



UIT

**THE ARCTIC
UNIVERSITY
OF NORWAY**

Faculty of Science and Technology

Department of Computer Science

Implementation of Cluster Detection Mechanism of Syndromic Surveillance System in EDMON

Prosper Kandabongee Yeng

INF-3997 Master's Thesis in Telemedicine and E-health-September 2019



Implementation of Cluster Detection Mechanism of Syndromic Surveillance System In EDMON

PROSPER KANDABONGEE YENG

INF-3997

Faculty of Science and Technology, Department of Computer Science

Master's Thesis in Telemedicine and E-health

September, 2019

ABSTRACT

Background

Early detection of disease outbreak has become a global challenge because existing disease surveillance systems, ostensibly, appears not to be efficient enough. As a result, there still exists disease outbreaks such as Ebola, heatwaves, malaria and flu with high case fatality rates in some parts of the world. New disease surveillance methods are therefore being explored to enhance the disease outbreak detection capabilities for timely interventions. For this reason, Electronic Disease Monitoring Network (EDMON) was initiated. EDMON is an ongoing research in syndromic surveillance at University of Tromsø, The Arctic University of Norway. The broad goal of this project is to detect the spread of contagious diseases at the earliest possible moment, and potentially before people know that they have been infected thus as early as the incubation stage of infection.

The results shall be visualized on real-time maps as well as presented in digital communication. The project uses self-recorded health related data from people with type-1 diabetes as input. The problem is that most syndromic surveillance systems do not detect disease outbreak as early enough. They detect outbreaks during or after visible symptoms stage of the infection which results in higher time lag. Therefore, health management is unable to manage the outbreaks early enough and this often lead to high disease burden.

Appropriate algorithms were explored through systematic review towards the implementation of a cluster detection mechanism in EDMON. In this study, a Hybrid of K-nearest Neighbour (KNN) and Cumulative Summation (CUSUM) known as EDMON-Cluster, were proposed and explored to assess the dual combination ability to augment for the gap of loss of power to detect outbreaks in a geographically disaggregated data.

Objective

The main aim of EDMON-cluster was to implement and assess clustering methods of detecting infectious disease outbreak in EDMON. Specifically, spatial and temporal algorithms were hybridized in the implementation and their performance of detection such as sensitivity, specificity and timeliness were evaluated. Various challenges such as privacy and security, geographical location estimation and visualization were considered.

Materials and Methods

Synthetic or simulated data was generated to consist of required parameters such as infected Individuals' detections, geolocations and respective time stamp of occurrences. Synthetic dataset of

geolocations of centroid of post codes was also generated. K-nearest neighbour spatial classifier was used to cluster the detected infected Individuals into various centroid of post code areas. This was based on proximity of distance between geolocation of detected individuals and centroid of post codes of near neighbours. Cumulative summation (CUSUM) was then used to implement the temporal aspect of the clustering. A vertical baseline data of an average of one week was used to compare to a week's scanning window. Z-score was used for thresholding while prototyping was adopted in the entire study. The performance of the KNN algorithm was assessed by determining the proportion of infections which were accurately classified. The Sensitivity, Specificity of the CUSUM method were also evaluated by varying the input data through injection of outbreak spikes at various times.

Results

The KNN algorithm, which was implemented in the EDMON-Cluster, recorded 99.52% accuracy when it was evaluated with simulated dataset containing geolocation coordinates among other features and ScikitLearn KNN algorithm achieves an accuracy of 93.81% when it was tested with the same dataset. After injection of spikes of known outbreaks in the simulated data, the CUSUM module was totally specific and sensitive by correctly identifying all outbreaks and non-outbreak clusters. Indication of outbreaks on visual maps and through alarm and SMS alerts were successful. The entire process was estimated to be 12.5 minutes with the simulated data. One-way hashing and deidentification were some of the data anonymization techniques which were adopted in the study to obscure privacy as recommended by the General Data Protection Regulation (GDPR).

CONCLUSION

Basically, KNN and CUSUM algorithms were fused together as a spatiotemporal measure known as EDMON- Cluster. A prototype approach was adopted with synthetic data. With reference to the outstanding performance of the EDMON- Cluster, there is enormous motivation to further evaluate the dual paired algorithms with real dataset towards empirical implementation in EDMON. EDMON-Cluster exhibited a potentially useful method in comparison with other surveillance methods which can further be assessed with real data for practical implementation in EDMON. Suitable methods for obtaining a balance point of anonymizing geolocation attributes towards obscuring the privacy and confidentiality of diabetes subjects while maintaining the data requirements for public good, disease surveillance, remains a challenge.

PREFACE

Having worked as Management Information Systems (MIS) Manager and Regional ICT coordinator for National Health Insurance Authority of Ghana, my results were mostly supported with research evidence. This drew attention to my interest in research. After pursuing MSc in Information and Network Security in the United Kingdom, I decided to follow my dream of becoming a researcher in healthcare by pursuing a second masters in Telemedicine and eHealth in 2017. In my pursuance of this course, the visibility of my dream was still fuzzy until I met Ashenafi Zebene Weldearegay and Professor Gunnar Hartvigsen through the guidance of my wife who was a former student of Professor Gunnar Hartvigsen.

I discussed my research ambition with Gunnar and Ashenafi and asked for their guidance in selecting courses that would enable me to become a researcher. In fact, I felt welcomed as I pursued a capstone project under their guidance and supervision. Formal research process under the capstone project was initially challenging since I was not familiar with systematic review approaches. But my supervisors guided and encouraged me through this slippery road.

My capstone project was a systematic review and the development of a framework towards implementing cluster detection mechanism in EDMON. Through the grace of God Almighty, perseverance and the relentlessness of my supervisors, the project was completed. A scientific paper was published through Scandinavian Health Informatic Conference of 2018 and a journal paper was also extended and submitted to Journal of Medical and Internet Research.

My current research for the master's thesis work is an extension of the capstone project which involves implementing and evaluating a prototype of a cluster detection mechanism in EDMON. The entire development is majorly guided by the results of the capstone project work.

Aside the fact that I have learned better approach to scientific research, I gained better knowledge including how to apply for PhD jobs. The evidence is my success in gaining a PhD job in Healthcare Security Practice Analysis, Modelling and Incentivization with the Norwegian University of Science

and Technology while in the process to complete this current master's degree. About four months into the PhD work, two scientific papers have been accepted. Observational measures for profiling healthcare staff security practices was accepted by IEEE Conference (COMPSAC 2019) in Wisconsin, USA and Healthcare Staffs' Information Security Practices Towards Mitigating Data Breaches in psycho-socio-cultural context was also accepted by pHealth conference 2019 in Italy.

I thank God Almighty, who linked me to my supervisors and provided them with the resilience to support me in spite of my possible weaknesses. I want to also greatly thank professor Gunnar Hartvigsen who transformed me with better research skills and provided me with tremendous ideas to becoming a researcher. Ashenafi is also highly appreciated through his motivation by exemplary hard work and by poking me out of my comfort zone to roll my sleeve to gain these research experiences. My appreciation also goes to my wife, Portia Eren-muo, who always push me to align my efforts towards my dreams. Last but not the least, I wish to extend my sincere appreciation to the Department of Computer Science department of the University of Tromsø - The Arctic University of Norway, who sponsored all these research work publications and conferences.

Gjøvik, September 16, 2019

Prosper Kandabongee Yeng

TABLE OF CONTENTS

TABLE OF CONTENTS.....	v
LIST OF FIGURES	vii
LIST OF TABLES.....	ix
ACRONYMS.....	x
CHAPTER 1: INTRODUCTION	1
1.0 Background and Motivation	1
1.2 EDMON Background	3
1.4 Objective.....	6
1.5 Justification of the study	6
1.6 Scope of the system evaluation.....	8
1.7 Assumptions, Biases and Limitations	9
1.8. Organization.....	9
CHAPTER 2: THOERITICAL FRAMEWORK AND STATE OF THE ART	11
2.1 Introduction.....	11
2.1.1 Terminologies, Preliminaries and Definitions	11
2.2 Disease Surveillance	11
2.3 Clustering.....	12
CHAPTER THREE: Literature Review	24

3.1 Literature Review	24
3.2 Inclusion and Exclusion Criteria.....	24
3.3 Data Collection and Categorization	25
3.4 Literature Evaluation and Analysis.....	26
3.5 Principal Findings and Discussion.....	26
Chapter 4: Materials and Methods.....	31
4.1 Introduction.....	31
4.2 Materials Used	31
4.3 Methods Used	36
4.4 Justification and Critique of the methods	39
Chapter Five: System Analysis.....	42
5.1 Introduction.....	42
5.2 System Description.....	42
5.3 Requirement gathering and analysis	43
5.4 Source of Requirements.....	45
5.5 Functional Requirement.....	45
5.6 Use Case	48
4.7. Non-functional requirements	54
5.8. Summary.....	54
Chapter Six: System Design	55
6.1 Introduction.....	55
6.2 Framework and design considerations.....	55
6.2.1 Prototyping.....	56
CHAPTER 7: IMPLEMENTATION AND RESULTS	60

Figure 4. 6: Nulling technique of anonymization.....	41
Figure 5. 1: diagrammatic view of prototyping model(Kenneth, 1986)	44
Figure 5. 2: Use Case Diagram	47
Figure 6. 1: Layout of Framework	56
Figure 6. 2: Clustering mechanism.....	58
Figure 7. 1 Graphical representation of initial synthetic data	62
Figure 7. 2: Data points of all detections.....	64
Figure 7. 3: Clustering around centroid.....	67
Figure 7. 4: Determination of K in KNN	67
Figure 7. 5: Output of Classified data size and K in KNN.....	68
Figure 7. 6: Determining the Euclidean Distance	69
Figure 7. 7: Sample values of computed Euclidean distances	69
Figure 7. 8: KNN implementation.....	70
Figure 7. 9: Sorted K NNN of infected individuals data points	71
Figure 7. 10: Sorted K NNN of infected individuals data points	71
Figure 7. 11: voting and counting of infected individuals	72
Figure 7. 12: Voting results of infected individuals' proximity to postcode areas.	72
Figure 7. 13: Pie chart indicating percentages of nearness of data point	73
Figure 7. 14: Posting of clustering Results.....	74
Figure 7. 15: Baseline data	74
Figure 7. 16: Baseline data merged with postcodes (code).....	75
Figure 7. 17: Observed data	76
Figure 7. 18: Observed data merged with post codes centroids	76
Figure 7. 19: Determination of standard deviation and mean	76
Figure 7. 20: Data for CUSUM	77
Figure 7. 21: Output data gathered for CUSUM	77
Figure 7. 22: Aberration detection function	77
Figure 7. 23: Sample map presentation	78
Figure 7. 24: Improvement of Clusters on map.....	79
Figure 7. 25: Improvement of Clusters on map.....	79
Figure 7. 26: Single cluster view	80
Figure 7. 27: Dynamic graph visualization of Infections	81
Figure 7. 28: One-way hashing of Person IDS.....	81
Figure 7. 29: Nulling of Person names.....	82
Figure 7. 32: Sample of Observed counts for aberration detection.....	84

Figure 7. 33: Observed and corresponding baseline values	84
--	----

LIST OF TABLES

Table 3. 1: Data categories and their definitions.....	25
Table 3. 2: Principal findings on a systematic review of cluster detection mechanism for implementation.....	28
Table 3. 3: Evaluation metrics of some algorithms.....	29
Table 4. 1: Simulated Centroid of post codes of study area.....	31
Table 4. 2: Unclassified Data	32
Table 4. 3: Classified synthetic data of people with type-1 diabetes	33
Table 4. 4: Programming tools used.....	34
Table 4. 5: Methods used for implementation and evaluation	36
Table 4. 6: Sensitivity and Specificity analysis(Josseran et al., 2010).....	39
Table 5. 1: Functional Requirement number 1	45
Table 5. 2: Functional Requirement number 2.....	45
Table 5. 3: Functional Requirement number 3.....	45
Table 5. 4: Functional Requirement number 4.....	45
Table 5. 5: Functional Requirement number 5.....	46
Table 5. 6: Functional Requirement number 6.....	46
Table 5. 7: Functional Requirement number.....	46
Table 5. 8: Functional Requirement number 8.....	47
Table 5. 9 : Functional Requirement number 9.....	48
Table 5. 10: Functional Requirement number 10.....	48
Table 7. 1: Initial Simulated Data.	61
Table 7. 2: Synthetic data with detections (Classified dataset)	63
Table 7. 3: Centroid of post codes.....	65
Table 7. 4: Unclassified data	66
Table 7. 5: Cluster of number of infected individuals around centroid	66
Table 7. 6: Unclassified data point.....	68
Table 7. 8: Sensitivity and specificity of outbreak clusters.....	88

ACRONYMS

BG-Blood Glucose

CUSUM-Cumulative Summation

DSR-Design Science Research

EDMON-Electronic Disease Monitoring Network

MIT-Medical Informatics and Telemedicine Research Group

STPSS-Space Time Permutation Scan Statistics

EDMON- Cluster - KNN and CUSUM combined algorithm

KNN-Nearest Neighbor

PPM-Privacy Preserving Mechanisms

CHAPTER 1: INTRODUCTION

1.0 Background and Motivation

Electronic Disease Monitoring Network (EDMON) is an ongoing research in syndromic surveillance at the University of Tromsø, The Arctic University of Norway (Woldaregay et al., 2017). One of the main aims of EDMON project is to detect the spread of contagious diseases at the earliest possible moment, and potentially before people know that they have been infected thus as early as the incubation stage of infection (Woldaregay et al., 2017) through detecting infection incidences in people with type 1 diabetes and clustering them on time and geographical region. The results shall be visualized on real-time maps as well as presented in digital communication. The project uses self-recorded health related data from people with type-1 diabetes as input (Woldaregay et al., 2017).

In following the trend of disease surveillance, traditional disease surveillance systems mostly depend on laboratory confirmations as input data to detect disease outbreak (Hope, Durrheim, d'Espaignet, & Dalton, 2006). This results in significant time lag between infection time and the time of detection of infection through laboratory confirmation (Hope et al., 2006). This was transformed to syndromic surveillance systems (Hope et al., 2006) which greatly relied on visible signs and symptoms with data sources from emergency department records (Choi, Cho, Shim, & Woo, 2016), school absenteeism, work absenteeism, disease reporting systems and over-the-counter medication sales (Nie et al., 2014; Woldaregay et al., 2017). But significant delays have been observed between infection time and up to the visible sign and symptoms stage (Nie et al., 2014; Woldaregay et al., 2017). These types of disease surveillance systems do not detect the disease outbreak early enough and their data sources excludes the incubation stage of the infection (Woldaregay et al., 2017; Woldaregay et al., 2018). They mostly detect disease outbreak after the infected person is at the illness or after terminal stage, thereby increasing the disease burden such as infection rates (IR) and case fatality rates (CFR) (Kulldorff, 2005; WHO, 2017a, 2018).

These shortcomings of the surveillance systems possess a global health security threat resulting in higher mortality and morbidity rates (Daulaire, 2018; Kulldorff, 2007; Kulldorff, 2005; WHO, 2017a). For instance, mankind is still battling with the burden of infectious disease outbreaks such as mortality rate, morbidity rate, case fatality rate, economic losses, global fear and panic (WHO, 2015, 2017a, 2018). Seasonal disease outbreaks such as influenza still remains a global health threat (Quinn & Kumar, 2014). The outbreak of Ebola Virus Disease (EVD) in Liberia in West Africa, claimed over 11000 lives and resulted in national case fatality rate of about 70%. Local economic losses of \$3-4 billion was realized in this outbreak. Various continents descended to fight this outbreak partly for fear of spreading patterns into other parts of the Globe (Jafarpour Khameneh, 2014; Marí Saéz et al., 2015; WHO, 2015, 2017a).

The late detection and their related impact by most disease surveillance systems has since been noticed and efforts are being made by researchers to bridge the gap. For instance, the New York City Department of Health and Mental Hygiene developed syndromic surveillance system with data sources from emergency department visits and chief complaint information were electronically analyzed daily to detect disease outbreaks early (Heffernan et al., 2004). Recently, other enhanced syndromic surveillance systems have been proposed to be dependent on electronic health record data collected at the emergency department and urgent care settings (Jacquez, 2018). This calls for concerted efforts to develop better and effective syndromic surveillance systems that can detect disease in real time. The traditional and most syndromic surveillance systems have helped in detecting and managing disease outbreaks but with the current prevailing technology, it is feasible to provide a better lead time (Pedersen & Hartvigsen, 2015; Struchen, Vial, & Andersson, 2017; Woldaregay et al., 2018). The internet is becoming much more available and cheaper with the passage of time (Zickuhr & Smith, 2012). In addition to this, electronic devices including smart phones, watches and cameras are becoming much cheaper and smarter (Bonnington, 2015). The combination of the internet and ubiquitous devices has presented a huge opportunity for developing Information Technology systems for the management of chronic diseases including diabetes and cardiovascular diseases (Pedersen & Hartvigsen, 2015). This has resulted in the abundance of space-time data being generated by omnipresence and location-aware devices including GPS, smart phones and body area networks (BAN) (Wang, 2014). Such health-related data can be mined to enhance disease surveillance (Lauritzen, Årsand, Vuurden, Bellika, & Hejle, 2011; Struchen et al., 2017; Woldaregay et al., 2018).

So, in EDMON, a systematic review of cluster detection mechanisms for syndromic surveillance system was conducted (Yeng, Woldaregay, Solvoll, & Hartvigsen, 2018b). The aim was to pinpoint the state-of-the-art cluster detection mechanisms for the implementation of a syndromic surveillance system in the EDMON system. Various challenges such as user mobility, geographical location estimation and other factors were considered. To this end, the study revealed several space, time and spatiotemporal algorithms. One of the most used spatiotemporal algorithms was Space Time Permutation Scan Statistics (STPSS) (Yeng et al., 2018b). Though STPSS was mostly used in practically implemented algorithms among syndromic surveillance systems, a combination of temporal methods and near neighbor algorithms were desired to improve the power of detection in a geographically disaggregated surveillance data in order for these measures to augment for increase in sparseness of data towards preventing in a loss of power to detect in areas with local excess aberrations in spatial and spatiotemporal methods (Abellan J J, 2007; Isobel et al., 2016). Therefore, EDMON-Cluster was explored to implement clustering methods of detecting infectious disease outbreak in EDMON. Specific objectives include developing a spatial classifier with a classification error margin of 1% and implementing a temporal method with 1% error margin of sensitivity, and specificity. The timeliness and methods to deal with privacy and location

estimation challenges while generating visualization alarm and alert of outbreaks were also explored.

1.2 EDMON Background

Diabetes Mellitus (DM) is a medical condition which relates to the deficiency of insulin secretion (Type 1 Diabetes) or action (Type 2 Diabetes) (Casqueiro & Alves, 2012; Woldaregay et al., 2018). Diabetes can be treated, and its impact mitigated through diet, physical activity, medication, regular screening and treatment for complications (WHO, 2017b). People with diabetes often experience high Blood Glucose (BG) levels during infection incidents (Casqueiro & Alves, 2012; Diabetes Research and Wellness Foundation, 2018). The correlation between incidence of infections and an elevated blood glucose levels in diabetes has been known for a long time. Moreover, recent studies also support the evidence that there is a strong positive correlation between infection incidence and hyperglycemia episodes (Arsand et al., 2005; Botsis, Bellika, & Hartvigsen, 2009; Botsis & Hartvigsen, 2010; Botsis, Hejlesen, Bellika, & Hartvigsen, 2008; Botsis et al., 2012; J. N. Lauritzen et al., 2011; Woldaregay, Årsand, Botsis, & Hartvigsen, 2017; Woldaregay et al., 2018). For instance, Botsis et al. (Botsis et al., 2012) conducted a proof of concept study towards using blood glucose data as a potential surveillance indicator parameter based on daily glycemic control data of 248 people with type 2 diabetes and reported that blood glucose were significantly elevated during infection. Furthermore, the study concluded that a wide set of variables included in the diabetes profile could be used as supporting indicators.

Electronic Disease Monitoring Network (EDMON) is a kind of syndromic surveillance system which relies on self-recorded health related data from people with type 1 diabetes as secondary source of information to detect the incidence of infections possibly during the incubation period (Heffernan et al., 2004; Jacquez, 2018; A. Woldaregay et al., 2017; Woldaregay et al., 2018). EDMON has a high ambition to detect infectious disease outbreak as early as the incubation stage of the infection. This attribute makes it unique from other syndromic surveillance systems (Woldaregay et al., 2017). Most syndromic surveillance systems detect disease outbreak at the illness stage of infection or after the terminal stage. This usually results in a delay in detecting the disease outbreak which results in high impact of the disease burden (Woldaregay et al., 2017). EDMON has three tiers, which includes sensor and wearable unit, mobile computing unit and remote server unit, as shown in the Figure 1.1. The sensor and wearable unit encompass the patient unit, where the patient is expected to record the diabetes related data, as shown in the Figure 1.2. The mobile computing unit is a secure communication, which delivers the data from the subject to the dedicated central server for analysis. The remote server unit is divided into a personalized health module, a clustering module and data visualization module. The personalized health module tracks the individual subject on daily basis (morning, afternoon and evening) and detects unexpected blood glucose deviation from the previous normal patterns, as shown in the Figure 1.1

and 1.2 above. The clustering module tracks and detects incidence of any aberration patterns on the population levels with a spatio-temporal clustering algorithm.

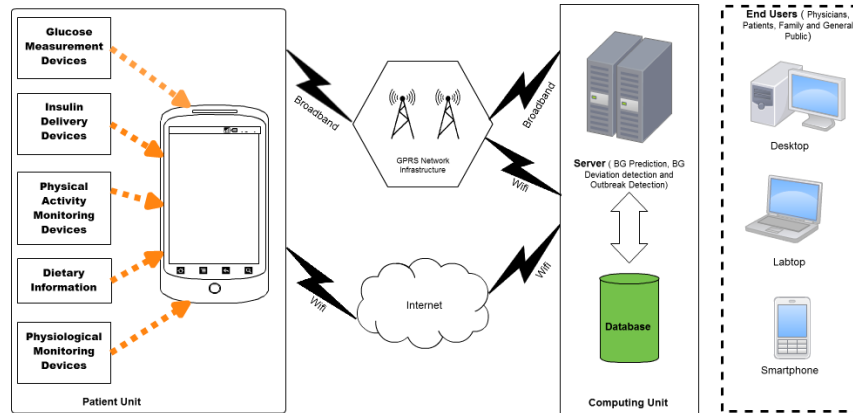


Figure 1. 1: EDMON system architecture (Woldaregay et al., 2017).

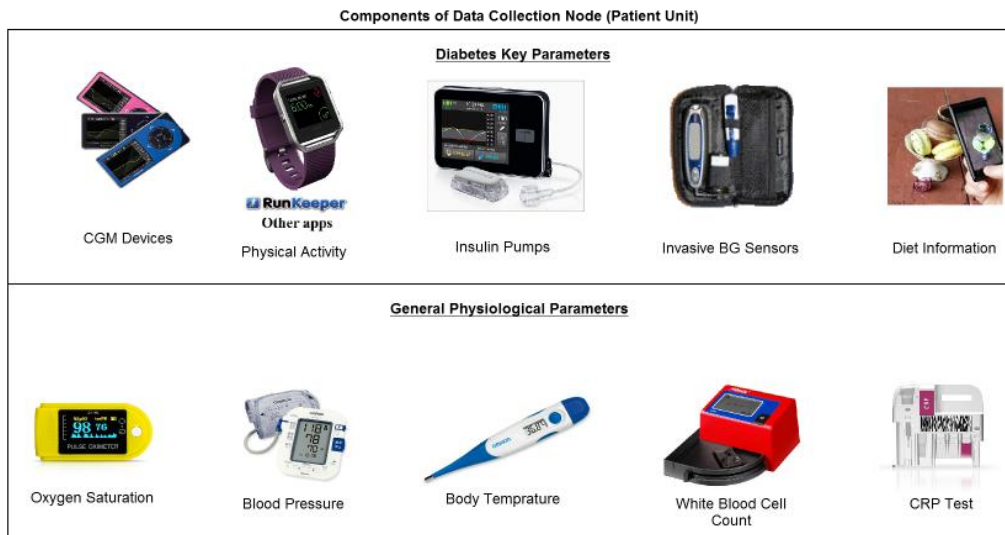


Figure 1. 2: EDMON patient units, where the patient records the necessary information (Woldaregay et al., 2017).

Therefore, the plan regarding EDMON is to track diabetes profile such as blood glucose levels, insulin dosage, diet (carbohydrate consumption), physical activity performed and other physiological parameters for infected persons towards infectious disease surveillance. EDMON will detect outbreaks of all pathogens that have significance effect on evolution of BG dynamics.

The general idea is that infection incidences in people with diabetes triggers a stress response which causes the release of glucose into the blood. However, due to their deficiency of insulin secretion or action, hyperglycemia persists (Casqueiro & Alves, 2012; Diabetes Research and Wellness Foundation, 2018; WHO, 2017b).

Recently, the availability of the internet and ubiquity of systems such as smart phones, tablets, smart watches, laptops and other systems have created greater opportunity for the advancement of diabetes management technologies (Wang, 2014). Through the electronic management of diabetes, big data is being generated as a “by-product” which can be processed to detect disease outbreak at an earlier stage in time. In the right mix of cluster detections, big data from self-management of diabetes, internet availability and the prevailing pervasiveness of devices, it is feasible and efficient to detect infectious disease outbreak as early as the incubation stage by using the vulnerability of diabetes persons as a sensor (Heffernan et al., 2004). Detection of disease outbreak at the incubation stage is important for reducing morbidity and mortality through early prevention and control (Marshall, Reynolds, Birch, Woodall, & Spitzner, 2009; Kulldorff, 2005; MedicineNet, 2017; Study.com, 2018). Cluster of blood glucose elevation in diabetic patients within a defined space, time, or both would help in predicting disease outbreak if other environmental factors which also causes stress response in diabetic patients are suppressed (Ali et al., 2016; Duangchaemkarn, Chaovatut, Wiwatanadate, & Boonchieng, 2017). Hence, the proposed Electronic Disease Surveillance Monitoring Network (EDMON) will use a personal self-collected health related data, state of the art cloud technologies, a dedicated mathematical model, i.e. personalized blood glucose deviation detection, and clustering techniques for an early detection of infectious disease outbreak.

The general objective of this study is to therefore implement an efficient cluster detection mechanism in EDMON and other similar syndromic surveillance systems for infectious diseases using the state-of-the-art cluster detection algorithms. Various challenges such as user mobility, privacy and confidentiality, geographical location estimation and other factors to shield the security and privacy of the study subjects have been considered.

1.3 Clustering

Generally, outbreak of infectious or communicable diseases are more likely to be presented in cluster form either in space, time, or both (Fanaee-T, 2012; P.N. Tan, Vipin Kumar, & Steinbach, 2005). Clustering methods in disease outbreak detection helps in the identification of environmental factors and spreading patterns linked with certain diseases (Wang, 2014). This has been realized many years ago by John Snow. A correlation was observed between cholera disease and source of public water (Colwell, 2004). Also, in an outbreak of Ebola virus disease which occurred in West Africa, there was a strong correlation of the spread of the virus from a 2-year old boy to his neighbors which resulted in their death (Marí Saéz et al., 2015). Furthermore, the spread of influenzas virus has been realized among clusters of people through hands resulting in person-to-person transmission (J. Barker, 2001).

1.4 Objective

Combining spatial and temporal algorithms has the tendency of boosting the power of cluster detection even when detected points are geographically dispersed (Abellan J J, 2007; Duangchaemkarn et al., 2017; Isobel et al., 2016; Mathes et al., 2017; Wang et al., 2010; Yih et al., 2010). Therefore, space and temporal algorithms are being explored through the development of prototype in EDMON system. Specific objectives include:

- I. Developing a spatial classifier with classification margin of error of 1%
- II. Implementing a temporal method with the performance metrics such as sensitivity and specificity of 1% margin of error and less than half an hour timeliness.
- III. Providing methods to deal with privacy and location estimation challenges
- IV. Generating visualization, alarms and alerts

1.5 Justification of the study

The importance of the study and its associated contribution to the body of knowledge are also laid out in this section.

1.5.1 Problem statement

The main goal of this study was to develop spatiotemporal prototype algorithm for syndromic surveillance with inputs from Type-1 Diabetes persons which could be improved through evaluations for empirical studies. The improved version of the system is aimed towards implementation in various syndromic surveillance systems such as EDMON (Woldaregay et al., 2017). In EDMON, the input sources such as the Diabetes persons with infection states in the surveillance area could be disaggregated at different times and locations such as postcodes (Woldaregay et al., 2017). Therefore, there is the need to develop algorithms that would be able to efficiently detect disease outbreak in such a disaggregated nature of data.

A systematic review was conducted (Yeng, Woldaregay, Solvoll, & Hartvigsen, 2018a), to explore potential methods, evaluation techniques, visualization methods and other dimensions. The systematic review revealed various algorithms that could be used to achieve the spatiotemporal objective of EDMON (Yeng et al., 2018a). Space Time Permutation Scan Statistics (STPSS), CUSUM, K Nearest Neighbor (KNN), K means clustering, WSARE, DBSCAN and Space Scan Statistics (SSS) (Yeng et al., 2018b) were some of the algorithms identified. STPSS and CUSUM were found to be the most used algorithms. From the review, STPSS could have been adopted in EDMON-Cluster since STPSS does not require population at risk data to draw the expected baseline value. STPSS dwells on the detected cases to determine the expected count (Kulldorff,

2005). This approach provides significant trend of baseline data while avoiding inclusion of historical data that is irrelevant to the current period. On the hind side of STPSS, the algorithm is only efficient on outbreaks that start locally (Kulldorff, 2005). According to Chen et al., who studied into “Spatial and temporal aberration detection methods for disease outbreaks in syndromic surveillance systems”, (Chen, Cunningham, Moore, & Tian, 2011); spatial scan methods only detect clusters in simple regular shapes such as cylindrical, circular or spherical. The spatial scan algorithm does not also consider prior knowledge such as the impact of the infection rate or size or shape of the outbreak and it is computationally expensive as local cluster search require searching over a large geographical region.

These short-comings suggest that STPSS is not suitable for detecting disease outbreaks which occur simultaneously in the entire surveillance area. For instance, disease outbreaks which occur through exposure to an infectious agent implies that subjects might be living in different neighborhood. So STPSS will not detect disease outbreaks with very few cases like one case of small pox or three cases of anthrax in the anthrax bioterrorism which occurred in 2001(Kulldorff, 2005). STPSS is only efficient on disease outbreaks with higher rate of early symptoms (Kulldorff, 2005). An evaluation which was performed through injection of spikes of known outbreak revealed low detection in the space and spatiotemporal algorithms (Abellan J J, 2007; Isobel et al., 2016; Mathes et al., 2017; Wang et al., 2010; Yih et al., 2010). For instance, in an evaluation exercise, at a specificity of 95%, the STPSS detected none (Isobel et al., 2016). This was due to the geographically disaggregated data which resulted in a loss of power of detection by the STPSS algorithm (Mathes et al., 2017).

Syndromic surveillance system is optimally effective when both spatial and temporal cluster detection methods work in unison to track emerging infectious diseases at an early stage over the surveillance area (Chen et al., 2011; Rogerson, 2005).

Another problem area is the balance point of using diabetes persons location tracking and identification data for public good. If the vital few of the diabetes persons data are to be used for disease surveillance to the benefit of the masses, then it is important to preserve the subjects’ privacy. Effective syndromic systems in space-time require location points, time stamps and identification of data points as input sources. Therefore, appropriate privacy and confidentiality preserving methods need to be adopted in the fulcrum to meeting both privacy and surveillance systems requirement(Chen et al., 2011).

In the light of these, there exists the need to explore for suitable algorithms that can be used in EDMON to effectively detect disease outbreak in a geographically disaggregated data. In addition to this, methods for handling privacy and security relating to sensitive data were explored.

1.5.2 Significance and contribution

Though STPSS was mostly used in practically implemented algorithms among syndromic surveillance systems (Yeng et al., 2018b), a combination of temporal methods and near neighbor algorithms were desired for optimal performance (Chen et al., 2011) in a geographically dispersed data. This allusion was supported by Khanita D. et al in their conclusion after evaluating their proposed study on “Symptom-based Data Preprocessing for the Detection of Disease Outbreak”, with time series and KNN algorithm (Duangchaemkarn et al., 2017). Nearest Neighbor and CUSUM were also statistically demonstrated to illustrate its feasibility of monitoring nearest neighbor statistics (Rogerson, 1997). When there is an aberration in the surveillance area, the CUSUM can spot this with the mean distances of emerging diseases of various points in the surveillance area (Rogerson, 1997, 2005). Kulldorff also support this opinion by emphasizing that “efficient disease surveillance will need the parallel use of different methods, each with their own strengths and weaknesses” (Kulldorff, 2005).

This study implemented a prototype algorithm by combining CUSUM and KNN towards improving upon the power of detection in a geographically disaggregated data point. The specific contribution includes:

- I. Developing a spatial classifier with classification margin of error of 1%
- II. Implementing a temporal method with the performance metrics such as sensitivity and specificity of 1% margin of error and less than half an hour timeliness.
- III. Providing methods to deal with privacy and location estimation challenges
- IV. Exploring for generating visualization, alarms and alerts

1.6 Scope of the system evaluation

Evaluation consists of measuring or describing something, aimed to answer questions to inform choice and decision making of a product, process or service (Charles P. & Jeremy C., 2006). Evaluation of a system in medical informatics is essential for promotional, scholarly, pragmatic, ethical, and other related purposes (Charles P. & Jeremy C., 2006). The pragmatic reason is pursued to indicate the techniques or methods which are more effective in this study. This reason was also pursued to point out why certain methods are not the best option. This has been justified since the pragmatic reason could enhance the clarity of the contribution and significance of the study approach. Ultimately, since the study modules include the combination of spatial and temporal clustering algorithms, the implementation and assessment of the clustering algorithm and the sensitivity and specificity of the temporal algorithm (CUSUM) were conducted.

1.7 Assumptions, Biases and Limitations

Maintaining the privacy and confidentiality of study subjects is always a major problem in obtaining accuracy of spatial and spatiotemporal detection methods essentially for individual data involving location tracking and their respective time stamps (Chen et al., 2011). It is quite challenging currently to find the balance point to obscure personal information in geolocation-tracked related dataset and the requirement to use this dataset for the public good thus public health disease surveillance systems (Chen et al., 2011). Therefore, it is assumed the simulated data containing geolocations have been treated in such a way of which the locations are not exact to the extent of revealing the privacy of individuals but valid for maintaining accuracy of the surveillance system.

1.8. Organization

The rest of the manuscript is organized as follows:

Chapter 2: Theoretical framework and State of the art: - This chapter describes the basic theoretical concepts and framework of the study. It discusses spatial clustering, temporal clustering and their related evaluation techniques. It further discusses the state of the art methods of clustering in syndromic surveillance.

Chapter 3: Literature Review: - This chapter presents a systematic literature review on clustering algorithms used in syndromic surveillance. It provides an overview of the state-of-the-art clustering methods for developing syndromic surveillance systems. Other dimensions such as location estimation, evaluation methods, nature of location, aberrations and thresholding methods and generation of alarms to alert possible outbreaks

Chapter 4: Materials and Method: - This chapter presents the materials and methods used in this study.

Chapter 5: Requirements specification: - This chapter describes the necessary functional and nonfunctional requirements and specifications of the cluster detection mechanism in EDMON

Chapter 6: Design: - This chapter describes the strategies and techniques used to develop the clustering methods for outbreak detection mechanisms.

Chapter 7: Implementation and Results: - This chapter presents the implementations and results of the various models of the design in Chapter 6. It also presents the execution of testing and evaluation results of the clustering system.

Chapter 8: Discussion: - This chapter discusses the implementations, evaluations, comparisons and analysis of the test results and the research findings.

Chapter 9: Further works/Recommendations: - This chapter describes future works and gaps that the author identified during this research

Chapter 10: Conclusion: - This chapter summarizes and concludes on the study outcomes and findings.

References: - This section presents list of references used in this thesis project.

Appendix: - This section contains the list of files and folders for data, files of algorithms and evaluation results.

CHAPTER 2: THOERITICAL FRAMEWORK AND STATE OF THE ART

2.1 Introduction

This chapter focuses on the theoretical frameworks and concepts, which form the foundation for the implementation of the cluster detection mechanism of syndromic surveillance in EDMON. It describes the basics of disease surveillance, clustering, methods and evaluations relating to the implementation of cluster detection mechanism of syndromic surveillance in EDMON. The whole chapter is organized as follows;

The first section presents terminologies, preliminaries and definitions that are fundamental in this research project. The second discusses issues related to electronic disease surveillance systems and early outbreak. The third section delved into clustering and state of the art algorithms. The last section gives a background of the methods and evaluation techniques relating to this implementation.

2.1.1 Terminologies, Preliminaries and Definitions

The section presents terminologies, preliminaries and definitions that are fundamental in this research project. These terminologies are also used throughout the entire project. The intention is to give the reader the fundamental understanding to be able to determine the basics of the study case definition.

2.1.2. Definitions

Specificity: is defined as “the proportion of true non-events correctly classified as such, the inverse being the false alarm rate” (DREWE, HOINVILLE, COOK, FLOYD, & STÄRK, 2011; Woldaregay et al., 2017).

Sensitivity: refers to “the proportion of actual cases in a population that are detected and notified through the system” (WHO, 2006).

Positive predicative value (PPV): refers to the proportion of clusters, that have been correctly detected as outbreaks (WHO, 2006)

2.2 Disease Surveillance

The term disease surveillance is referred to an actively ongoing systematic collection, analysis, interpretation and disseminations of health data for decision making in public health management (Brennan, 2002; Choi, 2012). Disease surveillance essentially involve watching over the occurrences of symptoms and transmission of a disease in a given population and geographical area or time frame for the planning, implementations and evaluation of public health actions (Choi, 2012). Over the years, disease surveillance has provided some knowledge for public health management to control mortality and morbidity (Savel & Foldy, 2012). Such interventions mostly

include vaccinations, quarantining or isolation and public trainings for awareness creations (Choi, 2012; Delisle, Roberts, Munro, Jones, & Gyorkos, 2005; Savel & Foldy, 2012). Governmental and non-governmental organizations can also depend on surveillance results to take the necessary actions on policy making and implementations (Delisle et al., 2005).

2.3 Clustering

One of the most basic ways of understanding and learning is through organizing data into sensible groupings (Jain, 2010). Clustering involves the study of methods and algorithms for grouping, objects or data points into measured or perceived fundamental characteristics or similarities as shown in figure 3 (Fanaee-T, 2012; Jain, 2010). In figure 5, objects that are red, green or blue in color are grouped together based on their similar in color.

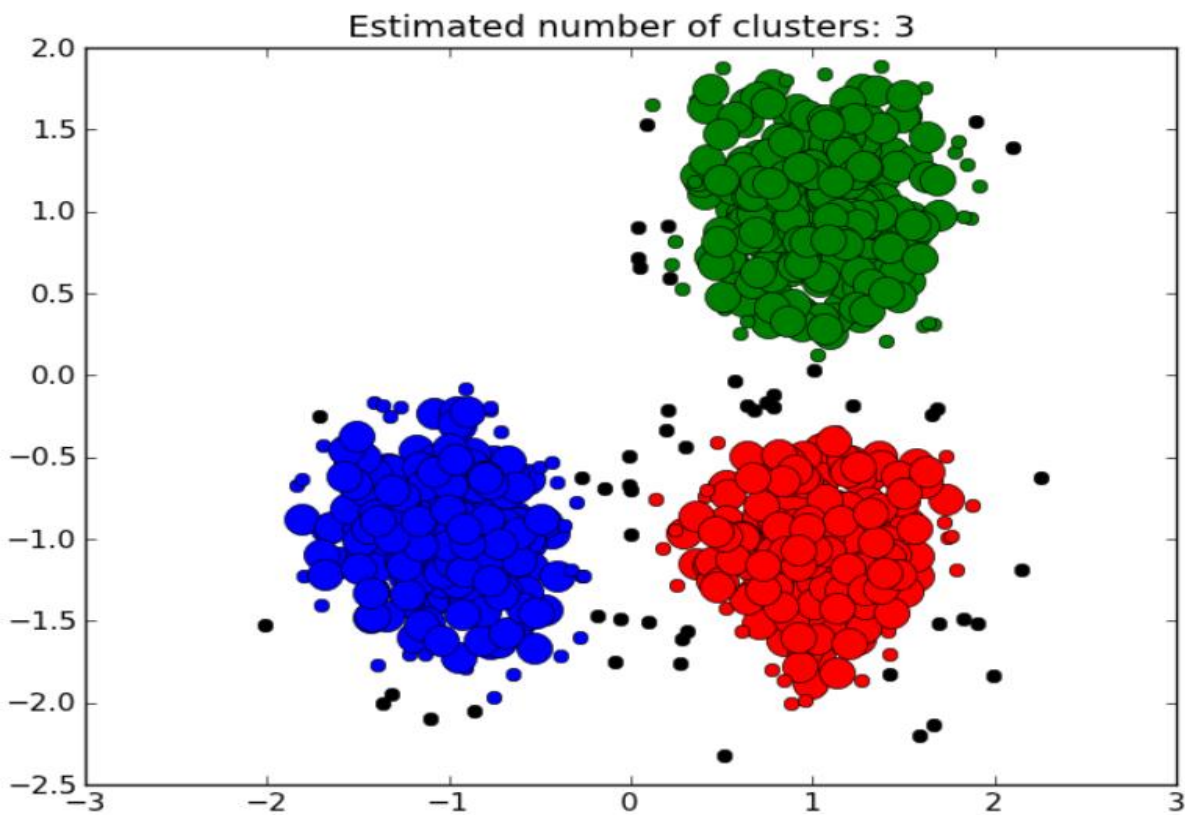


Figure 2. 1: Items grouped into three clusters with estimated outliers (scikit-learn developers, 2011)

Clustering approach could be roughly categorized as temporal, spatial and spatiotemporal. Spatial clustering uses multi-dimensional vectors with longitudinal and latitudinal coordinates. There are variety of such algorithms including K-Nearest Neighbor (KNN) (Bremner et al., 2005; Kim, Kim, & Savarese, 2012), Space Scan Statistics (SSS) and density-based spatial clustering of applications with noise (DBSCAN) (Birant & Kut, 2007; Fanaee-T, 2012; P.N. Tan et al., 2005). Temporal clustering deals with data points associated with time (Chan, Teng, & Hwang, 2015; Hutwagner,

Browne, Seeman, & Fleischauer, 2005). It includes various algorithms such as cumulative summation (CUSUM) and what is strange about recent event (WSARE) (Chen et al., 2011; Kleinman, Abrams, Kulldorff, & Platt, 2005; Kulldorff, 2007). Spatiotemporal clustering occurs when there is the involvement of time dimension (temporal information) and space dimension (spatial information) (Birant & Kut, 2007; Fanaee-T, 2012; P.N. Tan et al., 2005). There are variety of strategies including different distance functions (Jeung, Yiu, Zhou, Jensen, & Shen, 2008; Khokhar & Nilsson, 2009), importing time to the spatial data, transform spatiotemporal data to the new objects, progressive clustering and spatiotemporal pattern discovery (Birant & Kut, 2007; Fanaee-T, 2012).

An insight into clustering methods has been provided by various literatures to help understand the theoretical framework of cluster detection mechanism in syndromic surveillance (Fanaee-T, 2012). This has been initiated with classical data. In classical data, each data point is represented by its corresponding x and y coordinate values in a 2-Dimension axis (Fanaee-T, 2012). It does not show the location of the data point in a geographical space as shown in figure 6.

6						
5						
4						
3						
2					x (3,4)	
1						
0						
	0	1	2	3	4	5

Figure 2. 2: Classical data representation

It is basically an abstract representation of data points in an x and y axis as shown in figure 2.2. Spatial data is much like classical data. But the data points are represented by their corresponding longitudinal and latitudinal values in a 2-Dimensional space indicating their location in space. It usually indicates their location on earth. In spatial data, aside the longitudinal and latitudinal coordinate values, no further information is provided about the data point or the data item (Fanaee-T, 2012; P.N. Tan et al., 2005). Spatio-Temporal data occur when there is the involvement of time dimension or when temporal information is associated with spatial data (Birant & Kut, 2007; Fanaee-T, 2012; P.N. Tan et al., 2005). For instance, a person can be located at latitude L1, Longitude G1 and at a particular time T1, (L1, G1, T1). Considering these three axes scenario, the data points in EDMON can be associated with space, time or Spatio-Temporal data. When a diabetic patient infected at a point in time, the person here represents the data point or data item in

space who can be located with longitudinal and latitudinal coordinate values on earth at a given time.

Further to this, according to (Fanaee-T, 2012; P.N. Tan et al., 2005) there are three different types of Spatio-Temporal Data. These are Events, Geo-reference and Moving Points data. Event data is data in which the data items have no correlation with each other in space and the data set have no identification or the identifications are of no importance. Geo-Referenced data items have spatio-temporal data attributes with non-spatial value related to the data item. For instance, a weather station location and corresponding temperature values at different times. In moving data Item, the data items are involved in movement in time with associated Identifications. The data set involves longitude and latitude coordinates, time and Identification.

Another interesting point worth understanding is spatial data clustering. Spatial data clustering uses multi-dimensional vectors with longitudinal and latitudinal coordinates. It can be done with density-base method or distance base method. Density-base method uses Density-Based Spatial Clustering of Applications with Noise algorithm (DBSCAN). Also, Density-base method uses Ordering Points To Identify the Clustering Structure (OPTICS) algorithm. Spatial Clustering method would only be applicable in EDMON if the time dimension in EDMON is not used (Birant & Kut, 2007; Fanaee-T, 2012; P.N. Tan et al., 2005).

From sources (Birant & Kut, 2007; Fanaee-T, 2012) Spatiotemporal Data Clustering method is almost like Spatial Clustering except that the spatiotemporal method uses time variable as part of the data point in the dataset or algorithm. EDMON is also in line with Spatiotemporal data. When a diabetic patient infected a point in time, the person at a location represents the data point or data item in space who can be located with longitudinal and latitudinal coordinate values on earth at a given time. The data set involves longitude and latitude coordinates and time which maps to an identified object (Madhulatha, 2012).

2.3.1. Thresholds of Aberration Detection

Aberration detection in the context of cluster detection in syndromic surveillance, are statistical tools which monitor clusters and create alerts when the observed number of counts of syndromes in a cluster exceeds the threshold of the baseline or expected number of occurrences in space, time or space-time (Chen et al., 2011; Tsui et al., 2003). Temporal aberration detection deals with the excess count of syndromes occurrence during a specified time while the excess of syndromes detected within a specified geographic location defines spatial aberration detection. But if the aberration detection deals with excess count of syndromes of both temporal and space, then spatiotemporal aberration detection is defined (Chen et al., 2011; Kleinman et al., 2005; Kulldorff, 2007). Aberration detection is mainly performed through thresholding mechanisms including various forms such as number of standard deviations set from the mean (z-score), generalized likelihood ratio, RI and confidence intervals (Chen et al., 2011; Kajita, Luarca, Wu, Hwang, & Mascola, 2017; Sharip, 2006). The threshold in this scenario is a set value which triggers alerts if

the test statistics exceeds the set value (Chan et al., 2015; Hutwagner et al., 2005). Thresholds are very important in aberration detection. Thresholds set to a very high sensitivity may not be able to detect some aberrations. On the other hand, if the sensitivity is low, high false positive rate would occur. As a result, some guidelines have been provided by the Centers for Disease Control as how to apply some of these thresholds (Chan et al., 2015; Hutwagner, Thompson, Seeman, & Treadwell, 2003). But their applications largely depend on the type of surveillance and other factors (Chan et al., 2015; Hutwagner et al., 2003).

2.3.2 K-nearest neighbor algorithm

k-nearest neighbor algorithm (KNN) (Bremner et al., 2005; Kim et al., 2012) is a data classification technique which depends on the proximity of the unclassified data point to the training sets in the feature space. KNN is a non-parametric, lazy learning algorithm with the purpose to use a database in which the data points are separated into several classes for the prediction of the classification of the unlabeled or unclassified data point (Analytics Vidhya, 2018; Bolandraftar & Imandoust, 2013; Bronshtein, 2017). KNN being non-parametric, implies that it does not consider any assumptions on the underlying data distribution (Analytics Vidhya, 2018; Bolandraftar & Imandoust, 2013; Bronshtein, 2017). So, the structure of the model is derived from the sample data without adequate or prior knowledge of the data distribution (Analytics Vidhya, 2018; Bolandraftar & Imandoust, 2013; Bronshtein, 2017). KNN being a lazy learner means that generalization is not done based on the training data points (Bronshtein, 2017). The lack of generalization implies that KNN retains the entire training data. In order to be more precise, the entire training dataset (or most) is required during the testing phase (Bronshtein, 2017). This rule in KNN basically retains the whole training set data during learning and measures to each query of a class through the majority votes or labels of its k-nearest neighbors in the training data set. The KNN algorithm is among the machine learning algorithms with the training process for this algorithm consisting of storing feature vectors and labels of the training sets. In the classification process, the unlabeled or unclassified data point is assigned to the label of its k nearest neighbors. The “K” in the KNN algorithm is the nearest neighbors that the vote should be considered from. Essentially, the data point or object is classified based on the labels of its k nearest neighbors by simple majority vote.

With the KNN technique each unclassified or unlabeled data point should be classified similarly to its classified surrounding data points. Therefore, an unclassified or unlabeled sample is categorized by taken into account the classification of the proximity of its classified neighbor samples. The KNN method therefore involve an unknown sample which can be referred to as unclassified or unlabeled data points and a training set which is a classified data set. The distances between the unclassified data point and all the classified data samples in the training set can be computed. The computed distance between the classified data set and the unclassified data point with the smallest value is the closest to the unknown sample. Therefore, the unknown data point could be classified based on this classification technique of this nearest neighbor.

Figure 2.3 shows a diagram of the KNN algorithm. In figure 2.3 (a), with $K=1$ in the 1-NN decision rule, the unknown data point (?) in the diagram is assigned to the class on the left. In figure 2.3(b), with $K=4$ in the 4-NN decision rule, the unknown point (?) is also assigned to the class on the left.

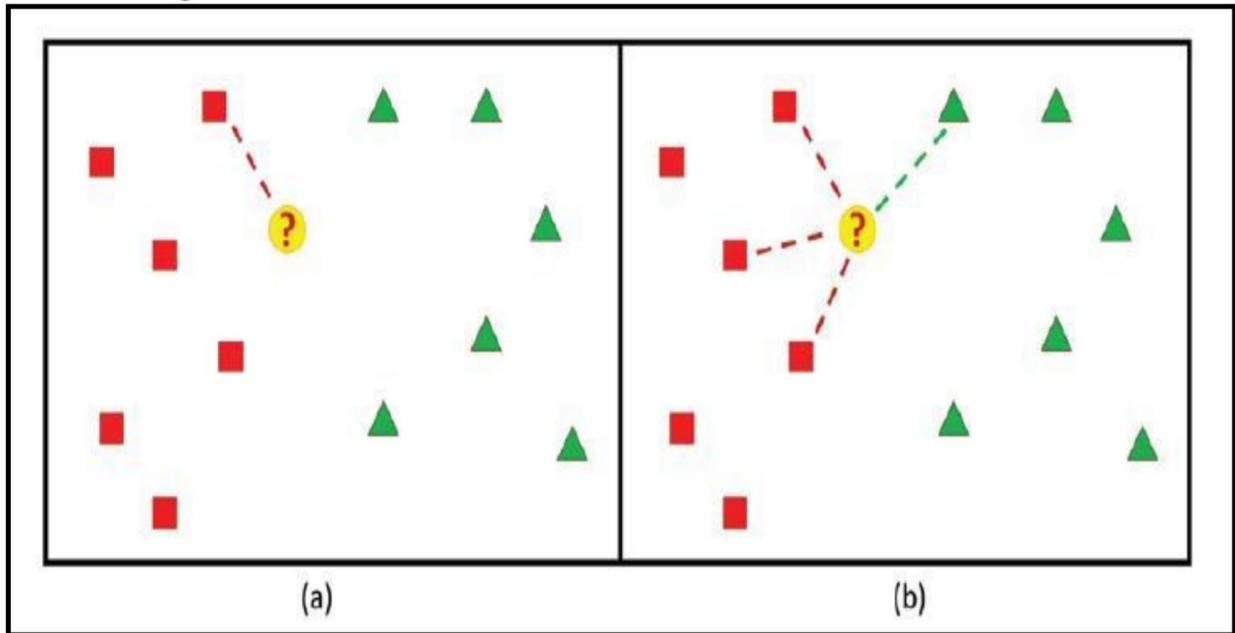


Figure 2. 3: Example of K-NN classification (Bolandraftar & Imandoust, 2013).

2.3.3 How to choose K in KNN

The efficiency of the KNN classifier is primarily dependent on the choice of parameters including one known as K (Bolandraftar & Imandoust, 2013; Gil-García & Pons-Porrata, 2006; Silverman & Jones, 1989). The K is the chosen total number of data points in the training dataset which are considered to be closer to the unknown or unclassified data point. These data points are considered to participate in a voting to determine the class in which the unknown or unclassified data point is to be categorized into. The estimate is impacted by the choice of the neighborhood size K, since the radius of the local region is determined by the distance of the Kth nearest neighbor to the query or unclassified data point. Apparently, different K results in different conditional class probabilities. If the value of K is small as shown in figure 2.4(a) and (b), the local estimate tends to be less efficient because of the disaggregation of the data which can result in noise and mislabeled data points. The accuracy and efficiency of K increases with increasing value of K as shown in figure 2.4 and 2.5. Considering figures 2.4(a) and 2.4(b) through to figure 2.5(a) and 2.5(b) it is observed that the boundaries become smoother with increasing value of K. With progressive increase of K, to an optimum value, a smoothed boundary is reached where all blue or all red are being separated depending on the total majority (Analytics Vidhya, 2018; Jirina & jr., 2008; Jirina, 2010).

However, if a large value of K is estimated, then over smoothing do set in which leads to classification performance degradation resulting in the occurrences of outliers from other classes. In gist, a small value of k results in noise which have a higher significant of poor estimation on the result and larger value of K results in higher computationally cost in the algorithm (Analytics Vidhya, 2018; Jirina & jr., 2008; Jirina, 2010).

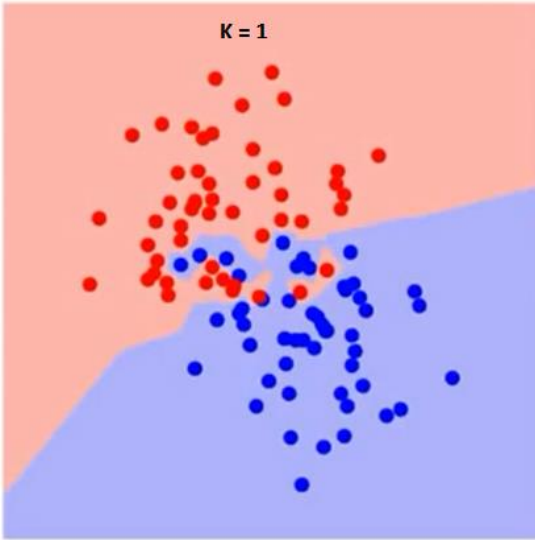


Figure 2. 4(a) (Analytics Vidhya, 2018)
(Analytics Vidhya, 2018)

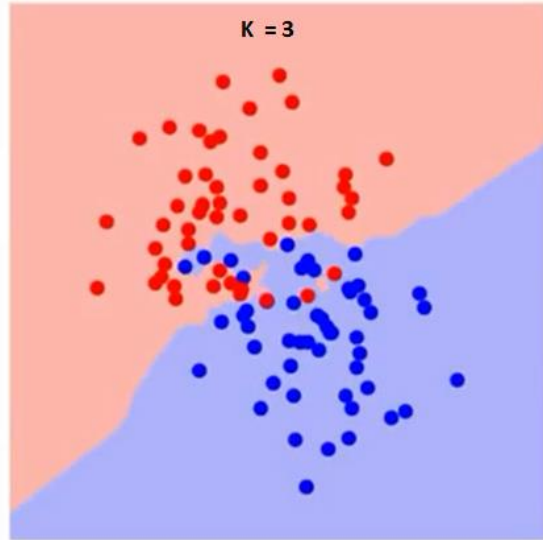


Figure 2.4(b)

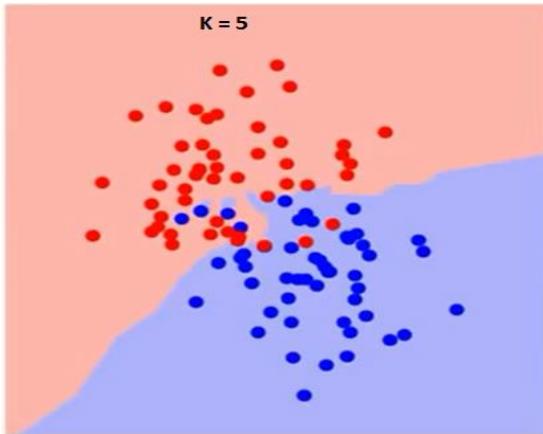


Figure 2. 5(a)(Analytics Vidhya, 2018)

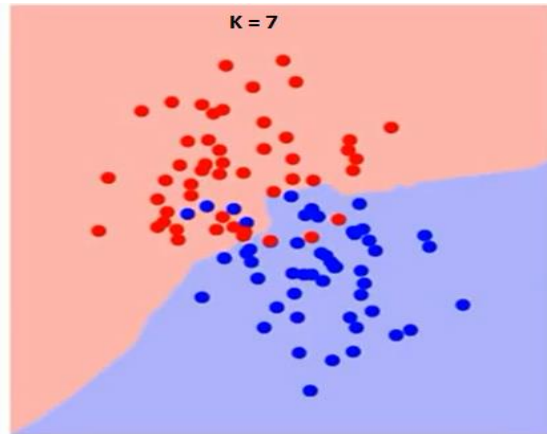


Figure 2.5(b) (Analytics Vidhya,
2018)

To this end, various methods have been adopted to ensure the choice of K results in optimal performance (zakka, 2018). One of such methods of choosing K is based on a rule of thumb where the K is selected to be the square root of the total number of the training dataset (Analytics Vidhya, 2018; Jirina & jr., 2008; Jirina, 2010). This rule of thumb where $K=\sqrt{n}$, with n being the number of samples in the training dataset, often perform well(Jordan, 2017) resulting in a balance of

speed and enhanced accuracy of the algorithm (Analytics Vidhya, 2018; Jirina & jr., 2008; Jirina, 2010).

2.3.4 Distance Metrics in KNN

The KNN algorithm is dependent on the measured distances between the unlabeled data and each of the training dataset, to decide on the final classification outcome (Analytics Vidhya, 2018; Jirina & jr., 2008; Jirina, 2010). Various approaches for the calculation of the measured distance include Euclidean, cosine, Chi square, Chebychev Distance and Minkowsky distances (Hu, Huang, Ke, & Tsai, 2016; Michael Greenacre & Primicerio, 2013; Singh et al., 2013).

Euclidean distance emanates from the concept of Pythagoras’s theorem(Michael Greenacre & Primicerio, 2013; Teknomo, 2017) with the theory that the squared length of a vector $x = [x_1 \ x_2]$ as shown in figure 2.6, is determined as the sum of the squares of its coordinates as shown in triangle OPA in figure 2.6. Also, the squared distance between two data points or vectors $x = [x_1 \ x_2]$ and $y = [y_1 \ y_2]$ is calculated as the sum of squared differences in their coordinates as shown in triangle PQD in figure 10.

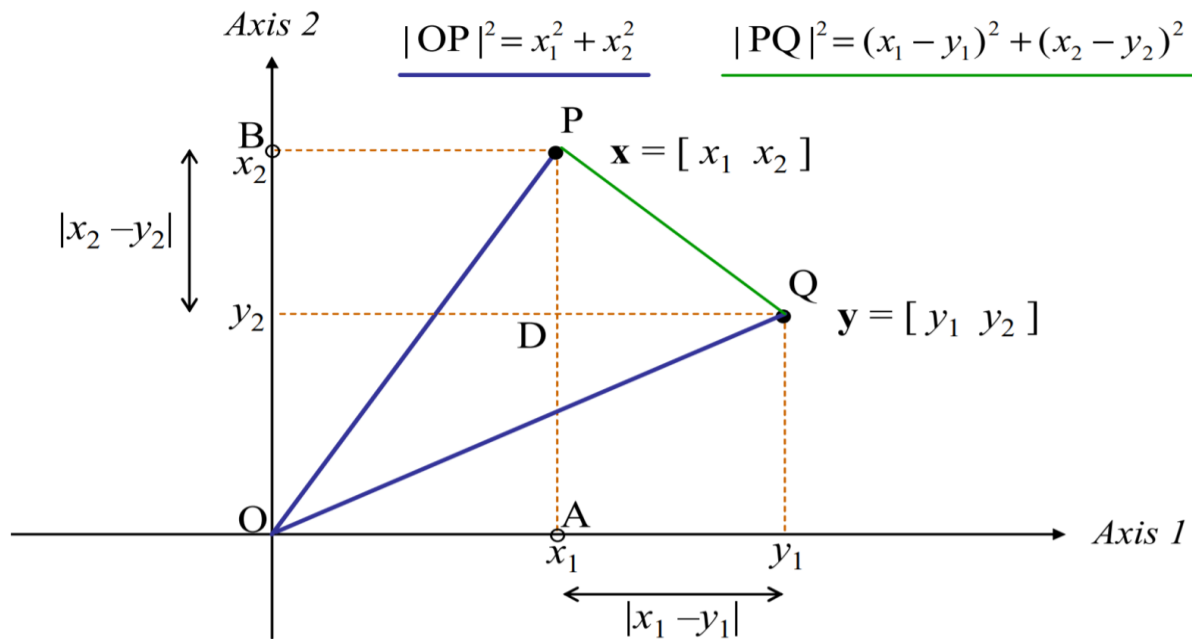


Figure 2. 6: Application of Pythagoras’s theory in distances between two data points (Michael Greenacre & Primicerio, 2013)

From figure 10, the distance between vectors x and y can be noted as $d_{x,y}$ (Michael Greenacre & Primicerio, 2013) such that the result can be represented as follows;

$$d_{x,y}^2 = (x_1 - y_1)^2 + (x_2 - y_2)^2 \dots\dots\dots\text{eqn (1)}$$

$$d_{x,y} = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} \dots\dots\dots\text{eqn(2)}$$

Referring to eqn(1) and eqn (2), Euclidean Distance between two data points such as x and y is therefore defined as the square root of the sum, over all dimensions, of the weighted squared differences between the values for the data points or cases(Borgatti, 2018; Center, 2018) as indicated in eqn (3).

$$d(x, y) = \sqrt{\sum_i^n (x_i - y_i)^2} \dots\dots\dots\text{eqn(3)}$$

Cosine similarity is one of the distance measures defined as the cosine of the angle between two n -dimensional vectors in an n -dimensional space. It is expressed as the dot product of the two vectors divided by the product of the length or magnitudes of the two vectors(Nguyen & Ba, 2010) as shown in equation(eqn) 4.

$$\text{similarity}(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n A_i^2} \times \sqrt{\sum_{i=1}^n B_i^2}} \dots\dots\dots\text{eqn (4)}$$

The chi-squared distance is known to be a nonlinear metric which is mostly used in comparing histograms (Yang, Xu, Chen, Zheng, & Liu, 2015). Its name was obtained from the view of the mathematical expression that the chi-squared distance is similar to that found in the expansion of the chi square goodness of fit test (Boscovich Roger J. & E.P.George, 2018).

For instance, in a frequency table with r rows and c columns, the row and column profiles can be determined (Boscovich Roger J. & E.P.George, 2018; Yang et al., 2015). The r and c points can then be plotted from the profile and the corresponding weight of each term can be calculated as the inverse of its frequency (Boscovich Roger J. & E.P.George, 2018; Yang et al., 2015). The chi-squared distance is therefore defined as the Euclidean distance between the components of the profiles, of the defined weights (Boscovich Roger J. & E.P.George, 2018; Yang et al., 2015).

Manhattan distance involves the calculation of the absolute differences between coordinates of pair of objects as shown in eqn(5)(Singh et al., 2013).

$$\text{Dist}_{xy} = |X_{ik} - X_{jk}| \dots\dots\dots\text{eqn(5)}$$

Chebyshev Distance which is also known as the maximum value distance is calculated as the absolute magnitude of the differences between coordinate of a pair of data points. Chebyshev

distance is a type of Minkowski distance such that $p=\infty$ (taking a limit). This distance can be used for both ordinal and quantitative variables (Singh et al., 2013)

$$Dist_{XY} = \max_k |X_{ik} - X_{jk}| \dots\dots\dots eqn(6)$$

Minkowski Distance is the generalized metric distance. Note that when $p=2$, the distance becomes the Euclidean distance. When $p=1$ it becomes city block distance (Singh et al., 2013).

$$Dist_{XY} = \left(\sum_{k=1}^d |X_{ik} - X_{jk}|^{\frac{1}{p}} \right)^p \dots\dots\dots eqn(7)$$

Contemporary research in healthcare rely on Geographical Information Systems and spatial related data. This has triggered a study to establish the effectiveness of the combined effects of using Euclidean measurements and geographic zip-code centroids in health care research. The objective of this study was to determine if there exists statistically significant variance in distance values in using Euclidean measurements and zip-code centroid geolocations methods in comparison with more precise spatial analytical methods such as drive distance data and residential geocoded address. The study revealed that “geocoded address was highly correlated ($r=0.99$) with the Euclidean distance from the zip-code centroid” (Jones, Ashby, Momin, & Naidoo, 2010). With this high significance of results, this current study in cluster detection mechanism in EDMON would apply this study results as a conventional data aggregation technique which is less time consuming and easier to obtain(Hu et al., 2016).

2.3.5 Cumulative Summation

Cumulative Summation (CUSUM) is a statistical control method which has traditionally been used for industrial process control. It has been predominantly used in tracking changes in production process average levels since in the 1950s (O'Brien & Christie, 1997; Woodward, 1964). The main role of CUSUM in the production control was to generate alert if products from a production process were nonconforming to defined limits (PAGE & Statistical Laboratory, 1954). But with the advent of electronic disease surveillance, CUSUM has been found to be very useful in this direction. CUSUM algorithm accumulates the variances between detected or observed cases and baseline values over a given time (O'Brien & Christie, 1997; Peter A., 2005). If the cumulative summation value is greater than the baseline by a specified thresholding, a likelihood aberration is detected (O'Brien & Christie, 1997). In disease surveillance, CUSUM demonstrated to be very sensitive, fast reactive method of detecting disease outbreaks and generates less false positive alarms than more conventional methods (Abellan J J, 2007; Isobel et al., 2016; O'Brien & Christie, 1997). CUSUM is also among the most commonly used temporal algorithms due to its powerful and straightforward to design and implement (Watkins, Eagleson, Veenendaal, Wright, & Plant, 2008; Yeng et al., 2018b)

The formula used to express CUSUM is as follows;

$$CuSum_t = \sum_1^t e_t$$

The e represents the observed number of events minus the reference value (the baseline), and the t represents the time associated. Conventionally, the CUSUM value is initialized to zero (O'Brien & Christie, 1997). A positive result indicates a change above expected (O'Brien & Christie, 1997). A zero outcome signifies a period when the observed number of events are the same as the expected number (Hutwagner et al., 2003; O'Brien & Christie, 1997). While a negative value of the result indicates that events have fallen below expected levels (O'Brien & Christie, 1997). There is different type of CUSUM algorithms, which is generally referred to as the Early Aberration Reporting Systems (EARS) (Groeneveld et al., 2017; Hutwagner et al., 2005; Hutwagner et al., 2003; Watkins et al., 2008). The known EARS algorithms are C1-MILD(C1), C2-MEDIUM(C2) and C3-ULTRA(C3) (Groeneveld et al., 2017; Hutwagner et al., 2005; Hutwagner et al., 2003; Watkins et al., 2008). The C1, C2 and C3 names were given according to their levels of sensitivities with C1 being less sensitive to C3 being more sensitive (Hutwagner et al., 2005). The C1 algorithm aberration method depends on a conventional alarm level of $Cl=2$ (Groeneveld et al., 2017; Hutwagner et al., 2005; Hutwagner et al., 2003; Watkins et al., 2008). This means in the C1 algorithm, the current detected value is greater than the baseline means with an addition of three standard deviations which has been calculated based on the past 7days of historical data (Groeneveld et al., 2017; Hutwagner et al., 2005; Hutwagner et al., 2003; Watkins et al., 2008). The use of a guard band of two days period between the current day under evaluation and the baseline is the distinguishing factor in C2 when compared with the C1(Groeneveld et al., 2017; Hutwagner et al., 2005; Hutwagner et al., 2003; Watkins et al., 2008). Essentially, the C2 used 7 days background data while ignoring the most recent 2 days data (Hutwagner et al., 2005). Aside the guard band of two days duration, the C3 algorithm computes a partial sum of the current value from the mean for the last three days of the positive deviations (Watkins et al., 2008). Most syndromic surveillance systems depend on about 3 to 5 years long historical data to form a baseline for aberration detections (Hutwagner et al., 2003). But recent developments in biological attacks such as the release of *Bacillus anthracis* in the United State and higher case fatality rate, there is the need to develop efficient syndromic surveillance systems which are independent on long term historical data (Hutwagner et al., 2003). In a study “comparing aberration detection methods with simulated data” (Hutwagner et al., 2005), the aberration detection algorithms with short term duration baseline data (C1, C2 and C3) are as effective as the methods requiring long term historical data in terms of specificity, sensitivity and timeliness(Hutwagner et al., 2005). C1, C2 and C3 algorithms have also been developed to accommodate daily and seasonal variations. Their mean and standard deviations are based on a week’s information which are computed in the same season (Hutwagner et al., 2005).

2.3.6 Geographical location of the diabetes subjects

The geographical location of the diabetes subjects is deemed essential for clustering and detecting aberrations of infected individuals (Hutwagner et al., 2005; Musa et al., 2013). One of the methods

involves partitioning the region of interest into different small equal cells as shown in Figure 8 (Woldaregay et al., 2017) (Yang & Abraham O. Fapojuwo, 2015). In wireless system within the telecommunication industry, a cell is defined as a geographical area covered by a transmitter of a cellular telephone. The transmitting system is known as a cell site (Yang & Abraham O. Fapojuwo, 2015). The diameter of a cell ranges between one mile to twenty miles under a cell site and this is much related to the terrain and transmission power (Yang & Abraham O. Fapojuwo, 2015). A group of coordinated cell sites is termed as cell system (Yang & Abraham O. Fapojuwo, 2015). Every cellular telephone provider generally has their local cell system for their users who subscribe to that. When the users are travelling out of the coverage of this network or cell system, the cell system automatically transfers the user to a neighboring company's cell system which is termed as roaming service(Yang & Abraham O. Fapojuwo, 2015). Each cell is always divided into hexagonal shape for enhance coverage as shown in figure 11(Yang & Abraham O. Fapojuwo, 2015). In this regard as shown in figure 2.7, the exact locations of the users are tract with GPS/GSM mobile network’s address. The subject is considered to be in one of the cells based on the input address at the instance of data submission (Groeneveld et al., 2017; Hutwagner et al., 2005; Hutwagner et al., 2003; Watkins et al., 2008). This forms a dynamic nature of location in which the location of the subject is tagged to the date and time of data submission. In this method, the mobility of the subjects is taken into consideration. So, the baseline data and current detected cases can easily be computed by counting the number of subjects in each cell at a defined date, time or both. Therefore, if the counted subjects in a cell which is the geographical area at a given time exceeds a defined threshold, that cell which is a cluster can be queried for disease outbreak (Groeneveld et al., 2017; Hutwagner et al., 2005; Hutwagner et al., 2003; Watkins et al., 2008).

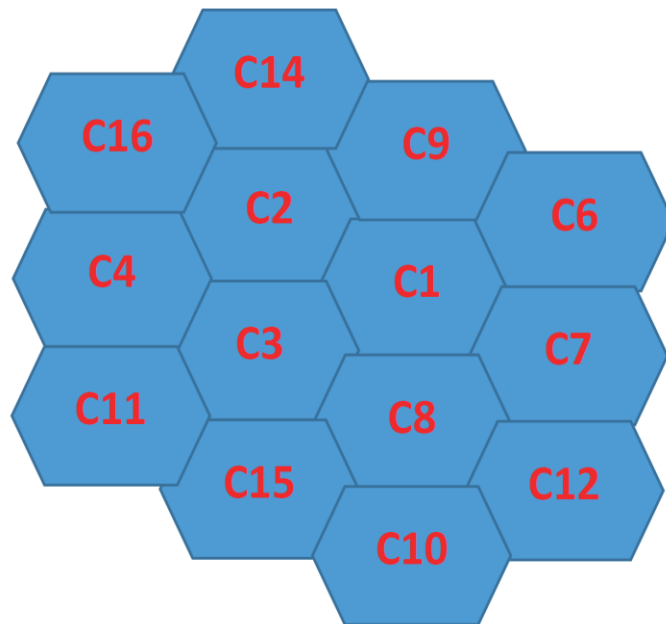


Figure 2. 7: Partitioning the entire region of the map into equal cells(Woldaregay et al., 2017; Yang & Abraham O. Fapojuwo, 2015).

For instance, consider cell 1 (C1) as shown in figure 2.7, to form a section of the city under surveillance as the others. This implies that, any subject, who is in C1 at the time of data submission are classified to be in C1. However, if a subject relocates to another cell such as C14 and the subject happen to submit the data, while in C14, then the subject is classified to be in C14. Computationally, this method is deemed inexpensive and its implementation is deemed to be very costly as high resources are required to divide the study area into equal hexagonal cells (Yang & Abraham O. Fapojuwu, 2015).

Another approach includes using the partitions of map of the study area such as the postal code areas (Musa et al., 2013). The diabetes subject's infection status occurs at specific geographical locations of latitude and longitudes which can be captured by geolocation enabled devices such as smart phones (Golden & Schell, 2008). The locations of such data points of infected individuals are classified to the nearest postal code regions. Hence, clustering is monitored at the postal code levels for aberrations (Heffernan et al., 2004). For instance, with reference to Figure 2.8, an infected subject located at the four start point (a) on the study area of the map would be classified into the nearest postcode (Kvaløysletta:9100) using suitable clustering algorithms.

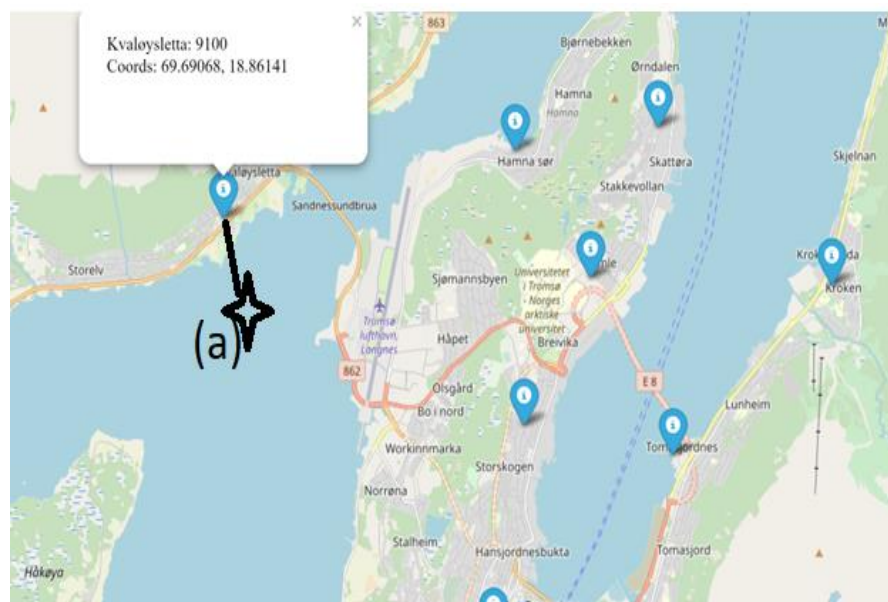


Figure 2. 8: Map of study area with postcodes

CHAPTER THREE: Literature Review

3.1 Literature Review

In EDMON, a systematic review of cluster detection mechanism for syndromic surveillance system was conducted (Yeng et al., 2018b). The aim was to pinpoint the state of the art cluster detection mechanisms for the implementation of a syndromic surveillance system in EDMON (Yeng et al., 2018b). Various challenges such as user mobility, geographical location estimation and other factors were considered.

Between January 2018 and March 2018, literature search was conducted through Google Scholar, Science Direct, PubMed, IEEE Xplore, ACM Digital Library and Scopus. Key words such as “Spatiotemporal Clustering”, “Syndromic Surveillance”, “Real Time”, “Cell Phone”, “Mobile Phone”, “Smart Phone”, trajectory, “Aberration Detection” and “Clustering” were used. These key words were combined with Boolean functions of ‘AND’, ‘OR’ and ‘NOT’. Peer-reviewed journals and articles were considered. The inclusions and exclusions criteria were developed based on the objective of the study and through rigorous discussions among the authors. While focusing on the inclusion and exclusion criteria, basic selection was done through reading the titles, abstracts and keywords for relevant literatures. Duplicates were removed and articles, which seems relevant, based on the inclusion and exclusion criteria, were fully read and judged. Reference lists were also the source of other relevant literatures. Preferred Reporting Items for Systematic Reviews and Meta-Analysis flow diagram was used to record the article selection and screening [42].

3.2 Inclusion and Exclusion Criteria

For an article to be included in the review, that article was expected to be a practically implemented syndromic surveillance system with cluster detection mechanisms. Practically implemented algorithms were being accepted because the result of the study was towards the development of a framework and practical implementation of a syndromic surveillance system in EDMON and such similar systems. The major focus includes articles relating to infectious disease surveillance such as influenza, cholera, Sever Acute Respiratory Syndrome (SARS) and Ebola Virus Disease. All the articles were also limited to human species and not plants or animals’ species. In a bit to judiciously use resources in this project the study focused on practically implemented algorithms in syndromic surveillance but limited exploration of theoretical and unimplemented algorithms in the study (Omicsonline, 2018). The study was also focused on English language only due to limited resources (Omicsonline, 2018). The publication type included journal articles, conference abstracts and presentations. There were no time restrictions and any other article outside the above stated scope were excluded in the study.

3.3 Data Collection and Categorization

The data collection and categorization were done with the guide of the objective of the study. through literature reviews and authors discussions. The categories have been defined exclusively to assess, analyzed and evaluate the literatures, as shown in Table 3.1.

Table 3. 1: Data categories and their definitions

Category	Definition
Clustering and Aberration Detection Algorithm	This category defines the kind of clustering and Aberration detection algorithm which the study has used and implemented.
Type of Clustering Algorithm	This category defines the type of algorithm. The type of algorithms includes spatial, temporal and spatiotemporal algorithms.
Threshold	This category defines the type of threshold used to generate alarms and alerts in the study.
Clustering Category	The clustering algorithms has been categorized (Fanaee-T, 2012). This dimension tags the specified clustering algorithm used to their respective category.
Design Method	This category indicates the design method such as prototype, participatory or joint application development, Agile or waterfall model, that has been used in implementing the system.
Evaluation Criteria	In this category, the evaluation criteria used in evaluating the algorithms have been specified.
Performance Metrics	This category specifies the performance metrics such as sensitivity, specificity, positive predictive value etc., which were used in the evaluation of the algorithms.
Type of Location	Different types of locations are being used in clustering. These include geolocation, postcodes, counties and many others. This category specifies the exact type of location which was used in the system.
Source of Location	The source of location is defined as the location where the type of location information was obtained from.

Nature of Location	The nature of the location is defining the state of the location as static or dynamic nature.
Visualization Tool used	This category also records the type of visualization tool used in the implementation of the visualization aspect of the system.
Display Report	This category records the type of visual displays (graphs, maps, time series etc.) which were implemented by the various systems in the study.
Design Layout	This category records the stages and processes used in the architectural design of the syndromic surveillance system. For example, a layout may consist of data acquisition, clustering and aberration detection and visualization (Sharip, 2006). While other design layout could include privacy preserving mechanisms, machine learning techniques in processing the data and other layers (Ali et al., 2016; Groeneveld et al., 2017).

3.4 Literature Evaluation and Analysis

Eligible literatures were assessed, analyzed and evaluated, based on the categories in table 3.1. Analysis was performed on each of the categories to evaluate the state-of-the-art approaches. Percentages of the attributes of the categories were determined based on the total number of counts (n) of each type of the attribute. Some articles used multiple categories, therefore, the number of counts of these categories could exceed the total number of articles of these articles presented in the study.

3.5 Principal Findings and Discussion

A total of 5,936 records were retrieved From Scopus, ACM, IEEE Explore, Google Scholar, PubMed and Science Direct, in the initial search of the literatures with only the keywords. Reading of the titles of the literatures led to an initial record exclusion of 4165 and removal of 97 duplicates. Skimming through the abstracts and keywords, led to a further removal of 1549 records. So, 125 literatures, which were fully read and judged. After full text reading, a total of 28 articles were included in the study and analysed as shown in Figure 3.1.

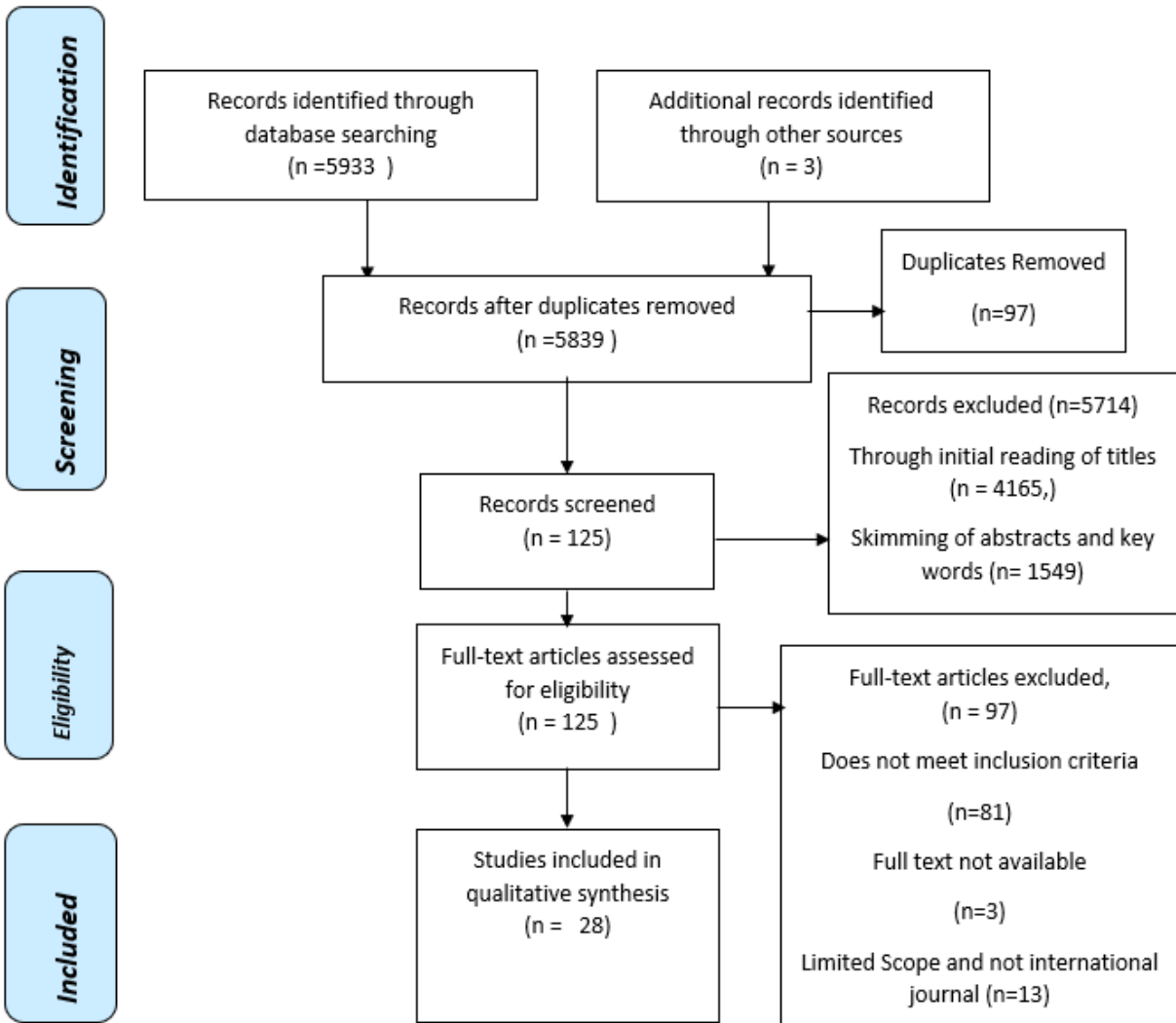


Figure 3. 1: Flowchart of the review process.

To this end, the study revealed a number of space, time and spatiotemporal algorithms including Space Time Permutation Scan Statistics (STPSS), K Nearest Neighbor (KNN), K means clustering, DBSCAN and Space Scan Statistics (SSS) (Yeng et al., 2018b). In the review, the principal findings are shown in table 3.2 below.

Table 3. 2: Principal findings on a systematic review of cluster detection mechanism for implementation

Category	Most Used
Clustering Algorithm	Space Time Permutation Scan Statistics
Type of Clustering	Spatiotemporal type
Threshold	Recurrence Interval
Algorithm Category	Threshold base Clustering
Design Method	Participatory Design
Evaluation Method	Simulation with historical data
Performance Metric	Sensitivity
Type of Location	Geocode
Source of Location	Patient Health Record
Nature of Location Source	Static
Visualization Tool Used	ArcGIS
Displayed Output	Maps
Layout	DCADAA

Considering table 3.3 and figure 3.2, at an average sensitivity and specificity of 82%, STPSS detected high cases of about 26. At a very high sensitivity and specificity up to 99.5%, the special and spatiotemporal algorithms detect some number of cases. Despite the disaggregated nature of data, the special and spatiotemporal clusters detecting some average number of cases at a higher range of sensitivity and specificity signifies good power of case detection (Abellan J J, 2007; Isobel et al., 2016; Mathes et al., 2017; Wang et al., 2010; Yih et al., 2010). With slightly lower sensitivity and specificity ranging from 82% to 92%, the temporal algorithms also detected some number of cases.

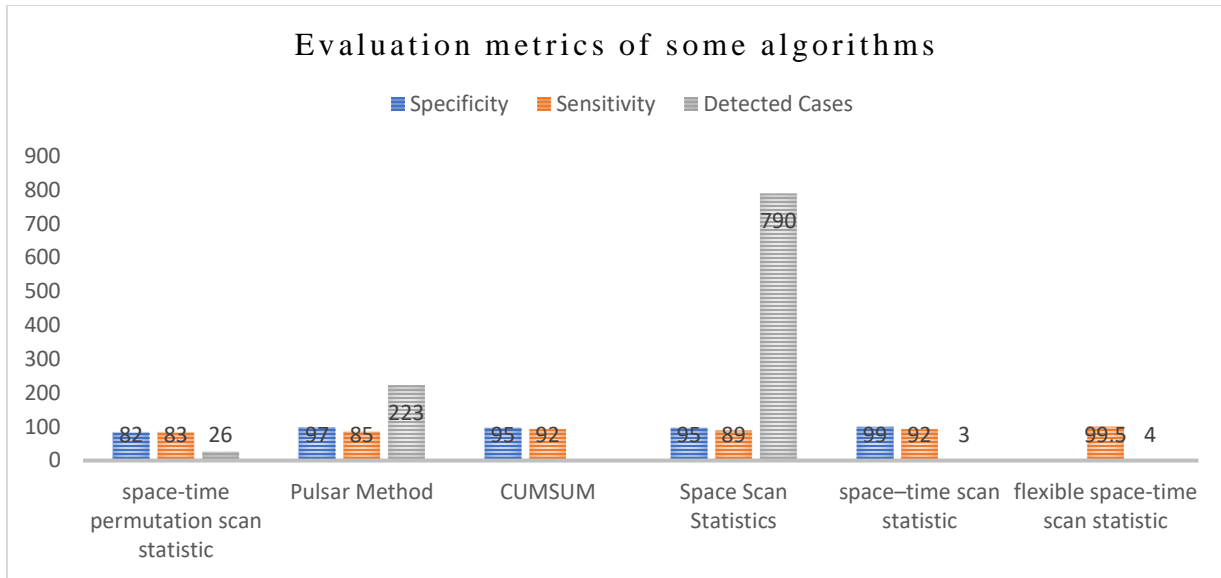


Figure 3. 2: Performance metrics of some clustering algorithms

Table 3. 3: Evaluation metrics of some algorithms

Algorithms	Specificity	Sensitivity	Detected Cases
space-time permutation scan statistic	82	83	26
Pulsar Method	97	85	223
CUMSUM	95	92	
Space Scan Statistics	95	89	790
space-time scan statistic	99	92	3
flexible space-time scan statistic		99.5	4

In using spatiotemporal clustering algorithms in syndromic surveillance, various methods such as temporal methods and near neighbors could be considered. These measures may augment for the increase in sparseness of data which causes loss of power to detect areas with local excess aberrations in spatial and spatiotemporal methods (Abellan J J, 2007; Isobel et al., 2016; Mathes et al., 2017; Wang et al., 2010; Yih et al., 2010).

An evaluation which was performed through injection of spikes of known outbreak revealed low detection in the space and spatiotemporal algorithms (Abellan J J, 2007; Isobel et al., 2016; Mathes et al., 2017; Wang et al., 2010; Yih et al., 2010). Spatial scan statistic detected 3% of all injects but a space time permutation scan statistic detected none at a specificity of 95% (Isobel et al., 2016). But the temporal algorithms detected higher percentages ranging from about 2% to 19%

percent of the injects under the same level of sensitivity (Abellan J J, 2007; Isobel et al., 2016; Mathes et al., 2017; Wang et al., 2010; Yih et al., 2010). The low detection in space and spatiotemporal algorithm was as a result that, the algorithms were not adjusted to increase their power of detection on disaggregated data (Abellan J J, 2007; Isobel et al., 2016; Mathes et al., 2017; Wang et al., 2010; Yih et al., 2010). Also, the performance of the algorithms could be enhanced with higher number of input cases and better coverage in spatial and spatiotemporal algorithms (Isobel et al., 2016). Spatial algorithms are implemented together with temporal algorithms to give the surveillance system the spatiotemporal properties (Duangchaemkarn et al., 2017). This approach has the tendency of boosting the power of cluster detection even when detected points are geographically dispersed (Abellan J J, 2007; Duangchaemkarn et al., 2017; Isobel et al., 2016; Mathes et al., 2017; Wang et al., 2010; Yih et al., 2010). Therefore, a combination of the state-of-the-art spatial algorithm (KNN) and temporal algorithms are being used to implement the cluster detection mechanism in EDMON.

Chapter 4: Materials and Methods

4.1 Introduction

A design Science research (DSR) and prototyping approach were adopted in the implementation of the EDMON-Cluster (Doyle, Sammon, & Neville, 2016; Offermann, Levina, Schönherr, & Bub, 2009). The problem of the study was identified through literature review which led to the development of the study objective and design. The results were evaluated and communicated in the discussion section.

The main task of this project was to develop a spatiotemporal cluster detection method for a syndromic surveillance system which can timely detect infectious disease outbreak. A dataset was simulated for certain time period containing an infected diabetes individual tagged with time and geographical location (postal codes). The synthetic data was classified into post code area using k-nearest neighbor algorithms. This data was further classified into temporal dimension using the cumulative summation (CUSUM), algorithm. The combination of the spatial algorithm (nearest neighbor) and temporal algorithm (CuSUM) formed the spatiotemporal method (Fanaee-T, 2012). Deidentification and one-way hashing technique were adopted for preserving privacy of the subjects involved in the study. A prototype and system development life cycle approach were adopted for the implementation of the system. The output was displayed on maps with indications of the level of aberrations from the baseline mean. Comparison method was used to evaluate the classification algorithm. Sensitivity, Specificity and Positive Predictive Values of the detection algorithm were determined while System Usability Scale (SUS) was used to assess the usability of the system. Additionally, a visualization module was implemented to visualize various dimensions. The detection time of the surveillance system, privacy preserving technique and other performances measures were also evaluated. The entire study was organized under materials and method sections.

4.2 Materials Used

4.2.1 Synthetic data used

Various synthetic data were simulated through DSR, conceptualization and through feedback from experts in the iterations of the prototype, for usage in the project. The simulated datasets were 21 post codes centroids in the Tromsø area of Norway, unclassified data as test set and classified or reference dataset as training set. To overcome over fitting, under fitting and class imbalance issues, 660 training and 209 testing datasets of approximately, 70%: 30% were randomly simulated (Cochran, 1977; Liu & Cocea, 2017). The postcode dataset majorly has the centroid coordinate features of Latitude (Lat) and Longitude (Lon) for each observable with postcode (Code) feature which was used as a response vector. The postcodes were also tagged to their various places as shown in table 4.0.

Table 4. 1: Simulated Centroid of post codes of study area

Lat	Lon	Code	Centroid_ID	Place
69.55799	19.33103	9027	1	Ramfjordbotn
69.57781	18.55499	9106	2	Straumsbukta
69.627957	18.915001	9006	3	Lanesvegen
69.63077	19.04736	9020	4	Tromsdalen
69.63702	17.981	9110	5	Sommarøy
69.640574	18.927288	9007	6	Kveldrovegen
69.64225	18.90889	9013	7	Tromsø
69.65079	18.95493	9008	8	Tromsø
69.251024	18.54714	9272	9	Tromsøya

The synthetic subjects in the study were also simulated to contain the location coordinates and Date stamp of where and when the infection incidences occurred in the form of Lat, Lon and D_Date. The detection values and some personal identifiable features such as Names and IDs are shown in Table 4.2. The unclassified simulated dataset lacked postcode attribute and therefore, the detections were required to be classified into their respective postcode areas.

Table 4. 2: Unclassified Data

DID	Lat	Lon	PID	D_Date	Detections	Name
1	69.55799	19.33103	1	20.01.2018 15:00	1	Alexander
2	69.57781	18.55499	2	21.01.2018 18:00	1	Bjørn
3	69.627957	18.915001	3	22.01.2018 6:00	1	Andreas
4	69.63077	19.04736	4	23.01.2018 15:00	1	Daniel
5	69.63702	17.981	5	24.01.2018 18:00	1	Frank
6	69.640574	18.927288	6	25.01.2018 6:00	0	Erling
7	69.64225	18.90889	7	26.01.2018 15:00	1	Geir
8	69.65079	18.95493	8	27.01.2018 18:00	1	Harald
9	69.651024	18.954714	9	28.01.2018 6:00	0	Mat
10	69.65103	18.9557499	10	29.01.2018 15:00	1	Jan
11	69.65494	18.95376	11	30.01.2018 18:00	-1	Håkon
12	69.66153	18.94791	12	31.01.2018 6:00	1	Johan
13	69.66299	18.96545	13	2.01.2018 15:00	1	Jørgen
14	69.68476	18.98854	14	2.02.2018 18:00	0	Kenneth
15	69.69068	18.86141	15	2.04.2018 6:00	1	Marius
16	69.69755	18.96246	16	4.02.2018 15:00	0	Thomas
17	69.69995	19.01186	17	5.02.2018 18:00	1	Nils
18	69.649	18.955	18	6.02.2018 6:00	-1	Ola
19	69.667	19.017	19	7.02.2018 15:00	-1	Vegard

Table 4. 3: Classified synthetic data of people with type-1 diabetes

DID	Lat	Lon	Code	PID	D_Date	D	Name
1	69.55799	19.33103	9027	1	20.01.2018 15:00	1	Alexander
2	69.57781	18.55499	9106	2	21.01.2018 18:00	1	Bjørn
3	69.62796	18.915	9006	3	22.01.2018 6:00	1	Andreas
4	69.63077	19.04736	9020	4	23.01.2018 15:00	1	Daniel
5	69.63702	17.981	9110	5	24.01.2018 18:00	1	Frank
6	69.64057	18.92729	9007	6	25.01.2018 6:00	1	Erling
7	69.64225	18.90889	9013	7	26.01.2018 15:00	1	Geir
8	69.65079	18.95493	9008	8	27.01.2018 18:00	1	Harald
9	69.65102	18.95471	9272	9	28.01.2018 6:00	1	Mat
10	69.65103	18.95575	9037	10	29.01.2018 15:00	1	Jan
11	69.65494	18.95376	9009	11	30.01.2018 18:00	1	Håkon
12	69.66153	18.94791	9011	12	31.01.2018 6:00	1	Johan
13	69.66299	18.96545	9010	13	2.01.2018 15:00	1	Jørgen
14	69.68476	18.98854	9019	14	2.02.2018 18:00	1	Kenneth
15	69.69068	18.86141	9100	15	2.04.2018 6:00	1	Marius
16	69.69755	18.96246	9017	16	4.02.2018 15:00	1	Thomas
17	69.69995	19.01186	9018	17	5.02.2018 18:00	1	Nils
18	69.649	18.955	9016	18	6.02.2018 6:00	1	Ola

As shown in table 4.3, a synthetic dataset was also simulated to represent the classified dataset of the detected infection incidences. Each subject with a detection ID (DID), location features of Lat and Lon and temporal feature of timestamp was classified into their respective response vectors of post codes (Code) area using the KNN algorithm. The classified dataset was also used as a training dataset for the KNN algorithm during classification of new observables in the unclassified dataset.

4.2.2 Synthetic data generation data quality

The data was manually generated by first, creating estimated decimal degree coordinates of centroids of postcode areas using google GPS coordinate lookup system as shown in figure 4.1.

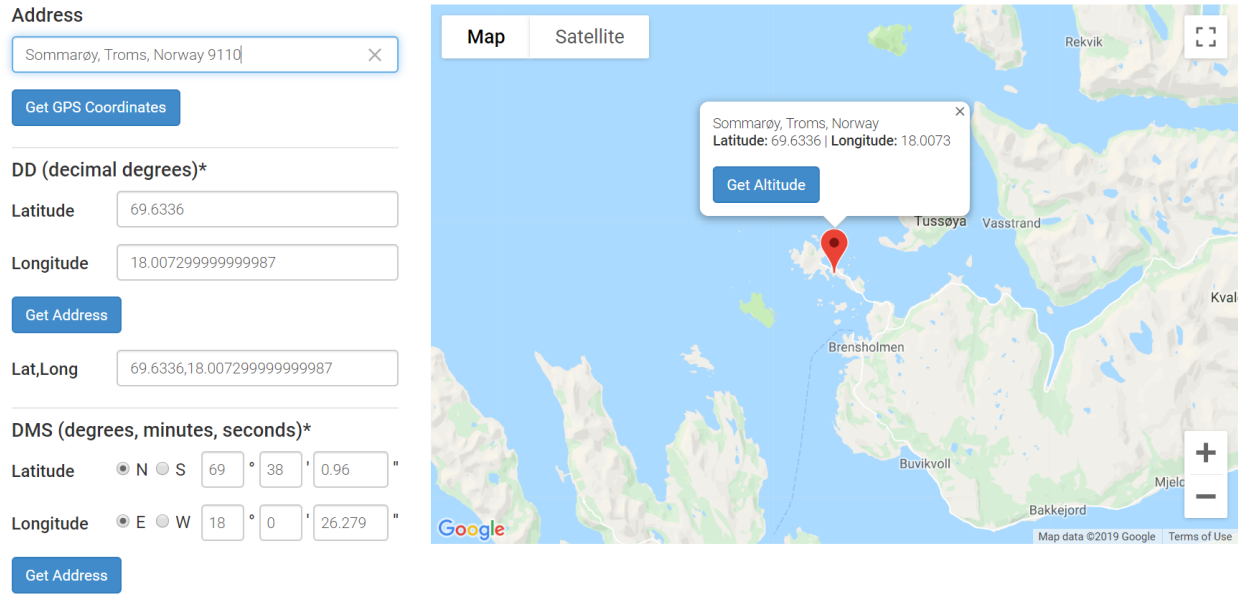


Figure 4. 1: Source of simulated geocoordinates (gps-coordinates, 2019).

Decimal Degree Coordinates (DDC) were then varied randomly to create artificial locations for the fictitious data subjects. The random variation of the DDC could overlap to other post code areas which might introduce some errors. However, carefulness was being taken not to introduce large degree of variations of the DDC. About 350 classified synthetic data was created and this was used to classify about 303 unclassified datasets.

4.2.2 System development tools used

Various materials were used at the system development stage. These include the Python, Leaflet.js, DC.js, D3.js, Crosfilter.js and visual Studio Code. Python was used as a programming and data manipulation language. The leaflet.js was used to display the map for visualization alarm and alert of the surveillance system. DC, D3 and Crossfiliter were also used to visualize the classified data into different dimensions. Visual Studio Code was used as a source code editor for the python. The details such as type and purpose of the material, and their versions are shown in table 7.0 below.

Table 4. 4: Programming tools used

No.	Material	Type/Purpose	Version
1.	Python	Programming/Data manipulation language	3.6
2.	Leaflet.js	JavaScript Library	4.2
3.	DC.js	JavaScript Library	3.0.10
4.	D3.js	JavaScript Library	5.7.0

5.	Cross-filter.js	JavaScript Library	2.0
6.	Visual Studio Code	Node.js	1.3.3

A pictorial representation of the Leaflet.js is shown in figure 4.2. Further a laptop of, a 64-bit windows 10 operating with I5 processor, 8 Gigabyte of RAM and 150Gigabyte of hard disk drive was used in the study as shown in Figure 4.3

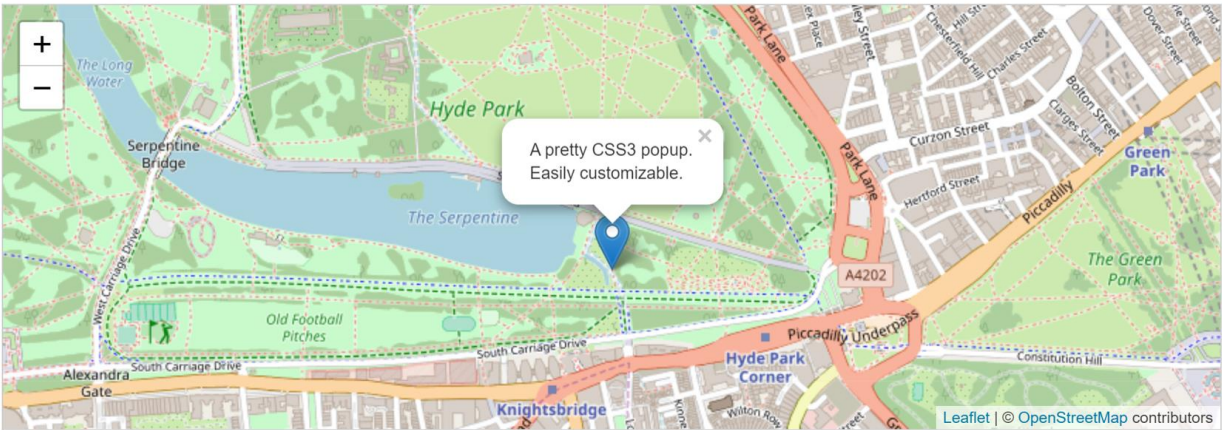


Figure 4. 2: Leaflet-Open Source JavaScript Library for interactive Map

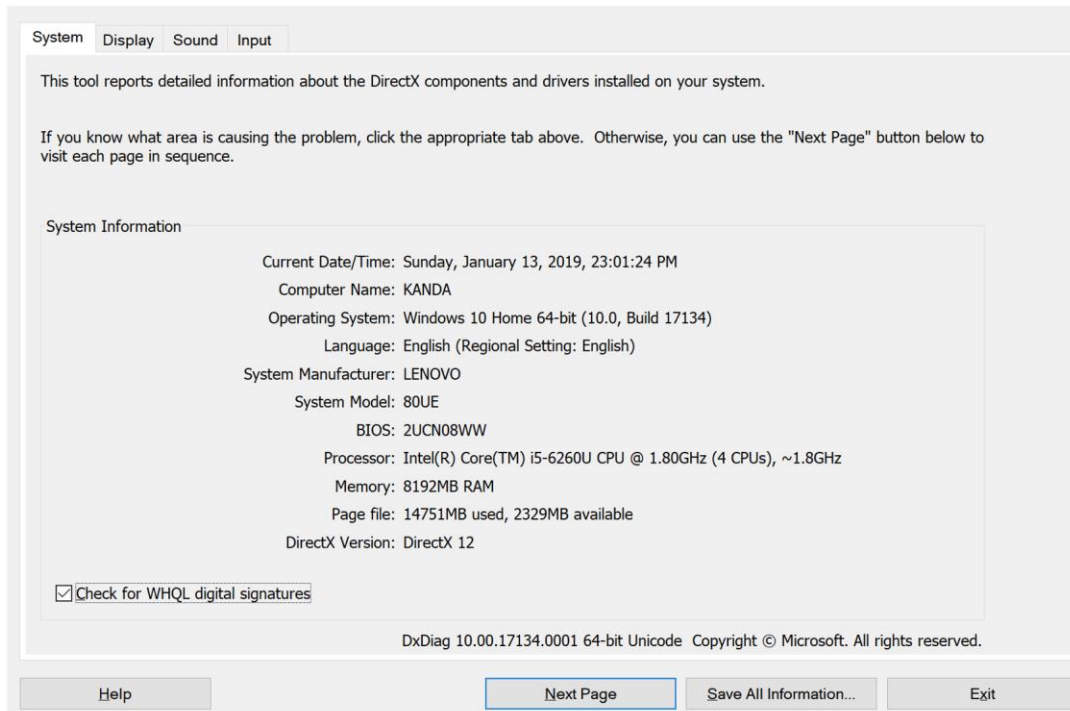


Figure 4. 3: operating system and hardware features of laptop used

4.3 Methods Used

Different methods were used for various purposes in this project work. The methods include prototyping, KNN, CUSUM, Privacy Preserving Mechanisms (PPM) and performance measures as shown in Table 4.5. The method column in the Table 4.5 shows the methods that were used in the study. The category column defines implementation or evaluation aspect in which the specified methods were applied. The purpose column explains the role in which the method played in the study.

Table 4. 5: Methods used for implementation and evaluation

No	Method	Category	Purpose
1	Prototype	Implementation	Requirement and system functionality measures
2	KNN	Implementation	Clustering
3	CUSUM	Implementation	Aberration Detection

4	PPM	Implementation	Privacy Preserving
5	SciKitlearn	Evaluation	Evaluating KNN method
6	Sensitivity (Se),	Evaluation	Evaluating the Efficacy of the surveillance system
7	Specificity (Sp)	Evaluation	Evaluating the Efficacy of the surveillance system
8	Positive Predictive Value (PPV)	Evaluation	Evaluating the Efficacy of the surveillance system
9	Detection Time	Evaluation	Evaluating the Efficacy of the surveillance system

The performance metrics such as the Se and Sp adopted the methods in table 4.6 while the detection time also adopted the structure of figure 4.3.

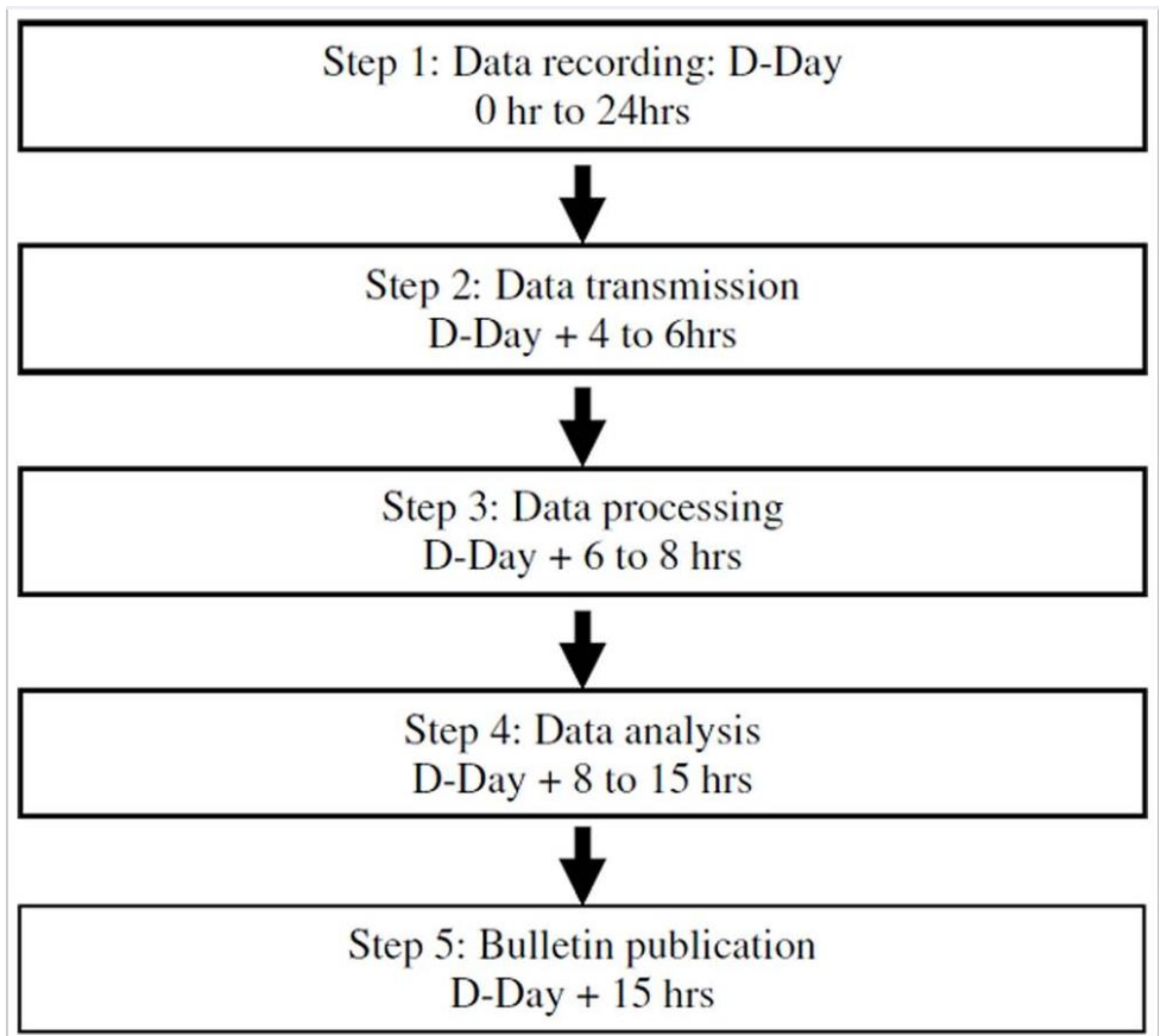


Figure 4.4: Clustering and Detection Time (Josseran et al., 2010)

Table 4. 6: Sensitivity and Specificity analysis (Josseran et al., 2010)

		ONAP		
		Yes	No	
Alert Days - Syndromes	Yes	True positive A	False positive B	A + B
	No	False negative C	True negative D	C + D
		A + C (13 days)	B + D (79 days)	

ONAP: On Alert Period

Sensitivity (Se) = $A/A+C$
 $IC_{95\%(Se)} = Se \pm 1.96 * \sqrt{(Se * (1 - Se) / n)}$

Specificity (Sp) = $D/B+D$
 $IC_{95\%(Sp)} = Sp \pm 1.96 * \sqrt{(Sp * (1 - Sp) / n)}$

Predictive positive value (PPV) = $A/A+B$
 $IC_{95\%(PPV)} = PPV \pm 1.96 * \sqrt{(PPV * (1 - PPV) / n)}$

4.4 Justification and Critique of the methods

Prior to the implementation of the clustering methods in EDMON (EDMON-Cluster), a systematic review was conducted, (Yeng et al., 2018a), to explore for methods including, algorithms, evaluation techniques, visualization methods and other dimensions. The results were intended to be used for implementing an efficient syndromic surveillance system. Various dimensions of the results were obtained in which this implementation primarily depends on. In addition to relying on the results of the systematic review, various reasons guided the choice of the implementation materials and methods as follows;

4.4.1 Synthetic Data

With regards to the data sources, synthetic data was simulated for the implementation. In syndromic surveillance system development, the application needs to be tested and results evaluated with data to assess performance and robustness regarding erratic data requirements (Jafarpour Khameneh, 2014; Kajita et al., 2017). Apparently, actual data or semi-synthetic data can be used in the assessment (Chen H, Zeng D, & P., 2010; Karami, 2012), however, there are regulatory hurdles and stringent privacy laws across the globe (Beredskapsdepartementet, 2018; e-helse, 2019). Further to this healthcare data is considered to be one of the most sensitive personal data which cannot be toyed with (ISO, 2016). Additionally, surveillance data for many of the disease outbreaks do not exist (Karami, 2012). To succeed in implementing this prototype in the midst of these challenges, synthetic data is an obvious choice. Synthetic data serve as a playground or surveillance range which can be manipulated in different ways to test the scalability or the robustness of new algorithms without transgressing on privacy laws (Burgard, Kolb, Merkle, & Münnich, 2017; RIAKTR, 2016). The test results can also be shared broadly, as open data without regulatory impedance (Burgard et al., 2017; RIAKTR, 2016).

4.4.2 Algorithms

In EDMON framework, the outbreak detection was to be done in both space and time in order for users to know where and when the aberrations are occurring (Woldaregay et al., 2017). The systematic review towards the implementation of cluster detection mechanism in EDMON revealed various algorithms that could be used to achieve the spatiotemporal objective of EDMON (Yeng et al., 2018a). Space Time Permutation Scan Statistics (STPSS) was found to be the most used algorithm for spatiotemporal measures. STPSS does not require population at risk data to draw the expected baseline value. But it dwells on the detected cases to determine the expected count (Kulldorff, 2005). This approach provides significant trend of baseline data while avoiding inclusion of historical data that is irrelevant to the current period. However, STPSS has a major drawback. STPSS, algorithm is only efficient on outbreaks that start locally (Kulldorff, 2005). This suggests that STPSS is not suitable for detecting disease outbreaks which occur simultaneously in the entire surveillance area. STPSS is only efficient on disease outbreaks with higher rate of early symptoms (Kulldorff, 2005). It has low power of detection in geographically disaggregated data. But this gap of low power of detection can be filled by combining temporal methods and near neighbors' methods (Abellan J J, 2007; Isobel et al., 2016; Mathes et al., 2017; Wang et al., 2010; Yih et al., 2010). This measure could boost the power of detection in sparseness of data through local excess aberration detections in spatial and spatiotemporal methods (Abellan J J, 2007; Duangchaemkarn et al., 2017; Isobel et al., 2016; Mathes et al., 2017; Wang et al., 2010; Yih et al., 2010). Therefore, a combination of the state-of-the-art spatial algorithm (KNN) and temporal algorithms are being explored to implement the cluster detection mechanism in EDMON.

4.4.3 Evaluation methods

Various evaluation techniques exist but what is relevant to syndromic surveillance systems include performance metrics such as timeliness of detection, sensitivity and specificity (Yeng et al., 2018a). Additionally, the accuracy of the classification is equally important which is mostly evaluated with comparison with known outbreaks and simulation with historical data (Yeng et al., 2018a). But in synthetic data where historical patterns of outbreaks are not known, comparison of the classification accuracy can be conducted with injects of spikes of outbreaks.

To overcome over fitting, under fitting and class imbalance issues, 660 training and 209 testing datasets of 70%: 30% were randomly simulated (Cochran, 1977; Liu & Cocea, 2017).

6.4.4 Visualization, Alarm and Alerts

The main output of the framework includes timely alerts through alarms and visualizations of detected aberrations. From the studies, various visualization tools for output displays such as bar charts, pie charts, graphs and maps have been realized. Guided with the results of the systematic review (Yeng et al., 2018a), ArcGIS, Leaflet-Open Source or Google Map tool was used to

implement the visualization module such as what was used in Google flu trend visualization and Flu near you as shown in figure 4.4. This visual display would mainly be map with other displays such as time series and graph. The maps would indicate where and when clustering and aberrations occur Leaflet map was chosen for the prototype due to it being open source, less expensive and does not require acquiring license to use (Leafletjs, 2019). Also, alerts would be triggered through alarms and messaging. The short messaging service (SMS) was created with a trial version of an application development interface (API) known as Twilio (Twilio, 2019). The Twilio API was selected based on cost, ease and flexibility of use.

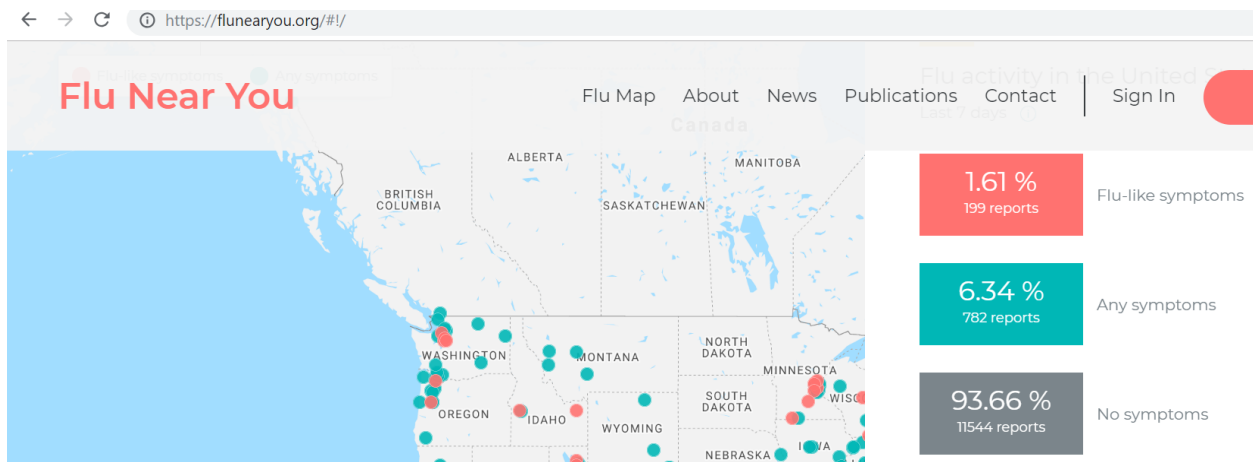


Figure 4. 5: Sample map for visualization (Flu Near You, 2019)

On privacy preserving and data security of the diabetes subjects, MD5, one-way hashing algorithm and nulling were selected for implementation. The choice was guided by GDPR on anonymization of personal data for research purpose as shown in figure 4.6

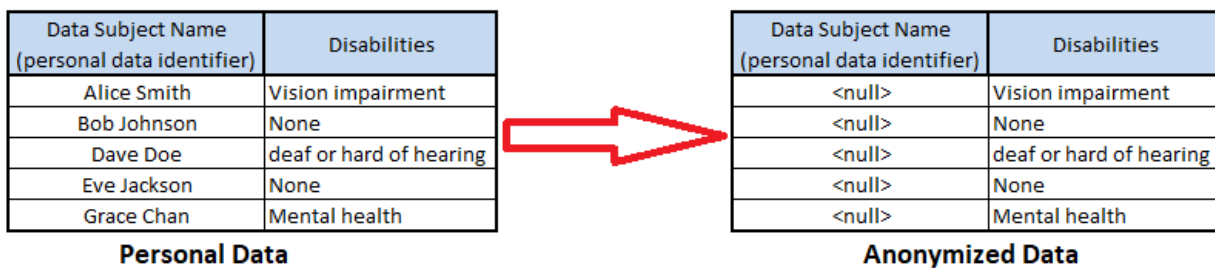


Figure 4. 6: Nulling technique of anonymization

Chapter Five: System Analysis

5.1 Introduction

According to (Avison & Fitzgerald, 2003; UKEssays, 2018), system analysis is the understanding of the goals and purpose of a business area to develop a system and procedures with the aim of achieving these goals and purposes in the most efficient fashion. In system development, detailed analysis is performed to obtain specific data which helps to meet both functional and non-functional requirement of the business study area. A general explanation has therefore been given in this chapter regarding the system analysis for requirement gathering and specification for the EDMON-Clustering project. In the development of this project, higher consideration was given to the clustering of elevated BGs, aberration detections and alerts and visualization of various dimensions. The concentration was on the development of the required software application and its associated evaluation without consideration for the conversion of the system to mobile application version due to time constraint.

First, the system was described in considering several assumptions such as the potential use of the system and related dependences. This was followed by the reasons of the choice of methods used for the requirement specification and the requirements sources. Both functional and non-functional requirements were specified for the whole system and a summary of the chapter was provided.

5.2 System Description

This project presents a prototype of a syndromic surveillance system (EDMON-Cluster). The system uses synthetic data, simulated as type-1 diabetes persons data who have recorded their BG dynamics in space and time in the Tromsø area of Norway. The system classifies infection incidence in people with type 1 diabetes into postal-code spaces and establishes a one-week baseline of infection incidences. This is used to detect possible aberrations of subsequent infection incidences to be investigated for possible disease outbreaks. EDMON-Cluster is aimed to be used by public health department of the health ministry. The system would be able to;

- Provide overview of infectious disease status in various postal code areas.
- Indicate potential outbreaks in various postal codes for investigation and other actions.
- Provide visualization of various dimensions in the surveillance data.

5.2.1 Constraints

- Privacy preserving would be required to protect data subjects of their right to privacy (Beredsksdepartementet, 2018)
- The EDMON-Cluster shall use nearness neighbor and distance metrics for the clustering or aggregation.

5.2.2 Users or Stakeholders

Presumably, the users of the system include but not limited to;

- Public health unit of the ministry of health

- Diabetes patients, families and relatives
- General Practitioners (GPs), physicians, nurses and other healthcare workers

5.2.3 Interoperability & Communication

EDMON-Cluster is designed to use a common data conversion and importing standard such as comma separated value (CSV) file. It is assumed daily data from the BG dynamics detection system were to be imported into the EDMON-Cluster. The data is then classified in combination with GDPR recommended anonymization standards for deidentification of the individual's data to preserve the confidentiality.

5.3 Requirement gathering and analysis

In system development, requirements and their respective specifications are much needed. Various requirements methods with their associated advantages and disadvantages exist. These requirement methods usages therefore depend on the problem scenarios in which they suit best (Robertson & Robertson, 2006).

In EDMON-Cluster project, the requirements are uncertain and needs an iterative approach that brings change at any course of time during development. This therefore requires a method that can facilitate this kind of task. The business requirements analysis includes eliciting and documenting the different business modules of the users' requirements, modelling, analyzing and documenting for the purpose of system design (Robertson & Robertson, 2006). There are various methods used in system development. These include but not limited to the Waterfall model, the prototyping, Incremental, Spiral and Rapid Application Development (RAD).

In the Waterfall model, the system development life cycle (SDLC) is partitioned into stages of which each stage must be entirely completed before subsequent stages. The output is then obtained from one stage to the next with the flow of progress moving from top to bottom, in a waterfall nature (PK.Ragunath, S.Velmourougan, P. Davachelvan, S.Kayalvizhi, & R.Ravimohan, 2010). The waterfall development model is known to be highly structured and basically costly (Robertson & Robertson, 2006). The Waterfall model can be easily managed because of its rigidity in nature, it's simple and easy to use. Additionally, the waterfall model is into phases, has specific output and a review process, making it suitable for smaller projects since the requirements are more clearly defined. But the downside of the Waterfall model includes challenges in adjusting of the scope during the project life cycle which can disrupt the project entirely (Robertson & Robertson, 2006). Also, implementation is not done until later part of the life cycle which is quite risky and is associated with higher degree of uncertainty. It is deemed not suitable for complex and object orientated, long and ongoing projects and where requirements are subjected to changes (PK.Ragunath et al., 2010).

Prototyping involves building, testing, and iterative reworks as necessary, and early approximation of the final system until an acceptable product of the prototype is finally achieved from which the complete system or product can be completely developed (Ashwin, 2017; Tavolato & Vincena, 1984). The prototyping method is most relevant in circumstances where the project requirements are not priority known in detail. This iterative trial-and-error process often occur between the developers and the user's model (Ashwin, 2017; Tavolato & Vincena, 1984). The Iterative process begins with a very basic and simple implementation of how the software requirements were understood by the developers and iteratively enhances the evolving versions until the full system is implemented as indicated in the diagrammatic view in Figure 5.1.

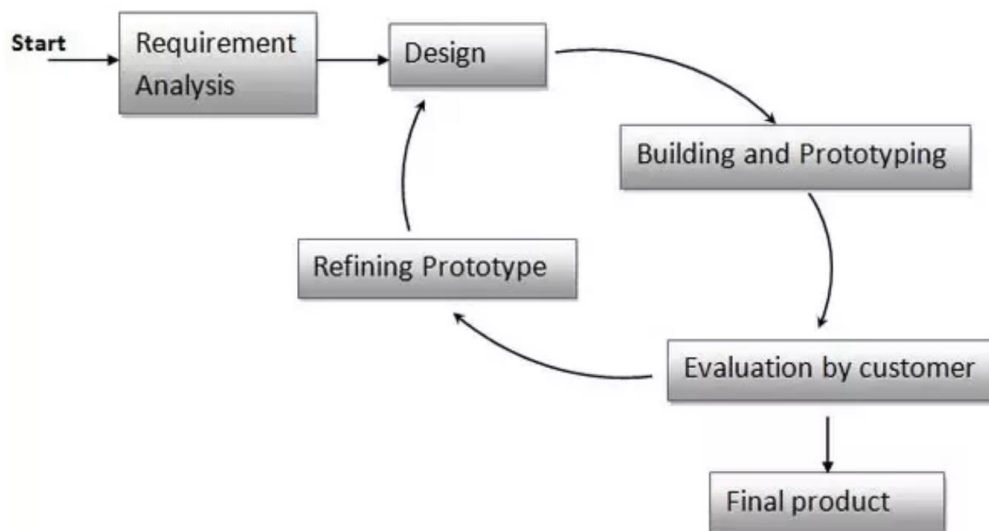


Figure 5. 1: diagrammatic view of prototyping model (Kenneth, 1986)

In the requirements determination aspect of the systems analysis phase, information about the organization's operations procedures and processes which are related to the proposed information system are gathered (Kenneth, 1986). User interviews and documents review are also carried out. The proposed system requirements can then be gathered from such a study. Using prototyping can augment this entire process because it converts the initial basic understanding of the users' intangible specifications into a tangible but limited working version of the expected information system. The users' desires, learned from developing the tangible prototype, simplifies an evaluative response that can be used to enhance the existing requirements while developing new requirements (Ashwin, 2017; Tavolato & Vincena, 1984).

Prototyping therefore reduces development time, cost, encourages user participation, provides quantifiable user feedback and results in higher user satisfaction among others. Though prototyping can lead to insufficient analysis, and may cause incomplete documentation, it has increasingly been adopted due to its high-quality product and enhanced user participation in the product development (Ashwin, 2017; Tavolato & Vincena, 1984).

5.4 Source of Requirements

The source of requirements is a prerequisite for a successful project implementation since it helps to realize stakeholder's involvement and other vital sources. Functional requirements are features of the system which are combined to form a comprehensive and coherence of the system which can be executed to perform the specified task. Functional requirements can therefore be concretely measured with data values, decision making logic and algorithms (Robertson & Robertson, 2006). Non-functional requirements are the intangible or invisible attributes that the system must possess such as, security, performance, usability and others (Robertson & Robertson, 2006). Relevant literature reviewed in Chapter 3 and the prototyping iteration method of do-review-update approach were adopted for both functional and non-functional requirement.

5.5 Functional Requirement

This section basically describes the specified functional requirement in relation to the sources outlined in the source of requirement section.

Table 5. 1: Functional Requirement number 1

Requirement #: 1	Event/Use case #: 1
Description: The system shall load post codes study area's data	
Rationale: For processing and clustering.	
Source: Author and Background knowledge from Literatures	
Fit Criterion: Each post code should contain the centroid geographical coordinates	
Dependencies: None	Conflict: None

Table 5. 2: Functional Requirement number 2

Requirement #: 2	Event/Use case #: 2
Description: The system shall load unclassified infection status data from BG dynamics detection sources.	
Rationale: For processing and classification.	
Source: Author and Background knowledge from Literatures	
Fit Criterion: The data shall be accessed by permitted modules of the system.	
Dependencies: None	Conflict: None

Table 5. 3: Functional Requirement number 3

Requirement #: 3	Event/Use case #: 3
Description: The system shall load classified detected BG dynamics data Classified data file	
Rationale: For processing and clustering.	
Source: Author and Background knowledge from Literatures	
Fit Criterion: The data shall be accessed by permitted modules of the system.	
Dependencies: None	Conflict: None

Table 5. 4: Functional Requirement number 4

Requirement #: 4	Event/Use case #: 4
Description: The system shall classify infected individuals	

Rationale: For clustering base on post code areas.
Source: Author and Background knowledge from Literatures
Fit Criterion: The elevated BGs shall be aggregated to their post code areas
Dependencies: 2 Conflict: None

Table 5. 5: Functional Requirement number 5

Requirement #: 5	Event/Use case #: 5
Description: The system shall form a baseline	
Rationale: The base line is for detecting aberrations.	
Source: Background knowledge from Literatures	
Fit Criterion: The base line data shall be formed from the previous one week	
Dependencies: 3 and 4 Conflict: None	

Table 5. 6: Functional Requirement number 6

Requirement #: 6	Event/Use case #: 6
Description: The system shall form observed counts	
Rationale: To compare with baseline for detecting aberrations.	
Source: Background knowledge from Literatures	
Fit Criterion: The observed count data shall be formed from the current week	
Dependencies:3 and 4 Conflict: None	

Table 5. 7: Functional Requirement number

Requirement #: 7	Event/Use case #: 7
Description: The developed system shall compare baseline data to observed count	
Rationale: To detect deviation from baseline.	
Source: Background knowledge from Literatures	
Fit Criterion: The result shall agree with the thresholding mechanism	
Dependencies: 5 and 6 Conflict: None	

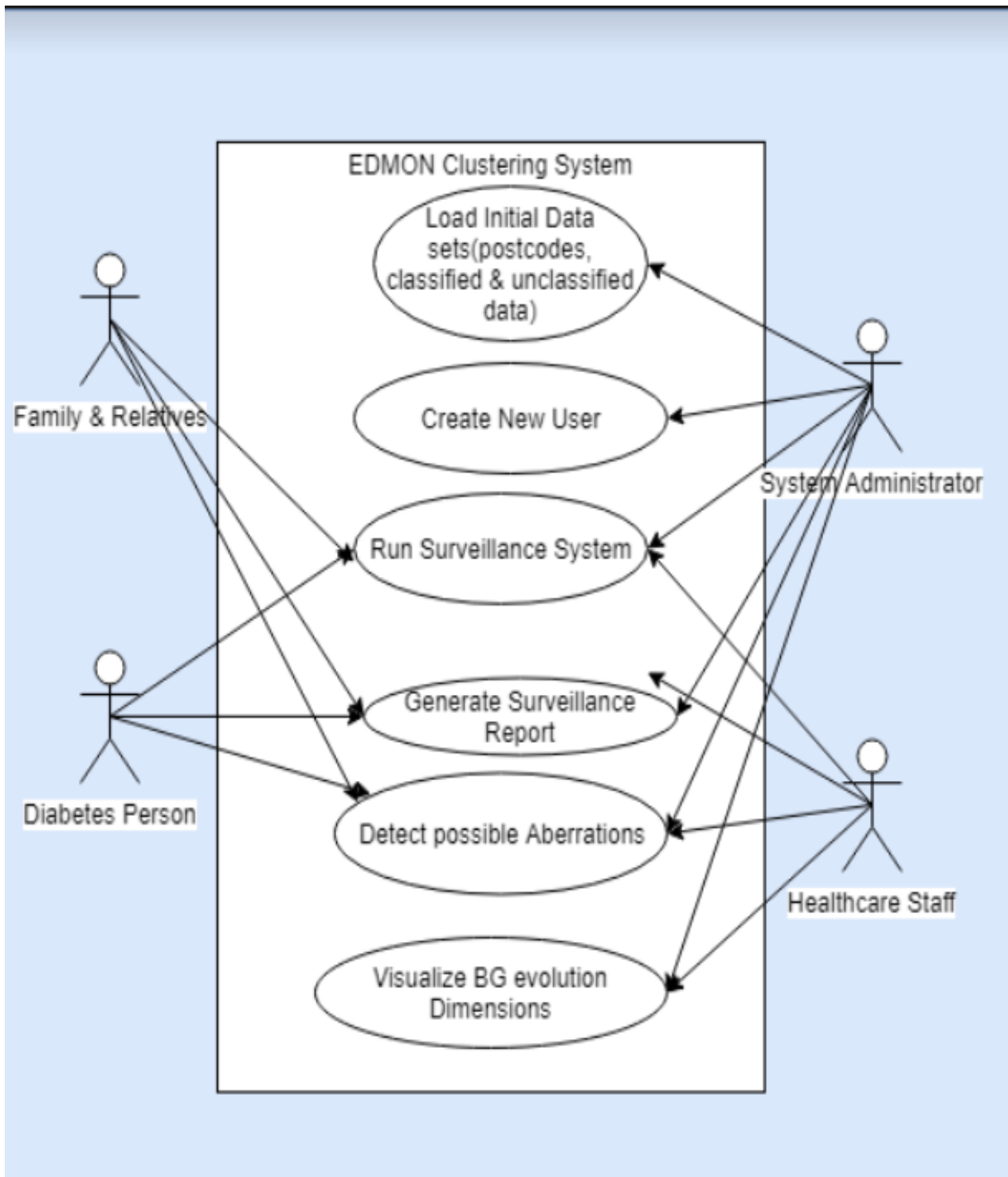


Figure 5. 2: Use Case Diagram

Table 5. 8: Functional Requirement number 8

Requirement #: 8	Event/Use case #: 8
Description: The system shall detect significant BG elevations at postcode area	
Rationale: To detect outbreak of infectious diseases.	
Source: Author Background knowledge from Literatures	
Fit Criterion: The system should detect elevated BG aberrations	
Dependencies: 1,2,3,4,5,6 and 7	Conflict: None

Table 5. 9 : Functional Requirement number 9

Requirement #: 9	Event/Use case #: 9
Description: The developed system shall indicate a detection status for notification	
Rationale: For users to be aware of possible infectious disease outbreak.	
Source: Background knowledge from Literatures	
Fit Criterion: The cluster status should indicate green, yellow or red to show no outbreak, near outbreak or outbreak respectively	
Dependencies: 1,2,3,4,5,6,7 and 8	Conflict: None

Table 5. 10: Functional Requirement number 10

Requirement #: 10	Event/Use case #: 10
Description: The developed system shall store the clustering results	
Rationale: For users to be able to view, visualize and analyze at any point in time.	
Source: Author and Background knowledge from Literatures	
Fit Criterion: Users can visualize various dimensions of the classified data	
Dependencies: 1,2 and 3	Conflict: None

5.6 Use Case

From the requirement specification above, use cases were developed to depict the interaction between actors and the system features as shown in Figure 5.1.

Use case #: 1

Purpose: Get the post codes study area's data.

The sequential events are as follows;

1. System checks post code file for validity
2. System reads the data from the post code file
3. System formats the data
4. System stores the data

Alternatives:

1.1 if postcode file is not valid, generate error message for administrators' attention

2.1 If the algorithm fails to read the data, generate error message for administrators' attention

4.1 if write error exists, generate error message for administrators' attention

Exceptions:

None Identified.

Use case #: 2

Purpose: Get unclassified detected BG dynamics data from BG dynamics detection sources.

The sequential events are as follows;

1. System checks unclassified data file for validity
2. System reads the data from the unclassified data file
3. System formats the data
4. System stores the data

Alternatives:

1.1 if unclassified data file is not valid, the system should generate error message for administrators' attention

2.1 If the algorithm fails to read the data, the system should generate error message for administrators' attention

4.1 if write error exists during storage, the system should generate error message for administrators' attention

Exceptions:

None Identified.

Use case #: 3

Purpose: Get infected persons data from classified data file

The sequential events are as follows;

1. System checks classified data file for validity

2. System reads the data from the classified data file
3. System formats the data
4. System stores the data

Use case #: 4

Purpose: Classify the unclassified elevated detected BG dynamics data

The sequential events are as follows;

1. System checks unclassified data file for data extraction
2. System sequentially reads the data
3. System compares the distance of BG detection location to the distance of other BG detection locations in various postcode areas
4. System classifies the detected BG elevated location to the nearest post code area
5. System stores the data
6. System indicates the classification on a map

Alternatives:

1.1 if unclassified data file is not valid, the system should generate error message for administrators' attention

2.1 If the algorithm fails to read the data, the system should generate error message for administrators' attention

5.1 if write error exists during storage, the system should generate error message for administrators' attention

Exceptions:

None Identified.

Use case #:5

Purpose: Form a baseline

The sequential events are as follows;

1. System checks classified data file for past week BG elevations
2. System reads the detections per post code area

3. System sums the number of BG elevations per post code area
4. System stores the data

Alternatives:

- 1.1 if classified data file is not valid, the system should generate error message for administrators' attention
- 2.1 If the algorithm fails to read the data, the system should generate error message for administrators' attention
- 4.1 if write error exists during storage, the system should generate error message for administrators' attention

Exceptions:

None Identified.

Use case #: 6

Purpose: Form observed counts

The sequential events are as follows;

1. System checks classified data file for current week BG elevations
2. System reads the detections per post code area
3. System sums the number of BG elevations per post code area
4. System stores the data

Alternatives:

- 1.1 if classified data file is not valid, the system should generate error message for administrators' attention
- 2.1 If the algorithm fails to read the data, the system should generate error message for administrators' attention
- 4.1 if write error exists during storage, the system should generate error message for administrators' attention

Exceptions:

None Identified.

Use case #: 7

Purpose: compare baseline data to observed count.

The sequential events are as follows;

1. System checks for baseline sums
2. System checks for observed sums
3. System compute standard deviation values per post code area from baseline sum
4. System compares the observe counts to the sum of baseline and three times the standard deviation of the baseline

Alternatives:

- 1.1 if baseline sum is not valid, the system should generate error message for administrators' attention
- 2.1 if observed sum is not valid, the system should generate error message for administrators' attention
- 4.1 if error exists, the system should generate error message for administrators' attention

Exceptions:

None Identified.

Use case #: 8

Purpose: Make space and time triggered detections of outbreak based on detected BG elevations

The sequential events are as follows;

1. System checks the comparison results of baseline and observed computation
2. System execute the CUSUM algorithm if the baseline and observed values are valid

Alternatives:

- 1.1 If comparison results are not valid, the system should generate error message for administrators' attention
- 2.1 if observed sum is not valid, the system should generate error message for administrators' attention
- 2.2 if error exists, the CUSUM algorithm, the system should generate error message for administrators' attention

Exceptions:

None Identified.

Use case #: 9

Purpose: Indicate detection status

The sequential events are as follows;

1. If observed count is less than the baseline value and three times of its standard deviation, then green color is indicated on the map to show no possible outbreak
2. If observed count is equal to the baseline value and three times of its standard deviation, then yellow color is indicated on the map to show possible outbreak is near threshold
3. If observed count is greater than the baseline value and three times of its standard deviation, then red color is indicated on the map to show possible outbreak

Alternatives:

1.1 If the comparison results are not valid, the system should generate error message for administrators' attention

2.1 if observed sum is not valid, the system should generate error message for administrators' attention

2.2 if error exists, the CUSUM algorithm, should generate error message for administrators' attention

Exceptions:

None Identified.

Use case #: 10

Purpose: Store the clustering results

The sequential events are as follows;

1. System aggregate data and stores it with respective dates and postcodes

Alternative:

- 1.1 if write error exists during storage, the system should generate error message for administrators' attention

Exceptions:

None Identified.

4.7. Non-functional requirements

The following non-functional requirements were defined in accordance with Maarten et al. (Steen & Tanebaum, 2017).

5.7.1 Scalability

The ability of the system to conform with future expansion and integration with other functionalities. So, the EDMON-Cluster system shall have the ability to scale in line with additions of other geographical regions, diabetes subjects, reporting and visualization features.

5.7.2 Usability

EDMON-Clustering system will consist of efficient and easy usability features such as visualization of various dimensions and intuitive measures which will enable easy investigation and quick decision making.

5.7.3. Security and privacy

Security measures towards the confidentiality, integrity and availability of the subjects' data would be highly considered. Privacy preserving mechanisms recommended by GPDR and various security measures would be adopted to safe guide the privacy right of the subjects.

5.7.4. Ethics

Ethical consideration during actual implementation with personal data would be taken into consideration. Concerns of the subjects would be sorted prior to the usage of their data in the surveillance.

5.7.5 performance

Performance of the system such as power of detection, sensitivity, specificity, positive predictive values and timeliness would be considered

5.8. Summary

The chapter provided analysis of the functional requirement (as shown in Table 5.1 to 5.10) and non-functional requirements that should be met by the system and this was guided by various sources of requirement analysis (Robertson & Robertson, 2006; Steen & Tanebaum, 2017) (Ashwin, 2017; Tavolato & Vincena, 1984). Major sources of the requirements were obtained from literatures, discussion with diabetes experts and colleagues. These requirements served as input to the design and implementation.

Chapter Six: System Design

6.1 Introduction

The aim of this project was to develop a prototype of a cluster detection mechanism to be used in syndromic surveillance system like EDMON . The main input data was to be from a BG dynamic detection system (BGD-DS) (Woldaregay et al., 2018). BGD-DS was envisaged to be monitoring BG dynamics of Type-1 Diabetes persons and detecting changes which corresponded to their disease infection stages (Woldaregay et al., 2018). The output was conceptualized as to whether their BGs were high, low, or normal in relation to space and time in which the subjects were infected with diseases as proposed in EDMON framework (Woldaregay et al., 2018). Detections, which were in correspondence with infections, were to be clustered in both space and time. Aberrations were to be detected and alerted for disease outbreak detection. Due to limited availability of real data, synthetic data was simulated.

The entire project was partitioned into development and research components. The development component consists of implementing the necessary algorithms and visualizations and further converting the software into a mobile application. The research components consist of implementing the necessary algorithms and mathematical functions in the whole surveillance structure (clustering, aberration detection and output alerts) and assessing these methods. The entire research component and aspect of the development were executed, except the conversion of the system into mobile application aspect. Due to time constrain, the conversion to mobile app was hence deferred for future works and is not considered in this thesis work.

Essentially, the design, implementation and evaluation consideration of the mathematical model including clustering with KNN with Euclidian distance functions, aberration detection with CUSUM algorithm and Z-Score, visualization on maps, alerts and graphical representation of other dimensions have been presented in this thesis work.

The prototype of EDMON-Cluster can detect disease outbreak with synthetic data based on the past one-week baseline and the current week observation using a z-score. The alerts are to be indicated in the form of traffic lights indications such that red cluster is indicating outbreak detections, yellow cluster depicts near outbreak and green cluster suggests no outbreak possibility. These indications were exhibited on a map. The entire design has been depicted on the design frame framework and layout as shown in figure 20.0.

6.2 Framework and design considerations

Prior to the implementation of the clustering methods in EDMON (EDMON-Cluster), a systematic review was conducted (Yeng et al., 2018a), to explore for methods, algorithms, evaluation techniques, visualization methods and other dimensions. The results were intended to be used for implementing an efficient syndromic surveillance system. Various dimensions of the results were obtained in which this implementation primary depends on. In addition to relying on the results of

the systematic review, various reasons guided the development of the framework and the choice of the implementation methods as shown in figure 6.1. The layout of this framework consists of Input data, pre-processing of the input data, Clustering and Aberration detection, Visualization, Alarm and Alerts as shown in figure 6.1.

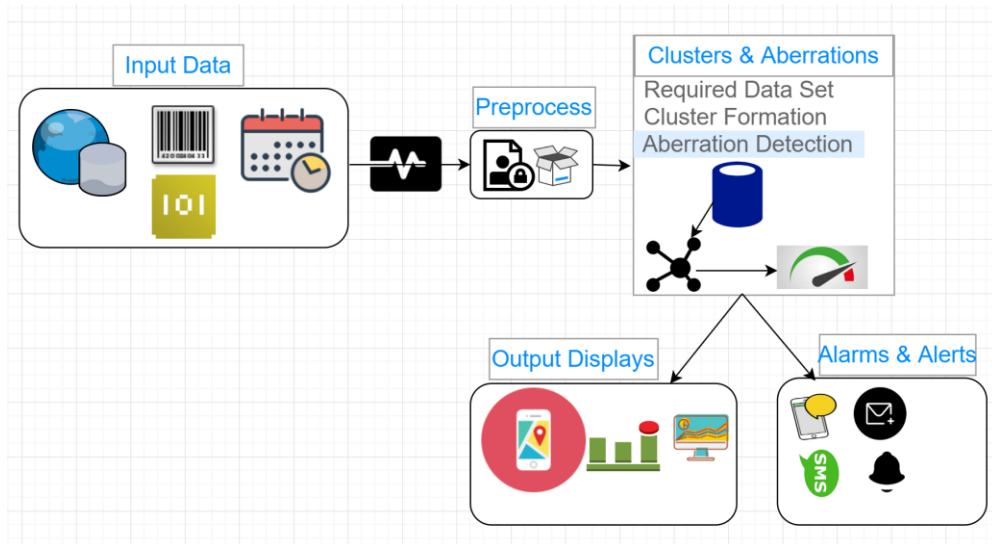


Figure 6. 1: Layout of Framework

6.2.1 Prototyping

Prototyping method was selected for this implementation due to the uncertain nature of EDMON requirements which essentially required iterative and collaborative approach between developers and users (Woldaregay et al., 2018). The systematic review results confirmed the prototyping method was most appropriate for use in this study (Yeng et al., 2018a).

6.2.2 Input data

Generally, syndromic surveillance systems with spatiotemporal methods require input data varying from structured to semi-structured data such as csv, xml or json format (Duangchaemkarn et al., 2017) as shown in figure 6.1. Ultimately, some key data input elements are highly required for the algorithm. These data elements include the data points with their associated geolocations, date and time of occurrences (Duangchaemkarn et al., 2017). The data points would also have unique non-personal identifications. Considering EDMON, the input data point would consist of the real-time infection status of an individual (micro level) in sequential binary format, where a binary value of 1 represent an infected individual, -1 represent suspicious individuals and a binary digit of 0 represent a normal individual. The data points would be associated with their

corresponding date, time and geolocation of occurrences. The data could be in a certain format such as csv or xml which can be accessed online.

6.2.3 Distance Measures

This section involves the location and clustering of the diabetes patients into various clusters based on their proximity to specified geographical points. Various methods were identified in the literature review including partitioning the region of interest into different small equal cells as shown in Figure 8 (Woldaregay et al., 2017) (Yang & Abraham O. Fapojuwo, 2015), calculating the Euclidean distance in KNN as shown in figure 12. In EDMON framework, distance measure was conceptualized to be used in clustering the occurrences of the elevated BGs (Woldaregay et al., 2017). The systematic review which was conducted for methods towards the implementation of the syndromic surveillance system also revealed various measures including KNN and K-means clustering (Yeng et al., 2018a).

6.2.4 Pre-processing

The preprocessing phase is to ensure that the input data is in the right format and sanity for the cluster and aberration detection phase to use. Therefore, the framework has provision for data conversion. For instance, online data in xml format can be converted to JSON format. Missing data would also be handled in various ways. In most instances, missing data has been excluded from the analysis (Nicholas Thapen, Donal Simmie, Chris Hankin, & Gillard, 2016). This and other methods would be used.

Another provision is to ensure privacy preserving mechanisms. This framework has a provision in the data preprocessing section to ensure that the input data is devoid of personal data. This would be done by following layout standards and regulations such as the General Data Protection Regulation (GDPR) established by the European Union (Bertino & Ferrari, 2018; GDPR, 2018). According to (GDPR:Report, 2017) the data is considered non-personal if pseudonymization and anonymization methods of privacy preserving mechanisms are used. Such techniques mitigate risk and assist the data processors in meeting their data compliance requirement. Pseudonymization replaces the most identifying fields within a data record with artificial identifiers, or pseudonyms but it does not replace all personal identifiable information from the data. It basically reduces the link-ability of a dataset with the original identity of an individual. Pseudonymization method uses techniques including encryption schemes. With anonymization, a variety of methods are available, and the choice will depend on the degree of risk and the intended use of the data. Some of the methods includes direct replacement, scramble, masking and blurring.

6.2.5 Cluster and Aberration Detection

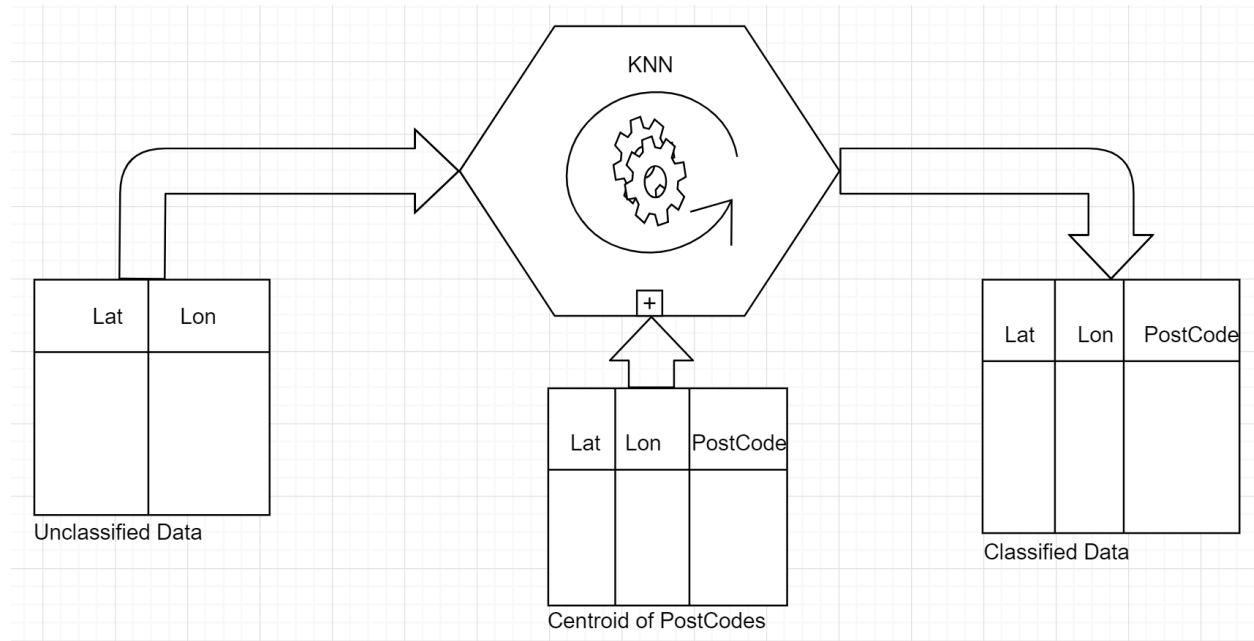


Figure 6. 2: Clustering mechanism

In EDMON framework, the outbreak detection was to be done in both space and time in order for users to know where and when the aberrations are occurring (Woldaregay et al., 2017). KNN and CUSUM algorithms were being explored in this study due to various reasons. The systematic review towards the implementation of cluster detection mechanism in EDMON revealed various algorithms that could be used to achieve the spatiotemporal objective of EDMON (Yeng et al., 2018a). Space Time Permutation Scan Statistics (STPSS) was found to be the most used algorithm for spatiotemporal measures since it does not require population at risk data to draw the expected baseline value (Kulldorff, 2005). This approach provides significant trend of baseline data while avoiding inclusion of historical data that is irrelevant to the current period. However, STPSS has a major drawback. It has low power of detection in geographically disaggregated data. But this gap of low power of detection can be filled by combining temporal methods and near neighbors' methods (Abellan J J, 2007; Isobel et al., 2016; Mathes et al., 2017; Wang et al., 2010; Yih et al., 2010). This measure could boost the power of detection in sparseness of data through local excess aberration detections in spatial and spatiotemporal methods (Abellan J J, 2007; Duangchaemkarn et al., 2017; Isobel et al., 2016; Mathes et al., 2017; Wang et al., 2010; Yih et al., 2010). Therefore, a combination of the state-of-the-art spatial algorithm (KNN) and temporal algorithm (CUSUM) were being considered for the implementation of the cluster detection mechanism in EDMON. Additionally, CUSUM happened to be the most used temporal algorithm and KNN was the most used near neighbour algorithm in the review (Yeng et al., 2018a).

6.2.7 Visualization, Alarm and Alerts

The main output of the framework includes timely alerts through alarms and visualizations of detected aberrations. From the studies, various visualization tools and output displays have been realized. Guided with the results and discussion sections of this study, ArcGIS, Leaflet-Open Source or Google Map tool can be used to implement the visualization module. This visual display would mainly be map with other displays such as time series and graph. The maps would indicate where and when clustering and aberrations occur. Also, alerts would be triggered through alarms and messaging. Leaflet map was chosen for the prototype due to it being open source, less expensive and does not require acquiring license to use (Leafletjs, 2019).

6.2.8 Summary of the chapter

This chapter focused on the system design and the design choices based on the systematic review (Yeng et al., 2018a). The design and implementation options were based upon the review and a framework which were developed in a capstone project in this study context. The various modules in the study has been specified and the reasons of their design options, methods and tools used were provided. The entire idea of the clustering aspect is as summarized in figure 6.1 and 6.2. The clustering unit receives unclassified data and uses KNN algorithm to determine the class of the unclassified data point using the combination of centroid and the classified data. The determined class of the unclassified dataset is posted onto the classified dataset with its associated new or determined class.

CHAPTER 7: IMPLEMENTATION AND RESULTS

This chapter presents both implementation and evaluation results of a cluster detection mechanism in EDMON, which was obtained through prototyping approach. After a systematic review on cluster detection mechanism, various methods, evaluation techniques and their related challenges among other dimensions were obtained. These details provided a guideline for the design of a framework and the implementation and evaluation of the prototype of the cluster detection mechanism. Synthetic data was simulated for this exercise. In addition, KNN and CUSUM algorithms were combined to form a spatiotemporal method for the study. The KNN was employed as a spatial algorithm for the classification of the detected infected diabetes individuals into their respective postal code areas. Euclidian distance was used in the KNN algorithm as a distance measure. Further, CUSUM algorithm was used as a temporal algorithm for the detection of the aberrations. Z-score or the number of standard deviations from the mean was used as the thresholding mechanism in the CUSUM method. The output was obtained in graphs, charts and alerts. Since the prototyping approach was adopted, various iterative development scenarios were implemented to arrive at the desired results as presented in the following sections. The source codes, simulated data and evaluation results are kept in a folder as described in the Appendix.

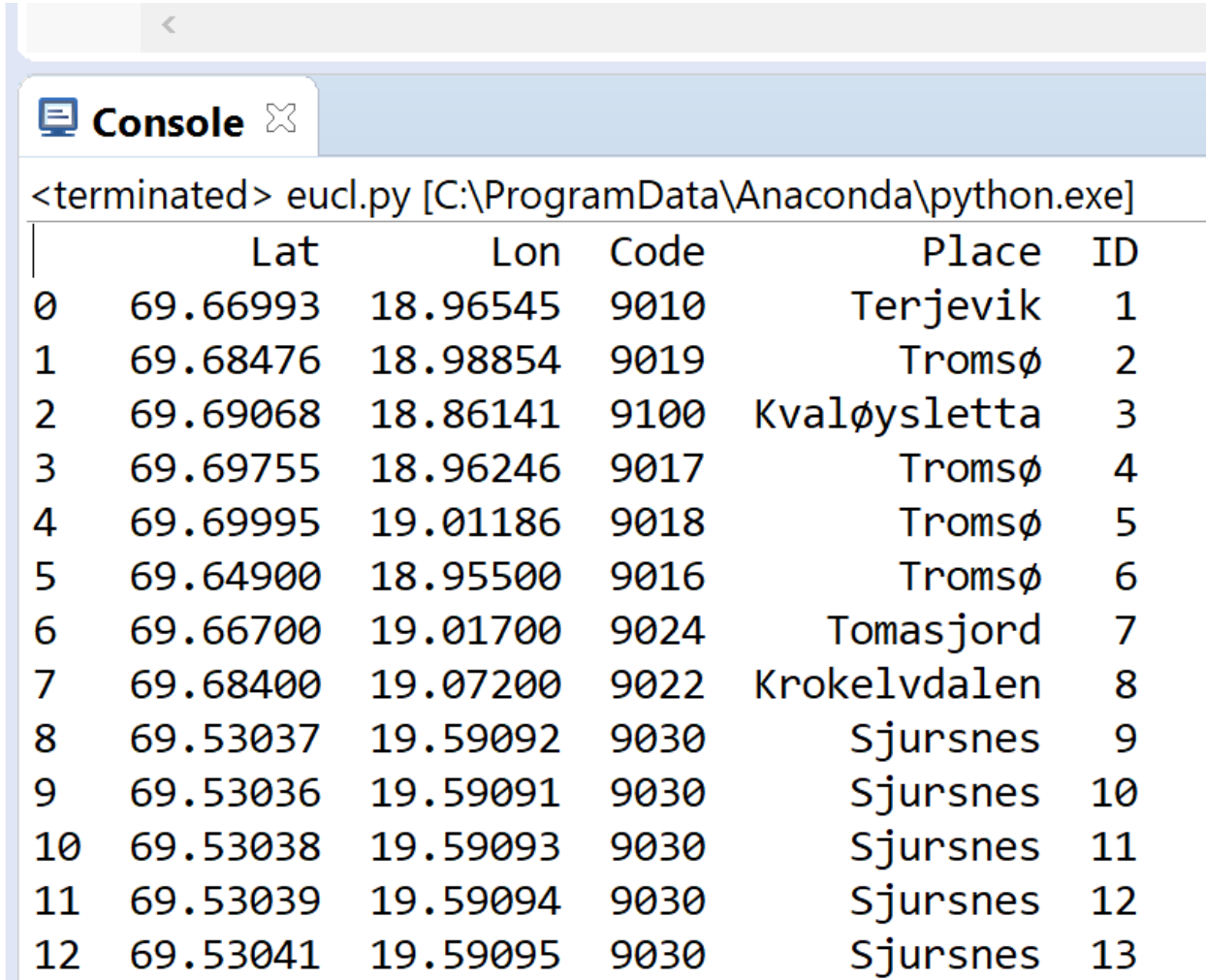
7.1 Synthetic data

This section indicates the results of the iteration of the development of the synthetic data which has evolved to meet the desire of the surveillance requirement.

7.1.1 Initial Synthetic data

Initially, the synthetic data was simulated and represented on a map as shown in Table 7.1 and figure 7.1. But the synthetic data lacked the infected individuals and time stamps of their occurrences.

Table 7. 1: Initial Simulated Data.



```
<terminated> eucl.py [C:\ProgramData\Anaconda\python.exe]
|      Lat      Lon  Code      Place  ID
0  69.66993  18.96545  9010   Terjevik  1
1  69.68476  18.98854  9019   Tromsø    2
2  69.69068  18.86141  9100  Kvaløysletta  3
3  69.69755  18.96246  9017   Tromsø    4
4  69.69995  19.01186  9018   Tromsø    5
5  69.64900  18.95500  9016   Tromsø    6
6  69.66700  19.01700  9024   Tomasjord  7
7  69.68400  19.07200  9022  Krokeldalen  8
8  69.53037  19.59092  9030   Sjursnes  9
9  69.53036  19.59091  9030   Sjursnes  10
10 69.53038  19.59093  9030   Sjursnes  11
11 69.53039  19.59094  9030   Sjursnes  12
12 69.53041  19.59095  9030   Sjursnes  13
```

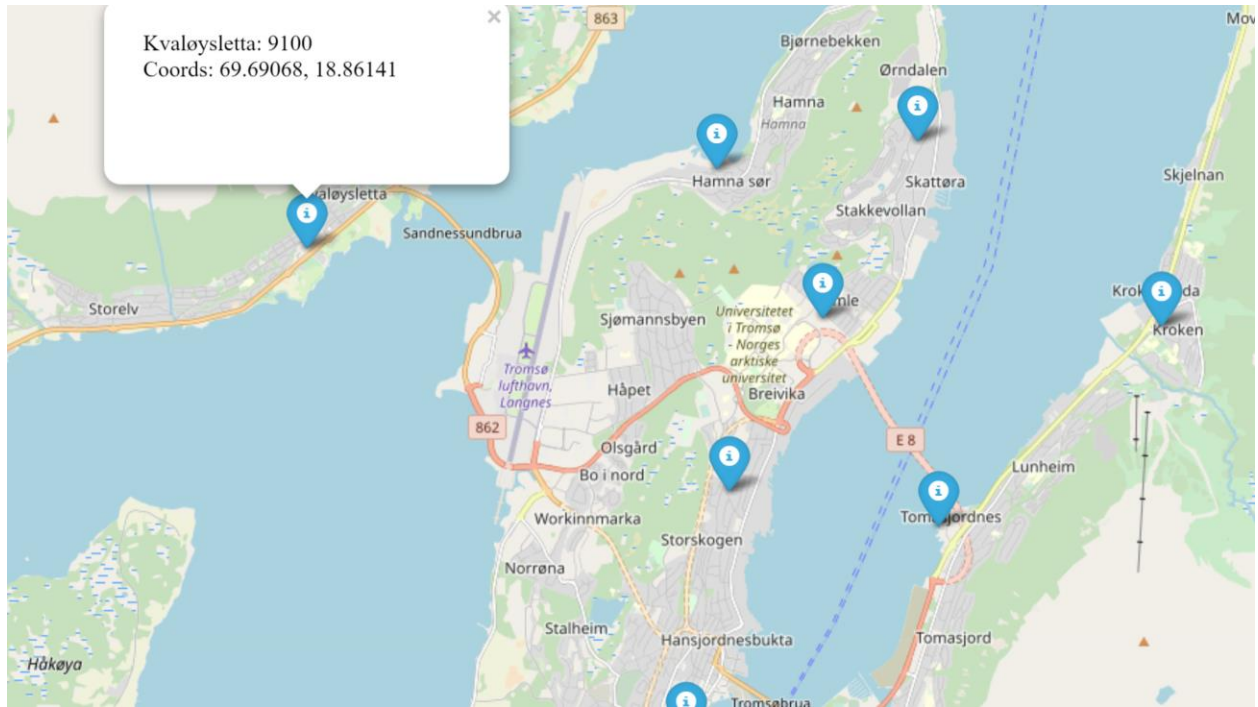


Figure 7. 1 Graphical representation of initial synthetic data

7.1.2 Synthetic data with infection status

The synthetic data was improved with status of detections of infected individuals and their time stamp of occurrences as shown in Table 7.2. The detections of the subjects have 1,0 and -1, representing infected, normal and suspicious state of infection status. These states were plotted on the map as shown in figure 7.2. However, the data representation on the graph was still undesirable since it was difficult to visualize the various clusters involve.

Table 7. 2: Synthetic data with detections (Classified dataset)

DID	Lat	Lon	Code	PID	Place	D_Date	Detections	Name
1	69.55799	19.33103	9027	1	Ramfjordbotn	20.01.2018 15:00	1	Alexander
2	69.57781	18.55499	9106	2	Straumsbukta	21.01.2018 18:00	1	Bjørn
3	69.62796	18.915	9006	3	Lanesvegen	22.01.2018 6:00	1	Andreas
4	69.63077	19.04736	9020	4	Tromsdalen	23.01.2018 15:00	1	Daniel
5	69.63702	17.981	9110	5	Sommarøy	24.01.2018 18:00	1	Frank
6	69.65124	18.54714	9272	9	Tromsøya	28.02.2018 6:00	0	Mat
7	69.65503	18.95499	9037	10	Tromsø	16.02.2018 15:00	0	Jan
10	69.66899	18.46545	9010	13	Terjevik	19.02.2018 15:00	-1	Jørgen
11	69.68976	18.38854	9019	14	Tromsø	20.02.2018 18:00	-1	Kenneth



Figure 7. 2: Data points of all detections

7.2 Clustering approach

A further improvement was explored by filtering the infected individuals and clustering them around the centroid. The number of detections for each postcode area was indicated as shown in Table 7.2 and figure 7.1. However, the representation did not distinguish between clusters of the various infection status such as normal, suspicious or infected. An improvement method was adopted by using three datasets known as centroid of postcode, unclassified and classified datasets as shown in Tables 7.3, 7.4 and 7.5. Unknown classes of infection status (as shown Table 7.4) were classified onto various simulated centroids of post code areas serving as the classes (as shown in Table 7.3) to form the classified infection status as shown in Table 7.5.

7.2.1 Centroid of post code

The simulated centroid of the post codes of the study area (Tromsø) consists of the geocoordinates of the centers of the various postcodes areas as shown in table 7.3.

Table 7. 3: Centroid of post codes

Lat	Lon	Code	Centroid_ID	Place
69.55799	19.33103	9027	1	Ramfjordbotn
69.57781	18.55499	9106	2	Straumsbukta
69.62796	18.915	9006	3	Lanesvegen
69.63077	19.04736	9020	4	Tromsdalen
69.63702	17.981	9110	5	Sommarøy
69.64057	18.92729	9007	6	Kveldrovegen
69.64225	18.90889	9013	7	Tromsø
69.65079	18.95493	9008	8	Tromsø
69.25102	18.54714	9272	9	Tromsøya
69.61103	18.35579	9037	10	Tromsø
69.95494	17.95376	9009	11	Tromsø
69.66153	18.94791	9011	12	Tromsø
69.66993	18.96545	9010	13	Terjevik
69.68476	18.98854	9019	14	Tromsø
69.69068	18.86141	9100	15	Kvaløysletta
69.69755	18.96246	9017	16	Tromsø
69.69995	19.01186	9018	17	Tromsø
69.70019	18.10055	9016	18	Tromsø
69.667	19.017	9024	19	Tomasjord
69.684	19.072	9022	20	Krokeldalen
69.53037	19.59092	9030	21	Sjursnes

7.2.2 Unclassified data

The simulated, unclassified data are detections of the infection status of the diabetes persons with their respective geocoordinates of their occurrences as shown in Table 7.4.

Table 7. 4: Unclassified data

DID	Lat	Lon	PID	D_Date	Detections	Name
1	69.55799	19.33103	1	20.01.2018 15:00	1	Alexander
2	69.57781	18.55499	2	21.01.2018 18:00	1	Bjørn
3	69.627957	18.915001	3	22.01.2018 6:00	1	Andreas
4	69.63077	19.04736	4	23.01.2018 15:00	1	Daniel
5	69.63702	17.981	5	24.01.2018 18:00	1	Frank
6	69.640574	18.927288	6	25.01.2018 6:00	1	Erling
7	69.64225	18.90889	7	26.01.2018 15:00	1	Geir
8	69.65079	18.95493	8	27.01.2018 18:00	1	Harald
9	69.651024	18.954714	9	28.01.2018 6:00	1	Mat

Table 7. 5: Cluster of number of infected individuals around centroid

Counts	1	2	2	1	1	1	2	1	1	1	1	2	1	1	1	5	1	2	1	2	1
Code	9006	9007	9008	9009	9010	9011	9013	9016	9017	9018	9019	9020	9022	9024	9027	9030	9037	9100	9106	9110	9272
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

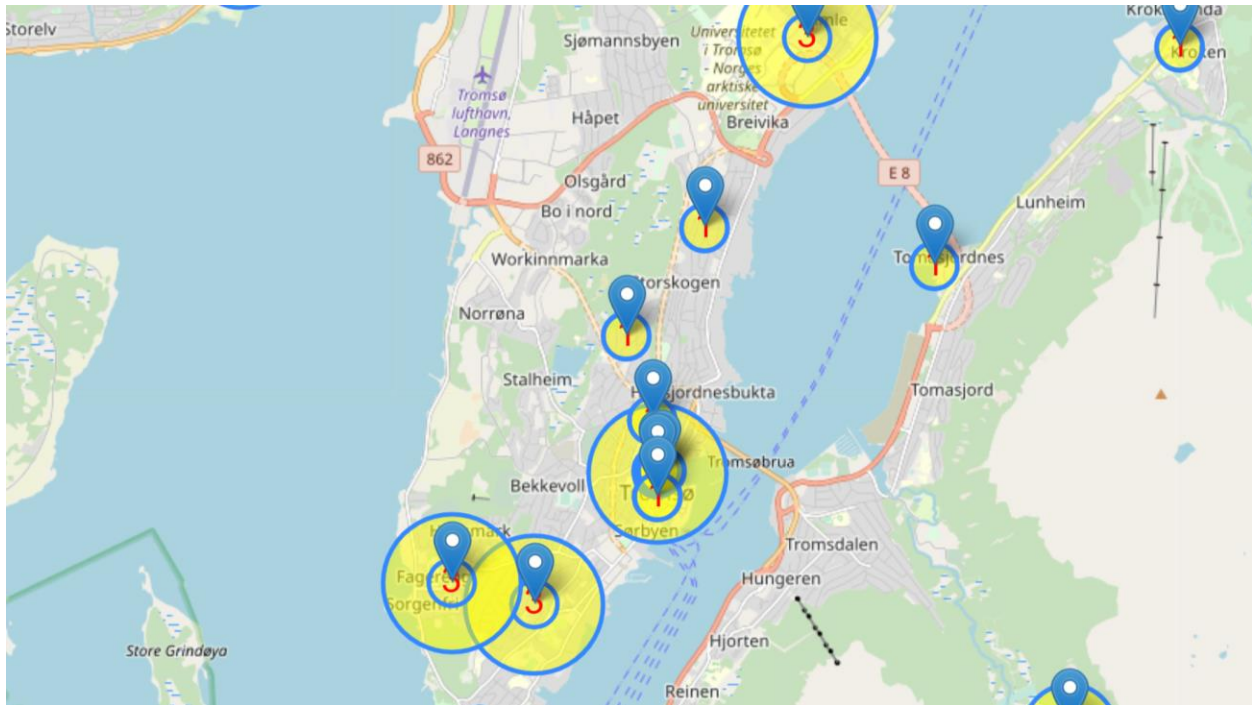


Figure 7. 3: Clustering around centroid

7.2 Classification

This section consists of the implementation results obtained from the clustering mechanism. Three modules were involved in the clustering of the infected individuals around the centroids of each postcode area. These include the K of KNN, computation of the Euclidian distance and the determination of the class based on the K factor.

7.2.1 The K factor

The K factor in the KNN was determined by calculating the odd integer value of the square root of the total number of the classified data as shown in Figures 7.4 and 7.5.

```
#create a function for K
def k(datasize):
    myk=math.sqrt(datasize)
    if round(myk,0) % 2==0:
        myk=myk-1
    return round(myk,0)
print("The size of classified infected individuals:")
print(len(read_classified_data))
print("The value for K: ")
print(k(len(read_classified_data)))
```

Figure 7. 4: Determination of K in KNN

The Size of Classified Infected Persons' Data:

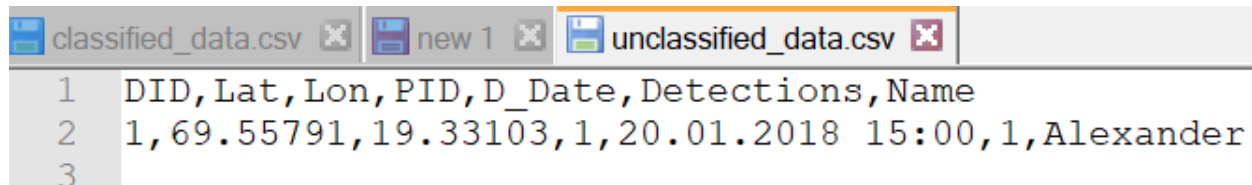
```
270
The value for K:
15.0
```

Figure 7. 5: Output of Classified data size and K in KNN

7.2.2 Euclidean distance

The Euclidean distance of each unclassified infected individuals (as shown in Table 7.6 and Figure 7.6) was computed by using their respective geolocation coordinates (Lat, Lon), to each of the centroid coordinates of the infected classified persons' location points as shown in Figure 7.6 and 7.7. The Euclidean distances were sorted in ascending order as shown in Figure 7.8, 7.9 and 7.10.

Table 7. 6: Unclassified data point



	DID	Lat	Lon	PID	D_Date	Detections	Name
1	1	69.55791	19.33103	1	20.01.2018	15:00	1,Alexander
2							
3							

Table 7.6 consists a sample feature of unclassified dataset with its associated attributes. Figure 7.6 is a sample piece of code of a function which was used to calculate the Euclidean distances.

```
# Defining a function which calculates euclidean distance between two data points
length = mytestset.shape[1]
number_of_unclassified_records=mytestset.shape[0]
distances = (Groeneveld et al., 2017)

def euclideanDistance(data1, data2, length):
    distance = 0
    for x in range(length):
        distance += np.square(data1[x] - data2[x])
    return np.sqrt(distance)
##### Start of STEP 3
```

```

# Calculating Euclidean distance between each row of training data and test data

read_classified_data = read_classified_data[['Lat', 'Lon', 'Code', 'DID', 'PID', 'Place',
'D_Date', 'Detections']]
print('read_classified_data-----')
print(read_classified_data)
for i in range(len(read_classified_data)):
    dist = euclideanDistance(test, read_classified_data.iloc[i], length)

    distances[i] = dist[0]

```

Figure 7. 6: Determining the Euclidean Distance

Figure 7.7 is a portion of the output of the Euclidian distances which were computed by the function in Figure 7.6.

```

distances:
{0: 0.14319714592128407, 1: 0.1350026762697738, 2: 0.1350026762697738, 3: 0.14319714592128407, 4: 8.000000001118224e-05, 5
: 0.0, 6: 8.000000001118224e-05, 7: 8.000000001118224e-05, 8: 0.7762951060002875, 9: 0.7762951060002875, 10: 0.77629716088
62147, 11: 0.7762951060002875, 12: 0.4218847129844821, 13: 0.4218847129844821, 14: 0.4207791024397459, 15: 0.4218847129844

```

Figure 7. 7: Sample values of computed Euclidean distances

7.2.3 Determination of KNN

The distances of each of the unclassified infected individuals (eg as shown in Figure 7.7) were sorted in ascending order and the first K number of the shorter distances were obtained as shown in figure 7.8, 7.9 and 7.10. Since at an instance, the K was determined to be 15 as shown in figure 7.5, the first 15 shorter distances were obtained as shown in Figure 7.9 and 7.10.

```

#perform knn
def knn(trainSet, k):

    ##### Start of STEP 3.2
    # Sorting them on the basis of distance
    sort_distances =sorted(distances.items(), key=operator.itemgetter(1))
    print('sort_distances-----')
    print(sort_distances)
    ##### End of STEP 3.2

    neighbors = []

    ##### Start of STEP 3.3

```

```

# Extracting top k neighbors
for x in range(k):
    neighbors.append(sort_distances[x][0])
print('K neighbors-----')
print(neighbors)

##### End of STEP 3.3
classVotes = {}

##### Start of STEP 3.4
# Calculating the most freq class (post code or zipcode) in the neighbors
for x in range(len(neighbors)):
    response = trainSet.iloc[neighbors[x]][2]
    #row index=neighbors[x]
    #column index in the data set =[2] starting from the left
    print(response)
    if response in classVotes:
        classVotes[response] += 1
        print('classVotes[response]-COUNT OF PROXIMITY TO CLASSES.....')
        print(classVotes[response])
    else:
        classVotes[response] = 1
        print(classVotes[response])

##### End of STEP 3.4

##### Start of STEP 3.5
sortedVotes = sorted(classVotes.items(), key=operator.itemgetter(1), reverse=True)
print("sorted Votes ")
print(sortedVotes)
return(sortedVotes[0][0], neighbors)
##### End of STEP 3.5

# Running KNN model
result,m= knn(read_classified_data,int(the_k))
print(result,m)

```

Figure 7. 8: KNN implementation

Figure 7.9 showed the sorted distances in ascending order. The ID and the Euclidean distance (ID, Euclidean distance) of the subjects involves are shown in Figure 7.9

```

sort_distances-----
[(5, 0.0), (4, 8.000000001118224e-05), (6, 8.000000001118224e-05), (7, 8.000000001118224e-05), (1, 0.135002676269773
8), (2, 0.1350026762697738), (0, 0.14319714592128407), (3, 0.14319714592128407), (87, 0.1613973608830074), (96, 0.16
13973608830074), (84, 0.1620777088312925), (85, 0.1620777088312925), (89, 0.17126619076747246), (93, 0.1712661907674

```

Figure 7. 9: Sorted K NNN of infected individuals data points

Figure 7.10 showed the IDs of the Euclidean distances in ascending order of the Euclidean distances.

```

K neighbors-----
[5, 4, 6, 7, 1, 2, 0, 3, 87, 96, 84, 85, 89, 93, 177]

```

Figure 7. 10: Sorted K NNN of infected individuals data points

7.2.4 Voting and counting of data points to the nearness of their various postcode areas.

After the selection of the K number of data points which were closer to various classified data points, the K data points further ‘voted’ or were categorized and tagged to various postcodes or classes based on their proximity to the centroid coordinates of the simulated postcodes as shown in figure 7.11. The final counts of votes or tagged K number of data point distances to each postcode area were declared and the post code with the higher number of K data points was also declared as shown in figure 7.12 and 7.13. In demonstrating with the synthetic data (figure 7.14), 40% of the 15-total number of K were closer to the postcode, 9030 as shown in figure 7.12 and 7.13.

```
K neighbors-----
[5, 4, 6, 7, 1, 2, 0, 3, 87, 96, 84, 85, 89, 93, 177]

9106
1

9027
1

9027

classVotes[response]-COUNT OF PROXIMITY TO CLASSES.....
2

9027

classVotes[response]-COUNT OF PROXIMITY TO CLASSES.....
```

Figure 7. 11: voting and counting of infected individuals

Figure 7.11 showed the counting process of how many of the K of 15 were closer to each postcode. Figure 7.12 displayed the final counted distances which were closer to each of the post codes. Figure 7.12 showed 9030 post code to have the highest count of 6 of the Euclidian distances which were closer to it.

```
sorted Votes:

[(9030, 6), (9019, 4), (9027, 3), (9106, 1), (9018, 1)]
9030
```

Figure 7. 12: Voting results of infected individuals' proximity to postcode areas.

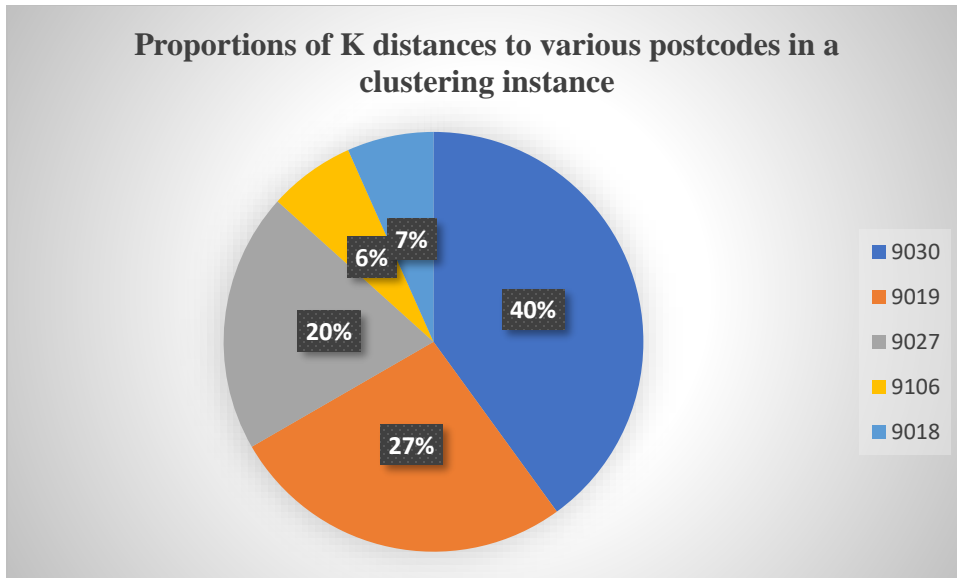


Figure 7. 13: Pie chart indicating percentages of nearness of data point

7.2.5 Saving Clustering Results

The geocodes which were associated to the selected postcodes in the KNN clustering and related attributes were then written to the classified data file as shown in Figure 7.14

```
#writing clustered data point with the selected post code to classified data file
'Lat', 'Lon', 'Code', 'DID', 'PID', 'Place', 'D_Date'
voted_pc=result
DID=DD+z
la=read_unclassified_data.iloc[z][0]
lo=read_unclassified_data.iloc[z][1]

personalID=read_unclassified_data.iloc[z][3]
PID=hashlib.md5(str(personalID).encode()).hexdigest()
place=""
D_Date=read_unclassified_data.iloc[z][4]
Detections=read_unclassified_data.iloc[z][5]
Name=str(None)
htm='\n'

#Built test code to check for duplications
check_duplication=[la,lo,PID,D_Date]
#combine data item in a list
classified_data_list=[DID,la,lo,voted_pc,PID,place,D_Date,Detections,Name +'\n']

for tc in check_duplication:
```

```

with open(read_classified_file,'r+') as file:
    content=file.read()
    if str(tc) in content:
        pass
        print('pass')
    else:
        file.write(','.join(str(x) for x in classified_data_list))
        DID=len(read_classified_data)+1
        print(classified_data_list)

```

Figure 7. 14: Posting of clustering Results

7.3 Aberration Detection

The aberration detection section determined excess of the observational data in comparison with the established baseline. The excess measure is three times the standard deviation of the baseline data, known as the Z-score which was determined in the literature (Yeng et al., 2018a).

7.3.1 Baseline Data

A baseline data was formed with elevated classified data by grouping 7 days of the previous week's data for each post code area as shown in figure 7.15 and 7.16.

```

#Identify those BG elevations that were detected in the most recent 14 days of the dataset to
form the baseline //(check date.unique)
historic_data = read_classified_data.D_Date.isin(read_classified_data.D_Date.unique()[-14:])
historic_data = read_classified_data[is_elevated_BGs &
historic_data].drop_duplicates('DID').reset_index(drop = True)
print('historic_data')
print(historic_data)

base_line=historic_data.D_Date.isin(historic_data.D_Date.unique()[:7])
base_line=historic_data[base_line].drop_duplicates('DID').reset_index(drop = True)

```

Figure 7. 15: Baseline data

```
merge_data_baseline_count_Grouped-----
```

	Lat	Lon	Code	Centroid_ID	Place	Counts
0	69.557990	19.331030	9027	1	Ramfjordbotn	3
1	69.627957	18.915001	9006	3	Lanesvegen	1
2	69.650790	18.954930	9008	8	Tromsø	5
3	69.699950	19.011860	9018	17	Tromsø	1
4	69.684000	19.072000	9022	20	Krokelvdalen	1

Figure 7. 16: Baseline data merged with postcodes (code)

7.3.2 Observations

The observed data was also formed with classified infected individuals in the most recent 7 days or the current week as shown in figure 7.17 and 7.18

```
#DETERMINE OBSERVED DATA
#Identify those BG elevations that were detected in the most recent 7 days of the dataset to form
the baseline // (check date.unique)
most_resent = read_classified_data.D_Date.isin(read_classified_data.D_Date.unique()[-7:])
observed_counts = read_classified_data[is_elevated_BGs &
most_resent].drop_duplicates('DID').reset_index(drop = True)
print('observed count.....')
print(observed_counts)
#Count the number of detections in each post code area in the Observed count using groupby
method
print('observed_count_Grouped .....')
observed_count_Grouped=observed_counts.groupby('Code').size().reset_index(name='Counts')

#Count the number of detections in each post code area by date in the observed count using
groupby method
observed_count_Grouped=observed_counts.groupby(['Code','D_Date']).size().reset_index(name
='Counts')
print('observed Count_Grouped.....')
print(observed_count_Grouped)
#merge centroid data to observed count on post code
merge_data_observed_count_Grouped=pd.merge(centroid_data, observed_count_Grouped,
on='Code')
print('merge_data_observed_count_Grouped-----')
print(merge_data_observed_count_Grouped)
```


Figure 7. 17: Observed data

```
merge_data_observed_count_Grouped-----
```

	Lat	Lon	Code	Centroid_ID	Place	Counts
0	69.557990	19.331030	9027	1	Ramfjordbotn	2
1	69.627957	18.915001	9006	3	Lanesvegen	2
2	69.630770	19.047360	9020	4	Tromsdalen	1
3	69.650790	18.954930	9008	8	Tromsø	3

Figure 7. 18: Observed data merged with post codes centroids

7.3.3 Standard deviation and mean

The standard deviation and mean were then computed for each postcode data in the baseline dataset as shown in figure 7.19.

```
print basline for SD.....
   Code      D_Date  Counts
0  9008 2018-07-02 15:00:00    1
1  9008 2018-08-02 18:00:00    1
2  9008 2018-11-20 15:00:00    3
type.....
<class 'pandas.core.series.Series'>
SD.....
0.9428090415820634
M.....
1.6666666666666667
```

Figure 7. 19: Determination of standard deviation and mean

7.3.3 Cumulative sum (CUSUM) dataset

A dataset was formed out of the data gathered in the previous sections such as the baseline, observational, standard deviation and mean, for the computation of the CUSUM and the determination of the aberrations as shown in Figure 7.20 and 7.21

```
print ('data for
cumsum!!!!!!!!!!!![i,post_code,m,standDev,observed_data_mergedWith_baseline.iloc[i][-
2],observed_data_mergedWith_baseline.iloc[i][-1]])
data_for_CUSUM=[i,post_code,m,standDev,observed_data_mergedWith_baseline.iloc[i][-
2],observed_data_mergedWith_baseline.iloc[i][-1]]
```

Figure 7. 20: Data for CUSUM

```
data for cumsum!!!!!!!!!!!![i,post_code,m,standDev,observed_data_mergedWith_baseline.iloc[i][-2],observed_data_mergedWith_b
aseline.iloc[i][-1]]
[3, 9008, 1.6666666666666667, 0.9428090415820634, 3, 5]
0.9428090415820634
observed count
3
baseline
5
```

Figure 7. 21: Output data gathered for CUSUM

7.3.4 CUSUM Aberration Detection

Having obtained the necessary values for all the variables in the CUSUM calculation, a function was therefore developed as such, as shown in Figure 7.22. From figure 7.22 if the observed count value was less than three times of the standard deviation plus the baseline cumulative value within the postcode area in the defined time frames, the cluster shows green to indicate no detections of aberrations.

```
#function get color for aberrations
def Color_for_aberration_detection():
    if observed_count < 3*(standDev)+(baseline_count):
        mycolor='green'
    elif observed_count > 3*(standDev)+(baseline_count):
        mycolor='red'
    elif observed_count == 3*(standDev)+(baseline_count):
        mycolor='yellow'

    return mycolor
```

Figure 7. 22: Aberration detection function

However, if the observed count value was more than three times of the standard deviation plus the baseline cumulative value within the postcode area in the defined time frames (Watkins et al., 2008), the cluster shows red to indicate detections of aberrations or outbreak of infected individuals as shown in figure 7.23 to 7.26. A cluster shows yellow if the observed count value was equal to three times of the standard deviation plus the baseline cumulative value within the postcode area in the defined time frames (Cochran, 1977; Groeneveld et al., 2017).

7.4 Surveillance data presentation

In the surveillance data presentation, the maps aspect was transformed from figure 7.1, 7.2, 7.3 to 7.23 to 25 and finally to 7.43 where the cluster point and its background showed red to indicate possible outbreak as the status of the cluster.

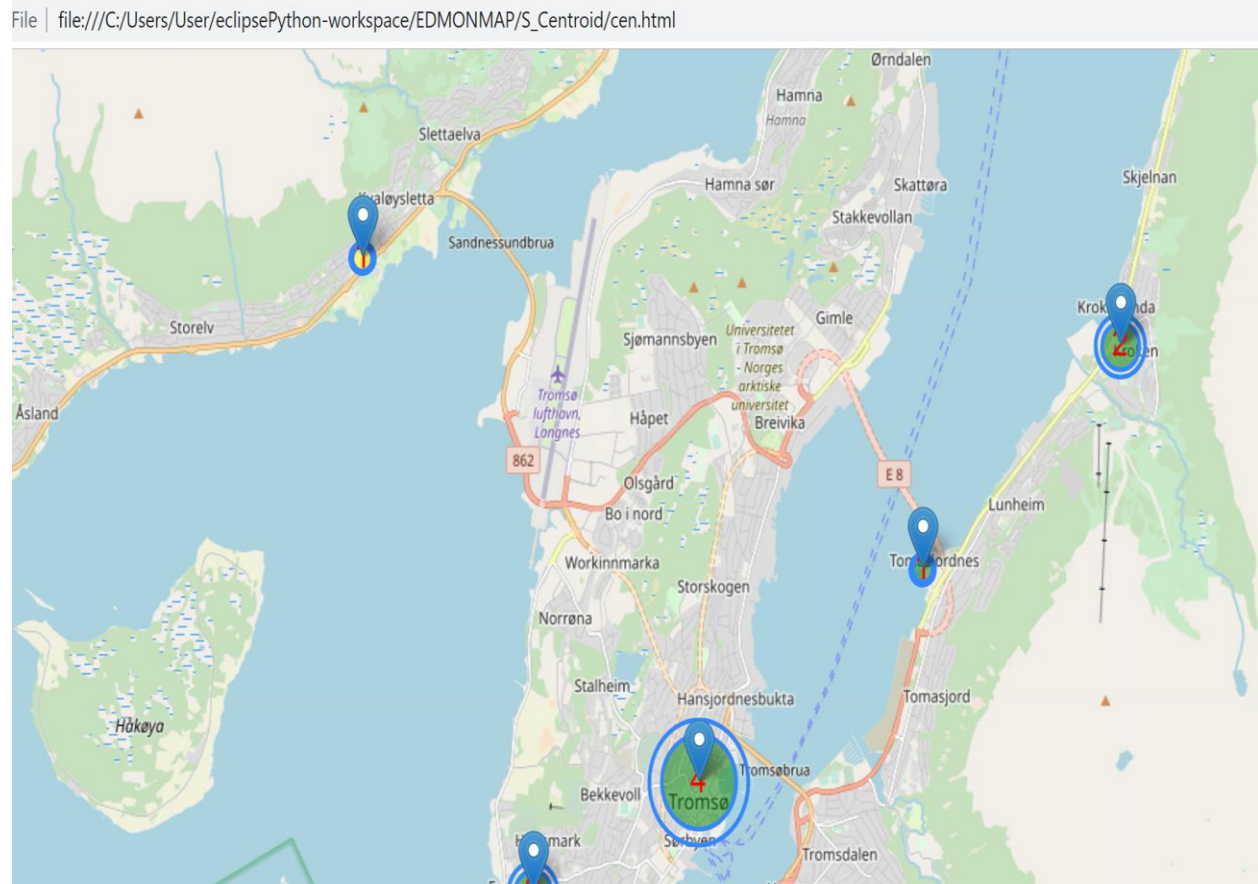


Figure 7. 23: Sample map presentation

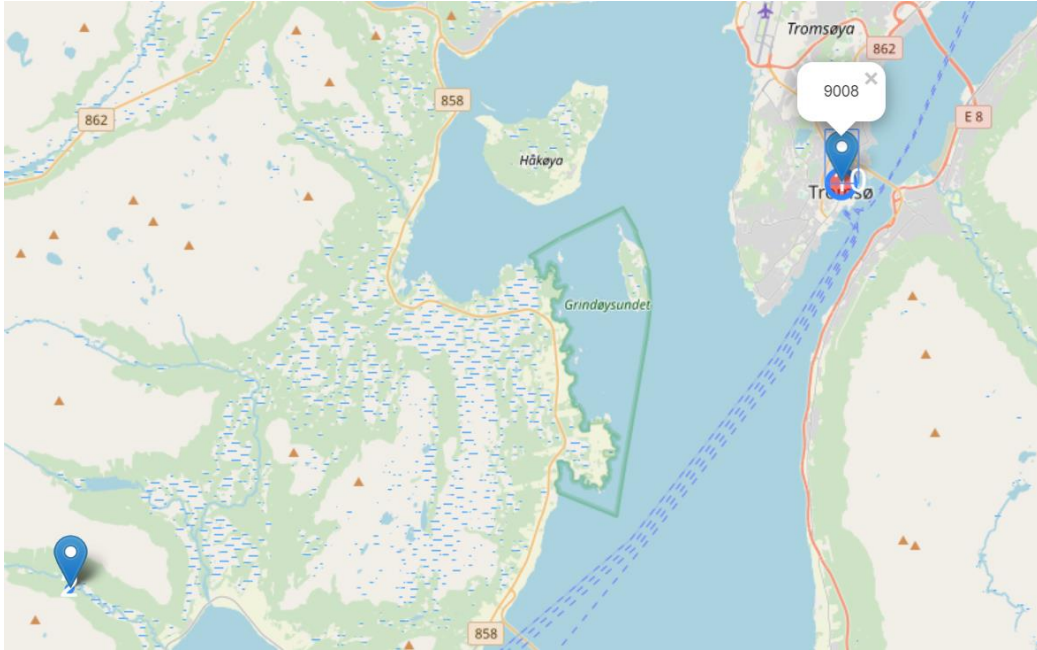


Figure 7. 24: Improvement of Clusters on map

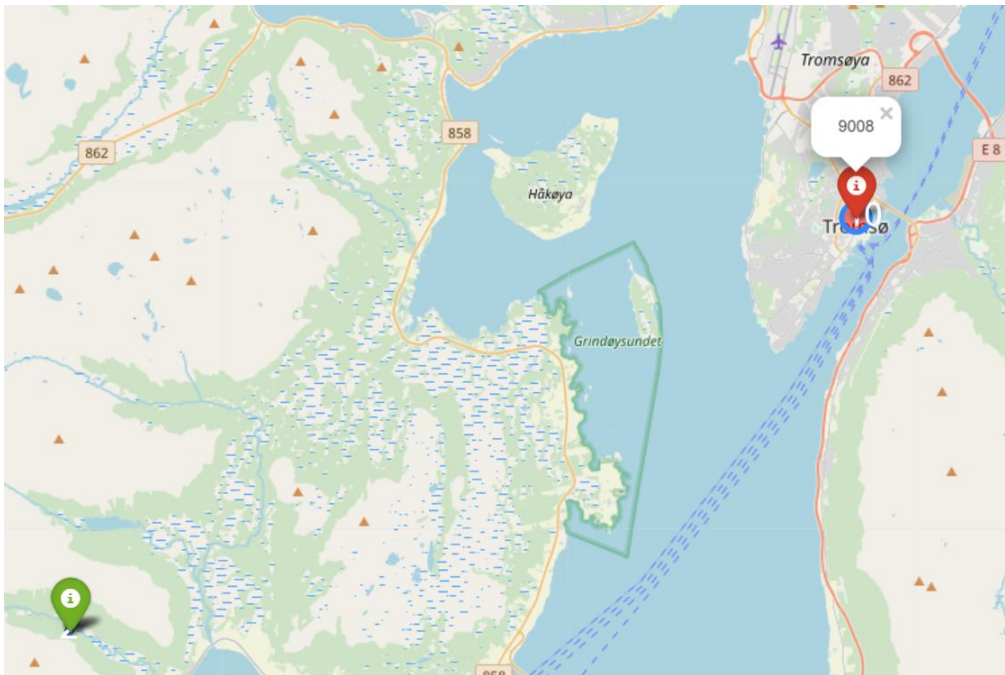


Figure 7. 25: Improvement of Clusters on map

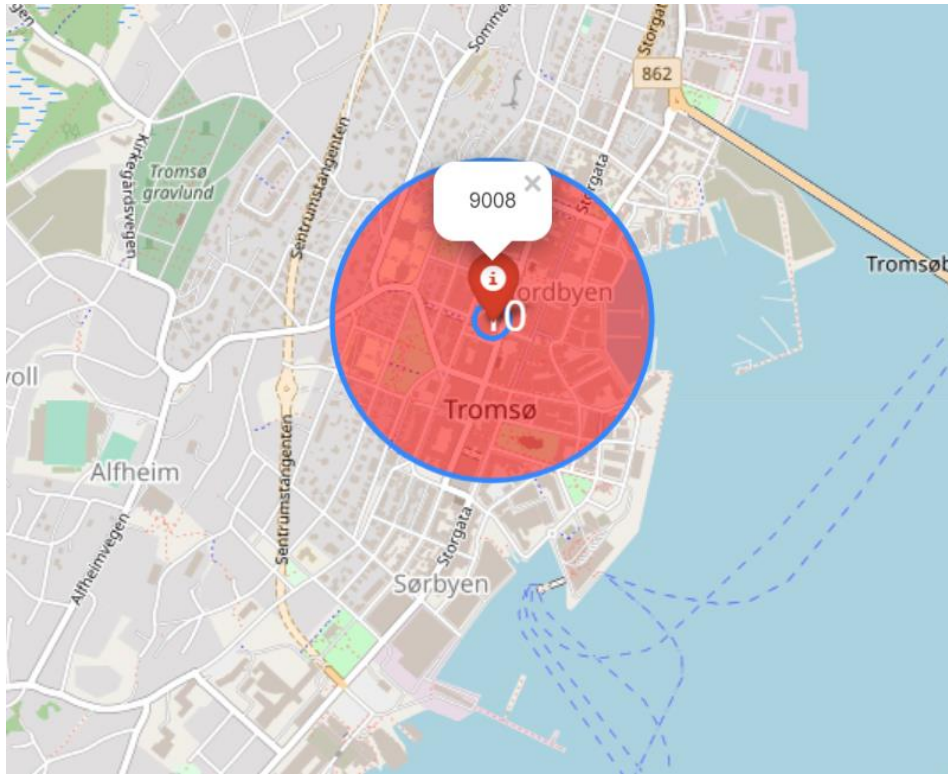


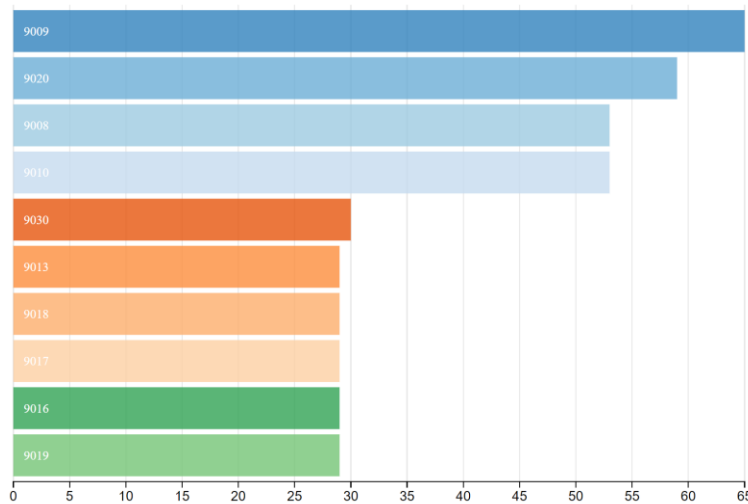
Figure 7. 26: Single cluster view

7.4.2 Graphs

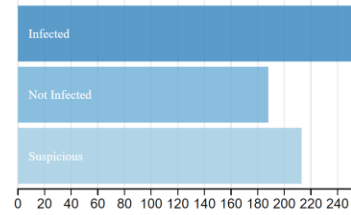
The detected status of individuals as normal, suspicious and infected were also presented in dynamic graphs as shown in figure 7.27. A selection of the status of the individual detection (Infected, Normal or Suspicious) on the pie chart indicates the quantities on the post code areas as shown in figure 7.27.

Disease Infection Reports:

By Post Code Area(Top 10):



Infection Rates, Bar Chart:



Infection Rates, Pie Chart:

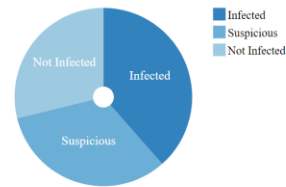


Figure 7. 27: Dynamic graph visualization of Infections

7.5 Security and privacy implementation

A one-way hashing with MD5 and nulling of anonymization methods were implemented as a security and privacy measures on personal attributes of personal identifiers such as personal IDs and names as shown in figure 7.28 and 7.29 respectively.

```

655 652,69.557909999999999,19.33103,9106,c4ca4238a0b923820dcc509a6f75849b,,2018-01-20 15:00:00,1,Alexander
656 653,69.57789,18.55499,9008,c4ca4238a0b923820dcc509a6f75849b,,2018-01-21 18:00:00,1,Alexander
657 654,69.620956999999999,18.915001,9006,eccbc87e4b5ce2fe28308fd9f2a7baf3,,2018-01-22 06:00:00,1,Andreas
658 655,69.637769999999999,19.04736,9008,a87ff679a2f3e71d9181a67b7542122c,,2018-01-23 15:00:00,1,Daniel
659 656,69.63002,17.980999999999998,9008,e4da3b7fbbce2345d7772b0674a318d5,,2018-01-24 18:00:00,1,Frank
660 657,69.641574,18.927288,9006,1679091c5a880faf6fb5e6087eb1b2dc,,2018-01-25 06:00:00,1,Erling
661 658,69.64025,18.90889,9006,8f14e45fceeal67a5a36dedd4bea2543,,2018-01-26 15:00:00,1,Geir
662 659,69.65579,18.954929999999997,9006,c9f0f895fb98ab9159f51fd0297e236d,,2018-01-27 18:00:00,1,Harald
663 660,69.656024,18.954714000000003,9006,45c48cce2e2d7fbdeaf5c51c7c6ad26,,2018-01-28 06:00:00,1,Mat
664 661,69.65603,18.955749899999997,9006,83d9446802a44259755d38e6d163e820,,2018-01-29 15:00:00,1,Jan
665 662,69.65094,18.95376,9006,83d9446802a44259755d38e6d163e820,,2018-01-30 18:00:00,1,Jan
666 663,69.666530000000001,18.94791,9006,c20ad4d76fe97759aa27a0c99bfff6710,,2018-01-31 06:00:00,1,Johan
667 664,69.66799,18.96545,9006,c51ce410c124a10e0db5e4b97fc2af39,,2018-02-01 15:00:00,1,Jorgen
668 665,69.680759999999999,18.98854,9006,aab3238922bcc25a6f606eb525ffdc56,,2018-02-02 18:00:00,1,Kenneth
669 666,69.69768,18.86141,9006,9bf31c7ff062936a96d3c8bd1f8f2ff3,,2018-02-04 06:00:00,1,Marius
670 667,69.69055,18.96246,9006,c74d97b01eae257e44aa9d5bade97baf,,2018-04-02 15:00:00,1,Thomas
671 668,69.69095,19.011860000000002,9006,70efd2ec9b086079795c442636b55fb,,2018-05-02 18:00:00,1,Nils
672 669,69.64019,18.955,9006,6f4922f45568161a8cdf4ad2299f6d23,,2018-06-02 06:00:00,1,Ola
673 670,69.660069999999999,19.017,9008,1f0e3dad99908345f7439f8ffabdfc4,,2018-07-02 15:00:00,1,Vegard
674 671,69.68004,19.072,9008,98f13708210194c475687be6106a3b84,,2018-08-02 18:00:00,1,Robert

```

Figure 7. 28: One-way hashing of Person IDS

```

1 652,69.55790999999999,19.33103,9106,c4ca4238a0b923820dcc509a6f75849b,,2018-01-20 15:00:00,1, None
2 653,69.57789,18.55499,9008,c4ca4238a0b923820dcc509a6f75849b,,2018-01-21 18:00:00,1, None
3 654,69.62095699999999,18.915001,9006,eccbc87e4b5ce2fe28308fd9f2a7baf3,,2018-01-22 06:00:00,1, None
4 655,69.63776999999999,19.04736,9008,a87ff679a2f3e71d9181a67b7542122c,,2018-01-23 15:00:00,1, None
5 656,69.63002,17.98099999999999,9008,e4da3b7fbbce2345d7772b0674a318d5,,2018-01-24 18:00:00,1, None
6 657,69.641574,18.927288,9006,1679091c5a880faf6fb5e6087eb1b2dc,,2018-01-25 06:00:00,1, None
7 658,69.64025,18.90889,9006,8f14e45fceeal67a5a36dedd4bea2543,,2018-01-26 15:00:00,1, None
8 659,69.65579,18.95492999999999,9006,c9f0f895fb98ab9159f51fd0297e236d,,2018-01-27 18:00:00,1, None
9 660,69.656024,18.954714000000003,9006,45c48cce2e2d7fbdea1afc51c7c6ad26,,2018-01-28 06:00:00,1, None
10 661,69.65603,18.95574989999999,9006,d3d9446802a44259755d38e6d163e820,,2018-01-29 15:00:00,1, None
11 662,69.65094,18.95376,9006,d3d9446802a44259755d38e6d163e820,,2018-01-30 18:00:00,1, None
12 663,69.66653000000001,18.94791,9006,c20ad4d76fe97759aa27a0c99bfff6710,,2018-01-31 06:00:00,1, None
13 664,69.66799,18.96545,9006,c51ce410c124a10e0db5e4b97fc2af39,,2018-02-01 15:00:00,1, None
14 665,69.68075999999999,18.98854,9006,aab3238922bcc25a6f606eb525ffdc56,,2018-02-02 18:00:00,1, None
15 666,69.69768,18.86141,9006,9bf31c7ff062936a96d3c8bd1f8f2ff3,,2018-02-04 06:00:00,1, None
16 667,69.69055,18.96246,9006,c74d97b01eae257e44aa9d5bade97baf,,2018-04-02 15:00:00,1, None
17 668,69.69095,19.011860000000002,9006,70efdf2ec9b086079795c442636b55fb,,2018-05-02 18:00:00,1, None
18 669,69.64019,18.955,9006,6f4922f45568161a8cdf4ad2299f6d23,,2018-06-02 06:00:00,1, None
19 670,69.66006999999999,19.017,9008,1f0e3dad99908345f7439f8ffabdfcc4,,2018-07-02 15:00:00,1, None
20 671,69.68004,19.072,9008,98f13708210194c475687be6106a3b84,,2018-08-02 18:00:00,1, None

```

Figure 7. 29: Nulling of Person names

7.6 Evaluations

This aspect of the study aimed to assess the results of the adopted methods to determine their level of validity and effectiveness.

7.6. 1 Evaluation of Results of KNN in EDMON-Cluster and KNN in Scikit LearnThe effectiveness of the KNN algorithm which was implemented in this study (K-KUSUM), was initially assessed with simulated infectious data containing location features with known targets or classes. The algorithm was initially trained with an entire dataset and was tested with the same dataset. All the features were correctly predicted to be the true classes. To overcome over fitting, under fitting and class imbalance issues, 660 training and 209 testing datasets of 70%: 30% were randomly simulated (Cochran, 1977; Liu & Cocea, 2017) and used to evaluate the KNN in EDMON- Cluster and the KNN in Scikit-Learn algorithms. 99.52% of the test dataset was accurately classified by the KNN in EDMON- Cluster algorithm. The same datasets were tested with Scikit Learn KNN algorithm which resulted in 93.81% classification accuracy. The accuracies were determined by computing the proportion of the test sets which were correctly classified by the algorithms. As shown in Figure 7.30, the evaluation algorithms predicted the various classes or postcodes of the tested dataset.

```
Prdict class for
      0      1
0  69.55799  19.33103

[9019]

tests-----
      0      1
0  69.57781  18.55499

Prdict class for
      0      1
0  69.57781  18.55499

[9272]
```

Figure 7.30: Output of prediction in the clustering

7.6.2 Assessment of CUSUM

The CUSUM method in the EDMON- Cluster was also evaluated to determine its validity. In the CUSUM evaluation in EDMON- Cluster, the baseline values of past one-week infections (Figure 7.31), were compared with the observed values of current one week (figure 7.32) while taking into consideration, the thresholding of the standard deviations of the baseline values figure 7.33 and 7.34.


```
base_line_Grouped.....
```

	Code	D_Date	Counts
0	9008	2018-08-02 18:00:00	1
1	9022	2018-08-02 18:00:00	4
2	9027	2018-10-02 06:00:00	1
3	9030	2018-09-02 06:00:00	4

Figure 7.31: Sample Baseline data set

```
observed_Count_Grouped.....
```

	Code	D_Date	Counts
0	9006	2018-12-02 06:00:00	2
1	9006	2018-12-03 06:00:00	2
2	9008	2018-11-20 15:00:00	10
3	9020	2018-12-03 06:00:00	2
4	9106	2018-11-02 18:00:00	2
5	9106	2018-11-03 18:00:00	2

Figure 7. 30: Sample of Observed counts for aberration detection

```
observed_data_mergedWith_baseline.....
```

	Lat	Lon	Code	Centroid_ID	Place	D_Date	Counts_x	Counts_y
0	69.65079	18.95493	9008	8	Tromsø	2018-11-20 15:00:00	10	1

Figure 7. 31: Observed and corresponding baseline values

```
data for cumsum!!!!!!!!!!!![1,post_code,m,standDev,observed_data_mergedWith_baseline.iloc[i][-2],observed_data_mergedWith_baseline.iloc[i][-1]]
[0, 9008, 1.0, 0.0, 10, 1]
```

Figure 7.34: Dataset for CUSUM

In the post code area of 9008, the observe count which was 10 infected individuals was indeed more than the average baseline value (1) in addition to three times the standard deviation (0) of

the baseline as shown in the CUSUM dataset in figure 7.34. This indicates excess of infected individuals in the 9008 post code area which indicated a possible outbreak as shown in figure 7.35 and figure 7.36.

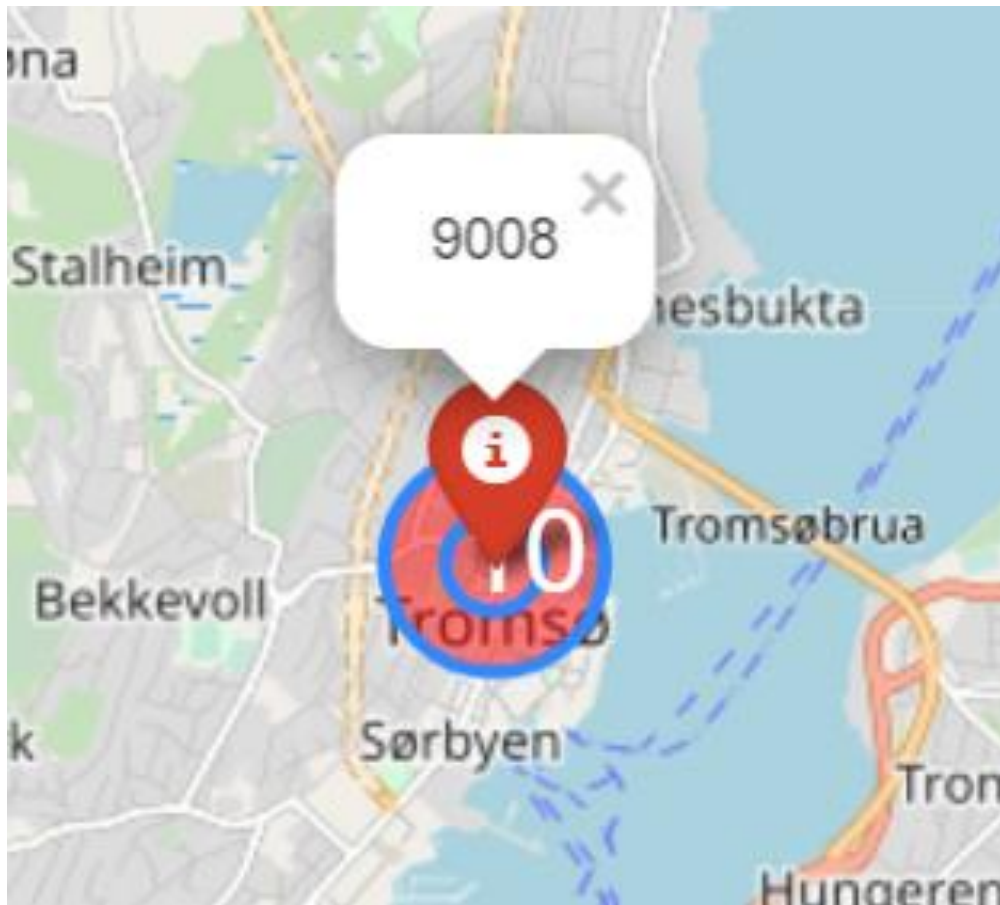


Figure 7.35: Outbreak Cluster



Figure 7.36 Details of outbreak cluster

7.6.4 Visualization, alert and alarms

The results of visualization and SMS alerts are shown in figure 7.37 and figure 7.38. In post code 9022, when there was excess observation like as shown in figure (7.35 and 7.36), an SMS alert was sent as shown in figure 7.37

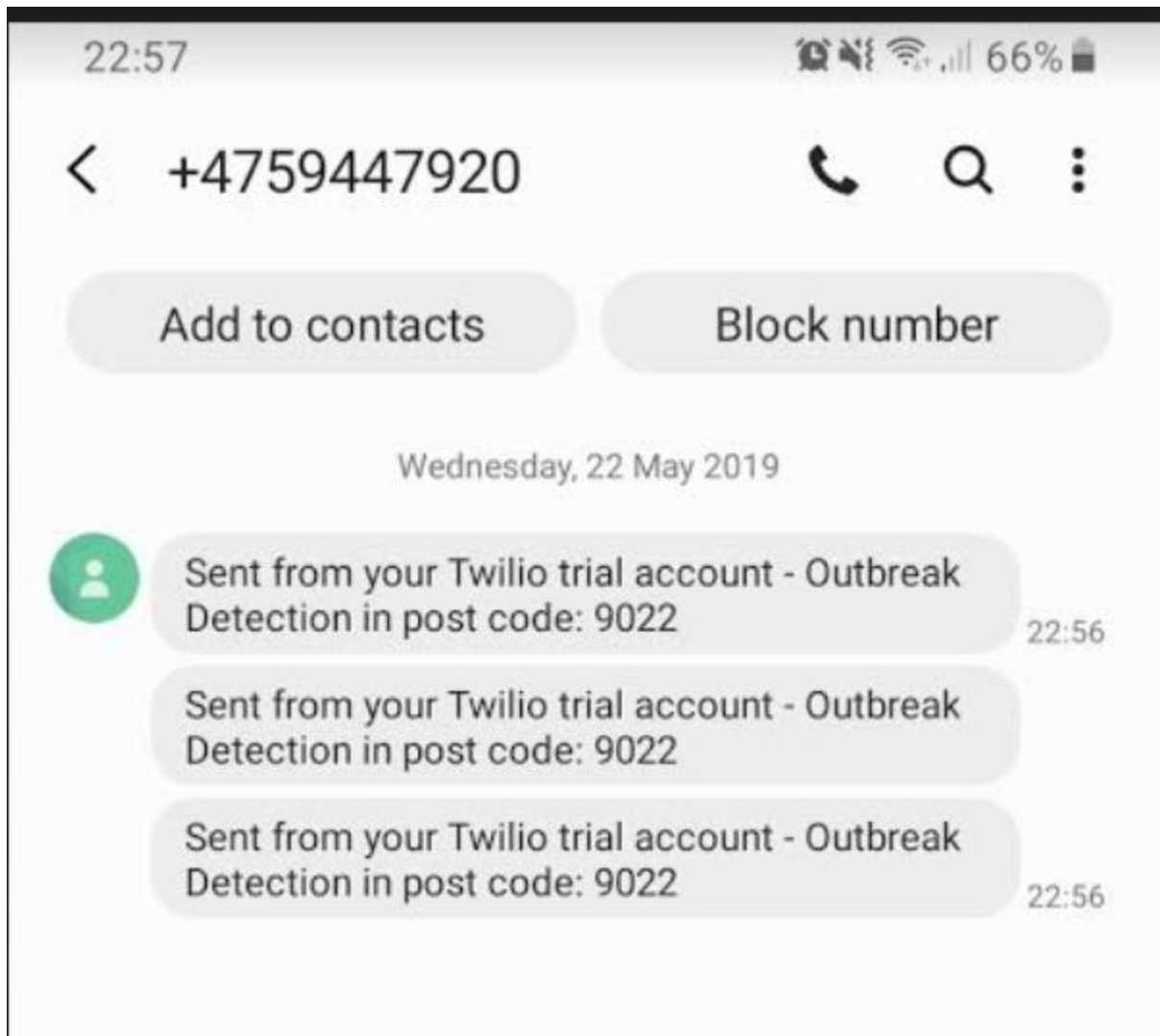


Figure 7.37: SMS alert of outbreak Cluster

7.6.5 Sensitivity, specificity and Timeliness

Out of about 40 spikes of simulated infections which were injected, 13 of them which were spikes of outbreaks were all identified as outbreaks and the remaining 27 which were not outbreak were indeed determined by the CUSUM algorithm as either green or yellow clusters signifying no outbreaks as shown in Table 7.8.

Table 7. 7: Sensitivity and specificity of outbreak clusters

	On Alert Period (ONAP)		
	Yes	No	
Yes	True Positive (TP) A=13	False Positive (FP) B=0	A+B
No	False Negative (FN) C=0	True Negative (TN) D=27	C+D
	A+C ()	B+D	

Therefore, the sensitivity (Se) = $A/A+C$

$$=13/13=1.0 \text{ OR } 100\%$$

The Specificity (Sp) = $B/B+D$

$$=27/27=1.0 \text{ OR } 100\%$$

7.6.6 Timeliness

The timeliness was estimated as shown in figure 7.47 to be about 12.5 minutes in processing from data recording through to visualization and alerts.

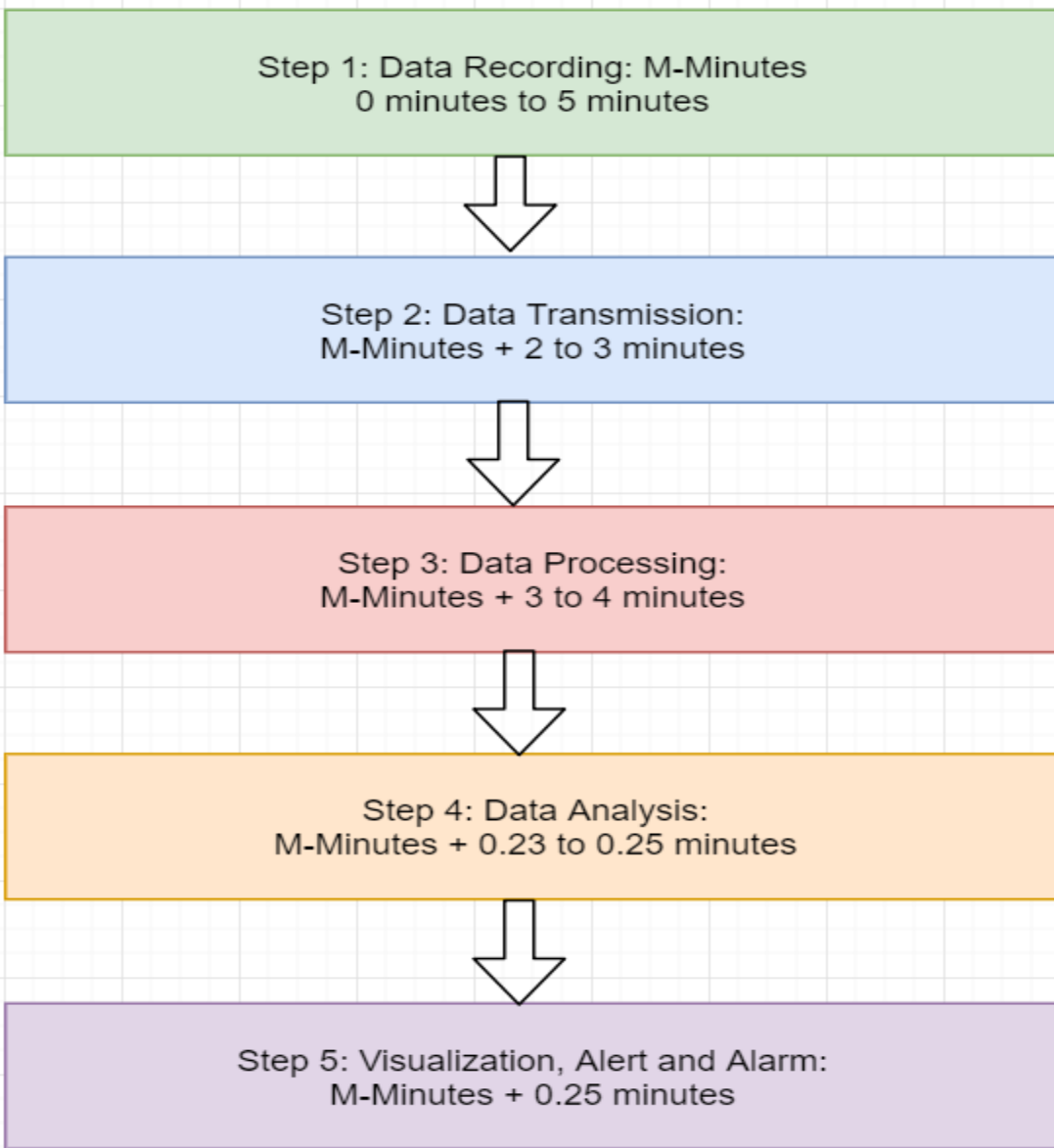


Figure 7.37: Timeliness of clustering and aberration detection in EDMON

Chapter 8: Discussion

8.1 introduction

The main goal of EDMON-Clustering was to develop and assess the performance of the hybridization of KNN and CUSUM spatiotemporal method towards real-time implementation of cluster detection mechanism in Electronic Disease Monitoring Network (EDMON). A design science research (DSR) and prototyping approach with synthetic data was adopted for this study. Various challenges including location estimation and security were considered. Visualization, alert and alarm were also implemented and assessed. The generally results of the study is summarized in table 8.1.

Table 8. 1: Summary of EDMON-Clustering Results

#	Dimension	Description	Results
1	Classification of KNN in EDMON- Cluster	Comparative performance of classification of KNN	99.52%
2	Sensitivity	Combined effect of KNN and CUSUM	100%
3	Specificity	Combined effect of KNN and CUSUM	100%
4	Timeliness	Estimated time of detection	12.5 minutes
5	Alert/Alarm	Trigger, indicating outbreak	Successful
6	Visualization	Graphical display of Detections	Map and Dynamic Graphs
7	Privacy and security	Implementation of one-way hashing and nulling	Successful

The KNN was evaluated with 209 test datasets of which 208 records, representing 99.52% were correctly classified with simulated training set of 660. The results of this can traced in appendix. The CUSUM algorithm in this study was also able to accurately identify all spike injects of infected person's data as either outbreak clusters or non-outbreak clusters. The entire surveillance time was estimated to be 12.5 minutes with the input data.

The presentation of the results did not follow the conventional discrete implementation and respective output of results, but the results were represented by following detailed iterative improvement of the chronology of the results in the prototype approach. This did not only show the trend of revelation of unclear results to the clear understanding of requirement but indicates as well the metamorphosis of the efforts kept in persevering from the genesis to achieve the final presentation of the results. The DSR method was adopted to specify the problem, develop the objective and create the design, evaluation and communication (Doyle et al., 2016). Prototyping involves building, testing, and iterative reworks as necessary, and early approximation of the final system until an acceptable product of the prototype is finally achieved from which the complete system or product can be completely developed (Ashwin, 2017; Tavolato & Vincena, 1984). The

prototyping method is most relevant in circumstances where the project requirements are not priority known in detail (Ashwin, 2017; Tavolato & Vincena, 1984). This iterative trial-and-error process often occur between the developers and the users' model (Ashwin, 2017; Tavolato & Vincena, 1984). The Iterative process began with a very basic and simple implementation of how the software requirements were understood by the developers and iteratively enhanced the evolving versions until the full system was implemented to achieve the desire results as summarized in table 8.1.

In reference to the summary of the results in table 8.1, the performance of K-CUMSUM indeed showed higher prospects and deemed promising for real implementation in EDMON system. The idea of the study approach of synthetic data simulation was adopted from military shooting ranges and cyber ranges. These reserves are used for training and simulating attack-defense scenarios while preserving intrinsic rules such as preserving lives and property (Karami, 2012). In a similar vein, synthetic data for simulating disease outbreak detection in syndromic surveillance, addresses concerns of privacy rights of data subjects in a prototype study such as this. In the absence of synthetic data, a real data of people with type 1 diabetes would have been used in the study, which has the tendency to expose their health conditions and their location trajectories. This contravenes the privacy regulations of GDPR and the personal data acts of individual rights to private life (e-helse, 2018; Yeng, Yang, & Snekenes, 2019). However, the synthetic data served as a reserve or a range for simulating outbreak detections in developing the disease surveillance system in EDMON. The synthetic data also covered the gap of the lack of available known outbreak data in EDMON since the study is a novel area (Karami, 2012).

Further to this, the prototyping approach was highly useful in EDMON- Cluster as the study area was quite new. Determining disease outbreak at the pre-symptomatic stage in EDMON is quite a novel area. As the certainty of the requirements for EDMON-Cluster at the unset was not clear, the iterative, try-and-error-approach of prototyping was ideal to systematically reveal the needed requirement out of the initial fuzzy and unclear visibility of the study area. The required datasets (unclassified, postcode centroid and classified data sets) for the study was therefore developed based on this approach.

In a review for practically implemented algorithms towards the implementation of cluster detection mechanism in EDMON system, Space Time Permutation Scan Statistics (STPSS) was identified to be mostly used (Yeng et al., 2018a). However, the revelations of the weakness of SPSS and the desire for the hybridization of spatial and temporal algorithm to form spatiotemporal algorithm towards overcoming the hind side of the SPSS, motivated this study. Though SPSS is strong in detecting local outbreaks, it lacks the ability to detect outbreaks in a geographically disaggregated data or outbreaks which simultaneously occur in the entire surveillance area (Kulldorff, 2005). In using spatiotemporal clustering algorithms in syndromic surveillance, various methods such as temporal methods and near neighbors could be considered. These measures may augment for the increase in sparseness of data which causes loss of power to detect areas with local excess aberrations in spatial and spatiotemporal methods (Abellan J J, 2007; Isobel et al., 2016; Mathes et al., 2017; Wang et al., 2010; Yih et al., 2010).

The KNN algorithm in the EDMON-Cluster demonstrated high accuracy by correctly classifying 99.52% of the tested dataset with error margin of 0.48%. A further test with another KNN algorithm in Scikit Learn with the same training and test dataset showed that the KNN in EDMON-Cluster performed better, as the Scikit-Learn KNN had 93.81% classification accuracy with higher margin of error of 6.19%. The results can be traced in appendix d. Scikit-learn is a Python module which has been integrated with various state-of-the-art machine learning algorithms for supervised and unsupervised problems. Scikit-learn has good performance, ease of use, documentation, and API consistency and is fit for use in both academic and commercial environment (Brownlee, 2016). It has reasonable precision measure(Kramer, 2016) which provide basis for comparison in evaluations(Olson, La Cava, Orzechowski, Urbanowicz, & Moore, 2017). Disease surveillance systems which relies on geographical location of each detection point with the aim of aggregating the detections in smaller spatial units such as the zip codes for aberration detection, can easily rely on KNN with distance measures. In EDMON, the infection persons (unclassified or unknown classes) are geographically located on their respective latitude and longitude coordinates. If other detections of infectious persons have reference of post code in their geolocations, the Euclidean distances between the unclassified infected person and the referenced subjects with labeled post codes can be computed with them geocodes. What remains a hurdle is to locate a balance point of using geocodes of the surveillance subjects for detecting disease outbreak to safeguard the health of the entire community while maintaining privacy of the subjects.

Out of about 40 diabetes individuals which were formed at various periods through injection of infection status, 13 clusters indicated outbreaks and 27 in total indicated yellow and green to suggest near and no outbreaks respectively. These clusters were evaluated by analyzing the values of the variables in the aberration detection function such as the baseline, observe counts, standard deviations and thresholding. It resulted that all the 13 clusters which indicated red for outbreaks were indeed having excess observational counts which were evaluated to exceed the disease outbreak thresholding of 3 times of the z-score which led to 100% sensitivity score. Similarly, the evaluation results of the 27 clusters met the defined criteria of non-outbreak clusters which resulted in 100% specificity score. The measures seem valid since the detections were not analogue but binary values of 1s and 0s and could be discretely counted to obtain exact non-approximation baseline and observational variables for the CUSUM relation. The sensitivity and specificity results obtained in this study is deemed feasible for actual implementation in EDMON.

Regarding security, personal attributes such as identification numbers (PID) were hashed with MD5 as shown in figure 7.29. The data subjects' names were also nulled as shown in figure 7.30. These methods were guided by the GDPR (Beredskapsdepartementet, 2018; e-helse, 2019; IMPERVA, 2019). The GDPR pointed out anonymization and pseudonymization or encryption methods to be able to mitigate the data subject's privacy risk (Beredskapsdepartementet, 2018; e-helse, 2019; IMPERVA, 2019). Anonymization transforms personal data which is no longer possible to identify individuals (data subjects). According to GDPR (Beredskapsdepartementet, 2018; e-helse, 2019; IMPERVA, 2019), data anonymization is a one-way process and it should be impossible to reverse the process and transform the dataset back into personal data. Personal data that has been sufficiently anonymized is no longer considered as personal data therefore, GDPR

takes the data out of scope of all the GDPR legal privacy obligations (Recital 26). The recommended personal data anonymization methods include nulling, deletion, redaction or obscuring of part of a text for legal or security purposes. However, some of these techniques cannot be employed in datasets which require statistical analysis. Therefore, under GDPR Article 4 (5) some pseudonymization techniques have been recognized to fill this gap. Pseudonymization is a security technique for replacing sensitive data with realistic fictional data that cannot be attributed to a specific individual without additional dataset. Such additional dataset is to be separated during storage and technical and organization measures are to be used to ensure non-attribution to specific individuals

Pseudonymization is necessary because it maintains referential integrity and statistical accuracy, thereby enabling business processes, development and testing systems, training programs, and analysis to operate effectively GDPR (Beredskapsdepartementet, 2018; e-helse, 2019; IMPERVA, 2019). Anonymized data always de-links personal data from identifiable attributes. For instance, personal data is encrypted, and the encryption key is destroyed. However, pseudonymized data is not considered anonymous, when a specific individual can be identified if the pseudonymized and additional non-pseudonymized information are combined to identify the individual GDPR (Beredskapsdepartementet, 2018; e-helse, 2019; IMPERVA, 2019). For pseudonymized data to be taken as anonymous, GDPR requires appropriate and effective technological and organization measures to be implemented to protect the pseudonymized data. For instance, supplementary dataset that have the tendency to help identify specific personal data should be kept separate with the required technical measures such as encryption, hashing, or tokenization. The organizational policies should also prevent unauthorized reversal of the pseudonymization. The MD5 hashing which was used to obscure the personal IDs in the dataset is non-reversal and the de-identification used to remove the names of the data subjects as shown in figure 7.29, adopted the recommended provisions of the GDPR.

The visualization aspect of EDMON-Cluster underwent various transformation to meet expectation. The visualization aspect in EDMON-Cluster, was improved through discussion with experts and colleges from an initial concept as shown in figure 7.3, 7.4 and 7.5 through to 7.35 and 7.36. In the final map output as shown in figure 7.36, the ring size was implemented to indicate the background color as well as the pop-up color with the least value (1) of elevated BG observed since the background color is important indicator of the level of aberration and the status of the cluster. The ring size of the cluster linearly and dynamically changes with the corresponding changes of the observations data. But the ring size remains constant if the observations per cluster becomes greater than 11. This prevents extreme overlaps of clusters and presented adequate views for users.

Dynamic graphs were also presented in figure 7.27 with D3.js, DC.js and cross filter tools. These tools were used to present classified dataset of the conventional graphs to provide users with options of visualization. The combined tools enable a single dataset to be presented in different views such as bar charts, pie-charts among others as shown in figure 7.27. This demonstration is

deemed to be promising which can be improved and inculcated in EDMON-Cluster to provide variety of graphical views for users.

Table 8.2: Comparison of performance metrics

Algorithms	Specificity	Sensitivity	Detected Cases
space-time permutation scan statistic	82	83	26
Pulsar Method	97	85	223
CUMSUM	95	92	
Space Scan Statistics	95	89	790
space-time scan statistic	99	92	3
flexible space-time scan statistic		99.5	4
EDMON- Cluster	100	100	40

The EDMON-Cluster was found to perform better when it was compared with other methods which were found in the review (Yeng et al., 2018b) as shown in table 8.2 and figure 7.36. From Table 8.2, the EDMON-Cluster detected all the outbreak clusters (Sensitivity=100%) and non-outbreak clusters (specificity=100%) of the 40 injected spikes as the true outbreak or the non-outbreak clusters. During the evaluation, there was a cluster as shown in Figure 7.39 which indicated yellow on the cluster background but not on the popup which could have affected the sensitivity and specificity scores of the results but further analysis showed there was a problem with the python module of folium.Icon color variable. The color variable does not include yellow which always results in the popup to take red color when yellow color is called. Other users have encountered related issues and are hoping the issue would be corrected in future updates (Reddit, 2019).

Reflecting the entire results to the onset objectives, EDMON-Cluster approach can be adopted in the EDMON for empirical study. Other studies (Dailey, Watkins, & Plant, 2007; O'Brien & Christie, 1997; Watkins et al., 2008) also revealed CUSUM to be a highly sensitive technique. Therefore, the hybridization of KNN and the CUSUM in this study is deemed promising for implementation in EDMON.

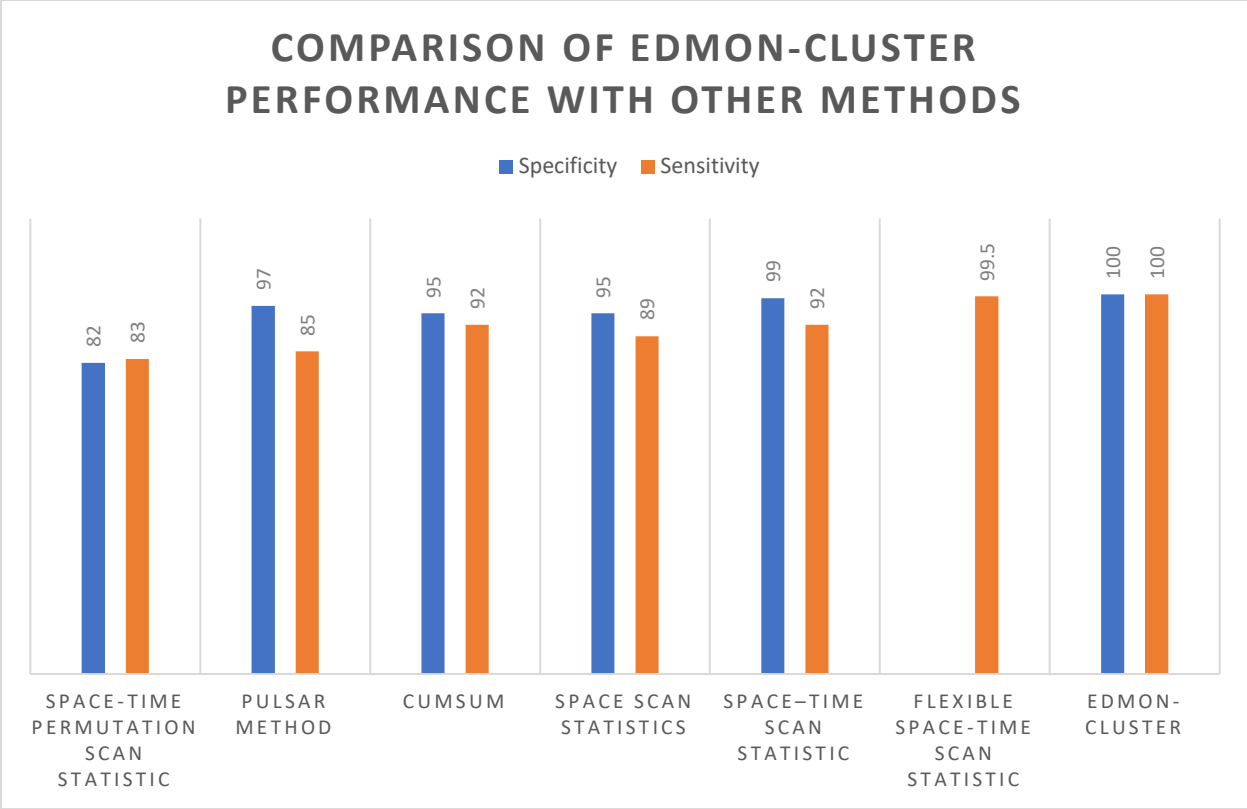


Figure 7.36: Comparison of performance metrics



Figure 7.39: Folium popup color issue

8.2 Conclusion Limitation and Future Works

Basically, KNN and CUSUM algorithms were fused together as a spatiotemporal measure known as EDMON-Clustering. The purpose was to implement an effective algorithm that can be able to cluster a geographically disaggregated data in an empirical cluster detection mechanism in EDMON. The choice of the KNN and CUSUM for this assessment was based on a systematic review which was conducted for algorithms towards implementing cluster detection mechanism in EDMON. That review revealed various methods including Space Time Scan Statistics, KNN, and CUSUM among others. STPSS and CUSUM were mostly used for syndromic surveillance systems however, STPSS was associated with low detection of outbreak in geographically disaggregated data. So, an integration of a temporal method such as CUSUM and a spatial method

such as KNN were proposed to be explored for their combined effectiveness in a geographically disaggregated data.

A prototype method was adopted in the study with synthetic simulated data. The EDMON-Clustering demonstrated higher sensitivity and specificity. This suggests that, the method can be further assessed with real data towards its implementation in EDMON. One-way hashing and nulling were found appropriate for anonymization measures. However better techniques of anonymization are required to be explored towards securing privacy relating to geo-coordinate attributes of the subjects. Various visualization tools have been incorporated and can be improved to meet specific users' requirement.

EDMON-Clustering depends on a one-week baseline data in order to detect aberrations which still have some time lag. Other methods such as prediction mechanism could be explored towards improving this study. Also, there might still be the impact of diseases in a normal distribution below the z-score of 3 standard deviation thresholding mechanism. Future studies could explore towards transforming the EDMON-Clustering by exploring towards reducing the size of this thresholding. The KNN of EDMON-Cluster is also a supervised learning which require historical data for its intelligence of clustering new cases. This makes it challenging for the system to be used without training dataset. EDMON-Cluster can be extended to unsupervised method such as K-means clustering in future where the clustering would not depend on training dataset for intelligence but on only distance measure.

Moreover, EDMON-Clustering was based on simulated data sources, whereas the algorithms in which the EDMON-Cluster was compared with, used actual disease case data as shown in Table 7.39. More informed comparison could be than when EDMON-Clustering is evaluated with real syndromic cases data. The adoption of EDMON-Cluster in EDMON would be a consolidation of the strengths of KNN and CUSUM algorithms. This would help in detecting early disease outbreaks in real time.

REFERENCE

- Abellan J J, R. S., Best. N., (2007). Spatial Versus Spatiotemporal Disease Mapping : Epidemiology. Retrieved from https://journals.lww.com/epidem/fulltext/2007/09001/Spatial_Versus_Spatiotemporal_Disease_Mapping.365.aspx. doi:10.1097/01.ede.0000288446.95319.0a
- Ali, M. A., Ahsan, Z., Amin, M., Latif, S., Ayyaz, A., & Ayyaz, M. N. (2016). ID-Viewer: a visual analytics architecture for infectious diseases surveillance and response management in Pakistan. *Public Health, 134*, 72-85. Retrieved from <http://dx.doi.org/10.1016/j.puhe.2016.01.006>. doi:10.1016/j.puhe.2016.01.006
- Analytics Vidhya. (2018). Introduction to k-Nearest Neighbors: Simplified (with implementation in Python). Retrieved from <https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/>.
- Arsand, E., Walseth, O. A., Andersson, N., Fernando, R., Granberg, O., Bellika, J. G., & Hartvigsen, G. (2005). Using Blood Glucose Data as an Indicator for Epidemic Disease Outbreaks. *Connecting Medical Informatics and Bio-Informatics, 116*, 217-222. Retrieved from <Go to ISI>://WOS:000273025900036.
- Ashwin. (2017, 2017-10-16). Incremental / Prototyping Model- Advantages Disadvantages and when to use. Retrieved from <https://www.daaminotes.com/2017/10/16/prototyping-model-advantages-disadvantages-and-when-to-use/>
- Avison, D., & Fitzgerald, G. (2003). *Information systems development: methodologies, techniques and tools (3rd edition)* (3rd ed.): McGraw-Hill.
- Proposition 56 LS (2017–2018)/Act on the processing of personal data (the Personal Data Act), (2018).
- Bertino, E., & Ferrari, E. (2018). Big Data Security and Privacy | SpringerLink. Retrieved from https://link.springer.com/chapter/10.1007/978-3-319-61893-7_25. doi:10.1007/978-3-319-61893-7_25
- Birant, D., & Kut, A. (2007). ST-DBSCAN: An algorithm for clustering spatial–temporal data. *Data & Knowledge Engineering, 60*(1), 208-221. Retrieved from <http://www.sciencedirect.com/science/article/pii/S0169023X06000218>. doi:<https://doi.org/10.1016/j.datak.2006.01.013>
- Bolandraftar, M., & Imandoust, S. B. (2013). Application of K-nearest neighbor (KNN) approach for predicting economic events theoretical background. Retrieved from https://www.researchgate.net/publication/304826093_Application_of_K-nearest_neighbor_KNN_approach_for_predicting_economic_events_theoretical_background. doi:<http://dx.doi.org/>
- Bonnington, C. (2015). In Less Than Two Years, a Smartphone Could Be Your Only Computer. Retrieved from <https://www.wired.com/2015/02/smartphone-only-computer/>.
- Borgatti, S. (2018). Distance and Correlation.
- Boscovich Roger J., & E.P.George, B. (2018). Chi-Square Distance | SpringerLink. Retrieved from https://link.springer.com/referenceworkentry/10.1007%2F978-0-387-32833-1_53. doi:10.1007/978-0-387-32833-1_53
- Botsis, T., Bellika, J. G., & Hartvigsen, G. (2009). New Directions in Electronic Disease Surveillance: Detection of Infectious Diseases during the Incubation Period. *International Conference on Ehealth, Telemedicine, and Social Medicine: Etelemed 2009, Proceedings*, 176-183. Retrieved from <Go to ISI>://WOS:000267007800031. doi:DOI 10.1109/eTELEMED.2009.9
- Botsis, T., & Hartvigsen, G. (2010). Exploring new directions in disease surveillance for people with diabetes: lessons learned and future plans. *Stud Health Technol Inform, 160*(Pt 1), 466-470.

- Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/20841730>. doi:doi:10.3233/978-1-60750-588-4-466
- Botsis, T., Hejlesen, O., Bellika, J. G., & Hartvigsen, G. (2008). Electronic disease surveillance for sensitive population groups - the diabetics case study. *Stud Health Technol Inform*, 136, 365-370. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/18487758>.
- Botsis, T., Lai, A. M., Palmas, W., Starren, J. B., Hartvigsen, G., & Hripcsak, G. (2012). Proof of concept for the role of glycemic control in the early detection of infections in diabetics. *Health Informatics Journal*, 18(1), 26-35. Retrieved from <http://journals.sagepub.com/doi/abs/10.1177/1460458211428427>. doi:doi:10.1177/1460458211428427
- Bremner, D., Brunswick, U. o. N., Demaine, E., Technology, M. I. o., Erickson, J. G., Science, C., . . . University, M. (2005). Output-sensitive algorithms for computing nearest-neighbour decision boundaries. *Discrete and Computational Geometry*, 33(4), 593-604. Retrieved from <https://experts.illinois.edu/en/publications/output-sensitive-algorithms-for-computing-nearest-neighbour-decis>. doi:10.1007/s00454-004-1152-0
- Brennan, P. F. (2002). AMIA Recommendations for National Health Threat Surveillance and Response. In *J Am Med Inform Assoc* (Vol. 9, pp. 204-206).
- Bronshtein, A. (2017). A Quick Introduction to K-Nearest Neighbors Algorithm. Retrieved from <https://medium.com/@adi.bronshtein/a-quick-introduction-to-k-nearest-neighbors-algorithm-62214cea29c7>.
- Brownlee, J. (2016, 2016-05-31). How To Compare Machine Learning Algorithms in Python with scikit-learn. Retrieved from <https://machinelearningmastery.com/compare-machine-learning-algorithms-python-scikit-learn/>
- Burgard, J. P., Kolb, J.-P., Merkle, H., & Münnich, R. (2017). Synthetic data for open and reproducible methodological research in social sciences and official statistics. *AStA Wirtschafts- und Sozialstatistisches Archiv*, 11(3), 233-244. Retrieved from <https://doi.org/10.1007/s11943-017-0214-8>. doi:10.1007/s11943-017-0214-8
- Casqueiro, J., & Alves, C. (2012). Infections in patients with diabetes mellitus: A review of pathogenesis. In *Indian J Endocrinol Metab* (Vol. 16, pp. S27-36).
- Center, I. K. (2018). IBM Knowledge Center - Distance Metric (nearest neighbor algorithms). Retrieved from https://www.ibm.com/support/knowledgecenter/en/SSLVMB_22.0.0/com.ibm.spss.statistics.algorithms/alg_knn_training_distance-metric.htm.
- Chan, T. C., Teng, Y. C., & Hwang, J. S. (2015). Detection of influenza-like illness aberrations by directly monitoring Pearson residuals of fitted negative binomial regression models. In *BMC Public Health* (Vol. 15).
- Charles P., F., & Jeremy C., W. (2006). Evaluation Methods in Biomedical Informatics | SpringerLink. Retrieved from <https://link.springer.com/book/10.1007%2F0-387-30677-3>.
- Chen, D., Cunningham, J., Moore, K., & Tian, J. (2011). Spatial and temporal aberration detection methods for disease outbreaks in syndromic surveillance systems. <http://dx.doi.org/10.1080/19475683.2011.625979>. Retrieved from <https://www.tandfonline.com/doi/abs/10.1080/19475683.2011.625979>. doi:Annals of GIS, Vol. 17, No. 4, December 2011, pp. 211–220
- Chen H, Zeng D, & P., Y. (2010). *Infectious Disease Informatics Syndromic Surveillance for Public Health and Bio Defense* (1st ed ed.). New York: Springer Science and Business Media.
- Choi, B. C. K. (2012). The Past, Present, and Future of Public Health Surveillance. *Scientifica (Cairo)*, 2012. Retrieved from <http://dx.doi.org/10.6064/2012/875253>. doi:10.6064/2012/875253

- Choi, J., Cho, Y., Shim, E., & Woo, H. (2016). Web-based infectious disease surveillance systems and public health perspectives: a systematic review. *BMC Public Health*, *16*(1), 1238. Retrieved from <https://bmcpublichealth.biomedcentral.com/articles/10.1186/s12889-016-3893-0>. doi:doi:10.1186/s12889-016-3893-0
- Cochran, W. G. (1977). *Sampling Techniques* (3rd ed.). New York: JOHN WILEY & SONS.
- Colwell, R. R. (2004). Infectious disease and environment: cholera as a paradigm for waterborne disease. *INT. MICROBIOL.*, *7*(4), 285-289. Retrieved from <http://scielo.isciii.es/pdf/im/v7n4/Colwell.pdf>.
- Dailey, L., Watkins, R. E., & Plant, A. J. (2007). Timeliness of data sources used for influenza surveillance. *J Am Med Inform Assoc*, *14*(5), 626-631. Retrieved from <https://www.ncbi.nlm.nih.gov/pubmed/17600101>. doi:10.1197/jamia.M2328
- Daulaire, N. M. (2018). Global Health Security.
- Delisle, H., Roberts, J. H., Munro, M., Jones, L., & Gyorkos, T. W. (2005). The role of NGOs in global health research for development. In *Health Res Policy Syst* (Vol. 3, pp. 3).
- Diabetes Research and Wellness Foundation. (2018). Illness And Diabetes. Retrieved from <https://www.diabeteswellness.net/sites/default/files/Illness%20and%20Diabetes.pdf>
- Doyle, C., Sammon, D., & Neville, K. (2016). A design science research (DSR) case study: building an evaluation framework for social media enabled collaborative learning environments (SMECLEs). *Journal of Decision Systems*, *25*(sup1), 125-144. Retrieved from <https://doi.org/10.1080/12460125.2016.1187411>. doi:10.1080/12460125.2016.1187411
- DREWE, J. A., HOINVILLE, L. J., COOK, A. J. C., FLOYD, T., & STÄRK, K. D. C. (2011). Evaluation of animal and public health surveillance systems: a systematic review | *Epidemiology & Infection* | Cambridge Core. Retrieved from <https://www.cambridge.org/core/journals/epidemiology-and-infection/article/evaluation-of-animal-and-public-health-surveillance-systems-a-systematic-review/37E0197F65E69C2C0A3B4D7E69C0C5F7>. doi:doi:10.1017/S0950268811002160
- Duangchaemkarn, K., Chaovatut, V., Wiwatanadate, P., & Boonchieng, E. (2017). *Symptom-based data preprocessing for the detection of disease outbreak - IEEE Conference Publication*. Paper presented at the 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Seogwipo. <https://ieeexplore.ieee.org/document/8037393/citations>
- Code of conduct for information security and data protection in the healthcare and care services sector, (2018).
- e-helse, D. f. (2019). Implementation of GDPR in health care sector in Norway. Retrieved from <https://ehelse.no/personvern-og-informasjonssikkerhet/eus-personvernforordning/implementation-of-gdpr-in-health-care-sector-in-norway>
- Fanaee-T, H. (2012). *Spatio-Temporal Clustering Methods Classification*.
- Flu Near You. (2019). Flu Near You. Retrieved from <https://flunearyou.org/#/>
- GDPR, E. (2018). EU GDPR Information Portal. Retrieved from <http://eugdpr.org/eugdpr.org.html>
- GDPR:Report. (2017). Data masking: anonymization or pseudonymization? - GDPR:Report. Retrieved from <https://gdpr.report/news/2017/09/28/data-masking-anonymization-pseudonymization/>.
- Gil-García, R., & Pons-Porrata, A. (2006). A New Nearest Neighbor Rule for Text Categorization | SpringerLink. Retrieved from https://link.springer.com/chapter/10.1007/11892755_84. doi:10.1007/11892755_84
- Golden, R., & Schell, J. D. (2008). Using ZIP Code and GIS Studies to Assess Disease Risk. In *Environ Health Perspect* (Vol. 116, pp. A18).
- gps-coordinates. (2019). GPS coordinates, latitude and longitude with Google Maps. Retrieved from <https://www.gps-coordinates.net/#>

- Groeneveld, G. H., Dalhuijsen, A., Kara-Zaitri, C., Hamilton, B., de Waal, M. W., van Dissel, J. T., & van Steenberg, J. E. (2017). ICARES: a real-time automated detection tool for clusters of infectious diseases in the Netherlands. *BMC Infect Dis*, *17*(1), 201. Retrieved from <https://www.ncbi.nlm.nih.gov/pubmed/28279150>. doi:10.1186/s12879-017-2300-5
- Heffernan, R., Mostashari, F., Das, D., Karpati, A., Kulldorff, M., & Weiss, D. (2004). Syndromic surveillance in public health practice, New York City. *Emerg Infect Dis*, *10*(5), 858-864. Retrieved from <http://dx.doi.org/10.3201/eid1005.030646>. doi:10.3201/eid1005.030646
- Hope, K., Durrheim, D. N., d'Espaignet, E. T., & Dalton, C. (2006, 17/04/2006). Syndromic surveillance: is it a useful tool for local outbreak detection? *J Epidemiol Community Health*, pp. 374-375. Retrieved from <http://dx.doi.org/10.1136/jech.2005.035337>
- Hu, L. Y., Huang, M. W., Ke, S. W., & Tsai, C. F. (2016). The distance function effect on k-nearest neighbor classification for medical datasets. In *Springerplus* (Vol. 5).
- Hutwagner, L., Browne, T., Seeman, G. M., & Fleischauer, A. T. (2005). Comparing Aberration Detection Methods with Simulated Data. In *Emerg Infect Dis* (Vol. 11, pp. 314-316).
- Hutwagner, L., Thompson, W., Seeman, G. M., & Treadwell, T. (2003). The bioterrorism preparedness and response Early Aberration Reporting System (EARS). *J Urban Health*, *80*(2 Suppl 1), i89-96. Retrieved from <http://dx.doi.org/>.
- IMPERVA. (2019). Pseudonymization | Data Security & Compliance Center | Imperva. Retrieved from <https://www.imperva.com/data-security/compliance-101/pseudonymization/>
- ISO. (2016). ISO 27799:2016(en), Health informatics – Information security management in health using ISO/IEC 27002. In.
- Isobel, M., Chenoweth, P., Okayasu, H., Donnelly, C. A., Aylward, R. B., & Grassly, N. C. (2016). Faster Detection of Poliomyelitis Outbreaks to Support Polio Eradication - Volume 22, Number 3— March 2016 - Emerging Infectious Disease journal - CDC. Retrieved from https://wwwnc.cdc.gov/eid/article/22/3/15-1394_article.
- J. Barker, D. S., S.F. Bloom. (2001). Spread and prevention of some common viral infections in community facilities and domestic homes *Journal of Applied Microbiology*, *91*(1), 7-21. Retrieved from <https://onlinelibrary.wiley.com/doi/full/10.1046/j.1365-2672.2001.01364.x>. doi:10.1046/j.1365-2672.2001.01364.x
- Jacquez, G. (2018). Spatial Clustering and Autocorrelation in Health Events | SpringerLink. Retrieved from https://link.springer.com/referenceworkentry/10.1007%2F978-3-642-23430-9_80. doi:10.1007/978-3-642-23430-9_80
- Jafarpour Khameneh, N. (2014). *Machine Learning for Disease Outbreak Detection Using Probabilistic Models*. (PhD NonPeerReviewed), UNIVERSITÉ DE MONTRÉAL, PolyPublie. Retrieved from <https://publications.polymtl.ca/1659/>
- Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, *31*(8), 651-666. Retrieved from <http://dl.acm.org/citation.cfm?id=1755267.1755654>. doi:10.1016/j.patrec.2009.09.011
- Jeung†, H., Yiu, M. L., Zhou, X., Jensen, C. S., & Shen, H. T. (2008). Discovery of Convoys in Trajectory Databases. Retrieved from <http://www.vldb.org/pvldb/1/1453971.pdf>. doi:10.1192/bjp.bp.113.142612
- Jirina, M., & jr., M. J. (2008). *Classifier Based on Inverted Indexes of Neighbors II- Theory and Appendix*. Retrieved from <http://www.marceljirina.cz/files/classifier-based-on-inverted-indexes-of-neighbors-ii-theory-and-appendix.pdf>
- Jirina, M. J. M. (2010). Using singularity exponent in distance based classifier. doi:10.1109/ISDA.2010.5687263

- Jones, S. G., Ashby, A. J., Momin, S. R., & Naidoo, A. (2010). Spatial implications associated with using Euclidean distance measurements and geographic centroid imputation in health care research. *Health Serv Res, 45*(1), 316-327. Retrieved from <https://www.ncbi.nlm.nih.gov/pubmed/19780852>. doi:10.1111/j.1475-6773.2009.01044.x
- Jordan, J. (2017). K-nearest neighbors. Retrieved from <https://www.jeremyjordan.me/k-nearest-neighbors/>.
- Josseran, L., Fouillet, A., Caillère, N., Brun-Ney, D., Ilef, D., Brucker, G., . . . Astagneau, P. (2010). Assessment of a Syndromic Surveillance System Based on Morbidity Data: Results from the OScour® Network during a Heat Wave. In *PLoS One* (Vol. 5).
- Kajita, E., Luarca, M. Z., Wu, H., Hwang, B., & Mascola, L. (2017). Harnessing Syndromic Surveillance Emergency Department Data to Monitor Health Impacts During the 2015 Special Olympics World Games. *Public Health Rep, 132*(1_suppl), 99s-105s. Retrieved from <http://dx.doi.org/10.1177/0033354917706956>. doi:10.1177/0033354917706956
- Karami, M. (2012). Validity of evaluation approaches for outbreak detection methods in syndromic surveillance systems. *Iranian journal of public health, 41*(11), 102-103. Retrieved from <https://www.ncbi.nlm.nih.gov/pubmed/23304684>.
- Kenneth, E. L. (Ed.) (1986). *The prototyping methodology*: Prentice-Hall, Inc.
- Khokhar, S., & Nilsson, A. A. (2009, 2009//). *Introduction to Mobile Trajectory Based Services: A New Direction in Mobile Location Based Services*. Paper presented at the Wireless Algorithms, Systems, and Applications, Berlin, Heidelberg.
- Kim, J., Kim, B.-S., & Savarese, S. (2012, 01/25/2012). *Comparing image classification methods: K-nearest-neighbor and support-vector-machines*. Paper presented at the Proceedings of the 6th WSEAS international conference on Computer Engineering and Applications, and Proceedings of the 2012 American conference on Applied Mathematics.
- Kleinman, K. P., Abrams, A. M., Kulldorff, M., & Platt, R. (2005). A model-adjusted space-time scan statistic with an application to syndromic surveillance. *Epidemiol Infect, 133*(3), 409-419. Retrieved from <http://dx.doi.org/>.
- Kramer, O. (2016). Scikit-Learn. In *Machine Learning for Evolution Strategies* (pp. 45-53). Cham: Springer International Publishing.
- Kulldorff, M. (2007). A spatial scan statistic. <http://dx.doi.org/10.1080/03610929708831995>. Retrieved from <https://www.tandfonline.com/doi/abs/10.1080/03610929708831995>. doi:Communications in Statistics - Theory and Methods, Vol. 26, No. 6, 1997, pp. 1481–1496
- Lauritzen, A. s. J. N., Årsand, E., Vuurden, K. V., Bellika, J. G., & Hejle, O. K. (2011). Towards a mobile solution for predicting illness in Type 1 Diabetes Mellitus: Development of a prediction model for detecting risk of illness in Type 1 Diabetes prior to symptom onset - IEEE Conference Publication. Retrieved from <https://ieeexplore.ieee.org/document/5940877/authors#authors>.
- Lauritzen, J. N., Arsand, E., Van Vuurden, K., Bellika, J. G., Hejlesen, O. K., & Hartvig-sen, G. (2011). Towards a mobile solution for predicting illness in Type 1 Diabetes Mellitus: Development of a prediction model for detecting risk of illness in Type 1 Diabetes prior to symptom onset. 1-5. doi:10.1109/wirelessvitae.2011.5940877
- Leafletjs. (2019). Leaflet — an open-source JavaScript library for interactive maps. Retrieved from <https://leafletjs.com/>
- Liu, H., & Cocea, M. (2017). Semi-random partitioning of data into training and test sets in granular computing context. *Granular Computing, 2*(4), 357-386. Retrieved from <https://doi.org/10.1007/s41066-017-0049-2>. doi:10.1007/s41066-017-0049-2
- Madhulatha, T. S. (2012). AN OVERVIEW ON CLUSTERING METHODS. *IOSR Journal of Engineering, 24*(719-725). Retrieved from <https://arxiv.org/ftp/arxiv/papers/1205/1205.1117.pdf>.

- Marí Saéz, A., Weiss, S., Nowak, K., Lapeyre, V., Zimmermann, F., Düx, A., . . . Leendertz, F. H. (2015). Investigating the zoonotic origin of the West African Ebola epidemic. *EMBO Mol Med*, 7(1), 17-23. Retrieved from <http://dx.doi.org/10.15252/emmm.201404792>. doi:10.15252/emmm.201404792
- Marshall, J. B., Reynolds, M. R. J., Birch, J. B., Woodall, W. H., & Spitzner, D. J. (2009). Prospective Spatio-Temporal Surveillance Methods for the Detection of Disease Clusters. Retrieved from <https://vtechworks.lib.vt.edu/handle/10919/29639>. doi:<http://scholar.lib.vt.edu/theses/available/etd-11172009-233449/>
- Martin Kulldorff, R. H., Jessica Hartman, Renato Assunção, Farzad Mostashari. (2005). A Space–Time Permutation Scan Statistic for Disease Outbreak Detection. *PLOS Medicine*, 2(3), 126-224. Retrieved from <http://journals.plos.org/plosmedicine/article?id=10.1371/journal.pmed.0020059>. doi:10.1371/journal.pmed.0020059
- Mathes, R. W., Lall, R., Levin-Rector, A., Sell, J., Paladini, M., Konty, K. J., . . . Weiss, D. (2017). Evaluating and implementing temporal, spatial, and spatio-temporal methods for outbreak detection in a local syndromic surveillance system. *PLoS One*, 12(9), e0184419. Retrieved from <https://www.ncbi.nlm.nih.gov/pubmed/28886112>. doi:10.1371/journal.pone.0184419
- MedicineNet. (2017). *Modeling Infectious Diseases in Humans and Animals*.
- Michael Greenacre, & Primicerio, R. (2013). Measures of Distance Between Samples:Euclidean. In *Multivariate Analysis of Ecological Data* (pp. 47-60).
- Musa, G. J., Chiang, P. H., Sylk, T., Bavley, R., Keating, W., Lakew, B., . . . Hoven, C. W. (2013). Use of GIS Mapping as a Public Health Tool—From Cholera to Cancer. In *Health Serv Insights* (Vol. 6, pp. 111-116).
- Nguyen, H. V., & Ba, L. (2010). Cosine Similarity Metric Learning for Face Verification | SpringerLink. Retrieved from https://link.springer.com/chapter/10.1007/978-3-642-19309-5_55. doi:10.1007/978-3-642-19309-5_55
- Nicholas Thapen, Donal Simmie, Chris Hankin, & Gillard, J. (2016). DEFENDER: Detecting and Forecasting Epidemics Using Novel Data-Analytics for Enhanced Response. Retrieved from <http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0155417>. doi:10.1371/journal.pone.0155417
- Nie, S., Lau, E., Lawpoolsri, S., Khamsiriwatchara, A., Liulark, W., Taweeseenepitch, K., . . . Singhasivanon, P. (2014). Real-Time Monitoring of School Absenteeism to Enhance Disease Surveillance: A Pilot Study of a Mobile Electronic Reporting System. In *JMIR Mhealth AND Uhealth* (Vol. 2, pp. 1-10): JMIR.
- O'Brien, S. J., & Christie, P. (1997). Do CuSums have a role in routine communicable disease surveillance? *Public Health*, 111(4), 255-258. Retrieved from <http://dx.doi.org/>.
- Offermann, P., Levina, O., Schönherr, M., & Bub, U. (2009). *Outline of a design science research process*.
- Olson, R. S., La Cava, W., Orzechowski, P., Urbanowicz, R. J., & Moore, J. H. J. B. M. (2017). PMLB: a large benchmark suite for machine learning evaluation and comparison. *10*(1), 36. Retrieved from <https://doi.org/10.1186/s13040-017-0154-4>. doi:10.1186/s13040-017-0154-4
- Omicsonline. (2018). Inclusion and Exclusion Criteria and Rationale. Retrieved from <https://www.omicsonline.org/articles-images/2157-7595-5-183-t001.html>
- P.N. Tan, Vipin Kumar, & Steinbach, M. (2005). Cluster Analysis: Basic Concepts and Algorithms.
- PAGE, E. S., & Statistical Laboratory, U. o. C. (1954). CONTINUOUS INSPECTION SCHEMES. *Biometrika*, 41(1-2), 100-115. Retrieved from <https://academic.oup.com/biomet/article-pdf/41/1-2/100/1243987/41-1-2-100.pdf>. doi:10.1093/biomet/41.1-2.100

- Pedersen, S., & Hartvigsen, G. (2015). Lessons learned from 25 years with telemedicine in Northern Norway. Retrieved from http://www2.telemed.no/publikasjoner/prosjektrapporter/NST-rapport_2015-06_Lessons_learned_from_25_years_with_telemedicine_in_Northern_Norway-20MB.pdf. doi:978-82-8242-053-2
- Peter A., R. (2005). Spatial and Syndromic Surveillance for Public Health (Lawson/Spatial and Syndromic Surveillance for Public Health) || References - [PDF Document]. Retrieved from <https://vdocuments.mx/spatial-and-syndromic-surveillance-for-public-health-lawsonspatial-and-syndromic-5852891dd9971.html>.
- PK.Ragunath, S.Velmourougan , P. Davachelvan, S.Kayalvizhi, & R.Ravimohan. (2010). Evolving A New Model (SDLC Model-2010) For Software Development Life Cycle (SDLC). *International Journal of Computer Science and Network Security*, 10(1), 112-120. Retrieved from http://paper.ijcsns.org/07_book/201001/20100115.pdf.
- Quinn, S. C., & Kumar, S. (2014). Health Inequalities and Infectious Disease Epidemics: A Challenge for Global Health Security. In *Biosecur Bioterror* (Vol. 12, pp. 263-273).
- Reddit. (2019). python - Trying to change marker colors using Folium map module. Retrieved from https://www.reddit.com/r/learnpython/comments/4bs9t2/trying_to_change_marker_colors_using_folium_map/
- RIAKTR. (2016, 2016-09-23). Why synthetic data is about to become a major competitive advantage - Riaktr. Retrieved from <https://www.riaktr.com/synthetic-data-become-major-competitive-advantage/>
- Robertson, S., & Robertson, J. (2006). *Mastering the Requirements Process (3rd Edition)*: Addison-Wesley Professional.
- Rogerson, P. A. (1997). Surveillance systems for monitoring the development of spatial patterns. *Stat Med*, 16(18), 2081–2093 Retrieved from <https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291097-0258%2819970930%2916%3A18%3C2081%3A%3AAID-SIM638%3E3.0.CO%3B2-W>. doi:[https://doi.org/10.1002/\(SICI\)1097-0258\(19970930\)16:18<2081::AID-SIM638>3.0.CO;2-W](https://doi.org/10.1002/(SICI)1097-0258(19970930)16:18<2081::AID-SIM638>3.0.CO;2-W)
- Rogerson, P. A. (2005). Spatial Surveillance and Cumulative Sum Methods. In A. B. Lawson & K. Kleinman (Eds.), *Spatial and Syndromic Surveillance for Public Health* (1st ed., pp. 269): John Wiley & Sons, Ltd.
- Savel, T. G., & Foldy, S. (2012). *The Role of Public Health Informatics in Enhancing Public Health Surveillance*. Retrieved from <https://www.cdc.gov/mmwr/preview/mmwrhtml/su6103a5.htm>
- scikit-learn developers. (2011). Clustering.
- Sharip, A. (2006). *Preliminary Analysis of SaTScan's Effectiveness to Detect Known Disease Outbreaks Using Emergency Department Syndromic Data in Los Angeles County*.
- Silverman, B. W., & Jones, M. C. (1989). E. Fix and J.L. Hodges (1951): An Important Contribution to Nonparametric Discriminant Analysis and Density Estimation: Commentary on Fix and Hodges (1951). *International Statistical Review / Revue Internationale de Statistique*, 57(3), 233-238. Retrieved from <http://www.jstor.org/stable/1403796>. doi:10.2307/1403796
- Singh, A., Yadav, A., Rana, A., Singh, A., Yadav, A., & Rana, A. (2013). K-means with Three different Distance Metrics. Retrieved from <https://www.ijcaonline.org/archives/volume67/number10/11430-6785>.
- Steen, M. V., & Tanebaum, A. S. (2017). *Distributed Systems 3rd edition (2017) | DISTRIBUTED-SYSTEMS.NET* (2nd Ed ed.): Pearson International Edition.
- Struchen, R., Vial, F., & Andersson, M. G. (2017). Value of evidence from syndromic surveillance with cumulative evidence from multiple data streams with delayed reporting. *Sci Rep*, 7(1), 1191.

- Retrieved from <https://www.ncbi.nlm.nih.gov/pubmed/28446757>. doi:10.1038/s41598-017-01259-5
- Study.com. (2018). Progress of Disease: Infection to Recovery - Video & Lesson Transcript | Study.com. Retrieved from <http://study.com/academy/lesson/progress-of-disease-infection-to-recovery.html>
- Tavolato, P., & Vincena, K. (1984). *A Prototyping Methodology and Its Tool*, Berlin, Heidelberg.
- Teknomo, K. (2017). Euclidean Distance. Retrieved from <https://people.revoledu.com/kardi/tutorial/Similarity/EuclideanDistance.html>.
- Tsui, F. C., Espino, J. U., Dato, V. M., Gesteland, P. H., Hutman, J., & Wagner, M. M. (2003). Technical Description of RODS: A Real-time Public Health Surveillance System. In *J Am Med Inform Assoc* (Vol. 10, pp. 399-408).
- Twilio. (2019). Twilio SMS Python Quickstart - Send & Receive SMS. Retrieved from https://www.twilio.com/docs/sms/quickstart/python?utm_source=docs&utm_medium=social&utm_campaign=guides_tags
- UKEssays. (2018). Systems Analysis And Requirements Analysis Information Technology Essay. Retrieved from <https://www.ukessays.com/essays/information-technology/systems-analysis-and-requirements-analysis-information-technology-essay.php>
- Wang, H. (2014). *Pattern Extraction From Spatial Data - Statistical and Modeling Approches*. University of South Carolina, Retrieved from <https://scholarcommons.sc.edu/etd/3035>
- Wang, X., Zeng, D., Seale, H., Li, S., Cheng, H., Luan, R., . . . Wang, Q. (2010). Comparing early outbreak detection algorithms based on their optimized parameter values. *J Biomed Inform*, 43(1), 97-103. Retrieved from <http://dx.doi.org/10.1016/j.jbi.2009.08.003>. doi:10.1016/j.jbi.2009.08.003
- Watkins, R. E., Eagleson, S., Veenendaal, B., Wright, G., & Plant, A. J. (2008). Applying cusum-based methods for the detection of outbreaks of Ross River virus disease in Western Australia. In *BMC Med Inform Decis Mak* (Vol. 8, pp. 37).
- WHO. (2006). Communicable diseasesurveillance and responsesystems Guide to monitoring and evaluating. Retrieved from <https://www.scribd.com/document/212386787/WHO-CDS-EPR-LYO-2006-2-eng>.
- WHO. (2015). *WHO STRATEGIC RESPONSE PLAN West Africa Ebola Outbreak*. Retrieved from http://apps.who.int/iris/bitstream/10665/163360/1/9789241508698_eng.pdf
- WHO. (2017a, June 2017). Ebola Virus Disease. Retrieved from <http://www.who.int/mediacentre/factsheets/fs103/en/>
- WHO. (2017b). WHO | Diabetes. *WHO*. Retrieved from <http://www.who.int/mediacentre/factsheets/fs312/en/>. doi:/entity/mediacentre/factsheets/fs312/en/index.html
- WHO. (2018). Countries prioritize Health Security to address disease outbreaks. Retrieved from <https://www.afro.who.int/news/countries-prioritize-health-security-address-disease-outbreaks>.
- Woldaregay, A., Årsand, E., Botsis, T., & Hartvigsen, G. (2017). An Early Infectious Disease Outbreak Detection Mechanism Based on Self-Recorded Data from People with Diabetes. *Studies in health technology and informatics*, 245, 619-623. doi:10.3233/978-1-61499-830-3-619
- Woldaregay, A. Z., Årsand, E., Giordanengo, A., Albers, D., Mamykina, L., Botsis, T., & Hartvigsen, G. (2017). *EDMON-A Wireless Communication Platform for a Real-Time Infectious Disease Outbreak Detection System*. Paper presented at the 15th SHI Conference, Kristiansand. <http://www.ep.liu.se/ecp/145/003/ecp17145003.pdf>
- Woldaregay, A. Z., Årsand, E., Giordanengo, A., Albers, D., Mamykina, L., Botsis, T., & Hartvigsen, G. (2018). *EDMON-A Wireless Communication Platform for a Real-Time Infectious Disease Outbreak De-tetection System Using Self-Recorded Data from People with Type 1 Diabetes*. Paper presented

- at the Proceedings from The 15th Scandinavian Conference on Health Informatics 2017 Kristiansand, Norway, August 29–30, 2017.
- Woodward, R. H. (1964). Cumulative sum techniques / by R.H. Woodward [and] P.L. Goldsmith. - Version details. Retrieved from <https://trove.nla.gov.au/version/32819702>.
- Yang, W., Xu, L., Chen, X., Zheng, F., & Liu, Y. (2015). Chi-Squared Distance Metric Learning for Histogram Data. *Mathematical Problems in Engineering, 2015*, 1-12. doi:10.1155/2015/352849
- Yang, X., & Abraham O. Fapojuwo. (2015). Performance analysis of hexagonal cellular networks in fading channels - Yang - 2016 - Wireless Communications and Mobile Computing - Wiley Online Library. Retrieved from <https://onlinelibrary.wiley.com/doi/full/10.1002/wcm.2573>. doi:10.1002/wcm.2573
- Yeng, P., Yang, B., & Snekenes, E. (2019, 15-19 July 2019). *Observational Measures for Effective Profiling of Healthcare Staffs' Security Practices*. Paper presented at the 2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC).
- Yeng, P. K., Woldergay, A. z., Solvoll, T., & Hartvigsen, G. (2018a, 2018-08-24). *A systematic review of cluster detection mechanisms in syndromic surveillance: Towards developing a framework of cluster detection mechanisms for EDMON system* Paper presented at the Scandinavian Conference on Health Informatics, Aalborg, Denmark.
- Yeng, P. K., Woldergay, A. z., Solvoll, T., & Hartvigsen, G. (2018b). A systematic review of cluster detection mechanisms in syndromic surveillance: Towards developing a framework of cluster detection mechanisms for EDMON system | Request PDF. Retrieved from https://www.researchgate.net/publication/327884946_A_systematic_review_of_cluster_detection_mechanisms_in_syndromic_surveillance_Towards_developing_a_framework_of_cluster_detection_mechanisms_for_EDMON_system. doi:http://dx.doi.org/
- Yih, W. K., Deshpande, S., Fuller, C., Heisey-Grove, D., Hsu, J., Kruskal, B. A., . . . Platt, R. (2010). Evaluating Real-Time Syndromic Surveillance Signals from Ambulatory Care Data in Four States. In *Public Health Rep* (Vol. 125, pp. 111-120).
- zakka, k. (2018). A Complete Guide to K-Nearest-Neighbors with Applications in Python and R. Retrieved from <https://kevinzakka.github.io/2016/07/13/k-nearest-neighbor/>
- Zickuhr, K., & Smith, A. (2012). *Digital differences*. Retrieved from

Appendix

The appendix contains the project source code files and simulated data in respective folders.

- a. In the Final_Project_files folder, K_CUSUM is the python file with the developed KNN and CUSUM algorithm
- b. The basic test folder in the knn folder contains the test data, the algorithm of KNN in K-CUSUM which was tested and the correctly classified dataset.
- c. The evaluation folder in the knn folder, contains the test data, the algorithm of KNN in K-CUSUM which was tested and the correctly classified dataset
- d. The Scikit Learn folder in the Final_Project_Files folder contains the evaluation data and the code used to evaluate the performance of Scikit Learn KNN. The data folder contains the data and the evaluation code contains the algorithm code
- e. Cluster Visualization is the data visualization model of the study. Cluster_Visualization folder contains the libraries folder (Scripts and Content, idea), the source code file (index) and the simulated data which was visualized.