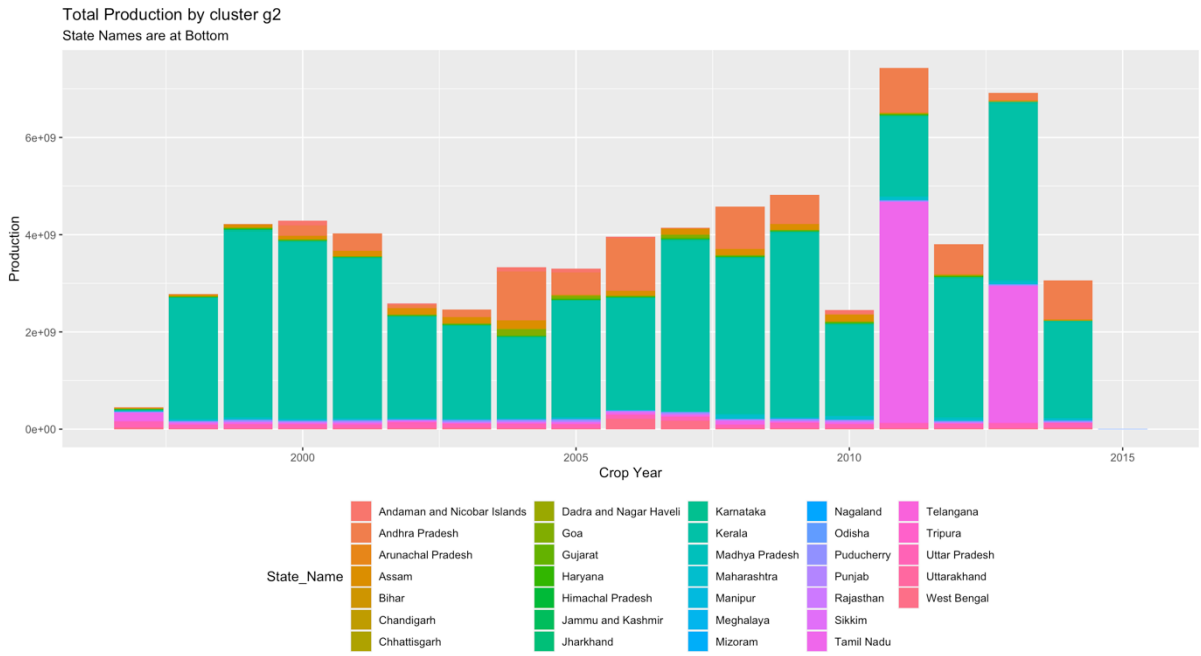


Figure 4.22: Total Production in Group 2

- As compared with group 1 Kerala state has increased the production in 2011 and 2014. But Kerala and Andhra Pradesh is highest among all states in group 2.

  ⇒ Now check that which states are in cluster g3

```
> summary(g3)
         State_Name      Crop_Year             Season          Area
 Uttar Pradesh : 2882   Min.   :1997   Autumn     : 416   Min.   :      1
 Madhya Pradesh: 2011   1st Qu.:2002   Kharif     :8294   1st Qu.:     85
 Karnataka     : 1874   Median :2006   Rabi       :5747   Median :    604
 Bihar         : 1647   Mean   :2006   Summer     :1314   Mean   :  12292
 Assam         : 1317   3rd Qu.:2010   Whole Year :5075   3rd Qu.:   4362
 Tamil Nadu    : 1208   Max.   :2015   Winter     : 543   Max.   :8580100
 (Other)       :10450
   Production
 Min.   :        0
 1st Qu.:       88
 Median :      729
 Mean   :   721174
 3rd Qu.:     7000
 Max.   :948000000
```

Figure 4.23: Summary of Group 3

- g3.prod.year <- aggregate(g3$Production ~ g3$State_Name+g3$Crop_Year, FUN = sum)
- names(g3.prod.year) <- c("State_Name","Year","Production")
- g<-ggplot(data = g3.prod.year) + aes(x=Year, y = Production, fill=State_Name) + geom_bar(stat = "identity") + xlab("State_Name") + ylab("Production") + ggtitle("Total Production by cluster g3")
- g + theme(legend.position="bottom", legend.box = "horizontal") + labs(subtitle="State Names are at Bottom")

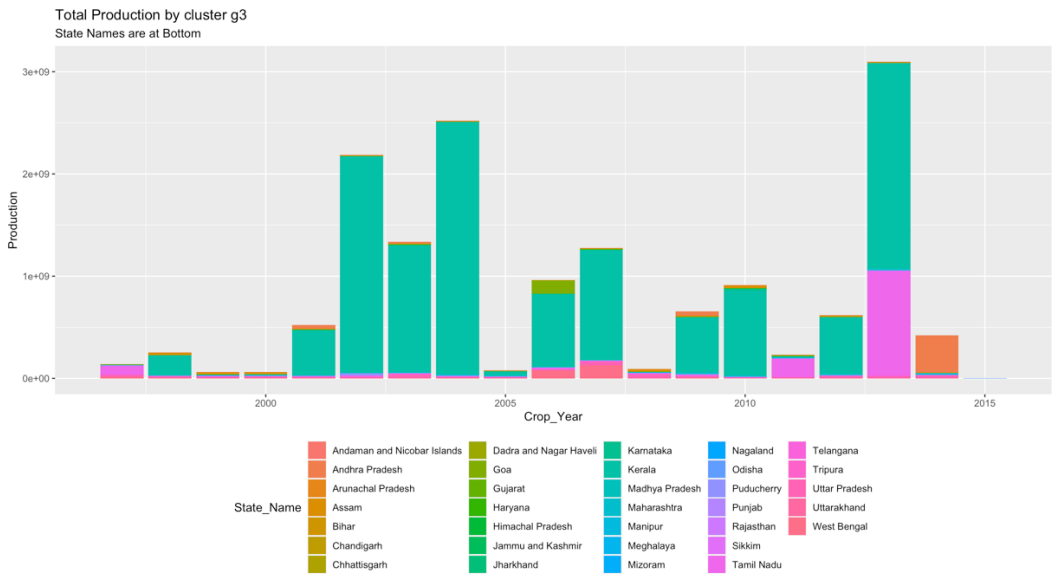


***Figure* 4.24: Total Production in Group 3**

- In the plot g3 the highest production is maintained by the Kerala state. As this group is the least crop maintained states among them we can see which states are very low and medium in this plot.

  ⇒ which states are in the top and least production groups

- top <- which.max(c(max(g1$Production),max(g2$Production), max(g3$Production)))
- bottom <- which.min(c(max(g1$Production),max(g2$Production), max(g3$Production)))
- print(paste("production Group is g", top, sep=""))
- print(paste("Least production Group is g", bottom, sep=""))

```
> top <- which.max(c(max(g1$Production),max(g2$Production), max(g3$Production)))
> bottom <- which.min(c(max(g1$Production),max(g2$Production), max(g3$Production)))
> print(paste("production Group is g", top, sep=""))
[1] "production Group is g2"
> print(paste("Least production Group is g", bottom, sep=""))
[1] "Least production Group is g3"
```

Figure 4.25: Top and Bottom in 3 Groups

- Assuming that true labels are available, then one could use the external validation methods(rand index, adjusted rand index, Jaccard index, Fowlkes Mallows index) to validate the output clusters.

- res = external_validation(model$classification,data$Production, method = "adjusted_rand_index", summary_stats = T)

```
> res = external_validation(model$classification,data$Production,
+
+                           method = "adjusted_rand_index", summary_stats = T)

---------------------------------------
purity                          : 0.6227
entropy                         : 7.6945
normalized mutual information   : 0.0431
variation of information        : 13.2275
normalized var. of information : 0.978
---------------------------------------
specificity                     : 0.9982
sensitivity                     : 0.0018
precision                       : 0.4292
recall                          : 0.0018
F-measure                       : 0.0036
---------------------------------------
accuracy OR rand-index          : 0.5685
adjusted-rand-index             : 0
jaccard-index                   : 0.0018
fowlkes-mallows-index           : 0.0278
mirkin-metric                   : 25347039584
---------------------------------------
```

F*igure* 4.26: External validation of the model classification

- we can check the accuracy of the classification clusters in the above figure
  - ➡ accuracy : 0.5685(56.85%)

Step 4

⇒ predicting the clustered output without the production attribute using the package “library(ClusterR)”

⇒ install the “ClusterR” package and then apply prediction on the gmm result(training).

- install.packages("ClusterR")
- library(ClusterR)

⇒ apply the GMM algorithm by removing the production attribute

- data.matrix <- data.matrix(data)
- train <- scale(data.matrix[,-7])

- gmm <- GMM(train,3)

```
> gmm$weights
[1] 0.1219441 0.4051882 0.4728677
> gmm$covariance_matrices
          [,1]      [,2]      [,3]      [,4]      [,5]       [,6]
[1,] 0.9460129 0.9741448 1.0435821 1.0011886 0.8758319 5.97992108
[2,] 0.4110996 1.0275164 1.0146733 0.9352242 0.9869578 0.01779588
[3,] 0.5036174 0.9812958 0.9719012 1.0310829 0.9671661 0.00106520
```

Figure 4.27: GMM weights and covariance matrices

Step 5

⇒ Apply the result of the GMM algorithm for predicting the output result

⇒ predict centroids, covariance matrix and weights

- pr = predict_GMM(train, gmm$centroids, gmm$covariance_matrices, gmm$weights)
- pr$cluster_labels

```
> pr$cluster_labels
  [1] 2 2 2 2 2 0 2 2 2 2 2 2 2 2 0 2 2 2 2 2 2 2 2 0 2 2 2 2 2 2 2 2 0 2 2 2 2 2 2 0 2 2 2
 [46] 2 2 2 2 0 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 0 2 2 2 0 2 2 2 2 2 2 2 2
 [91] 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 0 2 2 2 2 2 2 2
[136] 2 2 0 2 2 2 2 2 2 2 2 2 0 2 2 2 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
[181] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 0 2 2 0 2 0 2 0 2 2 2 2 2 0 2 0 2 2 0 0 2 0 2 2
[226] 2 0 0 2 0 2 2 0 2 0 2 2 2 2 2 2 2 0 0 0 2 0 0 2 0 2 2 2 0 0 2 2 2 2 2 2 0 0 2 0 2 0 2 2 0 2 2 2
[271] 0 0 2 0 2 2 2 2 2 0 2 2 2 2 2 2 0 2 0 0 2 2 0 2 2 2 2 2 2 2 2 2 2 0 2 2 2 0 2 2 2 2 2 0 0 2 0
[316] 2 2 2 2 2 0 2 2 2 2 2 2 2 2 0 2 2 2 2 2 2 0 2 2 2 0 2 2 0 2 2 2 2 2 2 2 2 2 2 0 2 0 2 2 2 2 2 0 0 2 0
[361] 2 2 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 2 2 2 0 2 2 2 0 2 0 2 2 2 2 2 2 2 2 2 2 0 2 2 2 0 2 2 2 0 0 2 0 2
[406] 2 2 2 2 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 2 0 2 2 2 2 0 2 0 2 2 2 2 2 2
[451] 0 2 2 2 0 2 2 2 2 0 0 2 0 2 2 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 0 2 2 0
[496] 2 0 2 2 2 2 2 2 2 2 2 0 0 2 2 2 0 2 0 0 2 0 2 2 2 2 2 2 2 0 2 2 2 0 2 2 2 2 2 2 2 2 2 2 2 0 2 2 2 2 0 2
[541] 2 2 2 2 2 2 2 0 2 2 2 2 2 2 0 0 2 0 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 2 0 2 2 2 0 2 2 2 2 2 0 2 0 2 2 2
[586] 2 2 0 2 2 2 2 2 0 0 2 0 2 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 0 2 2 0 2 2 2 2 2 0 2 2 2 2 0 0 0 0 0 2 0
[631] 0 0 2 2 2 2 2 0 2 2 2 2 2 2 2 2 0 2 2 2 0 0 2 0 2 2 2 2 2 0 2 2 2 2 2 2 2 2 0 2 2 0 2 2 2 2 2 0 2 0
[676] 0 2 2 2 2 2 0 2 2 2 2 2 0 0 2 0 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 2 0 2 2 0 2 2 2 2 0 2 2 0 2 0 2 2 2 2 0
[721] 2 2 2 2 2 0 0 2 0 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 0 2 2 2 2 0 2 0 0 2 2 0 2 2 0 2 2 2 2 2 2 2 2 0 2 2 2
[766] 2 2 2 2 2 0 0 2 0 2 2 2 2 2 0 2 2 2 2 2 1 2 2 2 2 2 0 2 2 2 2 2 0 2 2 0 2 0 2 0 2 2 2 2 0 2 0 2 2 0 2 2
[811] 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 0 0 2 2 2 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 2 2 2 2 0 2 2 2
[856] 2 0 0 2 2 2 2 0 2 0 2 2 2 0 0 2 2 2 2 0 2 2 2 2 2 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 0 0 2 2 0 2 2
[901] 2 2 0 2 2 2 2 2 2 2 2 2 1 2 0 2 2 2 2 2 2 2 2 0 2 2 2 2 2 0 2 0 2 2 2 0 2 0 2 0 0 2 2 2 2 2 2 2
[946] 2 2 2 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 0 2 2 2 0 0 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 0 2 2 2 2 0 2
[991] 2 2 2 2 2 0 0 2 2 0 2
[ reached getOption("max.print") -- omitted 241361 entries ]
```

Figure 4.28: Predicted clusters Labels

Step 6

⇒ Start the cluster labels with 1 with the below mentioned function

- pred <- as.factor(pr$cluster_labels+1)

❖ Top Group

⇒ Predict which states is in the top crop production group

- top.pred <- print(paste("Predict which states is in the top crop production group, Group", top, sep=""), row.names = FALSE)
- pred.top <- print(data[pred == top,], row.names = FALSE)
- summary(pred.top)

```
> summary(pred.top)
          State_Name            District_Name      Crop_Year
 Bihar          :15986   TUMKUR          :   715   Min.   :1997
 Madhya Pradesh:15896   BILASPUR        :   709   1st Qu.:2002
 Karnataka      :15634   BANGALORE RURAL:   703   Median :2006
 Assam          :12954   SHIMOGA         :   694   Mean   :2006
 Odisha         : 9445   HAVERI          :   673   3rd Qu.:2010
 Chhattisgarh   : 9282   VISAKHAPATANAM  :   670   Max.   :2015
 (Other)        :41379   (Other)         :116412
         Season                       Crop            Area
 Autumn     : 2385   Maize             : 6220   Min.   :   0.1
 Kharif     :41983   Sesamum           : 5064   1st Qu.:  57.0
 Rabi       :31260   Moong(Green Gram): 4844   Median : 276.0
 Summer     : 7750   Urad              : 4341   Mean   : 837.8
 Whole Year :33854   Potato            : 4280   3rd Qu.:1057.0
 Winter     : 3344   Sugarcane         : 4146   Max.   :7631.0
                     (Other)           :91681
   Production
 Min.   :        0
 1st Qu.:       52
 Median :      300
 Mean   :    43655
 3rd Qu.:     1436
 Max.   :160779000
```

***Figure* 4.29: Summary of top predicted cluster**

⇒ Add the ggplot2 Library to plot the states, crop year and Production of the predicted top states group

- library(ggplot2)
- g<-ggplot(data = pred.top) + aes(x=pred.top$Crop_Year, y = Production, fill=State_Name) + geom_bar(stat = "identity") + xlab("State_Name") + ylab("Production") + ggtitle("predicted top crop production states")
- g + theme(legend.position="bottom", legend.box = "horizontal") + labs(subtitle="State Names are at Bottom")

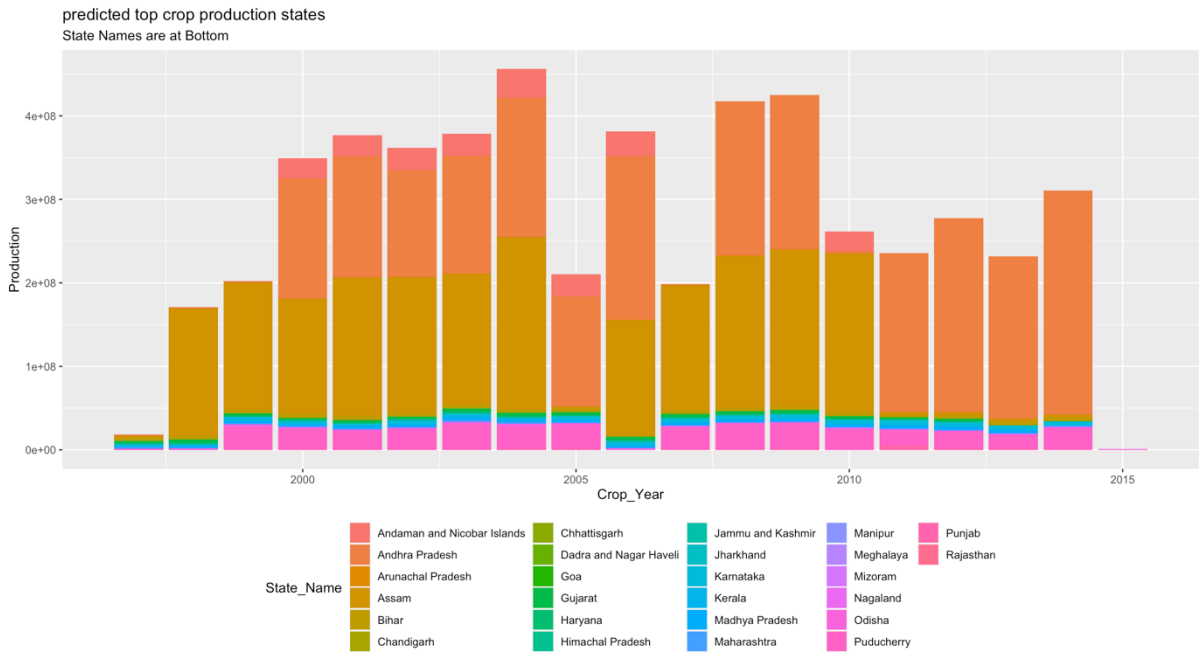


***Figure* 4.30: Predicted Top Crop Production States**

⇒ As per the plot we can see that Andaman and Nicobar Islands and Andhra Pradesh is the highest predicted production states among other states and second is the Arunachal Pradesh which seems to be the highest predicted production state.

❖ Bottom Group

⇒ Predict which states is in the bottom crop production group

- print(paste("Predict which states is in the least crop production group, Group", bottom, sep=""), row.names = FALSE)
- pred.bottom <- print(data[pred == bottom,], row.names = FALSE)
- summary(pred.bottom)

```
> summary(pred.bottom)
          State_Name    District_Name   Crop_Year             Season
 Karnataka      :263   DAVANGERE: 20   Min.   :1997   Autumn      :  8
 Uttar Pradesh :248   MEDAK    : 19   1st Qu.:2002   Kharif      :420
 Tamil Nadu     :155   FIROZABAD: 18   Median :2006   Rabi        :269
 Madhya Pradesh:116   MYSORE   : 18   Mean   :2006   Summer      : 66
 Telangana      : 76   TUMKUR   : 18   3rd Qu.:2010   Whole Year :264
 Haryana        : 32   HASSAN   : 17   Max.   :2014   Winter      :  3
 (Other)        :140   (Other)  :920
           Crop           Area             Production
 Maize        : 64   Min.   :     1.0   Min.   :       0
 Groundnut    : 47   1st Qu.:    74.2   1st Qu.:      76
 Dry chillies: 43   Median :   530.5   Median :     614
 Jowar        : 42   Mean   :  6901.0   Mean   :   31446
 Rice         : 42   3rd Qu.:  4266.5   3rd Qu.:    5745
 Urad         : 42   Max.   :392173.0   Max.   :11703240
 (Other)      :750
```

***Figure* 4.31: Summary of bottom predicted cluster**

⇒ Plot the states, crop year and Production of the predicted bottom states group

- g<-ggplot(data = pred.bottom) + aes(x=pred.bottom$Crop_Year, y = Production, fill=State_Name) + geom_bar(stat = "identity") + xlab("Crop_Year") + ylab("Production") + ggtitle("predicted low crop production states")
- g + theme(legend.position="bottom", legend.box = "horizontal") + labs(subtitle="State Names are at Bottom")

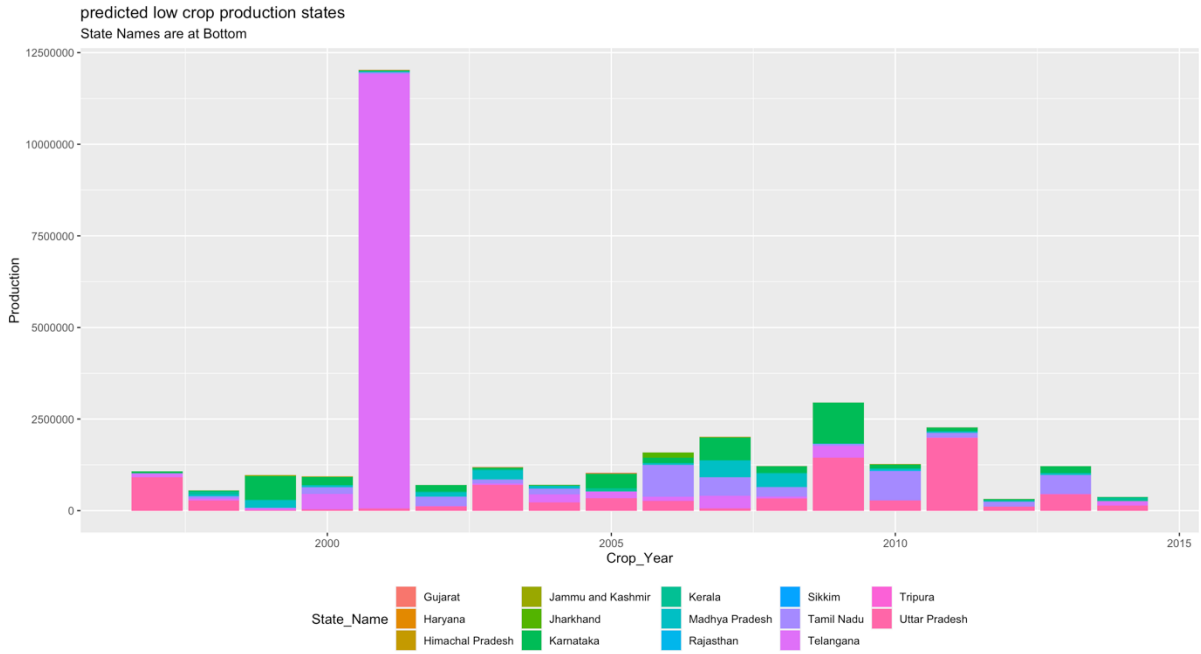


Figure 4.32: Predicted Bottom Crop Production States

⇒ As per the plot we can see that Telangana was highest in the year 2001, but later years Jharkhand is the highest predicted production state among other states and second is the Tripura which seems to be the highest predicted production state.

- Assuming that true labels are available, then one could use the external validation methods(rand index, adjusted rand index, Jaccard index, Fowlkes Mallows index) to validate the predicted output clusters.

```
res = external_validation(pr$cluster_labels,data.na$Production,
                          method = "adjusted_rand_index", summary_stats = T)
```

```
> res = external_validation(pr$cluster_labels,data$Production,
+
+                           method = "adjusted_rand_index", summary_stats = T)

---------------------------------------
purity                          : 0.6991
entropy                         : 7.4784
normalized mutual information   : 0.0921
variation of information        : 12.6212
normalized var. of information  : 0.9517
---------------------------------------
specificity                     : 0.9987
sensitivity                     : 0.0025
precision                       : 0.5585
recall                          : 0.0025
F-measure                       : 0.005
---------------------------------------
accuracy OR rand-index          : 0.5952
adjusted-rand-index             : 0.0014
jaccard-index                   : 0.0025
fowlkes-mallows-index           : 0.0373
mirkin-metric                   : 23775866182
---------------------------------------
```

F*igure* 4.33: External validation for Predicted clusters

- we can check the accuracy of the predicted clusters in the above figure

  ➡ accuracy : 0.5952 (59.52%)

CHAPTER 5 – CONCLUSION

5.1 Introduction

This chapter concludes the thesis with a summary of the overall work that has been carried out as well as the findings and observations obtained from the work. This also addresses ways in which agriculture can benefit from the use of data mining techniques . The performance of the classification algorithms implemented in r studio are shown.

5.2 Summary of Performance of Classification Models in R studio

Algorithms	Accuracy
K means	92.5%
Gaussian Mixture Modelling	56.85%

Table 5.1: Summary of Performance of Classification Models in R studio

- Overall, K means with classification of data in to the three clusters was the best performing classification model with an accuracy of 92.5%.
- Gaussian Mixture Modelling with classification of data into the three clusters was performed with an accuracy of 56.85%.

5.3 Summary of Performance of Predicting Classification Models in R studio

Algorithms	Accuracy
K means	66%
Gaussian Mixture Modeling	59.52%

Table 5.2: Summary of Performance of Predicting Classification Models in R studio

- Overall, K means for prediction of the classification of data was been performed with an accuracy of 66%.

- Gaussian Mixture Modeling for prediction of the classification of data was been performed with an accuracy of 59.52%.
- However, the comparison between the k means and Gaussian Mixture Modeling was performed successfully by the accuracy values for both classification and prediction of the classified clusters.
- Further, k means accuracy is high for both classification and prediction when compared with Gaussian Mixture Modeling.

5.4 Summary of Results and Conclusion

One large collection of dataset for clustering research was included in this study. This thesis introduces our research on the refinement of a large data set by extracting the garbage (incomplete binary files) from a large data set. The higher the precision of the classification method, the better it is to interact with the large data sets in our situation. We began the implementation of the k means algorithm by using the Elbow Method, which gives us an indication of what a reasonable k number of clusters will be cantered on the amount of the squared distance (SSE) between the data points and their allocated cluster centroids. In addition, to evaluate the optimal number of clusters, we use the Bayesian Information Criterion (BIC) when applying the GMM Algorithm.

As defined in section 1.6, this purpose of this research was to “to not only develop effective and efficient models to recognize low crop production states, but also to apply prediction techniques in a classified clusters to find top and least crop producing states.

To successfully achieve the objectives of this research, as defined in section 1.7, a publicly available agriculture crop dataset which contains 2,46,091 records with seven variables corresponding to state, district, year, season, crop, area, production of crop was

used. The data was been cleaned into a readable format and hence significant data pre-processing was necessary. R studio was used for implementing the data mining models. The results were almost similar in both the model. K-mean cluster and the Gaussian Mixture Model clustering were the two classification algorithms that were used to predict agriculture crop production and understand which of the states in the dataset were top and least crop producing groups. The results show that these algorithms are not successful in predicting crop dataset. The attributes that were used to show which algorithm are significant in predicting crop production are state name, crop year, production.

It was found that states with high production in the starting years are not maintained till the end of the years till 2015. The crop year attribute is also highly influential in determining top and low crop producing states. Some states are having high production in one single year. Extensive research on cluster formation algorithms and their application to data mining and agriculture have been performed in this study. Agriculture is demographically the main economic segment and plays a major role in India's overall socio-economic structure.

5.5 Future Work

The algorithm is currently working with the area and overall production as an operating parameter. The plan is to incorporate the other aspects that have an effect on the final result in the future Even, in the current algorithm, we have classified only the production and prediction of production values may not have been possible due to data being inaccessible. We will therefore try to gather the data and include that component in the model as well. This data mining algorithms was been applied on the production of states to begin with. However, it is also appropriate for certain states

or products. The only downside here is that we have no data collection available for many other states. In reality, the work has taken too long to come to a conclusion due to the lack of proper data collection. Our emphasis would therefore also be on building a proper study database and having it accessible and usable to the public. This will make it easier for any researcher who wants to contribute to this area to have access to data. In the future, we should be able to evaluate an efficient algorithm based on its accuracy criterion, which will help to pick an efficient algorithm for the classification of crop production in different states. Our existing data collection should be viewed as the foundation for potential research and any possible development should be taken on the basis of this structure. We intend to create an API that will make it easy for us to access the collected data. We aim to contribute to the development of the nation by carrying out and taking forward our research in the future. Our existing data set should be seen as the foundation for future study and any prospective contribution will be made on the basis of this structure.

References

1) Open data ecosystem as per National Data Sharing and Accessibility Policy (NDSAP) initiated Open Government Data (OGD) Platform,*https://data.gov.in/catalog/district-wise-season-wise-crop-production-statistics*.
2) Sekhar, C.C., UdayKumar, J., Kumar, B.K. and Sekhar, C., 2018. Effective use of Big Data Analytics in Crop planning to increase Agriculture Production in India.
3) Berkhin, P., 2006. A survey of clustering data mining techniques. In Grouping multidimensional data (pp. 25-71). Springer, Berlin, Heidelberg.
4) Jain, A.K., 2010. Data clustering: 50 years beyond K-means. Pattern recognition letters, 31(8), pp.651-666.
5) Shamir, O. and Tishby, N., 2010. Stability and model selection in k-means clustering. Machine learning, 80(2-3), pp.213-243.
6) Windarto, A.P. (2017). Implementation of Data Mining on Rice Imports by Major Country of Origin Using Algorithm Using K-Means Clustering Method. International Journal of Artificial Intelligence Research, 1(2), p.26.
7) Agricultural Recommender Using Data Mining Techniques by Mr.Omkar B. Bhalerao, Prof. L. M. R. J. Lobo Volume: 5 — Issue: 4 — April 2015 — ISSN - 2249-555X.
8) G. NasrinFathima and R. Geetha, “Agriculture Crop Pattern Using Data Mining Techniques”, International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE), ISSN: 2277 128X, vol. 4, no. 5, (2014) May, pp. 781-786.

9) Majumdar, J., Naraseeyappa, S. and Ankalaki, S. (2017). Analysis of agriculture data using data mining techniques: application of big data. Journal of Big Data, 4(1).

10) Jain, A.K., Murty, M.N. and Flynn, P.J., 1999. Data clustering: a review. *ACM computing surveys (CSUR)*, *31*(3), pp.264-323.

11) Dabbura, I., 2018. K-means Clustering: Algorithm, Applications, Evaluation Methods, and Drawbacks. Towards Data Science. Saatavissa: https://towardsdatascience. com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a. Hakupäivä, 7, p.2019.

12) Teknomo, K., 2006. K-means clustering tutorial. *Medicine*, *100*(4), p.3.

13) Kanungo, T., Mount, D.M., Netanyahu, N.S., Piatko, C.D., Silverman, R. and Wu, A.Y., 2002. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE transactions on pattern analysis and machine intelligence*, *24*(7), pp.881-892.

14) Yuan, C. and Yang, H., 2019. Research on K-value selection method of K-means clustering algorithm. *J—Multidisciplinary Scientific Journal*, *2*(2), pp.226-235.

15) Madhulatha, T.S., 2012. An overview on clustering methods. *arXiv preprint arXiv:1205.1117*.

16) Gursharan S, Harpreet K and Gursharan S 2014 A Novel Approach Towards K-Mean Clustering Algorithm With PSO *International Journal of Computer Science and Information Technologies (IJCSIT)* **5** 5978

17) rpubs.com. (n.d.). RPubs - Predictive Modeling - Cluster analysis. [online] Available at: https://rpubs.com/corey_sparks/539484 [Accessed 29 Apr. 2020].

18) J. M. Chambers, “Software for Data Analysis: Programming with R (Statistics and Computing)”, Springer, (2012).

19) Fraley, C. and Raftery, A.E., 1998. MCLUST: Software for model-based cluster and discriminant analysis. Department of Statistics, University of Washington: Technical Report, (342).

20) O. Shamir and N. Tishby, “Stability and model selection in k-means clustering,” *Mach. Learn.*, vol. 80, no. 2–3, pp. 213–243, 2010

21) Kuyuk, H.S., Yildirim, E., Dogan, E. and Horasan, G., 2012. Application of k-means and Gaussian mixture model for classification of seismic activities in Istanbul. Nonlinear Processes in Geophysics, 19(4).

22) Sanghyuk Chun (2014). K-means and GMM. [online] Available at: https://www.slideshare.net/sanghyukchun/kmeans-and-gmm [Accessed 29 Apr. 2020].

23) GeeksforGeeks. (2018). Gaussian Mixture Model. [online] Available at: https://www.geeksforgeeks.org/gaussian-mixture-model/ [Accessed 29 Apr. 2020].

24) Liu, S. and Barbu, A., 2018. Unsupervised Learning of Mixture Models with a Uniform Background Component. arXiv preprint arXiv:1804.02744.

25) Chambers, J., 2008. Software for data analysis: programming with R. Springer Science & Business Media.

26) Reynolds, D.A., 2009. Gaussian Mixture Models. Encyclopedia of biometrics, 741.

27) Steele, R.J. and Raftery, A.E., 2010. Performance of Bayesian model selection criteria for Gaussian mixture models. *Frontiers of statistical decision making and bayesian analysis*, *2*, pp.113-130.

28) Berkhin, P., 2006. A survey of clustering data mining techniques. In *Grouping multidimensional data* (pp. 25-71). Springer, Berlin, Heidelberg.

29) Zhao, Y., 2013. RDataMining. com: R and Data Mining. *RDataMining. com: R and Data Mining*.

30) Gupta, G.K., 2014. Introduction to data mining with case studies. PHI Learning Pvt. Ltd..

31) Han, J., Pei, J. and Kamber, M., 2011. *Data mining: concepts and techniques*. Elsevier.