Spring 5-1-2016

Design and Development of Data Fusion-based Approach to Minimize the False Alarm Rate in ICU

Sarin Shrestha
Dakota State University

Design and Development of Data Fusion-based Approach to Minimize the False Alarm Rate in ICU

By:

Sarin Shrestha

B.E., Tribhuvan University (2006)

M.S., Fairfield University (2010)

A dissertation submitted to the Faculty of the Graduate School of the College of Business & Information Systems at Dakota State University in partial fulfillment of the requirements for the degree of Doctor of Science in Information Systems

2016

Dissertation Approval Form

This thesis entitled: Design and Development of Data Fusion-based Approach to Minimize the False Alarm Rate in ICU written by Sarin Shrestha has been approved for the College of Business and Information Systems, Dakota State University.

Dr. Surendra Sarnikar (Chair)

Dr. Jun Liu

Dr. Viki Johnson

Dr. Dorine Bennett

Acknowledgements

I could not have accomplished this work without the blessings and support of numerous faculties, friends, and family members. I would like to start by thanking my dissertation chair, Dr. Surendra Sarnikar, on whom I could always count for the excellent guidance and support from him throughout my years as a doctoral student. I can never thank him enough for being so patient with me and for his contributions and suggestions that tremendously helped shape this study. He has been a great mentor for me!

I would also like to extend my sincere thanks to committee members Dr. Jun Liu, Dr. Dorine Bennett and Dr. Viki Johnson who helped me during different stages of this study. I would also like to thank my friends for providing all the encouragement, affections and for helping me through out my life and this program.

This was certainly unachievable without the unfaltering support of my Mom and Dad. I thank them for all their lifetime blessings, teachings, love, and support. To my wife, Ranju, for being there for me all the time! I thank her for providing all the unconditional love, and support that kept me going every day and making every moment of my life worth living!

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

The rapid advancement in technology is being leveraged in healthcare to develop many innovative systems for measuring and monitoring health with a potential for providing higher quality of care due to better information availability and decision support. Clinical Decision Support Systems (CDSS) support clinical decision making by providing specific guidance based on clinical input and real-time patient physiological data. However, there is limited integration of the data from multiple monitoring devices into a unified clinical decision support infrastructure. Clinicians employ multiple devices in intensive care unit (ICU) to access information from multiple sensors and devices to monitor and understand patient health status. However, as the number of monitoring devices increases, the number of false alarms also increases (Kacmarek & Tobin, 1998). The information from individual devices is often processed and displayed using patient monitoring information systems. The patient monitoring information systems generate alarms to alert clinicians so that they can provide immediate attention to a patient in need. These alarms are designed to make the clinician aware of the condition of a patient, but false alarms occur frequently. These frequent false alarms result in alarm fatigue and reduce the probability of clinician to respond to the alarms. In healthcare, it is known as the cry wolf effect (Breznitz, 1984). Alarm fatigue may occur when the number of alarms overwhelm providers because of the false alarms, technical problem in alarms, inappropriate alarm settings, inappropriate protocols for inactivation, and over utilization of physiologic monitoring devices (Cvach, 2012; Graham & Cvach, 2010; Shrestha, Sarnikar, & Timsina, 2013).

In healthcare, false alarms are defined as alarms without clinical relevance or alarms without therapeutic consequence. Numerous studies (Chambrin et al., 1999; Lawless, 1994; Siebig, Kuhls, Imhoff, Langgartner, et al., 2010; Tsien & Fackler, 1997) acknowledge high rate of false alarms in ICU led to care disruption and an increase in the workload of ICU staff (Allen &

Murray, 1996) and eventually to alarm desensitization (Chambrin, 2001; Drew et al., 2014). Lawless (1994) identified that up to 94% of the alarms are false in ICU and Siebig, Kuhls, Imhoff, Langgartner, et al. (2010) state that only 17% of the alarms are clinically significant. Imhoff, Kuhls, and Gather (2009) reported that about 359 alarms occur per cardiac surgery procedure at 1.2 per minute and about 80% of the alarms have no beneficial effect. American College of Clinical Engineering survey more than 1300 healthcare professionals and reports that 81% of clinicians agreed that nuisance alarms occur frequently, and 78% agreed on disabling them due to reduced trust in alarms (Drew, Califf, & Funk, 2004). Chambrin et al. (1999) assess the significance of current monitoring alarms in the ICU among 131 adult patients and reported 3188 alarms with an average of one alarm every 37 min: 23.7% were due to staff manipulation, 17.5% to technical problems and 58.8% to the patients and identified a false positive rate of alarm was 74.2%. Siebig, Kuhls, Imhoff, Gather, et al. (2010) identified 70% of alarms generated were due to threshold alarms and 45% were related to arterial blood pressure among 5934 alarms during 982 hours of observation. Emergency Care Research Institute (ECRI) state alarm hazard as the number one medical device technology hazard for the year 2012, 2013, 2014 and 2015 (ECRI, 2011, 2012, 2013, 2014) and Alarm fatigue has been identified as one of the top 10 medical hazards (Keller, 2012). A patient in Massachusetts General Hospital died after the alarm on a heart monitor was accidentally left off (Wallis, 2010). Federal investigators concluded the incident as alarm fatigue experienced by clinicians functioning among frequently beeping monitors. U.S. Food and Drug Administration (FDA) Manufacturer and User Facility Device Experience (MAUDE) database reported 566 patient deaths related to the monitoring device alarms between January 2005 and June 2010 and the Joint Commission’s Sentinel Event database reported that 98 alarms related events were recorded between January 2009 and June 2012. Among 98 reported events, 80 resulted in death, 13 in permanent loss of function, and five in unexpected additional care (TJC, 2013).

Many times, alarms are triggered as they are out of range or beyond the threshold values generating false positives without any clinical relevance. The false alarm rate has basically not changed over the decades despite the advancement in technology (Baumgartner, Ro ̈del, Schreiber, & Knoll, 2012). However, research in the field of machine learning, and artificial intelligence has shown promising results. There is a need of monitoring systems with fewer alarms, but with preserved sensitivity for clinically relevant alarms. As a consequence, we propose multi-parameter analysis data fusion-based approach to reduce the rate of false alarms.

In this study, we focus specifically on reducing false arrhythmia alarms in an ICU setting using a data fusion-based approach. Data fusion is a method designed to compute from multiple sensors data, integrate them and generate more meaningful information that can be of greater value than single source data. We plan to minimize the false alarm rates for life threatening arrhythmia alarms; especially bradycardia, tachycardia, and ventricular tachycardia generated in the ICU's patient information systems with the use of multi-parameter analysis in different time domains utilizing various data transformation techniques. The data were obtained from Physionet's MIMIC II (Multi-parameter Intelligent Monitoring in Intensive Care) database. The major objective of this research is to achieve high false alarms suppression rates and low true alarm suppression rates, and investigate the effect of false alarms on nursing staffs by a simulation approach.

1.2 Layout of Dissertation

This study is divided into seven chapters.

- Chapter one provides an introduction of the dissertation, and the alarms, the problem associated with the alarms such as alarm fatigue, false alarms and the motivation of this study.
- The second chapter discusses the extant literature in the areas of reduction of false alarms in non-clinical and clinical domain as well as the data fusion approach in regards to machine learning in general and in clinical areas.
- The third chapter elaborates on the approach of data fusion for decision support developed in this study with the list of objectives and artifacts. This section also describes the data fusion-based decision support architecture.
- The fourth chapter describes the research methods used to accomplish the goals of this study.
- Chapter five focuses on the discussion of results and summarizes the important findings of the study. Furthermore, the chapter also illustrates the ensemble learning approach and use of the approach to reduce the rate of false alarms and evaluates our approach with other approach to validate the results.
- Chapter six investigates the effect of false alarms on ICU and develops a discrete event simulation model to study the effect of false alarms and different alarm

policies.

- Finally, the seven chapters conclude the thesis by providing contributions, limitations and implications for future research.

Chapter 2

Review of Literature

Research on reducing false alarms can be found in clinical as well as non-clinical areas where false positive alerts are needed to be reduced significantly and improve true positive alerts.

2.1 Reduction of False Alarms

Significant research on false alarms reduction is found in non-clinical areas such as intrusion detection systems, smoke detection systems, explosive detection systems, and other areas. Arrue, Ollero, and Martinez (2000) developed a False Alarm Reduction (FAR) System for forest fire detection using an infrared image processing techniques and Artificial Neural Networks, and a decision function designed using a fuzzy expert rule. Pietraszek and Tanner (2005) grouped the alert management in intrusion detection system mainly into two categories: (i) improving the quality of alerts and (ii) alert correlation. Utilizing supplementary information such as alert context can enhance the quality of alerts. Sommer and Paxson (2003) used alert context approach to develop Bro's byte-level alert signatures (Paxson, 1999). Valdes and Skinner (2001) illustrated a heuristic approach to alert correlation using a weighted sum of attribute similarities that allow to group alerts into scenarios. The other approach to address the problem for false positive alert is by building an alert classifier that notifies true from false positives alerts (Pietraszek & Tanner, 2005). Jazzar and Jantan (2008) proposed a solution to reduce the false alert rate in intrusion detection system by using fuzzy cognitive map which is a soft computing modeling techniques generated from the compensation of fuzzy logic and neural network.

Furthermore, Merzbacher and Gable (2010) applied data mining techniques for the reduction of false positives in aviation explosives detection computed tomography imaging systems. Choi, Akin, Kwak, and Toliyat (2014) proposed an error management algorithm to minimize the rate of false alarms of motor faults in hybrid electric vehicles. Xu et al. (2015) used process context to reduce the false positive alarms during sporadic operations on cloud applications called Process-Oriented Dependability (POD) Monitor and improves the precision up to 0.226 resulting in 36.1% improvement.

2.2 Reduction of False Alarms in Clinical Domain

Several studies have been conducted to analyze the issue of clinical alarms and reducing false alarms. Zong, Moody, and Mark (2004) developed an algorithm that reduces false alarms related to changes in arterial blood pressure (ABP) in ICU monitoring by evaluating the ABP signal quality and examining the ECG-ABP relationships using a fuzzy logic approach. Aboukhalil, Nielsen, Saeed, Mark, and Clifford (2008) reduced the rate of false critical ECG arrhythmia alarms from 42.7% to 17.2% by relating the ECG data with the arterial blood pressure curve. Blum, Kruger, Sanders, Gutierrez, and Rosenberg (2009) recommended computer architecture based on reactive intelligent agent technology to improve the physiologic alarms sensitivity in a critical care unit. Borowski, Siebig, Wrede, and Imhoff (2011) suggested higher rate of false alarm can be minimized using statistical signal extraction algorithm like adaptive online Repeated Mediation (Schettlinger, Fried, & Gather, 2010), adaptive online Trimmed Repeated Median-Least Squares (Borowski, Schettlinger, & Gather, 2009) that separates significant signals from noise. Sayadi and Shamsollahi (2011) develop a novel nonlinear joint dynamical model that is designed for being used in Bayesian estimation procedures such as the Kalman filter to provide synchronized estimations of pulsatile cardiovascular signals including the ECG, ABP, PPG, CVP, and PAP and used for false arrhythmia suppression with an overall false suppression rate reduced from 42.3% to 9.9%. Scalzo, Liebeskind, and Hu (2013) introduce a smart alarm detection system for intracranial pressure signal (ICP) based on advanced pattern recognition methods and use an adaptive discretization to reduce the dimensionality of the input features that led to decrease of 30% of false ICP alarms without compromising sensitivity. Behar, Oster, Li, and Clifford (2013) used an automated algorithm to assess electrocardiogram quality for both normal and abnormal rhythms for false arrhythmia alarm suppression in ICU where the signal quality indices were derived from the ECGs segments and used as the inputs to a support vector machine classifier with a Gaussian kernel. Salas-Boni, Bai, Harris, Drew, and Hu (2014) developed a robust methodology that suppresses false positive ventricular tachycardia alarms by applying a multi resolution wavelet transform to the ECG data using L1-regularized logistic regression classifier where 21% of false alarm suppression with zero true alarm suppression was achieved. Roychoudhury, Ghalwash, and Obradovic (2015) investigate a cost-sensitive approach for false alarm suppression on two life-threatening cardiac arrhythmia alarm Asystole and

Ventricular Tachycardia from MIMIC II database and achieved moderate false alarm suppression rate of 34.29% for Asystole and 20.32% for Ventricular Tachycardia while keeping near 100% true alarm detection.

Other approaches are also proposed to improve alarm system such as use of the median filter (P. L. Davies, Fried, & Gather, 2003; Mäkivirta, Koski, Kari, & Sukuvaara, 1991) to eliminate noise, development of control chart method to detect the onset of changes in systolic blood pressure during the use of anesthesia (Kennedy, 1995), a trend based alarm system (Charbonnier & Gentil, 2007; Jakob et al., 2000; Schoenberg, Sands, & Safran, 1999) to improve patient monitoring, multivariable fuzzy temporal profile modeling for designing intelligent alarms capable of addressing the flaws and limitations of threshold alarms (Otero, Félix, Barro, & Palacios, 2009).

In this regard, several strategies for alarm management have been suggested to reduce false alarms and improve patient safety. Numerous studies have shown the potential to reduce the alarm rate by the adjustment of alarm default settings. Customizing the alarm parameters according to the patient have resulted in decrease of false alarms rate (Graham & Cvach, 2010; Phillips, 2006). Research has shown that changing the heart rate alarm from 120 bpm to 130 bpm has resulted in a 50% decrease in the number of alarms (Gross, Dahl, & Nielsen, 2011). Similarly, when default alarm parameters were changed including customization of the alarms, 43% reduction in critical monitor alarms was observed (Graham & Cvach, 2010). Delaying the setting on the SpO2 alarm to 15 seconds (Welch, 2011) or 19 seconds (Go¨rges, Markewitz, & Westenskow, 2009) can reduce the frequency of alarms by 50% and 70%, respectively. Setting the alarm threshold based on each patient's condition can also reduce the frequency of alarms resulting in decrease of alarm fatigue. Welch (2011) reduced the SpO2 alarm threshold from 90 % to 88%, and the alarm rate was decreased by 45%. Whalen et al. (2014) changed the limits for heart rate low to 45 bpm and high to 130 bpm, and the alarms level for bradycardia, tachycardia to "crisis," requiring the staff to act on the alarm each time it sounded and overall 89% reduction in total mean weekly audible alarms was achieved with the improvement in staff and patient satisfaction.

Various research efforts have been made in the areas to reduce false alarm, however it seems inadequate as these sensors may be of different types with different requirements. The examples of sensor types include radar, thermal, acoustic, laser, optical, and clinical. These different types of sensors have different strengths and weaknesses. Therefore, integrating data from multiple sensors of different types provides a better result because the strengths of one type can compensate for the weaknesses of another type. This is where the data fusion comes in a picture. D. L. Hall and Llinas (1997) defined data fusion as a technique that combines data from multiple sensors and relates information from associated databases to achieve improved accuracy and more specific inferences than could be achieved by the use of a single sensor alone.

2.3 Data Fusion with Machine Learning

Different approaches have been proposed for data fusion algorithms. Chan, Fan, Prodromidis, and Stolfo (1999) used naive Bayes, decision tree, CART, and RIPPER as base classifiers and combine them to propagate fusion algorithm. Maes, Tuyls, Vanschoenwinkel, and Manderick (2002) used Bayesian networks and back propagation algorithm for neural networks called STAGE algorithm to identify alert in credit transactional fraud detection. M. Kim and Kim (2002) proposed a decision tree and back propagation neural network to generate an integrated algorithm for weighted suspicion score on credit card transactions. Phua, Alahakoon, and Lee (2004) recommended back propagation neural networks, naive Bayes, and decision tree as base classifiers on data partitions to develop fusion algorithm to produce the best cost savings on insurance claims. Algorithms such as neural networks, Bayesian networks, and decision trees have been applied in a sequential fashion to improve results (Phua, Lee, Smith, & Gayler, 2010).

Other developments in data fusion include hybrid models such as those suggested by Phua et al. (2010) where optimal results can be derived from a hybrid model which combines multiple algorithms. Using true positive rate with no false positives as the performance measure, Taniguchi, Haft, Hollmen, and Tresp (1998) state that supervised neural networks and Bayesian networks on labeled data achieve significantly better outcomes on non-fraud user to detect anomalous phone calls. Kumar and Rathee (2011) compared the results from classification method using J48 classifier with the outcomes from fusion of clustering and classification method using WEKA and the results illustrate that the fusion algorithm gives promising results with utmost accuracy rate even when the data set contain missing values. Moreover, Zheng (2015) explores the relationship and difference between different data fusion methods states that the proliferation of data in today’s world calls for techniques that can fuse knowledge from multiple disparate datasets by the process of advanced data fusion methods.

2.4 Data Fusion with Machine Learning in Clinical Domain

The concept of data fusion has been implemented in clinical settings from several years. Factor, Gelernter, Kolb, Miller, and Sittig (1991) apply the process trellis, a domain and hardware independent software architecture in building the Intelligent Cardiovascular Monitor (ICM) prototype, a real-time clinical decision-support system by the process of data fusion. Feldman, Ebrahim, and Bar-Kana (1997) used robust sensor fusion method that is designed to fuse data from multiple sensors with redundant data to improve the quality of alarm detection and the outcome was a fused estimate of heart rate which was better than the estimates available from any individual sensor and that also minimized the occurrence of false positive alarms. Chen, Huang, Chen, Chen, and Luh (2006) propose a multi-level sensor data fusion approach that infers inactivity of an older people based on accelerometer and implies that with data fusion, the existing Personal Emergency Response Systems (PERS) can be improved to provide timely emergency alarms that will potentially save lives. An automated system called BioSign was developed to generate early warning of patient deterioration through data fusion of heart rate, breathing rate, oxygen saturation, temperature, and blood pressure (Hann, Tarassenko, Patterson, Barber, & Young, 2006). Zhang (2007) studied the feasibility of developing patient-specific alarm algorithms in real time in pediatric intensive care unit using classification trees and neural networks to bring adaptive capabilities to the patient monitoring and achieved a sensitivity of 96%, a specificity of 99%, a positive predictive value of 79&, and an accuracy of 99% with neural network.

Numerous methods have been proposed for data fusion in clinical domain with machine learning approach. Tan and Gilbert (2003) used three different supervised machine learning techniques C4.5 decision tree, bagged and boosted decision tree in cancer classification and observed that ensemble learning- bagged and boosted decision tree often performs better than single decision tree in classification task. Polikar et al. (2008) used an ensemble of classifiers based data fusion approach to combine information from two or more sources for early diagnosis of Alzheimer's disease using learn^{++} algorithm. Vyas, Farringdon, Andre, and Stivoric (2011) provide insight into the BodyMedia FIT armband system—a wearable multi-sensor technology that continuously monitors physiological events related to energy expenditure for weight management and demonstrates the use of machine learning and multi-sensor data fusion techniques provide accurate results for various activities for a large range of users in both lab and free-living settings. D. Kim, Shin, Song, and Kim (2012) used a graph-based semi-supervised learning as a classification algorithm for prediction of clinical outcomes in cancer and accuracy of prediction increases because of incorporation of information fused over heterogeneous biological data sources. Salimi-Khorshidi et al. (2014) introduce FMRIB's ICA-based X-noiseifier (FIX) that provides an automatic solution for cleaning functional MRI data of various types of structured noise via accurate classification of independent component analysis (ICA) and fusion of multiple classifiers such as linear SVM, SVM with RBF kernel, random forest, and conditional-inference tree, a stacking ensemble technique and achieve 95% of over all accuracy.

While several of the statistical and algorithmic solutions have been explored for reducing false alarm rate and alarm sensitivity in healthcare, there is limited research in data fusion for decision support in healthcare context where there are a wide variety of data types (for e.g. waveform, numerical and text), with different periodicities, and very high accuracy and performance requirements. Furthermore, research in multi-parameter data in ICU context is inadequate where comparative analysis with various feature sets is studied in time domain with different data transformation techniques. Moreover, to our knowledge, no study has been conducted to simulate the effect of false alarms in clinicians.

Chapter 3

Data Fusion

3.1 Problem Description - Clinical Decision Complexity in ICU

The clinical decision processes in an Intensive Care Unit is particularly complex due to multiple factors. Patients are admitted to ICU when they are in a critical condition. Upon admission to the ICU, clinicians perform a general assessment and then use various devices to measure the patient's vital signs such as blood pressure, heart rate, respiration rate, oxygen saturation (SpO_2), and other significant physiological parameters to understand the patient health status. The physiological parameters are measured at different time intervals. Some of these parameters are measured continuously, and some are measured in every 12-16 hours, and some are measured in an hourly basis depending on the physiological signal. The measured physiological parameters have different data types and units as well; some parameters are recorded directly in numeric measurements such as SpO_2, and some parameters are recorded in waveforms such as Electrocardiogram (ECG) that needs to be interpreted. Clinicians need to constantly monitor and process several of these parameters that are measured at different time intervals, with varying data types, different thresholds and ranges, and convey information about different underlying conditions so that they can take preventive measure. In addition, clinician also needs to identify additional tests to perform to generate a better picture of patient's health and determine their health condition.

In this setting, the clinician is burdened with numerous data that may lead to information overload; it is probable that the clinician may miss something significant. In addition to the above complexity, there are typically many false alarms due to device errors, sensor misplacement, patient movements, and other operational and non-clinical triggers leading to alarm fatigue. Moreover, the alarms do not match the critical condition of the patient and impede the clinician's ability to respond.

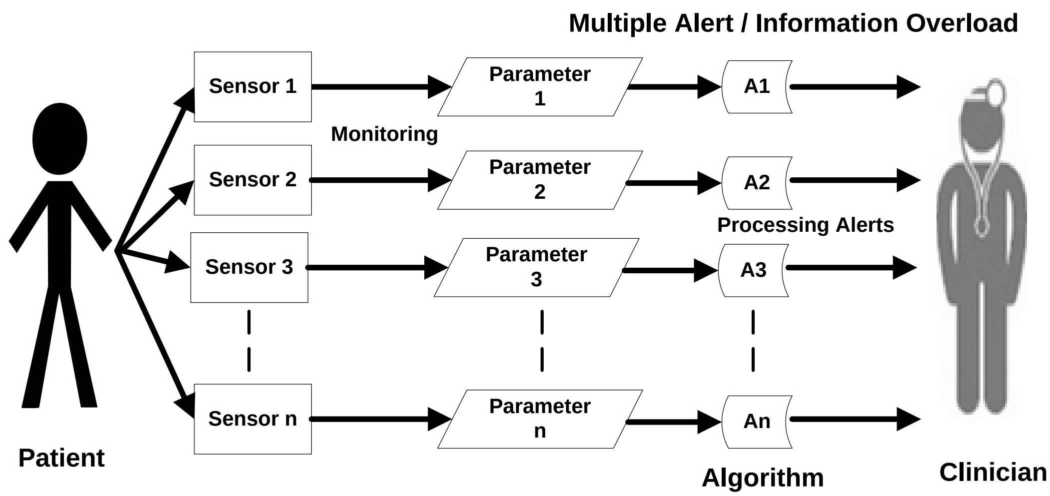

Figure 1: Multiple Sensor Single Parameter Approach

Figure 1 illustrates the current approach where the different sensors measure various physiological parameters such as blood pressure, temperature, heart rate, oxygen saturation etc. and each is individually conveyed to the clinician through multiple display monitors and alarms. Singe-parameter algorithms use individual device data to generate range-based alerts and make clinician aware of the condition of the patient. In this case, clinician receives multiple alerts. Some of these alerts may be vital, and some may be unwarranted and duplicate alerts leading to severe information overload and alarm fatigue in clinicians.

3.2 Data Fusion Approach for Decision Support in ICU

In order to reduce the complexity of the clinical decision making process in intensive care units and to reduce information overload, and false alarms, we propose to use the data fusion approach that encompasses multi-parameter data and provides a data fusion-based system for decision support. Figure 2 illustrates a data fusion approach where the different sensor measures various physiological parameters that are process through data fusion algorithms to generate alert of higher precision and higher recall that will help to reduce the false alarm as well as information overload in clinicians.

The dissertation goal is to minimize the false alarms by the process of data fusion-based decision support. There are three major objectives: 1) Comparative analysis of feature sets & algorithms in time domain with data transformation, and 2) Data fusion-based analysis to maximize false alarm suppression rates and minimize true alarm suppression rates 3) Develop simulation model to study the effect of false alarms on nursing staffs.

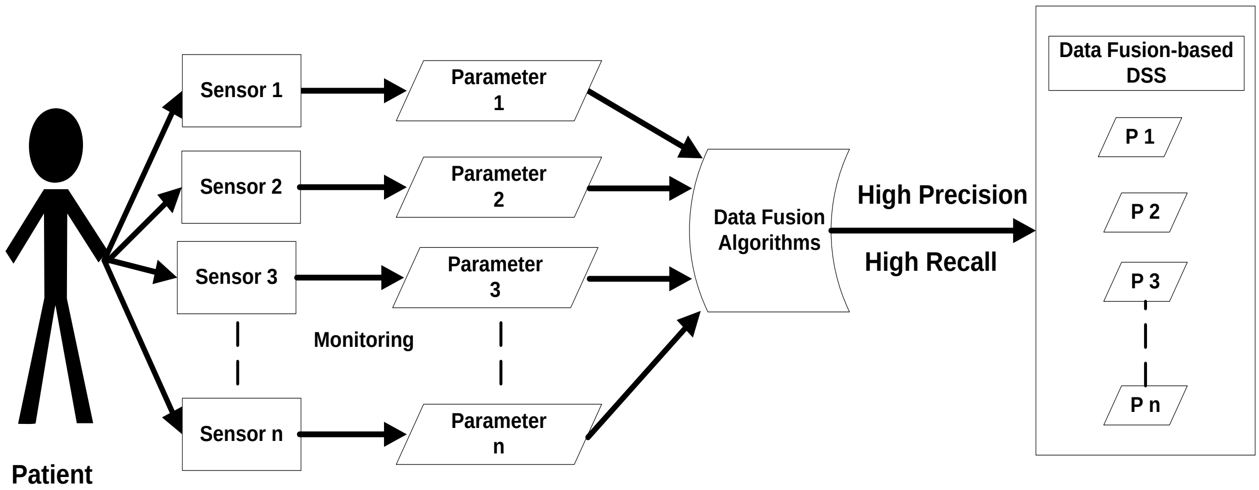

Figure 2: Data Fusion Approach

3.3. Objectives and Artifacts

We address three major objectives in this study, which are listed below.

Table 1: Objectives & Artifacts

Objective	Theory	Artifact
1. Comparative analysis with features sets and algorithms in time domain with data transformation	• Trend extraction methodology (Charbonnier & Gentil, 2007) • Physiologic parameter relationship (Zong et al., 2004) • Statistical signal processing and filtering methods (Borowski et al., 2011)	Optimal model & Parameter optimization
2. Data fusion-based analysis	• Data fusion approach (Borges & Brusamarello, 2014; Zong et al., 2004) • Multi-parameter data mining approach (Baumgartner, Rödel, & Knoll, 2012)	Data-fusion method
3. Develop simulation model to study the effect of false alarm in clinician	• Discrete event model for exploring interruptions on performance of knowledge worker (Gupta, 2007)	Simulation model

In the context of achieving our objectives, we intend to study various dimensions, which is described next.

Time Dimension

The general concept of using various time dimensions in research is to study the impact of time in the research output, and identifying the adequacy for prediction. Our major objective is to minimize false alarm rates, so we intend to study various time domains to investigate at which time the high false alarm suppression and low true alarm suppression rates can be achieved. Therefore, the research objective is to:

Explore time ranges to predict better outcome for alarm suppression rates.

Data Transformation

Artifacts, noise, and missing values often corrupt the physiological signals that lead to errors. The data transform methods such as taking average, standard deviation, Fourier transforms, where the original data samples are transformed in the hope of achieving better performance. Therefore, the research objective is to:

Explore transformation methods to predict better outcome for alarm suppression rates.

3.4 System Architecture

We propose the data fusion-based decision support architecture to gain high precision, high recall alert that consists of data collection module, data pre-processing module, data fusion module, and decision support module.

Data Collection Module

The data collection module is the process of gathering useful information about a phenomenon. Once the data are collected from the individual sensors they are analyzed and used for monitoring purposes. Data are collected to establish factual basis for decision-making processes. It is an important aspect in the research study, as accurate data is essential to maintain the research integrity. The module is designed to handle data with different data types, varying units with different time intervals. While collecting the data, data standard is maintained so that the issues such as interoperability do not arise later. The collected data is organized, normalized and accurate as the quality of data collected impacts directly on the quality of analysis that eventually will impact the quality of decision that can be made.

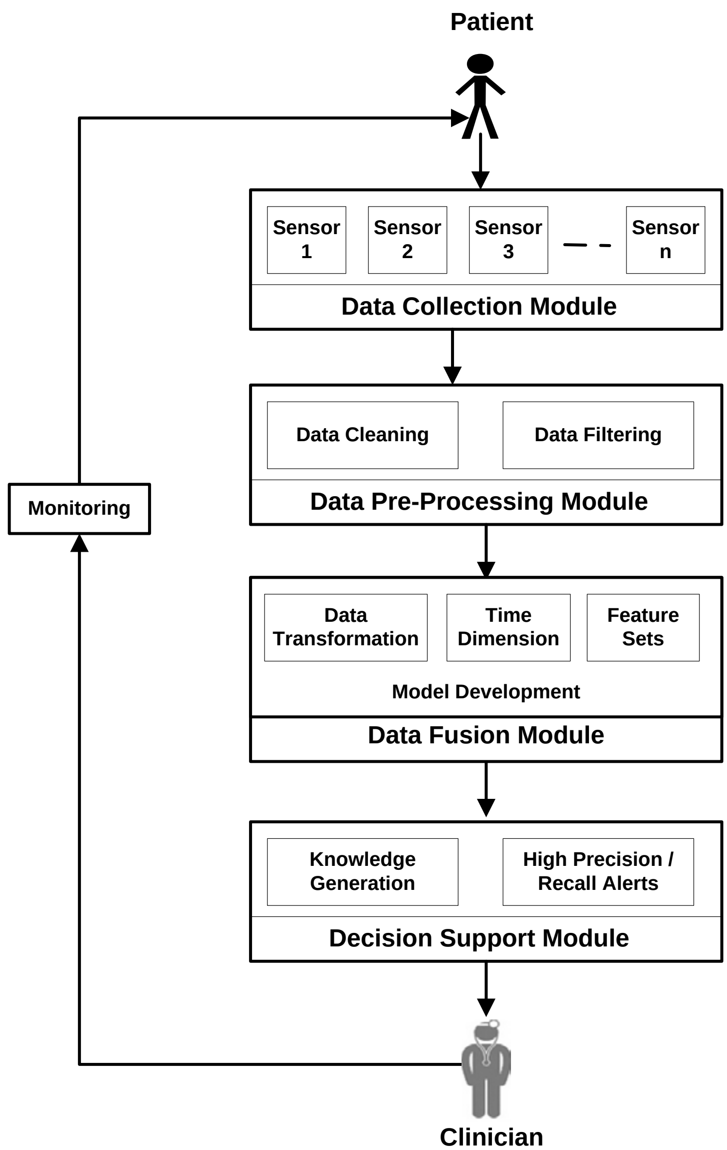

Figure 3: Data Fusion-based Decision Support Architecture

Data Pre-Processing Module

The data pre-processing module comprises of a several individual steps such as understanding the available data, extracting the data and cleaning, merging relevant data files then creating and coding different variables. Data quality check and cleaning are essential steps to consider while building an accurate model as data anomalies and impurities may result in inefficient data analyses and erroneous decisions. The following steps are followed while processing the gathered data. The data sets are thoroughly checked to see if all the variables considered are fully populated with correct formats and illogical values are adjusted. Remove variables with high missing rate. Different kinds of variables can be observed within the datasets. Identifying different variable types within the datasets and recoding them with appropriate missing treatments are also a part of preprocessing module. After applying the missing treatments datasets are merged together forming a training data set. The training dataset contains all the independent and dependent variables extracted through the process of variable selection. The module is accountable for standardizing and normalizing the data into a structural format in order to enhance the performance of the subsequent modules.

Data Fusion Module

The data fusion module is the process of convergence of data from different sources to generate more meaningful information that can be of better value than single source data. The data fusion approach encompasses data streams from multiple devices and develops a system for decision support. It combines data in order to remove the influence of irrelevant data, so that the optimal analysis of information is obtained. Information may be of different data types such as numeric, text and waveform. The fusion module is designed to handle various data types, so the fusion process can be performed precisely.

Decision Support Module

Decision Support System (DSS) in a clinical context is a system that takes input as clinical information and produces as output inferences that can assist clinician in their decision making (Musen, 1997). It is a potential way for delivering the precise information, but requires careful design of the interactions required for selecting and displaying information. The data fusion module and decision support module will be the main contribution.

Chapter 4

Research Methods

4.1 Data Sources

For the study, a subset from PhysioNet's Multi-parameter Intelligent Monitoring in Intensive Care (MIMIC II) database was used (Saeed et al., 2011). The database includes 4458 measurement records. The records have a ”waveform” as well as a ”numeric” part that is sampled per minute. Additionally, metadata is also available with age and gender information of the patients. The alarm notifications of the monitors are also included in separated alarm annotation file, which consist of a timestamp, the “aux” field of each annotation is the text of the alert, and the “subtype” field indicates the severity of the detected event, 3 as "red" or most critical and 2 as "yellow" or less critical. The “chan” and “num” fields of these annotations were unused.

Aboukhalil et al. (2008) chose a subset of the MIMIC II database for alarm labeling and classification that fulfilled two criteria: a critical arrhythmia alarm was issued and one channel of ECG and an ABP waveform were present at the time of the alarm. They labeled five types of arrhythmia alarms: 1) Asystole - alarms were triggered by a default asystolic pause of 4 s, 2) Bradycardia - heart rate (HR) less than 40 bpm, 3) tachycardia -HR greater than 140 bpm, 4) ventricular tachycardia - a run of ventricular beats at a rate of at least 100 bpm, lasting 5 or more beats, and 5) ventricular fibrillation -a fibrillatory waveform lasting for at least 4 s. Furthermore, the alarm was judged to be true or false with expert human review creating the gold standard alarm database.

As of time limit, we only used patient records with alarm annotation for bradycardia (BRADY), tachycardia (TACHY), and ventricular tachycardia (VTACH) for the study. Numerous studies used the “waveform part” to study false alarms and proposed techniques for false alarm reduction (Aboukhalil et al., 2008; Baumgartner, Ro¨del, & Knoll, 2012; Behar et al., 2013; Eerika¨inen, Vanschoren, Rooijakkers, Vullings, & Aarts, 2015; Li & Clifford, 2012). To our knowledge, no research has been done using “numeric part”. The waveform records may contain up to four signals such as ABP, PAP, ECG digitized at 125 Hz with 8-bit resolution, and the "numeric" record may contain 10 or more time series of vital signs sampled once per minute. In our work, all classifiers were trained using physiological parameters obtained from numeric part with 10-fold cross validation. In 10-fold cross-validation, the original dataset is randomly partitioned into 10 equal size subsamples. Of the 10 subsamples, a single subsample is retained as the validation data for testing the model, and the remaining 9 subsamples are used as training data. The cross-validation process is then repeated 10 times with each of 10 subsamples used exactly once as the validation data. The 10 results from the folds can then be averaged to produce a single estimation. The advantage of this method is that all observations are used for both training and validation, and each observation is used for validation exactly once.

Table 2: Alarm Distribution

Alarm	False Alarms	True Alarms	Total Alarms
Bradycadia (BRADY)	218	490	708
Tachycardia (TACHY)	415	1551	1966
Ventricular Tachycardia (VTACH)	956	1104	2060
Total	1589	3145	4734

4.2 Alarm Definitions

In an ICU, patient monitoring system normally generate two types of alarms: 1) ‘‘yellow" alarm to notify something abnormal, and 2) ‘‘red" alarm to notify a critical event. The ‘‘yellow" alarms are not very loud and usually last only for few seconds. However, ‘‘red" alarms are much louder and have a unique tone that remains on until they are acknowledged by on duty nursing staffs. In this study, we considered only critical ‘‘red" arrhythmia alarms.

4.3 Physiological Parameters

The predictor variables are Respiration rate (RR), Arterial Blood Pressure (ABP), Pulmonary Artery Pressure (PAP), Mean Arterial Pressure (MAP), Heart rate (HR), Central Venous Pressure (CVP), Non-Invasive Blood Pressure (NBP) Oxygen Saturation (SpO_2), Pulmonary Artery Wedge Pressure (PAWP), and Cardiac Output (CO) are defined below.

Table 3: Physiological Parameter Definitions

Parameter	Definition
RR	Breathing frequency
HR	Speed of heart beat
ABP	Pressure exerted upon arteries during heart contractions
PAP	Measure of the blood pressure found in the pulmonary artery
CVP	Pressure of blood in thoracic vena cava
NBP	Pressure exerted by circulating blood on the walls of blood vessels
SpO_2	Concentration of oxygen in blood
PAWP	Pressure generated by left ventricle
CO	Volume of blood pumped by the heart in time interval of 1 min

The parameters normal and abnormal ranges are listed below.

Table 4: List of Parameter with Normal and Abnormal Range

Parameter	Unit	Normal Range	Abnormal Range	Device / Sensor / Method
RR	bpm	12-18 (Sherwood, 2005)	>20 = Unwell >24 = Critically ill (Cretikos et al., 2008)	Piezoelectric Sensor
HR	bpm	60-100 (Laskowski, 2012)	>100 = Tachycardia <60 = Bradycardia (MedlinePlus, 2012)	Pulse Oximeter
ABP–Systolic[1]	mmHg	90-140 (Lidco, 2014)	>140= Hypertension <90 = Hypotension (AHA, 2012)	Sphygmomanometer
ABP–Diastolic[2]	mmHg	60-90 (Lidco, 2014)	>90 = Hypertension <60 = Hypotension (AHA, 2012)	Sphygmomanometer

PAP-Systolic[1]	mmHg	15-25 (Lidco, 2014)	>25 = Hypertension (Grünig et al., 2000)	PA Catheter
PAP-Diastolic[2]	mmHg	8-15 (Edwards, 2014; Lidco, 2014)	--------	PA Catheter
CVP	mmHg	2-6 (Edwards, 2014)	--------	Transducer / Manometer
NBP-Systolic[1]	mmHg	<120 (AHA, 2012)	>140= Hypertension <90 = Hypotension (AHA, 2012)	Sphygmomano meter
NBP-Diastolic[2]	mmHg	<80 (AHA, 2012)	>90 = Hypertension <60 = Hypotension (AHA, 2012)	Sphygmomano meter
SpO_2	%	95-100 (Edwards, 2014)	<90 = Hypoxemia (MayoClinic, 2013)	Pulse Oximeter
PAWP	mmHg	6-12 (Lidco, 2014)	--------	Swan-Ganz Catheter
CO	L/min	4-8 (Lidco, 2014)	--------	Doppler Ultrasound

* [1]Systolic refers to BP when the heart beats while pumping blood. * [2]Diastolic refers to BP when the heart is at rest between beats.

4.4 Design Science Research Approach

The study embraces the design science research approach as research methodology. The most popular guidelines for design science research have been proposed by the Hevner, March, Park, and Ram (2004) and Peffers, Tuunanen, Rothenberger, and Chatterjee (2007). We follow Peffers et al. (2007) guidelines which is illustrated below:

- **Problem-Centered Approach:**

False alarms has been one of the major issues in clinical domain, particularly in

ICU that causes alarm fatigue, waste of human resources, and increased workload for care providers as well as risk to patient's health. Numerous studies document the adverse effect of false alarms on both patients and staff that affects quality of care and patient safety. Minimizing the false alarm has become an utmost importance.

- **Problem Identification and Motivation:**

Various studies acknowledge high rate of false alarms in ICU (Chambrin et al., 1999; Lawless, 1994; Siebig, Kuhls, Imhoff, Langgartner, et al., 2010; Tsien & Fackler, 1997) that led to care disruption and an increase in the workload of ICU staff (Allen & Murray, 1996) and eventually to alarm desensitization (Chambrin, 2001; Drew et al., 2014) including patient's death (TJC, 2013; Wallis, 2010).

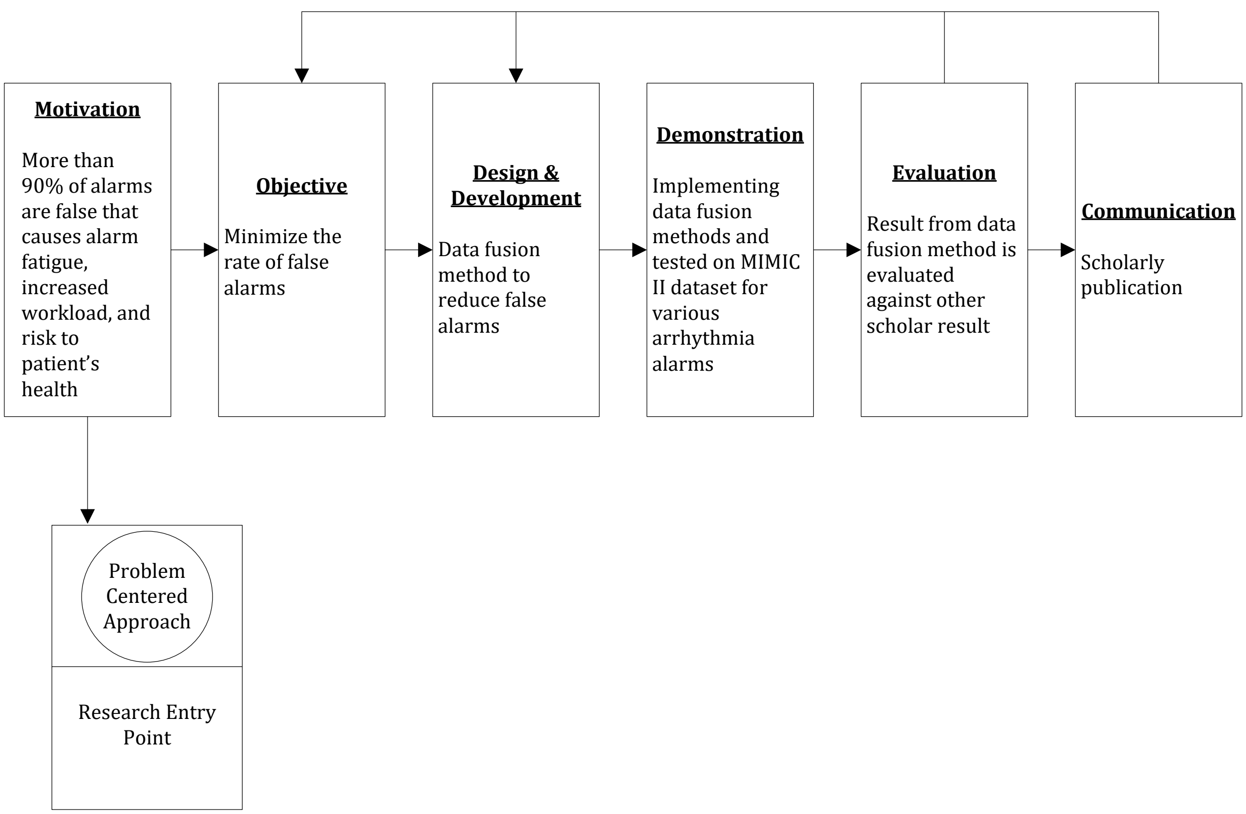

Figure 4: Design Science Research based on Peffers et al. (2007) Guidelines

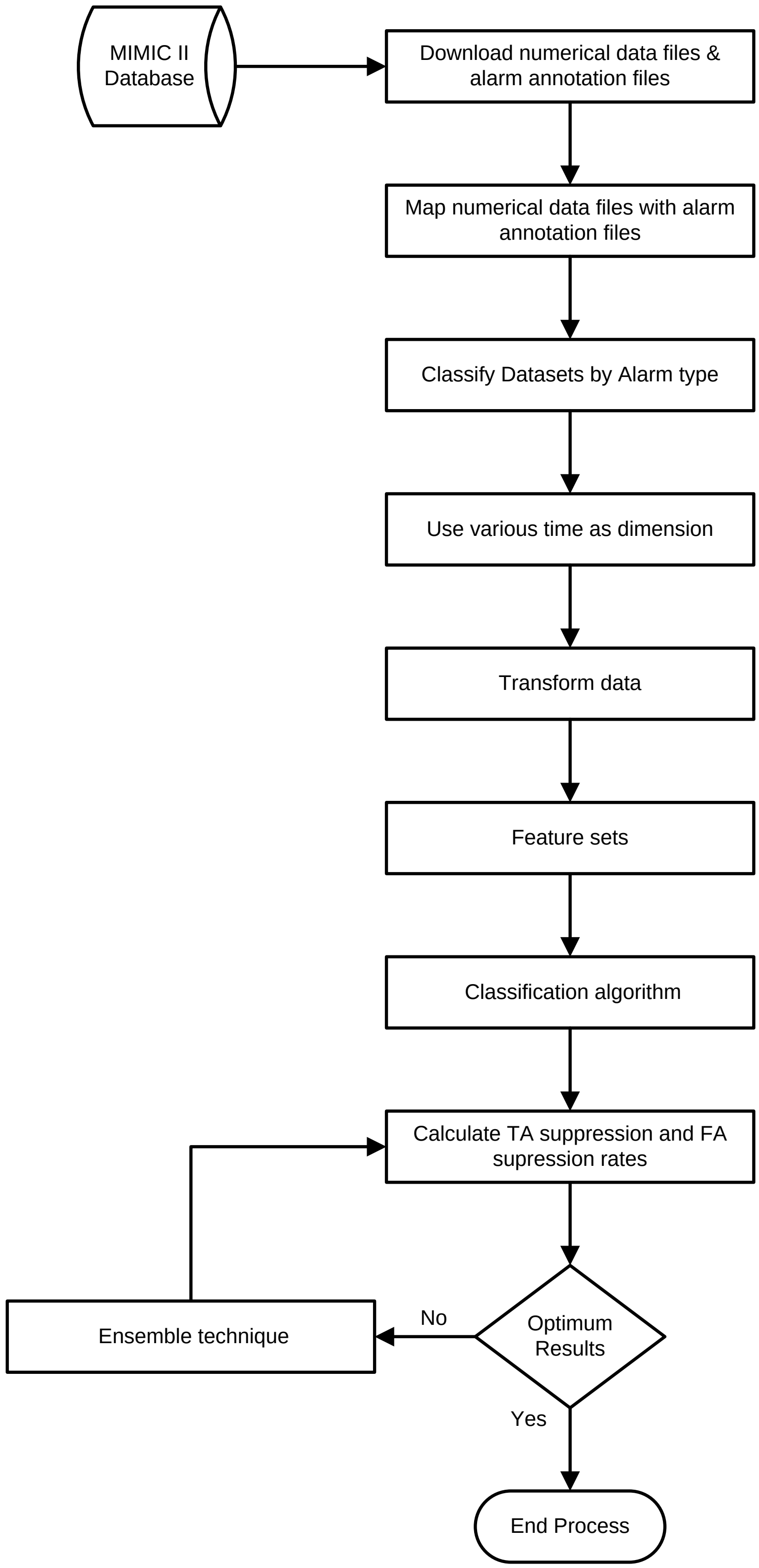

Figure 5: Process Flowchart

- **Objective of the Solution:**

  The objective is to minimize the rate of false alarm by the process of data fusion-based approach. Moreover, we plan to use multi-parameter analysis in different time domains utilizing various data transformation techniques as well as study the effect of false alarms among nursing staffs by the process of simulation.

- **Design and Development:**

  We developed a data fusion-based method to minimize the rate of false alarms in ICU. We used different transformation technique such as mean, median, standard deviation, and Discrete Fourier transform taking various time ranges such as 30, 60, 90, and 120 minutes in consideration with several feature sets. Furthermore, we also developed a simulation model to study the effect of false alarms in clinicians.

- **Demonstration:**

  We implemented the data fusion-based approach and tested it on MIMIC II dataset for various arrhythmia alarms such as bradycardia, tachycardia, and ventricular tachycardia. Furthermore, the effect of false alarm was demonstrated through a simulation model.

- **Evaluation:**

  Result from data fusion method is evaluated against other scholar's result.

- **Communication:**

  The communication is done through scholarly publication.

Our study seeks to understand the influence of the time dimension, and data transformation to develop a model for decision support that can reduce the rate of false alarms. Figure 5 illustrates the flowchart for the process.

4.5 Data Processing

4.5.1 Data Transformation

We extracted from the signal statistical parameters such as mean, median, standard deviation, and Discrete Fourier Transform (DFT) that were calculated from the sample data and aim at characterizing the physiological parameters available.

- **Mean**

An obvious transformation method of the time-series data is the mean's value. Here, x is the number of unique alarm events

$$f_1(x) = \frac{1}{n}\sum_{i=0}^{n} x_i$$

- **Median**

We also consider taking the median value as other transformation method

$$f_2(x) = \left\{\frac{(n+1)}{2}\right\}$$

where the value of $f_2(x)^{\text{th}}$ is the median value

- **Standard deviation**

The other transformations we consider is the standard deviation and illustrate how much the samples deviate from the average

$$f_3(x) = \sqrt{\frac{1}{n-1}\sum_{i=0}^{n}(x_i - f_1(x))^2}$$

- **Discrete Fourier Transform**

We also consider taking the Discrete Fourier Transform (DFT) as the transformation method where given a sequence of *N* samples *f* (*n*), indexed by *n* = 0…*N*-1, the DFT is defined as $f_4(x)$, where *k* = 0...*N*-1:

$$f_4(x) = \frac{1}{\sqrt{\mathrm{N}}}\sum_{n=0}^{N-1} f(n)\, e^{-j2\boldsymbol{\pi} xn/N}$$

Here, $f_4(x)$ are often called the 'Fourier Coefficients'. In our study, we consider only 4 coefficients.

4.5.2 Feature Selection

Feature selection is the process of selecting a subset of relevant features for use in model construction. Feature section is also known as attribute selection, variable selection. It is of great importance in the field of machine learning and data mining. The methods used for selecting feature sets can be classified into two types: Wrapper and filter method. Wrapper method considers the selection of a set of features based on the learning algorithm used to train the model itself where different combinations are prepared, evaluated and compared to other combinations. These methods generally result in better performance than filter methods because the feature selection process is optimized for the classification algorithm to be used. However, wrapper methods are expensive for large dimensional database in terms of computational complexity and time since each feature set considered must be evaluated with the classifier algorithm used. In filter feature selection method, the selection procedure is independent of learning algorithm. Filter approach apply a statistical measure to assign a scoring to each feature and the features are ranked by the score and either selected to be kept or removed from the dataset. Examples of some filter feature selection method are Correlation-based Feature Selection, Gain Ratio attribute evaluator, Information gain evaluator, Principal Component Analysis, Chi-square Feature Evaluation, Fast Correlation-based Feature selection, Euclidean distance, i-test, Markov blanket filter and so on.

We use different feature sets from Weka Explorer such as CFS Subset Evaluator, Wrapper Subset Evaluator (using Bayes Net, NaiveBayes, J48, and Random Forest), Gain Ratio and Info Gain Evaluator.

- **Correlation based Feature Selection (CFS)**:

  It is a filter algorithm that ranks feature subsets according to a correlation heuristic evaluation function which evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them (M. A. Hall, 1999). Correlation coefficients is used to estimate correlation between subset of attributes and class, as well as inter-correlations between the features. The selection method assumes that useful features subsets contains feature that are highly correlated with the class, but uncorrelated with each other. Moreover, irrelevant features are ignored as they have low correlation with the class. CFS's feature subset evaluation function is illustrated here:

$$M_s = \frac{k\overline{r_{cf}}}{\sqrt{k + k(k-1)\overline{r_{ff}}}}$$

where M_s is the correlation between the summed feature subsets and the class variable, k is the number of subset features, $\overline{r_{cf}}$ is the average of the correlations between the subset features an the class variable, and $\overline{r_{ff}}$ is the average inter-correlation between subset features.

The numerator of the above equation provides an indication of how predictive of the class a set of features are, and the denominator of how much redundancy there is among the feature (M. A. Hall, 1999).

- **Information Gain Evaluator**:

  It evaluates the worth of an attribute by measuring the information gain with respect to the class.

  Let S be set consisting of s data samples with m distinct classes. The expected information needed to classify a given sample is given by (Karegowda, Manjunath, & Jayaram, 2010)

$$I(S) = -\sum_{i=1}^{n} x_i \, log_2(x_i)$$

Where x_i is the probability that an arbitrary sample belongs to class C_i and is estimated by s_i/s.

Let attribute A has v distinct values. Let s_{ij} be number of samples of class C_i in a subset S_j. S_j contains those samples in S that have value $a_j$of A. The entropy, or expected information based on the partitioning into subsets by A, is given by (Karegowda et al., 2010)

$$E\ (A) = -\sum_{i=1}^{n} I(S) \frac{S_{1i} + S_{2i} + S_{3i} + \cdots S_{ni}}{S}$$

The information that would be gained by branching on A is

$$Gain\ (A) = I(S) - E(A)$$

- **Gain Ratio Attribute Evaluator**:

Gain ratio is a ratio of information gain to the intrinsic information that is used to reduce a bias by taking the number and size of branches into account when choosing an attribute. The gain ratio is defined as (Karegowda et al., 2010)

$$Gain\ Ratio\ (A) = Gain(A)/SplitInfo_A\ (S)$$

Where, gain ratio which applies normalization to information gain using a value defined as (Karegowda et al., 2010)

$$SplitInfo_A\ (S) = -\sum_{i=1}^{v}(\frac{|S_i|}{|S|})\ log_2(\frac{|S_i|}{|S|})$$

The above value represents the information generated by splitting the training data set S into v partitions corresponding to v outcomes of a test on the attribute A.

- **Wrapper Subset Evaluator**: It evaluates attribute sets by using a learning scheme that evaluates subset of variables that allow to detect the possible interactions between variables. Cross validation is used to estimate the accuracy of the learning scheme for a set of attributes (Kohavi & John, 1997).

4.6 Algorithms

4.6.1 J48 Decision Tree

A decision tree is a decision support tool that uses a tree-like graph that decides the end value i.e. dependent variable of a new sample based on various attribute values of the available data. A decision tree includes: a root node, leaf nodes, branches and, internal nodes. Each internal node represents a test conditions applied on an attribute (e.g. whether a coin flip comes up heads or tails), each branch represents the outcome of the test and each leaf node represents a class label and the topmost node in the tree represents the root node. The path from root to leaf represents classification rules. A decision tree consists of 3 types of nodes: 1) Decision nodes - commonly represented by squares 2) Chance nodes - represented by circles and 3) End nodes - represented by triangles. Decision trees are the most powerful approaches in knowledge discovery and data mining that includes the technology of research large and complex bulk of data in order to discover useful patterns.

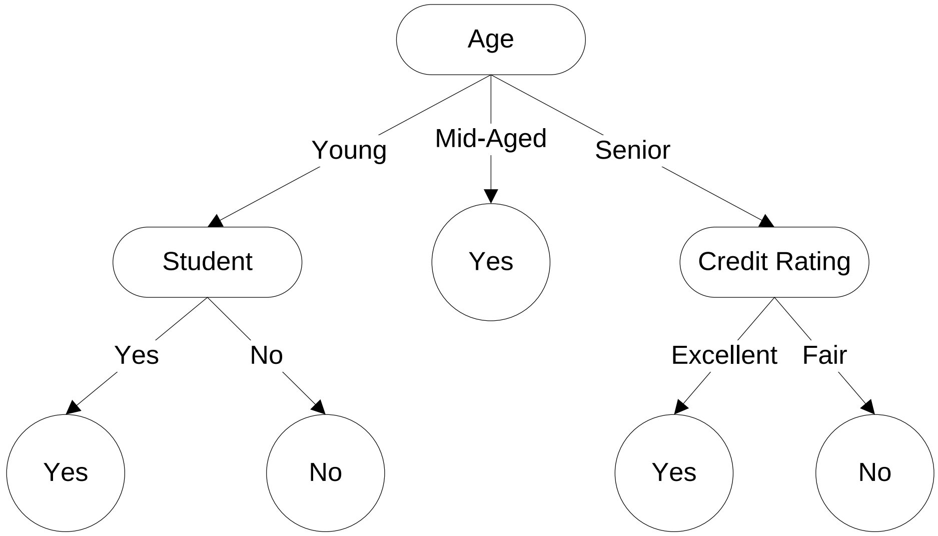

Figure 6: Decision Tree

Figure shown above is the decision tree is for the concept buy computer that indicates whether a customer is likely to buy a computer or not.

J48 is a class for generating a pruned or un-pruned C4.5 decision tree. C4.5 is an algorithm used to generate a decision tree developed by Ross Quinlan, which is an extension of Quinlan's earlier ID3 algorithm (Quinlan, 1993). It is an open source Java implementation of the C4.5 algorithm in the Weka data-mining tool.

4.6.2 Random Forest

A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and use averaging to improve the predictive accuracy and control over-fitting. Ho (1995) created the first algorithm for random decision forests using the random subspace method (Ho, 1998). An extension of the algorithm was developed by Breiman (1996) and Adele Cutler and recognized as their trademark.

Random Forests are a combination of tree predictors where each tree depends on the values of a random vector sampled independently with the same distribution for all trees in the forest. The basic principle is that a group of “weak learners” can come together to form a “strong learner” (El-Atta, Moussa, & Hassanien, 2014). In random forest, many decision trees are representing weak learners and together they are representing a strong learner (random forest). The subsets of the training data are selected randomly and each subset is used to train a decision tree. Each tree is grown as follows (El-Atta et al., 2014):

- Subset (about 66% of the total training data) is sampled at random with replacement to create a subset of the data.
- At each node:
    - Some predictor variables are selected at random from all the predictor variables
    - The predictor variable that provides the best split. According to some objective function, is used to do a binary split on that node
    - At the next node, choose other predictor variables at random from all predictor variables and do the same
- Each tree is grown to the largest extent possible and not pruned.

Random Forests are a wonderful tool for making predictions considering they do not over fit because of the law of large numbers. Introducing the right kind of randomness makes them accurate classifiers and regressors.

4.6.3 Bayes Net

A Bayes network is a probabilistic graphical model that represents a set of random variables and their conditional dependencies via a directed acyclic graph where the nodes represent random variables and the edges between the nodes represent probabilistic dependencies among the corresponding random variables (Ben-Gal, 2007). Let us say the weather can of three types: sunny, cloudy, or rainy, also that the grass can be either wet or dry, and that the sprinkler can be on or off. Now, there are some causal links. If it is rainy, then it will make the grass wet directly, but if it is sunny for a long time, that too can make the grass wet, indirectly, by causing us to turn on the sprinkler. When actual probabilities are entered into this network that reflect the reality of real weather, lawn, and sprinkler-use-behavior, such a network can be made to answer a number of useful questions, like, "if the lawn is wet, what are the chances it was caused by rain or by the sprinkler", and "if the chance of rain increases, how does that affect my having to budget time for watering the lawn".

In a Bayes net, the links may form loops, but they may not form cycles that makes possible very fast update algorithms, since there is no way for probabilistic influence to "cycle around" indefinitely.

4.6.4 NaiveBayes

Naive Bayes methods are a set of supervised learning algorithms based on applying Bayes' theorem with the "naive" assumption of independence between the features. Bayes theorem provides a way of calculating the posterior probability, *P(c|x)*, from *P(c)*, *P(x)*, and *P(x|c)*. Naive Bayes classifier assumes that the effect of the value of a predictor *(x)* on a given class *(c)* is independent of the values of other predictors. This assumption is called class conditional independence.

$$P\ (C|\mathcal{X}) = \frac{P\ (\mathcal{X}|C)\ P(c)}{P\ (x)}$$

Where,

P(c|x) is the posterior probability of class (target) given predictor (attribute)

P(c) is the prior probability of class

P(x|c) is the likelihood which is the probability of predictor given class

P(x) is the prior probability of predictor

4.6.5 Multilayer Perceptron

A multilayer perceptron is a feedforward neural network model that maps sets of input data onto a set of appropriate outputs. Feedforward means that data flows in one direction from input to output layer. It consists of an interconnected group of artificial neurons working in unison to solve specific problems. In most cases, a neural network is an adaptive system that changes its structure during a learning phase. This type of network is trained with the back-propagation learning algorithm and are used to model complex relationships between inputs and outputs or to find patterns in data. An example, system has three layers. The first layer has input neurons, which send data via synapses to the middle layer of neurons, and then via more synapses to the third layer of output neurons. More complex systems will have more layers of neurons with some having increased layers of input neurons and output neurons. The synapses store parameters called "weights" that manipulate the data in the calculations.

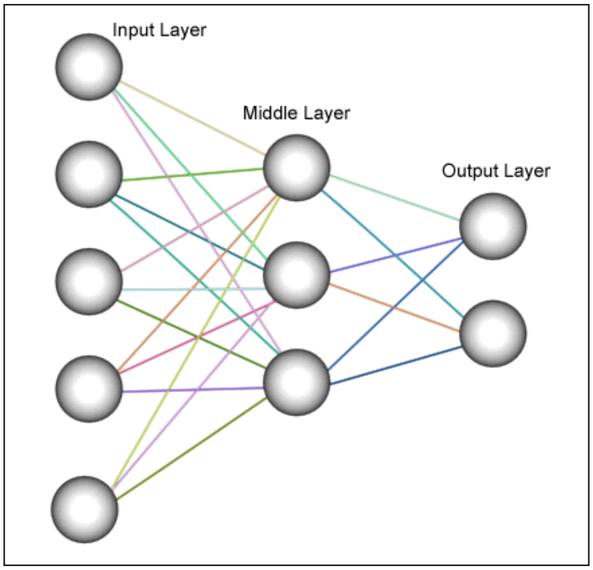

Figure 7: Multilayer Perceptron

It is typically defined by three types of parameters: 1) the interconnection pattern between different layers of neurons, 2) the learning process for updating the weights of the interconnections and 3) the activation function that converts a neuron's weighted input to its output activation.

Mathematically, a neuron's network function $f(x)$ is defined as a composition of other functions $g_i(x)$, which can further be defined as a composition of other functions. A widely used type of composition is the nonlinear weighted sum,

$$f(x)=K\left(\sum_i w_i g_i(x)\right)$$

Here K is some predefined function. It will be convenient for the following to refer to a collection of functions g_i as simply a vector $g = (g_1, g_2 \ldots. g_n)$.

4.7 Ensemble Approach

Ensemble methods use multiple learning algorithms to obtain better predictive performance than could be obtained from any of the constituent learning algorithms alone (Rokach, 2010). An ensemble method is itself a supervised learning algorithm, because it can be trained and then used to make predictions that tend to yield better results (Kuncheva & Whitaker, 2003; Sollich & Krogh, 1996).

The popular approaches for combining classifiers are voting and stacking. In voting approach, the class predicted by majority of the models is selected, whereas in stacking approach the predictions from each different model is given as input to a meta-level classifier whose output is the final class. In voting no learning takes place at the meta level, as the final classification is decided by the majority of votes casted by the base level classifiers whereas in stacking learning takes place at the meta level. Whether it is voting or stacking, there are two ways of making an ensemble: homogenous and heterogeneous ensemble. The ensemble technique where the classifiers are of same type is called homogeneous ensemble and where the classifiers are different, it is called heterogeneous ensemble.

4.7.1 Stacking

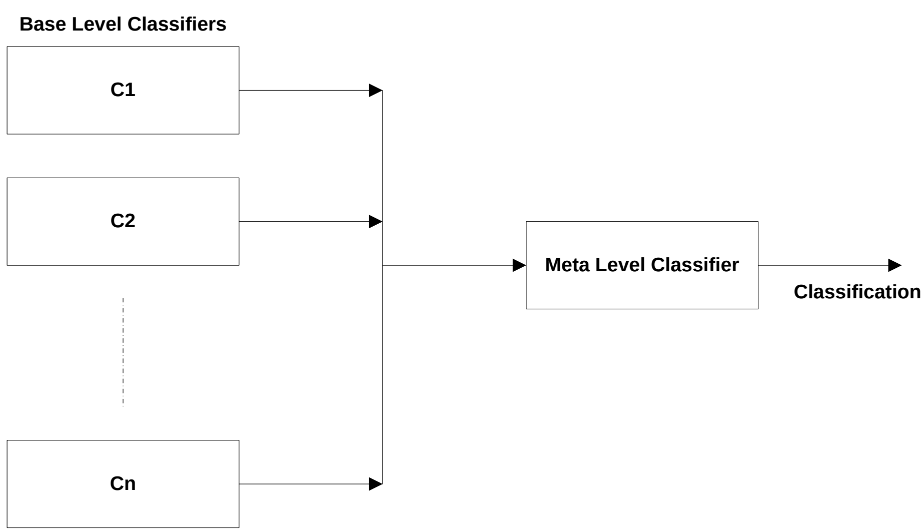

Figure 8: Ensemble Model Using Stacking Approach

Stacking is the process of combining multiple classifiers generated by different learning algorithms on a single dataset. In the first phase, a set of base level classifiers C_1, C_2...C_n is generated. In the second phase, a meta level classifier is developed by combining the base level classifier. In this work, the effort is made how the performance of classifier can be improved using the stacking approach. While conventional data mining research focuses on how the performance of a single model can be improved, this work focuses on how heterogeneous classifiers can be combined to improve classifier performance.

4.7.2 Voting

The simplest way to combine the output of multiple classifiers is within a voting framework. Let $C_{1,}$ C_2 ...C_n be the set of classifiers that are induced by training n different learning algorithms. To classify a new instance at runtime, the classifiers $C_{1,}$ C_2 ...C_n are queried for a class value and the class with the highest count is finally selected which is known as majority voting. The variations include weighted majority voting and voting using class probability distributions (Dietterich, 1997). In the probabilistic approach, each classifier outputs a probability distribution vector over all relevant classes. For each class, the individual probability values are averaged by all classifiers, and the class with the maximum value is finally selected (Sigletos, Paliouras, Spyropoulos, & Hatzopoulos, 2005). The voting approach on a set $C_{1,}$ C_2 ...C_n of classifiers in boosting (Shapire, Freund, Bartlett, & Lee, 1998) and bagging (Breiman, 1996) are generated by applying a single learning algorithm to "n" different versions of a given data set, rather than training "n" different algorithms.

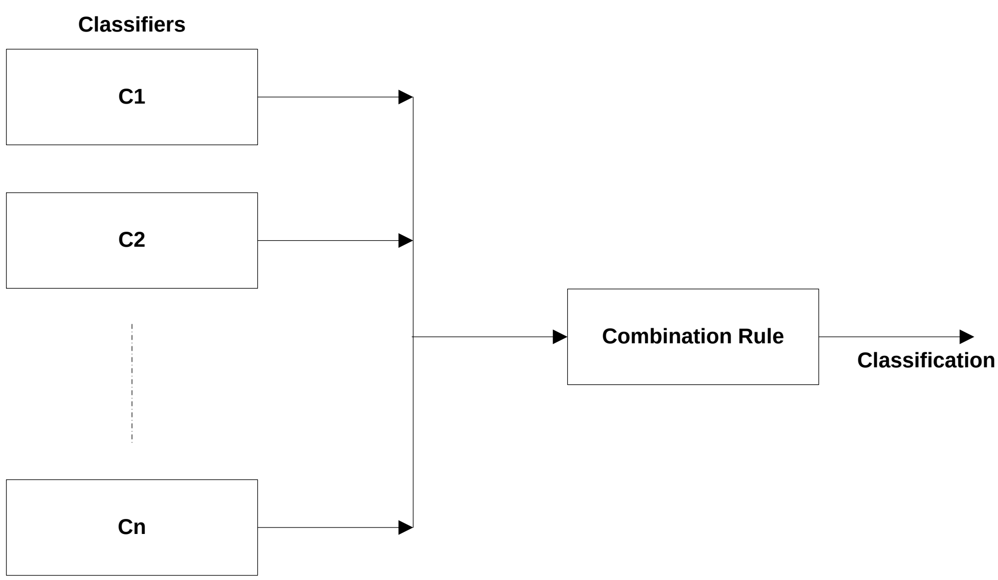

Figure 9: Ensemble Model Using Voting Approach

Chapter 5

Results and Discussion

In this section, we present a comparative analysis of alternative feature sets and algorithms in classifying alarms.

5.1 Bradycardia (Brady)

5.1.1 Comparative Analysis in Time domain

We used 30, 60, and 90 and 120 minutes of time window to investigate the efficacy of classification algorithms to determine under which time domain, the false alarm rates can be minimized, retaining the true alarm suppression rate.

5.1.1.1 Time Domain with Mean Value

Table 5: Comparison of Mean Value in Time Domain for Brady

	30 Min		**60 Min**		**90 Min**		**120 Min**	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	12.45	71.56	11.63	73.39	13.27	76.61	11.43	76.15
Random Forest	10.20	75.23	11.43	85.32	10.41	83.94	8.98	81.19
BayesNet	21.22	52.75	15.71	54.59	15.51	51.38	14.49	56.88
NaiveBayes	25.31	52.29	24.49	48.62	22.65	49.54	20.00	47.71
Multilayer Perceptron	12.65	77.52	11.22	78.44	11.63	72.94	10.61	71.56

When the mean value was taken in consideration in time domain analysis, we observed that Random Forest outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates (S-Rate). We also observed in Random Forest that true alarm and false alarm suppression rate was initially increasing when time window was increased from 30 minutes to 60 minutes. However, when time window was still increased, both true alarm and false alarm suppression rates started decreasing.

Furthermore, in Multilayer Perceptron, when time window was increased from 30 to 60 minutes, the true alarm suppression rate was initially decreasing where as false alarm suppression rate was increasing, but as time window is increased to 90 minutes; the alarm suppression rate was reverse i.e. true alarm suppression rate is increased and false alarm suppression rate was decreased. Our objective is to achieve high rate of false alarm suppression and low rate of true alarm suppression. From the Table 5, we observed Random Forest in 60 min time window achieved the highest false alarm suppression rate of 85.32% with 11.43 true alarm suppression rate and 120 min time window has the lowest true alarm suppression rate of 8.98% with 81.19% of false alarm suppression.

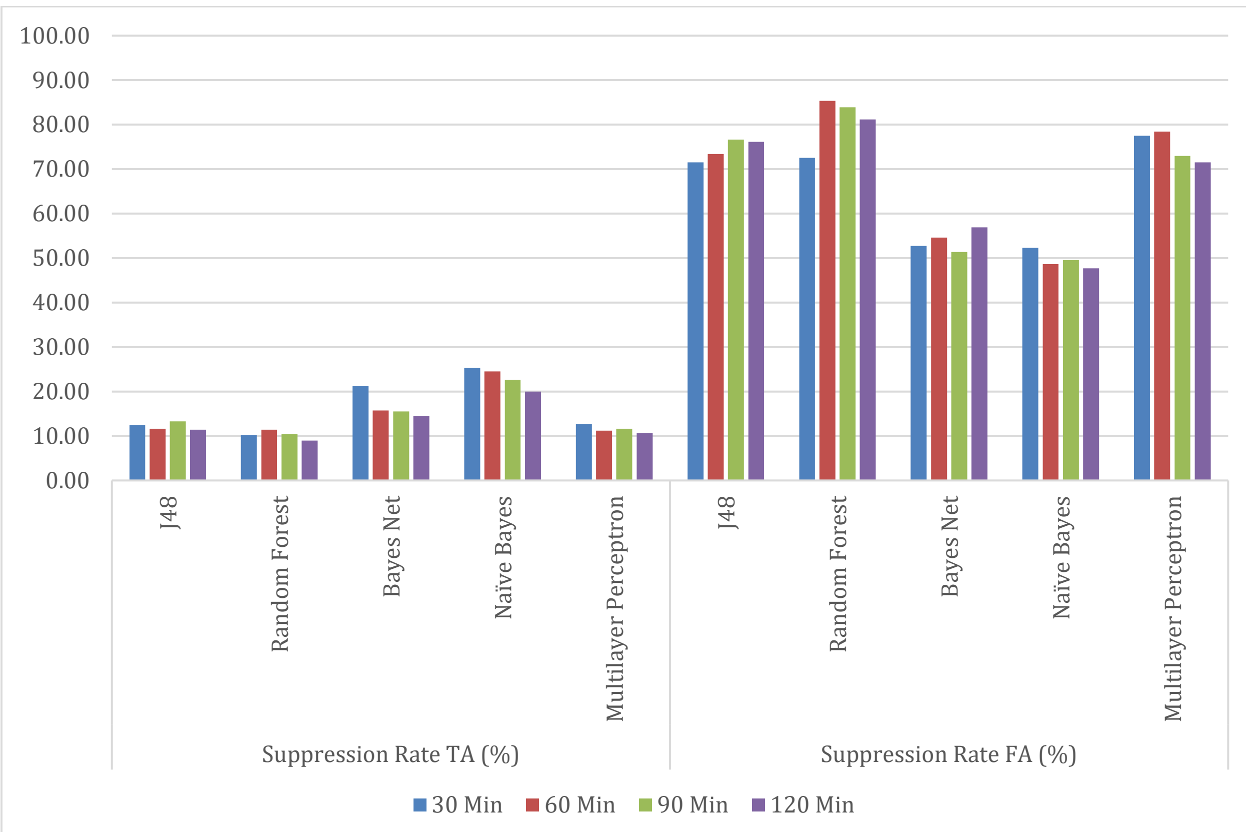

Figure 10: True Alarm & False Alarm Suppression Rates with Mean Value in Time Dimension for Brady

5.1.1.2 Time Domain with Median Value

From Table 6, when the median data transformation was taken in consideration in time domain analysis, we observed that Random Forest still outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates. We also observed in Random Forest that true alarm suppression rate has decreasing trend when time window was increased from 30 minutes to 120 minutes, and false alarm suppression rate has increasing trend.

Table 6: Comparison of Median Value in Time Domain for Bardy

	30 Min		60 Min		90 Min		120 Min	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	15.10	73.39	12.04	75.69	10.61	75.69	12.86	82.57
Random Forest	11.22	79.82	10.61	83.49	10.61	83.03	10.41	87.16
BayesNet	19.80	65.14	21.43	72.94	18.16	68.35	18.16	73.39
NaiveBayes	26.12	51.38	24.49	54.13	24.90	55.05	23.67	55.05
Multilayer Perceptron	12.65	77.52	11.43	74.77	13.47	78.44	10.82	71.10

However, the true alarm suppression rate was constant and suppression rates for false alarms almost similar when time window was increased from 60 to 90 min. Moreover, in Multilayer Perceptron, when time window was increased from 30 to 60 minutes, both true alarm suppression rate and false alarm suppression rate was initially decreasing, but as time window is increased to 90 minutes; both alarm suppression rate was increased, and again increased in time window to 120 min, both true alarm and false alarm suppression rate starts decreasing.

We also observed Random Forest in 120 min time window achieved the highest false alarm suppression rate of 87.16% and 10.41% true alarm suppression rates.

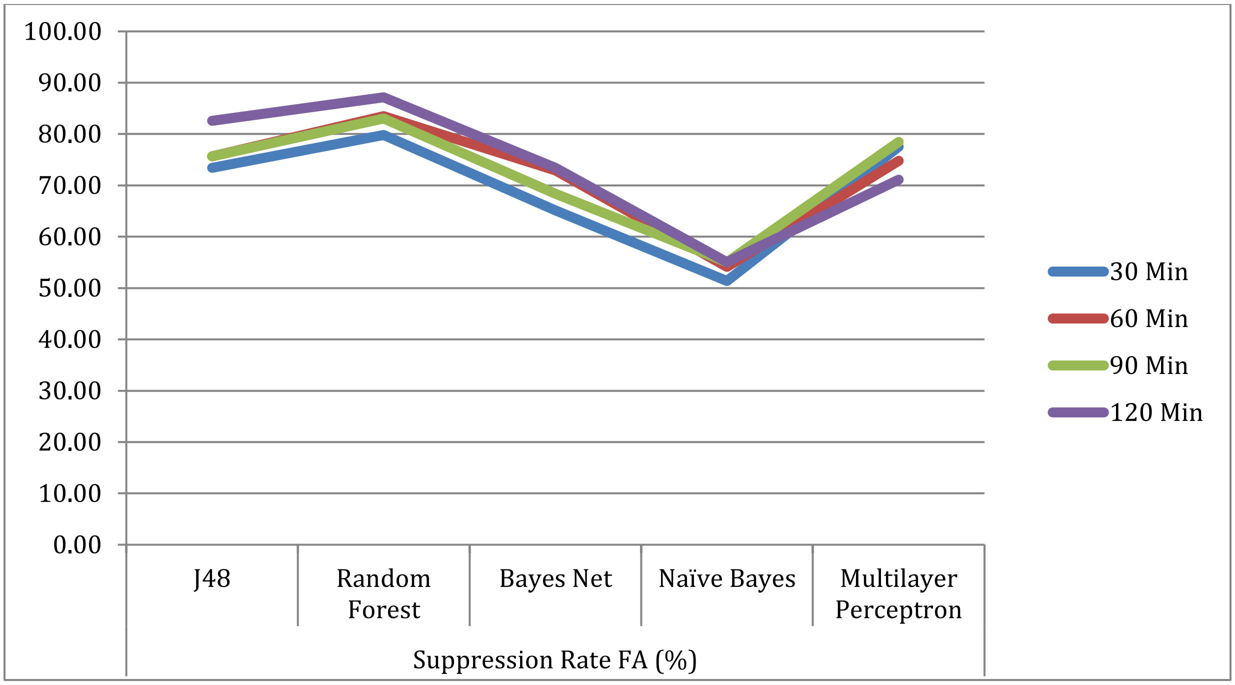

Figure 11: False Alarm Suppression Rate with Median Value in Time Dimension for Brady

5.1.1.3 Time Domain with Standard Deviation Value

Table 7: Comparison of Standard Deviation Value in Time Domain for Brady

	30 Min		**60 Min**		**90 Min**		**120 Min**	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	11.02	71.56	8.57	71.10	10.00	72.94	7.76	70.18
Random Forest	8.78	76.61	7.55	80.73	6.94	79.82	7.55	77.52
BayesNet	19.59	52.75	14.69	52.29	13.88	55.96	15.71	49.54
NaiveBayes	15.51	47.71	17.14	46.79	14.49	56.88	15.10	65.60
Multilayer Perceptron	8.98	69.72	10.41	66.97	9.18	70.18	9.39	72.94

In Table 7, the data was transformed through standard deviation with varying time window; we observed that Random Forest performed best among other classification algorithms. We also observed in Random Forest that when time window was increased from 30 to 60 minutes, the true alarm suppression rate was initially decreasing and the false alarm suppression rate was increasing, but when time window was increased to 90 min both alarm suppression rate was decreasing, and when time window increased to 120 min false alarm suppression rate starts increasing, and true alarm suppression rates starts decreasing.

Furthermore, in Multilayer Perceptron, when time window was increased from 30 to 60 minutes, true alarm suppression rate was initially increasing, whereas false alarm rate was decreasing, but as time window is increased to 90 minutes; it was reverse, true alarm suppression rate was increased, and false alarm suppression rate was decreased. Again increased in time window to 120 min, both true alarm and false alarm suppression rate starts increasing.

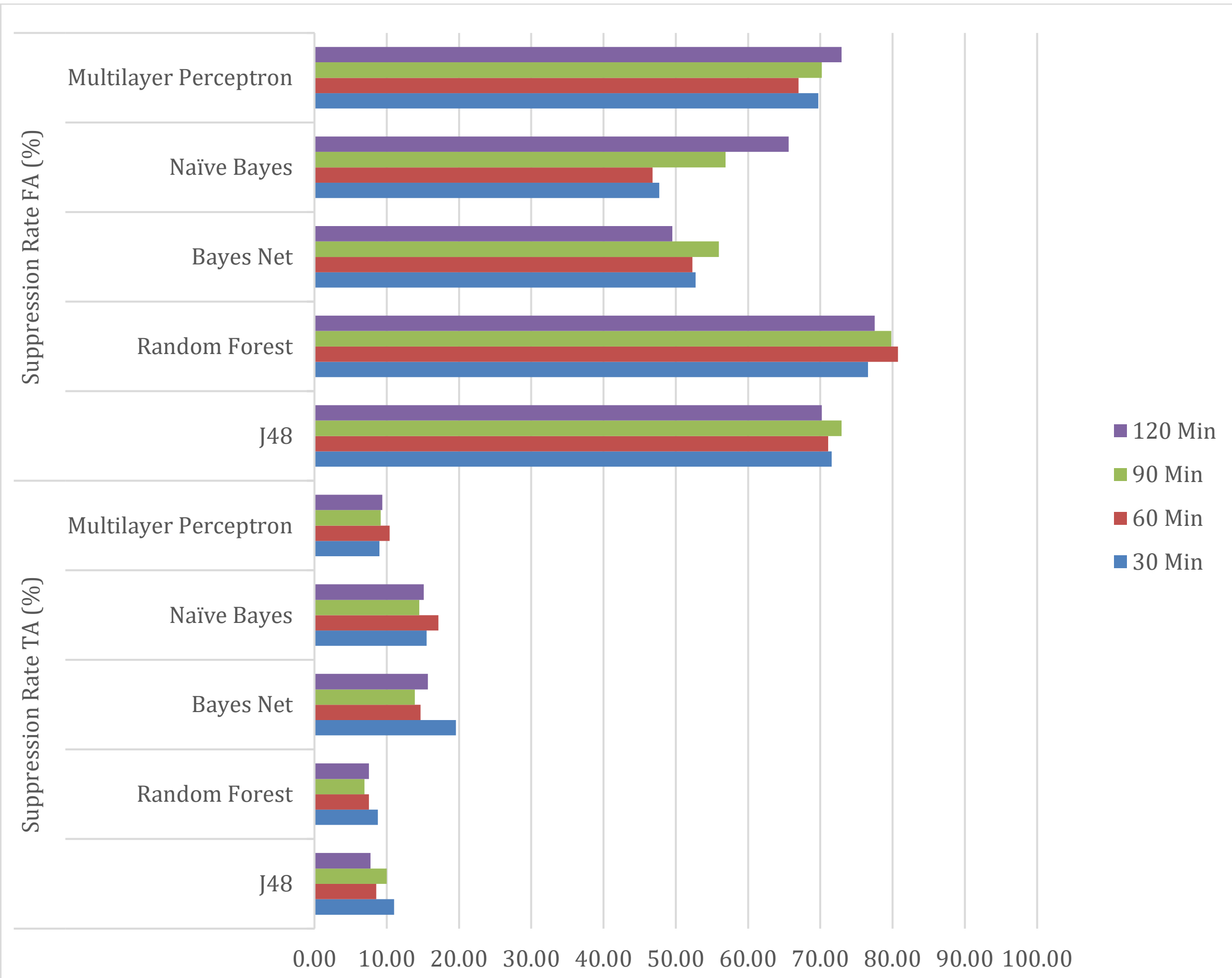

Figure 12: False Alarm & True Alarm Suppression Rates with Median Value in Time Dimension for Brady

5.1.1.4 Time Domain with DFT Value

Table 8: Comparison of DFT Value in Time Domain for Brady

	30 Min		60 Min		90 Min		120 Min	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	15.51	61.47	11.43	66.51	12.45	67.89	10.20	67.43
Random Forest	14.49	69.27	12.24	74.31	11.63	76.61	9.39	77.52
BayesNet	21.02	62.84	18.37	63.76	16.73	63.30	14.08	56.42
NaiveBayes	17.76	53.21	13.67	46.33	16.94	52.29	21.02	57.80
Multilayer Perceptron	12.24	67.43	10.20	66.97	8.78	74.77	11.02	69.72

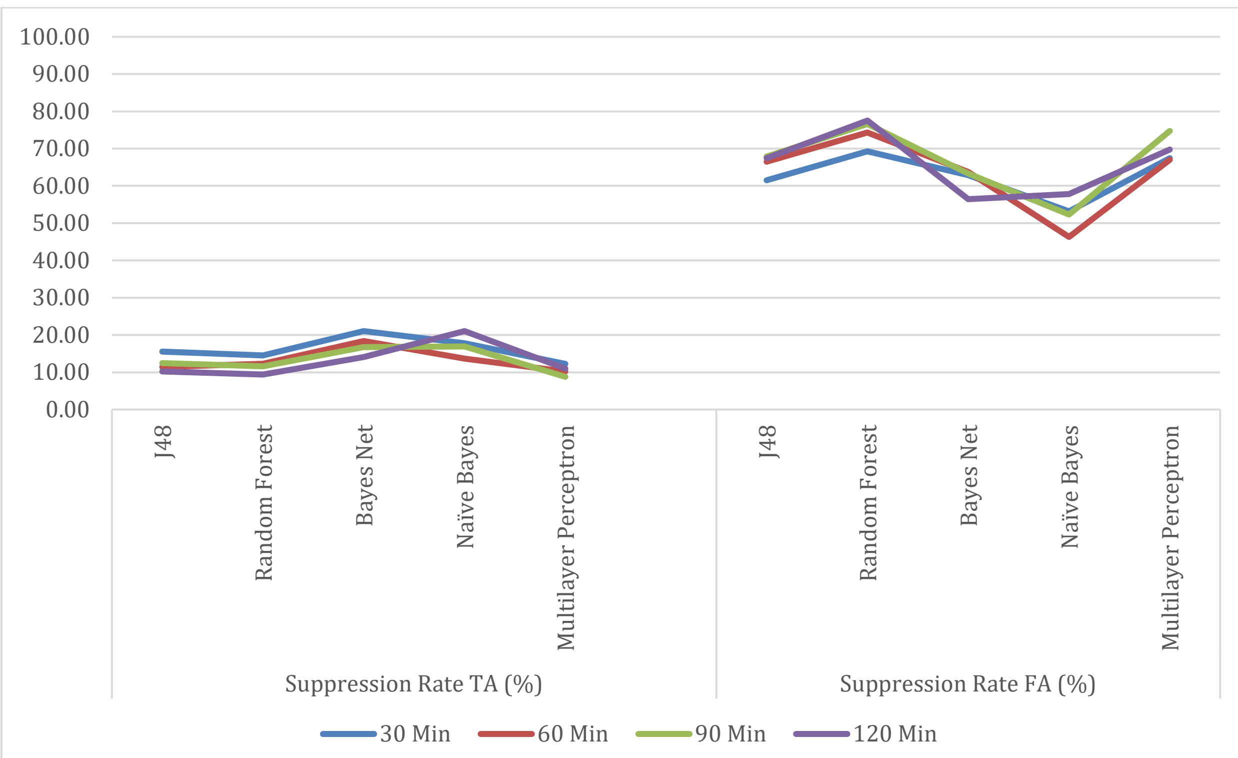

Figure 13: False Alarm & True Alarm Suppression Rates with DFT Value in Time Dimension for Brady

In Table 8, the data was transformed through DFT with varying time window; we observed that Random Forest performed best among other classifiers. We also observed in Random Forest that when time window was increased from 30 to 120 minutes, the true alarm suppression rate was decreasing and the false alarm suppression rate was increasing. However, in J48, when time window was increased from 30 to 60 minutes, true alarm suppression rate was initially decreasing, whereas false alarm rate was increasing, but as time window is increased to 90 minutes, true alarm suppression rate was increased, and false alarm suppression rate was still increased. Again increased in time window to 120 min, true alarm suppression rate starts decreasing and false alarm suppression rate was almost constant.

5.1.2 Comparative Analysis with Data Transformation

We transform data through mean, median, standard deviation, and DFT.

5.1.2.1 Data Transformation in 30 Minutes Time Window

Table 9: Comparison of Data Transformation in 30 Minutes Time Window for Brady

	Mean		Median		Std. Deviation		DFT	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	12.45	71.56	15.10	73.39	11.02	71.56	15.51	61.47
Random Forest	10.20	75.23	11.22	79.82	8.78	76.61	14.49	69.27
BayesNet	21.22	52.75	19.80	65.14	19.59	52.75	21.02	62.84
NaiveBayes	25.31	52.29	26.12	51.38	15.51	47.71	17.76	53.21
Multilayer Perceptron	12.65	77.52	12.65	77.52	8.98	69.72	12.24	67.43

When 30-minute time window was taken in consideration with various data transformation, we observed that Random Forest outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates. We also observed in Random Forest that true alarm and false alarm suppression rate was initially increasing when data transformation was altered from mean to median.

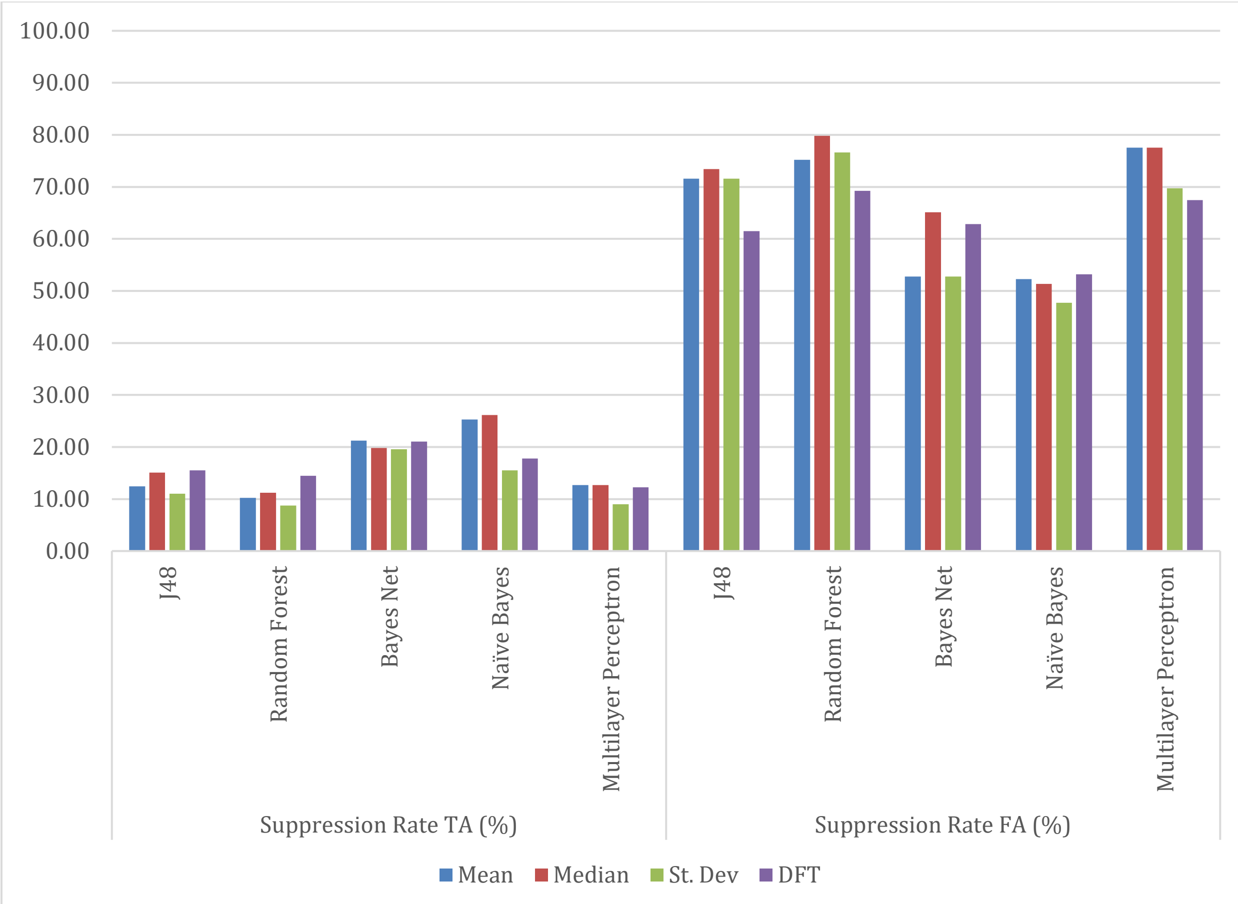

Figure 14: False Alarm & True Alarm Suppression Rates with Data Transformation for 30 Minutes Time Window for Brady

However, when data transformation was changed to standard deviation, both true alarm and false alarm suppression rates started decreasing. Again data transformation technique was altered to DFT, the true alarm suppression rates started increasing and false alarm started decreasing. Furthermore, BayesNet and NaiveBayes had high true alarm suppression rates and low false alarm suppression rates.

5.1.2.2 Data Transformation in 60 Minutes Time Window

Considering 60-minutes time window with various data transformation technique, we observed that Random Forest outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates. We also observed from Table 10, Random Forest with standard deviation perform the best with low true suppression rate of 7.55% and false alarm suppression rate 0f 80.73%.

Table 10: Comparison of Data Transformation in 60 Minutes Time Window for Brady

	Mean		Median		Std. Deviation		DFT	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	11.63	73.39	12.04	75.69	8.57	71.10	11.43	66.51
Random Forest	11.43	85.32	10.61	83.49	7.55	80.73	12.24	74.31
BayesNet	15.71	54.59	21.43	72.94	14.69	52.29	18.37	63.76
NaiveBayes	24.49	48.62	24.49	54.13	17.14	46.79	13.67	46.33
Multilayer Perceptron	11.22	78.44	11.43	74.77	10.41	66.97	10.20	66.97

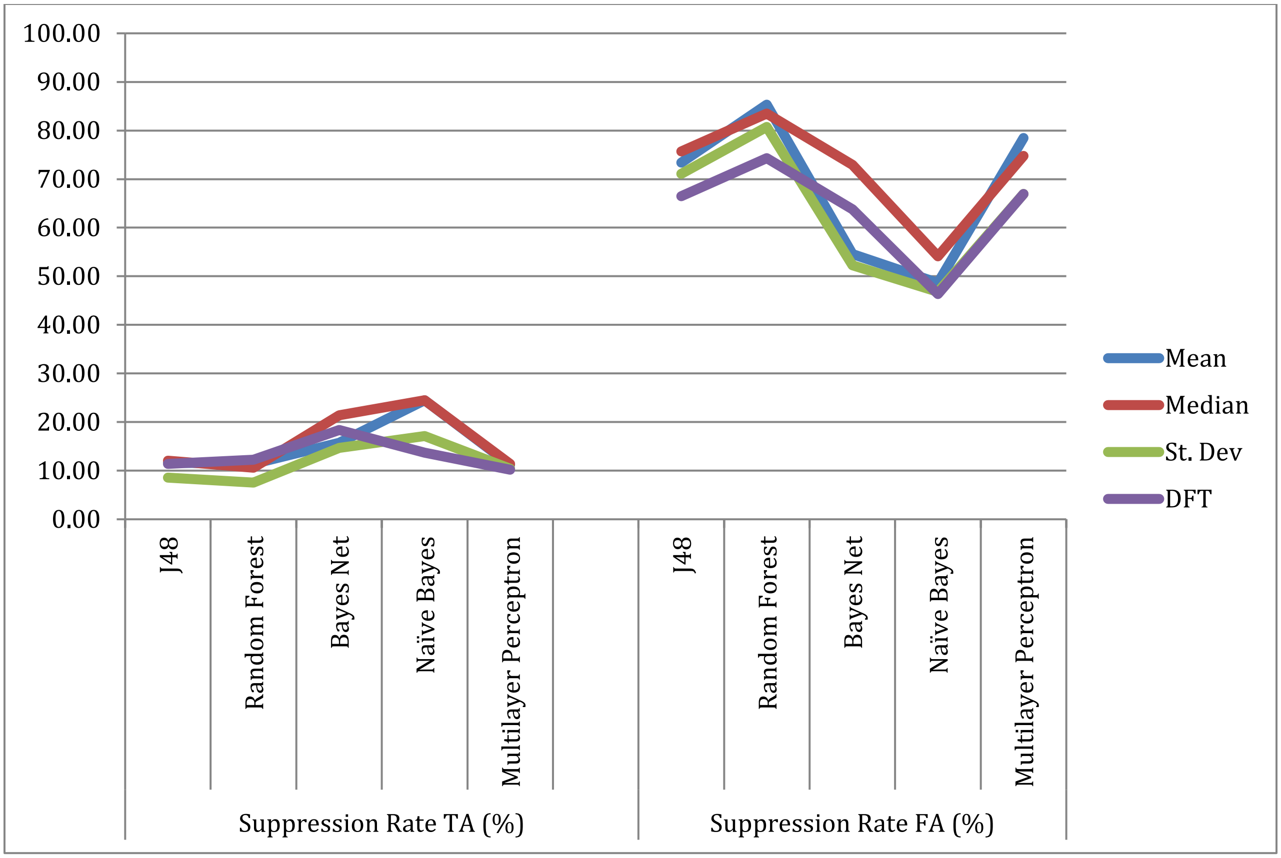

Figure 15: False Alarm & True Alarm Suppression Rates with Data Transformation in 60 Minutes Time Window for Brady

5.1.2.3 Data Transformation in 90 Minutes Time Window

Table 11: Comparison of Data Transformation in 90 Minutes Window for Brady

	Mean		**Median**		**Std. Deviation**		**DFT**	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	13.27	76.61	10.61	75.69	10.00	72.94	12.45	67.89
Random Forest	10.41	83.94	10.61	83.03	6.94	79.82	11.63	76.61
BayesNet	15.51	51.38	18.16	68.35	13.88	55.96	16.73	63.30
NaiveBayes	22.65	49.54	24.90	55.05	14.49	56.88	16.94	52.29

Multilayer Perceptron	11.63	72.94	13.47	78.44	9.18	70.18	8.78	74.77

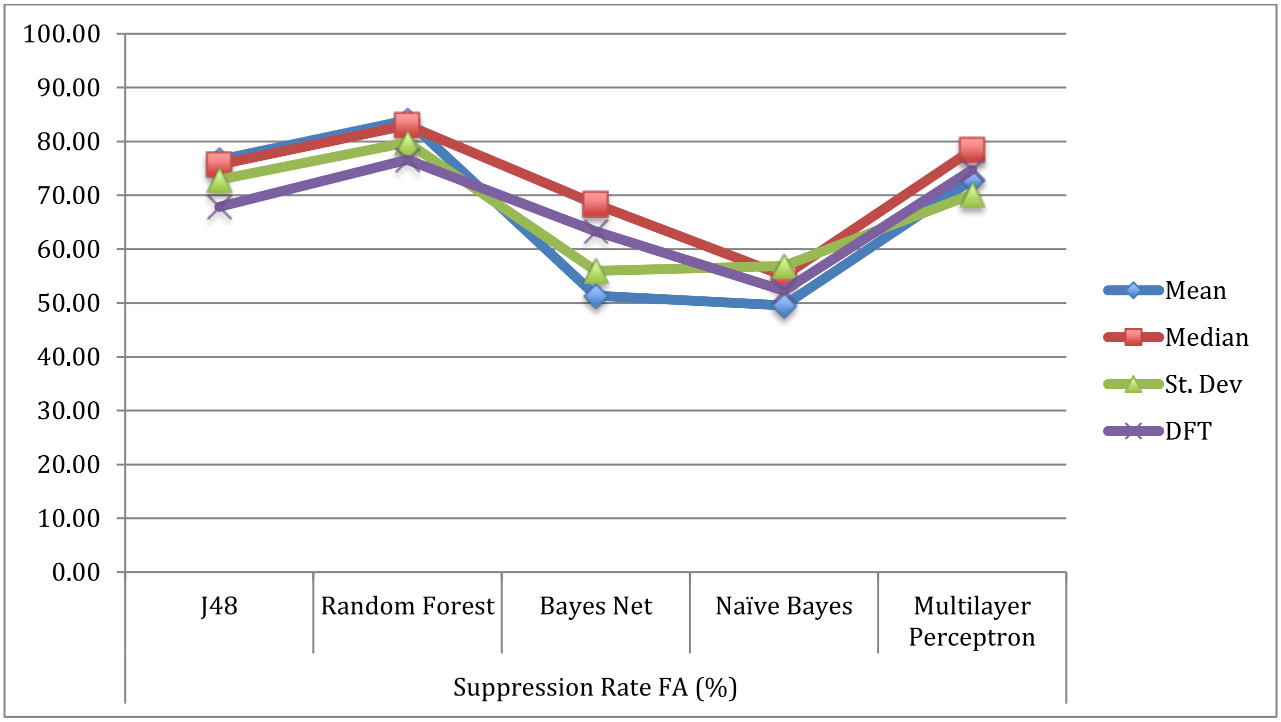

Figure 16: False Alarm Suppression Rates with Data Transformation in 90 Minutes Time Window for Brady

When 90-minutes time window was taken in consideration with various data transformation, we observed that Random Forest outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates. Furthermore, BayesNet and NaiveBayes had high true alarm suppression rates and low false alarm suppression rates.

5.1.2.4 Data Transformation in 120 Minutes Time Window

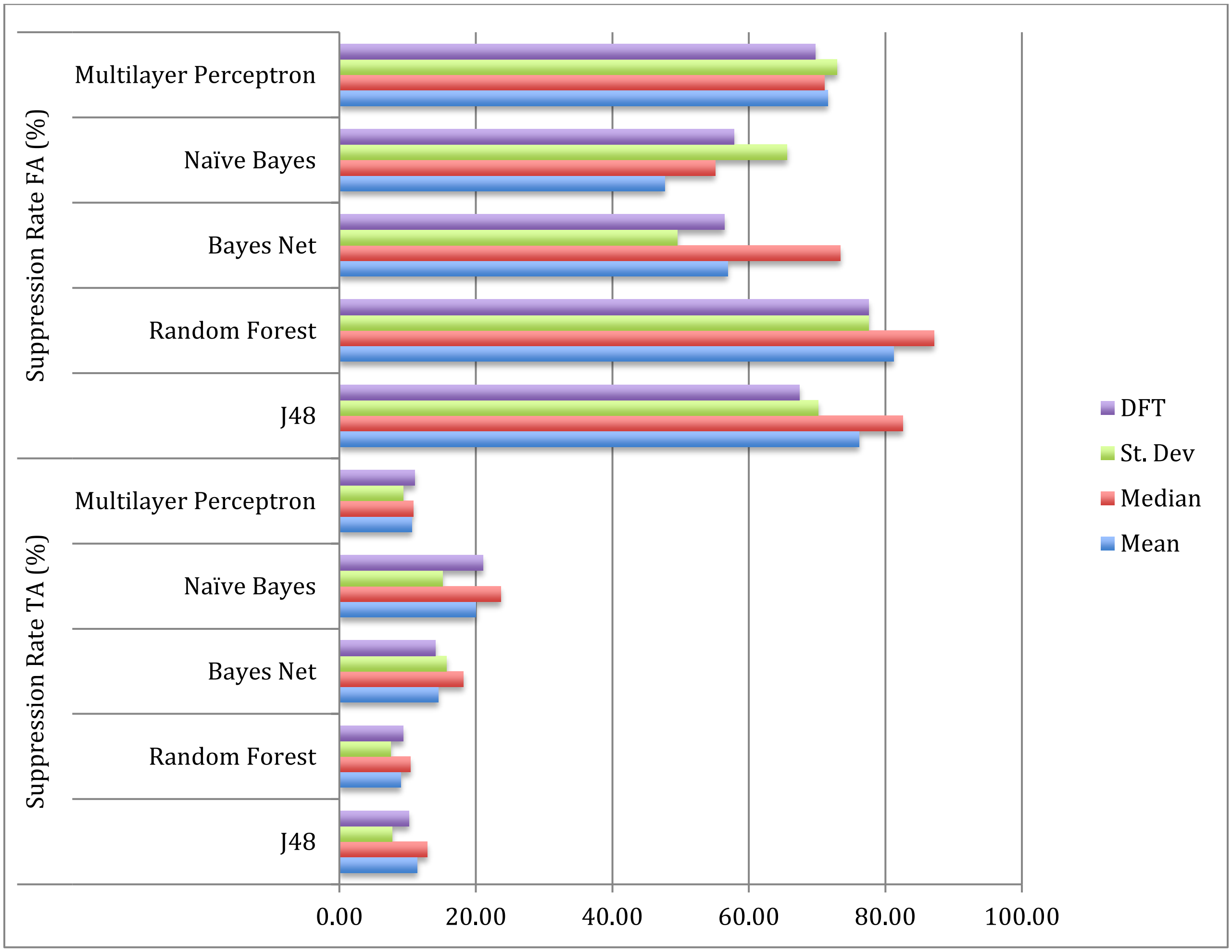

Figure 17: False Alarm & True Alarm Suppression Rates with Data Transformation in 120 Minutes Time Window for Brady

Considering 120-minutes time window with various data transformation technique, we observed that Random Forest outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates.

Table 12: Comparison of Data Transformation in 120 Minutes Time Window for Brady

	Mean	**Median**	**Std. Deviation**	**DFT**

Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	11.43	76.15	12.86	82.57	7.76	70.18	10.20	67.43
Random Forest	8.98	81.19	10.41	87.16	7.55	77.52	9.39	77.52
BayesNet	14.49	56.88	18.16	73.39	15.71	49.54	14.08	56.42
NaiveBayes	20.00	47.71	23.67	55.05	15.10	65.60	21.02	57.80
Multilayer Perceptron	10.61	71.56	10.82	71.10	9.39	72.94	11.02	69.72

5.1.3 Comparative Analysis with Feature Sets

5.1.3.1 Feature Sets with Mean Value

We use different feature sets such as CFS Subset Evaluator, Wrapper Subset Evaluator (using Bayes Net, NaiveBayes, J48, Random Forest), and Info Gain Evaluator.

5.1.3.1.1 Feature Sets with Mean Value in 30 Minutes Window

Table 13: Comparison of Feature Sets with Mean Value in 30 Minutes Time Window for Brady

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	13.27	72.02	18.16	61.47	13.67	72.02	11.84	64.68
Random Forest	10.61	76.15	14.29	62.39	9.80	77.06	11.43	75.23
BayesNet	20.82	57.80	23.27	61.01	18.57	51.83	12.65	42.66
NaiveBayes	21.02	47.71	13.27	41.28	22.45	52.29	20.41	48.62
Multilayer Perceptron	13.06	68.81	15.51	59.63	13.67	72.02	9.80	61.01

Considering 30-minutes time window with mean value, we observed that feature selection obtained from Wrapper method including J48 performed best with Random Forest as classifier in comparison to the feature sets obtained from CFS, Wrapper method including NaiveBayes, Wrapper method including Random Forest, and Information gain. Wrapper method including J48 resulted in low true alarm suppression rate of 9.8% and high false alarm suppression rate of 77.46%.

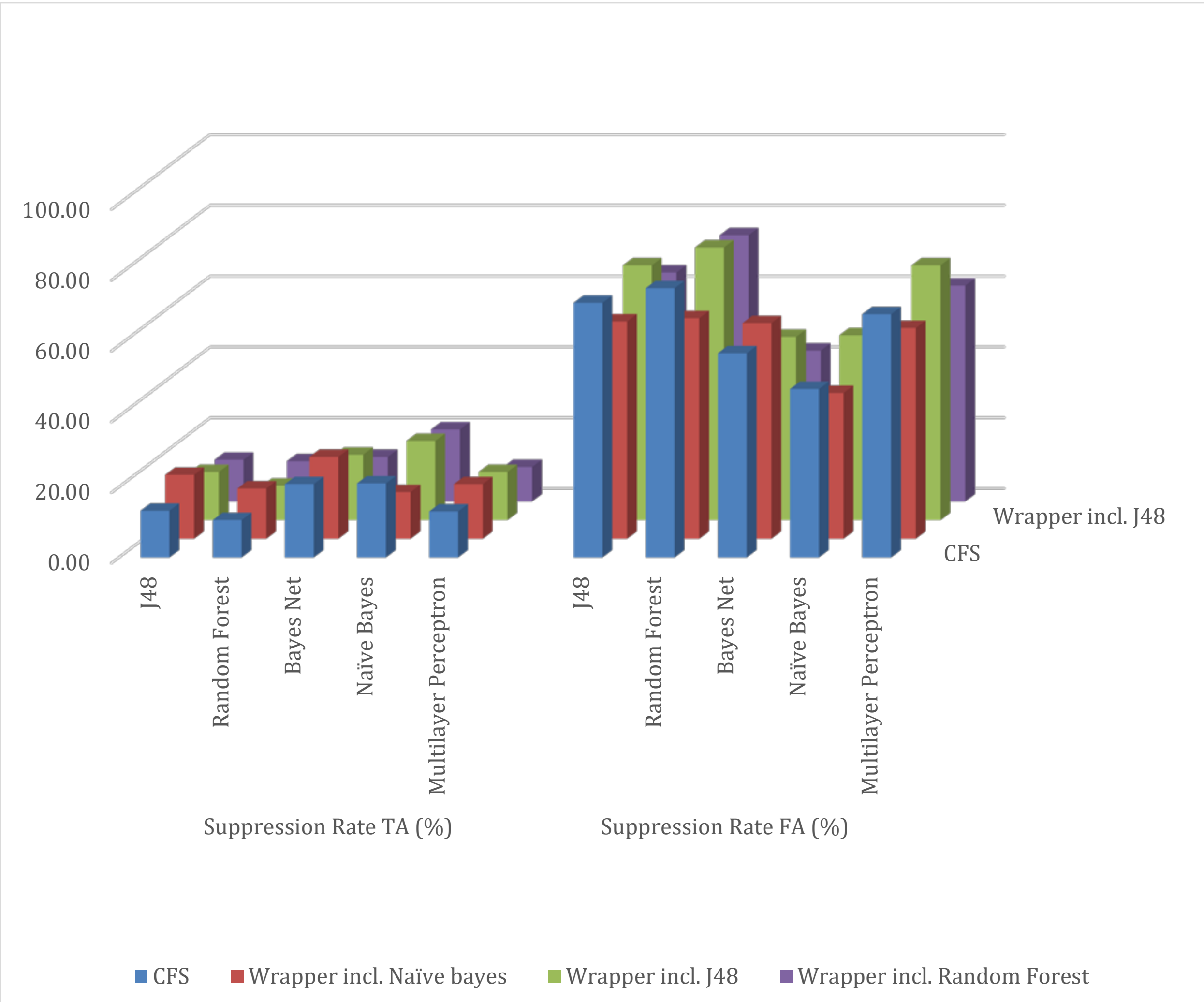

Figure 18: Comparison of Feature Sets with Mean Value in 30 Minutes Time Window for Brady True & False Alarm Suppression Rates

5.1.3.1.2 Feature Sets with Mean Value in 60 Minutes Window

Table 14: Comparison of Feature Sets with Mean Value in 60 Minutes Time Window for Brady

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	12.86	73.85	15.10	74.31	11.43	68.35	8.98	68.35
Random Forest	10.82	82.57	11.02	79.82	10.41	78.90	9.18	80.28
BayesNet	16.53	61.01	16.94	54.13	18.16	58.26	12.65	44.95
NaiveBayes	18.98	49.54	13.88	46.33	16.94	48.62	21.22	44.04
Multilayer Perceptron	13.88	60.09	11.43	72.02	11.43	75.23	12.86	67.43

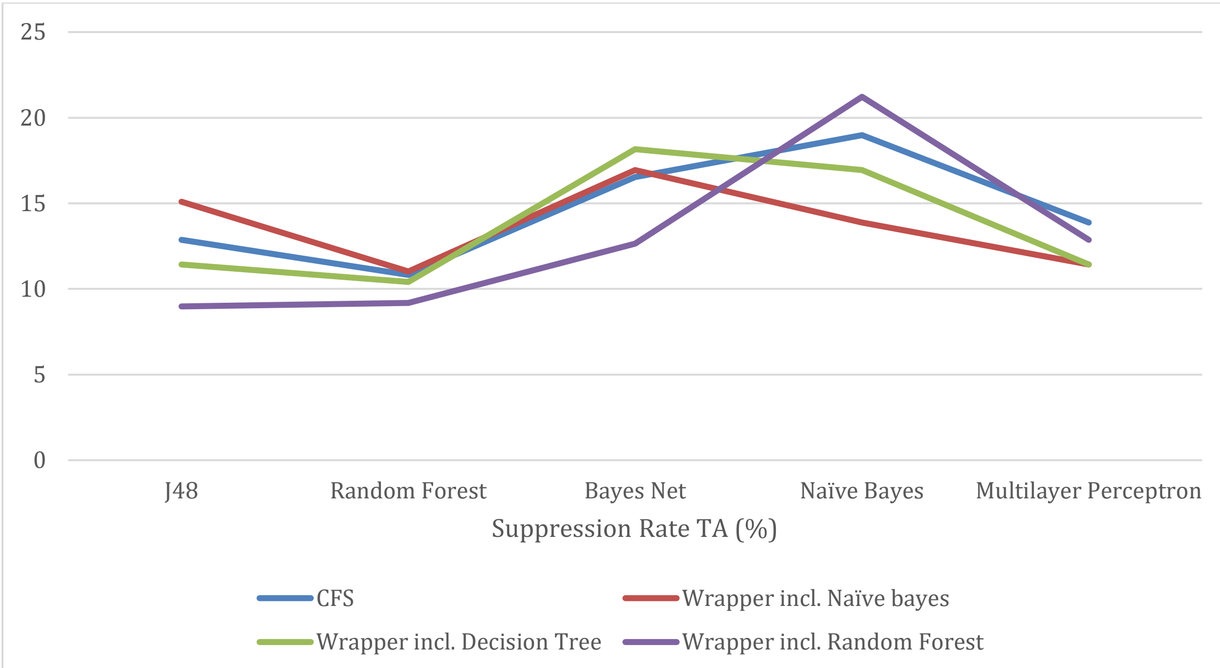

Figure 19: Comparison of Feature Sets with Mean Value in 60 Minutes Time Window for Brady True Alarm Suppression Rates

5.1.3.1.3 Feature Sets with Mean Value in 90 Minutes Window

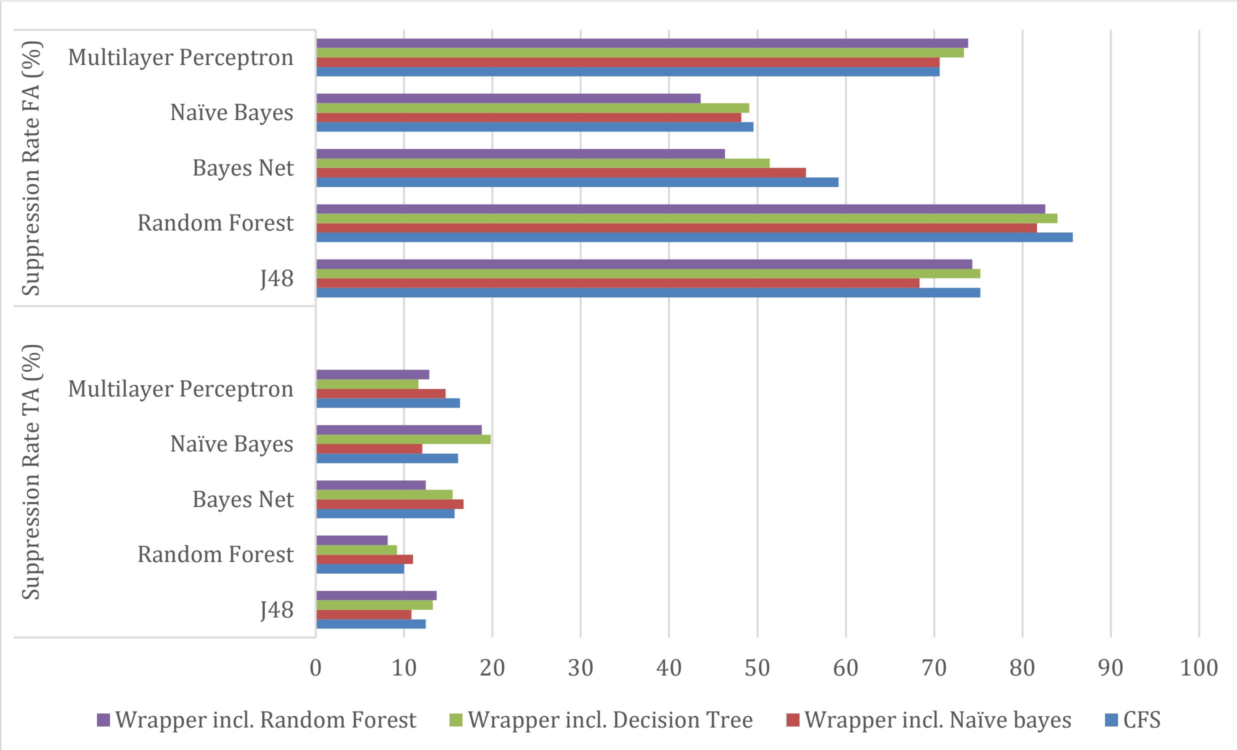

Figure 20: Comparison of Feature Sets with Mean Value in 90 Minutes Time Window for Brady True & False Alarm Suppression Rates

Table 15: Comparison of Feature Sets with Mean Value in 90 Minutes Time Window for Brady

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	12.45	75.23	10.82	68.35	13.27	75.23	13.67	74.31
Random Forest	10.00	85.71	11.02	81.65	9.18	83.94	8.16	82.57
BayesNet	15.71	59.17	16.73	55.50	15.51	51.38	12.45	46.33
NaiveBayes	16.12	49.54	12.04	48.17	19.80	49.08	18.78	43.58
Multilayer Perceptron	16.33	70.64	14.69	70.64	11.63	73.39	12.86	73.85

5.1.3.1.4 Feature Sets with Mean Value in 120 Minutes Window

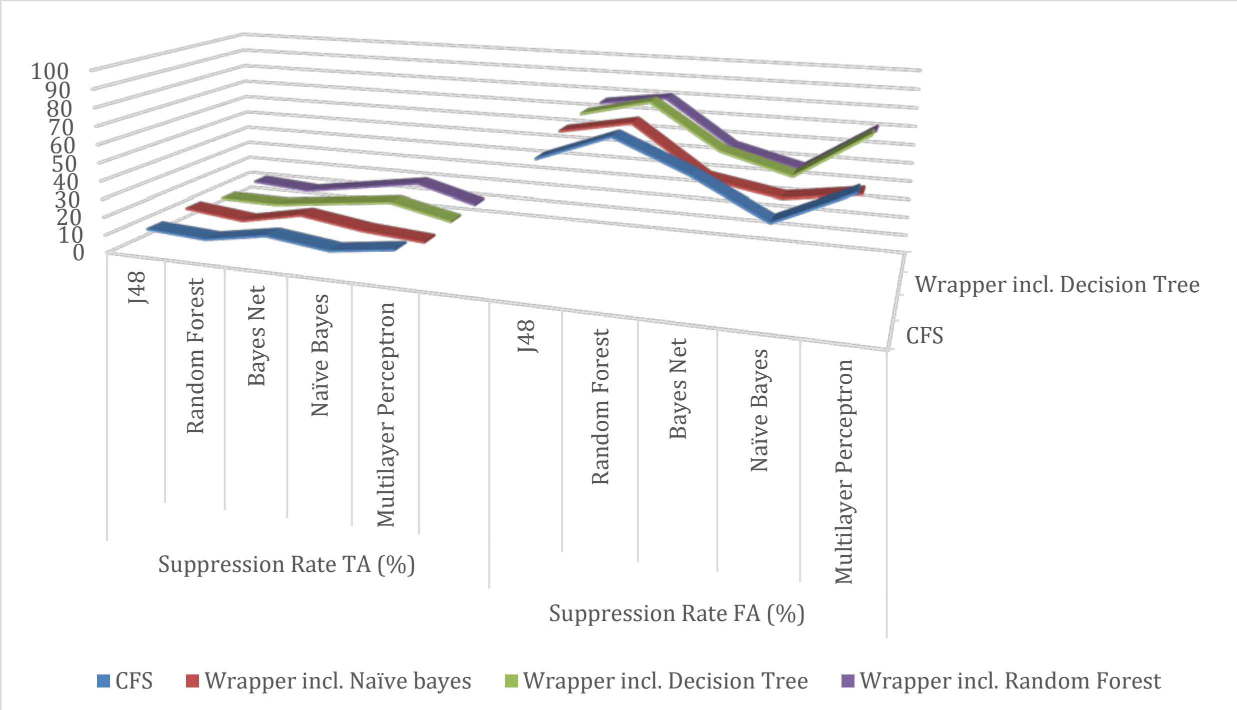

Figure 21: Comparison of Feature Sets with Mean Value in 120 Minutes Time Window for Brady True & False Alarm Suppression Rates

Table 16: Comparison of Feature Sets with Mean Value in 120 Minutes Time Window for Brady

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	12.04	69.27	13.88	75.23	10.20	77.06	11.43	76.15
Random Forest	10.00	83.03	10.20	82.11	9.80	85.78	8.98	81.19
BayesNet	15.92	67.89	17.14	57.34	14.08	62.84	14.49	56.88
NaiveBayes	11.63	49.08	12.04	50.92	18.16	52.75	20.00	47.71
Multilayer Perceptron	16.33	66.51	9.39	56.42	10.61	77.06	10.61	71.56

Taking consideration of 60-minutes, 90-minutes, and 120-minutes time window with mean value, we observed that feature selection obtained from Wrapper method including Random Forest as classifier performed best with low true alarm suppression rate in both 60 and 90 minutes of data with comparatively high false alarm suppression rates. In 120 minutes of time window, Wrapper method including J48 performed the best in comparison to the feature sets obtained from CFS, Wrapper method including NaiveBayes, Wrapper method including Random Forest, and Information gain.

5.1.3.2 Feature Sets with Median Value

5.1.3.2.1 Feature Sets with Median Value in 30 Minutes Window

Table 17: Comparison of Feature Sets with Median Value in 30 Minutes Time Window for Brady

	CFS		Wrapper incl. Naïve Bayes		Wrapper incl. J48		Wrapper incl. Random Forest	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	13.88	69.27	16.12	38.99	14.49	70.18	11.43	72.48
Random Forest	10.61	79.82	28.78	60.55	10.61	76.61	11.02	76.15
BayesNet	16.94	67.43	12.86	35.32	18.98	66.51	20.41	66.97
NaiveBayes	17.21	43.58	11.43	34.86	20.20	50.46	13.67	43.58
Multilayer Perceptron	13.47	25.87	11.84	34.86	13.06	66.06	14.08	60.55

Considering 30-minutes, 60-minutes, 90-minutes, and 120-minutes time window with median value, we observed that feature selection obtained from CFS method with Random Forest as classifier performed best in 30 minutes time window with true alarm suppression rate of 10.61%, and false alarm suppression rate of 79.82% whereas Wrapper including Random Forest with Random Forest as classifier performed well in 60, 90, 120 minutes time window.

5.1.3.2.2 Feature Sets with Median Value in 60 Minutes Window

Table 18: Comparison of Feature Sets with Median Value in 60 Minutes Time Window for Brady

	CFS		Wrapper incl. Naïve Bayes		Wrapper incl. J48		Wrapper incl. Random Forest	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	11.84	66.51	15.51	67.43	16.53	77.52	10.61	72.48
Random Forest	10.00	82.11	13.47	76.15	12.45	76.61	9.59	84.40
BayesNet	20.00	75.23	20.20	73.39	18.37	47.25	21.43	71.56
NaiveBayes	15.92	48.17	15.31	49.54	11.02	35.78	24.29	52.29
Multilayer Perceptron	12.86	58.72	12.45	53.67	6.53	42.20	13.67	72.94

5.1.3.2.3 Feature Sets with Median Value in 90 Minutes Window

Table 19: Comparison of Feature Sets with Median Value in 90 Minutes Time Window for Brady

	CFS		Wrapper incl. Naïve Bayes		Wrapper incl. J48		Wrapper incl. Random Forest	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	12.24	72.94	15.51	77.98	9.39	77.98	12.45	81.19
Random Forest	8.78	83.03	10.82	79.82	11.02	83.03	9.39	86.70
BayesNet	16.53	74.31	18.98	77.52	17.76	73.39	17.14	74.31
NaiveBayes	15.51	50.92	13.88	49.54	20.41	53.67	19.18	59.63
Multilayer Perceptron	14.29	59.63	13.27	56.42	13.27	64.22	12.45	72.94

5.1.3.2.4 Analysis of Feature Sets with Median Value in 120 Minutes Window

Table 20: Comparison of Feature Sets with Median Value in 120 Minutes Time Window for Brady

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	14.90	81.65	20.20	76.61	15.31	83.03	11.84	73.85
Random Forest	10.61	84.86	15.92	82.57	11.22	84.86	9.59	84.86
BayesNet	16.12	72.94	12.65	45.41	16.12	72.94	16.53	61.47
NaiveBayes	14.08	44.95	8.37	37.16	17.35	50.00	22.65	50.00
Multilayer Perceptron	19.39	72.48	13.47	62.84	14.90	73.39	14.69	69.27

5.1.3.3 Feature Sets with Standard Deviation Value

5.1.3.3.1 Feature Sets with Standard Deviation Value in 30 Minutes Window

Table 21: Comparison of Feature Sets with Standard Deviation Value in 30 Minutes Time Window for Brady

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	13.27	67.43	14.49	64.22	10.41	68.81	10.82	73.39
Random Forest	11.02	69.72	13.67	67.89	10.82	78.44	10.00	79.82
BayesNet	15.31	60.55	13.47	44.50	19.80	56.88	17.76	54.59

NaiveBayes	12.24	49.54	9.18	49.54	11.02	38.53	15.92	49.08
Multilayer Perceptron	10.20	59.17	14.49	66.51	13.88	66.51	9.18	67.89

5.1.3.3.2 Feature Sets with Standard Deviation Value in 60 Minutes Window

Table 22: Comparison of Feature Sets with Standard Deviation Value in 60 Minutes Time Window for Brady

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	13.67	74.31	13.27	65.60	9.80	76.15	9.39	69.27
Random Forest	8.98	78.44	13.47	73.39	8.37	80.28	8.98	81.65
BayesNet	15.10	61.47	13.88	51.83	12.04	44.95	14.69	55.50
NaiveBayes	12.86	41.74	5.71	42.66	13.67	47.71	16.12	44.04
Multilayer Perceptron	12.86	71.56	11.22	57.34	12.24	66.97	11.63	75.23

5.1.3.3.3 Feature Sets with Standard Deviation Value in 90 Minutes Window

Table 23: Comparison of Feature Sets with Standard Deviation Value in 90 Minutes Time Window for Brady

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	12.24	75.69	12.24	71.56	7.14	74.31	12.24	77.52
Random Forest	8.37	77.52	10.82	76.61	8.37	77.98	6.94	82.11
BayesNet	16.12	65.14	10.00	42.66	16.33	64.68	12.65	51.38

NaiveBayes	13.47	48.62	12.24	57.34	11.22	53.67	14.08	53.67
Multilayer Perceptron	9.59	61.01	11.22	67.89	8.98	69.72	12.24	76.61

Among 30-minutes, 60-minutes, 90-minutes, and 120-minutes time window with standard deviation value, we observed that feature selection obtained from Wrapper including Random Forest method with Random Forest as classifier performed best in 90 minutes time window with true alarm suppression rate of 6.94%, and false alarm suppression rate of 82.11%.

5.1.3.3.4 Feature Sets with Standard Deviation Value in 120 Minutes Window

Table 24: Comparison of Feature Sets with Standard Deviation Value in 120 Minutes Time Window for Brady

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	9.80	72.02	12.24	72.48	11.22	72.02	9.59	66.51
Random Forest	8.16	75.69	9.59	75.23	9.59	76.61	8.98	79.36
BayesNet	14.29	59.17	10.41	44.50	13.88	61.01	15.92	52.75
NaiveBayes	14.08	50.00	12.04	60.55	10.82	55.96	14.49	56.42
Multilayer Perceptron	10.61	65.14	11.22	72.02	11.22	62.39	10.20	75.69

5.1.3.4 Feature Sets with DFT Value

5.1.3.4.1 Feature Sets with DFT Value in 30 Minutes Window

Table 25: Comparison with Feature Sets with DFT Value in 30 Minutes Time Window for Brady

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	11.02	63.76	15.92	48.17	12.65	67.43	10.00	57.80
Random Forest	13.06	69.72	15.92	57.80	13.27	68.81	11.22	77.52
BayesNet	17.96	64.22	15.31	53.67	14.69	54.13	18.37	63.76
NaiveBayes	11.43	54.13	9.39	55.05	9.80	37.16	13.67	48.17
Multilayer Perceptron	13.67	58.26	13.88	44.50	11.63	67.89	8.78	31.19

5.1.3.4.2 Feature Sets with DFT Value in 60 Minutes Window

Table 26: Comparison of Feature Sets with DFT Value in 60 Minutes Time Window for Brady

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	14.49	72.02	14.08	49.08	10.20	72.48	13.47	69.72
Random Forest	12.24	78.44	15.10	66.06	10.41	75.23	10.20	76.61
BayesNet	17.14	67.43	18.57	55.50	15.92	44.95	16.53	50.00
NaiveBayes	14.49	57.80	8.37	48.17	19.80	51.83	20.00	47.25
Multilayer Perceptron	9.59	69.72	12.04	48.62	9.80	64.22	10.82	61.01

5.1.3.4.3 Feature Sets with DFT Value in 90 Minutes Window

Table 27: Comparison with Feature Sets with DFT Value in 90 Minutes Time Window for Brady

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	12.04	65.14	11.43	54.13	10.61	76.15	11.02	74.31
Random Forest	10.41	75.69	12.24	63.30	10.20	78.90	9.80	82.57
BayesNet	13.27	71.10	11.43	53.67	16.53	56.42	15.71	62.39
NaiveBayes	17.14	56.42	11.22	56.88	15.71	41.74	13.27	42.20
Multilayer Perceptron	11.84	71.10	12.65	58.26	10.61	57.34	6.94	53.21

Among 30-minutes, 60-minutes, 90-minutes, and 120-minutes time window with DFT, we observed that feature selection obtained from Wrapper including Random Forest method with Random Forest as classifier performed best in 120 minutes time window with true alarm suppression rate of 9.39%, and false alarm suppression rate of 81.19%.

5.1.3.4.1 Feature Sets with DFT Value in 120 Minutes Window

Table 28: Comparison of Feature Sets with DFT Value in 120 Minutes Time Window for Brady

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	9.59	62.39	11.84	51.38	10.41	72.02	10.20	76.15
Random Forest	11.43	77.52	17.35	59.17	9.39	79.36	9.39	81.19
BayesNet	16.94	66.51	11.02	47.71	18.98	67.89	15.51	44.95
NaiveBayes	18.57	61.93	10.41	52.75	15.71	35.78	21.63	47.71

Multilayer Perceptron	9.59	69.27	10.82	49.54	13.27	63.30	14.08	67.43

5.2 Tachycardia (Tachy)

5.2.1 Comparative Analysis in Time domain

We used 30, 60, and 90 and 120 minutes of window to investigate the efficacy of classification algorithms to determine under which time domain, the false alarm rates can be minimized.

5.2.1.1 Time Domain with Mean Value

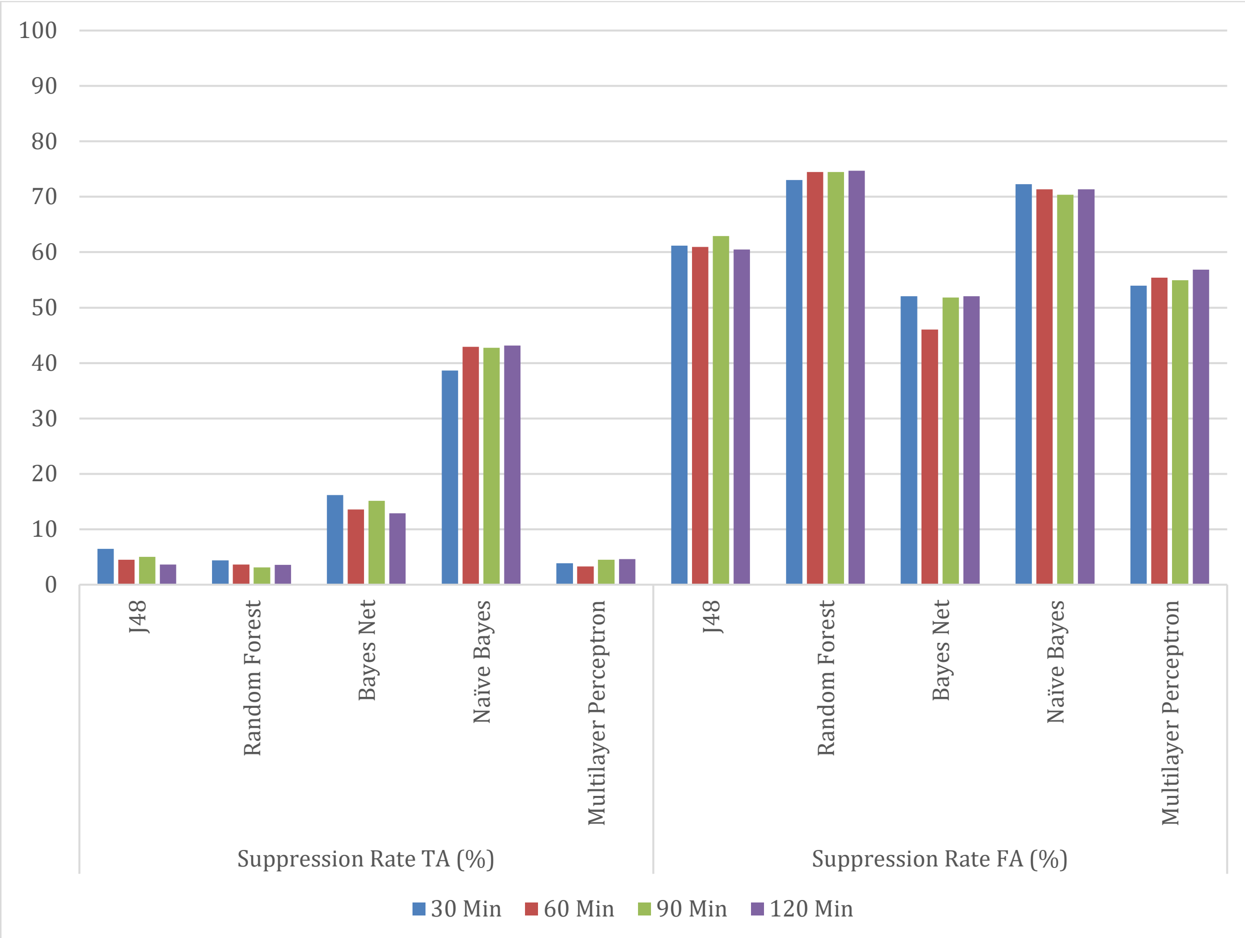

Figure 22: True Alarm & False Alarm Suppression Rates with Mean Value in Time Dimension for Tachy

When the mean value was taken in consideration in time domain analysis, we observed that Random Forest outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates. We also observed in Random Forest that true alarm suppression rate was initially decreasing and false alarm suppression rate was initially increasing when time window was increased from 30 minutes to 60 minutes. However, when time window was increased to 90 and 120 minutes, true alarm suppression rate was decreasing and then started increasing, and false alarm suppression rates was almost constant.

Furthermore, in Multilayer Perceptron, when time window was increased from 30 to 60 minutes, the true alarm suppression rate was initially decreasing where as false alarm suppression rate was increasing, but as time window is increased to 90 minutes; the alarm suppression rate was reverse i.e. true alarm suppression rate is increased and false alarm suppression rate was decreased.

Table 29: Comparison of Mean Value in Time Domain for Tachy

	30 Min		60 Min		90 Min		120 Min	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	6.45	61.20	4.51	60.96	5.03	62.89	3.61	60.48
Random Forest	4.38	73.01	3.61	74.46	3.09	74.46	3.55	74.70
BayesNet	16.18	52.05	13.60	46.02	15.15	51.81	12.89	52.05
NaiveBayes	38.62	72.29	42.94	71.33	42.75	70.36	43.13	71.33
Multilayer Perceptron	3.87	53.98	3.29	55.42	4.51	54.94	4.64	56.87

5.2.1.2 Time Domain with Median Value

Table 30: Comparison of Median Value in Time Domain for Tachy

	30 Min		60 Min		90 Min		120 Min	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	4.64	58.07	4.38	61.20	4.51	62.89	4.19	63.61
Random Forest	3.55	69.88	3.80	70.84	3.55	72.53	3.35	72.77
BayesNet	11.28	50.60	11.22	47.47	10.51	48.19	9.61	50.60
NaiveBayes	25.40	59.04	11.48	33.01	11.61	32.05	12.19	34.46
Multilayer Perceptron	7.09	50.60	4.32	47.71	5.09	50.60	4.90	55.42

When the median value was taken in consideration in time domain analysis, we observed that Random Forest still outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates. We also observed in Random Forest that false alarm suppression rate has increasing trend when time window was increased from 30 minutes to 120 minutes, and false alarm suppression rate was increasing when time window was increased from 30 minutes to 60 minutes. However, the true alarm suppression rate was constant when time window was increased from 90 to 120 min. Moreover, in Multilayer Perceptron, when time window was increased from 30 to 60 minutes, both true alarm suppression rate and false alarm suppression rate was initially decreasing, but as time window is increased to 90 minutes; both alarm suppression rate was increased.

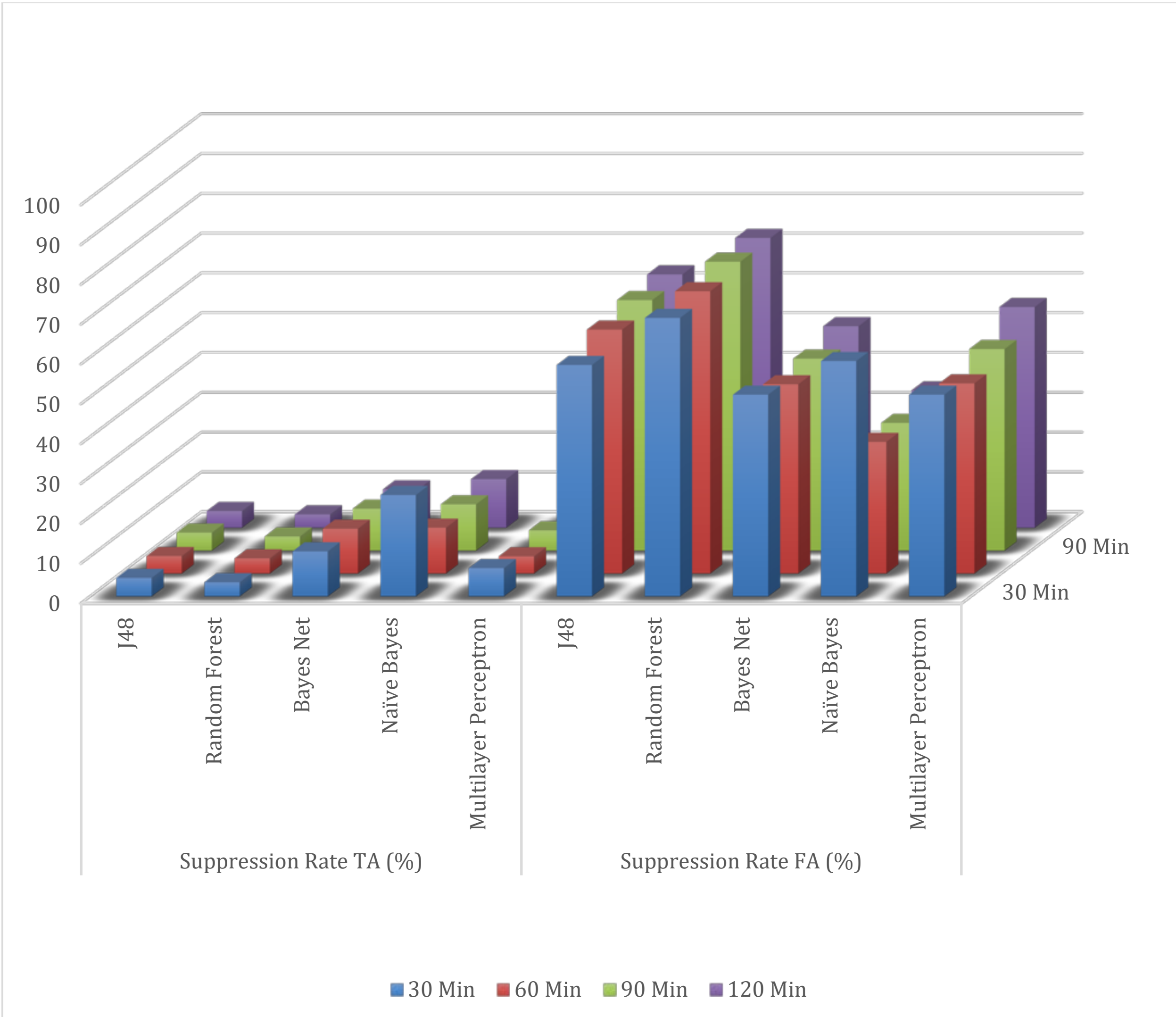

Figure 23: True Alarm and False Alarm Suppression Rates with Median Value in Time Dimension for Tachy

5.2.1.3 Time Domain with Standard Deviation Value

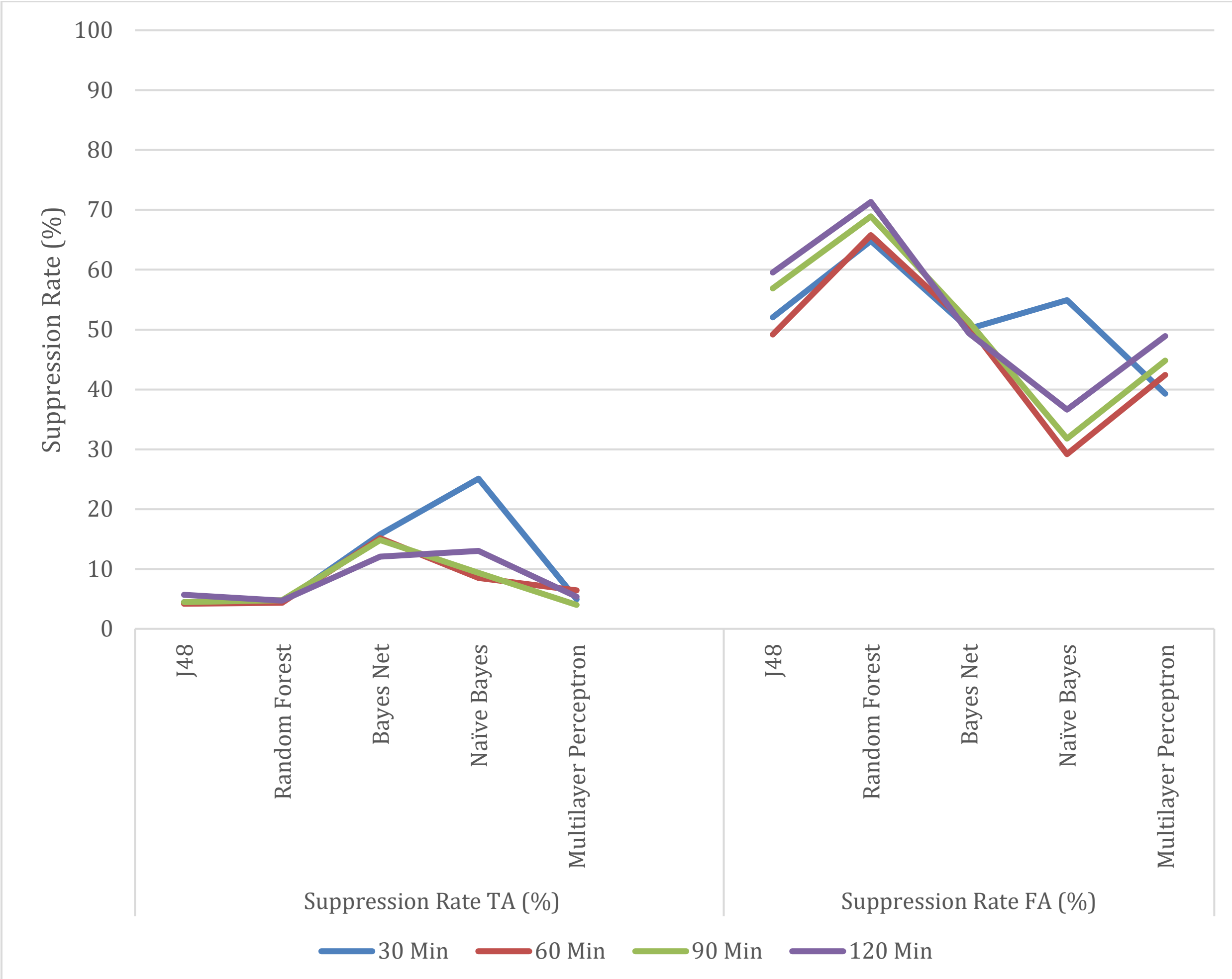

Figure 24: True Alarm & False Alarm Suppression Rates with Standard Deviation Value in Time Dimension for Tachy

In Table 31, the data was transformed through standard deviation with varying time window; we observed that Random Forest performed best among other classification algorithms. We also observed in Random Forest that when time window was increased from 30 to 60 minutes, the true alarm suppression rate was constant and the false alarm suppression rate was increasing, but when time window was increased to 90 min both alarm suppression rate was increasing, and when time window increased to 120 min false alarm suppression rate starts decreasing, and true alarm suppression rates starts increasing.

Table 31: Comparison of Standard Deviation Value in Time Domain for Tachy

	30 Min		60 Min		90 Min		120 Min	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	4.45	52.05	4.19	49.16	4.45	56.87	5.67	59.52
Random Forest	4.38	64.82	4.38	65.78	4.77	68.92	4.71	71.33
BayesNet	15.80	50.12	15.15	50.60	14.83	51.33	12.06	49.40
NaiveBayes	25.08	54.94	8.51	29.16	9.35	31.81	13.02	36.63
Multilayer Perceptron	4.96	39.28	6.45	42.41	4.00	44.82	5.35	48.92

Furthermore, in Multilayer Perceptron, when time window was increased from 30 to 60 minutes, true alarm suppression rate was initially increasing, but as time window is increased to 90 minutes; the true alarm suppression rate was decreased, and when increased in time window to 120 min, true alarm rate starts increasing, but false alarm suppression rate was increasing and increasing when time window was increased from 30 to 120 minutes.

5.2.1.4 Time Domain with DFT Value

In Table 32, the data was transformed through DFT with varying time window; we observed that Random Forest performed best among other classifiers. We also observed in Random Forest that when time window was increased from 30 to 120 minutes, the true alarm as well as false alarm suppression rate was initially decreasing and then increasing.

However, in J48, when time window was increased from 30 to 60 minutes, true alarm suppression rate was initially decreasing, whereas false alarm rate was increasing, but as time window is increased to 90 minutes, true alarm suppression rate was increased, and false alarm suppression rate was almost constant. Again increased in time window to 120 min, true alarm suppression rate was still increasing and false alarm suppression rate was also increasing.

Table 32: Comparison of DFT Value in Time Domain for Tachy

	30 Min		60 Min		90 Min		120 Min	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	8.45	59.52	7.03	60.72	7.41	60.48	8.12	61.69
Random Forest	5.29	58.31	3.29	60.48	3.35	57.83	4.38	65.06
BayesNet	15.67	54.70	20.18	52.77	17.21	52.77	15.93	52.29
NaiveBayes	61.83	87.71	59.25	82.41	63.31	82.41	62.41	86.27
Multilayer Perceptron	5.42	56.87	7.03	58.31	7.48	63.37	5.80	67.47

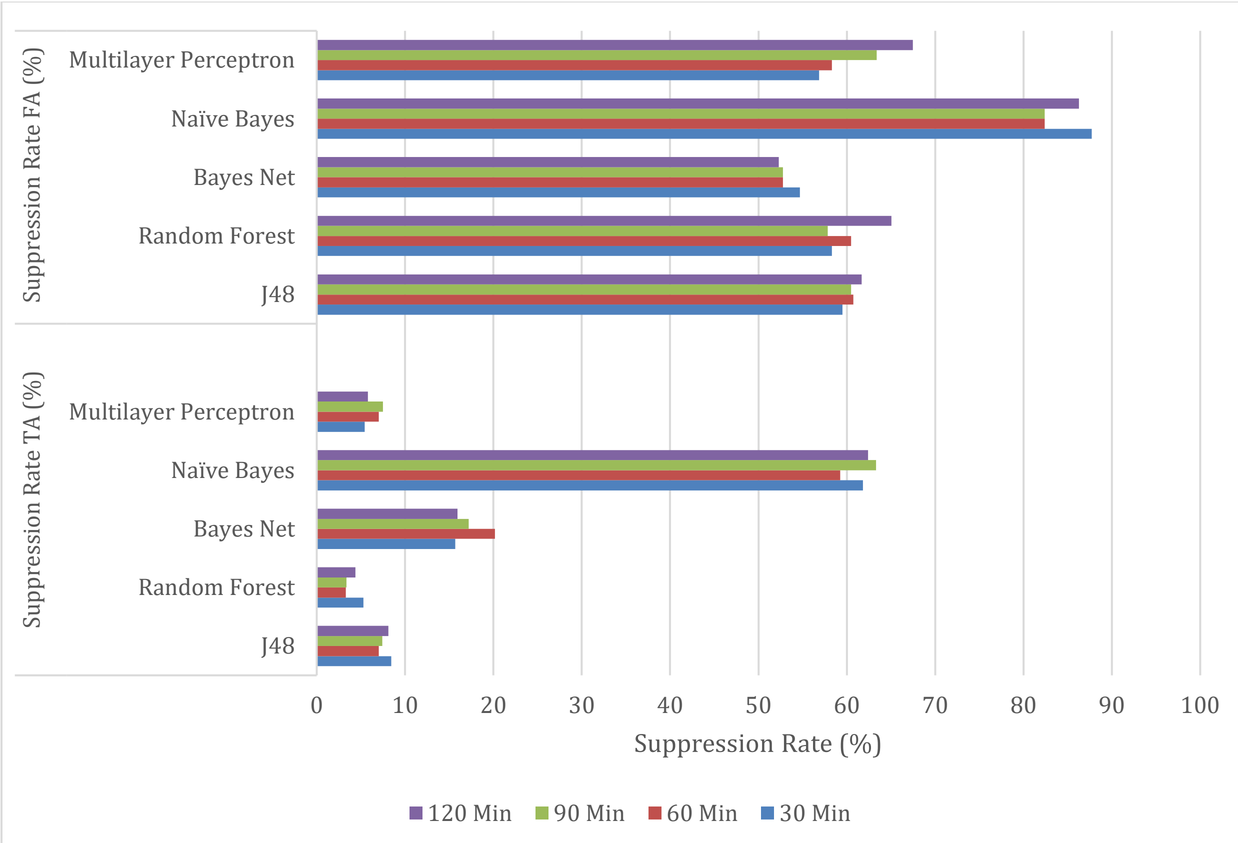

Figure 25: True Alarm & False Alarm Suppression Rates with Discrete Fourier Transform Value in Time Dimension for Tachy

5.2.2 Comparative Analysis with Data Transformation

We transform data through mean, median, standard deviation, and DFT.

5.2.2.1 Data Transformation in 30 Min Time Window

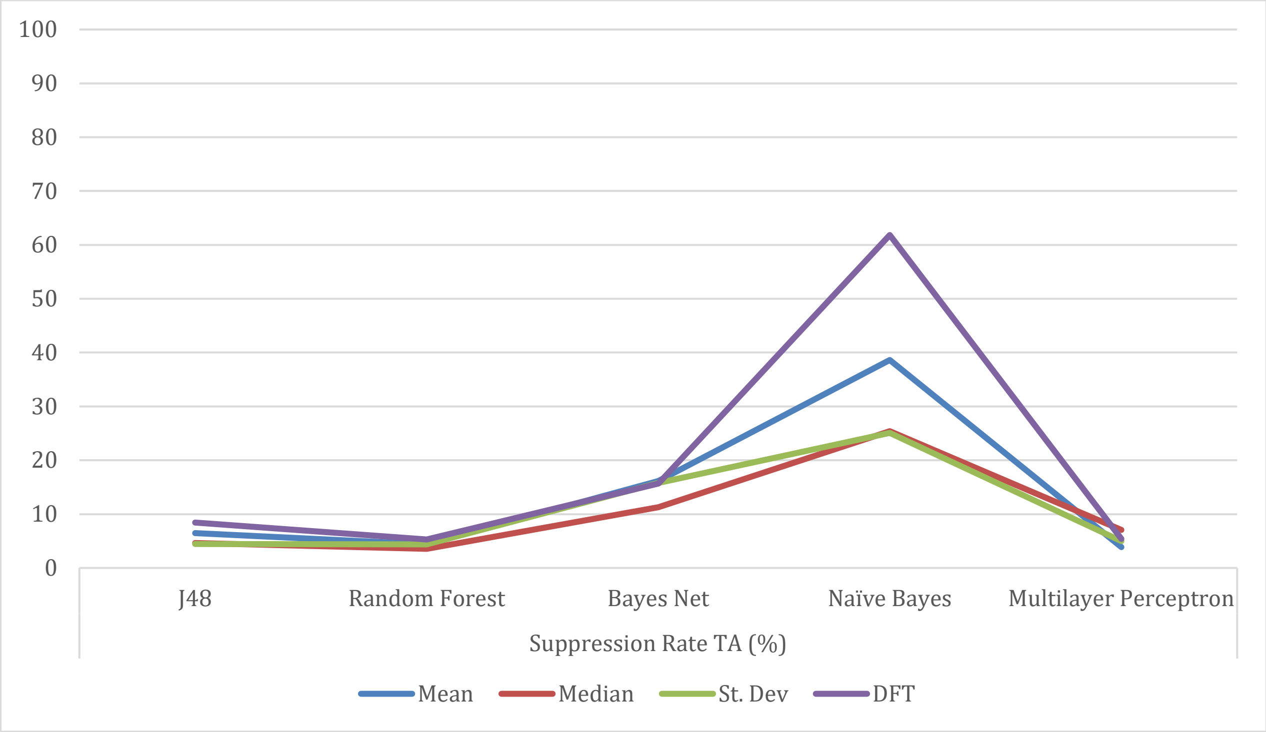

Figure 26: True Alarm & False Alarm Suppression Rates with Data Transformation in 30 Minutes Time Window for Tachy

When 30-minute time window was taken in consideration with various data transformation, we observed that Random Forest outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates. We also observed in Random Forest that true alarm and false alarm suppression rate was initially decreasing when data transformation was altered from mean to median. However, when data transformation was changed to standard deviation, true alarm suppression rates started increasing, and false alarm suppressing rate started decreasing. Again data transformation technique was altered to DFT, the true alarm suppression rates was increasing and false alarm was decreasing.

Table 33: Comparison of Data Transformation in 30 Minutes Time Window for Tachy

	Mean	**Median**	**Std. Deviation**	**DFT**

Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	6.45	61.20	4.64	58.07	4.45	52.05	8.45	59.52
Random Forest	4.38	73.01	3.55	69.88	4.38	64.82	5.29	58.31
BayesNet	16.18	52.05	11.28	50.60	15.80	50.12	15.67	54.70
NaiveBayes	38.62	72.29	25.40	59.04	25.08	54.94	61.83	87.71
Multilayer Perceptron	3.87	53.98	7.09	50.60	4.96	39.28	5.42	56.87

5.2.2.2 Data Transformation in 60 Min Time Window

Considering 60-minutes time window with various data transformation technique, we observed that Random Forest outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates. We also observed from Table 34, Random Forest with mean value performed the best with low true suppression rate of 3.61% and false alarm suppression rate 0f 74.76%.

Table 34: Comparison of Data Transformation in 60 Minutes Time Window for Tachy

	Mean		Median		Std. Deviation		DFT	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	4.51	60.96	4.38	61.20	4.19	49.16	7.03	60.72
Random Forest	3.61	74.46	3.80	70.84	4.38	65.78	3.29	60.48
BayesNet	13.60	46.02	11.22	47.47	15.15	50.60	20.18	52.77
NaiveBayes	42.94	71.33	11.48	33.01	8.51	29.16	59.25	82.41
Multilayer Perceptron	3.29	55.42	4.32	47.71	6.45	42.41	7.03	58.31

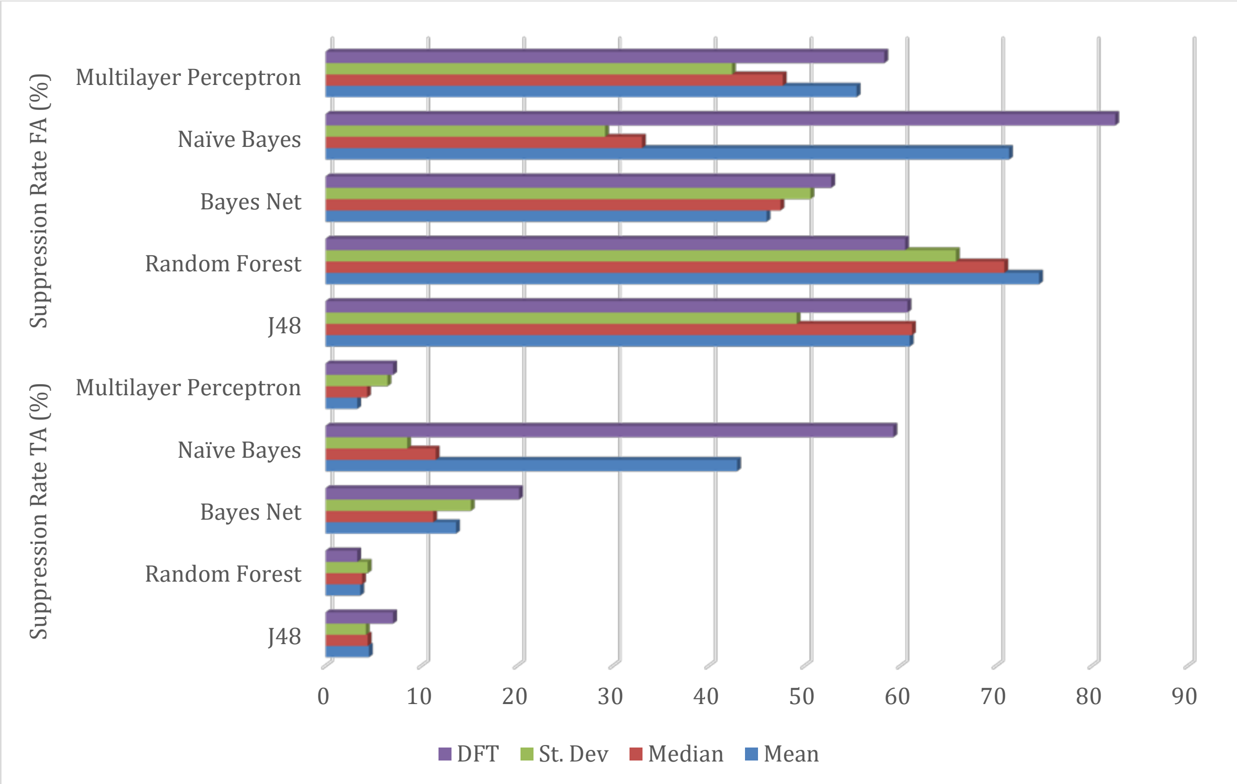

Figure 27: False Alarm & True Alarm Suppression Rates with Data Transformation in 60 Minutes Time Window for Tachy

5.2.2.3 Data Transformation in 90 Min Time Window

Table 35: Comparison of Data Transformation in 90 Minutes Time Window for Tachy

	Mean		**Median**		**Std. Deviation**		**DFT**	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	5.03	62.89	4.51	62.89	4.45	56.87	7.41	60.48
Random Forest	3.09	74.46	3.55	72.53	4.77	68.92	3.35	57.83
BayesNet	15.15	51.81	10.51	48.19	14.83	51.33	17.21	52.77
NaiveBayes	42.75	70.36	11.61	32.05	9.35	31.81	63.31	82.41
Multilayer Perceptron	4.51	54.94	5.09	50.60	4.00	44.82	7.48	63.37

When 90-minutes time window was taken in consideration with various data transformation, we observed that Random Forest outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates. Moreover, Random Forest with mean value has lowest true alarm suppression rate of 3.09% and high false alarm suppression rate of 74.46%. Furthermore, BayesNet had high true alarm suppression rates and low false alarm suppression rates.

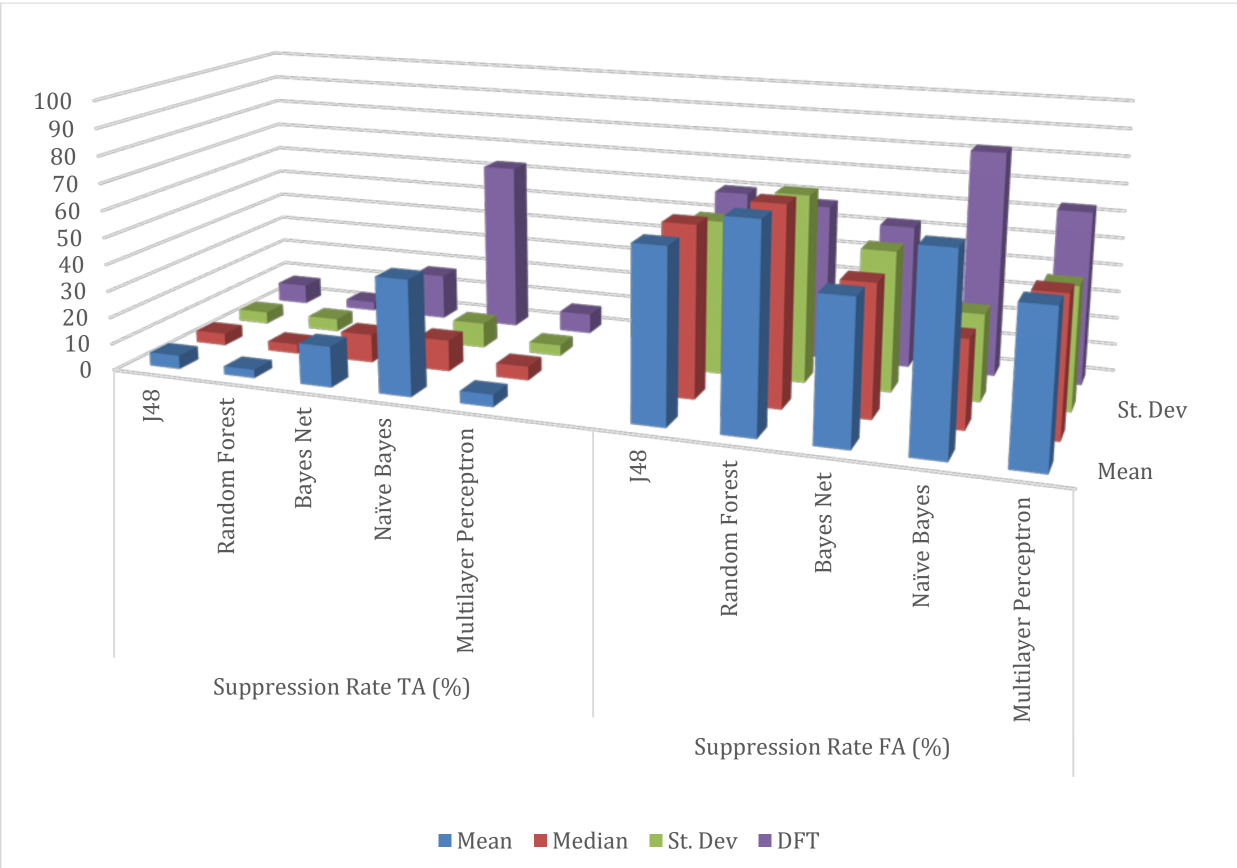

Figure 28: False Alarm & True Alarm Suppression Rates with Data Transformation in 90 Minutes Time Window for Tachy

5.2.2.4 Data Transformation in 120 Min Time Window

Considering 120-minutes time window with various data transformation technique, we observed that Random Forest still outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates.

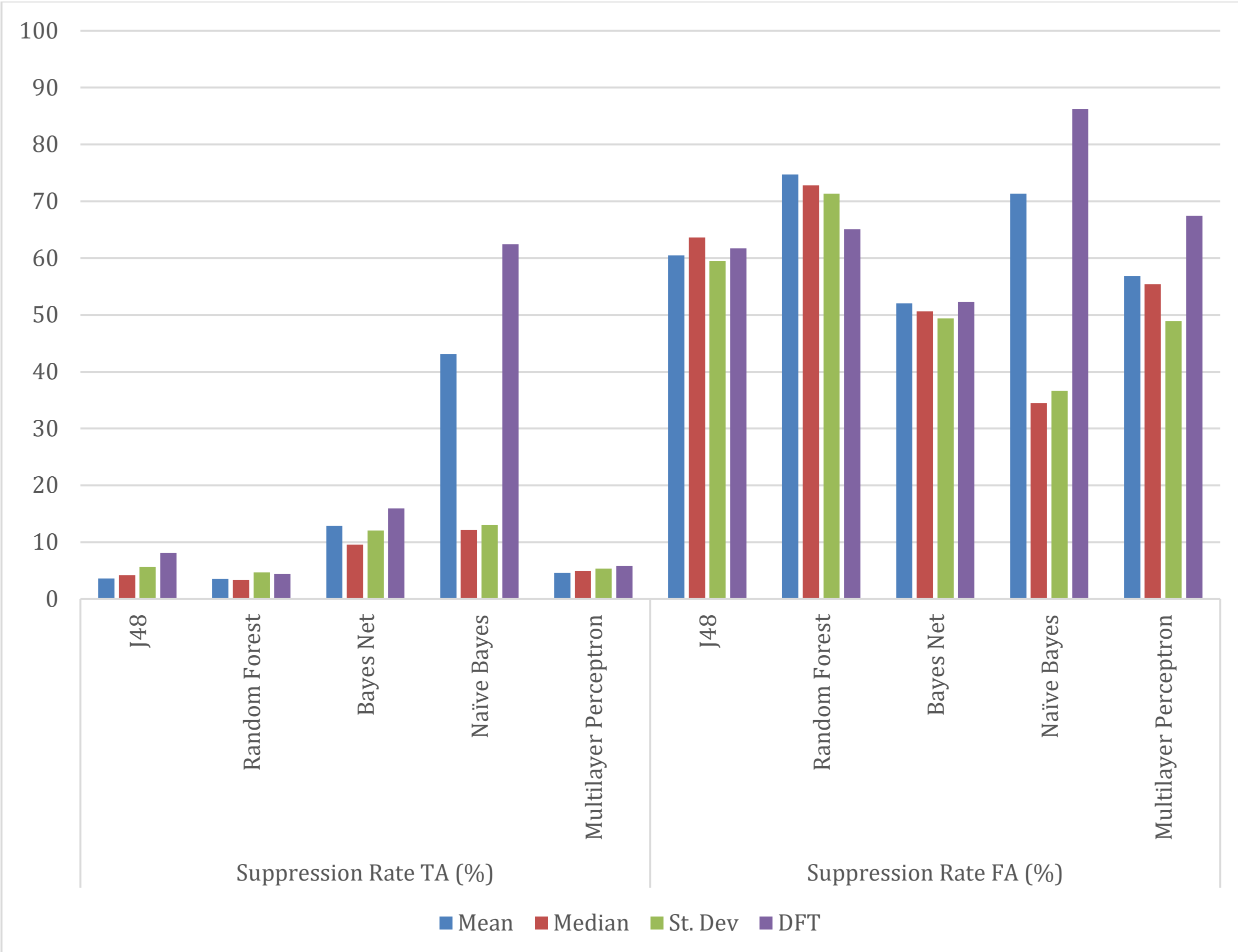

Figure 29: False Alarm & True Alarm Suppression Rates with Data Transformation in 90 Minutes Time Window for Tachy

Table 36: Comparison of Data Transformation in 120 Minutes Time Window for Tachy

	Mean		**Median**		**Std. Deviation**		**DFT**	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	3.61	60.48	4.19	63.61	5.67	59.52	8.12	61.69
Random Forest	3.55	74.70	3.35	72.77	4.71	71.33	4.38	65.06
BayesNet	12.89	52.05	9.61	50.60	12.06	49.40	15.93	52.29
NaiveBayes	43.13	71.33	12.19	34.46	13.02	36.63	62.41	86.27

Multilayer Perceptron	4.64	56.87	4.90	55.42	5.35	48.92	5.80	67.47

5.2.3 Comparative Analysis with Feature Sets

We use different feature sets such as CFS Subset Evaluator, Wrapper Subset Evaluator (using Bayes Net, NaiveBayes, J48, Random Forest), and Info Gain Evaluator.

5.2.3.1 Feature Sets with Mean Value

5.2.3.1.1 Feature Sets with Mean Value in 30 Minutes Window

Considering 30-minutes time window with mean value, we observed that feature selection obtained from Wrapper method including J48 performed best with Random Forest as classifier in comparison to the feature sets obtained from CFS, Wrapper method including NaiveBayes, Wrapper method including Random Forest, and Information gain. Wrapper method including J48 resulted in low true alarm suppression rate of 4.32% and high false alarm suppression rate of 71.33%.

Table 37: Comparison of Feature Sets with Mean Value in 30 Minutes Time Window for Tachy

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	5.54	57.11	3.93	29.40	4.71	61.20	5.48	59.28
Random Forest	5.22	68.19	8.90	51.33	4.32	71.33	4.77	69.64
BayesNet	10.70	50.12	7.80	36.87	14.83	52.05	15.22	50.60
NaiveBayes	8.58	24.82	1.03	15.42	24.11	60.72	23.40	61.93
Multilayer Perceptron	4.32	41.69	3.80	28.92	4.64	52.53	5.54	55.42

5.2.3.1.2 Feature Sets with Mean Value in 60 Minutes Window

Table 38: Comparison of Feature Sets with Mean Value in 60 Minutes Time Window for Tachy

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	3.87	50.84	2.58	41.93	3.74	57.35	4.00	58.07
Random Forest	4.58	69.16	5.80	61.93	3.68	73.01	3.87	73.98
BayesNet	5.61	39.76	7.93	32.77	9.80	45.78	12.31	45.06
NaiveBayes	8.51	25.78	1.42	19.28	9.48	27.95	38.36	69.64
Multilayer Perceptron	5.09	36.63	2.71	30.60	3.09	48.43	4.38	56.63

5.2.3.1.3 Feature Sets with Mean Value in 90 Minutes Window

Table 39: Comparison of Feature Sets with Mean Value in 90 Minutes Time Window for Tachy

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	4.90	47.47	1.87	20.24	3.93	55.90	1.48	19.04
Random Forest	4.71	68.43	16.44	38.80	5.16	72.05	2.90	29.88
BayesNet	5.16	40.48	1.55	17.35	8.06	44.34	1.48	18.07
NaiveBayes	6.58	26.99	1.81	18.55	7.16	27.23	1.87	18.31

Multilayer Perceptron	3.61	33.25	2.13	22.41	3.03	29.40	1.61	17.83

5.2.3.1.4 Feature Sets with Mean Value in 120 Minutes Window

Table 40: Comparison of Feature Sets with Mean Value in 120 Minutes Time Window for Tachy

	CFS		Wrapper incl. Naïve Bayes		Wrapper incl. J48		Wrapper incl. Random Forest	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	3.68	49.88	2.45	36.87	3.29	63.37	3.35	63.86
Random Forest	4.90	73.49	7.29	63.37	3.42	73.98	3.74	74.70
BayesNet	6.58	44.58	4.64	25.78	11.35	52.53	9.09	51.81
NaiveBayes	7.74	29.16	2.32	21.20	34.95	68.43	20.37	51.33
Multilayer Perceptron	3.93	32.77	1.87	24.58	3.74	53.73	3.22	49.64

Taking consideration of 60-minutes, 90-minutes, and 120-minutes time window with mean value, we observed that feature selection obtained from Wrapper method J48 method with Random Forest as classifier performed best with low true alarm suppression rate in 60 minutes of data and CFS method with Random Forest as classifier performed well in 90 minutes of data. In 120 minutes of time window, Wrapper method including J48 performed the best in comparison to the feature sets obtained from CFS, Wrapper method including NaiveBayes, Wrapper method including Random Forest, and Information gain.

5.2.3.2 Feature Sets with Median Value

5.2.3.2.1 Feature Sets with Median Value in 30 Minutes Window

Table 41: Comparison of Feature Sets with Median Value in 30 Minutes Time Window for Tachy

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	3.87	54.70	3.87	20.96	3.93	51.57	4.13	55.18
Random Forest	4.45	69.40	13.15	49.64	4.51	66.02	4.32	71.08
BayesNet	7.29	45.54	6.13	27.47	6.64	41.93	11.15	49.64
NaiveBayes	8.06	13.25	0.97	6.27	8.45	12.77	22.31	53.73
Multilayer Perceptron	3.29	32.77	2.77	21.45	3.09	31.33	4.13	46.27

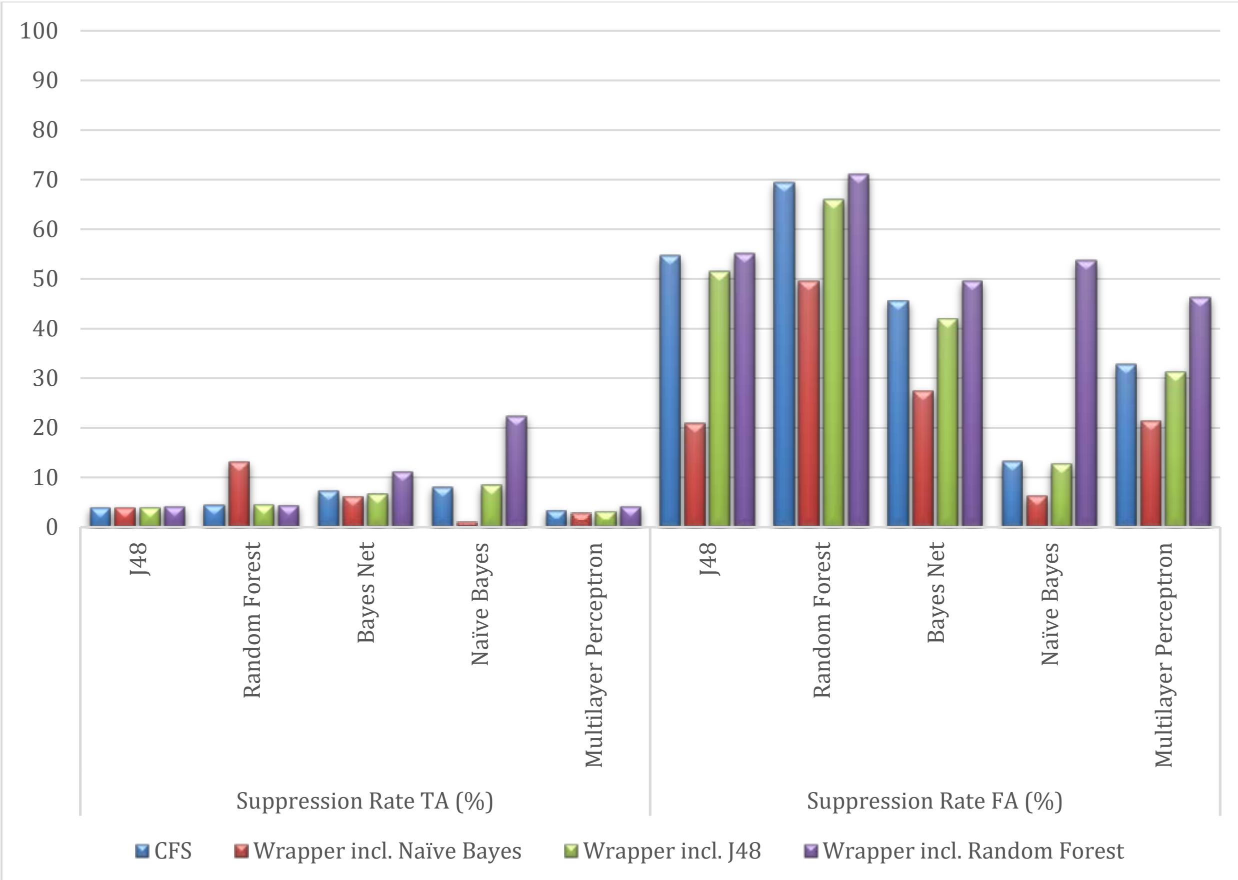

Figure 30: Comparison of Feature Sets with Median Value in 30 Minutes Time Window for Tachy True & False Alarm Suppression Rates

5.2.3.2.2 Feature Sets with Median Value in 60 Minutes Window

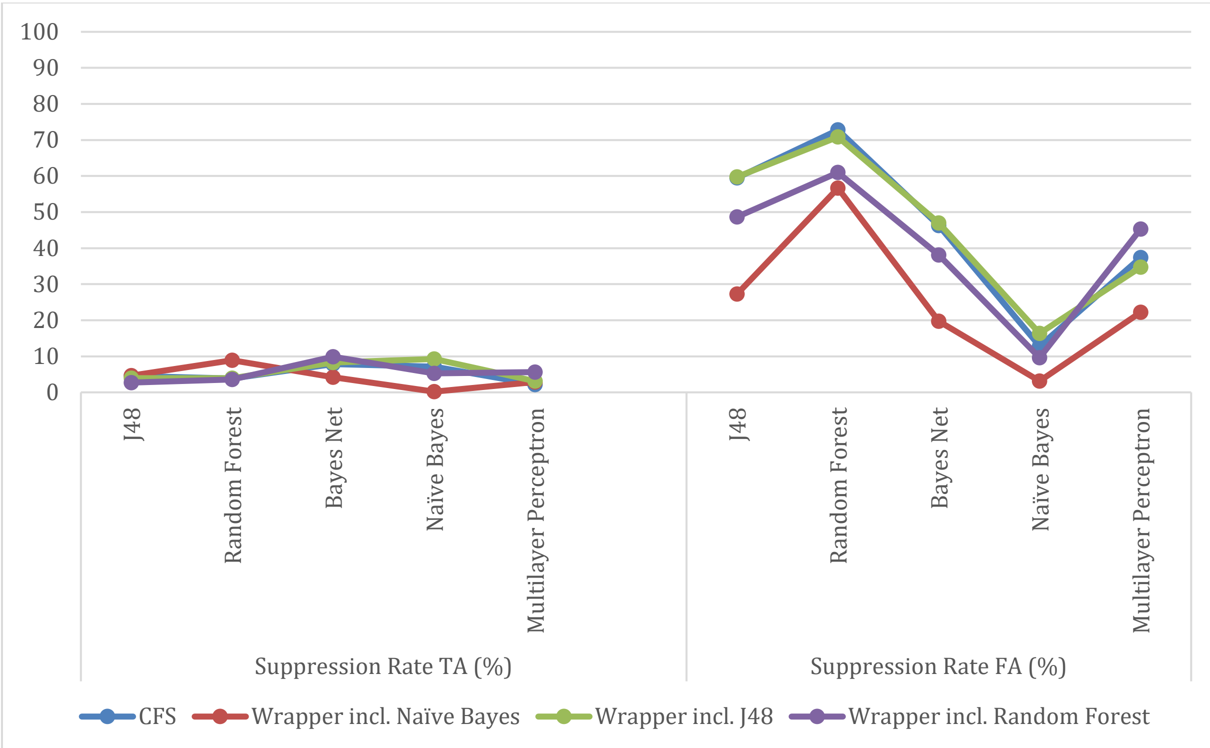

Figure 31: Comparison of Feature Sets with Median Value in 60 Minutes Time Window for Tachy True & False Alarm Suppression Rates

Table 42: Comparison of Feature Sets with Median Value in 60 Minutes Time Window for Tachy

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	4.58	59.52	4.64	27.23	3.93	59.76	2.71	48.67
Random Forest	3.87	72.77	8.90	56.63	3.93	70.84	3.55	60.96
BayesNet	7.80	46.27	4.19	19.76	8.32	46.99	9.86	38.07
NaiveBayes	7.09	12.77	0.19	3.13	9.22	16.39	5.29	9.64
Multilayer Perceptron	2.19	37.35	2.90	22.17	3.09	34.70	5.61	45.30

5.2.3.2.3 Feature Sets with Median Value in 90 Minutes Window

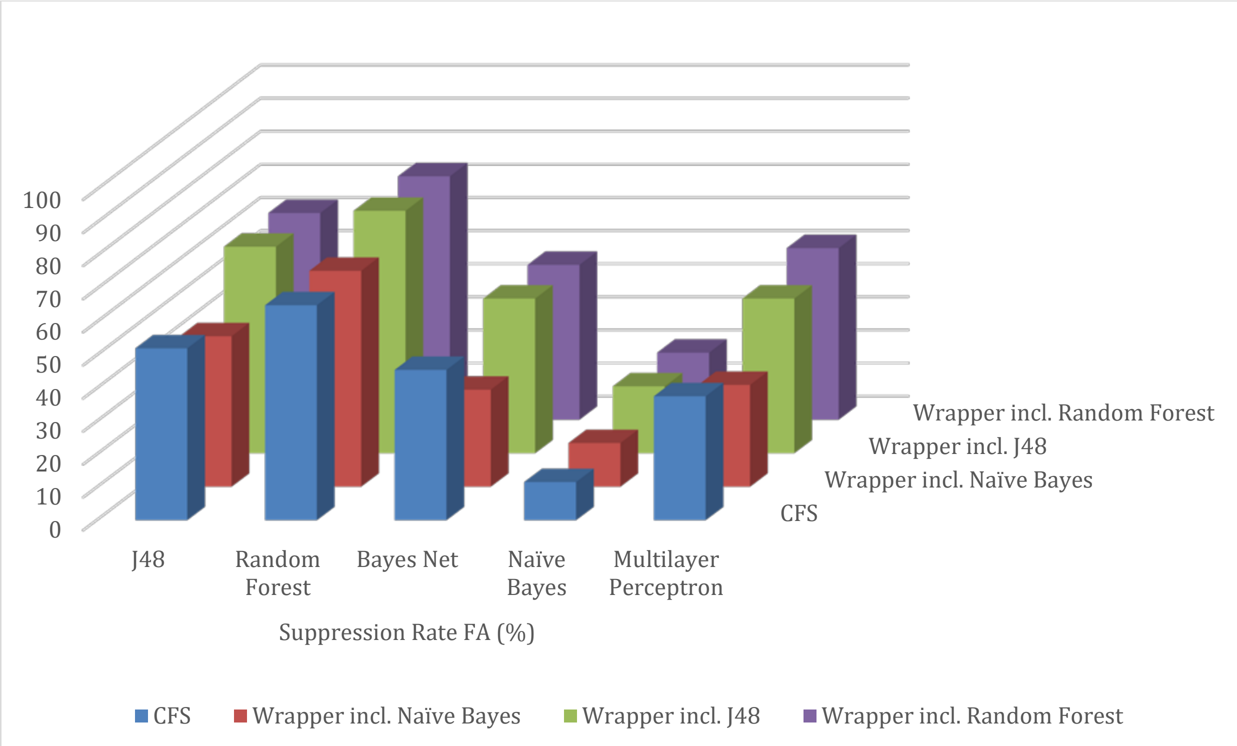

Figure 32: Comparison of Feature Sets with Median Value in 90 Minutes Time Window for Tachy False Alarm Suppression Rates

Table 43: Comparison of Feature Sets with Median Value in 90 Minutes Time Window for Tachy

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	2.64	52.05	3.09	45.54	4.58	62.41	4.58	62.41
Random Forest	4.58	65.06	7.16	65.30	3.29	73.25	3.16	73.49
BayesNet	5.42	45.54	6.71	29.40	9.35	46.75	9.35	46.75
NaiveBayes	6.45	11.57	1.93	13.25	8.96	20.24	8.96	20.24
Multilayer Perceptron	4.64	37.59	2.71	30.84	3.61	46.75	3.61	51.81

5.2.3.2.4 Feature Sets with Median Value in 120 Minutes Window

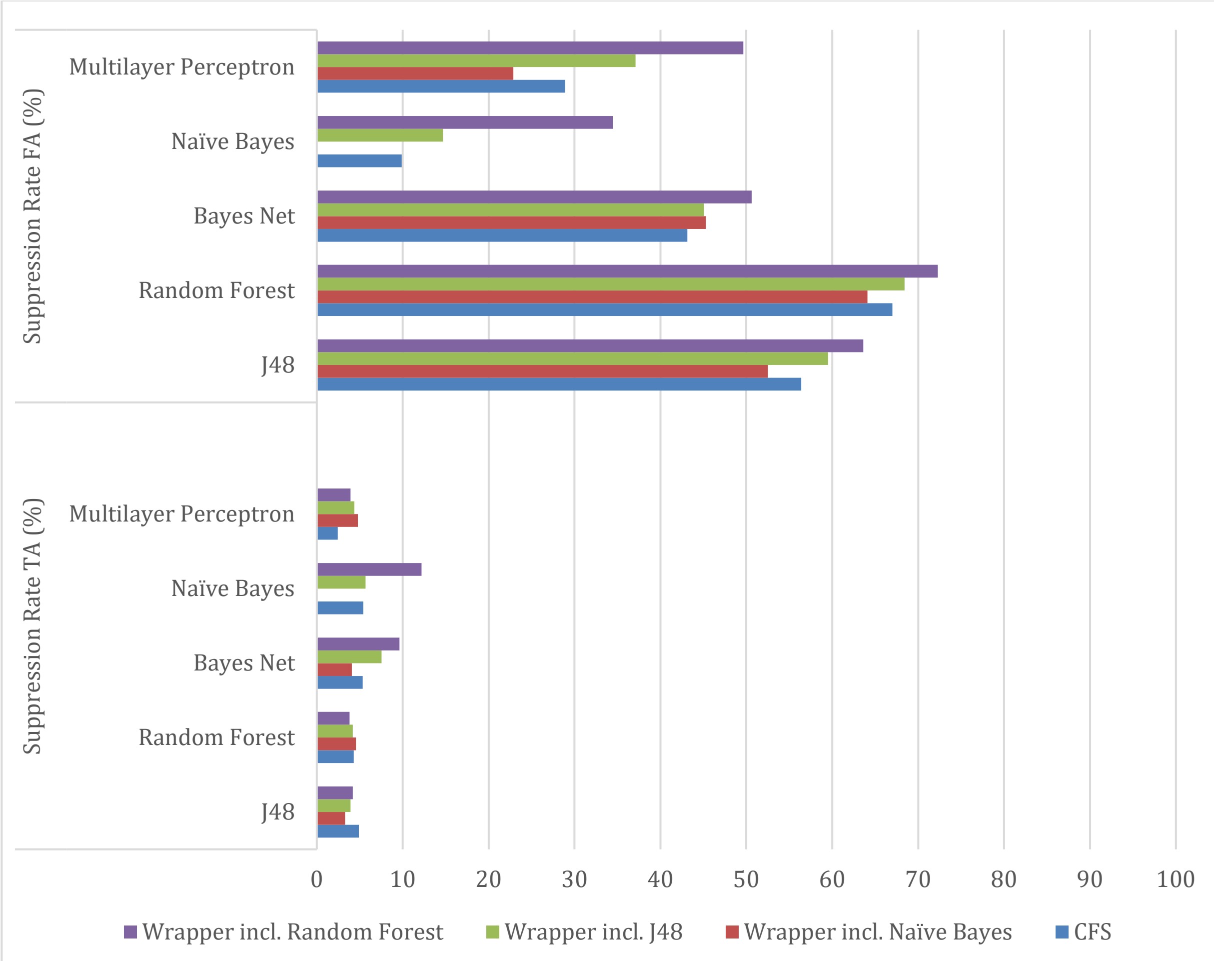

Figure 33: Comparison of Feature Sets with Median Value in 120 Minutes Time Window for Tachy True & False Alarm Suppression Rates

Considering 30-minutes, 60-minutes, 90-minutes, and 120-minutes time window with median value, we observed that feature selection obtained from CFS method with Random Forest as classifier performed best in 30 minutes time window with true alarm suppression rate of 4.45%, and false alarm suppression rate of 69.40% whereas Wrapper including J48 method with Random Forest as classifier performed well in 60 minutes time window. Wrapper including Random Forest method with Random Forest as classifier performed best in 90 and 120 minutes time window.

Table 44: Comparison of Feature Sets with Median Value in 120 Minutes Time Window for Tachy

	CFS	Wrapper incl. BayesNet	Wrapper incl. J48	Wrapper incl. Random Forest

Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	4.90	56.39	3.29	52.53	3.93	59.52	4.19	63.61
Random Forest	4.32	66.99	4.58	64.10	4.19	68.43	3.80	72.29
BayesNet	5.35	43.13	4.06	45.30	7.54	45.06	9.61	50.60
NaiveBayes	5.42	9.88	0.00	0.00	5.67	14.70	12.19	34.46
Multilayer Perceptron	2.45	28.92	4.77	22.89	4.38	37.11	3.93	49.64

5.2.3.3 Feature Sets with Standard Deviation Value

5.2.3.3.1 Feature Sets with Standard Deviation Value in 30 Minutes Window

Table 45: Comparison of Feature Sets with Standard Deviation Value in 30 Minutes Time Window for Tachy

	CFS		Wrapper incl. Naïve Bayes		Wrapper incl. J48		Wrapper incl. Random Forest	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	2.39	46.27	3.16	40.48	3.55	53.85	0.77	8.43
Random Forest	4.84	62.41	6.77	60.96	4.77	62.89	1.61	16.87
BayesNet	6.25	36.63	5.67	41.20	10.12	44.34	0.71	8.67
NaiveBayes	4.32	21.45	2.32	22.65	4.06	23.13	0.71	8.92
Multilayer Perceptron	3.48	22.17	5.42	30.84	5.03	39.04	0.84	9.16

5.2.3.3.2 Analysis of Feature Sets with Standard Deviation Value in 60 Min Window

Table 46: Comparison of Feature Sets with Standard Deviation Value in 60 Minutes Time Window for Tachy

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	2.45	47.95	1.42	25.54	2.13	45.78	1.16	16.39
Random Forest	4.19	61.20	6.25	45.06	5.54	60.96	2.00	25.54
BayesNet	6.77	41.45	1.74	22.65	8.64	43.13	0.90	14.70
NaiveBayes	5.35	24.58	2.71	24.58	5.48	24.58	1.35	14.94
Multilayer Perceptron	3.03	25.06	1.48	24.10	1.68	21.45	1.55	15.66

5.2.3.3.3 Feature Sets with Standard Deviation Value in 90 Minutes Window

Table 47: Comparison of Feature Sets with Standard Deviation Value in 90 Minutes Time Window for Tachy

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	3.16	44.34	1.29	25.06	2.71	52.77	1.29	21.20
Random Forest	4.58	63.61	14.96	45.54	4.26	66.27	2.51	31.33
BayesNet	7.29	42.89	1.93	26.75	6.06	42.89	1.35	20.24
NaiveBayes	5.54	27.23	2.06	26.51	3.35	27.47	1.87	18.31
Multilayer Perceptron	3.87	29.40	1.81	25.54	3.74	37.35	1.61	17.59

5.2.3.3.4 Feature Sets with Standard Deviation Value in 120 Minutes Window

Table 48: Comparison of Feature Sets with Standard Deviation Value in 120 Minutes Time Window for Tachy

	CFS		Wrapper incl. Naïve Bayes		Wrapper incl. J48		Wrapper incl. Random Forest	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	3.93	50.60	2.97	41.45	3.93	52.77	1.81	26.75
Random Forest	5.03	62.89	5.87	58.80	5.03	66.51	2.39	33.01
BayesNet	4.32	39.52	5.48	35.66	9.93	43.13	1.03	24.34
NaiveBayes	3.80	27.23	2.39	28.43	5.74	29.16	2.32	20.48
Multilayer Perceptron	4.32	32.53	2.51	29.40	4.84	47.71	1.81	19.76

Among 30-minutes, 60-minutes, 90-minutes, and 120-minutes time window with standard deviation value, we observed that feature selection obtained from Wrapper including J48 method with Random Forest as classifier performed best in 90 minutes time window with true alarm suppression rate of 4.26%, and false alarm suppression rate of 66.27%.

5.2.3.4 Feature Sets with DFT Value

5.2.3.4.1 Feature Sets with DFT Value in 30 Minutes Window

Among 30-minutes, 60-minutes, 90-minutes, and 120-minutes time window with DFT, we observed that feature selection obtained from Wrapper including Random Forest method with Random Forest as classifier performed best in 120 minutes time window with true alarm suppression rate of 3.35%, and false alarm suppression rate of 70.12%.

Table 49: Comparison of Feature Sets with DFT Value in 30 Minutes Time Window for Tachy

	CFS	Wrapper incl. Naïve Bayes	Wrapper incl. J48	Wrapper incl. Random Forest

Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	5.67	56.14	3.03	24.58	5.67	54.46	3.16	31.81
Random Forest	4.45	68.19	9.35	49.40	5.35	66.02	5.54	50.60
BayesNet	8.70	52.53	4.38	27.23	11.54	50.84	13.15	42.41
NaiveBayes	12.51	41.20	3.48	26.75	34.69	68.92	75.24	93.01
Multilayer Perceptron	5.87	45.54	3.80	30.60	6.25	45.06	4.58	21.93

5.2.3.4.2 Feature Sets with DFT Value in 60 Minutes Window

Table 50: Comparison of Feature Sets with DFT Value in 60 Minutes Time Window for Tachy

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	4.58	54.46	1.10	22.65	4.77	66.02	1.87	36.87
Random Forest	4.64	70.36	14.51	38.07	4.32	72.53	3.93	55.42
BayesNet	8.51	46.27	3.22	24.82	14.44	46.51	12.96	44.10
NaiveBayes	12.31	37.83	3.74	27.23	23.02	56.39	72.02	87.23
Multilayer Perceptron	4.19	43.86	3.09	23.13	4.96	53.98	1.74	23.61

5.2.3.4.3 Feature Sets with DFT Value in 90 Minutes Window

Table 51: Comparison of Feature Sets with DFT Value in 90 Minutes Time Window for Tachy

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	4.96	56.39	0.97	16.39	4.38	54.70	1.10	19.76
Random Forest	5.09	70.12	4.77	21.93	4.13	69.64	1.55	29.64
BayesNet	8.90	46.99	1.81	17.59	9.74	45.06	1.74	20.24
NaiveBayes	13.73	38.31	1.68	18.31	7.99	34.70	72.15	85.30
Multilayer Perceptron	4.51	52.77	1.55	17.83	4.26	43.61	1.42	18.07

5.2.3.4.4 Feature Sets with DFT Value in 120 Minutes Window

Table 52: Comparison of Feature Sets with DFT Value in 120 Minutes Time Window for Tachy

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	5.80	49.16	1.87	23.86	4.71	65.78	5.22	64.34
Random Forest	5.48	67.95	7.48	44.34	3.55	69.64	3.35	70.12
BayesNet	7.67	46.51	5.61	26.51	11.93	46.99	19.60	56.63
NaiveBayes	13.60	45.30	2.71	23.61	57.58	79.76	68.28	90.36
Multilayer Perceptron	5.09	43.86	1.23	21.20	4.38	47.23	5.48	54.94

5.3 Ventricular Tachycardia (Vtach)

5.3.1 Comparative analysis in Time domain

We used 30, 60, and 90 and 120 minutes of time window to investigate the efficacy of classification algorithms to determine under which time domain, the false alarm rates can be minimized, retaining the true alarm suppression rate.

5.3.1.1 Time Domain with Mean Value

Table 53: Comparison of Mean Value in Time Domain for Vtach

	30 Min		60 Min		90 Min		120 Min	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	27.08	74.69	25.82	77.41	26.27	76.99	21.01	75.52
Random Forest	24.64	81.49	22.83	84.31	21.01	84.31	21.29	84.00
BayesNet	32.79	66.95	30.25	62.13	33.06	64.23	30.98	60.56
NaiveBayes	54.53	72.91	26.00	43.31	29.17	48.95	39.04	58.16
Multilayer Perceptron	26.45	74.79	29.08	81.90	25.82	79.60	24.37	76.57

When the mean value was taken in consideration in time domain analysis, we observed that Random Forest outperformed other classifier. We also observed in Random Forest that true alarm suppression rate was decreasing and false alarm suppression rate was increasing when time window was increased from 30 minutes to 120 minutes. 90 minutes of time window with Random Forest performed best with false alarm suppression rate of 84.31% and true alarm suppression rate of 21.01%. Furthermore, in Multilayer Perceptron, when time window was increased from 30 to 120 minutes, the true alarm suppression rate was almost constant through out the time where as false alarm suppression rate was decreasing when time window was changed from 30 minutes to 60 minutes, but as time window is increased to 90 and 120 minutes; the false alarm suppression rate was increasing.

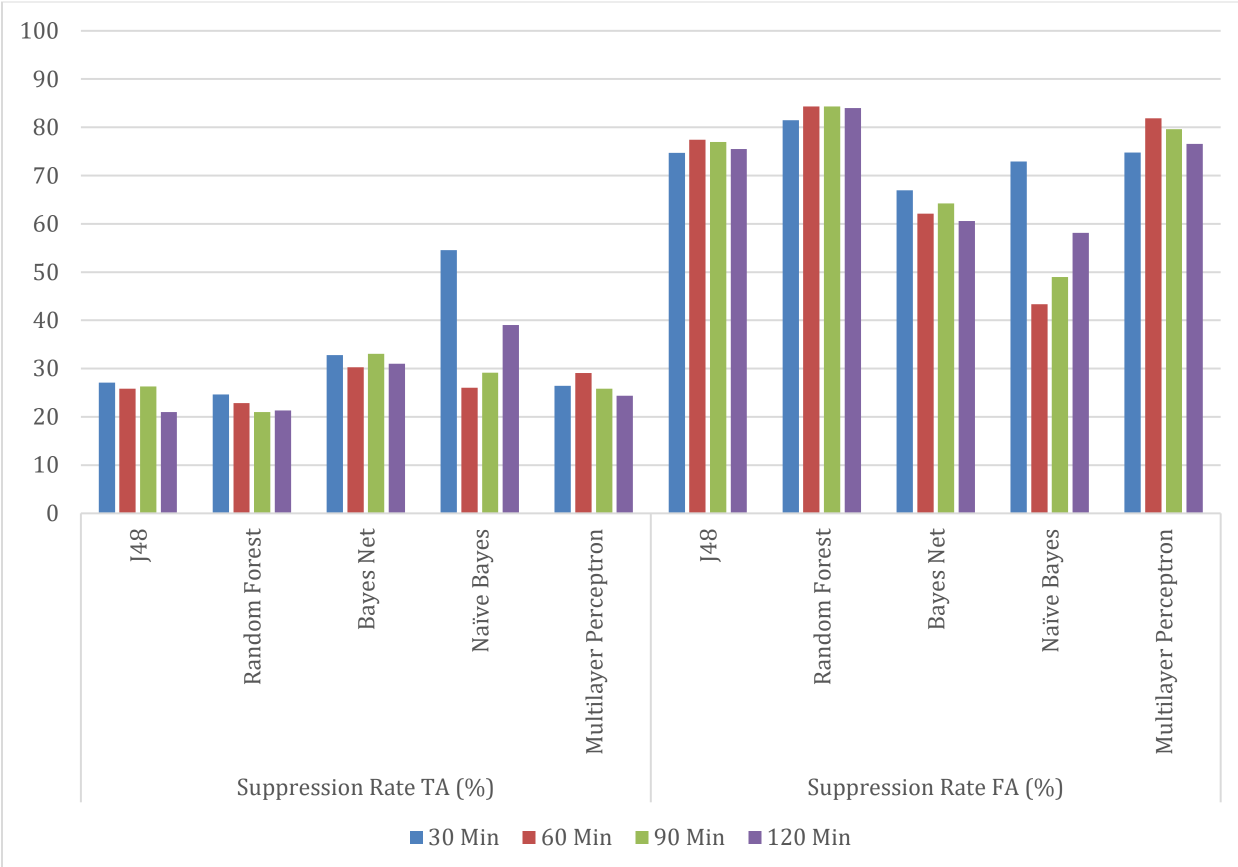

Figure 34: True Alarm & False Alarm Suppression Rates with Mean Value in Time Dimension for Vtach

5.3.1.2 Time Domain with Median Value

Table 54: Comparison of Median Value in Time Domain for Vtach

	30 Min		60 Min		90 Min		120 Min	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	23.46	76.88	26.99	79.81	22.74	79.50	22.46	75.84
Random Forest	22.74	82.53	22.28	82.64	19.93	82.74	20.38	80.96
BayesNet	34.96	69.87	33.79	68.93	34.51	65.48	36.23	64.54
NaiveBayes	32.97	48.85	29.80	45.08	35.42	51.88	34.87	51.67

Multilayer Perceptron	25.82	78.24	26.54	80.13	25.18	78.24	24.73	75.84

When the median data transformation was taken in consideration in time domain analysis, we observed that Random Forest still outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates. We also observed in Random Forest that true alarm suppression rate has decreasing trend when time window was increased from 30 minutes to 90 minutes, and false alarm suppression rate was very much similar. 90 minutes of time window with Random Forest performed best with true alarm suppression rate of 19.93 and false alarm suppression rate of 82.74%. Moreover, in Multilayer Perceptron, when time window was increased from 30 to 60 minutes, both true alarm suppression rate and false alarm suppression rate was initially increasing, but as time window is increased to 90 and 120 minutes; both alarm suppression rate decreased.

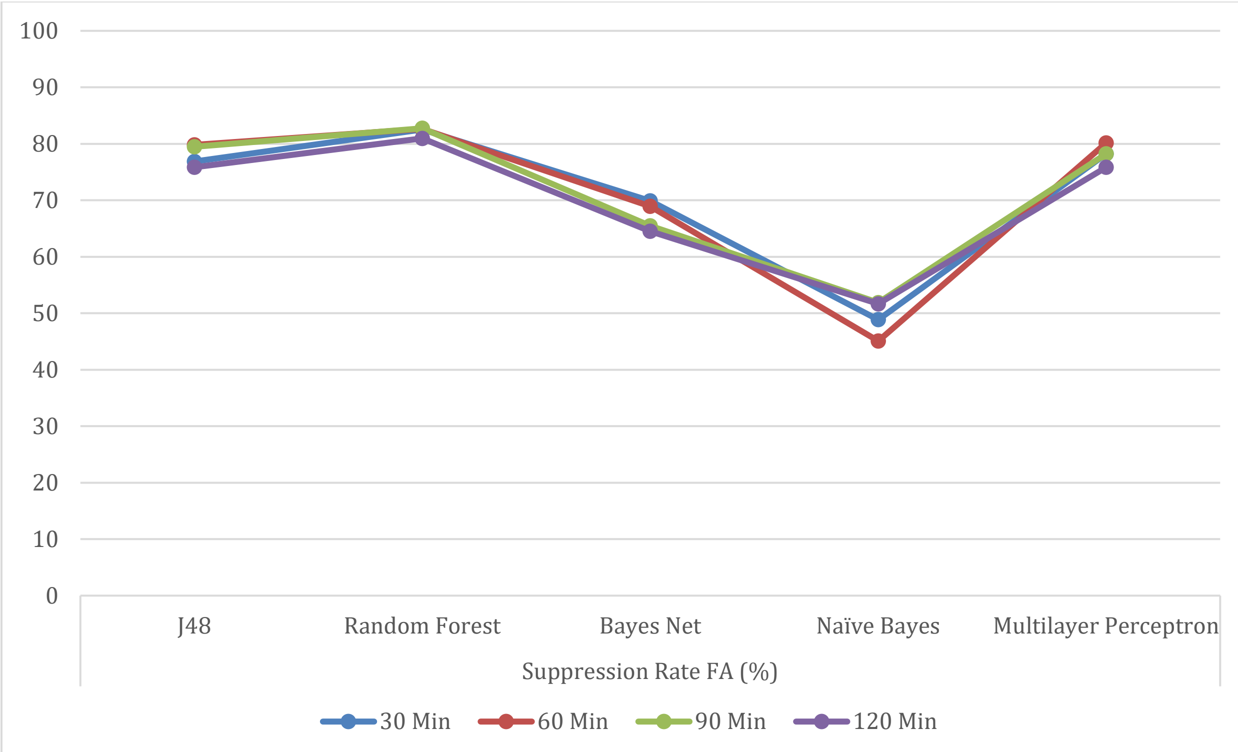

Figure 35: True Alarm & False Alarm Suppression Rates with Median Value in Time Dimension for Vtach

5.3.1.3 Time Domain with Standard Deviation Value

In Table 31, the data was transformed through standard deviation with varying time window; we observed that Random Forest performed best among other classification algorithms. We also observed in Random Forest that when time window was increased from 30 to 60 minutes, the true alarm suppression rate was decreasing and the false alarm suppression rate was increasing, but when time window was increased to 90 and 120 minutes, true alarm suppression rate was almost constant and false alarm suppression rate started increasing.

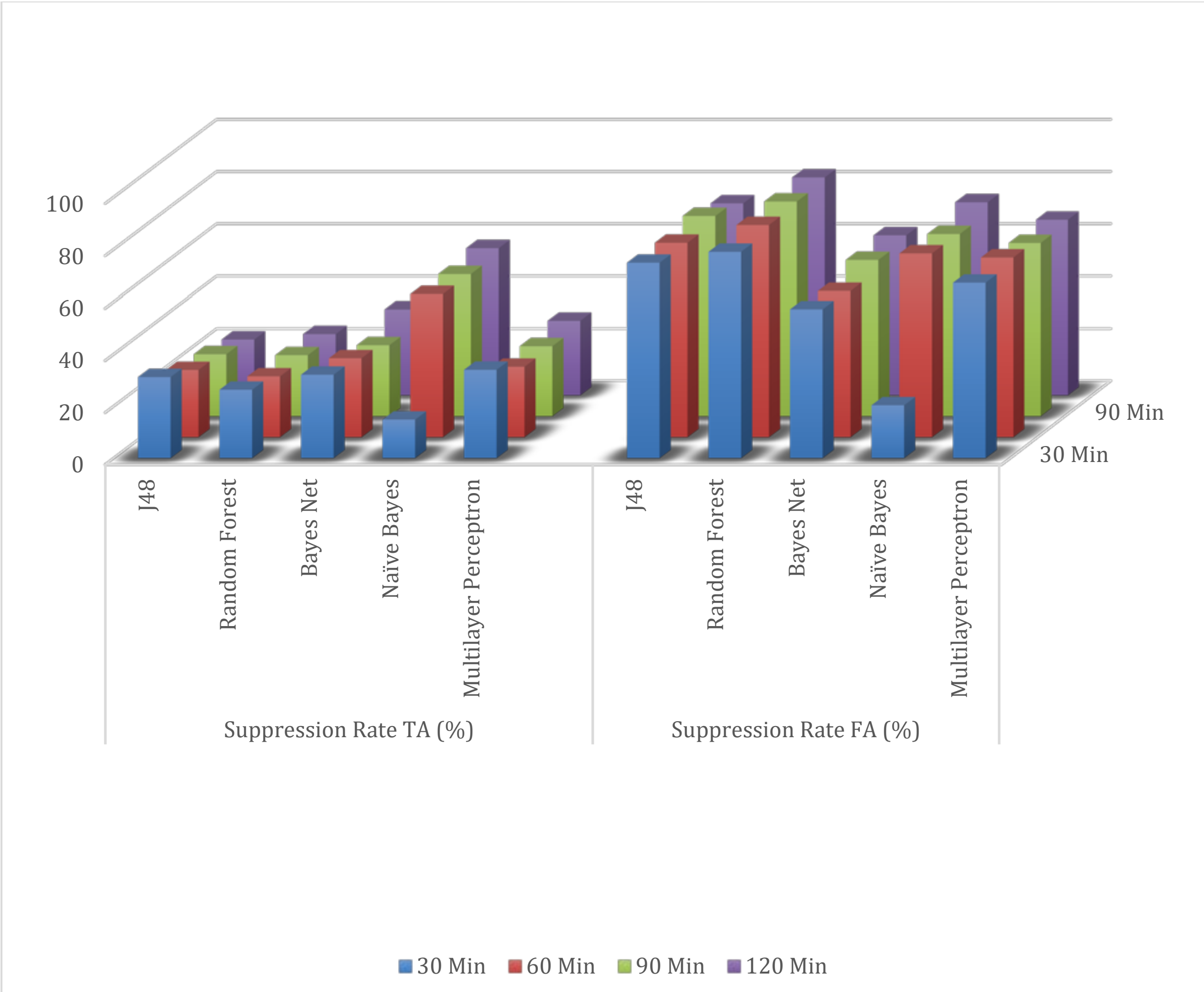

Figure 36: True Alarm & False Alarm Suppression Rates with Standard Deviation Value in Time Dimension for Vtach

Furthermore, in Multilayer Perceptron, when time window was increased from 30 to 60 minutes, true alarm suppression rate was initially decreasing, whereas false alarm rate was increasing, but as time window is increased to 90 minutes; both false alarm suppression rate was decreased. Again increased in time window to 120 min, both true alarm and false alarm suppression rate increased.

Table 55: Comparison of Standard Deviation Value in Time Domain for Vtach

	30 Min	**60 Min**	**90 Min**	**120 Min**

Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	31.25	74.90	25.91	74.48	23.91	76.67	21.47	73.43
Random Forest	26.36	79.08	23.46	81.28	23.55	82.22	23.55	83.37
BayesNet	32.16	57.22	30.43	56.07	27.36	59.83	32.97	61.19
NaiveBayes	14.86	20.40	54.89	70.40	54.44	69.77	56.25	73.85
Multilayer Perceptron	34.06	67.26	27.17	68.83	26.99	66.32	28.62	67.15

5.3.1.4 Time Domain with DFT Value

In Table 32, the data was transformed through DFT with varying time window; we observed that Random Forest performed best among other classifiers. We also observed in Random Forest that when time window was increased from 30 to 120 minutes, the true alarm suppression rate was initially decreasing and then increasing and the false alarm suppression rate was increasing. However, in J48, when time window was increased from 30 to 120 minutes, true alarm suppression rate was decreasing, whereas false alarm rate was initially increasing and then decreasing.

Table 56: Comparison of DFT Value in Time Domain for Vtach

	30 Min		60 Min		90 Min		120 Min	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	27.08	71.13	26.72	72.49	25.00	71.23	24.73	72.49
Random Forest	29.89	73.43	28.44	76.67	26.54	76.15	27.63	79.50
BayesNet	26.63	60.25	24.55	55.54	22.37	54.81	26.72	59.41
NaiveBayes	35.24	50.52	74.55	84.10	63.86	76.57	78.71	84.94
Multilayer Perceptron	27.63	71.23	24.91	74.27	22.55	74.58	27.99	74.27

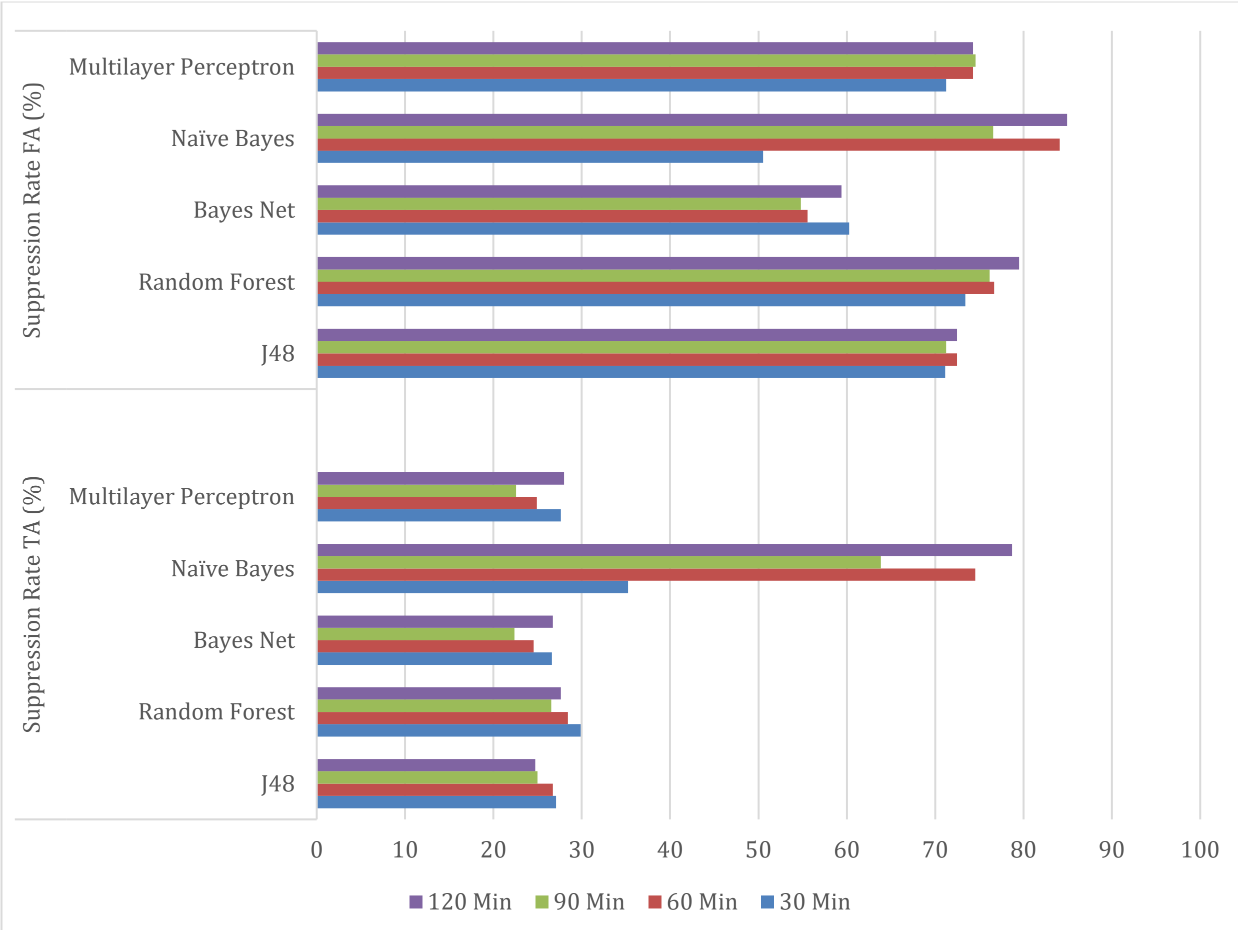

Figure 37: True Alarm & False Alarm Suppression Rates with DFT Value in Time Dimension for Vtach

5.3.2 Comparative Analysis with Data Transformation

We transform data through mean, median, standard deviation, and DFT.

5.3.2.1 Data Transformation in 30 Minutes Time Window

When 30-minute time window was taken in consideration with various data transformation, we observed that Random Forest outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates. We also observed in Random Forest that true alarm suppression rate was initially decreasing and false alarm suppression rate was increasing when data transformation was altered from mean to median. However, when data transformation was changed to standard deviation, true alarm suppression rates started increasing and false alarm suppression rates started decreasing. Again data transformation technique was altered to DFT, the true alarm suppression rates started increasing and false alarm was decreased.

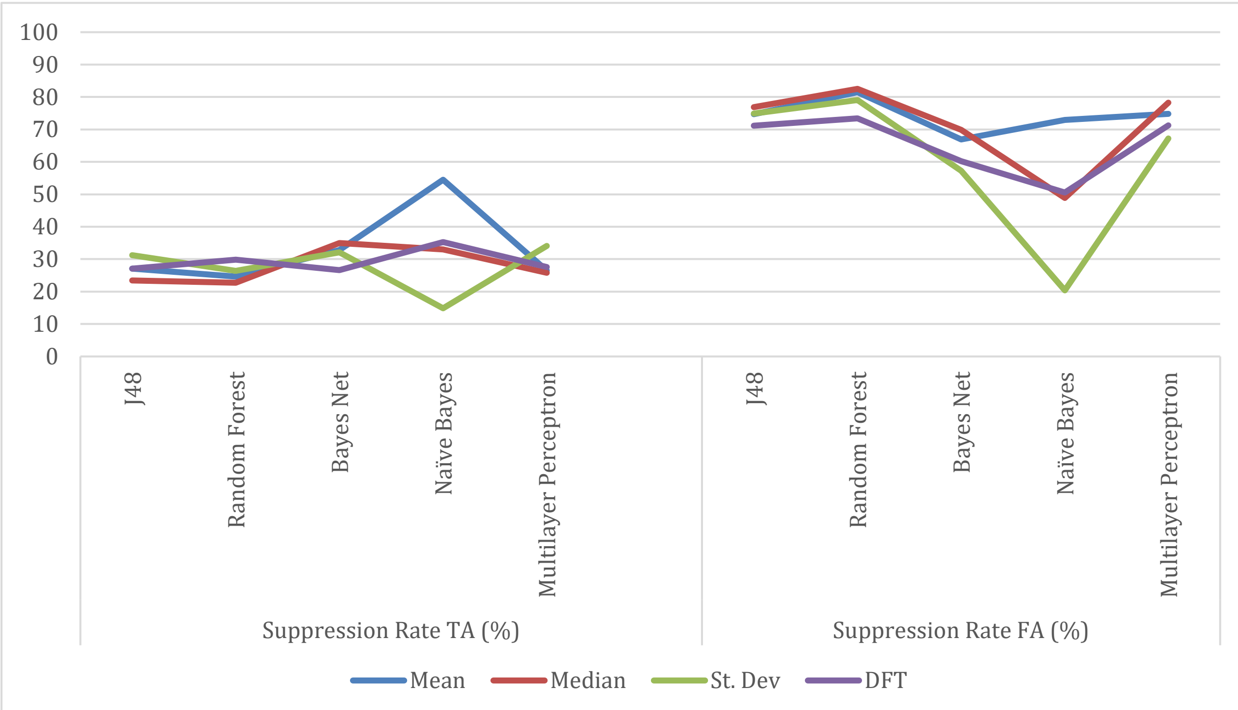

Figure 38: False Alarm & True Alarm Suppression Rates with Data Transformation in 30 Minutes Time Window for Vtach

Table 57: Comparison of Data Transformation in 30 Minutes Time Window for Vtach

	Mean		**Median**		**Std. Deviation**		**DFT**	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	27.08	74.69	23.46	76.88	31.25	74.90	27.08	71.13
Random Forest	24.64	81.49	22.74	82.53	26.36	79.08	29.89	73.43
BayesNet	32.79	66.95	34.96	69.87	32.16	57.22	26.63	60.25
NaiveBayes	54.53	72.91	32.97	48.85	14.86	20.40	35.24	50.52
Multilayer Perceptron	26.45	74.79	25.82	78.24	34.06	67.26	27.63	71.23

5.3.2.2 Data Transformation in 60 Minutes Time Window

Table 58: Comparison of Data Transformation in 60 Minutes Time Window for Vtach

	Mean		Median		Std. Deviation		DFT	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	25.82	77.41	26.99	79.81	25.91	74.48	26.72	72.49
Random Forest	22.83	84.31	22.28	82.64	23.46	81.28	28.44	76.67
BayesNet	30.25	62.13	33.79	68.93	30.43	56.07	24.55	55.54
NaiveBayes	26.00	43.31	29.80	45.08	54.89	70.40	74.55	84.10
Multilayer Perceptron	29.08	81.90	26.54	80.13	27.17	68.83	24.91	74.27

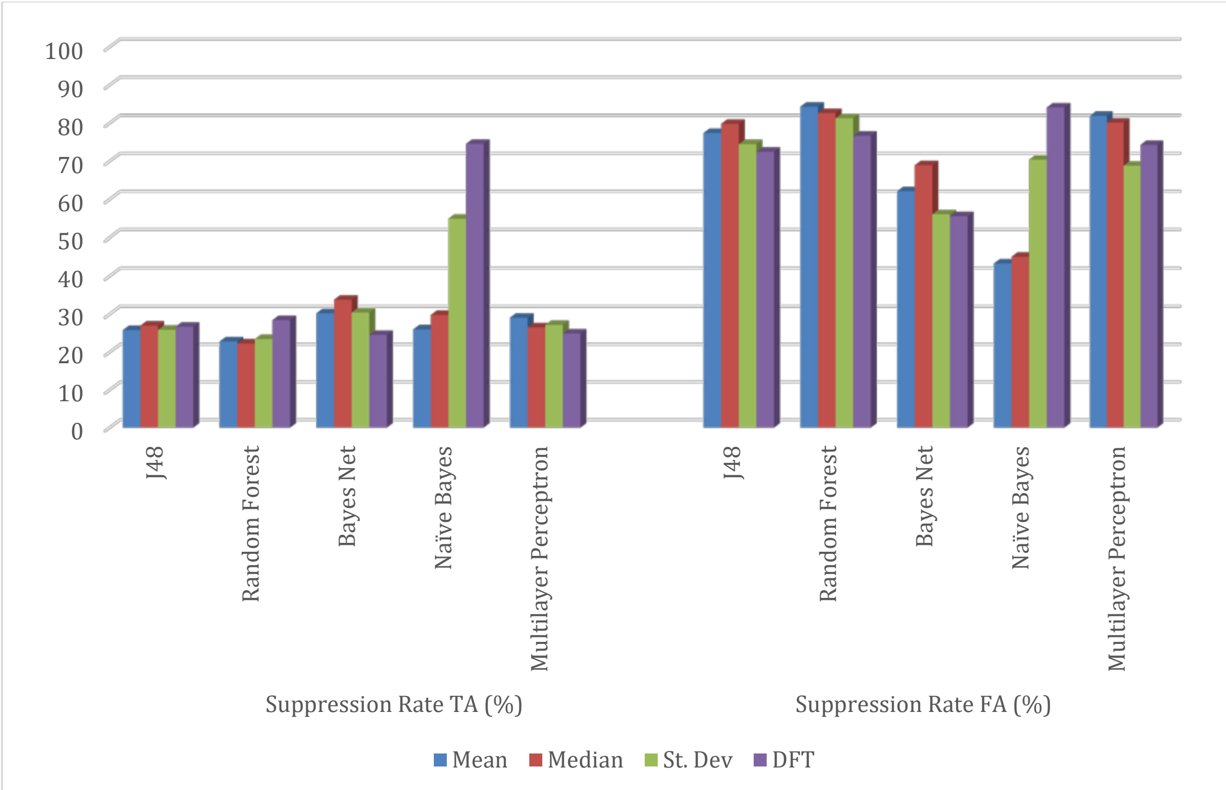

Figure 39: False Alarm & True Alarm Suppression Rates with Data Transformation in 60 Minutes Time Window for Vtach

Considering 60-minutes time window with various data transformation technique, we observed that Random Forest outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates. We also observed from Table 34, Random Forest with standard deviation performed the best with true suppression rate of 22.83% and false alarm suppression rate of 82.64%.

5.3.2.3 Data Transformation in 90 Minutes Time Window

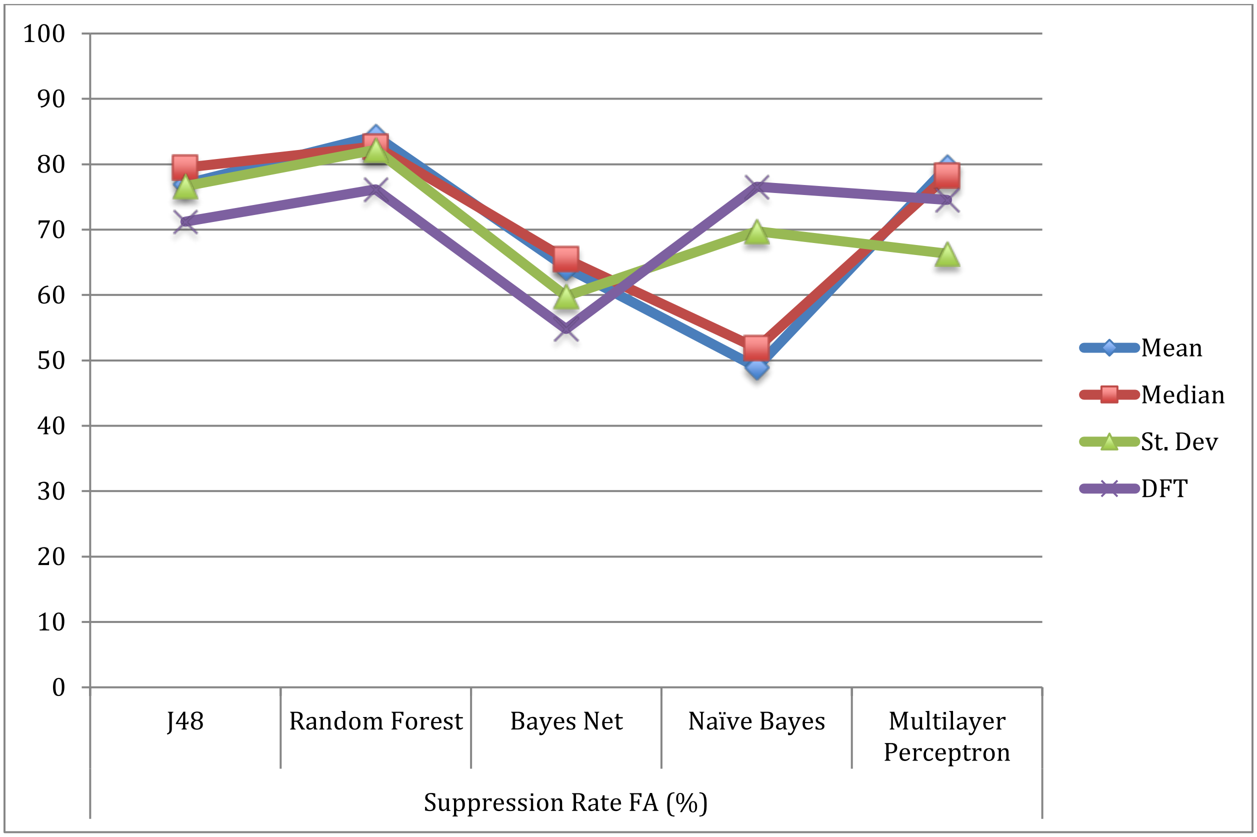

Figure 40: False Alarm Suppression Rates with Data Transformation in 90 Minutes Time Window for Vtach

When 90-minutes time window was taken in consideration with various data transformation, we observed that Random Forest outperformed other classifier resulting in high false alarm suppression and low true alarm suppression rates. Furthermore, BayesNet had high true alarm suppression rates and low false alarm suppression rates.

Table 59: Comparison of Data Transformation in 90 Minutes Time Window for Vtach

	Mean		**Median**		**Std. Deviation**		**DFT**	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	26.27	76.99	22.74	79.50	23.91	76.67	25.00	71.23
Random Forest	21.01	84.31	19.93	82.74	23.55	82.22	26.54	76.15

BayesNet	33.06	64.23	34.51	65.48	27.36	59.83	22.37	54.81
NaiveBayes	29.17	48.95	35.42	51.88	54.44	69.77	63.86	76.57
Multilayer Perceptron	25.82	79.60	25.18	78.24	26.99	66.32	22.55	74.58

5.3.2.4 Data Transformation in 120 Minutes Time Window

Table 60: Comparison of Data Transformation in 120 Minutes Time Window for Vtach

	Mean		**Median**		**Std. Deviation**		**DFT**	
Classification Algorithms	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**	**S-Rate TA**	**S-Rate FA**
J48	21.01	75.52	22.46	75.84	21.47	73.43	24.73	72.49
Random Forest	21.29	84.00	20.38	80.96	23.55	83.37	27.63	79.50
BayesNet	30.98	60.56	36.23	64.54	32.97	61.19	26.72	59.41
NaiveBayes	39.04	58.16	34.87	51.67	56.25	73.85	78.71	84.94
Multilayer Perceptron	24.37	76.57	24.73	75.84	28.62	67.15	27.99	74.27

Considering 120-minutes time window with various data transformation technique, we observed that Random Forest with mean value outperformed other classifier resulting in false alarm suppression of 84% and low true alarm suppression rates of 21.29% when compared with other data transformation technique.

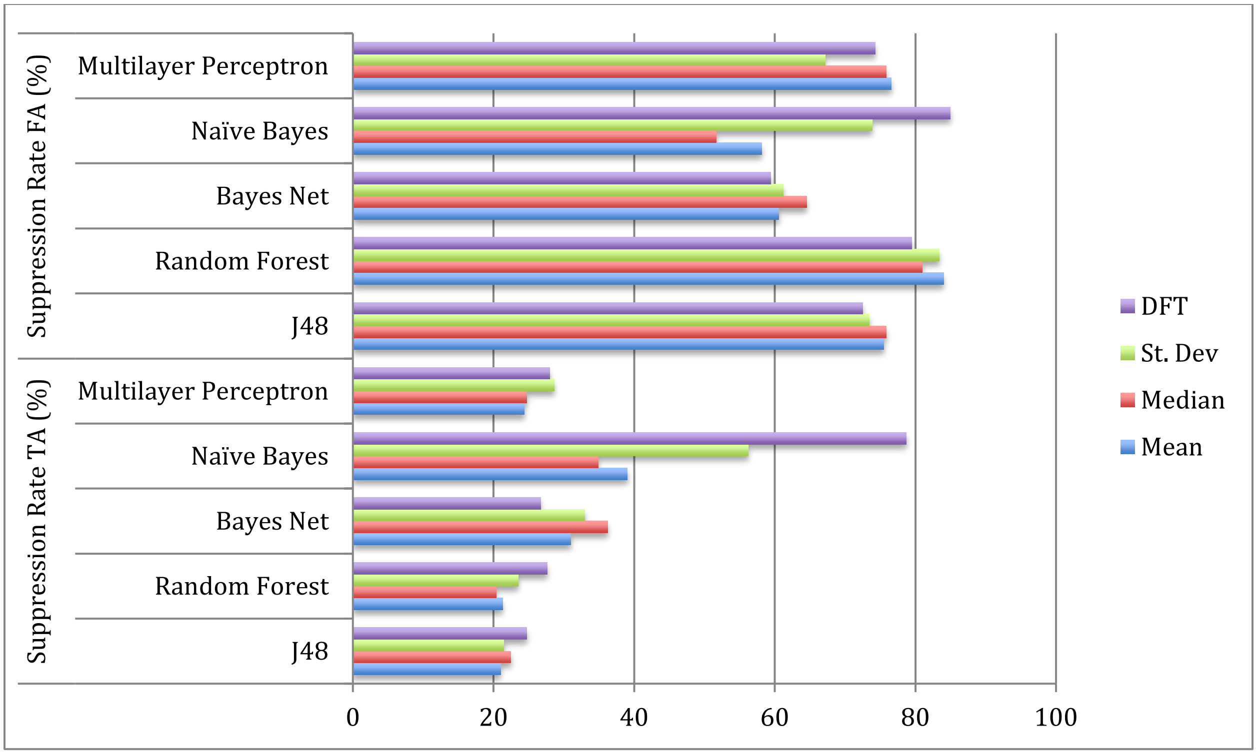

Figure 41: False Alarm & True Alarm Suppression Rates with Data Transformation in 120 Minutes Time Window for Vtach

5.3.3 Comparative Analysis with Feature Sets

We use different feature sets such as CFS Subset Evaluator, Wrapper Subset Evaluator (using Bayes Net, NaiveBayes, J48, Random Forest), and Info Gain Evaluator.

5.3.3.1 Feature Sets with Mean Value

5.3.3.1.1 Feature Sets with Mean Value in 30 Minutes Window

Considering 30-minutes time window with mean value, we observed that feature selection obtained from Wrapper method including J48 performed best with Random Forest as classifier in comparison to the feature sets obtained from CFS, Wrapper method including NaiveBayes, Wrapper method including Random Forest, and Information gain. Wrapper method including J48 resulted in low true alarm suppression rate of 23.91% and high false alarm suppression rate of 79.5%.

Table 61: Comparison of Feature Sets with Mean Value in 30 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	28.44	73.74	30.07	65.06	21.83	71.44	27.17	73.95
Random Forest	25.27	79.71	27.81	73.12	23.91	79.50	24.73	80.65
BayesNet	28.44	64.44	31.88	60.46	29.62	60.25	30.07	66.00
NaiveBayes	24.28	43.62	28.17	50.84	17.93	36.82	20.11	38.08
Multilayer Perceptron	30.07	70.29	33.15	58.37	29.80	68.10	30.34	78.87

5.3.3.1.2 Feature Sets with Mean Value in 60 Minutes Window

Table 62: Comparison of Feature Sets with Mean Value in 60 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	32.61	81.28	30.16	68.31	28.80	78.77	23.64	76.15
Random Forest	24.91	81.49	29.35	74.58	25.09	81.17	22.64	83.37
BayesNet	25.82	61.51	26.18	56.38	23.64	51.46	32.79	66.95
NaiveBayes	27.81	46.13	16.76	37.87	18.75	34.73	20.83	33.89
Multilayer Perceptron	35.42	78.24	35.60	64.54	29.26	70.50	27.90	78.24

5.3.3.1.3 Feature Sets with Mean Value in 90 Minutes Window

Table 63: Comparison of Feature Sets with Mean Value in 60 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	26.81	74.16	31.25	70.29	25.00	76.88	24.28	73.33
Random Forest	22.55	82.85	26.09	75.21	22.74	82.74	22.64	83.89
BayesNet	31.52	62.03	26.54	54.92	26.09	50.73	33.15	64.23
NaiveBayes	28.80	49.69	17.57	38.81	56.34	77.20	55.43	74.79
Multilayer Perceptron	36.14	78.56	40.94	69.87	26.45	66.32	29.62	77.62

5.3.3.1.4 Feature Sets with Mean Value in 120 Minutes Window

Table 64: Comparison of Feature Sets with Mean Value in 120 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	18.93	66.95	21.56	74.16	22.74	75.00	22.37	73.54
Random Forest	23.28	80.33	15.62	77.26	21.38	83.37	21.47	84.21
BayesNet	30.71	59.62	29.98	62.13	37.59	66.42	30.25	59.10
NaiveBayes	37.14	57.53	33.15	58.05	24.46	39.54	52.36	75.84
Multilayer Perceptron	46.29	77.30	27.45	74.69	24.18	77.93	25.82	78.35

Taking consideration of 60-minutes, 90-minutes, and 120-minutes time window with mean value, we observed that feature selection obtained from Wrapper method including Random Forest as classifier performed best in 60, 90 and 120 minutes of data with comparatively high false alarm suppression rates. In 120 minutes of time window, Wrapper including Random Forest method with Random Forest performed the best in comparison to the feature sets obtained from CFS, Wrapper method including NaiveBayes, Wrapper method including Random Forest, and Information gain with 21.47% of true alarm suppression rates and false alarm suppression rates of 84.21%.

5.3.3.2 Feature Sets with Median Value

Considering 30-minutes, 60-minutes, 90-minutes, and 120-minutes time window with median value, we observed that feature selection obtained from Random Forest including Random Forest method with Random Forest as classifier performed best in 120 minutes time window with true alarm suppression rate of 20.92%, and false alarm suppression rate of 82.32% when comparing with all time-windows.

5.3.3.2.1 Feature Sets with Median Value in 30 Minutes Window

Table 65: Comparison of Feature Sets with Median Value in 30 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	25.27	76.26	27.54	76.26	20.83	73.95	23.19	76.26
Random Forest	22.37	81.69	23.10	79.08	22.19	76.99	23.10	80.02
BayesNet	29.53	65.48	25.54	60.36	37.95	73.85	31.16	68.72
NaiveBayes	21.47	40.90	19.66	41.53	21.92	41.00	27.63	44.87
Multilayer Perceptron	31.34	73.85	29.17	69.87	29.26	63.60	29.44	73.43

5.3.3.2.2 Feature Sets with Median Value in 60 Minutes Window

Table 66: Comparison of Feature Sets with Median Value in 60 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	25.82	75.52	30.53	69.77	22.92	77.30	25.09	75.84
Random Forest	24.28	80.44	27.72	75.21	20.20	81.28	16.41	76.06
BayesNet	33.06	64.96	28.17	56.59	30.90	64.64	33.06	66.84
NaiveBayes	19.47	39.33	11.68	30.86	27.99	50.00	22.92	40.48
Multilayer Perceptron	29.89	75.73	24.46	41.32	25.18	73.01	26.18	74.69

5.3.3.2.3 Analysis of Feature Sets with Median Value in 90 Minutes Window

Table 67: Comparison of Feature Sets with Median Value in 90 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	25.36	77.62	21.83	69.25	23.46	77.82	23.55	79.60
Random Forest	21.20	81.07	21.29	80.02	20.47	82.01	21.01	82.64
BayesNet	25.00	54.50	32.16	59.62	33.06	63.60	33.70	62.24
NaiveBayes	22.55	41.95	18.03	40.17	35.60	53.14	27.45	43.72
Multilayer Perceptron	33.15	78.66	28.35	58.05	27.54	74.58	28.53	80.75

5.3.3.2.4 Analysis of Feature Sets with Median Value in 120 Minutes Window

Table 68: Comparison of Feature Sets with Median Value in 120 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	22.46	71.23	36.05	73.43	22.46	75.31	23.19	77.20
Random Forest	21.38	81.59	28.44	73.64	21.11	83.47	20.92	82.32
BayesNet	29.89	60.98	51.09	80.33	35.69	64.96	34.33	62.66
NaiveBayes	21.56	40.79	19.93	42.15	27.26	42.26	27.17	41.95
Multilayer Perceptron	35.24	79.29	29.35	45.19	27.36	78.66	26.90	76.46

5.3.3.3 Feature Sets with Standard Deviation Value

5.3.3.3.1 Feature Sets with Standard Deviation Value in 30 Minutes Window

Table 69: Comparison of Feature Sets with Standard Deviation Value in 30 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	24.55	55.23	30.16	55.54	28.44	71.97	32.25	75.52
Random Forest	33.51	67.26	47.92	61.40	26.81	79.60	24.91	78.14
BayesNet	24.64	53.56	26.54	52.41	28.89	54.08	32.16	57.22

NaiveBayes	10.87	16.42	8.24	18.93	18.84	26.26	21.11	27.72
Multilayer Perceptron	30.62	55.44	28.44	49.16	29.53	66.11	31.70	69.04

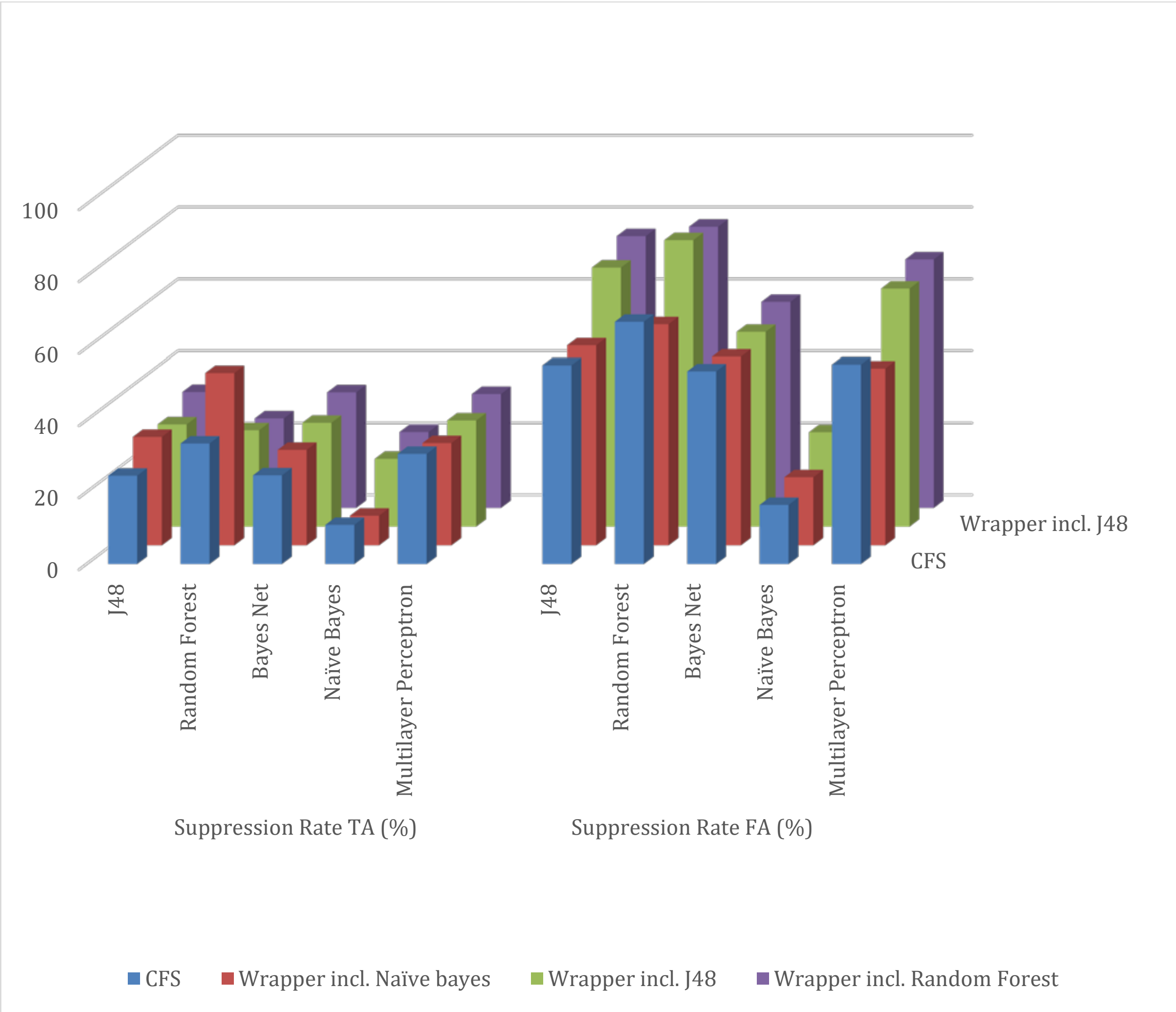

Figure 42: Comparison of Feature Sets with Standard Deviation Value in 30 Minutes Time Window for Vtach True & False Alarm Suppression Rates

Among 30-minutes, 60-minutes, 90-minutes, and 120-minutes time window with standard deviation value, we observed that feature selection obtained from Wrapper including Random Forest method with Random Forest as classifier performed best in 120 minutes time window with true alarm suppression rate of 23.1%, and false alarm suppression rate of 82.11%.

5.3.3.3.2 Feature Sets with Standard Deviation Value in 60 Minutes Window

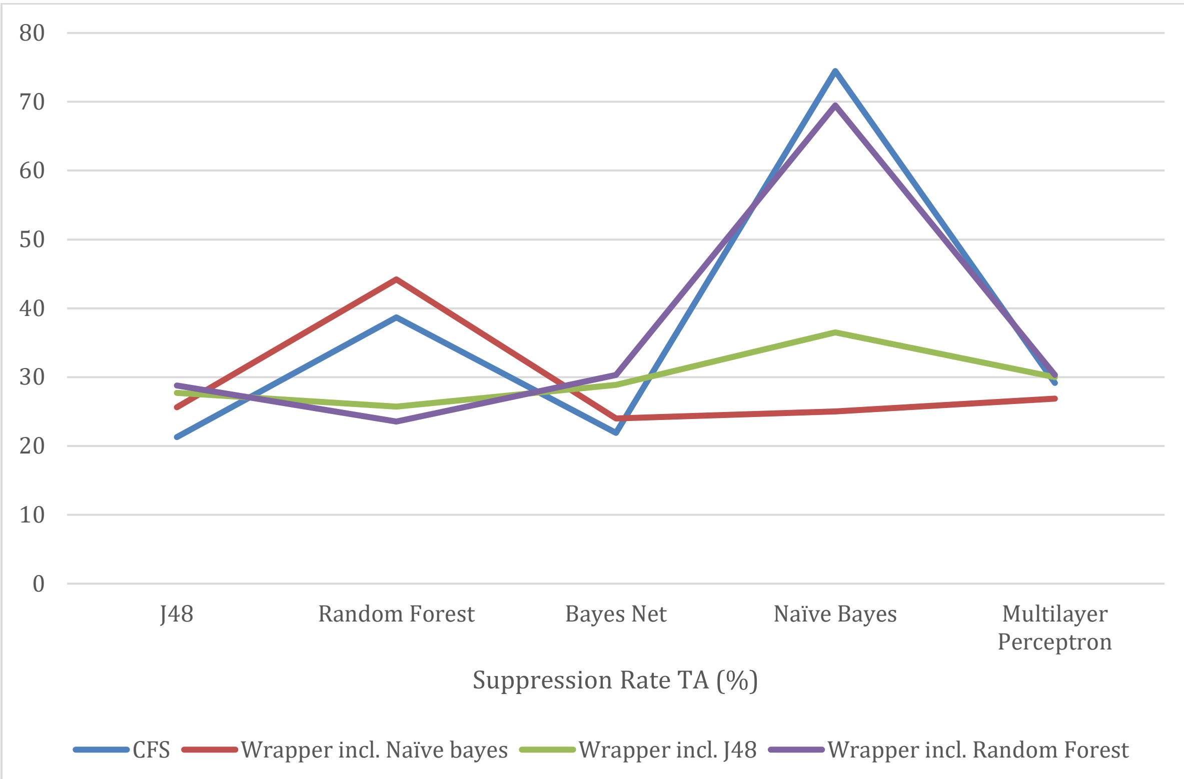

Figure 43: Comparison of Feature Sets with Standard Deviation Value in 60 Minutes Time Window for Vtach True Alarm Suppression Rates

Table 70: Comparison of Feature Sets with Standard Deviation Value in 60 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	21.29	53.77	25.63	55.02	27.72	76.36	28.80	78.77
Random Forest	38.68	66.74	44.20	62.76	25.72	81.28	23.55	81.38
BayesNet	21.92	51.26	24.00	51.46	28.89	55.54	30.34	56.49

NaiveBayes	74.46	81.69	25.00	50.73	36.50	55.23	69.47	79.81
Multilayer Perceptron	29.17	56.59	26.90	52.72	29.98	70.61	30.34	66.74

5.3.3.3.3 Feature Sets with Standard Deviation Value in 90 Minutes Window

Table 71: Comparison of Feature Sets with Standard Deviation Value in 90 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	30.43	73.33	28.08	59.21	22.01	76.78	26.72	78.45
Random Forest	25.63	76.99	35.69	68.10	23.64	79.08	24.55	84.10
BayesNet	22.46	52.20	29.62	55.65	26.18	59.52	27.54	55.23
NaiveBayes	56.07	68.20	25.91	50.42	66.03	76.57	39.86	59.31
Multilayer Perceptron	37.23	71.55	34.06	62.03	29.62	70.40	30.25	71.23

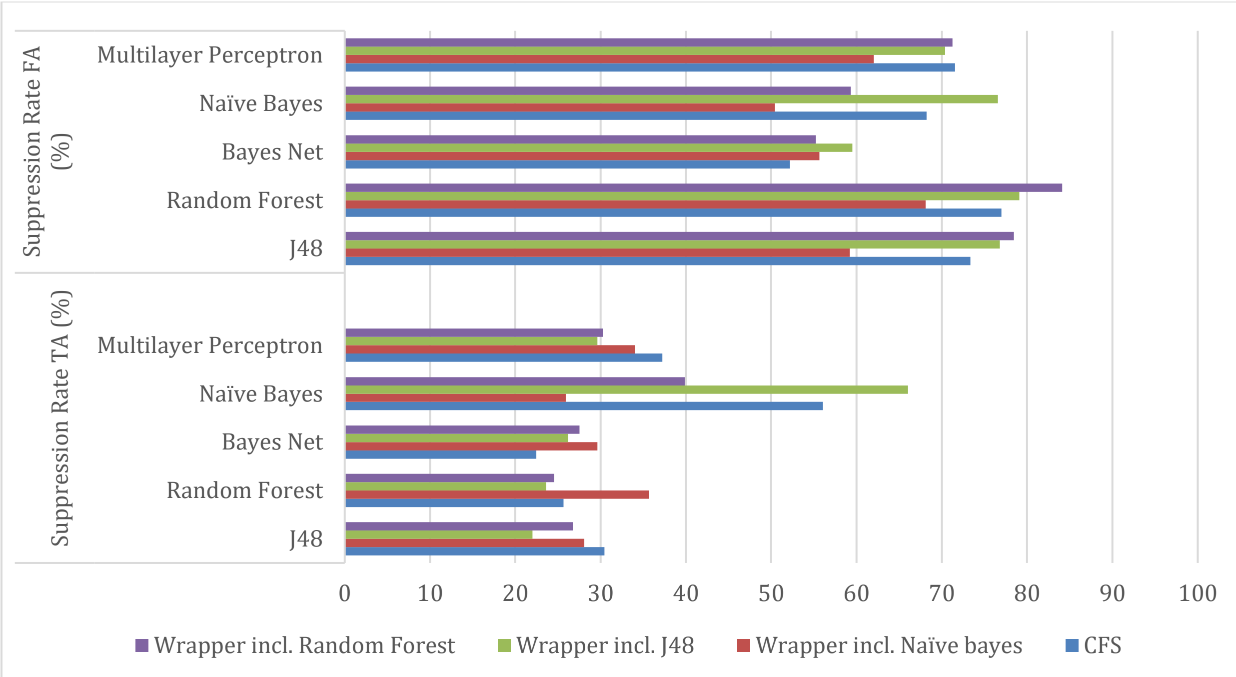

Figure 44: Comparison of Feature Sets with Standard Deviation Value in 90 Minutes Time Window for Vtach True & False Alarm Suppression Rates

5.3.3.3.4 Feature Sets with Standard Deviation Value in 120 Minutes Window

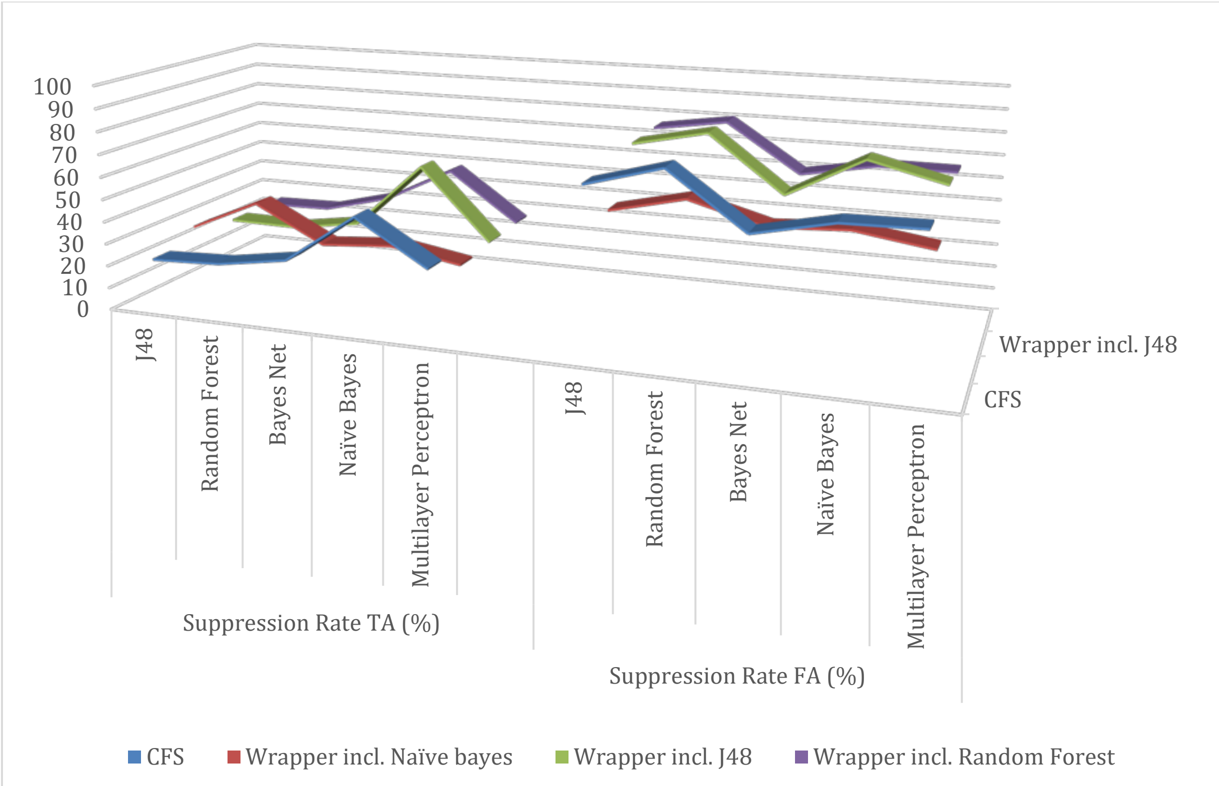

Figure 45: Comparison of Feature Sets with Standard Deviation Value in 120 Minutes Time Window for Vtach True & False Alarm Suppression Rates

Table 72: Comparison of Feature Sets with Standard Deviation in 120 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	21.83	71.97	28.44	53.56	23.55	75.21	23.73	75.84
Random Forest	23.10	81.07	42.30	61.19	23.10	82.11	24.28	80.75
BayesNet	27.99	58.16	26.27	52.41	27.81	58.05	32.25	59.83
NaiveBayes	50.00	65.38	28.80	54.29	57.25	75.63	48.19	65.90
Multilayer Perceptron	30.80	66.00	23.46	49.48	25.00	67.68	25.91	65.79

5.3.3.4 Feature Sets with DFT Value

5.3.3.4.1 Feature Sets with DFT Value in 30 Minutes Window

Table 73: Comparison of Feature Sets with DFT Value in 30 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	31.07	69.56	28.44	74.90	22.83	71.97	26.54	76.57
Random Forest	27.54	76.57	26.90	76.67	26.00	80.02	25.00	81.07
BayesNet	23.19	56.07	26.36	53.35	37.41	63.08	30.71	64.96
NaiveBayes	24.46	45.61	18.39	42.78	51.72	69.56	28.08	47.28
Multilayer Perceptron	35.05	72.07	28.89	69.25	27.99	70.82	26.00	77.72

5.3.3.4.2 Feature Sets with DFT Value in 60 Minutes Window

Table 74: Comparison of Feature Sets with DFT Value in 60 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	30.89	74.37	33.97	73.64	24.18	72.80	31.43	80.33
Random Forest	27.99	76.99	31.79	73.64	21.92	81.28	23.01	82.01
BayesNet	23.10	55.02	22.28	48.95	24.00	48.33	29.17	55.75
NaiveBayes	32.44	57.99	18.21	44.46	24.55	44.46	54.35	73.22

Multilayer Perceptron	38.41	77.82	34.87	60.98	22.74	69.25	29.62	79.92

5.3.3.4.3 Feature Sets with DFT Value in 90 Minutes Window

Table 75: Comparison of Feature Sets with DFT Value in 90 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	28.53	73.85	39.67	75.94	21.29	74.27	22.46	72.80
Random Forest	25.82	78.66	29.17	74.90	23.46	79.92	22.10	79.81
BayesNet	19.93	52.09	38.23	65.79	20.92	49.06	22.28	46.86
NaiveBayes	50.45	75.84	15.40	41.63	24.91	45.19	53.80	80.23
Multilayer Perceptron	39.95	77.62	36.05	62.13	31.70	70.19	25.91	66.32

5.3.3.4.4 Feature Sets with DFT Value in 120 Minutes Window

Table 76: Comparison of Feature Sets with DFT Value in 120 Minutes Time Window for Vtach

	CFS		**Wrapper incl. Naïve Bayes**		**Wrapper incl. J48**		**Wrapper incl. Random Forest**	
Classification Algorithms	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA	S-Rate TA	S-Rate FA
J48	24.18	74.69	23.10	64.33	20.11	76.78	20.92	76.15
Random Forest	24.82	81.49	24.73	74.48	22.10	82.85	22.55	81.49
BayesNet	33.03	69.83	24.09	49.58	30.71	56.90	26.27	53.14
NaiveBayes	46.92	77.09	21.74	47.80	73.01	83.47	79.08	89.12

Multilayer Perceptron	29.98	75.10	25.36	53.03	25.82	71.23	28.08	63.60

Among 30-minutes, 60-minutes, 90-minutes, and 120-minutes time window with DFT, we observed that feature selection obtained from Wrapper including J48 method with Random Forest as classifier performed best in 120 minutes time window with true alarm suppression rate of 22.1%, and false alarm suppression rate of 82.85%.

5.4 Using Ensemble Approach

5.4.1 Stacking

The comparative analysis in time domain, data transformation technique and use of feature set helps to analyze and build a model that can suppress the high rate of false alarm preserving the true alarms. 90 min data with median transformation with stacking approach resulted in high false alarm suppression. In the stacking approach, the base classifiers were IBK, J48, KStar, and random forest where the meta classifier was J48 with confidence factor 0.3. More than 85% missing data were deleted. A little adjustment was made and the record that does not have ABP systolic, and diastolic was deleted. The false alarm suppression rate achieved was 1.33% and true alarm suppression rate was 80.7%.

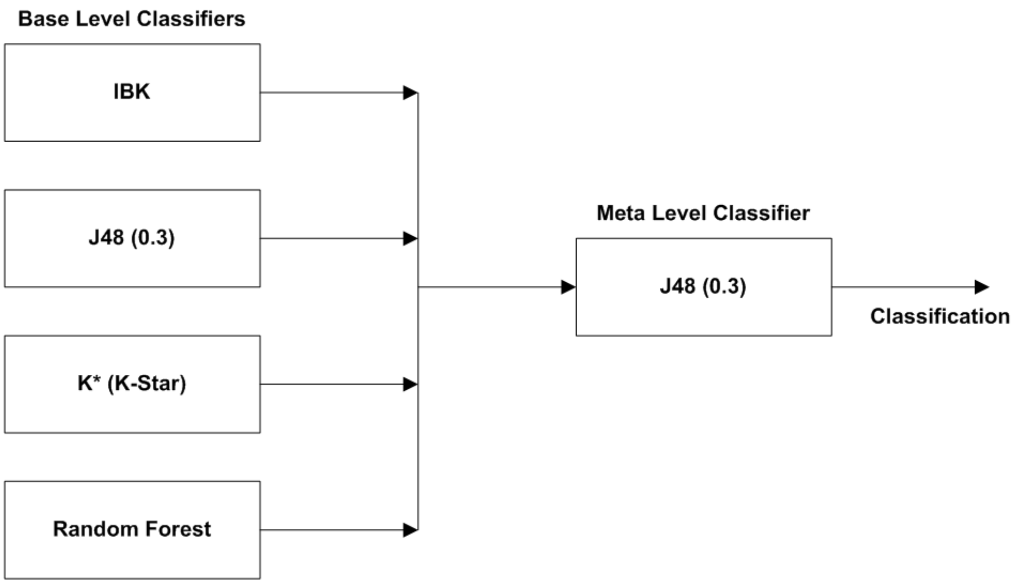

Figure 46: Model Ensemble using Stacking Approach for TACHY

5.4.2 Voting

Data of 90 minutes with median data transformation with feature sets based on information gain taking voting approach in account resulted in high false alarm suppression. In the voting approach, the classifiers were IBK, random forest and KStar where the combination rule used was average of probabilities. A little adjustment was made and the records that does not have ABP systolic, diastolic and pulse were deleted. The false alarm suppression rate achieved was 15.87% and true alarm suppression rate was 80.19%.

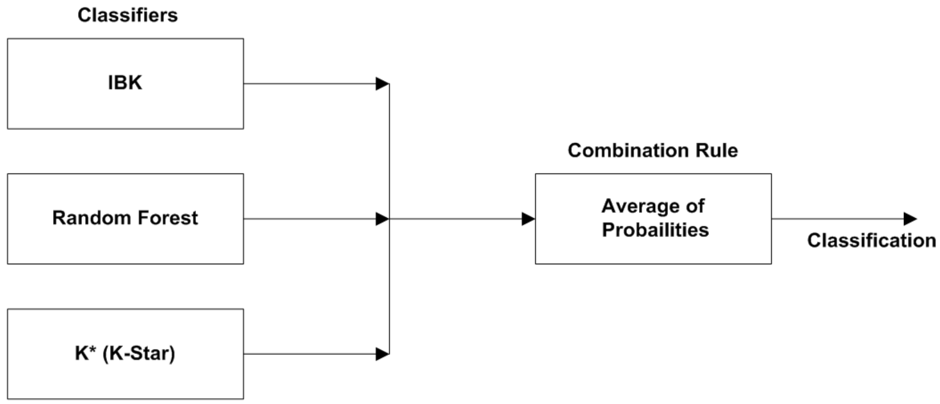

Figure 47: Model Ensemble using Voting Approach for VTACH

Data of 90 minutes with standard deviation data transformation with voting approach resulted in high false alarm suppression. In the voting approach, the classifiers were IBK and KStar where the combination rule used was average of probabilities. More than 85% missing data were deleted. A little adjustment was made and the records that does not have ABP systolic, diastolic and pulse were deleted. The false alarm suppression rate achieved was 2.38% and true alarm suppression rate was 81.88%.

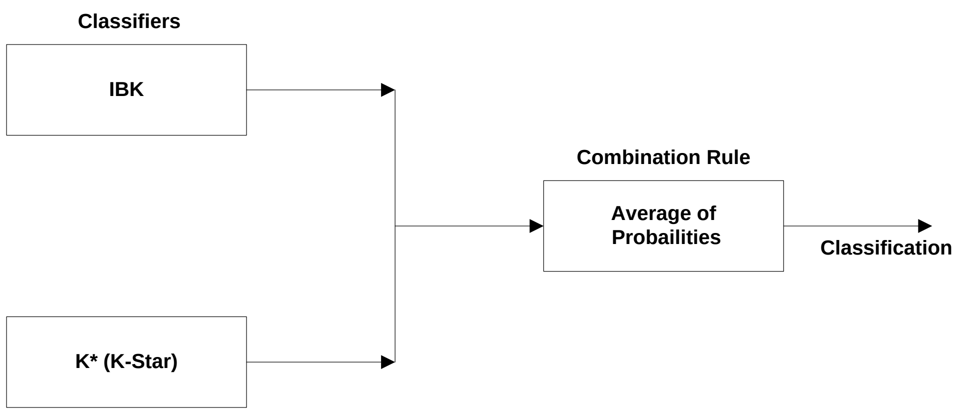

Figure 48: Model Ensemble using Voting Approach for BRADY

Table 77: Result through Ensemble Learning

	BRADY	TACHY	VTACH
S-Rate TA (%)	2.38	1.33	15.87
S-Rate FA (%)	81.88	80.7	80.19

Precision	0.933	0.936	0.824
Recall	0.934	0.935	0.824
F-Measure	0.932	0.933	0.824

5.5 Evaluation

Our method resulted in 81.88% of false alarms and 2.38 % of true alarms with 90 minutes of data with standard deviation value with voting approach and classifier as IBK and K-Star in bradycardia alarms. Likewise, in tachycardia alarms, data of 90 minutes with median value with stacking approach and base classifiers as IBK, J48, K-Star, and Random Forest with meta classifier as J48 (0.3) resulted in 80.7% of false alarm suppression and 1.33% of true alarm suppression. Again, ventricular tachycardia alarms were hard to classify with a suppression rate of false alarms of 80.19% and a suppression of true alarms of 15.87% with 90 minutes of data with median value with voting approach and classifier as IBK, K-Star, and Random Forest.

Table 78: Evaluation of Result

	BRADY			**TACHY**			**VTACH**		
	S-Rate TA	S-Rate FA	Alarms	S-Rate TA	S-Rate FA	Alarms	S-Rate TA	S-Rate FA	Alarms
Our Approach	2.38	81.88	708	1.33	80.7	1966	15.87	80.19	2060
Baumgartner et al. (2012)	2.6	81.54	258	1.62	80.47	971	17.73	75.24	597
Aboukhalil et al. (2008)	0	81	717	0	63.7	1877	9.4	33	1900

Compared to the work of Baumgartner et al. (2012), our data fusion approach resulted in high false alarm suppression with minimal true alarm suppression for bradycardia, tachycardia, and ventricular tachycardia alarms with significantly high number of alarm than Baumgartner et al. (2012). In case of Vtach alarms, our method resulted in high true alarm suppression compared to bradycardia and tachycardia alarms, but it still outperforms the Baumgartner et al. (2012)

approach. When compared with Aboukhalil et al. (2008), our result in suppressing the false alarms was significantly better for tachycardia and ventricular-tachycardia alarms. However, Aboukhalil et al. (2008) reported a 0% reduction of true alarms except for ventricular tachycardia alarms, but the number of alarms was greater in tachycardia and ventricular tachycardia alarms, and the number of alarms was almost similar in bradycardia alarms in our case.

Overall, our data fusion approach resulted in high false alarm suppression rates with very low true alarm suppression rates except for ventricular tachycardia. We believe our approach has promising results as we have more number of alarms, studied in different time domain with various feature sets and algorithms.

5.6 Result Implications - Bradycardia

- ***Explanation of better performing algorithm***

  Phua et al. (2010) illustrate that optimal results can be derived from a model that combines multiple algorithms. We found that ensemble technique with voting approach of IBK and K-Star algorithm with standard deviation transformation and 90 minutes time window is the best performing combination among others that are explored in the paper for reducing false alarms in bradycardia. Numerous studies (Veerappan et al., 2000; Yahalom et al., 2013) acknowledge that standard deviation has been used to study in case of bradycardia. Aboukhalil et al. (2008) used 17 seconds of fixed time slots to study bradycardia alarms and achieved 81% of false alarm suppression. To our knowledge, no research has been conducted in the bradycardia alarms that studied across various time slots. We used 30, 60, 90, and 120 minutes of the time window for the study of bradycardia alarms and identified 90 minutes with the above combination resulted in 81.88% of false alarm suppression in bradycardia.

- ***Impact of results for Medical device makers***

  Given the result of bradycardia, medical device makers should design smart alarm systems that use algorithms to interpret data in order to alert clinicians instead of alarm threshold values. The algorithm in medical device to generate alarms for bradycardia is recommended to use an ensemble technique i.e. voting approach with IBK and K-Star algorithm and perform standard deviation data transformation and use 90 minutes time to generate more accurate alarms.

- ***Impact of results for Clinicians/Hospitals***

The development of alarm systems utilizing the above mentioned combination of algorithms and data transformation techniques to reduce false alarms in bradycardia impacts clinician in reducing alarm fatigue. Therefore, clinician can devote more time in patient care that improves the patient safety, which is the goal in ICU.

- ***Future Research***

The existing MIMIC II dataset we used to study bradycardia alarms were a mix of false and true alarms, but were not comprised of missed alarms (Baumgartner, Rödel, & Knoll, 2012). This is an area of future research. Such datasets need to be developed and investigate missed alarm event for bradycardia is a part of future work.

5.7 Result Implications - Tachycardia

- ***Explanation of better performing algorithm***

We found that ensemble technique with stacking approach of IBK, J48, Random Forest and K-Star algorithm with median transformation and 90 minutes time window perform better than other algorithms in minimizing false alarms in tachycardia. Nandhini and Subhasini (2013) illustrate that median has been used to study in case of tachycardia. Aboukhalil et al. (2008) achieved 63.7% of false alarm suppression in tachycardia alarms, however our method resulted in 80.7% of false alarm suppression with a significantly higher number of alarm records. Aboukhalil et al. (2008) reported a 0% reduction of true alarms in tachycardia with low false alarm suppression. It means it minimizes fewer false alarms, which may lead to alarm fatigue and probably in switching off the alarms that may suppress true alarms as well.

- ***Impact of results for Medical device makers***

Medical device manufacturers should develop intelligent alarms, in which the alarm system takes into account multi-parameters. By doing so the alarm system may minimize false alarms, for instance: alarm for a high pulse rate caused by pulse oximetry sensor motion can be avoided if the heart rate determined by the ECG signal remains stable.

However, medical devices alert clinician based on a single parameter. So, it is advisable to change the architecture of the device from a single parameter to multi-parameter approach. Implementing multi-parameter approach could bring changes in medical device software in parameter acquisition as well as in alarming techniques as the medical device may require more than one criterion to be met to alert clinician.

- ***Impact of results for Clinicians/Hospitals***

With the implementation of multi-parameter approach, the changes in medical devices may occur with operating and managing the device that may lead clinicians for further training. Institutions need to provide effective education and training to better understand the proper operation, the implications of misconfiguration, advantages and the limitations of alarm systems. The training should be designed so that devices are operated in their normal clinical environments and should include information on the institution's alarm setting and response protocols.

- ***Future Research***

The smart alarms for tachycardia are likely to enhance patient outcomes by incorporating multi-parameter data. In tachycardia alarms, we studied in 30 minutes time interval from 30 minutes up to 2 hours and identified 90 minutes resulted in better prediction of false alarms. In future, we consider studying different time intervals to generate more accurate alarms in tachycardia.

5.8 Result Implications - Ventricular Tachycardia

- ***Explanation of better performing algorithm***

Ensemble approach tends to yield better performance than single algorithms (Kuncheva & Whitaker, 2003; Sollich & Krogh, 1996). We found that ensemble technique with voting approach of IBK, Random Forest and K-Star algorithm with median transformation and 90 minutes time window perform better than other algorithms in minimizing false alarms in ventricular tachycardia. Nandhini and Subhasini (2013) illustrate that median has been used to study in case of ventricular tachycardia. Aboukhalil et al. (2008) achieved 33% of false alarm suppression for ventricular tachycardia alarms. However, our method resulted in 80.19% of false alarm suppression with a significantly higher number of alarm records. Our combination also resulted better in false alarm suppression when compared to Baumgartner, Ro¨del, and Knoll (2012).

- ***Impact of results for Medical device makers***

FDA has adopted IEC 60601-1-8, as a reference standard that provides general requirements for alarm systems. It is the only focused alarm standard intended to be applied to all medical devices with alarms (American College of Clinical Engineering, 2006). If the architecture of the device is changed to multi-parameter approach, the standard for alarm systems also needs to be updated. As the new architecture is implemented, the alarm standard should be redesigned and reevaluated to incorporate new changes.

- ***Impact of results for Clinicians/Hospitals***

Ventricular tachycardia is common in patients with congestive heart failure (Baher & Valderrabano, 2013). The numerous false alarms for ventricular tachycardia interrupt clinical workflow and can result in missed tasks as well as reduced productivity in clinician. The above combination of algorithms with ensemble technique, and data transformation minimizes the false alarms in ventricular tachycardia. Reducing ventricular tachycardia false alarms may result in increased productivity in clinicians and improve patient outcomes.

- ***Future Research***

Our result shows that the combination of ensemble technique with voting approach of IBK, Random Forest and K-Star algorithm with median data transformation for 90 minutes time window data perform better in ventricular tachycardia comparing to Aboukhalil et al. (2008). It resulted in true alarm suppression of 15.87%, which is relatively high when compared to other alarms such as bradycardia and tachycardia. The future work in ventricular tachycardia alarm is to improve true alarm suppression rates by studying other data transformation technique such as logarithmic transformations.

Chapter 6

Simulation

6.1 Simulation in Healthcare

In previous chapters, we talk about false alarms and its reduction, but no study has been conducted to study the effect of false alarms. In this chapter, we extended our research further in studying the impact of false alarm on clinician and decision-making. Moreover, we develop a discrete event simulation model to test the impact. Simulation has been a beneficial tool to conduct virtual experiments (Winsberg, 2003). In general, modeling is a popular tool to support decision-making. There are various techniques used in healthcare modeling such as Markov modeling (Bauerle et al., 2000), Monte-Carlo simulation (Sebille & Valleron, 1997), discrete event simulation, and many more.

The most extensively used simulation approach in healthcare is discrete-event simulation (DES) method. Jun, Jacobson, and Swisher (1999) review the literature regarding applications of DES modeling to healthcare clinics. Fone et al. (2003) perform an extensive review on the use of simulation in healthcare. Sobolev, Sanchez, and Vasilakis (2011) analyze the use of simulation for modeling patient flow. Harper and Shahani (2002) presented the various types of patient flows when simulating bed occupancies and patient rejection rates. Shahani, Ridley, and Nielsen (2008) developed a simulation model for a critical care unit to implement changes in bed numbers, patient length of stay, discharges in order to explore their effects on bed occupancy and refused admissions. Investigating the flow of patients (Caro, 2005; R. Davies & Davies, 1994; Sobolev et al., 2011), studying healthcare workflows (Sarnikar, 2010), and resource allocation (Steins & Walther, 2013) are most common examples of use of discrete event simulation in healthcare.

6.2 Simulation Research Model

In this section, we present our research model for evaluating the impact of false alarms on patient safety and clinician workload. Specifically, we extend the approach proposed by Gupta , Sharda , Greve, and Kamath (2005) that includes discrete event simulation for modeling email interruptions based on email policy, task complexity, and workload level in the workplace to the context of alarm interruptions in ICU. An overview of our proposed model is shown in Figure 49. The simulation model is designed to study different alarm policies in varying clinical contexts and study its effect on patient safety and various performance variables in clinicians.

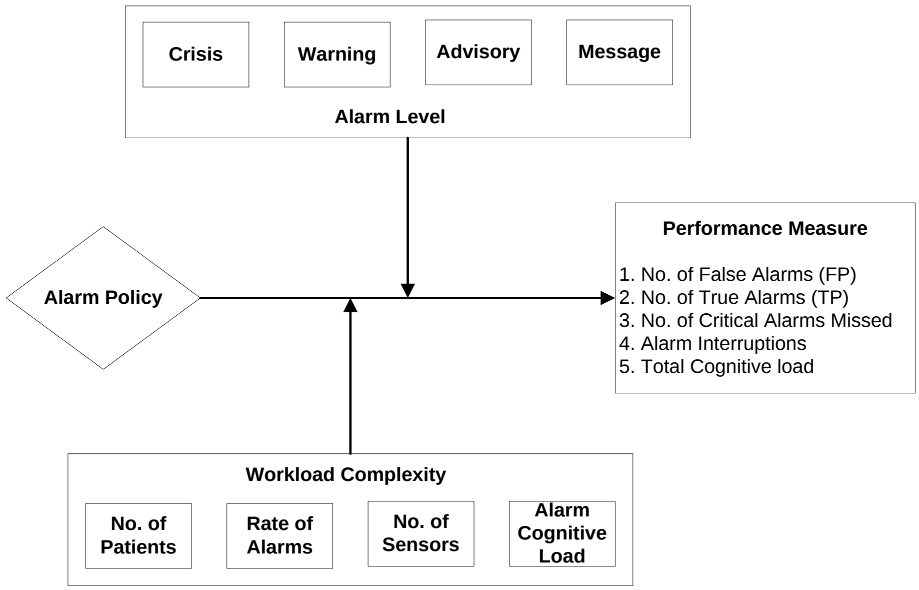

Figure 49: Simulation Research Model

6.2.1 Research Model Component

The research model consists of four major components:

- Alarm Levels: Alarm levels are categorized as crisis, warning, advisory, and message alarms (Graham & Cvach, 2010). Crisis alarms are most serious alarms such as Asystole (a state of no cardiac electrical activity i.e. flat line), Extreme Tachycardia (heart rate is dangerously high, typically above 200 beats per minute). Warning alarms are to alert the clinicians that the condition is likely to occur and clinicians should take preventive actions such as Tachycardia (heart rate is faster than normal range), Bradycardia (heart rate is slower than normal range). Advisory alarms are meant to advise the condition such as Low Pulse Oximetry (low oxygen level in blood), Premature Ventricular Contractions (PVC - abnormal heartbeats from the ventricles of the heart). Message alarms are common notification to clinician such as atrial fibrillation (rapid irregular heartbeat).

- Workload Complexity: Workload complexity can vary from low, medium to high. Workload complexity is directly proportional to the number of patients in ICU, number of sensors attached to the patient, rate of alarms, and increase in cognitive load because of alarm.
- Alarm Policy: Alarm policies guide the process of alarm notification including thresholds, routing, formats etc. For instance, policy outcomes guide which patients to monitor and suggest parameters to optimize the alarm systems that can reduce false alarms. In this paper, we explore the alarm policy related to role-based routing of alarms.
- Performance Measure: The performance measures are the number of true alarms in ICU, total number of false alarms, number of critical alarm missed, alarm interruptions, and total cognitive load of the clinician.

6.3 Alarm Policies

The two policies we explore in this paper are described below.

6.3.1 Policy 1: All Alarms routed to Nurse

In policy 1, all the alarms are routed to a nurse for response. When a sensor triggers the alarm and alarm notification enabled, the alarm is sent to a nurse. If nurse is available, the nurse responds to the alarm by monitoring the patient's vital signs and other physiologic parameters to determine the patient condition. If the alarm is assessed to be valid, the nurse takes appropriate patient care actions and records the alarm in documentation. If the alarm is identified as false, the nurse ignores the alarm or may switch the alarm off based on a threshold value signifying too many false alarms. A flow chart depicting this process flow in more detail is presented in Figure 50.

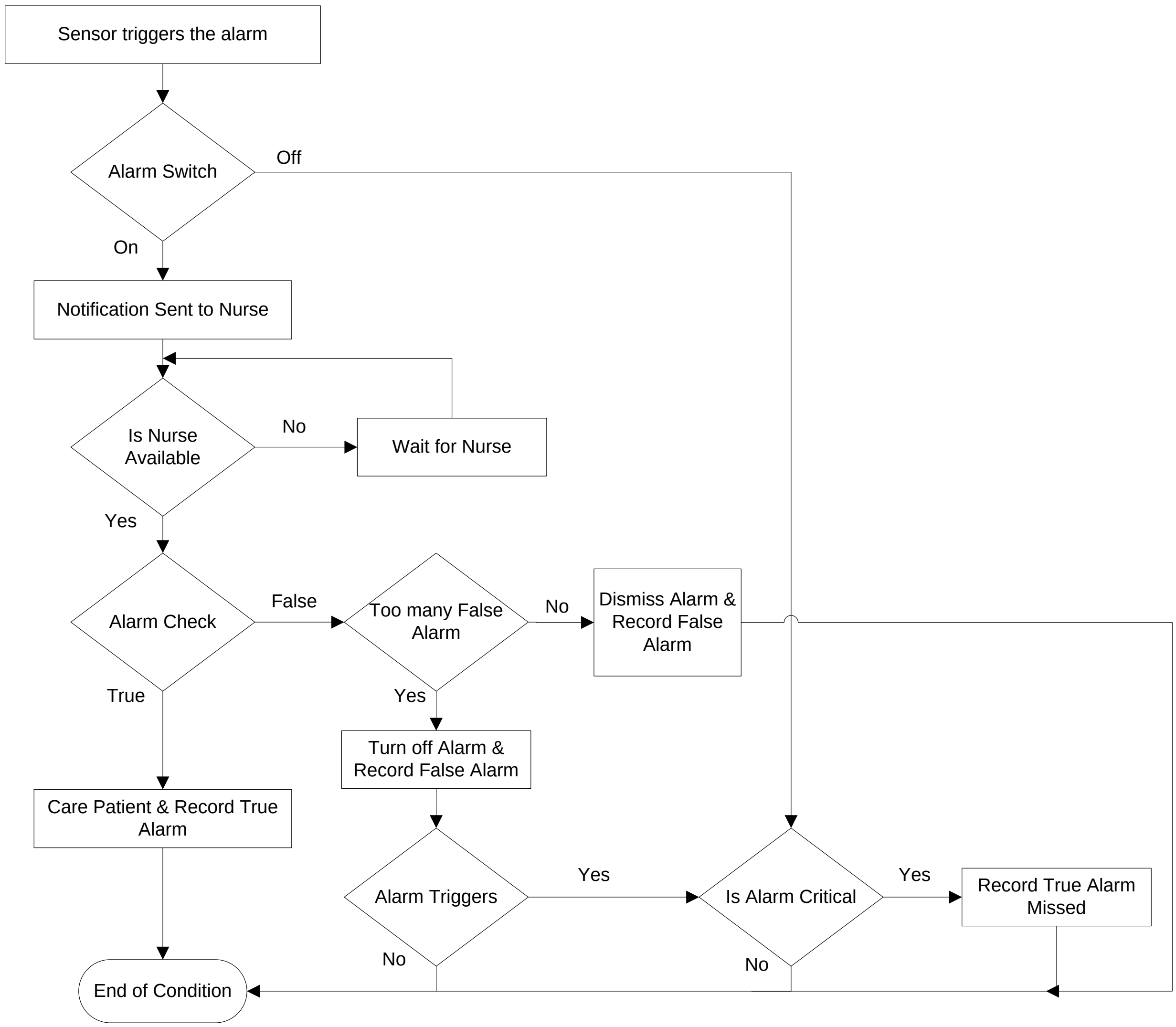

Figure 50: Process Flow for Policy 1

*Alarm Triggers -> Alarm occurrence after the alarm is switched off.

* Is Alarm critical -> Alarm of crisis level.

6.3.2 Policy 2: Role-based Routing of Alarms

In the policy 2, alarms are routed based on role of nurse and technician in ICU. The role of physician is not considered in this scenario, but we plan to model it in the future.

The policy 2 is similar to policy 1 with the addition of role of technician. When sensor triggers the alarm and alarm switch is on, alarm notification is sent to technician if the alarm is classified as a technical alarm or non-clinical alarm by the alarm notification system. A detailed overview of policy 2 is presented in Figure 51.

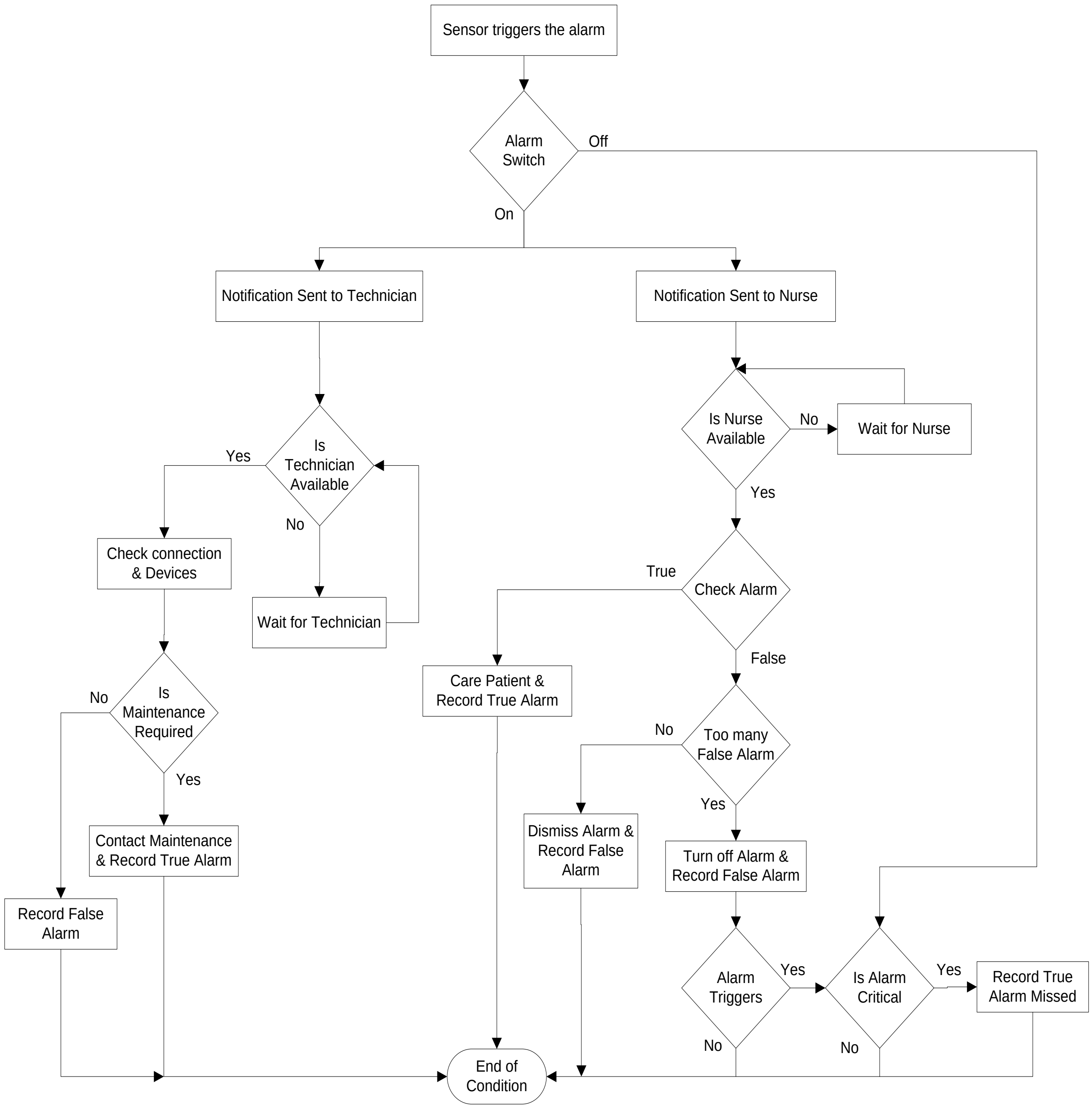

Figure 51: Process Flow for Policy 2

6.4 Simulation Model

Designing the simulation model advances our understanding of the complex nature of healthcare processes, and helps develop insights that otherwise would be expensive and time consuming. It allows testing different scenarios, and the result evaluates various strategies for effective operation of the system. In this context, we build a simulation model on two policies mentioned above to investigate the routing of alarms based on roles.

6.4.1 Simulation Software

We use JaamSim simulation software developed by Ausenco for modeling which is an open source simulation package coded in the Java programming language (King & Harrison, 2013).

6.4.2 Simulation Parameters

AAMI (2012) reported that 771 alarm conditions occur per bed per day on average in one ICU i.e. an alarm occurs on an average of every 112 seconds. The inter-arrival time for an alarm is used as exponential distribution (Ricciulli & Shacham, 1996), so the mean of 112 seconds of exponential distribution is used as inter-arrival time for alarm for simulation purpose. Lawless (1994) suggested that up to 94% of the alarms are false in ICU, so the alarm generated is set using discrete probability distribution of 0.06 and 0.94 for true and false alarms respectively. Since, the alarms are generated only for a patient, the capacity of the resource “Nurse” is assigned 1. Pergher and Silva (2014) stated that average respond time for alarms was 2 minutes and 45 seconds, so the entity delay of 165 second is used to respond to alarms after the nurse is available. Upon available, nurse monitors the vital sign, examines different parameters and determines the condition of a patient, and identifies the true alarm, takes care of the patient and then documents it. Tang, Mazabob, Weavind, Thomas, and Johnsin (2006) suggested that nurses spend 46% of time monitoring and caring patient and 30% of time documenting it in 6 hours shift when workload for each nurse was 35 patients. So, the time spent on caring a patient was used 4.73 minutes and for documenting was 3.08 minutes. The processing time to care a patient in simulation model is used 283.8 seconds and the nurse is released in 184.8 seconds after documentation. If the nurse identifies the alarm is false, it goes through a counter that keeps the track of false alarms. AAMI (2011) stated that alarm fatigue is when a nurse is overwhelmed with 350 alarm conditions per patient per day, or 0.004 false alarms per second. In our model, when there are too many false alarms, i.e. the rate for false alarm goes higher than 0.004, then the nurse is overwhelmed and alarm is switched off, and the alarms generated subsequently are recorded as missed alarms. After switching it off, the rate of false alarm starts decreasing, as the processing rate goes below then 0.004, the alarm is switched back on again. A serious problem may occur in patient’s health when true alarm is missed so, the model also captures how many true alarms are missed when the alarm is switched off.

The role of technician is added in the scenario in the Policy 2. When the alarm is switched on, the alarms are distributed based on roles of nurse and technician. Konkani, Oakley, and Bauld (2012) addressed that 17.5% of alarms are due to technical problems. So, we use the discrete probability distribution of 0.175 for technician alarms and 0.825 for alarms to nurse. Since, the alarms are generated only for a patient, the capacity of the resource “Technician” is also assigned 1. We use the same average time for nurse and clinician to respond to alarms i.e. 2 minutes and 45 seconds, so the entity delay of 165 seconds is used. Upon available, technician checks the connection problem, medical equipment and instrument. The processing time of 3 minutes is used for the scenario. If the maintenance is required, technician contacts the maintenance department. We used 195 seconds as time to contact technical support and report a problem and then release the technician for other work.

6.4.3 Simulation Analysis and Recommendation

Table 79 illustrates that nurse administered 348 false alarms in Policy 1 and 98 false alarms in Policy 2. It implies that implementing routing of alarms based on role eases the workload on the nurse and helps to reduce alarm fatigue. The total time spent on false alarms in Policy 1 is 5.76 hours, and 1.632 hours in Policy 2. This allows the nurse to spend adequate time in caring for a patient when role-based routing of alarms is implemented. The other significant measure is number of critical alarm missed. 18 critical alarms are missed in Policy 1 compared to none in Policy 2. It increases patient safety, which is the ultimate goal of setting up alarms. But at the same time, we also identified that there are numerous alarms waiting in queue to be responded by nurse in Policy 2 as the simulation is run only for 24 hours with a nurse at time.

Based on the simulation results, it suggests that the number of critical alarms missed by nurse in Policy 1 is significantly reduced when role-based routing of alarms is implemented. It also demonstrates that in role-based routing, nurse assessed more number of true alarms. It implies that such an approach would increase patient safety and save more lives which is the major goal in ICU. Moreover, it may also attract future researchers to develop such alarming device based on roles. Hence, we recommend role-based routing of alarms policies where nurse perform much better in their roles.

Table 79: Comparison of Policy 1 and Policy 2

Measure	Policy 1	Policy 2	
	Nurse	Nurse	Technician
Total Number of Alarms	769	769	
Alarms State Time	24 hours	24 hours	
Number of Critical Alarm Missed	17	10	
Number of True Positive Alarms Processed	17	20	4
Number of False Positive Alarms Processed	348	347	88
Number of Unprocessed False Positive Alarms	387	300	

Chapter 7

Conclusion

7.1 Summary

In this thesis, we studied the false alarms, basically bradycardia (Brady), tachycardia (Tachy), and ventricular tachycardia (VTACH) occurred in ICU that was retrieved from MIMIC database. We examined time ranges from 30 minutes time window up to two hours with different data transformation technique such as mean, median, standard deviation, and Discrete Fourier transform (DFT) in regards with various computing algorithms and feature sets to achieve the goal to reduce false alarm suppression rates and retaining the true alarm suppression rates.

Data of 90 minutes with median and standard deviation with Random Forest resulted in better false alarm suppression rates with high true alarm suppression rate as well. However, ensemble approach such as stacking, and voting was employed to improve the alarm suppression rates.

Table 80: Result Summary with Precision, Recall & Alarm Suppression Rates

Alarm Data Set	**Data Transformation**	**Feature Selection**	**Ensemble Approach**	**Precision (%)**	**Recall (%)**	**S - Rate TA (%)**	**S- Rate FA (%)**
Brady	Standard Deviation	None	Voting (IBk & KStar)	93.3	93.4	2.38	81.88
Tachy	Median	None	Stacking (Base: IBK, J48, Random Forest, Kstar & Meta: J48)	93.6	93.5	1.33	80.7
Vtach	Median	Information Gain	Voting (IBK, Random Forest & KStar)	82.4	82.4	15.87	80.19

Table 80 lists the most successful combination of data transformation, feature selection, and ensemble approach for three different alarm data sets with respect to precision and recall.

Furthermore, this thesis also explored the role-based routing processes involved in responding to alarms in ICU. We developed a simulation model that helps to better comprehend the effect of alarm policies on patient safety. The result suggests that the critical alarm missed by nurse is significantly reduced when role-based routing of alarms is implemented. The model implies that such an approach would increase patient safety.

7.2 Contributions

Our method, the data fusion-based approach looked across different time domain with various data transformation technique and the false alarm rate for bradycardia, tachycardia, and ventricular tachycardia was minimized through the comparative analysis with multiple feature sets and algorithms. Furthermore, the ensemble approach was also studied in order to see if the use of multiple learning algorithms may have better performance than the single learning algorithms, and certainly it does in our case. We believe it is a best way to study as it explores multiple dimensions.

We not only developed a data fusion method to minimize the rate of false alarms in ICU, we also examine the effect of false alarms in ICU. Moreover, we developed a simulation model to investigate the impact of false alarm on clinician workload and patient safety. We investigated two alarm policies 1) all alarms are routed to nurse and 2) role-based routing of alarms. When role-based routing of alarm policy was implemented, the critical alarm missed was significantly less, and nurses assess more true alarms. These findings will certainly increases patient safety that is the ultimate goal in ICU.

7.3 Limitations and Future Work

Using only a subset of the data to include the same number of alarms in every set solves the distribution problem, but also limits the overall data size. The accessible data obtained from MIMIC database has further significances: all alarm labels were obtained manually and even though they were declared gold standard by Aboukhalil et al. (2008), some labels are arguable as illustrated by the preliminary results of an online survey (Baumgartner, Rödel, Schreiber, et al., 2012). Moreover, the existing data was a mix of false and true alarms, but were not comprised of missed alarm events. Including such events is a part of future work.

The data used for the simulation model was extracted from various time and motion studies and research reports. This is a limitation of the current work, but in future, future plan is to collect estimates of all parameters based on a single context. We intend to add the role of physician as well and plan to enhance the process by modeling alarm levels, and workload complexity in future for the role-based routing of alarms.

Over all, the focus of this work was not to find an optimal method for alarm classification, but to illustrate the applicability of data mining to the problem of false alarm rate suppression. A larger dataset with equally distributed alarm types is desirable to foster the results. However, one should be aware that patient safety is the primary goal in ICU monitoring and that an alarm classification system as presented suppresses true alarms.

References

AAMI. (2011). A Siren Call for Action: PRIORITY ISSUES FROM THE MEDICAL DEVICE ALARMS SUMMIT *2011 Summit Publications: Clinical Alarms*. Arlington, VA: Association for the Advancement of Medical Instrumentation.

AAMI. (2012). Using Data to Drive Alarm System Improvement Efforts - The John Hopkins Hospital Experience. Baltimore, MD: AAMI Foundation HTSI.

Aboukhalil, A., Nielsen, L., Saeed, M., Mark, R. G., & Clifford, G. D. (2008). Reducing false alarm rates for critical arrhythmias using the arterial blood pressure waveform. *Journal of Biomedical Informatics, 41*, 442-451.

AHA. (2012). Understanding Blood Pressure Readings. http://www.heart.org/HEARTORG/Conditions/HighBloodPressure/AboutHighBloodPressure/Understanding-Blood-Pressure-Readings_UCM_301764_Article.jsp

Allen, J., & Murray, A. (1996). Assessing ECG signal quality on a coronary care unit. *Physiological Measurement, 17*, 249-258.

American College of Clinical Engineering. (2006). Impact of Clinical Alarms on Patient Safety (pp. 1-19). Plymouth Meeting, PA, USA: ACCE Healthcare Technology Foundation.

Arrue, B. C., Ollero, A., & Martinez, D. J. R. (2000). An Intelligent System for False Alarm Reduction in Infrared Forest-Fire Detection. *IEEE Intelligent Systems and their Applications*, *15*(3), 64-73.

Baher, A., & Valderrabano, M. (2013). Management of Ventricular Tachycardia in Heart Failure. *Methodist DeBakey Cardiovascular Journal, 9*(1), 20-25.

Bauerle, R., Rucker, A., Schmandra, T. C., Holzer, K., Encke, A., & Hanisch, E. (2000). Markov cohort simulation study reveals evidence for sex-based risk difference in intensive care unit patients. *The American Journal of Surgery, 179*, 207-211.

Baumgartner, B., Ro¨del, K., & Knoll, A. (2012). *A Data Mining Approach to Reduce the False Alarm Rate of Patient Monitors*. Paper presented at the 34th Annual International Conference of the IEEE EMBS, San Diego, California, USA.

Baumgartner, B., Ro¨del, K., Schreiber, U., & Knoll, A. (2012). A Web-Based Survey for Expert Review of Monitor Alarms. *Computing in Cardiology, 39*, 209-212.

Behar, J., Oster, J., Li, Q., & Clifford, G. D. (2013). ECG Signal Quality During Arrhythmia and Its Application to False Alarm Reduction. *IEEE Transaction on Biomedical Engineering, 60*(6), 1660-1666.

Ben-Gal, I. (2007). Bayesian Networks. In F. Ruggeri, F. Faltin & R. Kenett (Eds.), *Encyclopedia of Statistics in Quality & Reliability*: Wiley & Son.

Blum, J. M., Kruger, G. H., Sanders, K. L., Gutierrez, J., & Rosenberg, A. L. (2009). Specificity Improvement for Network Distributed Physiologic Alarms Based on a Simple Deterministic Reactive Intelligent Agent in the Critical Care Environment. *Journal of Clinical Monitoring and Computing, 23*(1), 21-30.

Borges, G. d. M., & Brusamarello, V. (2014). *Bayesian Fusion of Multiple Sensors for Reliable Heart Rate Detection.* Paper presented at the Proceedings of 2014 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Montevideo.

Borowski, M., Schettlinger, K., & Gather, U. (2009). Multivariate real time signal processing by a robust adaptive regression filter. *Communications in Statistics—Simulation and Computation, 38*(2), 426-440.

Borowski, M., Siebig, S., Wrede, C., & Imhoff, M. (2011). Reducing False Alarms of Intensive Care Online-Monitoring Systems: An Evaluation of Two Signal Extraction Algorithms. *Computational and Mathematical Methods in Medicine, 2011*, 11. doi: 10.1155/2011/143480

Breiman, L. (1996). Bagging predictors. *Machine Learning, 24*(2), 123-140.

Breznitz, S. (1984). Cry wolf: The Physchology of False Alarm (pp. 280). Hillsdale, NJ: Lawrence Erlbaum Associates.

Caro, J. J. (2005). Pharmacoeconomic analyses using discrete event simulation. *PharmacoEconomics, 23*, 323-332.

Chambrin, M. C. (2001). Alarms in the intensive care unit: How can the number of false alarms be reduced? *Critical Care, 5*, 184-188.

Chambrin, M. C., Ravaux, P., Calvelo-Aros, D., Jaborska, A., Chopin, C., & Boniface, B. (1999). Multicentric study of monitoring alarms in the adult intensive care unit (icu): a descriptive analysis. *Intensive Care Medicine, 25*(12), 1360-1366.

Chan, P., Fan, W., Prodromidis, A., & Stolfo, S. (1999). Distributed Data Mining in Credit Card Fraud Detection. *IEEE Intelligent Systems, 14*, 67-74.

Charbonnier, S., & Gentil, S. (2007). A trend-based alarm system to improve patient monitoring in intensive care units. *Control Engineering Practice, 15*(9), 1039–1050.

Chen, H.-H., Huang, P., Chen, C.-H., Chen, H.-S., & Luh, J.-J. (2006). *Sensor data fusion for timely emergency alarm*. Paper presented at the 8th International Conference on e-Health Networking, Applications and Services, New Delhi, India.

Choi, S.-D., Akin, B., Kwak, S., & Toliyat, H. A. (2014). A Compact Error Management Algorithm to Minimize False-Alarm Rate of Motor/Generator Faults in (Hybrid) Electric Vehicles. *IEEE Journal of Emerging and Selected Topics in Power Electronics, 2*(3), 618-625.

Cretikos, M. A., Bellomo, R., Hillman, K., Chen, J., Finfer, S., & Flabouris, A. (2008). Respiratory rate: the neglected vital sign. *The Medical Journal of Australia, 188*(11), 657-659.

Cvach, M. (2012). Monitor Alarm Fatigue An Integrative Review. *Biomedical Instrumentation & Technology, 46*(4), 268-277.

Davies, P. L., Fried, R., & Gather, U. (2003). Robust signal extraction for on-line monitoring data. *Journal of Statistical Planning and Inference, 122*, 65-78.

Davies, R., & Davies, H. (1994). Modelling patient flows and resource provision in health systems. *Omega (Oxford), 22*, 123-131.

Dietterich, T. G. (1997). Machine Learning research: Four current directions. *AI Magazine, 18*(4), 97-136.

Drew, B. J., Califf, R. M., & Funk, M. (2004). Practice standards for electrocardiographic monitoring in hospital settings. *Circulation, 110*(17), 2721-2746.

Drew, B. J., Harris, P., Zgre-Hemsey, J. K., Mammone, T., Schindler, D., Salas-Boni, R., . . . Hu, X. (2014). Insights into the problem of alarm fatigue with physiologic monitor devices: A comprehensive observational study of consecutive intensive care unit patients. *PLoS ONE, 9*(10). doi: 10.1371/journal.pone.0110274

ECRI. (2011). Top 10 Health Technology Hazzards for 2012 *Health Devices* (pp. 1-16): ECRI Institute.

ECRI. (2012). Top 10 Health Technology Hazzards for 2013 *Health Devices* (pp. 1-23): ECRI Institute.

ECRI. (2013). Top 10 Health Technology Hazzards for 2014 *Health Devices* (pp. 1-13): ECRI Institute.

ECRI. (2014). Top 10 Health Technology Hazzards for 2015. *Health Devices*, 1-31.

Edwards. (2014). Normal Hemodynamic Parameters – Adult. *Normal Hemodynamic Parameters and Laboratory Values.*

Eerika¨inen, L. M., Vanschoren, J., Rooijakkers, M. J., Vullings, R., & Aarts, R. M. (2015). *Decreasing the False Alarm Rate of Arrhythmias in Intensive Care Using a Machine Learning Approach*. Paper presented at the Computing in Cardiology.

El-Atta, A. H. A., Moussa, M. I., & Hassanien, A. E. (2014). *Predicting Biological Activity of 2,4,6-trisubstituted 1,3,5-trazines Using Random Forest*. Paper presented at the Fifth International Conference on Innovations in Bio-Inspired Computing and Applications, Ostrava, Czech Republic.

Factor, M., Gelernter, D. H., Kolb, C. E., Miller, P. L., & Sittig, D. F. (1991). Real-time data fusion in the intensive care unit. *Computer, 24*(11), 45-54. doi: 10.1109/2.116850

Feldman, J. M., Ebrahim, M. H., & Bar-Kana, I. (1997). Robust sensor fusion improves heart rate estimation: clinical evaluation. *Journal of Clinical Monitoring and Computing, 13*, 379-384.

Fone, D., Hollinghurst, S., Temple, M., Round, A., Lester, N., Weightman, A., . . . Palmer, S. (2003). Systematic review of the use and value of computer simulation modelling in population health and health care delivery. *Journal of Public Health, 25*(4), 325-335.

Go¨rges, M., Markewitz, B. A., & Westenskow, D. R. (2009). Improving alarm performance in the medical intensive care unit using delays and clinical context. *Anesthesia & Analgesia, 108*(5), 1546-1552.

Graham, K. C., & Cvach, M. (2010). Monitor Alarm Fatigue: Standardizing Use of Physiological Monitors and Decreasing Nusiance Alarm. *American Journal of Critical Care, 19*(1), 28-37.

Gross, B., Dahl, D., & Nielsen, L. (2011). Physiologic monitoring alarm load on medical/surgical floors of a community hospital. *Biomedical Instrumentation & Technology*, 29-36.

Grünig, E., Bart Janssen, Mereles, D., Barth, U., Borst, M. M., Vogt, I. R., . . . Kübler, W. (2000). Clinical Investigation and Reports: Abnormal Pulmonary Artery Pressure Response in Asymptomatic Carriers of Primary Pulmonary Hypertension Gene. *Circulation, 102*, 1145-1150. doi: 10.1161/01.CIR.102.10.1145

Gupta, A. (2007). *Understanding The Rhythms of Email Processing Strategies in a Network of Knowledge Workers.* (Doctor of Philosophy), Oklahoma State University.

Gupta , A., Sharda , R., Greve, R., & Kamath, M. C., Mohanraj. (2005). *How often should we check our email? Balancing interruptions and quick response times*. Paper presented at the BigXII+ MIS Research Symposium, Norman, Oklahoma.

Hall, D. L., & Llinas, J. (1997). An Introduction to Multisensor Data Fusion. *Proceedings of the IEEE, 85*(1), 6-23.

Hall, M. A. (1999). *Correlation-based Feature Subset Selection for Machine Learning.* The University of Waikato, Hamilton, New Zealand.

Hann, A., Tarassenko, L., Patterson, A., Barber, V., & Young, D. (2006). *The Value of Data Fusion for Predicting Alarms in Critical Care*. Paper presented at the 3rd International Conference On Advances in Medical, Signal and Information Processing, Glasgow, UK.

Harper, P., & Shahani, A. (2002). Modelling for the planning and management of bed capacities in hospitals. *Journal of the Operational Research Society, 53*(1), 11-18.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information System Research. *MIS Quarterly, 28*(1), 75-105.

Ho, T. K. (1995). *Random Decision Forests.* Paper presented at the 3rd International Conference on Document Analysis and Recognition, Montreal, QC.

Ho, T. K. (1998). The Random Subspace Method for Constructing Decision Forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20*(8), 832-844.

Imhoff, M., Kuhls, S., & Gather, U. (2009). Smart alarms from medical devices in the OR and ICU. *Best Practice & Research Clinical Anesthesiology, 23*, 39-50.

Jakob, S., Korhonen, I., Ruokonen, E., Virtanen, T., Kogan, A., & Takala, J. (2000). Detection of artifacts in monitored trends in intensive care. *Computer Methods and Programs in Biomedicine, 63*(3), 203-209.

Jazzar, M., & Jantan, A. B. (2008). *Using fuzzy cognitive maps to reduce false alerts in som-based intrusion detection sensors*. Paper presented at the Second Asia International Conference on Modelling & Simulation, Kuala Lumpur.

Jun, J., Jacobson, S., & Swisher, J. (1999). Application of discrete-event simulation in health care clinics: A survey. *The Journal of the Operational Research Society, 50*, 109-123.

Kacmarek, R., & Tobin, M. J. (1998). Alarms *Principle and Practice of Intensive Care Monitoring* (pp. 133-139). New York: McGraw-Hill, Inc.

Karegowda, A. G., Manjunath, A. S., & Jayaram, M. A. (2010). Comparative Study of Attribute Selection Using Gain Ratio and Correlation based Feature Selection. *International Journal of Information Technology and Knowledge Management, 2*(2), 271-277.

Keller, J. P. (2012). Clinical alarm hazards: a top ten health technology safety concern. *Journal of Electrocardiology, 45*(6), 588-591. doi: 10.1016/j.jelectrocard.2012.08.050

Kennedy, R. R. (1995). A modified Trigg's tracking variable as an 'advisory' alarm during anaesthesia. *International Journal of Clinical Monitoring and Computing, 12*(4), 197-204.

Kim, D., Shin, H., Song, Y. S., & Kim, J. (2012). Synergistic effect of different levels of genomic data for cancer clinical outcome prediction. *Journal of Biomedical Informatics, 45*(6), 1191-1198. doi: 10.1016/j.jbi.2012.07.008

Kim, M., & Kim, T. (2002). *A Neural Classifier with Fraud Density Map for Effective Credit Card Fraud Detection*. Paper presented at the IDEAL.

King, D. H., & Harrison, H. S. (2013). *Open-Source Simulation Software "JaamSim"*. Paper presented at the 2013 Winter Simulation Conference.

Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence in Medicine, 97*(1), 273-324.

Konkani, A., Oakley, B., & Bauld, T. J. (2012). Reducing Hopsita; Noise: A Review of Medical Device Alarm Management. *Biomedical Instrumentation & Technology, 46*, 6478-6487. doi: 10.2345/0899-8205-46.6.478

Kumar, V., & Rathee, N. (2011). Knowledge discovery from database Using an integration of clustering and classification. *International Journal of Advanced Computer Science and Applications, 2*(3), 29-33.

Kuncheva, L., & Whitaker, C. (2003). Measures of diversity in classifier ensembles. *Machine Learning, 51*, 181-207.

Laskowski, E. (2012). What's a normal resting heart rate? *Health Lifestyle Fitness*. http://www.mayoclinic.org/heart-rate/expert-answers/faq-20057979

Lawless, S. T. (1994). Crying Wolf: False alarms in a pediatric intensive care unit. *Critical Care Medicine, 22*(6), 981-985.

Li, Q., & Clifford, G. (2012). Signal quality and data fusion for false alarm reduction in the intensive care unit. *Journal of Electrocardiology, 45*(6), 596-603.

Lidco. (2014). Normal Hemodynamic Parameters. from http://www.lidco.com/clinical/hemodynamic.php

Maes, S., Tuyls, K., Vanschoenwinkel, B., & Manderick, B. (2002). *Credit Card Fraud Detection using Bayesian and Neural Networks.* Paper presented at the 1st International NAISO Congress on Neuro Fuzzy Technologies.

Mäkivirta, A., Koski, E., Kari, A., & Sukuvaara, T. (1991). The median filter as a preprocessor for a patient monitor limit alarm system in intensive care. *Computer Methods and Programs in Biomedicine, 34*, 139-144.

MayoClinic. (2013). Hypoxemia Low blood Oxygen.

MedlinePlus. (2012). Heart palpitations. http://www.nlm.nih.gov/medlineplus/ency/article/003081.htm

Merzbacher, M., & Gable, T. (2010). *Applying Data Minining to False Alarm Reduction in Aviation Explosives Detection System*. Paper presented at the IEEE International Carnahan Conference on Security Technology (ICCST), San Jose, CA.

Musen, M. A. (1997). Modelling of Decision Support in Handbook of Medical Informatics. In J. H. V. Bemmel & M. A. Musen (Eds.), (pp. 1-26). Springer-Verlag.

Nandhini, R., & Subhasini, S. (2013). Classification of ECG Images Using Probabilistic Neural Network Based on Statistical Feature Analysis. In S. Mohan & S. Suresh Kumar (Eds.), *Proceedings of the Fourth International Conference on Signal and Image Processing 2012 (ICSIP 2012)* (Vol. 2): Springer India.

Otero, A., Félix, P., Barro, S., & Palacios, F. (2009). Addressing the flaws of current critical alarms: a fuzzy constraint satisfaction approach. *Artificial Intelligence in Medicine, 47*(3), 219-238. doi: 10.1016/j.artmed.2009.08.002

Paxson, V. (1999). Bro: A System for Detecting Network Intruders in Real-Time. *Computer Networks, 31*(23-24), 2435-2463.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems, 24*(3), 45-77.

Pergher, A. K., & Silva, R. C. (2014). Stimulus-response time to invasive blood pressure alarms: implications for the safety of critical-care patients. *Rev Gaucha Enferm, 35*(2), 135-141.

Phillips, J. (2006). Clinical alarms: complexity and common sense. *Critical Care Nursing Clinics of North America*, *18*(2), 145-156.

Phua, C., Alahakoon, D., & Lee, V. (2004). Minority Report in Fraud Detection: Classification of Skewed Data. *SIGKDD Explorations*, *6*(1), 50-59.

Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). A Comprehensive Survey of Data Mining-based Fraud Detection Research. *Arxiv preprint arXiv:1009.6119*, 1-14. doi: 10.1016/j.chb.2012.01.002

Pietraszek, T., & Tanner, A. (2005). Data mining and machine learning—Towards reducing false positives in intrusion detection. *Information Security Technical Report, 10*(3), 169-183.

Polikar, R., Topalis, A., Parikh, D., Green, D., Frymiare, J., Kounios, J., & Clark, C. M. (2008). An ensemble based data fusion approach for early diagnosis of Alzheimer’s disease. *Information Fusion 9, 9*(1), 83-95. doi: 10.1016/j.inffus.2006.09.003

Quinlan, R. (1993). *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann Publishers.

Ricciulli, L., & Shacham, N. (1996). *Modeling Correlated Alarms in Network Management Systems.* Paper presented at the Western SImulation Multiconference.

Rokach, L. (2010). Ensemble-based classifiers. *Artificial Intelligence Review*, *33*(1-2), 1-39. doi: 10.1007/s10462-009-9124-7

Roychoudhury, S., Ghalwash, M. F., & Obradovic, Z. (2015). *False Alarm Suppression in Early Prediction of Cardiac Arrhythmia.* Paper presented at the IEEE 15th International Conference on Bioinformatics and Bioengineering (BIBE), Belgrade.

Saeed, M., Villarroel, M., Reisner, A., Clifford, G., Lehman, L.-W., Moody, G., . . . Mark, R. (2011). Multiparame- ter intelligent monitoring in intensive care ii: a public-access intensive care unit database. *Critical Care Medicine, 39*(5), 952-960.

Salas-Boni, R., Bai, Y., Harris, P. R., Drew, B. J., & Hu, X. (2014). False ventricular tachycardia alarm suppression in the ICU based on the discrete wavelet transform in the ECG signal. *Journal of Electrocardiology, 47*(6), 775-780. doi: 10.1016/j.jelectrocard.2014.07.016

Salimi-Khorshidi, G., Douaud, G., Beckmann, C. F., Glasser, M. F., Griffanti, L., & Smith, S. M. (2014). Automatic Denoising of Functional MRI Data: Combining Independent

Component Analysis and Hierarchical Fusion of Classifiers. *NeuroImage, 90*, 449-468. doi: 10.1016/j.neuroimage.2013.11.046

Sarnikar, S. (2010). *Using Simulation and DEA for Optimizing Post-EMR Workflows*. Paper presented at the 5th INFORMS Workshop on Data Mining and Health Informatics, Austin, TX.

Sayadi, O., & Shamsollahi, M. B. (2011). Life-Threatening Arrhythmia Verification in ICU Patients Using the Joint Cardiovascular Dynamical Model and a Bayesian Filter. *IEEE Transactions on Biomedical Engineering, 58*(10), 2748-2757. doi: 10.1109/TBME.2010.2093898

Scalzo, F., Liebeskind, D., & Hu, X. (2013). Reducing False Intracranial Pressure Alarms Using Morphological Waveform Features. *IEEE Transactions on Biomedical Engineering, 60*(1), 235-239.

Schettlinger, K., Fried, R., & Gather, U. (2010). Real-time signal processing by adaptive repeated median filters. *International Journal of Adaptive Control and Signal Processing, 24*(5), 346–362.

Schoenberg, R., Sands, D. Z., & Safran, C. (1999). *Making ICU alarms meaningful: a comparison of traditional vs. trend-based algorithms.* Paper presented at the American Medical Informatics Association Symposium, Washington D.C.

Sebille, V., & Valleron, A. J. (1997). A computer simulation model for the spread of noso- comial infections caused by multidrug-resistant pathogens. *Computers and Biomedical Research, 30*, 307-322.

Shahani, A., Ridley, S., & Nielsen, M. (2008). Modelling patient flows as an aid to decision making for critical care capacities and organisation. *Anesthesia, 63*(10), 1074-1080.

Shapire, R., Freund, Y., Bartlett, P., & Lee, W. (1998). Boost- ing the margin: A new explanation for the effec- tiveness of voting methods. *Annals of Statistics, 26*(5), 1651-1686.

Sherwood, L. (2005). *Fundamentals of Physiology: A Human Perspective* (3rd ed.): Cengage Learning.

Shrestha, S., Sarnikar, S., & Timsina, P. (2013). *Data Integration in CDSS for Alerts & Reminders: A Review*. Paper presented at the Eighth Midwest Association of Information System (MWAIS), Normal, IL.

Siebig, S., Kuhls, S., Imhoff, M., Gather, U., Schölmerich, J., & Wrede, C. E. (2010). Intensive care unit alarms--how many do we need? *Critical Care Medicine*, *38*(2), 451-456. doi: 10.1097/CCM.0b013e3181cb0888

Siebig, S., Kuhls, S., Imhoff, M., Langgartner, J., Reng, M., Schlmerich, J., . . . Wrede, C. E. (2010). Collection of annotated data in a clinical validation study for alarm algorithms in intensive care: A methodologic framework. *Journal of Critical Care*, *25*(1), 128-135.

Sigletos, G., Paliouras, G., Spyropoulos, C. D., & Hatzopoulos, M. (2005). Combining Information Extraction Systems Using Voting and Stacked Generalization. *Journal of Machine Learning Research, 6*, 1751-1782.

Sobolev, B., Sanchez, V., & Vasilakis, C. (2011). Systematic Review of the Use of Computer Simulation Modeling of Patient Flow in Surgical Care. *Journal of Medical Systems, 35*(1), 1-16.

Sollich, P., & Krogh, A. (1996). Learning with ensembles: How overfitting can be useful. *Advances in Neural Information Processing Systems, 8*, 190-196.

Sommer, R., & Paxson, V. (2003). *Enhancing byte-level network intrusion detection signatures with context*. Paper presented at the 10th ACM conference on Computer and Communication Security, Washington DC.

Steins, K., & Walther, S. M. (2013). A generic simulation model for planning critical care resource requirements. *Anaeshesia, 68*, 1148-1155. doi: 10.1111/anae.12408

Tan, A. C., & Gilbert, D. (2003). Ensemble machine learning on gene expression data for cancer classification. *Applied Bioinformatics, 2*(3 Suppl), S75-83.

Tang, Z., Mazabob, J., Weavind, L., Thomas, E., & Johnsin, T. R. (2006). *A Time-Motion Study of Registered Nurses' Workflow in Intensive Care Unit Remote Monitoring*. Paper presented at the 2006 AMIA Symposium Washington D.C.

Taniguchi, M., Haft, M., Hollmen, J., & Tresp, V. (1998). *Fraud Detection in Communication Networks using Neural and Probabilistic Methods*. Paper presented at the IEEE International Conference in Acoustics, Speech and Signal Processing.

TJC. (2013). Sentinel Event Alert (pp. 1-3): The Joint Comission.

Tsien, C. L., & Fackler, J. C. (1997). Poor prognosis for existing monitors in the intensive care unit. *Critical Care Medicine, 25*, 614-619.

Valdes, A., & Skinner, K. (2001). Probabilistic alert correlation *Recent Advances in Intrusion Detection (RAID2001)* (Vol. 2212, pp. 54-68): Springer-Verlag.

Veerappan, S., Rosen, H., Craelius, W., Curcie, D., Hiatt, M., & Hegyi, T. (2000). Spectral Analysis of Heart Rate Variability in Premature Infants with Feeding Bradycardia. *Pediatric Research, 46*, 659–662. doi: 10.1203/00006450-200005000-00017

Vyas, N., Farringdon, J., Andre, D., & Stivoric, J. (2011). *Machine Learning and Sensor Fusion for Estimating Continuous Energy Expenditure.* Paper presented at the Twenty-Third Innovative Applications of Artificial Intelligence Conference, San Francisco, CA.

Wallis, L. (2010). Alarm Fatigue Linked to Patient's Death. *American Journal of Nursing, 110*(7), 16.

Welch, J. (2011). An evidence-based approach to reduce nuisance alarms and alarm fatigue. *Biomedical Instrumentation & Technology*, 46-52.

Whalen, D. A., Covelle, P. M., Piepenbrink, J. C., Villanova, K. L., Cuneo, C. L., & Awtry, E. H. (2014). Novel approach to cardiac alarm management on telemetry units. *The Journal of Cardiovascular Nursing, 29*(5), E13-22. doi: 10.1097/JCN.0000000000000114

Winsberg, E. (2003). Simulated experiments: methodology for a virtual world. *Philosophy of Science, 70*(1), 105-121.

Xu, X., Zhu, L., Fu, M., Sun, D., Tran, A. B., Rimba, P., . . . Bass, L. (2015). *Crying Wolf and Meaning it: Reducing False Alarms in Monitoring of Sporadic Operations through POD-Monitor.* Paper presented at the 1st International Workshop on COmplex faUlts and Failures in LargE Software Systems (COUFLESS), Firenze, Italy.

Yahalom, M., Spitz, M., Sandler, L., Heno, N., Roguin, N., & Turgeman, Y. (2013). The Significance of Bradycardia in Anorexia Nervosa. *International Journal of Angiology, 22*(2), 83-94.

Zhang, Y. (2007). *Real-Time Development of Patient-Specific Alarm Algorithms for Critical Care*. Paper presented at the 9th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EBS), Lyon, France.

Zheng, Y. (2015). Methodologies for Cross-Domain Data Fusion: An Overview. *IEEE Transactions on Big Data, 1*(1), 16-34. doi: 10.1109/TBDATA.2015.2465959

Zong, W., Moody, G. B., & Mark, R. G. (2004). Reduction of false arterial blood pressure alarms using signal quality assessment and relationships between the electrocardiogram and arterial blood pressure. *Medical and Biological Engineering and Computing, 42*(5), 698-706.