A MACHINE LEARNING APPROACH FOR ALERT BEHAVIOR RESPONSE MODELING TO MITIGATE ALERT FATIGUE IN HEALTH INFORMATION SYSTEMS

by

Jamey Allyson Martell

Submitted in partial fulfilment of the requirements
for the degree of Master of Health Informatics

at

Dalhousie University
Halifax, Nova Scotia
April, 2019

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

This research investigates novel approaches to reduce the burden of alert fatigue faced by primary care physicians using Clinical Decision Support Systems (CDSS) within EMR systems. CDSS issue a range of alerts to assist physicians in patient management with respect to clinical guidelines and institutional clinical pathways. However, the generation of alerts is usually suboptimal, and does not consider the physician's clinical context. Our approach is to understand the physician's practice to triage alert issuance, ensuring that alerts are adequately addressed by physicians without causing unnecessary alert fatigue. We utilize machine learning techniques to: cluster physicians into distinct practice groups based on their practice data, stratify the wide range of CDSS alerts based on key, defining attributes, and learn a classification based mapping between physician practice groups and alert types to develop an innovative alert issuance strategy that greatly reduces the volume of alerts presented to each physician group.

LIST OF ABBREVIATIONS USED

AE	Adverse Event
AHRQ	Agency for Healthcare Research and Quality
AI	Artificial Intelligence
ANN	Artificial Neural Network
AUC	Area Under the Curve
CDS	Clinical Decision Support
CDSS	Clinical Decision Support Systems
CPOE	Computerized Physician/Provider Order Entry
CPSNS	College of Physicians and Surgeons of Nova Scotia
CSV	Comma Separated Values
DDI	Drug-Drug Interaction
DI	Diagnostic Imaging
DT	Decision Tree
EMR	Electronic Medical Record
GUI	Graphical User Interface
HIT	Health Information Technology
HITECH	Health Information Technology for Economic and Clinical Health
INR	International normalized ratio
IOM	Institute of Medicine
MeSH	Medical Subject Headings
MIT	Massachusetts Institute of Technology
MLP	Multilayer Perceptron
NHS	National Health Service
POC	Point of Care
RCT	Randomized Control Trial
ROC	Receiver Operating Characteristic
RRC	Response Rate Category
Weka	Waikato Environment for Knowledge Analysis

ACKNOWLEDGEMENTS

I would like to extend sincere thanks to the following people without whom this work would not have been possible. To my supervisor, Dr. Raza Abidi. For his guidance, patience, support and expert advice throughout the course of this project, I will be forever grateful. To Dr. Sam Stewart, for his help with data generation, and to Patrice Roy and all of the members of the NICHE research team who provided a supportive environment in which to explore questions, techniques and to engage in rich discussion that helped shape the direction of this work.

Most importantly, I must thank my family who provided me the love, support, and time required to embark upon this endeavor. Ayla, Senna and Aidan: I wish for each of you a healthy, happy life of endless learning. With time, effort and will you can accomplish anything. To my husband, John, I will be forever indebted. Thank you today and always.

CHAPTER 1: INTRODUCTION

With the increased adoption of electronic medical records in the primary care setting, there is an opportunity to leverage Clinical Decision Support (CDS) to improve patient care delivery and associated health outcomes. Electronic, automated CDSS (Clinical Decision Support *System*) alerting is one mechanism by which physicians can receive real time, evidence-based information relating to the patient, whether it be in the form of a suggestion, reminder or an urgent care-related alert, in order to help inform care delivery [6, 7]. While CDS alerting has been associated with increased patient safety and improved patient outcomes [17, 18, 24, 25, 27, 28], significant challenges with alert delivery and alert acceptance persist resulting in the widely reported phenomenon known as 'alert fatigue' [19, 20, 27]. Alert fatigue can be defined as the process by which clinicians become desensitized to alerts, and as a result ignore or fail to respond appropriately to such warnings [16]. Essentially, alert fatigue is a direct result of higher than necessary volumes of alerts being presented to clinicians [19, 20, 27]. With studies reporting alert override rates as high as 90% [68, 70], there exists significant risk to patient safety as important, evidence-based, care related alerts are being ignored. Physicians have expressed the importance of delivering the right alert to the right provider at the right time and in the right way, noting that alert content is not always aligned with the needs of different clinician groups and expressing frustration in their inability to influence alert issuance to align with personal preferences [60]. There is a need to explore a more personalized approach to

CDSS alert delivery in order to reduce the volume of alerts delivered, thereby minimizing alert fatigue.

1.1 Research Intent and Objectives

This research aims to investigate novel approaches to reduce the burden of alert fatigue faced by primary care physicians, thereby ensuring that CDSS-generated alerts are actually acted upon in a timely manner in order to improve patient safety, and ultimately, patient care. Given the wide variety and volume of alerts being generated (with varying degrees of acuity) it is important that these alerts are presented to physicians in keeping with the dynamic of their clinical practice—i.e. alert response behavior, clinical schedule, alert acuity and patient case-mix. Essentially, our approach is to understand the physician's clinical practice in order to triage the issuance of alerts to ensure that the alerts are adequately attended to by physicians without any unnecessary fatigue due to the plethora of alerts presented. Our goal, therefore, is to investigate and develop a personalized, alert triaging mechanism to reduce alert fatigue faced by physicians. For personalization, we will pursue the following research objectives:

1) Stratification of physicians into distinct practice groups to design a group-level alert issuance strategy;

2) Classification of the wide range of CDSS alerts in terms of their source, acuity and response expectations;

3) Establishing a mapping between physician groups and alerts types; and

4) Development of a strategy to issue alerts based on physician's practice and alert response behavior in order to minimize alert fatigue.

1.2 Research Design

In order to address our research objectives, we first explore physician characteristics, related to both patient and practice attributes. This will permit the grouping of physicians into distinct clusters. We seek to leverage as much real world data as possible, coupled with attributes taken from current literature. We will then utilize well documented, machine learning-based clustering techniques to identify physician types.

Next we will explore CDSS alert classification in the literature. We will seek out a framework that could be applied to the primary care setting. From there, we develop our own framework for CDSS alert type classification. The goal is to utilize this framework for our research; but it will be developed such that it can be shared and used by others to support future research as well.

Finally, we will use our physician clusters and alert classification to generate personalized physician response data which will be used to develop a predictive model that will help identify how the various physician types (groups) respond to the various alert types set forth in our framework. This will enable us to develop an innovative strategy for alert issuance based on a physician's practice and alert response behavior. If successful, this will lead the way for future research exploring a more personalized approach to CDSS alert delivery that could offer significant opportunities to combat alert fatigue.

1.3 Contribution

This research will provide an innovative strategy for the reduction of alert fatigue based on a more personalized approach to CDSS alert issuance based on the stratification of physicians. While many CDSS related studies leverage patient characteristics with respect to alert issuance, we were unable to identify any that reported utilizing physician attributes to determine which CDSS alerts are displayed. Our approach could lead to a significant reduction in alert rejection, and in the volume of alerts displayed to a physician – alerts that we know would be ignored based on evidence used to develop our predictive models. This work will pave the way for additional work in the area of personalized CDSS alerting.

1.4 Thesis Organization

The remainder of this thesis is organized as follows: Chapter 2 presents the background and main concepts related to this research, Chapter 3 presents our research methods and design, Chapter 4 presents our classification results, Chapter 5 offers our proposed alert issuance strategy, and Chapter 6 provides a discussion on study contributions, limitations and future work.

CHAPTER 2: BACKGROUND

2.1 Overview

This chapter will provide the reader with a general overview of health information technology, specifically focusing on current CDSS implementations including reported benefits as well as challenges and barriers to adoption. It then provides specific information on CDSS alerting and the phenomenon of 'alert fatigue', and finally describes the use of data mining in healthcare, focusing on the use of clustering and classification techniques.

2.2 Healthcare IT

Over the past number of years, organizations governing the delivery of healthcare in industrialized nations have identified the need for significant change in care delivery models in order to address a variety of challenges including increased life expectancy [1], higher volumes of patients with chronic conditions and multiple comorbidities [2], mounting financial pressures, and growing demands for improved quality and positive patient outcomes [3]. As a result, care delivery models have shifted whereby services that have traditionally been addressed via acute care models are now being delivered in ambulatory care settings [4, 5]. More, and increasingly complex care, creates challenges for primary care clinicians in providing patient care. Information technology provides one mechanism to overcome challenges faced by primary care providers in terms of patient management and care delivery.

Health information technology (HIT) presents a wide range of opportunities for improving and transforming healthcare including improvement of practice efficiencies, facilitating care coordination amongst clinicians, reducing human error, improving clinical outcomes, and tracking data over time [50]. Since the IOM's 2001 release of 'Crossing the Quality Chasm' report, and with ongoing technological advances, HIT continues to be suggested as an important mechanism for advancing clinical care and addressing quality and patient safety concerns. In fact, in 2009, the HITECH Act was passed into US law, promoting the "adoption and meaningful use of health information technology" [51]. Meaningful Use goals were established, and set out in stages, beginning in 2011 through to 2015 (Stages 1, 2 and 3). The Meaningful Use criteria focus on discrete data capture and sharing, advanced clinical processes and clinical decision support, HIT adoption, improved outcomes, and truly transforming care delivery through HIT [52]. Similarly, the NHS claims to be the most globally advanced in terms of its use of IT in primary health care, and in the 'NHS Five Year Forward View' report released in 2017, they commit to being fully paperless across the entire healthcare delivery sector by 2020 [53]. These initiatives have contributed to further implementation and adoption of HIT applications globally, across the entire spectrum of care delivery including computerized physician order entry (CPOE), electronic prescribing, electronic patient handover tools, bar coded medication administration, automated medication dispensing cabinets, electronic medication administration records, patient portals, electronic referrals, remote patient monitoring technologies, integrated biomedical devices (i.e. Smart pumps), EMR/EHRs, as well as CDS systems.

2.3 Clinical Decision Support Systems

Clinical decision support systems (CDSS) provide "… clinicians, staff, patients or other individuals with knowledge and person-specific information, intelligently filtered or presented at appropriate times, to enhance health and health care" [6]. The term CDSS is most widely used for computer-based interventions/support delivered through clinical information systems [7]. Some examples of CDSS functionality include electronic reminders and alerts to providers and/or patients, integration of clinical practice guidelines into electronic workflows, condition-specific order sets, diagnostic support, or context-based reference information, to name a few [6]. CDSS are "not intended to replace clinician judgment, but rather to provide information to assist care team members in managing the complex and expanding volume of biomedical and person-specific data needed to make timely, informed, and higher quality decisions based on current clinical science" [8].

Although a significant volume of research has been published around CDSS (a PubMed search through January, 2018 returns 1468 papers), most studies involve small, internally developed CDSS applications that have been designed for a specific disease or condition [54]. As a result, comparability, and therefore generalizability, proves challenging. None the less, many recent papers do report positive outcomes related to CDSS implementation. For example, a 2014 systematic review of US based acute care facility implementation of Computerized Provider Order Entry (CPOE) with CDSS found that use of CPOE reduced preventable adverse drug events by more than 50% when compared with facilities using paper based ordering [55]. The same study also reported a reduction in medication errors by approximately 50% when using CPOE with clinical decision support versus paper based ordering [55]. Yet, as of 2012, only 44% of US based acute care hospitals had implemented an EHR with CPOE and clinical decision support [55].

More recently, Varghese et al. conducted a review examining the effects of CDSS implementations on patient outcomes in inpatient care. This review was based on Kawamoto et al.'s definition of CDSS, which states "A CDSS is any electronic system designed to aid directly in clinical decision making, in which characteristics of individual patients are used to generate patient-specific assessments or recommendations that are then presented to clinicians for consideration" [56]. This study is unique in that it focuses more broadly on patient outcomes as opposed to improvements in physician adherence to clinical practice guidelines, error reduction or improving clinical workflows which much of the research has reported on to date. Seventy papers were included in the review, and researchers found that almost all reported CDSS use as being associated with positive patient outcome effects; but with substantial differences with respect to clinical impact [57]. The reviewers actually suggest that there are some specific use cases for CDSS in terms of improved patient outcomes, specifically around blood glucose management, blood transfusion management, physiological surveillance, pressure ulcer prevention, acute kidney injury prevention and VTE prophylaxis [57].

A 2014 paper published by Loja et al. examined RCT's "assessing the effectiveness of computerized decision support systems (CDSSs) featuring rule- or algorithm-based software integrated with electronic health records (EHRs) and evidence-based knowledge" [58]. This study focused on RCT's whose goals were to evaluate mortality and morbidity outcomes. Based on the 28 RCT's who met the inclusion criteria, there was no evidence that CDSS improved mortality rates, and minimal evidence for a decrease in morbidity associated with CDSS use [58]. That being said, the authors also note the limited number of studies reporting on solid, tangible outcomes. Yet another, more recent paper, published in 2018 highlights the need for caution around CDSS implementations. Stone shares two cases in which CDSS's led to unintended adverse patient outcomes related not necessarily to the system itself; but to the reliance on integrated applications and the need to be aware of changes to those point of care (POC) and other ancillary systems [59]. That being said, there has been minimal evidence reported to date around negative outcomes or unintended adverse events related to CDSS use.

While much evidence exists supporting the use of CDSS in some way, and HIT in general, there remain significant barriers to CDSS adoption. Although 87% of primary care practices in the US report EMR use [12], it is unclear to what extent rich, evidence-based knowledge rules and support are integrated into those systems. Current research notes key CDSS adoption barriers which include a lack of physician involvement in CDSS rule development, the need for some level of physician control or choice (for example, addressing physician preferences in terms of the timing of when patient care recommendations are presented), and the perceived lack of physician autonomy [60,61]. A 2015 study conducted focus groups with primary care physicians, and explored perceived barriers to using CDSS in the primary care setting. Commonly perceived barriers included: insufficient knowledge of the CDSS, irrelevant alerts, too high an intensity of alerts, a lack of flexibility and lack of learning in the CDSS application, a negative effect on patient interaction, and additional amount of time and effort required to use the CDSS [60]. This study discusses physician concern around the reliability and accuracy of the CDSS content; but also highlights the need for flexibility and adaptability of the system to allow for personal physician preferences [60]. A 2016 study conducted focus groups with both physicians and nurses, found incorrect reminders as well as an inability to prioritize reminders to be two important barriers to EMR CDSS adoption [77]. Liberati et al. describe a CDSS implementation framework to help address challenges with CDSS adoption [67]. These challenges include selecting and presenting relevant alerts, in order to avoid alert fatigue, and as in the 2016 study, Liberati et al. also discuss the need for regular updates and continual improvement of the CDSS application [67].

Despite some of the challenges and barriers associated with CDSS implementation and adoption, its use will likely continue to increase given continued calls for meaningful use of HIT, and increased familiarity with technology across the clinician population, and growing concerns around quality and safety related to patient care [62]. The challenge lies in presenting the right information to the right clinician at the right time in order to ensure that clinician is able to make an informed decision and quickly take action on that decision. This work attempts to address some of these challenges, particularly in the areas of CDSS alerting and alert fatigue.

2.4 CDSS Alerts

With clinical practice guidelines changing every three to five years [63], and with over one million new biomedical papers published each year [65], the sheer volume of clinical information available to physicians and other clinicians to help inform their clinical practice is undoubtedly overwhelming. As the adoption of electronic clinical information systems continues to grow, the volume of more easily accessible, electronic patient data climbs. CDSS applications attempt to address this challenge by linking patient-specific data to a ‘knowledge base’ in order to generate information and suggestions meant to directly improve patient care delivery. A knowledge base is simply a database that stores knowledge in a suitable form depending on its use [13]. Alerting is one form of clinical decision support, and can be defined, at a high level, as providing relevant information back to the clinician electronically via the clinician’s EMR application. Alerts are electronic notifications intended to advise a clinician on a course of action based on relevant patient information coupled with supporting clinical evidence. CDSS alerts can be synchronous or asynchronous [21]:

- Asynchronous alerts are non-interruptive notifications or supports [64]. An example might be an automatically generated print out to the pharmacy when a patient’s INR level is greater than 4. Not triggered by a specific user task at point of care; but could be event-driven (for example, notification of an abnormal lab result) or display of filtered reference information or other knowledge resources requested of the CDSS by the physician for a particular patient or disease/condition type. Asynchronous alerts are commonly thought of as 'inbox notifications' including referral responses or test results ready for review.

- Synchronous alerts are the focus of this research, and are typically delivered in the form of immediate, interruptive activities such as "pop-up" alerts, information displays, links, or through targeted highlighting of relevant data [14]. This type of alert serves two main functions: either to remind the clinician to perform a task, or to alert the clinician about the potential consequences with not performing a task [66].

Synchronous CDSS alerts generally take the form of a notification or reminder meant to help inform the clinician around best evidence related to patient care. These alerts present at the point of decision making. For example, the alert may take the form of a preventative care reminder, such as a patient being overdue for a vaccine or it could take the form of a more urgent notification or warning of a patient drug allergy or contraindication as the physician attempts to order a patient prescription. Alerts are either accepted (meaning the provider has acknowledged the alert/adjusted course of action in support of the alert) or rejected (meaning the alert was overridden, or ignored). Some CDSS applications are designed such that alerts deemed critical or urgent enforce a ‘hard stop’ preventing a clinical workflow from continuing, although more often these high priority alerts simply require physician justification for overriding the alert and as a result the workflow is permitted to continue. The latter measure aids in supporting physician autonomy – a widely reported barrier to physician adoption to date [61, 67].

Both alert types can be either knowledge-based or data-driven. Asynchronous alerts are less likely to contribute to alert fatigue given their non-interruptive nature. On the contrary, synchronous alerts, which often take the form of a ‘pop-up’ box, are a significant contributor to alert fatigue given the interruption to clinical workflow [14, 66], and as such, are the focus of this research.

2.5 Alert Classification

Recognizing that differences exist amongst physicians, we also acknowledge the heterogeneity amongst EMR alerts. We expect that differences in CDSS EMR alerts will invoke varied responses from physicians and physician groups. In order to validate these assumptions; however, we require a means of classifying the various EMR alerts.

An extensive literature review was conducted to determine whether any alert classification frameworks had been developed and/or reported on to date. PubMed is a free search engine “developed and maintained by the National Center for Biotechnology Information (NCBI)” [46], and was utilized extensively for this work. Only one paper, published in 2011 by Wright et al., discussed development of a CDSS alert taxonomy [21]. The authors had completed an extensive comparison of front-end CDSS tools in both commercially developed and internally developed EHRs/EMRs (these terms are frequently used interchangeably in the literature). After examination of systematic literature reviews on CDSS, coupled with previously conducted qualitative research as well as extensive experience in the field of CDSS, the authors developed a preliminary list of 46 CDS tools [21]. They subsequently organized and facilitated an in person conference attended by eleven US based HIT and CDSS experts, as well as the researchers themselves. From there, CDS types were divided into 6 categories: medication dosing support, order facilitators, point of care alerts/reminders, relevant information display, expert systems and workflow support [21]. While this taxonomy was not specific to primary care EMR systems, relevant CDSS alert types were taken from this taxonomy to create the EMR alert classification used in this research.

2.5.1 CDS Group and Type

As noted in Section 2.5, Wright et al. identified six CDS categories in their taxonomy. For the purposes of our research, relevant information display and workflow support were excluded. Relevant information display includes CDS tools such as 'tall man lettering' (where applications vary the case of medication names that resemble one another in order to highlight critical differences), or 'context-sensitive information retrieval' often referred to as info buttons. This type of CDS simply *occurs* and does not offer the clinician to accept or ignore the support, and therefore was excluded from this study. Wright et al.'s 'Workflow Support' category includes things like parsing of free text orders into structured fields, as well as documentation aids such as templates and other tools for documenting care in both structured (discrete) and non-structured forms [21]. Similar to the ‘relevant information display’ category, these supports are not actionable by a physician and cannot be recorded as accepted nor ignored and therefore were also excluded from this research. Medication dosing support, order facilitators, point of care alerts/reminders and expert systems; however, do all consist of synchronous alerts whose acceptance can be captured, and are also relevant in the primary care setting, and therefore these categories were included in this research.

Wright et al. also identified, based on extensive research and engagement with experts, sub-categories under each of the six previously discussed categories. For each of the four categories included in our research, we reviewed each of the sub-categories listed, the evidence referenced for each, and determined whether or not they should be included in our research. The inclusion criteria included: a) was the CDS delivered in such a way that it could be accepted/rejected and could the action taken by the physician be recorded?, and b) was the CDS sub-category relevant to the primary care environment? Tables 2-1 through 2-4 provide detail on each CDS group and associated CDS types that met the inclusion criteria for our research [21].

Table 2-1: Medication Dosing Support

CDS Group	CDS Type	Description
1	Medication dose adjustment	Assistance with adjusting or calculating medication doses based on patient characteristics such as age, weight, etc.
1	Formulary checking	Check medication orders against hospital or payer formularies. May also suggest more cost effective therapies
1	Single dose range checking	Checking to see whether a single dose of a medication falls outside of an allowable range
1	Maximum daily dose checking	Checking to see whether a combined daily dose of a medication exceeds a specified maximum daily dose.
1	Default doses/pick lists	Providing common doses of a medication for a provider to choose from.
1	Indication-based dosing	Adjusting default medication doses based on indications entered by an ordering provider.

Table 2-2: Order Facilitators

CDS Group	CDS Type	Description
2	Medication order sentences	Complete statements of orders which a provider can order as a single unit.
2	Subsequent or corollary orders	Suggesting or automatically ordering something based, on or in response to, another order.
2	Indication-based ordering	Suggesting orders based on the indication entered by the ordering provider.
2	Condition-specific order sets	Order sets (collections of common orders) based on a disease or problem that the patient has.
2	Non-medication order sentences	Complete statements of non-medication orders which a provider can order as a single unit.

Table 2-3: Point of Care Alerts/Reminders

CDS Group	CDS Type	Description
3	Drug-condition interaction checking	Checking medication orders against the patient problem list for possible contraindications.
3	Drug-drug interaction checking	Checking medication orders and the medication list for possible contraindications.
3	Drug-allergy interaction checking	Checking medication orders against the allergy list for possible contraindications.
3	Plan of care alerts	Time-based alerts relating to planning of care.
3	Critical laboratory value checking	Comparing laboratory results to reference ranges and alerting providers to critical values.
3	Duplicate order checking	Checking active medication orders and the medication list for possible duplication.
3	Care reminders	Reminders to order a diagnostic or therapeutic procedure based on patient parameters including preventative care reminders, chronic disease reminders, or palliative care reminders.
3	Look-alike/sound-alike medication warnings	Warn providers when they order a medication whose name looks or sounds like another drug.

CDS Group	CDS Type	Description
3	Ticklers	Time-based alerts that an order has not been fully carried out.
3	Problem list management	Alerts, reminders and automated documentation tools that help providers maintain an accurate problem list.
3	Radiology ordering support	Assistance in selecting appropriate radiology studies based on patient conditions.
3	High-risk state monitoring	Alerting the provider to high risk states.
3	Polypharmacy alerts	Alerting the provider when patients are on a high number of medications.

Table 2-4: Expert Systems

CDS Group	CDS Type	Description
4	Antibiotic ordering support	Antibiotic suggestions based on patient history, culture results, patient characteristics, etc.
4	Diagnostic support	Differential diagnosis suggestions based on patient signs and symptoms.

2.6 Alert Fatigue

There are a number of challenges related to CDSS alerts. Arguably, the most widely reported impediment is the concept of ‘alert fatigue’. The AHRQ defines alert fatigue as the process by which clinicians become desensitized to alerts, and as a result ignore or fail to respond appropriately to such warnings [16]. There is a general consensus amongst clinicians and administrators that CDSS alerts are an important tool in enabling improved, standardized patient care [17, 18, 24, 25, 27, 28]; however the majority of systems implemented to date struggle to achieve appropriate sensitivity and specificity levels, and generate higher volumes of alerts than are necessary and/or clinically relevant [19, 20, 31]. Alert sensitivity describes the ability of a CDSS to alert clinicians correctly when patients are at risk of experiencing harm [20]. The specificity of a CDSS is a measure of its ability to distinguish between events that put a patient at risk of harm as opposed to non-events that will not [20]. The higher the volume of false positives, the lower the specificity, resulting in increased clinician frustration and lower alert acceptance rates. While alert specificity and sensitivity are important, they do not necessarily contribute to the volume of alerts presented to clinicians, although arguably, could contribute to a reduction in volume should the lower sensitivity and specificity alerts be prevented from firing unnecessarily.

Several conceptual models have been identified in terms of alert fatigue. One has been coined by Ancker et al. as ‘cognitive overload’, which suggests that alert fatigue is caused by receipt of large volumes of information with “insufficient time or cognitive resources to distinguish relevant from irrelevant information” [68]. This theory implies that a reduction in the volume of clinician ‘workload’ including CDSS alerts will improve override rates which have been widely reported as between 50 and as high as 96% across a variety of systems and settings [68, 69, 70]. A 2017 study which examined provider acceptance responses over time, while factoring in provider workload factors such as volume of alerts, patient complexity, volume of patients and quantity of orders reported a significant reduction in acceptance rates as increases in the number of reminders, increases in the volume of repeated reminders for the same patient, and patient complexity also increased [68]. This is particularly relevant in the primary care environment, where, as we noted above, with changing care models providers are seeing increasingly complex patients. Most CDSS are developed based on treatment guidelines designed to treat single diseases [66], resulting in increased reminder volumes for primary health care clinicians.

Other research, however, reports no evidence of a relationship between volume of alerts and override rates [72, 73]. These studies lend support to another model: ‘desensitization’, theorizing that repeated exposure to alerts leads to decreasing provider responsiveness. This theory suggests that acceptance rates are higher for newly added CDSS alerts; but that acceptance rates of those alerts decreases over time [68]. There is little evidence to support this theory; however, and some evidence actually goes against this hypothesis suggesting that physicians do continue to accept alerts even after longer term exposure to the same alert when the alert is appropriate [68]. The ability to isolate compounding factors, in particular, the appropriateness of the alert in each specific context, makes this desensitization theory difficult to evaluate.

Some mechanisms/techniques have been identified to help reduce alert fatigue, although no real ‘best practices’ surrounding the development of CDSS alerts, in order to prevent alert fatigue, have been published to date [74]. McCoy et al. (2014) show that increasing clinical context integration increases alert appropriateness thereby increasing alert acceptance rates [71]. Alert appropriateness is crucial, given findings that at least 70% of alerts need to be appropriate if providers are to have any confidence in the CDSS application [75]. A recent 2018 review of important CDSS design elements required for provider acceptance and subsequent improved outcomes, identified the need for the

CDSS to adapt its alerting based on relevant actions taken by physicians in similar situations [74] in order to reduce the volume of inappropriate alerts.

Another mechanism to address alert fatigue is to review all possible alerts and tier or rank the alerts. This method presents alerts differently based on alert severity and there is evidence to suggest that the tiering of alerts improves alert acceptance. In a randomized control trial (RCT), Paterno et al. tiered DDI alerts at one of two facilities using the same knowledge base. Level 1 alerts interrupted the ordering process and presented as a 'hard stop' preventing the user from proceeding without taking some action at the tiered site. The tiered facility reported 100% alert compliance for Level 1 alerts compared with only 34% at the non-tiered sight [26] demonstrating the importance of alert stratification as a means of presenting the right alert, at the right time and in the right way, in order to reduce alert fatigue. Another 2017 study looked at grouping similar medication related alerts into single alerts, and showed that by performing this clustering in alert generation, they were able to reduce the alert rate within these clusters by 53-70% [76].

Integrating more clinical and patient context to refine alerts and improve alert appropriateness is clearly an important area of CDSS development. Provider context, however, must be considered as well. Research in the area of personalized provider alert presentation is limited. Coleman et al. identified the need for additional research in this area, identifying key gaps in CDSS alert research. Presentation and personalization of alerts as well as the design and firing of alerts/rules were two of the eight key areas identified [20]. Although significant bodies of knowledge have been published around CDSS alerting, most of these studies take place in acute care settings. The Primary Health Care environment is very different with providers seeing a wide range of patients over extended periods of time. Patient care needs in this setting are in stark contrast to those who find themselves in the acute care setting, and the needs of, and demands placed on, primary care providers are also in contrast to their counterparts in acute care. This further supports the need for more research in the area of CDSS alerting in primary care. We believe that a greater understanding and evaluation of the physician's context can lead to significant improvements CDSS alert delivery and acceptance, as a means of addressing alert fatigue, particularly in the complex primary care environment.

2.7 Data Mining and Machine Learning in Healthcare

Machine learning itself is not a new concept, having been discussed since the introduction of computers in the first half of the nineteenth century. The terms "machine learning" and "data mining" are often used interchangeably; however there are some notable differences. Data mining pulls from existing data sets in order to identify patterns in the data that can help shape decision making, whereas machine learning involves observing patterns and *learning* from them to in order to adapt behavior for future incidents [29]. Although data scientists can set up data mining to automatically look for specific types of data and parameters, it doesn't learn and apply knowledge on its own without human interaction [29]. One of the most commonly cited data mining definitions comes from MIT in 2001, and states that "data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner" [30]. In that same year, a technology review (again published by MIT) listed data mining as one of the top ten emerging technologies that would change the world [31]. The goal of data mining is to gain a deeper understanding of large volumes of data in order to advance domain knowledge and support improved decision making. The digital world continues to grow as more and more information is captured electronically across all industries and sectors. Healthcare is no different, with massive amounts of data accumulating as much of the patient care delivery processes are captured via electronic methods be it through electronic physiological monitoring and biomedical devices, or through an expanded use of hospital information systems. With electronic storage costs and computational power continually decreasing in cost, the field of data mining, and the opportunities within, continues to expand.

Although data mining has been successfully employed across various sectors (financial, retail, marketing, and manufacturing) to date, the use of machine learning in healthcare has been minimal. With growing volumes of increasingly complex data, and increasing pressures to reduce costs and improve outcomes, traditional analysis methods are no longer feasible, and machine learning tools are becoming a necessity. Data mining techniques can be broadly categorized by their capabilities which include data description and visualization, clustering and association, and finally, classification and estimation (predictive modeling) [33]. Clustering refers to "the division of data into groups of similar objects" [40]. Clustering is used when there is no class to be predicted; rather the goal is to separate the instances into natural groups. A variety of different clustering methods exist, each accepting and analyzing training data in order to generate clusters by which test data can be evaluated and classified against. Each method serves a unique purpose, and results may be expressed as overlapping, exclusive, probabilistic or hierarchical [41]. Classification is a supervised learning approach in which a computer program learns from the data input given to it (training data), and then uses this learning to classify a new observation [81]. Essentially, classification techniques are used to group membership for data instances. Classification is considered 'supervised' given that the algorithm learns based on labeled data – meaning the group membership, or 'class' of an instance is known [81].

Leveraging these techniques in healthcare can be challenging for several reasons. Healthcare data is typically stored across a number of systems: administrative, laboratory, clinics, health records, etc. Data must be collected and integrated before it can be analyzed, which can be a time consuming, resource intense, and therefore costly endeavor. Heterogeneity of patients as well as treatments adds additional complexity. Because traditional hospital information systems (HIS) were developed for billing and financial objectives, patient specific data is often of low quality (i.e. missing, inconsistent, unstructured, and/or non-standardized) resulting in poor algorithm accuracy insufficient for use in a clinical environment. Successful data mining requires both domain expertise and data mining proficiency. Health professionals, while domain experts, are generally not well versed in data mining techniques required for successful development, implementation and evaluation.

Despite the challenges mentioned above, there have been some notable studies published in the area of data mining in healthcare. Findings highlight success in the areas of fraud detection, cost prediction, decreased length of stay, reduction of readmission by identifying patients who are at high risk for readmission and addressing care concerns thereby preventing the need for readmission [readmission], disease identification [32], predicting and preventing adverse events [35] and personalized medicine [33, 34, 35, 36, 37, 38]. This research focuses on predictive modeling using traditional clustering and classification algorithms to address the well documented challenge of alert fatigue in a novel way.

2.8 Conclusion

In this chapter, we have provided an overview of HIT and Clinical Decision Support systems. We have provided a review of recent studies which highlight some of the key benefits and challenges associated with CDSS, and CDSS alerts, in particular. We discussed alert classification and alert types. We also reviewed the phenomenon of alert fatigue, and the implications of alert fatigue in the primary care setting. Finally, we have provided some background on data mining and machine learning in healthcare.

Lugtenburg et al. highlight the need to improve flexibility and learning capability of systems in order to increase options to adapt the CDSS applications to meet the varying needs of different physicians as an important area for future research [60]. In the next chapter, we will discuss our research methodology and solution design which aims to address some of the gaps in existing research related to physician context and CDSS alerting.

CHAPTER 3: RESEARCH METHODS AND DESIGN

This chapter will provide an overview of the research methods that were used to design, develop, and implement this research study. First, we will discuss our overall research design. Next, we will discuss the approach taken to assemble physicians into clusters based on a selection of physician and practice attributes. We will discuss the need for an alert classification framework, as well as the methods and justification for doing so. Finally, in this chapter we discuss our approach to patient typing for inclusion in our models.

3.1 Solution Approach

In this section, we will discuss the approach to be taken to address each of the research objectives outlined below in Table 3-1.

Table 3-1: Research objectives guiding the solution approach.

	Research Objectives
1	To stratify physicians into distinct practice groups to design a group-level alert issuance strategy
2	To develop a classification scheme for EMR-based CDS alerts based on a review of current literature.
3	To establish a mapping between physician groups and alerts types
4	To develop a strategy to issue alerts based on physician’s practice and alert response behavior in order to minimize alert fatigue

This work will be designed as a ‘Proof of Concept’ utilizing both collected and simulated data to investigate our research objectives and to develop an alert issuance strategy. There are several reasons for this approach. Firstly, this research will be conducted in the Province of Nova Scotia. While increasing numbers of primary care physicians are using electronic medical records, the degree to which CDSS is implemented across the primary health care setting in this province is currently limited. Additionally, multiple EMRs are in use provincially, and the lack of consistency and standards introduces some challenges in terms of our ability to extract meaningful data related to CDSS alerts and associated acceptance rates. For these reasons, we chose a proof of concept approach, and acknowledge that the outcomes of this research may identify opportunities to further explore these concepts with real world data.

The Waikato Environment for Knowledge Analysis (Weka) is a suite of open-source, java-based, machine learning software developed at the University of Waikato, New Zealand [43]. Weka offers a variety of data mining algorithms and its use is widely reported in the literature [44]. In addition, it provides an intuitive GUI. For all of these reasons, Weka was selected as the data mining software for this research.

Section 3.2 will discuss, in detail, the approach taken to address our first research question. This is a critical first step as the physician clustering will provide the foundation for the rest of this research work. We will discuss the data collection and preparation methods, and will highlight the clustering techniques proposed and the rationale for their selection.

Section 3.3 focuses on the steps we will take to develop an EMR alert classification framework. This is needed in in order to address both our second and third research goals: classification of the wide range of CDSS alerts based on their source, acuity and response expectations and our ability to establish a mapping between physician groups and alert types. A clear alert framework which outlines alert severity and indicates alert ‘type’ will enable meaningful predictive modeling. This section will detail the methods used to develop a clear, meaningful and re-usable alert classification.

Finally, Section 3.4 will discuss our approach to stratifying patients. We will seek out current research around patient typing, as we acknowledge that volumes of CDSS alerts will be higher for patients who take multiple medications or who have multiple comorbidities. We will seek out evidence to support a tiering of patients as we believe this to be an important factor to be considered as part of our proof of concept. We believe that if we are able to cluster physicians into distinct ‘types’, and if we are able to design an alert classification, we can then leverage machine learning classification algorithms to develop predictive models, leading to novel approaches in alert issuance and reductions in alert fatigue. Figure 3-1 provides an overview of our research approach.

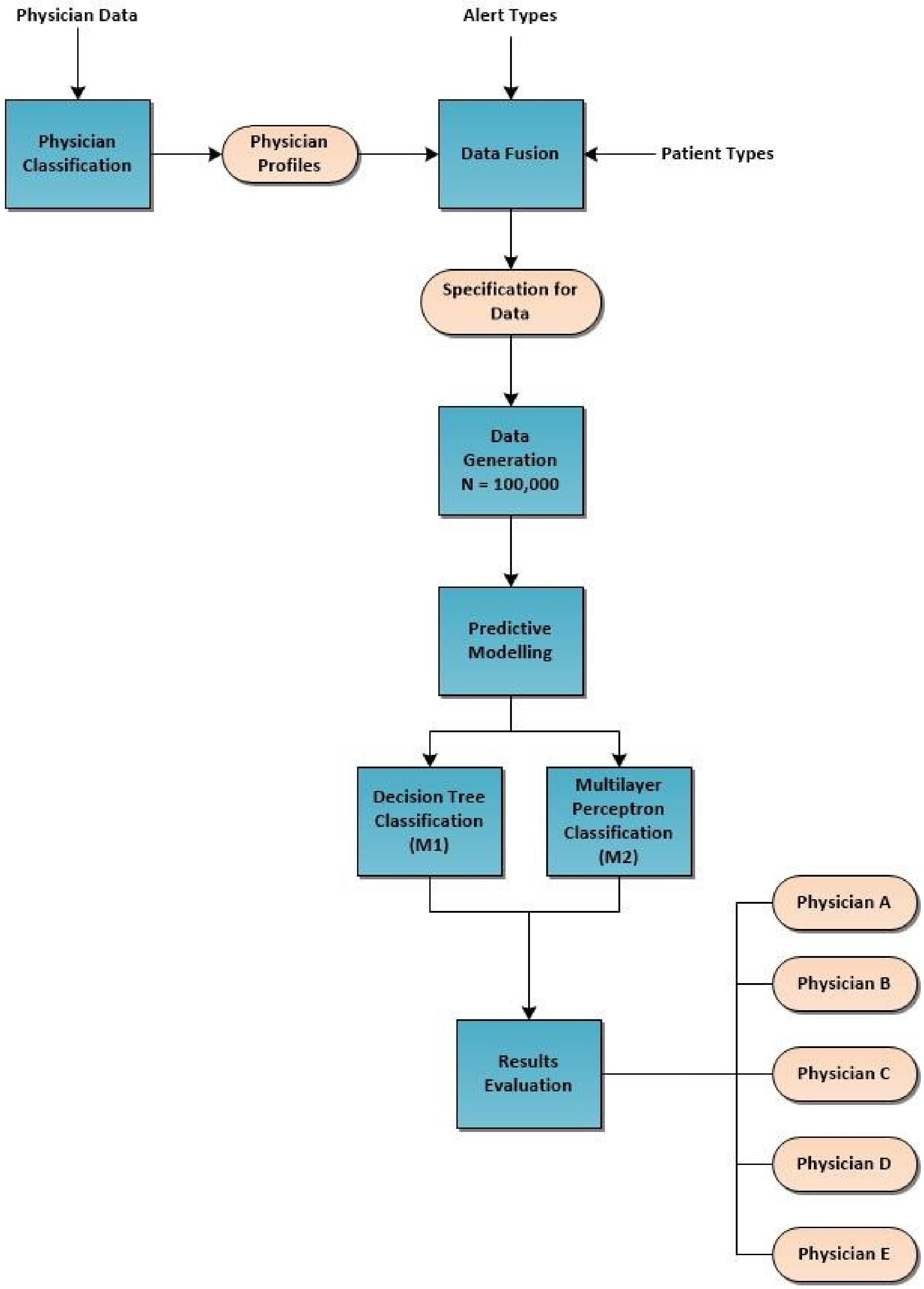

Figure 3-1: Solution Design Overview

3.2 Physician Classification

All physicians are not the same, and this research argues that differences amongst physicians and their practices will influence how they respond to EMR-based CDSS alerts. Although this work is a proof of concept, it was important that the data used here is reflective, as best possible, of the real world. Physician data was procured from the College of Physicians and Surgeons of Nova Scotia (CPSNS) [39] covering data for all physicians with a specialty of 'Family Medicine' licensed to practice in the province of Nova Scotia. The listing included a total of 1544 physicians. Table 3.2 presents the CPSNS data attributes.

Table 3-2: List of Physician Attributes (CPSNS)

Attribute	Description	Possible Values
Registration Number	Unique Identifier provided by the college	Numeric
Gender	Sex	Male, Female
Year of Graduation	Year physician graduated from Medical School	4-digit year
University	The university the physician received their degree from	String
Specialty 2	Describes a secondary specialty (aside from Family Medicine)	String
Office City	City in NS where physician's primary office is located	String
Office Postal Code	Canada Post provided postal code of primary practice location	Var Char

Of the 1544 physicians, only 34 contained a secondary specialty. Therefore, this attribute was removed. Office postal code was also removed, as the 'Office City' attribute provided sufficient detail and the postal code would have been redundant. We did not believe that the CPSNS information alone included sufficient detail for determination of a physician ‘type’. For this reason, additional attributes were identified and added based on our understanding of physician characteristics that might influence a family physician’s response to an EMR alert.

A number of studies have reported differences in terms of EMR adoption dependent upon physician age. A 2012 paper reviewed EMR adoption over a ten year period beginning in 2002 and reported a significant lag in adoption of these electronic tools in physicians aged 55 and older [82]. Another systematic literature review of health care provider adoption of eHealth technologies sought to identify ‘influential factors’ to provider acceptance of various eHealth systems (including CDSS). This study confirmed physician age, gender and years in practice all to be important acceptance factors [83]. This aligns with our hypothesis that physician age and experience (both in medical practice and in technology use) would drive different CDSS alert acceptance behaviors and for these reasons, ‘Age’ and ‘Years in Practice’ were added.

Physician perception that use of EHR or CDS systems leads to increased workload has been widely reported in the literature [25, 28, 68]; however we believe that physician workload itself is also an important factor to consider in terms of CDSS alert acceptance. A recent study supports this theory, noting that physician workload and work complexity are important considerations when studying alert fatigue [68]. This study considered the number of patients seen each day, as well as the comorbidity index of these patients as important attributes contributing to physician workload. The volume of chronic patients is important because the existence of multiple comorbidities requires additional evaluation and management on the physician's part, and would trigger a higher volume of CDSS alerts for an individual patient. We believe that another important measure of both workload and physician experience is how frequently the physician is seeing patients. For these reasons, 'Days/week', '% Chronic Patients' and 'Patients/Day' attributes were included.

We believe workload complexity to be another important attribute that would drive different CDSS alert responses depending upon physician type. Anker et al. suggest that the volume of lab orders per patient encounter is an important indicator of physician workload complexity [68]. For the purposes of our study, we include both the volume of investigation orders (i.e. lab, diagnostic imaging) placed per month as well as the percentage of patients referred to other physicians as markers of physician workload complexity. Table 3-3 provides a listing of the additional physician attributes included in our study.

Table 3-3: List of Additional Physician Attributes

Attribute	Description	Possible Values
Age	Physician age in years	Numeric
Years in Practice	Number of years the physician has been practicing medicine	Numeric
Days/week	Number of days the physician practices each week	1, 2, 3, 4, 5
Patients/day	Number of patients the physician sees, on average, each day	20, 25, 30
% Chronic Patients	Percentage of physician's patient roster who have chronic conditions	10, 20, 30, 40, 50, 60, 70%
% Referrals	Percentage of patients who are referred to other physicians	10, 15, 20, 25, 30, 40, 50%
# of Investigation Orders/month	Number of investigation orders (lab, DI, etc.) ordered by the physician per month	50, 100, 200, 300

- 'Age' was generated based on some assumptions related to year of graduation and the location of the university graduated from (years to complete medical school being fewer in Europe versus North America, as an example).
- 'Years in Practice' was simulated based on similar assumptions as those for 'Age' above.
- 'Days/week' was simulated based on the assumption that more senior physicians were likely working less days/week compared to younger physicians who were in the earlier stage of their careers.
- The number of patients a physician sees per day was simulated randomly.

- The percentages of chronic patients on a physician's patient roster were simulated randomly.
- The '% Referrals' and '# of Investigation Orders/month' were loosely tied to the number of days the physician worked each week, and the average number of patients seen per day.

Next, attributes 'Age' and 'Years in Practice' were discretized. Figure 3-2 provides a visual representation of the physician data. Here we list 'Years in Practice' across the horizontal access (0-10, 11-20, 21-30, 31-40, 41-50 and >50). Next we show 'Age' based on years of practice, across the horizontal access, and split based on physician gender. So, for example, for physicians who have been practicing 0-10 years, we see that just under 100 males and just under 140 females are between the ages of thirty and forty.

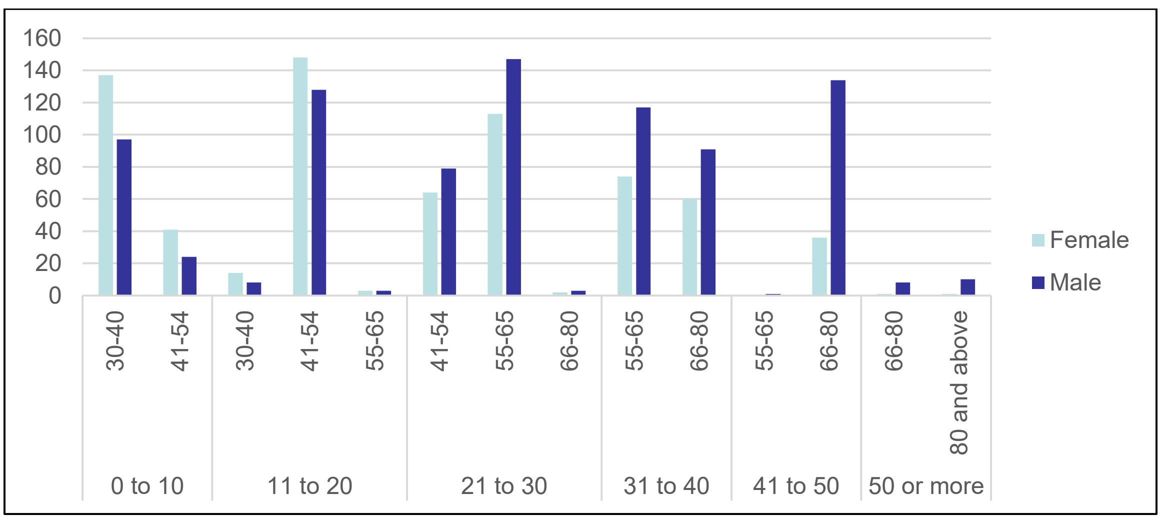

Figure 3-2: Overview of Physician Data. Count of Physician by Gender, Age, and Years in Practice.

3.2.1 Physician Data Clustering

The objective of this task was to determine whether we could identify distinct physician groups using clustering techniques. We utilized the k-means algorithm [42] against the physician data discussed in Section 3.2. Our choice of algorithm was based on the view that k-means provides a simple, straightforward and effective technique to identify clusters within a given dataset [41]. K-means works as follows: the parameter, *k*, represents the number of clusters. The number of clusters being sought must be specified in advance, and then *k* points are chosen as random cluster centers [41]. All instances in the data set are assigned to their closest cluster center according to the Euclidean distance metric. The 'means' of the algorithm descriptor represents the calculation of the mean of the instances in each cluster – termed "centroid" [41]. These centroids then become the new center value for their respective cluster, and the process continues until the same points are assigned to each cluster in consecutive rounds, indicating that the cluster centers have stabilized.

Parameter selection is a critical step, and a full listing of available parameters for the SimpleKMeans algorithm can be found in Appendix A. The input parameter of 'k' is arguably the most important, and requires some experimentation in order to determine the most appropriate setting. Because we were targeting identification of distinct physician groups which could be used to develop predictive classification models at later stages of our study, we set the following values for k: k=4, k=5, k=6, k= 7 and k=8. We set the cluster initialization method to 'Random' meaning that the initial cluster center locations were set at random. Euclidean distance was used for the distance function, and the seed was set to 10. The maximum number of iterations was set to 500. We felt this to be reasonable given 1544 instances.

Unlike clustering, classification and association learning make predictions on test data and those predictions are simply true or false [41]. This makes the evaluation of clustering techniques challenging. While some techniques for k-means cluster validation have certainly been reported [45], they are complicated to implement and often impractical [41]. We did, however, add the cluster assignment as a class, and we subsequently performed a classes to clusters evaluation. We ran each value of k through the algorithm ten times (n= 10), and then calculated the average error rate for each value of k tested. We found k= 5 delivered the best fit, as it produced the least cluster variance. Table 3-4 lists the error rates for each value of k. k=5 had the lowest error rate and therefore this model was selected. With an error rate of 11.33%, we believe that the results have balance across the clusters, and are useful and adequate in this application context.

Table 3-4: Error rates for each value of k tested (n=10).

K value	Error Rate (%)
4	14.25
5	11.33
6	13.92
7	11.72
8	12.05

The selected model of five distinct clusters is detailed in Table 3-5.

Table 3-5: Grouping of physician data into five distinct clusters.

	Cluster A	Cluster B	Cluster C	Cluster D	Cluster E
Gender	Female	Male	Male	Male	Female
Age	55-65	66-80	55-65	66-80	41-54
Years In Practice	21-30	31-40	21-30	41-50	11-20
Work Days/Week	4	2	5	1	5
Patients/Day	20	25	25	20	30
Patients/Week	80	50	125	20	150
% Referrals	30	10	20	40	20
Investigation Orders/Month	100	50	200	100	100
TOTAL (% physicians assigned to cluster)	**28%**	**18%**	**18%**	**8%**	**28%**

The emergent clusters have some interesting differences across attributes. We note 2 out of 5 clusters are female, and these represent the largest clusters. Clusters B, C and D represent males in our older age categories (55+). We see variation across the remaining cluster attributes. We review Figure 3-2 and are confident that our clusters are representative of our data set. At this point, we are satisfied that it is, in fact, possible to utilize clustering algorithms to group physicians into 'types' based on a set of well-defined attributes.

3.3 EMR Alert Classification

As discussed in Section 2, an extensive literature review did not result in the identification of an alert classification scheme. Therefore, we set out to develop our own alert classification model. We acknowledge that differences in the urgency of the alert and the type of alert are likely to affect the physician response. The majority of studies published on CDSS alerting to date reference the urgency of the CDSS alert; however the way in which ‘urgency’ is defined differs substantially. There is consensus that different levels of urgency exist, and approaches to defining this have included both expert panels as well as researcher judgement based on knowledge/experience collected and the uniqueness of study design. Some studies identify clearly defined alert severity levels. For example Cornu et al. developed a six-level severity classification for DDI alerts [78]. For the purposes of this study, we reviewed each CDSS type and assigned one of two severity levels based on the following definitions:

- Urgent (1): Requiring immediate response
- Non-urgent (2): Not requiring immediate response

We conducted an extensive literature review surrounding CDSS alerts (as discussed in Section 2), and found that CDSS alerts are generally categorized into one of the following groupings (alert types): alert, reminder, or suggestion. We therefore defined these broad categories as follows:

- Alert: A high priority alert meant to notify a provider that some sort of intervention is required.
- Suggestion: An alert that notifies a provider that they *may* want to consider a particular course of action.
- Reminder: An alert that notifies a provider that a time-sensitive care related activity is due. For example, a patient is due for their annual physical.

We then reviewed each CDS Group/Type combination, as outlined in Section 2.5.1. We assigned both a severity level (1 or 2) as well as an 'alert type' (alert, suggestion, or reminder) to each based on the definitions described above. We assigned all CDS types categorized as 'alert' with a severity level equal to 1 (urgent). We then validated our classification with a Nova Scotia-based primary care physician who confirmed the reasonability of this design, and the applicability of the framework [79]. Our final classification work is detailed in Table 3-6.

Table 3-6: EMR Alert Classification

CDS Group #	CDS Group	CDS Type	Alert Severity	Alert Type
1	Medication Dosing	Medication dose adjustment	1	Alert
1	Medication Dosing	Formulary checking	2	Suggestion
1	Medication Dosing	Single dose range checking	1	Alert
1	Medication Dosing	Maximum daily dose checking	1	Alert
1	Medication Dosing	Default doses/pick lists	2	Suggestion
1	Medication Dosing	Indication-based dosing	2	Suggestion
2	Order Facilitator	Medication order sentences	2	Suggestion
2	Order Facilitator	Subsequent or corollary orders	2	Suggestion
2	Order Facilitator	Indication-based ordering	2	Suggestion
2	Order Facilitator	Condition-specific order sets	2	Suggestion
2	Order Facilitator	Non-medication order sentences	2	Suggestion
3	Point of Care Alert/Reminder	Drug-condition interaction checking	1	Alert

CDS Type	CDS Group	CDS Type	Alert Severity	Alert Type
3	Point of Care Alert/Reminder	Drug-drug interaction checking	1	Alert
3	Point of Care Alert/Reminder	Drug-allergy interaction checking	1	Alert
3	Point of Care Alert/Reminder	Plan of care alerts	1	Reminder
3	Point of Care Alert/Reminder	Critical laboratory value checking	1	Alert
3	Point of Care Alert/Reminder	Duplicate order checking	1	Alert
3	Point of Care Alert/Reminder	Care reminders	1	Reminder
3	Point of Care Alert/Reminder	Look-alike/sound-alike medication warnings	2	Suggestion
3	Point of Care Alert/Reminder	Ticklers	1	Reminder
3	Point of Care Alert/Reminder	Problem list management	2	Suggestion
3	Point of Care Alert/Reminder	Radiology ordering support	2	Suggestion
3	Point of Care Alert/Reminder	High-risk state monitoring	1	Alert
3	Point of Care Alert/Reminder	Polypharmacy alerts	1	Alert
4	Expert Systems	Antibiotic ordering support	2	Suggestion
4	Expert Systems	Diagnostic support	2	Suggestion

Figure 3-3 provides a visual overview of our EMR alert classification. Twenty six CDS types have been included across four CDS groups. Of those, ten have been categorized as ‘alerts’, three as ‘reminders’, and thirteen as ‘suggestions’. All suggestions have been categorized as non-urgent (severity 2), while all alerts and reminders included in this work have been classified as urgent (severity 1).

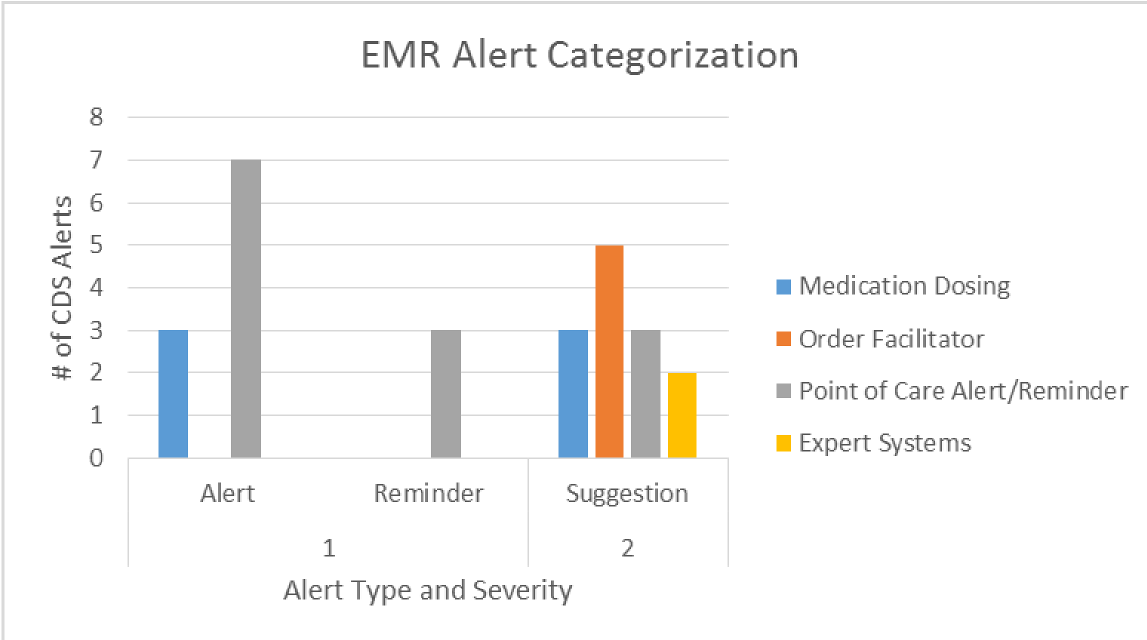

Figure 3-3: Overview of EMR alert categorization.

3.4 Patient Type

In order for this proof of concept to be meaningful, it was important to think about the context of the patient given that different patient types would invariably invoke different CDS alerts. We scanned the literature seeking patient classification methods and found that patient classification is heavily dependent upon the nature of the research and each study's specific research objectives. For example, many studies site patient attributes such as age or gender as highly relevant. For our research, we know that different CDS rules will be invoked based on these factors, perhaps; but more importantly, based on the general health (and therefore the complexity) of the patient. For this reason, we leverage work developed by the Advisory Board® [47] who suggest that there are three main patient types: 1) high risk, 2) rising risk, and 3) low risk as shown in Figure 3-4.

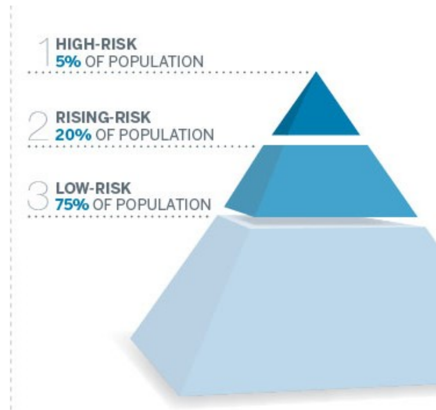

Figure 3-4: Population Health and the 3 Types of Patients [47]

The ‘high-risk’ patient is defined as having at least one complex illness, multiple co-morbidities and psychosocial problems. These patients make up approximately 5% of the overall patient population. The ‘rising-risk’ patient is defined as having multiple risk factors that threaten to push them into the higher risk category if not addressed/managed. For example, a diabetic patient who also smokes. These patients represent roughly 20% of the population. Finally, the ‘low-risk’ patient, who represents the remaining 75% of the population, are generally healthy, or have a well-managed condition.

It is assumed that given the volume and types of alerts generated for each patient type will be different, as will the way in which each physician type responds to these alerts (i.e. whether they accept or ignore the alert).

3.5 Chapter Summary

In this chapter, we have discussed our solution design beginning with our approach to physician ‘typing’ using the k-means clustering algorithm. This addresses our first research objective confirming the ability to stratify physicians based on a selected group of attributes. Next, we introduce an EMR alert classification model, built upon an alert classification of CDS group and type, alert severity, and alert type. This addresses our second research objective of classifying the wide range of CDSS alerts in terms of their source, acuity and response expectations. Lastly, we discuss the approach used for patient type categorization.

CHAPTER 4: CLASSIFICATION RESULTS

In this chapter we seek to answer our third research objective: To establish a mapping between physician groups and types. To do this, we look to machine learning classification algorithms to develop predictive models that will help predict how certain physician 'types' will respond specific CDSS EMR alert types. To do this, we will utilize two well documented, industry standard classification algorithms in an attempt to develop a predictive model. The Decision Tree (DT) algorithm, C4.5, was selected due to its strong performance in terms of classification accuracy, and easily interpreted output [48, 49]. The Multilayer Perceptron (MLP) algorithm is a robust algorithm with an ability to derive meaning from complicated, voluminous data. They work well for classification prediction problems, and for these reasons, we selected the MLP algorithm for our research. Our goal will be to develop a model that can predict, with a high level of accuracy, the alert response rate category for a given physician group.

Section 4.1 speaks to the alert response data generation process and sections 4.2 and 4.3 discuss the algorithms used, the reasons for their selection, the tuning of the algorithm parameters, and finally, the results for each. Section 4.4 provides a chapter summary.

4.1 Alert Response Data Simulation

We next set out to develop data in order to simulate alert response results for physicians based on their physician group, the EMR alert presented, and the patient type. The data was simulated based on the following assumptions: (i) volume of alerts is a function of patient type, (ii) whether a physician responds to an alert is a function of physician type, patient type, severity, CDS group and alert type. 100,000 alert responses were simulated. Figure 4-1 provides a sample of the simulated data, and Appendix B includes the simulation script used.

Physician Type	Patient Type	CDS Group	CDS Type	Severity	Alert Type	Volume	Respond	Ignore
A	Rising	Order Facilitator	Subsequent or corollary orders	2	Suggestion	56	32	24
B	High	Medication Dosing	Medication dose adjustment	1	Alert	28	12	16
C	Low	Point of Care Alert/Reminder	Drug-allergy interaction checking	1	Alert	236	74	162
B	Low	Medication Dosing	Medication dose adjustment	1	Alert	96	45	51
D	Low	Medication Dosing	Formulary checking	2	Suggestion	289	124	165
C	Low	Expert Systems	Antibiotic ordering support	2	Suggestion	201	61	140
E	Rising	Medication Dosing	Single dose range checking	1	Alert	55	29	26
D	Low	Order Facilitator	Subsequent or corollary orders	2	Suggestion	190	31	159
E	Low	Point of Care Alert/Reminder	High-risk state monitoring	1	Alert	273	123	150
A	Rising	Medication Dosing	Maximum daily dose checking	1	Alert	62	50	12
A	Low	Medication Dosing	Default doses/pick lists	2	Suggestion	316	146	170
A	Low	Point of Care Alert/Reminder	Drug-condition interaction checking	1	Alert	260	110	150
D	Low	Point of Care Alert/Reminder	Polypharmacy alerts	1	Alert	316	142	174

Figure 4-1: Sample physician response simulation data.

Alert acceptance is defined as a physician responding to an alert, whereas alert rejection is defined as a physician ignoring an alert. Response rate was then calculated for each instance based on Respond/Volume, and 'Response Rate' categories were assigned based on both quartiles and quantiles. Response rate categories and associated alert acceptance rates are captured in Table 4-1. The response rate categories are important for classification purposes.

Table 4-1: Response Rate categories and associated Alert Acceptance Rates based on Quartiles and Quantiles

Quantiles		Quartiles	
Response Rate Category	**Alert Acceptance Rate (%)**	**Response Rate Category**	**Alert Acceptance Rate (%)**
1	0-30	1	0-32
2	31-40	2	33-46
3	41-50	3	47-60
4	51-63	4	61-100
5	64-100		

4.2 Decision Tree Classification

Decision trees are a supervised learning method commonly used for classification purposes. Supervised learning can be defined as the data mining task of learning or inferring a function based upon labeled training data [41]. Decision trees classify instances by "… sorting them based on feature values. Each node in a decision tree represents a feature in an instance to be classified, and each branch represents a value that the node can assume. Instances are classified starting at the root node and sorted based on feature values." [48]. The C4.5 algorithm [49], developed by Ross Quinlan, was selected for its speed, simplicity, and ability to create a visual, easily understood/interpreted decision tree.

4.2.1 C4.5 Experimentation and Results

To begin experimentation, the simulated alert data was loaded into Weka. Feature selection is an important pre-processing step, and at this point attributes 'Volume', 'Respond' and 'Ignore' were removed given the addition of the

‘Response Rate’ category. The ‘Numeric to Nominal’ filter was applied across all remaining attributes, which included physician type, patient type, CDS group, CDS type, severity, alert type and response rate category ensuring that numerical attributes were treated as categorical by the classifier. There are a number of parameters available for tuning this algorithm (see Appendix C); but of note are pruning, the confidence factor, and minimum number of objects. The C4.5 algorithm provides a pruning option as a mechanism to prevent overfitting. C4.5 uses a pruning technique based on statistical confidence estimates. The core of this is the calculation of the confidence interval for the error rate [49]. In order to decide whether to replace a near-leaf node and its child leaves by a single leaf node, C4.5 compares the upper limits of the error confidence intervals for both the pruned and unpruned trees. For the unpruned tree, the upper error estimate is calculated as a weighted average over its child leaves. Finally, the tree with the lower estimated upper limit on the error rate "wins" and is selected. In Weka, the default confidence value is set to 25%; however can be adjusted. A lower value will lead to more drastic pruning. The ‘minimum number of objects’ parameter refers to the minimum number of instances – tests are not incorporated into the decision tree unless they have at least two outcomes that have at least the value set for the minimum number of objects [41]. The default in Weka is 2; however this can be increased to help address noisy data.

The first step was to import the simulation data into Weka. Preliminary tests were performed based on both the quartile and quantile data sets.

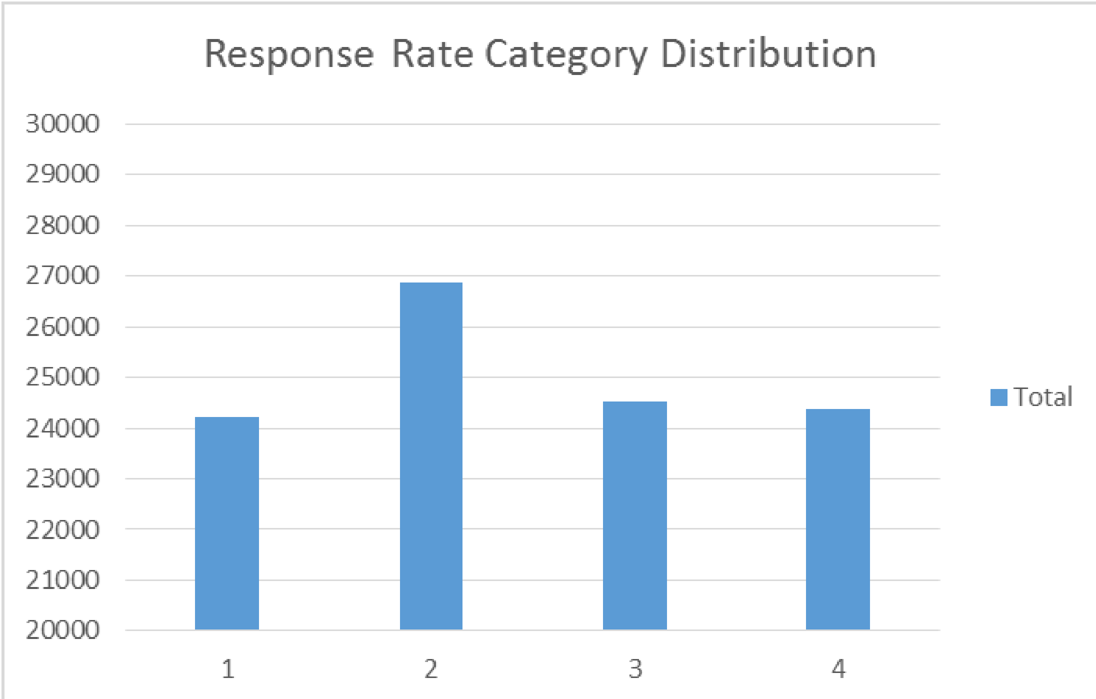

Figure 4-2: Response Rate Category data (quartiles)

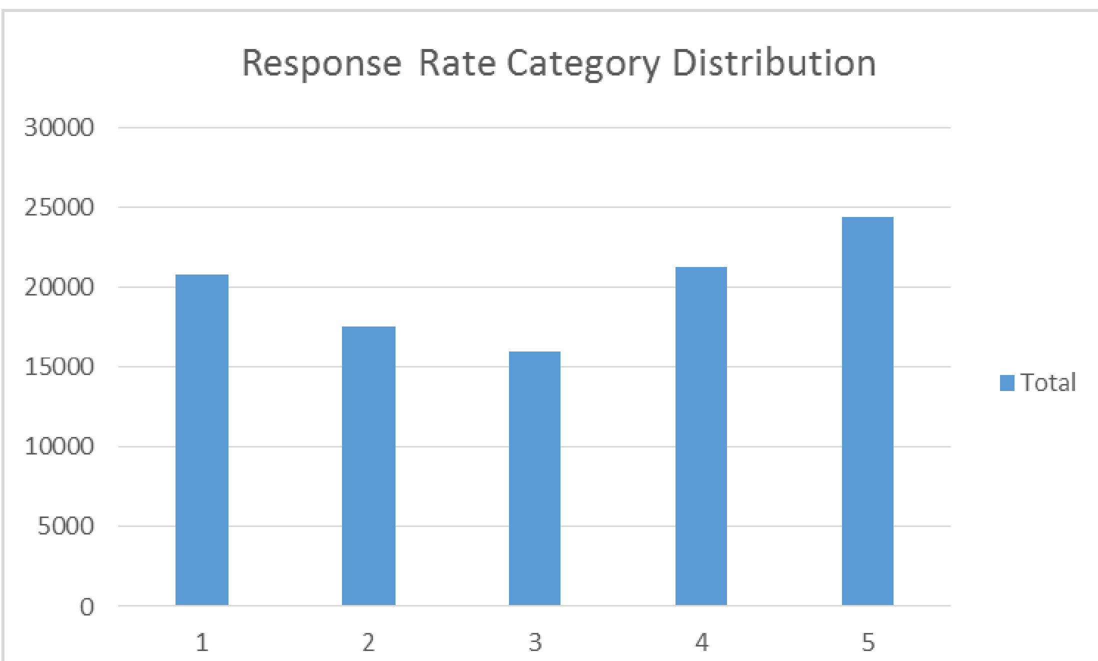

Figure 4-3: Response Rate Category data (quantiles)

Figures 4-2 and 4-3 show the response rate category assignment for both quartile and quantiles, respectively. Both demonstrate balance across the response rate category output class which is important for classifier algorithm performance. The C4.5 classifier algorithm is found in Weka under the title of 'J48'. The algorithm was initially run with parameter defaults (confidence factor = 0.25, minNumObj = 2), and testing options of 80% and 90% training data as well as 10 fold cross validation were selected. Results are listed in Table 4-2. Classification accuracy was significantly better on the quartile data, and therefore further analysis was performed on this data.

Table 4-2: Initial results with both Quartile and Quantile Data

	Quartile Data	**Quantile Data**
Algorithm Learning Method	**Correctly classified instances (%)**	**Correctly classified instances (%)**
80% training/20% test	77.32	67.52
90% training/10% test	77.41	67.21
10 fold cross validation	77.08	67.07

Next, the Weka Experimenter tool was used to further experiment with algorithm tuning. This tool provides the ability to run our algorithm with various parameter values, as outlined in Tables 4-3, 4-4, and 4-5.

Table 4-3: Experimenter Data based on a 10-fold cross validation.

No	Confidence Factor	Minimum Number of Objects	% Correctly Classified Instances
1	0.25	2	77.08
2	0.25	10	77.08
3	0.25	20	77.08
4	0.25	100	77.08
5	0.24	2	77.08
6	0.24	10	77.08
7	0.24	20	77.08
8	0.24	100	77.08
9	0.23	2	77.08
10	0.23	10	77.08
11	0.23	20	77.08
12	0.23	100	77.08
13	0.20	2	77.08
14	0.20	10	77.08
15	0.20	20	77.08
16	0.20	100	77.07
17	0.18	2	77.08
18	0.18	10	77.08
19	0.18	20	77.08
20	0.18	100	77.08

Table 4-4: Experimenter Data based on 80% training data, with holding 20% for testing.

No	Confidence Factor	Minimum Number of Objects	% Correctly Classified Instances
1	0.25	2	77.28
2	0.25	10	77.28
3	0.25	20	77.28
4	0.25	100	77.25
5	0.24	2	77.29
6	0.24	10	77.29
7	0.24	20	77.29
8	0.24	100	77.25
9	0.23	2	77.28
10	0.23	10	77.28
11	0.23	20	77.28
12	0.23	100	77.25
13	0.20	2	77.29
14	0.20	10	77.29
15	0.20	20	77.29
16	0.20	100	77.25
17	0.18	2	77.29
18	0.18	10	77.29
19	0.18	20	77.29
20	0.18	100	77.25

Table 4-5: Experimenter Data based on 90% training data, with holding 10% for testing.

No	Confidence Factor	Minimum Number of Objects	% Correctly Classified Instances
1	0.25	2	77.27
2	0.25	10	77.27
3	0.25	20	77.27
4	0.25	100	77.23
5	0.24	2	77.27
6	0.24	10	77.27
7	0.24	20	77.27
8	0.24	100	77.23
9	0.23	2	77.27
10	0.23	10	77.27
11	0.23	20	77.27
12	0.23	100	77.23
13	0.20	2	77.34
14	0.20	10	77.34
15	0.20	20	77.34
16	0.20	100	77.29
17	0.18	2	77.34
18	0.18	10	77.34
19	0.18	20	77.34
20	0.18	100	77.29

Although each of the methods of cross validation, 80/20 and 90/10 testing/training data splits produce similar results in terms of classification accuracy, a split using 90% of the data for training and 10% of our data for testing our model produces the highest classification accuracy. Neither lowering the confidence factor value below 0.2 nor increasing the minimum number of objects above 2 did not yield improved classification accuracy. Based on these results, we explore and tune the classifier further using C = 0.20 and M = 2 (values that provided our highest classification accuracy as seen in Table 4-4).

We use 90% of the data to train our classifier leaving 10% for testing (10,000 instances). The classifier was re-run both including and excluding the CDS type. Significant improvements were not observed in terms of classification accuracy; however the output is more easily interpretable in the absence of CDS type given the reduced complexity of the tree itself. These results are displayed in Tables 4-6, 4-7 and 4-8 below.

Table 4-6: J48 Classifier Performance Summary

Correctly Classified Instances	7745	77.45%
Incorrectly Classified Instances	2255	22.55%
Total Number of Instances	10000	

Table 4-7: Detailed Accuracy by Class (DT Classifier)

Class	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area
1	0.954	0.076	0.802	0.954	0.872	0.831	0.971	0.902
2	0.560	0.054	0.789	0.560	0.655	0.571	0.895	0.730
3	0.707	0.114	0.667	0.707	0.696	0.583	0.906	0.702
4	0.896	0.056	0.839	0.896	0.867	0.822	0.979	0.924
Weighted Average	0.775	0.075	0.775	0.775	0.768	0.699	0.937	0.813

Table 4-8: Confusion Matrix (DT Classifier)

a	b	c	d	classified as
2325	101	10	1	a = 1
562	1493	601	8	b = 2
10	293	1726	413	c = 3
1	5	250	2201	d = 4

This tuned model gives us 77.45% correctly classified instances. This was our best performing model in terms of accuracy.

Another metric used to examine classifier performance is the Receiver Operating Characteristic (ROC) Curve which uses a two dimensional graph whereby the ‘true positive rate’ is plotted on the Y-axis and the ‘false positive rate' is plotted against the X=axis [80, 84]. In this case, the point (0,1) will represent a perfect classifier, meaning a curve that visually hugs the upper left corner of the graph represents a strong classifier. ROC curves for each of our output classes based on this model are shown below (Figures 4-5 through 4-8). Classifier performance was relatively strong overall, with particularly high ROC areas for Response Rate Categories 1 and 4.

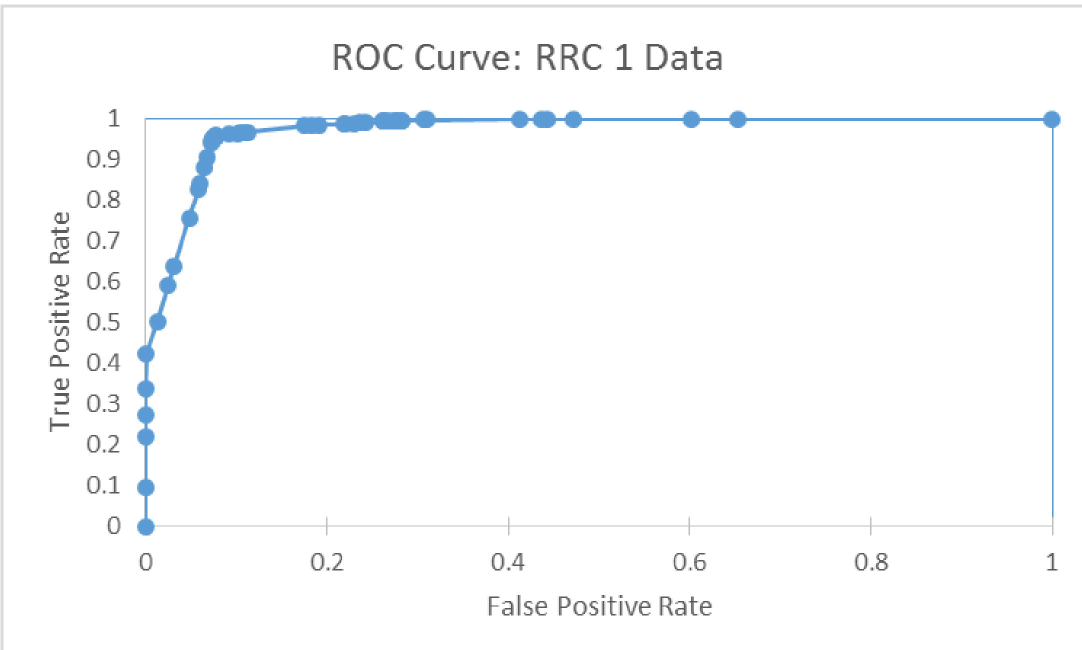

Figure 4-5: ROC Curve demonstrating classifier accuracy for Response Rate Category = 1 (0-32% acceptance rate).

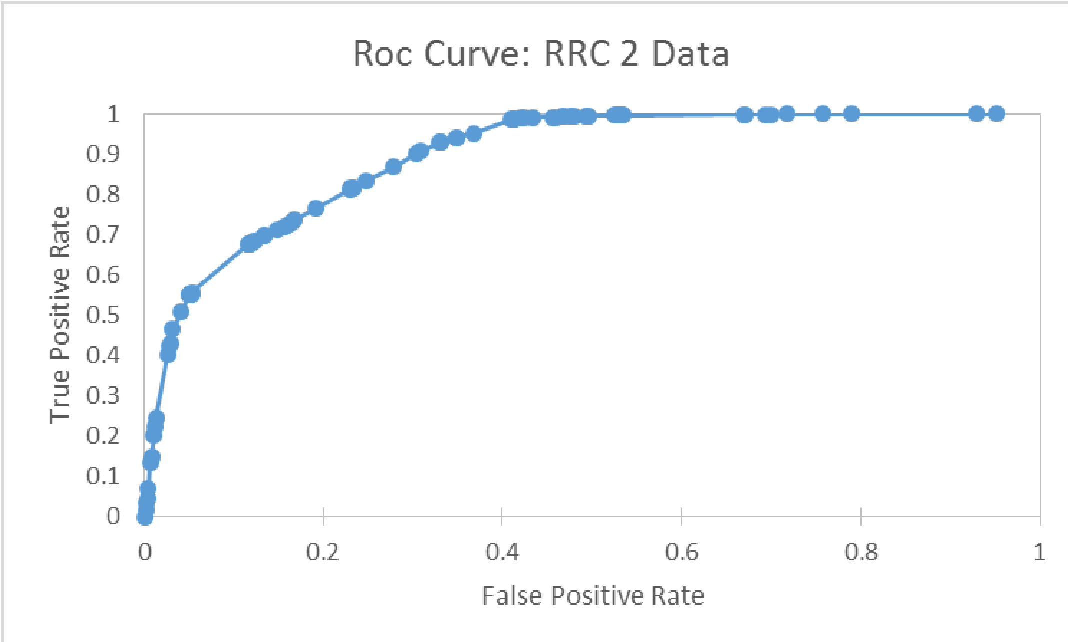

Figure 4-6: ROC Curve demonstrating classifier accuracy for Response Rate Category = 2 (33-46% acceptance rate).

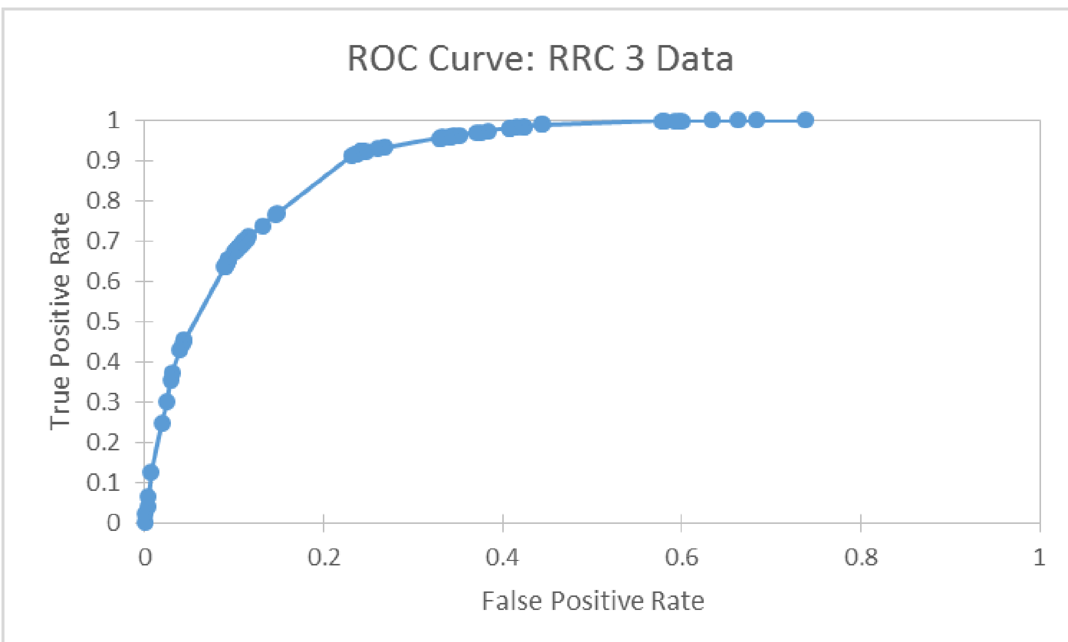

Figure 4-7: ROC Curve demonstrating classifier accuracy for Response Rate Category = 3 (47-59% acceptance rate).

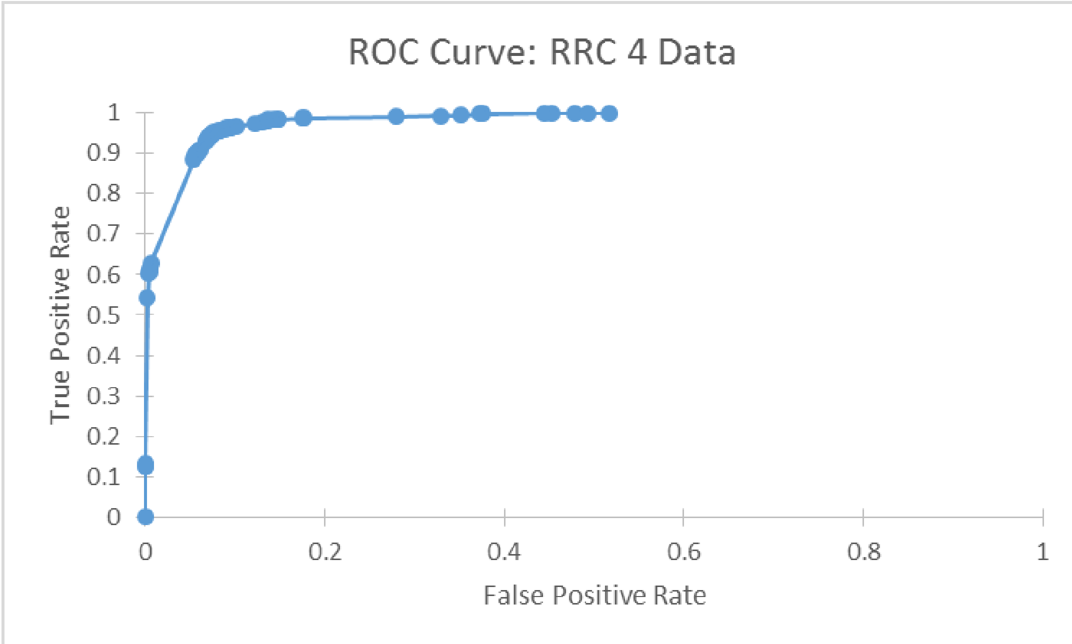

Figure 4-8: ROC Curve demonstrating classifier accuracy for Response Rate Category = 4 (60-100% acceptance rate).

A diagonal line (y=x) represents the strategy of randomly guessing a class, whereas a value in the lower right hand quadrant of the graph is indicative of a classifier that performs worse than guessing [84]. ROC curves are frequently used in classification evaluation due, in part, to their ability to visualize and compare classifier performance. Another attractive feature is their insensitivity to changes in class distribution [84]. In reviewing Figures 4-4 through 4-8, we note that in all cases, our classifier performs well in that all of our points are higher than the y=x line. The AUC is greater when RRC = 1 and RRC = 4, representing improved accuracy for our classifier when assigning classes to these categories.

The confusion matrix (Table 4-8) is another helpful tool in terms of interpreting results. Here we can see additional detail in terms of the number of correctly and incorrectly classified instances and which output class, specifically, they were assigned to. For RRC = 1, we see that for n = 2437, 2325 instances were correctly classified, while 112 were incorrectly assigned to RRC = 2 (101 instances), RRC = 3 (10 instances) or RRC = 4 (1 instance). As we would expect based on our ROC analysis above, for RRC = 2, we see poorer classification performance with only 1493 correctly classified instances out of 2664 in total. Similarly, for RRC = 3 we see a slight improvement (ROC area = 0.906) with 1726 out of 2442 correctly classified instances. Our confusion matrix in this case shows only 6 instances incorrectly classified to RRC = 1 or RRC = 2, and 250 instances incorrectly assigned to RRC = 3. We acknowledge there is room for classifier improvement; but are pleased with our classifier performance over all.

A clear advantage of the C4.5 classifier is the ability to generate a visualization of the Decision Tree. Figures 4-9 and 4-10 show a small subsection of the visualization of the decision tree generated by our model. With 73 leaves, it proves difficult to include the full visualization of the tree in a static document and the figures here are included to demonstrate to the reader the output of this algorithm, and the ease to which this can be shared and explored electronically. Appendix D includes the detailed decision tree and associated rules generated by our model.

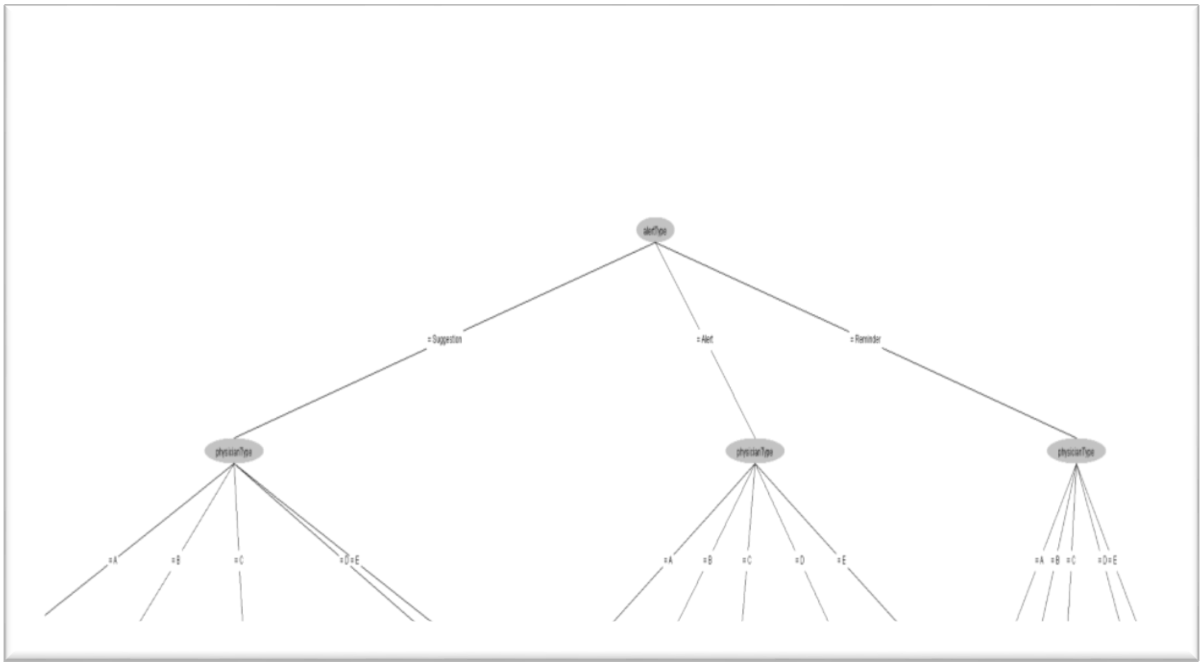

Figure 4-9: Top nodes of the decision tree generated by the J48 classifier. Although difficult to read, this image illustrates the concept of the lead-node splitting.

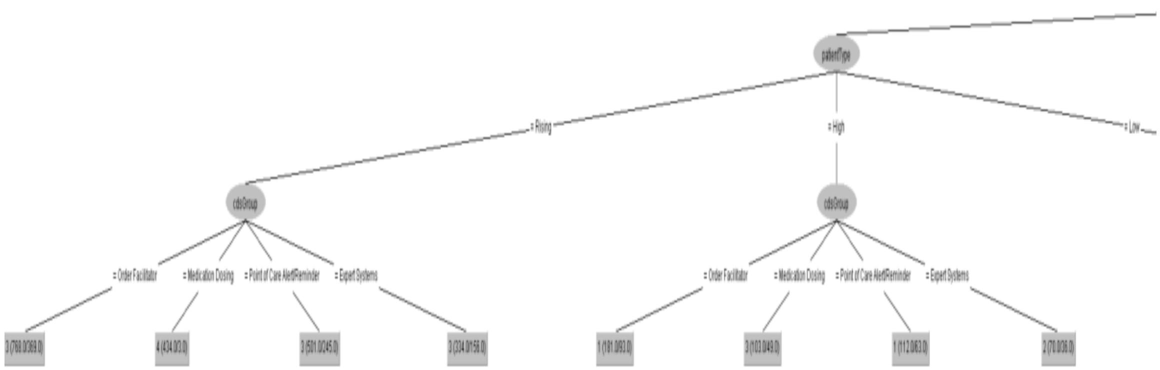

Figure 4-10: Second layer nodes of the decision tree generated by the J48 classifier. This image shows how the tree branches after physician type = A, splitting next on patient type.

4.1.2 Decision Tree: Interpretation of Results

Close examination of the model's output (see Appendix D) permits the following interpretations. First, alert severity is not considered in leaf development. This makes sense given that all 'Alerts' were classified as Severity 1, all 'Suggestions' were classified as Severity 2 and all 'Reminders' except three of the point of care alerts/reminders were classified as Severity 2. Our top level node is 'Alert Type'.

Our next observation has to do with the 'Reminders' alert type. This portion of the tree is fairly simple, given that CDS group is not important – rather the acceptance rate category for each physician type is based solely on patient type. We note that physician types B and E are the least likely of all physician types to respond to a reminder type alert. On the contrary, physician types A and C respond to most reminders, whereas D falls somewhere in the middle, although is more likely to respond to reminders for high risk patients (acceptance rate of 47-60%).

When we look at 'Suggestions', we note that physician types B and E almost always respond to suggestions (61-100% acceptance rate) regardless of patient type or CDS group. Physicians of type D do not respond well to suggestions except those related to medication dosing (and even those they respond to only ~50% of the time), whereas physicians of type C are likely to respond well to suggestions related to medication dosing or those related to high risk patient types. Physicians of type A respond well to suggestions when patient type is rising risk or the suggestion relates to medication dosing.

For the alert type category of ‘Alerts’, we note that all physician types respond well to medication dosing alerts (all physician groups show a 47-100% acceptance rate). Type C physicians are least likely to respond to alerts from the ‘Order Facilitator’ group, regardless of patient type. Similarly, Type E physicians show low acceptance rates for point of care alerts/reminders regardless of patient type.

The output of this model provides us with insight in to how physicians of the various groups are likely to respond when faced with these types of alerts for particular patient populations. Insights into how a physician is likely to respond can help inform decisions – for example, given that physicians from groups B and E are unlikely to respond to reminder alerts, why not prevent these non-urgent reminders from appearing to these physician groups at all? Physician types B and E do respond well to suggestions, though, so we could ensure that those alert types always display for these groups. With the alert type of ‘Alerts’, we note that all physician groups respond well to medication dosing alerts; however differences exist for other CDS groups. We could consider removing order facilitator alerts for Type C physicians, and perhaps removing point of care alerts/reminders for physicians categorized as group E. In doing so, we reduce alerts that we are quite confident will be ignored based on our analysis, thereby reducing the ‘noise’ and consequently contributing to a reduction in alert fatigue. Application of these findings, including the development of an alert issuance strategy, is explored in detail in Chapter 5.

4.3 Multilayer Perceptron Classification

As with the Decision Tree classification discussed previously, the first step was to load the simulated data into Weka. The response rate categories used were based on the quartiles given the analysis and experimentation already completed and discussed in Section 4.1.1 above. The available parameters for the MLP algorithm are listed in Appendix E. As with other classification algorithms, the tuning process requires some experimentation. Of note is the ability to specify the number of hidden layers. Witten et al. suggest that a single hidden layer is often sufficient [41]; however identifying the appropriate number of units for that layer requires experimentation in order to maximize the estimation accuracy of the classifier. The default hidden layer value in Weka is 'a', which is defined as:

$$\frac{\text{total number of attributes + the total number of classes}}{2}$$

The 'training time' allows the user to specify the number of epochs to train through. The default value is 500. Some preliminary tests were run with 300, 400, 500, 600 and 700 epochs; however the classification accuracy was not significantly different with any of these options and therefore all further testing was performed with epochs = 500. The learning rate specifies the amount by which the weights are updated, and the momentum provides an opportunity to adjust the momentum applied to the weights during updating. The default learning rate is set to 0.3, and the default momentum to 0.2.

Table 4-9 lists the experimentation performed with the MLP classifier. A variety of training methods were employed including using 70%, 80% and 90% of the data for training (and the remainder for testing), as well as ten-fold cross validation. We also adjusted the number of hidden layers, the learning rate and the momentum in an attempt to find the best model.

Table 4-9: MLP Experimentation Parameters and Results

Training type	# of Hidden Layer Units	Learning Rate	Momentum	Classification Accuracy
90/10	11	0.1	0.2	77.54
90/10	11	0.01	0.2	77.49
90/10	11	0.2	0.2	77.43
90/10	11	0.3	0.2	77.36
90/10	7	0.3	0.2	75.58
90/10	6	0.2	0.2	74.63
90/10	5	0.2	0.2	74.44
90/10	3	0.3	0.2	61.89
90/10	2	0.3	0.2	57.46
80/20	11	0.3	0.2	76.99
80/20	3	0.3	0.2	65.07
70/30	11	0.1	0.2	77.17
70/30	11	0.3	0.2	77.12
70/30	11	0.25	0.2	77.1
70/30	11	0.3	0.2	77.09
70/30	2	0.3	0.2	57.46
70/30	1	0.3	0.2	41.48
10 fold cross validation	11	0.3	0.2	76.67
10 fold cross validation	4	0.3	0.2	71.01
10 fold cross validation	2	0.3	0.2	59.94
10 fold cross validation	1	0.3	0.2	48.49

The first model listed in Table 4-9 provided the best classification accuracy, at 77.54% accuracy – just slightly higher than our best performing Decision Tree model (Figure 4-6). Additional details for this MLP model (Hidden

Layer = a, Learning rate = 0.1, Momentum = 0.2), which was run against 90% of the training data, can be seen in Tables 4-10. 4-11 and 4-12.

Table 4-10: MLP Classifier Accuracy

Correctly Classified Instances	7754	77.54%
Incorrectly Classified Instances	2246	22.46%
Total Number of Instances	10000	

Table 4-11: Detailed Accuracy by Class (MLP Classifier)

Class	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area
1	0.957	0.077	0.800	0.957	0.871	0.831	0.970	0.910
2	0.557	0.052	0.796	0.557	0.655	0.574	0.895	0.743
3	0.705	0.112	0.670	0.705	0.687	0.583	0.907	0.715
4	0.902	0.058	8.360	0.902	0.868	0.823	0.978	0.940
Weighted Average	0.775	0.074	0.776	0.775	0.768	0.700	0.937	0.825

Table 4-12: Confusion Matrix (MLP Classifier)

a	b	c	d	classified as
2333	93	9	2	a = 1
567	1484	603	10	b = 2
15	283	1722	422	c = 3
2	4	236	2215	d = 4

Similar to the best performing Decision Tree model, examination of the ROC curve indicates that our MLP model performs best in correctly assigning test instances to Response Rate Category 1 (0-32% acceptance) and Category 4 (61-100% acceptance). ROC curve visualizations can be seen in Figures 4-11 through 4-14.

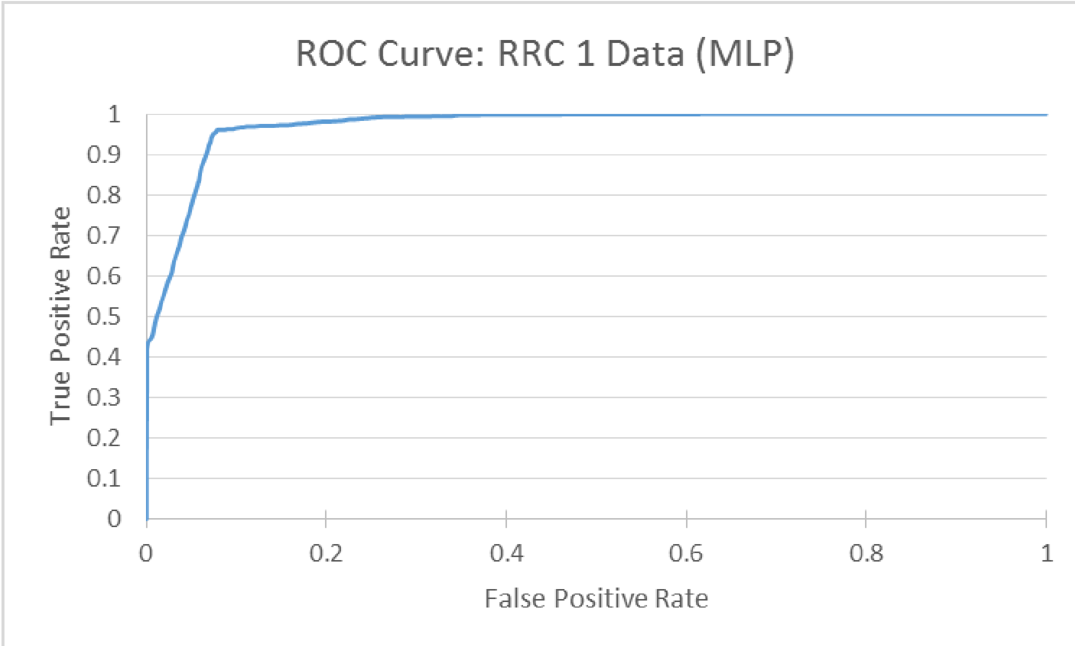

Figure 4-11: ROC Curve demonstrating MLP classifier accuracy for Response Rate Category = 1 (0-32% acceptance rate).

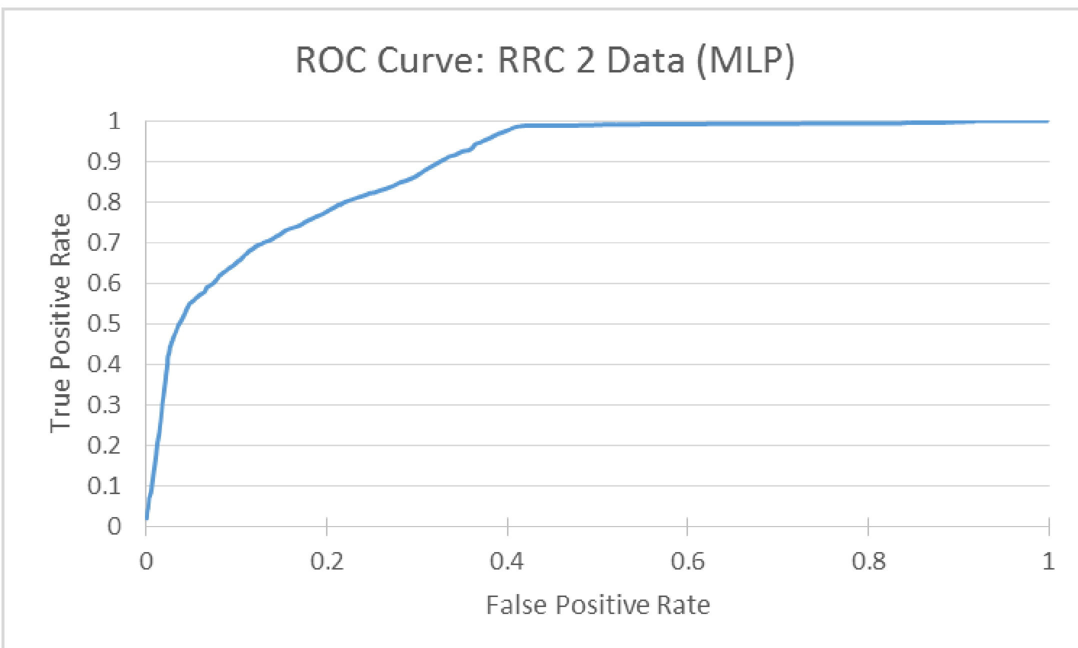

Figure 4-12: ROC Curve demonstrating MLP classifier accuracy for Response Rate Category = 2 (33-47% acceptance rate).

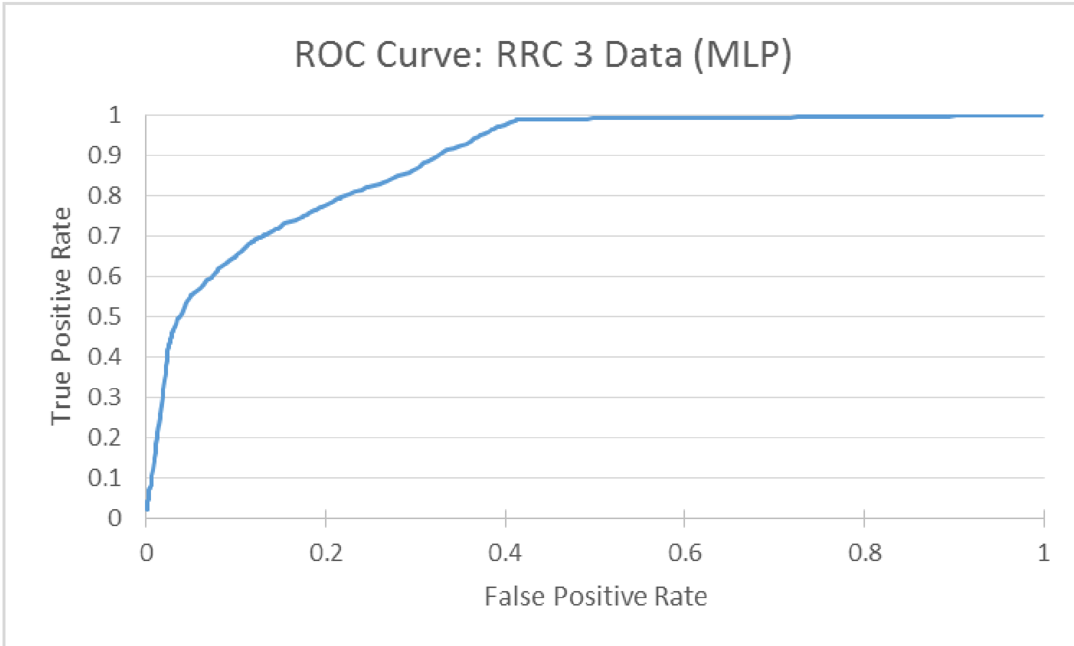

Figure 4-13: ROC Curve demonstrating MLP classifier accuracy for Response Rate Category = 3 (47-59% acceptance rate).

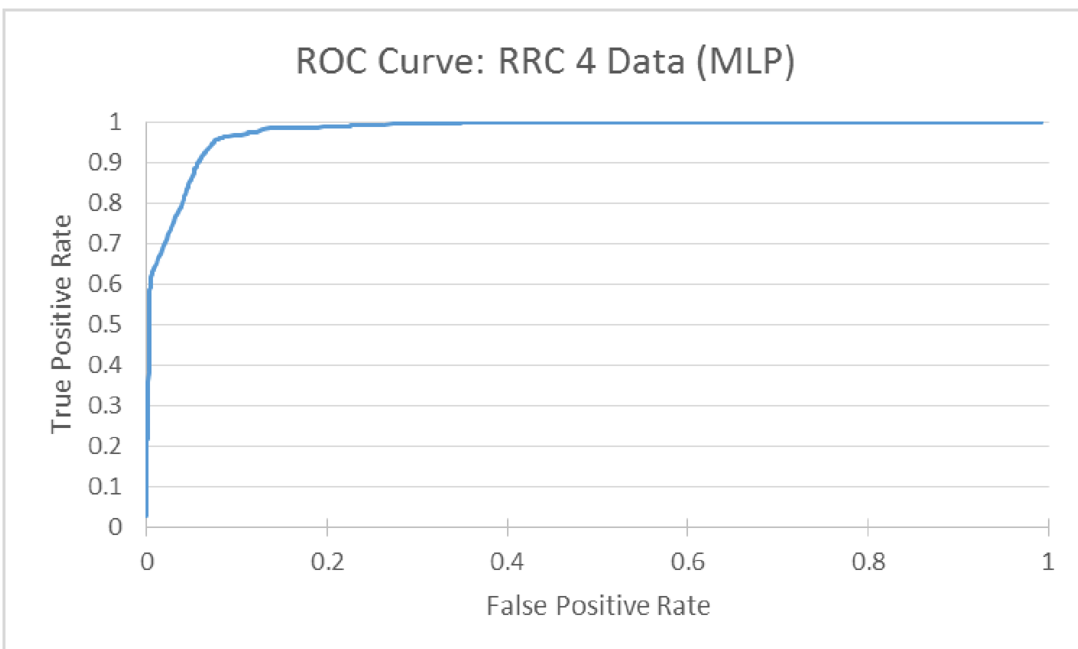

Figure 4-14: ROC Curve demonstrating MLP classifier accuracy for Response Rate Category = 4 (60-100% acceptance rate).

4.3.1 Multilayer Perceptron: Interpretation

While classification accuracy was slightly better with the MLP model, our ROC area values and confusion matrix outputs were almost identical. Figure 4-15 and 4-16 show side by side comparisons of our DT and MLP outputs values for ROC area and our confusion matrix, respectively.

Class	ROC Area (DT)	ROC Area (MLP)
1	0.971	0.970
2	0.895	0.895
3	0.906	0.907
4	0.979	0.978
Weighted Average	0.937	0.937

Figure 4-15: ROC Area comparison for DT and MLP classifiers

Confusion Matrix (DT)

a	b	c	d	classified as
2325	101	10	1	a = 1
562	1493	601	8	b = 2
10	293	1726	413	c = 3
1	5	250	2201	d = 4

Confusion Matrix (MLP)

a	b	c	d	classified as
2333	93	9	2	a = 1
567	1484	603	10	b = 2
15	283	1722	422	c = 3
2	4	236	2215	d = 4

Figure 4-16: Confusion Matrix comparison for DT and MLP classifiers

One notable disadvantage of this method is what has been described as the ‘black box’ processing of this algorithm, meaning that it’s more difficult to understand or interpret the output of an MLP classifier given the absence of a decision tree or rule list that is favored by domain experts for its ease of interpretation [41].

4.4 Chapter Summary

In this chapter we have discussed our data generation methods. We have successfully leveraged this data to develop predictive models using both Decision Tree and MLP classification techniques. Experimentation details for our DT and MLP models and the tuning techniques utilized in both cases have been described in detail. We chose to work with two different classification algorithms to determine whether we would achieve improved performance with different modeling techniques. While our classification performance was similar across both techniques, these results are also reassuring as they suggest that we have likely achieved maximum classification accuracy based on our simulated data set.

In the next chapter we will further discuss our findings, and how these classification models can be used to adapt alert issuance based on physician practice and alert response behavior in order to minimize alert fatigue.

CHAPTER 5: CDSS ALERT ISSUANCE STRATEGIES

In this chapter we will explore strategies for reducing alert fatigue based on the predictive models developed in Chapter 4. We will review the output of our DT model in greater detail, for each CDSS alert type (Reminders, Suggestions, and Alerts) in order to propose changes with respect to alert issuance based on physician practice and alert response data in order to address our fourth and final research goal. We also intend to demonstrate the benefit of our proposed strategy by applying this to our simulated data for each alert type, and finally we'll discuss the implications of our strategy for each physician type and the overall implications.

5.1 Alert Issuance Strategy: Reminders

In order to contemplate strategies to reduce alert fatigue based on our classification models, closer examination of our model's predictions related to physician alert acceptance rates are required. Table 5-1 highlights our DT classifier's predictions related to 'Reminders'. Based on our model, all 'reminder' alert types of from the CDS Group 'Point of Care Alerts/Reminders', and as a result, the CDS group is not a contributor to acceptance rate for these alert types.

Table 5-1: Predicted Physician Response to 'Reminder' Alerts

Physician Type	Patient Type	Acceptance Rate Category
A	Rising	4
A	High	3
A	Low	3
B	Rising	1
B	High	1
B	Low	1
C	Rising	4
C	High	4

Physician Type	Patient Type	Acceptance Rate Category
C	Low	3
D	Rising	3
D	High	3
D	Low	2
E	Rising	1
E	High	1
E	Low	1

We note that physicians in groups B and E are least likely to respond to reminder alert types, compared to physicians in groups A and C who are most likely to respond to CDSS reminder alerts. Physicians from group D are most likely to respond to reminders when patient type is equal to ‘Rising’ or ‘High’ risk.

We can directly apply the learnings from our model to adjust alert issuance to better support physicians based on their alert response behavior, clinical schedule, alert acuity and patient case-mix. In this case, we would suggest blocking reminder alert types for physicians of groups B and E. For Physician Group D, we could adjust the CDSS reminder alerts and only present reminders when patient type is equal to ‘Rising’ or ‘High’. If we apply this strategy to our simulated data for these physician groupings, we see a reduction in issuance of ‘Reminder’ alerts from 11,497 (total number of reminder alerts issued across all physician types) out of a total of 100,000 to 4,470 ‘Reminder’ alerts – a percentage decrease in ‘Reminder’ alerts issued across all physician groupings of approximately 62%.

5.2 Alert Issuance Strategy: Suggestions

The output of our classification model for the CDSS alert type of 'Suggestions' is detailed in Table 5-2. In this case, we see that CDS Group is an important contributor to alert acceptance rate categories.

Table 5-2: Predicted Physician Response to 'Suggestion' Alerts

Physician Type	Patient Type	Acceptance Rate Category	Physician Type
A	Rising	Order Facilitator	3
A	Rising	Medication Dosing	4
A	Rising	Point of Care Alert/Reminder	3
A	Rising	Expert System	3
A	High	Order Facilitator	1
A	High	Medication Dosing	3
A	High	Point of Care Alert/Reminder	1
A	High	Expert System	2
A	Low	Order Facilitator	1
A	Low	Medication Dosing	2
A	Low	Point of Care Alert/Reminder	1
A	Low	Expert System	1
B	Rising	Order Facilitator	4
B	Rising	Medication Dosing	4
B	Rising	Point of Care Alert/Reminder	4
B	Rising	Expert System	4
B	High	Order Facilitator	4
B	High	Medication Dosing	4
B	High	Point of Care Alert/Reminder	4
B	High	Expert System	4
B	Low	Order Facilitator	4
B	Low	Medication Dosing	4
B	Low	Point of Care Alert/Reminder	4
B	Low	Expert System	4

Physician Type	Patient Type	Acceptance Rate Category	Physician Type
C	Rising	Order Facilitator	2
C	High	Order Facilitator	3
C	Low	Order Facilitator	1
C	Rising	Medication Dosing	3
C	High	Medication Dosing	4
C	Low	Medication Dosing	2
C	Rising	Point of Care Alert/Reminder	1
C	High	Point of Care Alert/Reminder	2
C	Low	Point of Care Alert/Reminder	1
C	Rising	Expert System	2
C	High	Expert System	3
C	Low	Expert System	1
D	Rising	Order Facilitator	1
D	High	Order Facilitator	2
D	Low	Order Facilitator	1
D	Rising	Medication Dosing	3
D	High	Medication Dosing	3
D	Low	Medication Dosing	2
D	Rising	Point of Care Alert/Reminder	1
D	High	Point of Care Alert/Reminder	1
D	Low	Point of Care Alert/Reminder	1
D	Rising	Expert System	1
D	High	Expert System	1
D	Low	Expert System	1
E	Rising	Order Facilitator	4
E	Rising	Medication Dosing	4
E	Rising	Point of Care Alert/Reminder	4
E	Rising	Expert System	4
E	High	Order Facilitator	4
E	High	Medication Dosing	4
E	High	Point of Care Alert/Reminder	4
E	High	Expert System	4

Physician Type	Patient Type	Acceptance Rate Category	Physician Type
E	Low	Order Facilitator	4
E	Low	Medication Dosing	4
E	Low	Point of Care Alert/Reminder	4
E	Low	Expert System	4

Based on the detailed outputs of our model, we propose the following strategy to adjust alert issuance: we recommend no changes to alert issuance for physician groups B and E given that they almost always respond to suggestion alerts. For Physician Group A, we recommend suppressing all suggestion alerts except where patient type is equal to 'Rising Risk', or the CDS group is equal to medication dosing. Physicians in group C respond well to CDSS alerts where CDS group is equal to medication dosing, or where patient type is equal to 'High Risk'. In this case, we recommend suppressing all other suggestions. For those in group D, we propose suppressing all suggestion alerts with the exception those from the medication dosing CDS group for patients of 'High' and 'Rising' risk. If we apply these rules to our simulated data, we see a reduction from 50,163 out of 100,000 suggestion alerts to 20,663 suggestion alerts (across all physician types). This is a percent decrease of close to 60%.

5.3 Alert Issuance Strategy: Alerts

Next, we examine the output of our classification models for the CDSS alert types of 'Alert' more closely. Table 5-3 details our DT model's prediction of alert acceptance rate category for each physician group.

Table 5-3: Predicted Physician Response to 'Alerts'

Physician Type	Patient Type	CDS group	Response Rate Category
A	Rising	Order Facilitator	4
A	High	Order Facilitator	3
A	Low	Order Facilitator	2
A	Rising	Medication Dosing	4
A	High	Medication Dosing	3
A	Low	Medication Dosing	3
A	Rising	Point of Care Alert/Reminder	4
A	High	Point of Care Alert/Reminder	2
A	Low	Point of Care Alert/Reminder	2
A	Rising	Expert Systems	4
A	High	Expert Systems	3
A	Low	Expert Systems	2
B	Rising	Order Facilitator	3
B	High	Order Facilitator	3
B	Low	Order Facilitator	3
B	Rising	Medication Dosing	3
B	High	Medication Dosing	3
B	Low	Medication Dosing	3
B	Rising	Point of Care Alert/Reminder	3
B	High	Point of Care Alert/Reminder	3
B	Low	Point of Care Alert/Reminder	3
B	Rising	Expert Systems	3
B	High	Expert Systems	3
B	Low	Expert Systems	3
C	Rising	Order Facilitator	1
C	High	Order Facilitator	1
C	Low	Order Facilitator	1
C	Rising	Medication Dosing	3
C	High	Medication Dosing	4
C	Low	Medication Dosing	3

Physician Type	Patient Type	CDS group	Response Rate Category
C	Rising	Point of Care Alert/Reminder	2
C	High	Point of Care Alert/Reminder	3
C	Low	Point of Care Alert/Reminder	1
C	Rising	Expert Systems	1
C	High	Expert Systems	1
C	Low	Expert Systems	1
D	Rising	Order Facilitator	3
D	High	Order Facilitator	3
D	Low	Order Facilitator	3
D	Rising	Medication Dosing	4
D	High	Medication Dosing	4
D	Low	Medication Dosing	4
D	Rising	Point of Care Alert/Reminder	3
D	High	Point of Care Alert/Reminder	4
D	Low	Point of Care Alert/Reminder	3
D	Rising	Expert Systems	3
D	High	Expert Systems	3
D	Low	Expert Systems	3
E	Rising	Order Facilitator	2
E	High	Order Facilitator	2
E	Low	Order Facilitator	2
E	Rising	Medication Dosing	3
E	High	Medication Dosing	3
E	Low	Medication Dosing	3
E	Rising	Point of Care Alert/Reminder	2
E	High	Point of Care Alert/Reminder	2
E	Low	Point of Care Alert/Reminder	2
E	Rising	Expert Systems	2
E	High	Expert Systems	2
E	Low	Expert Systems	2

Physicians in group B and D respond well to 'alert' types. For these physicians, we would not recommend any change in alert issuance for CDSS alerts classified as 'alert'. For physicians in group C, our strategy would be to remove all 'point of care alerts/reminders', where patient type is equal to 'low' or 'rising' risk given the low acceptance rates. For physicians in group E, we suggest removing all 'point of care alerts/reminders' as they are actioned less than half of the time. Physicians in group A show very low acceptance rates for 'point of care alerts/reminders' when patient type is equal to 'High' risk. For these reasons, we suggest altering alert issuance to prevent presentation of these alert types to these physicians. Applying these strategies to the simulated alert response data would result in a decrease from 38,340 (total number of 'alert' type CDSS alerts issued across all physician groups) out of 100,000 alters issued, to 23,506 alerts - a decrease of approximately 40%.

5.4 Alert Issuance Reduction: Overall Approach and Impact

In this section we examine the impact of the changes we are proposing to alert issuance, and the impact that approach would have on each physician group. It's important to note, that our approach focuses not just on alert reduction; but on delivering alerts that are most meaningful to a physician based on their practice, and are therefore most likely to be appropriately actioned by each physician type.

5.4.1 Alert Issuance: Overall Strategy

The procedural rules that we propose based on our DT classification model, in order to implement our alert issuance strategy are described in Table 5-4 below. These rules were generated manually, upon careful review of our DT classification model's output.

Table 5-4: Proposed Alert Issuance Strategy – Procedural Rules

Rule	Description
1	If Physician Type = A AND Alert Type = Suggestion BLOCK ALL ALERTS except where Patient Type = 'Rising' OR CDS Group = 'Medication Dosing'
2	If Physician Type = A AND Alert Type = Alert BLOCK ALL ALERTS where CDS Group = 'Point of Care Alert/Reminder' AND Patient Type = 'High'
3	If Physician Type = B AND Alert Type = Reminder BLOCK ALL ALERTS
4	If Physician Type = C AND Alert Type = Suggestion BLOCK ALL ALERTS except where CDS Group = 'Medication Dosing' OR Patient Type = 'High'
5	If Physician Type = C AND Alert Type = Alert BLOCK ALL ALERTS where CDS group = 'Point of Care Alerts/Reminders' AND Patient Type = 'Low'
6	If Physician Type = C AND Alert Type = Alert BLOCK ALL ALERTS where CDS group = 'Point of Care Alerts/Reminders' AND Patient Type = 'Rising'
7	If Physician Type = D AND Alert Type = Reminder BLOCK ALL ALERTS except where Patient Type = 'Rising' OR Patient Type = 'High'
8	If Physician Type = D AND Alert Type = Suggestion BLOCK ALL ALERTS except where CDS Group = 'Medication Dosing' AND Patient Type = 'High' OR CDS Group = 'Medication Dosing' and Patient Type = 'Rising'
9	If Physician Type = E AND Alert Type = Reminder BLOCK ALL ALERTS
10	If Physician Type = E AND Alert Type = Alert BLOCK ALL ALERTS where CDS Group = 'Point of Care Alerts/Reminders'

Table 5-5 demonstrates the impact of our proposed changes to alert issuance based on our simulated data.

Table 5-5: Physician Alert Issuance Data – Pre and Post Intervention (includes all Reminders, Suggestions and Alert CDSS alert types)

Physician Type	# of Alerts Issued Pre-Intervention	# of Alerts Issued Post-Intervention	% Reduction
A	19,977	9,645	48%
B	19,987	12,014	60%
C	20,181	8,860	44%
D	20,021	11,984	60%
E	19,834	6,436	32%

Through implementation of our alert issuance strategy, we are able to achieve a significant reduction in the volume of alerts presented across each physician type. Additionally, we are providing a more personalized method of alert delivery– delivering specific alerts to each physician type based on their personalized response data based on patient and practice characteristics. This method allows us to discontinue presentation of alerts that we are confident will be ignored based on the machine learning experimentation conducted and discussed in Chapter 4.

Certainly, there is an implication to suppressing alerts of any type. Our proposed alert strategy requires continuous review and engagement with clinical end users. For example, for the groups for which alerts were suppressed, some of the users in those physician groups would have been responding to those alerts. So an assessment of what the impact of suppressing those alerts is will be necessary. Is it that the physician continues to perform an action in the absence of the alert? Additionally, through engagement with physicians and ongoing monitoring of alert response data, we may determine that changes are required to our alert classification framework in order to support increased delivery of more important, relevant alerts.

5.5 Chapter Summary

In this chapter we have discussed an innovative CDSS alert issuance strategy to address current challenges with alert fatigue in the primary care setting developed based on our classification models. We have demonstrated a more meaningful, personalized alert issuance approach, which, based on our simulated data, shows a significant reduction in the volume of alerts delivered across all physician types.

CHAPTER 6: DISCUSSION

This work commenced with four clear objectives. In this chapter, we will discuss how we have addressed each of these objectives. We will also discuss our research contributions as well as some study limitations, and finally, we will highlight some opportunities for future work.

Our first research objective was to stratify physicians into distinct practice groups in order to design a group-level alert issuance strategy. To do this, we utilized well documented clustering algorithms and developed physician groupings based on various physician attributes. This provided us with an opportunity to later design group-level alert delivery strategies. We were able to utilize the k-means algorithm to cluster our physicians into five distinct groups based on both physician specific, as well as physician practice attributes. This was a crucial first step in our solution design.

Our second objective was to classify the wide range of CDSS alerts in terms of their source, acuity and response expectations for EMR- based alerts based on a review of current literature. As outlined in Chapters 2 and 3, a detailed literature review was conducted, and subsequently we were able to develop an alert classification. Our classification work was validated by a Nova Scotia-based primary care physician. This classification could be used by others researching CDSS alerts in the primary care setting. Additionally, while our classification was developed specifically for the primary care setting, the methods used for its development could be applied to develop a similar classification model for the acute care setting.

Our third goal was to establish a mapping between physician groups and alert types. To do this, we employed classification algorithms in order to develop predictive models that would help predict how certain physician ‘types’ would respond specific EMR alert types. We were able to generate physician response data, based on our five physician clusters and our EMR alert classification (n = 100,000). From there, we were able to successfully utilize DT and MLP algorithms to develop predictive alert acceptance rate models. While we would have liked to have seen a higher classification accuracy than what we were able to achieve (77.45%), we feel these models show promise and do meet our research objectives; however an important next step would be to work with real-world data and subsequently re-evaluate our models.

Finally, we were able to develop a strategy to issue alerts based on physician’s practice and alert response behavior in order to minimize alert fatigue. We were able to develop a strategy, and demonstrate its impact on alert reduction across all physician groups based on our simulated alert response data.

6.1 Research Contributions

Our final research objective highlights our contribution. This objective was to develop a strategy to issue CDSS alerts based on a physician’s practice (group) and alert response behavior. Understanding how certain groups of physicians will respond to certain CDS alerts provides us with an opportunity to present only the most relevant alerts for that particular physician group, thereby reducing the volume of alerts that will almost certainly be ignored based on the personalized response data. For example, our results show that physicians of group B and E have very low acceptance rates for reminders. In this case, alert issuance is modified such that reminder alerts no longer fire for physicians belonging to group B and E, thereby reducing the volume of what these groups likely consider to be unnecessary alerts. Similarly, we found that physicians of group D have very low alert acceptance rates for 'suggestion' alerts with the exception of medication dosing related suggestions. In this case, we suppress all 'suggestion' alert types from firing – again reducing the volume of unnecessary alerts thereby contributing to a reduction in alert fatigue. Modification of the CDSS alert issuance mechanisms could be adjusted programmatically based on our predictive models. Parameters could be set based on acceptance rates – for example, do not fire an alert where physician group acceptance of that alert is predicted to be less than 33% (Response Rate Category 1). Acceptance rates would need to continue to be monitored over time, and alert issuance parameters could be readjusted as required in an effort to further reduce alert fatigue.

Other aspects of this work also offer additional opportunities. For example, the physician clustering work could be used to evaluate physician educational opportunities or groups that may require additional CDSS related training. It may also identify groups who are suited to research initiatives based on patient and practice types. With physician engagement, there is a real opportunity to evaluate our alert issuance strategy and seek feedback for evaluation and improvement. Our strategy focused on alert suppression; but it could be that some physician groups could benefit from increased alert issuance based on

CDSS alert type. Ongoing monitoring and engagement would be an important aspect of this future work.

This work demonstrates a novel approach to contemplating CDSS alerts. To date, much of the CDSS-related research has focused on acute care settings and often on specific disease/patient populations. Our work focuses specifically on the primary care setting – an important area of research given the reported increased demands on physicians in this setting and therefore their susceptibility to alert fatigue [4, 5, 68]. In addition, this work focuses broadly on all patient types/diseases, an important distinction from most other work published to date. Our ability to establish a mapping between physician groups and alert types, and to be able to utilize this, coupled with actual response data and data mining techniques provides a meaningful tool that can be used to deliver more meaningful, personalized alerts to primary care physicians.

6.2 Study Limitations

There are a number of limitations of this study that should be noted. Firstly, while based on a portion of real, Nova Scotia-based physician data, other physician attributes used in our cluster modelling were added based on some documented workload [68] and patient characteristics [47]; but more focused research in this area is needed. Additionally, domain experts should be consulted to identify more scientifically supported attributes for consideration in physician ‘typing’. Obviously, this was a proof of concept, and our classification models were built on artificial, simulated data. A better approach, and recommended next step, would be to use real world physician alert response data. Unfortunately, that was not possible as part of the scope of this initiative.

Another important limitation to note, is that my research assumes that alerts presented are appropriate. We do not evaluate whether an alert *should* have fired or not. We also do not consider repeated alerts – meaning the same alert might present to a primary care physician repeatedly given that a number of patients would see their primary care physician regularly, have multiple comorbidities, so, as an example, a medication risk/reminder might fire at each patient visit. Perhaps the clinician accepted the alert the first time, and then ignored all subsequent alerts. We have no way of tracking this based on our simulated data and parameters of this study; but it is an important consideration that should be included in future work.

While our alert classification was reviewed by a family physician, this should be reviewed by a more robust, representative panel to confirm its validity. Additionally, it should be reviewed against a wide range of primary care EMR systems to confirm its applicability. While we did receive endorsement from the family physician we reviewed with, and do feel our classification covers most primary care CDSS alert types, some additional validation of this work would lend further credence to its future use.

6.3 Future Work

Our classification results show promise in terms of CDSS alert response predictability; however real world data is required to further test and to further improve classification accuracy of these algorithms. Opportunity exists to further explore physician grouping via clustering perhaps by partnering with a group of primary care physicians, and domain experts to improve upon the attributes included and the clustering techniques used. Leveraging real world alert response data will provide additional opportunities to further validate the concepts and results reported here. The classification models used in this research were purposely selected; however future research could explore other classification techniques as well. Once these models are implemented and validated against live data, there is almost certainly an opportunity to track physician responses in order to optimize and continuously improve the classification models.

Future work in this area should also include physician engagement - there is an opportunity to assess physician satisfaction with the CDSS alerting pre and post intervention. The real opportunity lies in changing the way these CDSS alerts are delivered based on the physician context, and measuring impact of this change in alert delivery on alert acceptance rates, provider satisfaction, and ultimately, patient outcomes.

APPENDIX A

SimpleKMeans: Available Parameters and their Definitions in Weka

seed -- The random number seed to be used.

displayStdDevs -- Display std deviations of numeric attributes and counts of nominal attributes.

numExecutionSlots -- The number of execution slots (threads) to use. Set equal to the number of available cpu/cores

dontReplaceMissingValues -- Replace missing values globally with mean/mode.

canopyMinimumCanopyDensity -- If using canopy clustering for initialization and/or speedup this is the minimum T2-based density below which a canopy will be pruned during periodic pruning

canopyT2 -- The T2 distance to use when using canopy clustering. Values < 0 indicate that this should be set using a heuristic based on attribute standard deviation

numClusters -- set number of clusters

doNotCheckCapabilities -- If set, clusterer capabilities are not checked before clusterer is built (Use with caution to reduce runtime).

preserveInstancesOrder -- Preserve order of instances.

maxIterations -- set maximum number of iterations

canopyPeriodicPruningRate -- If using canopy clustering for initialization and/or speedup this is how often to prune low density canopies during training

canopyMaxNumCanopiesToHoldInMemory -- If using canopy clustering for initialization and/or speedup this is the maximum number of candidate canopies to retain in main memory during training of the canopy clusterer. T2 distance and data characteristics determine how many candidate canopies are formed before periodic and final pruning are performed. There may not be enough memory available if T2 is set too low.

initializationMethod -- The initialization method to use. Random, k-means++, Canopy or farthest first

distanceFunction -- The distance function to use for instances comparison (default: weka.core.EuclideanDistance).

canopyT1 -- The T1 distance to use when using canopy clustering. Values < 0 are taken as a positive multiplier for the T2 distance

fastDistanceCalc -- Uses cut-off values for speeding up distance calculation, but suppresses also the calculation and output of the within cluster sum of squared errors/sum of distances.

reduceNumberOfDistanceCalcsViaCanopies -- Use canopy clustering to reduce the number of distance calculations performed by k-means

APPENDIX B

B.1: Simulation Script

```
#Simulating Jamey's thesis data
#For now I'll simulate the characteristic variables naively
library(xlsx)
physicianType = LETTERS[1:5]
patientType = c("High",rep("Rising",4),rep("Low",15))
alertTypes = read.xlsx2("Alert Types for Model - 2018-04-29.xlsx",sheetName="Alert
classification detail",startRow=2,stringsAsFactors=FALSE)
names(alertTypes) = c(NA,'cdsGroup','cdsType','severity','alertType',NA,NA,NA,NA,NA)
alertTypes = alertTypes[,which(!is.na(names(alertTypes)))]
alertTypes$cdsType = gsub("\n"," ",alertTypes$cdsType)
#so for a random physician and patient type I need to get ALL the alert types
frame =
expand.grid(physicianType=physicianType,patientType=patientType,1:dim(alertTypes)[1])
frame = cbind(frame[,1:2],alertTypes[frame[,3],])

###two random components:
#1. Volume is a function of patient severity
#2. Volume needs to be split into "accept" and "ignore" based on a bunch of variables
n=100000
k=dim(frame)[1]
set.seed(11)
#getting the samples
dat = frame[sample(1:k,n,replace=TRUE),]
#getting the Volumes
pts = as.numeric(dat$patientType)
dat$Volume = round(runif(n,min=(pts*3)^2,max=(pts*6)^2))
###for the decision making: response should be a function of
# patientType, severity and alertType (leaving the CDS components out)
# each physician should have their own response probabilities

pMatrix = #matrix(ncol=9,nrow=5)
  rbind(#  Int    Low Rising   High     1     2 Suggestion  Alert Reminder
    A=c(0.100, 0.100, 0.400, 0.150, 0.000, 0.000,     0.000, 0.100,  0.200),
    B=c(0.125, 0.000, 0.000, 0.000, 0.000, 0.400,     0.100, 0.350,  0.000),
    C=c(0.000, 0.000, 0.100, 0.200, 0.200, 0.000,     0.100, 0.000,  0.250),
    D=c(0.000, 0.100, 0.200, 0.250, 0.000, 0.000,     0.000, 0.400,  0.350),
    E=c(0.100, 0.000, 0.000, 0.000, 0.100, 0.400,     0.200, 0.100,  0.000)
  )
colnames(pMatrix) = c("Int","Low","Rising","High","1","2","Suggestion","Alert","Reminder")

#adding cdsGroup probabilites
pMatrix2 = rbind(
     #Expert, Dosing, Order, POC Alert
  A = c(0.100,  0.250,  0.100, 0.100),
  B = c(0.000,  0.000,  0.000, 0.000),
  C = c(0.200,  0.300,  0.200, 0.100),
  D = c(0.100,  0.300,  0.100, 0.000),
  E = c(0.000,  0.200,  0.000, 0.100)
)
```

```
colnames(pMatrix2) = names(table(dat$cdsGroup))

p0 = pMatrix[dat$physicianType,1]
p1 = pMatrix[as.matrix(cbind(dat[,c("physicianType","patientType")]))]
p2 = pMatrix[as.matrix(cbind(dat[,c("physicianType","severity")]))]
p3 = pMatrix[as.matrix(cbind(dat[,c("physicianType","alertType")]))]
p4 = pMatrix2[as.matrix(cbind(dat[,c("physicianType","cdsGroup")]))]
p = p0+p1+p2+p3+p4+runif(n,-0.05,0.05)

dat$Respond = rbinom(n,size=dat$Volume,p=p)
dat$Ignore = dat$Volume-dat$Respond

write.csv(dat,file='simDataForJamey.V3.csv',row.names = FALSE)
#testing the results
dat$responseRate = dat$Respond/dat$Volume

library(rpart)
library(rattle)
library(e1071)
library(nnet)
n=dim(dat)[1]
nTrain = round(n*0.7)
train = dat[1:nTrain,]
test = dat[(nTrain+1):n,]
mod =
rpart(responseRate~physicianType+patientType+alertType+severity+cdsGroup,data=train,control
=list(cp=0.0000))
fancyRpartPlot(mod)
pred = predict(mod,newdata=test)



library(klaR)
clust01 = kmodes(dat[,1:6],modes = 5)
clust02 = kmodes(dat[,c(1,2,5,6)],modes=5)
```

B.2: Complete Proof of Concept Data (n=100,000)

simDataV3_quartiles.csv

APPENDIX C

J48 (C4.5) Available Algorithm Parameters in Weka

seed -- The seed used for randomizing the data when reduced-error pruning is used.

unpruned -- Whether pruning is performed.

confidenceFactor -- The confidence factor used for pruning (smaller values incur more pruning).

numFolds -- Determines the amount of data used for reduced-error pruning. One fold is used for pruning, the rest for growing the tree.

numDecimalPlaces -- The number of decimal places to be used for the output of numbers in the model.

batchSize -- The preferred number of instances to process if batch prediction is being performed. More or fewer instances may be provided, but this gives implementations a chance to specify a preferred batch size.

reducedErrorPruning -- Whether reduced-error pruning is used instead of C.4.5 pruning.

useLaplace -- Whether counts at leaves are smoothed based on Laplace.

doNotMakeSplitPointActualValue -- If true, the split point is not relocated to an actual data value. This can yield substantial speed-ups for large datasets with numeric attributes.

debug -- If set to true, classifier may output additional info to the console.

subtreeRaising -- Whether to consider the subtree raising operation when pruning.

saveInstanceData -- Whether to save the training data for visualization.

binarySplits -- Whether to use binary splits on nominal attributes when building the trees.

doNotCheckCapabilities -- If set, classifier capabilities are not checked before classifier is built (Use with caution to reduce runtime).

minNumObj -- The minimum number of instances per leaf.

useMDLcorrection -- Whether MDL correction is used when finding splits on numeric attributes.

collapseTree -- Whether parts are removed that do not reduce training error.

APPENDIX D

J48 Pruned Tree

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     simDataV3_quartiles-weka.filters.unsupervised.attribute.Remove-R7-10-
weka.filters.unsupervised.attribute.NumericToNominal-Rfirst-last-weka.filters.unsupervised.attribute.Remove-R4
Instances:    100000
Attributes:   6
              physicianType
              patientType
              cdsGroup
              severity
              alertType
              Response Rate Category
Test mode:    split 90.0% train, remainder test

=== Classifier model (full training set) ===

J48 pruned tree
------------------

alertType = Suggestion
|   physicianType = A
|   |   patientType = Rising
|   |   |   cdsGroup = Order Facilitator: 3 (768.0/369.0)
|   |   |   cdsGroup = Medication Dosing: 4 (434.0/3.0)
|   |   |   cdsGroup = Point of Care Alert/Reminder: 3 (501.0/245.0)
|   |   |   cdsGroup = Expert Systems: 3 (334.0/156.0)
|   |   patientType = High
|   |   |   cdsGroup = Order Facilitator: 1 (181.0/93.0)
|   |   |   cdsGroup = Medication Dosing: 3 (103.0/49.0)
|   |   |   cdsGroup = Point of Care Alert/Reminder: 1 (112.0/63.0)
|   |   |   cdsGroup = Expert Systems: 2 (70.0/36.0)
|   |   patientType = Low
|   |   |   cdsGroup = Order Facilitator: 1 (2888.0/835.0)
|   |   |   cdsGroup = Medication Dosing: 2 (1728.0/669.0)
|   |   |   cdsGroup = Point of Care Alert/Reminder: 1 (1633.0/471.0)
|   |   |   cdsGroup = Expert Systems: 1 (1137.0/372.0)
|   physicianType = B: 4 (9979.0/3433.0)
|   physicianType = C
|   |   cdsGroup = Order Facilitator
|   |   |   patientType = Rising: 2 (771.0/170.0)
|   |   |   patientType = High: 3 (197.0/115.0)
|   |   |   patientType = Low: 1 (2864.0/811.0)
|   |   cdsGroup = Medication Dosing
|   |   |   patientType = Rising: 3 (491.0/155.0)
|   |   |   patientType = High: 4 (116.0/61.0)
|   |   |   patientType = Low: 2 (1776.0/256.0)
|   |   cdsGroup = Point of Care Alert/Reminder
|   |   |   patientType = Rising: 1 (501.0/159.0)
|   |   |   patientType = High: 2 (112.0/63.0)
|   |   |   patientType = Low: 1 (1733.0/7.0)
|   |   cdsGroup = Expert Systems
|   |   |   patientType = Rising: 2 (324.0/82.0)
|   |   |   patientType = High: 3 (86.0/48.0)
|   |   |   patientType = Low: 1 (1130.0/361.0)
|   physicianType = D
|   |   cdsGroup = Order Facilitator
|   |   |   patientType = Rising: 1 (783.0/253.0)
|   |   |   patientType = High: 2 (208.0/114.0)
|   |   |   patientType = Low: 1 (2935.0/4.0)
|   |   cdsGroup = Medication Dosing
|   |   |   patientType = Rising: 3 (488.0/180.0)
```

```
|   |   |   patientType = High: 3 (137.0/78.0)
|   |   |   patientType = Low: 2 (1750.0/242.0)
|   |   cdsGroup = Point of Care Alert/Reminder: 1 (2314.0/36.0)
|   |   cdsGroup = Expert Systems: 1 (1586.0/163.0)
|   physicianType = E: 4 (9993.0/198.0)
alertType = Alert
|   physicianType = A
|   |   patientType = Rising: 4 (1551.0/74.0)
|   |   patientType = High
|   |   |   cdsGroup = Order Facilitator: 3 (0.0)
|   |   |   cdsGroup = Medication Dosing: 3 (125.0/68.0)
|   |   |   cdsGroup = Point of Care Alert/Reminder: 2 (272.0/155.0)
|   |   |   cdsGroup = Expert Systems: 3 (0.0)
|   |   patientType = Low
|   |   |   cdsGroup = Order Facilitator: 2 (0.0)
|   |   |   cdsGroup = Medication Dosing: 3 (1803.0/269.0)
|   |   |   cdsGroup = Point of Care Alert/Reminder: 2 (4039.0/567.0)
|   |   |   cdsGroup = Expert Systems: 2 (0.0)
|   physicianType = B: 3 (7709.0/3384.0)
|   physicianType = C
|   |   cdsGroup = Order Facilitator: 1 (0.0)
|   |   cdsGroup = Medication Dosing
|   |   |   patientType = Rising: 3 (443.0/219.0)
|   |   |   patientType = High: 4 (130.0/13.0)
|   |   |   patientType = Low: 3 (1712.0/432.0)
|   |   cdsGroup = Point of Care Alert/Reminder
|   |   |   patientType = Rising: 2 (1109.0/286.0)
|   |   |   patientType = High: 3 (250.0/137.0)
|   |   |   patientType = Low: 1 (4090.0/1249.0)
|   |   cdsGroup = Expert Systems: 1 (0.0)
|   physicianType = D
|   |   cdsGroup = Order Facilitator: 3 (0.0)
|   |   cdsGroup = Medication Dosing: 4 (2253.0)
|   |   cdsGroup = Point of Care Alert/Reminder
|   |   |   patientType = Rising: 3 (1076.0/538.0)
|   |   |   patientType = High: 4 (251.0/93.0)
|   |   |   patientType = Low: 3 (3951.0/946.0)
|   |   cdsGroup = Expert Systems: 3 (0.0)
|   physicianType = E
|   |   cdsGroup = Order Facilitator: 2 (0.0)
|   |   cdsGroup = Medication Dosing: 3 (2252.0/654.0)
|   |   cdsGroup = Point of Care Alert/Reminder: 2 (5324.0/943.0)
|   |   cdsGroup = Expert Systems: 2 (0.0)
alertType = Reminder
|   physicianType = A
|   |   patientType = Rising: 4 (417.0)
|   |   patientType = High: 3 (114.0/63.0)
|   |   patientType = Low: 3 (1767.0/434.0)
|   physicianType = B: 1 (2299.0/2.0)
|   physicianType = C
|   |   patientType = Rising: 4 (498.0/112.0)
|   |   patientType = High: 4 (119.0/14.0)
|   |   patientType = Low: 3 (1729.0/284.0)
|   physicianType = D
|   |   patientType = Rising: 3 (490.0/128.0)
|   |   patientType = High: 3 (126.0/70.0)
|   |   patientType = Low: 2 (1673.0/660.0)
|   physicianType = E: 1 (2265.0/705.0)

Number of Leaves  :     73

Size of the tree :      101
```

APPENDIX E

Weka MLP Parameters

seed – Seed used to initialise the random number generator. Random numbers are used for setting the initial weights of the connections between nodes, and also for shuffling the training data.

momentum -- Momentum applied to the weights during updating.

nominalToBinaryFilter – This will preprocess the instances with the filter. This could help improve performance if there are nominal attributes in the data.

hiddenLayers – This defines the hidden layers of the neural network. This is a list of positive whole numbers. 1 for each hidden layer. Comma seperated. To have no hidden layers put a single 0 here. This will only be used if autobuild is set. There are also wildcard values 'a' = (attribs + classes) / 2, 'i' = attribs, 'o' = classes , 't' = attribs + classes.

validationThreshold -- Used to terminate validation testing.The value here dictates how many times in a row the validation set error can get worse before training is terminated.

GUI – Brings up a gui interface. This will allow the pausing and altering of the nueral network during training.

normalizeAttributes – This will normalize the attributes. This could help improve performance of the network. This is not reliant on the class being numeric. This will also normalize nominal attributes as well (after they have been run through the nominal to binary filter if that is in use) so that the nominal values are between -1 and 1

numDecimalPlaces – The number of decimal places to be used for the output of numbers in the model.

batchSize – The preferred number of instances to process if batch prediction is being performed. More or fewer instances may be provided, but this gives implementations a chance to specify a preferred batch size.

decay -- This will cause the learning rate to decrease. This will divide the starting learning rate by the epoch number, to determine what the current learning rate should be. This may help to stop the network from diverging from the target output, as well as improve general performance. Note that the decaying learning rate will not be shown in the gui, only the original learning rate. If the learning rate is changed in the gui, this is treated as the starting learning rate.

validationSetSize – The percentage size of the validation set.(The training will continue until it is observed that the error on the validation set has been consistently getting worse, or if the training time is reached).

If This is set to zero no validation set will be used and instead the network will train for the specified number of epochs.

trainingTime -- The number of epochs to train through. If the validation set is non-zero then it can terminate the network early

debug – If set to true, classifier may output additional info to the console.

autoBuild -- Adds and connects up hidden layers in the network.

normalizeNumericClass -- This will normalize the class if it's numeric. This could help improve performance of the network, It normalizes the class to be between -1 and 1. Note that this is only internally, the output will be scaled back to the original range.

learningRate – The amount the weights are updated.

doNotCheckCapabilities – If set, classifier capabilities are not checked before classifier is built (Use with caution to reduce runtime).

reset -- This will allow the network to reset with a lower learning rate. If the network diverges from the answer this will automatically reset the network with a lower learning rate and begin training again. This option is only available if the gui is not set. Note that if the network diverges but isn't allowed to reset it will fail the training process and return an error message.

REFERENCES

[1] The World Bank Group. "Life expectancy at birth, total (years) - Canada." *The World Bank*, World Bank Group, data.worldbank.org/indicator/SP.DYN.LE00.IN?locations=CA. Accessed 6 Mar. 2019.

[2] Schaink, A.K., et al. "A scoping review and thematic classification of patient complexity: offering a unifying framework." *J Comorb,* vol. 2, 10 Oct. 2012, pp. 1-9.

[3] Porter, M. "A Strategy for Healthcare Reform – Toward a Value-Based System." *New England Journal of Medicine*, vol. 361, no. 2, July 2009, pp. 109-12.

[4] Munnich, Elizabeth L., and Stephen T. Parente. "Procedures Take Less Time At Ambulatory Surgery Centers, Keeping Costs Down And Ability To Meet Demand Up." *Health Affairs*, vol. 33, no. 5, May 2014, pp. 764-69.

[5] Khangura, Sara, et al. *Evidence Review Summary: Health Services for the 21st Century*. The Ottawa Hospital Research Institute.

[6] The Office of the National Coordinator for Health Information Technology (ONC). "Clinical Decision Support." *HealthIT.gov*, Office of the National Coordinator for Health Information Technology (ONC), 10 Apr. 2018, www.healthit.gov/topic/safety/clinical-decision-support. Accessed 7 Mar. 2019.

[7] Wright, A., et al. "Clinical Decision Support Capabilities of Commercially-available Clinical Information Systems." *J Am Med Inform Assoc*, vol. 16, no. 5, Sept.-Oct. 2009, pp. 637-44.

[8] Tcheng, J. E., S. Bakken, D. W. Bates, H. Bonner III, T. K. Gandhi, M., Josephs, K. Kawamoto, E. A. Lomotan, E. Mackay, B. Middleton, J. M. Teich, S. Weingarten, and M. Hamilton Lopez, editors. 2017. *Optimizing Strategies for Clinical Decision Support: Summary of a Meeting Series.* Washington, DC: National Academy of Medicine.

[9] Lobach, D., G. Sanders, T. Bright, et al. *Enabling Health Care Decision making Through Clinical Decision Support and Knowledge Management*. Evidence Report No. 203. (Prepared by the Duke Evidence-based Practice Center under Contract No. 290-2007-10066-I.) AHRQ Publication No. 12-E001-EF. Rockville, MD: Agency for Healthcare Research and Quality. April 2012.

[10] Hannan, Terry J. ‘Electronic medical records’. *Health Informatics: An Overview*. 1996. Edited by Michael Kidd and E. Hovenga, Melbourne, Churchill Livingston, 2010: pp 133-148.

[11] Chang, Feng, and Nishi Gupta. "Progress in electronic medical record adoption in Canada." *Canadian Family Physician*, vol. 61, Dec. 2015, pp. 1076-84.

[12] Jamoom E., N. Yang. “Table of Electronic Health Record Adoption and Use among Office-based Physicians in the U.S., by State: 2015 National Electronic Health Records Survey”. 2016. Available at: https://www.cdc.gov/nchs/data/ahcd/nehrs/2015_nehrs_web_table.pdf

[13] Mylopoulos, J., et al. "Building knowledge base management systems." *The VLDB Journal*, vol. 5, no. 4, Dec. 1996, pp. 238-63.

[14] The Office of the National Coordinator for Health Information Technology (ONC). " Clinical Decision Support: More than Just 'Alerts' Tipsheet." *HealthIT.gov*, Office of the National Coordinator for Health Information Technology (ONC), Jul. 2014, https://www.healthit.gov/sites/default/files/clinicaldecisionsupport_tipsheet.pdf Accessed 7, Mar. 2019.

[15] Centers for Disease Control and Prevention. "Meaningful Use is about how technology helps humanity." *CDC: Centers for Disease Control and Prevention*, U.S. Department of Health & Human Services, 24 Sept. 2018, www.cdc.gov/ehrmeaningfuluse/index.html. Accessed 8 Mar. 2019.

[16] Patient Safety Network, and Agency for Patient Safety and Quality. "Alert Fatigue." Patient Safety Net, Agency for Healthcare Research and Quality, Jan. 2019, psnet.ahrq.gov/primers/primer/ 28/alert-fatigue. Accessed 8 Mar. 2019.

[17] Berner ES. Clinical decision support systems: State of the Art. AHRQ Publication No. 09-0069-EF. Rockville, Maryland: Agency for Healthcare Research and Quality. June 2009.

[18] Bryan, Cathy, and Suzanne Boren. "The use and effectiveness of electronic clinical decision support tools in the ambulatory/primary care setting: a systematic review of the literature." *Inform Prim Care*, vol. 16, no. 2, 2008, pp. 79-91.

[19] Shah T, Patel-Teague S, Kroupa L, et al. “Impact of a national QI programme on reducing electronic health record notifications to clinicians”. *BMJ Qual Saf,* vol.28, no. 1, 5 Feb. 2018, pp. 10-14.

[20] Coleman, Jamie J., et al. "On the alert: future priorities for alerts in clinical decision support for computerized physician order entry identified from a European workshop." BMC Medical Informatics and Decision Making, vol. 13, 1 Oct. 2013, pp. 111-19.

[21] Wright, A., et al. "Development and evaluation of a comprehensive clinical decision support taxonomy: comparison of front-end tools in commercial and internally developed electronic health record systems." *J Am Med Inform Assoc*, vol. 18, no. 3, 1 May 2011, pp. 232-42.

[22] Smithburger, P. L., et al. "A critical evaluation of clinical decision support for the detection of drug-drug interactions." Expert Opin Drug Saf, vol. 10, no. 6, Nov. 2011, pp. 871-72.

[23] Guiriguet, C., et al. "Alerts in electronic medical records to promote a colorectal cancer screening programme: a cluster randomised controlled trial in primary care." Br J Gen Pract, vol. 66, no. 648, July 2016, pp. e483-90.

[24] Glassman, P. A., et al. "Improving recognition of drug interactions: benefits and barriers to using automated drug alerts." *Med Care*, vol. 40, no. 12, Dec. 2002, pp. 1161-71.

[25] Beeler, P. E., et al. "Clinical decision support systems." *Swiss Medical Weekly*, vol. 144, no. w14073, 23 Dec. 2014.

[26] Paterno, M. D., et al. "Tiering drug–drug interaction alerts by severity increases compliance rates." *Journal of the American Medical Informatics Association*, vol. 16, no. 1, Jan.-Feb. 2009, pp. 40-46.

[27] Weingart, S. N., et al. ": Clinicians' assessments of electronic medication safety alerts in ambulatory care." *Arch Intern Med*, vol. 169, no. 17, 28 Sept. 2009, pp. 1627-32.

[28] Bright, T. J., et al. "Effect of clinical decision-support systems: a systematic review." *Ann Intern Med*., vol. 157, no. 1, 3 July 2012, pp. 29-43.

[29] "Data Mining vs. Machine Learning: What's The Difference?" *import.io*, Import.io, 21 Oct. 2017, www.import.io/post/data-mining-machine-learning-difference/. Accessed 10 Mar. 2019.

[30] Hand, D., et al. *Principles of data mining.* Cambridge, The MIT Press, 2001. Adaptive Computation and Machine Learning.

[31] Snow, Jackie. "10 Breakthrough Technologies 2018." *MIT Technology Review*, The MIT Group, www.technologyreview.com/lists/technologies/2018/. Accessed 10 Mar. 2019.

[32] Yoo, I., et al. "Data mining in healthcare and biomedicine: a survey of the literature." *J Med Syst*, vol. 36, no. 4, Aug. 2012, pp. 2431-48.

[33] Koh, H. C., and G. Tan. "Data Mining Applications in Healthcare." *J Healthc Inf Manag*, vol. 19, no. 2, Spring 2005, pp. 64-72.

[34] Ng, K., et al. "Personalized Predictive Modeling and Risk Factor Identification using Patient Similarity." *AMIA Jt Summits Transl Sci Proc*, vol. 2015, 25 Mar. 2015, pp. 132-36.

[35] Escobar, G. J., et al. "Stratification of risk of early-onset sepsis in newborns ≥34 weeks' gestation." *Pediatrics*, vol. 133, no. 1, Jan. 2014, pp. 30-36.

[36] Shameer, K., et al. "Pharmacological risk factors associated with hospital readmission rates in a psychiatric cohort identified using prescriptome data mining." *BMC Medical Informatics and Decision Making*, vol. 18, nos. Suppl 3, 14 Sept. 2018, pp. 79-90.

[37] He, D., and S. C. Matthews. "Mining high-dimensional administrative claims data to predict early hospital readmissions." *J Am Med Inform Assoc*., vol. 21, no. 2, Mar.-Apr. 2014, pp. 272-9.

[38] Vijayakrishnan, R., et al. "Prevalence of Heart Failure Signs and Symptoms in a Large Primary Care Population Identified Through the Use of Text and Data Mining of the Electronic Health Record." *J Card Fail.*, vol. 20, no. 7, July 2014, pp. 459-64.

[39] *College of Physicians and Surgeons of Nova Scotia*. cpsns.ns.ca/. Accessed 10 Mar. 2019.

[40] Berkhin P. (2006) *A Survey of Clustering Data Mining Techniques.* In: Kogan J., Nicholas C., Teboulle M. (eds) Grouping Multidimensional Data. Springer, Berlin, Heidelberg (2006).

[41] Witten, I. H., Frank, Eibe, and Hall, Mark A. *Data Mining Practical Machine Learning Tools and Techniques.* 3rd Ed. / Ian H. Witten, Eibe Frank, Mark A. Hall.. ed. Burlington, MA: Morgan Kaufmann, 2011. Print. Morgan Kaufmann Ser. in Data Management Systems.

[42] Bock, Hans-Hermann. "Origins and extensions of the k-means algorithm in cluster analysis." *Electronic Journ@l for History of Probability and Statistics*, vol. 4, no. 2, Dec. 2008, p.1-18.

[43] "Weka 3: Data Mining Software in Java." *Weka*, www.cs.waikato.ac.nz/ml/weka/. Accessed 11 Mar. 2019.

[44] Gera, Mansi, and Shivani Goel. "Data Mining - Techniques, Methods and Algorithms: A Review on Tools and their Validity." *International Journal of Computer Applications*, vol. 113, no. 18, Mar. 2015, pp. 22-29.

[45] Wu, Junjie, et al. "Adapting the Right Measures for K-means Clustering." *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, July 2009, pp. 877-85.

[46] Canese, Kathi, and Sarah Weis. *The NCBI Handbook*. 2nd ed., Internet, National Center for Biotechnology Information (US), 2013.

[47] Hasan, Hamza. "Population health managers, meet the three patient types central to your success." *Advisory Board Care Transformation Center Blog*, Advisory Board, 23 Oct. 2013, www.advisory.com/research/care-transformation-center/care-transformation-center-blog/2013/06/ three-care-delivery-models. Accessed 11 Mar. 2019.

[48] Kotsiantis, S. B. "Supervised Machine Learning: A Review of Classification Techniques." *Informatica*, 16 July 2007, pp. 249-68.

[49] Quinlan, J. Ross. *C4.5: Programs for Machine Learning.* San Mateo, California, Morgan Kaufman, 1993.

[50] Alotaibi, Yasser K., and Frank Federico. "The impact of health information technology on patient safety." *Saudi Med J*, vol. 38, no. 12, Dec. 2017, pp. 1173-80.

[51] "HITECH Act Enforcement Interim Final Rule." *HHS.gov*, U.S. Department of Health & Human Services, www.hhs.gov/hipaa/for-professionals/special-topics/hitech-act-enforcement-interim-final-rule/index.html. Accessed 11 Mar. 2019.

[52] "Meaningful Use." *Centers for Disease Control and Prevention*, 18 Jan. 2017, www.cdc.gov/ehrmeaningfuluse/introduction.html. Accessed 11 Mar. 2019.

[53] National Health Service. *NEXT STEPS ON THE NHS FIVE YEAR FORWARD VIEW*. NHS, Mar. 2017.

[54] Moja, Lorenzo, et al. "Implementing an evidence-based computerized decision support system to improve patient care in a general hospital: the CODES study protocol for a randomized controlled tria." *Implementation Science*, vol. 11, 7 July 2016, pp. 89-99.

[55] Nuckols, T. L., et al. "The effectiveness of computerized order entry at reducing preventable adverse drug events and medication errors in hospital settings: a systematic review and meta-analysis." *Systematic Reviews*, vol. 3, 4 June 2014, pp. 56-68.

[56] Kawamoto, K., et al. "Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success." *British Medical Journal*, vol. 330, 14 Mar. 2005, pp. 765-73.

[57] Varghese, J., et al. "Effects of computerized decision support system implementations on patient outcomes in inpatient care: a systematic review." *J Am Med Inform Assoc*, vol. 25, no. 5, 1 May 2018, pp. 593-602.

[58] Moja, L., et al. "Effectiveness of computerized decision support systems linked to electronic health records: a systematic review and meta-analysis." *Am J Public Health*, vol. 104, no. 12, Dec. 2014, pp. e12- 22.

[59] Stone, E. G. "Unintended adverse consequences of a clinical decision support system: two cases." *J Am Med Inform Assoc*, vol. 25, no. 5, 1 May 2018, pp. 564-67.

[60] Lugtenberg, M., et al. "Implementation of multiple-domain covering computerized decision support systems in primary care: a focus group study on perceived barriers." *BMC Medical Informatics and Decision Making*, vol. 15, no. 82, 12 Oct. 2015.

[61] Kilsdonk, E., et al. "Factors influencing implementation success of guideline-based clinical decision support systems: A systematic review and gaps analysis." *International Journal of Medical Informatics*, vol. 98, Feb. 2017, pp. 56-64.

[62] Alotaibi, Y., and F. Federico. "The impact of health information technology on patient safety." *Saudi Med*, vol. 38, no. 12, Dec. 2017, pp. 1173-80.

[63] Vernooij, R. W., et al. "Guidance for updating clinical practice guidelines: a systematic review of methodological handbooks." *Implement Sci*, vol. 9, no. 3, 2 Jan. 2014.

[64] Saxena, K., et al. "Improving Patient Safety by Modifying Provider Ordering Behavior Using Alerts (CDSS) in CPOE system." *AMIA Annu Symp Proc*, vol. 2011, 22 Oct. 2011, pp. 1207-16.

[65] Landhuis, Esther. "Scientific literature: Information overload." *Nature*, vol. 535, 21 July 2013, pp. 457-58.

[66] Backman, R., et al. "Clinical reminder alert fatigue in healthcare: a systematic literature review protocol using qualitative evidence." *Systematic Reviews*, vol. 6, 2017, p. 255.

[67] Liberati, E. G., et al. "What hinders the uptake of computerized decision support systems in hospitals? A qualitative study and framework for implementation." *Implementation Science*, vol. 12, no. 1, 15 Sept. 2017, p. 113.

[68] Ancker, J. S., et al. "Effects of workload, work complexity, and repeated alerts on alert fatigue in a clinical decision support system." *BMC Medical Informatics and Decision Making*, vol. 17, 10 Apr. 2017, p. 36-45.

[69] Press, A., et al. "Avoiding alert fatigue in pulmonary embolism decision support: a new method to examine 'trigger rates.'" *Evid Based Med*, vol. 21, no. 6, Dec. 2016, pp. 203-07.

[70] Footracer, K. G. "Alert fatigue in electronic health records." *Journal of the American Academy of PAs*, vol. 28, no. 7, July 2015, pp. 41-42.

[71] McCoy, A. B., et al. "Clinical Decision Support Alert Appropriateness: A Review and Proposal for Improvement." *Ochsner Journal*, vol. 14, no. 2, Summer 2014, pp. 195-202.

[72] Van der Sijs, H., et al. "Overriding of drug safety alerts in computerized physician order entry." *J Am Med Inform Assoc*, vol. 13, no. 2, Mar.-Apr. 2006, pp. 138-47.

[73] Bryant, A. D., et al. "Drug interaction alert override rates in the meaningful use era. No evidence of progress." *Appl Clin Inform*, vol. 5, no. 3, 3 Sept. 2014, pp. 802-13.

[74] Miller, K., et al. "Interface, information, interaction: a narrative review of design and functional requirements for clinical decision support." *J Am Med Inform Assoc*, vol. 25, no. 5, 1 May 2018, pp. 585-92.

[75] Seilding, H. M., et al. "What, if all alerts were specific - estimating the potential impact on drug interaction alert burden." *Int J Med Inform*, vol. 83, no. 4, Apr. 2014, pp. 285-91.

[76] Heringa, M., et al. "Lower alert rates by clustering of related drug interaction alerts." *J Am Med Inform Assoc*, vol. 24, no. 1, Jan. 2017, pp. 54-59.

[77] Koskela, T., et al. "User perspectives on an electronic decision-support tool performing comprehensive medication reviews – a focus group study with physicians and nurses." *BMC Medical Informatics and Decision Making*, vol. 16, 22 Jan. 2016, pp. 6-15.

[78] Cornu, P., et al. "High-priority and low-priority drug–drug interactions in different international electronic health record systems: A comparative study." *Int J Med Inform*, vol. 111, Mar. 2018, pp. 165-71.

[79] Sadler, Dr Laura. Interview. 23 Feb. 2019.

[80] Bradley, A. "The use of the area under the ROC curve in the evaluation of machine learning algorithms." *Pattern Recognition*, vol. 30, no. 7, July 1997, pp. 1145-59.

[81] Neelamegam, S., and E. Ramaraj. "Classification algorithm in Data mining: An Overview." *International Journal of P2P Network Trends and Technology (IJPTT)*, vol. 3, no. 5, Sept.-Oct. 2013, pp. 1-5.

[82] Decker, S., et al. "Physicians In Nonprimary Care And Small Practices And Those Age 55 And Older Lag In Adopting Electronic Health Record Systems." *Health Affairs*, vol. 31, no. 5, May 2012, pp. 1108-14.

[83] Li, J., et al. "Health Care Provider Adoption of eHealth: Systematic Literature Review." *Interact J Med Res*, vol. 2, no. 1, 16 Apr. 2013, p. e7.

[84] Fawcett, T. An introduction to ROC analysis. *Pattern Recognition Letters*, vol. 27, no. 8, Jun 2006, pp. 861-874.