DEVELOPING CLINICAL DECISION SUPPORT SYSTEMS FOR SEPSIS PREDICTION USING TEMPORAL AND NON-TEMPORAL MACHINE LEARNING METHODS

By

Akash Gupta

Bachelor of Technology in Computer Engineering
Uttar Pradesh Technical University
Lucknow, India
2009

Master of Technology in Control Systems
Indian Institute of Technology (BHU)
Varanasi, India
2012

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
Doctor of Philosophy
July 2019

DEVELOPING CLINICAL DECISION SUPPORT SYSTEMS FOR SEPSIS PREDICTION USING TEMPORAL AND NON-TEMPORAL MACHINE LEARNING METHODS

Dissertation Approved:

Dr. Tieming Liu

Dissertation Adviser

Dr. Christopher Crick

Dr. Dursun Delen

Dr. Sunderesh Heragu

Dr. Farzad Yousefian

ACKNOWLEDGEMENTS

I am in gratitude to all the people who inspired and supported me during the process to pursue Ph.D.. First and foremost, I thank my parents for their blessings and shaping me to fight against the odds. I thank my siblings who do not spare a single opportunity to brag about me.

I thank my dissertation advisor, Dr. Tieming Liu, for introducing me to the field of medical informatics. He guided me to formulate and redefine the research questions. This dissertation is a product of his continued support and mentorship. My committee members, Drs. Christopher Crick, Dursun Delen, Sunderesh Heragu, and Farzad Yousefian provided me insights on a variety of subjects including data mining and machine learning. I appreciate their time and efforts. I especially like to thank Dr. Crick for amplifying my curiosity in the area of probabilistic graphical modeling. Dr. Debjit Roy planted and nourished the seed of research to flourish my academic success. During my formative years in the realm of research, he encouraged me to test my limits and was patient with my mistakes.

The School of Industrial Engineering & Management and the Center for Health Systems Innovation (CHSI) supported this research financially during different phases of my Ph.D. program. Dr. William Paiva fueled me with a passion for helping rural communities. I thank him and his generous staff at CHSI. In my efforts to bridge the gap between medicine and engineering, Dr. Scott Shepherd advised me on the clinical aspect of the problem and directed me to find the relevant problem statement. I thank him for setting the foundation of the work.

My friends, Ana, Ashwin, Ashwin Raman, Babak, Harry, Ishahaq, Kelly, Ruby, Saeed, Sagnik and Shelly helped me to sail through many storms throughout my graduate program. Thanks to all the fun and beers!

Finally, I thank everyone who directly or indirectly supported me to culminate this research.

GO POKES !!!

Acknowledgements reflect the views of the author and are not endorsed by committee members or Oklahoma State University.

Name: Akash Gupta

Date of Degree: July 2019

Title of Study: DEVELOPING CLINICAL DECISION SUPPORT SYSTEMS FOR SEPSIS PREDICTION USING TEMPORAL AND NON-TEMPORAL MACHINE LEARNING METHODS

Major Field: Industrial Engineering & Management

Abstract:

In healthcare, diagnostic errors represent the biggest challenge to synthesize accurate treatments. In the United States, patient deaths due to misdiagnoses are estimated at 40,000 to 80,000 per year. It was also found that 30% of the annual healthcare spending was consumed on unnecessary services and other inefficiencies. The diagnostic errors could be reduced, and public health can be improved by applying machine learning and artificial intelligence in healthcare problems. This dissertation is an attempt to formulate clinical decision support systems and to develop new algorithms to reduce clinical errors.

This dissertation aims at developing clinical decision support systems to diagnose sepsis in the early stages. The key feature of our work is that we captured the dynamics among body organs using Bayesian networks. The richness of the proposed model is measured not only by achieving high accuracy but also by utilizing fewer lab results.

To further improve the accuracy of the clinical decision support system, we utilize longitudinal data to develop a mortality progression model. This part of the dissertation proposes a hidden Markov model (HMM) framework to model the mortality progression. In comparison to existing approaches, the proposed framework leverages the longitudinal data available in the electronic health records (EHR).

In addition, this dissertation proposes an initialization procedure to train the parameters of HMM efficiently. The current HMM learning algorithms are sensitive to initialization. The proposed method computes an initial set of parameters by relaxing the time dependency in sequential time series data and incorporating the multinomial logistic regression.

Finally, this dissertation compares the prognostic accuracy of two popularly used early sepsis diagnostic criteria: Systemic Inflammatory Response Syndrome (SIRS) and quick Sepsis-related Organ Failure Assessment (qSOFA). Using statistical and machine learning methods, we found that qSOFA is a better diagnostic criteria than SIRS. These findings will guide healthcare providers in selecting the best bedside diagnostic criteria.

Dedicated to my

beloved parents, Shri Awdhesh Kumar Gupta and Smt. Kanti Devi Gupta,

beloved sisters, Beena, Abha, Rajni, Alka and Manjula

beloved brother, Vikas alias Baba,

and beloved friend, Shelly

Dedication reflect the views of the author and are not endorsed by committee members or Oklahoma State University.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter I

INTRODUCTION

An ounce of prevention is better than a pound of cure

-Benjamin Franklin

This phrase sums the thrust of this dissertation: early identification of the onset of disease and timely treatment is the best modus operandi to improve public health. This dissertation is an attempt to help both doctors and patients by integrating engineering and medicine.

1 Healthcare in the United States

The United States spent about 3.5 trillion dollars (or $10,000 per person) in 2017 on healthcare, which accounted for 18% of Gross Domestic Product (GDP) [1]. Figure 1.1 illustrates per capita healthcare expenditures across all the states. The states with light blue color have low per capita healthcare expenditures, whereas states with dark blue color have high per capita healthcare expenditures. With per capita healthcare expenditures of $7,627, the total healthcare budget of Oklahoma is about $24 billion dollars [2].

Of the total healthcare expenditures in the United States, 30% is consumed due to inefficiencies caused by an error in diagnosis and delay in treatment. These inefficiencies could

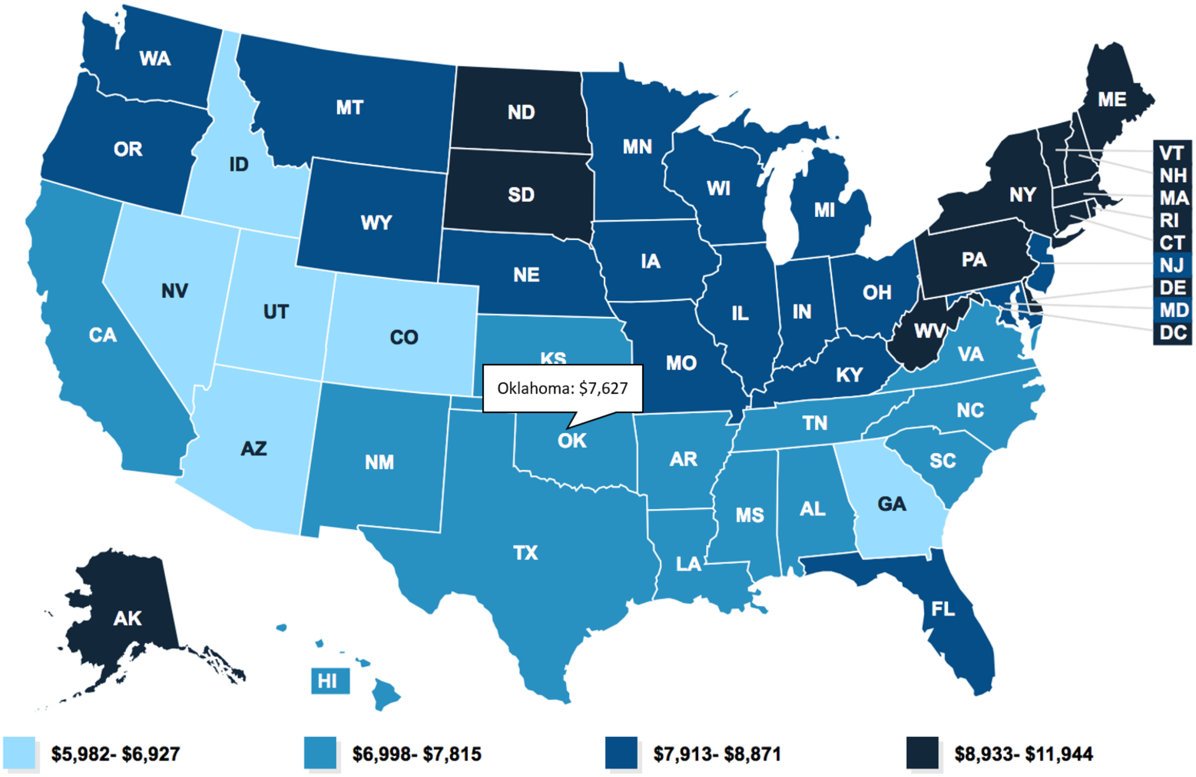


Figure 1.1: Healthcare expenditures per capita across states in the United States (data source: Centers for Medicare and Medicaid Services)

be minimized by using a Clinical Decision Support System (CDSS). A CDSS is a software program that evaluates multiple clinical signs, translating this information into meaningful insights that can be utilized by the physicians to make informative decisions.

The benefits of CDSSs to rural areas are enormous. In the United States, 20% of the population is living in rural areas [3]. According to the Oklahoma Department of Commerce, 67 of all 77 (or 87%) Oklahoma counties are rural. The concern to sustain the health and economy of these counties is immense. The population living in these counties is aging, lacks adequate transportation, and has limited financial resources. These factors alone demand technological advancements in healthcare. To make the problem more severe, these rural counties are struggling to attract young physicians due to lack of facilities and amenities.

Hence, the shortage of doctors in rural counties is imminent. For example, in Alabama, seven rural hospitals closed their operations in the last eight years [4] resulting to 48 hospitals [5]. To manage the shortage of physicians, Blue Cross Blue Shield in collaboration with the medical college of the University of Alabama announced a $3.6 million scholarship program for medical students to promote medical practices in rural areas [6].

Financial incentives are one of the possible ways to deal with the impending scarcity of doctors, but the multitude of doctors cannot be trained promptly. Therefore, the challenge of lack of doctors necessitates an alternative solution. The utility of CDSS provides an alternate solution to manage the lack of properly-trained physicians. These tools could be used by primary care providers or nurses, in the absence of experts, to assess the risk of disease and to deploy limited hospital resources efficiently.

CDSSs facilitate early diagnosis for some diseases and this early identification may be the difference between life and death. For example, delaying sepsis treatment each hour results increase in mortality by 7.6% [7]. Early diagnosis not only prevents untimely deaths but also saves trillions of dollars for healthcare providers. For example, in 2018, the Alzheimer's Association estimated a $7 trillion savings from early Alzheimer detection [8]. In addition, the early detection can be recognized with understanding the dynamics of physiological variables, thus the development of effective CDSSs stems from quality of medical data.

The advent of technology has led to a robust expansion of medical data. Hospitals store both structured (numeric) and unstructured (images) data. Electronic Health Record (EHR) includes information about vital signs, lab results, medications, procedures, demographics and so on. The generated health data from different sources embed this wealth of information. However, the hidden patterns are cryptic and more complex than can be deciphered by a human brain. Therefore, the use of statistical and machine learning methods has been favored to transform health data into knowledge.

Machine learning solves problems like humans, by learning patterns from historical data (or past experiences). It selects a combination of variables that reliably predicts outcomes. Specifically, in medicine, machine learning is helpful in many clinical aspects: *prognosis*, *diagnosis*, *prescription* and *radiology.* The terms *prognosis* and *diagnosis* are often used interchangeably. However, there exists a slight difference – *prognosis* refers to *"future prediction"* while *diagnosis* refers to *"identification of a condition"*.

The traditional prognostic models such as Sepsis-related Organ Failure Assessment (SOFA) [9] rely on manual calculations, while the machine learning based models draw variables directly from EHR and assess the risk with more granularity and accuracy. One of the critical areas where machine learning can contribute significantly is curtailing errors in prognosis. By continuously monitoring physiological data from EHR and triggering a timely alarm to alert physicians for close inspections, the applications of machine learning approaches in healthcare hold great promises. In this dissertation, we explore both temporal (HMM, [10], Dynamic Bayesian Networks (DBN) [11], etc.) and non-temporal (support vector machine, decision tree and so on) statistical and machine learning methods to develop clinical prognostic (or predictive) models.

2 Sepsis at a Glance

Sepsis, traditionally known as the systemic response to infection, is *a life-threatening organ dysfunction caused by a dysregulated host response to an infection* [12]. The incidences of sepsis are growing rapidly in the United States. According to the Centers for Disease Control and Prevention, the number of admissions to Intesive Care Units (ICU) due to sepsis increased about 84% from 621,000 in 2000 to 1,141,000 in 2008 [13]. A worldwide study showed that the incidence of sepsis and severe sepsis were 437 and 270, respectively, per 100,000 people per year for 2003-2015 in high-income countries [14]. Nearly two-thirds of septic patients enter in hospitals through Emergency Departments (ED). The high influx of septic patients makes it the most expensive disease to treat with an annual cost of $24

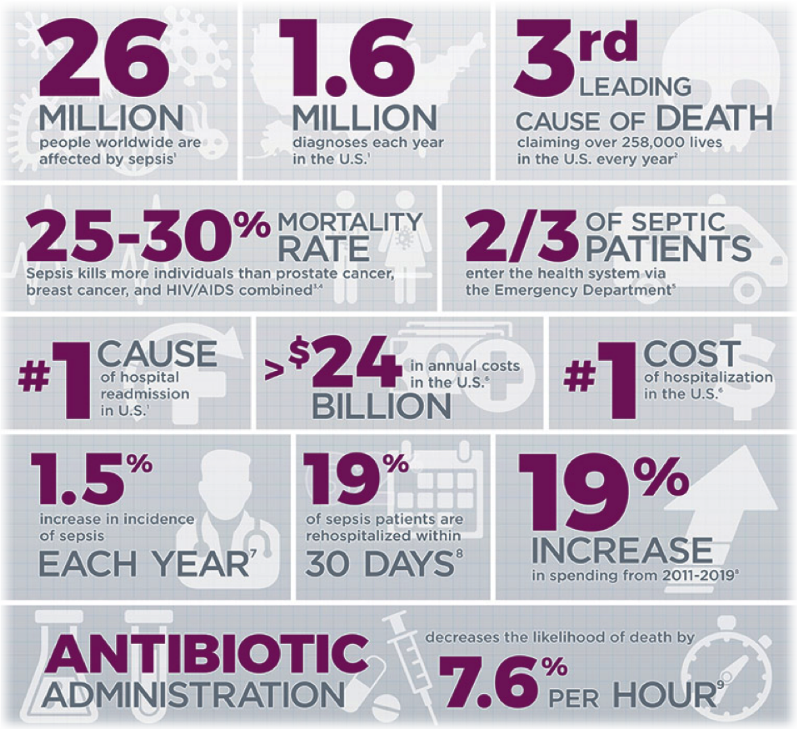

26 MILLION
people worldwide are affected by sepsis[1]
1.6 MILLION
diagnoses each year in the U.S.[1]
3rd LEADING CAUSE OF DEATH
claiming over 258,000 lives in the U.S. every year[2]
25-30% MORTALITY RATE
Sepsis kills more individuals than prostate cancer, breast cancer, and HIV/AIDS combined[3,4]
2/3 OF SEPTIC PATIENTS
enter the health system via the Emergency Department[5]
#1 CAUSE
of hospital readmission in U.S.[1]
>$24 BILLION
in annual costs in the U.S.[6]
#1 COST
of hospitalization in the U.S.[6]
1.5%
increase in incidence of sepsis
EACH YEAR[7]
19%
of sepsis patients are rehospitalized within
30 DAYS[8]
19% INCREASE
in spending from 2011-2019[8]
ANTIBIOTIC ADMINISTRATION
decreases the likelihood of death by
7.6% PER HOUR[9]

Figure 1.2: Sepsis statistics (adapted from `https://www.beckmancoulter.com`)

billion [15]. Patients with sepsis are at considerable risk for severe complications and death. It is the third leading cause of death, after heart disease and cancer, with mortality rates ranging from about 25-30% depending on sepsis severity [16]. Liu et al. (2014) found that the admission to hospitals due to sepsis consisted only 11% of total admissions to hospitals, but the in-hospital mortality among the sepsis patients was considerable, ranging from 37% to 55% [17]. Figure 1.2 summarizes the statistics of sepsis.

Understanding the needs of rural communities, this dissertation is a progression against the waste of medical resources (time and money due to diagnostic errors) and an impending shortage of doctors and attempts to achieve the following objectives:

1. Developing a CDSS using Bayesian networks to effectively diagnose sepsis

2. Engineering a framework to develop mortality monitoring systems for early diagnosis

3. Proposing an initialization method for the learning algorithm of HMM parameters

4. Comparing prognostic accuracy of two popular sepsis diagnosis criteria

3 Brief Overview of Analytics

The analytics is divided into three dimensions: descriptive, predictive, and prescriptive. Usually, predictive analytics follows descriptive, and prescriptive analytics follows predictive. Descriptive analytics focuses on preparing summary tables, performs hypothesis testing (chi-square test statistics, etc.) to understand the structure and relations among variables. The purpose of descriptive analytics is to present the data graphically and to devise possible hypotheses. The predictive analytics is the next level of analytics, where we use statistical or machine learning techniques to learn inherited patterns in data. The learning could be two types: supervised learning and unsupervised learning. In supervised learning, the outcome is known, and we maximize/minimize the prediction accuracy/error. However, in unsupervised learning, the outcome variable is not accessible. The data points are clustered based on the similarity. The predictive analytics aims to determine the possible future outcome. Once we know the probable future outcome, prescriptive analytics helps to derive the control actions to avoid the adverse event.

This dissertation primarily focuses on descriptive and predictive analytics. In terms of descriptive analysis, we initially visualized the data to investigate the trends among variables. For example, using bar chart we noticed that patients who did not survive during the hospital stays showed the low value of the Glasgow Coma Scale compared to patients who survived (Figure 3.9). We also used table to explore the difference in discharge type based on gender, census region etc. (Table 6.2). However, results based on visualization could lead to the wrong conclusion. Therefore, we required to perform statistical tests (chi-square, etc.) to prove the statistically significant difference. In the predictive side, we used logistic regression, decision tree, Bayesian networks and support vector machine to predict the future outcome. Specifically, we used temporal and non-temporal data to predict (or prognosis) sepsis (Chapters IV, V and VI).

4 Problem Statements

4.1 Developing a CDSS for Sepsis

The physiological mechanism of sepsis is a complicated process due to stochastic behavior of individual variable. A proper understanding of dynamics among variables is important to interpret the reasoning for the state of the disease. There exists a few diagnostic criteria to diagnose sepsis but each has its limitations: SIRS criteria are known to have poor specificity [18], qSOFA have low sensitivity [19], Modified Early Warning Score (MEWS) have limited accuracy [20], and SOFA is complex. It requires knowledge of four laboratory results, ventilation support and administration of antibiotics.

Additionally, all the aforementioned diagnostic criteria assume that each clinical variable is independent. However, this assumption does not hold true in reality. It is possible that the state of one organ affects the state of other organs. For example, pneumonia causes lung infection but it is possible that the impacted lung may cause dysfunction in other respiratory system organs, distress to the kidneys, and cognitive dysfunction due to fever. Therefore, the interconnection among body organs necessitates the development of models that can capture interrelation among organs. As such, the first problem statement is ***"how to develop an easy-to-use CDSS for sepsis that facilitates balanced sensitivity and specificity and captures organ interrelation ?"***

4.2 A Framework to Develop Mortality Progression Models

Mortality progression monitoring is instrumental to early diagnosis and to composing efficient treatment strategies. Mortality progression models assess the risk of mortality throughout a patient's hospital stay using longitudinally/temporally observed clinical signs such as heart rate, blood pressure etc. The risk of mortality varies over time; therefore, mortality progression provides relevant temporal insights such as time of change in patients' health and time duration for a disease being in a specific state. The current state of the art mortality prediction models uses aggregated (mean, median or peak) clinical measurements [21, 22]. Goldstein et al. (2017) showed in their comprehensive review paper on risk prediction models using EHR that 93% of studies do not leverage longitudinal information present in EHR data [23]. These longitudinal data capture important variations in clinical signs and can be used to develop mortality progression models. From 2009 to 2014, the number of hospitals utilizing a basic EHR system increased from 12% to 76% [24], but EHR data also present its challenges: sparsity of lab measurements, missing clinical measurements and unrecorded intermittent severity of disease during the hospital stay. Therefore, designing a framework to develop mortality progression models using EHR is essential. Hence, the second problem statement is ***"how to develop a framework to design mortality progression models using real-world medical records ?"***

4.3 Initialization of the Learning Algorithm of HMM

In Section 4.2, we emphasize the importance of developing mortality progression models using temporal data. There exist several methods to analyze temporal data such as HMMs and recurrent neural networks. Although HMMs are primarily used to solve speech recognition problems, these methods hold a great promise to solve problems of other applications as well. A salient feature of HMMs is that they facilitate interpretation. However, the learning of HMM is computationally expensive. Due to a high volume of data, the training of HMM is the biggest impediment to its usability. The speed of the learning algorithms is dictated by the initial values. The problem of initialization becomes prominent as the number of variables grows. For a large data set, the algorithm converges slowly and gets stuck at a local solution. Therefore, the third problem statement is ***"how to address the***

issue of slow convergence and local solution of HMM learning algorithm ?”

4.4 Early Bedside Sepsis Scoring Criteria

Identifying sufficient early bedside sepsis scoring criteria is paramount to timely treatment. To detect the disease in early stages, many early diagnostic criteria were proposed [12, 25, 9, 26]. The two early diagnostic criteria, SIRS and qSOFA (Table 1.1), are predominantly used in clinics to assess the criticality of disease. The foundation of SIRS relies on inflammatory response to infection while the basis of qSOFA relies on the organ failures. The presence of a criterion at the designated threshold yields a score of 1, otherwise 0. SIRS scores range from 0 to 4, with scores of 2 indicative of SIRS and, subsequently, an increased likelihood of mortality. qSOFA scores range from 0 to 3, with scores of 2 indicative of high risk for mortality. Of the seven indicators between the two assessments, only respiratory rate is common to both, with the threshold slightly higher in qSOFA.

Table 1.1: SIRS and qSOFA criteria.

Criterion	Threshold	
	SIRS	qSOFA
Body temperature	< 36°C or > 38°C	
Heart rate	> 90 beats/min	
While blood cell count	< 4K/μ L or $>$12K/μ L	
Respiratory rate	> 20 breaths/min	≥ 22 breaths/min
Systolic blood,pressure		≤ 100 mmHg
Glasgow Coma Scale		≤ 13

Singer et al. (2016) proposed a paradigm shift by moving the focus of sepsis diagnoses from inflammatory response to organ failures [12]. The new definitions tied the risk of mortality in suspected infection patients with risk of sepsis. The performance of proposed qSOFA criteria was compared against SIRS. The authors argued for using qSOFA in clinics because of its high specificity. However, several practitioners and researchers argued against accepting qSOFA because of its low sensitivity that could lead to many patients undiagnosed.

Given the necessity of early sepsis detection and divided opinion among researchers, a detailed comparison of SIRS and qSOFA can establish a clarity in a practitioner's mind for the use of more accurate diagnostic criteria. Therefore, the fourth problem statement is ***"which scoring system between SIRS and qSOFA is better ?"***

In summary, this dissertation addresses the following problems:

1. How to develop an easy-to-use CDSS for sepsis that facilitates balanced sensitivity and specificity and captures organ interrelation ?

2. How to develop a framework to design mortality progression models using real-world medical records ?

3. How to address the problem of slow convergence and local solution of HMM learning algorithm ?

4. Which scoring system between SIRS and qSOFA is better ?

5 Contributions

As mentioned above, this dissertation is a compendium of four studies. Here, we summarize the contributions of each study:
Study 1 contributions:

- Develop a sepsis predictive model with high sensitivity and high specificity by utilizing new definitions of sepsis

- Identify relevant biomarkers that enable quick and easy diagnosis

- Capture probabilistic interrelations among biomarkers rather than considering them independent to better evaluate the risk of mortality

- Propose an imputation method that is suitable for electronic medical records where the majority percentage of data is missing

Study 2 contributions:

- Propose an HMM framework to develop mortality progression using EHR data
- Underline the advantages of using temporal methods over non-temporal techniques
- Extract relevant time information to compose better treatment strategies and efficiently utilize hospital resources
- Integrate the principals of multinomial regression to speed up the learning of HMM

Study 3 contribution:

- Propose an initialization method for the HMM learning algorithm
- Employing additional information available in data to efficiently design a multi-stage disease progression model
- Provide a time-efficient approach to initialize HMM learning for the data size grows

Study 4 contributions:

- Determine the early bedside diagnostic criteria using machine learning and statistical approaches
- Identify the most important predictor for sepsis

The remainder of this dissertation is organized as follows. Chapter II discusses the literature review to explore the existing tools and methods. Chapter III implements a CDSS for sepsis using Bayesian networks (Problems 1). This chapter also includes the comparison of the proposed model with alternate clinically acceptable models. Chapter IV describes the framework to develop mortality progression models using HMM and EHR data (Problem 2). Chapter V explains the proposed initialization method for the HMM (Problem 3). Finally, Chapter VI uses statistical and machine learning methods to compare the prognostic accuracy of SIRS and qSOFA (Problem 4).

Chapter II

LITERATURE REVIEW

If I have seen further, it is by standing on the shoulders of giants

-Issac Newton

1 Statistical and Machine Learning Methods

Statistical and machine learning methods can be categorized into two classes: non-temporal and temporal. The applications of both types of methods in healthcare are explored and discussed below.

1.1 Non-temporal Methods

The use of non-temporal machine learning and statistical methods in combination with aggregated value (mean, median, or peak) of clinical signs is growing. Matheny et al. (2010) used peak or mean value (depending on the type of lab) to architect a risk stratification model for hospital-acquired acute kidney injury using logistic regression [21]. Saltzman et al. (2011) developed an in-hospital mortality prediction model using decision trees [27]. The authors used clinical measurements recorded at the time of admission into an emergency department. Ramchandran et al. (2013) proposed a mortality prediction model for cancer patients using logistic regression [28]. To implement the model, the authors used measurements recorded within the first 24 hours of the admission. Instead of developing a specific disease mortality model, Tabak et al. (2013) developed a generic mortality prediction model using 23 lab results and 2 demographic measurements [22]. The authors developed the logistic model using the first measurement of hospital visits. All of these proposed models use either aggregated values or a specific measurement (first or last observation during hospital visit) for model implementation. Due to the use of a single value, these models do not capture the dynamic behavior of the diseases.

Bayesian Networks (BN) are also popularly applied in healthcare; these approaches are equipped to manage linear, non-linear, stochastic and combinatorial relations among variables in order to capture the complex mechanism of medical problems. Therefore, we investigate these models in detail. Aronsky and Haug (2000) developed a diagnostic system to identify patients likely to have community-acquired pneumonia using Bayesian Network (BN)s [29]. Burnside et al. (2000) devised a probabilistic graphical model to differentiate benign and malignant cancer tumors [30]. Stojadinovic et al. (2009) applied BNs to predict malignant thyroid nodules and achieved high discrimination capabilities with AUROC [31] of 0.88 using clinical variables acquired through non-invasive methods [32].

Nissan et al. (2010) developed a model using BNs to predict the growth of tumors in the sentinel lymph node, which is a challenging task for oncologists even with the support of radiologic mapping [33]. Identification of thyroid nodules is a difficult problem as it requires unnecessary diagnostic surgery and among them only 5% of patients have malignant thyroid nodules. Pang et al. (2004) used the tongue images to diagnose common diseases [34]. The proposed computerized model, designed using BN, extracts two kinds of features from these images - chromatic and textural from tongue images, and these features are used to predict diseases. The authors employed the proposed model to interpret data from 455 patients and 13 classes of disease. The results showed that the model performed with a moderate accuracy of about 75%.

Kahn et al. (1997) developed a clinical diagnosis model (MammoNet) to estimate the probability of tumor malignancy [35]. MammoNet requires 5 patient's demographic information, 2 physical signs and 15 features extracted from images of mammography. The proposed model performed well with an accuracy of 0.88. BNs are also applied in the dental domain and in detecting lung cancers. Sesen et al. (2013) used BN to generate insights about lung cancer care, including information about the survival probabilities and treatment selection recommendations [36]. The authors used both expert based approaches and algorithmic approaches to learn the structure of BN. The structure elicited from algorithmic approach (AUROC = 0.81) performed better than the structure obtained from the experts' opinion (AUROC= 0.75).

Mago et al. (2008) developed BNs for the dentist to diagnose dental caries and decide the course of treatment [37]. There is another similar study on identifying the presence or absence of dental caries using BNs [38]. The main feature of this study is that it also incorporates the spatial behavior of caries. BNs were not only applied for diagnostic purposes but they were also used for the management of hospital resources. Marshall et al. (2001) developed a BN model to predict the duration of stay and destination of discharge to manage the resources of geriatric hospitals [39]. Acid et al. (2004) developed BNs with the combination of different structure learning algorithms to understand the interaction among variables that influence the patients' management [40]. The non-temporal methods work well and provide informative insights about the future outcome. However, the performance of models can be improved, and additional insights on the progression of the disease can be drawn by utilizing the temporal information available in the EHR

1.2 Temporal Methods

Temporal methods are applied to understand the dynamic behavior of diseases. Henry et al. (2015) used Cox proportional hazard modeling techniques to develop the septic shock progression model. This study developed a targeted real-time warning score (TREW) to predict septic shock in the early stages [7]. Although authors utilized longitudinal data using the proportional hazards model, this modeling technique does not incorporate past information when estimating the mortality risk. The principal of a proportional hazards model lies in the number of subjects that survive over the study time window. There are only a few studies focusing on developing mortality progression, but several studies have been proposed to model disease risk progression using time series techniques. Peelen et al. (2010) investigated the sequence of organ failure in ICU patients using DBN [41]. Cai et al. (2015) developed the non-disease-specific progression model using DBN [42].

A derivative of probabilistic graphical models, HMM enable modeling the longitudinal behavior of disease. A few researchers have applied HMM to model the progression of diseases. Liu et al. (2013) proposed a 2D-continuous time HMM to model glaucoma progression using longitudinal data [43]. Sukkar et al. (2012) developed a slowly progressing disease risk model using HMM [44]. This work set the foundation of one of our works (Chapter IV) where we propose a mortality progression model rather than disease progression using EHR data. The primary difference is that Sukkar et al. (2012) did not test the proposed framework on EHR data. The study collected data from brain images. With the rapid growth of EHR data, developing the framework to establish a mortality progression model using EHR is instrumental. The secondary difference is that the author did not comment on the significance of using an HMM model over non-temporal techniques. In Chapter IV, we bridge this gap by proposing a framework that can model mortality progression and can be easily replicated for any disease. The alternate approach to exploit longitudinal data available in EHR is deep learning.

Deep learning methods stem from neural networks. In recent days, the application of deep learning in healthcare received special attention. Garske (2018) adapted deep learning to predict diabetes from EHR data [45]. Che et al. (2015) applied deep learning to determine the physiologic patterns associated with the clinical phenotypes [46]. Esteva et al. (2019) summaries the multiple healthcare domains where deep learning can make a significant difference [47]. They pointed out that the medical imaging analysis, unstructured data analysis using natural language processing and reinforcement learning for robot-assisted surgery.

Kam and Kim (2017) developed sepsis detection models with deep learning and compared the performance of the new deep learning method with regression. The authors applied temporal deep learning architecture (long short-term memory (LSTM) and showed that LSTM improved the performance compared to regression models. The results indicate the potential of improving the performance of sepsis prediction models by exploiting the longitudinal data available in EHR. This study used open-source EHR data from MIMIC-III [48].

Deep learning methods are capable of handling time series data. Hammerela et al. (2015) used time-series data obtained from wearable sensors to predict the severity of Parkinson's disease [49]. Lipton et al. (2015) employed the derivative of deep learning known as LSTM to determine the multi-label diagnosis using ICU time-series data [50]. Choi et al. (2017) utilized time-stamped EHR data to diagnose heart failure in the early stages. Although deep learning methods are powerful tools to unearth the hidden patterns, the lack of interpretation is a hurdle to their real-world implementation.

In this dissertation, we focus on employing HMM to exploit longitudinal data available in EHR. The applications of HMM in healthcare hold a great promise because it can model the stochastic nature of the diseases and equips healthcare providers with interoperability.

Therefore, we studied HMM in details. HMM was introduced in 1966 by Baum and Petie [51]. It was first successfully applied to model the real world problem of speech recognition. In 1972, Baum devised a forward-backward recursion to estimate the parameters of a HMM [52]. This algorithm is also widely known as a Baum-Welch algorithm in honor of Lloyd Welch. The study on automatic speech recognition using HMM by Rabiner popularized the use of HMM in other applications as well [10]. A detailed introduction of HMM can be found in work done by Ephraim and Marhav [53].

Vairavan et al. (2012) proposed a model using HMM and logistic regression to predict the mortality in ICU [54]. In the proposed method, the output of HMM used as one of the features of logistic regression. This study used supervised HMM with two hidden states and estimated the HMM parameters using maximum likelihood method.

Using a combination of Temporal Boot-strap method and HMM, Li et al. (2013) developed disease progression model with cross-sectional data [55]. For estimating HMM parameters, the study used iterative Expectation-Maximization method. Chen and Pham (2013) developed a dementia detection tool using HMM framework [56]. In this work, the MRI images were used to extract the sequence of observations. The procedure to develop a clinical support system can be summarized as follows. First, identify gray matter from MRI images. Later, generate time series data by the distance measured from subsequent outer boundary points to the centroid of gray matter. Now, use the obtained sequence of distance to estimate the parameters using the Baum-Welch algorithm. The proposed framework was able to identify mild Alzheimer's disease in the early stage with a detection rate of 0.90. Nicola et al. (2011) modeled the progression of liver cirrhosis using HMM [57]. Stanculescu et al. (2014) used Autoregressive HMM to detect the neonatal sepsis without the need for lab tests [58]. In this study, using clinical experts, the unsupervised HMM problem was translated into supervised HMM to employ the maximum likelihood for estimation of transition matrix and emission matrix. Khorasani and Daliri (2014) developed a classifier model for Parkinson's disease using HMM [59].

Smyth (1997) suggested the idea of controlled initialization to improve the performance of Forward-Backward algorithm [60]. The authors utilized *K-means* clustering for initialization of parameters. This approach works well for a small set of sequence of observations but could be computationally expensive for a large set of sequence of observations. Liu et al. (2014) [17] proposed a Segmentation and Clustering (SnC) approach for initialization of Baum-Welch algorithm. The authors devised four steps initialization procedure. Step 1 includes segmentation of data sequence based on similarity of observations. In Step 2, the segments obtained from Step 1 are clustered using $k-mean$ clustering. Step 3 is an extension of Step 2. In this step, we estimate the optimal number of clusters to explain the data. In Step 4, the identified clusters serve as hidden states and are used to estimated initial parameters using maximum likelihood method.

The current procedures for the initialization of HMM learning algorithm rely on heuristic approaches. There exists a need to establish a computationally efficient method for initialization of HMM learning algorithm. We proposed an initialization framework in Chapter V.

In summary, given the successful applications of probabilistic graphical modeling in medical domain and benefit of interpretability, we aim to contribute in two aspects of probabilistic graphical modeling: *Application* and *Methodology.* In the application aspect, we develop CDSS using graphical modeling techniques to improve on existing tools by capturing the interactions among physiological signs. Other classification methods such as logistic regression and decision tree fail to capture the dynamics of variables that are predominantly present in the clinical domain. In the methodological aspect, we devise a framework to improve the parameter learning of graphical models. The current HMM learning algorithms suffer the deficiencies of local maxima and slow convergence.

2 Sepsis Diagnostic Criteria

Sepsis develops into septic shock when body organs start behaving abnormally. According to the new definition proposed in 2016, the organ dysfunction is identified using the increment of 2 or more in the SOFA score. Due to technological advancements in pathobiology and epidemiology of sepsis, sepsis definition and diagnostic procedures have been revised to control the high mortality and growing incidence. The physiological mechanism of sepsis is a complicated process. The complication involves the effect of individual clinical variables as well as the random variations. In addition, symptoms of sepsis are similar to other conditions, therefore, the diagnosis of sepsis in the early stage is a challenging problem for practitioners. Research has shown that every hour delay in antibiotic therapy is associated with an increased risk of mortality by 5-10% [61]. Therefore, early identification is critical for the management of sepsis patients. Early identification requires a proper understanding of dynamics among variables to reason the state of a disease. However, the multivariate stochasticity and non-linear behavior of body response to sepsis make sepsis' physiology difficult to understand by the human.

In 1991, Sepsis was first formally defined by the consensus conference on sepsis and organ failure [25]. Participants attempted to overcome the problems of multiple definitions and meanings by addressing critical questions, such as whether sepsis was the appropriate term for a condition unaccompanied by infection and if organ failure was an accurate description of dysfunctions that may not necessarily be complete failure [62]. Rather than formally defining sepsis, the conference resulted in a reinterpretation of sepsis and related disorders as SIRS, a term describing the widespread inflammation or response associated with a range of disorders, including infection, ischemia, and hemorrhagic shock.

Sepsis as a diagnostic term was to be reserved only for patients with infection, making sepsis a subcategory of SIRS rather than SIRS being indicative of sepsis. SIRS scores range from 0 to 4, with scores of 2 indicative of SIRS and, subsequently, an increased likelihood of mortality. The intent of the SIRS definition was to broaden the thinking about the inflammatory response, including its etiology and measurement. However, several researchers have shown that the SIRS criteria fail to adequately differentiate patients by the level of mortality risk due to high sensitivity and low specificity, making the assessment a poor indicator of illness severity [18, 63, 64]. Such imprecise measurement has led to numerous obstacles for clinical trials [64, 65, 66, 67].

With advances in medicines understanding of the pathophysiology of sepsis coupled with systemic inflammatory response occurring in both the presence and absence of infection [68], sepsis definitions were evaluated in 2016 by the Third International Consensus Definitions for Sepsis and Septic Shock (Sepsis-3) task force. The committee proposed a paradigm shift in the definition of sepsis by moving away from the emphasis on systemic inflammation to *life-threatening organ dysfunction caused by a dysregulated host response to infection* [12]. The new qSOFA criteria were designed to be a predictor of mortality used with patients with suspected infection. qSOFA scores range from 0 to 3, with scores of 2 indicative of high risk for mortality. Of the seven indicators between the two assessments, only respiratory rate is common to both (SIRS and qSOFA), with the threshold slightly higher in qSOFA.

Further investigations found that although qSOFA has a stronger association with mortality than SIRS for predicting in-hospital mortality among Emergency Department (ED) patients, it had lower sensitivity [69]. Due to the poor sensitivity of qSOFA, patients with sepsis might remain undiagnosed. This misdiagnosis in the early stage could lead to life-threatening outcomes as timely treatment is critical [70]. However, the high sensitivity of SIRS could lead to unnecessary burdening of ICUs due to improper referrals. Researchers differ in their preference between sensitivity and specificity while selecting diagnostic criteria. Freund et al. (2017) argued that the high specificity of qSOFA criteria make it suitable to replace SIRS for efficient stratification of sepsis patients in the ED [71]. On the other hand, Akim et al. (2017) preferred sensitivity and presented results against the use of qSOFA [72].

From the above discussion, it is evident that there exists a dilemma about the selection of best early bedside sepsis scoring criteria. In Chapter VI, we elicit evidence using statistical and machine learning approaches to bring clarity about the selection of best diagnostic criteria.

3 CDSS for Sepsis

Many studies have been dedicated to the development of decision support systems for sepsis identification to aid practitioners in diagnosis. Mathe et al. (2009) developed a sepsis surveillance system that monitors specific lab and vital signs in real time and visualizes them on a screen [73]. The system notifies health care providers in case of observed abnormalities. However, this tool does not analyze the information recorded and provide any suggestion about the future outcome. Brandt et al. (2015) derived a flow chart for sepsis diagnosis and developed an electronic sepsis surveillance system. The major drawback of the system was that it had low specificity [74]. Amland and Hahn-Cover (2016) designed a computerized early warning tool for sepsis diagnosis [75]. This tool generates two kinds of alerts: SIRS alert and sepsis alert. The results showed that the system was able to alert physicians with a median of 8.6 hours earlier than the time of suspicion of infection noted by doctors. The limitation of this tool was that it triggered many false positive alarms. Only 25% of patients that were alerted required the physician's attention.

Lukeaszewski et al. (2008) used Reverse Transcription Polymerase Chain Reaction (RT-PCR) in combination with neural networks to predict the onset of sepsis [76]. RT-PCR is a commonly used test to determine genetic diseases. The study was able to predict sepsis with 95% accuracy. However, extracting input information to employ such models is not feasible in healthcare practice. The input information requires blood sampling and expensive gene sequencing reading. Henry et al. (2015) proposed a Target Real-time Early Warning

(TREW) score to predict septic shock using the Cox-proportional hazard model [7]. This support system triggers a warning of septic shock based on the change in clinical variables over time. The proposed model achieved the discrimination ability of 0.83. The limitation of this analysis is that it does not capture the effect of the previous condition of disease and also does not comment about the mortality risk. Most expert systems for sepsis consider clinical variables independently. However, it is highly unlikely that physiological variables are independent.

A few attempts have been made to apply Bayesian networks to understand the dynamics of sepsis. Mani et al. (2014) developed a predictive model using non-invasive clinical observations for sepsis prediction among neonates (infants during the first month) [77]. The authors compared the performance of nine machine learning algorithms and found that naive Bayes, with Area under Receiver Operating Characteristic (AUROC) curve of 0.79, outperformed other learning techniques. Gultepe et al. (2014) applied the Bayesian network structure learning algorithm (hill-climbing) to unearth the interaction among clinical variables used for sepsis diagnosis [78]. Along with developing the Bayesian network structure to understand the dependencies among variables, the authors also proposed a model for mortality prediction using naive Bayes and Support Vector Machine (SVM). With AUROC of 0.73, SVM based model performed better than naive Bayes. The possible reason for the poor performance of naive Bayes model was that the dataset was small (number of patients was 741 where only 151 had sepsis and others were of the control group). Nemzek et al. (2015) applied Bayesian networks to the protein and cellular components of sepsis-related organ failure and elucidated the intricate relationships among the variables that cause lung dysfunction [79]. Although this study provided detailed insights for the development of sepsis-related lung dysfunction, it used the protein and cellular components that are not directly accessible and required laboratory analysis.

Some studies use the variant of Bayesian networks known as DBN to improve the performance of sepsis diagnostic system. Peelen et al. (2010) proposed a clinical support system for medical practitioners to predict organ failures in intensive care units (ICUs) [41]. They developed three different models of the problem, and each model is an advancement of the previous. Model 1 takes into account the number of organ failures, Model 2 considers the transition of one organ failure to other, and Model 3 incorporates the dependency of one organ failure to other organs, i.e, independent assumption was relaxed. The authors captured the dynamic nature of patients' recovery in ICU using a Markov chain. The states in the Markov chain denote various conditions of patient's stay in the ICUs. The transition probabilities are calculated using logistic regression. The limitation of the study is that the authors used data only from ICU admissions; therefore, it was not clear to what extent the findings could be generalized to non-ICU admissions. Another study by Sandri et al. (2014) developed a model to predict the sequence of organ failure in a patient using DBN [80]. The main limitation of this study was that the organ dysfunctional time was not known, therefore, authors used orders of organ failure based on physicians' recommendations.

In Chapter III, we capture important interactions among biomarkers to develop a model for predict sepsis. The inclusion of dynamics among biomarkers differentiates our model from existing models. In addition, the existing sepsis diagnostic models lack in providing the balance between sensitivity and specificity [18, 19]. Due to unbalanced nature between sensitivity and specificity, both qSOFA and SIRS were criticized and considered unhelpful. Therefore, practitioners require a more sophisticated diagnostic system. MEWS provides a better alternative, but its sensitivity, specificity, and AUROC are not strong. Although SOFA provides high performance accuracy, its complexity is the biggest challenge in resource constrained environments [81]. In the next chapter, we develop an easy-to-use expert system using Bayesian network classifier that captures the interaction among clinical variables and facilitates a balance between sensitivity and specificity by selecting the optimal set of biomarkers.

Chapter III

SEPSIS MORTALITY PREDICTION USING BAYESIAN NETWORKS

This chapter explains the development of a CDSS for sepsis using Bayesian networks. The novelty of this work is that we attempt to capture the physiological interactions among organs. Designing of a CDSS is a multi-step process. The procedure can be summarized using Figure 3.1. We used the following steps to build a predictive model: 1) data extraction, 2) data preprocessing (imputation, feature selection, and discretization), and 3) building Bayesian network. An optimal set of variables was identified using feature extraction, and then these variables were used for further construction of the model. Using the combination of elastic net and recursive partitioning, irrelevant or redundant variables were eliminated. After pruning the search space, the selected biomarkers were discretized using Hartemink method, suitable for using with Bayesian classifiers [82]. Following the discretization, the interactions among selected biomarkers were captured by utilizing a specific class of Bayesian classifiers.

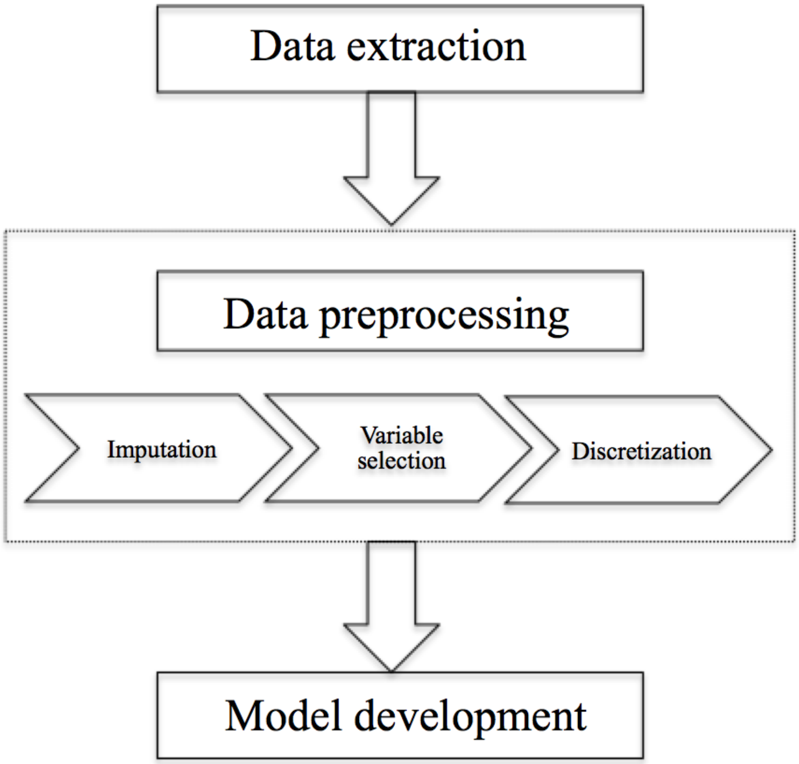


Figure 3.1: Model development procedure

1 Method

1.1 Data Extraction

This study used de-identified data from Cerner Corporations HIPAA-compliant Health Facts database. The database comprises EHR for 379 million patient visits, about 63 million patients, from 480 affiliated hospitals across the United States. These comprehensive clinical records for individual patient visits include demographic, vital signs, laboratory results, medication, diagnosis codes (both ICD-9 and ICD-10), pharmacy, procedures performed, and date- and time-stamped information on admission and discharge. This dataset is one of the largest EHR in the United States [83].

The population of suspected infection was selected using the gold standard explained in Sepsis-3 [84]. The gold standard includes a combination of culture drawn (Event 1) and antibiotics administration (Event 2) within a defined time interval. There are two scenarios illustrated in Figure 3.2: 1) Event 1 occurs first, then Event 2 occurs within 72 hours, 2) Event 2 occurs first, then Event 1 occurs within 24 hours. The study cohort includes patient visits that satisfy either of two aforementioned scenarios. The time of infection is the time of occurrence of the first event. The selection of culture and antibiotic was based on the consultation with practitioners. In the final dataset, we had 16,909 patient visits. The outcome variable was discharge type (expired or non-expired).

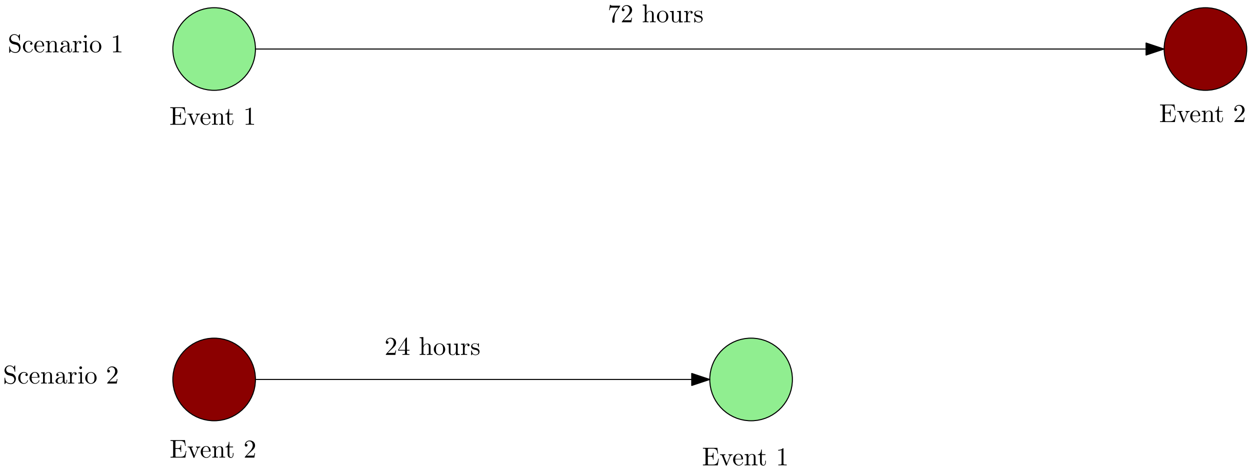


Figure 3.2: Data selection scenarios

1.2 Imputation

This section explains the proposed *left-center-right* imputation method and compares it to alternate imputation approaches. Imputation is a critical step to develop an effective clinical model using EHR data. EHR data are collected for the billing purpose rather than performing analytics, therefore, the data has a lot of missing clinical measurements. Also, variables that are easy to measure such as routine vital signs are frequently present while others, such as lab tests, are sparsely available. Since we wanted to capture the interactions among variables, it was important to use values recorded in the same time intervals instead of taking the aggregate values such as *mean* or *median* over the hospital stay of the patient.

Considering measurements at the same time interval is especially important for sepsis as it progresses aggressively, therefore aggregate measurements are clinically less meaningful.

In the proposed imputation method, we utilized observed biomarker measurements to impute unobserved biomarkers of individual patient's visit. For this study, we divided the patient's length of stay after the first suspicion of infection into 24-hour (or one day) windows. We selected 24-hour windows because the popular existing criteria such as SOFA uses the worst value recorded in 24 hours [9]. The imputation procedure is easy to explain with the example shown in Figure 3.3. In Figure 3.3, t = 0 is the time at which patient is suspected to have infection. The subsequent hospital stay is divided into 24-hour intervals until the discharge of the patient (i.e., t = 8). For a specific variable in a patient visit, suppose we know the observation for t = 2, 5 and 7 (X_1, X_2 and X_3). To impute the values in remaining intervals, we use carry-forward (*right* in Figure 3.3) and carry-backward (*left* in Figure 3.3) methods along with the mean in intermittent intervals (*center* in Figure 3.3). In the carry-forward and carry-backward methods, the most recent value is carried forward or backward, respectively, to fill empty intervals. We referred to this imputation method as *left-center-right* for convenience. Using observed data at t = 2, 5 and 7 (shown in black), the other time intervals were imputed (shown in blue).

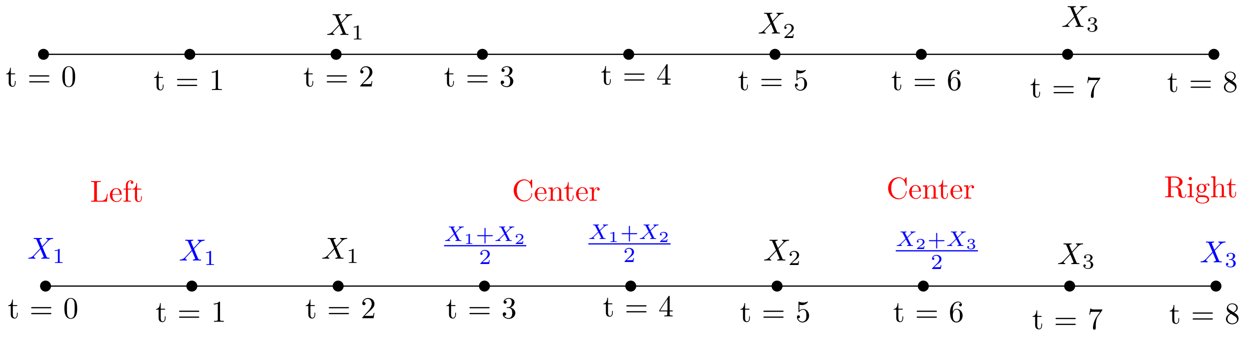


Figure 3.3: *Left-center-right* imputation. t = 0, time of suspicion of infection; t = 8, time of discharge

We compared the imputed data generated from the proposed imputation method to the *mean* and the *median* imputation. This comparison was established for each individual variable. For illustration purposes, let us consider *bilirubin*. The same population data (explained in Section 1.1) were used for this comparison purpose. We expect that good imputed data should follow the density plot of data without imputation. Figure 3.4 shows the density plot of three imputation techniques along with density plot of data without imputation. The density plot for the *left-center-right* method closely aligns with the data without imputation as desired. The density plots obtained from other imputation techniques changed the shape of distribution and showed peaks at different points than the original mean. The reason for peaks at different points is attributed to the sparsity of EHR data. Therefore, we avoided the use of *mean* or *median* imputation for our analysis.

1.3 Variable Selection

Variable selection is the most critical part of the analysis because most clinical results have lead time and cost associated with it. Also, Capan et al. (2018) has shown that not all organ failures are equally associated with sepsis mortality [85]. Therefore, to keep our model simple, and at the same time maintain high performance, we need to establish the best subset of variables. Table 3.1 includes the names and types of the variables initially selected in this study. The primary selection of biomarkers was based on the literature and discussion with practitioners. Later, the best subset of biomarkers among the primarily selected biomarkers was obtained using the machine learning techniques.

We used both elastic net [86] and recursive partitioning [87] for variable selection. The two methods stand on different principals. The elastic net pulls the coefficients of unimportant variables towards zero by minimizing the sum of square errors and regularization term. We used the elastic net rather than the stepwise selection because it facilitates shrinkage and avoids over-fitting by using the regularizing factor. The shrinkage implies reducing the complexity of the model by dragging the weights of unimportant biomarkers to zero. Shrinkage

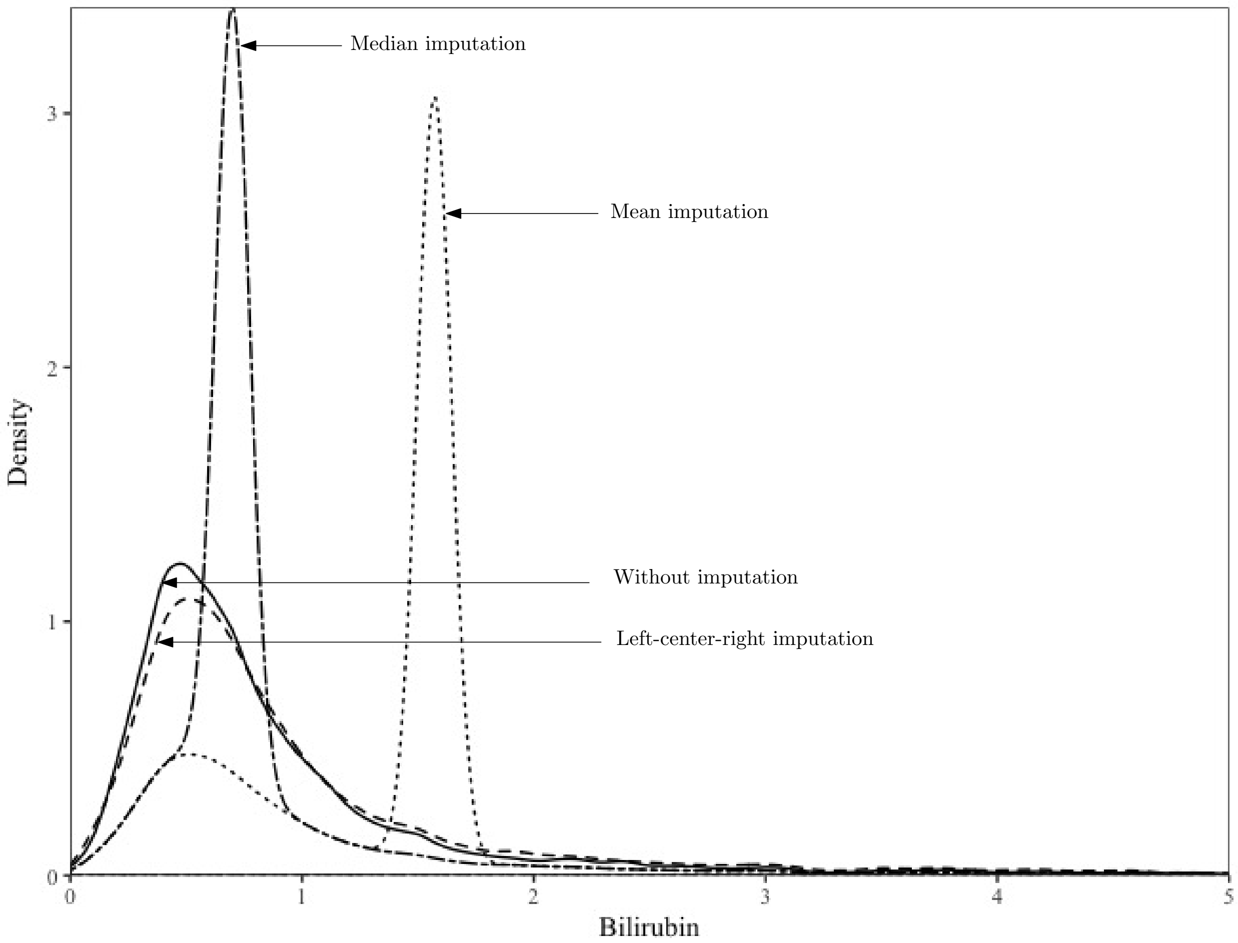


Figure 3.4: Density plots for different imputation techniques

is an important characteristic for clinical applications because each biomarker (lab test or vital signs) is associated with a significant cost. The recursive partitioning focuses on minimizing the impurity present in the data by partitioning. For the final model, a conservative approach was applied and only variables that were suggested by both variable selection methods were used.

Zou and Hastie (2005) proposed elastic net regression that utilized both *L1 norm* and *L2 norm* [86] The objective function (or prediction error) for the elastic net is given by Equation (III.1). The variable of interest is the weight vector at which objective function is minimum.

Table 3.1: Clinical variables

Description	Associated organ dysfunction indicator	Type
Body temperature	General	Vital
Mean arterial pressure	General	Vital
Systolic blood pressure	Cardiovascular	Vital
Glasgow Coma Scale	Nervous	Vital
Respiratory rate	Respiratory	Vital
Glucose	General	Lab
Platelet count	Coagulation	Lab
Prothrombin time (PT)	Coagulation	Lab
White blood cell count	Inflammatory	Lab
Bilirubin	Liver	Lab
Creatinine	Renal	Lab
Age	NA	Demographic
Gender	NA	Demographic

$$\hat{w} = arg\min_{w}\left\{\sum_{n=1}^{N}(Y^n - w^T X^n)^2 + \lambda\alpha\|w\|_1 + \frac{\lambda(1-\alpha)}{2}\|w\|_2\right\} \tag{III.1}$$

Here Y^n is the observed output (discharge type) of n^{th} data point, X^n is the input vector of length p (number of biomarkers; $p = 13$) of n^{th} data point, n is the index for admission visit ($n \in \{1, 2, \ldots, N\}$). N is the number of admission visits ($N =$ 16,909). α is a tuning parameter, w is a estimated weight vector, and $\|w\|_1$, $\|w\|_2$ are *L1 norm* and *L2 norm*, and equal to $\sum_{j=1}^{p}|w_j|$ and $\sqrt{\sum_{j=1}^{p} w_j^2}$, respectively. For a specific α, we search for best λ that provides the least cross-validation error (CV-E) with the minimum set of biomarkers.

To obtain an optimal set of variables, we searched for different α ($0 \leq \alpha \leq 1$) and λ. Figure 3.5 shows the plot between CV-E and $\log(\lambda)$ on different α. 5-fold cross-validation was used to avoid the bias due to selection of sample. Each dot (in red) and bar (in gray) in the graph represents the CV-E and one standard deviation from mean, respectively corresponding to a specific $\log(\lambda)$. Two vertical lines in each plot represent λ at which CV-E is minimum and λ above which CV-E is greater than one standard deviation of the minimum CV-E. To compute the optimal number of variables, we used λ above which CV-E is greater than one standard deviation of the minimum CV-E instead of using λ at which CV-E is minimum. This selection of λ enables significant reduction in the size of optimal set of variables with the minimal increase in error. For example, at $\alpha = 1$, minimum CV-E was achieved at $\log \lambda$ = - 7.32 with 12 variables (shown in the top of each plot) and maximum CV-E that is within one standard deviation of minimum CV-E is at $\log \lambda$ = - 5.27 with 9 variables. Therefore, we selected 9 as an optimal set of variables for $\alpha = 1$. We expected as α decreases, sparsity of the model decreases due the nature of Equation (III.1). It is evident from Figure 3.5 that the optimal number of variables increases as α decreases from 1 to 0. The size of optimal set of variables for $\alpha = 0.33$ and $\alpha = 0$ were 10 and 13, respectively (bottom two plots of Figure 3.5). Figure 3.5 depicts number of selected variables on the top of each plot. This number of selected variables is identified using variables with non-zero weights. The clinical variables with zero weights for $\alpha = 0.66$ and $\log(\lambda)$ = -4.3 are struck out in Table 3.2.

In recursive partitioning, the data are recursively partitioned to reduce the impurity present in the data. Each step of the recursive partitioning includes the splitting of data into two subgroups. The split is determined by the variable that reduces the impurity most. There are two methods to measure the impurity: *Gini index* and *Information index* [88].

Gini index is defined as:

$$Gini \quad index = \sum_{i=1}^{2} p_i(1 - p_i)$$

Information index is defined as:

$$Information \quad index = -\sum_{i=1}^{2} p_i log(p_i)$$

Where p_i is the probability of outcome class i. We have two classes in outcome variable (1 = non-expired, 2 = expired).

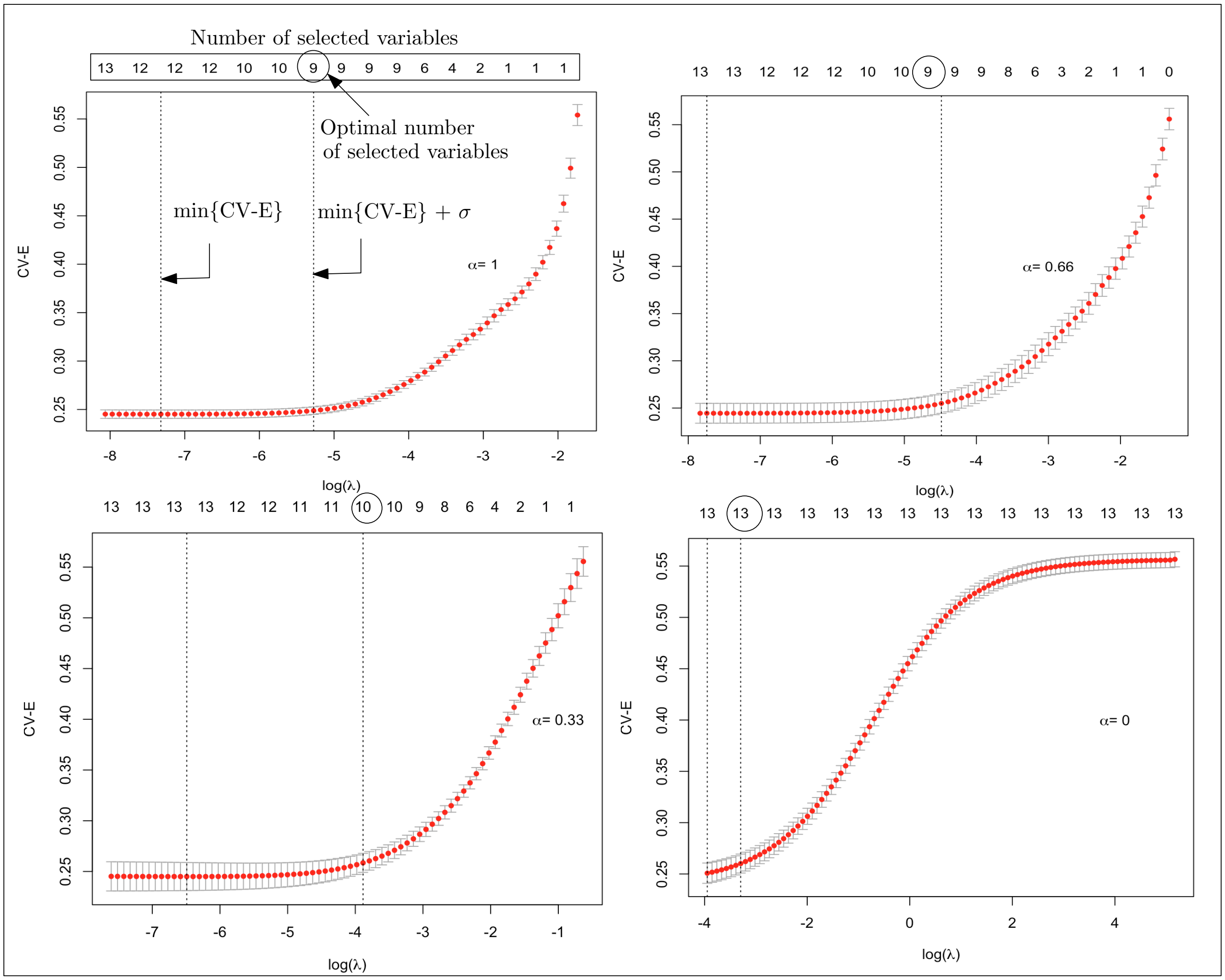


Figure 3.5: CV-E vs. log(λ) on α = 1, 0.66, 0.33, 0

The recursive partitioning can be summarized as follows:

1. Select variable that provides highest information gain (or reduction in impurity). The information gain is calculated using the following expression:

$$\text{Information Gain} = \text{Impurity (before split) - Impurity (after split)}$$

2. Split data using the selected variable

3. Repeat 1 and 2 until information gain is greater than a pre-defined threshold

Table 3.2: Selected variables for α = 0.66 and log(λ) = -4.3; strike out indicates variable having zero weight

Variable	Variable
~~Body temperature~~	Prothrombin time (PT)
Mean arterial pressure	White blood cell count
Systolic blood pressure	~~Bilirubin~~
Glasgow Coma Scale	Creatinine
Respiratory rate	Age
~~Glucose~~	~~Gender~~
Platelet count	

For our data, recursive partitioning using both splitting criteria resulted in the same classification accuracies (0.96). Since we had the same classification accuracies and the research has shown that both criteria disagree only in 2% cases [89], we selected information index for further explanation of the results.

The variable importance of each biomarker obtained by recursive partitioning using information index is shown in Figure 3.6. Figure 3.6 illustrates a significant drop in variable importance from respiratory rate to mean arterial pressure. Therefore, we performed a sensitivity analysis to inspect the change in accuracy for two scenarios: considering only first five important variables (dotted variables in Figure 3.6), considering all variables with non-zero variable importance. The classification accuracy of the model (0.96) with 5 variables (dotted variables in Figure 3.6) was the same as the classification accuracy of the model (0.96) with 9 variables (both dotted and shaded variables in Figure 3.6). Therefore, we considered only five biomarkers from recursive partitioning: Glasgow Coma Scale, creatinine, blood pressure, white blood cell and respiratory rate.

We summarized the variables selection using elastic net (α = 0, 0.33, 0.66, 1) and recursive partitioning in Table 3.3. Although for our experiments, the variables, selected by recursive partitioning, are the subset of variables selected by the elastic net (α = 1), it is not always the case. The variables used for further analysis are shown in bold font in Table 3.3.

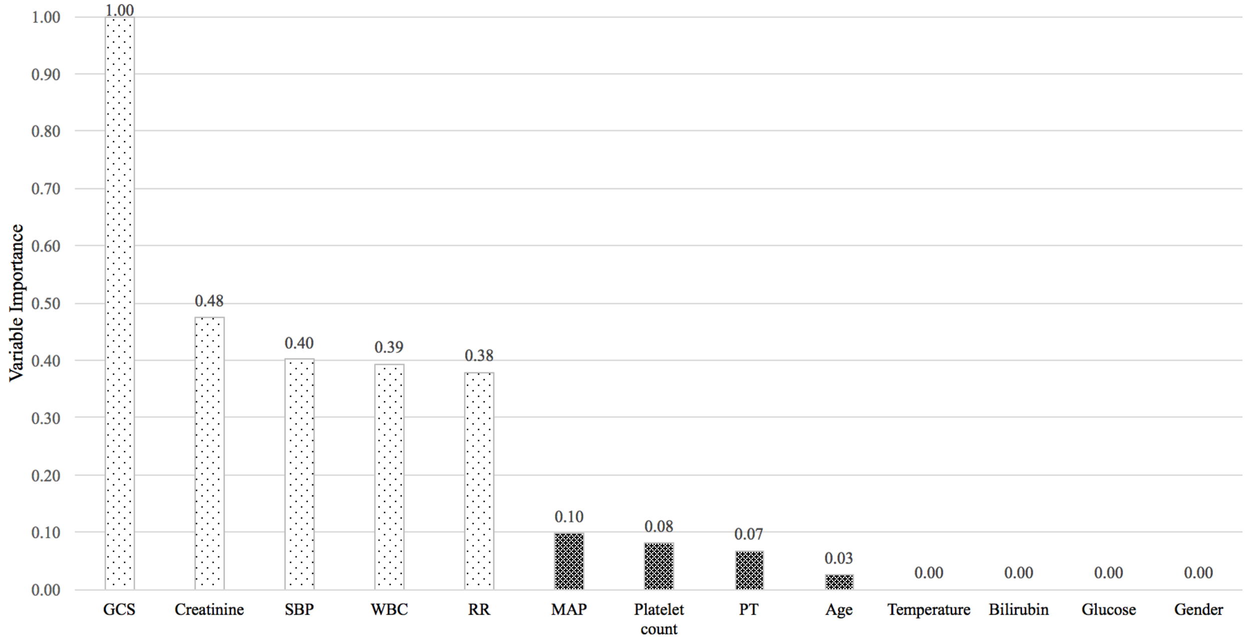


Figure 3.6: Variable importance from recursive partitioning (GCS: Glasgow Coma Scale, SBP: systolic blood pressure, WBC: white blood cell, RR: respiratory rate, MAP: mean arterial pressure and PT: prothrombin time)

1.4 Discretization

Many methods have been developed to discretize continuous variables, and the performance of these methods depends on the shape and the nature of the data. Some approaches such as quantile and equal intervals utilize variables in isolation, while others such as Hartemink [82] consider relationship among variables to decide the cutoffs.

For our data, the correlation analysis uncovers significant dependencies among biomarkers. Figure 3.7 illustrates correlations among biomarkers. The size of the circle represents the strength of the correlation between biomarkers. The cross signs show that the correlation was non-significant. Therefore, a relationship among biomarkers should be utilized for discretization to avoid the loss of information. Hence, we selected Hartemink method to efficiently compute the discrete levels of biomarkers. Another reason why we chose it is that research has shown that Hartemink discretization performs better with Bayesian classifiers [82].

Table 3.3: Final set of selected variables (check mark indicates that variable is selected by individual procedure, bold fonts means finally selected)

Variable	$\alpha = 0$	$\alpha = 0.33$	$\alpha = 0.66$	$\alpha = 1$	Recursive partitioning
Body temperature	✓				
Mean arterial pressure	✓	✓	✓	✓	
Systolic blood pressure	✓	✓	✓	✓	✓
Glasgow Coma Scale	✓	✓	✓	✓	✓
Respiratory rate	✓	✓	✓	✓	✓
Glucose	✓				
Platelet count	✓	✓	✓	✓	
Prothrombin time (PT)	✓	✓	✓	✓	
White blood cell count	✓	✓	✓	✓	✓
Bilirubin	✓	✓			
Creatinine	✓	✓	✓	✓	✓
Age	✓	✓	✓	✓	
Gender	✓				

Hartemink Discretization assumed that the continuous variables are normal by distribution. Therefore, we inspected the density plot of each variable. Using the Box-Cox method [90], the transformed variables were obtained when original variables were deviating from normal, except for the Glasgow Coma Scale sore. For example, the original distribution of platelet count and its transformed distribution are shown in Figure 3.8. The Glasgow Coma Scale score was heavily left skewed, therefore, no better distribution was found using the Box-Cox method. Hence, the cutoffs ($\leq 6, (6, 10], (10, 13], > 13$) defined in the literature were applied for Glasgow Coma Scale [9].

In the Hartemink method, each continuous variable initially is segregated into a large number of levels using any of the traditional methods such as equal intervals. This large number of levels reduces to a specific number, defined by the users, by coalescing selected neighboring pair to a single level. The mutual information between the selected variable and other variables is calculated. The levels that result in a minimum loss of information with any of the other variables is coalesced with neighboring level. For our experiments, we first divided each variable into 10 levels and then coalescing was done with the Hartemink procedure

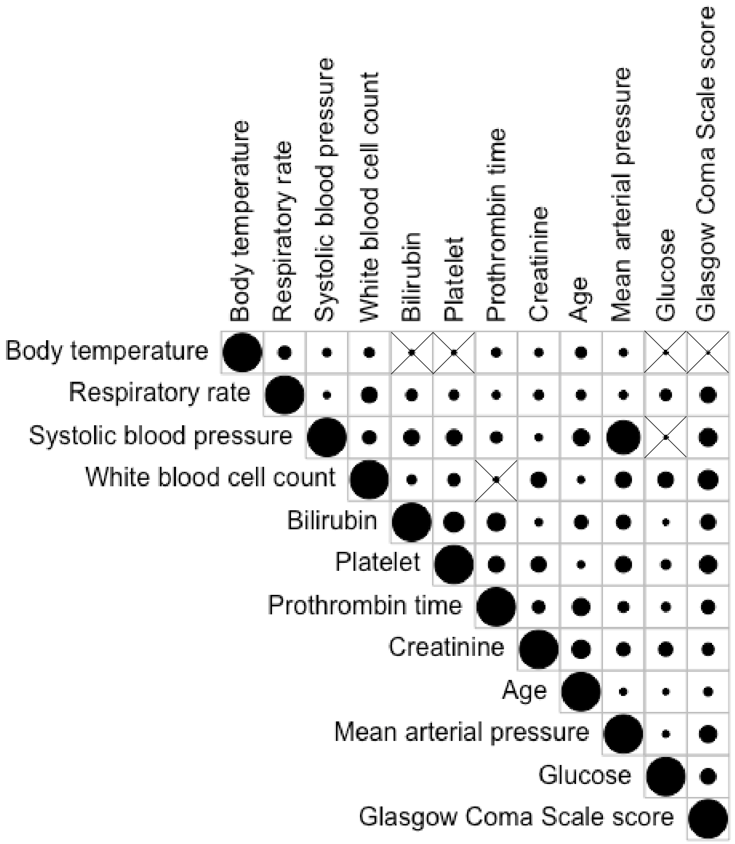


Figure 3.7: Correlation analysis among biomarkers (size of the circle represents the strength of correlation, the cross signs show non-significant correlation)

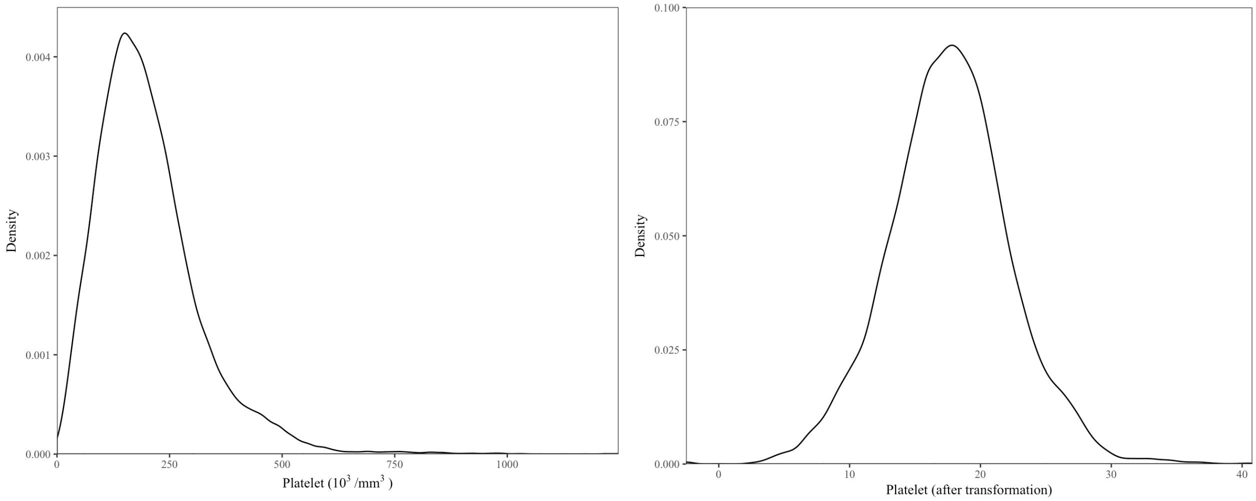


Figure 3.8: Platelet (before and after variable transformation)

until each variable reached three levels. Based on physicians' opinion, three discrete levels facilitate simplicity and meaningful clinical interpretation. Some biomarkers are considered as *normal* at mid-range value and as *abnormal* at either extreme end. The three discrete levels are able to discover the mid-range boundaries.

The computed discrete levels of the selected variables provide important insights. The patterns of mortality over different discrete levels of biomarkers (shown in Figure 3.9) are supported by practitioner's knowledge. For example, practitioners suggested that both low and high respiratory rates are associated with increased mortality. The discrete levels obtained from our analysis also showed the same pattern. However, there exists slight differences in the standard cutoffs and cutoffs obtained from discretization. The standard respiratory rate for a normal patient is 12 to 20 breaths per minute. However, we obtained least mortality for respiratory rate between 16 to 22 breaths per minute. It is possible that generated cutoffs work well for sepsis specifically because the upper cutoff is supported by the recent Sepsis-3 study, where respiratory rate greater than 22 was proposed as the cut-off for increased risk of mortality.

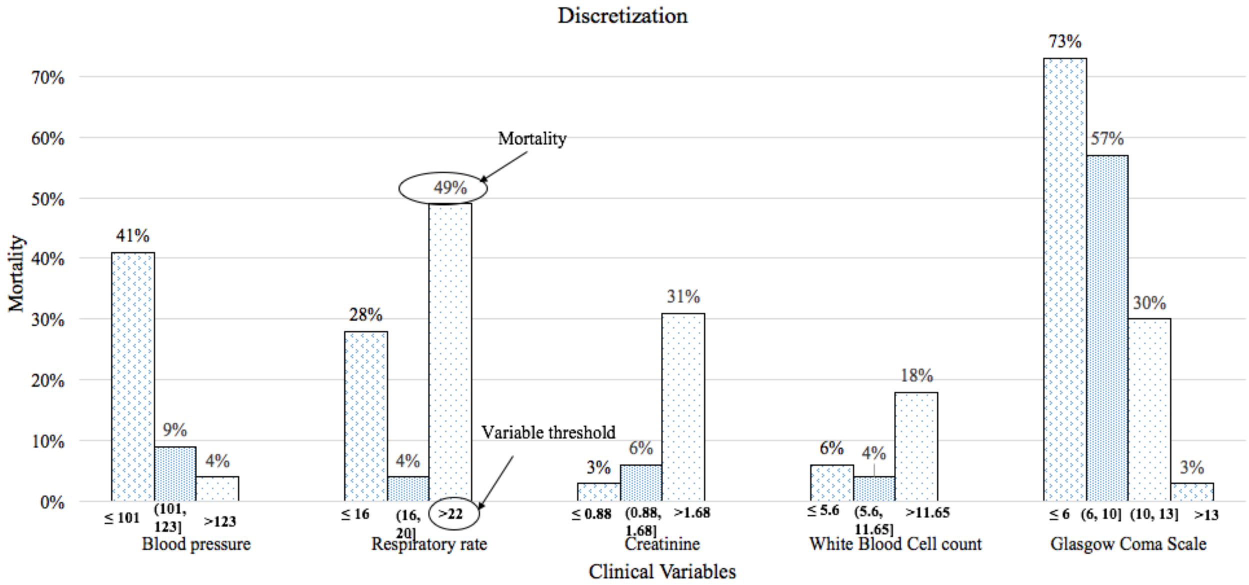


Figure 3.9: Discrete levels of selected biomarkers and corresponding mortality rate

1.5 Model Development

Bayesian network is a probabilistic model with directed acyclic graph. Combination of nodes and arcs is used for the graphical representation. Each node represents a random variable, and each arc implies conditional dependencies between variables. Bayesian networks are popularly used in medical research because they capture the uncertainties and non-linearities presented in data using graphical representation that is easy to understand [11].

There are two popular Bayesian classifiers: Naive Bayesian and TAN Bayesian. The Naive Bayes classifiers are simple and assume conditional independence between variables given the outcome variable. TAN improves Naive Bayes classifier to capture important interactions among variables and at the same time maintains the mathematical simplicity. In TAN, the outcome variable (mortality risk) has no parent, and each predictor can have at most two parents: class variable and any other biomarker. The procedure to learn the structure of the proposed model is explained as follows [91].

- Step 1: Calculate the mutual information (MI) between each pair of biomarkers using Equation III.2

  $$MI(X_i, X_j|Y) = \sum_{k=1}^{C_{X_i}} \sum_{l=1}^{C_{X_j}} \sum_{m=1}^{C_Y} P(k,l,m) \log \frac{P(k,l|m)}{P(k|m)P(l|m)} \tag{III.2}$$

  where C_{X_i}, C_{X_j} and C_Y are the number of levels in variable X_i, X_j and Y, respectively. i, $j \in \{$Glasgow Coma Scale (GCS), Systolic Blood Pressure (SBP), Respiratory Rate (RR), White Blood Cell (WBC), Creatinine$\}$

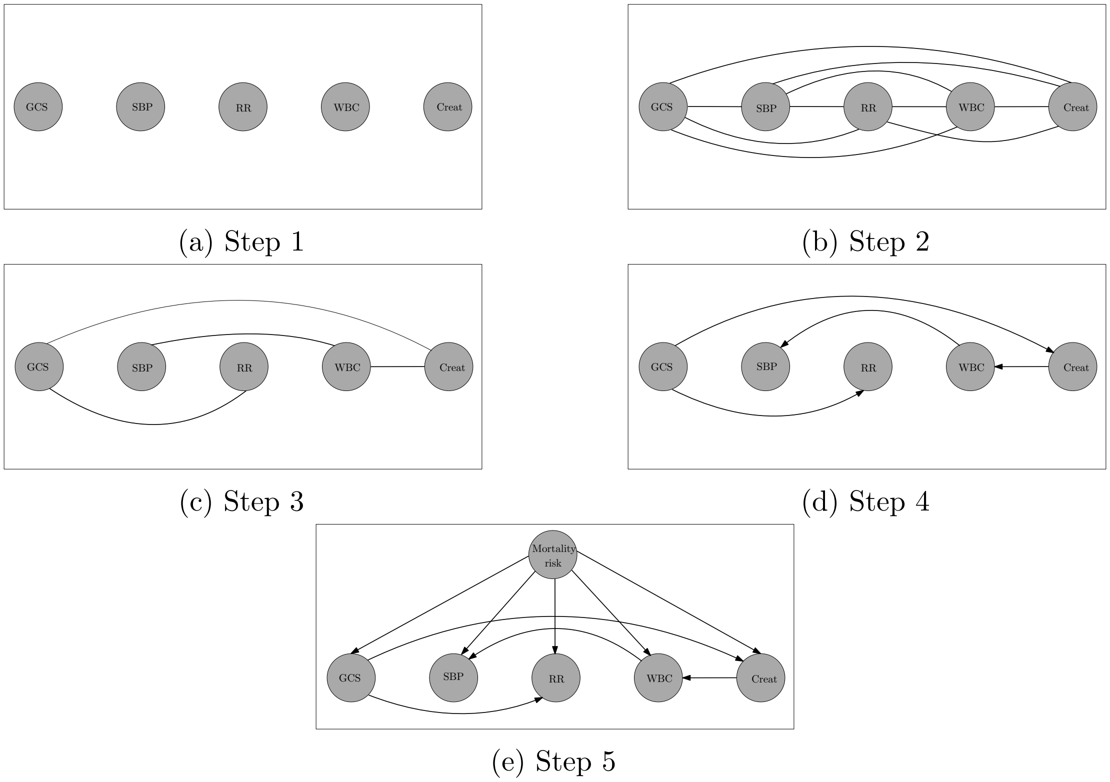
- Step 2: Build a complete undirected graph with the mutual information as the weight of edges

- Step 3: Find the maximum weighted spanning tree

- Step 4: Transform the undirected tree obtained from Step 3 to a directed one by choosing a root variable either arbitrarily or by expert's opinion, and setting the direction of all edges to be outward from it. In this study, the selection of root node was performed using an expert's opinion such that the selection does not violate any domain knowledge. The establishment of direction does not change the data likelihood because MI(X_i, X_j) = MI(X_j, X_i)

- Step 5: Include new node (mortality risk) and connect it with the rest of the graph obtained in Step 4

The steps to construct TAN Bayesian network for the current sepsis model are shown in Figure 3.10.



(a) Step 1

(b) Step 2

(c) Step 3

(d) Step 4

(e) Step 5

Figure 3.10: Structure learning of the TAN Bayesian network model (GCS: Glasgow Coma Scale; SBP: systolic blood pressure; RR: respiratory rate; WBC: white blood cell count; Creat: creatinine)

After learning the TAN structure, the conditional probabilities are calculated by using Equation (III.3).

$$P(X_i|X_{pa(i)}) = \frac{P(X_i, X_{pa(i)})}{P(X_{pa(i)})} \qquad \forall i \in \{GCS, SBP, RR, WBC, Creat\}. \tag{III.3}$$

$X_{pa(i)}$ are the parents of X_i. For example, in Figure 3.10, the parents of *respiratory rate* are *Glasgow Coma Scale* and *mortality risk*. Therefore,

$$P(X_{RR}|X_{GCS}, Y) = \frac{P(X_{RR}, X_{GCS}, Y)}{P(X_{GCS}, Y)}$$

The estimated conditional probabilities were used for the inference. For our model, the inference can be derived by using Bayes' theorem as follows.

$$\begin{aligned} &P(Y|X_{GCS}, X_{SBP}, X_{RR}, X_{WBC}, X_{Creat}) \\ &= \frac{1}{Z} P(X_{GCS}|Y)P(X_{SBP}|Y, X_{WBC})P(X_{RR}|Y, X_{GCS})P(X_{WBC}|Y, X_{Creat})P(X_{Creat}|Y, X_{GCS})P(Y) \end{aligned} \tag{III.4}$$

Z is a scaling factor and equal to $P(X_{GCS}, X_{SBP}, X_{RR}, X_{WBC}, X_{Creat})$.

The inclusion of interactions among biomarkers differentiates our model from others. Most relationships discovered by TAN Bayesian network were supported by practitioner knowledge except the relationship between WBC and blood pressure. Based on expert's opinions, the Glasgow Coma Scale is most likely to relate with respiratory rate and creatinine, and creatinine possibly affects WBC; but the relationship between WBC and blood pressure is not so apparent. Our results show eliminating the arc between WBC and blood pressure will marginally lower the overall performance of the model. Furthermore, we inspected the correlation coefficient between WBC and blood pressure and observed statistically significant negative correlation (ρ = -0.12). The negative correlation coefficient implies that as WBC increases, blood pressure decreases and vice-versa. The magnitude of correlation coefficient represents the strength of association. Our findings suggest there might exist a relationship between WBC and blood pressure, and it warrants further investigation.

2 Results and Discussion

In this section, we compare the performance of the proposed model with SIRS [25], qSOFA [12], MEWS [92] and SOFA [9]. It is worth noting that MEWS and SOFA are not designed to predict mortality. MEWS is used to detect criticality of ill patients, and SOFA is used to understand the health of body organs. However, these criteria are commonly used in medical settings and provide a good benchmark to evaluate the performance of a model [93].

The metrics to evaluate the performance of a classifier are AUROC, sensitivity, specificity, and geometric mean (G-Mean). The AUROC is most commonly used in medical literature to evaluate the diagnostic capability of a classifier as its discriminative threshold is varied. The AUROC of 1 corresponds to an ideal model while 0.5 corresponds to the worst. Sensitivity is the true positive rate, and specificity is the true negative rate. G-Mean is the geometric mean of sensitivity and specificity (Equation III.5). G-mean is often used to inspect the balance between sensitivity and specificity [94].

$$G - Mean = \sqrt{sensitivity \times specificity} \tag{III.5}$$

Our model estimates the risk of mortality. The estimated probabilities were translated into categorical outcome by using a cut-off point. This cut-off point governs the trade-off between sensitivity and specificity, therefore, the selection of cut-off point was given special attention. Instead of using default decision threshold (0.5), we explored previous studies to understand the best approaches to find the optimal cut-off point without considering the misclassification cost and prevalence. There exist two popular methods to find the optimal cut-off point: 1) point in receiver operating characteristic curve closer to (0, 1); first index represents false positive rate and second represents true positive rate, and 2) Youden index [95, 96] Youden index determines the cut-off at which (sensitivity + specificity -1) is maximum. Perkins and Schisterman (2006) compared these two methods and identified advantages and disadvantages of selecting one over the other. The ideal model passes through the point (0, 1) [95]. Therefore, any point in receiver operating characteristic curve closer to (0, 1) is visually more meaningful [95]. Hence, cut-off point corresponding to the point closer to (0, 1) was considered as an optimal cut-off point and was used to estimate sensitivity and specificity.

The model validation was performed using 5-fold cross-validation. Figure 3.11 illustrates the validation procedure. We divided the data into five equal folds. One fold was used for the validation and the remaining were used to train the model. This procedure was carried out iteratively to consider each fold for validation. The final performance measures (AUROC, sensitivity, specificity and G-mean) were estimated by taking the average of performance measures obtained from five mutually exclusive data folds.

We compared the AUROC of the proposed model against the alternate models (SIRS, qSOFA, MEWS and SOFA) over different time instances prior to discharge. Figure 3.12 illustrates the AUROC of our model and competing models preceding the discharge time. The result indicates that our model performed better than competing models over all time instances before the discharge. The percentage differences in AUROC between our model and the alternate models (SIRS, qSOFA, MEWS, and SOFA) ranged from 30-50%, 22-32%, 8-11%, 3-5%, respectively. This longitudinal analysis presents results in favor of the robustness of our model.

We also compared the time-average performance of our model to alternate models (SIRS, qSOFA, MEWS and SOFA). The sensitivity, specificity, G-Mean, and AUROC of our

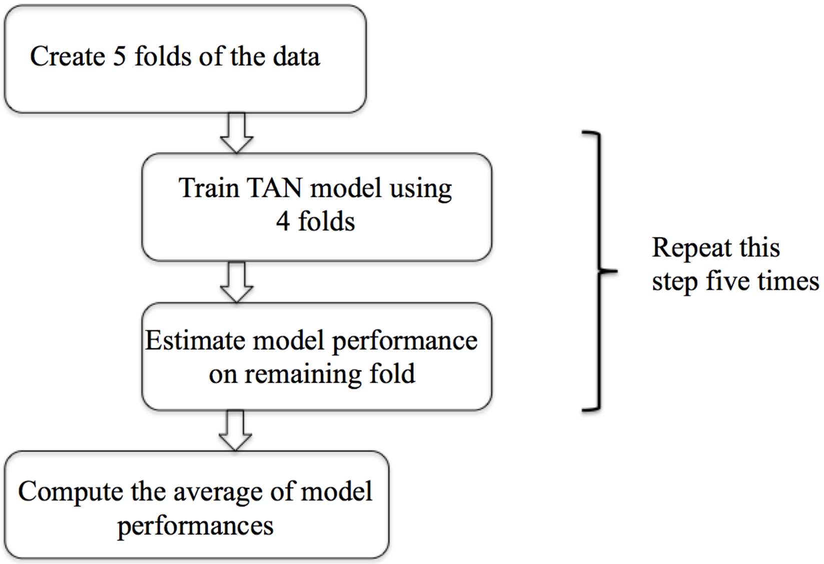

Create 5 folds of the data
Train TAN model using 4 folds
Estimate model performance on remaining fold
Compute the average of model performances
Repeat this step five times

Figure 3.11: Model validation approach

model and competing models are shown in Figure 3.13. With AUROC of 0.84, our model outperformed other criteria (SIRS = 0.59, qSOFA = 0.65, MEWS = 0.75, SOFA = 0.80). In addition, our model provides the best balance between sensitivity and specificity, and this can be seen evidently by using G-Mean. The G-Mean of our model (0.75) is greater than SIRS (0.55), qSOFA (0.58), MEWS (0.70), and SOFA (0.73).

Many clinical applications of a model necessitate high sensitivity. Therefore, for the same AUROC (0.84), Table 3.4 reports specificity on varying values of sensitivity. The results show that at very high sensitivity (0.99), the model loses the specificity significantly. Depending on the requirement on sensitivity, we can increase the specificity.

Table 3.4: Specificity on varying values of sensitivity

Sensitivity	0.99	0.95	0.90	0.85	0.80	0.85
Specificity	0.10	0.35	0.54	0.68	0.72	0.76

The primary reason for the better performance using TAN Bayesian network is the presence of correlations among biomarkers. TAN considers correlations, while other modeling techniques treat all variables as independent. TAN Bayesian network usually performs better

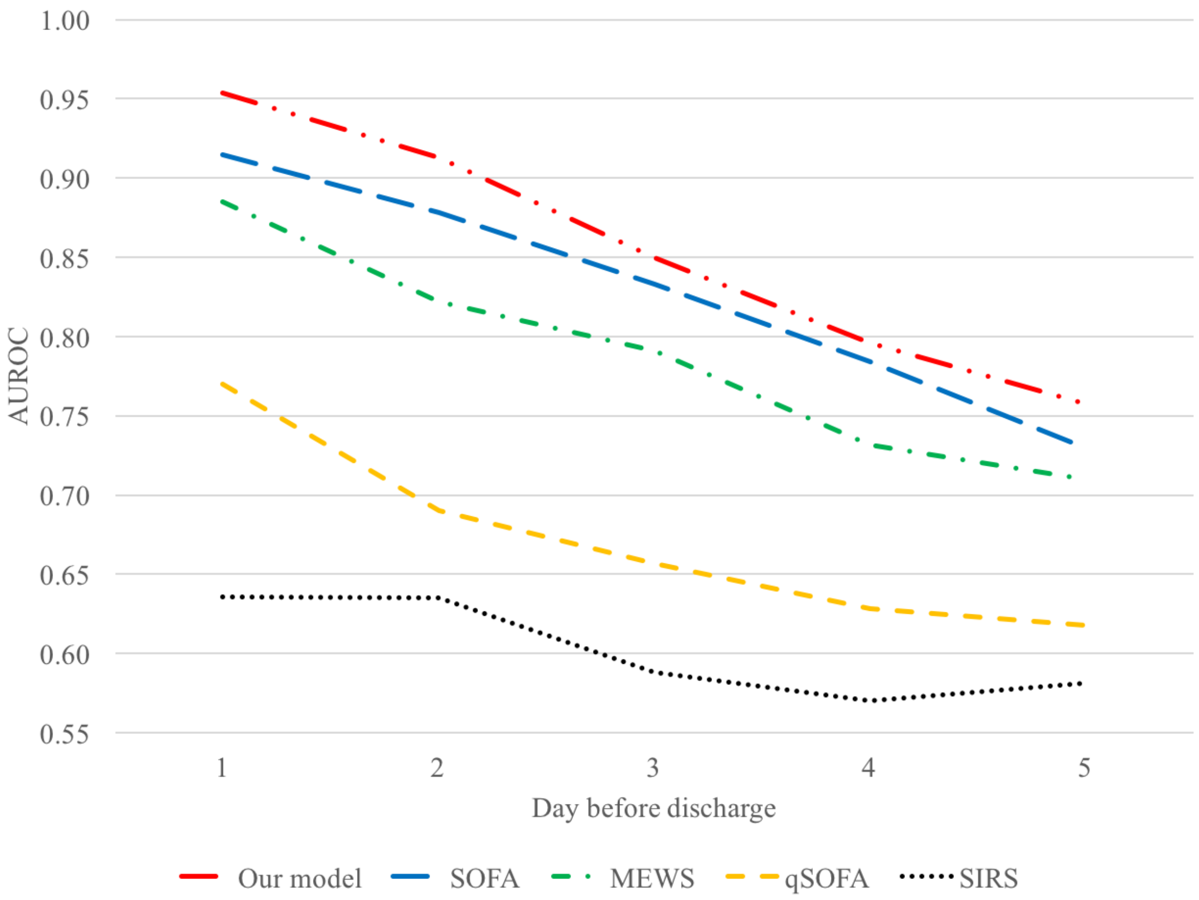


Figure 3.12: AUROC over the different time before the discharge

when variables are correlated [97]. The physiological variables more often than not show correlation among biomarkers. For example, in our EHR data, the correlation between WBC and blood pressure is -0.12. Therefore, the performance gain of our model could be attributed to the existence of correlation among biomarkers.

Table 4.1 summarizes selected biomarkers and the performance characteristics of each model. Although our model uses more variables than SIRS and qSOFA, it performs significantly better than these models. SOFA score requires the knowledge of four laboratory results. Our model with only two laboratory results performs marginally better than SOFA. The use of fewer laboratory results enables practitioners to predict the risk of sepsis easily and quickly.

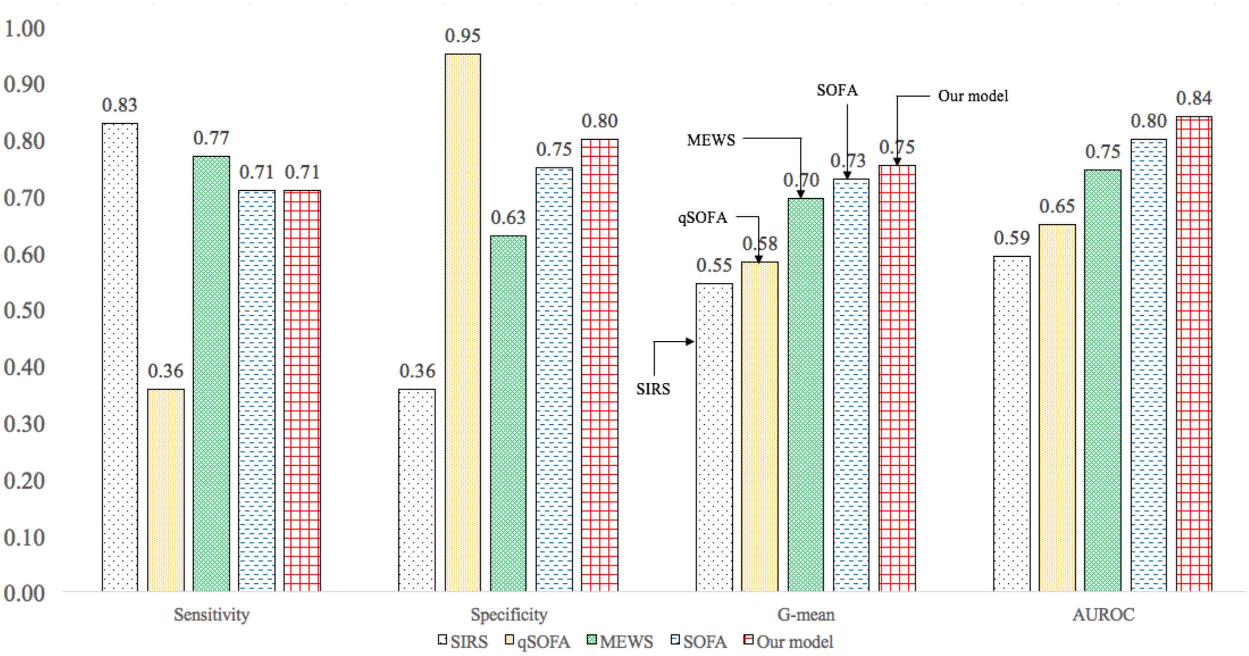


Figure 3.13: Comparison of performance of different diagnostic criteria

Table 3.5: Biomarkers and the performance characteristics of the proposed model

		SIRS	qSOFA	MEWS	SOFA	Our model
Biomarkers	Systolic blood pressure		✓	✓	✓	✓
	Glasgow Coma Scale		✓	✓	✓	✓
	Respiratory rate	✓	✓	✓		✓
	White blood cell	✓				✓
	Creatinine				✓	✓
	PaO_2/FiO_2				✓	
	Platelet count				✓	
	Mean arterial pressure				✓	
	Bilirubin				✓	
	Blood pressure support				✓	
	Body temperature	✓		✓		
	Hourly urine			✓		
	Heart rate	✓		✓		
	No. of biomarkers	4	3	6	8	5
Performance	AUROC	0.59	0.65	0.75	0.80	0.84
	G-Mean	0.55	0.58	0.70	0.73	0.75
	Sensitivity	0.83	0.36	0.77	0.71	0.71
	Specificity	0.36	0.95	0.63	0.75	0.80

3 Conclusions

In this work, we developed a predictive model using Bayesian networks that inherently captures the important interactions among biomarkers and compared the performance of our model with SIRS, qSOFA, MEWS and SOFA. The proposed model was trained with the population selected using the new sepsis definitions. The model development process includes the following main steps: data extraction, data preprocessing, learning of Bayesian network, and model evaluation. The variable selection identifies biomarkers meaningful to predict sepsis. Out of a set of 13 variables, 5 variables were selected (systolic blood pressure, Glasgow Coma Scale, respiratory rate, white blood cell count, and creatinine) to develop the model.

The structure of the Bayesian network facilitates visual interpretation of the interactions among biomarkers. The model identified relations that were in align with practitioners' knowledge. The comparative analysis showed that our model outperformed alternate models. The results underline the robustness of our model by comparing the AUROC of various models over different time instances prior to discharge. The proposed model eliminates the existing problem of unbalanced sensitivity (SIRS = 0.83, qSOFA = 0.36) and specificity (SIRS = 0.36, qSOFA = 0.95) and delivers balanced sensitivity (0.71) and specificity (0.80). The key characteristic of our model is that it outperformed MEWS and SOFA and uses fewer variables than them. The use of fewer lab results enables quick prognosis of sepsis. In comparison with SIRS, qSOFA, MEWS and SOFA, our model provides the best alternative and can easily be integrated in EHR environment and autonomously identify the patient at high risk of sepsis.

The limitation of this study is that the data come only from hospitals using Cerners EHR system. So, there could be potential sources of bias in the data. Although we included data from hospitals located in all four U.S. census regions, we cannot generalize the results to all U.S. hospitals as there could be distinct differences between hospitals using the Cerner system and other hospitals. The possible reason of difference is that the Cerner's EHR is mainly used in big hospitals because it is expensive to implement. We did not consider the effect of intervention by assuming that each patient was subjected to a similar intervention.

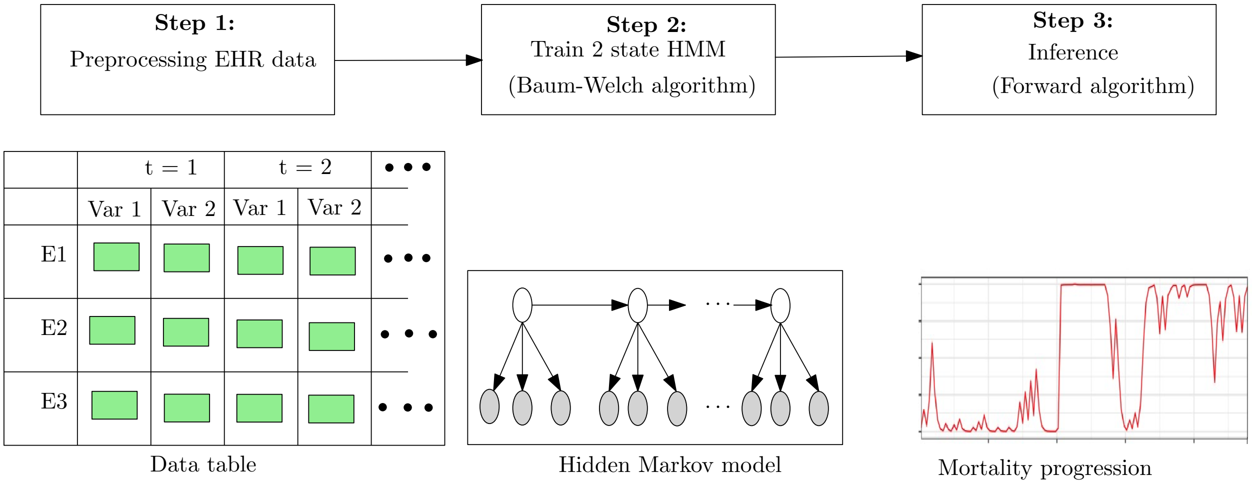
In this chapter, we developed a predictive model that uses the observations obtained at a particular time interval. However, when a patient visits hospitals, the physicians do not only examine the current symptoms but also browse through the previous records. In the next chapter, we propose a framework to capture such phenomena where a model uses both current and past information to assess the risk of the disease.

Chapter IV

MORTALITY PROGRESSION MODEL

In Chapter III, we applied non-temporal methods to develop a clinical decision support system for sepsis diagnosis. In Chapter IV and V, we introduce a temporal method (HMM) to elaborate on utilizing the longitudinal EHR data. This chapter demonstrates the application of HMM to model the mortality progression, while the next chapter unearths the limitation of the current HMM learning algorithm and proposes a method to efficiently compute the HMM parameters.

The proposed framework to develop a mortality monitoring system using an HMM is shown in Figure 4.1. A brief explanation of HMM and their training is provided in Sections 1.2 and 1.3, respectively. The proposed framework computes mortality risk in combination with present observations and past trends of clinical signs. This modeling approach is closely aligned with the real-world phenomenon where physicians use historical data in addition to current vital signs to assess the health of patients. The proposed framework includes three steps: data preprocessing, training HMM parameters and inference. EHR data are stored in long form where each clinical measurement of a patient's hospital stay is stored in separate rows. The proposed framework requires data to be in wide form where longitudinal measurements are stored in one row. The data preprocessing steps are explained in detail in Section 1.1 using sepsis case data. The preprocessed data is used to learn the 2-state HMM parameters using the *Baum-Welch algorithm*. The two states represent stable and critical patient status. The learned model combined with newly observed longitudinal clinical measurements is used for generating the inference for mortality progression using the *forward algorithm* (explained in Section 1.4).



	t = 1		t = 2		• • •
	Var 1	Var 2	Var 1	Var 2	
E1	🟩	🟩	🟩	🟩	• • •
E2	🟩	🟩	🟩	🟩	• • •
E3	🟩	🟩	🟩	🟩	• • •

Figure 4.1: Proposed hidden Markov model framework. E1 stands for Encounter 1. Similarly E2, E3 etc.

1 Method

1.1 EHR Data Extraction and Preprocessing

This section explains the data extraction and preprocessing steps for EHR data. The procedure is generic and could be applied to any EHR dataset. The study data was collected from Cerner Corporation's Health Facts warehouse, one of the largest de-identified and HIPAA compliant health care data repositories in the United States. The data included longitudinal date- and time-stamped information on admission and discharge, laboratory results, diagnosis code, patient demographics, and additional clinical and billing information. We demonstrate the performance of the proposed framework on sepsis data. As mentioned in

Chapter I, progresses rate of sepsis is high, and the likelihood of survival reduces with delay in treatment. Therefore, developing a mortality progression model to monitor the criticality of sepsis is meaningful.

The selection of population was derived following the Sepsis-3 study [84]. The emphasis of the study is not to develop a new predictive model but to underline the benefits of applying the proposed temporal framework over traditional non-temporal methods. Therefore, we borrowed the set of three clinically useful variables (blood pressure, Glasgow Coma Scale score and respiratory rate) identified by the Sepsis-3 study [84] to detect sepsis.

The data extraction steps are illustrated in Figure 4.2 (left). Although HMM parameters can be trained for varying length of time series data, we keep our analysis to sequences of the same time length to eliminate the additional computational complexity for training the parameters. In this study, we considered encounter visits that resulted in discharge on the eleventh day. For such encounters we only used the data till tenth day for training the model.

Figure 4.2 (right) shows the preprocessing steps of data preparation. The Sepsis-3 study had shown, with comprehensive experiments, that blood pressure, respiratory rate and Glasgow Coma Scale score are the most critical identifiers of sepsis mortality [84]. Therefore, we limited our study to only these routinely available vital signs because such clinical variables are frequently recorded in EHR and present fewer missing longitudinal data. The EHR data may suffer many challenges. For example, clinical variables are stored with multiple names and varying units, and some measurements of clinical signs are meaningless. To ameliorate the inconsistencies, we preprocessed the data. The pruning of a variable includes considering outliers or meaningless measurements as unrecorded. For example, a respiratory rate of 250 is infeasible, therefore, we considered it as unknown measurement.

After pruning variables, time abstraction was performed. Time abstraction is the process of aggregating the clinical measurements over time into time windows. Figure 4.3 graphically represents the time abstraction. In EHR data, a few clinical signs were recorded

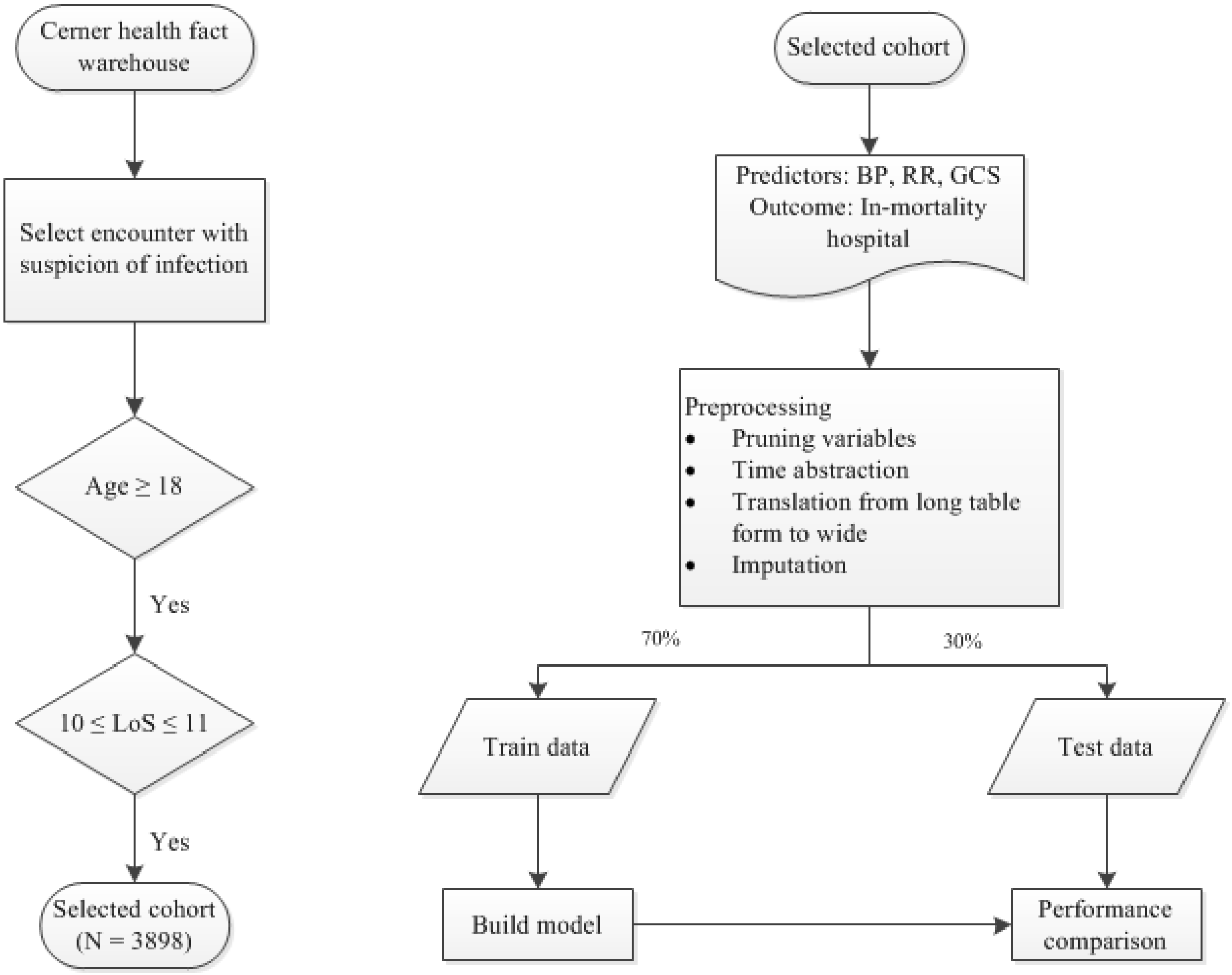


Figure 4.2: Data extraction (left) and data preprocessing (right). BP:blood pressure, RR: respiratory rate, GCS: Glasgow Coma Scale score

in intervals of fifteen minutes, and some were a couple of hours apart. We aggregated the data into two hour blocks to maintain the original variations in clinical signs and at the same time to reduce the volume of missing records. As shown in Figure 4.3, if more than one result is present in a two hour interval, the mean of all results was used for analysis.

Following time abstraction, imputation was carried out. Imputation is a process of finding the best fit measurement for missing values. The *left-center-right* technique was employed that was discussed in Section 1.2 (Chapter III). The preprocessed data was split into training (70%) and testing (30%) for model building and testing the performance, respectively.

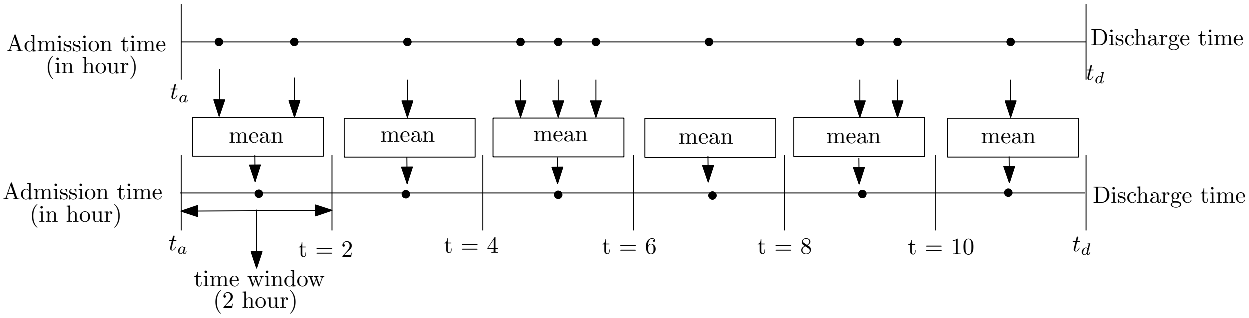


Figure 4.3: Time abstraction (t_a: admission time, t_d: discharge time)

The statistical summary of selected three variables after preprocessing is shown in Table 4.1.

Table 4.1: Statistical summary of three selected clinical variables

Clinical variable	Minimum	Maximum	Mean	Median
Blood pressure (mmHg)	36	243	123.2	122
Glasgow Coma Scale score	2	15	14.26	15
Respiratory rate (breaths/min)	8	40	18.85	18

1.2 Hidden Markov Model

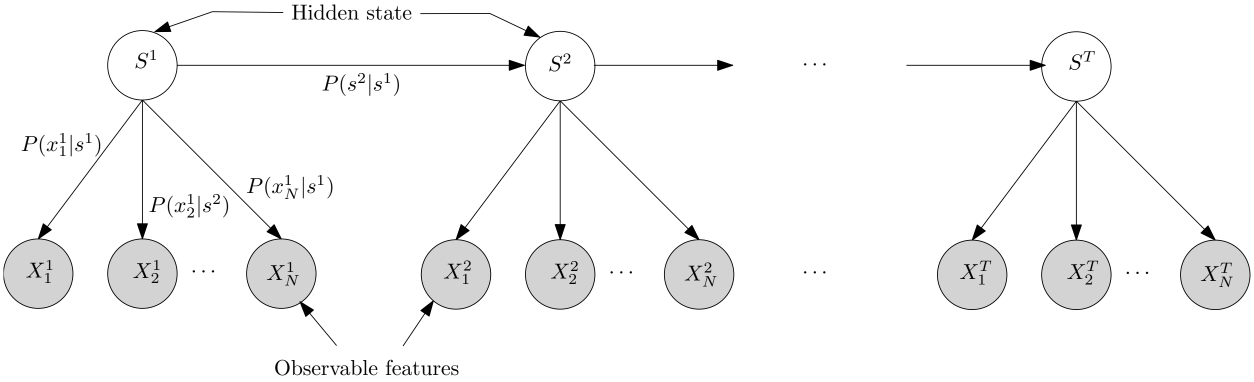


Figure 4.4: Hidden Markov model

HMMs are popularly known for successfully modeling time series problems of speech recognition. HMMs consist of observed states and hidden (unknown) states. Unlike Markov models, in HMMs, the state of interest is unobservable. Therefore, we refer such modeling methods hidden. The aim of HMMs is to infer about the hidden state of the system using the observable time series data. The HMM can be explained easily with an example. Consider I record how many ice cream I eat every day. Given my consumption of ice cream reflects the the state of my mind (happy or sad), Pistol Pete can infer about my mood (hidden state) from the number of ice cream I have.

The basic framework of an HMM is shown in Figure 4.4. The shaded circles represent the observable variables and un-shaded circles represent hidden variables. The arcs represent conditional probabilities. The conditional probabilities are divided into two classes:

1. *Transition probabilities*

   The transition probabilities reflect the probability of transition from one state to another state in one time step:

   $$P(s^t|s^{t-1})$$

2. *Emission probabilities*

   The emission probabilities represent the probability of emitting an observation given the state.

   $$P(x_i^t|s^t)$$

   We assume that the parameters are time-invariant. Therefore, $P(x_i^1|s^1) = P(x_i^2|s^2) = P(x_i^t|s^t) = P(x_i|s)$ and $P(s^2|s^1) = P(s^3|s^2) = P(s^T|s^{T-1})$

$x_i^1, x_i^2, \ldots, x_i^T$ represent the sequence of observed measurements for X_i ($i \in \{$1, 2, ..., N$\}$), where T is the number of time steps and N is the number of observed variables. We assume that given the state at time t, the observed variables are independent from each other. This assumption is in-align with the basic principal of non-temporal methods such as logistic regression and naive Bayes. Let $X_1^t, X_2^t, \ldots, X_N^t$ be random variables representing each clinical variable at time t. For simplicity let X^t be $[X_1^t, X_2^t, \ldots, X_N^t]$

$$P(X^t|s^t) = \prod_{i=1}^{N} P(x_i^t|s^t) \tag{IV.1}$$

As mentioned earlier, each sequence could be of a different length but to maintain the simplicity, we considered sequences of the same length. s^t denotes the state of the disease (1 = non-critical or 2 = critical) at time t ($t \in \{1, 2, \dots, \mathrm{T}\}$).

1.3 Learning HMM Parameters

The HMM parameters (*emission* and *transition* probabilities) are estimated using the maximum likelihood method. The objective is to maximize the likelihood of sequences of observations given parameter as shown in Equation (IV.2)

$$logL = \sum_{i=1}^{M} logP(\vec{\mathbf{Z}}_i|\lambda) \tag{IV.2}$$

where Z_i is i^{th} sequence of observations, $\vec{\mathbf{Z_i}} = \{X^1, X^2, \dots, X^T\}_i$. λ is the collection of HMM parameters both transition and emission probabilities. Each sequence of observation is obtained from an independent hospital stay. M is the number of independent observed sequences corresponding to the number of patient visits (or widely known as *encounters*) in the dataset. We used *Baum-Welch algorithm* to estimate the HMM parameters such that the log-likelihood of observed data could be maximized [10]. A mathematical description of how to maximize log-likelihood is included in Appendix A.

1.4 HMM Inference

HMM inference includes computation of the most probable sequence of states. The most probable sequence of state is derived by estimating the probability of being in state s

at time t, given the observed data until t and HMM parameters $(P^t(s|\mathbf{X^1}, \mathbf{X^2}, \ldots, \mathbf{X^t}, \lambda))$. The inference is computed using a recursive procedure partially adapted from Rabiner [10]. Following naive Bayes principle, the recursive procedure is modified to accommodate multiple features instead of considering only one feature at a time. We used multiplication of conditional probabilities $(\prod_{i=1}^{N} P(x_i^t|s^t))$ to account for the effect of N variables in state probabilities.

Let $\alpha^t(s)$ be the forward variable and is defined as joint probability of partial observed sequence until time t $(\mathbf{X^1}, \mathbf{X^2}, \ldots, \mathbf{X^t})$ and system being in state s, given HMM parameters.

$$\alpha^t(s) = P^t(\mathbf{X^1}, \mathbf{X^2}, \ldots, \mathbf{X^t}, S^t = s|\lambda)$$

The recursive procedure to compute $\alpha^t(s)$ is explained as follows:

1. *Initialization:* Initialize the joint probability of initial observation $\mathbf{X^1}$ and state s

$$\alpha^1(s) = \alpha^0(s) \prod_{i=1}^{N} P(x_i^1|S^1 = s, \lambda) \quad for \quad s \in \{1, 2\}$$

   where $\alpha^0(s)$ is the initial probability of system being in state s when no observation was observed.

2. *Induction:* for t = 2 to T-1

$$\alpha^t(s) = \big[\sum_{i \in \{1,2\}} \alpha^{t-1}(i) P(S^t = s|S^{t-1} = i) \big] \prod_{i=1}^{N} P(X_i^t|S^t = s) \quad for \quad s \in \{1, 2\}$$

   This step shows how a patient's health transitions to one of two states at time t from previous state at time $t-1$. $\alpha^{t-1}(i)P(S^t = s|S^{t-1} = i)$ is joint probability that $\mathbf{X^1}, \mathbf{X^2}, \ldots, \mathbf{X^{t-1}}$ are observed and the state s is reached via state i. Summing this product over all possible states results in all paths reaching s at time t. Since, now we know the state distribution at time t, it is easy to compute $\alpha^t(s)$ by incorporating the observed data.

After computing $\alpha^t(s)$, we normalize the probability to compute the conditional probability distribution:

$$P^t(s|\mathbf{X^1}, \mathbf{X^2}, \ldots, \mathbf{X^t}, \lambda) = \frac{\alpha^t(s)}{\alpha^t(1) + \alpha^t(2)} \quad for \quad s \in \{1, 2\}$$

2 Results and Discussion

2.1 Performance Comparison between Proposed Temporal Framework and Non-temporal Modeling Techniques

The AUROC, explained in Chapter III, was used to compare the performance of temporal and non-temporal methods to model mortality progression using EHR data. The AUROC is a popular performance measure to evaluate the discrimination power of classifiers. The classification accuracy could be an alternate performance measure. However, due to high proportion of negative cases in our data (negative cases; 93%, positive cases: 7%), the choice of classification accuracy is not optimal. Non-temporal models include decision trees, logistic regression, naive Bayes, random forests and support vector machines. In typical scenarios with non-temporal methods, we compute one class probability for an individual patient's visit using aggregated measurements and compare it against the actual outcome to evaluate the AUROC. The HMM computes class probability at each time interval using current and past measurements. Therefore, for comparison, we computed the class probabilities for non-temporal models at each time interval using the measurement recorded in that time interval only. The parameters of non-temporal models were estimated using the data observed in that time series window. Figure 4.5 illustrates the procedure to compute the AUROC at each time interval. Each model predicts the severity of disease for each time interval, and the predicted risk for each time interval is compared against actual discharge type to compute the AUROC.

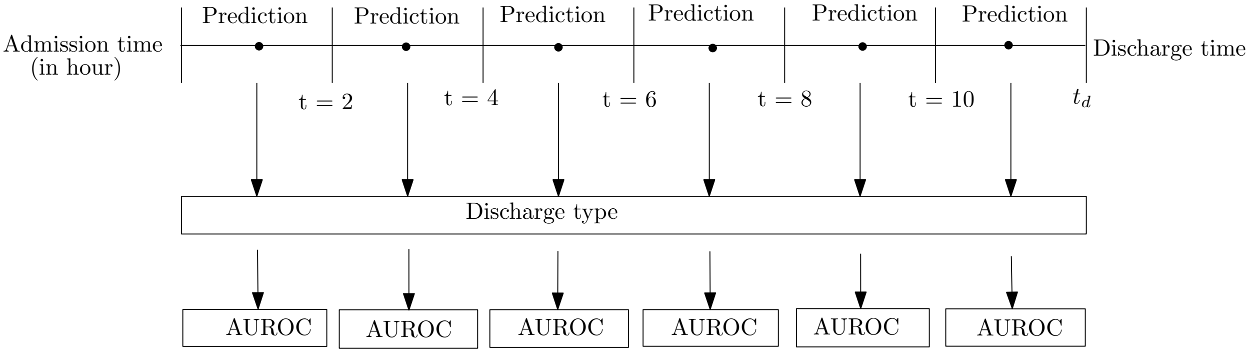


Figure 4.5: Procedure to compute AUROC at each time step

The data used to perform numerical experiments are explained in Section 1.1. Figure 4.6 shows the AUROC obtained at different time steps prior to discharge time for HMM and non-temporal models on sepsis data. It is clearly evident from Figure 4.6 that the AUROC of HMM model is significantly better than non-temporal models. In addition to high performance, HMM model presents a robust increase in the AUROC. The non-temporal models showed high variation in AUROC over time. The primary reason of such behavior is that the non-temporal models estimate the mortality risk by considering only the present measurements of clinical signs but do not incorporate the previous status of the patients. Therefore, the non-temporal models are sensitive to rapid changes in clinical signs.

The difference of AUROC between temporal and non-temporal methods reduces as prediction time moves closer to discharge time. This characteristic indicates that when compared to non-temporal methods, the use of the proposed framework has better prediction advantage in the early stages than the later stages of patient hospital stay. The early identification of a possible bad outcome is an additional novelty of the proposed temporal framework.

We performed the trend analysis to clearly inspect the AUROC gap between the proposed temporal framework and the non-temporal methods. Figure 4.7 includes five plots

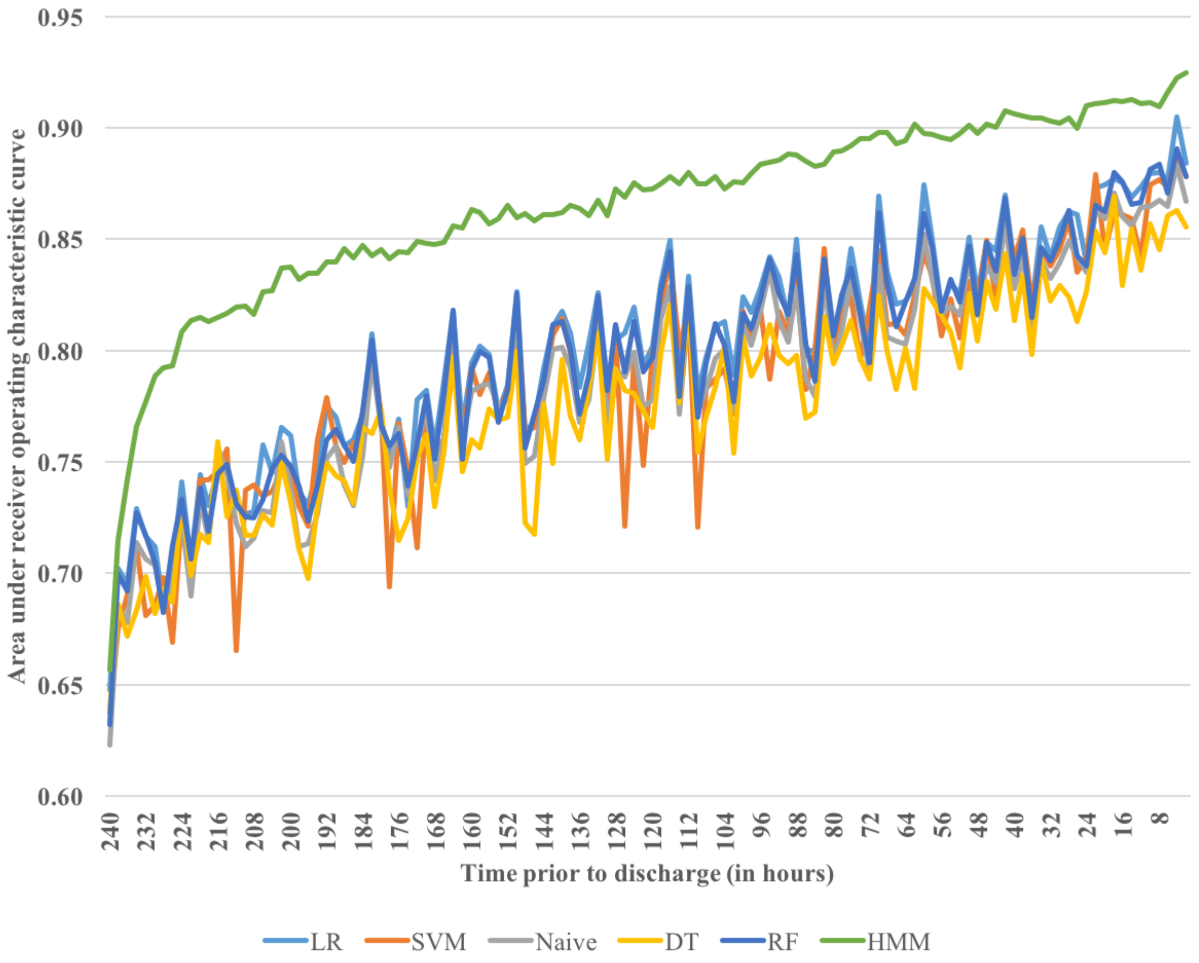
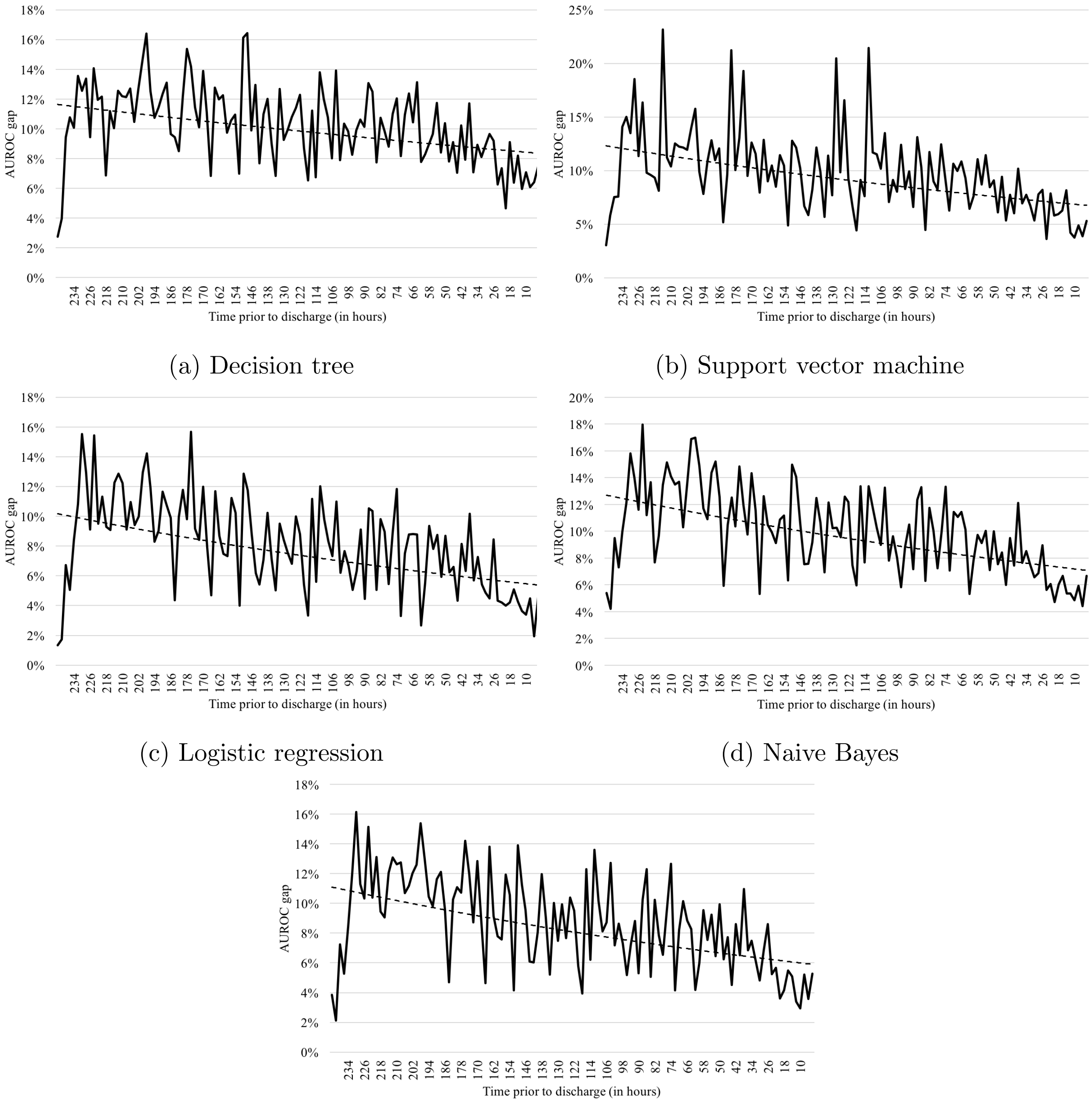


Figure 4.6: Comparison of area under operating characteristic curve prior to discharge time between HMM and non-temporal methods (LR: logistic regression, SVM: support vector machine, DT: decision tree, RF: random forest, HMM: hidden Markov model)

each comparing an individual non-temporal method to the proposed framework. The X-axis represents time prior to discharge and Y-axis shows difference of AUROC between the proposed framework and individual methods. In each plot, at the beginning, the AUROC gap is small because the proposed framework takes time to reach steady state from initial state. It is evident from trend lines that the AUROC gap decreases with prediction time moving closer to discharge time. This characteristic emphasizes that the proposed framework is effective for early prediction, which is an important advantage for healthcare applications.

The early prediction enables clinicians to inject timely intervention and to efficiently manage hospital resources.



(a) Decision tree

(b) Support vector machine

(c) Logistic regression

(d) Naive Bayes

(e) Random forest

Figure 4.7: Trend of AUROC difference between proposed framework and non-temporal methods. The dotted line represents the trend line.

Table 4.2 summarizes the mean of AUROC obtained by both temporal and non-temporal methods. The mean is calculated by taking the average of all AUROC obtained at different time intervals. The mean AUROC of HMM is 9-12% greater than non-temporal methods. The better performance of HMM is attributed to its strength in leveraging longitudinal clinical data.

Table 4.2: Mean of AUROC for both temporal and non-temporal models

Type	Model	Mean AUROC
Non-temporal	Decision tree	0.78
	Naive Bayes	0.79
	SVM	0.79
	Logistic regression	0.80
	Random forest	0.80
Temporal	HMM	0.87

2.2 Mortality Progression for Sepsis

In this section, we aim to elaborate on another characteristic of the proposed framework that is tracking the mortality progression. We developed a real-time sepsis mortality progression model by combining the time series data of three clinical variables mentioned in Table 4.1.

Figure 4.8 shows the trajectory of three variables (blood pressure, respiratory rate and Glasgow Coma Scale score) along with computed inference (procedure explained in Section 1.4) for the disease severity (or mortality) risk. The inference (red) represents the dynamic behavior of the disease criticality. The region within the dashed rectangle includes the detection point of change in criticality of the patient's condition. This change in state of condition can probably be explained using the drop in blood pressure, continuous low respiration and low Glasgow Coma Scale score. We also noticed in inference that there are further changes in the disease states along the time. The probable reason is the active treatment. The inference using HMM provides critical time information about the change in state and duration of a state. This knowledge can enable practitioners to effectively decide treatment strategies.

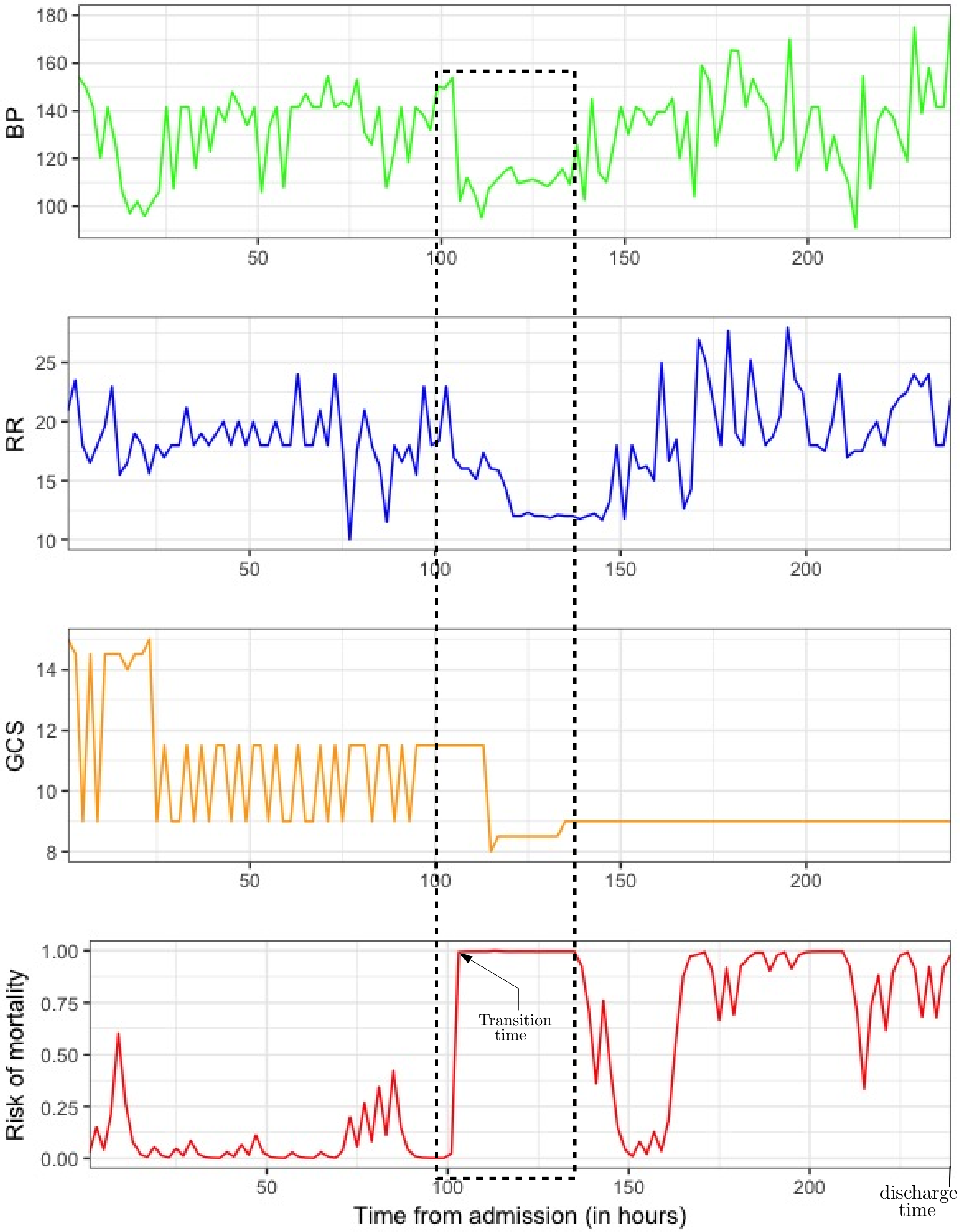


Figure 4.8: Example encounter features and mortality trajectory. BP: blood pressure, RR: respiratory rate, and GCS: Glasgow Coma Scale score

For practical usage of clinical tools, the sensitivity and specificity are important measures. Therefore, we computed these performance measures as well. Sensitivity is defined as correctly classifying positive cases, and specificity is correctly classifying negative cases. We examined the performance of our model five days prior to discharge time because it provides enough time for intervention. Figure 4.9 shows the receiver operating characteristic curve for both temporal and non-temporal models. The sensitivity of our model (0.80) is greater than the sensitivity of all non-temporal models (DT: 0.78, LR: 0.74, Naive: 0.73, RF: 0.74 and SVM: 0.60), which is an important characteristic for diagnosis. But the specificity of our models (0.76) is marginally lower than most of the non-temporal models (DT: 0.78, LR: 0.78, Naive: 0.78, RF: 0.79 and SVM: 0.90). For predictive tools, the sensitivity is an important performance parameter because neglecting the patient who has disease is more detrimental than treating patients that are actually not at high risk.

2.3 Discussion

Sukkar et al. (2012) developed an HMM based disease risk progression model for slow progressing disease [44]. The author employed six state HMM to model Alzheimer's disease. However, the multi-state model possesses its own challenges such as learning complexity. This learning problem becomes significant especially for EHR where high volumes of data are missing. This study describes a step by step process to build a time series model from EHR data. This study directs future clinical researchers to consider time variations in variables instead of merely using aggregated values (mean/median). The proposed approach also provides time related insights that enable practitioners to understand the trajectory of disease.

The mortality prediction model designed using traditional non-temporal approaches (decision tree, logistic regression, naive Bayes, random forest and support vector machine) are very sensitive to short-time intervention effects. The non-temporal models compute mortality

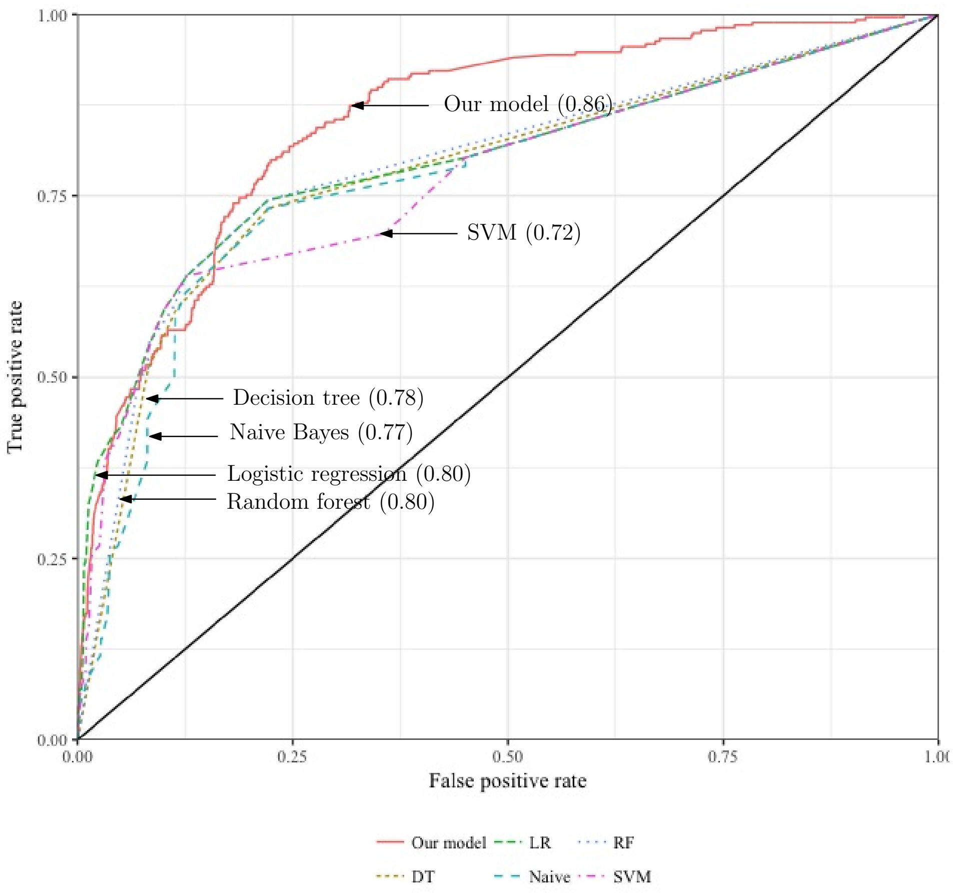


Figure 4.9: Receiver operating characteristic curve. Prediction time is five days prior to discharge. SVM: support vector machine

risk using only the latest measurements. Therefore, temporary change in clinical signs due to intervention can significantly affect the prediction of mortality risk. Such predictions could be misleading and can lead to poor treatment decisions. Instead of relying only on current clinical observations, the more robust mortality risk could be obtained by combining the previous health of patients and chances of transitioning patient's health from one state to others. Consequently, the change in mortality risk does not depend on instant clinical signs, rather, it depends on the continuous trend of clinical signs. Using the proposed temporal framework, the more robust mortality prediction can be achieved by combining the trends of individual clinical signs.

The proposed framework provides continuous update on the health of a patient. The status of the patient's health gets updated periodically with a newly observed set of clinical observations. This real time information facilitates early detection of acute events, discharge planning and managing limited resources of hospitals. Although sepsis case study data in this paper only includes vital signs due to their dense availability in EHRs, the lab results could also be incorporated. The periodic update on the mortality risk requires knowledge of all clinical signs at each time period. In case any clinical variable is not available, the possible alternative is to use the last available measurement to infer about mortality risk.

The sepsis mortality progression model, developed as a part of this study, showed high sensitivity and specificity. The developed model provides an alternative to existing criteria. Most existing sepsis diagnostic criteria suffer deficiencies: SIRS is known to have poor specificity [18, 31], MEWS have limited accuracy [20], and SOFA is highly complex and requires knowledge of four laboratory results that can be difficult to measure in a resource constrained environment [81]. Our model only uses three easy-to-measure non-invasive clinical measurements to infer about mortality progression. This model equips physicians with early diagnosis in order to provide timely treatment.

3 Conclusions

In this paper, we propose a two state hidden Markov model framework for real-time mortality prediction using EHR data. This study bridges a significant research gap in that the majority of studies on disease risk models do not leverage longitudinal data of EHR [23]. By using a two state HMM, we combine time series clinical data from EHR and show that the proposed framework performs better than non-temporal models. The modeling foundation of this work is to capture the trend of clinical variables in such a way that is clinically meaningful as well as easy to replicate for aggressively progressing diseases.

This study has a few limitations. We assumed that HMM parameters are stationary or do not vary over time. This is a legitimate assumption for fast progressing diseases. Another assumption of this study is that hidden states follow Markovian property. The Markovian property implies that given the previous state, the current state is independent of other states. The intuition behind that assumption is that the previous state provides a summary of what had happened one step prior to current time.

Also, due to computational complexity, this study considers only two hidden states. In the next chapter, we propose a procedure to improve the performance of the HMM algorithm to incorporate multiple states.

Chapter V

MULTINOMIAL REGRESSION BASED INITIALIZATION OF BAUM-WELCH ALGORITHM

This chapter explains the proposed initialization method for the HMM training algorithm. Figure 5.1 graphically depicts the standard procedure for estimate the parameters of HMM. The performance of the HMM learning algorithm (*Baum-Welch* [51]) is sensitive to initialization. In this chapter, we propose a guided initialization method to improve the performance of the *Baum-Welch* algorithm.

Initial parameters
Baum-Welch algorithm
Estimated parameters

Figure 5.1: The standard approach of learning HMMs

1 Method

HMMs are used to model system phenomena that evolve through a finite number of states. The structure and the learning of HMMs are described in Chapter IV. As explained, HMMs are characterized using two parameters: *transition probabilities* $P(s^t|s^{t-1})$ and *emission probabilities* $P(x^t|s^t)$. The transition probabilities are the likelihood of moving from one state to another in one time step, whereas the emission probabilities are the chances of emitting an observations given the hidden state. The parameters are assumed to be time-invariant as in Chapter IV.

The parameters of HMMs are estimated using the *Baum-Welch* algorithm. In this algorithm, the initial distributions of transition and emission probabilities are randomly selected; then, the probability distributions are updated iteratively to maximize the *log-likelihood* of the data. This algorithm is sensitive to initial probability distributions. If the initial parameters are far away from true value, then the convergence might be slow, or the solution might get stuck in a local optimum. Therefore, it is critical to intelligently select initial parameters. In this section, we explain a time efficient initialization procedure for *Baum-Welch* algorithm.

1.1 Proposed Method

This initialization procedure relaxes the time dependency of an observed sequence. In other terms, we assume that the data observed for a subject in different time intervals are independent. The time independence is assumed only to compute initial values; later, time dependencies will be captured by using HMM modeling approach.

In the proposed method, we first concentrate on the final time interval. For the final time interval, the input variables and outcome variable are known; therefore, the problem translates into supervised learning and the weights can be estimated using the maximum *log-likelihood* method. For the remaining time intervals, although the true states of the system are not known, the observed variables are recorded. Hence, using the combination of estimated weights and observed variables, we infer probable state of the system. Subsequently, from the estimated sequence of states, we are able to compute approximate transition probabilities. Finally, the sequence of states and the observed data is used to compute emission probabilities. The proposed method is a mix of supervised learning and the prediction. Our approach can be summarized as follows:

1. Retrieving data of the final time interval and estimating weights for each variable (*supervised learning*)

2. Employing estimated weights to compute probable states in the remaining time step (*prediction*)

3. Computing transition and emission probabilities using probable states computed in Step 2

To elaborate, we also explain the proposed method using an example of healthcare area. Figures 5.2 to 5.6 illustrate a step by step procedure to compute initial transition probabilities. Figure 5.2 shows a sample of the data organization. Let us assume for each patient's visit, we monitor three clinical signs at a constant frequency (shown below as one day). Each clinical sign indicates the health of a specific organ (brain, lung and kidney). Each row in the table represents each individual visit by a single patient. For illustrative purpose, we show each visit constitutes the same length of stay, i.e. T. Each visit, however, can have varying length of stay. In EHR, along with recording clinical signs such as respiratory rate and heart rate, the type of discharge (expired, transfer to special nursing facility and so on) is also reported. The discharge type is an indicator of the patient's true state. For example, if the patient is discharged as expired, then the patient's true state is *critical.* On the other hand, if the patient is discharged to home, then the patient's true state is *healthy.* In Figure 5.3, S^T is a random variable designating the true state of the patient at time T. The combination of observed clinical variables and known true states at time T is used to estimate weights associated with each clinical variable by using maximum likelihood (Figure 5.4). Now, for the remaining time intervals of $\{1, 2, \ldots, T-1\}$, the true states are inferred using the estimated weights and the observed data (Figure 5.5). The true states are derived by computing the posterior probabilities of classes of the patient's state of health. The class with the highest class probability is assigned to that time interval. In Figure 5.6, $S^1, S^2, \ldots, S^{T-1}$ are the random variables designating the true states of the patient's health status from time intervals 1 to $T-1$.

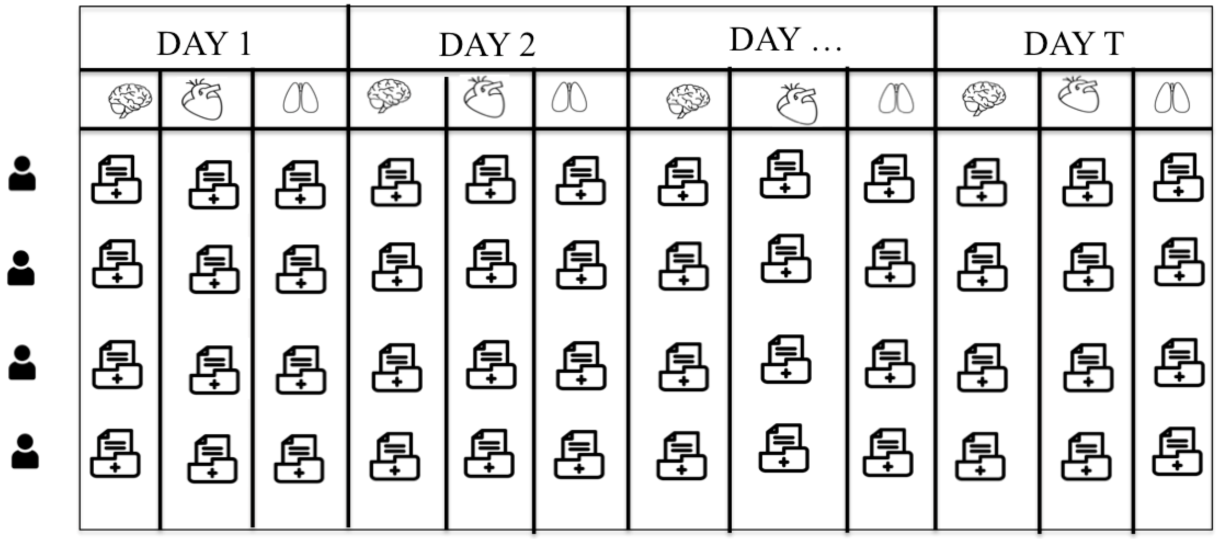


Figure 5.2: Temporal data from EHR

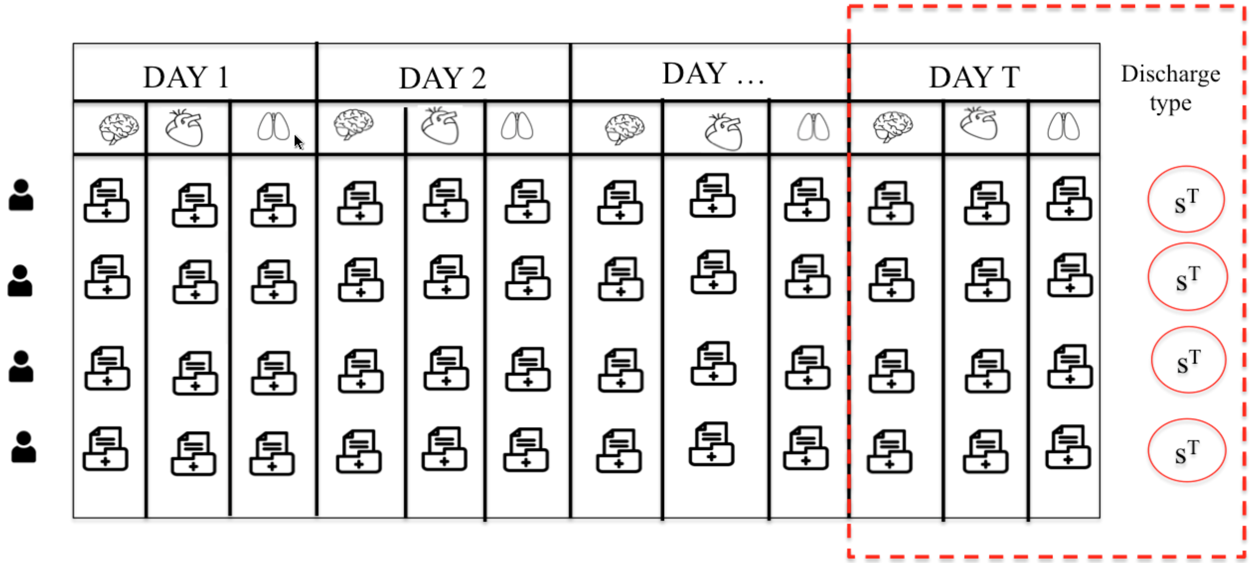


Figure 5.3: At final time interval, the known observation and true states translate into supervised learning

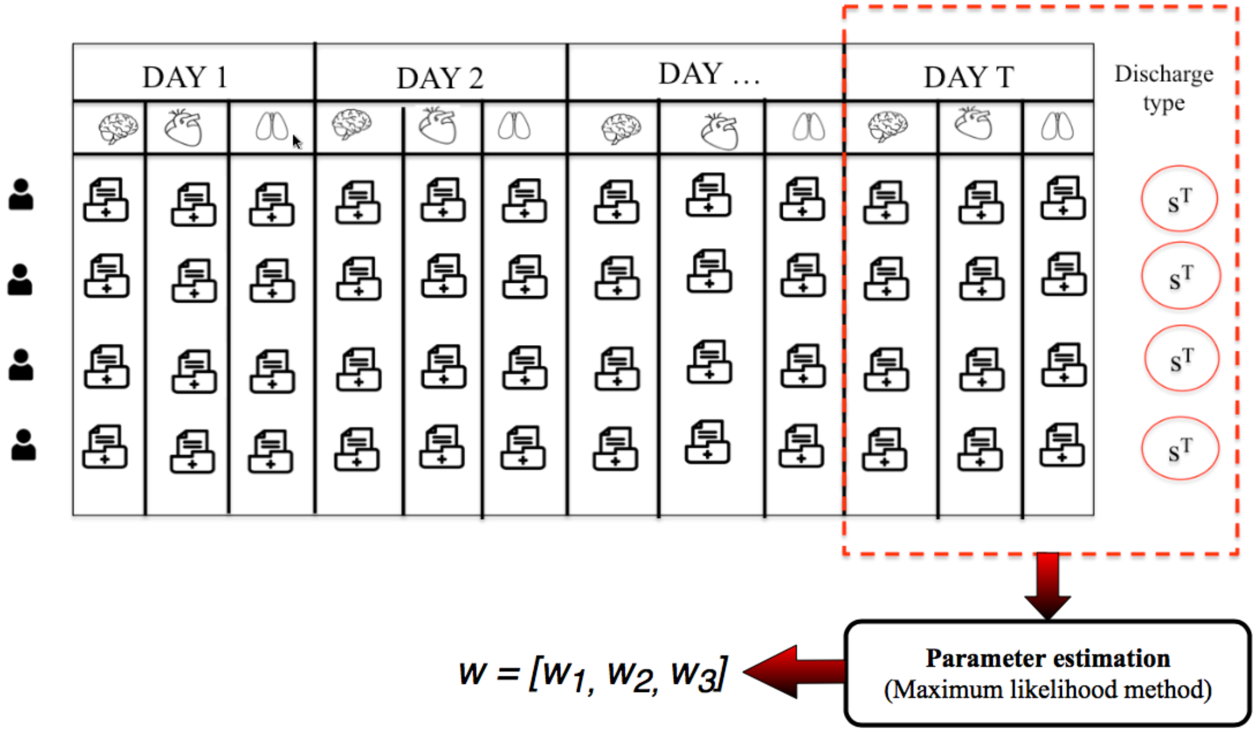


Figure 5.4: Weight estimation using maximum likelihood method

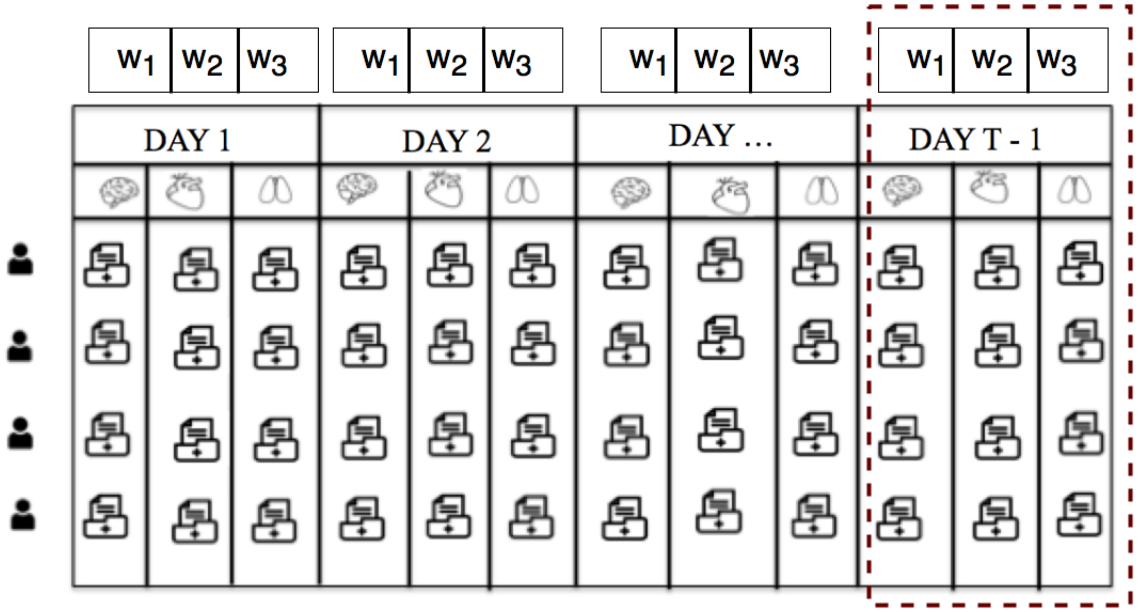


Figure 5.5: State prediction using estimated weights

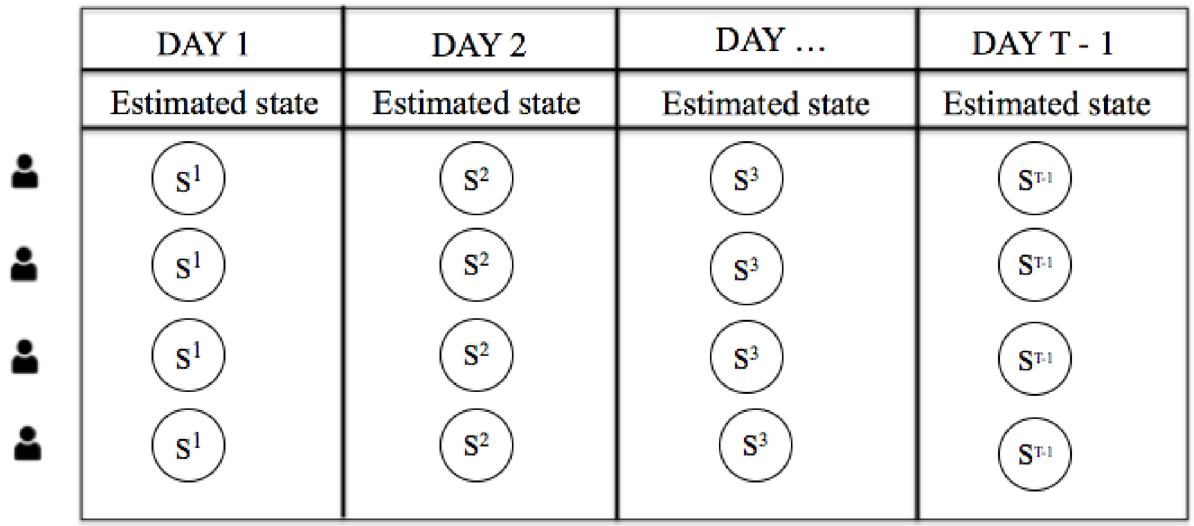


Figure 5.6: States are used to compute transition probabilities

To explain the proposed method mathematically, let $\phi(x_m^t)$ be

$$\phi(x_m^t) = [1, x_{m1}^t, x_{m2}^t, \dots, x_{mN}^t]$$

Where $m \in \{1, 2, 3, \dots, M\}$, $t \in \{1, 2, 3, \dots, T\}$.

- Retrieving training data observed at the final time interval

$$\phi(x_m^T) = [1, x_{m1}^T, x_{m2}^T, \dots, x_{mN}^T]$$

- Deriving cost function to estimate parameters

$$f(w) = \frac{1}{2} w^\tau w - \sum_{m=1}^{M} \left[\sum_{k=1}^{C} t_{mk} w_k^\tau \phi(x_m^T) - log \sum_{k=1}^{C} exp(w_k^\tau \phi(x_m^T)) \right] \qquad \text{(V.1)}$$

τ represents transpose. t_{mk} is a binary variable that signifies that k^{th} output state is observed for m^{th} data sequence.

- Estimate parameters

$$\hat{w} = \min_{w} f(w)$$

- Compute state probabilities for remaining time intervals {1, 2, 3, ..., T-1}

$$s^t_{ml} = \frac{exp(\hat{w}_l \phi(x^t_m))}{\sum_{k=1}^{C} exp(\hat{w}_k \phi(x^t_m))} \quad \forall \quad l \in \{1, 2, \ldots, C\} \tag{V.2}$$

$$S^t_m = arg \max\{s^t_{m1}, s^t_{m2}, \ldots, s^t_{mC}\}$$

For the purpose of notation, the capital letters (S) indicate random variables, and the lower-case letters (s) indicate observed value. Now, using assigned states, transition and emission probabilities can be estimated using following expression:

$$P(S^t = s^{t-1} | S^{t-1} = s^t) = \frac{\text{Expected number of transitions from } s^t \text{ to } s^{t-1}}{\text{Expected number of transistion from state } s^t}$$

$$P(X = x | S = s) = \frac{\text{Expected number of times system in state } s \text{ and observation is } x}{\text{Expected number of times system in state } s}$$

1.2 Time Complexity of the Proposed Method

In this section, we derive the time complexity per iteration of the proposed approach in terms of big $\mathcal{O}$ notation. As explained earlier in Section 1.1, our method is the combination of two phases: supervised learning and prediction; hence, we show the complexity of each step separately. For simplicity, we calculate an approximate number of operations rather than the actual number of operations.

Supervised learning:

In this step, we first compute the cost function and then update weights to optimize the cost function. For the calculation of the cost function, we need to evaluate Equation V.1, that is the linear combination of three parts. Part 1 requires $(N + 1)$ multiplications (+1 for bias) and N summations. Part 2 requires about $2 * M * C * (N + 1)$ operations and Part 3 requires approximately $2 * M * C * N$ operations. In all, the computation of Equation V.1 is the order of $\mathcal{O}(M * C * N)$. Newton - Conjugate Gradient is employed to update weights. The time complexity per iteration for Newton - Conjugate Gradient is $\mathcal{O}(M * N)$ [98]. Hence, the overall complexity of the learning phase is $\mathcal{O}(M * C * N)$.

Prediction:

The prediction phase requires the weighted sum of the observed data. The state probabilities can be computed using $\mathcal{O}(M * N * C)$ operations (Equation V.2). The state with maximum probability can be found in $\mathcal{O}(C)$. The aggregated number of operations per iteration from both phases is bounded by $\mathcal{O}(M * C * N)$. Therefore, overall complexity of the proposed method is $\mathcal{O}(M * C * N)$.

K-means is an alternate approach to initialize the *Baum-Welch* algorithm [60]. Therefore, we wanted to investigate the complexity of our approach against *K-means*. *K-Means* is an NP-hard problem [99]. However, a few popular heuristic methods provide approximate solution [100, 101]. The complexity of *K-means* is very well established; therefore, we briefly explain the concept. Each iteration of *K-means* consists of three primarily steps: calculation of distance, data point assignment and re-estimation of the centroid. The time complexity per iteration is bounded by $\mathcal{O}(M * K * N)$ operations, where K is the number of centroids.

We notice that the time complexity per iteration for both approaches is linear in terms of data size (given $M >> N, K$). The missing part of time complexity analysis is the number of iterations to converge. Therefore, we compare both methods empirically.

1.3 Modified Baum-Welch Algorithm

In this section, we present an overview of the Baum-Welch algorithm with multiple features observed. In this algorithm, the likelihood of the observed set of sequences is maximized. Although theory to estimate parameters for HMMs with one variable observed is readily available, the details for HMM with multiple variables observed is not easily found. Therefore, in this section, we present a modified *Baum-Welch* algorithm to accommodate multiple features.

The estimation of transition and emission probabilities requires computation of forward ($\alpha^t(s)$) and backward ($\beta^t(s)$) variables [10]. The forward variable is defined as joint probability of partial observed sequence ($\mathbf{X^1}, \mathbf{X^2}, \ldots, \mathbf{X^t}$) until time t and system being in state s given HMM parameters (λ).

$$\alpha^t(s) = P^t(\mathbf{X^1}, \mathbf{X^2}, \ldots, \mathbf{X^t}, S^t = s|\lambda)$$

where $\mathbf{X^t} = \{x_1^t, x_2^t, \ldots, x_N^t\}$

We assume that features observed at time t are independent of each other. The modified procedure to compute forward probabilities is explained as follows.

1. *Initialization:* Initialize the joint probability of initial observation

$$\alpha^1(s) = \alpha^0(s) \prod_{i=1}^{N} P(x_i^1|S^1 = s, \lambda) \quad for \quad s \in \{1, 2, \ldots, S\}$$

where $\alpha^0(s)$ is the initial probability of the system being in state s when no observation was observed.

2. *Induction:* for t = 2 to T-1

$$\alpha^t(s) = \left[\sum_{i=1}^{S} \alpha^{t-1}(i) P(S^t = s|S^{t-1} = i) \right] \prod_{n=1}^{N} P(x_n^t|S^t = s) \quad for \quad s \in \{1, 2, \dots, S\} \tag{V.3}$$

The backward variable ($\beta^t(s)$) is defined as a probability of partial observation from $t+1$ to the end T ($\mathbf{X^{t+1}}, \mathbf{X^{t+2}}, \dots, \mathbf{X^T}$) given system state at time t and the HMM parameters.

$$\beta^{t+1}(s) = P^t(\mathbf{X^{t+1}}, \mathbf{X^{t+2}}, \dots, \mathbf{X^T}|S^t = s, \lambda)$$

The modified procedure to compute backward probabilities is explained as follows.

1. *Initialization:* This terms arbitrarily defines backward variable at time T as 1.

$$\beta^T(s) = 1 \quad for \quad s \in \{1, 2, \dots, S\}$$

2. *Induction:* for t = T-1 to 1

$$\beta^t(s) = \sum_{i=1}^{S} \left[P(S^{t+1} = i|S^t = s) \left[\prod_{n=1}^{N} P(x_n^{t+1}|S^{t+1} = i) \right] \beta^{t+1}(i) \right] \quad for \quad s \in \{1, 2, \dots, S\} \tag{V.4}$$

The modified forward and backward procedures are explained in Algorithm 1 and 2, respectively. Algorithm 3 uses modified forward and backward algorithm to learn HMM parameters. Let Z_i be i^{th} sequence of observations, $\vec{\mathbf{Z_i}} = \{\mathbf{X^1}, \mathbf{X^2}, \dots, \mathbf{X^T}\}_i$.

Algorithm 1 Forward algorithm

```
function Forward($\vec{\mathbf{Z_m}}, \alpha^0, P(S^{t-1}|S^t), \big[P(X_1|S), P(X_2|S), \ldots, P(X_N|S)\big]$)
    $B_j(\mathbf{X^1}) \leftarrow \prod_{n=1}^{N} P(x_n^1|s^t = j)$        ▷ $\forall j \in \{1, 2, \ldots, S\}$
    $\alpha_j^1 \leftarrow \alpha_j^0 B_j(\mathbf{X^1})$
    for $t \in \{1, 2, \ldots, T-1\}$ do
        for $j \in \{1, 2, \ldots, S\}$ do
            $B_j(\mathbf{X^{t+1}}) = \prod_{n=1}^{N} P(x_n^{t+1}|S^{t+1} = j)$
            $\alpha_j^{t+1} = \big[\sum_{i=1}^{S} \alpha_i^t P(S^{t+1} = j|S^t = i)\big] B_j(\mathbf{X^{t+1}})$
        end for
    end for
    $P(\vec{\mathbf{Z_m}}|\lambda) = \sum_{i=1}^{S} \alpha_i^T$        ▷ $\lambda$ is a collection of parameters
    return $\alpha, P(\vec{\mathbf{Z_m}}|\lambda)$
end function
```

Algorithm 2 Backward algorithm

```
function Backward($\vec{\mathbf{Z_m}}, \alpha^0, P(S^{t-1}|S^t), \big[P(X_1|S), P(X_2|S), \ldots, P(X_N|S)\big]$)
    $\beta_j^T \leftarrow 1$        ▷ $\forall j \in \{1, 2, \ldots, S\}$
    for $t = \{T-1, T-2, \ldots, 1\}$ do
        for $i = \{1, 2, \ldots, S\}$ do
            $sum \leftarrow 0$
            for $j = \{1, 2, \ldots, S\}$ do
                $B_j(\mathbf{X^{t+1}}) \leftarrow \prod_{n=1}^{N} P(x_n^{t+1}|S^{t+1} = j)$
                $sum \leftarrow P(S^{t+1} = j|S^t = i) B_j(\mathbf{X^{t+1}}) \beta_j^{t+1} + sum$
            end for
            $\beta_i^t \leftarrow sum$
        end for
    end for
    return $\beta$
end function
```

We also include Python implementation for the readers in Appendix B. This code is an extension of the script written by [102].

2 Results and Discussion

The performance of the proposed method is measured in three dimensions:

1. Speed

Algorithm 3 Parameter learning

```
procedure LEARNING ($\vec{\mathbf{Z}}, \alpha^0, P(S^{t-1}|S^t), \big[P(X_1|S), P(X_2|S), \ldots, P(X_N|S)\big]$)
    for $m \in \{1, 2, \ldots, M\}$ do
        $\alpha, P(\vec{\mathbf{Z_m}}|\lambda) \leftarrow Forward(\vec{\mathbf{Z_m}}, \alpha^0, P(S^{t-1}|S^t), \big[P(X_1|S), P(X_2|S), \ldots, P(X_N|S)\big])$
        $\beta \leftarrow Backward(\vec{\mathbf{Z_m}}, \alpha^0, P(S^{t-1}|S^t), \big[P(X_1|S), P(X_2|S), \ldots, P(X_N|S)\big])$
                                                            ▷ $\alpha$ and $\beta$ are matrices
        $\gamma_i^t \leftarrow \frac{\alpha_i^t \beta_i^t}{P(\vec{\mathbf{Z_m}}|\lambda)}$          ▷ $\forall i \in \{1, 2, \ldots, S\}, \forall t \in \{1, 2, \ldots, T\}$
        for $t \in \{1, 2, \ldots, T-1\}$ do
            $B_j(\mathbf{X^{t+1}}) \leftarrow \prod_{n=1}^{N} P(x_n^{t+1}|S^{t+1} = j)$          ▷ $\forall j \in \{1, 2, \ldots, S\}$
            $\zeta^t(i,j) \leftarrow \frac{\alpha_i^t P(S^{t+1}=j|S^t=i) B_j(\mathbf{X^{t+1}}) \beta_j^{t+1}}{P(\vec{\mathbf{Z_m}}|\lambda)}$          ▷ $\forall i, j \in \{1, 2, \ldots, S\}$
        end for
        $P^m(i,j) \leftarrow \sum_{t=1}^{T-1} \zeta^t(i,j)$          ▷ $\forall i, j \in \{1, 2, \ldots, S\}$
        $Q_i^m \leftarrow \sum_{t=1}^{T-1} \gamma_i^t$          ▷ $\forall i \in \{1, 2, \ldots, S\}$
        $R^m(i, x_n) \leftarrow \sum_{t=1; X_n = x_n}^{T} \gamma_i^t$
        $S_i^m \leftarrow \sum_{t=1}^{T} \gamma_i^t$
    end for
    $P(S^{t+1} = j|S^t = i) \leftarrow \frac{\sum_{m=1}^{M} P^m(i,j)}{\sum_{m=1}^{M} Q_i^m}$
    $P(x_n|S = i) \leftarrow \frac{\sum_{m=1}^{M} R^m(i,x_n)}{\sum_{m=1}^{M} S_i^m}$
end procedure
```

2. Proximity

3. Integration

In *speed*, we investigate the time to compute the initial values. In *proximity*, we determine the deviation between the true parameters and the estimated initialization. In *integration*, we aim to probe the performance of the *Baum-Welch* algorithm after integrating the proposed initialization.

2.1 Speed

In Section 1.2, we determined the time complexity per iteration for the proposed method to be $\mathcal{O}(M * C * N)$. Theoretically, for a large data ($M >> N$), the time complexity is linear in time. We also have the knowledge that the time complexity per iteration for initialization using *K-means* is $\mathcal{O}(M * K * N)$ which is also linear in time for the large data $(M >> N)$. For this work, the number of centroid K is dictated by the number of classes in the output variable. Therefore, we compare the overall time complexity of both initialization methods empirically.

The experiments are performed using the simulated data. For the given HMM parameters (transition and emission probabilities), multiple sequences of states were generated. The parameters were selected following the principals explained in [60]. Figure 5.7 presents a flow chart to systematically illustrate the procedure used to generate state sequences. Let us assume that we are interested in simulating M independent observation sequences. Here, we explain the procedure used to generate one sequence of observations, with each observation of length N (number of features). M sequences can be generated by replicating the same procedure. To obtain a sequence of observations, first, we need to generate a sequence of states, then, by integrating known states and emission probabilities, the observations of multiple features in each time interval were probabilistically determined. Note that each feature has its emission probabilities that reflect the chances of appearing an observation given the state.

For simulating a sequence of length T, first a vector of size T was generated with elements having a uniform distribution between 0 and 1 (Figure 5.7). Using each element of this vector, a state is determined following the *Roulette-Wheel* algorithm [103]. The *Roulette-Wheel* algorithm uses the uniform random number in combination with a probability distribution vector to determine the outcome. For our application, this algorithm can be explained as follows and is embedded in Figure 5.7.

1. Generate a uniform random value u in the range (0, 1]

2. Using binary search, find the index i of the smallest element in the state transition probability vector larger than u

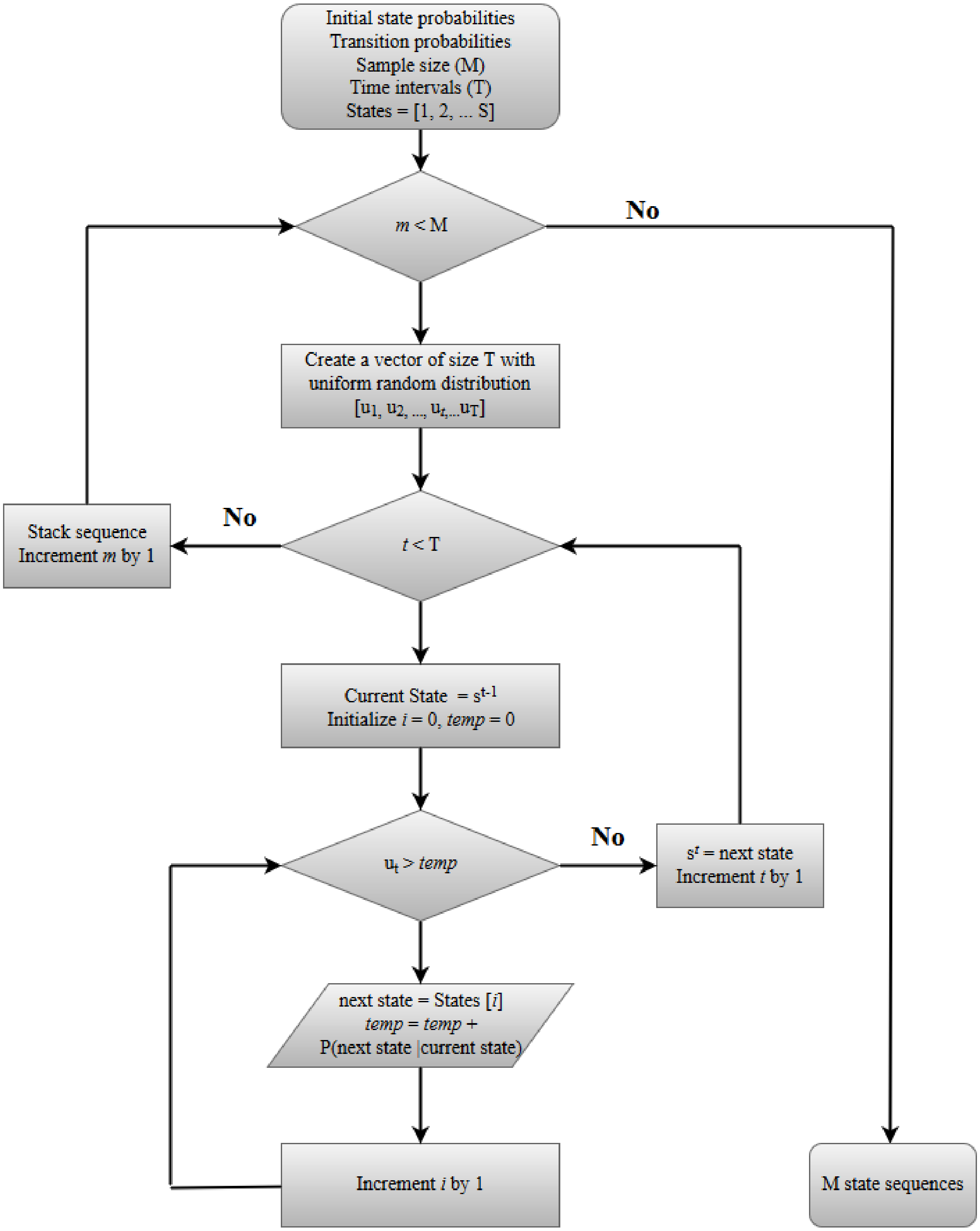


Figure 5.7: Flow chart for generating the simulated data

Following the data simulation, we compare the time taken by both *K-means* and our approach to compute initial values. In Figure 5.8, X-axis represents the sample size (in 1000s) and Y-axis represents the computation time (in seconds) for initial values. As expected the computation time grows linearly in both cases. However, as the sample size increases, the difference in the computation time grows. With a 10 fold increase in the sample size, the computation time difference grows to 50%. This implies that the scale factor associated with the time complexity of our approach is smaller than the scale factor associated with the time complexity of *K-means* approach. We also performed similar experiments with the three numbers of classes in the output variable (Figure 5.9).

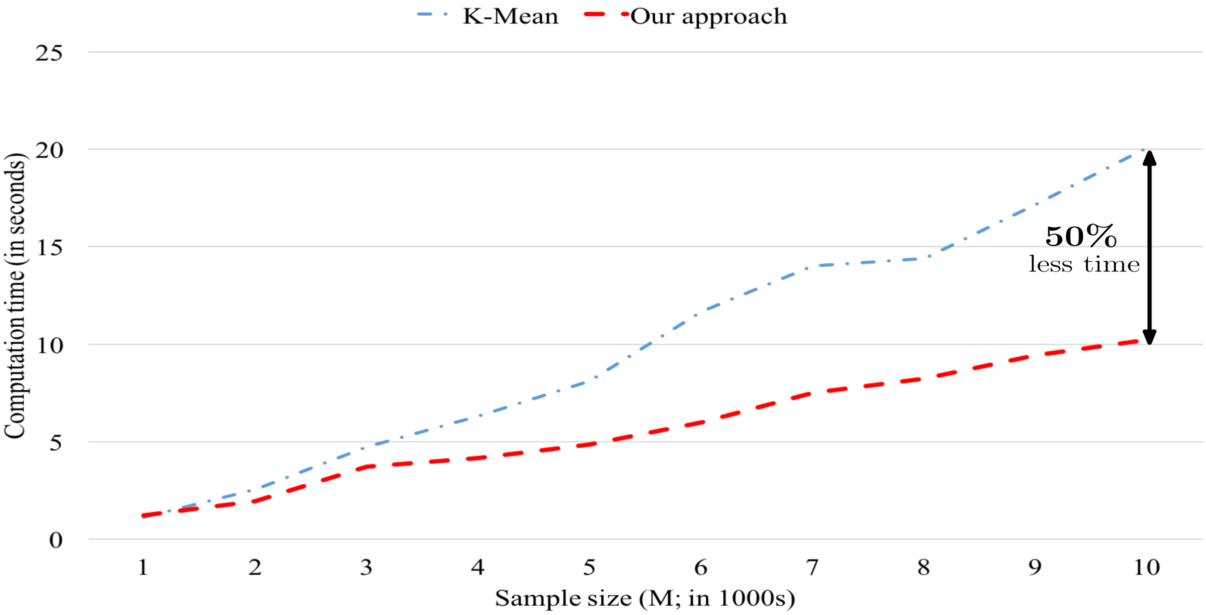


Figure 5.8: Computation time to determine initialization (number of features (N): 4, number of classes in output variable (C) = 4, time length for each sequence (T) = 50)

2.2 Proximity

The objective of measuring the proximity is to explore the goodness of the estimated initial values. A low deviation implies that the estimated initial probability distribution is

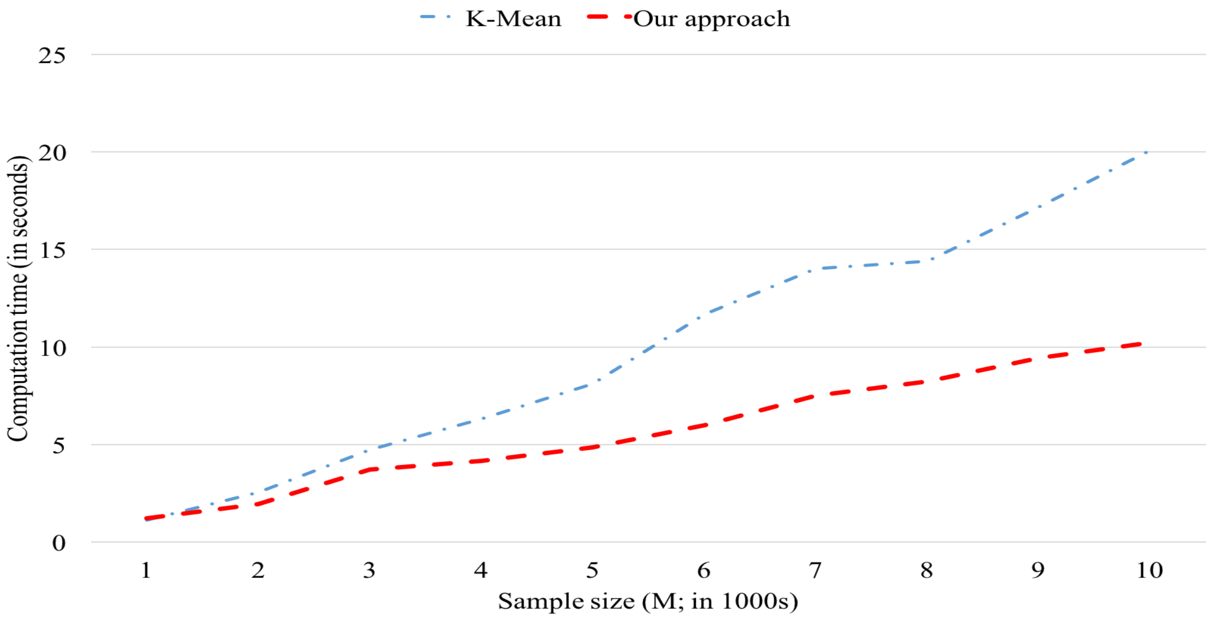


Figure 5.9: Computation time to determine initialization (number of features $(N) = 4$, number of classes in output variable $(C) = 3$, time length for each sequence (T): 50)

closer to the true structure of the data and exists a possibility of early convergence of the *Baum-Welch* algorithm. The proximity is measured using *Kullback-Leibler (KL) Divergence* and *log-likelihood*.

Kullback-Leibler (KL) Divergence: It measures the difference between two probability distributions using Equation V.5.

$$D_{KL}(p \parallel q) = \sum_{s^t=1}^{S} \sum_{s^{t-1}=1}^{S} P(s^t|s^{t-1}) ln \frac{P(s^t|s^{t-1})}{Q(s^t|s^{t-1})} \tag{V.5}$$

where p and q are the estimated and true state (or referenced) state transition distributions. The smaller the *KL-Divergence* the better the proximity.

Log-likelihood: It measures the fitness of the observed data on the given parameters, and can be expressed using Equation V.6.

$$l = \sum_{m}^{M} logP(\tilde{\mathbf{Z_m}}|\lambda) \tag{V.6}$$

The greater the *log-likelihood* the better the proximity. In our experiments, the *log-likelihood* obtained from true parameters is compared to the *log-likelihood* obtained from both the random and the proposed initialization methods. The results enable us to understand the performance differences of both initialization approaches.

Table 5.1 shows computed *KL-Divergence* and *log-likelihood* on the different instances of the simulated data. In the table, the notation Data_S_N_I incorporates three numeric values representing the number of hidden states (S), the number of features (N) and the index for data obtained using varying parameters. The transition and emission probabilities were randomly selected to obtain multiple instances of the data. We followed the structure, explained in [60], to determine the transition probabilities for generating the data. The number of hidden states varies from 3 to 5 with an increment of 1.

We were not able to compare the proposed method to the *K-means* method in terms of *KL-Divergence* and *log-likelihood* because *K-means* does not facilitate the annotation of states. Therefore, we investigate the strengths of our method by comparing the performance with the alternative approach usually used in literature to initialize the *Baum-Welch* algorithm. In the absence of prior knowledge, a uniform distribution is considered for transition probabilities. The emission probabilities are assumed to be randomly distributed. We avoided the use of uniform distribution for emission probabilities because a fix distribution causes the loss in degree of freedoms. Due to the randomness in the observation probabilities, 50 trials were executed to calculate the average performances (*KL-Divergence* and *log-likelihood*). The *KL-Divergence* between the true transition probabilities and the transition probabilities obtained from the uniform distribution is recoded in Column *KL-Divergence - Uniform distribution* *(avg.))*. Column *KL-Divergence - Our approach* shows the *KL-Divergence* between the true parameters and the guided initial value of parameters. The *log-likelihood* was also computed for the random and the guided initialization following the same procedure. We also list the initial value computation time to understand the time expense in gaining the benefit from the proposed method.

From Table 5.1, we observed that the proximity of the true parameter distribution from the guided initial values, in terms of *KL-Divergence*, is closer than the uniform values. These results indicate that the probability distribution obtained from our approach is similar to the true distribution. In terms of *log-likelihood*, the parameters which fit the data well show high *log-likelihood*. In Table 5.1, our approach shows greater *log-likelihood* than the uniform initialization. An average difference between maximum achievable *log-likelihood* and both the guided and the uniform distribution is 1% and 29%, respectively.

The reason for the improved performance can be attributed to careful assignment of hidden states from the known observation. In our approach, the assignment of initial hidden states is determined by the combination of the observed data and estimated weights corresponding to each feature. However, in random initialization, the initial state assignment is independent of the observations.

Table 5.1: KL-Divergence and log-likelihood on the simulated data. M = 500, T = 50, N = 4, Number of random trial = 50

		KL-Divergence		Log-likelihood					
Data	Hidden states	Uniform distribution (avg)	Our approach	Uniform distribution (U)	Our approach (G)	True parameters (T)	(T-U)/T	(T-G)/T	Time to compute initialization (in sec)
Data_3_4_1	3	3.03	0.35	-65757.42	-47082.83	-46180.06	42%	2%	0.29
Data_3_4_2	3	0.94	0.07	-65382.26	-49851.02	-49559.64	32%	1%	0.44
Data_3_4_3	3	0.94	0.17	-65561.65	-53983.34	-53503.29	23%	1%	0.46
Data_3_4_4	3	0.77	0.12	-65824.18	-49642.97	-49192.34	34%	1%	0.36
Data_3_4_5	3	3.98	0.61	-65533.40	-45577.65	-44615.24	47%	2%	0.26
Data_4_4_1	4	2.91	1.08	-63936.83	-47024.44	-46238.89	38%	2%	0.60
Data_4_4_2	4	1.90	0.46	-62847.99	-50235.23	-49822.92	26%	1%	0.44
Data_4_4_3	4	1.81	0.49	-63975.96	-47460.53	-46845.34	37%	1%	0.38
Data_4_4_4	4	1.73	0.44	-62888.75	-50319.54	-49677.48	27%	1%	0.36
Data_4_4_5	4	0.72	0.16	-63630.40	-50019.66	-49619.05	28%	1%	0.48
Data_5_4_1	5	3.37	2.05	-62279.74	-52093.18	-51096.76	22%	2%	0.34
Data_5_4_2	5	1.07	0.78	-62207.62	-52524.08	-51764.65	20%	1%	0.40
Data_5_4_3	5	2.27	1.17	-62068.68	-52871.01	-52106.73	19%	1%	0.38
Data_5_4_4	5	1.19	1.19	-62538.71	-51259.13	-50523.04	24%	1%	0.38
Data_5_4_5	5	0.94	0.56	-61783.81	-56458.79	-55697.96	11%	1%	0.32
Mean log-likelihood difference from the true log-likelihood							29%	1%	

2.3 Integration

The objective of the integration is to probe how well our approach works with the *Baum-Welch* algorithm. To establish the advantage of our method, two scenarios were considered. Each scenario is different by the type of initialization employed to start the *Baum-Welch* algorithm. Figure 5.10 shows the experimental settings; the setting is divided into four steps: data generation, initialization, learning, and evaluation. Each phase is explained as follows.

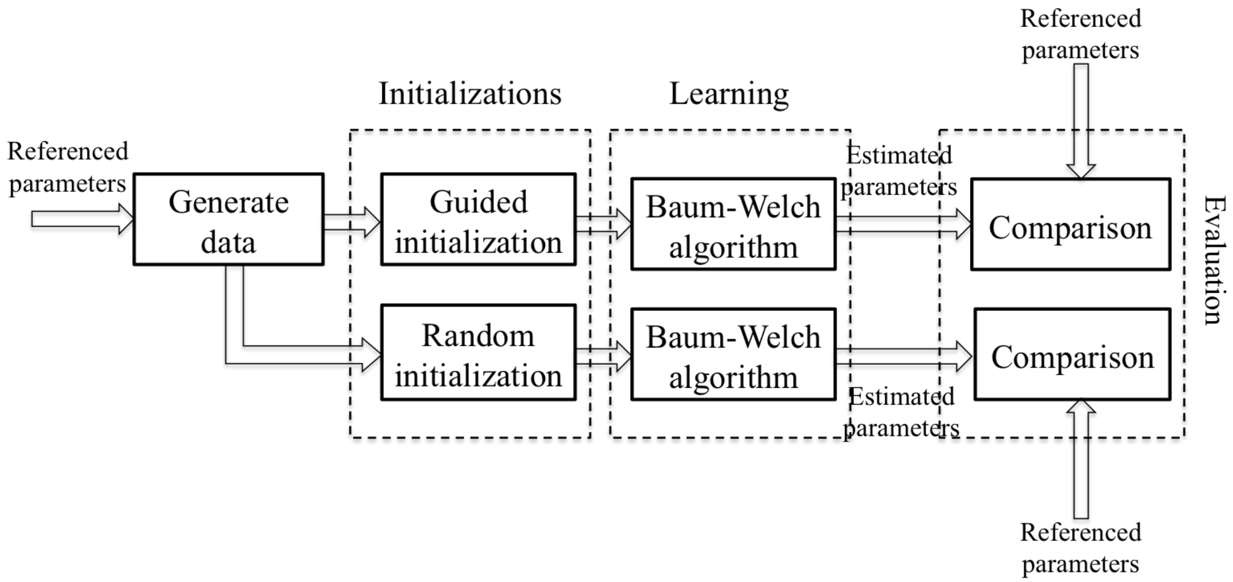


Figure 5.10: Experimental design using synthetic data

1. **Generate data:** The synthetic data was generated from referenced parameters (*transition* and *emission* probabilities). For the experiment in this section, the transition probabilities are adapted from [104] and explained in Table 5.2. From known parameters, pseudo-sequences of states were generated. Then, each state probabilistically generates observations guided by the emission probabilities. A detailed description of the generation of the simulated data is provided in Section 2.1.

2. **Initialization:** The learning of HMM parameters requires an initial set of parameters to begin. Two initialization approaches (uniform and guided) were used to initialize the *Baum-Welch* algorithm. The guided initialization procedure (our approach) is explained in Section 1.1.

3. **Learning:** After initialization, the learning was performed using the *Baum-Welch* algorithm. In each iteration, the algorithm updates parameters to maximize the *log-likelihood.*

4. **Evaluation:** To understand the integration of the proposed method with *Baum-Welch* algorithm, we capture the trajectory of parameter updates. For each iteration, we computed *log-likelihood* to illustrate the learning path.

For simulating the data, as mentioned earlier, the referenced parameters are adapted from [104]. Table 5.2 shows the transition probabilities. In this table, the different states of the health (Excellent, Very Good, Good, Fair and Poor) represent five hidden states. Jung (2006) does not facilitate the emission probabilities [104]; therefore, random distributions are selected to generate data.

Table 5.2: Transition probabilities adapted from Jung (2006)

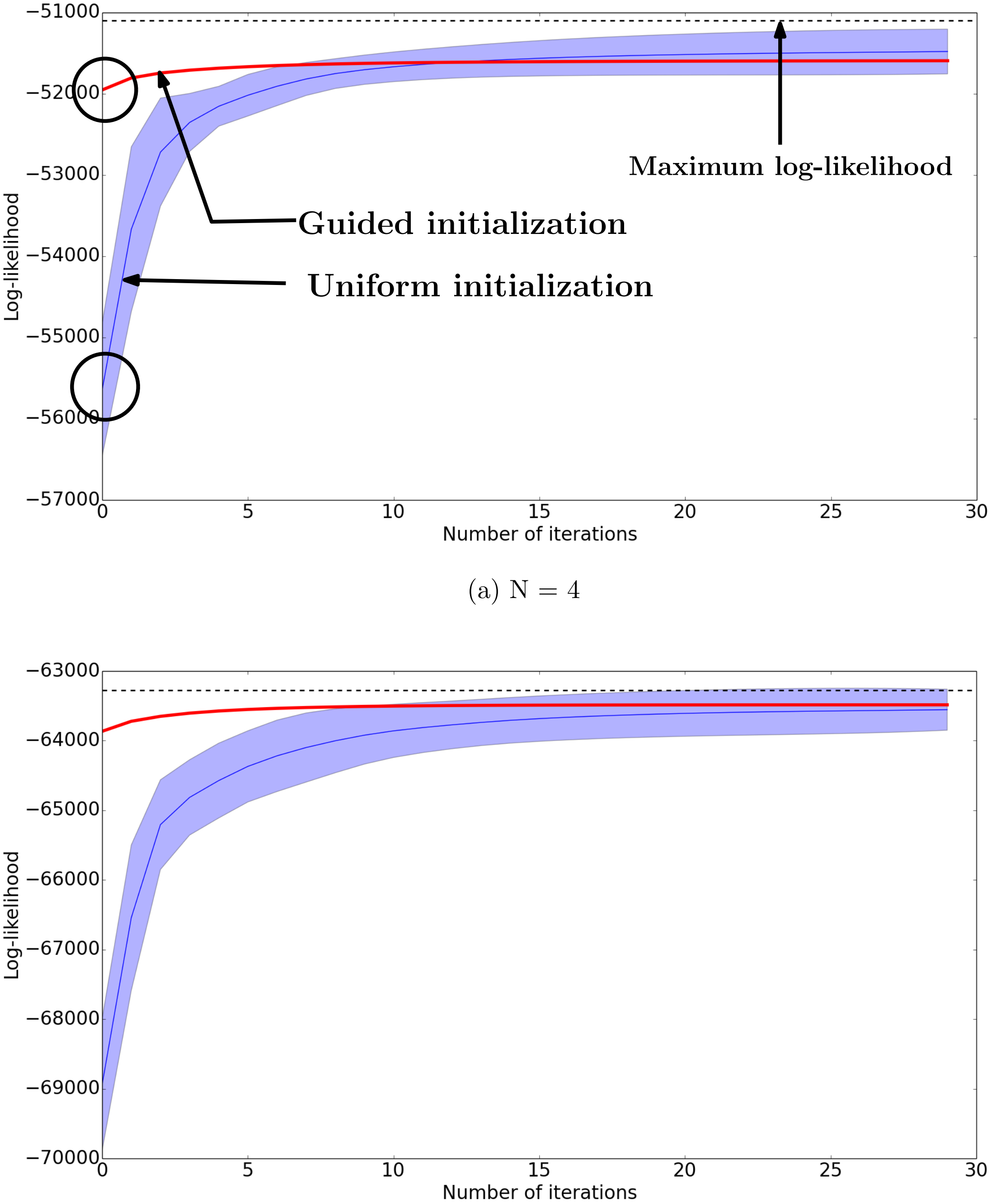
	Health				
Health	Excellent	Very Good	Good	Fair	Poor
Excellent	0.514	0.332	0.117	0.029	0.009
Very Good	0.136	0.512	0.274	0.063	0.015
Good	0.040	0.222	0.504	0.192	0.043
Fair	0.015	0.070	0.255	0.491	0.169
Poor	0.006	0.023	0.089	0.304	0.578

The simulated data from the referenced parameters were used to train the HMM. The *Baum-Welch* algorithm was initialized using both initialization methods (uniform distribution and our approach). The numeric experiments are performed considering the number of sequences be 200 (M), with each sequence of length 50 (T).

Figures 5.11 and 5.12 shows the benefit of good initial values for the *Baum-Welch* algorithm by comparing the *log-likelihood* trajectory. In the figure, X-axis represents the number of iterations, and Y-axis represents the *log-likelihood*. The blue color shows the trajectory of the uniform initialization and the red color shows the trajectory of the guided initialization. In the case of uniform initialization of transition probabilities, the emission probabilities are selected randomly. Therefore, we performed 50 trials to compute the mean performance with 95% confidence interval. The solid blue line shows the mean and the filled blue region shows the 95% confidence interval. The black dotted line indicates the maximum achievable *log-likelihood* obtained from true parameters. The experiments were performed considering the different number of features (N). It is evident that the better initial values lead to faster performance of the *Baum-Welch* algorithm. Out of four, three cases reach greater maximum *log-likelihood* using our guided initialization than the maximum *log-likelihood* obtained from the uniform initialization. These experiments present evidence that our initialization approach can help the *Baum-Welch* alogrithm to converge in a local optimum.

3 Conclusions

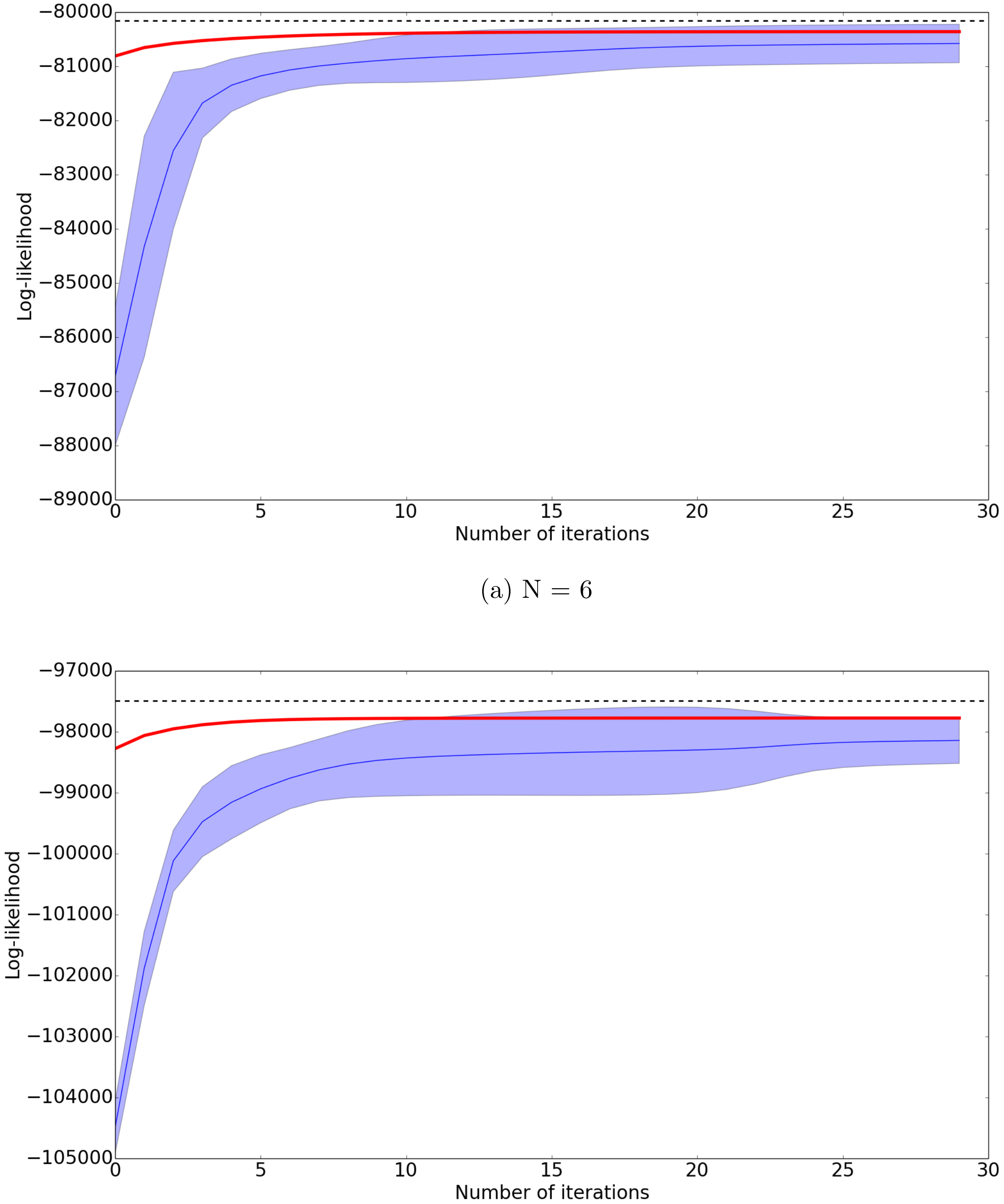
In this study, we developed an initialization procedure following the principal of multinomial logistic regression. This procedure generates a pseudo-sequence of hidden states based off observed data. Using pseudo-hidden states, the initial values of the HMM parameter are derived. We evaluated the performance of the proposed method in three dimensions: *speed*, *proximity* and *integration*. The results showed that the proposed initialization methodology is significantly better than the alternate initialization methods, and yields distribution which is closer to the true distribution. This method can be applied to build a multi-stage progression model using EHR data to monitor the trajectory of the disease. The main limitation of this study is that data need to have known true state at the final time interval.



(a) N = 4

(b) N = 5

Figure 5.11: Log-likelihood trajectory of *Baum-Welch* algorithm with random (blue) and guided initialization (red). M = 200, T = 50



(a) N = 6

(b) N = 7

Figure 5.12: Log-likelihood trajectory of *Baum-Welch* algorithm with random (blue) and guided initialization (red). M = 200, T = 50

Chapter VI

PROGNOSTIC ACCURACY OF SIRS AND qSOFA

In this chapter, the performance characteristics of SIRS and qSOFA are evaluated. The association of SIRS ≥ 2 and qSOFA ≥ 2 with mortality was compared using OR approach. We divided the patient visits at the ED into ten classes to investigate the trend of OR of SIRS ≥ 2 and qSOFA ≥ 2 across varying initial risk. The results obtained from OR analysis provided unbalanced combination of sensitivity and specificity of SIRS and qSOFA. Therefore, instead of considering SIRS or qSOFA score, we used SIRS and qSOFA variables to estimate the probability of risk using modeling approaches. Adopting the approach from [95] for finding the threshold to translate predicted probabilities into labels, the modeling approaches provided a balance between sensitivity and specificity. This study has great clinical implications as it provides evidence in favor of effective screening criteria that enables healthcare providers to effectively manage limited resources of ED.

1 Method

1.1 Study Setting and Population

Similar to the study explained in Chapter III, this study also used data from the Cerner Corporations HIPAA-compliant Health Facts database. All visits for patients admitted to the ED between January 1, 2009 to December 31, 2015 with primary or secondary diagnoses of sepsis (ICD-9-CM code 995.91), severe sepsis (ICD-9-CM code 995.92), septic shock (ICD-9-CM code 785.52), and unspecified sepsis (ICD-9-CM 038.xx) were included in the initial data extraction. Extracted data included age, gender, U.S. Census region (Midwest, Northeast, South, and West), hospital location (urban/rural), SIRS/qSOFA clinical variables and discharge type. The outcome of interest was 28-day all-cause mortality. The study included patients 18 years of age and older with length of stay less than or equal to 28 days. Of the 230,451 extracted encounters, 97,747 encounters were excluded (20,639 under 18 years old; 12,276 with length of stay (LOS) greater than 28 days; and 64,832 with non-emergency admission). Our analytical sample consisted of 132,704 encounters occurring in the ED (Figure 6.1).

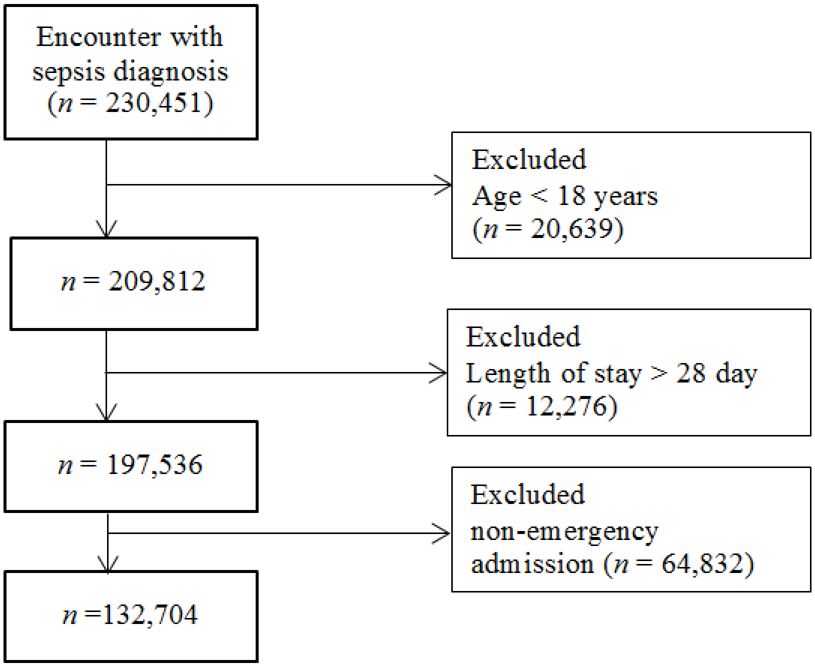

Encounter with sepsis diagnosis (n = 230,451)
Excluded Age < 18 years (n = 20,639)
n = 209,812
Excluded Length of stay > 28 day (n = 12,276)
n = 197,536
Excluded non-emergency admission (n = 64,832)
n =132,704

Figure 6.1: Encounter flow chart

1.2 Data Analysis

Data preparation and statistical analysis were performed using R (Version 3.2.5). The two approaches, OR and modeling methods (Logistic Regression (LR), Decision Tree (DT) and Nave Bayes (NB)), were used to compare the associations of SIRS and qSOFA with 28-day in-hospital mortality. The OR is a statistical parameter used to determine the strength of association between two categorical variables. Note that the association is statistically significant if the range of 95% Confidence Interval (CI) of OR does not contain the value 1.

To investigate the robustness of two diagnostic criteria, the patient cohort was divided into ten classes of varying baseline risk [105]. The baseline risk was estimated using patient demographic and hospital characteristic including age, sex, hospital location, and U.S. census region. The OR was computed to compare the association of mortality with SIRS (≥ 2 vs. ≤ 2) and qSOFA (≥ 2 vs. ≤ 2) across each decile.

For further investigation, the modeling approaches (LR, DT [106] and NB [107]) were employed on SIRS and qSOFA variables. Following the literature, the performance matrices of SIRS and qSOFA were evaluated above the baseline risk [95, 108]. The set of variables used for modeling purpose are summarized as follows:

- **Baseline variables:** Age, sex, hospital location (urban/rural), and U.S. census region.
- **Base + SIRS variables:** Baseline variables plus SIRS clinical variables.
- **Base + qSOFA variables:** Baseline variables plus qSOFA clinical variables.

We assessed the discriminatory power of each model using the AUROC curve. Other performance parameters such as sensitivity, specificity, positive predictive value and negative predictive value were also used. Table 6.1 represents the confusion matrix. Equations 1-6 are the expressions of performance metrics derived from Table 6.1. The threshold to determine the predicted class from estimated probability was decided using the roc plot. The cutoff that resulted a point in the roc plot closest to (0, 1) was selected as threshold [95]. In (0, 1), the first index represents false positive rate and second represents true positive rate.

Table 6.1: Confusion matrix

		Actual value	
		Yes	No
Predicted value	Yes	TP	FP
	No	TN	TF

$$OR = \frac{TP \times TN}{FP \times FN} \tag{VI.1}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{VI.2}$$

$$Specificity = \frac{TN}{TN + FP} \tag{VI.3}$$

$$Positive \quad predictive \quad value = \frac{TP}{TP + FP} \tag{VI.4}$$

$$Negative \quad predictive \quad value = \frac{TN}{TN + FN} \tag{VI.5}$$

$$Accuracy = \frac{TN + TP}{TN + FP + FN + TN} \tag{VI.6}$$

2 Results and Discussion

2.1 Population Characteristics

Table 6.2 compares demographic, geographic, and clinical characteristics of non-expired and expired encounters. The mortality rate for ED encounters was 14%, with a median age of 75 years (Inter-quartile Range (IQR)= 60 to 83 years) and 50.4% male. In comparison, the non-expired group had a median age of 65 years (IQR = 52 to 78 years) and was 48.7% male. The Moods median test [108] showed a significant difference in median age between the groups, with age increasing mortality risk. Using 95% CI of OR, we found that gender, hospital location, and census region were significantly associated with mortality. The results showed that males were at higher risk than females (OR = 1.07, 95% CI = 1.01 to 1.10), and encounters at urban hospitals had lower levels of risk than rural hospitals (OR = 0.77, 95% CI = 0.75, 0.80). Similarly, we compared the association of mortality with hospital census region and found encounters in the Midwest and West to have lower risk for mortality, while those in the Northeast and South had higher risk. Table 6.2 also shows the association of individual variables of qSOFA and SIRS with mortality. For the qSOFA, each criterion showed a positive association with mortality: blood pressure (OR = 2.00; 95% CI = 1.94 to 2.08), Glasgow Coma Scale (OR = 2.96; 95% CI = 2.87 to 3.06), and respiratory rate (OR = 1.66; 95% CI = 1.61 to 1.72). Whereas, for SIRS, heart rate (OR = 1.09; 95% CI = 1.05 to 1.12), respiratory rate (OR = 1.67; 95% CI = 1.62 to 1.72), and white blood cell (OR = 1.04; 95% CI = 1.01 to 1.07) were associated with a higher risk of mortality. While respiratory rate had the smallest association with mortality among qSOFA criteria, it had the largest association among SIRS criteria. The association of the SIRS criterion of body temperature ($<$ 36°C or $>$ 38°C) with mortality (OR = 0.84; 95% CI = 0.80 to 0.87) was counterintuitive as it suggests that high or low body temperature reduces the risk of mortality.

Figure 6.2 shows the distribution of encounters by qSOFA and SIRS scores. There were 23,260 (17.5%) encounters that met the qSOFA criteria for mortality risk (qSOFA $\geq$ 2), whereas 80,015 (39.7%) encounters met the definition of (SIRS $\geq$ 2). Figure 6.3 shows the distribution of mortality for the qSOFA and SIRS scores, with rates markedly increasing across qSOFA scores compared to the relatively uniform distribution of SIRS scores. For qSOFA scores, mortality rates were 40.1% for the highest possible score of 3 and 8% for a score of 0. In contrast, for SIRS scores, mortality rates were 18.9% for the highest possible score of 4 and 11.3% with a score of 0.

Table 6.2: Population characteristic.

Variables	Non-expired (n = 114,030)	Expired (n = 18,674)	OR	95% CI
Demographics				
Age, median (IQR), year	65 (52-78)	75 (60-83)		
Male, n (%)	55,554 (48.7)	9,424 (50.4)	1.07	1.01, 1.10
Hospital location, n (%)				
Urban	118,193 (81.8)	22,028 (77.7)	0.77	0.75, 0.80
Census region, n (%)				
Midwest	21,857 (19.2)	2,691 (14.4)	0.71	0.68, 0.74
Northeast	35,862 (31.5)	6,357 (34.0)	1.13	1.09, 1.16
South	40,820 (35.8)	7,410 (39.7)	1.18	1.14, 1.22
West	15,491 (13.6)	2,216 (11.9)	0.86	0.82, 0.90
qSOFA criteria met, n (%)				
Systolic BP $\leq$ 100 mmHg	23,451(20.6)	6,387 (34.2)	2.00	1.94, 2.08
Glasgow Coma Scale $\leq$ 13	20,745 (18.2)	7,417 (39.7)	2.96	2.87, 3.06
Respiratory rate $\geq$ 22 breaths/min	35,667 (31.3)	8,046 (43.1)	1.66	1.61, 1.72
SIRS criteria met, n (%)				
Temperature < 36°C or > 38°C	27,676(24.3)	3,950(21.2)	0.84	0.80, 0.87
Heart rate > 90 beats/min	69,712 (61.1)	11,791(63.1)	1.09	1.05, 1.12
Respiratory rate > 20 breaths/min	37,253 (32.7)	8,356 (44.8)	1.67	1.62, 1.72
WBC count < 4 or > 12 K/μL	67,509 (59.2)	11,218 (60.0)	1.04	1.01, 1.07

2.2 Comparison of Prognostic Accuracy of qSOFA and SIRS Scores using OR

Confusion matrices for both SIRS and qSOFA are shown in Table 6.3 and 6.4, respectively. Both SIRS and qSOFA were significantly associated with mortality. The association was considerably stronger for qSOFA ≥ 2 (OR = 3.06; 95% CI = 2.96 to 3.17) than for SIRS ≥ 2 (OR = 1.22; 95% CI = 1.18 to 1.26). The classification accuracy, sensitivity, and specificity for qSOFA were 0.78, 0.35, and 0.85, respectively. The same performance parameters for SIRS were 0.44, 0.64, and 0.40, respectively. Using the cutoff of ≥ 2 for both measures, the qSOFA outperformed SIRS on accuracy and specificity, but demonstrated lower sensitivity.

Figure 6.4 compares the association of qSOFA and SIRS with mortality over deciles of baseline risk. The deciles of patients were derived by feeding baseline variables to multivariable

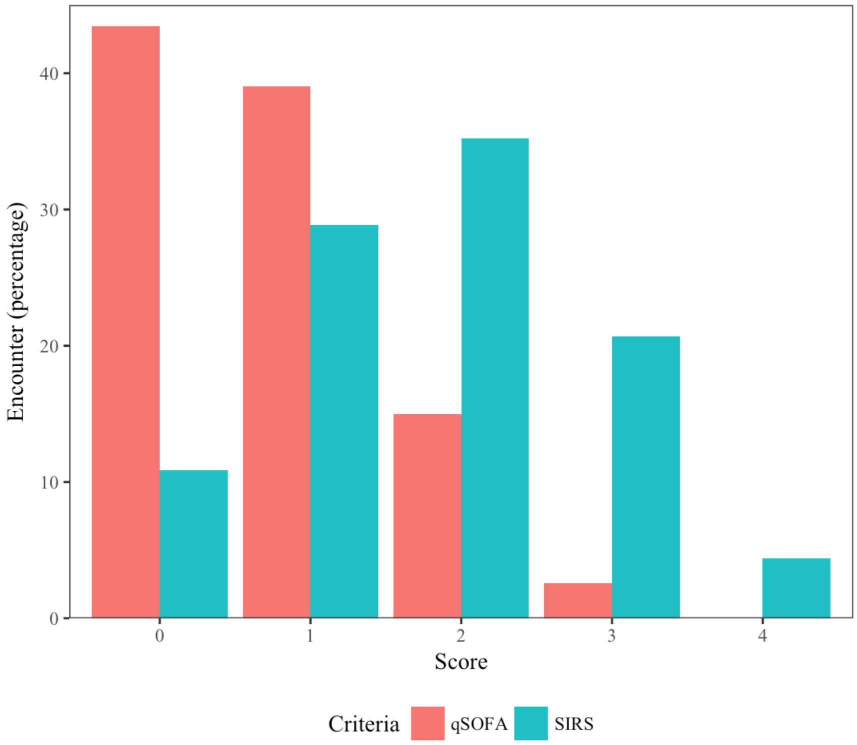


Figure 6.2: Distribution of encounters across qSOFA and SIRS scores.

Table 6.3: Confusion matrix for SIRS.

		Actual value	
		Yes	No
Predicted value	Yes	12021	67994
	No	6653	46036

Table 6.4: Confusion matrix for qSOFA.

		Actual value	
		Yes	No
Predicted value	Yes	6464	16796
	No	12210	97234

logistic regression. For each decile, the OR for qSOFA score ($\geq$ 2 vs. <2) was greater than for SIRS score ($\geq$ 2 vs. <2). The OR of qSOFA ranged from 4.3 among those in the lowest baseline risk decile to 2.4 among those in the highest decile, while ORs for SIRS across deciles were more or less constant.

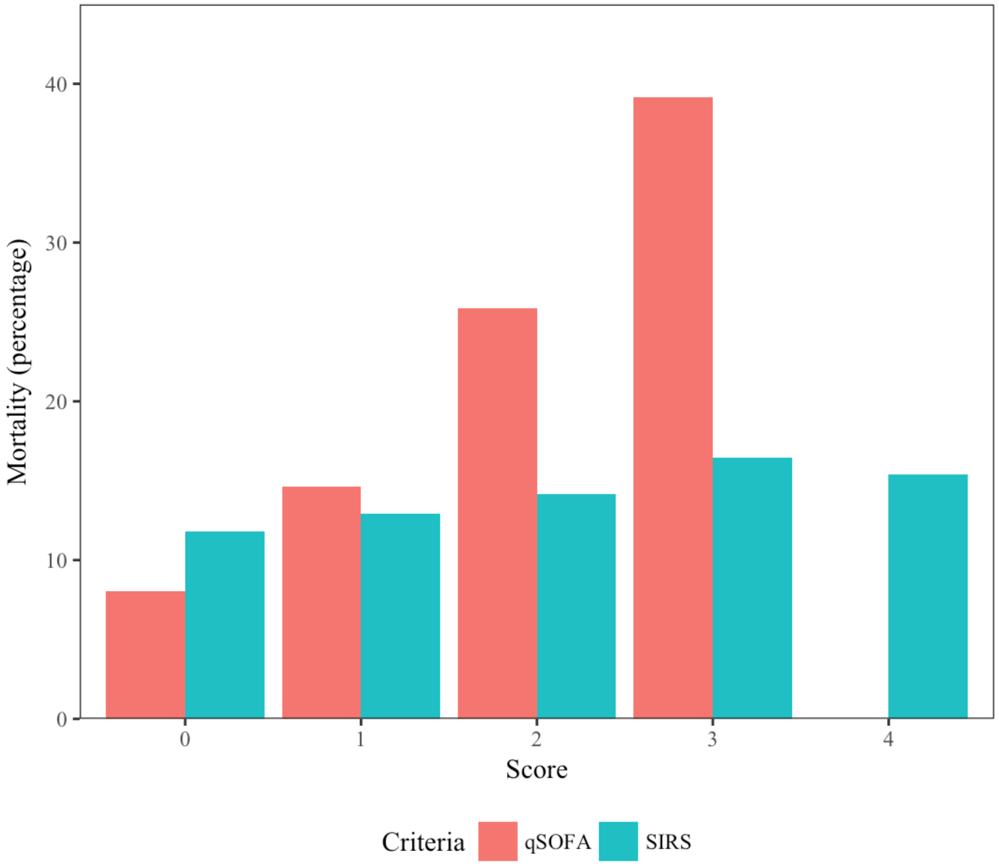


Figure 6.3: Distribution of mortality across qSOFA and SIRS scores

2.3 Comparison of Prognostic Accuracy of qSOFA and SIRS using Modeling Approach

Table 6.5 shows the performance measures of different modeling techniques on two set of variables: baseline + SIRS variables and baseline + qSOFA variables. Considering either sensitivity or specificity a model selecting criteria is a debatable subject in the medical domain. Based on sensitivity, logistic regression and nave Bayes performed well. Based on specificity, decision tree showed the best result. Figure 6.5 shows the receiver operating characteristic curve using logistic regression for baseline risk, baseline + qSOFA, and baseline + SIRS. The qSOFA criteria demonstrate better discrimination power than SIRS (baseline (AUROC = 0.63; 95% CI = 0.61, 0.63), baseline + SIRS (AUROC = 0.64; 95% CI = 0.64 to

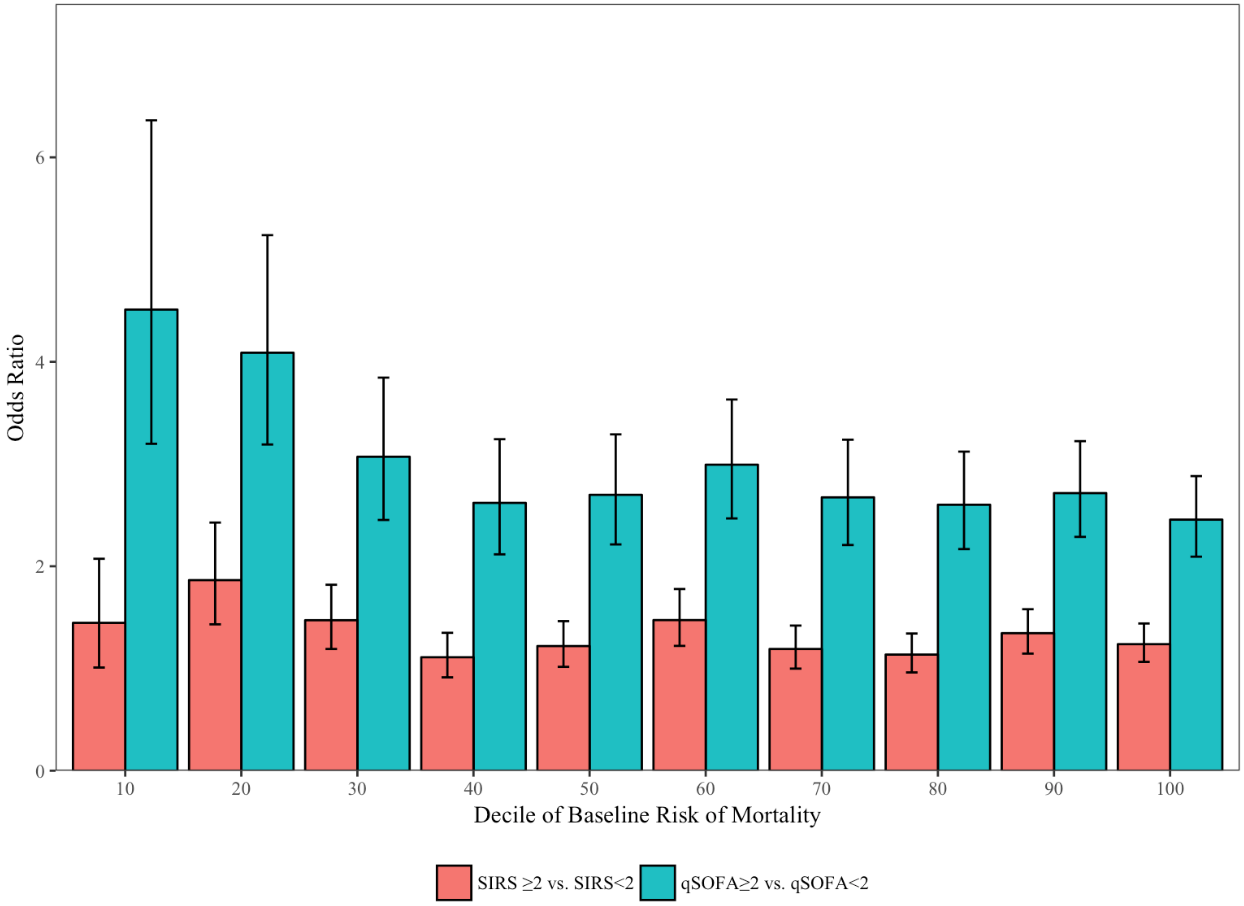


Figure 6.4: OR for in-hospital mortality for each decile of baseline risk.

0.65), and baseline + qSOFA (AUROC = 0.70; 95% CI = 0.69, 0.70)).

Table 6.5: Performance measures of SIRS and qSOFA (10 fold cross validation)

Performance	Baseline + SIRS			Baseline + qSOFA		
metrics	DT	LR	NB	DT	LR	NB
AUROC	0.622	0.643	0.653	0.612	0.696	0.696
95% CI	(0.615 0.630)	(0.637 0.652)	(0.635 0.650)	(0.605 0.618)	(0.689 0.703)	(0.688 0.703)
Sensitivity	0.459	0.617	0.623	0.394	0.642	0.616
Specificity	0.703	0.585	0.580	0.819	0.634	0.664
Positive Predictive value	0.203	0.196	0.195	0.263	0.223	0.230
Negative predictive value	0.891	0.903	0.903	0.892	0.915	0.914
Accuracy	0.669	0.590	0.586	0.759	0.631	0.657

This study compared the prognostic power of SIRS and qSOFA. The results obtained using OR and modeling approaches indicated that qSOFA was more accurate than SIRS for assessing the risk of mortality among patients at the ED. In addition, the results showed that

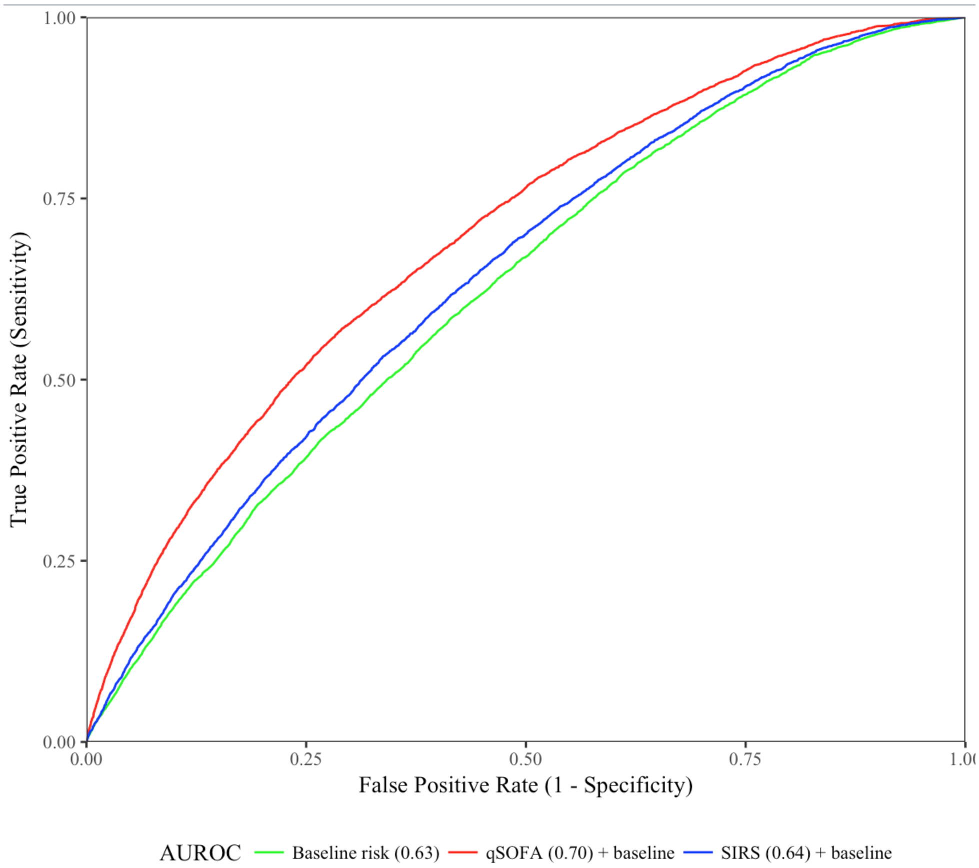


Figure 6.5: Receiver operating characteristic curve (using logistic regression).

qSOFA criteria provide better balance between sensitivity and specificity.

The individual variable analysis of SIRS and qSOFA revealed interesting results (Table 6.2). We found that body temperature was not directly associated with 28-day in-hospital mortality. This may explain the poor performance of SIRS criteria and slight dip in mortality between scores of 3 and 4 on the SIRS. A close inspection of individual SIRS criteria showed that all variables except respiratory rate were weakly associated with in-hospital mortality. However, among qSOFA criteria, all variables were strongly associated with in-hospital mortality, with the Glasgow Coma Scale having the strongest association. The trends of mortality showed a steep increase in mortality with qSOFA scores, while steady change in mortality with SIRS score. The increase of each unit of qSOFA score provides more information about the mortality risk than the unit change in SIRS score. We also investigated the possibility of either qSOFA or SIRS performing well to a group with population of specific characteristics. The investigation revealed that the association of qSOFA with mortality is greater than association of mortality with SIRS across all groups of different initial risk of mortality.

Further investigation found that although qSOFA ≥ 2 has stronger association with mortality than SIRS ≥ 2 for predicting in-hospital mortality among ED patients, it had lower sensitivity. Due to the poor sensitivity of qSOFA, patients with sepsis might remain undiagnosed. This misdiagnosis in the early stage could lead to life-threatening outcomes as timely treatment is critical [70]. However, the high sensitivity of SIRS could lead to unnecessary burdening of ICUs due to improper referrals. Researchers differ in their preference between sensitivity and specificity while selecting diagnostic criteria. Freund et al. (2017) argued that the high specificity of qSOFA criteria make it suitable to replace SIRS for efficient stratification of sepsis patients in the ED [71]. On the other hand, Akim et al. (2017) preferred sensitivity and presented results against the use of qSOFA [72]. Apart from the contradictory nature of qSOFA and SIRS about sensitivity and specificity, other performance characteristics such as positive and negative likelihood ratios showed evidence in favor of qSOFA. Later, we explain how carefully selection of threshold to determine labels from predicted probabilities, obtained from modeling approaches, can facilitate balance between sensitivity and specificity and make qSOFA more suitable for clinical use.

All three modeling approaches presented results in favor of qSOFA criteria. The area under receiver operating characteristic was greater, in most cases, for qSOFA than SIRS. The findings are aligned with work that established the foundation of qSOFA criteria [84]. Since most patients with sepsis are initially assessed in the ED [109], several other studies compared the performance of qSOFA and SIRS. Freund et al. (2017) noted the AUROC curve for predicting in-hospital mortality among ED patients was 0.80 for qSOFA and 0.65 for SIRS [71]. One of the reasons for the difference in results between this study and ours is the use of different baseline risk variables. Churpek et al. (2017) found that for non-ICU patients, the AUROC was 0.69 for qSOFA and 0.65 for SIRS, which is closely aligned with our results obtained using logistic regression [110]. The sensitivity and specificity obtained from modeling approach is more balanced for both qSOFA and SIRS than obtained directly considering qSOFA ≥ 2 and SIRS ≥ 2 (Table 6.3, 6.4). The reason for this is careful selection of threshold, instead of considering default 0.5, to determine the predicted labels from predicted probabilities. Adapting [95], we computed threshold that resulted in a point in roc plot nearest to (0, 1). From Table 6.5, it is clear that sensitivity and specificity are greater for qSOFA than SIRS for most of the modeling approaches.

Our findings support the conclusions of other studies that qSOFA criteria can better differentiate low- and high-risk patients in the ED across varying levels of baseline risk. We know that early diagnosis of sepsis is the most effective way to reduce mortality [111]. Therefore, since qSOFA does not require any laboratory results, the use of qSOFA criteria in the ED could lead to early identification of critically ill patients and, consequently, improved outcomes.

2.4 Limitations

This study has some limitations that should be considered when interpreting the results. We selected encounters using ICD-9 codes. However, these codes have been criticized in the literature due to lack of clear definitions [112, 113]. Future studies are needed that overcome this limitation by using more precise methods of identifying patients at risk for sepsis. For example, in their assessment of clinical criteria for sepsis, Seymour et al. (2016) used a combination of antibiotics and body fluid cultures occurring within a specific timeframe to define suspected infection [84].

3 Conclusions

Our findings suggest that the discrimination power for qSOFA is greater than SIRS. The baseline risk analysis showed the robustness of qSOFA. Given the continued use of SIRS to assess mortality risk, the negative association of body temperature with mortality is of particular concern as this suggests normal body temperatures could increase risk of mortality. These findings contribute to our understanding of the prognostic power of qSOFA as a rapid bedside assessment requiring no laboratory data. This study also identified the Glasgow Coma Scale as the most important variable within the qSOFA clinical variables. We also found that careful selection of threshold to translate the predicted probabilities to labels can facilitate better balance between sensitivity and specificity. These findings have important implications for the implementation and use of sepsis-related clinical scoring systems.

Chapter VII

CONCLUSIONS

In this dissertation, we applied temporal and non-temporal machine learning methods to solve clinical problems. The proposed CDSS can be easily integrated with the existing data architecture of hospitals and have the potential to reduce waste of medical resources. The additional significance of this research is that the models developed in this work can be implemented in the form of software applications, and can be easily utilized by primary care. These clinical softwares equip rural healthcare providers to manage the shortage of doctors. This research also adds knowledge to the scientific community by making python codes publicly available for future researchers to develop disease progression models for a variety of diseases. We provide synopsis of each study as follows:

Study 1: The diagnosis of sepsis is a challenging problem due to its complex physiology and symptoms similar to other diseases. Early and accurate diagnosis enables practitioners to take timely preventive actions. The current diagnostic criteria suffer from deficiencies, such as triggering false alarms or leaving conditions undiagnosed. This study aims to develop an expert system to predict the risk of sepsis using BN by identifying the optimal set of biomarkers. Using the gold standards explained in Sepsis-3 study [84], the population of suspected infection was selected from the retrospective data obtained from Cerner's Corporations. We present a four-stage framework for the predictive model: 1) *feature selection:* the combination of elastic net and recursive partitioning was used to determine the best set of biomarkers, 2) *discretization:* Hartemink method was employed, 3) *building Bayesian network*, and 4) *evaluation*.

The key difference between our approach and the existing predictive model of sepsis diagnosis in literature is that we captured the dynamics among biomarkers. We also propose *left center right* imputation method suitable for EHR that intrinsically include high percentage of missing data. The sensitivity, specificity, AUROC, and G-mean were used to compare the performance of the proposed model to SIRS, qSOFA, MEWS and SOFA. We inspected the robustness of our model by comparing its performance with competing models over different time intervals.

Using data of 16,909 patient visits, we identified blood pressure, respiratory rate, Glasgow Coma Scale score, white blood cell and creatinine as the most important biomarkers to evaluate the risk of sepsis. The most important biomarker Glasgow Coma Scale score also influences the creatinine level and respiratory rate. With an AUROC of 0.84, the proposed model outperformed the competing diagnostic criteria (SIRS = 0.59, qSOFA = 0.65, MEWS = 0.75, SOFA = 0.80). G-mean of our model was 0.75 better than SIRS = 0.55, qSOFA = 0.58 and SOFA = 0.73. The longitudinal analysis showed that the AUROC of our model is always 4-5% greater than alternative criteria over all time intervals.

We proposed an easy-to-use predictive model to asses the risk of sepsis using the important biomarkers. The richness of our proposed model is measured not only by achieving high accuracy but also by utilizing the least number of biomarkers. This model can easily be integrated into an EHR environment and autonomously identify the patient at high risk of sepsis.

Study 2: Continuous mortality monitoring is instrumental to manage a patient's care and to efficiently utilize the limited hospital resources. Due to incompleteness and irregularities of EHR, developing mortality progression using EHR data is a challenge. In this dissertation, we propose a two state hidden Markov model framework (temporal technique) to continuously monitor mortality risk by considering both the previous state of patient's health and the current value of clinical signs.

In comparison to existing non-temporal techniques to predict mortality, the proposed framework leverages the longitudinal data available in EHR. We demonstrate the performance of the proposed framework by applying it to real-world sepsis data. On the sepsis data, the AUROC of the proposed temporal framework is 9-12% greater than the non-temporal methods. The proposed framework also provides information related to the time of change of the patient's health that helps practitioners to plan early and develop effective treatment strategies.

Study 3: Disease progression models are increasingly gaining popularity in healthcare. These models can directly be integrated with the EHR and can help physicians to make informative decisions by generating insights about the status of the patient's health. One of the popular methods to develop such models is HMM. However, the time complexity to learn parameters of HMM grows with the number of stages of diseases. In this study, we propose an initialization method to learn HMM parameters efficiently. By comparing our approach to alternative methods in three dimensions (*speed*, *proximity* and *integration*), we were able to show that our approach significantly improves the performance of HMM learning algorithm.

Study 4: There exist two popular bedside sepsis diagnostic criteria: SIRS and qSOFA. It is critical to understand the pros and cons of both. In this study, we compare the performance of these two sepsis diagnostic criteria using statistical and machine learning approaches. This study examined patient visits in ED with sepsis related diagnosis. SIRS and qSOFA scores were derived from values for individual indicators for each assessment based on the first observation for each encounter. The outcome was 28-day all-cause mortality.

Using OR and modeling methods (decision tree, logistic regression and nave Bayes), the relationships between diagnostic criteria and mortality were examined. Of 132,704 eligible patient visits, 14% died within 28 days of ED admission. The association of qSOFA$\geq$ 2 with mortality (OR = 3.06; 95% CI = 2.96 to 3.17) greater than the association of SIRS $\geq$ 2 with mortality (OR = 1.22; 95% CI = 1.18 to 1.26). The AUROC for qSOFA (AUROC = 0.70) was significantly greater than for SIRS (AUROC = 0.63). The sensitivity (0.64) and specificity (0.63) for qSOFA was greater than the sensitivity (0.62) and specificity (0.59) for SIRS.

The evidences suggest that qSOFA is a better diagnostic criteria than SIRS. The low sensitivity of qSOFA can be improved by carefully selecting the threshold to translate the predicted probabilities into labels. These findings can guide healthcare providers in selecting risk-stratification measures for patients presenting to an ED with sepsis.

Bibliography

[1] Centers for Medicare and Medicaids. National Health Expenditures 2017 Highlights. Technical report, 12 2018.

[2] Centers for Medicare and Medicaid Services. National health expenditure data: Health expenditures by state of residence.

[3] United States Census Bureau. One in five americans live in rural areas. *https://www.census.gov/*.

[4] Hospital closing highlights problems of rural healthcare. https://www.al.com.

[5] Rural Health Information Hub. *https://www.ruralhealthinfo.org/states/alabama*.

[6] University of Alabama News. Uab, blue cross blue shield of alabama announce plan to tackle states rural physician shortage. *https://www.uab.edu*.

[7] Katharine E Henry, David N Hager, Peter J Pronovost, and Suchi Saria. A targeted real-time early warning score (trewscore) for septic shock. *Science Translational Medicine*, 7(299):299ra122–299ra122, 2015.

[8] Alzheimer's Association et al. 2018 alzheimer's disease facts and figures. *Alzheimer's & Dementia*, 14(3):367–429, 2018.

[9] J-L Vincent, Rui Moreno, Jukka Takala, Sheila Willatts, Arnaldo De Mendonça, Hajo Bruining, CK Reinhart, PeterM Suter, and LG Thijs. The sofa (sepsis-related organ

failure assessment) score to describe organ dysfunction/failure. *Intensive Care Medicine*, 22(7):707–710, 1996.

[10] Lawrence R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[11] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques.* MIT press, 2009.

[12] Mervyn Singer, Clifford S Deutschman, Christopher Warren Seymour, Manu Shankar-Hari, Djillali Annane, Michael Bauer, Rinaldo Bellomo, Gordon R Bernard, Jean-Daniel Chiche, Craig M Coopersmith, et al. The third international consensus definitions for sepsis and septic shock (sepsis-3). *Journal of the American Medical Informatics Association*, 315(8):801–810, 2016.

[13] N. Williams Sonja Jean Hall, Margaret and Aleksandr Golosinskiy. Inpatient care for septicemia or sepsis: A challenge for patients and hospitals. Nchs data brief, National Center for Health Statistics.

[14] Carolin Fleischmann, André Scherag, Neill KJ Adhikari, Christiane S Hartog, Thomas Tsaganos, Peter Schlattmann, Derek C Angus, and Konrad Reinhart. Assessment of global incidence and mortality of hospital-treated sepsis. current estimates and limitations. *American Journal of Respiratory and Critical Care Medicine*, 193(3):259–272, 2016.

[15] Celeste M. Torio and Brian J. Moore. *National Inpatient Hospital Costs: The Most Expensive Conditions by Payer, 2013.*

[16] Vidant Beaufort Hospital. *The Third-leading Cause of Death: Sepsis.*

[17] Vincent Liu, Gabriel J Escobar, John D Greene, Jay Soule, Alan Whippy, Derek C Angus, and Theodore J Iwashyna. Hospital deaths in patients with sepsis from 2 independent cohorts. *Jama*, 312(1):90–92, 2014.

[18] Kirsi-Maija Kaukonen, Michael Bailey, David Pilcher, D Jamie Cooper, and Rinaldo Bellomo. Systemic inflammatory response syndrome criteria in defining severe sepsis. *New England Journal of Medicine*, 372(17):1629–1638, 2015.

[19] Maia Dorsett, Melissa Kroll, Clark S Smith, Phillip Asaro, Stephen Y Liang, and Hawnwan P Moy. qsofa has poor sensitivity for prehospital identification of severe sepsis and septic shock. *Prehospital Emergency Care*, pages 1–9, 2017.

[20] Nesrin O Ghanem-Zoubi, Moshe Vardi, Arie Laor, Gabriel Weber, and Haim Bitterman. Assessment of disease-severity scoring systems for patients with sepsis in general internal medicine departments. *Critical Care*, 15(2):R95, 2011.

[21] Michael E Matheny, Randolph A Miller, T Alp Ikizler, Lemuel R Waitman, Joshua C Denny, Jonathan S Schildcrout, Robert S Dittus, and Josh F Peterson. Development of inpatient risk stratification models of acute kidney injury for use in electronic health records. *Medical Decision Making*, 30(6):639–650, 2010.

[22] Ying P Tabak, Xiaowu Sun, Carlos M Nunez, and Richard S Johannes. Using electronic health record data to develop inpatient mortality predictive model: Acute laboratory risk of mortality score (alarms). *Journal of the American Medical Informatics Association*, 21(3):455–463, 2013.

[23] Benjamin A Goldstein, Ann Marie Navar, Michael J Pencina, and John Ioannidis. Opportunities and challenges in developing risk prediction models with electronic health records data: a systematic review. *Journal of the American Medical Informatics Association*, 24(1):198–208, 2017.

[24] Dustin Charles, Meghan Gabriel, and Michael F Furukawa. Adoption of electronic health record systems among us non-federal acute care hospitals: 2008-2012. *ONC Data Brief*, 9:1–9, 2013.

[25] Roger C Bone, Robert A Balk, Frank B Cerra, R Phillip Dellinger, Alan M Fein, William A Knaus, Roland MH Schein, and William J Sibbald. Definitions for sepsis and organ failure and guidelines for the use of innovative therapies in sepsis. *Chest*, 101(6):1644–1655, 1992.

[26] J Gardner-Thorpe, N Love, J Wrightson, S Walsh, and N Keeling. The value of modified early warning score (mews) in surgical in-patients: a prospective observational study. *The Annals of The Royal College of Surgeons of England*, 88(6):571–575, 2006.

[27] John R Saltzman, Ying P Tabak, Brian H Hyett, Xiaowu Sun, Anne C Travis, and Richard S Johannes. A simple risk score accurately predicts in-hospital mortality, length of stay, and cost in acute upper gi bleeding. *Gastrointestinal Endoscopy*, 74(6):1215–1224, 2011.

[28] Kavitha J Ramchandran, Joseph W Shega, Jamie Von Roenn, Mark Schumacher, Eytan Szmuilowicz, Alfred Rademaker, Bing Bing Weitner, Pooja D Loftus, Isabella M Chu, and Sigmund Weitzman. A predictive model to identify hospitalized cancer patients at risk for 30-day mortality based on admission criteria via the electronic medical record. *Cancer*, 119(11):2074–2080, 2013.

[29] Dominik Aronsky and Peter J Haug. Automatic identification of patients eligible for a pneumonia guideline. In *Proceedings of the AMIA Symposium*, page 12. American Medical Informatics Association, 2000.

[30] Elizabeth Burnside, Daniel Rubin, and Ross Shachter. A bayesian network for mammog-

raphy. In *Proceedings of the AMIA Symposium*, page 106. American Medical Informatics Association, 2000.

[31] Akash Gupta, Tieming Liu, Scott Shepherd, and William Paiva. Using statistical and machine learning methods to evaluate the prognostic accuracy of sirs and qsofa. *Healthcare Informatics Research*, 24(2):139–147, 2018.

[32] Alexander Stojadinovic, George E Peoples, Steven K Libutti, Leonard R y, John Eberhardt, Robin S Howard, David Gur, Eric A Elster, and Aviram Nissan. Development of a clinical decision model for thyroid nodules. *BMC Surgery*, 9(1):12, 2009.

[33] Aviram Nissan, Mladjan Protic, Anton Bilchik, John Eberhardt, George E Peoples, and Alexander Stojadinovic. Predictive model of outcome of targeted nodal assessment in colorectal cancer. *Annals of Surgery*, 251(2):265–274, 2010.

[34] Bo Pang, David Zhang, Naimin Li, and Kuanquan Wang. Computerized tongue diagnosis based on bayesian networks. *IEEE Transactions on Biomedical Engineering*, 51(10):1803–1810, 2004.

[35] Charles E Kahn Jr, Linda M Roberts, Katherine A Shaffer, and Peter Haddawy. Construction of a bayesian network for mammographic diagnosis of breast cancer. *Computers in Biology and Medicine*, 27(1):19–29, 1997.

[36] M Berkan Sesen, Ann E Nicholson, Rene Banares-Alcantara, Timor Kadir, and Michael Brady. Bayesian networks for clinical decision support in lung cancer care. *PloS one*, 8(12):e82349, 2013.

[37] Vijay Mago, Bhanu Prasad, Ajay Bhatia, and Anjali Mago. A decision making system for the treatment of dental caries. *Soft Computing Applications in Business*, pages 231–242, 2008.

[38] Dipankar Bandyopadhyay, Brian J Reich, and Elizabeth H Slate. Bayesian modeling of multivariate spatial binary data with applications to dental caries. *Statistics in Medicine*, 28(28):3492–3508, 2009.

[39] Adele H Marshall, Sally I McClean, CM Shapcott, IR Hastie, and Peter H Millard. Developing a bayesian belief network for the management of geriatric hospital care. *Health Care Management Science*, 4(1):25–30, 2001.

[40] Silvia Acid, Luis M de Campos, Juan M Fernández-Luna, Susana Rodrıguez, José Marıa Rodrıguez, and José Luis Salcedo. A comparison of learning algorithms for bayesian networks: a case study based on data from an emergency medical service. *Artificial Intelligence in Medicine*, 30(3):215–232, 2004.

[41] Linda Peelen, Nicolette F de Keizer, Evert de Jonge, Robert-Jan Bosman, Ameen Abu-Hanna, and Niels Peek. Using hierarchical dynamic bayesian networks to investigate dynamics of organ failure in patients in the intensive care unit. *Journal of biomedical informatics*, 43(2):273–286, 2010.

[42] Xiongcai Cai, Oscar Perez-Concha, Enrico Coiera, Fernando Martin-Sanchez, Richard Day, David Roffe, and Blanca Gallego. Real-time prediction of mortality, readmission, and length of stay using electronic health record data. *Journal of the American Medical Informatics Association*, 23(3):553–561, 2015.

[43] Yu-Ying Liu, Hiroshi Ishikawa, Mei Chen, Gadi Wollstein, Joel S Schuman, and James M Rehg. Longitudinal modeling of glaucoma progression using 2-dimensional continuous-time hidden markov model. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 444–451. Springer, 2013.

[44] Rafid Sukkar, Elyse Katz, Yanwei Zhang, David Raunig, and Bradley T Wyman. Disease progression modeling using hidden markov models. In *Engineering in Medicine*

and Biology Society (EMBC), 2012 Annual International Conference of the IEEE, pages 2845–2848. IEEE, 2012.

[45] Thomas Garske. *Using Deep Learning on EHR Data to Predict Diabetes.* PhD thesis, University of Colorado at Denver, 2018.

[46] Zhengping Che, David Kale, Wenzhe Li, Mohammad Taha Bahadori, and Yan Liu. Deep computational phenotyping. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 507–516. ACM, 2015.

[47] Andre Esteva, Alexandre Robicquet, Bharath Ramsundar, Volodymyr Kuleshov, Mark DePristo, Katherine Chou, Claire Cui, Greg Corrado, Sebastian Thrun, and Jeff Dean. A guide to deep learning in healthcare. *Nature medicine*, 25(1):24, 2019.

[48] Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-wei, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3:160035, 2016.

[49] *PD disease state assessment in naturalistic environments using deep learning*, 2015.

[50] Zachary C Lipton, David C Kale, Charles Elkan, and Randall Wetzel. Learning to diagnose with lstm recurrent neural networks. *arXiv preprint arXiv:1511.03677*, 2015.

[51] Leonard E Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state markov chains. *The Annals of Mathematical Statistics*, 37(6):1554–1563, 1966.

[52] Leonard Baum. An inequality and associated maximization technique in statistical estimation of probabilistic functions of a markov process. *Inequalities*, 3:1–8, 1972.

[53] Yariv Ephraim and Neri Merhav. Hidden markov processes. *IEEE Transactions on Information Theory*, 48(6):1518–1569, 2002.

[54] Srinivasan Vairavan, Larry Eshelman, Syed Haider, Abigail Flower, and Adam Seiver. Prediction of mortality in an intensive care unit using logistic regression and a hidden markov model. In *Computing in Cardiology (CinC), 2012*, pages 393–396. IEEE, 2012.

[55] Yuanxi Li, Stephen Swift, and Allan Tucker. Modelling and analysing the dynamics of disease progression from cross-sectional studies. *Journal of Biomedical Informatics*, 46(2):266–274, 2013.

[56] Ying Chen and Tuan D Pham. Development of a brain mri-based hidden markov model for dementia recognition. *Biomedical Engineering Online*, 12(1):S2, 2013.

[57] Nicola Bartolomeo, Paolo Trerotoli, and Gabriella Serio. Progression of liver cirrhosis to hcc: an application of hidden markov model. *BMC Medical Research Methodology*, 11(1):38, 2011.

[58] Ioan Stanculescu, Christopher KI Williams, and Yvonne Freer. Autoregressive hidden markov models for the early detection of neonatal sepsis. *IEEE journal of Biomedical and Health Informatics*, 18(5):1560–1570, 2014.

[59] Abed Khorasani and Mohammad Reza Daliri. Hmm for classification of parkinsons disease based on the raw gait data. *Journal of Medical Systems*, 38(12):147, 2014.

[60] Padhraic Smyth. Clustering sequences with hidden markov models. In *Advances in Neural Information Processing Systems*, pages 648–654, 1997.

[61] Álvaro Castellanos-Ortega, Borja Suberviola, Luis A García-Astudillo, María S Holanda, Fernando Ortiz, Javier Llorca, and Miguel Delgado-Rodríguez. Impact of the surviving sepsis campaign protocols on hospital length of stay and mortality in septic shock

patients: results of a three-year follow-up quasi-experimental study. *Critical Care Medicine*, 38(4):1036–1043, 2010.

[62] Roger C Bone, William J Sibbald, and Charles L Sprung. The accp-sccm consensus conference on sepsis and organ failure. *Chest*, 101(6):1481–1484, 1992.

[63] Nathan Shapiro, Michael D Howell, David W Bates, Derek C Angus, Long Ngo, and Daniel Talmor. The association of sepsis syndrome and organ dysfunction with mortality in emergency department patients with suspected infection. *Annals of Emergency Medicine*, 48(5):583–590, 2006.

[64] Jean-Louis Vincent. Dear sirs, i'm sorry to say that i don't like you. *Critical Care Medicine*, 25(2):372–374, 1997.

[65] Edward Abraham, Michael A Matthay, Charles A Dinarello, Jean-Louis Vincent, Jonathan Cohen, Steven M Opal, Michel Glauser, Polly Parsons, Charles J Fisher Jr, and John E Repine. Consensus conference definitions for sepsis, septic shock, acute lung injury, and acute respiratory distress syndrome: time for a reevaluation. *Critical Care Medicine*, 28(1):232–235, 2000.

[66] Mitchell M Levy, Mitchell P Fink, John C Marshall, Edward Abraham, Derek Angus, Deborah Cook, Jonathan Cohen, Steven M Opal, Jean-Louis Vincent, Graham Ramsay, et al. 2001 sccm/esicm/accp/ats/sis international sepsis definitions conference. *Intensive Care Medicine*, 29(4):530–538, 2003.

[67] Michael M Liao, Dennis Lezotte, Steven R Lowenstein, Kevin Howard, Zachary Finley, Zipei Feng, Richard L Byyny, Jeffrey D Sankoff, Ivor S Douglas, and Jason S Haukoos. Sensitivity of systemic inflammatory response syndrome for critical illness among ed patients. *The American Journal of Emergency Medicine*, 32(11):1319–1325, 2014.

[68] Jean-Louis Vincent, Steven M Opal, John C Marshall, and Kevin J Tracey. Sepsis definitions: time for change. *Lancet (London, England)*, 381(9868):774, 2013.

[69] Eli J Finkelsztein, Daniel S Jones, Kevin C Ma, Maria A Pabón, Tatiana Delgado, Kiichi Nakahira, John E Arbo, David A Berlin, Edward J Schenck, Augustine MK Choi, et al. Comparison of qsofa and sirs for predicting adverse outcomes of patients with suspicion of sepsis outside the intensive care unit. *Critical Care*, 21(1):73, 2017.

[70] Jessica L Nelson, Barbara L Smith, Jeremy D Jared, and John G Younger. Prospective trial of real-time electronic surveillance to expedite early care of severe sepsis. *Annals of Emergency Medicine*, 57(5):500–504, 2011.

[71] Yonathan Freund, Najla Lemachatti, Evguenia Krastinova, Marie Van Laer, Yann-Erick Claessens, Aurélie Avondo, Céline Occelli, Anne-Laure Feral-Pierssens, Jennifer Truchot, Mar Ortega, et al. Prognostic accuracy of sepsis-3 criteria for in-hospital mortality among patients with suspected infection presenting to the emergency department. *Journal of the American Medical Informatics Association*, 317(3):301–308, 2017.

[72] Åsa Askim, Florentin Moser, Lise T Gustad, Helga Stene, Maren Gundersen, Bjørn Olav Åsvold, Jostein Dale, Lars Petter Bjørnsen, Jan Kristian Damås, and Erik Solligård. Poor performance of quick-sofa (qsofa) score in predicting severe sepsis and mortality–a prospective study of patients admitted with infection to the emergency department. *Scandinavian Journal of Trauma, Resuscitation and Emergency Medicine*, 25(1):56, 2017.

[73] Janos L Mathe, Jason B Martin, Peter Miller, Akos Ledeczi, Liza M Weavind, Andras Nadas, Anne Miller, David J Maron, and Janos Sztipanovits. A model-integrated, guideline-driven, clinical decision-support system. *IEEE Software*, 26(4), 2009.

[74] Bristol N Brandt, Amanda B Gartner, Michael Moncure, Chad M Cannon, Elizabeth Carlton, Carol Cleek, Chris Wittkopp, and Steven Q Simpson. Identifying severe sepsis via electronic surveillance. *American Journal of Medical Quality*, 30(6):559–565, 2015.

[75] Robert C Amland and Kristin E Hahn-Cover. Clinical decision support for early recognition of sepsis. *American Journal of Medical Quality*, 31(2):103–110, 2016.

[76] Roman A Lukaszewski, Adam M Yates, Matthew C Jackson, Kevin Swingler, John M Scherer, AJ Simpson, Paul Sadler, Peter McQuillan, Richard W Titball, Timothy JG Brooks, et al. Presymptomatic prediction of sepsis in intensive care unit patients. *Clin. Vaccine Immunol.*, 15(7):1089–1094, 2008.

[77] Subramani Mani, Asli Ozdas, Constantin Aliferis, Huseyin Atakan Varol, Qingxia Chen, Randy Carnevale, Yukun Chen, Joann Romano-Keeler, Hui Nian, and Jörn-Hendrik Weitkamp. Medical decision support using machine learning for early detection of late-onset neonatal sepsis. *Journal of the American Medical Informatics Association*, 21(2):326–336, 2014.

[78] Eren Gultepe, Jeffrey P Green, Hien Nguyen, Jason Adams, Timothy Albertson, and Ilias Tagkopoulos. From vital signs to clinical outcomes for patients with sepsis: a machine learning basis for a clinical decision support system. *Journal of the American Medical Informatics Association*, pages 315–325, 2014.

[79] Jean A Nemzek, Andrew P Hodges, and Yongqun He. Bayesian network analysis of multi-compartmentalized immune responses in a murine model of sepsis and direct lung injury. *BMC research notes*, 8(1):516, 2015.

[80] Micol Sandri, Paola Berchialla, Ileana Baldi, Dario Gregori, and Roberto Alberto De Blasi. Dynamic bayesian networks to predict sequences of organ failures in patients admitted to icu. *Journal of Biomedical Informatics*, 48:106–113, 2014.

[81] Colin K Grissom, Samuel M Brown, Kathryn G Kuttler, Jonathan P Boltax, Jason Jones, Al R Jephson, and James F Orme. A modified sequential organ failure assessment score for critical care triage. *Disaster Medicine and Public Health Preparedness*, 4(4):277–284, 2010.

[82] Alexander John Hartemink. *Principled computational methods for the validation discovery of genetic regulatory networks*. PhD thesis, Massachusetts Institute of Technology, 2001.

[83] Jonathan P DeShazo and Mark A Hoffman. A comparison of a multistate inpatient ehr database to the hcup nationwide inpatient sample. *BMC Health Services Research*, 15(1):384, 2015.

[84] Christopher W Seymour, Vincent X Liu, Theodore J Iwashyna, Frank M Brunkhorst, Thomas D Rea, André Scherag, Gordon Rubenfeld, Jeremy M Kahn, Manu Shankar-Hari, Mervyn Singer, et al. Assessment of clinical criteria for sepsis: for the third international consensus definitions for sepsis and septic shock (sepsis-3). *Journal of the American Medical Informatics Association*, 315(8):762–774, 2016.

[85] Muge Capan, Stephen Hoover, Julie S Ivy, Kristen E Miller, Ryan Arnold, et al. Not all organ dysfunctions are created equal–prevalence and mortality in sepsis. *Journal of Critical Care*, 2018.

[86] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.

[87] Elizabeth J Atkinson and Terry M Therneau. An introduction to recursive partitioning using the rpart routines. *Rochester: Mayo Foundation*, 2000.

[88] Terry M Therneau, Elizabeth J Atkinson, et al. An introduction to recursive partitioning using the rpart routines, 1997.

[89] Laura Elena Raileanu and Kilian Stoffel. Theoretical comparison between the gini index and information gain criteria. *Annals of Mathematics and Artificial Intelligence*, 41(1):77–93, 2004.

[90] RM Sakia. The box-cox transformation technique: a review. *The Statistician*, pages 169–178, 1992.

[91] C Chow and Cong Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968.

[92] CP Subbe, M Kruger, P Rutherford, and L Gemmel. Validation of a modified early warning score in medical admissions. *QJM*, 94(10):521–526, 2001.

[93] Thomas Desautels, Jacob Calvert, Jana Hoffman, Melissa Jay, Yaniv Kerem, Lisa Shieh, David Shimabukuro, Uli Chettipally, Mitchell D Feldman, Chris Barton, et al. Prediction of sepsis in the intensive care unit with minimal electronic health record data: a machine learning approach. *JMIR Medical Informatics*, 4(3), 2016.

[94] Kazim Topuz, Hasmet Uner, Asil Oztekin, and Mehmet Bayram Yildirim. Predicting pediatric clinic no-shows: a decision analytic framework using elastic net and bayesian belief network. *Annals of Operations Research*, pages 1–21, 2017.

[95] Neil J Perkins and Enrique F Schisterman. The inconsistency of optimal cutpoints obtained using two criteria based on the receiver operating characteristic curve. *American Journal of Epidemiology*, 163(7):670–675, 2006.

[96] Anthony K Akobeng. Understanding diagnostic tests 3: receiver operating characteristic curves. *Acta Paediatrica*, 96(5):644–647, 2007.

[97] Harini Padmanaban. Comparative analysis of naive bayes and tree augmented naive bayes models. 2014.

[98] Thomas P Minka. A comparison of numerical optimizers for logistic regression. *Unpublished draft*, pages 1–18, 2003.

[99] Andrea Vattani. The hardness of k-means clustering in the plane. *Manuscript, accessible at http://cseweb. ucsd. edu/avattani/papers/kmeans_hardness. pdf*, 617, 2009.

[100] SP Lloyd. Least square quantization in pcm. bell telephone laboratories paper. published in journal much later: Lloyd, sp: Least squares quantization in pcm. *IEEE Trans. Inform. Theor.(1957/1982)*, 18, 1957.

[101] Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, and Angela Y Wu. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (7):881–892, 2002.

[102] https://github.com/ananthpn/pyhmm - accessed on 4/25/2019.

[103] Adam Lipowski and Dorota Lipowska. Roulette-wheel selection via stochastic acceptance. *Physica A: Statistical Mechanics and its Applications*, 391(6):2193–2196, 2012.

[104] Juergen Jung. Estimating markov transition probabilities between health states in the hrs dataset. *Indiana University*, pages 1–42, 2006.

[105] Eamon P Raith, Andrew A Udy, Michael Bailey, Steven McGloughlin, Christopher MacIsaac, Rinaldo Bellomo, and David V Pilcher. Prognostic accuracy of the sofa score, sirs criteria, and qsofa score for in-hospital mortality among adults with suspected infection admitted to the intensive care unit. *JAMA*, 317(3):290–300, 2017.

[106] Saeed Piri, Dursun Delen, Tieming Liu, and Hamed M Zolbanin. A data analytics approach to building a clinical decision support system for diabetic retinopathy: Developing and deploying a model ensemble. *Decision Support Systems*, 101:12–27, 2017.

[107] Kevin P Murphy. Naive bayes classifiers. *University of British Columbia*, 18, 2006.

[108] Robert C Amland and Bharat B Sutariya. Quick sequential [sepsis-related] organ failure assessment (qsofa) and st. john sepsis surveillance agent to detect patients at risk of sepsis: An observational cohort study. *American Journal of Medical Quality*, 33(1):50–57, 2018.

[109] Stephen PJ Macdonald, Glenn Arendts, Daniel M Fatovich, and Simon GA Brown. Comparison of piro, sofa, and meds scores for predicting mortality in emergency department patients with severe sepsis and septic shock. *Academic Emergency Medicine*, 21(11):1257–1263, 2014.

[110] Matthew M Churpek, Ashley Snyder, Xuan Han, Sarah Sokol, Natasha Pettit, Michael D Howell, and Dana P Edelson. Quick sepsis-related organ failure assessment, systemic inflammatory response syndrome, and early warning scores for detecting clinical deterioration in infected patients outside the intensive care unit. *American Journal of Respiratory and Critical Care Medicine*, 195(7):906–911, 2017.

[111] Emanuel Rivers, Bryant Nguyen, Suzanne Havstad, Julie Ressler, Alexandria Muzzin, Bernhard Knoblich, Edward Peterson, and Michael Tomlanovich. Early goal-directed therapy in the treatment of severe sepsis and septic shock. *New England Journal of Medicine*, 345(19):1368–1377, 2001.

[112] Daniel A Ollendorf, A Mark Fendrick, Karen Massey, G Rhys Williams, and Gerry Oster. Is sepsis accurately coded on hospital bills? *Value in Health*, 5(2):79–81, 2002.

[113] Chongthawonsatid Sukanya. Validity of principal diagnoses in discharge summaries and icd-10 coding assessments based on national health data of thailand. *Healthcare Informatics Research*, 23(4):293–303, 2017.

Appendix A

Appendix

Let $\vec{Z_m}$ be the m^{th} sequence of observation

$$\vec{Z}_m = [\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^t, \ldots \mathbf{X}^T]$$

where $\mathbf{X}^t$ is a vector of observed features at time t, i.e., $\mathbf{X}^t = [x_1^t, x_2^t, \ldots, x_N^t]$

The objective is to estimate parameters by maximizing the log-likelihood of the observed sequences. The likelihood function is given by Equation A.1

$$logL = \sum_{m=1}^{M} logP(\vec{Z_m}|\lambda) \tag{A.1}$$

where λ is a set of HMM parameters (transition and emission probabilities). The estimation of $P(\vec{Z}_m|\lambda)$ is computationally expensive and requires enumeration of all possible sequence of states i.e. $O(S^T)$ operations, where S is the number of hidden states and T is the number of time intervals. However, there exists an alternate method where instead of computing $P(\vec{Z_m}|\lambda)$, we use $P(\vec{Z_m}, \vec{s}|\lambda)$ that is tractable using *Baum-Welch* algorithm. $\vec{s}$ is a sequence of hidden states.

Using Bayesian formula, Equation A.1 can be written as:

$$\begin{aligned} logP(\vec{Z_m}|\lambda) =& log\frac{P(\vec{Z_m},\vec{s}|\lambda)}{P(\vec{s}|\vec{Z_m}\lambda)} \\ =& log\frac{P(\vec{Z_m},\vec{s}|\lambda)}{Q(\vec{s})} - log\frac{P(\vec{s}|\vec{Z_m},\lambda)}{Q(\vec{s})} \end{aligned} \tag{A.2}$$

$Q(\vec{s})$ is a probability mass function. Later, we explain how to define this distribution.

$$\begin{aligned} Q(\vec{s})logP(\vec{Z_m}|\lambda) =& Q(\vec{s})log\frac{P(\vec{Z_m},\vec{s}|\lambda)}{Q(\vec{s})} - Q(\vec{s})log\frac{P(\vec{s}|\vec{Z_m},\lambda)}{Q(\vec{s})} \\ \sum_{\vec{s}} Q(\vec{s})logP(\vec{Z_m}|\lambda) =& \sum_{\vec{s}} Q(\vec{s})log\frac{P(\vec{Z_m},\vec{s}|\lambda)}{Q(\vec{s})} - \sum_{\vec{s}} Q(\vec{s})log\frac{P(\vec{s}|\vec{Z_m},\lambda)}{Q(\vec{s})} \end{aligned}$$

Since $Q(\vec{s})$ is a mass function, therefore, $\sum_{\vec{s}} Q(\vec{s}) = 1$

$$logP(\vec{Z_m}|\lambda) = \sum_{\vec{s}} Q(\vec{s})log\frac{P(\vec{Z_m},\vec{s}|\lambda)}{Q(\vec{s})} - \sum_{\vec{s}} Q(\vec{s})log\frac{P(\vec{s}|\vec{Z_m},\lambda)}{Q(\vec{s})}$$

The term $-\sum_{\vec{s}} Q(\vec{s})log\frac{P(\vec{s}|\vec{Z_m},\lambda)}{Q(\vec{s})}$ is always greater than zero. Therefore,

$$logP(\vec{Z_m}|\lambda) \geq \sum_{\vec{s}} Q(\vec{s})log\frac{P(\vec{Z_m},\vec{s}|\lambda)}{Q(\vec{s})}$$

Hence, instead of maximizing $logP(\vec{Z_m}|\lambda)$, the lower bound of $logP(\vec{Z_m},\vec{s}|\lambda)$ is maximized to estimate HMM parameters. To provide the tighter bound, we select $Q(\vec{s})$ such that $log\frac{P(\vec{s}|\vec{Z_m},\lambda)}{Q(\vec{s})}$ becomes zero (i.e., $Q(\vec{s}) = P(\vec{s}|\vec{Z_m},\lambda)$).

Appendix B

Appendix

```
import json
import os
import sys
import numpy as np
# defining a class to initialize HMM with multiple features
class hmm_msmv(object):
def __init__(self, fname, nVar):
if fname == None:
print "Fatal␣Error:␣You␣should␣provide␣the␣model␣file␣name"
sys.exit()
self.nVar = nVar  #number of features
self.parameters = json.loads(open(fname).read())
# reading parameter file with initial parameters;
# the file includes transition matrix, observation matrix and \
prior hidde state distribution
self.A = self.parameters["A"] # transistion matrix
self.states = self.A.keys() # get the list of states
self.nStates = len(self.states) # number of states of the model
self.B = [] #set emission matrix
self.LevelsVar = []
self.nLevelsVar = []
for i in range(self.nVar):
self.B.append(self.parameters["B" + str(i)])
#each features are stored in parameter file as B1, B2, so on
self.LevelsVar.append(self.B[i].values()[0].keys())
self.nLevelsVar.append(len(self.B[i].values()[0].keys()))
self.pi = self.parameters["pi"] # prior distribution

def forward(self, obs):
# this part implements Algorithm 1
self.fwd = [{}]
for y in self.states:
self.fwd[0][y] = self.pi[y]
for nVar_i in range(self.nVar):
self.fwd[0][y] = self.fwd[0][y] * \
self.B[nVar_i][y][obs[0][nVar_i]]
for t in range(1, len(obs)):
self.fwd.append({})
for y in self.states:
temp = 1
for nVar_i in range(self.nVar):
temp = self.B[nVar_i][y][obs[t][nVar_i]] * temp
```

```
self.fwd[t][y] = sum((self.fwd[t-1][y0] * self.A[y0][y] \
* temp) for y0 in self.states)
# refer to parameter learning section of this chapter\
(forward variable)
prob = sum((self.fwd[len(obs) - 1][s]) for s in self.states)
# return probability of observing the sequence
return prob
def backward(self, obs):
# this part implements Algorithm 2
self.bwk = [{} for t in range(len(obs))]
T = len(obs)
for y in self.states:
self.bwk[T - 1][y] = 1
for t in reversed(range(T - 1)):
for y0 in self.states:
sum = 0
for y1 in self.states:
temp = 1
for nVar_i in range(self.nVar):
temp = self.B[nVar_i][y1][obs[t + 1][nVar_i]]*temp
sum = self.bwk[t + 1][y1]*self.A[y0][y1]*temp + sum
self.bwk[t][y0] = sum

prob = sum((self.pi[y] * self.B[y][obs[0]] * self.bwk[0][y])\
for y in self.states)
return prob
```

```
def parameterLearning(self, obsSeqMultiSub):
#this part implements Alogrithm 3
gamma = [{} for t in range(len(obsSeqMultiSub[0]))]
zi = [{} for t in range(len(obsSeqMultiSub[0]))]
for t in range(len(obsSeqMultiSub[0]) - 1):
for s in self.states:
zi[t][s] = {}
for s1 in self.states:
zi[t][s][s1] = 0

for t in range(len(obsSeqMultiSub[0])):
for s in self.states:
gamma[t][s] = 0



num = [{} for i in range(len(obsSeqMultiSub))]
denom = [{} for i in range(len(obsSeqMultiSub))]
numGamma = [[{} for nVar_i in range(self.nVar)] \
for i in range(len(obsSeqMultiSub))]
denomGamma = [[{} for nVar_i in range(self.nVar)] \
for i in range(len(obsSeqMultiSub))]

for j in range(len(obsSeqMultiSub)):  # number of data points
p_obs = self.forward(obsSeqMultiSub[j])
self.backward(obsSeqMultiSub[j])
```

```
for t in range(len(obsSeqMultiSub[0])):
for y in self.states:
gamma[t][y] = (self.fwd[t][y] * self.bwk[t][y]) / p_obs
if t == 0:
self.pi[y] = gamma[t][y]
if t == len(obsSeqMultiSub[j]) - 1:
continue

for y1 in self.states:
temp = 1
for nVar_i in range(self.nVar):
temp = self.B[nVar_i][y1][obsSeqMultiSub[j][t + 1]\
[nVar_i]]*temp
zi[t][y][y1] = (self.fwd[t][y] * self.A[y][y1] * \
temp * self.bwk[t + 1][y1])/ p_obs

for y in self.states:
num[j][y] = {}
denom[j][y] = {}
for y1 in self.states:
num[j][y][y1] = sum([zi[t][y][y1] \
for t in range(len(obsSeqMultiSub[0]) - 1)])
denom[j][y][y1] = sum([gamma[t][y] \
for t in range(len(obsSeqMultiSub[0]) - 1)])

for nVar_i in range(self.nVar):
```

```
for s in self.states:
numGamma[j][nVar_i][s] = {}
denomGamma[j][nVar_i][s] = {}
for i_varLevel in self.LevelsVar[nVar_i]:
numGamma[j][nVar_i][s][i_varLevel ] = 0.0
for t in range(len(obsSeqMultiSub[0])):
if obsSeqMultiSub[j][t][nVar_i] == i_varLevel:
numGamma[j][nVar_i][s][i_varLevel ] += \
gamma[t][s]
denomGamma[j][nVar_i][s][i_varLevel] =\
sum([gamma[t][s] for t in range(\
len(obsSeqMultiSub[0]))])
for s in self.states:
for s1 in self.states:
val = sum([num[k][s][s1] \
for k in range(len(obsSeqMultiSub))])
val /= sum([denom[k][s][s1] \
for k in range(len(obsSeqMultiSub))])
self.A[s][s1] = val
for nVar_i in range(self.nVar):
for s in self.states:
for i_varLevel in self.LevelsVar[nVar_i]:
val = sum([numGamma[k][nVar_i][s][i_varLevel] \
for k in range(len(obsSeqMultiSub))])
val /= sum([denomGamma[k][nVar_i][s][i_varLevel] \
for k in range(len(obsSeqMultiSub))])
```

```
self.B[nVar_i][s][i_varLevel] = val
```

VITA

Akash Gupta

Candidate for the Degree of:

Doctor of Philosophy

Dissertation: DEVELOPING CLINICAL DECISION SUPPORT SYSTEMS FOR SEPSIS PREDICTION USING TEMPORAL AND NON-TEMPORAL MACHINE LEARNING METHODS

Major Field: Industrial Engineering & Management

Biographical:

Education:

Completed the requirements for Doctor of Philosophy in Industrial Engineering & Management at Oklahoma State University, Stillwater Oklahoma in July, 2019.

Completed the requirements for Master of Technology in Control Systems at Indian Institute of Technology (BHU) - Varanasi, Varanasi India in 2012.

Completed the requirements for Bachelor of Technology in Computer Engineering at Uttar Pradesh Technical University, Lucknow India in 2009.