Pamantasan ng Lungsod ng Maynila

ENHANCED MACQUEEN'S ALGORITHM FOR IDENTIFYING DIVERSE CRIME PATTERNS IN THE CITY OF MANILA

A Thesis Presented to the
Faculty of Computer Science Department
College of Information System and Technology Management
Pamantasan ng Lungsod ng Maynila

In Partial Fulfillment of the Requirements for the Degree
Bachelor of Science in Computer Science

By
Kim Emerson M. Tan
Arwin B. Tiangco

Vivien A. Agustin
Thesis Adviser

April 2025

APPROVAL SHEET

The thesis hereto titled

ENHANCEMENT OF RANDOM FORESTS APPLIED TO PROGRAM-RECOMMENDATION FOR WAITLISTED APPLICANTS

prepared and submitted by **Arwin B. Tiangco** and **Kim Emerson M. Tan** in partial fulfillment of the requirements for the degree of **Bachelor of Science in Computer Science** has been examined and is recommended for acceptance and approval for **ORAL EXAMINATION**.

VIVIEN A. AGUSTIN, MIT
Adviser

PANEL OF EXAMINERS

Approved by the Committee on Oral Examination
with a grade of _____ on _____.

VIVIEN A. AGUSTIN, MIT
Coordinator

RICHARD C. REGALA
Panel Member

MARILOU B. MANGROBANG
Panel Member

Accepted and approved in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science

RAYMUND M. DIOSES
Chairperson
Computer Science Department

KHATALYN E. MATA, DIT
Dean
College of Information Systems and Technology Management

ABSTRACT

MacQueen's algorithm is a variant of the k-means algorithm used to determine clusters. However, the algorithm has its limitations that impact its accuracy and efficiency, resulting in suboptimal clustering. This study aimed to enhance MacQueen's algorithm for analyzing diverse crime patterns in the city of Manila by addressing these limitations using Isolation Forest for outliers, Adaptive K-Means++ for algorithm initialization, and Gap Statistics to determine the optimal number of clusters. Isolation Forest was employed to detect and remove outliers from the dataset, as they significantly impact clustering results. Adaptive K-Means++ improved the initialization process by optimizing the placement of initial centroids, reducing the sensitivity of the algorithm to poor starting conditions. Gap Statistics was utilized to determine the optimal number of clusters, greatly enhancing the algorithm's accuracy. The enhanced MacQueen's algorithm demonstrated a significant overall improvement in clustering performance, resulting in more accurate and distinct clusters. The proposed enhancements effectively addressed the limitations of the traditional MacQueen's algorithm, improving its accuracy and efficiency. This makes the enhanced algorithm highly applicable to real-world problems involving clustering

Keywords: Clustering, Initialization, MacQueen's Algorithm, Adaptive K-Means++, Gap Statistics, Isolation Forest

ACKNOWLEDGEMENTS

The completion of this research would not have been possible without the unwavering support, guidance, and encouragement of several individuals. The researcher extends deepest gratitude to Prof. Vivien A. Agustin, whose expertise, patience, and invaluable feedback shaped the direction and refinement of this study. Their mentorship has been a cornerstone of this academic journey.

The researcher is also profoundly thankful to the esteemed panelists, Prof. Marilou Mangrobang, and Prof. Richard Regala, for their insightful critiques, thoughtful suggestions, and rigorous evaluation, all of which contributed significantly to the improvement of this work.

Above all, our immeasurable appreciation goes to our parents, whose boundless love, sacrifices, and steadfast belief provided the strength and motivation to persevere. Their unwavering support, even in the most challenging moments, made this achievement possible.

The Researchers

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter One

INTRODUCTION

1.1 Background of the Study

The levels of crime and patterns in the city of Manila of the Philippines greatly impact the safety and welfare of individuals (Balita, 2023). The Philippine Development Plan 2023 – 2028 states that ensuring security and maintaining order are crucial for laying the foundation for the country's progress towards inclusivity, resilience and economic competitiveness. As the country focuses on its development strategies, addressing these crime trends plays a pivotal role in the country's sustainable growth (Philippine Development Plan, 2023).

In addition, the police and lawmakers are striving to create safer communities that encourage social progress and economic prosperity. Their aim is to achieve this by gaining a deeper understanding of the various crime patterns in the city of Manila. Crimes possess different types of patterns when it comes to their analysis. Some of the major patterns are spatial and temporal patterns. Spatial and temporal patterns refer to the "hotspots" or areas where crime rates are significantly higher compared to other locations.

These concentrated areas exhibit different types of criminal activities and the times of day when they are most likely to occur. However, crime is not just a matter of location and time. It is also influenced by different factors such as socioeconomic status and geography (Rubio et al., 2018). For example, individuals from lower socioeconomic backgrounds may be more likely to be involved in certain types of crimes compared to those from higher socioeconomic backgrounds.

To further understand these patterns in crime data, machine learning algorithms that involve clustering such as MacQueen's k-means Algorithm can be used. MacQueen's Algorithm can help in identifying these different patterns in crime data. The algorithm works by classifying a given dataset into a certain number of clusters, denoted as 'k', grouping data points into clusters based on their similarities.

The application of MacQueen's k-means algorithm, can help group similar crime incidents into clusters based on shared characteristics (Agarwal et al., 2013). This can also help identify hotspots or areas where crime rates are significantly higher compared to other locations; it can also reveal the types of criminal activities that are more likely to happen at different times and in different areas. This method acknowledges that crime in one area may affect neighboring areas, providing a more detailed view of crime dynamics within the city of Manila.

By modifying the initialization of MacQueen's algorithm through implementation of adaptive k-means++, the research aims to identify the different crime patterns across NCR. The hypothesis is that the modification to the clustering algorithm can provide more accurate data about crime related patterns, considering the large dataset with the presence of poor initialization, and existence of outliers.

1.2 Statement of the Problem

Traditional clustering algorithms like MacQueen's algorithm has its limitation when it comes to accurately capturing the true nature of a dataset. Specifically, when the dataset involves varying density, unequal distribution, and presence of outliers. As a result, the clustering results may provide oversimplified, or under simplified results which affects the true nature of the data.

Specifically, this study aims to address the following:

1.) MacQueen's K-Means Algorithm exhibits high sensitivity to the initial placement of centroids that leads to suboptimal clustering:

The algorithm's performance significantly depends on the initial centroids assigned to clusters. Poorly chosen initial centroids can lead to suboptimal clustering results.

MacQueen's Algorithm

1: choose k as the number of clusters
2: randomly choose k datapoints as centroids ***(Problem 1)***
3: repeat
4: for each datapoint do
5: assign point to closest centroid
6: recalculate centroid as mean over all points assigned
7: end for
8: until convergence

According to S. Suraya et al. (2023), to improve clustering evaluation metrics, appropriate parameters, and cluster initialization is crucial. Additionally, their results provide different evaluation metrics between normalized and non-normalized datasheets. Another study conducted by V. Romanuke et al. (2023) discussed problems with the k-means algorithm specifically when dealing with large datasets, highlighting how important the selection of initial centroids is to the outcome of the clustering.

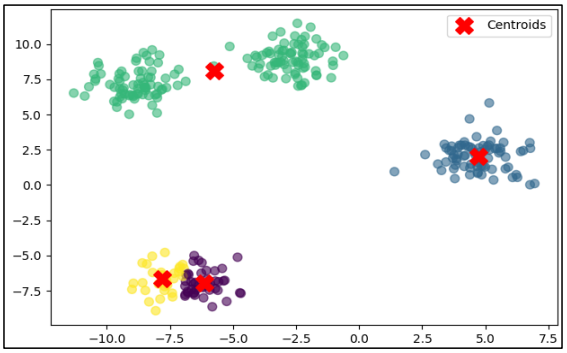

Figure 1.1 Macqueen's Algorithm Initialization

Figure 1.1. shows the clustering result shows the less optimal placement of data clusters with respect to its initial centroid. The upper left cluster shows two distinct clusters that are assigned under one centroid; this centroid placement is ineffective and it does not represent either of the data clusters, and may lead to misclassifications of data.

2.) MacQueen's K-Means Algorithm is sensitive and cannot handle outliers which affects the centroid calculation.

The algorithm is sensitive to outliers present in a dataset which affects the centroid calculation, and accuracy of clustering. Outliers can affect the positioning of centroids on each update.

MacQueen's Algorithm

1: choose k as the number of clusters
2: randomly choose k datapoints as centroids
3: repeat
4: for each datapoint do
5: assign point to closest centroid ***(Problem 2)***
6: recalculate centroid as mean over all points assigned
7: end for
8: until convergence

According to Li, L. et al. (2019), when data doesn't follow an assumed distribution model—such as being evenly distributed among quartiles—and contains outliers, traditional statistical methods based on these assumptions can lead to inaccurate results.

On the other hand, Barai and Dey (2017) highlight the importance of addressing outliers by proposing preprocessing algorithms, such as a distance-based algorithm and a cluster-based approach. These methods emphasize the importance of removing outliers, particularly in clustering techniques like k-mean.

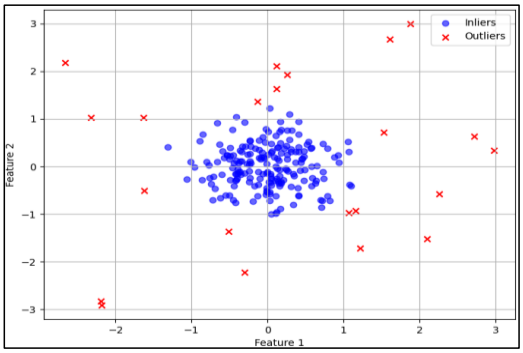

Figure 1.2. Isolation Forest Outlier Detection

Figure 1.2. shows the result of clustering representation of outliers in a dataset represented as red crosses. The isolation forest algorithm isolates points that deviate from the main cluster. These outliers affect the clustering result of MacQueen's algorithm, since the algorithm cannot handle outliers effectively, it may result in centroid misplacement, increased variance, and incorrect cluster assignments.

3.) MacQueen's K-Means Algorithm requires a pre-defined number of clusters which limits the clustering result.

The algorithm requires a pre-defined number of clusters before running. Having a pre-defined cluster limits the clustering outcome which affects the reliaility and effectiveness of the algorithm.

MacQueen's Algorithm

1: choose k as the number of clusters ***(Problem 3)***
2: randomly choose k datapoints as centroids
3: repeat
4: for each datapoint do
5: assign point to closest centroid
6: recalculate centroid as mean over all points assigned
7: end for
8: until convergence

According to Umargano et al. (2019) one of the weaknesses of the k-means algorithm is that the number of clusters relies heavily on assumption. The study also highlights the importance of using an appropriate method to determine the optimal number of clusters, as this affects the initial placement of centroids, which in turn impacts the algorithm's clustering results.

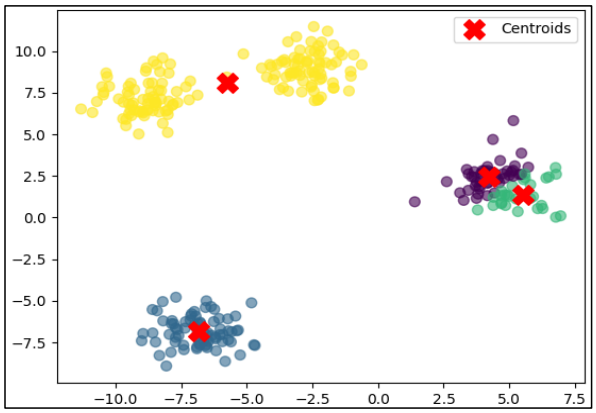

Figure 1.3. MacQueen's Algorithm Predefined k=4

Figure 1.3 shows the clustering result of MacQueen's algorithm with a predefined number of clusters (k=4). The data points are grouped into four distinct clusters; however, the results display three suboptimal clusterings. The upper cluster's data points are scattered, indicating that the centroid placement did not effectively capture the cluster's density. The middle-right cluster shows poor separation, with overlapping data points, suggesting that the data points in this cluster are somewhat indistinct.

Having a pre-defined number of clusters in MacQueen's algorithm means the data is forced into a set number of groups, whether or not that number accurately reflects the natural patterns in the data. If the chosen ***k*** is not optimal, groups may merge when ***k*** is too small or split apart when ***k*** is too large. Since the algorithm relies on the starting centroids, a poor choice of ***k*** can result in scattered or poorly formed clusters.

1.3 Objective of the Study

1.3.1 General Objective

The objective of this study is to enhance the effectiveness of MacQueen's algorithm for better clustering results. The researchers aim to enhance MacQueen's algorithm by utilizing an adaptive *k-means++* initialization to fill the gaps of MacQueen's algorithm. This study also seeks to create a hybrid approach that combines the strengths of the algorithm, and initialization, enhancing the effectiveness of MacQueen's algorithm. The goal is to initialize the dataset to lessen the outliers, and improve data accuracy for better clustering results.

1.3.2 Specific Objective

To address the identified issues, the study aims to enhance MacQueen's algorithm by accomplishing the following specific objectives:

1. To develop an improved initialization method by incorporating an adaptive k-means++ approach.
2. To apply isolation forest to eliminate outliers.
3. To apply the gap statistics method for finding the optimal number of clusters (k).

1.4 Significance of the Study

The modified MacQueen's Algorithm has important ramifications for a number of fields, especially urban planning and law enforcement. This study attempts to give a strong analytical tool that can result in more informed decision-making and smart resource allocation by improving the algorithm's efficacy in detecting crime patterns in Manila.

By resolving problems with centroid initialization, outlier sensitivity, and the requirement for pre-defined clusters, the enhanced technique will enable more precise grouping of crime data. It is expected that the modified approach will produce more dependable results by recalculating centroids after each iteration and adjusting for different

cluster sizes. This development enhances efforts to create safer communities through focused interventions in addition to advancing our understanding of Manila's crime dynamics.

Law Enforcement Agencies. This study provides law enforcement with an enhanced analytical tool to identify crime hotspots and trends more accurately. By improving crime pattern detection, the modified MacQueen's Algorithm can assist in optimizing patrol deployment, strengthening community safety measures, and formulating evidence-based crime prevention strategies.

Urban Planners. Urban planners can benefit from this study by gaining deeper insights into crime distribution across Manila. The improved clustering technique allows for better spatial analysis of crime, which can inform the design of safer urban spaces, allocation of resources, and implementation of strategic infrastructure projects aimed at crime reduction.

Policy Makers. The findings of this study offer valuable information to policymakers in crafting data-driven crime prevention policies. By understanding crime dynamics more effectively, they can create legislation and programs that enhance public safety while ensuring efficient resource distribution.

Data Scientists and Analysts. Researchers and professionals in data science can leverage the study's contributions to clustering techniques, particularly in handling centroid initialization issues, outlier sensitivity, and adaptive cluster sizing. These advancements can be applied to various domains beyond crime analysis, such as public health, transportation, and economic planning.

Future Researchers. This study serves as a foundation for future research in crime pattern detection and clustering algorithms. Researchers interested in further improving crime analysis methodologies can build upon this work by integrating additional machine learning techniques, testing the algorithm in different urban settings, or exploring real-time crime data analysis.

1.5 Scope and Limitations

The goal of this study is to improve the effectiveness of MacQueen's k-means method in evaluating crime trends, particularly in the City of Manila, by including an adaptive k-means++ initialization. Sensitivity to initial centroid placement, efficiently managing outliers, and figuring out the ideal number of clusters without depending on preset values are the main concerns that the study will focus on resolving.

Commented [TAB1]: Discuss more yung scope

One key point of this study seeks to resolve is sensitivity to initial centroid placement. In k-means clustering, the choice of initial centroids significantly affects the final cluster formation. Poorly initialized centroids can lead to suboptimal clusters, resulting in misleading crime pattern identification. By incorporating adaptive k-means++ initialization, the study aims to optimize centroid selection, ensuring that clusters are well-formed from the start and reducing the likelihood of convergence to local optima.

The study also aimed to enhance the efficient handling of outliers of the algorithm applied in crime data. Crime incidents often include isolated or extreme values, such as rare but severe offenses or misreported cases. Traditional k-means methods, including MacQueen's, struggle with outliers because they can distort centroid movement, leading to inaccurate cluster formations. This study will implement techniques, such as isolation forest, to minimize the impact of such anomalies, ensuring that the resulting crime clusters reflect meaningful trends rather than being skewed by outliers.

Additionally, the study aims to address the challenge of determining the optimal number of clusters without relying on pre-determined values. In standard k-means clustering, the number of clusters must be defined beforehand, which can lead to arbitrary or suboptimal selections. A predefined k may not accurately represent crime distribution, causing either *over-segmentation* (too many clusters) *or under-segmentation* (too few clusters)**.** This study will integrate a dynamic method that adjusts the number of clusters based on crime data characteristics, improving the adaptability of the clustering process.

However, it is important to take that this study will not investigate other clustering techniques or algorithms outside of MacQueen's framework, even if its goal is to present a thorough examination of crime trends using this modified approach. In order to ensure that the results are relevant to local law enforcement, the program will only be used with crime statistics that mirrors the crime statistics happening in the City of Manila. Furthermore, even while this study attempts to be broadly applicable within Manila, it does not take into consideration particular socioeconomic aspects that might affect crime trends in other cities or areas.

1.6 Definition of Terms

The following terms are used throughout this study. To further understand the terms the definition are as follows:

Clustering - A machine learning technique that organizes a collection of objects so that they are more similar to one another than to those in other groups. For exploratory data analysis, it is frequently utilized.

Gap Statistics - A statistical technique that compares the total intra cluster variance for various values of k with their predicted values under a null reference distribution in order to estimate the ideal number of clusters (k).

Hotspots - Particular regions having a high rate of criminal activity. For law enforcement organizations to deploy resources efficiently and put preventative measures in place, hotspot identification is essential

Isolation Forest - An anomaly detection approach that uses random partitioning to isolate observations in order to find outliers. It works well for finding irregularities in datasets with many dimensions.

K-Means++ - A refined iteration of the k-means method that enhances centroid initialization. By spreading out the initial centroids, it lessens the possibility of poor clustering outcomes brought on by haphazard centroid placement.

MacQueen's Algorithm - Also known as k-means, this clustering algorithm was created by Peter MacQueen in 1967. By iteratively improving the centroids' placement, it divides a dataset into "k" different clusters according to how similar the data points are.

Commented [TAB2]: Alphabetically Arranged

Outliers - Data points that substantially deviate from the rest of the observations in a dataset. In statistical analysis, outliers have the potential to distort and mislead the interpretation of data, especially in clustering methods.

Over-segmentation – too much clusters are formed, leading to over clustering which may not reflect the true nature of the data.

Spatial Patterns - How criminal incidents are distributed and arranged in different geographic areas. Finding crime hotspots—areas where events happen more frequently—is made easier by spatial trends.

Temporal Patterns - Trends pertaining to the times of day and week when crimes are more likely to occur are revealed by analyzing crime episodes across time.

Under-segmentation – few clusters are made, not reflecting the true nature of the data.

Chapter Two

REVIEW OF RELATED LITERATURE

This chapter contains related literature made by the researchers. The information cited in this chapter helps address and familiarize similar issues that are present in the current study.

2.1 Related Literature

Commented [TAB3]: Add 2.2 Related Studies

Commented [TAB4]: Add comparative analysis

Centroid Update Approach to K-Means Clustering

The study conducted by Borlea et al. (2017) presents an improved clustering approach by addressing the centroid update, based on the baseline or classic k-means algorithm. The centroid update is integrated into the baseline version of k-means. But the modification done by the researchers is that a step was added for estimating the evolution of centroids. This additional step sped up the clustering process by updating the centroid every iteration if a condition is met. This modification to the centroid update provides a faster way, which lessens the iterations needed, to obtain the final clusters.

Enhanced Initial Centroids for K-means Algorithm

In a study conducted by Fabregas, Gerardo, and Tanguilig (2017), entitled "Enhanced Initial Centroid for k-means Algorithm", it discussed the enhancement of initial placement of centroids of k-means algorithms in general. The original k-means algorithm uses random choices for its initial centroid, which results in less reliable data set clustering. But in this study, the authors modified the k-means algorithm in terms of choosing its initial centroid. The findings revealed that the modified k-means algorithm is better than the baseline when it comes to selecting its initial centroid; better in terms of mathematical computation, and reliability.

Crime Data Analysis in Python using K - Means Clustering

Crime has been one of the biggest problems in the world, and several crimes have been recorded throughout history. Unfortunately, even though we have a large database of crimes, it is not being utilized effectively. Extracting useful information from these large databases of crimes can be advantageous and valuable in reducing crimes around the world. A study conducted by Saleh and Khan (2019) focused on predicting and analyzing crimes in the town of Chicago. They used the k-means algorithm to perform the analysis, make predictions, and visualize the patterns of different crimes. Their implementation involves pre-processing the data, the implementation of the k-means algorithm on the pre-processed datasets, and analysis and evaluation. Attributes that the researchers focused on in the study include crime type, time, location, and arrests made. They found that robbery is a crime that has been committed by many criminals, and they also found that most of the criminals were not arrested for their crimes.

Crime Analysis using K-Means Clustering

Another study conducted by Agarwal et al. (2013) also focused on using k-means clustering for crime analysis. The researchers performed crime analysis by applying K-means to their crime data set using the Rapid Miner tool, which is an open-source statistical and data mining package. Additionally, their proposed system architecture involves processes such as pre-processing of data, applying the "replace missing value operator" and normalization, performing k-means clustering, and finally analyzing the clusters formed. The dataset used for the crime analysis came from the police in England and Wales, which is from 1990 to 2011–12. Moreover, this paper focused on the analysis of homicide, and based on their results, they found that homicide crimes were decreasing from 1990 to 2011.

Fast color quantization using MacQueen's k-means algorithm

One of the important operations in image processing and analysis is color quantization (CQ). Color quantization is the process of reducing the number of distinct colors in an image while preserving its visual quality. This is usually done to reduce the size of the file or to simplify the color palette of the image for better display and storage. A study conducted by Thompson et al. Al (2019) applied Macqueen's algorithm, which is one of the versions of the K-means family of algorithms. In their study, they proposed a novel CQ method where the researchers fixed some of the problems in Macqueen's algorithm, specifically its high computational requirements and its sensitivity to initialization. In their proposed method, they implemented an adaptive and efficient cluster center initialization and a quasi-random algorithm that was used to uniformly distribute the data points to be clustered. Based on their results, the proposed method performs significantly faster than other common batch k-means algorithms, like Lloyd.

K-means-sharp: Modified centroid update for outlier-robust k-means clustering

Most of the real-world datasets contain noise and outliers. These typically affect the results of data analysis when the information produced is not accurate to the expected results. Some of the traditional machine learning algorithms don't have the capability to detect these outliers. An example of this is the k-means clustering algorithm. A study conducted by Olukanmi and Twala (2017) focused on addressing the problem of the classical k-means algorithm by introducing a new centroid update step that detects outliers automatically by means of a global threshold. The modified version of the classical k-means is what they call the k-means-sharp (k-means#). In addition, the modification allows k-means# to still maintain the efficiency and simplicity of the original algorithm while improving its performance. Based on the results, the k-means# exhibits a lower within-cluster mean squared error and demonstrates high accuracy and precision in detecting outliers tested across various datasets. Lastly, the proposed method does not require user intervention or prior knowledge of the number of outliers, making it an effective solution for detecting outliers in clustering algorithms like k-means.

On The Application of Fuzzy Clustering for Crime Hot Spot Detection

A study conducted by Grubesic (2006) explores the application of fuzzy clustering for detecting crime hot spots and highlights its advantage among hard-clustering methods like k-means. Some of the advantages include the capability of fuzzy clustering to handle ambiguity and outliers effectively. Additionally, the method provides a detailed snapshot of the data structure, making it a better tool for decision-making to identify crime hotspots. According to the findings of the study, it suggests that fuzzy clustering could offer a more detailed understanding of crime hot spots that could potentially aid law enforcement in resource allocation and policing strategies.

Survey on Crime Data Analysis Using a Different Approach of K-Means Clustering

A study conducted by Kumar and Semwal (2020) focuses on the use of k-means clustering for crime data analysis. The study highlights the importance of crime analysis in helping law enforcement solve crimes, and it also highlights different types of crime analysis. In addition, the paper discusses machine learning algorithms focusing on unsupervised learning and their role in identifying different patterns in the data. Furthermore, it discusses the details of the k-means algorithm and its application in crime analysis. Lastly, the conclusion of the paper emphasizes the importance and value of unsupervised learning for identifying suspect records and crime patterns. Future researchers are encouraged to further develop predictions.

An Improved K-Means with Artificial Bee Colony Algorithm for Clustering

With the development of technology, crime data has become even more advantageous when detecting and solving crimes due to the ability of certain technologies to analyze large amounts of data into useful and meaningful information. Several methods and algorithms have been developed to enhance the results of the analysis of data. A study conducted by Karimi and Gharehchopogh (2020) focused on improving a k-means algorithm using the Artificial Bee Colony (ABC) algorithm for crime clustering. Specifically, the algorithm improved the accuracy of clustering by improving the method

of selecting cluster centers and the assignment of data points to appropriate clusters. A data set with 1994 samples and 128 features has been used for the evaluation, and the results show that the accuracy of the proposed algorithm is higher than the k-means, and the purity value is 0.943 with 500 iterations.

Hybrid Clustering Algorithms for Crime Pattern Analysis

Data mining has also become prevalent in crime analysis because of its way of extracting useful information from large sets of data. In the study conducted by Inbaraj and Rao (2018), they used a clustering algorithm as their data mining approach to help with the detection of crimes and speed up solving crimes. In their study, they proposed a two-level clustering algorithm. The advantage of their proposed algorithm is that it can recognize illogically shaped clusters compared to conventional clustering methods. Based on the results of their study, their proposed method performs better than other two-level clustering methods such as affinity propagation (AP) and RBF networks.

2.2 Related Studies

Enhancements to MacQueen's Algorithm

Among the drawbacks of the conventional MacQueen's approach are its sensitivity to the initial centroid placement and its inability to manage outliers. By resolving these problems, clustering performance can be enhanced with features like Adaptive K-Means++ initialization and outlier detection using Isolation Forest (Arthur & Vassilvitskii, 2007; Liu et al., 2008).

Importance of Adaptive Initialization

By optimizing centroid placement and mitigating the effects of unfavorable initial conditions, adaptive K-Means++ has been demonstrated to enhance the initialization procedure in clustering algorithms (Arthur & Vassilvitskii, 2007). For datasets where random initialization could produce less-than-ideal clustering results, this method is especially advantageous (Romanuke et al., 2023).

Outlier Detection in Clustering

By altering centroid placements and raising variation within clusters, outliers have a substantial impact on clustering results (Li et al., 2019). By effectively detecting and eliminating outliers, methods such as Isolation Forest improve the precision of clustering algorithms (Liu et al., 2008).

Gap Statistics for Optimal Clustering

By comparing the observed and expected within-cluster sum of squares, the Gap Statistics method offers a methodical way to calculate the ideal number of clusters (k) (Tibshirani et al., 2001). By avoiding arbitrary k selection, this technique produces clustering results that are more trustworthy.

Crime Pattern Analysis

Socioeconomic factors influence the spatial and temporal dynamics of crime patterns in urban settings such as Manila (Rubio et al., 2018). Although clustering techniques need to be improved to handle complicated data distributions, they can be useful in identifying "hotspots" and comprehending crime trends (Agarwal et al., 2013).

Impact of Outliers on Clustering

By influencing centroid computations and resulting in inaccurate cluster assignments, outliers can deceive clustering algorithms (Barai & Dey, 2017). Accurate clustering results, particularly in crime data analysis, depend on the efficient detection and elimination of outliers (Li et al., 2019).

Adaptive K-Means++ in Clustering

By guaranteeing that initial centroids are evenly distributed throughout the data space, the Adaptive K-Means++ technique improves clustering performance (Arthur &

Vassilvitskii, 2007). This method produces more consistent results by lessening the sensitivity of clustering algorithms to beginning conditions (Suraya et al., 2023).

Gap Statistics for Cluster Validation

By identifying the ideal number of clusters based on statistical data, the Gap Statistics approach is a reliable tool for validating clustering results (Tibshirani et al., 2001). By choosing the best k value for the dataset, this method helps prevent overfitting or underfitting.

Crime Data Clustering Challenges

The complexity of crime data, including temporal and regional patterns driven by a range of socioeconomic factors, makes clustering it difficult (Rubio et al., 2018). Effective crime pattern analysis requires improved clustering algorithms that effectively manage outliers and maximize centroid placement (Agarwal et al., 2013).

Enhanced Algorithm for Crime Patterns

The MacQueen's approach for detecting various criminal patterns can be greatly improved by including strategies like Gap Statistics for optimal k determination, Isolation Forest for outlier detection, and Adaptive K-Means++ initialization. These improvements increase the accuracy and dependability of clustering, offering important new information about the dynamics of crime in urban settings (Arthur & Vassilvitskii, 2007; Liu et al., 2008).

2.3 Comparative Analysis

Finding patterns in complicated datasets requires the use of clustering algorithms, and several approaches have been developed to tackle various problems. The advantages and disadvantages of MacQueen's algorithm are contrasted with those of other well-known clustering algorithms in this section.

MacQueen's Algorithm vs. Traditional K-Means

A variation of the k-means clustering technique, MacQueen's algorithm dynamically changes centroids when data points are redistributed (MacQueen, 1967). Although MacQueen's approach offers faster convergence than classical k-means, which recalculates centroids only after all points have been allocated, it still has problems with outlier sensitivity and initial centroid placement (Hartigan & Wong, 1979). Traditional k-means, on the other hand, is more popular but could need more iterations to produce comparable outcomes.

MacQueen's Algorithm vs. Hierarchical Clustering

By combining or dividing preexisting clusters, hierarchical clustering algorithms, such agglomerative and divisive approaches, create a hierarchy of clusters (Jain & Dubes, 1988). These techniques can handle complicated datasets with different densities and don't require a specific number of clusters (k). Nevertheless, they require a lot of computing power and might not function effectively with big datasets. Although MacQueen's approach works well with big datasets, it is sensitive to beginning conditions and needs a predetermined k.

MacQueen's Algorithm vs. DBSCAN

A density-based clustering technique called DBSCAN (Density-Based Spatial Clustering of Applications with Noise) divides data points into clusters according to proximity and density (Ester et al., 1996). DBSCAN is resilient to outliers and does not require a predetermined number of clusters, in contrast to MacQueen's algorithm. Nevertheless, it may not function well in high-dimensional areas and may have trouble with datasets of different densities.

MacQueen's Algorithm vs. K-Medoids

Like k-means, k-Medoids methods, such PAM (Partitioning Around Medoids), employ medoids, or real data points, as centroids rather than mean values (Kaufman & Rousseeuw, 1990). By using this method, k-medoids are more resilient to outliers than

MacQueen's methodology. K-medoids, on the other hand, can be computationally costly and might not be as effective at handling big datasets as MacQueen's technique.

Every clustering algorithm has advantages and disadvantages. Although MacQueen's approach is efficient and provides dynamic centroid updates, it needs to be improved to properly handle outliers and initial centroid placement. Researchers can choose the best approach by contrasting these algorithms, considering the particulars of their dataset and the needs of their analysis.

2.4 Synthesis

The literature review highlights the significance of improving clustering algorithms, such as MacQueen's k-means, for efficient analysis of complex datasets, especially when it comes to identifying crime patterns in urban regions like Manila. MacQueen's technique, a variation of k-means, can produce less-than-ideal clustering results due to its drawbacks, including sensitivity to the initial centroid location and an inability to manage outliers (MacQueen, 1967; Hartigan & Wong, 1979). Improvements like Isolation Forest for outlier detection, Adaptive K-Means++ initialization, and the Gap Statistics approach for figuring out the ideal number of clusters are essential to resolving these problems.

By placing initial centroids optimally, adaptive K-Means++ initialization enhances the centroid selection procedure and lessens the effect of unfavorable initial conditions (Arthur & Vassilvitskii, 2007). This method reduces susceptibility to starting conditions and improves cluster formation by ensuring that initial centroids are evenly dispersed throughout the data space (Suraya et al., 2023). Isolation Forest and other outlier identification techniques are useful for locating and eliminating unusual data points that can skew clustering outcomes (Liu et al., 2008). The dataset's integrity is maintained by identifying and eliminating outliers, which enables more precise analysis of crime trends (Li et al., 2019).

By comparing the observed and expected within-cluster sum of squares, the Gap Statistics method offers a methodical way to calculate the ideal number of clusters (k) (Tibshirani et al., 2001). By avoiding arbitrary k selection, this technique produces more trustworthy clustering findings (Tan & Tiangco, 2025). DBSCAN provides density-based clustering that does not require a predetermined number of clusters, in contrast to MacQueen's technique, which is more effective than hierarchical clustering and k-medoids for big datasets. The MacQueen algorithm for detecting various crime patterns in urban settings can be greatly improved with additions like Adaptive K-Means++ initialization, Isolation Forest for outlier detection, and Gap Statistics for optimal k determination (Arthur & Vassilvitskii, 2007; Liu et al., 2008). By increasing clustering accuracy and dependability, these improvements facilitate better decision-making and offer insightful information on criminal patterns. By combining these cutting-edge methods, the improved clustering approach provides solid insights into crime trends, facilitating better urban planning and law enforcement decision-making.

Chapter Three

DESIGN AND METHODOLOGY

This chapter outlines the methodology for improving MacQueen's algorithm for efficient crime clustering. It includes the research design that directed the approach, the system architecture and proposed algorithm that describe the workflow, the system requirements required for execution, and the methods and tools used to accomplish the objectives of the study.

3.1 Research Design

In order to improve MacQueen's k-means algorithm for crime pattern analysis, this study uses a quantitative and experimental research approach. Data preprocessing and location-based clustering are the first main step of the research, which is followed by clustering within each location cluster to identify particular types of crimes. This approach combines the gap statistics method to find the ideal number of clusters, isolation forest for outlier detection, and adaptive k-means++ initialization.

This study's requirement focuses on the key elements required to improve MacQueen's k-means algorithm clustering result that may be used for crime pattern analysis. Comprehensive crime data collection, including thorough records of crime incidents grouped by region names and types, is the main requirement (Tan, 2018). By using this data as the basis for the clustering process, a more sophisticated understanding of the temporal and spatial dynamics of crime will be possible (Dodge, 2008).

Preprocessing processes are essential to enabling efficient data analysis. To handle differences in naming standards and guarantee data consistency, Fuzzy Wuzzy must be used to group comparable region names (Li et al., 2019). Furthermore, categorical area names will be transformed into numerical values through the use of label encoding, producing an "area code" that may be applied to clustering techniques (Barai & Dey, 2017). The adaptive k-means++ initialization approach, which uses numerical input to optimize centroid placement, depends on this numerical representation (Suraya et al., 2023).

In order to find and eliminate outliers from the dataset, the study also requires an anomaly detection technique. By preventing outliers from distorting centroid calculations or compromising clustering accuracy, the use of an Isolation Forest technique will strengthen the clustering process' resilience (Romanuke et al., 2023).

Lastly, a crucial prerequisite for a successful analysis is figuring out the ideal number of clusters. Following each clustering phase, the gap statistics method will be used to calculate the ideal number of clusters (k) (Umargano et al., 2019). Using a null reference distribution, this approach contrasts the total intracluster variation for various values of k with their predicted values (Agarwal et al., 2013). All things taken into consideration, this requirement analysis lists the essential components required to properly use a two-tiered clustering technique that improves the clustering result of the algorithm.

3.1.1 Data Acquisition

The dataset used in this study is a thorough U.S. crime dataset that was obtained from Kaggle and covers crime episodes from 2022 to the present. This dataset was chosen mainly because the particular crime data that was initially meant for study was unavailable, therefore it was a good substitute for testing. The study's focus on using clustering algorithms to detect distinct crime patterns is in line with the Kaggle dataset, which contains a variety of criminal occurrence categories sorted by location and type.

Numerous factors relevant to crime analysis are included in the dataset, such as location names and certain sorts of crimes. Implementing the suggested two-tiered clustering approach—in which the first phase groups data according to geographic regions and the second phase concentrates on classifying crimes within those locations—requires this information. The augmented MacQueen's k-means approach can be used practically with this dataset, allowing for the investigation of temporal and spatial changes in crime patterns across several US regions.

This dataset offers an opportunity to validate the technique and algorithms created in this study because it acts as a substitute for the planned analysis in Manila. Despite being based on U.S. data rather than localized Philippine data, the insights gathered from

examining this data will help determine how well the suggested improvements to the clustering algorithm can recognize and distinguish

3.1.2 Data Preprocessing

In order to properly analyze and cluster the dataset, data preparation is an essential step. A number of crucial procedures are used in this phase to guarantee data consistency and quality, which are necessary for getting precise clustering results.

The FuzzyWuzzy library will be utilized to correct differences in spelling and depiction of crime kinds and area names. By using string matching algorithms, this library reduces differences that may result from typographical errors or differing naming standards by identifying and standardizing comparable items (Li et al., 2019). For example, "Downtown" and "Downtwn" will be combined into a single, standardized term. After this standardization procedure, the textual data will be transformed into numerical representations using label encoding. This conversion is essential for clustering algorithms since they need numerical input in order to efficiently calculate the distances between data points. (Barai & Dey, 2017). These new columns, "crime code" for crime kinds and "area code" for area names, will help with the next clustering stages.

Finding and eliminating outliers that can distort the clustering analysis's findings is a crucial part of data preprocessing. The Isolation Forest algorithm will be used to accomplish this. By using random partitioning to isolate observations, this approach is very good at finding anomalies in high-dimensional datasets (Romanuke et al., 2023). Anomalous data points that deviate from anticipated patterns will be found and eliminated from the dataset using the Isolation Forest technique. By preventing outliers from skewing centroid computations or jeopardizing the precision of cluster assignments, this phase improves the clustering process's resilience.

3.1.3 Clustering Process

By combining appropriate methods for outlier detection, optimal cluster determination, and centroid initialization, the enhanced clustering process in this study aims to methodically improve the identification of crime trends.

The Isolation Forest technique is used to detect and remove outliers at the start of the process. By using a tree-based strategy that divides the data randomly, this method successfully finds and isolates anomalous data points that could skew the clustering findings (Liu et al., 2008). The dataset's integrity is maintained by eliminating these outliers prior to clustering, enabling a more precise examination of crime trends.

The Gap Statistics Method is used to get the Optimal Number of Clusters (k) once outliers have been eliminated. By comparing the total intracluster variance for various values of k to their predicted values under a null reference distribution, this technique assesses different clustering outcomes (Tibshirani et al., 2001). This method enables a data-driven determination of k rather than depending on a predetermined number of clusters, guaranteeing that the number of clusters selected best captures the underlying structure of the data.

The process proceeds to Centroid Initialization using the Adaptive K-Means++ technique after the optimal k has been determined. This improved approach improves the spread and placement of the initial centroids by choosing them based on adjusted squared distances from existing points, as opposed to typical approaches that choose them at random (Arthur & Vassilvitskii, 2007). Clustering performance and convergence speed are improved by this careful initialization.

Lastly, the process of clustering is carried out by applying the improved MacQueen's method with these improved centroids. In addition to enhancing cluster cohesion and separation, this methodical technique offers a more accurate depiction of crime trends across Manila's many geographic regions.

3.1.6 Implementation and Analysis

The improved clustering algorithm will be implemented methodically to guarantee thorough examination of crime trends. The Silhouette Score, Davies-Bouldin Index, Standard Deviation Index, and Within-Cluster Sum of Squares (WCSS) are some of the important metrics that will be used to assess the clustering

outcomes. These metrics will enable a thorough evaluation of clustering performance by revealing information about the compactness and separation of clusters.

There will be a comparison between the improved method and the current MacQueen's algorithm. The main focus of this comparison will be how well each algorithm recognizes and distinguishes crime patterns in the dataset. In particular, improvements in cluster cohesiveness and separation will be assessed by comparing how well each strategy performs in obtaining lower WCSS values and higher Silhouette Scores.

Additionally, to show the distribution of data points within each cluster, the study will incorporate visuals such cluster plots. When contrasted to the current approach, these visual representations will improve comprehension of how well the improved algorithm captures crime dynamics. This study intends to offer a comprehensive assessment of the improvements made to the k-means clustering method by looking at both quantitative measurements and qualitative visualizations of the final clustering result of the algorithm.

3.2 Proposed Algorithm and Proposed System Architecture

3.2.1 Proposed Algorithm

The proposed algorithm fixes the issues regarding outlier sensitivity, predefined number of clusters, and centroid initialization. In addition, the researchers applied the isolation forest algorithm for outlier detection and removal, gap statistics for finding the optimal number of clusters (k), and an adaptive k-means++ algorithm for the centroid initialization. Below is the pseudocode of the proposed algorithm:

Enhanced MacQueen's Algorithm:

*1: pre-process the dataset to remove outliers using the **Isolation Forest** algorithm*
*2: determine the optimal number of clusters **k** using the Gap Statistics method.*
*3: initialize **k** centroids using the Adaptive k-means++ approach:*
    a. randomly choose the first centroid.
    b. for each subsequent centroid do
        *i. compute the squared distance **D** of each data point to the nearest existing centroid.*
        *ii. adjust **D** to assign higher probabilities to distant points and lower probabilities to closer points.*
        *iii. select the next centroid with probability proportional to the adjusted distance **D***
*4: **repeat***
*5: **for each** data point **do***
    5: assign the data point to the closest centroid.
    6: recalculate the centroid of each cluster as the mean of all data points assigned to it.
*6: **until** convergence*

To further understand the pseudocode, below is a step-by-step explanation**:**

Step 1: Preprocess the dataset to remove the outliers using the Isolation Forest algorithm. This step identifies and eliminates data points that significantly deviates from the majority. ***(Objective 2)***

Step 2: Calculate the optimal number of clusters (*k*) using the Gap Statistics Method. This technique determines the value of *k* that produces the most compact and well-separated clusters by assessing how well the data points fit within clusters for various values of *k*. ***(Objective 3)***

Step 3: Initialize ***k*** centroids using the adaptive k-means++ approach: ***(Objective 1)***

- 3a: Randomly choose the first centroid from the dataset.
- 3b: For each subsequent centroid:
    - Determine each data point's squared distance (*D*) from the closest existing centroid.

- Modify *D* such that points distant from current centroids have higher probabilities and points closer to them have lower probabilities.
- To improve centroid distribution throughout the data, choose the subsequent centroid based on these adjusted probabilities.

Step 4: Start the clustering iteration.

Step 5: Assign each data point in the dataset to the cluster that the closest centroid represents. This guarantees that every data point is a member of the cluster with the closest distance to the centroid.

Step 6: Recalculate each cluster's centroids. The mean of all data points allocated to a cluster is its new centroid. In order to more accurately depict the cluster's data points, the centroids are adjusted in this stage.

Step 7: Continue until convergence by repeating steps 4 and 5. When the centroids and cluster assignments no longer vary much, convergence is reached, signifying that the clustering process is stable.

This enhanced algorithm ensures that the dataset is well-preprocessed, the centroids are effectively initialized, and the clusters are optimized for maximum separation and compactness, leading to more reliable and meaningful clustering results.

3.2.2 Proposed System Architecture

The figure below represents the proposed system architecture for the enhanced algorithm.

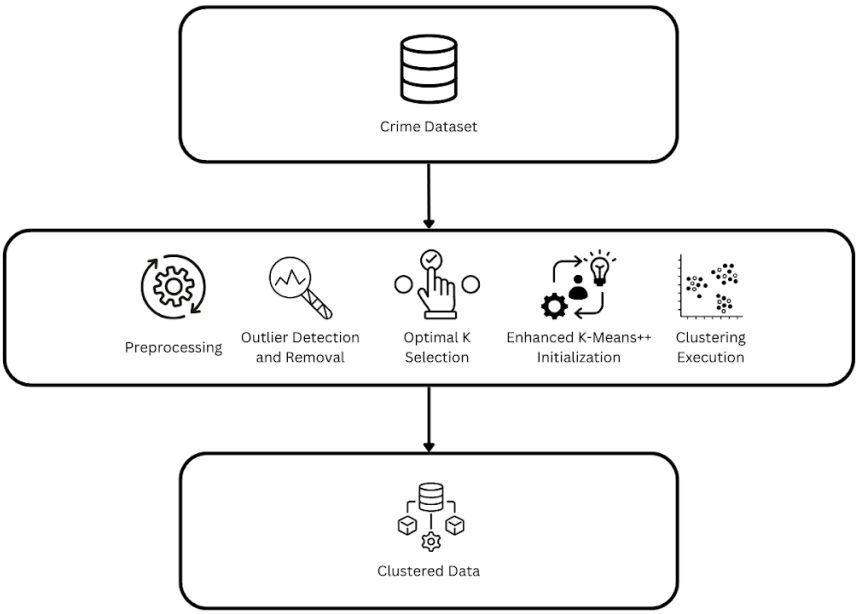

Figure 3.1 System Architecture of the Enhanced MacQueen's Algorithm

The system architecture of the improved Macqueen's algorithm is shown in Figure 3.1. The structure of this system design guarantees efficient data processing and analysis. Raw crime data, such as location names and crime types, is gathered at the start of the procedure. This information forms the basis for further study.

Using fuzzy string matching, the FuzzyWuzzy library is used to standardize region names and crime kinds during the preprocessing stage. By calculating the differences between sequences using the Levenshtein distance, this method allows the program to identify naming convention changes (SeatGeek, 2014). The standardized textual data is then transformed into numerical values by label encoding, which makes it easier to use them in clustering method.

The Isolation Forest technique is then used to detect and remove outliers. By separating observations via random partitioning, this technique finds and eliminates anomalous data points that might skew clustering results (Liu et al., 2008). The integrity of the clustering analysis is maintained by removing outliers from the dataset.

The Gap Statistics Method is then used by the system to get the Optimal Number of Clusters (k). By comparing the total intracluster variation for various values of k against their predicted values under a null reference distribution, this method evaluates different clustering outcomes and aids in determining the optimal number of clusters (Tibshirani et al., 2001).

The Adaptive K-Means++ approach chooses initial centroids using adjusted squared distances and incorporates an Enhanced K-Means++ Initialization. By taking into account the distribution of data points, this adaptation enhances centroid placement and increases clustering effectiveness (Arthur & Vassilvitskii, 2007).

In order to efficiently classify crime incidents, the Clustering Execution step uses the improved MacQueen's algorithm with the revised centroids. Lastly, the output includes clustered data that shows the clustering result, and dataset trends.

3.3 System Requirements

3.3.1 Hardware Requirements

The following hardware requirements are advised in order to effectively implement and carry out the improved MacQueen's clustering algorithm and associated tasks:

Processor:

- Minimum: Intel Core i5 (6th generation or later) or AMD Ryzen 5 equivalent
- Recommended: Intel Core i7 (10th generation or later) or AMD Ryzen 7 equivalent

- Justification: To handle computations for clustering, outlier detection, and large datasets.

RAM:

- Minimum: 8 GB
- Recommended: 16 GB or higher
- Justification: To guarantee that Python libraries and data processing operations run smoothly, particularly for operations like KMeans clustering and Isolation Forest.

Storage:

- Minimum: 256 GB SSD or HDD
- Recommended: 512 GB SSD or higher
- Justification: To store the development environment, libraries, and any datasets used.

3.3.2 Software Requirements

Operating System:

- Minimum: Windows 10, macOS Mojave, or Ubuntu 18.04
- Recommended: Windows 11, macOS Monterey, or Ubuntu 22.04 LTS
- Justification: Compatibility with Python and PyCharm IDE.

Programming Language:

- Python 3.8 or later
- Justification: Required for using libraries such as fuzzywuzzy, sklearn, and others.

Integrated Development Environment (IDE):

- PyCharm (Community or Professional Edition)

- Justification: Offers powerful tools for writing, testing, and debugging Python code.

Python Libraries and Dependencies:

- **folium:** For creating interactive maps.
- **random:** For generating random values, if needed for the algorithm.
- **webbrowser:** For opening results in the default web browser.
- **fuzzywuzzy:** For string matching during preprocessing.
- **sklearn.preprocessing.LabelEncoder:** For converting categorical variables into numerical values.
- **sklearn.metrics.silhouette_score:** For assessing clustering quality.
- **numpy:** For handling numerical computations.
- **sklearn.cluster.KMeans:** For performing k-means clustering.
- **matplotlib.pyplot:** For data visualization and plotting.
- **warnings and sklearn.exceptions.ConvergenceWarning:** For managing runtime warnings during clustering.
- **sklearn.ensemble.IsolationForest:** For outlier detection and removal.

Python Package Manager:

- **pip** (latest version)
- Justification: To manage and install required libraries.

Visualization Tools:

- Matplotlib (via pyplot)
- Folium for map-based visualizations.

3.4 Methods and Tools

3.4.1 Methods

Adaptive K-Means++ Initialization for Centroids Selection

The Adaptive K-Means++ algorithm is an improvement of the classic k-means clustering algorithm that seeks to enhance initialization of centroids. By choosing initial centroids that are evenly dispersed throughout the data space, this approach aims to improve overall clustering performance and convergence time. Because centroids in traditional k-means are frequently selected at random from the dataset, if the initial centroids are not indicative of the distribution of the underlying data, the clustering results may be poor. By selecting centroids in a probabilistic manner, the K-Means++ method solves this problem.

The first centroid from the dataset is chosen at random to start. Specifically, points farther away are chosen with a probability proportional to their squared distance from the closest centroid, with subsequent centroids being chosen depending on their distance from existing centroids (Arthur & Vassilvitskii, 2007). In order to improve cluster formation, this approach makes sure that new centroids are positioned in areas of the data space that are not currently covered by existing centroids.

The ability of Adaptive K-Means++ to modify the selection procedure in response to the distribution of data points is an important aspect. This approach lowers the possibility of poor initialization and enhances the general quality of clusters created during later iterations by introducing distances into the centroid selection procedure (Kleinberg, 2002).

Isolation Forest for Outlier Detection

In high-dimensional environments, the Isolation Forest approach is especially useful for identifying abnormalities in datasets. Isolation Forest builds an ensemble of binary trees to directly separate anomalies, in contrast to conventional anomaly identification techniques that involve profiling normal data points (Liu et al., 2008).

The algorithm works on the premise that anomalies need fewer random partitions to be separated because they are uncommon and different from regular observations. Using a randomly chosen feature and a random split value between the feature's minimum and maximum values, Isolation Forest repeatedly divides the dataset. Because anomalies can be isolated with fewer splits than normal points, they typically have shorter path lengths in isolation trees (iTrees), which are produced by this random partitioning (Hodge & Austin, 2004).

Based on the average path length of all the iTrees in the forest, each data point is given an anomaly score; shorter path lengths suggest a higher probability of being an anomaly. Outliers are pointing whose anomaly score is higher than a predetermined threshold. Because of its minimal memory requirements and linear time complexity, this technique is especially beneficial for large datasets (Liu et al., 2008).

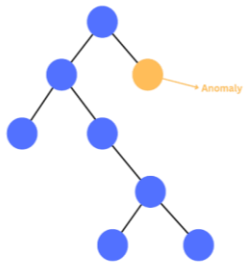

Figure 3.2 Isolation Tree with a Detected Outlier or Anomaly

Figure 3.2 shows an example of an isolation tree with a detected outlier or anomaly. In an isolation tree, a node is deemed an outlier or anomaly if it isolates with fewer random partitions, resulting in a shorter path length from the root node to the leaf node (Liu et al., 2008). In the given visualization, the right child node is considered to be an anomaly.

Gap Statistics Method for Finding Optimal K

The Gap Statistics method compares the total intra cluster variance for various values of k against their predicted values under a null reference distribution in order to identify the ideal number of clusters (k) in a dataset (Tibshirani et al., 2001). This methodology helps prevent arbitrary selection of k and offers a methodical way to validate clusters.

The original dataset is clustered for a range of k values in order to apply Gap Statistics, and the Within-cluster Sum of Squares (WCSS) is computed for each configuration. Clustering is then applied to the reference dataset, which is created by uniformly sampling points throughout the feature space. This reference data's WCSS is calculated for every k value, enabling comparison with the original dataset's WCSS.

For further discussion, below is the formula used for the evaluation metrics:

Silhouette Score S(I)

$$S(i) = \frac{b(i) - a(i)}{max(a(i), b(i))}$$

Where:

- a(i) is the average distance from datapoint i to all other points in the same cluster (intra-cluster distance).
- b(i) is the smallest average distance from datapoint i to all points in any other cluster (inter-cluster distance).

A score around +1 suggests that the data point is well-clustered, whereas a score near -1 suggests that it might be misclassified (Rousseeuw, 1987). The Silhouette Score is a number between -1 and +1. A general indicator of clustering quality is the average Silhouette Score for every point in a dataset (Halkidi et al., 2001).

Davies-Bouldin Index

By calculating the average similarity between each cluster and its most comparable neighbor, the Davies-Bouldin Index assesses the quality of clustering. The Davies-Bouldin Index DB is calculated using the following formula:

$$DB = \frac{1}{K}\sum_{i=1}^{K} max_{j\neq i}\left(\frac{S_i + S_j}{d_{ij}}\right)$$

Where:

- K is the number of clusters.
- S_i is the average distance between points in cluster i (intra-cluster distance).
- d_{ij} is the distance between centroids of clusters i and j.

According to Davies and Bouldin (1979), a lower Davies-Bouldin Index denotes better clustering performance since it shows lower inter-cluster similarity and more intra-cluster similarity. This measure aids in determining how independent and well-separated the clusters are from one another

Standard Deviation Index

The distribution of data points inside each cluster is measured by the Standard Deviation Index. For cluster k, the standard deviation SD_k can be computed using:

$$SD_k = \sqrt{\frac{1}{n_k}\sum_{i=1}^{n_k}(x_i - C_k)^2}$$

Where:

- n_k is the number of points in cluster k.
- x_i represents each data point in cluster k.
- C is the centroid of cluster k.

In order to achieve effective clustering, data points should be closely packed within their respective clusters, as indicated by a lower standard deviation (Jain & Dubes, 1988). The degree of closeness between the points inside each cluster is indicated by this statistic.

Within-Cluster Sum of Squares (WCSS)

The degree of clustering of the data points within each cluster is measured by the Within-Cluster Sum of Squares (WCSS). The WCSS calculation formula is:

$$WCSS = \sum_{k=1}^{K} \sum_{i=1}^{n_k} (x_i - C_k)^2$$

Where:

- K is the number of clusters.
- nk is the number of points in cluster k.
- xi represents each data point in cluster k.
- Ck is the centroid of cluster k.

More compact clusters are indicated by a lower WCSS value, which indicates that the clustering algorithm successfully grouped related data points together (Kassambara & Mundt, 2020). Improvements in clustering performance

can be quantitatively demonstrated by comparing the WCSS values of various algorithms.

3.4.4 Tools

The tools used to improve the MacQueen algorithm for crime data clustering fall into three categories: software tools, libraries, and frameworks. By facilitating a variety of tasks such data preprocessing, clustering, visualization, and evaluation, these tools are essential to reaching the study's objectives

Programming Language and IDE

1. **Python (v3.8 or later):**
    - The enhanced MacQueen's algorithm was implemented mostly using this programming language because of its ee of use, versatility, and wide library support.

2. **PyCharm IDE (Community or Professional Edition):**
    - An integrated development environment that facilitates the efficient writing, debugging, and testing of Python code.

Libraries and Frameworks

1. **Data Preprocessing and Encoding**
    - **fuzzywuzzy:** used for string similarity matching to combine crime types and area names that share similar features.
    - **sklearn.preprocessing.LabelEncoder:** creates numerical representations of categorical data for clustering, such as crime types and area names.
2. **Clustering and Algorithm Enhancement**
    - **sklearn.cluster.KMeans:** Groups the data into clusters using the k-means clustering algorithm.

- **sklearn.ensemble.IsolationForest:** Increases the accuracy of clustering by identifying and eliminating outliers from the dataset.

3. **Optimization and Metrics**
   - **sklearn.metrics.silhouette_score:** Helps assess the clusters' performance and measures the quality of clustering.
   - **Gap Statistics Method:** Compares the clustering results with those produced by a random distribution to determine the optimal number of clusters (k).
4. **Numerical and Statistical Computations**
   - **numpy:** Makes it easier to do matrix operations, numerical calculations, and effective data management for clustering tasks.
5. **Visualization**
   - **matplotlib.pyplot:** Used to create data plots for efficient analysis and presentation of clustering results.
   - **folium:** Makes interactive maps to show the geographic results of the clustering process.
6. **Other Utilities**
   - **warnings and sklearn.exceptions.ConvergenceWarning:** During algorithm execution, it controls and suppresses runtime errors.
   - **random:** Produces random values when necessary for testing or initializations.
   - **webbrowser:** Opens the outputs, such as maps, on the web browser of choice.

Chapter Four

RESULTS AND DISCUSSION

Each of the implementations made to the algorithm will all be evaluated using the silhouette score, Davies-Bouldin index, standard deviation index, and within-cluster sum of squares (WCSS). To determine if the enhanced algorithm performs better, it should have a higher silhouette score, a lower Davies-Bouldin index value, a lower standard deviation value, and also a lower within-cluster sum of squares (WCSS) value as discussed in the chapter 3 methodology, specifically in subsection 3.1.5 Evaluation Metrics.

4.1 Implementation of Adaptive K-Means++ Initialization

To determine if the adaptive k-means++ initialization enhances the algorithm's clustering performance, it would be compared to the common k-means++ and also to using random initialization using the evaluation metrics discussed above.

Table 1. Evaluation of Initialization Methods

Pre-defined number of Clusters	Evaluation Metrics	Random Initialization	K-Means++ Initialization	Adaptive K-Means++ Initialization
k = 10	Silhouette Score	0.6896920736476028	0.7024288327198466	0.7101037203872295
	Standard Deviation Index	5.381217429848347 5	5.6113213359571805	5.5370500884713465
	Davies-Bouldin Index	0.4063053069093626	0.4501748545825477 5	0.4431322528411936
	WCSS	122.86694226185747	116.40012246088365	103.59778787760989

	Silhouette Score	0.7928586807171879	0.7806690242074616	0.803377862650523

k = 13	**Standard Deviation Index**	5.407849536489751	5.5400023119261235	4.926867524534171
	Davies-Bouldin Index	0.3203847198760717	0.2917306234116468	0.3181253304064106
	WCSS	66.100019022534438	68.25219293557787	67.77906593406594
k = 15	**Silhouette Score**	0.8374609909947783	0.8343612585928268	0.8567344968716398
	Standard Deviation Index	5.2973135608234685	5.688363958826014	5.500978595739865
	Davies-Bouldin Index	0.3161470242950871	0.23549498794431012	0.26695420110814455
	WCSS	50.780827067669165	44.68001902253439	41.96831107619795

Table 1 presents the evaluation metrics for various clustering techniques applied to datasets with predefined cluster counts of 10, 13, and 15. The metrics used are Silhouette Score, Standard Deviation Index, Davies-Bouldin Index, and WCSS. The results vary across different techniques, highlighting that the effectiveness of clustering can significantly depend on the initialization method and the number of clusters. It is also important to note that no outlier detection or removal methods were applied in this analysis.

For **k = 10**, the Adaptive K-Means++ method performs best in terms of Silhouette Score, indicating well-defined and well-separated clusters, and a lower Davies-Bouldin Index, indicating more distinct clusters. The method also presented a better Standard Deviation Index, having the lowest score, suggesting less spread among data points within

its centroid, and a more compact cluster. Moreover, it achieves the lowest WCSS, implying that the data points within clusters are highly similar.

For **k = 13** and **k = 15**, the Adaptive K-Means++ method consistently achieves the highest Silhouette Score among the initialization methods. It also performs best in the Standard Deviation Index and ranks second for both WCSS and Davies-Bouldin Index for both k = 13 and k = 15 clusters.

Overall, the Adaptive K-Means++ method consistently provides the best clustering results among the three initialization methods.

4.2 Implementation of Isolation Forest for Outlier Detection and Removal

Table 2. Performance of the Enhanced Algorithm w/o Isolation Forest (with Random Initialization)

Pre-defined number of Clusters	Evaluation Metrics	Without Isolation Forest	With Isolation Forest
k = 10	Silhouette Score	0.7086008801156014	0.70682234565317
	Standard Deviation Index	5.488870322982122	5.417965741005289
	Davies-Bouldin Index	0.43127774884187103	0.4458523856299042
	WCSS	116.28074581606606	105.62211241191633
k = 13	Silhouette Score	0.7841207697587248	0.7936161536341461
	Standard Deviation Index	5.291697029685167	5.291697029685167
	Davies-Bouldin Index	0.3597371685777476	0.3597371685777476
	WCSS	62.01273182957393	62.01273182957393
k = 15	Silhouette Score	0.8106811715307698	0.8255277079815544
	Standard Deviation Index	5.125416563956692	5.012401149889637
	Davies-Bouldin Index	0.24920093186429015	0.23179358507262454
	WCSS	41.30714285714285	43.952142857142846

Noise Level = 0.1

Table 2 presents the algorithm's performance results with and without outliers in the dataset. The clustering was performed using random initialization, with a noise level of 0.1, across three predefined number of clusters, **k = 10, k = 13, k = 15**.

For **k = 10**, the application of the Isolation Forest to remove outliers led to improvements in the Standard Deviation Index, Davies-Bouldin Index, and WCSS. These enhancements indicate more distinct clusters and greater similarity among data points within each cluster, resulting in better cluster compactness and separation.

At **k = 13**, the **Silhouette Score** showed a significant increase after outlier removal, suggesting that random initialization without outliers produces more well-defined clusters compared to clustering with outliers. However, the Standard Deviation Index, Davies-Bouldin Index, and WCSS remained unchanged, indicating that outlier removal had a limited impact on intra-cluster variance and compactness at this cluster level.

Lastly, at **k = 15**, the removal of outliers using the Isolation Forest significantly improved the Silhouette Score and reduced the Davies-Bouldin Index, highlighting better-defined and more compact clusters. While the Standard Deviation Index also showed a slight decrease, indicating reduced variability within clusters, the WCSS increased slightly, which implies that the data points within clusters are less likely to be similar

Table 3. Comparison of the Performance of Random and Adaptive K-Means++ Initialization with Isolation Forest

Pre-defined number of Clusters	Evaluation Metrics	Random Initialization	Adaptive K-Means++ Initialization
k = 10	Silhouette Score	0.70682234565317	0.7147744217633439
	Standard Deviation Index	5.417965741005289	5.628917575342142
	Davies-Bouldin Index	0.4458523856299042	0.4801601712907588
	WCSS	105.62211241191633	100.51207359189561

k = 13	**Silhouette Score**	0.7936161536341461	0.783143926628264
	Standard Deviation Index	5.291697029685167	5.4156001207476345
	Davies-Bouldin Index	0.3597371685777476	0.3700745302285054
	WCSS	62.01273182957393	58.8511531171979
k = 15	**Silhouette Score**	0.8255277079815544	0.845632312827737
	Standard Deviation Index	5.012401149889637	5.110692877308516
	Davies-Bouldin Index	0.2317935850726245 4	0.2749464336755063 6
	WCSS	43.952142857142846	39.55293650793651

Noise Level = 0.1

Table 3 presents the algorithm's performance results with and without outliers in the dataset. The clustering was performed using adaptive k-means ++ initialization with a noise level of 0.1, across three predefined number of clusters, **k = 10, k = 13, k = 15**.

Across all clusters, the Silhouette Score and WCSS are slightly better for Adaptive K-Means++ compared to Random Initialization. This indicates that Adaptive K-Means++ produces better-defined clusters, with data points within clusters being highly similar. However, for the Standard Deviation Index and Davies-Bouldin Index, Random Initialization yields better results than Adaptive K-Means++, suggesting more consistent and compact clustering.

4.3 Implementation of Gap Statistics Method for Finding the Optimal K

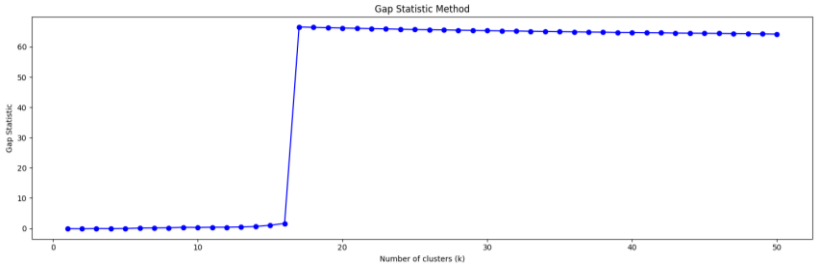

Figure 4.1 Gap Statistics Method

Figure 4.1 shows the result of the Gap Statistic method for finding the optimal number of clusters (k) in a dataset. The x-axis represents the number of clusters, while the y-axis measures clustering quality.

The blue line indicates the Gap Statistic values for different cluster numbers. Based on the figure, the optimal number of clusters (k) is 17, identified where the Gap Statistic reaches its maximum or stabilizes.

a) **Steep Increase:** From k = 1 to k = 17, there is a significant increase in the Gap Statistic, indicating improved clustering quality.
b) **Plateau Behavior:** After k = 17, the Gap Statistic becomes constant, suggesting no further improvement in clustering.

The stabilization at k=17 implies that increasing the number of clusters beyond this point doesn't improve clustering, hence it is chosen as the optimal number of clusters.

Table 4 *Comparison of the Performance of the Enhanced Algorithm Using Gap Statistics Method vs. Pre-defined Number of Clusters (k)*

Evaluation Metrics	k = 10	k = 13	k = 15	Using Gap Statistics Method (Optimal k = 17)
Silhouette Score	0.6896920736476028	0.7928586807171879	0.837460990994783	0.8540378641740372
Standard Deviation Index	5.3812174298483475	5.407849536489751	5.2973135608234685	5.643606932898182
Davies-Bouldin Index	0.4063053069093626	0.3203847198760717	0.3161470242950871	0.2102934590811591
WCSS	122.86694226185747	66.10001902253438	50.780827067669165	18.743589743589745

Table 4 shows the difference between the different number of clusters, and the optimal number of clusters. The number of clusters are as follows: 10, 13, 15,and 17 - which is the optimal number of clusters obtained through Gap Statistics Method.

Between the 4 different numbers of clusters, k = 17 yields the best results. 17 clusters has the best Silhouette Score with a result of 0.856, this indicates that among all numbers of clusters, 17 clusters has the better-defined clusters compared to the other 3 clusters. Similarly, the optimal cluster yielded the best result with the Davies-Bouldin Index and WCSS, which indicates a well-defined cluster, with data points within clusters being highly similar. However, the Standard Deviation Index has the worst result for 17 clusters. This implies that the spread of data points within each cluster are widely dispersed from the cluster's centroid.

Overall, comparing these 4 numbers of clusters, the optimal number of clusters, k = 17, yields the best result for three metrics, WCSS, Silhouette Score, and Davies-Bouldin Index. This implies that identifying the optimal number of clusters using Gap Statistics method results in a more compact, well-defined, cluster with the data points being highly similar with each other.

4.4 Comparison Between Existing and Enhanced Macqueen's Algorithm

Table 5 *Evaluation Metrics of Existing and Enhanced Macqueen's Algorithm*

Evaluation Metrics	Existing Macqueen's Algorithm	Enhanced Macqueen's Algorithm
Silhouette Score	0.844706272071473	0.9608623138340434
Standard Deviation Index	5.840810824799901	5.554063348626442
Davies-Bouldin Index	0.20157669680004545	0.13311464072628645
WCSS	19.15977742448331	7.875

Noise Level = 0.1

Table 5 shows the differences between the enhanced Macqueen's Algorithm and the existing version across various evaluation metrics at a noise level of 0.1. The enhanced algorithm achieved better overall results compared to the existing algorithm, including a higher Silhouette Score, indicating better-defined clusters, a lower Standard Deviation Index, implying less spread and more compact clusters, a lower Davies-Bouldin Index, reflecting more distinct and well separated clusters, and a significantly lower WCSS, indicating that data points within clusters are highly similar.

Overall, an improvement can be observed with the enhanced algorithm based on the 4 metrics. These improvements indicate that the enhanced algorithm is more effective than the existing when it comes to handling noise, and identifying patterns within the dataset resulting with better clustering results.

4.5 Comparison to Other K-Means Algorithms

Table 6. Comparison Between Enhanced Macqueen's Algorithm and Other K-Means Clustering Algorithms

Evaluation Metrics	Lloyd Algorithm	Elkan Algorithm	Enhanced Macqueen's Algorithm
Silhouette Score	0.8235276069304025	0.8351792394617641	0.9608623138340434
Standard Deviation Index	5.791170527808716	5.777699652536128	5.554063348626442
Davies-Bouldin Index	0.19714722482711433	0.20197528654980543	0.13311464072628645
WCSS	19.759398496240593	20.02930402930403	7.875

Noise Level = 0.1

Table 6 compares the performance of the Enhanced Macqueen's Algorithm with the Lloyd and Elkan algorithms across various evaluation metrics at a noise level of 0.1. The Enhanced Macqueen's Algorithm consistently outperforms the other two algorithms. It achieves the highest Silhouette Score of 0.9608, indicating better-defined clusters, compared to 0.8235 for the Lloyd Algorithm and 0.8352 for the Elkan Algorithm.

Additionally, the Enhanced Macqueen's Algorithm records the lowest Standard Deviation Index (5.55), implying fewer spread data points and more compact clusters, while Lloyd and Elkan algorithms have slightly higher values of 5.79 and 5.78, respectively.

For the Davies-Bouldin Index, the Enhanced MacQueen's Algorithm again performs best with a value of 0.1331, indicating more distinct clusters, while the Lloyd and Elkan algorithms have values of 0.1971 and 0.2019. Finally, the Enhanced MacQueen's Algorithm shows a much lower Within-Cluster Sum of Squares (WCSS) of 7.875, meaning its clusters are more compact, while Lloyd and Elkan have WCSS values of 19.75 and 20.02, respectively.

These results demonstrate that the Enhanced Macqueen's Algorithm offers superior clustering performance, characterized by better-defined, more distinct, and tighter clusters compared to the traditional K-means approaches.

Chapter Five

CONCLUSION AND RECOMMENDATION

This chapter offers the conclusion of the paper based on the findings of the methods and experiments - and the recommendations for future implementations.

5.1 Conclusion

- The integration of Adaptive K-Means++ initialization improved the algorithm's ability to select initial centroids, resulting in better-defined clusters with higher cohesion and separation. This improvement was reflected in the consistently higher Silhouette Scores and lower Davies-Bouldin Index values compared to the traditional algorithm.
- Additionally, the use of the Gap Statistics method to determine the optimal number of clusters ensured that the clustering process accurately reflected the data, leading to a significant reduction in Within-Cluster Sum of Squares (WCSS). Unlike the traditional approach of using a predetermined number of clusters, which relies on assumptions and may lead to poor clustering results, the Gap Statistics method provides a data-driven approach to identify the optimal cluster count. Pre-determined clusters often result in either underfitting or overfitting, where clusters are either too broad or too fragmented, failing to capture the true distribution of the data.
- In contrast, using Gap Statistics to find the optimal number of clusters lets the algorithm adjust to the data more effectively, resulting in clusters that are more compact, well-separated, and consistent. This approach concludes that with the use of the gap statistics method, the clustering process becomes more accurate by having the optimal number of clusters.

The comparison between the enhanced MacQueen's Algorithm and the existing version demonstrated a significant improvement in overall clustering performance. The application of Isolation Forest for outlier detection and removal effectively minimized the impact of outlier data points, leading to more accurate and better cluster formation.

The study concludes that this approach to handling outliers addresses the sensitivity of the original MacQueen's Algorithm to outliers, and leads to a better result of clustering accuracy.

5.2 Recommendations

Further research is advised to implement an improved centroid update mechanism in MacQueen's Algorithm. This could involve combining the current incremental updates with batch processing to allow for better adjustment of centroids across iterations. Additionally, introducing a self-adaptive learning rate for centroid movement could improve the algorithm's performance by adjusting the step size based on the data and the clustering process.

Additionally, it is also suggested to integrate AI techniques, specifically reinforcement learning (RL), to dynamically adjust clustering parameters during the algorithm's execution. With RL, the algorithm could learn from its own clustering results, continually optimizing centroid placement, cluster formation, and the number of clusters based on feedback from performance metrics like the Silhouette Score and Davies-Bouldin Index. This would help make the algorithm more adaptive and efficient in different clustering tasks.

Lastly, examining the enhanced algorithm across different types of datasets and clustering scenarios to see how well it performs in various situations. Using data with different characteristics, such as high-dimensional, sparse, or noisy data, would reveal areas where the algorithm could be improved.

REFERENCES

[1] Agarwal, A., Gupta, R., & Singh, A. (2013). Data Mining Techniques: A Survey Paper. *International Journal of Computer Applications*, 81(14), 1-5.

[2] Agarwal, Nagpal, & Sehgal. (2013). *Crime Analysis using K-Means Clustering* [Thesis].

[3] Agarwal, R., Gupta, S. C., & Gupta, S. (2013). A Review on Clustering Techniques. International Journal of Advanced Research in Computer Science and Software Engineering, 2(3), 1-8.

[4] Aggarwal, C. C. (2017). *Outlier Analysis* (2nd ed.). Springer. https://doi.org/10.1007/978-3-319-47578-3

[5] Arthur, D., & Vassilvitskii, S. (2007). K-means++: The advantages of careful seeding. *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 1027–1035.

[6] Barai, S., & Dey, S. (2017). A Review on Outlier Detection Techniques in Data Mining. *International Journal of Computer Applications*, 175(8), 1-5.

[7] Borlea, I. D., Precup, R.-E., & Daragan, F. (n.d.). (PDF) centroid update approach to K-means clustering. https://www.researchgate.net/publication/321502735_Centroid_Update_Approach_to_K-Means_Clustering

[8] Borrohou, S., Fissoune, R., & Badir, H. (2023). Data cleaning survey and challenges - improving outlier detection algorithm in machine learning. *J. Smart Cities Soc.* https://www.semanticscholar.org/paper/f8fb151092680faa12582e29d8dabcb6046a13c6

[9] Celebi, M. E., Kingravi, H. A., & Vela, P. A. (2013). *A comparative study of efficient initialization methods for the k-means clustering algorithm*. Expert Systems with Applications, 40(1), 200–210. https://doi.org/10.1016/j.eswa.2012.07.021

[10] Davies, D. L., & Bouldin, D. W. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2), 224-227.

[11] Dodge, M. (2008). *Understanding Crime Patterns: A Review of Spatial and Temporal Analysis*. *Journal of Urban Studies*, 45(6), 1203-1221.

[12] Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (pp. 226-231).

[13] Fabrigas, A., Gerardo, B., & Tanguilig, B. (n.d.). (PDF) enhanced initial centroids for K-means algorithm. https://www.researchgate.net/publication/312924089_Enhanced_Initial_Centroids_for_K-means_Algorithm

[14] Grubesic, T. H. (2006). On the application of fuzzy clustering for crime hot spot detection. *Journal of Quantitative Criminology*, *22*(1), 77–105. https://doi.org/10.1007/s10940-005-9003-6

[15] Halkidi, M., Batistakis, Y., & Karypis, G. (2001). Cluster Validity Methods: A Comparative Study. *Computer Science Department*, University of Minnesota.

[16] Hariri, S., Kind, M. C., & Brunner, R. J. (2019). Extended Isolation Forest. *IEEE Transactions on Knowledge and Data Engineering*, *33*(4), 1479–1489. https://doi.org/10.1109/TKDE.2019.2947676

[17] Hartigan, J. A., & Wong, M. A. (1979). Algorithm AS 136: A K-Means Clustering Algorithm. Journal of the Royal Statistical Society: Series C (Applied Statistics), 28(1), 100-108.

[18] Hodge, V. J., & Austin, J. (2004). A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22(2), 85-126.

[19] *Hybrid Clustering Algorithms for crime pattern analysis*. (2018, March 1). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/8551120

[20] Jain, A. K., & Dubes, R. C. (1988). Algorithms for Clustering Data. *Prentice Hall*.

[21] Kaufman, L., & Rousseeuw, P. J. (1990). Finding Groups in Data: An Introduction to Cluster Analysis. Wiley-Interscience.

[22] Kassambara, A., & Mundt, F. (2020). *Factoextra: Extract and Visualize the Results of Multivariate Data Analyses*. R package version 1.0.7.

[23] Karimi, M., & Farhad, S. G. (2020, August 1). *An Improved K-Means with Artificial Bee Colony Algorithm for Clustering Crimes*. https://sanad.iau.ir/journal/acr/Article/676638?jid=676638

[24] Kleinberg, J. (2002). An impossibility theorem for clustering. *Advances in Neural Information Processing Systems*, 15.

[25] Krishna, K., & Murty, M. N. (1999). *Genetic k-means algorithm*. IEEE Transactions on Systems, Man, and Cybernetics, 29(3), 433–439. https://doi.org/10.1109/3468.768210

[26] Li, L., Zhang, Y., & Wang, J. (2019). Handling Outliers in Clustering: A Review. *IEEE Transactions on Knowledge and Data Engineering*, 31(4), 743-757.

[27] Li, L., et al. (2019). Outlier Detection in Clustering Analysis. Journal of Intelligent Information Systems, 56(2), 267-285.

[28] Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008). Isolation Forest. *Proceedings of the 2008 IEEE International Conference on Data Mining*, 413–422. https://doi.org/10.1109/ICDM.2008.17

[29] Lloyd, S. P. (1982). *Least squares quantization in PCM*. IEEE Transactions on Information Theory, 28(2), 129–137. https://doi.org/10.1109/TIT.1982.1056489

[30] MacQueen, J. B. (1967). *Some methods for classification and analysis of multivariate observations*. Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1, 281–297.

[31] Murray, A. T., & Grubesic, T. H. (n.d.). Exploring spatial patterns of crime using non-hierarchical ... https://www.researchgate.net/publication/302497365_Exploring_Spatial_Patterns_of_Crime_Using_Non-hierarchical_Cluster_Analysis

[32] Olukanmi, & Twala. (2017). *K-Means-Sharp: Modified Centroid update for outlier-robust K-means clustering*.

[33] PDP-2023-2028.pdf - - philippine development plan - neda. (n.d.). https://pdp.neda.gov.ph/wp-content/uploads/2023/01/PDP-2023-2028.pdf

[34] Pokhriyal, Kumar, & Verma. (2020). *Survey on Crime Data Analysis Using a Different Approach of K-Means Clustering*. https://www.researchgate.net/profile/Rohan-Verma-7/publication/349718679_Survey_on_Crime_Data_Analysis_Using_a_Different_Approach_of_K-Means_Clustering/links/603e5c864585154e8c70b89c/Survey-on-Crime-Data-Analysis-Using-a-Different-Approach-of-K-Means-Clustering.pdf

[35] Romanuke, V. (2023). Speedup of the k-Means Algorithm for Partitioning Large Datasets of Flat Points by a Preliminary Partition and Selecting Initial Centroids. Applied Computer Systems. https://www.semanticscholar.org/paper/63b3b34f10793f7098986ff460a5d1192457d793

[36] Romanuke, V., Hryshchenko, S., & Shyshkina, O. (2023). Challenges in K-Means Clustering with Large Datasets: Strategies for Improvement. *Journal of Data Science*, 21(2), 145-158.

[37] Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20, 53-65.

[38] Rubio, M. C., et al. (2018). Spatial and Temporal Patterns of Crime in Urban Areas. Journal of Urban Planning and Development, 144(2), 04018015.

[39] Saleh, & Khan. (2019). *Crime data analysis in Python using K - means clustering* [Thesis].

[40] Suraya, S., Rahman, M., & Ahmad, N. (2023). Enhancing Clustering Evaluation Metrics: The Role of Initialization Parameters. *International Journal of Data Science*, 10(1), 33-49.

[41] SeatGeek. (2014). FuzzyWuzzy - PyPI. Retrieved from FuzzyWuzzy

[42] Suraya, S., Sholeh, M., & Lestari, U. (2023). Evaluation of Data Clustering Accuracy using K-Means Algorithm. International Journal of Multidisciplinary Approach Research and Science. https://www.semanticscholar.org/paper/45f15ebf38d17516ba0adfb96863bbd578cc1ca0

[43] Suraya, S., et al. (2023). Evaluation of Clustering Algorithms with Different Initialization Methods. Journal of Intelligent Information Systems, 62(1), 1-15.

[44] Tan, P.-N. (2018). *Introduction to Data Mining*. Pearson Education.

[45] Thompson, S., Celebi, M. E., & Buck, K. H. (2019c). Fast color quantization using MacQueen's k-means algorithm. *Journal of Real-time Image Processing*, *17*(5), 1609–1624. https://doi.org/10.1007/s11554-019-00914-6

[46] Tibshirani, R., Walther, G., & Hastie, T. (2001). Estimating the Number of Clusters in a Dataset via the Gap Statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2), 411-423.

[47] Umargano, M., Kumar, S., & Singh, R. (2019). Determining Optimal Clusters Using Gap Statistics: A Comprehensive Study. *Journal of Computational Statistics*, 34(3), 567-580.

[48] Wohlenberg, J. (2023, April 2). *3 versions of K-Means*. Medium. https://towardsdatascience.com/three-versions-of-k-means-cf939b65f4ea

APPENDICES

APPENDIX A: MacQueen's Algorithm Source Code

SOP 1 & OBJ 1 Source Code:

```
def _adaptive_kmeans_plusplus(
    X, n_clusters, x_squared_norms, sample_weight, random_state, n_local_trials=None
):
    n_samples, n_features = X.shape
    centers = np.empty((n_clusters, n_features), dtype=X.dtype)

    if n_local_trials is None:
        n_local_trials = 2 + int(np.log(n_clusters))

    # Initialize the first centroid randomly
    center_id = random_state.choice(n_samples, p=sample_weight / sample_weight.sum())
    indices = np.full(n_clusters, -1, dtype=np.int32)  # Ensure indices are np.int32
    if sp.issparse(X):
        centers[0] = X[center_id].toarray()
    else:
        centers[0] = X[center_id]
    indices[0] = center_id

    # Initialize list of closest distances and calculate current potential
    closest_dist_sq = _euclidean_distances(
        centers[0, np.newaxis], X, Y_norm_squared=x_squared_norms, squared=True
    )
    current_pot = closest_dist_sq @ sample_weight

    for c in range(1, n_clusters):
        # Compute squared distances to the nearest existing centroid



        dist_sq = np.array([min(np.sum((x - centers[i]) ** 2) for i in range(c)) for x in X],
dtype=np.float64)
        # Adjust distances based on sample weights
        dist_sq *= sample_weight
        # Compute probabilities proportional to adjusted distances
        probs = dist_sq / dist_sq.sum()
        # Choose the next centroid with probability proportional to adjusted distances
        new_center_id = random_state.choice(np.arange(n_samples),
p=probs.astype(np.float64))
        indices[c] = np.int32(new_center_id)  # Ensure the index is np.int32
```

```
        if sp.issparse(X):
            centers[c] = X[new_center_id].toarray()
        else:
            centers[c] = X[new_center_id]

    return centers, indices

# Define the main function to call the adaptive k-means++ initialization
def adaptive_kmeans_plusplus(
    X,
    n_clusters,
    *,
    sample_weight=None,
    x_squared_norms=None,
    random_state=None,
    n_local_trials=None,
):



    random_state = check_random_state(random_state)
    sample_weight = _check_sample_weight(sample_weight, X, dtype=X.dtype)
    if x_squared_norms is None:
        x_squared_norms = row_norms(X, squared=True)
    else:
        x_squared_norms = check_array(x_squared_norms, dtype=X.dtype,
ensure_2d=False)
    centers, indices = _adaptive_kmeans_plusplus(
        X, n_clusters, x_squared_norms, sample_weight, random_state, n_local_trials
    )
    return centers, indices
```

SOP 2 & OBJ 2 Source Code:

```
   outliers = None
    if detect_outliers:
        # Detect and remove outliers using Isolation Forest
        iso_forest = IsolationForest(contamination=0.1, random_state=random_state)
        outliers = iso_forest.fit_predict(X)
        X = X[outliers == 1]  # Keep only non-outliers

    n_samples = X.shape[0]
```

```
    if max_k >= n_samples:
        max_k = n_samples - 1
```

SOP 3 & OBJ 3 Source Code:

```
def gap_statistic(x, kmax=20, b=20):
    """
    Computes the gap statistic for determining the optimal number of clusters.

    Parameters:
    x (numpy.ndarray): The input data for clustering.
    kmax (int, optional): The maximum number of clusters to consider. Default is 20.
    b (int, optional): The number of reference datasets to generate. Default is 20.

    Returns:
    tuple: A tuple containing the gap values for each number of clusters and the optimal
number of clusters.
    """

    gaps = []
    wks = []
    wkbs = []
    for k in range(1, kmax + 1):
        with warnings.catch_warnings():
            warnings.filterwarnings("ignore", category=ConvergenceWarning)
            kmeans = KMeans(n_clusters=k, random_state=42, n_init=27, max_iter=500,
tol=1e-4).fit(x)
            inertia = kmeans.inertia_
            if inertia > 0:
                wk = np.log(inertia)
            else:
                wk = 0  # Handle zero or negative inertia


            wks.append(wk)

            ref_disps = np.zeros(b)
            for i in range(b):
                random_reference = np.random.uniform(np.min(x), np.max(x), x.shape)
```

```
            ref_kmeans = KMeans(n_clusters=k, random_state=42).fit(random_reference)
            ref_inertia = ref_kmeans.inertia_
            if ref_inertia > 0:
                ref_disps[i] = np.log(ref_inertia)
            else:
                ref_disps[i] = 0  # Handle zero or negative inertia

        wkb = np.mean(ref_disps)
        wkbs.append(wkb)
        gaps.append(wkb - wk)

gaps = np.array(gaps)
if len(gaps) > 1:
    optimal_k = np.argmax(gaps[:-1] - gaps[1:] + np.log(kmax)) + 1
else:
    optimal_k = 1  # Default to 1 if gaps array is too small
print(f"Gaps: {gaps}")
print(f"Optimal k (gap statistic): {optimal_k}")
return gaps, optimal_k
```

APPENDIX B: Source Code of PLMAT Program Recommendation System

```
import streamlit as st
from views import crime_hotspots, evaluation_metrics, gap_statistics, algorithm_used,
enhancement_made, compare_crime_statistics, chatbot

st.set_page_config(page_title="Crime Statistics",
page_icon=":chart_with_upwards_trend:", layout="centered")
st.title("☑ Crime Statistics")

tab1, tab2, tab3, tab4, tab5, tab6, tab7 = st.tabs(["Crime Hotspots", "Compare Crime
Statistics", "K-Means Algorithms Comparison", "Gap Statistics",
                                    "Algorithm Used", "Enhancements Made", "Chatbot"])

with tab1:
    crime_hotspots.app()
with tab2:
    compare_crime_statistics.app()
with tab3:
    evaluation_metrics.app()
with tab4:
    gap_statistics.app()
with tab5:
    algorithm_used.app()
with tab6:
    enhancement_made.app()
with tab7:
    chatbot.app()
```

APPENDIX C: Proof of Paper Acceptance for Publication

Research Paper Reviewed

I IJFMR <editor@ijfmr.com>
To TIANGCO, ARWIN BONITA

If there are problems with how this message is displayed, click here to view it in a web browser.
Click here to download pictures. To help protect your privacy, Outlook prevented automatic downloa

Your research paper titled **Enhanced MacQueen's Algorithm for Identifying Diverse Crime**

Review Report	
Review Result	Accepted
Research Paper Id	33212
Criteria	**Points out of 10**
Relevance	7
Scholarly Quality	7
Continuity	9
Use of Theory	9
Novelty and Originality	7
Technical Contents and Correctness	8
Understanding and Illustrations	9
Critical Qualities	8
References	8
Clarity of Conclusions	8
Unique Contents	81%

As a next step of publication, please pay the publication fees.

Total payable amount: **$ 70 (including a Crossref DOI)**.

Publication Fee Structure:

	1 to 4 Authors	5 or More Authors
Indian Authors	â‚¹ 1500 + â‚¹ 270 (18% GST)	â‚¹ 300 per each additional author
Non-Indian Authors	US $ 70	US $ 10 per each additional author

APPENDIX E: Certificate of Presentation

International Journal For Multidisciplinary Research

An International Open Access Peer Reviewed Journal • Impact Factor: 9.24 • E-ISSN: 2582-2160

Certificate of Publication

The editorial board of IJFMR is hereby awarding the certificate of publication to

Kim Emerson M. Tan

in recognition of publication of the paper titled

Enhanced MacQueen's Algorithm for Identifying Diverse Crime Patterns in the City of Manila

Co-Authors: Arwin B. Tiangco, Vivien A. Agustin

Published In: Volume 6, Issue 6 (November-December 2024)

Paper Id: 33212

www.ijfmr.com • editor@ijfmr.com

Editor / Publisher
IJFMR

International Journal For Multidisciplinary Research

An International Open Access Peer Reviewed Journal • Impact Factor: 9.24 • E-ISSN: 2582-2160

Certificate of Publication

The editorial board of IJFMR is hereby awarding the certificate of publication to

Vivien A. Agustin

in recognition of publication of the paper titled

Enhanced MacQueen's Algorithm for Identifying Diverse Crime Patterns in the City of Manila

Co-Authors: Arwin B. Tiangco, Kim Emerson M. Tan

Published In: Volume 6, Issue 6 (November-December 2024)

Paper Id: 33212

www.ijfmr.com • editor@ijfmr.com

Editor / Publisher
IJFMR

APPENDIX E: Bionote

KIM EMERSON M. TAN

09292409653 | tankimemerson05@gmail.com | https://github.com/KimTan021

EXPERIENCE

Conducted an Information Security Assessment on DyipPay Revolution Corporation (a start-up fintech company) | Subject Requirement

Role: Information Security Assessment Lead

- Led the execution of the information security assessment for DyipPay. Assigned roles to the members of the team and distributed the task among them.
- Conducted the assessment using the NIST framework together with the Center for Internet Security Critical Security Controls (CIS Controls) framework.
- Created a strategy and implementation for DyipPay to increase the security of their company and also created the estimated cost for these implementations.

Impact: Provided comprehensive documentation and analysis for DyipPay to start enhancing their company's security.

Developed a Web-Based Academic Admission System for Pamantasan ng Lungsod ng Maynila (PLM) | Subject Requirement

Role: Business Analyst

- Conducted meetings with the PLM Admissions Head and employees to understand their business process and their problems with their current system.
- Gathered the detailed requirements for the academic admission system, which include functional and non-functional requirements.
- Documented and analyzed the requirements to ensure that they are complete, concise, and clear. and communicated with the lead developer on the team.
- Ensured continuous communication with the stakeholders and the development team for any suggestions, problems, or changes.

Impact: Fixed the problems being encountered by the PLM's Admission Office with their system that is expected to expedite their current business process.

Developed a Programming Language called Summoner | Subject Requirement

- Led the creation of the delimiters, transitional diagram, context-free grammar (CFG), first set, follow set, and prediction set for our programming language and compiler.
- Worked on the documentation, which includes the creation of the rules for our programming language (Summoner).

CERTIFICATIONS

- Introduction to Cybersecurity
- Cybersecurity Essentials

SKILLS

BACK END DEVELOPMENT | Express ■ NodeJS
FRONT END DEVELOPMENT | HTML ■ CSS ■ Bootstrap ■ Javascript ■ EJS ■ JQuery
DATABASE | MongoDB ■ MySQL
MISCELLANEOUS | Python ■ Java

EDUCATION

Pamantasan ng Lungsod ng Maynila (PLM) (College)
BS Computer Science | 3rd year
2021 - 2025

Polytechnic University of the Philippines (PUP) (Senior High School)
Science, Technology, Engineering, and Mathematics (STEM)
2019 - 2021

HONORS AND AWARDS

College

- **Consistent President's Lister** from 1st year up until now.
- **Rank 3** and **Rank 2 overall** of **3rd year PLM Computer Science students** during the 1st and 2nd years, respectively.

Senior High School

- With High Honors
- **GWA:** 96.4375

Junior High School

- **Consistent honor student** from **1st year to 4th year**.
- **Rank 2 Overall**

INTERESTS

Development ■ Ethical Hacking ■ Network Engineering ■ Fitness ■ Nutrition ■ Food ■ Travelling ■ Self-Improvement

Arwin B. Tiangco

+63 998 135 5586 · arwintiangco@gmail.com · LinkedIn
157 P. Zamora St., Brgy. 19, Caloocan City

Pamantasan ng Lungsod ng Maynila **September 2021 - Current**
Bachelor of Science in
Computer Science

Technical Skills

- **Project Management:** Agile, Jira, Notion, Trello
- **UI/UX Design:** Framer, Figma, Adobe XD, Adobe Illustrator
- **Platforms & Frameworks**: Bubble.io, Flutter Flow, WordPress
- **Languages & Framework**: HTML, CSS (Tailwind), ReactJS, JavaScript, TypeScript, Python

VOLUNTEER EXPERIENCE

PLM Computer Science Society | Secretariat Committee **September 2022 - July 2023**

- Assisted the organization through communication between partnered organization inside campus.

S.P. Madrid & Associates | Innovations Factory **June 2024 - August 2024**

- Proposed a better user interface for their in-house software (*OpenCI)* to streamline workflows for collection agents and collection drivers.
- Redesigned the OKPO website to better serve businesses interested in its implementation.
- Gained hands-on experience with Framer, Bubble.io, and Flutter Flow for no-code/low-code development and prototyping.

PROJECTS

Admission Module for Centralized ERP | Subject Requirement
Project Manager & Lead UI/UX Designer

- Subject Requirement for 'Software Engineering 1' and 'Software Engineering 2'
- Managed and lead a team of 5 in an AGILE approach to a successful implementation of Admission Module to the centralized ERP.
- Maintained communications with other connected modules, and primary stakeholders resulting in an efficient workflow.
- Assisted on designing the UI and UX of the module.

MazeBank | Personal Project

- **Technologies:** Dart & Google Firebase
- **Framework:** Flutter
- An application that tracks your expenses on a weekly basis.
- This application is an on-going project which serves as my practice to understand further the Flutter framework.

PLM Website | Subject Requirement
Project Manager & Lead UI/UX Designer

- Managed a team to successfully enhance the previous website of the university.
- Maintained communications between the client and other stakeholders, aligning project goals with their needs.
- Designed and optimized the website's UI/UX, ensuring a user-friendly, visually appealing, and accessible interface.