VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“Jnana Sangama”, Belgaum – 590 018

A phase - II project report on

“Analysis and Prediction of Crimes by Clustering And Classification”

Submitted in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

INFORMATION SCIENCE & ENGINEERING

by

Deepak Koppal(1CR14IS032)

Madhushree.R (1CR15IS051)

Avesh Rawat (1CR15IS014)

Under the guidance of

Dr.Saravanan S
Associate Professor
Dept. of ISE, CMRIT, Bengaluru

CMR INSTITUTE OF TECHNOLOGY
DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING
#132, AECS Layout, IT Park Road, Bengaluru-560037

2019-2020

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“Jnana Sangama”, Belgaum – 590 018

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

Certificate

This is to certify that the project entitled, **“ANALYSIS AND PREDICTION OF CRIMES BY CLUSTERING AND CLASSIFICATION”**, is a bonafide work carried out by **Deepak Koppal. (1CR14IS032), Madhushree.R (1CR15IS051) and Avesh Rawat (1CR15IS014)** in partial fulfillment of the award of the degree of Bachelor of Engineering in Information Science & Engineering of Visvesvaraya Technological University, Belgaum, during the year 2019-20. It is certified that all corrections/suggestions indicated during reviews have been incorporated in the report. The project report satisfies the academic requirements in respect of the Phase II project work prescribed for the said Degree.

Name & Signature of Guide
Dr. Saravanan S
Assoc. Professor
Dept. of ISE, CMRIT

Name & Signature of HOD
Dr. M. Farida Begum
Professor and HOD
Dept. of ISE,CMRIT

External Viva

Name of the examiner **Signature with date**

1.

2.

CMR INSTITUTE OF TECHNOLOGY
BANGALORE-560037

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

Declaration

We, **Deepak Koppal (1CR14IS032), Madhushree.R (1CR15S051), Avesh Rawat (1CR15IS014),** bonafide students of CMR Institute of Technology, Bangalore, hereby declare that the dissertation entitled, **"Analysis And Prediction of Crimes By Clustering And Classification "** has been carried out by us under the guidance of **Dr. Saravanan S**, Associate Professor, CMRIT, Bangalore, in partial fulfillment of the requirements for the award of the degree of Bachelor of Engineering in Information Science Engineering, of the Visvesvaraya Technological University, Belgaum during the academic year 2019-2020. The work done in this dissertation report is original and it has not been submitted for any other degree in any university.

Deepak Koppal(1CR14IS032)

Madhushree.R(1CR15IS051)

Avesh rawat (1CR15IS014)

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany a successful completion of any task would be incomplete without the mention of people who made it possible. Success is the epitome of hard work and perseverance, but steadfast of all is encouraging guidance.

So with gratitude that we acknowledge all those whose guidance and encouragement served as beacon of light and crowned our effort with success.

We would like to thank Dr. Sanjay Jain, Principal, CMRIT, Bangalore for providing excellent academic environment in the college and his never-ending support for the B.E program.

We would also like to thank Dr.M. Farida Begam, Associate Professor and HOD, Department of Information Science and Engineering, CMRIT, Bangalore who provided guidance and gave valuable suggestions regarding the project.

We consider it a privilege and honor to express my sincere gratitude to my internal guide Dr.Saravanan S Associate Professor, Information Science and Engineering, CMRIT, Bangalore for their valuable guidance throughout the tenure of this project work.

We would also like to thank all the faculty members who have always been very co-operative and generous. Conclusively, we also thank all the non-teaching staff and all others who have done immense help directly or indirectly during my project

Deepak Koppal(1CR14IS032)

Madhushree.R(1CR15IS051)

Avesh rawat(1CR15IS014))

ABSTRACT

Crimes will somehow influence organizations and institutions when occurred frequently in a society. Thus, it seems necessary to study reasons, factors and relations between occurrence of different crimes and finding the most appropriate ways to control and avoid more crimes. The main objective of this report is to classify clustered crimes based on occurrence frequency during different years. Data mining is used extensively in terms of analysis, investigation and discovery of patterns for occurrence of different crimes. We applied a theoretical model based on data mining techniques such as clustering and classification to real crime dataset recorded by police in England and Wales and within 1990 to 2011. We assigned weights to the features in order to improve the quality of the model and remove low value of them. Today, collection and analysis of crime-related data are imperative to security agencies. The use of a coherent method to classify these data based on the rate and location of occurrence, detection of the hidden pattern among the committed crimes at different times, and prediction of their future relationship are the most important aspects that have to be addressed.

Keywords: Crime, Clustering, Classification, Genetic Algorithm, Weighting, Rapid Miner tool

Contents

Title **Page No**

List of Figures

Chapter 1

PREAMBLE

1.1 Introduction

There are mainly 5 sub categories where we are going to implement our project.
They are as follows.

A. Crime Analysis

Today, collection and analysis of crime-related data are imperative to security agencies. The use of a coherent method to classify these data based on the rate and location of occurrence, detection of the hidden pattern among the committed crimes at different times, and prediction of their future relationship are the most important aspects that have to be addressed. In this regard, the use of real datasets and presentation of a suitable framework that does not be affected by outliers should be considered. Pre-processing is an important phase in data mining in which the results are significantly affected by outliers. Thus, the outlier data should be detected and eliminated though a suitable method. Optimization of Outlier Detection operator parameters through the GA and definition of a Fitness function are both based on Accuracy and Classification error. The weighting method was used to eliminate low-value features because such data reduce the quality of data clustering and classification and, consequently, reduce the prediction accuracy and increase the classification error.

The main purposes of crime analysis are mentioned below:

- Extraction of crime patterns by crime analysis and based on available criminal information,
- Prediction of crimes based on spatial distribution of existing data and prediction of crime frequency using various data mining techniques

B. Clustering

Division of a set of data or objects to a number of clusters is called clustering. Thereby, a cluster is composed of a set of similar data which behave same as a group.

It can be said that the clustering is equal to the classification, with only difference that the classes are not defined and determined in advance, and grouping of the data is done without supervision.

c. Clustering by k-means algorithm

K-means is the simplest and most commonly used partitioning algorithm among the clustering algorithms in scientific and industrial software. Acceptance of the K-means is mainly due to its being simple. This algorithm is also suitable for clustering of the large datasets since it has much less computational complexity, though this complexity grows linearly by increasing of the data points. Beside simplicity of this technique, it however suffers from some disadvantages such as determination of the number of clusters by user, affectability from outlier data, high-dimensional data, and sensitivity toward centers for initial clusters and thus possibility of being trapped into local minimum may reduce efficiency of the K-means algorithm.

D. Classification

Classification is one of the important features of data mining as a technique for modeling of forecasts. In other words, classification is the process of dividing the data to some groups that can act either dependently or independently. Classification is used to make some examples of hidden and future decisions on the basis of the previous decision makings. Decision tree learning, neural network, nearest neighborhood, Nave Bayes method and support vector machine are different algorithms which are used for the purpose of classification.

1.2 Existing System

- The k-means clustering technique for extracting useful information from the crime dataset using Rapid Miner tool because it is solid and complete package with flexible support options.
- Linear regression for prediction the occurrence of crimes in Delhi (India).
- Prep Search has proposed.
- Intelligent criminal identification system called ICIS.
- An improved method of classification algorithms for crime prediction.
- Crime analysis and prediction using data mining.

1.2.1 Drawbacks

1. Although the coder designs an optimized code, the code readability is very inadequate and cannot be modified by the user even if required. Any change to the code requires change in the model itself.

1.3 Problem Statement

Extraction of crime patterns by crime analysis and based on available criminal information, prediction of crimes based on spatial distribution of existing data and prediction of crime frequency using various data mining techniques.

1.4 Plan of Implementation

Different algorithms will be used to come up with a good result in this contest. Each of them will be explained, tried and tested, and finally we will get to see which of them works best for this case. Cross-validation will be used to validate the models, so the database has to be split into test, train and validation subsets. This split has to be stratified to ensure that the initial proportion of elements (same amount of crimes per category) is maintained in each subdivision. The resulting train dataset is still too large (approx. 700.000), and running the testing programs would take too long. To speed up tests and development, we will reduce the database to approx. 8.000 records using a clustering algorithm. This algorithm will be K-Means. Then, having the 20 number of elements per cluster, we will be able to decide which element has more weight inside the algorithm. Technically there is no data loss. Once the data has been treated, the following algorithms will be tried (in order of complexity):

- K-Nearest Neighbors.
- Neural Networks.
- Confusion matrix.

Each of them will be deeply explained in its chapter later on. All the development and testing was done on a server lent by a university department. This way, executions could last all day without having to worry about them and were a little bit faster

Chapter 2

LITERATURE SURVEY

J. Agarwal, R. Nagpal and R. Sehgal [1] have analyzed crime and considered homicide crime taking into account the corresponding year. They have used the k-means clustering technique for extracting useful information from the crime dataset using Rapid Miner tool because it is solid and complete package with flexible support options. Fig1 shows the proposed system architecture. They review a dataset of the last 59 years to predict occurrence of some crimes including murder, burglary, robbery and etc. Their work will be helpful for the local police stations in decision making and crime supervision. After training systems will predict data values for next coming fifteen years. The system is trained by applying linear regression over previous year data. This will produce a formula and squared correlation The formula is used to predict values for coming future years. The coefficient of determination is useful because its gives the proportion of variance of one variable that is predictable from other variable. Fig2 shows the proposed system architecture.

An integrated system called Prep Search has proposed by L. Ding et al [8] It has been combined using two separate categories of visualization tools: providing the geographic view of crimes and visualization ability for social networks. ―It will take a given description of a crime including its location, type, and the physical description of suspects (personal characteristics) as input. To detect suspects, the system will process these inputs through four integrated components: geographic profiling, social network analysis, crime patterns and physical matching. Fig3 shows the system design and process of Prep Search. Increasing the efficiency in identifying possible suspects. In order to describe the system ICIS is divided to user interface, managed bean, multi agent system and database. Oracle Database is used for implementing of database, and identification of crime patterns has been implemented using Java platform.

In an improved method of classification algorithms for crime prediction has proposed by A. Babakura, N. Sulaiman and M. Yusuf. They have compared Naïve Bayesian and Back Propagation (BP) classification algorithms for predicting crime category for distinctive state in USA. In the first step phase, the model is built on the training and in the second phase the model is applied. The performance measurements such as Accuracy, Precision and Recall are used for comparing of the classification algorithms. The precision and recall remain the same when BP is used as a classifier.

In researches have introduced intelligent criminal identification system called ICIS which can potentially distinguish a criminal in accordance with the observations collected from the crime location for a certain class of crimes. The system uses existing evidences in situations for identifying a criminal by clustering mechanism to segment crime data in to subsets, and the Nave Bayesian classification has used for identifying possible suspect of crime incidents. ICIS has been used the communication power of multi agent system for

In researches have introduced crime analysis and prediction using data mining. They have proposed an approach between computer science and criminal justice to develop a data mining procedure that can help solve crimes faster. Also they have focused on causes of crime occurrence like criminal background of offender, political, enmity and crime factors of each day. Their method steps are data collection, classification, pattern identification, prediction and visualization.

Chapter 3

THEORITICAL BACKGROUND

Theoretical background highlighting some topics related to the project work is given below. The description contains several topics which are worth to discuss and also highlight some of their limitation that encourage going on finding solution as well as highlights some of their advantages for which reason these topics and their features are used in this project.

3.1 Result submission format and evaluation

Data submitted to the contest for its evaluation has to have a specific format to fulfill the requirements. To allow data to be evaluated correctly, the resulting dataset must contain the sample ID with a list of all categories and the probability of each sample to belong to each one of them. Remind that the training dataset is labeled with all sample's crime types (there are 39 different).

Submission format is in the below link:

https://drive.google.com/file/d/1ylPD-wCW0CpJuZRwF4XL2Xw5jNkw2-Z8/view?usp

sharing Then, instead of predicting to which category the given sample may belong, the output will always be probability vectors. Submissions are evaluated using the multi-class logarithmic loss function, which has the following formula (provided by Kaggle):

$logloss = -1\ /N\ \sum\sum yij \log(pij)$ where M, j=1 and N, i=1 21

Where N is the number of cases in the test set, M is the number of class labels, log is the natural Logarithm, yij is 1 if observation i is in class j and 0 otherwise, and pij is the predicted probability that observation i belongs to class j.

Using this function, which outputs a number, results can be compared and a measure of how good the results have been can be obtained. Also, it is useful for algorithms' parameters validation before making a final submission.

3.2 Data Set Analysis

The provided dataset has different “features”, each one being of a different relevance. In this chapter we will precede to analyses this database and extract the useful information out of it.

The record contains data sets of:

- Year- In which year which crime was held,
- Category- the different category of crimes.
- Descript- a short description of the crime.
- PD District: the district of the city where the crime was committed**.**
- Resolution: Short description of the resolution

3.2.1Data Set Selection

Data is the most import part when you work on prediction systems. It plays a very vital role your whole project i.e., you system depends on that data. So selection of data is the first and the critical step which should be performed properly, For our project we got the data from the government website. These data sets were available for all. There are other tons of websites who provide such data. The data set we choose was selected based on the various factors and constraints we were going to take under the consideration for our prediction system.

3.2.2 Data Cleaning

After we have selected the data set. The next step is to clean the data and transform it into the desired format as it is possible the data set we use may be of different format. It is also possible that we may use multiple data sets from different sources which may be in different file formats. So to use them we need to convert them into the format we want to or the type that type prediction system supports. The reason behind this step is that it is possible that the data set contains the constraints which are not needed by the prediction system and including them makes the system complicated and may extend the processing time. Another reason behind data cleaning is the data set may contain null value and garbage values too. So the solution to this issue is when the data is transformed the garbage values are replaced. There are many methods to perform that.

3.2.3 Data Processing

After the data has been cleaned and transformed it is ready to process further. After the data has been cleaned and we have taken the required constraints. We divide the whole data set into o the two parts that can be either 70-30 or 80-20. The larger portion of the data is for the processing. The data set obtained will now be subjected to various data mining techniques.

Data Preprocessing is a technique that is used to convert the raw data into a clean data set. In other words, whenever the data is gathered from different sources it is collected in raw format which is not feasible for the analysis.

- Importing the Dataset
- Importing the required Libraries
- Handling the Missing Data Encoding categorical data
- Splitting the Data set into Training set and Test Set Feature Scaling

Training:

- The idea of using training data in machine learning programs is a simple concept, but it is also very foundational to the way that these technologies work. The training data is an initial set of data used to help a program understand how to apply technologies like neural networks to learn and produce sophisticated results.

Testing:

- A test dataset is a dataset that is independent of the training dataset, but that follows the same probability distribution as the training dataset. If a model fit to the training dataset also fits the test dataset well, minimal over fitting has taken place.

3.2.4 Data flow diagrams

A data-flow diagram (DFD) is a way of representing a flow of a data of a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself.

DFD level -1

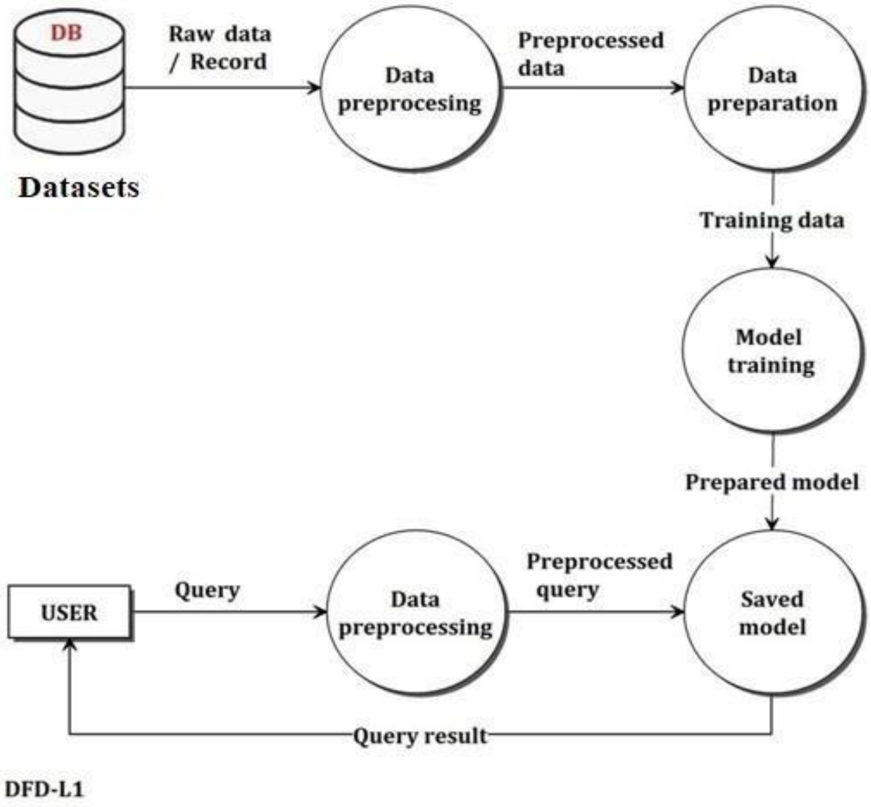


Figure 3.1: DFD level 1

DFD level -2

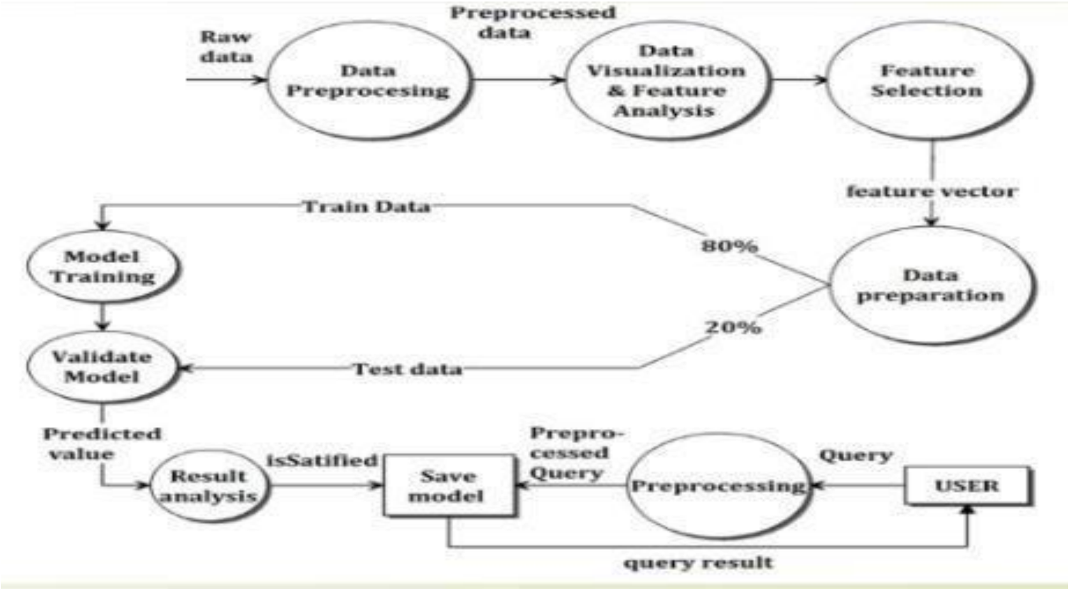


Fig:3.2 DFD level 2

3.2.5 Clustering

Clustering will be performed on the given data set. The main aim of performing clustering is to divide the data into different clusters or groups such that the objects within a group are similar to each other whereas objects in other clusters are different from each other. There are several clustering algorithms available: Hierarchical clustering technique like Ward method, single linkage, complete linkage etc, K means and latest class clustering (LCC).Other clustering algorithms like K-modes clustering is an enhanced version of K means clustering.The clusters are then subjected to other algorithms like Association rule mining and trend analysis.

3.3 Data Treatment

In the previous chapter it was explained how the dataset is distributed, and some useful information was extracted out of it. Now, having all this in mind, the useful information identification and transformation (if necessary) into a “computer-understandable” formatso it can be worked with, will be carried.

Also, as cross-validation is going to be used to validate some parameters, the dataset will be split into train, test and validation subsets. Even so, the train set will still be very big (nearly 650.000 samples) and this would slow down the tests. To make them faster, a clustering algorithm will be applied to reduce the dataset size.

3.3.1 Data Transformation

To differentiate between samples and to be able to classify them, they must be comparable in some way. They must be in a format which allows the algorithm to put one against another and see which one fits each classification.

3.3.2 Data Split

Results from the executions of the algorithms need to be evaluated to see which ones are better and also to see if the parameters’ values used are adequate. As only one labeled dataset is provided, it has to be split it into 3 smaller parts to train, test and validate the algorithms For this purpose, then, the main (train) database will be split into (as said previously) train, test and validate subsets. This division will be done in the following proportions: 80% train, 10% test and 10% validate.

With this reduced training database, which is also labeled, the algorithm will be shown which are the correct results and start making it learn to classify. With the test one, the accuracy of the algorithm will be tested, to see if any parameter adjustments have to be made or compare it with the other ones. Finally, the validate subset allows to, rapidly, validate the value of certain parameters of the training algorithm in order to choose their correct value or the one that fits best.

3.3.3 Dataset Reduction

Remember that the initial training database had nearly 900.000 records, which is pretty high. Then, with the split, it was reduced to 87% its size (which is about 720.000 records). Still too big if the tests are meant to run fast. So, it is required to keep reducing the training dataset until it has a manageable size. Deleting all the "extra" samples that are not wanted would end up with a messed up database that wouldn't have the same elements per category proportion than the original one. To avoid this happening, the clustering algorithm used to reduce the database size will be applied for each category. This way, it is ensured that the resulting set will maintain the same proportions from the initial one.

A clustering algorithm groups elements in a specified number of clusters. This clusters' centres are calculated and they act as the representatives of all the samples in their group. Because not all clusters will have the same number of elements, some will be more relevant than others. Their relevance is measured by their weight, which is the number of elements they contain.

The algorithm used for this purpose has been K-means. K-means clustering proceeds by selecting k initial cluster centres and then, iteratively refining them as follows:

1. Each instance di is assigned to its closest cluster centre.
2. Each cluster centre Cj is updated to be the mean of its constituent instances.

The algorithm converges when there is no further change in assignment of instances of clusters, or if the maximum total number of iterations defined have been.

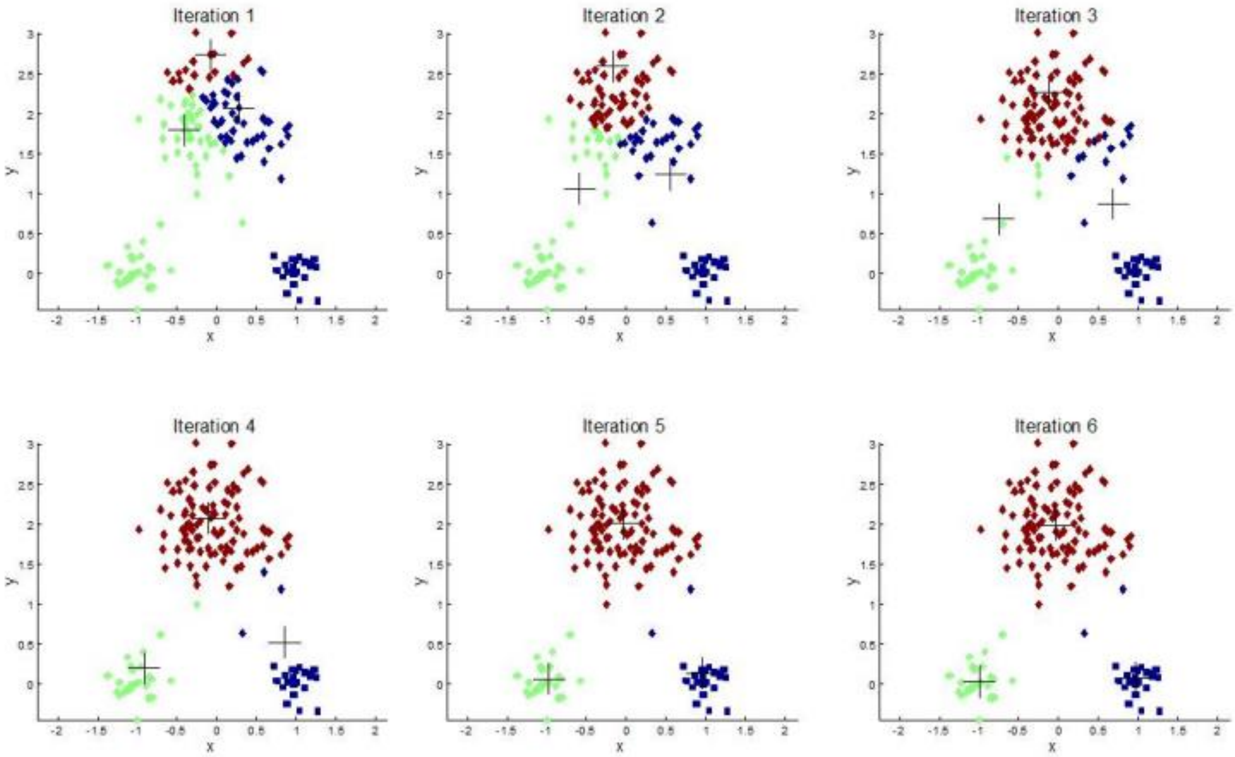


Fig . K-means Iterations

In this work, k-means algorithm has been applied for each category in the following way:

1. Get all the elements of a category

2. Obtain the number of centroids necessary to reduce the database to the number of elements that we want, bearing in mind that some classes might have very few members. 3. Apply K-means for this category.

4. Get the number of samples per cluster.

5. Add the resulting cluster centres with the number of elements they represent to the reduced database.

6. Follow up with the next category.

Note that this algorithm has to be applied after converting the values to the right format, otherwise it won’t be able to calculate the cluster centre properly

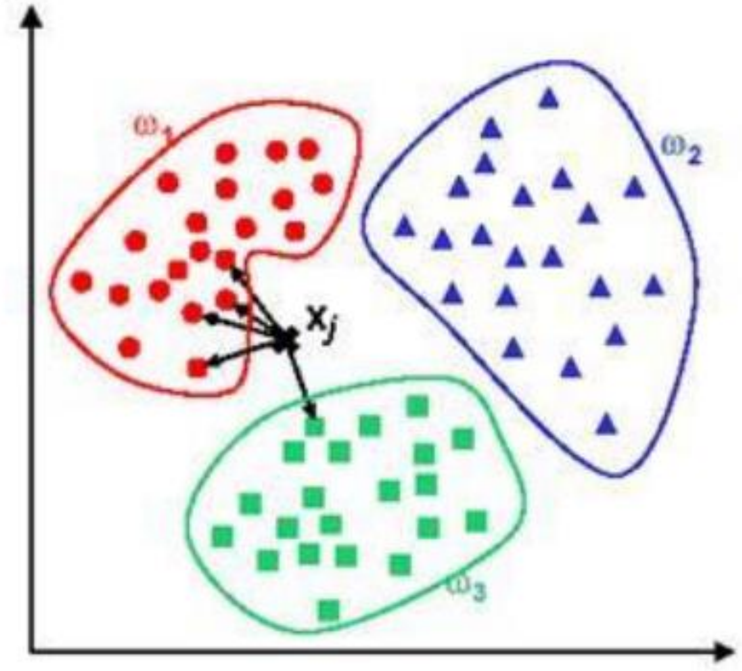
3.4 Machine learning algorithm

In this section we will explain which have been the algorithms that have been tried in order to solve this problem. Here, the theory and the implementation are discussed

3.4.1 k-nearest neighbor

The k-nearest neighbor classification rule is arguably the simplest and most intuitively appealing nonparametric classifier. It assigns a sample z to class X if the majority (i.e. more than ½) of the k values in the training dataset that are near z are from X, otherwise it assigns it to class Y.

This algorithm compares the given unclassified sample z with "all" (depending on the implementation, to speed up the execution, other comparison algorithms can be used so that it's not needed to compare it with every single value) the values in the training set and gets its k nearest values. Among them, it applies the previously described rule



Fig,K-nearestneighbor (k=5)

3.4.1.1 Implementation

Although it’s a pretty easy algorithm to implement from scratch, making it efficient and fast it’s not trivial. For this reason, Python scikit-learn library [20], which has a broad set of machine learning utilities, has been used. This library has a Nearest Neighbours package with all necessary functions. From this package, NearestNeighbors class has been used (concretely, fit and kneighbors methods). Because the results must be submitted giving a probability of each sample to belong to all categories, and not just the predicted category (as said in 3.1.2), the following formula was used to calculate this probabilities:

$$P(x \epsilon C_i) = \frac{\sum_{j \in C_i}(n_j \frac{1}{d_j})}{\sum_{l=1}^{k}(n_l \frac{1}{d_l})}$$

Where $(x \epsilon C\ )$ is the probability of the test sample to belong to category Ci , nj and nl are the number of elements represented by each train dataset sample, and dj and dl are the distance between test and the corresponding train sample

3.4.2 K-Means Clustering Algorithm

Kmeans algorithm is an iterative algorithm that tries to partition the dataset into *K*pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to only one group. It tries to make the intra-cluster data points as similar as possible while also keeping the clusters as different (far) as possible. It assigns data points to a cluster such that the sum of the squared distance between the data points and the cluster’s centroid (arithmetic mean of all the data points that belong to that cluster) is at the minimum. The less variation we have within clusters, the more homogeneous (similar) the data points are within the same cluster.

The way kmeans algorithm works is as follows:

1. Specify number of clusters *K*.
2. Initialize centroids by first shuffling the dataset and then randomly selecting *K* data points for the centroids without replacement.

3. Keep iterating until there is no change to the centroids. i.e assignment of data points to clusters isn't changing.
4. Compute the sum of the squared distance between data points and all centroids.
5. Assign each data point to the closest cluster (centroid).
6. Compute the centroids for the clusters by taking the average of the all data points that belong to each cluster.

The approach kmeans follows to solve the problem is called Expectation-Maximization. The E-step is assigning the data points to the closest cluster. The M-step is computing the centroid of each cluster. Below is a breakdown of how we can solve it mathematically (feel free to skip it).

The objective function is:

$$J = \sum_{i=1}^{m} \sum_{k=1}^{K} w_{ik} \|x^i - \mu_k\|^2 \qquad (1)$$

where wik=1 for data point xi if it belongs to cluster *k*; otherwise, wik=0. Also, μk is the centroid of xi's cluster.

It's a minimization problem of two parts. We first minimize J w.r.t. wik and treat μk fixed. Then we minimize J w.r.t. μk and treat wik fixed. Technically speaking, we differentiate J w.r.t. wik first and update cluster assignments (*E-step*). Then we differentiate J w.r.t. μk and recompute the centroids after the cluster assignments from previous step (*M-step*). Therefore, E-step is:

$$\frac{\partial J}{\partial w_{ik}} = \sum_{i=1}^{m} \sum_{k=1}^{K} \|x^i - \mu_k\|^2$$

$$\Rightarrow w_{ik} = \begin{cases} 1 & \text{if } k = argmin_j \|x^i - \mu_j\|^2 \\ 0 & \text{otherwise.} \end{cases} \qquad (2)$$

In other words, assign the data point xi to the closest cluster judged by its sum of squared distance from cluster's centroid And M-step is:

$$\frac{\partial J}{\partial \mu_k} = 2\sum_{i=1}^{m} w_{ik}(x^i - \mu_k) = 0$$

$$\Rightarrow \mu_k = \frac{\sum_{i=1}^{m} w_{ik}x^i}{\sum_{i=1}^{m} w_{ik}} \qquad (3)$$

3.4.3 Naïve Bayes Classifier

Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

Bayes' Theorem finds the probability of an event occurring given the probability of another event that has already occurred. Bayes' theorem is stated mathematically as the following equation:

$$P(A|B) = \frac{P(B|A)\ P(A)}{P(B)}$$

where A and B are events and P(B) ? 0.

- Basically, we are trying to find probability of event A, given the event B is true. Event B is also termed as evidence.
- P(A) is the priori of A (the prior probability, i.e. Probability of event before evidence is seen). The evidence is an attribute value of an unknown instance(here, it is event B).
- P(A|B) is a posteriori probability of B, i.e. probability of event after evidence is seen.

3.4.4 Confusion matrix

In the field of machine learning and specifically the problem of statistical classification, a confusion matrix, also known as an error matrix. A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known. It allows the visualization of the performance of an algorithm.It allows easy identification of confusion between classes e.g. one class is commonly mislabeled as the other. Most performance measures are computed from the confusion matrix.

A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key to the confusion matrix. The confusion matrix shows the ways in which your classification model is confused when it makes predictions. It gives us insight not only into the errors being made by a classifier but more importantly the types of errors that are being made.

Definition of the Terms:

- Positive (P) : Observation is positive (for example: is an apple).
- Negative (N) : Observation is not positive (for example: is not an apple).
- True Positive (TP) : Observation is positive, and is predicted to be positive.
- False Negative (FN) : Observation is positive, but is predicted negative.
- True Negative (TN) : Observation is negative, and is predicted to be negative.
- False Positive (FP) : Observation is negative, but is predicted positive.

Classification Rate or Accuracy is given by the relation.

$$\textbf{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

However, there are problems with accuracy. It assumes equal costs for both kinds of errors. A 99% accuracy can be excellent, good, mediocre, poor or terrible depending upon the problem

Recall:

$$\textbf{\textit{Recall}} = \frac{TP}{TP + FN}$$

Recall can be defined as the ratio of the total number of correctly classified positive examples divide to the total number of positive examples. High Recall indicates the class is correctly recognized (a small number of FN).

Precision:

$$\textbf{\textit{Precision}} = \frac{TP}{TP + FP}$$

To get the value of precision we divide the total number of correctly classified positive examples by the total number of predicted positive examples. High Precision indicates an example labelled as positive is indeed positive (a small number of FP).

Chapter 4

SYSTEM REQUIREMENT SPECIFICATION

A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and nonfunctional requirements, and may include a set of use cases that describe user interactions that the software must provide. In order to fully understand ones project, it is very important that they come up with an SRS listing out their requirements, how are they going to meet it and how will they complete the project. SRS also functions as a blueprint for completing a project with as little cost growth as possible. SRS is often referred to as the parent document because all subsequent project management documents, such as design specifications, statements of work, software architecture specifications, testing and validation plans, and documentation plans, are related to it. Requirement is a condition or capability to which the system must conform. Requirement Management is a systematic approach towards eliciting, organizing and documenting the requirements of the system clearly along with the applicable attributes. The elusive difficulties of requirements are not always obvious and can come from any number of sources.

4.1 Functional Requirements

Functional Requirement defines a function of a software system and how the system must behave when presented with specific inputs or conditions. These may include calculations, data manipulation and processing and other specific functionality. Following are the functional requirements on the system:

- Collecting Date sets and data pre-processing is performed for that data set
- The date set will be subjected to various date mining techniques ,Clustering will be performed on the given data set
- The clusters are then subjected to other algorithms like Association rule mining and trend analysis.

4.2 Non Functional Requirements

Nonfunctional requirements are the requirements which are not directly concerned with the specific function delivered by the system. They specify the criteria that can be used to judge the operation of a system rather than specific behaviors. They may relate to emergent system properties such as reliability, response time and store occupancy. Nonfunctional requirements arise through the user needs, because of budget constraints, organizational policies and the need for interoperability with other software and hardware systems.

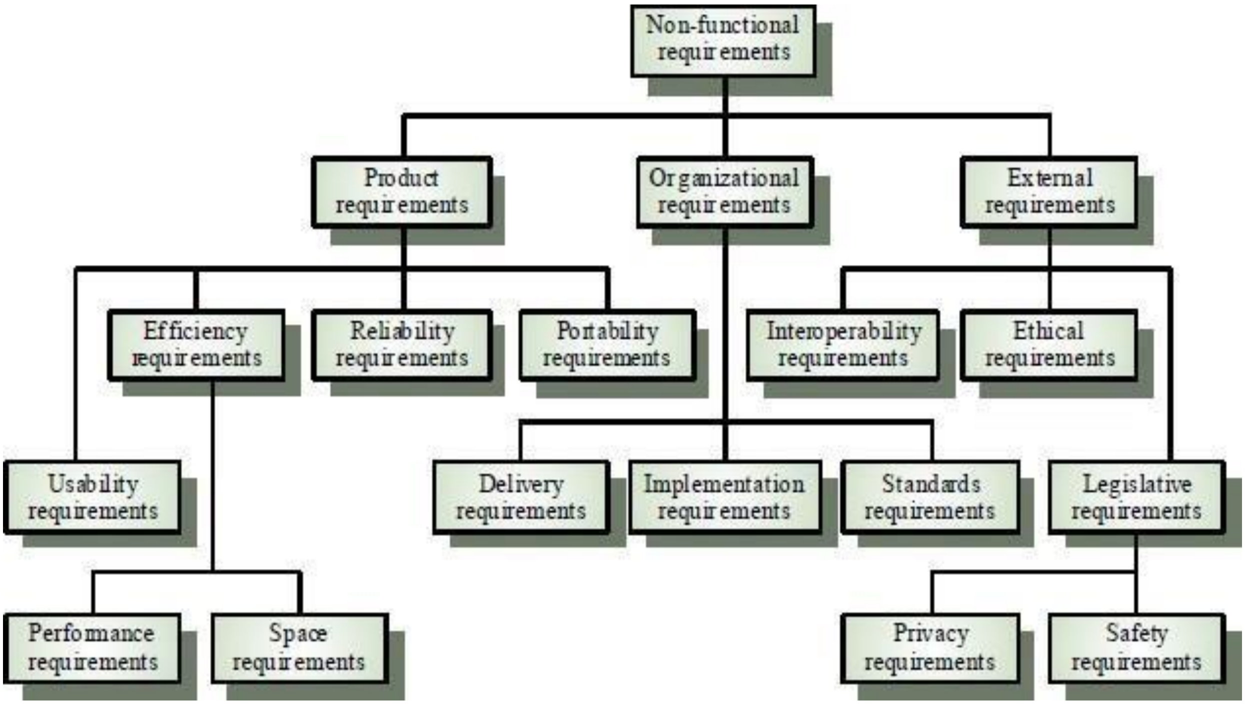


Figure 4.1 : Nonfunctional requirements

Some Non-Functional Requirements are as follows:

4.2.1 Reliability

The structure must be reliable and strong in giving the functionalities. The movements must be made unmistakable by the structure when a customer has revealed a couple of enhancements. The progressions made by the Programmer must be Project pioneer and in addition the Test designer.

4.2.2 Maintainability

The system watching and upkeep should be fundamental and focus in its approach. There should not be an excess of occupations running on diverse machines such that it gets hard to screen whether the employments are running without lapses.

4.2.3 Performance

The framework will be utilized by numerous representatives all the while. Since the system will be encouraged on a single web server with a lone database server outside of anyone's ability to see, execution transforms into a significant concern. The structure should not capitulate when various customers would use everything the while. It should allow brisk accessibility to each and every piece of its customers. For instance, if two test specialists are all the while attempting to report the vicinity of a bug, then there ought not to be any irregularity at the same time.

4.2.4 Portability

The framework should to be effectively versatile to another framework. This is obliged when the web server, which s facilitating the framework gets adhered because of a few issues, which requires the framework to be taken to another framework.

4.2.5 Scalability

The framework should be sufficiently adaptable to include new functionalities at a later stage. There should be a run of the mill channel, which can oblige the new functionalities.

4.2.6 Flexibility

Flexibility is the capacity of a framework to adjust to changing situations and circumstances, and to adapt to changes to business approaches and rules. An adaptable framework is one that is anything but difficult to reconfigure or adjust because of diverse client and framework prerequisites. The deliberate division of concerns between the trough and motor parts helps adaptability as just a little bit of the framework is influenced when strategies or principles change.

4.3 Product Requirements

Correctness:

It followed a well-defined set of procedures and rules to engage a conversation with the user and a pre-trained classification model to compute also rigorous testing is performed to confirm the correctness of the data.

Modularity:

The complete product is broken up into many modules and well defined interfaces are developed to explore the benefit of flexibility of the product.

Robustness:

This software is being developed in such a way that the overall performance is optimized and the user can expect the results within a limited time with utmost relevance and correctness. Nonfunctional requirements are also called the qualities of a system

4.4 Basic Operational Requirements:

The customers are those that perform the eight primary functions of systems engineering, with special emphasis on the operator as the key customer. Operational requirements will define the basic need and, at a minimum, will be related to these following points:-

Mission profile or scenario: It describes about the procedures used to accomplish mission objective. It also finds out the effectiveness or efficiency of the system.

Performance and related parameters: It points out the critical system parameters to accomplish the mission.

Utilization environments: It gives a brief outline of system usage. Finds out appropriate environments for effective system operation.

Operational life cycle: It defines the system lifetime.

4.5 Hardware system configuration

- Processor : Pentium Processor and Above
- Ram : 4GB
- Hard disk capacity : 500 GB

4.6 Software system configuration

- Operating system : Linux, Windows 7,8,10
- Programming Language : Python 3
- Framework :Anaconda
- IDE : Spider.
- DL Libraries : Numpy, Pandas/.

Chapter 5

SYSTEM DESIGN

5.1 System DevelopmentMethodology

System Development methodology is the development of a system or method for unique situation. Having a proper methodology helps us in bridging the gap between the problem statement and turning it into a feasible solution. It is usually marked by converting the System Requirements Specifications (SRS) into a real world solution.

System design takes the following inputs: Statement of work. Requirement determination plan. Current situation analysis. Proposed system requirements including a conceptual data model and metadata (data about data).The development method followed in this project is waterfall model.

5.2 Model Phases

The waterfall model is a sequential software development process, in which progress is seen as flowing steadily downwards (like a waterfall) through the phases of Requirement initiation, Analysis, Design, Implementation, Testing and maintenance.

Requirement Analysis: This phase is concerned about collection of requirement of the system. This process involves generating document and requirement review.

System Design: Keeping the requirements in mind the system specifications are translated in to a software representation. In this phase the designer emphasizes on: algorithm, data structure, software architecture etc.

Coding: In this phase programmer starts his coding in order to give a full sketch of product. In other words system specifications are only converted in to machine readable compute code

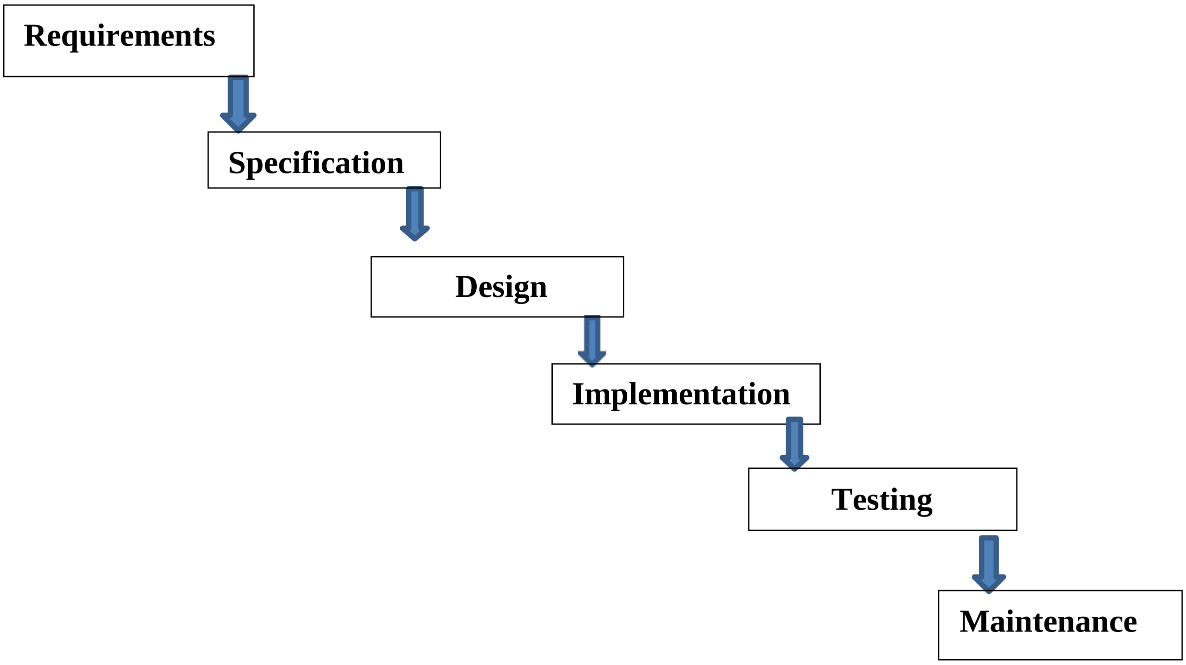


Figure 5.1: Waterfall Model

Implementation: The implementation phase involves the actual coding or programming of the software. The output of this phase is typically the library, executables, user manuals and additional software documentation

Testing: In this phase all programs (models) are integrated and tested to ensure that the complete system meets the software requirements. The testing is concerned with verification and validation.

Maintenance: The maintenance phase is the longest phase in which the software is updated to fulfill the changing customer need, adapt to accommodate change in the external environment, correct errors and oversights previously undetected in the testing phase, enhance the efficiency of the software.

5.3 Advantages of Waterfall model

- Clear project objective
- Stable project requirements
- Progress of system is measurable.
- Logic of software development is clearly understood.
- Better resource allocation.

5.4 System Architecture

The system architecture shown in Figure 5.2 gives overall view about all the modules in the proposed system of Crimes and the flow of the process right from data collection to detection.

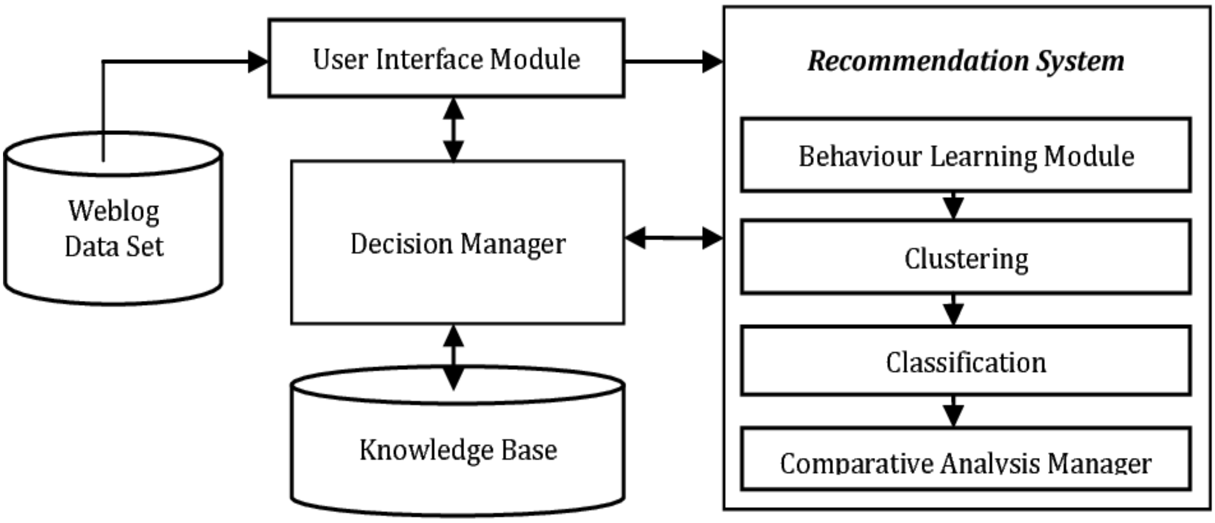


FIGURE 4.2: System Architecture for Crime Analysis

- The criminals can hold certain properties and their crimes characteristics and crime careers may vary from one criminal to another. Such type of information can be taken as input dataset.
- The input dataset is given to a pre-processor which performs the pre-processing based on the requirements
- Once the preprocessing is completed the features or attributes from those information are extracted which may be in the form of text content from emails, the crime factors for day, criminal characteristics, geo-location of the criminals etc..,
- The preprocessed results is further given to the classification algorithm or the clustering algorithm based on the requirements
- The requirements may be anything from selecting the crime prone areas to predicting the criminal based on the previous crime records.
- The classification algorithm works in a supervised learning manner in which the training and testing phase is required in order to train the classifier to identify the new unknown crime record.
- The clustering algorithm works in an unsupervised learning manner which automatically separates the crime records based on the number of groups.

CHAPTER 6

TESTING

6.1 Software testing introduction

Software testing is a process used to help identify the correctness, completeness and quality of developed computer software. Software testing is the process used to measure the quality of developed software .Testing is the process of executing a program with the intent of finding errors. Software testing is often referred to as verification & validation.

6.2 STLC (Software Testing Life Cycle):

Testing itself has many phases i.e. is called as STLC. STLC is part of SDLC

- Test Plan
- Test Development
- Test Execution
- Analyze Result
- Defect Tracking

6.2.1 Test Plan

It is a document which describes the testing environment, purpose, scope, objectives, test strategy, schedules, mile stones, testing tool, roles and responsibilities, risks, training, staffing and who is going to test the application, what type of tests should be performed and how it will track the defects.

6.2.2 Test Development

Preparing test cases, test data, Preparing test procedure, Preparing test scenario, Writing test script.

6.2.3Test execution

In this phase we execute the documents those are prepared in test development phase.

6.2.4 Analyze Result

Once executed documents will get results either pass or fail. We need to analyze the results during this phase.

6.2.5 Defect Tracking

Whenever we get defect on the application we need to prepare the bug report file and forwards to Test Team Lead and Dev Team. The Dev Team will fix the bug. Again we have to test the application. This cycle repeats till we get the software without defects.

6.3 TYPES OF TESTING:

White Box Testing

Black Box Testing

Grey box testing

6.3.1 White Box Testing

White box testing as the name suggests gives the internal view of the software. This type of testing is also known as structural testing or glass box testing as well, as the interest lies in what lies inside the box.

6.3.2 Black Box Testing

Its also called as behavioral testing. It focuses on the functional requirements of the software. Testing either functional or nonfunctional without reference to the internal structure of the component or system is called black box testing.

6.3.3 Grey Box Testing

Grey box testing is the combination of black box and white box testing. Intention of this testing is to find out defects related to bad design or bad implementation of the system.

6.4 LEVEL OF TESTING USED IN PROJECT

6.4.1 Unit testing

Initialization testing is the first level of dynamic testing and is the first responsibility of developers and then that of the test engineers.

6.4.2 Integration testing

All module which make application are tested . Integration testing is to make sure that the interaction of two or more components produces results that satisfy functional requirement.

6.4.3 System testing

To test the complete system in terms of functionality and non-functionality. It is black box testing, performed by the Test Team, and at the start of the system testing the complete system is configured in a controlled environment.

6.4.4 Functional testing

The outgoing links from all the pages from specific domain under test. Test all internal links. Test links jumping on the same pages. Check for the default values of fields. Wrong inputs to the fields in the forms.

6.4.5 Alpha testing

Alpha testing is final testing before the software is released to the general public. This testing is conducted at the developer site and in a controlled environment by the end user of the software.

6.4.6 Beta testing

The beta test is conducted at one or more customer sites by the end user of the software. The beta test is conducted at one or more customer sites by the end user of the software.

Chapter 7

IMPLEMENTATION

7.1 Reading The DataFrame

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import MinMaxScaler


from google.colab import drive
drive.mount('/content/drive')


data=pd.read_csv("/content/drive/My Drive/crime-1990-2011.csv",encoding = 'unicode_escape')
data.head(20)
```

7.2 Finding the number of rows and number of coloumns

```
data.shape

data.info()
```

7.3 Checking the Missing And NAN Values

```
data.isnull().sum()
```

7.4 Dropping the NAN Values into the Data

```
data1=data.dropna(axis=0,how="any")
data1.head()


data2=data1.drop(["Year","Force",'TOTAL OTHER MISCELLANEOUS OFFENCES','TOTAL ALL
OFFENCES'],axis=1)
data2.head()
```

7.5 Data Preprocessing

```
newdata=data2.replace('..',"0")
newdata.head()
```

```
newdata["Robbery of business property"]=newdata["Robbery of business property"].str.replace(",","")
newdata['Robbery of personal property']=newdata['Robbery of personal property'].str.replace(",","")
```

```
data3=newdata.apply(pd.to_numeric)
data3.head()
```

```
data3.shape
```

7.6 Data Normalization

```
from sklearn import preprocessing
data_normalize=preprocessing.normalize(data3)
data_normalize
```

```
data_normalize.shape
```

7.7 Elbow Method for Finding the Optimumu K Values

```
from scipy.spatial.distance import cdist
from sklearn.cluster import KMeans
distortions = []
K = range(1,7)
for k in K:
kmeanModel = KMeans(n_clusters=k).fit(data_normalize)
```

```
    kmeanModel.fit(data_normalize);
    distortions.append(sum(np.min(cdist(data_normalize, kmeanModel.cluster_centers_, 'euclidean'),
axis=1)) / data_normalize.shape[0])



# Plot the elbow
plt.plot(K, distortions, 'bx-')
plt.xlabel('k')
plt.ylabel('Distortion')
plt.title('The Elbow Method showing the optimal k')
plt.show()
```

7.8 Model Building of K-Means Clustering

```
kmeans = KMeans(n_clusters=6, random_state=0).fit(data_normalize)
labels=kmeans.labels_
labels

labels.shape
```

7.9 Split the data into training and testing

```
from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test=train_test_split(data_normalize,labels,test_size=0.2,random_state=1)
x_train

x_test
```

7.10 Model Building GaussianNB

```
from sklearn.naive_bayes import GaussianNB

gnb=GaussianNB()
gnb.fit(x_train,y_train)  # Train the model (Machine learning)
```

7.11 Model prediction of test data

```
y_pred=gnb.predict(x_test) #Prediction
y_pred

test_score=gnb.score(x_test,y_test)
test_score
```

7.12 Confusion Matrix

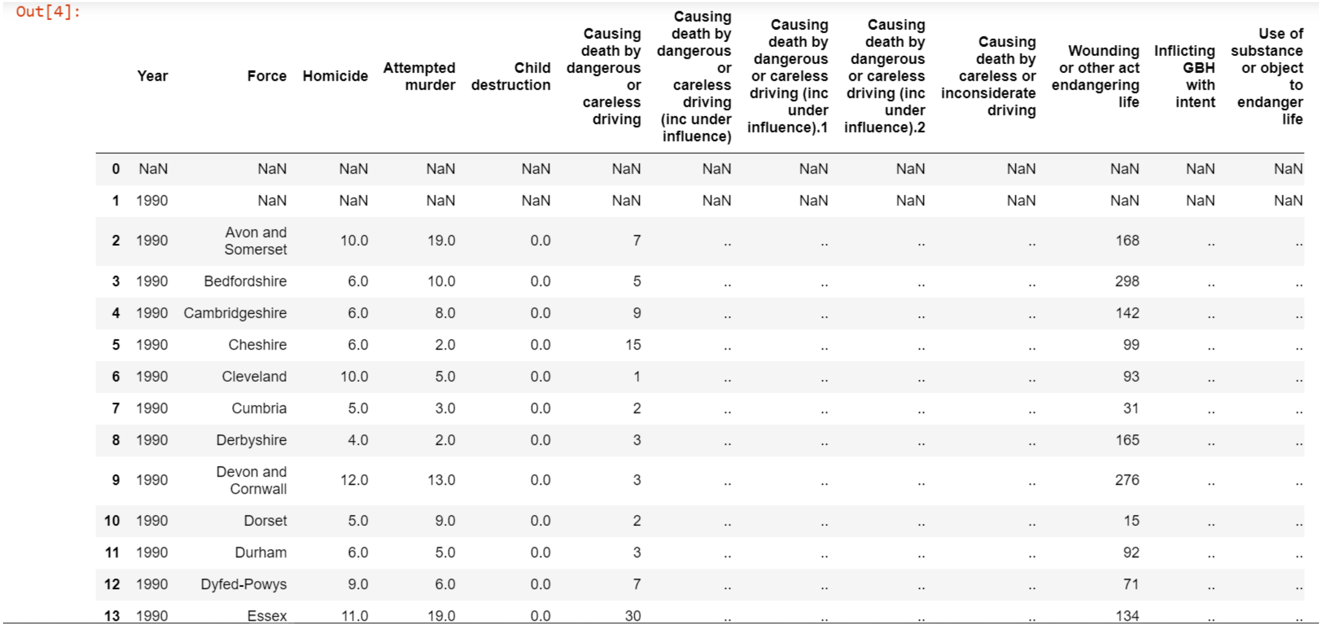
```
confusion_matrix=pd.crosstab(y_test,y_pred)
confusion_matrix

sns.heatmap(confusion_matrix, annot=True,fmt='d',cmap="YlGnBu")
plt.title("confusion matrix",fontsize=15)
plt.show()
```

Chapter 8

Screenshots

8.1 Reading the DataFrame

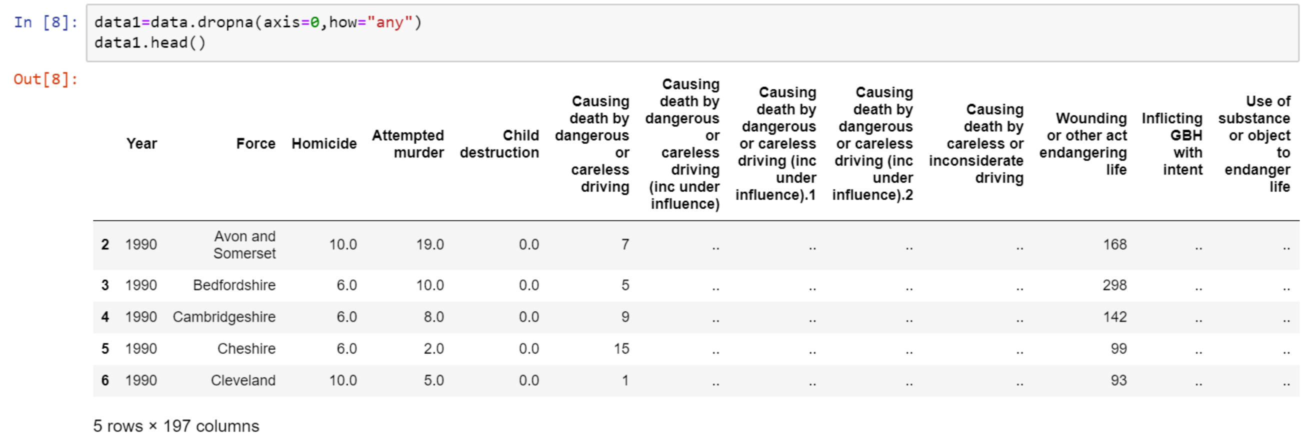


Out[4]:

	Year	Force	Homicide	Attempted murder	Child destruction	Causing death by dangerous or careless driving	Causing death by dangerous or careless driving (inc under influence)	Causing death by dangerous or careless driving (inc under influence).1	Causing death by dangerous or careless driving (inc under influence).2	Causing death by careless or inconsiderate driving	Wounding or other act endangering life	Inflicting GBH with intent	Use of substance or object to endanger life
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	1990	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	1990	Avon and Somerset	10.0	19.0	0.0	7	..	..	..	..	168	..	..
3	1990	Bedfordshire	6.0	10.0	0.0	5	..	..	..	..	298	..	..
4	1990	Cambridgeshire	6.0	8.0	0.0	9	..	..	..	..	142	..	..
5	1990	Cheshire	6.0	2.0	0.0	15	..	..	..	..	99	..	..
6	1990	Cleveland	10.0	5.0	0.0	1	..	..	..	..	93	..	..
7	1990	Cumbria	5.0	3.0	0.0	2	..	..	..	..	31	..	..
8	1990	Derbyshire	4.0	2.0	0.0	3	..	..	..	..	165	..	..
9	1990	Devon and Cornwall	12.0	13.0	0.0	3	..	..	..	..	276	..	..
10	1990	Dorset	5.0	9.0	0.0	2	..	..	..	..	15	..	..
11	1990	Durham	6.0	5.0	0.0	3	..	..	..	..	92	..	..
12	1990	Dyfed-Powys	9.0	6.0	0.0	7	..	..	..	..	71	..	..
13	1990	Essex	11.0	19.0	0.0	30	..	..	..	..	134	..	..

FIGURE 8.1: Reading the DataFrmae

8.2 Dropping The NAN Values into Data



```
In [8]: data1=data.dropna(axis=0,how="any")
        data1.head()
```

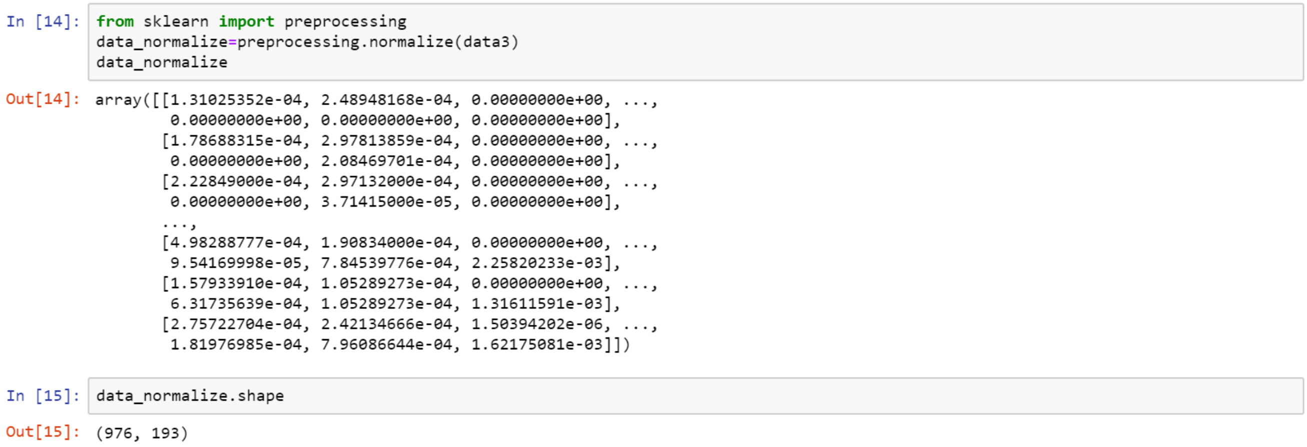
Out[8]:

	Year	Force	Homicide	Attempted murder	Child destruction	Causing death by dangerous or careless driving	Causing death by dangerous or careless driving (inc under influence)	Causing death by dangerous or careless driving (inc under influence).1	Causing death by dangerous or careless driving (inc under influence).2	Causing death by careless or inconsiderate driving	Wounding or other act endangering life	Inflicting GBH with intent	Use of substance or object to endanger life
2	1990	Avon and Somerset	10.0	19.0	0.0	7	..	..	..	..	168	..	..
3	1990	Bedfordshire	6.0	10.0	0.0	5	..	..	..	..	298	..	..
4	1990	Cambridgeshire	6.0	8.0	0.0	9	..	..	..	..	142	..	..
5	1990	Cheshire	6.0	2.0	0.0	15	..	..	..	..	99	..	..
6	1990	Cleveland	10.0	5.0	0.0	1	..	..	..	..	93	..	..

5 rows × 197 columns

Fig 2. Dropping the Nan Values into Data

8.3 Data Normalization



```
In [14]: from sklearn import preprocessing
         data_normalize=preprocessing.normalize(data3)
         data_normalize

Out[14]: array([[1.31025352e-04, 2.48948168e-04, 0.00000000e+00, ...,
                 0.00000000e+00, 0.00000000e+00, 0.00000000e+00],
                [1.78688315e-04, 2.97813859e-04, 0.00000000e+00, ...,
                 0.00000000e+00, 2.08469701e-04, 0.00000000e+00],
                [2.22849000e-04, 2.97132000e-04, 0.00000000e+00, ...,
                 0.00000000e+00, 3.71415000e-05, 0.00000000e+00],
                ...,
                [4.98288777e-04, 1.90834000e-04, 0.00000000e+00, ...,
                 9.54169998e-05, 7.84539776e-04, 2.25820233e-03],
                [1.57933910e-04, 1.05289273e-04, 0.00000000e+00, ...,
                 6.31735639e-04, 1.05289273e-04, 1.31611591e-03],
                [2.75722704e-04, 2.42134666e-04, 1.50394202e-06, ...,
                 1.81976985e-04, 7.96086644e-04, 1.62175081e-03]])

In [15]: data_normalize.shape

Out[15]: (976, 193)
```

8.4 Elbow Method for Finding the optimal K-Values

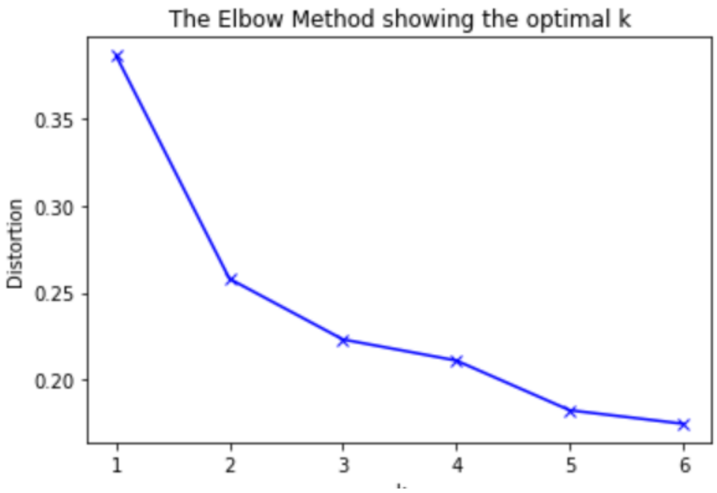
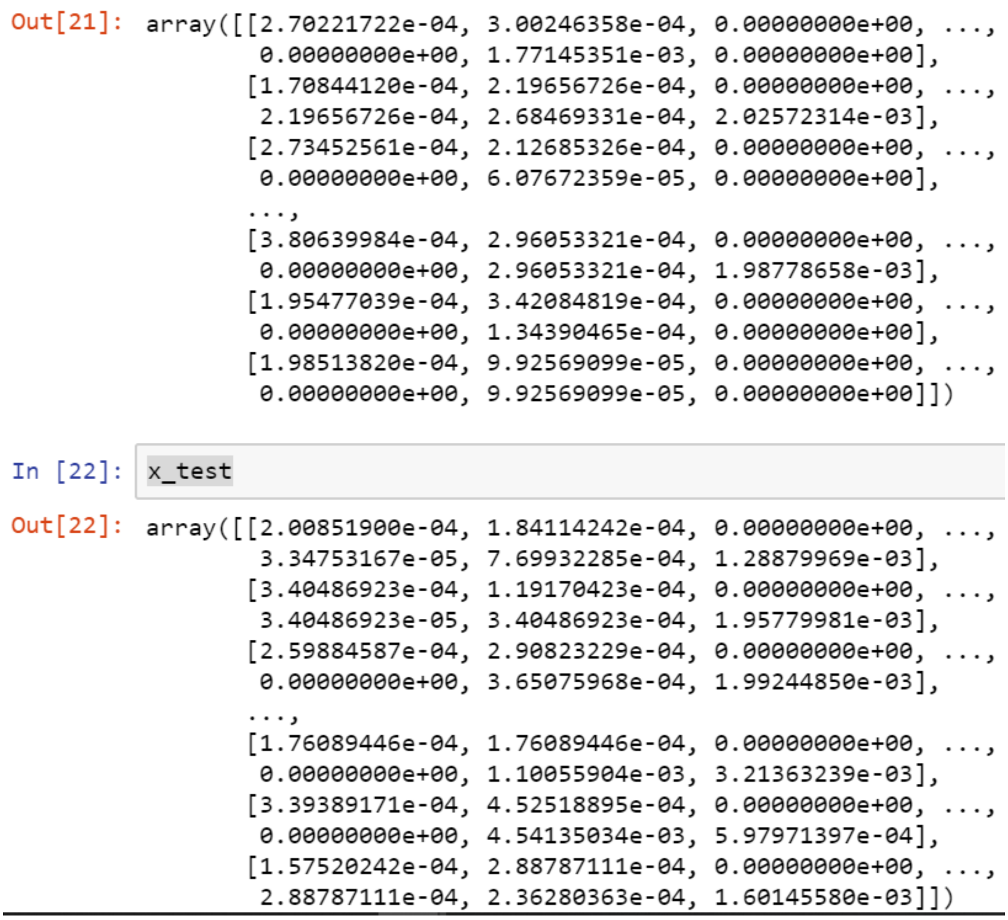


Fig 8.4: The Elbow Method

8.5 Split the data into training and testing



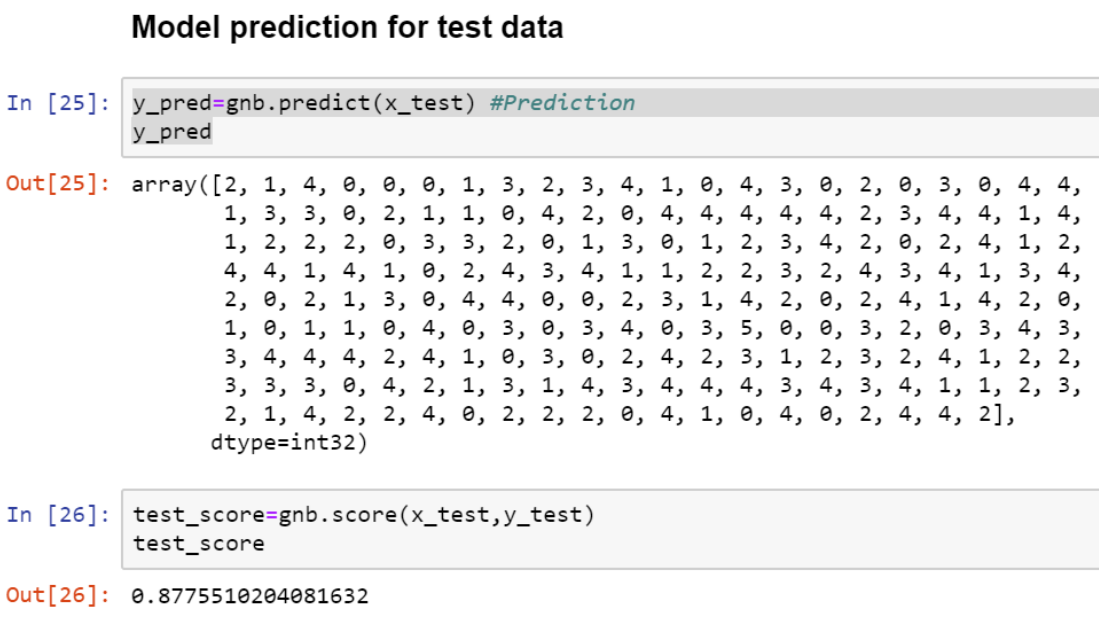
```
Out[21]: array([[2.70221722e-04, 3.00246358e-04, 0.00000000e+00, ...,
                 0.00000000e+00, 1.77145351e-03, 0.00000000e+00],
                [1.70844120e-04, 2.19656726e-04, 0.00000000e+00, ...,
                 2.19656726e-04, 2.68469331e-04, 2.02572314e-03],
                [2.73452561e-04, 2.12685326e-04, 0.00000000e+00, ...,
                 0.00000000e+00, 6.07672359e-05, 0.00000000e+00],
                ...,
                [3.80639984e-04, 2.96053321e-04, 0.00000000e+00, ...,
                 0.00000000e+00, 2.96053321e-04, 1.98778658e-03],
                [1.95477039e-04, 3.42084819e-04, 0.00000000e+00, ...,
                 0.00000000e+00, 1.34390465e-04, 0.00000000e+00],
                [1.98513820e-04, 9.92569099e-05, 0.00000000e+00, ...,
                 0.00000000e+00, 9.92569099e-05, 0.00000000e+00]])

In [22]: x_test

Out[22]: array([[2.00851900e-04, 1.84114242e-04, 0.00000000e+00, ...,
                 3.34753167e-05, 7.69932285e-04, 1.28879969e-03],
                [3.40486923e-04, 1.19170423e-04, 0.00000000e+00, ...,
                 3.40486923e-05, 3.40486923e-04, 1.95779981e-03],
                [2.59884587e-04, 2.90823229e-04, 0.00000000e+00, ...,
                 0.00000000e+00, 3.65075968e-04, 1.99244850e-03],
                ...,
                [1.76089446e-04, 1.76089446e-04, 0.00000000e+00, ...,
                 0.00000000e+00, 1.10055904e-03, 3.21363239e-03],
                [3.39389171e-04, 4.52518895e-04, 0.00000000e+00, ...,
                 0.00000000e+00, 4.54135034e-03, 5.97971397e-04],
                [1.57520242e-04, 2.88787111e-04, 0.00000000e+00, ...,
                 2.88787111e-04, 2.36280363e-04, 1.60145580e-03]])
```

Fig 8.5: Split the data into training and testing

8.6 Model Prediction of test Data



Model prediction for test data

In [25]:
```
y_pred=gnb.predict(x_test) #Prediction
y_pred
```

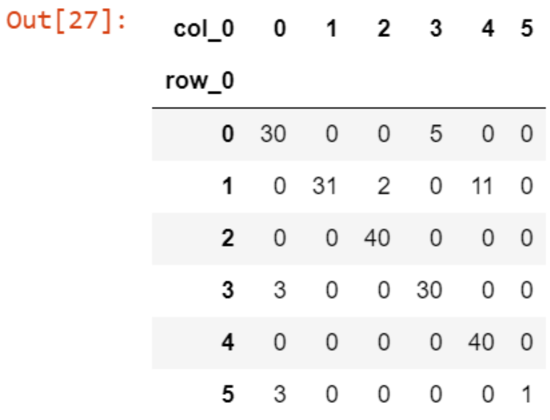
Out[25]:
```
array([2, 1, 4, 0, 0, 0, 1, 3, 2, 3, 4, 1, 0, 4, 3, 0, 2, 0, 3, 0, 4, 4,
       1, 3, 3, 0, 2, 1, 1, 0, 4, 2, 0, 4, 4, 4, 4, 4, 2, 3, 4, 4, 1, 4,
       1, 2, 2, 2, 0, 3, 3, 2, 0, 1, 3, 0, 1, 2, 3, 4, 2, 0, 2, 4, 1, 2,
       4, 4, 1, 4, 1, 0, 2, 4, 3, 4, 1, 1, 2, 2, 3, 2, 4, 3, 4, 1, 3, 4,
       2, 0, 2, 1, 3, 0, 4, 4, 0, 0, 2, 3, 1, 4, 2, 0, 2, 4, 1, 4, 2, 0,
       1, 0, 1, 1, 0, 4, 0, 3, 0, 3, 4, 0, 3, 5, 0, 0, 3, 2, 0, 3, 4, 3,
       3, 4, 4, 4, 2, 4, 1, 0, 3, 0, 2, 4, 2, 3, 1, 2, 3, 2, 4, 1, 2, 2,
       3, 3, 3, 0, 4, 2, 1, 3, 1, 4, 3, 4, 4, 4, 3, 4, 3, 4, 1, 1, 2, 3,
       2, 1, 4, 2, 2, 4, 0, 2, 2, 2, 0, 4, 1, 0, 4, 0, 2, 4, 4, 2],
      dtype=int32)
```

In [26]:
```
test_score=gnb.score(x_test,y_test)
test_score
```

Out[26]:
```
0.8775510204081632
```

FIGURE 8.6: Model Prediction of test data

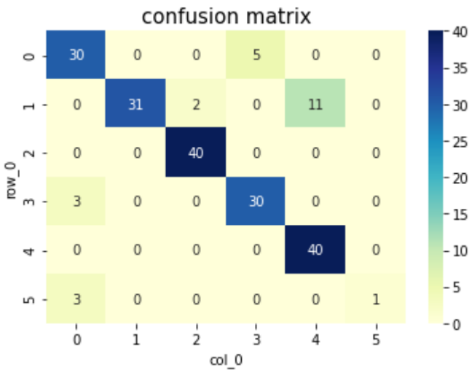
8.7Confusion Matrix



Out[27]:

col_0	0	1	2	3	4	5
row_0						
0	30	0	0	5	0	0
1	0	31	2	0	11	0
2	0	0	40	0	0	0
3	3	0	0	30	0	0
4	0	0	0	0	40	0
5	3	0	0	0	0	1

Fig 8.7. Confusion Matrix



Chapter 9

CONCLUSION

This project presents a new framework for clustering and predicting crimes based on real data. Examining the methods proposed for crime prediction shows that the parameters such as the effect of outliers in the data mining pre-processing, quality of the training and testing data, and the value of features have not been addressed before. In this framework, the K means algorithm was used to improve outlier detection in the pre-processing phase, and the fitness function was defined based on accuracy and classification error parameters. In order to improve the clustering process, the features were weighted, and the low-value features were deleted through selecting a suitable threshold. The proposed method was analysed with different clustering and classification algorithms

Chapter 10

REFERENCES

[1]. https://www.kaggle.com

[2]. https://www.kaggle.com/c/sf-crime

[3]. R. Solomonoff. “A New Method for Discovering the Grammars of Phrase Structure Language”, Proc. Int’l Conf. Information Processing, Unesco, 1959, pp. 285-290

[4]. E. Hunt, J. Marin, P. Stone. “Experiments in Induction”, Academic Press, 1966

[5]. L. Samuel. “Some Studies in Machine Learning Using the Game of Checkers, Part II”, IBM J. Research and Development, vol. II, no. 4, 1967, pp. 601-618

[6]. J. Quinlan. “Introduction of Decision Trees”, Machine Learning, vol. I, Mar. 1986, pp. 81-106 [7]. Z. Pawlak. “Rough Sets: Theoretical Aspects of Reasoning about Data”, Kluwer Academic Publishers, 1991

[8]. F. Rosenblatt. “The Perceptron: A Perceiving and Recognizing Automaton”, tech. report 85-460-1, Aeronautical Lab., Cornell Univ., 1957

[9]. D. E. Rumelhart, J. L. McClelland. “Parallel Distributed Processing”, MIT Press, 1986

[10].http://www.forbes.com/sites/bernardmarr/2016/02/19/a-short-history-of-machine-learning-every-managershould-read

[11].A. Smola, S.V.N. Vishwanathan. Introduction to Machine Learning, 1st ed. Cambridge University Press, UK, 2008

[12].A. L. Blum, P. Langley. “Selection of relevant features and examples in machine learning”, Elsevier, 1997

[13].http://www.astroml.org/sklearn_tutorial/general_concepts.html

[14].I. Hendrickx, A. Van den Bosch. “Hybrid algorithms with Instance-Based Classification”. Machine Learning: ECML2005. Springer. pp. 158-169.

[15].E. Alpaydin. “Introduction to Machine Learning”, The MIT Press, Cambridge, Massachusetts, 2010

[16]. L. Torrey, J. Shavlik. “Transfer Learning”. University of Wisconsin [Online] Available: ftp://ftp.cs.wisc.edu/machinelearning/shavlik-group/torrey.handbook09.pdf

[17].A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, T. Lillycrap. “One-Shot Learning with Memory-Augmented Neural Networks”. Google DeepMind. [Online]

[18].K. Wagsta, C. Cardie, S. Rogers, S. Schroedl. “Constrained K-means Clustering with Background Knowledge”. In Proceedings of the Eighteenth International Conference of Machine Learning, 2001, pp. 577-584.

[19].https://apandre.wordpress.com/visible-data/cluster-analysis/

[20].http://scikit-learn.org/stable/index.html

[21].http://www.byclb.com/TR/Tutorials/neural_networks/ch11_1.htm

[22].A. Abraham. “Handbook of Measuring System Design”, John Wiley & Sons, Ltd., 2005, pp. 901-908.

[24].P. Domingos. "A Few Useful Things to Know about Machine Learning". University of Washington. [Online] Available: https://homes.cs.washington.edu/~pedrod/papers/cacm12.pdf

[25].https://www.ovh.es/servidores_dedicados/details-servers-range-HOST-id-2016-HOST-32L.xml

[26] J. Agarwal, R. Nagpal, and R. Sehgal, ―Crime analysis using k-means clustering,‖ International Journal of Computer Applications, Vol. 83 – No4, December 2013.

[27] J. Han, and M. Kamber, ―Data mining: concepts and techniques,‖ Jim Gray, Series Editor Morgan Kaufmann Publishers, August 2000.

[28] P. Berkhin, ―Survey of clustering data mining techniques,‖ In: Accrue Software, 2003.

[29] W. Li, ―Modified k-means clustering algorithm,‖ IEEE Congress on Image and Signal Processing, pp. 616- 621, 2006.

[30] D.T Pham, S. Otri, A. Afifty, M. Mahmuddin, and H. Al-Jabbouli, ―Data clustering using the Bees algorithm,‖ proceedings of 40th CRIP International Manufacturing Systems Seminar, 2006.

[31] J. Han, and M. Kamber, ―Data mining: concepts and techniques,‖ 2nd Edition, Morgan Kaufmann Publisher, 2001. (IJARAI) International Journal of Advanced Research in Artificial Intelligence, Vol. 4, No.8, 2015 17 | P a g e www.ijarai.thesai.org

[32] S. Joshi, and B. Nigam, ―Categorizing the document using multi class classification in data mining,‖ International Conference on Computational Intelligence and Communication Systems, 2011.

[33] T. Phyu, ―Survey of classification techniques in data mining,‖ Proceedings of the International Multi Conference of Engineers and Computer Scientists Vol. IIMECS 2009, March 18 - 20, 2009, Hong Kong.