An Efficient Approach for Outlier Detection in Big Data

A Thesis
submitted in fulfillment of the requirements
for the award of degree of

Doctor of Philosophy

by
Bharti Saneja
(Roll No. : 901403024)

Under the Guidance of
Dr. Rinkle Rani
(Associate Professor, Computer Science and Engineering Department,
Thapar Institute of Engineering and Technology, Patiala)

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY
PATIALA – 147004

May 2019

Table of Contents

List of Figures

List of Tables

List of Algorithms

Certificate

I hereby certify that the work which is presented in this thesis entitled **"An Efficient Approach for Outlier Detection in Big Data"**, in fulfillment of the requirement for the award of degree of **"Doctor of Philosophy"** submitted in Computer Science and Engineering Department of Thapar Institute of Engineering and Technology, Patiala, is an authentic record of my own work carried out under the supervision of **Dr. Rinkle Rani** and refers other research works which are duly listed in the reference section.

The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.

Bharti

(Bharti Saneja)
Regn. No. 901403024

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Rinkle

(Dr. Rinkle Rani)

Associate Professor

Computer Science & Engineering Department

Thapar Institute of Engineering and Technology,

Patiala, India

Acknowledgement

I would like to express sincere appreciation to my supervisor **Dr. Rinkle Rani** for being a pillar of support and encouragement throughout my research work. Her experience, strength, tenderness and willfulness, have taught me valuable lessons of life, that are going to be of immense help to me in taking decisions throughout my life. My sincere thanks to **Dr. Maninder Singh**, Professor and Head, CSED, Thapar Institute of Engineering and Technology for providing me the necessary administrative assistance and infrastructure that helped me in completion of my research work. I am thankful to my doctoral committee members, **Dr. Vijay Kumar**, Assistant Professor, CSED, **Dr. Anju Bala**, Assistant Professor, CSED and **Dr. M.D. Singh**, Associate Professor, EIED, Thapar Institute of Engineering and Technology, Patiala for their helpful suggestions and regularly ensuring the progress of my research work. I am thankful to all the faculty, staff members and research scholars of Computer Science and Engineering Department for their support.

I offer my deepest gratitude to my father, **Mr. Chiman Lal** whose love and constant encouragement has been a major source of inspiration in turning my vision into a reality. I am also thankful to my mother **Mrs. Asha Saneja** and my brothers **Dr. Kuldeep Saneja and Sushil Saneja** for love, encouragement, motivation and confidence in me. I also offer my gratitude towards my grandparents **Sh. Ram Chander Saneja** and **Sh. Lachman Dass Saneja** for showering their blessings on me.

I also acknowledge the cooperation and encouragement by my friends for providing support and motivation in this journey. The chain of gratitude would be definitely

incomplete without thanking the supreme power, the **Almighty** for showering blessings in all of my endeavors.

Patiala (Bharti Saneja)

May, 2019

Abstract

Outlier detection is an important aspect of data mining which discovers the unusual events that occurs in data. Big data has large volume of unseen knowledge and many perceptions which have raised significant challenges in knowledge discovery. In certain kinds of data, the association among the different attributes is of much more significance than the information itself. Hence, in such datasets before detecting outliers these associations needs to be extracted. The associations can be mined by analyzing correlation among various attributes. However, it is very challenging to acquire ample benefits from the large amount of complex data. To overcome these issues, various methods for analyzing correlation are studied. Also, various existing approaches for outlier detection based on supervised and unsupervised learning models are studied. In recent times, these approaches have become an indispensable tool for detecting anomalous events in various domains.

With the advancement in sensor technologies, a lot of data is being generated by wireless sensors in various application domains. In this study, the main concern is on data generated from wireless body sensor networks. As caretaker may not be always available to monitor physiological parameters so, different sensors are attached with the body of patient to remotely monitor the health of the patient. Outlier detection in this domain detects the anomalous activities based on the sensor measurements and differentiates the sensor fault from true medical condition.

This thesis carried out research work in the field of outlier detection in wireless body area sensor networks. The key objective of the research is to explore the profits of using distributed map reduce framework for outlier detection. An approach is proposed to detect outliers based on the assumption that data attributes are linearly related to each other.

Further, as it is seen that in real application scenarios none of the sensors exhibit a truly linear relationship. Hence, to deal with non-linear aspect of data the proposed approach is further enhanced so that it can be able to detect outliers in dataset where data attributes are linearly or non-linearly correlated. The results of both the proposed approaches are proved to be effective than other competent approaches in terms of processing time and accuracy of outlier detection. The approaches are also tested for scalability by forming a multinode Hadoop cluster of eight nodes.

Furthermore, an integrated framework for outlier detection is proposed that is based on data compression, data clustering, and cluster refinement. The clustering algorithm in the proposed framework works on the principle of clonal selection algorithm and uses the objective function of fuzzy clustering. It is seen that the clusters formed by proposed clustering algorithm have more optimal structures than state of art clustering algorithms. The formed clusters are further refined using cluster refinement algorithm to increase accuracy of outlier detection. The results of the proposed framework show that it outperforms the competent algorithms in various aspects of processing time and detection rate.

It is suggested that the utilization of correlation between attributes detects discriminate and significant events, which can help in accurate classification of events and also reduce false alarms which can further aid in better utilization of resources.

CHAPTER 1

INTRODUCTION

With tremendous increase in the users of digital devices, the volume of data has also been increased to many folds. The expected increase in the data volume in 2020 is around 44 folds of the volume in 2009 [1]. The volume of data is increasing exponentially with respect to time. This leads to big data come into picture. Due to high complexity and variability in big data, it cannot be handled by traditional databases. Hence, there is a need for techniques and tools that can efficiently analyze information from such a large amount of data. Big data trend occurs due to large amount of hidden information in big datasets. Doug Laney [2] explained big data in terms of three V's i.e. volume, velocity and variety. Mark van Rijmenam redefined big data with four more v's that are variability, veracity, visualization and value.

Data plays a crucial role in various applications like business analytics, social network analysis, healthcare etc. New government policies are also being implemented based on social media analytics to discover new ideas or schemes that can facilitate their infrastructures. Apart from aforementioned applications of data analytics, e-commerce service providers also keep track of their customers purchasing history to predict future sales and provide special offers based on customer's interests. Hence, with the help of data analytics future prediction of market analysis is possible.

Outlier detection is an important aspect of data analytics which aims at detecting the objects that behave differently from other objects. Outlier detection is not a well-expressed issue, hence

requires a lot of attention in various areas like fraud detection, fault detection, disease detection, terrorism, novelty detection etc. With the advancement in sensor technologies, wireless sensor networks are pervasive and are generating tons of data every second in various application domains like communication networks, healthcare monitoring, weather forecasting etc. So, there is the need for big data analytics and outlier detection in wireless sensor networks. Outlier in a wireless sensor network is either due to an anomaly in the actual application scenario or due to some sensor fault. Hence, there is the need of techniques that detect outliers in WSNs and the cause of the occurrence of anomalous situation. The general reasons of outliers in a data set are [3]:

- Errors during the data entry caused by human mistakes
- Errors due to instrument fault
- Errors occurred during experimental analysis
- Any unusual event occurred in a dataset
- Deliberate outliers to test any method
- Outliers occurred due to data processing
- Outliers occurred while combining data from different sources
- Uniqueness or novelty in data

1.1 Need for Outlier Detection

Outlier detection is a subpart of data mining which refers to identification of data items or events which are deviated from normal behavior [4]. These deviated data items are referred to as

anomalies or outliers. Nowadays, huge amount of data is being produced every second and it is not possible to analyze every single piece of information. Most of the useful information remains unknown until we require that information or that information causes harm to the system. Therefore, to detect critical, malicious or new information, outlier detection came into picture which tells the data administrator about what is inside the data without having knowledge of the event.

1.2 Classification of Outlier Detection Approaches

Detecting anomalous events in the huge amount of data poses various processing challenges because of increase in storage and time overhead. Discovering hidden information from such frenzied data cannot be easily obtained without using appropriate methods. One way to analyze hidden information is to cluster the data based on some criteria with the help of which anomalous events can be identified. Various other popular approaches for outlier detection are data classification, nearest neighbor based, statistical based, information theoretic and spectral based. The classification of outlier detection approaches is given in Figure 1.1.

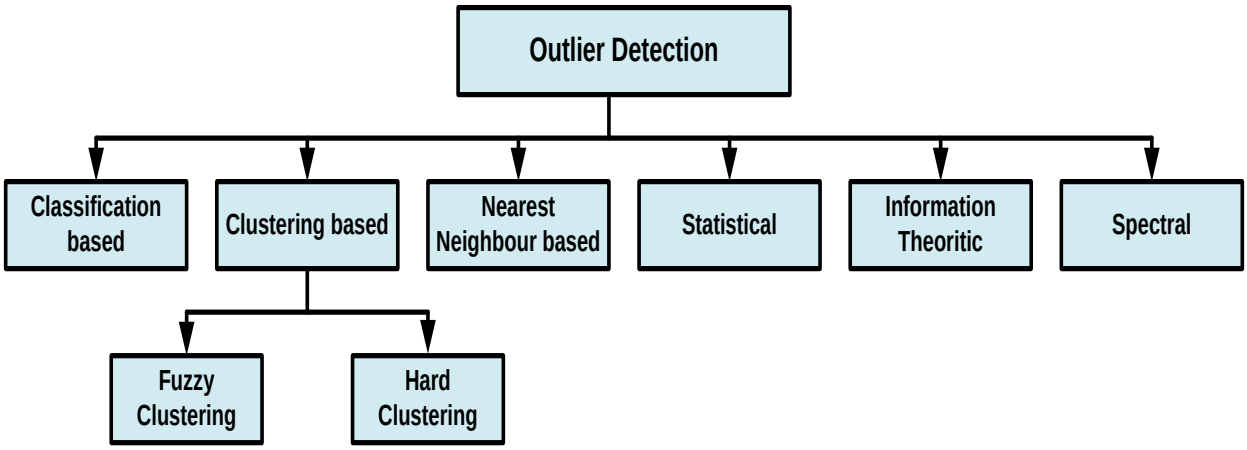
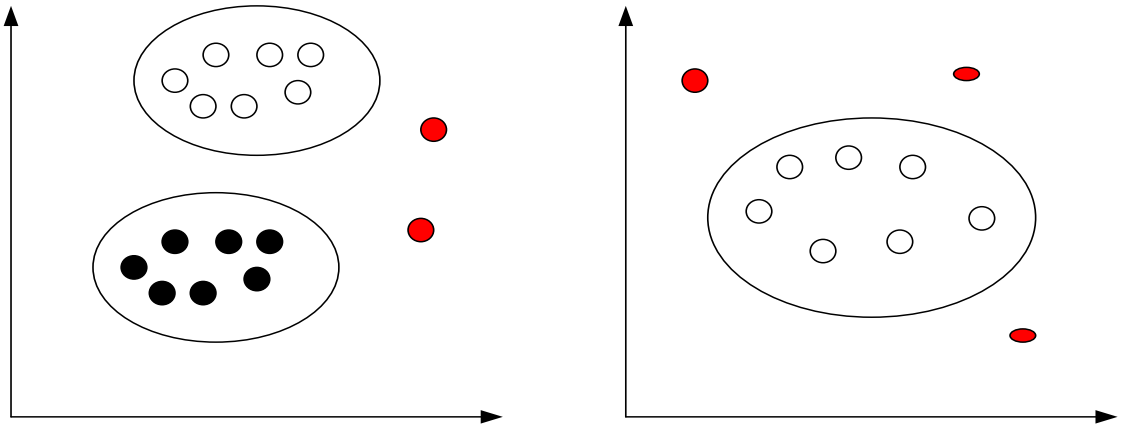


Figure 1.1: Classification of outlier detection approaches

The main focus of this research is on clustering and classification based approaches for outlier detection. The detailed explanation of various approaches is as follows:

Classification based techniques for outlier detection

In classification based techniques for outlier detection a model is trained based on the labeled data which is then used to classify the test instances [5, 6]. These techniques works in two phases: In the first phase a classifier is trained based on the existing labeled data, in the second phase a data instance is tested and classified as normal or outlier using the trained classifier model. Classification based techniques for detecting outliers is broadly divided into two categories: multi-class outlier detection and one-class outlier detection. In multi-class outlier detection techniques the labeled data consists of multiple normal classes as shown in Figure 1.1(a). Red data instances in the Figure 1.1 are outliers. The aim of multi-class classifier is to correctly identify the class of a data instance. In multi-class classifier there are multiple normal and multiple anomalous classes [7, 8]. In one-class outlier detection techniques there is a clear boundary outside the normal data instances as shown in Figure 1.1(b).



a) Multi-class outliers b) One-class outliers

Figure 1.2: Categorization of classification based techniques

Various examples of one-class classification are one-class support vector machines [9] Kernel Fisher Discriminants [10, 11] etc. In one-class classifiers any data instance outside the trained boundary for normal instances is classified as anomalous.

Nearest neighbor based techniques for outlier detection

Various outlier detection techniques are based on the idea of nearest neighbor exploration. These techniques are based on the assumption that normal data occur in dense regions, whereas outliers occur far away from their nearest neighbors. A distance measure between the two data points need to be defined in nearest neighbor based outlier detection. Eucledian distance measure is generally used for continuous feature space [5] and simple matching coefficient is more often used for categorical features [12]. Outlier detection techniques based on nearest neighbor can be classified into two categories:

- Techniques in which outlier score is calculated based on the distance of a data point from its nearest neighbor.
- Techniques in which outlier score is calculated based on the relative density of the data point.

Clustering based techniques for outlier detection

Clustering is an unsupervised learning technique to group similar data instances based on some criteria [13]. Although clustering is based on unsupervised learning model but currently clustering based on semi supervised learning is also explored [14]. Even though clustering and outlier detection seem to be basically different from one another, there are various clustering based outlier detection techniques. There exist three categories of clustering based outlier detection techniques:

- The first category is based on the assumption that outliers in a dataset do not belong to any cluster. Various existing clustering algorithms for outlier detection based on this assumption are DBSCAN, FindOut, ROCK, WaveCluster and SNN clustering [15-19]. The drawback of these algorithms is that the main goal is to find clusters so these algorithms are not efficient to find outliers.
- The second category is based on the assumption that normal data points lie near to the nearest cluster centroid, whereas outliers are far-off from the nearest cluster centroid. The approaches based on this assumption comprise of two phases: in the first phase, the data points are clustered and in the second phase, based on the distance of each data point from its nearest cluster centroid an outlier score is calculated. Various existing clustering algorithms that follow this two phase approach are: Self Organizing Maps that is based on semi supervised model, K-means clustering that is based on unsupervised learning, Expectation Maximization etc. [20-28]. In this category of algorithms if the training data consists of multiple classes then, to improve cluster quality semi-supervised clustering model can be used [29-30].
- A drawback with previous category of techniques is that it is not able to detect outliers if outliers in the dataset form a cluster. To deal with this issue another category of techniques is formed that is based on assumption that normal data belongs to dense clusters whereas outliers belongs to sparse clusters. Various existing approaches based on this assumption are FindCBLOF, k-d trees, CD-trees etc. [31-32].

Statistical techniques for outlier detection

The fundamental belief of a statistical outlier detection approach is that: "An outlier is a data instance which is assumed of being irrelevant since it is not created by the expected stochastic

model" [33]. The assumption on which stastical outlier detection techniques are based on is that the normal data points lie in high probability areas of a stochastic model whereas outliers occur in low probability areas. In statistics based techniques based on the available data a stastical model is fitted and then an inference test is applied on the test data to check whether the data instance belongs to model or not. The data instances having low probability of creation are considered as anomalies. Both parametric [34] and non-parametric [35] approaches are applied for fitting of a statistical model.

Information theoretic techniques for outlier detection

In information theoretic techniques for outlier detection the information is analyzed with the help of various information theoretic measures. The assumption in this technique is that outliers in a dataset cause irregularities in the data. The information theoretic measures used in these techniques are Kolomogorov complexity, size of data file, size of the regular expression, relative uncertainty etc. [36-41]. The basic idea behind these techniques is dual optimization that is to decrease the size of subset and also to reduce the complexity of the dataset. Information theoretic methods are used for naturally arranged sequential and spatial data. Initially the data is divided into pieces of sequences then using an information theoretic technique a subsequence S is detected such that $C(D) - C(D - S)$ is maximum. Here C(D) is complexity of dataset D. The main issue with these approaches is to detect the optimal subsequence with the help of which outliers can be detected.

Spectral techniques for outlier detection

The spectral techniques for outlier detection are generally applied to high dimensional datasets. The main assumption in these approaches is that data can be projected to low dimensional

subspace where there is significant difference between outliers and normal data instances. Thus, the main aim of any spectral technique is to find such embedding or projections in which outliers can be effortlessly identified [42]. Most of the existing spectral techniques for outlier detection are based on Principal Component Analysis (PCA) for reducing dimensionality of the dataset [43-46]. A spectral technique is proposed by Dutta et al. [47] for outlier detection in astronomy catalogs. Ide and Kashima [48] suggested a spectral technique for outlier detection in time series graph datasets. Fujimaki et al. [49] proposed an approach to detect outliers in space craft components.

1.3 Outlier Detection in Sensor Networks

Outlier detection is an evolving area in the field of data mining and analytics [5]. It is broadly applicable in various medical and scientific applications such as Cyber-Intrusion Detection, Fraud Detection, Medical Anomaly Detection, Fault Detection, Industrial Damage Detection, Image Processing, Textual Anomaly Detection, Sensor Networks etc. [50-57]. With the advancement in sensor technologies, sensor networks are an important area of research. There is a need for data analytics in sensor networks as the data collected from various wireless sensors has several unique characteristics. Outlier detection is a crucial step in data analytics to identify unusual events in different application domains. Various techniques for outlier detection in wireless sensor networks are proposed in the literature [5]. An outlier in sensor networks is either due to faulty sensors or due to unique or abnormal events that are important for analysis. Hence, outlier detection in wireless sensors can detect sensor fault detection or intrusion detection or both. Outlier detection in sensor networks poses a set of unique challenges. The outlier detection techniques need to be light because of sensor resource constraints. Another challenge is that in wireless sensor networks a distributed approach for analysis is required as data collection is done

in a distributed fashion [58]. Additionally, the noise in the data collected from the sensor makes outlier detection more challenging. In this study, the main concern is on wireless body sensor networks. As caretaker may not be always available to monitor physiological parameters so, different sensors are attached with the body of patient to monitor values of these parameters. Various medical sensors available nowadays are MICAZ, TelosB, IRIS etc. [59-61].

1.4 Outlier Detection in Medical Sensor Data

Outlier detection in the medical and public health domains typically work with patient records. The data can have outliers due to several reasons such as abnormal patient condition, instrumentation errors or recording errors. Several techniques have also focused on detecting disease outbreaks in a specific area [62]. Thus the outlier detection is considered to be very critical problem in this domain and requires high degree of accuracy. The data typically consists of records which may have several different types of features such as patient age, blood group, weight etc. The data might also have temporal as well as spatial aspect to it. Most of the current outlier detection techniques in this domain aim at detecting anomalous records (point anomalies). Typically, the labeled data belongs to healthy patients; hence most of the techniques adopt semi-supervised approach. Another form of data handled by outlier detection techniques in this domain is time series data, such as Electrocardiograms (ECG) and Electroencephalograms (EEG). Collective anomaly detection techniques have been applied to detect anomalies in case of such data [63]. The most challenging aspect of the outlier detection problem in this domain is that the cost of classifying an anomaly as normal can be very high.

1.5 Challenges of Outlier Detection in Big Sensor Networks

- Most of the existing techniques though are efficient but do not provide scalability that is they do not fit in the scenario of big data. Also the existing work in the field of outlier detection is based on the shared memory model which restricts their ability to manage very large volume of data.

- Existing techniques for outlier analysis in high dimensional datasets process all the dimensions for finding outliers but many of the features have no significant impact on detection of outlier. Therefore, in high dimensional data, analysis can be performed more efficiently using techniques of data reduction.

- Data instances are temporary in a data stream that is considered for a particular period of time but most of outlier detection techniques compare this data with all the previous data sets stored earlier. Because data in data streams are continuous therefore, it is not possible to store all the data.

- Most of the existing techniques detect a specific type of outlier, a point outlier or contextual outlier, thus there is need for a unified framework that can detect all kind of outliers.

- Existing methods for outlier detection are based on specific application domains. There is no such method that can detect all types of outliers in various domains. So, a generalized technique is required that can be applied to different domains.

- The existing methods for discovering outliers using machine learning are mostly focused on attributes or entities but, ignore relationships among them.

In the recent years, with the unprecedented growth of digital data in various domains, there is a surge for designing scalable, consistent and precise algorithms. Thus, distributed solutions which can deal with large amount of data with accuracy are in high demand.

1.6 Research Contribution

The main emphasis of the research in this thesis is designing and development of efficient algorithms for outlier detection. The key contributions of the research are as follows:

- A detailed review of the outlier detection approaches is provided. Special emphasis is given on outlier detection approaches in wireless sensor networks.
- The issue of outlier detection is discussed in detail and various proposed solutions of the problem are enlisted.
- Scalable outlier detection approaches for large datasets using different methods are proposed. The proposed approaches are:
    1. A scalable and efficient approach for outlier detection for finding outliers in linearly correlated datasets is proposed. The proposed approach uses Sequential Minimal Optimization Regression for finding point outliers. After finding point outliers, the contextual outliers are found using linear correlation among attributes. The proposed approach performs better than similar existing approach for outlier detection in terms of classification accuracy and time efficiency.
    2. The above proposed approach is further enhanced for detecting outliers in non-linearly correlated datasets as well.
    3. A parallel fuzzy c-means clustering algorithm based on clonal selection principle is proposed.

4. An integrated framework for detection of outliers using the proposed clonal selection based fuzzy c means algorithm is proposed. In the proposed framework, after clustering, clusters are further refined using the cluster refinement algorithm.

1.7 Thesis Organization

The organization of thesis is as follows:

Chapter 1: Introduction

This chapter introduces the concept of data mining in large datasets, describing it as a computational problem. It discusses various data mining techniques based on different machine learning models. Then, it moves on to describe the significance of outlier detection in data mining, laying a foundation for the introduction of the concept of outlier detection in big data. The problem of outlier detection is stated, and various solutions to the problem are also enlisted. This chapter also discusses the challenges faced while dealing with outlier detection in large datasets, and in distributed environment. In a nutshell, the chapter explains the basic concepts of data mining, and how it coincides with the concept of outlier detection. This chapter partially addresses Objective 1.

Chapter 2: Background and Related Work

This chapter contains a detailed review of the literature on data mining and outlier detection, with special emphasis on outlier detection in wireless sensor networks. It discusses use of various classification and clustering based approaches for outlier detection. There is a list of case studies, which show that finding correlation between data objects is the best way to deal with the problem of detecting anomalous events in ever increasing data. A segment on related research discusses the existing surveys on this topic. A significant portion of this chapter consists of an

extensive list of outlier detection for big data mining. In-depth discussion on outlier detection in wireless sensor networks is also presented. This chapter touches other approaches that are being used frequently by researchers and developers while finding outliers in big data. Finally, this chapter contains a few comparative studies highlighting the advantages and drawbacks of the discussed approaches. It accomplishes Objective 1.

Chapter 3: Problem Formulation

This chapter lists the research gaps that are identified initially, while proposing the research objectives. These gaps identify the issues that exist in the existing outlier detection approaches. Also, there is need for developing approaches for outlier detection in large datasets that cannot fit in the memory of single system. These approaches will be useful for developing applications that needs scalability. Based on the research gaps, the four research objectives are formulated which are addressed in different chapters of this thesis. This chapter partly addresses the Objective 2.

Chapter 4: Classification based Outlier Detection Approaches

In this chapter, two approaches for outlier detection are proposed. The first approach is based on SMO regression and linear correlation among attributes of the data which is implemented by amending the proposed approach in the distributed environment. The Hadoop based map-reduce model is used to design the proposed approach in the distributed setting. The second approach is proposed to overcome the issues present in the previous approach. The second approach handles outliers in both linearly and non-linearly related attributes. The approach is based on non-linear kernel function and detects non-linearly related attributes using continuous analysis of variance. This chapter also includes the details of the experiments performed on real and synthetic data sets, to validate the proposed approaches. The first phase of experimentation is testing the

performance of proposed classification based algorithms and comparing it with the existing classification based approaches. Standard parameters are used to evaluate the performance of the proposed approach. The next phase is to check the scalability of the proposed approach using distributed computing platform. This chapter partially accomplishes Objective 2, 3 and 4.

Chapter 5: Clustering based Framework for Outlier Detection

In this chapter, a clonal selection principle based parallel fuzzy clustering algorithm for data clustering is proposed. The proposed algorithm works on the principle of clonal selection algorithm and uses objective function of fuzzy clustering. The algorithm is implemented using distributed map reduce framework. An integrated framework for outlier detection is proposed based on the suggested algorithm. The working of the proposed framework is discussed in this chapter. The experiments are performed on real and synthetic data sets, to validate the proposed approaches. The first phase of experimentation is performed on testing the proposed clustering algorithm and comparing it with the existing state-of-art clustering algorithms. Standard parameters are used to evaluate the performance of the proposed algorithm. The next phase is to check the scalability and performance of the proposed framework using distributed computing platform. It accomplishes Objective 2, 3 and 4.

Chapter 6: Conclusion and Future Directions

This chapter concludes the thesis by providing a brief overview of the proposed approaches and framework for outlier detection. The insights about the future scope of work are also provided in the chapter. Sensor networks are an evolving field, and this chapter brings to light the huge amount of scope for developing outlier detection approaches and applications in this domain. It

highlights the contributions made by our work and enlists the points that need to be worked upon in the future.

1.8 Chapter Summary

In this chapter the meaning of outlier detection and what is the need of detecting outliers in a dataset is discussed in detail. Further, various types of techniques on the basis of which one can classify an outlier detection technique are explained. The light on the significance of outlier detection in remote health monitoring and various issues that occurs during remote monitoring is provided. A brief introduction of the approaches proposed in this research is also given. Finally the organization of thesis is discussed. In the next chapter various existing clustering and classification based approaches for outlier detection will be discussed in detail.

CHAPTER 2

BACKGROUND AND RELATED WORK

The aim of this chapter is three fold: initially, the metrics for outlier detection are discussed. Then, a detailed review of the related existing research works on outlier detection is presented. The literature review includes works specifically related to classification based and clustering based approaches for outlier detection. Finally various applications of outlier detection in different application domains are discussed. Section 2.1 discusses various outlier detection metrics. Section 2.2 discusses the literature review which is further divided into two subsections: Subsection 2.2.1 contains the review of existing classification based approaches for detecting outliers and subsection 2.2.2 gives the review of existing clustering based approaches for detecting outliers. Section 2.3 mentions various applications of outlier detection in wireless sensor networks.

2.1 Outlier Detection Metrics

Outlier detection is a process of identifying data instances that deviate drastically from the given dataset. Outlier detection approaches can be broadly classified by three characteristics: type of input data, presence of labeled data (anomalous vs. normal), and application specific constraints [64]. The main concern in an outlier detection approach based on input data is the number of features available and the type of values that a specific feature contains. The input data can be

univariate or multivariate depending on the number of features available. The values present in the features can also be categorized into three types that are categorical, continuous, and binary.

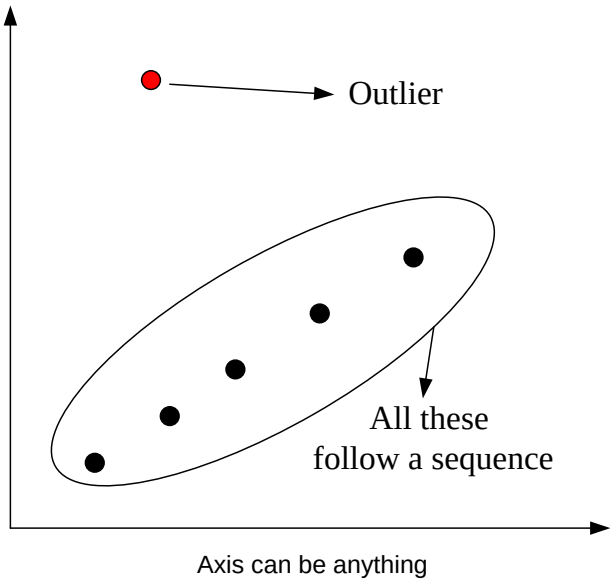


Figure 2.1: Outlier identification

In Figure 2.1, a data point is identified as outlier because it does not follow the pattern followed by other data points in dataset. An additional concern in input data is that there exist relationships within the features of data itself. Some applications assume the point outlier scenario that is they presume that no relationships are present between the features. Other applications assume that relationships may be present; these types of outliers are called contextual outliers. Based on the availability or non-availability of data labels the algorithms for outlier detection is categorized into supervised learning algorithm or unsupervised learning algorithm.

If there is apriori knowledge of data labels then the algorithm is referred as supervised learning algorithm. If the labeled data for normal and anomalous records are not available then the

algorithm is referred as unsupervised learning algorithm. Further, if labeled data of only normal instances are available, then the algorithm is categorized as semi-supervised learning algorithm [65].

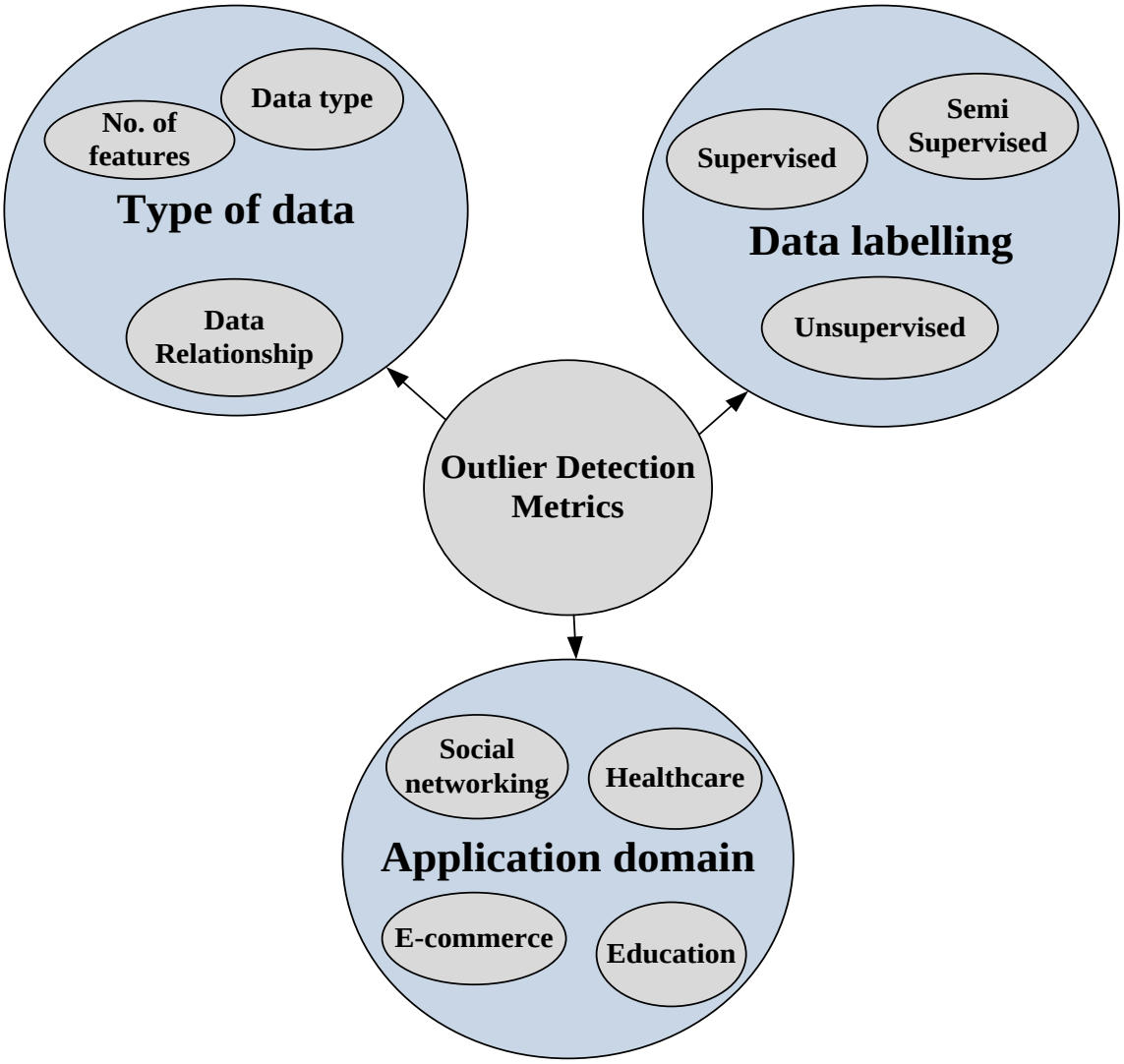


Figure 2.2: Outlier detection metrics

These types of algorithms presume that the occurrence of outliers in a dataset is very rare so, there is no labeled data available for model training. Also, the anomalous activities are dynamic that is it is very challenging to detect all types of anomalous events in a dataset [66]. The metrics

also depends on the type of application in which outlier needs to be detected. As in some applications only point outliers need to be detected however, in some other applications consideration of context between features plays a significant role. The different metrics on which any outlier detection approach is dependent is shown in the Figure 2.2.

2.1.1 Performance Metrics

This chapter also introduces concepts related to some metrics used in the contribution and evaluation chapters: Chapter 4 and Chapter 5 respectively. Firstly, concepts related to sensitivity and specificity [67] is given. These are the metrics used within classification tests to determine how well a classifier performs in classifying the correct versus incorrect errors. Sensitivity of an approach is referred as true positive rate or detection rate which is defined as:

$$DR = \frac{TP}{TP + FN}$$

where, *TP* is true positives that are the count of instances which are correctly identified as anomalous by the approach, *FN* is false negatives that are the count of instances which are misclassified as non-anomalous by the approach.

Specificity refers to true negative rate which calculates the total no. of negatives which are correctly identified. Specificity is calculated using false positive rate (FPR), which is defined as:

$$FPR = \frac{FP}{FP + TN}$$

where, FP is false positives that are the count of instances which are misclassified as anomalous, TN is true negatives that are the count of instances which are correctly classified as non-

anomalous. TP, FP, TN, FN are shown in the Table 2.1. The numeric value 1 in the table represents anomalous instance and 0 represents instance non-anomalous.

Table 2.1: Confusion Matrix

	TP	**FP**	**TN**	**FN**
Actual	1	0	0	1
Output	1	1	0	0

2.2 Literature Review

With the advancement in sensor technologies, there is the need for data analytics in various application domains such as communication networks, healthcare monitoring, weather monitoring, intelligent transport systems etc. [68, 69]. Outlier detection is a subpart of data mining [70] and is vital for effective utilization of these applications. Various techniques for outlier detection in wireless sensor networks are proposed in the literature [5, 71]. In this study, the main concern is on wireless body sensor networks. There is a need of healthcare monitoring using wireless sensors for intensive care unit patients, as the doctor may not be present twenty four hours in a day for monitoring of physiological parameters. Hence, different sensors are attached with the body of patient to monitor the values of these parameters. Various medical sensors available nowadays are MICAZ, TelosB, IRIS etc. [59-61].

Outliers in wireless body sensor networks can be broadly classified into two categories. The first category involves hardware failures caused by faulty sensor nodes or due to loss of connection between sensor and body. The second category contains the anomalous data due to actual medical emergency [72]. An Outlier caused by sensor fault generates false alarms due to which

there is wastage of human resources in healthcare centers. So, there is a need of techniques that can differentiate this type of anomalies from the actual medical emergency condition.

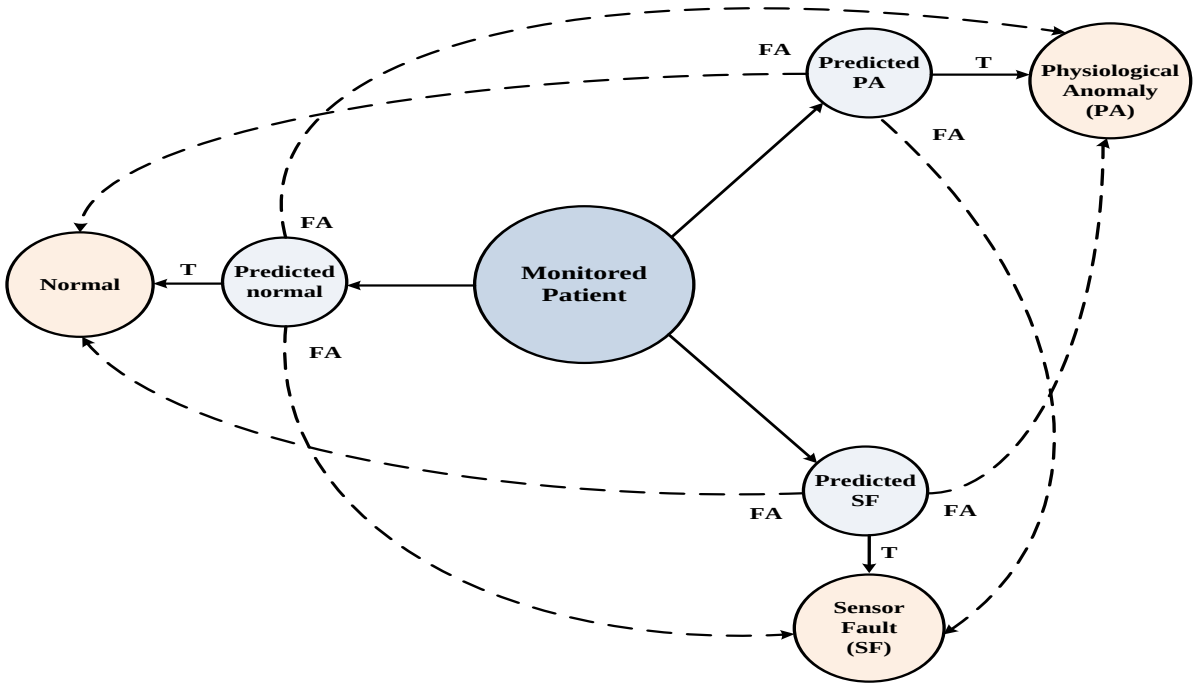


Figure 2.3: Different states of the patient in healthcare monitoring using WBSNs

Figure 2.3 shows various possible states of a patient in a healthcare monitoring application. Here, T signifies that the application identifies true state of the patient and FA denotes a false alarm showing that the predicted state is not actual state of the patient and connected to the probable true state with the dotted line.

The outliers in any domain are broadly classified into two categories that are point outliers and contextual outliers. If a single record is considered to be abnormal with respect to the other records in the dataset then that record is considered as point outlier. A record is anomalous within a specific context. For example, a sensor reading may only be considered anomalous

when evaluated in the context of temporal and spatial information otherwise it is non-anomalous. The case of point outlier is shown in Figure 2.1 and case of contextual outlier is shown in Figure 2.4. As our domain of study is body area wireless sensor networks, hence in this research work, we referred point outliers as sensor faults and used contextual outliers for detecting true medical condition.

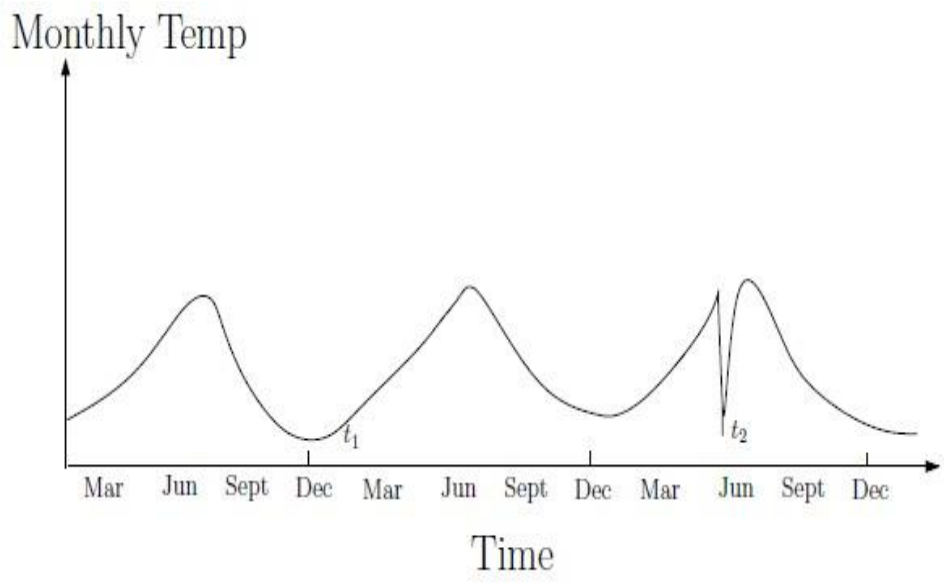


Figure 2.4: Outlier in temperature depiction

In sensor networks, point outliers are the points in which anomaly is present in a sensor node when considered independently. However, in real applications various sensor nodes in a network are related to each other so there is need of contextual outlier detection. Contextual outlier exists in correlated datasets where multiple sensor nodes are related to each other and the observations of one sensor also affect the readings of another sensor.

In this chapter, classification and clustering based approaches for outlier detection are discussed. The classification based approaches are discussed in Section 2.2.1 and clustering based approaches are surveyed in Section 2.2.2.

2.2.1 Supervised Learning for Outlier Detection

In this section, various Classification based outlier detection techniques are given. In particular, the pros and cons of each approach are discussed. Further, this section serves as a basis for why the algorithms for the contribution are selected in Chapter 4. There are some challenges in existing approaches for outlier detection in sensor networks. Therefore, the proposed approaches in this research worked on some of the short comings of existing approaches.

In supervised learning there is a predefined set of classes. The incoming data is classified using the labels assigned to different set of classes. The aim of supervised learning approaches is to train a model based on already labeled data that will help in identification of labels of novel data objects [73].

As we previously discussed outlier detection algorithms can be broadly classified into point outlier detection and context-aware outlier detection. Contextual anomalies are found in datasets where multiple attributes are related to each other based on some context. For example, medical sensors generally indicate the sensor reading of various physiological parameters. However, if different sensors are attached to the same patient, variation in the value of one sensor leads to change in values of other sensor also. Many existing anomaly detection algorithms in the sensor field target on using the time series information of the sensor reading to predict the next possible value and then compare this value to the actual reading.

Hill et al. [74] proposed an approach for outlier detection in environmental sensor data. In their work, they predict sensors value based on sliding window from historical data and compare them to actual sensed value to detect sensor faults. But, this data-driven approach detects only point anomalies, as no focus is given to correlation among sensor nodes. Mahalanobis distance–based

approach for outlier detection is proposed by Liu et al. [75] which considers correlation among different sensor nodes. This method uses sliding window concept for dynamic threshold calculation. The problem with this approach is that it assumes that neighboring sensor nodes always collect the same type of data that might not be possible in case of medical sensor data. Srivastava et al. [76] proposed an approach that takes into account both functional and contextual aspects of the data for anomaly detection. However, their approach requires dimensionality reduction to flatten data, which compromises accuracy and is not scalable to big sensor data. For anomaly detection in time series data, Yao et al. [77] proposed an algorithm using piecewise linear models. However, correlation among attributes is not considered in this approach, which leads to high false positive rate.

Salem et al. [53, 78, 79] developed 3 methods for anomaly detection in WSNs. In the first approach, abnormal instances are detected using linear SVM (support vector machine) model. The challenge with this method is that it is based on fixed threshold and also it uses linear regression, [80] which is not an efficient method for prediction. In the second approach, Salem et al. used linear regression and decision tree [81] J48 for detection of anomalies. The approach takes into account the correlation among attributes, but the window used in this approach is static. By combining Mahalanobis distance and kernel density estimator, Salem et al. proposed another algorithm for detecting outliers. However, predefined threshold still remains a challenge. Shilton et al. [82] proposed a classification based approach for outlier detection in WSNs. The issue with this approach is setting values of parameters, as with increase in parameter value false negatives also increases. Most of the approaches previously discussed are not scalable to big data due to lack of distributed framework.

An approach for anomaly detection in big sensor data has been proposed by Hayes et al. [83]. In this approach, contextual anomaly detection is used over point anomaly detection. Also the approach is scalable to big data requirements. The challenge with this approach is that it considers only one type of data that is tall datasets. Another technique for anomaly detection has been proposed by Haque et al. [84] to detect false alarms in medical sensor networks. The challenge with this approach is that it gives equal importance to all the sensors and thus it lacks weighted correlation among various sensors, which is not the case in real scenario. The above reviewed approaches considered two existing issues in existing approaches that are scalability to large amount of data and context aware outlier detection. Hence considering these two issues an approach based on linear correlation and using distributed framework is proposed in Section 4.2 of Chapter 4.

However, apart from linear, various other types of correlations also exist in real world. Thus, various approaches for anomaly detection [85, 86] are also studied by considering the type of correlation that exists among attributes. Lee et al. [87] proposed an approach by leveraging the capabilities of Hadoop, which is an open-source distributed framework to detect network anomalies. As network data are very large, so this approach typically addresses concerns related to outlier detection in big data. Recent work on data analytics has shown that outlier detection is an essential part of data driven and prediction based techniques for wireless sensor networks [88, 89]. The effectiveness of a technique depends on that whether the outlier detection approach accurately identifies the data that do not validate the estimated system behavior. The effectiveness of an outlier detection technique depends on various factors like data size, dimensionality of a dataset, statistical characteristics, machine learning technique, etc. For

instance, auto-regression is an efficient technique for linearly related datasets whereas for periodically repeated features cross correlation gives better results [5, 91].

Zhong et al. [92] proposed a correlation based approach for anomaly detection. In their approach, correlation coefficient among different sensor nodes are analyzed and converted into a vector. Then based on that correlation vector and the probabilistic model anomalous data has been identified. The challenge with the approach is its dimensionality problem when the number of sensors is large, thus is not scalable to big data. Saneja et al. [93] formalized an outlier detection method for WBSNs which handles both point and contextual outliers. The point anomalous sensors are detected using prediction based detection algorithm. The linear correlation among sensor nodes are analyzed and based on that contextual outliers are detected among point outliers. The approach is scalable to big data however in the approach only linearly related attributes are taken into consideration.

A distance based detection approach for outliers is proposed by Liu et al. [74]. Although the correlation among sensors is considered in the approach but, the hypothesis for the approach is that the neighboring sensors collect the same type of data which might not be the case in healthcare monitoring applications. The approach is also not scalable to big data. A piecewise linear model based approach for anomaly detection is proposed by Yao et al. [76]. The proposed approach has high false alarm rate as correlation among various attributes of the data is not taken into consideration. Salem et al. [78] discussed a distance based detection technique for outliers. The correlation between attributes is taken into account in their work however; the predefined threshold still remains the challenge. An approach for contextual anomaly detection in sensor data is proposed by Hayes et al. [83].

The approach is scalable to big data however it is only applicable to large datasets. Zhang et al. suggested a fault diagnosis method based on Bayesian network model [94]. The proposed model considers both spatial and temporal correlation in wireless body sensor networks. The model is not scalable to big data however by considering correlation there is a significant reduction in the number of errors. Various other methods have also been proposed for outlier detection in body sensor networks using spatial and temporal correlation. Kim et al. formalized an anomaly detection method for motion fault detection using nine motion sensors [95]. In their work, they proposed various history and non-history based fault detection strategies.

A linear regression based method for outlier detection is proposed by Salem et al. [53]. The spatial correlation among various attributes is used for prediction of next value. The challenges with the approach are that it considers only linear relationship among attributes and also it practices linear regression which is not an efficient method for prediction. In literature various other regression methods for prediction has been applied in different application domains and it is observed that performance of SMOReg (Sequential Minimal Optimization based Regression) is better than various other regression methods [96, 97].

In [98] Salem et al. proposed another approach for fault detection in Wireless Sensor Networks (WSNs). The proposed method predicts values of the current window of a signal with the help of a non-seasonal forecasting algorithm. The algorithm is based on exponential function which follows a linear trend. Naseem et al. [99] detects anomalous data using Gaussian mixture model. In their approach for detection of classification rules ant colony algorithm is used. Further, in order to detect true medical condition and sensor fault spatial correlation among data is considered. However, the approach is not scalable to large datasets.

From the above literature survey, it can be summarized that the challenges in the existing techniques can be attributed to following grounds. A key assumption in existing approaches is that the various attributes in the dataset are linearly related however in most of the real application scenarios the features extracted are far more complex and might be non-linearly correlated. Also, the monitored sensor data is temporal however various existing techniques do not take into consideration the temporal aspect of data hence may not be able to detect trend anomalies. Lastly, nowadays as sensors are used in various applications thus the data is growing tremendously but the existing techniques are also not scalable to big data. To address the above issues we therefore proposed a scalable approach for outlier detection in Section 4.4 of Chapter 4 that considers both linear and non-linear aspects of correlation among attributes.

In summary, though a lot of research has been done in the area of wireless body sensor networks. But, there is a need to increase efficiency, reliability and performance in various healthcare monitoring applications [100]. A brief comparison of existing classification based techniques of outlier detection based on various parameters is illustrated in Table 2.1.

2.2.2 Unsupervised Learning for Outlier Detection

Unsupervised learning is in contrast to supervised learning. In case of unsupervised learning there is no predefined set of class labels. Therefore, the objective in unsupervised learning is to decide which objects should be grouped together. Hence, the class labels are formed by learner itself. Of course, the success of classification learning is heavily dependent on the quality of the data provided for training. If the data is inadequate or irrelevant then the concept descriptions will reflect this and misclassification will result when they are applied to new data.

Table 2.2: Classification based approaches for outlier detection grounded on various parameters

Authors	Approach	Dataset	Outlier Type	Threshold selection	Programming Model	Correlation
Hill et al. [73]	Auto regression	Environmen-tal sensors	Point	Static	Centralized	No
Liu et al. [74]	Mahalanobis distance	Simulated	Point	Static	Centralized	Spatial
Srivastava et al. [75]	Univariate + Multivariate stastical hypothesis	Electrical power systems	Point Contextual	Static	Centralized	Inverse
Yao et al. [76]	SSA + Rule based	Environ-mental sensors	Point Contextual	Static	Distributed	Linear
Salem et al. [78]	Linear SVM	Medical wireless sensors	Point	Static	Centralized	No
Salem et al. [79]	Mahalanobis Distance + Kernel density estimator	Medical wireless sensors	Point Contextual	Static	Centralized	Majority voting
Hayes et al. [83]	Univariate and multivariate guassian predictor	Electricity sensors	Point Contextual	Static	Distributed	No
Haque et al. [84]	SMO Regression	Medical wireless sensors	Point contextual	Dynamic	Centralized	Majority voting
Zhong et al. [88]	LCAD	Simulated	Point	Static	Centralized	Latent
Naseem et al. [99]	Gaussian decomposition + Ant colony classifier	Real data	Point Contextual	-	Centralized	Spatial

In this section, various existing clustering based techniques will be discussed. In particular, the pros and cons of each approach will be discussed. Further, this section will serve as a basis for why the algorithm for the contribution is selected in Chapter 5. There are some challenges in existing approaches for outlier detection in sensor networks. Therefore, the proposed approaches worked on some of the short comings of existing approaches. The clustering techniques are broadly classified into two types that are hard/crisp clustering and soft/fuzzy clustering.

a) Hard Clustering

Clustering algorithms which group data into disjoint clusters are known as hard clustering algorithms. Most of the existing research focuses on finding disjoint clusters from the data. The most popular hard clustering algorithm is K-Means. Many modifications have been done on K-means to improve its performance [101-103]. The distributed improved version of K-Means was also introduced in literature [104]. Some other hard clustering algorithms were also proposed recently for graph datasets [105, 106]. But most of them lack consistency in results that is they produce different results each time the algorithm is executed. Also, in real life scenario most of the data in a dataset does not belong to a cluster completely. So there is need of fuzzy clustering approaches in which data partially belongs to a cluster based on its membership value.

b) Fuzzy Clustering

Fuzzy clustering may result in overlapping clusters, such that one data point can lie in multiple clusters. Fuzzy C-means (FCM) clustering is most popularly used algorithm that can find overlapping clusters. Dunn [107] developed this algorithm and further Bezdek [108] improved it by adding some more features. Random initialization of cluster center is one of the major

limitations of traditional FCM. Many researchers [109-110] have tried to remove this limitation of FCM by modifying objective function.

Several machine learning approaches were considered to improve the performance of FCM in several applications such as time-series data forecasting, Information retrieval [111-112] etc. Vertex similarity [113] and Euclidean distance [114] measures are also used in literature for fuzzy clustering. Apart from these aforementioned algorithms several other clustering algorithms were also proposed in literature [115, 116].

The concept of fuzziness is exploited in various other problem domains also, but the proposed approaches works on centralized systems [117-118]. However with the introduction of big data, there is requirement of distributed and parallel fuzzy clustering. Very few techniques have been proposed in literature for distributed clustering. An incremental multiple medoids based fuzzy clustering was proposed in literature for managing complex network data that is not well parted [119]. Ludwig [120] proposed the distributed clustering technique using map-reduce framework. Parallel Fuzzy Minimal (PFM), parallel fuzzy minimal clustering algorithm [121] are two of the parallel clustering algorithms.

Till date, various methods have been studied in the literature for detection of outliers based on data clustering [5]. Hence, to understand the literature of outlier detection some of the relevant techniques have been reviewed in this section. A network data mining technique for anomaly detection has been proposed in [122] based on k-means clustering for classification of anomalous and non-anomalous traffic. Kiss et al. [123] suggested a technique for detection of anomalies in industrial systems using a clustering approach which is the combination of k-means and subtractive clustering. An immune system mechanism that is negative selection algorithm based

anomaly detection was proposed by Dasgupta and Forest in [124]. In this approach, data was categorized as self and non-self where self refers to non-anomalous data and anomalous patterns was referred as non-self. Izakian and pedrycz proposed a unified algorithm in [125] for detecting amplitude and shape anomalies. The detection of anomalous and non-anomalous data was done using fuzzy c-means clustering. To detect amplitude anomalies actual time series data was taken whereas for detection of anomalies in shape the autocorrelation representation of data was considered. A support vector machine (SVM) and s-transform based anomaly detection algorithm was suggested by bhargava and raghuvanshi in [126]. S-transform was used for data reduction whereas SVM was used for classification of data. Guo et al. [127] proposed an algorithm for anomaly detection based on data reduction and improved k-means clustering. Piecewise aggregate approximation (PAA) was employed for data reduction. Then, on compressed data classification of anomalies was performed by means of improved k-means clustering. Zhang et al. [118] proposed a pruning approach based clustering algorithm to improve the robustness of hard, fuzzy and deterministic annealing c-means clustering. The approach is based on the centralized framework. A fuzzy inference based anomaly detection approach was proposed by Hoang et al. [128]. The proposed scheme was integration of ordinary database detection engine and Hidden Markov Model.

An online network anomaly detection method was proposed by Su in [129]. The method combines the genetic weighted KNN classifier and unsupervised clustering algorithm for anomaly detection. As temporal data is growing tremendously in the last decade, most of the existing methods do not take into consideration the issue of data overhead. Also, most of the above discussed algorithms are based on hard clustering techniques. For summarizing the literature survey, comparative analysis of various techniques based on different parameters is

demonstrated in Table 2.3. To resolve the problem of data overhead while considering fuzzy nature of data and ensuring high detection accuracy, an outlier detection framework has been proposed in this thesis based on data reduction and fuzzy clustering using distributed map reduce framework. The detailied explanation of the proposed algorithm is given in Chapter 5.

Table 2.3: Clustering based outlier detection approaches based on different parameters

Authors	**Approach**	**Dataset**	**Outlier Type**	**Handles Overlap**	**Programming model**	**Cluster Refinement**	**Clusters**
Kiss et al. [123]	K-means	Gas compressor stations	Point	No	Distributed	No	Crisp
Izakian et al. [125]	FCM	Weather sensor data	Point Contextual	Yes	Centralized	No	Fuzzy
Guo et al. [127]	K-means	Synthetic and real	Point	No	Centralized	No	Crisp
Munz et al. [122]	K-means	Traffic data	Point	No	Centralized	No	Crisp
Rajasegarar et al. [82]	Fixed width	Real data	Point	No	Distributed	No	Crisp
Moshtaghi et al. [90]	Hierarchical	IBRL	Point	No	Distributed	No	Crisp

2.3 Applications of WSNs

With the innovation of sensors and their incorporation into WSNs has boosted the research in the field of various key research domains such as healthcare, agriculture, military, etc. These sensors have ability to capture environmental conditions such as wind speed, wind direction, temperature, pressure, speed, light, noise, stress etc. This data helps in developing various monitoring, security & intelligence, health, climate, weather related applications. Figure 2.5 shows the areas in which WSN can help to develop research applications [130]. Some of the applications of WSNs are explained below:

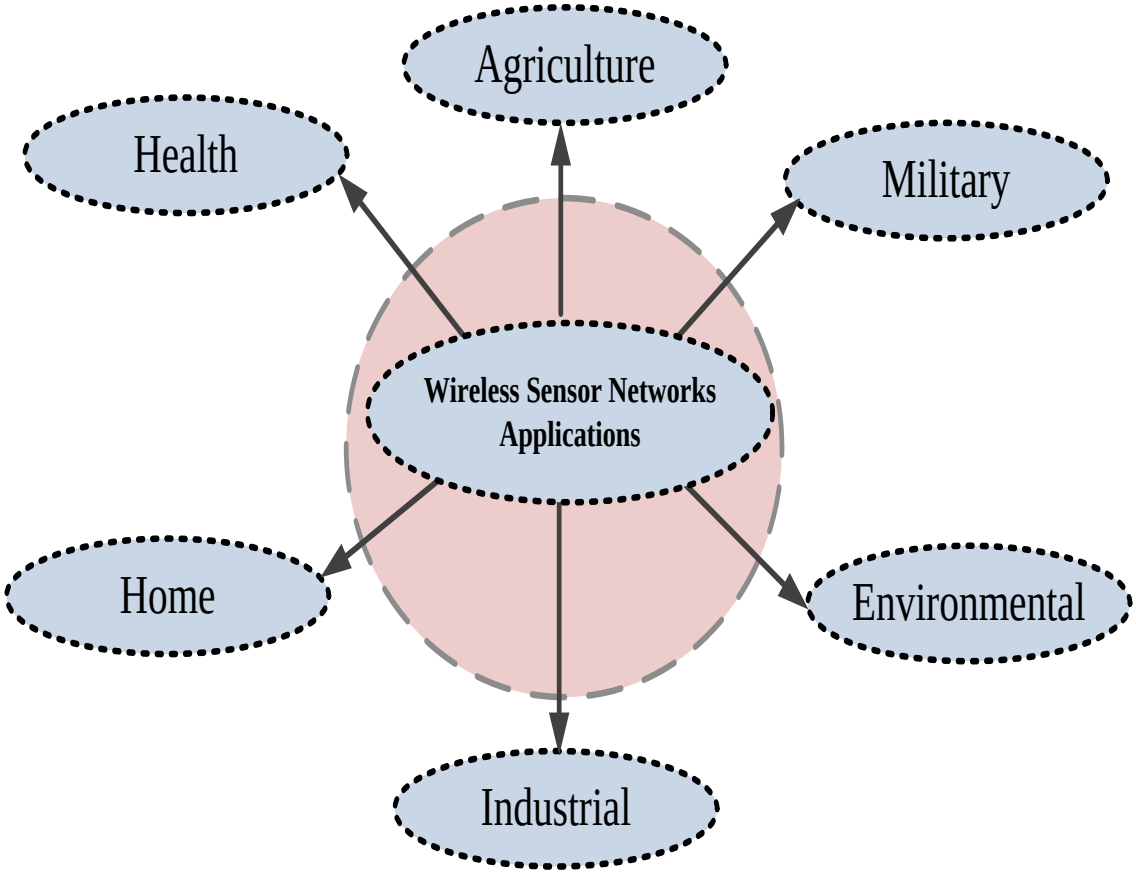


Fig 2.5: Applications of Wireless Sensor Networks

2.3.1 Military Applications [131]

WSNs are widely used in military monitoring, computing, surveillance, intelligence, investigation and targeting systems. These applications help in remote surveillance in military operations and target the enemy remotely. WSNs can reach into those areas where human access and reachability is not possible.

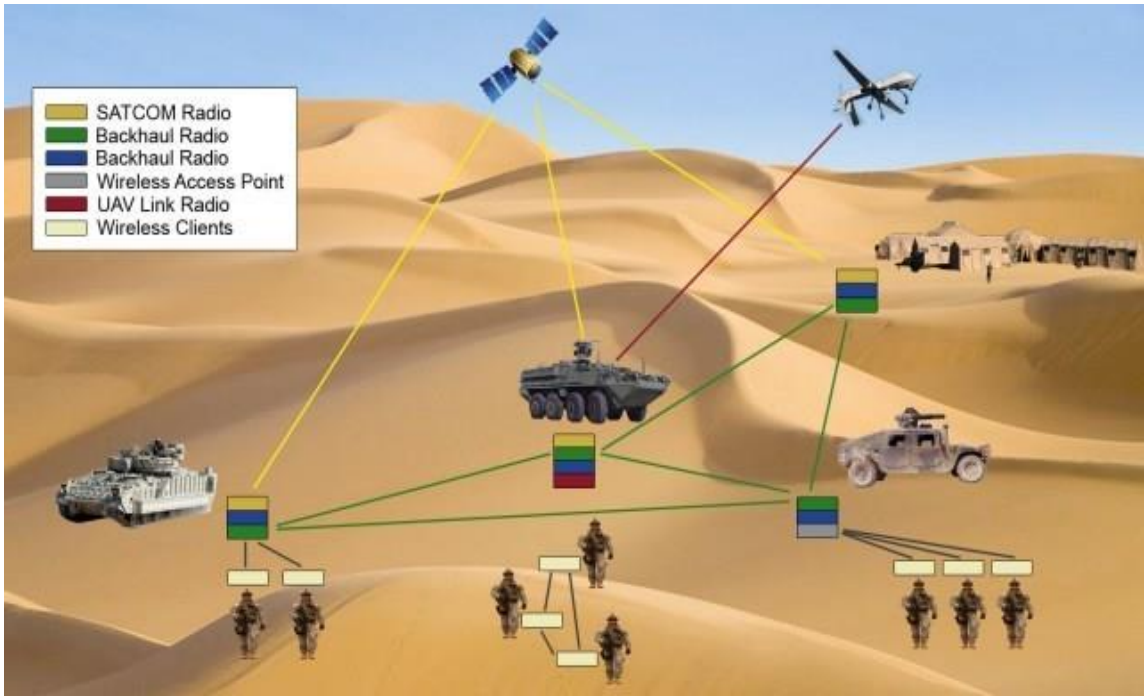


Figure 2.6: Sniper Detection System

- **Smart Dust [132]:** It is a WSNs application system that works in hostile environments where human access is dangerous. The small cubic millimeter sized sensors have been used to monitor, detect and track activities.
- **Sniper Detection System [133]:** It is used by military and law-enforcement agencies. The working is shown in Figure 2.6 [105].
- **VigilNet [134]:** It is WSN based target tracking surveillance network.

2.3.2 Environmental Applications [135, 136]

WSNs are also widely used in environmental monitoring, tracking, detection, and environmental research applications. Habitat selection of seabirds in Maine is also observed under different environmental conditions and forecasting of temperature, salinity, velocity and depth of water along with wind is done using environmental observation and forecasting system (EOFS) known as CORIE

- The **Great Duck Island** is a WSN project which aim to measure the residency of nesting burrows
- **Flood recognition and forecasting system:** It was developed by MIT and was tested in Honduras.
- **ZebraNet system:** It was deployed in Kenya for the tracking of two species of thirteen zebras.
- **Volcano observation:** WSN applications were used to observe the volcanos of "Volcán Tangurahua" in Ecuador in 2004-2005. So that timely action can be taken in case of any anomalous scenario.

2.3.3 Health Monitoring [137]

WSN applications have also provided benefits to health industry. Some of the applications help in monitoring, diagnosis, administration, tracking of different activities in hospitals and elsewhere. Some of the applications of WSN in health industry are following:

- Artificial Retina project benefits the visually impaired people by curing retinitis pigmentosa (RP) and age-related muscular degeneration (AMD).

- **Code Blue project** helps to monitor heart and muscles activity, pulse rate, blood oxygen. 14 different body sensors are used to measure these observations. This information can be submitted to doctor online and remote prescriptions can be asked.

2.3.4 Home Applications [138]

WSNs can be integrated in the devices used at home such as Air-Conditioners, refrigerators, ovens, Geysers, TVs etc. These devices can not only be monitored but can also be controlled by WSNs through remote places. One of the examples is the Nonintrusive Autonomous Water Monitoring System (NAWMS). It is detection and monitoring system for the water usage at homes.

2.3.5 Industrial Applications [139]

WSNs are also used in industrial automation and civil structure monitoring. Earlier wired monitoring was performed which had several drawbacks such as non-portability, wear and tear. So WSN helps in efficient, cost effective and accurate diagnosis, tracking of construction progress. Data is collected and analyzed to figure out the current situation. Following are the two applications of WSN:

- **Industrial Automation [140]:** Wireless sensors are replacing lead wires and helps in developing cost effective systems such as observing the strength for wind turbines, health care and location based services, environmental monitoring, and measure gaps in rubber seals.
- **Civil Structure Monitoring [141]:** Ben Franklin Bridge is deployed using wireless sensors over the bridge. Strain of the bridge is measured using these wireless sensors when a train

crosses the bridge. When the train reaches on the bridge, the strain waveform is recorded using sensor nodes.

2.4 Chapter Summary

In this chapter an overview of different outlier detection metrics is given. Also the detailed survey of various outlier detection techniques based on supervised and unsupervised machine learning models are provided. Finally, various application areas of wireless sensor networks where there is need of outlier detection are discussed. In the next chapter based on the literature survey of this chapter various research gaps in existing techniques will be identified. Further based on the research gaps various research objectives will be formulated.

CHAPTER 3

PROBLEM FORMULATION

This chapter lists the research gaps, which are identified initially while proposing the research objectives. These gaps recognize the issues with the existing outlier detection approaches. They identified the fact that there is a need to develop approaches for detection of anomalous events that consider the relationships among data attributes and detect outliers from datasets. Also, there is a need to work on developing approaches that detect outliers from big datasets, that is the datasets that are not able to fit in the memory space available in a single system, and applying the approaches for evolving applications that require scalability. Based on the research gaps discussed in this chapter, the four research objectives are formulated which are addressed in different chapters of the thesis.

3.1 Research Gaps

After the broad literature survey, the research gaps identified are as follows:

- Most of the existing techniques though efficient do not provide scalability that is they do not fit in the scenario of big data. Also, the existing work in the field of outlier detection is based on the shared memory model which restricts their ability to manage very large volume of data.
- Existing techniques for identification of outliers in high dimensional datasets process all the dimensions for finding outliers. However, many of the features have no significant impact on

detection of outlier. Therefore, in high dimensional data, analysis can be performed more efficiently using techniques of data reduction.

- Data instances are temporary in a data stream that is considered for a particular period of time but, most of outlier detection techniques compare this current data with all the previous data sets stored earlier. Because data in data streams are dynamic therefore, it is not possible to store all the data.
- Most of the existing techniques detect a specific type of outlier, it may be point outlier or contextual outlier thus there is need for a unified framework that can detect all kind of anomalies.
- Existing methods for outlier detection are based on specific application domains based on the nature of existing outliers. Hence, a generalized technique is required that can be applied to different domains.
- Most existing techniques use static threshold for outlier detection which is efficient for fixed datasets but, for scalable data dynamic threshold must be used that need to be updated based on values in the dataset.
- Today most of the data is time evolving but, most of the existing techniques for outlier detection do not consider temporal aspect of data.
- Also, the existing methods for discovering anomalies using machine learning are mostly focused on attributes or entities but, ignore relationships among them.

3.2 Research Objectives

The objectives of the proposed research work are as follows:

1. To understand and analyse various concepts, techniques and tools available for detection of outliers.

2. To propose an efficient approach for outlier detection in big data.
3. To implement the proposed methodology using real life data.
4. To verify and validate the proposed methodology.

3.3 Research Methodology

To accomplish our research objectives a certain set of steps has been followed, the workflow of which is demonstrated in the Figure 3.1. The methodology to achieve each objective is as follows:

Objective 1:

- Identified the techniques for outlier detection and data mining based on different machine learning models.
- A broad review of various outlier detection techniques and algorithms used for detecting outliers is done. There advantages and limitations are identified.
- Identified the big data analysis platforms and tools through which large data can be handled.

Objective 2:

- Proposed an efficient technique for outlier detection which identifies outliers in linearly correlated datasets. The technique finds outliers on the top of Hadoop so that it can efficiently handle large datasets.
- Further, the proposed technique is enhanced for outlier detection in linearly as well as non-linearly correlated datasets. The technique handles both linearly and non-linearly correlated datasets so the classification accuracy is further enhanced.
- Proposed a framework for detecting outliers based on unsupervised machine learning model. The framework considers the classical fuzzy c-Means algorithm as basis and proposed

parallel fuzzy c-means algorithm based on clonal selection principle for detection of outliers in large datasets. The results are further refined using the cluster refinement algorithm.

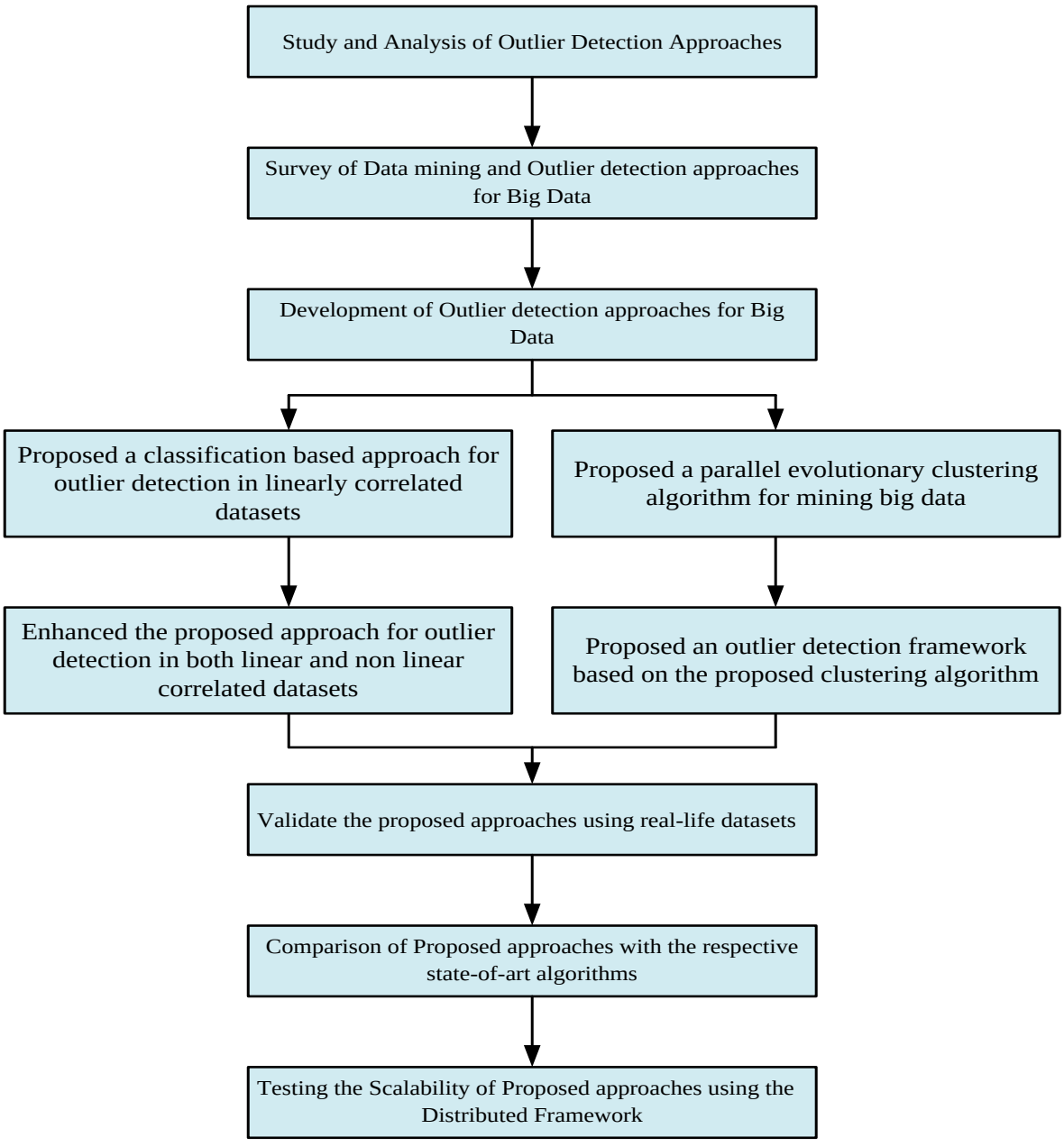


Figure 3.1: Workflow of the research methodology

Objective 3:

- The implementation of the proposed approaches are done on dell machines forming a Hadoop cluster of eight nodes, each node is having 8GB of RAM, 1 TB hard disk, and Ubuntu Linux Operating System installed. The nodes in the cluster are connected by means of a gigabyte network. One node in the cluster works as the master node, and the rest of the nodes are slave nodes.
- The proposed approaches are implemented using the map reduce framework on the top of Hadoop. The language used for implementation of the approaches is Java.
- The proposed approaches are implemented using real life and synthetic datasets of medical wireless sensors taken from Physionet library.

Objective 4:

- The efficiency of the proposed classification based algorithms for outlier detection is verified by comparing their performance with existing comparable algorithms.
- The efficiency of the clustering technique is proved by comparing its performance with state-of-art clustering algorithms.
- Tested the algorithms using different datasets and performance is evaluated in terms of true positive rate and false positive rate.
- The scalability of the proposed algorithms is confirmed by considering datasets of different sizes and is tested in terms of Speed-up by changing the data size and by varying the number of worker nodes.

3.4 Chapter Summary

In this chapter on the basis of literature survey the identified research gaps are discussed. To overcome the research gaps various research objectives are formulated. Further to achieve each of the objectives a research methodology is designed which is discussed in the last section of the chapter. In the next chapter, the proposed approaches for outlier detection based on correlation will be discussed in detail.

CHAPTER 4

CLASSIFICATION BASED OUTLIER DETECTION APPROACHES

In this chapter, the approaches proposed for outlier detection based on classification are explained. The existing algorithms for outlier detection lack scalability and consistency. First, the background and preliminaries are discussed in Section 4.1 of the chapter. Then, the working of the proposed algorithm for outlier detection in linearly correlated datasets is given in Section 4.2. The experimental evaluation of the proposed algorithm is explained in Section 4.3. Further, the proposed approach is enhanced to detect outliers in linearly as well as non-linearly correlated datasets which is discussed in Section 4.4. The performance evaluation of the improved proposed algorithm is explained in Section 4.5. Standard parameters are used to evaluate the performance of the proposed approaches. The scalability of approaches is tested using distributed computing platform.

4.1 Background

Outlier detection is one of the prominent research domains in the field of data mining and big data analytics. Nowadays, most of the data in healthcare centers is remotely monitored and is generated from different wireless sensors. The core objective of outlier detection in this domain is the recognition of the true physiologically anomalous data and the anomalies due to faulty sensors. In real healthcare monitoring scenario, various sensors are related to each other. So,

while detecting outliers in Wireless Body Sensor Networks (WBSNs), correlation among different sensor nodes is of major concern. Most of the existing outlier detection techniques do not consider the relationship among data attributes. Also, there exists various approaches for data mining that are scalable to big data [142] but, the traditional techniques for outlier detection are not scalable to big data. To address the above issues, in this chapter we propose an approach for outlier detection which is scalable to big data and considers linear relationships among data attributes. The technique is further enhanced to detect outliers in both linearly as well as non-linearly correlated attributes. The proposed approaches are implemented on Hadoop map reduce framework for the rapid processing of big data. The evaluation results are validated using the dataset of WBSNs taken from the Physionet library and simulated it for different cases. The results are compared with various existing outlier detection approaches and it is demonstrated that the proposed approaches are more effective in spotting the physiological outliers and sensor anomalies accurately. The various existing methods for finding correlation and data classification which are taken as background of the proposed approaches are discussed as follows:

4.1.1 SMO Regression

A support vector machine (SVM) is an optimized model to reduce prediction error and model complexity simultaneously [143]. Despite having several commendable traits, research in the field of SVMs has been hampered due to the fact that quadratic programming (QP) solvers were the providers of the sole known training algorithms for years.

In the year 1997, Osuna et al. [144] proved that subdividing a large QP problem into multiple smaller QP subproblems can optimize the SVMs. The optimization of each subproblem leads to minimization of the original QP problem. Once a stage is achieved where no further progress can be made with respect to the each of the smaller subproblems, the parent QP problem is optimized

and solved. Optimization of the QP problem via decomposition can be carried out with a defined size or linear memory footprint, since each of the subproblems can have fixed size. Moreover, several experimental results show that decomposition can be much quicker than QP. More recently, the sequential minimal optimization algorithm (SMO) was introduced [97, 145] as an extreme example of decomposition. Because, SMO is based on a subproblem of size two with each subproblem having an analytical solution.

Consider a set of data points, $\{(X_1, Y_1), (X_2, Y_2) \ldots (X_n, Y_n)\}$, such that $X_i \in R_n$ denotes an input, Y_i refers to a target output, and is the total number of exemplars. The weighted sum of kernel function outputs gives the SVM model result. The kernel function outputs can be an inner product, polynomial, Gaussian basis function, or any other function following Mercer's condition.

Thus, the result of an SVM model can either be a linear function of the inputs or a linear function of the kernel outputs. Because of the generalization of SVM models, they can take shape that seems similar to nonlinear regression, multilayer perceptron or radial basis function networks. The support vector machines and these methods are differentiated on the basis of the objective functions that they use to optimize and the optimization procedures used to achieve the minimization of these objective functions.

In the linear, noise-free case for classification, with $y_i \in \{-1, 1\}$, the output of an SVM is denoted as

$$g(x, w, b) = x \cdot w + b,$$

The optimization problem is defined as:

$$\text{minimize } \frac{1}{2}\|w\|^2$$

$$\text{subject to: } y_i(x \cdot w + b) \geq 1 \ \forall i$$

Inherently, this objective function represents the idea that one should find the model that is simplest to explain the data. The generalized SVM framework includes slack variables to consider miss-classifications, nonlinear kernel functions, regression, as well as other extensions for various other problem areas.

4.1.2 Pearson's Correlation Coefficient

In statistics, the Pearson correlation coefficient (PCC) [146] measures the linear correlation between two variables, X and Y. It is also called as Pearson's r, the bivariate correlation or the Pearson product-moment correlation coefficient (PPMCC). As per the Cauchy–Schwarz inequality, it takes up values in the range of +1 to −1, where the condition with its value as 1 is called as total positive linear correlation, when the value is 0 it is referred to as no linear correlation, and −1 is said to be total negative linear correlation. It was developed by the researcher named Karl Pearson based on a related theory given by another researcher Francis Galton in the 1880s and for which the mathematical formula was derived. The significance and identification of the Pearson's correlation coefficient is thus an example of the Stigler's Law.

The Pearson's correlation coefficient is mathematically defined as the covariance of two variables divided by the product of the standard deviations of the variables. This form of the definition has a "product moment", which is defined as the mean (the first moment about the origin) of the product of the mean-adjusted random variables; hence justifying the modifier product-moment in its name.

$$correlation\ coefficient\ (X,Y) = \frac{Covariance(X,Y)}{Standard\ deviation(X) * Standard\ deviation(Y)}$$

4.1.3 CANOVA

In Continuous Analysis Of Variance (CANOVA), [147] firstly we define a neighborhood with respect to each data point as per its *X* value, and then compute the variance of the *Y* value in the defined neighborhood. Finally, a permutation test for significance of the detected "within neighborhood variance" is performed. Given *X* and *Y* two random variables, where X_i and Y_i stand for the i^{th} observation. The sum square statistics within the neighborhood is defined as:

$$W = \sum_{i,j} (Y_i - Y_j)^2 \, , j < i, \left| rank\,(X_i) - rank(X_j) \right| < k$$

Where, *k* denotes the integer constant given by the user. $\left| rank(X_i) - rank(X_j) \right| < k$ represents the dataset's neighborhood structure. The assumption of CANOVA is that "analogous/neighbor X values lead to similar Y values". Thus, when X and Y are correlated, the *W* statistics have a tendency to be lesser than random expectation. A permutation test is performed to compute the significance of the observed *W* [148]. When *X* has same values (the tie condition), the rank of the tied *X* values is randomly shuffled in each permutation. In a tie condition, say, with the data:

X = [2, 2, 3, 4];

Y = [3, 2, 8, 5];

Since X has two twice, the sorting of data points does not come out to be unique. The algorithm randomly selects any of the following sorting sequences:

X = [2, 2, 3, 4]; Y = [3, 2, 8, 5]

X = [2, 2, 3, 4]; Y = [2, 3, 8, 5]

The algorithm is implemented using the CANOVA software available online as a project at https://sourceforge.net/projects/CANOVA/. The pseudo code of the CANOVA algorithm is presented in the shortened form as follows:

Sort data based on value of x

for (i=0; i < tie_shuffle; i++)

Shuffle y to tied x values

find observed W_i using observed Y

i ++

Sum = 0;

Observe w = avg(W_i)

for (i=0; i < no. of permutations; i++)

find random W by shuffling Y randomly

if (random W ≤ observed W)

Sum ++

Correlation_coefficient = Sum / no. of permutations

While calculating *W*, we exploit the fact that X_i is in a sorted sequence. Therefore, the algorithm complexity comes out to be in terms of *O(nlogn + np)*, where *n* stands for the size of the sample and *p* stands for the number of permutations. During performing the testing of multiple *X* variables against one *Y* variable, we need to perform just a single permutation of *Y* and the results of the permutation for all *X* variables can be reused.

4.2 Linear Correlation Based Approach for Outlier Detection

In the proposed approach for outlier detection based on linear correlation it is assumed that the correlation among the attributes in the dataset is linear. The proposed approach is composed of three different phases: correlation phase, prediction phase, and parallel computation phase.

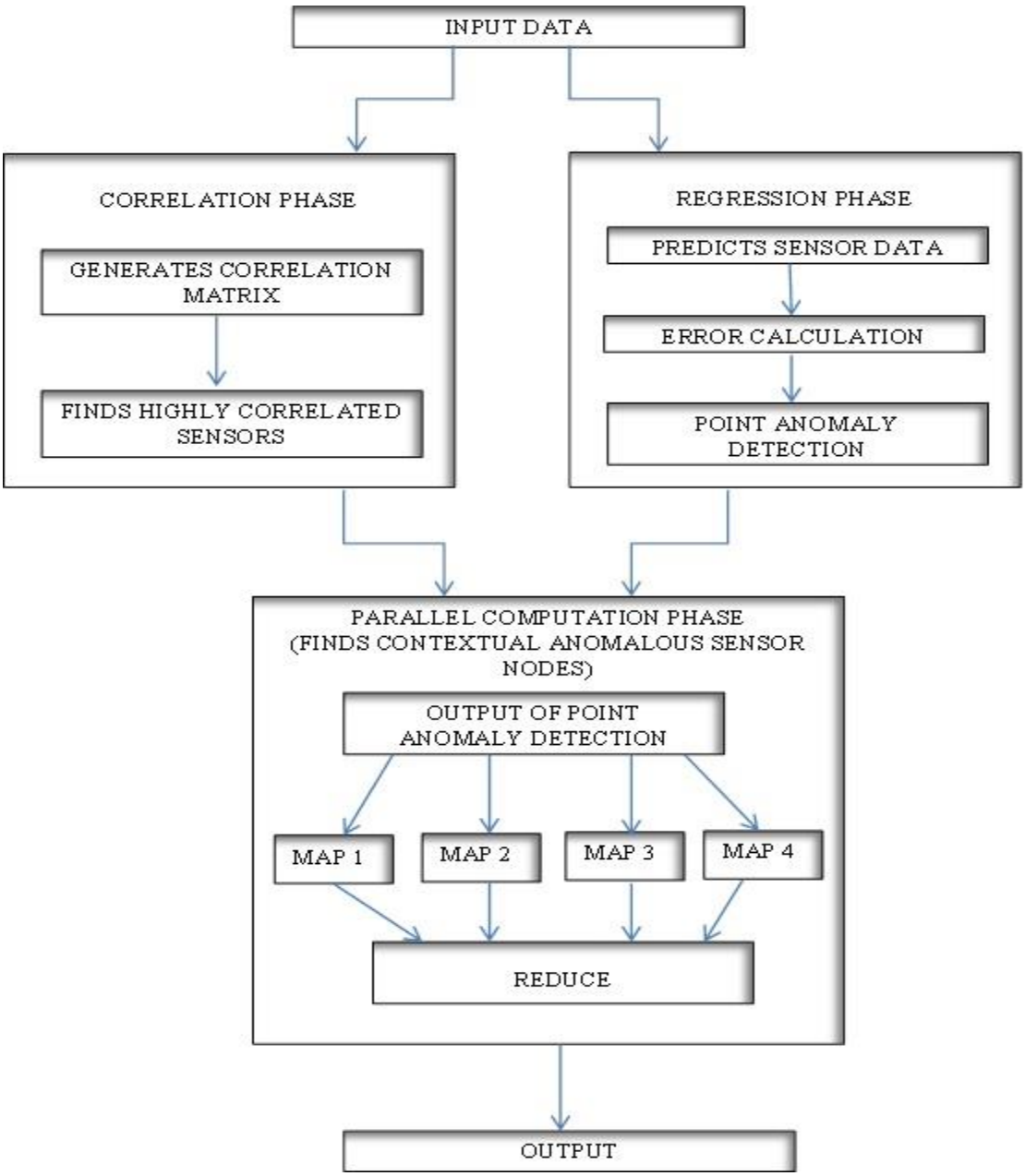


Figure 4.1: Workflow of the proposed approach

As a case study, a medical scenario is considered where a large number of wireless sensors are attached to the patient. The prediction phase is used to find out point anomalies in individual sensors that is detection of anomalous sensor reading. The correlation phase is designed to find out highly correlated sensors which are then used to find contextually anomalous data. By

identifying contextual anomalies we can then differentiate between the true medical emergency and sensor fault. The parallel computation phase is based on distributed map reduce framework and is executed on multiple nodes simultaneously hence reduces the time complexity and makes the proposed approach scalable to handle the large amount of data. The workflow of proposed approach is shown in Figure 4.1.

4.2.1 Correlation Phase

In the correlation phase values of correlation coefficient among different sensors are calculated. These coefficient values are represented in the form of a correlation matrix. In this scenario, N sensors collect values of different physiological parameters from the subjects. The dataset used in the correlation phase is an n×m matrix represented by Y.

$$Y = [y_{ij}] \quad \text{where } 1 \leq i \leq n$$

$$\text{and} \quad 1 \leq j \leq m$$

where, *i* corresponds to the number of sensors attached to a particular subject and *j* corresponds to the number of non-anomalous measurements which are used to build the correlation matrix.

$$Y = [y_{ij}] = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ \vdots \\ y_n \end{bmatrix}$$

In matrix Y, the i^{th} row is the i^{th} sensor profile across *m* measurements. For correlation matrix we find correlations among various sensors. Thus, for N sensors an n × n correlation matrix $C^{n \times n}$ is generated which is given as:

$$C = \begin{bmatrix} \text{Corr}(y_1, y_1) & \text{Corr}(y_1, y_2) & - - - & \text{Corr}(y_1, y_n) \\ \text{Corr}(y_2, y_1) & \text{Corr}(y_2, y_2) & - - - & Corr(y_2, y_n) \\ & \vdots & & \vdots \\ & \vdots & & \vdots \\ \text{Corr}(y_n, y_1) & \text{Corr}(y_n, y_2) & - - - & \text{Corr}(y_n, y_n) \end{bmatrix}$$

Correlation between two sensors is calculated using Pearson's correlation coefficient and is represented as r.

$$r = corr(y_1, y_2) = \frac{\sum y_1 y_2 - \sum y_1 \sum y_2 / m}{\sqrt{(\sum y_1^2 - \frac{(\sum y_1)^2}{m})(\sum y_2^2 - \frac{(\sum y_2)^2}{n})}}$$

Algorithm 4.1: Finding pairs of highly correlated sensors

Input: Data of m measurements for each of N sensors (Y).

Output: Pairs of strongly correlated sensors (p, q).

```
    // Calculates correlation coefficient between i & j sensors
1.  for i= 1 to n
2.      for j= 1 to n
a.          calculate C_ij = corr(y_i, y_j)
3.      end
4.  end
5.  Store correlation coefficient in n x n matrix C_pq.
6.  // Finds strongly correlated sensors
7.  for p = 1 to n          // C_pq is correlation matrix
8.      for q=1 to n
9.          while(p!=q)
10.             calculate max = max(C_pq)
11.             if 0.75≤max≤1 then
12.                 return (p, q)
13.     end
14. end
```

When two sensors are strongly correlated, the value of correlation coefficient lies in range (0.75 – 1.0). For instance, if y_1 and y_2 are strongly correlated then $0.75 \leq corr(y_1, y_2) \leq 1$. Based on the

correlation matrix the pairs of strongly correlated sensor nodes are identified, which are then used to find contextual sensor anomalies to reduce false positive rate. Algorithm 4.1 illustrates the procedure for finding pairs of highly correlated sensors.

4.2.2 Prediction Phase

Prediction phase is in particular used for point anomaly detection that is the detection of anomalies in individual sensors. In proposed technique, we used Sequential Minimal Optimization Regression algorithm (SMOReg) for building prediction model. SMOReg is used for prediction over other regression models such as Linear SVM, Univariate Gaussian predictor, linear regression etc. As time taken for analysis and average percentage error calculation is lowest in case of SMOReg in comparison to other regression models. In our approach, SMOReg works as a univariate predictor as it predicts values of each sensor independently based on its historical data. Suppose, we have training data from N sensors and each sensor has m sampled measurements defined as:

$$[(x_1, y_1), (x_2, y_2), \ldots\ldots\ldots\ldots\ldots, (x_m, y_m)] \in X \times R$$

The goal of the regression model is to estimate a function f(x) having least deviation from actual training data.

$$f(x) = \langle w, w \rangle + b \text{ where, } w \in X \text{ and } b \in R$$

$$Minimize \ \frac{1}{2}\|w\|^2$$

Subject to: $y_i - \langle w, w \rangle - b \leq \varepsilon$

$\langle w, w \rangle + b - y_i \leq \varepsilon$

where, b is bias, w is kernel function and Ɛ represents the error. Hence, SMOReg becomes a convex optimization problem. If error is lesser than the threshold then, sensor node is assumed non-anomalous otherwise it is anomalous.

4.2.2.1 Dynamic Prediction

For better accuracy, prediction is done using sliding window, where next value is predicted using the values present in the last window. Thus, for prediction of $(n+1)^{th}$ value SMOReg function is evaluated using last n values, where n is size of sliding window.

P_{n+1}	X_1	X_2			X_n	
P_{n+2}		X_2	X_3		X_n	X_{n+1}

Here, P_{n+1} and P_{n+2} are predicted values of $(n+1)^{th}$ and $(n+2)^{th}$ measurements respectively.

4.2.2.2 Dynamic Threshold

The threshold value for a particular sensor may vary from one patient to another based on different factors such as age, sex, lifestyle etc. The threshold value for the same subject may also vary based on its physiological condition. Therefore, by taking the fixed threshold error accuracy is compromised at various instances of time. Thus, it is important to adjust threshold value depending on different physiological conditions.

In the proposed approach threshold value (T_d) at a particular instance is the standard deviation which is computed as follows:

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)^2}$$

Here, μ is the mean of all x_i's. For finding dynamic threshold, the standard deviation is calculated on the same sliding window as used in dynamic prediction for a particular instance and compares the (Ɛ) error with the threshold value (Td).

Algorithm 4.2: Point anomaly detection algorithm

Input: Size of sliding window(n), Number of sensors(N), Actual sensor data(S_a)

Output: Sensor fault(s_f), No sensor fault(s_{nf})

```
1.  for i= 1 to N            // iterates for each sensor node
2.      for each sliding window
3.          Calculate predicted value S_{n+1}
4.          Calculate error in prediction Ɛ=S_a-S_p
5.          Calculate threshold (standard deviation) of that particular window (T_d).
6.      end
7.      if (T_d< Ɛ)  then
8.          return S_f
9.      else
10.         return S_nf
11.     Update sliding window
12. end
```

If Ɛ > Td then the sensor is assumed to be faulty otherwise there is no sensor fault. Algorithm 4.2 illustrates the procedure for finding point anomalous sensor nodes using dynamic regression and threshold.

4.2.3 Parallel Computation Phase

In this phase, Hadoop framework is used for contextual anomaly detection from point anomalous data. The map-reduce programming model is used as it is inherently parallel and is efficient for

large scale data analysis. The model is implemented using multi node Hadoop cluster. The output of correlation phase and point anomaly detection is given as input to map-reduce.

Suppose we have N sensors attached to a particular subject then at any time instance t the input to the mapper is pairs of highly correlated sensors and <t, s_1, s_2, s_3, ……., s_n>.

Algorithm 4.3: Contextual anomaly detection using correlation

Input: Pairs of highly correlated sensors, output of point anomaly detection algorithm < t, s_1, s_2, s_3, …….. , s_n >.

Output: Analysis of a particular subject at a particular time that is whether the sensors are faulty, true medical condition or normal condition.

```
1.  for key= 0 to t
2.      for each (i, j)        //(i, j) taken from correlation phase
3.          if Si.value = =1 then
4.              if Sj.value = = 1 then
5.                  Si.value=t and Sj.value=t
6.                  // true medical condition
7.              else
8.                  Si.value=s      //sensor fault
9.          end
10. end
11. for key= 0 to t
12.     if count(t) ≥1 then
13.         return (key, t)            // true medical condition
14.     else if count(s)≥1 then
15.         return (key, s)        // sensor anomaly
16.     else
17.         return (key, 0)        //  normal condition
18. end
```

where, s_n belongs to either 0 or 1, 0 represents non-anomalous sensor and 1 represents a faulty sensor. As map reduce framework works in key-value pairs. Thereby, time instance represents key for a particular input and corresponding sensor values represents value for map function.

The correlation matrix is used in the map-reduce framework to detect contextually anomalous sensor nodes. If both highly correlated sensor nodes are anomalous then it is true medical condition otherwise the sensor node is faulty. The output of reducer is a key-value pair where key represents a particular time instance and value represents whether the sensor is anomalous or there is a true medical condition or there is no fault in the system. Algorithm 4.3 illustrates the detailed procedure for finding contextual anomalies.

4.3 Experimental Evaluation

The fundamental evaluation of the proposed work is done on datasets taken from MIMIC II (Multiple Intelligent Monitoring in Intensive care) database of Physionet [149]. To validate our approach for wide data, the dataset having multiple attributes is taken and to validate data on distributed framework, the dataset with size greater than default chunk size of a single node in a Hadoop cluster is considered. Also, very few anomalous instances are present in actual application situation. So, to test the robustness of our approach we deliberately added anomalous instances to original data.

4.3.1 Dataset Used

The Physiobank is the large archive of data and contains physiological signals to be used for biomedical research. It currently includes data of healthy subjects and of patients with various health implications such as heart failure, respiratory failure, neurological disorders, brain injuries, Post op liver resection etc. For experimental evaluation, five datasets of intensive care

patients from MIMICDB of physionet are taken. The big data can be distinguished from traditional data using one or more of the four V's that are Volume, Velocity, Variety and variability. In the considered scenario a new data instance was generated every 10 ms from the sensor nodes attached to the patient which attributes towards the velocity of data. Also, the data is created from different sensor nodes hence generates varied data. Variability refers to inconsistency in data which can be seen in body sensor networks as many times an instance or two might get skipped due to loose contact of sensor with body. In order to efficiently handle the millions of data instances the experimental evaluation is performed on a Hadoop cluster with multiple nodes.

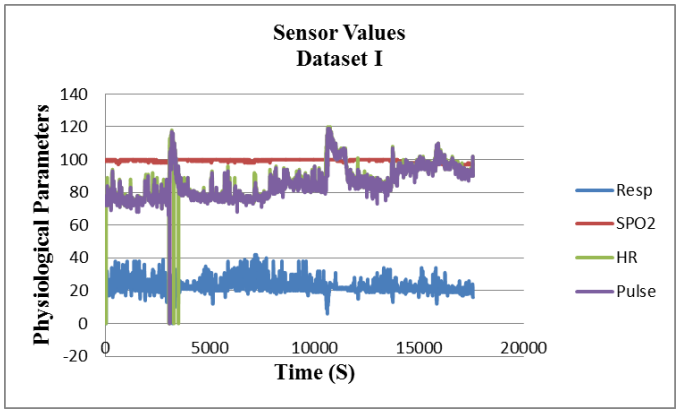


Figure 4.2: Sensor values for dataset D1

Table 4.1: Datasets used

Datasets	Number of Instances	Data Source
D1	18 K	http://www.physionet.org/physiobank/database/mimicdb/
D2	2 Lakhs	http://www.physionet.org/physiobank/database/mimicdb/
D3	1 Million	http://www.physionet.org/physiobank/database/mimicdb/
D4	2 Million	http://www.physionet.org/physiobank/database/mimicdb/
D5	2.5 Million	http://www.physionet.org/physiobank/database/mimicdb/

Table 4.2: Sensor readings for dataset D1 #221

RESP	Spo_2	HR	PULSE
23	99	75	75
23	99	76	75
22	99	76	76
22	99	76	76
20	99	76	75
20	98	75	75
20	99	75	74
23	99	74	73

Table 4.3: Sensor readings for D2 #276

PULSE	ABP	CVP	PAWP	Spo_2	ST_2	C.O.	Tblood	ST_1	ST_3	HR	PAP
109	82 124 61	12	14	98	-0.2	4.6	36.8	-0.1	-0.4	127	31 40 25
119	84 125 61	12	14	98	-0.2	4.6	36.8	-0.1	-0.4	127	29 38 22
120	83 124 62	12	14	98	-0.2	4.6	36.8	-0.1	-0.4	119	29 38 22
130	83 123 62	12	14	98	-0.2	4.6	36.8	-0.1	-0.4	119	29 40 22
132	83 123 62	12	14	98	-0.2	4.6	36.8	-0.1	-0.4	120	29 39 22
134	82 122 62	12	14	98	-0.2	4.6	36.8	-0.1	-0.4	123	30 39 24
132	79 118 60	12	14	98	-0.1	4.6	36.9	-0.1	-0.2	123	28 36 21
133	79 117 60	11	14	98	-0.1	4.6	36.9	-0.1	-0.2	128	30 39 24

4.3.2 Experimental Results

Experiments are performed on Dell Workstations T-5600 with INTEL Xeon e5 processor and 8 GB RAM. Java API of Weka is used for prediction, and parallel computation is done using multi node Hadoop cluster. The next value of each sensor node is predicted using dynamic SMO regression using a sliding window. Point anomalous nodes are detected by comparing error in prediction with the threshold value. To reduce false alarms, the output of point anomaly detection is again analyzed in map-reduce framework using Pearson's correlation coefficient. If both highly correlated sensors are anomalous then it is a case of true medical condition otherwise, it may be a case of sensor fault.

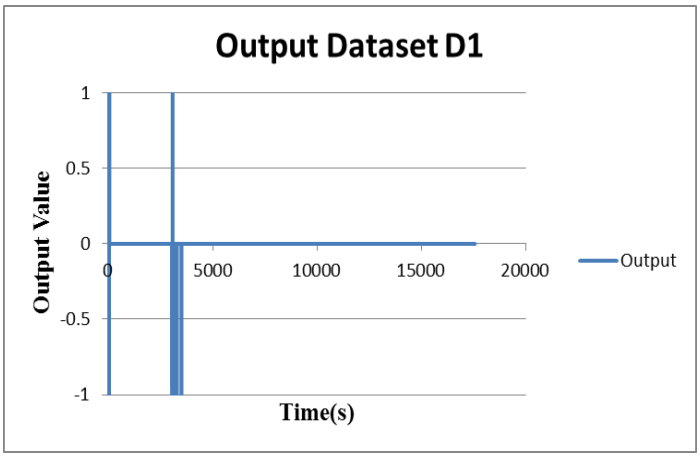


Figure 4.3: Output Dataset D1

The output of outlier analysis for dataset D1 is shown in Figure 4.3. In the figure, -1 represents the sensor fault, true medical condition is shown by 1, and 0 represents normal data. The dataset D1 consists of around 18k instances hence, the results of analysis are not clearly perceived. To understand the result analysis, the output of first 100 instances of dataset D1 is shown in Figure 4.4 with more clarity. From the Figure 4.4, it is observed that the first anomalous condition in D1 occurs near 30th instance which is identified as a sensor fault. After few instances the true

anomalous medical condition is detected. Datasets D2 and D3 are taken for validation of our approach on wide and big data.

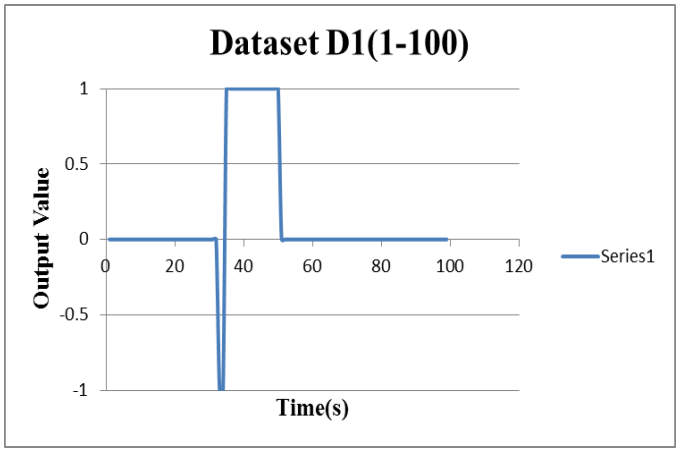


Figure 4.4: Dataset D1 (1-100 instances)

4.3.2.1 Scalability for Big Data

In order to validate the proposed approach on big data, a dataset having size greater than the default block size of Hadoop is considered. The proposed approach is implemented for all the datasets and is compared with existing approach given by Haque et al. [84] as shown in Figure 4.5. From the Figure 4.5, it is observed that for smaller data size (D1 and D2) existing approach performs slightly well in comparison to proposed approach as multinode Hadoop cluster takes time to fragment data into chunks, whereas when the size of data is increased (D3) the proposed approach takes a significant less amount of time in comparison to existing approach. The time is further decreased in case of dataset D4 and D5 as processing is done on Hadoop cluster with four nodes. The approach takes huge amount of time to process data in case of centralized framework which can be perceived from Figure 4.6.

The proposed approach is also analyzed by varying the number of worker nodes. For map-reduce processing each dataset is divided into chunks. If the dataset size is larger than the block size of

Hadoop then the dataset is divided into multiple chunks and each chunk is given to different mapper for data processing.

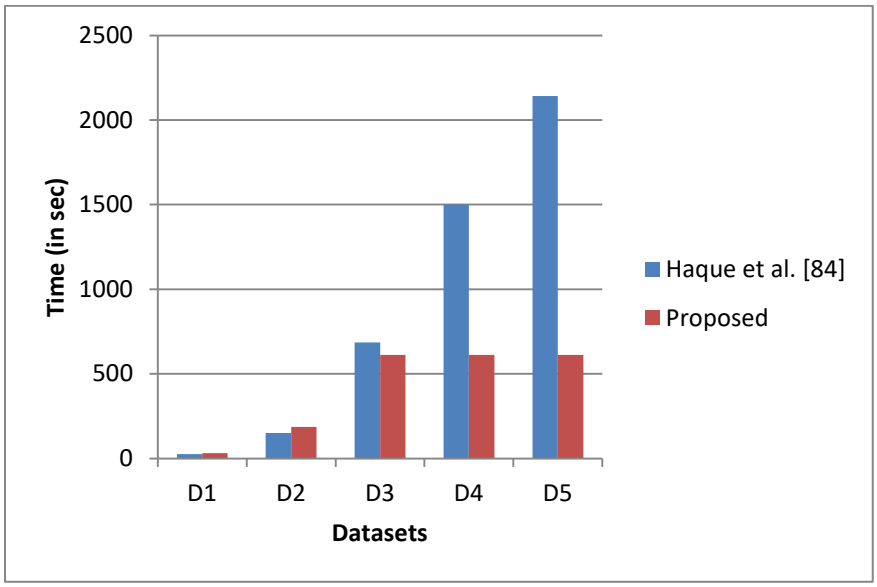


Figure 4.5: Comparison with existing approach

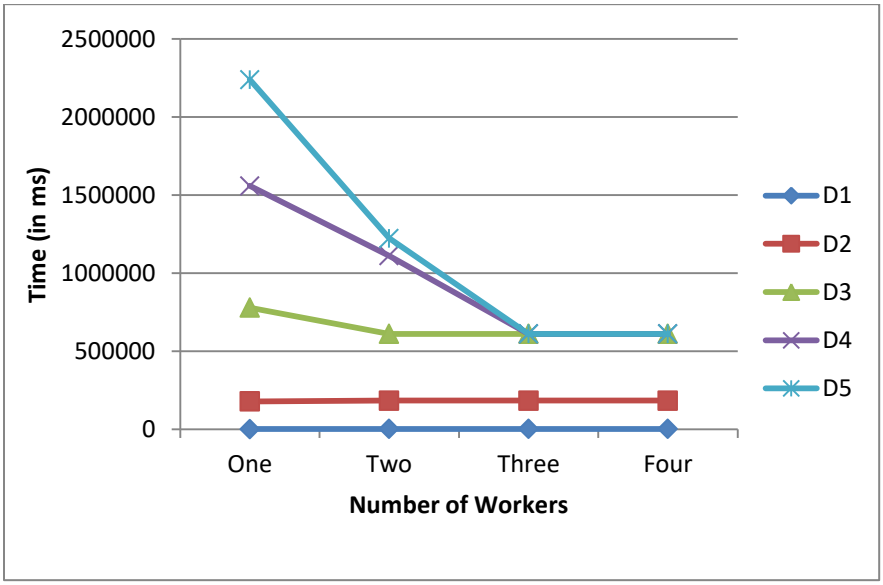


Figure 4.6: Run time with varying number of workers

Thus, datasets D1 and D2 are not segmented whereas, dataset D3 is divided into two chunks which can be processed using two worker nodes. It is clear from the Figure 4.6 that for datasets D1 and D2 time remains same for different number of worker nodes. For dataset D3 time taken remains same after two worker nodes and is not affected by increasing number of worker nodes. However, in case of D4 and D5 the time taken further reduces by using three nodes. Thus, as the data size increases the time taken for processing will be reduced by using more worker nodes. So, the proposed approach is scalable and efficiently handles big data with comparable execution time.

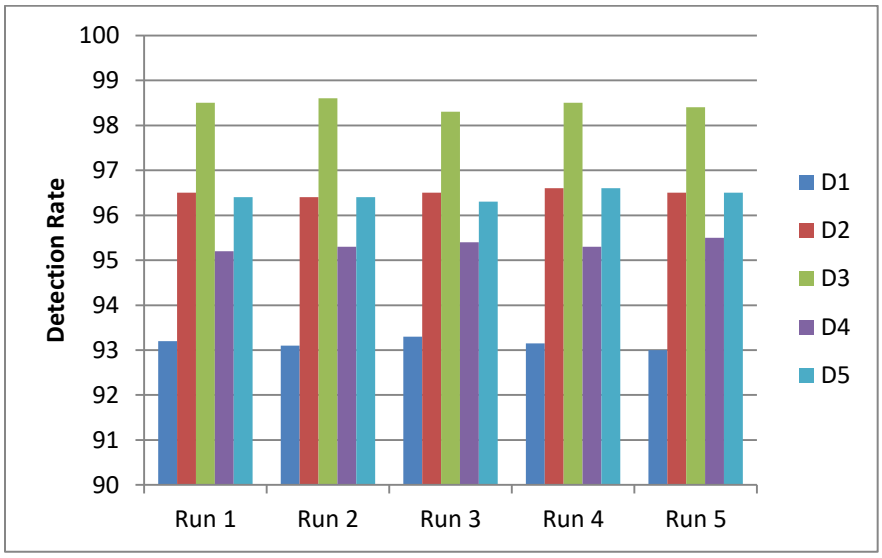


Figure 4.7: Detection rate in case of multiple runs

The consistency of the approach is shown in terms of detection rate during multiple executions. It is seen from the Figure 4.7 that for different datasets the detection rate during multiple runs remains consistent.

4.3.2.2 Performance Analysis

The performance is evaluated by comparing proposed approach with the existing technique given by Haque et al. [84] on the basis of two parameters that is detection rate and false positive rate.

Table 4.4: Performance Analysis

Approach	Anomalous instances	FN	TP	DR	FPR
Haque et al. [84]	468	15	453	0.97	0.0002
Proposed	468	3	465	0.99	0.0002

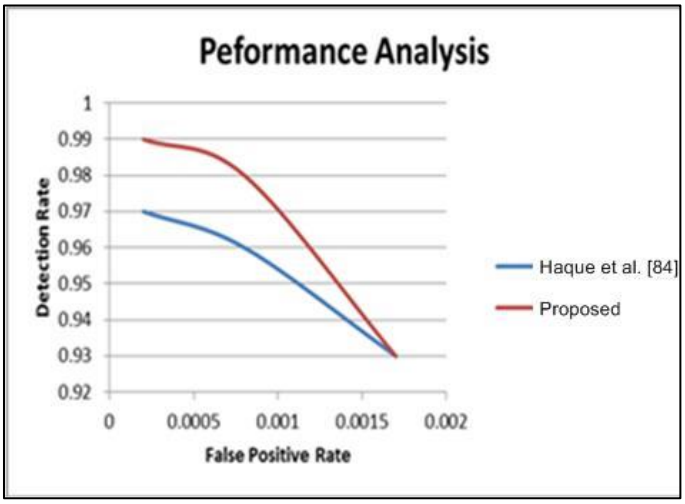


Figure 4.8: Performance Analysis

The proposed approach identifies false alarms and true medical conditions more accurately. In dataset D2 the sensor values are taken from twelve different sensors and at any instance of time in the dataset five correlated sensors become faulty. This is a situation of a true medical emergency. The existing approach considers it as a sensor fault only. Because in existing approach, for the occurrence of a true medical condition the number of faulty sensors must be larger than the average number of sensors. Comparative analysis of both the approaches is shown in Table 4.4. As FP and TN for both the approaches are same thus, FPR is also equal. From the Figure 4.8 it is clear that the proposed approach shows better detection rate in comparison to

existing approach by Haque et al. [84]. Therefore, the proposed approach can efficiently handle wider datasets having comparatively large number of attributes.

In the considered datasets, the sensor measurements are available from monitoring system and it is assumed that only linear correlation exists among various parameters. But, there exists certain application areas where these measurements cannot be presented linearly. For example, power law relationships, synthetic aperture radar data, hyper spectral sensor data, power demand prediction data etc. However, existing approaches for outlier detection considering non-linear relationships are not scalable to big data. So, the proposed work is further enhanced for outlier detection from large datasets having non-linear relationships. Correlation among the non-linear attributes can be measured using Randomized dependence coefficient, Kendall's tau coefficient, Spearman's rank correlation coefficient, continuous analysis of variance etc. The enhanced approach is discussed in the next section.

4.4 Non-Linear Correlation Based Approach for Outlier Detection

The enhanced approach for outlier detection in WBSNs is composed of correlation evaluator, prediction based outlier detector, and global integrated outlier checker. Again, as a case study, the healthcare monitoring scenario is considered in which various different sensors are attached to a patient. To evade dimensionality problem and for accurate detection of outliers in the first phase a correlation evaluator is designed that extracts the sets of sensor nodes which are strongly related to each other either linearly or non-linearly. Further, in prediction based outlier detector the currently observed data of every sensor is compared with its predicted value from past recorded data window based on a non linear kernel function. Further, based on the error in prediction point anomalous sensor nodes are detected. Finally, in the next phase the results are further refined for more accuracy by detecting contextual outliers using global integrated outlier

checker. Figure 4.9 presents the conceptual design of proposed outlier detection approach. The approach is implemented using distributed Hadoop map reduce framework with multiple nodes to make the approach scalable to big data. The detailed explanation of different phases of the proposed approach is given as follows:

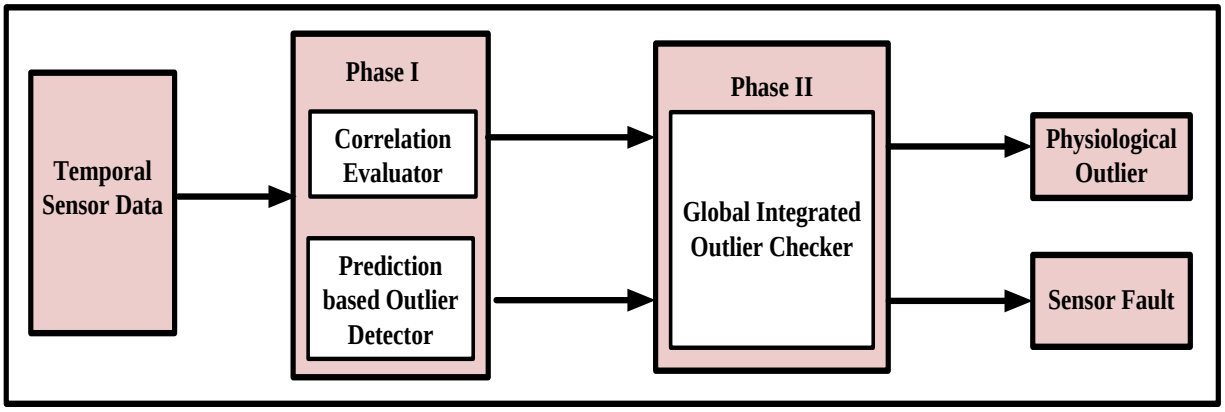


Figure 4.9: Conceptual design of the proposed technique

4.4.1 Correlation Evaluator

In a complex body network to monitor health of a person multiple sensors are attached with the body to gather different physiological attributes. Based on the sensor measurements of these attributes the anomalous instances in the health monitoring can be identified. Most of the conventional techniques for outlier detection based on correlation treat all the attributes equally and assume them to be linearly related. In view of this assumption, while dealing with multiple features it becomes challenging to identify the root cause of anomalous data. Also, in real life applications of sensors the nodes are linearly as well as non-linearly related. Hence, in the first phase of proposed approach a correlation evaluator is designed which mines both linearly correlated and non-linearly correlated sensor nodes. The working of correlation evaluator is shown in Figure 4.10.

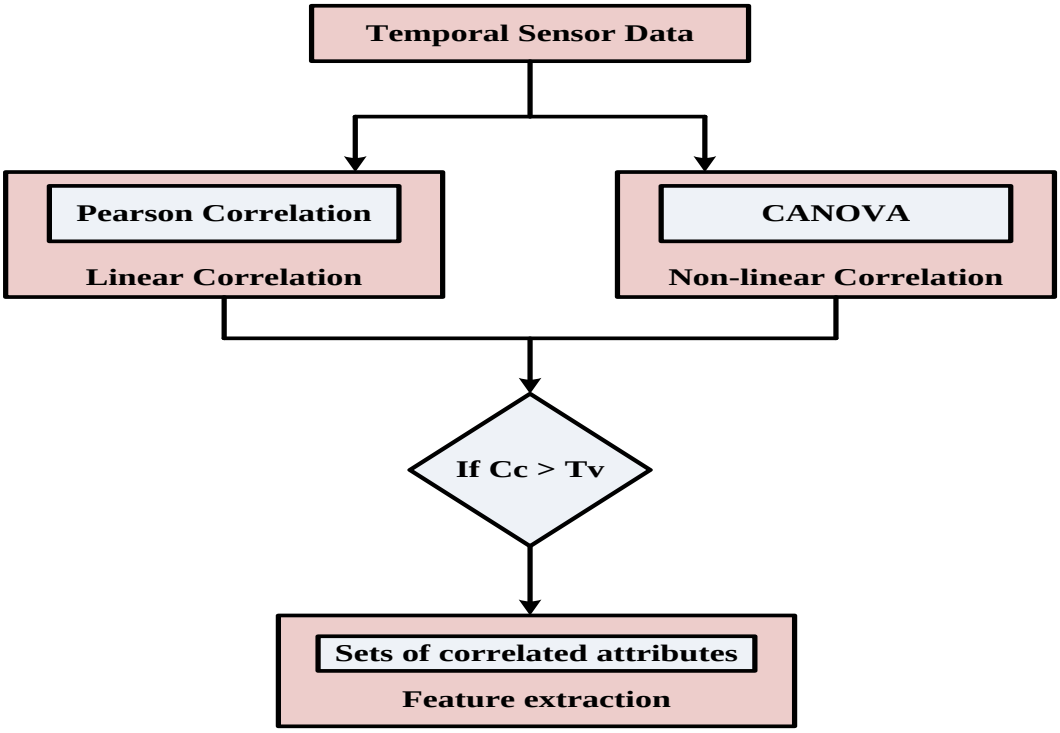


Figure 4.10: Correlation evaluator

Table 4.5: Correlation coefficient using different methods

Approach / Association	Pearson	Spearman	Kendall	Hoeffdig	MIC	CorGC	Maximal	CANOVA
Linear	0.99	0.97	0.96	0.95	0.98	0.97	0.97	0.88
Square	0.96	0.96	0.95	0.95	0.63	0.97	0.98	0.90
Cubic	0.61	0.69	0.93	0.94	0.65	0.98	0.98	0.92
Exponential	0.70	0.98	0.94	0.91	0.81	0.94	0.93	0.91
Periodic	0.33	0.31	0.75	0.56	0.61	0.49	0.91	0.86
Sine	0.31	0.32	0.32	0.40	0.57	0.38	0.76	0.88
Cosine	0.05	0.04	0.04	0.10	0.54	0.46	0.63	0.89

The most familiar measure for analyzing correlation among various nodes is with the help of correlation coefficient. Correlation coefficients using various techniques are evaluated on different linear and non-linear functions which are illustrated in Table 4.5. It can be perceived

from the table that for linear association the Pearson correlation coefficient performs well and for non-linear functions the performance of CANOVA [147] is more significant than the other methods. Hence, in the proposed technique the Pearson's correlation is used for detecting linearly correlated nodes and CANOVA is used for identification of non-linearly correlated nodes.

4.4.1.1 Linear Correlation

In the correlation evaluator phase for finding linear association, Pearson correlation coefficient is used on every pair of sensor node. Let the array of *m* observations of two sensors A and B be A = $\{a_1,\ldots,a_m\}$ and B = $\{b_1,\ldots,b_m\}$ then, the linear correlation between two sensors A and B are calculated with the help of following function:

$$r = LCorr(A,B) = \frac{\sum AB - \sum A \sum B / m}{\sqrt{(\sum A^2 - \frac{(\sum A)^2}{m})(\sum A^2 - \frac{(\sum B)^2}{m})}}$$

Here, *r* is correlation coefficient whose value lies in between -1 and 1. The sensors having |*r*| value near or equal to 1 are strongly correlated.

Let us consider the two arrays X and Y:

X = [8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19] and

Y = [4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]

The value of correlation coefficient (r) is calculated using the above equation for finding linear correlation and it comes out to be 1. Hence, it can be said that there is a strong linear correlation between the two arrays. It is further analyzed that the arrays are actually related to each other by function X = Y + 4.

4.4.1.2 Non-linear Correlation

Non-linear association is also widely used statistical relationship that is seen in healthcare monitoring applications. Continuous Analysis Of Variance (CANOVA) is applied to the dataset for detecting pairs of non-linearly correlated sensor nodes. CANOVA works in a similar manner like ANOVA but, it firstly finds within x neighborhood distance using a user defined constant *k* (rank(x_i) – rank(x_j) ≤ *k*) then within that locality calculates variance of y neighborhood. To apply CANOVA on the above sensors A and B we have to first sort observations according to one sensor let it be A. Then, the observations of sensor B are shuffled to tied values of sensor A in multiple permutations and calculate the *w* as follows:

$$w = \sum_{i,j} (b_i - b_j)^2$$

Subject to: $j < i$ and (rank (a_i) – rank (a_j)) ≤ *k*

If random *w* is lesser than the observed *w* during every permutation then, the two sensors are considered as highly correlated. The pseudo code of non-linear correlation calculation function is summarized as follows:

NLCorr(A, B)

{

Sort data of A sensor

for *(i = 0; i<tie_values; i++)*

Shuffle values of sensor B to tied A values

Calculate W_i;

Final W = average (W_i)

Count = 0;

for *(i=0; i<permutations; i++)*

Calculate random w

If *(random w ≤ observed w)*

Count ++

Return *Cc = count/permutations*

}

Let us consider another example of two arrays:

X = [-0.76, -0.96, -0.3, 0.65, 0.99, 0.41, -0.54, -0.99, -0.54, 0.42, 0.99, 0.65]

Y = [4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]

In this scenario the value of r calculated using the linear correlation equation comes out to be 0.42 so it can be said that the arrays are not related linearly. The arrays are further checked for non-linear correlation using CANOVA. As from the arrays it is seen that for same x value there are multiple y values so permutations has been used while finding correlation. The correlation between the arrays is calculated with the help of an R package "CANOVA". The Pseudo code for finding correlation is given using function NLCorr(A, B). The correlation is calculated by setting parameter values of *k*, permutations and tie_values. Here, *k* is the neighborhood structure of X values, permutations is the number of permutations and tie_values is the number of random shuffle times when there are ties in X. The correlation coefficient value of the above two arrays comes out to be 0.87 which signifies a strong non-linear correlation among the arrays. It is further analyzed that the arrays are related to each other with sine function.

4.4.1.3 Selection of Threshold Value for Strong Correlation

After the computation of correlation coefficients by both the methods, the sets of strongly correlated features are extracted based on their correlation coefficient value. The two arrays are considered as strongly correlated if they have correlation coefficient between 0.75 and 1. We have varied the correlation between 0.75 and 1 which is demonstrated in the Table 4.6. It is seen

Algorithm 4.4: Correlation Evaluator

Input: Data of m measurements for each of N sensors (Y_N)

Output: Set of highly correlated sensors S []

```
1.  Calculate linear and non-linear correlation coefficient between i^th and j^th sensor;
2.  for i ← 1 to n do
3.      for j ← 1 to n do
4.          C1 ← LCorr(y_i, y_j);              // linear correlation
5.          C_2 ← NLCorr(y_i, y_j);            // non-linear correlation
6.          if C_1 > C_2 then
7.              Store C_1 in C_ij where C is n×n matrix;
8.          else
9.              Store C_2 in C_ij where C is n×n matrix;
10.     end
11. end
12. Find sets of strongly correlated sensors;
13. for p ← 1 to n do
14.     for q ← 1 to n do
15.         k=1
16.         while p != q and k<n do
17.             S1[1] = p
18.             k++
19.             if C_pq > 0.85 then            // p and q sensor are strongly correlated
20.                 S1[k] ← [q]
21.             end
22.     end
23.     S[ ] = S1
24. end
25. return S;
```

that for the considered dataset the strongly correlated attributes have correlation coefficient above 0.85. Hence, the threshold value for the strong correlation in the proposed approach is set to 0.85. The sets having value greater than the threshold are considered highly correlated. It is seen from the table that if we further increase threshold to 0.9 then some of the strongly correlated functions that is cosine and polynomial might get excluded. So, we have taken the maximum value of threshold where all the correlated features can be extracted in the considered dataset. The threshold value can be set based on the dataset used or it can be generalized to be 0.75 for strong correlation. The detailed procedure of correlation evaluator is illustrated in Algorithm 4.4.

Table 4.6: Effect of varying threshold value on finding strongly correlated functions.

Threshold Value	Linear	Cosine	Exponential	Polynomial	Sine
0.75	✓	✓	✓	✓	✓
0.80	✓	✓	✓	✓	✓
0.85	✓	✓	✓	✓	✓
0.90	✓	×	✓	×	✓
0.95	✓	×	×	×	✓

4.4.2 Prediction based Outlier Detector

The prediction based outlier detector is used for detection of point outliers that is anomalies occurred in sensors taken independently. In the proposed approach modified SMOReg (Sequential minimal optimization based regression) [96] is used for prediction of next values. The reason for using SMOReg is that it is time efficient than other regression algorithms like linear regression, SVM, Gaussian regression etc.

The aim of the regression is to find a function that predicts values which have the least deviation from actual values. The next sensor value E_v is predicted using SMOReg function of weka used in default settings. Further, module for predicting next sensor value is implemented in java. The module predicts the next value using the SMO regression function applied to last n values inside the sliding window. The Complexity parameter is set to 1 as with value 0 no violations of margin is possible. The Polynomial Kernel is used as it will fit the data using a curved line, so it will fit the non- linear as well as linear training data accurately. The flowchart of the prediction based outlier detector is shown in Figure 4.11.

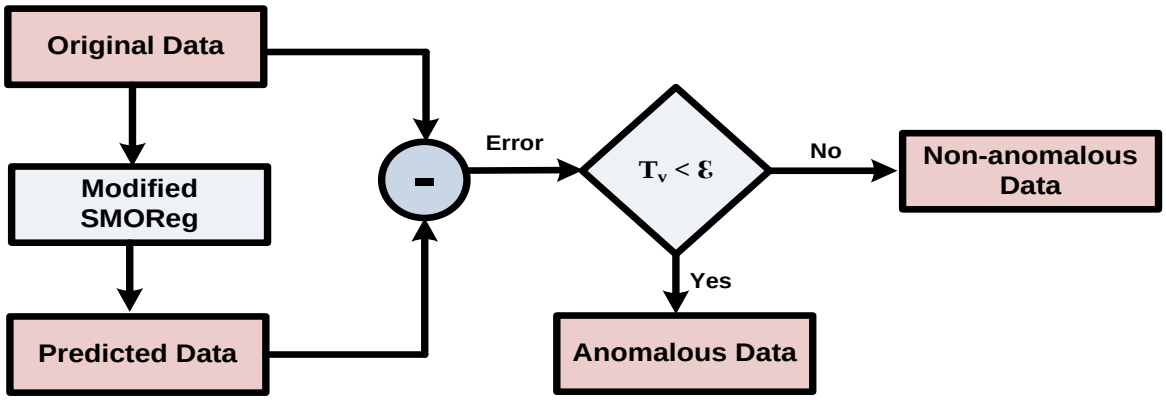


Figure 4.11: Prediction based outlier detector

For a sensor to be anomalous, error must be greater than a certain threshold. In modified SMOReg, two things are modified; the first is that prediction is done dynamically using a sliding window over last n recorded values. As the window size shrinks the accuracy of prediction decreases. However, by enhancing the size of window the accuracy increases but computational overhead also increases. In the proposed approach window size is taken as 25 because beyond that there is no significant increase in prediction accuracy which is shown in Figure 4.12.

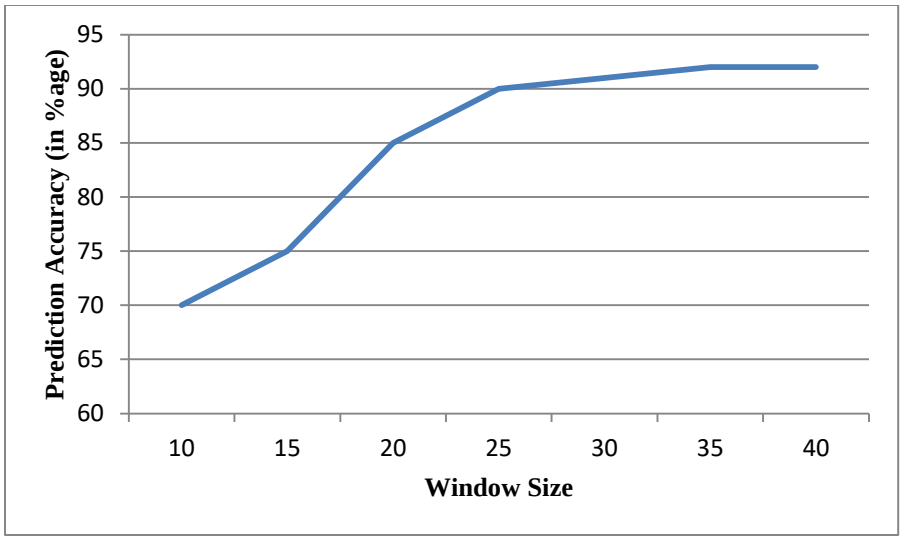


Figure 4.12: Variation in prediction accuracy with change of sliding window size

Algorithm 4.5: Prediction based Point Outlier Detection

Input: Window size (W_s), count of sensors (N_s), observed sensor value (O_v)

Output: Anomalous data (A_d), Non-anomalous data (N_d)

```
1.  for i ← 1 to N_s do
2.      for each sliding window do
3.          Calculate E_v            // predict sensor value
4.          Ɛ = O_v − E_v            // error calculation
5.          Calculate T_v
6.              if T_v < Ɛ then      // threshold check
7.                  return A_d;
8.              else
9.                  return N_d;
10.     end
11.     update window
12. end
```

The error threshold for a particular sensor node is also evaluated dynamically by using sliding window over n instances. The threshold is basically the value of standard deviation over *n* instances and calculated as:

$$T_v = \sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \bar{y})^2}$$

Here, T_v is the error threshold value and Ɛ is the error in prediction that is the difference between the actual and predicted sensor data. In Algorithm 4.5 the complete procedure of point outlier detection based on prediction is illustrated.

4.4.3 Global Integrated Outlier Checker

Global aggregated checker is used for detection of contextual outliers with the help of which we can distinguish between physiological anomalies and sensor faults. The output of point outlier detector acts as input to global outlier checker.

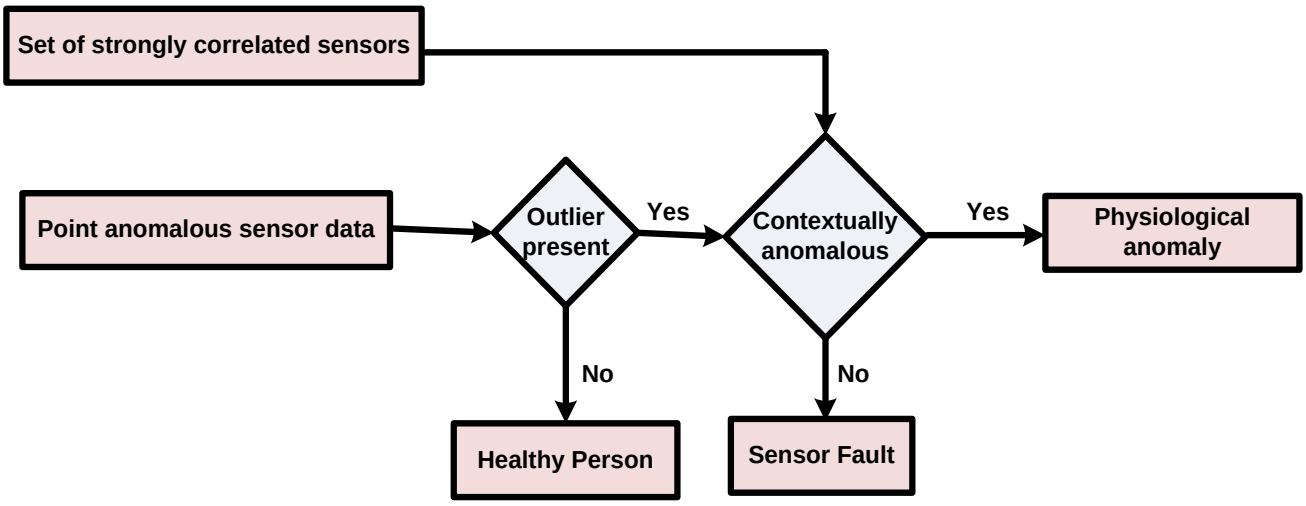


Figure 4.13: Global aggregated outlier checker

If in a set of correlated sensors, multiple sensors at a given time are point anomalous then, it is a case of the true medical condition. However, if the value of one sensor in a correlated set is

anomalous and other is not then it might be the case of sensor fault which can lead to false alarm if not handled accurately. The functionality of global integrated outlier checker is shown in Figure 4.13 and the procedure of contextual outlier detection is illustrated in Algorithm 4.6.

Algorithm 4.6: Global integrated outlier checker for contextual outliers

Input : Set of strongly correlated sensors, output of prediction based outlier detector (t, S_1, S_2, S_3, . . . , S_N)

Output : Physiological outlier (P_o), Sensor Fault (S_f), Healthy individual (H)

```
1.  for key ← 0 to t do                          // t is time instance of sensor reading
2.      for each array in set S[i] do            // S is set of arrays of correlated sensors
3.          for each element in set
4.              count = 0
5.              if S_i.value == A_d then
6.                  count++
7.          end
8.          if count>1 then
9.              return P_o;                      // Physiological outlier
10.         else if count == 1
11.             return S_f;                      // Faulty sensor
12.         else if count == 0
13.         for each sensor
14.             if S_i.value == A_d
15.                 return S_f;
16.             else
17.                 return H;                    // Healthy individual
18.     end
19. end
20. end
```

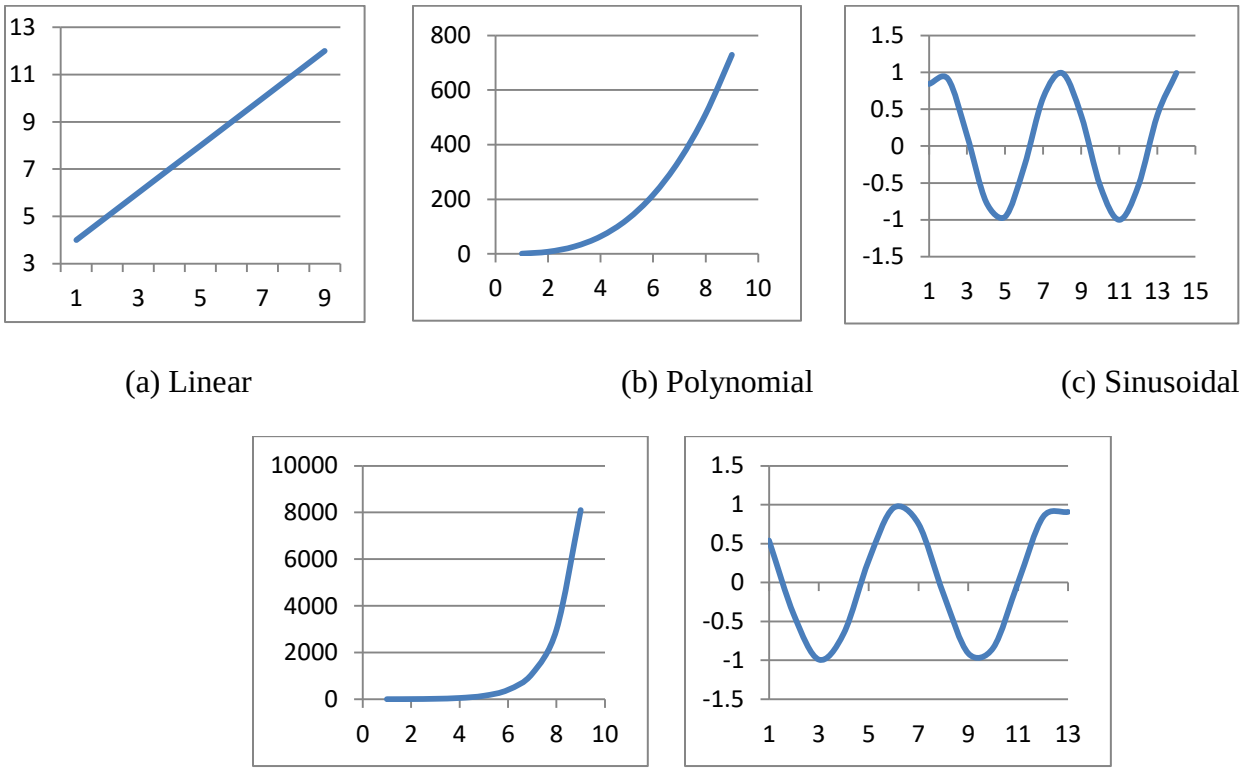
4.5 Experimental Evaluation

The proposed approach is implemented on Multinode Hadoop cluster with 8 Dell machines which are connected via a gigabyte network. Each machine is having 8GB RAM, 1 TB hard disk, Hadoop 2.6.0 and Ubuntu Linux operating system. HDFS (Hadoop Distributed File System) is used as the file system for storage. One node in the network acts as the master node in the Hadoop cluster and rest are considered as slave nodes. The evaluation of work is done by considering the scenario of wireless body sensor networks where different sensors are related to each other. Since in original data occurrence of outliers is very rare so to test the effectiveness of the proposed approach we inserted different types of outliers in the dataset. The proposed approach is compared with various existing approaches based on the detection rate, number of false alarms and scalability to big data.

4.5.1 Datasets Used

The Physiobank is a large archive which contains datasets of various sensors attached to different ICU patients [149]. We have taken the MIMIC II dataset of the physionet library as the base dataset and simulated this dataset using various functions. The simulated dataset consists of twenty-five sensors that are either correlated or independent. The various attributes of data are Pulse, ABP, CVP, PAWP, Spo_2, St_2, C.O., TBlood, St_1, St_3, Hr, PAP, and other features which are synthetically generated and related to each other with different linear and non-linear functions. The different types of correlations present in the dataset are shown in Figure 4.14. The dataset is synthetically expanded so that the scalability and correlation can be validated. The size of the dataset is much larger than the default block size of Hadoop so that the time efficiency can be observed by varying number of worker nodes. The data is distributed by map reduce

framework in chunks based on the block size of Apache Hadoop. The data chunks are then input to different worker nodes for computation. Hence, the time efficiency increases with increase in worker nodes.



(a) Linear (b) Polynomial (c) Sinusoidal

(d) Exponential (e) Cosine

Figure 4.14: Different relationships among various sensors for dataset simulation

4.5.2 Outlier Inserter

In the available real life application datasets, the occurrences of anomalous data are few so to test effectiveness of the proposed approach we inserted different outliers to the original dataset.

- Point outlier: Point outlier signifies that a single independent sensor node is anomalous. The occurrence of this type of outlier might be a physiological anomaly or a sensor fault. However, in healthcare monitoring probability of sensor fault is much higher than a true medical condition.

- Contextual Outlier: Contextual outlier signifies that there is a simultaneous deviation in the behavior of contextually similar nodes. This kind of outlier is helpful in differentiating sensor fault from the case of true medical emergency.

The data of Intensive Care Unit (ICU) patients is considered as basis. Further, we have synthetically created non-linear data using non- linear functions. We have inserted few anomalies in our dataset in such a way that we have deliberately added the cases where sensor may be detached to the patient or sensor may be providing wrong readings due to sensor fault. As in true medical scenario contextual outliers are present so there is no need for adding contextual anomalies.

4.5.3 Correlation Analysis

With the help of correlation analyzer highly correlated sensor nodes in the dataset are extracted. The output of correlation analyzer is shown in Figure 4.15. The threshold for strong correlation is taken as 0.85. The set of correlated sensors with their corresponding functions can be analyzed from the Figure 4.15. Here, S_i represents the i^{th} sensor node and C_c represents the correlation coefficient among sensor group. In case where more than two nodes are strongly correlated with each other, the correlation coefficient is calculated by taking the average of correlation coefficient of all the nodes. From the output of correlation analyzer it is observed that five groups of correlated sensors are formed that are linear (S_{11} S_{12} S_{15} S_{25}), cosine (S_1 S_3), polynomial (S_5 S_6 S_7), exponential (S_9 S_{20} S_{18}) and a sine function (S_{21} S_{23}). The correlation coefficient among different sensor nodes within a group is very strong while it is weak between sensor nodes belonging to different groups. The sensor nodes which are not correlated to any other sensor node based on their statistical characteristics are considered as independent and have low impact on the state of the person.

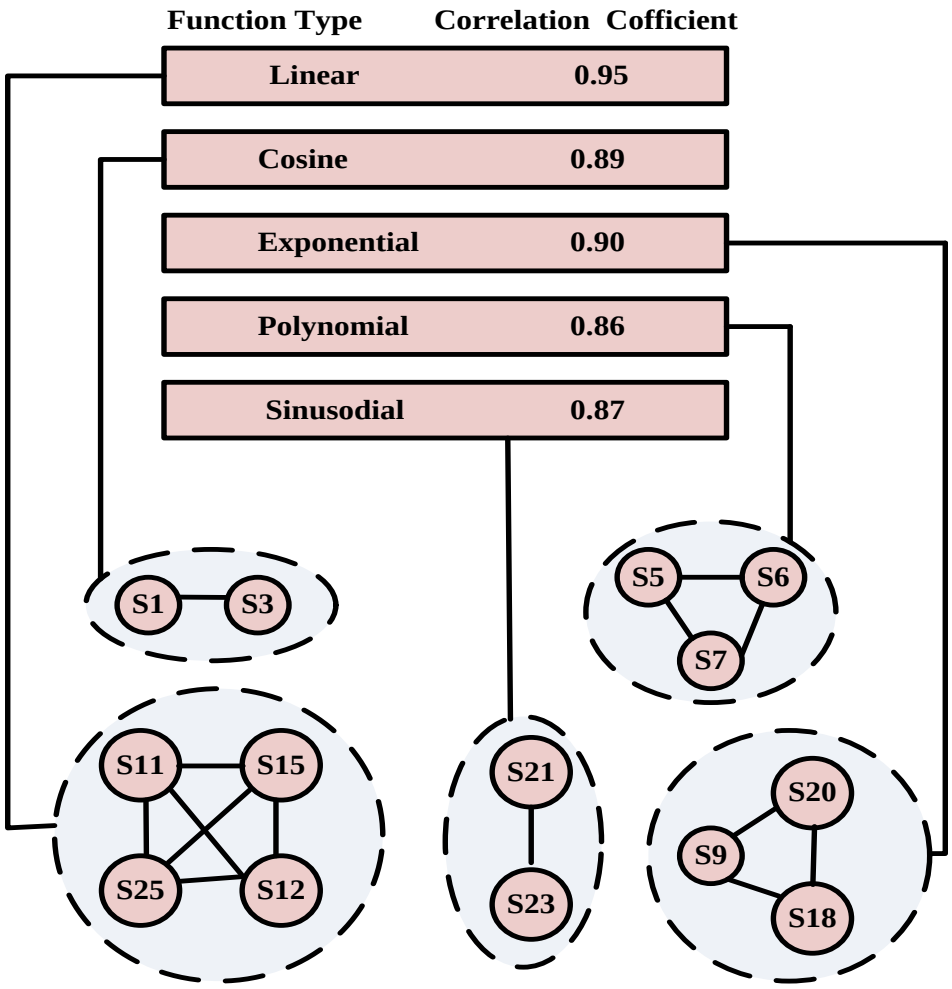


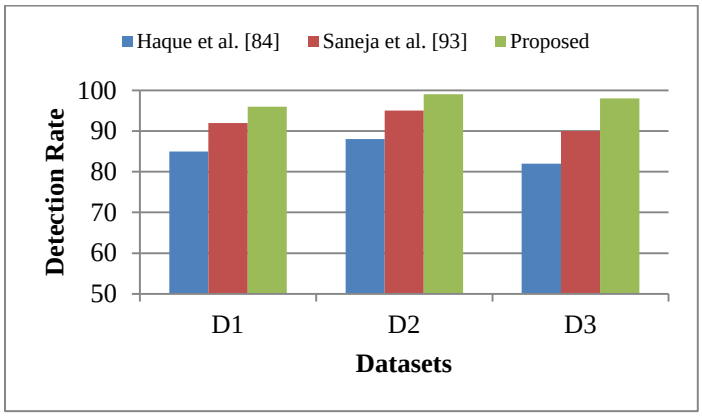
Figure 4.15: Sets of strongly correlated sensors

4.5.4 Performance Evaluation

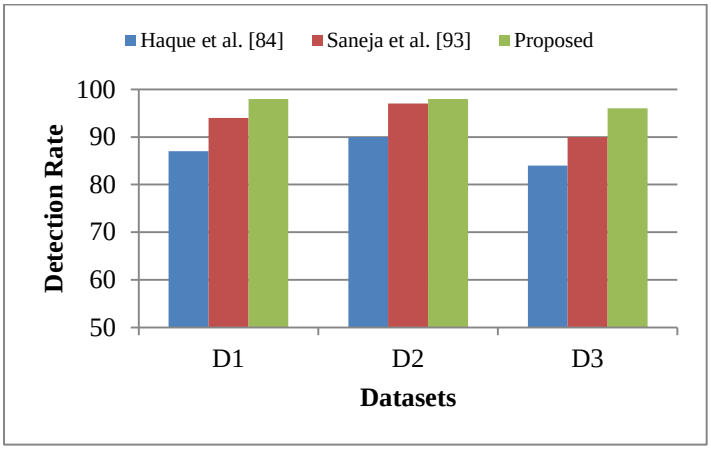
Experiments are carried out to evaluate the performance of the proposed approach in comparison to various other existing approaches for outlier detection. The point anomalous nodes are detected with the help of prediction based outlier detector and these are further refined for accuracy by analyzing highly correlated sensors using global aggregated outlier checker. The performance of the approach is evaluated on the basis various parameters given in Table 4.7. The performance comparison of the proposed approach with existing approach [84] and previously proposed approach [93] on the basis of detection rate is shown in Figure 4.16.

Table 4.7: Parameters taken for performance evaluation

Parameter	Description
DR_p	Detection rate of physiological outlier
DR_s	Detection rate of the sensor fault
FAR_p	False alarm rate of physiological outlier
FAR_s	False alarm rate of the sensor fault
Time(in ms)	Time taken by the proposed approach
Scalability	Scalability by varying no. of worker nodes

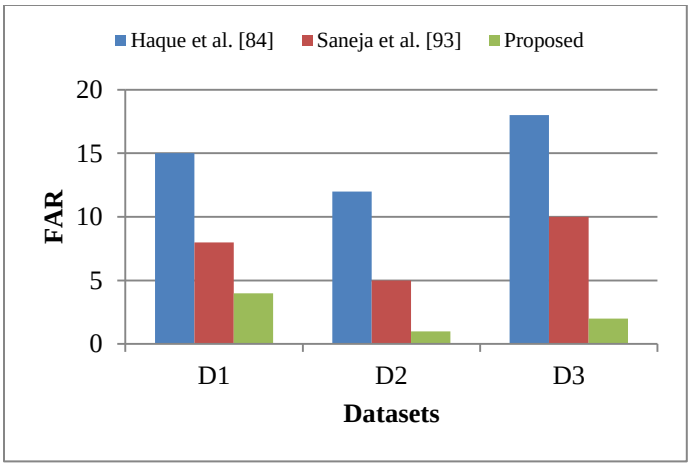


(a) Physiological Outlier

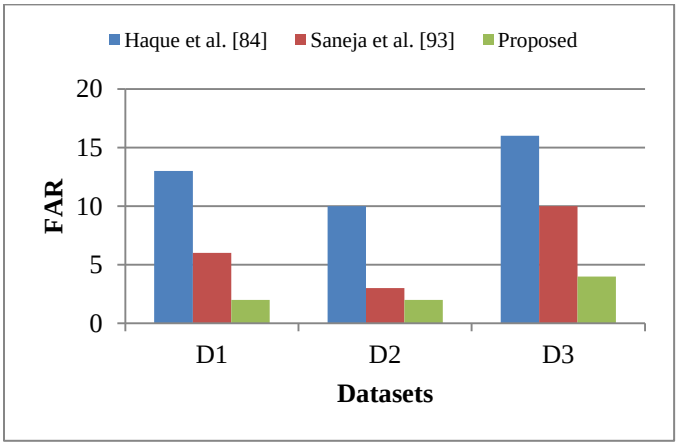


(b) Sensor Fault

Figure 4.16: Comparison with different approaches based on detection rate



(a) Physiological Outlier

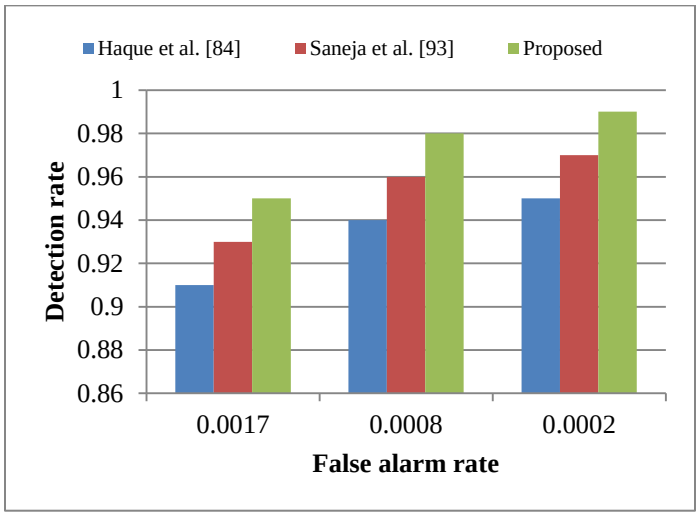


(b) Sensor Fault

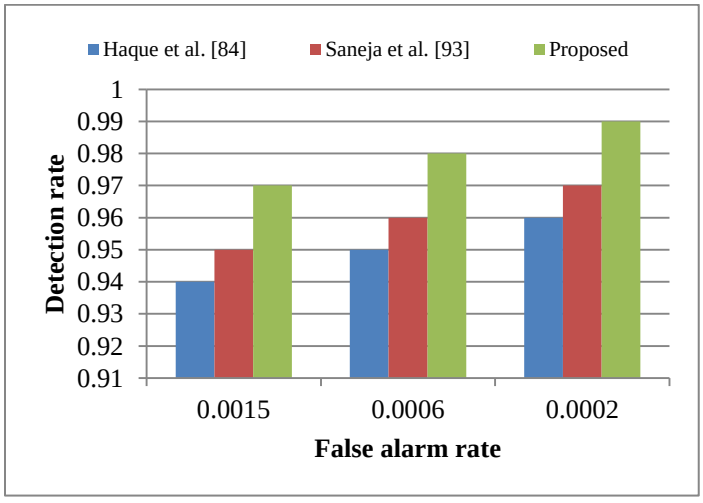
Figure 4.17: Comparison with different approaches based on false alarm rate

The performance comparison of the proposed approach with existing approach [84] and previously proposed approach [93] on the basis of false alarm rate is shown in Figure 4.17. It can be observed from the Figure 4.16 that the proposed approach is more efficient in detecting physiological anomalies and sensor faults in comparison to other techniques. The proposed approach is also efficient in the reduction of false alarms which can be seen from Figure 4.17. The variation of detection rate w.r.t false alarm rate for physiological outlier and sensor fault is shown in Figure 4.18. It is observed from the Figure 4.18 that for the same false alarm rate, the

detection rate of the proposed approach is higher than the other approaches. The time efficiency of the approach is shown in Figure 4.19. The proposed approach takes significantly less time in comparison to approach proposed by Haque et al. [84] and takes comparable time with our previously proposed approach [93].



(a) Physiological Outlier



(b) Sensor Fault

Figure 4.18: Detection rate (DR) w.r.t. False alarm rate (FAR)

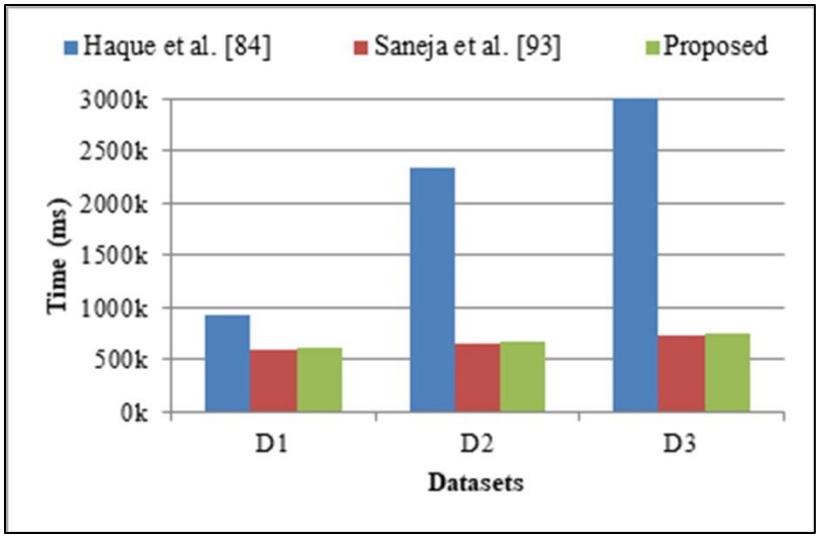


Figure 4.19: Comparison based on time by varying data size

The proposed approach is scalable to big data as it is executed on a distributed map reduce framework which is used for scalable and parallel computing. The proposed approach is implemented on a multinode Hadoop cluster having eight computational nodes. One node in the cluster is the master node and others are slave nodes. In the case where hardware is not available master node can also act as slave node.

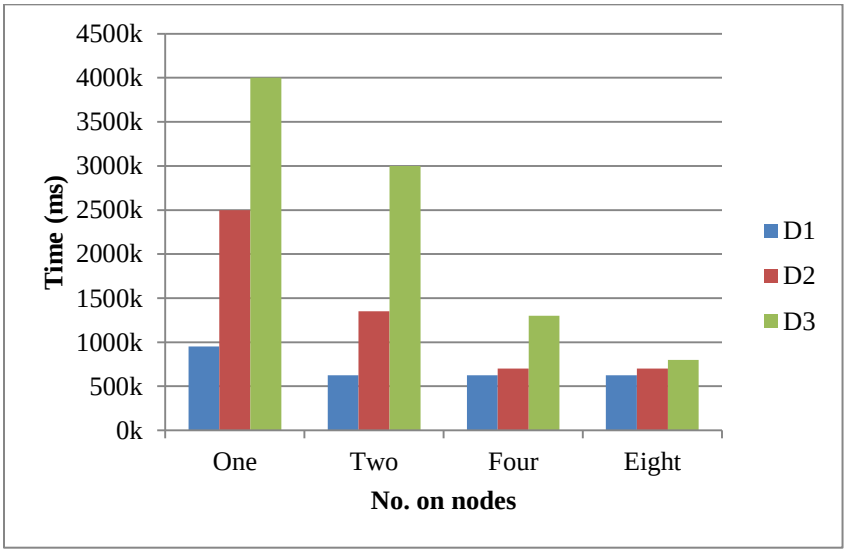


Figure 4.20: Scalability by varying number of worker nodes

The number of nodes can be further increased with the increase in data size which contributes to the scalability and time efficiency of the proposed approach. The validation of proposed approach is done by executing it multiple times by varying data size and the number of nodes in the Hadoop cluster. The computation time for different data sizes by varying number of worker nodes in Hadoop cluster is shown in Figure 4.20. It is seen that the proposed approach handles large datasets efficiently with comparable execution time.

4.6 Chapter Summary

Remote monitoring of patients in healthcare centers and assurance of alarms in case of emergency is the main application of WBSNs. Hence, it is mandatory that the application should be highly reliable and accurately detect physiological anomalies. The occurrence of false alarms in medical scenario dissipates the processing power and human resources. In this chapter, initially a linear correlation based outlier detection approach for big sensor data has been proposed. In the proposed approach, correlation evaluator has been formalized to find out the statistical relationship among sensors. Since in various application scenarios, the sensor nodes are correlated either linearly or non-linearly. Hence the proposed approach is further enhanced to detect outliers in both linearly as well as non-linearly correlated datasets. The prediction based anomaly checker detects point outliers using linear and non-linear kernel functions. The outputs of point outlier detection are further refined for contextual anomalies using the correlation matrix. The main advantage of the enhanced approach is that it extracts the meaningful information about both linear and non-linear attributes in the dataset. The approach is also scalable to big data as it is implemented using distributed map reduce framework by forming multinode Hadoop cluster of eight nodes. The experimental evaluation validates the high outlier detection rate and low false alarms in comparison to other techniques. Also, the scalability of the

approach has been validated by varying the data size and the number of worker nodes. The outlier detection approaches proposed in this chapter are based on supervised learning model. In the next chapter we propose a framework for outlier detection based on the unsupervised machine learning model.

Chapter 5

CLUSTERING BASED FRAMEWORK FOR OUTLIER DETECTION

In this chapter, a clustering based framework for outlier detection is proposed. The existing clustering algorithms lack scalability and consistency. Initially a clonal selection principle based parallel fuzzy clustering algorithm for data clustering is proposed. The proposed algorithm works on the principle of clonal selection algorithm and uses the objective function of fuzzy clustering. The algorithm is implemented using a distributed map reduce framework. Then a framework for outlier detection is proposed based on the proposed clustering algorithm. The framework performs data compression, data clustering, and cluster refinement. In Section 5.1 the background and preliminaries are given. Then, the proposed clustering algorithm is discussed in Section 5.2. Further, the working of the proposed framework is explained in Section 5.3. The performance evaluation and comparative analysis of the framework is given in Section 5.4. Standard parameters are used to evaluate the performance of the proposed approaches. The scalability of approaches are tested using distributed computing platform.

5.1 Background and Preliminaries

In this section, an intuitive explanation of the background algorithms is given. Firstly, the Piecewise Aggregate Approximation (PAA) compression algorithm with its improved version is explained in Section 5.1.1. Then, fuzzy c-means clustering and clonal selection principle are

discussed in Section 5.1.2 and 5.1.3 respectively. Finally, a brief explanation on working of map-reduce framework is given in Section 5.1.4. The various notations used in this chapter are mentioned in Table 5.1 along with their context.

Table 5.1: Notations and symbols used

Notation	**Description**
P	Length of original data series
Q	Length of compressed data series
K	Compression ratio / size of sliding window
k'	Number of steps sliding window moves
Y	Original data series
$\bar{Y}$	Compressed data series using PAA
Z	Compressed data series using improved PAA
A_b	Antibody group
A_g	Antigen group
J_{min}	Objective function of FCM
$\in$	Threshold for objective function of FCM
μ_{ij}	Membership function of i^{th} data value in j^{th} cluster
A_f	Affinity value / Similarity value
C_j	Centre of cluster j
C	Number of clusters
R	Weighting exponent / fuzziness factor
B_i	Boundary point
O_i	Outside point
I_i	Inside point
n_p	Total no. of points in all clusters
n_p^i	Number of points inside cluster i
T_j	Threshold for cluster j

5.1.1 Piecewise Aggregate Approximation

PAA is a simple compression method for scaling down dimensionality in time series data. It proximate a time series $Y = [y_1, y_2, y_3, \ldots, y_p]$ of length p in a q dimensional vector $\bar{Y} = [\bar{y}_1, \bar{y}_2, \bar{y}_3, \ldots, \bar{y}_q]$ where, $q \leq p$ and $\bar{Y}$ is calculated as follows:

$$\bar{Y} = [\bar{y}_i] = \frac{q}{p} \sum_{j=\frac{q}{p}(i-1)+1}^{(q/p)i} y_j$$

Although the compression technique is simple but, while detecting outliers it may not work well for some particular cases. For example, consider a situation where data value greater than 8 is presumed anomalous and let the original data series be Y = [3, 5, 7, 2, 11, 2]. The value of compression ratio (k) assumed is 3 in this case. After compression, the new series becomes $\bar{Y}$ = [5, 5] which is a non-anomalous series but, the original data series was anomalous.

To deal with this problem improved PAA was proposed [127]. In improved PAA the variance of particular sequence is also considered along with its compressed value. The variance of the i^{th} element of new data series is defined as:

$$var_i = \frac{q}{p} \sum_{j=k(i-1)+1}^{k.i} (y_j - \bar{y}_i)^2$$

The final outcome of compression is represented as $Z = [z_i] = (\bar{y}_i, var_i)$ and the sliding window size is equal to the compression ratio k.

5.1.2 Fuzzy C-Means Algorithm

The fuzzy c-means (FCM) clustering algorithm was developed by Dunn [107] and further augmented by Bezdek [108]. It is the most widely adopted fuzzy clustering algorithm that has been applied in various application areas such as data mining, medical diagnosis, pattern mining

etc. The algorithm partitions q data elements z_i (i = 1, 2, 3, …. , q) into c fuzzy clusters. The degree of membership of a data point z_i to a cluster j is defined by $\mu_{ij} \in [0, 1]$. The FCM algorithm generates clusters by minimizing the value of objective function which is represented by J_{min}.

$$J_{min} = \sum_{i=1}^{q} \sum_{j=1}^{c} (\mu_{ij})^r \left| z_i - C_j \right|^2$$

Where, $\left| z_i - C_j \right|^2$ is the dissimilarity of i^{th} data point to the j^{th} cluster center, μ_{ij} is the membership value of the i^{th} point in the j^{th} cluster, c is total number of clusters, $\in$ denotes threshold value and r is the fuzziness factor whose value lies in between 1 and ∞.

The sequence of steps of fuzzy c-means algorithm is given below:

1) Initialize c, r, z_i, and $\in$.
2) Randomly initialize the membership matrix U such that summation of membership values of every element in different clusters should always be unity.

$$U = [\mu_{ij}] \qquad \text{where } \sum_{j=1}^{c} \mu_{ij} = 1 \quad \forall\, i = 1\ to\ q$$

3) Find out the cluster centers C_j using μ_{ij}.

$$C_j = \sum_{i=1}^{q} (\mu_{ij})^r z_i \Big/ \sum_{i=1}^{q} (\mu_{ij})^r \qquad \forall\, j = 1\ to\ c$$

4) Calculate the new membership matrix based on cluster centers using:

$$\mu_{ij} = \left(\sum_{k=1}^{c} \left(\left| Z_i - C_j \right| \Big/ \left| C_k - C_j \right| \right)^{2/(r-1)} \right)^{-1}$$

5) If J_{min} reaches minimal optimal solution or maximum number of iterations are reached then stop the iteration else go to step 3.

5.1.3 Clonal Selection Algorithm

The clonal selection procedure is a heuristic algorithm similar to genetic algorithm. The two differs significantly in terms of inspiration, notation, and preliminaries. Genetic algorithm adopts terminology from natural genetics whereas CSA (clonal selection algorithm) uses immunological terminology. De castro and Von Zuben proposed CSA based on clonal selection principle and biological immune system [150].

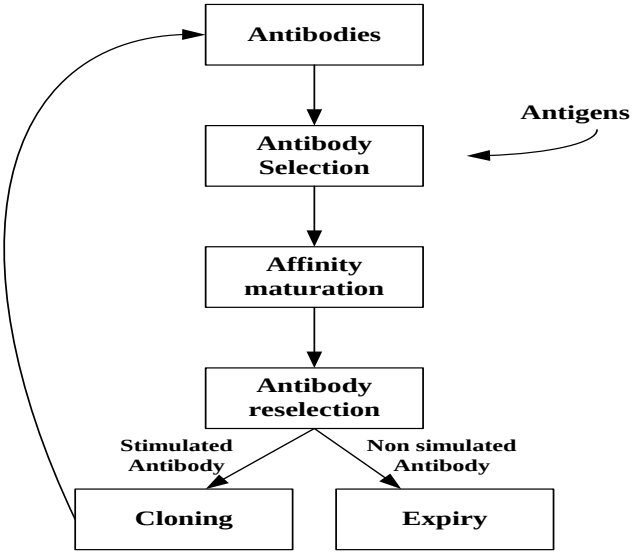


Figure 5.1: Workflow of clonal selection algorithm

The algorithm was developed for recognition of different patterns and to resolve problems of multimodal optimization. It acquires memory property of human body to identify similar antigens instantly and explores optimal solutions using mutation. The workflow of the CSA is shown in Figure 5.1. The process of the clonal selection algorithm is given as follows:

1) Randomly initialize initial antibody group, A_b.
2) For each antigen, record its similarity with every antibody.
3) Select N highest affinity elements whose value of objective function is lower in comparison to others.

4) Clone them based on their similarity (higher similarity means the higher number of clones).
5) Mutate all the clones based on their similarity value (higher similarity means less mutation rate).
6) Replace initial antibody group with these mutated clones.
7) Go to step 2 until stopping criteria is reached.

5.1.4 Map Reduce Framework

Hadoop is an open source framework which implements different algorithms using map reduce paradigm. It consists of two parts that are HDFS (Hadoop distributed file system) and map reduce [151]. HDFS is used to store large datasets and map reduce for processing of data. HDFS splits the dataset into smaller chunks according to default block size of Hadoop (64MB) or user defined block size. After that these chunks are processed with map reduce programming using multiple nodes. The two programs implemented in map reduce are mapper and reducer [152]. Both map and reduce functions work on <key, value> pairs.

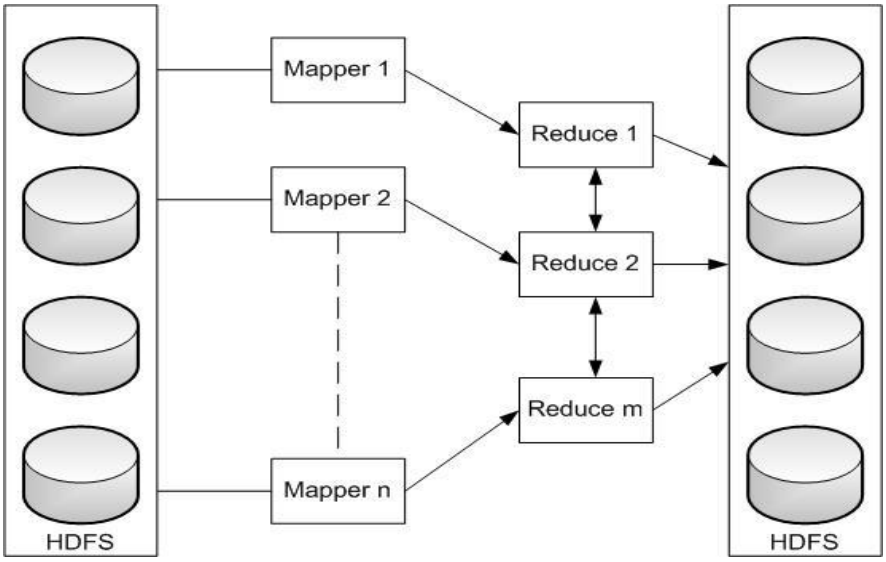


Figure 5.2: Working of Map-reduce

The map task takes input records in <key, value> pair, processes the data and generates intermediate data in the same format. The reduce function takes intermediate data as input, aggregates the data generated from different mappers and produces the final output. The working of map reduce is shown in Figure 5.2.

5.2 Design of the Proposed Framework

The proposed approach for anomaly detection is based on data compression and fuzzy clustering. It is analyzed from the literature that fuzzy clustering approaches tend to outperform hard clustering approaches in terms of accuracy. But, there is problem of local convergence in traditional fuzzy clustering algorithms. Hence, various different variants of traditional fuzzy clustering algorithms were proposed that overcome the problem of local convergence. But these approaches were based on centralized systems hence takes large execution time when there is need to process large amount of data. The traditional fuzzy c-means algorithm using centralized framework was proposed in 1981. With the advancement in technology the distributed map reduce framework was proposed in 2004 by Google which makes the algorithms fast and scalable. The map reduce programming model was implemented as open source software framework named Apache Hadoop in 2005. The map reduce framework based fuzzy c-means clustering was proposed by Ludwig in 2015. However, the problem of local convergence still remained in the proposed fuzzy clustering approach using distributed framework. To resolve the issue of local convergence, the proposed approach initializes the cluster centers based on an artificial immune system inspired algorithm that is clonal selection algorithm.

Also, it has been observed in the last decade that enormous amount of data is coming from various wireless sensor networks. To apply fuzzy clustering on large amount of data requires complex computations which increase computational time and overhead in comparison to hard

clustering algorithms. Thus, to reduce computational overhead and to maintain time efficiency comparable to hard clustering approaches, the data has been initially compressed using the modified piecewise aggregate approximation. Then, a distributed clustering approach has been proposed for data clustering. The algorithm for clustering is implemented using map reduce framework.

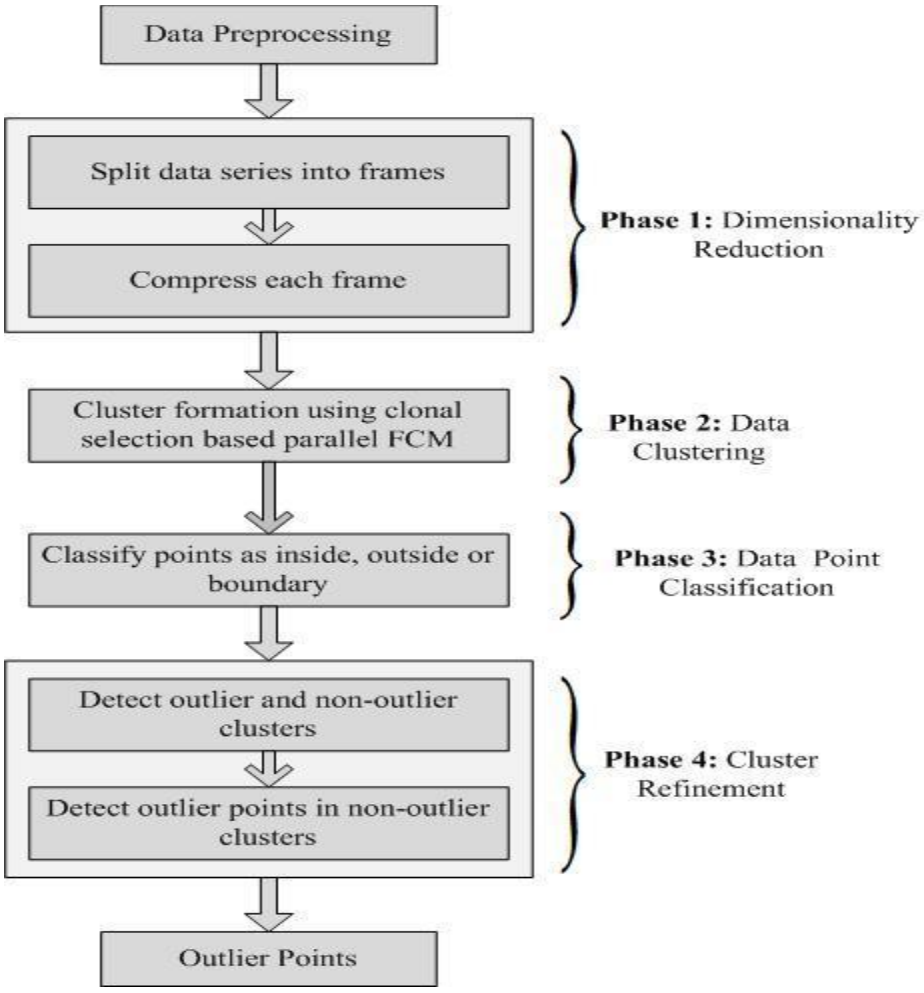


Figure 5.3: Workflow of the proposed framework

Most of the techniques surveyed for detection of anomalous data use clustering as the basis of classification. The Detection Rate (DR) of these approaches is comparatively lower as the detection was completely based on clustering results and the clusters are not refined. Thus, in order to detect anomalies more accurately with low False Alarm Rate (FAR), the clusters are

further refined in the proposed framework. The proposed cluster refinement algorithm specifically works on boundary points of large clusters by keeping in mind accuracy as well as the efficiency of the algorithm. The unified framework which integrates the proposed algorithms for different purposes is illustrated in Algorithm 5.1 and the workflow of the proposed framework is presented in Figure 5.3.

Algorithm 5.1: Integrated framework for anomaly detection

Input: original data series (Y)

Output: outlier clusters and outlier points in data series

```
1.  Begin
2.      read dataset
3.      if data is redundant then
4.          remove redundancy
5.      else if data is not normalized
6.          do data normalization
7.      call (algorithm 2)          // Phase 1: Dimensionality reduction
8.          compress(y_i, k)
9.            return Z
10.     call (algorithm 3)          // Phase 2: Cluster formation
11.         Cluster(Z_i, c, ∈)
12.           return c clusters
13.     call (algorithm 4)          // Phase 3: Data points classification
14.         classify points as O_i, B_i, I_i
15.     call (algorithm 5)          // Phase 4: Cluster refinement
16.         cluster refinement (c)
17.           return outlier clusters, outlier points
18. End
```

5.2.1 Phase 1: Dimensionality Reduction

To deal with the issue of data overhead modified piecewise aggregate approximation (PAA) is used in the proposed framework for compression of data. The main purpose of the proposed framework is efficient and accurate detection of anomalous subsequences. Thus, the data loss during compression is of least concern and the accuracy of detection is not affected by compression.

Algorithm 5.2: Modified PAA

Input: original data series(Y), compression ratio (k)

Output: compressed data series(Z)

1. **for each** sliding window i
2. **begin**
3. **for** j = 1 to k
4. **begin**
5. $Y_i = [y_j]$
6. calculate $\overline{Y_i}$
7. calculate var_i
8. $z_i = (\overline{Y_i}, var_i)$
9. **end**
10. **end**
11. **return** Z

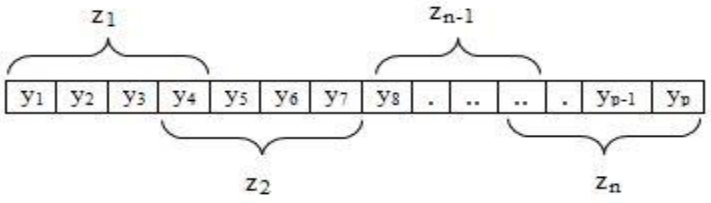


Figure 5.4: Movement of sliding window

The PAA compression algorithm is preferred as it is time efficient in comparison to other compression techniques like Fourier transform, wavelets and so on [153]. The number of steps sliding window moves is represented by k'. In the existing state of art algorithms for better accuracy the value of k' was taken 1 (k'=1). But, for small value of k' the time efficiency is low. In the proposed framework the modified PAA is used where the value of k' is taken (k-1) for better time efficiency. As variance of each subsequence is also considered so, the large value of k' will not affect the accuracy of the result. The movement of the sliding window is shown in Figure 5.4 with compression ratio 4. The procedure of the compression is illustrated in Algorithm 5.2.

5.2.2 Phase 2: Data Clustering

The proposed algorithm for clustering is prominent from different traditional algorithms for clustering. It is analyzed from the literature that fuzzy clustering approaches tend to outperform hard clustering approaches in terms of accuracy. However, the problem of local convergence still remained in fuzzy clustering using distributed framework. To overcome the limitations of parallel fuzzy clustering by considering the advantages of both clonal selection and fuzzy clustering a parallel algorithm is proposed for clustering data. The proposed algorithm works on the principle of clonal selection and objective function of parallel fuzzy c-means. The approach is implemented using map reduce framework in order to make the approach scalable to large amount of data. The workflow of the proposed clustering algorithm is presented in Figure 5.5.

The analogy of the notations used in clonal selection algorithm and fuzzy clustering is described as follows:

- A_b represents the antibodies group which consists of m randomly initialized antibodies. Each antibody represents a set of randomly initialized cluster centers.

$$A_b = [A_b^1, A_b^2, \ldots\ldots, A_b^m] \quad \text{and}$$

$$A_b^i = [C_{i1}, C_{i2}, \ldots\ldots\ldots, C_{ic}] \quad \text{where } i = 1 \text{ to } m$$

- A_g represents the group of antigens which denotes the data set to be clustered.

$$A_g = [A_g^1, A_g^2, \ldots\ldots, A_g^q]$$

where q is the number of antigen population that is the total count of unlabeled records.

- The similarity of the antigen group to a particular antibody is affinity and is represented by A_f and is calculated using objective function J_{min}.

$$A_f = A_f(A_b^i) = 1/(1+J_{min})$$

- The number of clones generated is calculated as follows:

$$N_c = m^2 + m$$

- M_c represents the memory cell that is, the best antibody having highest affinity value at convergence and is the optimal cluster centroid.

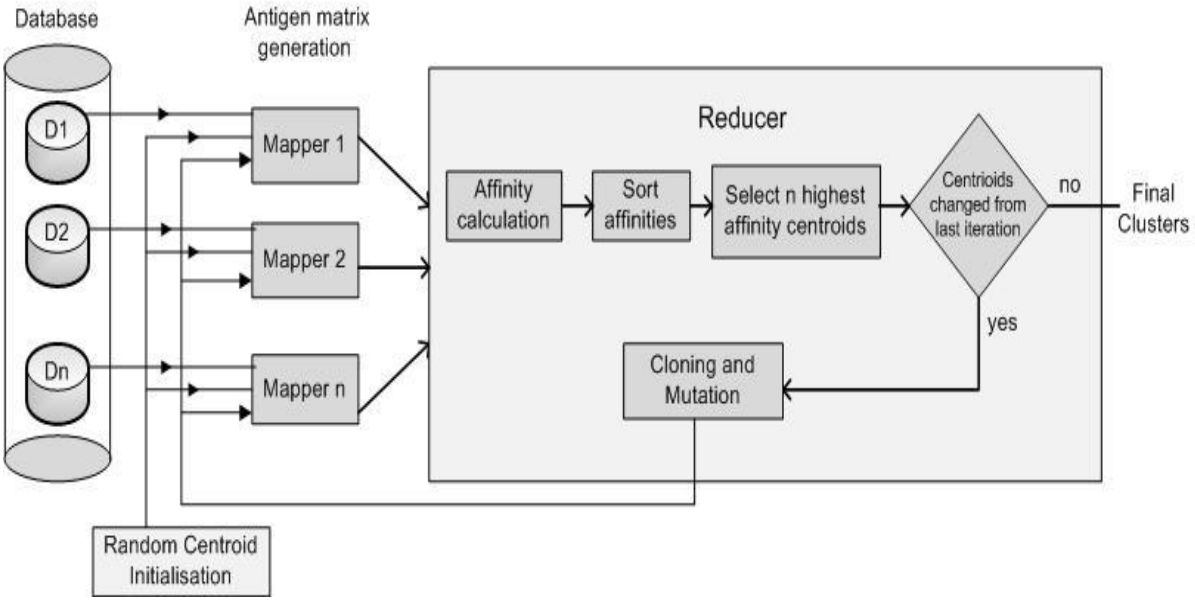


Figure 5.5: Flowchart of proposed clustering algorithm

Algorithm 5.3: Clonal Selection based Parallel FCM Clustering

Input: antibodies, antigens, threshold

Output: memory Cell, antigen matrix, data points

Map phase:

1. **if** *iteration* = 1 then
2. initialize A_b
3. **for** *i = 1 to m*
4. **begin**
5. Calculate antigen matrix (μ_{ij}) corresponding to each A_b^i
6. send output to reducer(A_b^i, A_g, μ_{ij})
7. **end**

Reduce phase:

8. **for** *i = 1 to m*
9. **begin**
10. calculate $A_f^i = A_f(A_b^i)$ // Affinity calculation
11. **end**
12. **if** *iteration* > 1 **then**
13. $M_c' = A_b^i$ where $A_f = \max(A_f^i)$ // Memory cell selection
14. **if** $M_c' - M_c < \in$ **then**
15. **return** M_c and exit
16. sort A_f^i desc // Affinity ordering
17. $M_c = A_b^i$ where $A_f = A_f^1$
18. **for** *i= 1 to n*
19. **begin**
20. $H_{af}^i = A_f^i$ // Antibody selection
21. **end**
22. **for each** H_{af}^i
23. clone corresponding A_b^i // Cloning and mutation
24. mutate all the clones based on their affinity values
25. choose m mutated antibodies
26. *iteration* ++
27. send mutated antibodies to mapper

The proposed clustering algorithm works in two phases that is the map phase and the reduce phase. In map phase, randomly selected antibodies are entered as input which is considered as keys in the (key, value) pair and the data points in the dataset as values. In map phase firstly randomly initialize some antibodies and then calculate the antigen matrix corresponding to each data point and antibody. Then this output is sent to reduce phase where affinity corresponding to each antibody is calculated based on objective function of FCM. The calculated affinities are then sorted and the antibodies corresponding to n highest affinities are cloned and mutated to generate new antibody population.

The newly generated antibody population is again sent to map phase and the whole process is repeated until convergence. The final set of centroids is the highest affinity antibody after convergence which is known as memory cell. The population is finally clustered according to this memory cell. The complete procedure of the proposed clustering approach is given in Algorithm 5.3.

5.2.3 Phase 3: Data Point Classification

In most of the anomalous datasets the edge where the normal data point and where the anomalous data point resides is not clearly distinct. As every data point in a dataset has a different probability of belonging to different clusters. So, the fuzzy set theory can be used in this situation to detect outliers. In fuzzy set theory, membership functions are employed to represent the likelihood of a data point belonging to a particular cluster.

In the proposed framework, the modification of basic fuzzy c-means is made. The three crisp partitions of data points are considered in the proposed framework that are inside points, outside points and boundary points of a cluster. These points are differentiated on the basis of their membership values. The internal points have membership values approximately equal to one, the

membership values of boundary points lies strictly in between 0.3 and 0.7. As the outside points do not belong to any cluster thus, their membership value is near zero. The procedure used for the data point classification is given in Algorithm 5.4.

Algorithm 5.4: Data point classification

Input: data points(Z), membership values

Output: classified data points

```
1.  begin
2.      for j = 1 to c
3.      begin
4.          for i = 1 to q
5.          begin
6.              if μ_ij < 0.3
7.                  return i as O_ij          // Outside point
8.              else if μ_ij > 0.7
9.                  return i as I_ij          // Inside point
10.             else if  0.3 ≤ μ_ij ≤ 0.7
11.                 return i as B_ij          // Boundary point
12.         end
13.     end
14. end
```

5.2.4 Phase 4: Cluster Refinement

In most of the traditional methods for anomaly detection after clustering, the data points residing inside small clusters are considered as outlier points. But, for the cases where outlier point lies within a big cluster with a low membership value, we have to compromise with the accuracy. In order to deal with this issue, a cluster refinement algorithm has also been proposed and used in

our framework. Each data point of the dataset is assigned to different clusters with varying membership values using CSPFCM. Then, the clustered data points have been labeled into three categories based on their membership values. The points having membership values near zero are considered as outside points (O_i), the points having membership values near 1 are inside points (I_i), and the points having membership values strictly in between 0.3 to 0.7 are boundary points (B_i).

Algorithm 5.5: Cluster refinement

Input: Clusters generated with classified data points

Output: outlier points, outlier clusters

1. **begin**
2. **for** j = 1 to c
3. **begin**
4. **if** $n_p{}^j < avg(n_p)/3$
5. **return** j as outlier cluster
6. **else**
7. calculate threshold for that cluster

$$T_j = 1.5\left(\frac{\sum_{i=1}^{n_p{}^j} Z_i \mu_{ij}}{n_p{}^j}\right)$$

8. **for each** boundary point B_i in cluster j
9. **begin**
10. **if** $\left|B_i - C_j\right| > T_j$
11. **return** B_i as outlier point
12. **end**
13. **end**
14. **end**

In the proposed cluster refinement algorithm, the data points lying inside small clusters that are the clusters having data points less than one third of the average points are labeled as outliers without refining that clusters. But, there is a probability that non-anomalous clusters may contain some outlier points. For detection of outlier points residing inside non-anomalous clusters, cluster refinement is done on every boundary point of the non-anomalous cluster. The detailed procedure of cluster refinement is given in Algorithm 5.5.

5.3 Experimental Evaluation

To illustrate the performance of the proposed framework, experimental analysis has been carried out on five different real datasets. Experiments are performed on Dell Workstations T-5600 with INTEL Xeon e5 processor and 8 GB RAM. The implementation has been done in java using a multinode Hadoop cluster. The source of datasets is MIMIC II (Multiple Intelligent Monitoring in Intensive care) database of Physionet library [149]. In real application scenario the number of anomalous instances present was very few, so for testing the strength of the proposed framework and its different modules the anomalous instances are artificially added to the original data. For comparative analysis, the traditional FCM algorithm, FCM algorithm with the clonal selection, and compressed k-means algorithm with AIS are taken as the baseline.

5.3.1 Datasets Used

The Physiobank is a large archive of physiological signals and data to be used for biomedical research. It currently includes data of healthy subjects and of patients with various health implications such as heart failure, respiratory failure, neurological disorders, trauma etc. For our experimental evaluation, we have taken various datasets which are cases of brain injury, sepsis, respiratory failure, cardiogenic shock, and trauma. The details about the datasets are given in

Table 5.2. The physiological parameters taken from the datasets are Heart rate (HR), Pulse, Respiration rate (Resp), Oxygen saturation in the blood (SPO2), arterial blood pressure (ABP) etc. Sensor values of different parameters for the datasets are shown in Figure 5.6. The spikes in the Figure 5.6 represent the outlier points for that particular physiological parameter.

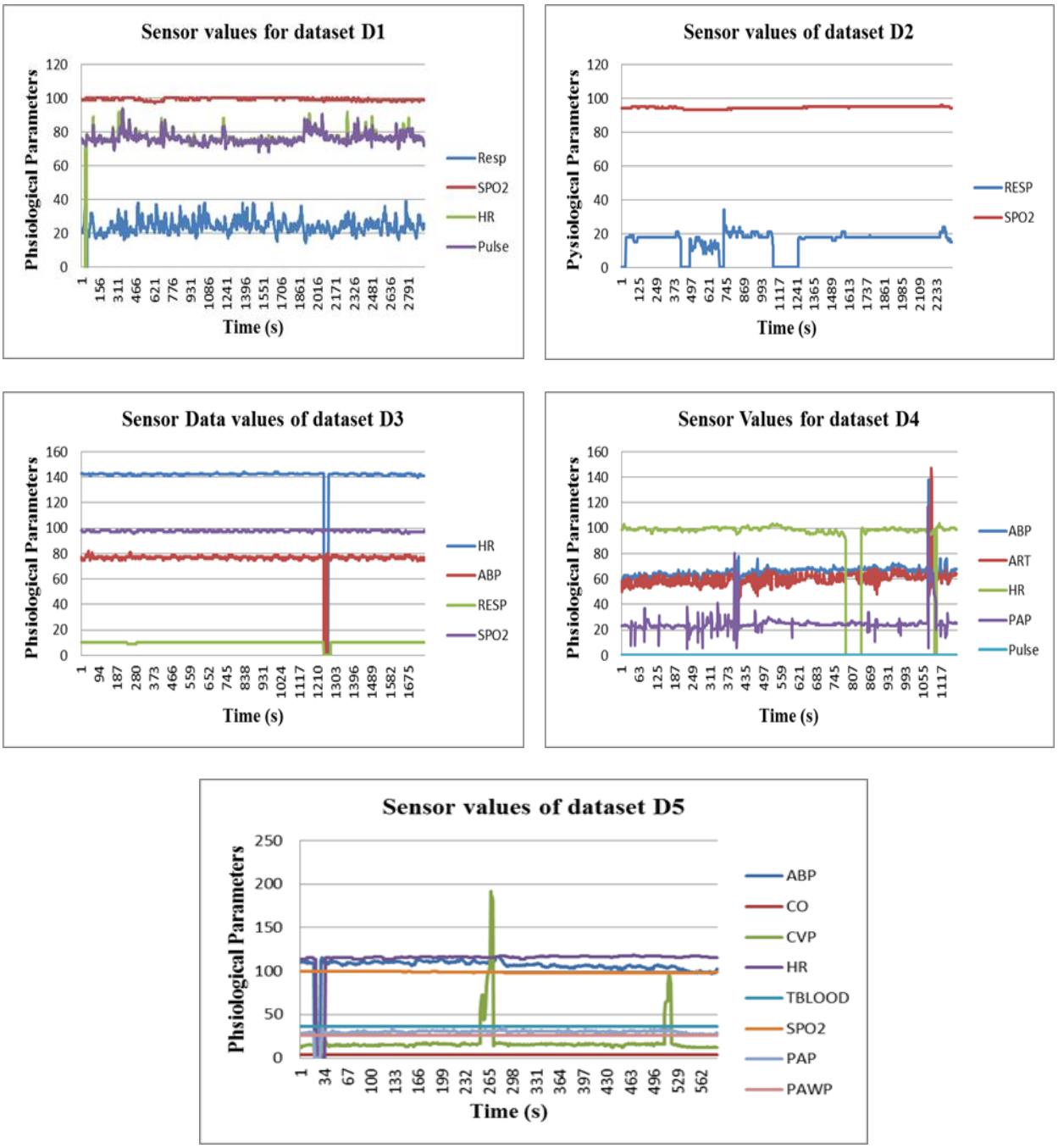


Figure 5.6: Sensor values for different datasets

Table 5.2: Datasets Used

Dataset	Record no.	Clinical Class	Measurements
D1	221	Brain injury	ABP, HR, PULSE, RESP, SpO2
D2	224	Sepsis	RESP, SpO2
D3	291	Sepsis	ABP, HR, RESP, SpO2
D4	405	Cardiogenic Shock	ABP, ART, HR, PAP, PULSE
D5	485	Trauma	ABP, C.O., CVP, HR, TBLOOD,SPO_2, PAP, PAWP

5.3.2 Selection for Optimal Value of Parameters

The size of sliding window that is compression ratio and the total count of number of clusters are two main parameters which have the considerable impact on the efficiency and accuracy of classification results. Hence, various trials are carried out to determine the suitable value of compression ratio and the number of clusters.

The suitable value of these two factors signifies that the detection results of the proposed framework are precise and robust. The results of the values of parameters are analyzed only for dataset D5. The accuracy percentage of the proposed approach for dataset D5 by varying compression ratio and number of clusters is demonstrated in Table 5.3.

Table 5.3: Accuracy %age under different k and c

Number of Clusters (*c*)	**Compression Ratio (*k*)**			
	5	10	15	20
2	97.34	96.01	95.08	94.65
3	97.01	96.33	95.68	95.01
4	98.90	98.55	97.43	96.97
5	98.99	98.67	98.00	97.54

It is perceived from the Table 5.3 that lesser the compression ratio, more the accuracy. But, with decrease in window size the number of frames increases hence computation time also increases. On the other hand, with the increase in compression ratio that is by increasing size of window the accuracy decreases. However, the effect on accuracy by increasing the number of clusters is opposite to the compression ratio. More clusters imply greater accuracy but, the time taken for computation also increases and the results are contradictory when the number of clusters decreases. Hence, in order to maintain a balance between time efficiency and accuracy, the most appropriate selected values of k and c are 10 and 4 respectively.

5.3.3 Performance Analysis

5.3.3.1 Optimality of Cluster Structure

The result of clustering depends on the predefined number of clusters. To obtain optimal value of number of clusters there exists various clustering indices which estimate goodness of clusters [154]. The Xie and Beni index is used in fuzzy clustering to check the compactness of clusters and to identify different fuzzy partitions. The index gives a validity criteria based on a function. The validity function depends on the distance between centroids of different clusters, dataset, distance matrix etc. and does not depend on the fuzzy approach used. The compactness of a cluster and intercluster distance are taken as parameters for evaluating goodness of the clustering algorithm. The XB index is calculated as follows:

$$I_{xb} = J_f / N d_{\min}^2$$

Where, I_{xb} is Xie Beni index value, J_f is fuzzy objective function, N is total number of data points and d_{min} is the minimum intercluster distance. The larger value of d_{min} signifies more separate clusters hence results in smaller value of XB index. The cluster structure is optimal in case of PCSFCM which is validated in Figure 5.7 using Xie Beni (XB) index [155]. As the value of XB

index is lowest in case of PCSFCM which signifies that the cluster structure is more optimal as compared to other two algorithms.

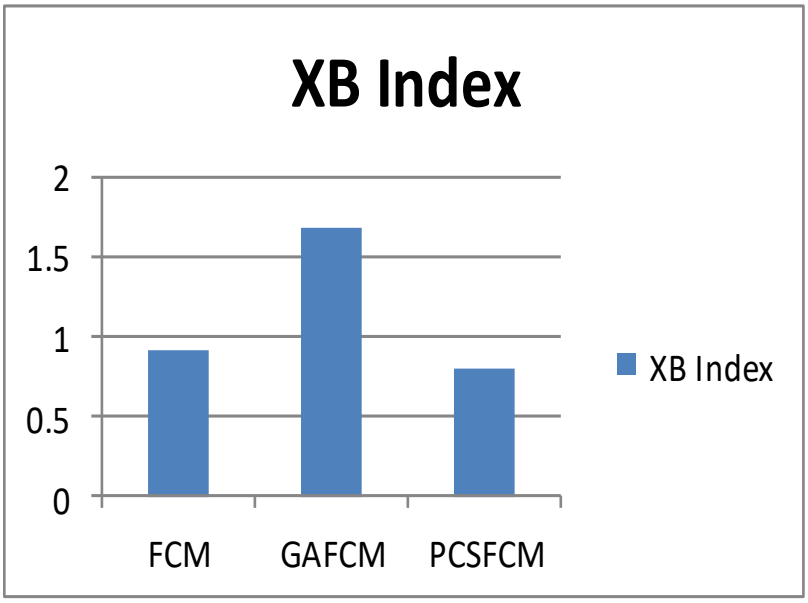


Figure 5.7: Xie Beni index for different algorithms

5.3.3.2 Accuracy

The performance of the different modules of the suggested framework is evaluated by comparing it with existing state of art approaches on the basis of detection rate and false alarm rate. The rate of detection is represented by *DR* and is calculated as:

$$DR = \frac{TP}{TP + FN}$$

where, *TP* represents the number of true positives that are the count of instances which are correctly identified as anomalous by the framework. *FN* represents false negatives that are the count of instances which are misclassified as non-anomalous by the framework. False positive rate symbolized by *FPR* and is calculated as:

$$FPR = \frac{FP}{FP + TN}$$

where, *FP* is false positives that are the count of instances which are misclassified as anomalous, *TN* is true negatives that are the count of instances which are correctly classified as non-anomalous. *TP*, *FP*, *TN*, *FN* are shown in Table 5.4. The numeric value 1 in the table represents an anomalous instance and 0 represents the instance as non-anomalous.

Table 5.4 Confusion Matrix

	TP	**FP**	**TN**	**FN**
Actual	1	0	0	1
Output	1	1	0	0

The comparison of the proposed framework with existing techniques on the basis of detection rate is shown in Figure 5.8. It is perceived from the figure that the detection rate of the proposed framework is better than the existing approaches such as fuzzy c-means, k-means with AIS, and clonal selection based fuzzy c-means.

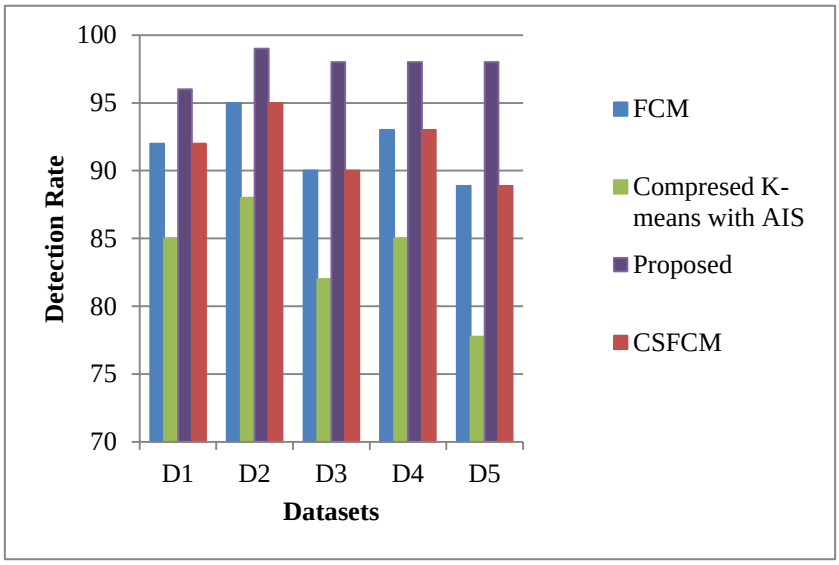


Figure 5.8: Comparison of proposed and existing approaches based on detection rate

The analysis of detection rate with respect to false alarm rate is demonstrated in Figure 5.9 and it is observed that for the identical value of the false alarm rate the accuracy of the proposed framework is much better in comparison with other methods.

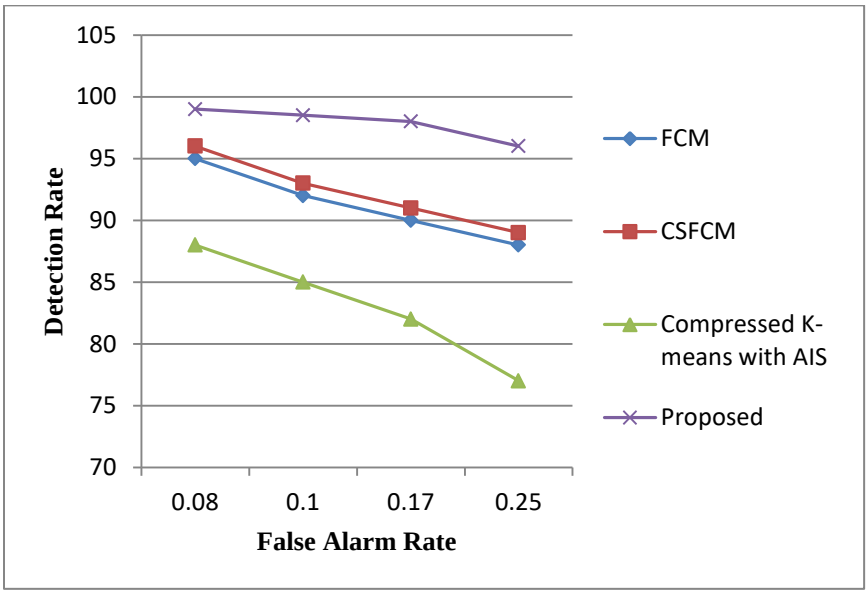


Figure 5.9: Comparison of detection rate w.r.t. False alarm rate

5.3.3.3 Scalability of Proposed Clustering Algorithm

The CSPFCM algorithm is proposed in the framework for scalability and is validated on big data by considering a dataset having size greater than the default block size of Hadoop.

The default block size in apache Hadoop is generally 64 MB. The proposed algorithm is implemented for different real datasets taken from physionet library and a comparative analysis is done with various similar existing algorithms as shown in Figure 5.10. It is perceived from the Figure 5.10 that for datasets D1 and D2 which are having data size less than default block size of Hadoop the existing approaches are equally time efficient in comparison to proposed algorithm for scalability. However as the data size increases, by using multiple worker nodes the proposed

clustering algorithm takes significant less amount of time for data clustering which is perceived using dataset D3, D4 and D5.

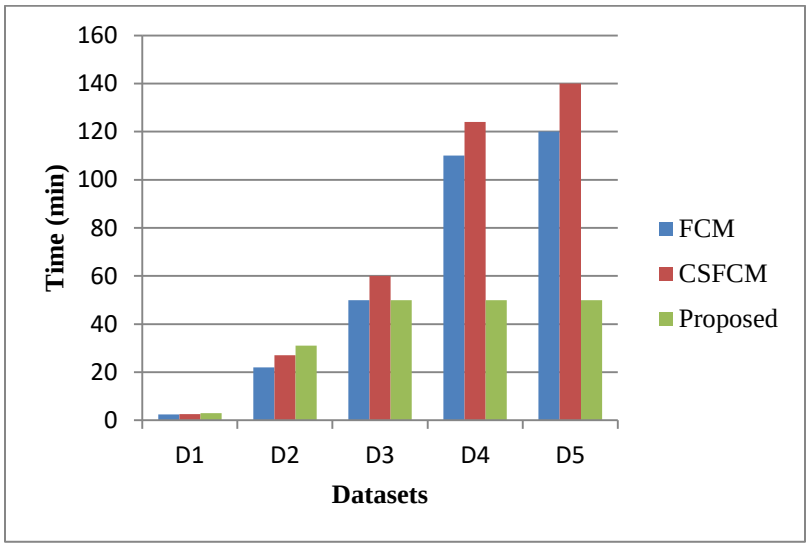


Figure 5.10: Comparison with existing clustering approaches

The analysis of the proposed algorithm is also done by varying the number of worker nodes in a Hadoop cluster. In map reduce framework every dataset is divided into chunks of size less than or equal to default block size. Each data chunk is processed by different mappers depending on the number of mappers. Here in our analysis datasets D1 and D2 are not segmented as they are having data size less than default block size of Hadoop. So, the time taken by D1 and D2 by varying the number of nodes is approximately equal as perceived from Figure 5.11. The dataset D3 is divided into two chunks hence, the time taken for processing decreases by adding second node and remains almost constant after two worker nodes. For dataset D3 and D4 number of data chunks are more than two so the time taken further decreases by increasing number of nodes. Hence, it is observed from the Figure 5.11 that the proposed algorithm is scalable and efficiently handles big data.

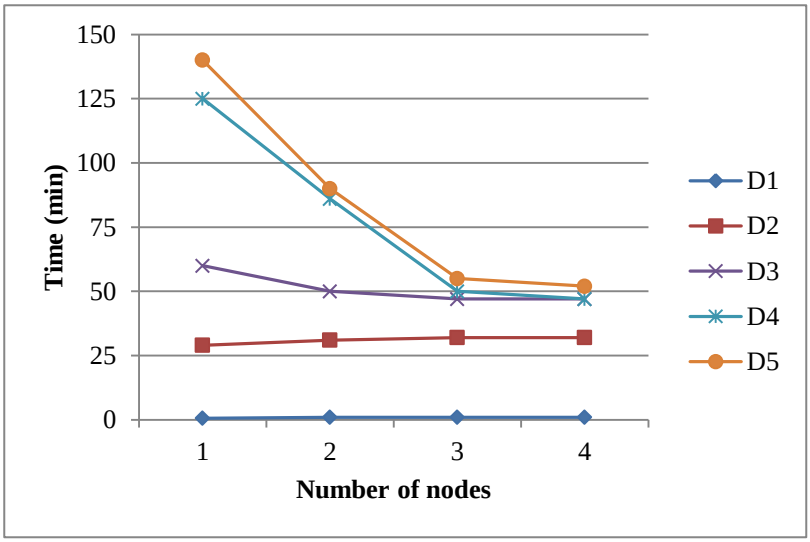


Figure 5.11: Run time with different number of nodes

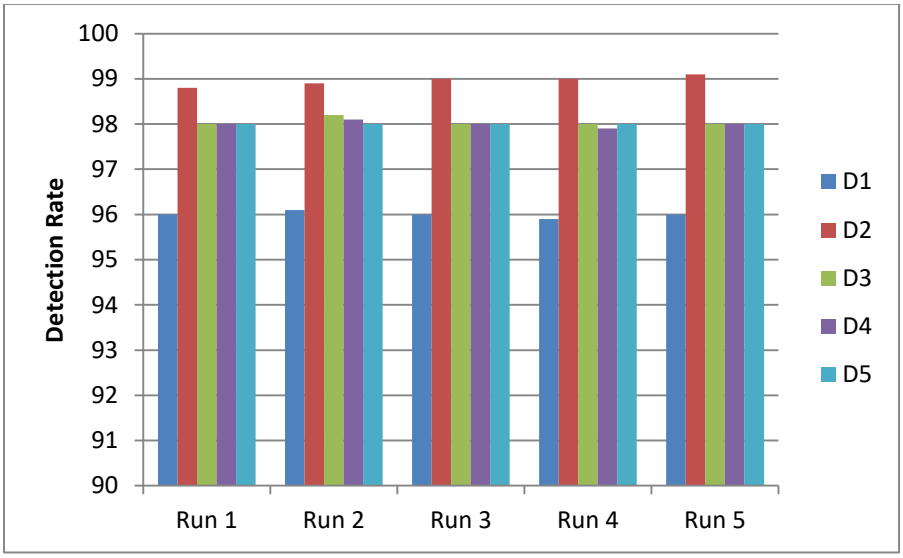


Figure 5.12: Detection rate in case of multiple runs

The consistency of the approach is shown in terms of detection rate during multiple executions. It is seen from the Figure 5.12 that for different datasets the detection rate during multiple runs remains consistent.

5.4 Chapter Summary

In this Chapter, an integrated framework for detection of anomalous data has been proposed. The framework is distributed into four phases. In the initial phase a set of subsequences are generated using a static sliding window. Then, these subsequences are compressed using modified PAA. Compressed subsequences are then clustered in second phase using proposed clustering algorithm and then the data points are classified based on their membership values. In the final phase the classified clusters are refined using proposed cluster refinement algorithm. For validation, the proposed framework is compared with existing state-of-art algorithms. The results obtained indicate that the proposed framework accomplishes high accuracy with lesser number of false alarms in comparison to its counterparts. The main edge of this approach is that it refines data after clustering and it is implemented on distributed framework which increases the classification accuracy and total time efficiency of the framework. In the next chapter the complete research work is concluded and future directions in the field of outlier detection are provided.

CHAPTER 6

CONCLUSION AND FUTURE DIRECTIONS

In this chapter the main contribution of the thesis and future directions in the field of outlier detection and data analytics are presented. The research contribution is discussed in Section 6.1 and future directions in Section 6.2.

6.1 Main Contribution

In this thesis, the main focus is on designing models and algorithms for outlier detection in wireless sensor networks. A distributed map reduce framework is used for processing large amount of data. It is seen that the sensors in real application scenarios are often correlated to one another in some manner. However, existing distributed techniques for outlier detection do not considered this aspect of data. Hence, an efficient approach based on linear correlation is proposed for finding outliers. The proposed approach uses Sequential Minimal Optimization regression for finding point outliers. After finding point outliers the contextual outliers are found considering linear correlation among attributes. The proposed approach performs better than similar existing approaches for outlier detection in terms of classification accuracy and time efficiency.

Further, it is observed that, in real applications none of the sensors in series exhibit a truly linear relationship and give unavoidable errors due to the assumption of linearity. But, the proposed approach is able to handle only linear relationships among data attributes. Hence, to overcome the above limitation the proposed approach is further enhanced to detect outliers in non-linearly

correlated datasets as well. The work assumes that collections of sensors in the datasets are linearly as well as non-linearly related. The enhanced approach works on non-linear kernel function of SMO regression. It finds both linearly and non-linearly correlated attributes using different algorithms. The approach is validated using the dataset of real medical sensors taken from Physionet library and simulated using various non-linear functions. The proposed approach is scalable and is able to detect outliers from large datasets efficiently. The performance of the proposed algorithm is compared with the existing algorithms and our previously proposed linear correlation based algorithm.

Further, an integrated framework for detection of outliers using the proposed clonal selection based Fuzzy C-means algorithm is proposed. The proposed algorithm works on the principle of clonal selection algorithm and uses the objective function of fuzzy clustering. The framework is based on data compression, data clustering, and cluster refinement. To validate the proposed framework, the experiments are performed on real and synthetic data sets. The first phase of experimentation is performed on testing the proposed clustering algorithm and comparing it with the existing state-of-art clustering algorithms. It is observed that the clusters formed by proposed clustering algorithm have more optimal structures than state of art clustering algorithms. The formed clusters are further refined using cluster refinement algorithm to increase accuracy of the outlier detection.

6.2 Future Scope

The research described in this thesis has a number of promising directions for future research.

The work might be explored to various applications like credit card fraud detection, recommendation system, malware detection etc. for obtaining interesting patterns. Further, there

are various areas in different domains that need to be touched for detection of unusual events so that one can contribute towards the betterment of the society.

Outlier detection in dynamic data

Data generated from domains like social networks, collaboration networks, weather forecast tend to change dynamically with time. Mining real time data is a very active area of research. Some examples include analyzing the properties of time-evolving sensor networks, mining dynamic datasets and growing recommendation systems.

Mining images for detecting outliers

The outlier detection can be explored by mining key patterns from image. In such applications, based on the similarity between the time evolving images interesting patters can be detected.

Evolutionary algorithms for outlier detection

Various other genetic algorithms might be explored for outlier detection to increase efficiency and accuracy of the existing approaches.

BIBLIOGRAPHY

[1] "White paper big data as a service by emc solutions group," July 2012.

[2] D. Laney, "3-d data management: Controlling data volume, velocity and variety," META Group Research Note, February 2001.

[3] A Brief Overview of Outlier Detection Techniques. available online: https://towardsdatascience.com/a-brief-overview-of-outlier-detection-techniques-1e0b2c19e561.

[4] P. N. Tan, M. Steinbach, and V. Kumar, "Introduction to data mining, Pearson Education," India, 2006.

[5] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," ACM computing surveys (CSUR), vol. 41, no. 3, pp. 15:1-58, 2009.

[6] A. L. Buczak, and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," IEEE Communications Surveys and Tutorials, vol. 18, no. 2, pp. 1153–1176, 2016.

[7] C. Stefano, C. Sansone, and M. Vento, "To reject or not to reject: that is the question– an answer in case of neural classifiers," IEEE Transactions on Systems, Management and Cybernetics, vol. 30, no. 1, pp. 84–94, 2000.

[8] D. Barbara, J. Couto, S. Jajodia, and N. Wu, "Detecting novel network intrusions using bayes estimators," In Proceedings of the SIAM International Conference on Data Mining, pp. 1-17, April 2001.

[9] B. Scholkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola,, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," Neural Computation, vol. 13, no. 7, pp. 1443–1471, 2001.

[10] V. Roth, "Outlier detection with one-class kernel fisher discriminants," Advances in Neural Information Processing Systems, pp. 1169-1176, 2005.

[11] V. Roth, "Kernel fisher discriminants for outlier detection," Neural Computation, vol. 18, no. 4, pp. 942–960, 2006.

[12] S. Boriah, V. Chandola, and V. Kumar, "Similarity measures for categorical data: A comparative evaluation," In Proceedings of the eighth SIAM International Conference on Data Mining, pp. 243–254, April 2008.

[13] A. K. Jain, and R. C. Dubes, "Algorithms for clustering data," Prentice Hall PTR, vol. 2, no. 3, pp. 283-304, 1988.

[14] S. Basu, M. Bilenko, and R. J. Mooney, "A probabilistic framework for semi-supervised clustering" In Proceedings of the tenth ACM SIGKDD International Conference on Knowledge discovery and data mining, pp. 59–68, August 2004.

[15] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," In Proceedings Knowledge Discovery and Data Mining, vol. 96, no. 34, pp. 226–231, August 1996.

[16] S. Guha, R. Rastogi, and K. Shim, "ROCK: A robust clustering algorithm for categorical attributes," Information Systems, vol. 25, no. 5, pp. 345–366, 2000.

[17] L. Ertoz, M. Steinbach, and V. Kumar, "Finding topics in collections of documents: A shared nearest neighbor approach," Network Theory and Applications, vol. 11, pp. 83–104, 2003.

[18] D. Yu, G. Sheikholeslami, and A. Zhang, “Findout: finding outliers in very large datasets,” Knowledge and Information Systems, vol. 4, no. 4, pp. 387–412, 2002.

[19] G. Sheikholeslami, S. Chatterjee, and A. Zhang, “Wavecluster: A multi-resolution clustering approach for very large spatial databases,” In Proceedings of the 24th International Conference on Very Large Data Bases, pp. 428–439, August 1998.

[20] R. Smith, A. Bivens, M. Embrechts, C. Palagiri, and B. Szymanski, “Clustering approaches for anomaly based intrusion detection,” In Proceedings of Intelligent Engineering Systems through Artificial Neural Networks, pp. 579–584, October 2002.

[21] T. Kohonen, “The self-organizing map,” Proceedings of the IEEE, vol. 78, no. 9, pp. 1464-80, 1990.

[22] K. Labib, and R. Vemuri, “Nsom: A real-time network-based intrusion detection using self-organizing maps,” Network Security, pp. 1-6, January 2002.

[23] M. Ramadas, S. Ostermann, and B. C. Tjaden, “Detecting anomalous network traffic with self-organizing maps,” In Proceedings of Recent Advances in Intrusion Detection, pp. 36–54, September 2003.

[24] W. Q. Wang, M. F. Golnaraghi, and F. Ismail, “Prognosis of machine health condition using neuro-fuzzy systems,” Mechanical Systems and Signal Processing, vol. 18, no. 4, pp. 813-831, 2004.

[25] A. Ypma, and R. P. W. Duin, “Novelty detection using self-organizing maps,” In Progress in Connectionist Based Information Systems, vol. 2, pp. 1322–1325, November 1998.

[26] V. Emamian, M. Kaveh, and A. Tewfik, “Robust clustering of acoustic emission signals using the kohonen network,” In Proceedings of the IEEE International Conference of Acoustics, Speech and Signal Processing, vol. 6, pp. 3891-3894, 2000.

[27] P. L. Brockett, X. Xia, and R. A. Derrig, “Using kohonen’s self-organizing feature map to uncover automobile bodily injury claims fraud,” Journal of Risk and Insurance, vol. 65, no. 2, pp. 245–274, 1998.

[28] D. Barbara, Y. Li, J. Couto, J. L. Lin, and S. Jajodia, “Bootstrapping a data mining intrusion detection system,” In Proceedings of the 2003 ACM symposium on Applied computing, pp. 421–425, March 2003.

[29] Z. He, S. Deng, and X. Xu, “Outlier detection integrating semantic knowledge,” In Proceedings of the International Conference on Advances in Web-Age Information Management, pp. 126–131, August 2002.

[30] Z. He, X. Xu, and S. Deng, “Discovering cluster-based local outliers,” Pattern Recognition Letters, vol. 24, no. 9-10, pp. 1641–1650, 2003.

[31] A. Chaudhary, A. S. Szalay, and A. W. Moore, “Very fast outlier detection in large multidimensional data sets,” In Proceedings of ACM SIGMOD Workshop in Research Issues in Data Mining and Knowledge Discovery (DMKD), June 2002.

[32] P. N. Tan, M. Steinbach, and V. Kumar, “Introduction to Data Mining,” Addison-Wesley, 2005.

[33] R. O. Duda, P. E. Hart, and D. G. Stork, “Pattern Classification (2nd Edition),” Wiley Interscience, 2000.

[34] E. Eskin, “Anomaly detection over noisy data using learned probability distributions,” In Proceedings of the Seventeenth International Conference on Machine Learning, pp. 255–262, July 2000.

[35] M. J. Desforges, P. J. Jacob, and J. E. Cooper, “Applications of probability density estimation to the detection of abnormal conditions in engineering,” In Proceedings of Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, vol. 212, no. 8, pp. 687–703, August 1998.

[36] M. Li, and P. Vitanyi, “An Introduction to kolmogorov complexity and its applications,” Springer, 2013.

[37] A. Arning, R. Agrawal, and P. Raghavan, “A linear method for deviation detection in large databases,” Knowledge Discovery and Data, vol. 1141, no. 50, pp. 972-981, 1996.

[38] E. Keogh, S. Lonardi, and C. A. Ratanamahatana,, “Towards parameter-free data mining,” In Proceedings of the 10th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp. 206–215, August 2004.

[39] W. Lee, and D. Xiang, “Information-theoretic measures for anomaly detection,” In Proceedings of the IEEE Symposium on Security and Privacy, IEEE Computer Society, pp. 130-143, May 2000.

[40] S. Ando, “Clustering needles in a haystack: An information theoretic analysis of minority and outlier detection,” In Proceedings of 7th International Conference on Data Mining, IEEE, pp. 13–22, October 2007.

[41] Z. He, S. Deng, X. Xu, and J. Z. Huang, “A fast greedy algorithm for outlier mining,” In Proceedings of 10th Pacific-Asia Conference on Knowledge and Data Discovery, pp. 567–576, April 2006.

[42] F. J. Anscombe, and I. Guttman, “Rejection of outliers,” Technometrics, vol. 2, no. 2, pp. 123–146, 1960.

[43] I. T. Jolliffe, “2nd edition Principal Component Analysis,” Springer, 2002.

[44] L. Parra, G. Deco, and S. Miesbach, “Statistical independence and novelty detection with information preserving nonlinear maps,” Neural Computation, vol. 8, no. 2, pp. 260–269, 1996.

[45] M. L. Shyu, S.C. Chen, K. Sarinnapakorn, and L. Chang, “A novel anomaly detection scheme based on principal component classifier,” In Proceedings of ICDM Foundation and New Direction of Data Mining workshop, pp. 172-179, January 2003.

[46] P. J. Huber, “Robust Statistics,” Springer, 2011.

[47] H. Dutta, C. Giannella, K. Borne, and H. Kargupta, “Distributed top-k outlier detection in astronomy catalogs using the demac system,” In Proceedings of 7th SIAM International Conference on Data Mining, pp. 473-478, April 2007.

[48] T. Ide, and H. Kashima, “Eigen space-based anomaly detection in computer systems,” In Proceedings of the 10th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 440–449, August 2004.

[49] R. Fujimaki, T. Yairi, and K. Machida, “An approach to spacecraft anomaly detection problem using kernel feature space,” In Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, pp. 401–410, August 2005.

[50] P. S. Bishnu and V. Bhattacherjee, “Software fault prediction using quad tree-based k-means clustering algorithm,” IEEE Transactions on knowledge and data engineering, vol. 24, no. 6, pp. 1146-50, 2012.

[51] A. L. Buczak and E. Guven, “A survey of data mining and machine learning methods for cyber security intrusion detection,” IEEE Communications Surveys and Tutorials, vol. 18, no. 2, pp. 1153–1176, 2016.

[52] R. J. Bolton, and D. J. Hand, “Statistical fraud detection: A review,” Statistical Science, vol. 17, no. 3, pp. 235–249, 2002.

[53] O. Salem, A. Guerassimov, A. Mehaoua, A. Marcus, and B. Furht, “Anomaly detection in medical wireless sensor networks using svm and linear regression models,” International Journal of E-Health and Medical Communications (IJEHMC), vol. 5, no. 1, pp. 20–45, 2014.

[54] A. Tabrizi, L. Garibaldi, A. Fasana, and S. Marchesiello, “Early damage detection of roller bearings using wavelet packet decomposition, ensemble empirical mode decomposition and support vector machine,” Meccanica, vol. 50, no. 3, pp. 865–874, 2015.

[55] Y. Xu, Z. Wu, J. Li, A. Plaza, and Z. Wei, “Anomaly detection in hyperspectral images based on low-rank and sparse representation,” IEEE Transactions on Geoscience and Remote Sensing, vol. 54, no. 4, pp. 1990–2000, 2016.

[56] D. Thom, H. Bosch, S. Koch, M. Worner, and T. Ertl, “Spatiotemporal anomaly detection through visual analysis of geolocated twitter messages,” In Proceedings of the IEEE Pacific Visualization Symposium (PacificVis), pp. 41–48, IEEE, February 2012.

[57] Y. Zhang, N. Meratnia, and P. J. Havinga, “Outlier detection techniques for wireless sensor networks: A survey.” IEEE Communications Surveys and Tutorials, vol. 12, no. 2, pp. 159–170, 2010.

[58] V. Chatzigiannakis, S. Papavassiliou, M. Grammatikou, and B. Maglaris, “Hierarchical anomaly detection in distributed large-scale sensor networks,” In Proceedings of the 11th IEEE Symposium on Computers and Communications, ISCC’06, pp. 761–767, IEEE, June 2006.

[59] “Crossbow Technology Inc., MICAZ/ZigBee Series (MPR2400),” http://www.willow.co.uk/html/.

[60] “Moteiv, i.; tmote sky,” http://www.snm.ethz.ch/Projects/TmoteSky.

[61] Q. Sun, F. Hu, and Q. Hao, “Mobile target scenario recognition via low-cost pyroelectric sensing system: Toward a context-enhanced accurate identification,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 44, no. 3, pp. 375–384, 2014.

[62] W. K. Wong, A. W. Moore, G. F. Cooper, and M. M. Wagner, “Bayesian network anomaly pattern detection for disease outbreaks,” In Proceedings of the 20th International Conference on Machine Learning (ICML-03), pp. 808–815, Washington 2003.

[63] J. Lin, E. Keogh, A. Fu, and H. V. Herle, “Approximations to magic: Finding unusual medical time series,” In the proceedings of the 18th IEEE Symposium on Computer-Based Medical Systems, pp. 329–334, IEEE, 2005.

[64] S. B. Kotsiantis, “ Supervised machine learning: A review of classification techniques,” In Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies, pp. 3–24, June 2007.

[65] T. Hastie, R. Tibshirani, and J. Friedman “The Elements of Statistical Learning: Data Mining, Inference, and Prediction,”Springer Series in Statistics, 2009.

[66] Committee on the Analysis of Massive Data; Committee on Applied, Theoretical Statistics; Board on Mathematical Sciences, Their Applications; Division on Engineering, and Physical Sciences; National Research Council. Frontiers in Massive Data Analysis. The National Academies Press, 2013.

[67] T. Hill, and P. Lewicki, “Statistics: methods and applications: a comprehensive reference for science, industry, and data mining,” StatSoft, 2007.

[68] Y. K. Ibrahim, “A Novel Algorithm to Estimate the Reliability of Hybrid Computer Communication Networks,” Journal of Signal and Information Processing, vol. 4, no. 4, pp. 394, 2013.

[69] F. Zhang, H. Liu, Y. W. Leung, X. Chu, and B. Jin, “CBS: Community-based bus system as routing backbone for vehicular ad hoc networks,” IEEE Transactions on Mobile Computing, vol. 16, no. 8, pp. 2132-46, 2017.

[70] N. Bele, P. K. Panigrahi, and S. K. Srivastava, “Knowledge discovery from vernacular expressions: An application of social media and sentiment mining,” International Journal of Knowledge Management (IJKM), vol. 14, no. 1, pp. 1-8, 2018.

[71] M. Agyemang, K. Barker, and R. Alhajj, “Framework for mining web content outliers,” In Proceedings of the 2004 ACM symposium on Applied computing, pp. 590-594, March 2004.

[72] A. Mahapatro, and P. M. Khilar, “Fault diagnosis in body sensor networks,” International Journal of Computer Information Systems and Industrial Management Applications vol. 5, pp. 252–259, 2012.

[73] S. Hore, S. Chatterjee, V. Santhi, N. Dey, A. S. Ashour, V. E. Balas, and F. Shi “Indian sign language recognition using optimized neural networks,” Information Technology and Intelligent Transportation Systems, AISC, vol. 455, pp. 553-563, 2017.

[74] D. J. Hill, and B. S. Minsker, “Anomaly detection in streaming environmental sensor data: A data-driven modeling approach,” Environmental Modelling and Software, vol. 25, no. 9, pp. 1014–1022, 2010.

[75] F. Liu, X. Cheng, and D. Chen, “Insider attacker detection in wireless sensor networks,” In Proceedings of 26th IEEE International Conference on Computer Communications, pp. 1937–1945, May 2007.

[76] N. Srivastava, and J. Srivastava, “A hybrid-logic approach towards fault detection in complex cyber-physical systems”, In Proceedings of Annual Conference of the Prognostics and Health Management Society, pp. 1-11, October 2010.

[77] Y. Yao, A. Sharma, L.Golubchik, and R.Govindan, “Online anomaly detection for sensor systems: A simple and efficient approach,” Performance Evaluation, vol. 67, no. 11, pp. 1059-1075, 2010.

[78] O. Salem, Y. Liu, and A. Mehaoua, “Anomaly detection in medical wireless sensor networks,” Journal of Computing Science and Engineering, vol. 7, no. 4, pp. 272–284, 2013.

[79] O. Salem, A.Guerassimov, A.Mehaoua, A. Marcus, and B. Furht, “Sensor fault and patient anomaly detection and classification in medical wireless sensor networks,” In Proceedings of IEEE International Conference on Communications (ICC), pp. 4373-4378, June 2013.

[80] M. Hall, I. Witten, and E. Frank, “Data mining: practical machine learning tools and techniques,” Morgan Kaufmann Publishers: Burlington, MA, USA, October 2011.

[81] X. Cheng, J.Xu, J. Pei, and J. Liu, “Hierarchical distributed data classification in wireless sensor networks,” Computer Communications, vol. 33, no. 12, pp. 1404-1413, 2010.

[82] A. Shilton, S. Rajasegarar, and M. Palaniswami, “Combined multiclass classification and anomaly detection for large-scale wireless sensor networks,” In Proceedings of IEEE Eighth International Conference on Intelligent Sensors, Sensor Networks and Information Processing, pp. 491–496, April 2013.

[83] M. A. Hayes, and M. A. Capretz, “Contextual anomaly detection in big sensor data,” In Proceedings of IEEE International Congress on Big Data, pp. 64-71, June 2014.

[84] S. A. Haque, M. Rahman, and S. M. Aziz, “Sensor anomaly detection in wireless sensor networks for healthcare,” Sensors, vol. 15, no.4, pp. 8764-8786, 2015.

[85] N. M. Nasrabadi, “A nonlinear kernel-based joint fusion/detection of anomalies using hyperspectral and sar imagery,” Image Processing, In Proceedings of 15th IEEE International Conference on, pp. 1864–1867, October 2008.

[86] H. Yu, X. J. Zhang, S. Wang, and S. M. Song, “Alternative framework of the gaussian filter for non-linear systems with synchronously correlated noises,” IET Science, Measurement and Technology, vol. 10, no. 4, pp. 306–315, 2016.

[87] J. R. Lee, S.K. Ye, and H.D. J. Jeong, “Detecting anomaly teletraffic using stochastic self-similarity based on hadoop,” In Proceedings of 16th International Conference on Network-Based Information Systems, pp. 282-287, September 2013.

[88] F. Ye, Z. Zhang, K. Chakrabarty, and X. Gu, “Board-level functional fault diagnosis using artificial neural networks, support-vector machines, and weighted-majority voting,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 32, no. 5, pp. 723–736, 2013.

[89] F. Ye, S. Jin, Z. Zhang, K. Chakrabarty, and X. Gu, “Handling missing syndromes in board-level functional-fault diagnosis,” 22nd Asian Test Symposium (ATS), IEEE, pp. 73–78, 2013.

[90] M. Moshtaghi, C. Leckie, S. Karunasekera, and S. Rajasegarar, “An adaptive elliptical anomaly detection model for wireless sensor networks,” Computer Networks, vol. 64, pp. 195-207, 2014.

[91] A. Patcha, and J. M. Park, “An overview of anomaly detection techniques: Existing solutions and latest technological trends,” Computer Networks, vol. 51, no. 12, pp. 3448–3470, 2007.

[92] S. Zhong, H. Luo, L. Lin, and X. Fu, “An improved correlation-based anomaly detection approach for condition monitoring data of industrial equipment,” In Proceedings of International Conference on Prognostics and Health Management (ICPHM), IEEE, pp. 1-5, June 2016.

[93] B. Saneja, and R. Rani., “An efficient approach for outlier detection in big sensor data of health care,” International Journal of Communication Systems, vol. 30, no. 17, p. e3352, 2017.

[94] H. Zhang, J. Liu, and R. Li, “Fault detection for medical body sensor networks under bayesian network model,” 11th International Conference on Mobile Ad-hoc and Sensor Networks (MSN), IEEE, pp. 37–42, December 2015.

[95] D. J. Kim, and B. Prabhakaran, “Motion fault detection and isolation in body sensor networks,” Pervasive and Mobile Computing, vol. 7, no. 6, pp. 727–745, 2011.

[96] A. J. Smola, and B. Scholkopf, “A tutorial on support vector regression,” Statistics and Computing, vol. 14, no. 3, pp. 199–222, 2004.

[97] J. Platt, “Fast training of support vector machines using sequential minimal optimization,” Advances in Kernel Methods Support Vector Learning, MIT Press, Cambridge, MA, pp. 185–208, 1999.

[98] O. Salem, Y. Liu, and A. Mehaoua, “Detection and isolation of faulty measurements in medical wireless sensor networks,” First International Symposium on Future Information and Communication Technologies for Ubiquitous HealthCare (Ubi-HealthTech), IEEE, pp. 1–5, July 2013.

[99] A. Naseem, O. Salem, Y. Liu, and A. Mehaoua, “Reliable vital sign collection in medical wireless sensor networks,” 15th International Conference on e-Health Networking, Applications & Services (Healthcom), IEEE, pp. 289–293, October 2013.

[100] A. Peiravi, “Connectance and reliability computation of wireless body area networks using signal flow graphs,” Life Science Journal, vol. 7, no. 2, pp. 52–56, 2010.

[101] D. Arthur, “K-means ++ : The Advantages of Careful Seeding,” In Proceedings of the 18th annual ACM-SIAM symposium on Discrete algorithms, pp. 1027–1035, January 2007.

[102] A. Bavelas, “Communication patterns in task oriented groups,” The Journal of the Acoustical Society of America, vol. 22, no. 6, pp. 725–730, 1950.

[103] L. Liu, L. Sun, S. Chen, M. Liu, and J. Zhong, “K -PRSCAN: A clustering method based on PageRank,” Neurocomputing, vol. 175, pp. 65–80, 2015.

[104] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, “An efficient k -means clustering algorithm: Analysis and implementation,” IEEE Transactions on Pattern Analysis and Machine Learning, vol. 24, no. 7, pp. 881–892, 2002.

[105] B. Bahmani, B. Moseley, A. Vattani, R. Kumar, and S. Vassilvitskii, "Scalable KMeans++," In Proceedings of 38th International Conference on Very Large Data Bases, pp. 622–633, March 2012.

[106] Y. Xu, W. Qu, Z. Li, G. Min, K. Li, and Z. Liu, "Efficient k-means ++ approximation with mapreduce," IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 12, pp. 3135–3144, 2014.

[107] J. C. Dunn, "A Fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," Journal of Cybernetic, vol. 3, no. 3, pp. 32–57, 1973.

[108] J. C. Bezdek, "Modified objective function algorithms pattern recognition with fuzzy objective function algorithms," Springer, pp. 155–201, 1981.

[109] S. Gregory, "Finding overlapping communities in networks by label propagation," New Journal of Physics, vol. 12, no. 10, pp. 1–21, 2010.

[110] P. Hu, K. C. C. Chan, and T. He, "Deep Graph Clustering in Social Network.," In Proceedings of the 26th International Conference on World Wide Web Companion, pp. 1425–1426, April 2017.

[111] K. Zhang, and X. W. Chen, "Large-scale deep belief nets with mapreduce," IEEE Access, vol. 2, pp. 395–403, 2014.

[112] A. Zafar, and S. H. Hasan, "A novel scheme for information retrieval from e-learning repository," Malaysian Journal of Computer Sciences, vol.28, no. 1, pp. 16-27, 2015.

[113] T. Nepusz, A. Petrczi, L. Ngyessy, and F. Bazs, “Fuzzy communities and the concept of bridgeness in complex networks,” Physical Review E - Statistical, Nonlinear, and Soft Matter Physics, vol. 77, no. 1, pp. 1–13, 2008.

[114] S. Wikaisuksakul, “A multi-objective genetic algorithm with fuzzy c-means for automatic data clustering,” Applied Soft Computing, vol. 24, pp. 679–691, 2014.

[115] E. Egrioglu, C. Hakan, and U. Yolcu, “Fuzzy time series forecasting with a novel hybrid approach combining fuzzy c-means and neural networks,” Expert Systems With Applications, vol. 40, no. 3, pp. 854–857, 2013.

[116] M. A. Khalilia, J. Bezdek, M. Popescu, and J. M. Keller, “Improvements to the relational fuzzy c -means clustering algorithm,” Pattern Recognition, vol. 47, pp. 3920–3930, 2014.

[117] C. Vehlow, S. Member, T. Reinhardt, D. Weiskopf, and I. C. Society, “Visualizing fuzzy overlapping communities in networks,” IEEE Transactions On Visualization And Computer Graphics, vol. 19, no. 12, pp. 2486–2495, 2013.

[118] J. S. Zhang, and Y. W. Leung, “Robust clustering by pruning outliers,” IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 33, no. 6, pp. 983-998, 2003.

[119] X. Chen, “A new clustering algorithm based on near neighbor influence,” Expert Systems with Applications, vol. 42, no. 21, pp. 7746–7758, 2015.

[120] S. A. Ludwig, “MapReduce based fuzzy c-means clustering algorithm: implementation and scalability,” International Journal of Machine Learning and Cybernetics, vol. 6, no. 6, pp. 923–934, 2015.

[121] S. Gregory, “Finding overlapping communities in networks by label propagation,” New Journal of Physics, vol. 12, no. 10, pp. 1–21, 2010.

[122] G. Münz, S. Li, and G. Carle, “Traffic anomaly detection using k-means clustering,” GI/ITG Workshop MMBnet, 2007.

[123] I. Kiss, B. Genge, P. Haller, and G. Sebestyen, “Data clustering-based anomaly detection in industrial control systems,” In Proceedings of IEEE 10th International Conference on Intelligent Computer Communication and Processing (ICCP), pp. 275-281, September 2014.

[124] D. Dasgupta, and S. Forrest, “Novelty detection in time series data using ideas from immunology,” In Proceedings of International Conference on Intelligent Systems, pp. 82-87, June 1996.

[125] H. Izakian, and W. Pedrycz, “Anomaly detection in time series data using a fuzzy c-means clustering,” in Proceedings of Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS), pp. 1513-1518, June 2013.

[126] A. Bhargava, and A. Raghuvanshi, “Anomaly detection in wireless sensor networks using s-transform in combination with SVM,” In Proceedings of 5th International Conference on Computational Intelligence and Communication Networks, pp. 111-116, September 2013.

[127] X. Guo, D. Wang, and F. Chen, “An anomaly detection based on data fusion algorithm in wireless sensor networks,” International Journal of Distributed Sensor Networks, vol. 11, no. 5, pp. 1-10, 2015.

[128] X. D. Hoang, J. Hu, and P. Bertok, “A program-based anomaly intrusion detection scheme using multiple detection engines and fuzzy inference,” Journal of Network and Computer Applications, vol. 32, no. 6, pp. 1219–1228, 2009.

[129] M. Y. Su, “Using clustering to improve the KNN-based classifiers for online anomaly network traffic identification,” Journal of Network and Computer Applications, vol. 34, no. 2, pp. 722–730, 2011.

[130] I. F. Akyildiz, and M. C. Vuran, “Wireless sensor networks,” John Wiley and Sons, vol. 4, 2010.

[131] M. P. Đurišić, Z. Tafa, G. Dimić, and V. Milutinović, “A survey of military applications of wireless sensor networks,” In Proceedings of Mediterranean Conference on Embedded Computing (MECO), IEEE, pp. 196-199, June 2012.

[132] B. Warneke, M. Last, B. Liebowitz, and K. S. Pister, “Smart dust: communicating with a cubic-millimeter computer,” Computer, vol. 34, no. 1, pp. 44-51, 2001.

[133] S. Srivastava, M. Singh, and S. Gupta, “Wireless sensor network: A survey,” In Proceedings of IEEE International Conference on Automation and Computational Engineering (ICACE), pp. 159-163, October 2018.

[134] T. He, S. Krishnamurthy, L. Luo, T. Yan, L. Gu, R. Stoleru, G. Zhou, Q. Cao, P. Vicaire, J. A. Stankovic, and T. F. Abdelzaher, “VigilNet: An integrated sensor network system for energy-efficient surveillance,” ACM Transactions on Sensor Networks (TOSN), vol. 2, no. 1, pp. 1-38, 2006.

[135] G. Xu, W. Shen, and X. Wang, “Applications of wireless sensor networks in marine environment monitoring: A survey,” Sensors, vol. 14, no. 9, pp. 16932-16954, 2014.

[136] M. T. Lazarescu, “Design of a WSN platform for long-term environmental monitoring for IoT applications,” IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol. 3, no. 1, pp. 45-54, 2013.

[137] L. Catarinucci, D. De Donno, L. Mainetti, L. Palano, L. Patrono, M. L. Stefanizzi, and L. Tarricone, “An IoT-aware architecture for smart healthcare systems,” IEEE Internet of Things Journal, vol. 2, no. 6, pp. 515-26, 2015.

[138] S. Hussain, S. Schaffner, and D. Moseychuck, “Applications of wireless sensor networks and rfid in a smart home environment,” In Proceedings of 2009 Seventh Annual Communication Networks and Services Research Conference, IEEE, pp. 153-157, May 2009.

[139] K. Yang, Wireless Sensor Networks: Principles and Applications, Springer, 2014.

[140] R. V. Arvind, R. R. Raj, and N. K. Prakash, “Industrial automation using wireless sensor networks,” Indian Journal of Science and Technology, vol. 9, no. 11, pp. 1-8, 2016.

[141] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, “Health monitoring of civil infrastructures using wireless sensor networks,” In Proceedings of the 6th international conference on Information processing in sensor networks, pp. 254-263, April 2007.

[142] S. Singh, R. Garg, and P. K. Mishra, "Performance optimization of map reduce based apriori algorithm on hadoop cluster," Computers & Electrical Engineering, vol. 67, pp. 348-64, 2018.

[143] V. Vapnik, "The nature of statistical learning theory," Springer, 1995.

[144] E. Osuna, R. Freund, and F. Girosi, "An improved training algorithm for support vector machines," In Proceedings of IEEE NNSP'97, pp. 276-285, September 1997.

[145] B. Schölkopf, C. J. Burges, and A. J. Smola, "Advances in Kernel methods—support vector learning," Cambridge, MIT Press, 1999.

[146] SPSS Tutorials: Pearson Correlation, Retrieved 2017-05-14

[147] Y. Wang, Y. Li, H. Cao, M. Xiong, Y. Y. Shugart, and L. Jin, "Efficient test for nonlinear dependence of two continuous variables," BMC Bioinformatics, vol. 16, no. 1, pp. 260:1–8, 2015.

[148] P. Good, "Permutation tests," Springer, 2000.

[149] Physionet. available online:. http://www.physionet.org/physiobank/database/mimicdb/.

[150] L. N. De Castro, and F. J. Von Zuben, "Learning and optimization using the clonal selection principle," IEEE Transactions on Evolutionary Computation, vol. 6, no. 3, pp. 239-251, 2002.

[151] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The hadoop distributed file system", In Proceedings of IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST), pp. 1-10, May 2010.

[152] Hadoop official site: http://hadoop.apache.org/core/.

[153] J. Lin, E. Keogh, S. Lonardi, and B. Chiu “A symbolic representation of time series, with implications for streaming algorithms,” In Proceedings of 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery – DMKD, pp. 2-11, June 2003.

[154] M. K. Pakhira, S. Bandyopadhyay, and U. Maulik, “Validity index for crisp and fuzzy clusters”, Pattern Recognition, vol. 37, no. 3, pp. 487-501, 2004.

[155] N. R. Pal, and J. C. Bezdek, “On cluster validity for the fuzzy c-means model,” IEEE Transactions on Fuzzy Systems, vol. 3, no. 3, pp. 370-379, 1995.

LIST OF PUBLICATIONS

Journals

1) Bharti Saneja and Rinkle Rani, "An efficient approach for outlier detection in big sensor data of health care", *International Journal of Communication Systems*, vol. 30, no. 17, pp. e3352:1-10, 2017. DOI: 10.1002/dac.3352 [SCIE Indexed, Impact Factor: 1.717]
2) Bharti Saneja and Rinkle Rani, "An integrated framework for anomaly detection in big data of medical wireless sensors", *Modern Physics Letters B*, vol. 32, no. 24, pp. 1850283:1-19, 2018. DOI: 10.1142/S0217984918502834 [SCI Indexed, Impact Factor: 0.731]
3) Bharti Saneja and Rinkle Rani, "A Scalable Correlation based Approach for outlier detection in wireless body sensor networks", *International Journal of Communication Systems*, vol. 32, no. 07, pp. e3918:1-15, 2019. DOI: 10.1002/dac.3918 [SCIE Indexed, Impact Factor: 1.717]
4) Bharti Saneja and Rinkle Rani, "Sensor based Intoxication Detection in Youths using Fuzzy Logic", *IETE Journal of Research*, SCIE Indexed, Impact factor: 0.829. [Under Review]

Conferences

1) Vandana Bhatia, Bharti Saneja and Rinkle Rani, "INGC: Graph Clustering & Outlier Detection algorithm using Label Propagation", in the proceedings of *IEEE International Conference on Machine Learning and Data Science*, pp. 68-74, 13-15 December 2017. DOI: 10.1109/MLDS.2017.14
2) Bharti Saneja and Rinkle Rani, "A Hybrid Approach for Outlier Detection in Weather Sensor Data", in the proceedings of *IEEE 8th International Advance Computing Conference (IACC 2018)*, pp. 321-326, 14-15 December 2018. DOI: 10.1109/IADCC.2018.8692127

3) Bharti Saneja and Rinkle Rani, “Clonal selection based parallel fuzzy clustering using map-reduce”, in the proceedings of *IEEE Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, pp. 442-447, 20-22 December 2018. [Scopus indexed]