A STUDY OF ADMINISTRATIVE DATA REPRESENTATION FOR MACHINE LEARNING

by

Negin Asadzadehzanjani

A Dissertation
Submitted to the
Graduate Faculty
of
George Mason University
in Partial Fulfillment of
The Requirements for the Degree
of
Doctor of Philosophy
Health Services Research

Committee:

Dr. Janusz Wojtusiak, Chair
Associate Professor, Program Director in Health Informatics, GMU

Dr. Sanja Avramovic, Member
Assistant Professor, Department of Health Administration and Policy, GMU

Dr. Özlem Uzuner, Member
Associate Professor, Department of Information Sciences and Technology, GMU

Dr. Y. Alicia Hong, Acting PhD Program Director
Department of Health Administration and Policy

Dr. P.J. Maddox, Chair
Department of Health Administration and Policy

Dr. Germaine M. Louis, Dean
College of Health and Human Services

Date: 2/25/2022

Spring Semester 2022
George Mason University, Fairfax, VA

A Study of Administrative Data Representation for Machine Learning

A Dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at George Mason University

by

Negin Asadzadehzanjani
Master of Science
University of Tehran, 2016
Bachelor of Science
Imam Khomeini International University, 2013

Director: Janusz Wojtusiak, Associate Professor
Department of Health Administration and Policy

Spring Semester 2022
George Mason University
Fairfax, VA

Copyright 2022 Negin Asadzadehzanjani
All Rights Reserved

DEDICATION

To my beloved family.

ACKNOWLEDGEMENTS

First and foremost, I'd like to express my sincere appreciation and gratitude to Prof. Janusz Wojtusiak, my doctoral adviser, for his unwavering support during my Ph.D. studies and research. Prof. Wojtusiak gave me the chance to start my Ph.D. in HAP department and always supported and encouraged me to pursue my research interests. His broad understanding of the field and extensive experience provided invaluable guidance, and I've enjoyed numerous helpful and thought-provoking conversations with him. Most importantly, he has always been patient with and accessible to me and has shown confidence in my ability, even when I wasn't confident in myself. Without his encouragement and assistance, this dissertation would not have been feasible.

I am thankful to my committee members. I want to express my gratitude to Prof. Sanja Avramovic and Prof. Özlem Uzuner for agreeing to serve on my thesis committee and providing precious and insightful feedback to improve this dissertation. Moreover, I have enjoyed working on a couple of projects with Prof. Avramovic in my first year in HAP that provided me with a solid foundation for the rest of my Ph.D. studies.

I was fortunate to receive guidance from several other faculty members at GMU. I am grateful to Prof. Farrokh Alemi for his input and fruitful discussions. I had an opportunity to collaborate with Prof. Alemi for the Functional Abilities project, and his insights have helped me understand the nuances of interdisciplinary research. I would also like to thank Prof. Panagiota Kitsantas for her support and for giving me the opportunity to work with her and to explore other close areas in the Health Services Research field. I am also very much blessed and grateful to be taught by the excellent professors in HAP.

Additionally, I would like to thank Prof. P.J. Maddox, chair of the Health Administration and Policy department and Prof. Wojtusiak for financially supporting my research studies in the past five years. I am also grateful to the provost office at GMU for their generosity in providing me with two summer fellowships, which allowed me to focus on my research without any distractions.

In addition, I would like to express my gratitude to Tracy and Regina for their continued support and assistance in HAP, which allowed me to focus entirely on research without worrying about administrative work.

I want to thank my lab mates in the MLI laboratory and other graduate students in the HAP department for creating such a great environment and atmosphere to study together and learn from each other. My special thanks go to Hedyeh and Reyhaneh, my fantastic friends and lab mates. I gratefully acknowledge their support, generosity with their time, and helpful advice in this journey. Hedyeh is not only a great friend but an awesome system administrator and has always been there for help. I also learned a lot from my friends Eman,

Fatemah, Kerry, Mary Lou, Elina, and Dina, and Atefeh who shaped my research in many ways.

Finally, I wish to express my deepest gratitude to my parents, Marzieh and Hassan, who have always believed in me and have sacrificed more than anyone could ever imagine making my dream a reality. I will forever remain in their debt for all they have done for me. I could never find enough words to express my gratitude to my husband, Aria. During my graduate program, the best thing that happened was that I met Aria. He has been my best friend, has supported me in my lows, and cheered for me in my highs. I am grateful for his love and companionship. A big thank you goes out to my sister Nazanin, who provided me with invaluable mental support. In addition, I would like to thank my sister-in-law Bahar and my brother-in-law Arsalan for their unending support and encouragement.

I apologize to those I have inadvertently forgotten.

I hope that all the ups and downs experienced in graduate school will make me a better person.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

Activity of Daily Living ..... ADL
Acute Physiologic and Chronic Health Evaluation ..... APACHE
Alzheimer's Association Interactive Network ..... GAAIN
Area Under Receiver-Operator Curve ..... AUC
Area Under Precision-Recall Curve ..... AUPCR
Artificial Intelligence ..... AI
Artificial Neural Network ..... ANN
Centers for Medicare and Medicaid Services ..... CMS
Chronic Condition Indicator CCI
Chronic Kidney disease ..... CKD
Chronic Obstructive Pulmonary Disease ..... COPD
Clinical Classification Software ..... CCS
Clinical Decision Support Systems ..... CDSS
Clinical Modification ..... CM
Computational Barthel Index Tool ..... CBIT
Congestive Heart Failure ..... CHF
Critical Limb Ischemia CLI
Current Procedural Terminology ..... CPT
Decision Tree ..... DT
Veterans Affairs ..... VA
Diagnosis-Related Group ..... DRG
Electronic Health Record ..... her
Electronic Medical Record ..... EMR
Elixhauser ..... ELIX
End Stage Renal Disease ..... ESRD
Gated Recurrent Unit ..... GRU
Gradient Boost ..... GB
Healthcare Common Procedure Coding System ..... HCPCS
Health Information Technology for Economic and Clinical Health ..... HITECH
Intensive Care Unit ..... ICU
International Classification of Diseases ..... ICD
K-nearest Neighbors ..... KNN
Logistic Regression ..... LR
Long Short-Term Memory ..... LSTM
Machine Learning ..... ML
Mean Square Error ..... MSE
Mean Absolute Error ..... MAE
Medicare Payment Advisory Committee ..... MedPAC
Missing at Random ..... MAR
Missing Completely at Random ..... MCAR

Minimum Data Set ..... MDS
Missing not at Random ..... MNR
Model Correlation Plot ..... MCP
Mortality Prediction Model ..... MPM
Multiple Organ Dysfunction Score ..... MODS
National Cardiovascular Data Register ..... NCDR
National Institute of Health ..... NIH
Organ Failure Assessment ..... SOFA
Principal Component Analysis ..... PCA
Recurrent Neural Network ..... RNN
Random Forest ..... RF
Simplified Acute Physiology Score ..... SAPS
Severity of Illness ..... SOI
Surveillance, Epidemiology, and End Results ..... SEER
Tuberculosis ..... TB

ABSTRACT

A Study of Administrative Data Representation for Machine Learning

Negin Asadzadehzanjani, Ph.D.

George Mason University, 2022

Dissertation Director: Dr. Janusz Wojtusiak

Administrative data, including medical claims, are frequently used to train machine learning-based models used for predicting patient outcomes. Despite many efforts in using administrative codes (medical codes) in claims data, little systematic work has been done in understanding how the codes in such data should be represented before model construction. Traditionally, the presence/absence of these codes representing diagnoses or procedures (Binary Representation) over a fixed period (typically one year) is used. More recently, some studies included temporal information into data representation, such as counting, calculating time from diagnosis, and using multiple time windows. However, these methods were not able to comprehensively capture temporal information in data and much of temporal information such as the exact time of the occurrence of an event, and the exact sequence of an event are missed. This dissertation presents the results of development and investigation of two additional methods of administrative data representation (Temporal Min-Max and Trajectory Representation) specific to diagnoses extracted from

claims data before applying machine learning algorithms. It then presents a large-scale experimental evaluation of these methods by comparing them with traditional Binary Representation using four classification problems: one-year mortality prediction and high utilization of medical services prediction, prediction of chronic kidney disease and prediction of congestive heart failure. It was shown that the optimal way of representing the data is problem-dependent, thus optimization of representation parameters is required as part of the modeling.

INTRODUCTION

Machine Learning in Healthcare

Machine Learning (ML) as a subfield of Artificial Intelligence (AI) is one of the fastest growing fields in computer science. According to Samuel, machine learning is "a field of study that gives computers the ability to learn without being explicitly programmed" (Samuel, 1959). The learning is often done by identifying patterns in data, which can subsequently be used as models for predicting the future, assessing unknown properties, controlling equipment, or making recommendations. Machine learning combines concepts from different fields such as computer science, statistics, logic, and optimization and aims at developing algorithms for creating models that are hard or impossible to build using traditional computer science methods (Wiens & Shenoy, 2018). In a sense, building models can be treated as formulating hypotheses (building new knowledge or skills) that can later be tested with traditional hypothesis testing statistical approaches.

When learning to solve specific tasks, machine learning algorithms can detect patterns in very large datasets, which makes them often suitable for health applications. ML-based models can be used to diagnose diseases, make effective treatment decisions and improve patient's healthcare quality as well as safety (Nithya & Illango, 2017). There are several spectacular successes achieved by ML methods in health care. IDx-DR is the first FDA-approved of machine learning tool with actual application in health care. In this software, the machine learning-based model analyzes retinal images and diagnoses if the

patient has diabetic retinopathy or not (Abràmoff et al., 2018). Other interesting works include applying a deep neural network to classify prostate cancer using ultrasound images (Azizi et al., 2017) and detecting lymph node metastases from breast cancer (Golden, 2017). Researchers also applied machine learning methods to medical claims and Electronic Health Records to predict myopia prognosis with accuracy of up to 99% (Lin et al., 2018), septic shock (Henry et al., 2015), lung cancer severity (Bergquist et al., 2017) to name a few.

The wide-spread use of Electronic Health Records (EHRs) in healthcare systems and the emergence of registries and claims data have provided the opportunity to apply ML methods to very large health data in terms of scope and size (Shah et al., 2018). In the United States, the significant investment into Health Information Technology (Health IT) infrastructure as part of Health Information Technology for Economic and Clinical Health (HITECH) Act in 2009, opened the possibility of using ML methods on a large scale. At the same time, the majority of medical claims processing has been shifted to electronic form that resulted in the creation of very large datasets that can be used in ML research and practice. This dissertation relies on such large medical claims datasets.

One important drawback in applying ML methods to EHR, claims and other types of health care data is that the data cannot be readily analyzed in their raw form. Significant preprocessing of data is needed to arrive at what is acceptable by ML algorithms. This dissertation attempts to address some of these challenges by studying representations of claims data appropriate for ML methods.

Machine learning Methods

Machine learning methods can be classified into three groups: Supervised Learning, Unsupervised Learning, and Reinforcement Learning (Figure 1). The latter is out of scope of this dissertation and the former two are described below.

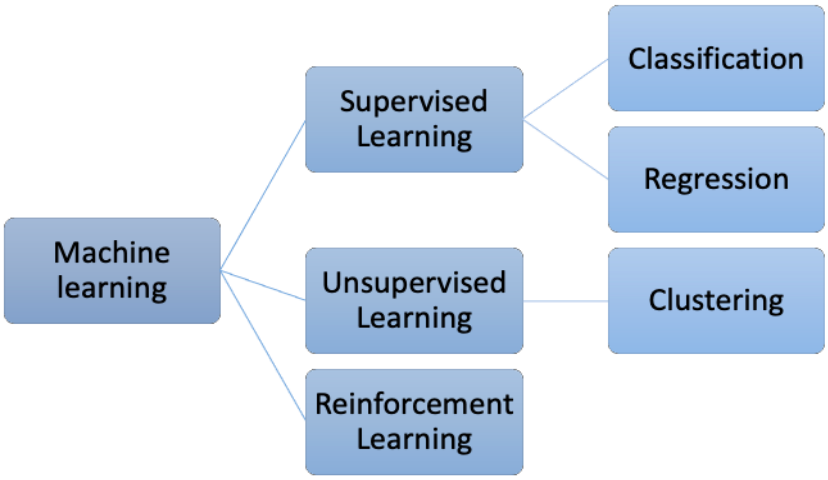

Figure 1: Graphic representation of machine learning methods classification.

Supervised Learning

Supervised Learning refers to the process of associating known inputs and outputs. This represents a situation in which examples are provided with 'true' answers or solutions to the problem. In most cases, the 'supervision' comes in the form of historical data for which the outputs (in health domain often referred to as outcomes) are known. For example, an algorithm may be applied to solve a problem of predicting high utilization of medical services. The algorithm is provided with examples of past patients for whom the

output (here high utilization or not) is known. By analyzing these cases, the algorithm learns how to solve the problem, and the results of learning are stored in a form of a model that can be applied to classify new cases. As a result, the trained model can predict the output/outcome for new cases that are different from ones on which the model was trained.

There are two types of Supervised Learning methods: classification and regression. Classification method has categorical outcomes with two or more groups (i.e., benign and malignant tumors), while regression problems have continuous outcomes (i.e., a dollar amount on a medical bill) (Sidey-Gibbons J. & Sidey-Gibbons C., 2019; Uddin et al., 2019). Some of the most important algorithms in supervised learning include Logistic Regression (LR), Decision Tree (DT), Random Forest (RF), Gradient Boosting (GB), Decision Rules (DR), Naïve Bayes and K-nearest Neighbors (KNN). The first four are used as example algorithms in the experimental section of this dissertation. In addition to these 'traditional' ML methods, there is a growing interest in Deep Learning approaches to data analysis, discussed later.

Selected supervised learning algorithms

Logistic Regression (LR) (Kleinbaum & Klein, 2010) is a type of linear regression in which the outcome in binary. The core of Logistic Regression is a logistic (sigmoid) function. It is basically an S-shaped curve and is formulated at Equation 1. Logistic function is continuous, differentiable, and non-decreasing which makes it suitable for gradient descent to find optimal solution. Logistic function can take any numbers and map it to values between 0 and 1.

Equation 1: Logistic Function Equation

$$f(z) = \frac{1}{1 + e^{-z}}$$

The logistic function takes the input in the form of a linear regression from all attributes of the regression line and returns the probability belonging to a class after training the data. The Logistic Regression equation is shown at Equation 2. In this equation, *y* corresponds to output probability, *b0* up to *bn* are the coefficients related to the underlying linear model, and *b0* is the intercept of the linear regression Logistic Regression can output the probability based on one or more input attributes.

Equation 2: Logistic Regression Equation

$$y = \frac{e^{b0+b1x1+b2x2+\cdots+bnxn}}{1 + e^{b0+b1x1+b2x2+\cdots+bnxn}}$$

These probabilities would take one value of 0 or 1 in final prediction; the cases with a probability greater than 0.5 is classified as class 1, and 0 otherwise (Uddin et al., 2019).

Decision Tree (DT) (Quinlan, 1986) is a popular model representation used in machine learning that works by classifying data into a tree-like structure. A Decision Tree consists of internal decision nodes and terminal leaves. Each internal node applies a test function on an attribute, with each branch taking a possible value of that attribute. The tree classifies instances by starting from the root node and applying a test function to the attribute represented by an internal node and then moving down based on the values of the

corresponding attribute. The splitting is continued within different internal nodes creating smaller subsets until reaching the terminal node (terminal leaf) representing the outcome. Decision Trees are constructed using different algorithms to decide how to split nodes into two or multiple sub-nodes including ID3, CART, C4.5. Each of these algorithms uses different criteria in selecting and ordering the attributes in each node including Entropy, Information Gain, Gini Index, Gain Ratio etc. (Mitchell, 1997; Alpaydin, 2020).

ID3 is a greedy top-down algorithm that uses Entropy as an impurity measure parameter and Information Gain computed to place the attributes within the tree. The algorithm works by constructing the tree through attributes with the highest information gain and smallest Entropy. CART (the algorithm used in in dissertation) is another Decision Tree algorithm that works for both classification and regression problems. CART uses Gini Impurity (GI) as a criterion in splitting a node to sub-node. Gini Impurity is the probability of incorrect classification of a particular attribute when selected randomly. It is calculated by subtracting the sum of the squared of the probabilities of for each class from 1. The formula to calculate the GI is given below:

Equation 3: GINI Impurity Equation

$$GI = 1 - \sum_{i=1}^{n} p_i^2$$

In the above equation, *pi* represents the probability of an attribute belonging to a class.

The GI is calculated for each attribute and the attribute with the smallest Gini Impurity is placed as the root node. The calculation is then continued for other internal nodes until the tree is structured (Mitchell, 1997; Alpaydin, 2020).

Random Forest (RF) is an ensemble method that consists of several (typically many) Decision Trees, each of which predicts its own outcome (Breiman, 2001). Random Forest inferred from randomly selected subsets of data, thus guaranteed to be different on sufficiently large data. Random Forests are created by applying bagging (a.k.a., bootstrap aggregation) (Breiman, 1996) to both sample and attributes. Standard top-down Decision Tree learning algorithms such as CART are used to create individual trees. The process is repeated to create multiple trees (typically in the order of tens or hundreds). After a forest is assembled, the final classification decision is made by applying all of the trees to new examples. When there is a disagreement in prediction, the trees vote on the predicted outcome. Random forests output classification scores which are calculated as a proportion of trees voting for a given outcome (Olson & Wyner, 2018). Since Random Forest models are based on several trees, they are less sensitive to small changes in the data than Decision Trees (Uddin et al., 2019).

Gradient Boosting (GB) (Friedman, 2001) is essentially a group of ensemble machine learning algorithms that are used for both classification and regression problems. Gradient Boosting is somewhat similar to a classic AdaBoost algorithm (Freund & Schapire, 1997). The idea behind AdaBoost is to construct a number of models in a series and reducing reduce the errors in previous models. The AdaBoost classifier combines multiple weak classifiers into a single strong classifier. In AdaBoost algorithm, a tree is

first trained by assigning equal weight to all observations. Then, a second tree is constructed by increasing the weight on observations that are difficult to classify and decrease the weight to observations that are easy to classify. The goal is to improve the classification of the observations that are not well classified in previous trees. Like AdaBoost algorithm, Gradient Boosting trains the models in a sequential manner but the difference is that Gradient Boosting uses loss function instead of high weight in improving the prediction. In Gradient Boosting, each tree tries to minimize the loss function and the result of each step are aggregated in the final model to achieve a stronger learner.

Naïve Bayes is another supervised learning algorithm that is based on the Bayes theorem, which describes the probability of an event based on prior understanding of conditions associated with an event. In this classifier, it is assumed that the attributes used in the models are independent of each other (Uddin et al., 2019).

K-nearest Neighbors is one of the simplest classification algorithms in which 'K' refers to the number of neighbors associated with a data point. In this algorithm, the classes belonging to each of the neighbors of a data point are taken and the data point belongs to the class for which most votes go to (Uddin et al., 2019).

Unsupervised Learning

Unsupervised Learning, on the other hand, refers to methods used to discover patterns in data. It is basically an attribute extraction method in which no attempt is made to predict outcome (Sidey-Gibbons J. & Sidey-Gibbons C., 2019). The principal method used in Unsupervised Learning is called clustering, where objects with similar characteristics are grouped together while the heterogeneity is maximized across the groups

(Liao et al., 2016). Several studies have used clustering in biomedical field, such as grouping psychiatric patients based on their symptoms, grouping genes with similar biological functions, and grouping patients in needs of intervention (Liao et al., 2016). Clustering can help find the hidden structure of a dataset when there is uncertainty about which group the entity belongs to (Liao et al., 2016).

Some of the most popular clustering methods include K-means Clustering, Hierarchical Clustering and Gaussian Mixture Clustering (Jiang et al., 2017). In K-means Clustering, '*K'* data points are selected as clusters centers (centroids). Then '*n*' observations are partitioned into '*K'* sets in a way that each subject is assigned to a cluster with the nearest mean. The recalculation is then performed by changing the position of the '*K*' centroids until all observations are separated into groups in which the distance is minimized (Liao et al., 2016). Hierarchical Clustering has two categories: Agglomerative and Divisive methods; the former assigns a cluster to each data point and then merges them into larger clusters, while the latter starts with one large cluster and then it is divided into small clusters (Belciug, 2009).

Deep Learning

Deep learning has gained a lot of popularity in recent years. Deep learning refers to very large, multilayered neural networks that often have the ability to handle a wide variety of correlations in data. It has specifically become popular in clinical informatics by the emergence of large amounts of patient data. Deep earning has shown to outperform traditional methods by preprocessing and handling attributes in a shorter timeframe (Shickel et al., 2017).

The most important characteristic of deep learning is data representation. In contrast to traditional methods in which attributes are preprocessed manually from raw data requiring expertise and knowledge of the task in hand, deep learning methods can discover an optimal set of attributes from hidden correlation in the data. Most of the deep learning algorithms are designed based on Artificial Neural Network (ANN) (shown in Figure 2), which consists of interconnected nodes (neurons) within input, output, and hidden layers. The neurons in hidden layers store a number of weights which are updated through model training, with ANN weights being optimized until the loss function is minimized (Shickel et al., 2017). Recurrent Neural Network (RNN) is a type of ANN which works best for temporal data, making them suitable for analysis of claims and EHRs. The output of Recurrent Neural Network depends on the previous elements in a sequence, unlike Artificial Neural Network that assumes that input and output are independent. It takes information from previous input to impact current input and output. In one study, RNNs were used to predict all diagnosis and medications of the next visit using a longitudinal time stamped data (Choi et al, 2016). In another study, RNN was applied in early detection of heart failure risks using EHR data and was shown to perform better than other machine learning methods including Logistic Regression, KNN, and multilayer perceptron etc. (Choi et al., 2017).

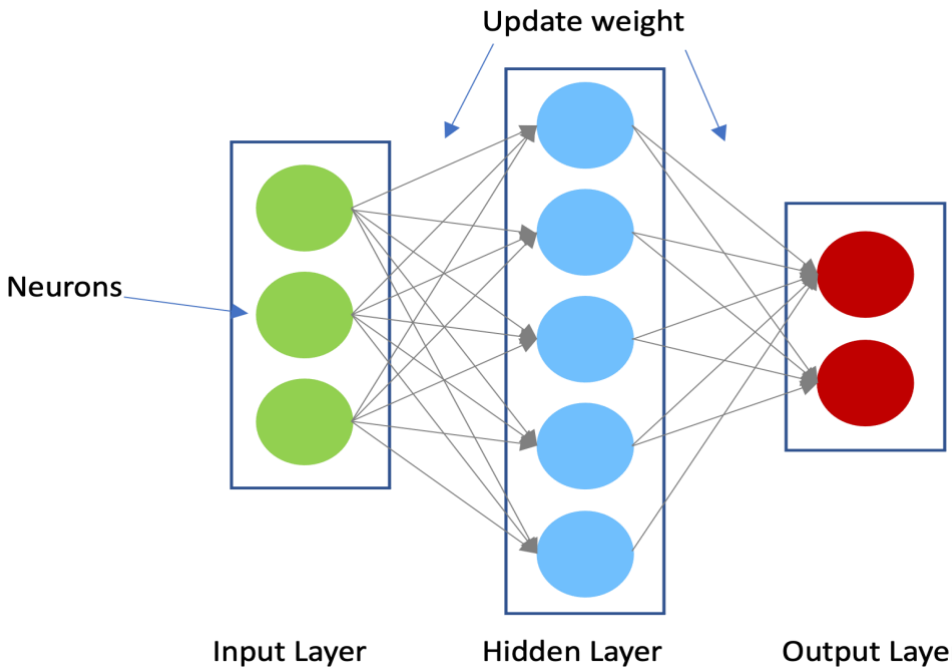

Figure 2: A connected Artificial Neural Network. The network consists of neurons within input, output, and hidden layers.

Health Data

Health data can be broadly categorized as those used in clinical, administrative, public/population health, and consumer applications. Health data are valuable source of information for medical and health services research. Some major sources of data used in health informatics research are: Electronic health records (EHRs), medical claims, registries, surveys, wearable sensors, and clinical trials. The following sections outline some of the most important types of health data.

Electronic Medical Records

Electronic Medical Records (EMRs) or Electronic Health Records (EHRs) are defined as "longitudinal electronic record of patient health information generated by one

or more encounters in any care delivery setting. Included in this information are patient demographics, progress notes, problems, medications, vital signs, past medical history, immunizations, laboratory data and radiology reports" (Atherton, 2011). Electronic Medical Records are tools to gather, store and represent patients' information and provide access to clinical information of patients. Over the past few years, there has been a growth in adaptation of hospitals with EHRs and according to Office of the National Coordinator for Health Information Technology (ONC), the adaptation of EHR systems has increased by 9-fold since 2008 (Shickel et al., 2017). The use of EHRs can improve patient's safety and healthcare quality and reducing the cost of care (Ajami & Bagheri-Tadi, 2013). Patient's information in EHR systems can also be used to extract medical concepts, modeling of patient's trajectory, disease inference and constructing clinical decision support tools (Shickel et al., 2017).

Medical Claims

Administrative data is a broad term referring to data that is used to process and document the registration and transactions for service delivery. Administrative data are collected to document a variety of services including education, healthcare, housing, taxation, etc. (Connelly et al., 2016). In healthcare, the most frequently used type of administrative data is medical claims. Often the terms administrative data and claims data are used interchangeably. Claims are essentially bills for provided medical services and include information required for the healthcare providers to receive payment. Therefore, the information included in claims data is limited to what is required by payers and typically corresponds to specific forms, such as CMS-1450 or CMS-1500 used by in the

United States by Medicare (CMS Forms List, n.d.). Health claims databases keep records of interactions that occurred between healthcare providers and patients which include all the billing information provided by hospitals, nursing homes, clinics, pharmacies, public and private insurance organizations such as Medicare/Medicaid and Blue Cross Blue Shield (Ferver, 2009). Claims data are typically generated at every encounter of the patient, which could be a procedure, a visit to doctors' office, admission to a hospital, or prescription (Cadarette, 2015).

For most patients, claims span longitudinally and provide a comprehensive summary of provided services when integrated by one payer. However, in certain situations, claims are incomplete, i.e., for dual- or triple-eligible patients or those seeking out-of-pocket paid services including uninsured. For example, Medicare beneficiaries may be also eligible for Medicaid and receive certain services from the Veteran's Affairs Health System for a military service-related disability. For research purposes, claims data are typically acquired from a single payer. For example, Medicare claims can be used to study populations 65 years and older in the United States. Private-pay claims data are typically used to study populations covered by a single insurer. In addition, it is sometimes possible to obtain integrated datasets from multiple payers (so-called all-payer data), but such data are often very costly and come with other types of limitations. Another reason for potential incompleteness of claims data is the inclusion of only billable items, i.e., those tied to reimbursement, which may miss additional services provided or the diagnosis that is not covered by insurance.

The types of information in claims data vary across different databases, but almost all claims datasets include date of claim (that may not be the same as date of service), diagnosis and procedure codes, provider information, site of service, charges and cost of healthcare delivery. They typically include demographic information including age (or date of birth), sex, race, and ethnicity, and sometimes education and income (Stein et al., 2014). The claims databases usually provide a list of all variables in a dictionary, yet the information is often vague and requires good understanding of coding systems and healthcare processes to correctly analyze data. Claims data have information in the forms of code, date, text, symbols etc., each of which requires special preprocessing steps for use in developing models. Claims data are typically structured, meaning that the data are stored in organized format, with little or no information provided as free text. This makes them suitable for data analysis and interpretation. In this structured data, there are standard healthcare coding systems (administrative codes) including International Classification of Diseases (ICD-9 or ICD-10) codes, Current Procedural Terminology (CPT), Healthcare Common Procedure Coding System (HCPCS) codes, etc. These standard codes, which are the focus of this dissertation, will be explained in detail in the next chapter.

Differences Between Claims and EHR Data

Claims data are broad in scope since they include patient information from potentially multiple healthcare providers. However, the information is limited to what is required to receive payment for services. Consequently, few details are available beyond billing codes. In contrast, EHR data are much more detailed and include laboratory values, vital signs, clinical notes, and orders. Yet, EHR data are often limited to one EHR system and one provider. As shown in Figure 3, while EHR data provide a comprehensive

overview of patient's information they, are narrow in scope; this is in contrast with claims data.

There are also other potential differences. Claims data only limit to insured patients whereas EHRs data contain information of both insured and uninsured individuals who received services from a given provider. Claims data captures accurate information if prescriptions were filled and refilled, while EHRs only contains prescription with no information about if it was filled or not (Wilson & Bock, 2012). Further, EHR data are typically up to date with most recent patient information available, while claims may be submitted with an allowable delay, sometimes months after an encounter.

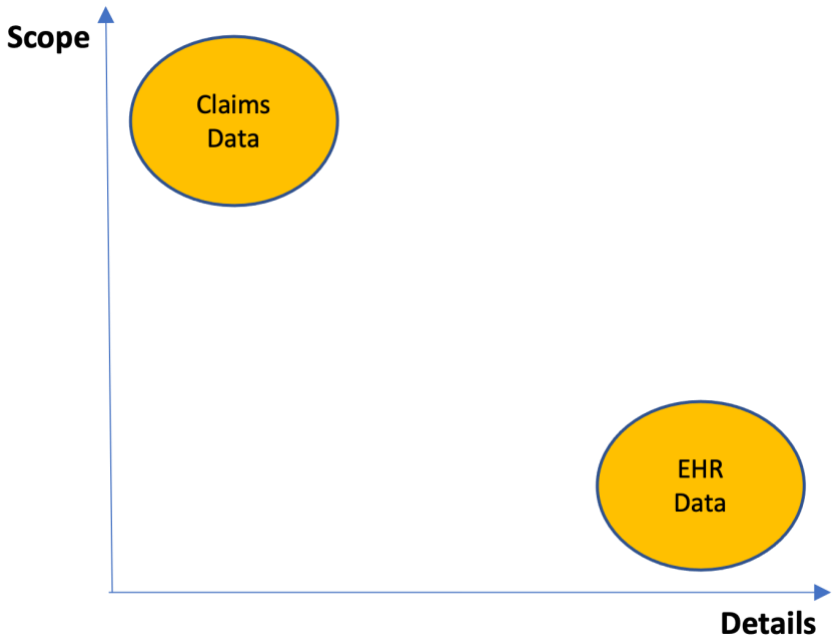

Figure 3: Claims versus EHR data

Health Registries, Clinical Trials, Surveys

Registry data is a type of health data collected for a group of individuals with specific health conditions such as cancer, heart disease, Alzheimer's disease, diabetes etc., typically collected for public health and epidemiological reasons. Many registries contain patient's information with a specific health condition, some of which are mandatory, such as reporting commutable diseases. Other registries focus on seeking volunteers with different health status to be involved in research for a particular disease. These registries provide valuable information about patients with particular conditions at an individual and group-level with the goal of increasing the understanding of that condition. Additionally, they can be used to track the prevalence of the diseases and their treatments. Some of the national-level registries include Global Alzheimer's Association Interactive Network (GAAIN), National Cardiovascular Data Register (NCDR), the Surveillance, Epidemiology, and End Results (SEER) Registries etc. (List of Registries, n.d.).

Clinical trials data are collected through clinical trial studies. According to national Health Institute (NIH), Clinical trial is "A research study in which one or more human subjects are prospectively assigned to one or more interventions (which may include placebo or other control) to evaluate the effects of those interventions on health-related biomedical or behavioral outcomes" ("NIH's definition", n.d.). Clinical trials are designed to determine if a treatment (e.g., drug and medical device) is effective and safe or if it is more effective and safer compared to standard ones. Clinical trials are useful in early detection of disease, preventing health problem and improving the quality of life for people with severe health conditions ("U.S. Department", n.d.).

Surveys are administered to seek information not typically present in claims or EHR data. Surveys are designed to systematically gather health and social information on a sample of patients in order to examine a larger population. Surveys can be divided into two categories: population surveys and provider surveys. Survey data can be collected using many different methods, including online questionnaires, mail, phone, and in-person interviews, with the majority of surveys are now conducted online. Some survey data collected by National Center for Health Statistics include: The National Health Interview Survey, The National Health and Nutrition Examination Survey, and The National Ambulatory Medical Care Survey ("Surveys", n.d.).

Machine Learning Model Construction in Healthcare

Although models differ in terms of outcome, dataset and algorithm, they follow common construction procedures when starting with raw claims or EHR data. In fact, the data preprocessing and preparation typically takes significantly more time than model learning and tuning. After defining the prediction problem and its clinical implications, one of the first technical steps is construction of the cohort of interest. Technically, the construction may be based on availability of data, and most importantly outcome in the training data, and include instances (e.g. patients) that have or do not have a target value or have target value in a numeric format. The concept of cohort construction is often overlooked in more technical ML publications, yet it has important implications for model generalization and the types of patients it can be applied to.

Among other preprocessing steps, one that is particularly related to the presented study is defining the observation window, which refers to how much we would like to go

back in time to collect information and construct the analytic files. The independent attributes are constructed in the observation window and any records occurred out of this timeline would be excluded.

There are two types of outcome prediction: 1. Static or One-time prediction and 2. Temporal outcome prediction. Static outcome prediction uses data from one encounter (such as heart failure prediction), while Temporal prediction uses time interval or time series data (such as predicting heart failure in the next six months) (Shickel et al., 2017). The time interval is called prediction window, which refers to how far we would like to make prediction in the future.

After defining cohort, observation and prediction window, the raw data is cleaned in order to identify and remove any incorrect or corrupt information in the data. The data is then be preprocessed to be suitable for constructing the model.

The prepared data are then split into train and test sets; the train dataset is used to apply the algorithm, learn the model and tune hyperparameters, while test data is used to evaluate the final model. The model with the best performance is selected for application. Figure 4 illustrates the high-level steps associated with constructing the models in claims or EHRs. It should be noted that the steps may overlap or appear in different order. The focus of dissertation is the preprocessing step of this framework specific to claims data. Later in the dissertation, the preprocessing framework for claims data is discussed in detail, which encompasses data cleaning, cohort construction, and observation/prediction window construction.

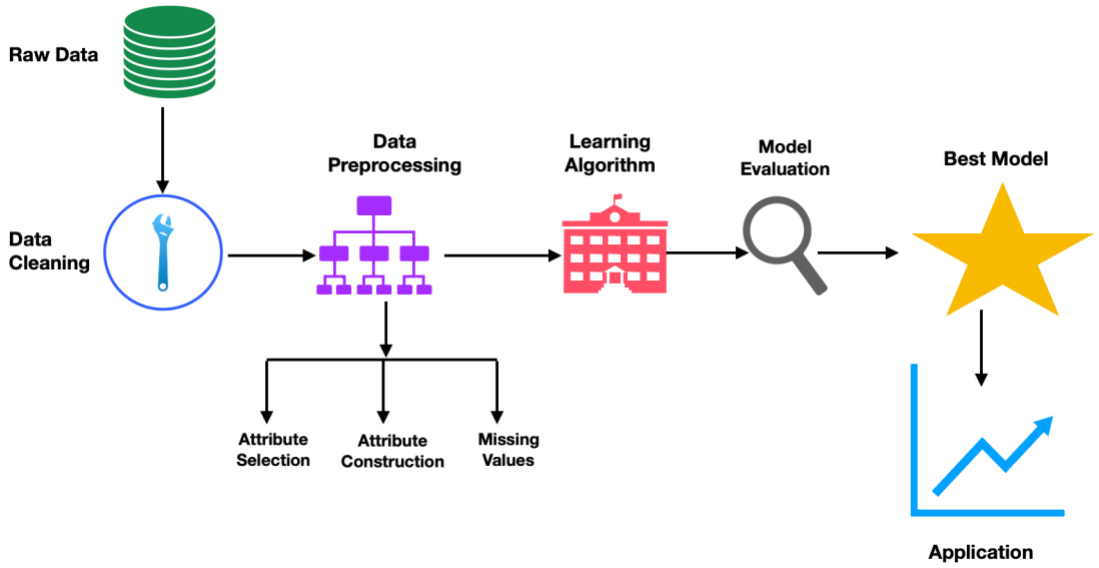

Figure 4: High-level steps in the building machine learning-based models starting from raw claims or EHR data.

Model Evaluation in Healthcare

There are standard measures in reporting the performance of models in machine learning. In classification problems the model quality is reported with Area Under Receiver-Operator Curve (AUC), accuracy, precision, recall, while in regression problems, Mean Square Error (MSE), Mean Absolute Error (MAE), and correlation coefficient are typically used. Accuracy is a metric that measures how many times the model makes correct, or incorrect classification, which is defined as total number of correct examples divided by the number of cases in the test set. Recall is defined as the total number of correctly classified positive examples divided by the total number of positive examples, while precision is the number of true positives divided by number of examples classified as positive. AUC measures the continuous relationship between true and false positive rates at different classification thresholds. Many authors combine precision and recall into a

single F1-score (Goutte & Gaussier, 2005). In the published literature, these measures are used to report results of modeling efforts, but also when applying hyperparameter tuning to achieve the highest performance (accuracy, AUC, or F1-score). Other metrics are also used such as Area Under Precision-Recall Curve (AUCPR) (Boyd et al., 2013), Kappa Statistic (McHugh, 2012), relative entropy, mutual information, and others (Baldi et al., 2000). Another method of model evaluation is Calibration, which refers to measuring how much the predicted probabilities differ from the actual probabilities (Chen et al., 2019). Calibration of the models can be tested using Calibration Curves, which plot the relative frequency of observed values against the frequency of predicted probabilities.

While all or some of the above statistical measures are used in virtually every published work that use ML methods in health applications, some authors argue that it is not sufficient. Wojtusiak (Wojtusiak, 2021) argued for more detailed testing of methods and specified ten criteria in evaluating models. Luo et al. (Luo et al., 2016) developed guidelines for reporting results of machine learning methods in biomedical research. Wojtusiak and Asadzadehzanjani (Wojtusiak, & Asadzadehzanjani, 2022) presented methods for comparing models (see “Temporal Min-Max Representation” chapter). Many other authors argued for the importance of other specific criteria, mainly transparency and reproducibility of models. Model transparency is concerned with allowing people to understand the internal workings of the constructed model. Transparency can be achieved using two main approaches: 1) Construction of models which are transparent in the first place and including models such as Decision Rules, Decision Trees, Bayesian Networks and linear models; 2) Providing explanation for black-box models (Wojtusiak, 2021).

Reproducibility refers to producing the same results from the same input data with a specific methodology. Reproducibility can be tested using different methods including intraclass correlation (ICC) and ANOVA test (Renard et al., 2020).

Selected Application Areas

Many studies have used machine learning to predict and study health outcomes. Some of the important works are to predict risk of mortality, readmissions, high utilization of medical services, disabilities, as well as many disease-specific outcomes. Depending on the application and population considered, the quality of these models varies. The following sections summarize the application of ML in predicting four health outcomes, which are used to describe methods in this dissertation. In addition, the application of machine learning in assessing functional abilities will be discussed in "Temporal Min-Max Representation" chapter.

High Utilization of Medical Services Prediction

In the United States, high healthcare spending is a significant burden on the economy, and it has grown since the 1980s (Hu et al., 2015). In some studies, a high utilizer is defined as a top 5% consumer of healthcare. Such patients consume about 40% to 55% of healthcare costs (Roysden & Wright, 2015). While most studies determine high utilization by the frequency of visits or the total amount of healthcare costs over a period, some studies have applied clustering methods. (Hyer et al., 2020a; Hyer et al., 2020b). In recent years, machine learning algorithms have been used to identify patients who are high utilizers of medical services. In one study, Bayesian algorithm was used to predict healthcare utilization based on all-cause patient's hospitalizations as well as predicting

associated healthcare costs among patients with Critical Limb Ischemia (CLI) (Berger et al., 2020). In another study, Random Forest algorithm was utilized to predict decrease in high utilization and ultra-high absolute utilization after patient's first behavioral visit. The utilization was predicted for the two models with AUC of 0.74 and 0.88, respectively (Roysden & Wright, 2015).

Mortality Prediction

An important metric in assessing a patient's health and predicting health outcomes is patient mortality. In recent years, several illness severity scoring systems have been used to evaluate in-hospital mortality. These systems include Simplified Acute Physiology Score (SAPS), Mortality Prediction Model (MPM), Organ Failure Assessment (SOFA), Multiple Organ Dysfunction Score (MODS), Acute Physiologic and Chronic Health Evaluation (APACHE) etc., which use data collected at the first 24 hours of Intensive Care Unit (ICU) admission or in longer period of time. The issue with these scoring systems is that they are appropriate for generalizing but may not be accurate when predicting individual patient's mortality (Wojtusiak et al., 2017).

In recent years, with the development of EHRs, a large number of ML-based models were developed to predict mortality and have shown better performance compared to the standard methods. Wojtusiak et al. constructed models (called C-LACE) to predict 30-day post-hospitalization mortality based on LACE models by using demographic information, diagnoses, laboratory values and medications achieving the AUC of 0.74. It was also shown that the model with top 20 attributes performs identical to the model with

the full set of attributes (308 attributes) (Wojtusiak et al., 2017). They later constructed another model (C-LACE2) to address the limitations of their first model by improving the representation of laboratory values, proper use of diagnoses codes, and improving the stability of the models resulting in model with AUC of 0.779 (Wojtusiak et al., 2018). In another study, note topics were extracted from data and were combined with static attributes to predict mortality using MIMIC data, resulting in the AUC of 0.84 (Ghassemi et al., 2014). Kim et al. developed a real-time warning score tool to predict real-time mortality 6 to 60 hours before death in pediatric ICUs and achieved the AUC of 0.89 to 0.97, respectively (Kim et al., 2019).

In addition to EHRs, claims and administrative data have been used in predicting mortality. In one study, Medicare claims data was used to predict 6-month mortality among patients over 65. It was shown that the ML-based model outperforms the traditional risk assessment models by capturing the severity and progression of the disease in constructing features (Makar et al., 2015). Aktuerk et. al used national administrative data to predict 1-year mortality after cardiac surgery, with C-statistic ranging from 72% to 81.6% (Aktuerk et al., 2016). In another work, 15-months mortality was predicted among community-dwelling Medicare beneficiaries. It was shown that the constructed models with the C-statistic of up to 0.795 had higher generalizability compared to other models developed on administrative databases (Berg & Gurley, 2019).

Chronic Kidney Disease Prediction

Chronic kidney disease (CKD), which is characterized by lack of kidney function, is a lifelong disease. Chronic kidney disease can progress to End Stage Renal Disease

(ESRD), which requires the patient to undergo dialysis and a kidney transplant (Segal et al., 2020). Considering the large number of patients affected by it, the risk of progression to ESRD, and the mortality rate, this disease causes a significant burden to the healthcare system (Krishnamurthy et al., 2021). Therefore, it is important to identify patients at risk of developing CKD to reduce healthcare expenditure and improve health outcomes. Ren Y et al. predicted CKD among patients with hypertension from EHRs using Neural Network framework and achieved AUC of 89.7%. In this study, bidirectional long short-term memory and auto-encoders were applied to represent the textual and numerical information, respectively (Ren et al., 2019). In another study, Logistic Regression was used to identify patients with CKD and at risk of kidney failure within 2 years from large claims dataset. The model used patients' demographic factors, CKD stage, patients' health status (history of having diabetes, congestive heart failure, hypertension etc.), and risk group score achieving an AUC of 0.844% (Dai et al., 2021). Ilyas et. al. also predicted different stages of CKD using J48 and Random Forest algorithms to predict different stages of CKD. The results indicated that J48 was performing better than Random Forest in predicting all stages with the AUC of 85.5% vs. 78.25% (Ilyas et al., 2021).

Congestive Heart Failure Prediction

Heart Failure or Congestive Heart Failure (CHF) is a public health crisis that contributes to significant mortality rates, morbidity, and health expense in older individuals, particularly those over 65 years of age (Desai et al., 2020). It is estimated that 1 out of 8 deaths in the United States is caused by heart failure. Many ML-based models have been developed to identify patients at risk of CHF or to predict related health

outcomes in order to provide better treatment in a timely manner. In one study, Random Survival Forest was used to predict cardiovascular events outcome, including heart failure, and the model achieved the AUC of 84% (Ambale-Venkatesh et al., 2017). König et al. developed models to predict in-hospital mortality rate among patients with CHF and showed that different machine learning algorithms outperformed classic regression approach specifically among Gradient Boosting and Extreme Gradient Boosting algorithms (König et al., 2021). A recent study showed that the incorporation of additional continuous variables into binary variables in ML-based models can improve the prediction of heart failure outcomes compared to traditional Logistic Regression (Desai et al., 2020).

The focus of this dissertation is on data preprocessing step specific to administrative codes (these codes are discussed in "Administrative Codes Representation Methods" chapter) applied to transform raw data into the final analytic file in supervised machine learning methods. The concepts discussed in this dissertation are described for claims data but can be generalized on any other types of health data. In general, construction of ML-based models follows two main steps: data preprocessing and hyperparameter tuning. Data preprocessing is transformation of data into appropriate format for data analysis (will be discussed in detail in the next chapter), but model tuning or hyperparameter tuning is the process of finding the best set of settings in model construction. Interestingly, most researchers only consider hyperparameter tuning as optimization of learning algorithm, while keeping fixed preprocessing steps. It is our experience that data preprocessing is as important as algorithm hyperparameter tuning or selection of specific ML methods. In fact, this work considers data preprocessing steps as part of model tuning. Changes in the

representation space caused by data preprocessing result in very different types of effects on the model than those that occur when tuning hyperparameters.

The concepts discussed in this dissertation will be tested on four learning problems using supervised machine learning: 1) Mortality Prediction, 2) High Utilization of Medical Services Prediction, 3) Chronic Kidney Disease Prediction and 4) Congestive Heart Failure Prediction.

ADMINISTRATIVE CODES REPRESENTATION METHODS

Preprocessing Definition

Due to high-dimensionality, heterogeneity, noise, incompleteness, sparseness, and errors in the data, modeling of health data, including claims, is difficult (Miotto et al., 2016). To remedy this problem, data preprocessing is an essential step in developing ML-based models. The reliability of the preprocessed data should be checked before training the model as any errors made in constructing the data would impact the accuracy of the models (Ngiam & Khor., 2019). Data preprocessing refers to several steps required to transform raw data into the appropriate format for analysis (Malley et al., 2016). As another definition, data preprocessing refers to the methods including constructing new attributes (attribute construction), removing irrelevant attributes (attribute selection) and modifying the attributes in which the initial representation space is improved (Wojtusiak, 2008). In data preprocessing, the goal is to reduce the complexity of the data and extract the relevant attributes from the data, which then can be used for further analysis (Castillo et al., 2011).

Preprocessing of Claims Data

Figure 5 shows how the raw claims data are transformed into an 'analytic file' for training and testing of classification and regression models. Claims data usually consist of tables from multiple sources including inpatient files, outpatient files, carrier files etc. collected over many years and stored in separate files. Different files/tables correspond to types of claims and are separated because they include different fields. Depending on the application, relevant information including medical codes, demographic information,

patient IDs, claim IDs, various dates etc. is first extracted from claims. Then one or more inclusion/exclusion criteria are applied to construct the targeted cohort from data. Once the prediction time is defined, the observation and prediction windows are established in which input and output attributes are constructed, respectively. The attributes are then processed in multiple steps including aggregation, discretization, normalization, and handling missing and integrated if needed, resulting in creation of the final analytic file. The analytic files are in the matrix format where rows represent instances and columns represent constructed attributes in the data. A wide variety of steps are defined for data preprocessing in the literature and depending on the application and needs, some or all are used to prepare the final analytic table. The focus is on the preprocessing of administrative codes and how they are represented before applying ML algorithm.

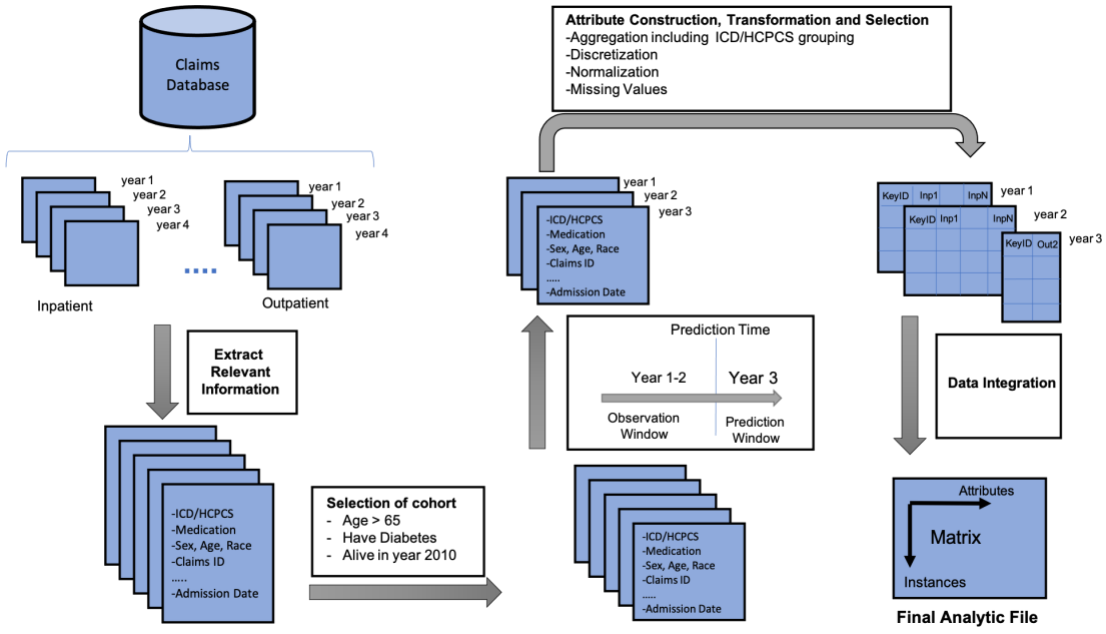

Figure 5: An example process of transforming raw claims data into the final analytic file.

Administrative Codes

Claims data contain standard healthcare coding systems (administrative codes) that are either universally or locally utilized by the healthcare systems. These include International Classification of Diseases (ICD-9 or ICD-10) codes, Current Procedural Terminology (CPT), Healthcare Common Procedure Coding System (HCPCS) codes. The codes refer to procedure or diagnosis codes that are used to report diagnoses, encounters, injuries, morbidities etc. withing the healthcare system. Diagnosis and procedure codes assigned to medical claims are known to contain errors and inaccuracies. Despite these errors, they are popular source of information in predictive modeling and there are many successful applications of these administrative codes in predicting health outcomes.

Types of Administrative Codes

International Statistical Classification of Diseases (ICD)

The 9th version of International Classification of Diseases (ICD-9) is used globally in healthcare settings to classify disease. In the United States, ICD-9-Clinical Modification (CM) which is developed based on the official version of ICD-9 is widely used by the U.S. healthcare organizations at any level (federal, state, local) to report diagnoses, encounters, injuries, and morbidities. In 1999, ICD-10-CM and ICD-10-PCS were developed by CMS for reporting medical diagnoses and procedure coding systems, respectively. The reason for such transition was that the ICD-9-CM coding system were limited and restrictive and was not sufficient enough for the needs of healthcare system. The ICD-10-CM codes have improved the quality of health data by tracking public health conditions, clinical decision making, outcome measurements and devising payment systems. ICD-9-CM has about

18000 codes for combined procedures and diagnoses, whereas ICD-10 has about 142,000 codes. ICD-10 procedure codes are 19 times more detailed than the ICD-9-CM codes and ICD-10 diagnoses codes are 5 times as many as the ICD-9-CM codes. In ICD-10-CM, more morbidity information is provided, making the coding system more flexible for expansion (Cartwright, 2013; "Centers for Disease Control and Prevention", 2015).

Current Procedural Terminology (CPT®)

The Current Procedural Terminology (CPT®) codes are used to report all kinds of healthcare services, such as surgical, radiologic and laboratory. The CPT codes have three categories with five digits in either numeric or alphanumeric form: Category 1 refers to procedures or services with subcategories dependent on the procedure or the services; Category 2 that consists of supplementary codes to measure performance; and Category 3 codes that are designed for emerging technology, procedure, and services. These codes are typically used for data collection, new services payment or any procedures that cannot be included in category 1 ("*CPT® Overview and Code Approva"l*, n.d.).

Healthcare Common Procedure Coding System (HCPCS)

HCPCS (Healthcare Common Procedure Coding System) is another type of coding system that has three levels and extends CPTs. Level one is essentially CPT codes, while level two refers to all non-physician services, products and supplies such as ambulance services, durable medical equipment and prescription drugs codes that are not included in CPT codes. Level three codes are local codes that are used when procedures do not fall within the first or second categories (Torrey, 2020).

Code Groupers

In many applications, including machine learning modeling, there is often no need to keep the original diagnosis codes, or the number of codes is simply too large to handle. Code groupers are broader categories of codes that are mapped from the original ICD codes into larger categories allowing for significant dimensionality reduction when constructing data. Some of the most popular types of code groupers are Clinical Classification Software (CCS), Charlson, Elixhauser, etc.

Clinical Classification Software (CCS) Codes

Clinical Classification Software (CCS) codes developed by the Agency for Healthcare Research and Quality (AHRQ) were created by collapsing ICD-9 codes to smaller number of codes. This coding system is used for cost, utilization and outcome analysis and comprises of diagnosis and procedure codes. The single-level coding system includes 285 diagnosis codes and 231 procedure codes with no hierarchical structure while the multi-level system consists of four levels of diagnosis and three levels of procedure codes with hierarchical structure ("CCS (Clinical Classifications Software) – Synopsis", n.d.; "Clinical Classifications Software (CCS) for ICD-9-CM Fact Sheet", n.d.). ICD-10 codes can also be mapped to CCS codes. At the time of writing this dissertation, the mapping of ICD-10 codes to CCS is still under development; however, the 'beta' version is available to the public. In this version, about 77.000 procedure codes are mapped into a limited number of meaningful clinical codes, which are more useful for demonstrating descriptive statistics of data ("Clinical Classifications Software (CCS) for ICD-10-PCS (beta version)", n.d.).

Charlson and Elixhauser Index

Charlson Comorbidity Index is another coding system that categorizes the comorbidities based on ICD9 codes. This system was first devised to predict one-year mortality from a range of comorbidities. It consists of 19 comorbidities including acute myocardial infarction, congestive heart failure, peripheral vascular disease, each of which is given a score of 1 to 6 based on the mortality risk and resource utilization. Score of zero means that the patient has no comorbidities, and higher scores indicate higher risk of dying (Charlson et al., 1987; "Concept: Charlson Comorbidity Index", n.d).

Like Charlson comorbidity index, Elixhauser is a method of categorizing comorbidities according to both ICD-9 and ICD-10 codes. The index started with 30 comorbidities but expanded to 31 categories (Gasparini, 2018).

In supervised machine learning, a combination of all coding systems is usually applied. Although grouper codes may not provide enough information in supervised machine learning, but they do well in preventing model overfitting, which is a common issue when a comprehensive coding system such as raw ICD codes are used.

Standard Methods of Representing Administrative Codes

Claims data can be viewed as a sequence of claims that include one or more diagnosis codes recorded over time. They are irregularity spaced in time. The data are also potentially censored on both sides because of benefit eligibility and events such as death. Since most machine learning (ML) algorithms cannot handle records with variable number of attributes, some summary functions including Boolean representation and counting the

occurrence of each event are used to aggregate the data and remove temporality before applying ML algorithms. This is not different from other health data such as EHRs.

The following sections describe increasingly complex approaches for representing diagnoses extracted from claims data. While these sections focus on diagnoses, the same methods can be applied to procedure codes.

Binary Representation

The simplest and most frequently used method is to represent presence/absence of diagnosis codes with a set of binary attributes (sometimes referred to as *dummies*). Let $Code_i$ be the administrative code representing a diagnosis code and C be the claim in the patient's record prior to the prediction time. Sometimes, the presence of codes within a certain time window is used instead of entire patient record. Either looking into entire records or a specific time frame, the frame is called the observation window. The administrative code ($Code_i$) associated with C is represented as 1 if the $Code_i$ belongs to claim C, 0 otherwise. The equation is given as follows:

Equation 4: Binary Representation

$$Code_i = \begin{cases} 1 & \exists\, C \quad : Code_i \in C \\ 0 & Otherwise \end{cases}$$

The above method however has the risk of information loss. Most health data collected during patient care are longitudinal, meaning that the patients are observed over a course of time. The health data have time-stamped entities, meaning that much of the

information such as emergency visit, hospitalization or blood test are recorded with time. Moreover, the time each patient is tracked varies across all patients (Tran et al., 2014). Also, the sequences of the events are highly correlated; for instance, a diagnosis could be made after a blood test result comes back (Liu et al., 2018). The above Binary Representation method cannot capture the heterogeneity and hidden temporal information in the data i.e., the severity of illness or the changes in prognosis of the disease over time. Therefore, there has been a growing interest to leverage such information in constructing the analytic file. Studies have shown that the incorporation of temporal information can improve the performance of the predictive models (Xie et al., 2016; Singh et al., 2015). For instance, Google proposed a method to learn temporal attributes from all attributes in EHRs using long short-term memory (LSTM) that can improve the AUC of three health outcomes (mortality, readmission and long hospital stay) by 10% (Rajkomar et al., 2018). Below, several methods used in the literature to introduce temporal information into data representation are explained.

Binary Representation with Multiple Time Bins

Another standard method to capture temporal attributes is to divide the observation window into multiple time bins and apply Binary Representation for each bin separately. Assume *w* is the bin in an observation window and *t(C)* is the time of claim. As shown, the administrative code ($Code_i$) associated with *C* at time window (bin) *w* is represented as 1 if the code belongs to claim *C* and *t(C)* falls in bin *w*, 0 otherwise. The method works as follows:

Equation 5: Binary Representation with Multiple time Bins

$$Code_i^w = \begin{cases} 1 & \exists\, C \quad : Code_i \in C \quad \wedge \quad t(C) \in w \\ 0 & Otherwise \end{cases}$$

The advantage of such representation is that the approximate time of an event is incorporated into the code representation and model. Such method was used in a study, in which equal time intervals (yearly, quarterly, monthly) were constructed to predict the number of hospitalization days in the upcoming year. The results showed that using smaller bins added more temporal information to the models, and the yearly model had significantly worse performance than the others (Xie et al., 2016).

Figure 6 graphically compares Binary Representation method with single vs. multiple time bins. In single observation window, diagnosis codes are extracted from raw data and then Binary Representation is applied to create separate column of each code shown as $Code_1$, $Code_2$,…, $Code_N$. The table shows six records associated with three patients. The data is then aggregated resulting in the final file with three records of three patients. However, when multiple time bins are used, the observation window is divided into multiple time bins (*w* bins) and within each bin, the Binary Representation is applied. Therefore, the total number of attributes in the final analytic file is *w* times more than the single window. It should be noted that these time bins can have overlap or can be disjoint.

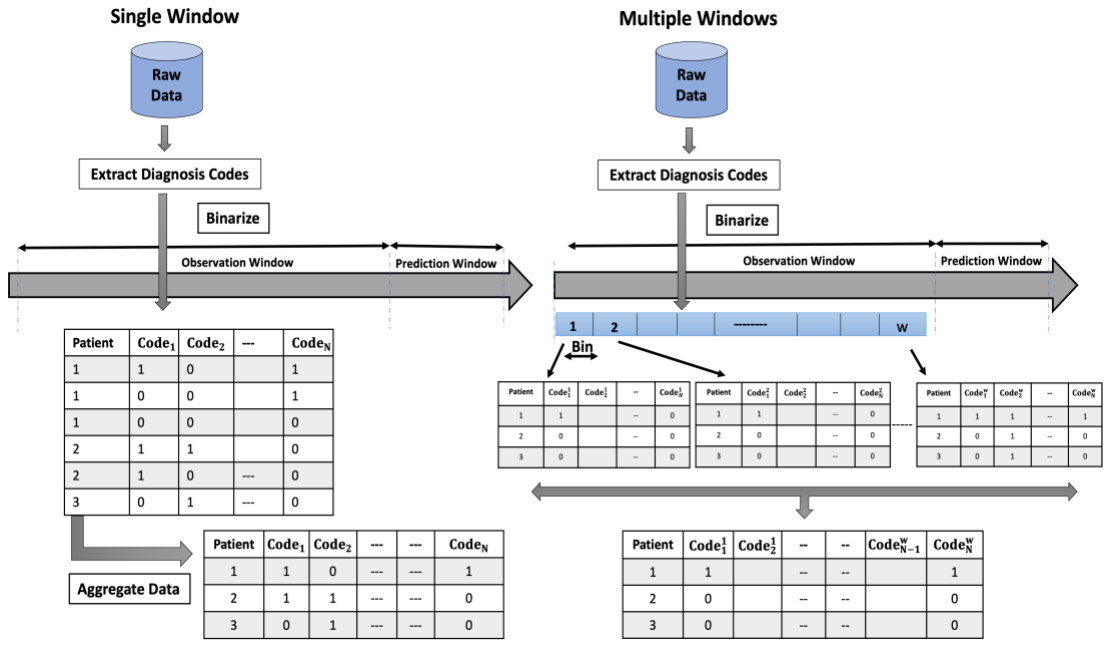

Figure 6: Comparison of single vs. multiple time bins in Binary Representation.

Enumeration Representation

Enumeration is another method in representing the administrative codes. In this method, the number of times a code is present in patient's medical history within a predefined time window is counted. Therefore, instead of using binary indicators, the present/non-present codes can be replaced by the number of times they occurred. The formula to create these codes are given below. As shown, each code ($Code_i^{cnt}$) is represented as sum of the present codes specific to claim C.

Equation 6: Enumeration Representation

$$Code_i^{cnt} = \sum_{c \in claims} \begin{cases} 1 & \exists\ C\ if\ code\ i\ present\ on\ claim\ c \\ 0 & Otherwise \end{cases}$$

This method of representing data clearly captures more information than simple Binary Representation. However, one needs to carefully plan for the specific type of classifier to construct models, and specifically how to represent diagnoses that are not present in patient's record. This approach was used to create a set of independent attributes that represent the total number of admissions, and total number of each CPT and diagnosis codes in predicting readmissions (He et al., 2014). In another study, the number of comorbidities as well as diagnoses were used in predicting high healthcare cost (Kim & Park, 2019).

Representation of Additional Information and Derived Attributes

In addition to the methods that represent individual diagnoses as attributes in the data, combinations of multiple attributes or attributes derived by some other means are often included in the data. For example, the total number of claims within a time window, time between hospitalizations, and the number of emergency care visits can be extracted from claims data to represent each diagnosis code. Further, individual and derived attributes can be modified by applying numerical transformation methods. For example, instead of considering time from the onset of a chronic condition, one may consider *log(time)* to emphasize recent changes and downplay small changes in distant past. Finally, global transformation methods such as those based on kernel methods or principal component can be used to transform all attributes in the space.

The methods mentioned above are some of the standard methods used in the literature to represent administrative codes (here diagnosis codes) in supervised machine learning. These methods are being used in limited settings and in some cases on a limited set of administrative codes. In this dissertation, two additional methods of representing administrative codes called 'Temporal Min-Max Representation' (*TMMR*) and 'Trajectory Representation' (*TJR*) will be introduced. The basis of these methods is in capturing more temporal information in representing the codes. The two methods are more complex but allow data to carry more information (Figure 7) than the standard administrative codes representation methods. Thus, they have a potential to improve the quality of the predictive models.

More importantly, there is no study available that has completed a systematic investigation of administrative code representation used in construction of machine learning-based models. The methods mentioned above, are used extensively in supervised machine learning without knowing if they are the best for the problem at hand. It is assumed that the method of representing data can be impacted by many factors including the type of algorithm, outcome, size of observation window, the type of administrative code etc. Thus, a large portion of this dissertation encompass a comprehensive evaluation and comparison of the administrative code representation specific two the two proposed methods. A large portion of this dissertation is dedicated to a comprehensive evaluation of the two proposed representation methods and its comparison with simple Binary Representation method. The concepts presented in this dissertation are described in context of four binary classification learning problems including predicting 1-year mortality, high

healthcare utilization of medical services, chronic kidney disease, and congestive heart failure. Comparison and evaluation of these codes are beyond standard model accuracy metrics and focus on the individual differences between the two representation methods, which will be thoroughly explained in the following chapters.

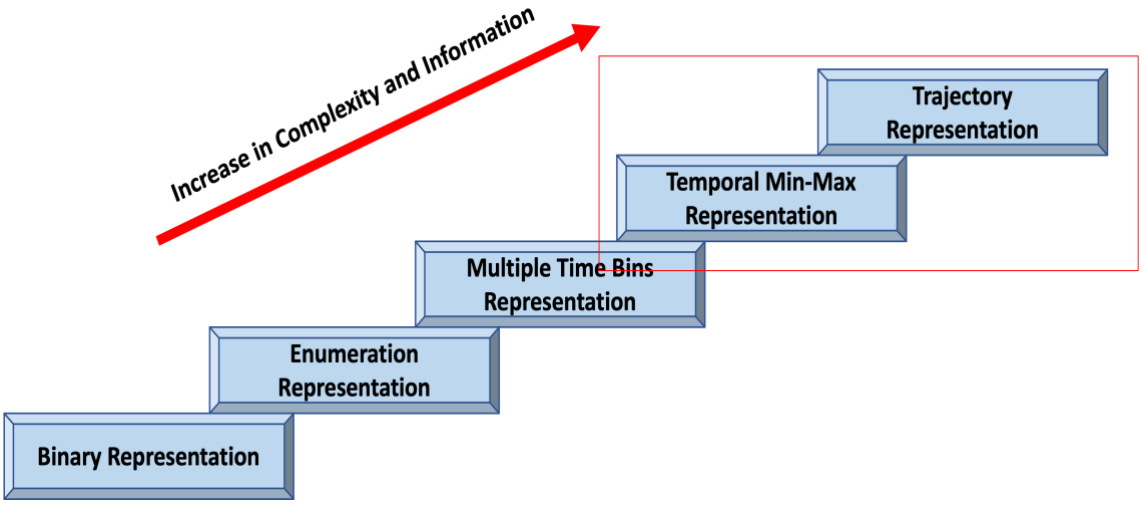

Figure 7: Administrative Codes Representation Methods

Preprocessing Methods

This chapter finishes by summarizing some of the general methods used in preprocessing of data before applying machine learning algorithm. These steps include attribute selection, attribute construction, attribute modification, and handling missing values. It should be noted that attribute modification is also explained within the attribute construction.

Attribute Selection

An effective approach to handle high-dimensional data is attribute selection, which is widely used in machine learning problems. Attribute selection refers to selecting a subset of data by removing the redundant and irrelevant attributes using a specific criterion. There are issues with using irrelevant attributes in models. Firstly, using high-dimensional data may cause curse of dimensionality meaning that the data gets sparser in space with high-dimensionality, negatively impacting the algorithms designed for low-dimensional space. Secondly, it can cause overfitting of models meaning that even though the models are highly accurate during training, they do not perform well on unseen data. Finally, when many inputs are used in data analytics, the computational associated cost increases. Attribute selection can improve computation time and learning accuracy as well as providing simpler and more understandable models (Li et al., 2017; Cai et al., 2018).

There are three main categories of attribute selection methods: Filter, Wrapper, and Embedded methods (Shown in Figure 8). The Filter methods select a subset of attributes independent of each other. Essentially, the selected attributes are selected before training the model by using information theoretic criteria or the correlation between the attributes and the outcome. Some of these methods include information gain, correlation, and chi-square test, V-score, Fisher Score etc. (Jović et al., 2015; Sondhi, 2009). The Wrapper method on the other hand, selects attributes based on learning algorithm; the attributes with the highest scores are selected after training the model. Even though the Filter methods are faster, scalable, and less computationally expensive than Wrapper methods, the selected subset might not be optimal for training the model (Sondhi, 2009). Embedded methods

apply attribute selection while training model, meaning that the training and attribute selection are applied simultaneously. These methods involve an optimization process in which an objective function both rewards the accuracy of the model and penalizes any unnecessary attributes (Kotsiantis, 2011; Sondhi, 2009). Some of the most important Embedded methods include Decision Tree algorithms such as CART, C4.5, Random Forest, Multinomial and Logistic Regression as well as attribute weighting approaches such as LASSO (L1 Regularization) (Jović et al., 2015).

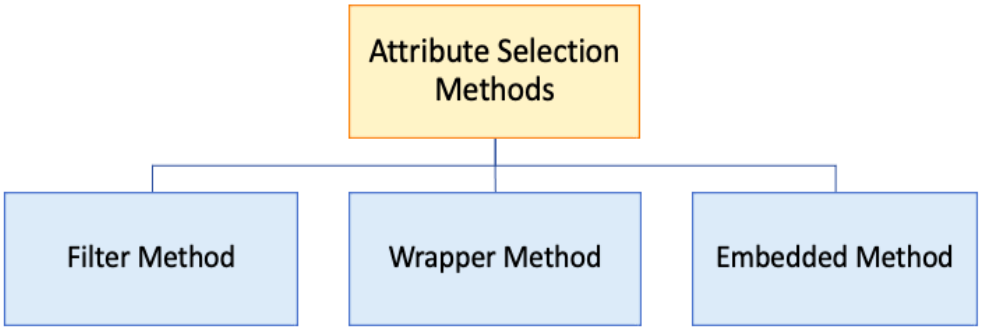

Figure 8: Attribute selection methods in machine learning

Attribute Construction

Attribute construction plays an important role in data preparation in machine learning task. The process involves building suitable attribute from existing attributes or transforming them into new forms. It includes handling categorical and continuous attributes, discretization, transformation etc. Below, some of the methods used in constructing health data are discussed in detail.

As discussed in representation of administrative codes, the most common approach is constructing structured data is using Boolean values (using 0/1 to represent

presence/absence of the value). These attributes can also be represented using enumeration representation (total number of visits, prescriptions, or admissions) (Alzoubi et al., 2019).

Occasionally, variables are measured multiple times over time in health data, including body temperature, blood pressure, etc., which can be handled by collapsing values into a single value by using the mean, mode, maximum or minimum of the first or last record. Selection of any of these methods can impact the models because they determine what kind of information to incorporate into the models (Ferrao et al., 2016).

Another method in constructing attributes is discretization. In discretization, a continuous variable is transformed into a discrete one by setting up a number of cut points within its range (Lustgarten et al., 2008). For example, a continuous variable such as blood pressure could be discretized into 'High', 'Normal' and 'Low' values, or different age groups could be determined from a continuous age attribute. Methods of discretization include supervised methods, where the output information is used to construct the discretized attribute, and unsupervised methods, in which the output information is not available or not used. While the unsupervised method has more application, the supervised one tends to build more predictive attributes (Maslove et al., 2013).

Transformation including Principal component analysis (PCA) and Normalization is another method of attribute construction. Principal component analysis method uses mathematical principles to transform large number of correlated attributes into small number of attributes called component principals. This method reduces a large number of variables to a small number of variables by using a vector space (Richardson, 2009). It is used in biomedical studies to control high-dimensional health data (Guo et al., 2016;

Jhajharia et al., 2016; Yang & Xu, 2019). The process of normalization involves scaling down values to low values, which is crucial to neural networks and K-nearest neighbors algorithms. Some of the most important Normalization methods include min-max normalization and z-score normalization (Kotsiantis et al., 2006).

Missing Values

Handling missing data is an important task in data preprocessing. There are three categories of missingness in data in statistics: Missing Completely at Random (MCAR), Missing at Random (MAR), and Missing not at Random (MNAR). Missing Completely at Random (MCAR) occurs when the probability of missingness of a data point is unrelated to that data point or other points values, while MAR happens when the probability of missingness is correlated to the data point controlling for all the other factors. Missing not at Random (MNAR) occurs when the probability of missingness really depends on the value of the data points or the unmeasured variables (Wells et al., 2013). Michaelski et al. categorize missing values differently than what is typically done in statistics. They categorized missing values into three different groups: 'Unknown', 'Non-applicable', and 'Irrelevant'. Unknown values are those that exist but are not recorded or not measured. Non-applicable means the value is not applicable to the definition of an attribute, while Irrelevant means the value is not relevant to a given task (Michalski & Wojtusiak, 2012). An example of Unknown missing value is date of death or cholesterol level, which can exist but is either not recorded or not measured. Prostate-Specific Antigen (PSA) scores for women or pregnancy for males fall into Non-applicable category, while the presence breast cancer biomarkers among other types of cancer is an example of Irrelevant missing

value. The methods for handling missing values are only applicable to Non-applicable and Irrelevant values, since they are mainly based on imputation methods (Michalski & Wojtusiak, 2012).

Missing values can be handled by removing records with missing information; while this method works on MCAR, it results in power loss and biases the results. Another method is to replace or impute missing values with data points (Wells) (Wells et al., 2013). Some of the imputation methods include simple mean imputation, Last Observation Carried Forward method in which missing value is replaced with the last observed values, Worst Observation Carried Forward method in which the missing value is replaced with the worst observed values, and multiple imputation method (Jakobsen et al., 2017).

TEMPORAL MIN-MAX REPRESENTATION

Definitions

'Temporal Min-Max Representation' (*TMMR*) represents diagnoses by using information about when the diagnosis was made for the first time and when it was made most recently (Wojtusiak et al., 2021a). This is possible as medical claims data are time-stamped and often longitudinal, thus allowing us to understand when an event occurred. In this method, the administrative codes are represented by calculating the number of days from the first known occurrence of the i-th, diagnosis or procedure code at time (t_i) to the time of prediction (t_p), named as $Code_i^{max}$ (*Max*), as well as last recorded occurrence of the diagnosis or procedure code relative to the time of prediction, named as $Code_i^{min}$(*Min*). This approach results in two values associated with each administrative code represented as the number of days. This is formalized using the below formulas.

Equation 7: Temporal Min-Max Representation

$$Code_i^{max} = \max_{t_i}(t_p - t_i)$$

$$Code_i^{min} = \min_{t_i}(t_p - t_i)$$

This method of representing the administrative codes provides information about how long a patient suffers from a given condition as well as if the condition is still present at the time of prediction (when was last time the patient diagnosed with the condition?).

The rationale behind this approach is that for many chronic conditions such as diabetes or cardiovascular disease that affect patients' health over time, it is important to know how long the condition has been present for the patient. Similarly, for many acute conditions such as falls and misuse of drugs that affect health status temporarily, only recent occurrences are important to consider. The method does not take into consideration what happens 'in between' the first and most recent occurrence of a diagnosis code, which will be later addressed in the "Trajectory Representation" chapter.

Representing Non-present Diagnoses

Since diagnoses and their corresponding administrative codes are represented by the number of days, special values need to be assigned to indicate diagnoses that are not present in patients' records. It is not reasonable to simply represent non-present diagnoses with '0' as codes are represented based on the number of days, and '0' means 'right now'. One assumption to indicate non-present codes is to use a large value (theoretically an infinity in time). Therefore, these codes can be represented with a very large positive and negative values, such as ±999999.0 (denoted as 6_9), ±99999.0 (denoted as 5_9), and ±9999.0 (denoted as 4_9) etc. Here, 10^n-1 is represented as n_9, where 'n' is the number of '9's. Positive and negative numbers also can indicate the positive (*Max* columns) and negative (*Min* columns) correlation between the number of days and predicted outcome. The main reason for selecting 10^n-1 as special values is that these numbers are easily visible when performing manual inspection of the data. In representation of non-present codes, it is assumed that the choice of the special value is impacted by the algorithm.

Figure 9 graphically shows how the proposed *TMMR* method works. The raw files consist of patient ID, claims ID, the corresponding date with the claim and codes etc. Let's assume that input attributes (diagnosis codes) are derived in year 2021 and the prediction time is '1/1/2022'. For each of the code, *TMMR* is applied resulting in two copies of each code represented by '*Min'* and '*Max'*. As shown, the first patient has two claims which are 180 days apart, while the second and third patients have only one claim. The first patient was diagnosed with $Code_1$ 180 days prior to the prediction time, and for the first time 365 days prior to the prediction time. This patient was also diagnosed with $Code_N$ only once, 180 days prior to the prediction time ($Code_N^{min}$ =$Code_N^{max}$=180). Similar interpretations can be made for the second patient; the patient was diagnosed with $Code_1$ only once 270 days prior to prediction time. Finally, the third patient was neither diagnosed with $Code_1$ nor with $Code_N$, which was represented with 999999.0 -999999.0 for *Min* and *Max* attributes, respectively.

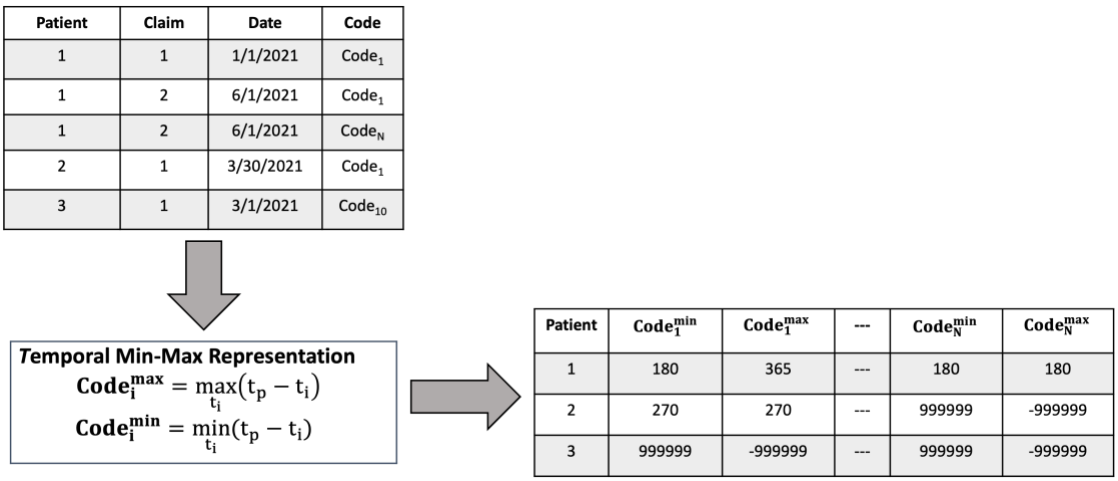

Figure 9: An Example Illustration of Temporal Min-Max Representation. Prediction time is set to 1/1/2022.

Initial Application of Temporal Min-Max Representation in Predicting Activities of Daily Living

Our group developed the *TMMR* and used it to construct models to predict Activity of Daily Living (ADL) up to one year ahead for the first time. Results clearly indicated that the representation method significantly affected results, and *TMMR* outperformed the standard Binary representation method (Wojtusiak et al., 2021a). The following sections describe how the *TMMR* was used in predicting ADLs and in constructing a decision support system called Computational Barthel Index (CBIT). The promising results that obtained from CBIT inspired large portion of this dissertation.

Assessment of Functional Disabilities

Understanding functional abilities of patients and their improvement or decline is crucial for making decisions about care provided to the elderly. For example, in a study by (Fried et al., 2002), it was reported that patients who were unlikely to return to baseline functional status were also less likely to comply with hospital treatment. It is also suggested that the quality of life matters more than living longer. Quality of life is impacted by many factors, one of which is patients' functional independence. The Functional ability assessment of patients in nursing homes is typically performed by a skilled nurse practitioner, which is a time consuming and expensive process. Such assessments are often reported through the Minimum Data Set (MDS), which is a standardized patient evaluation instrument collected by nurses through consultation with other healthcare members. In the United States, all Medicare and Medicaid-certified nursing homes collect the assessment data and enter them in MDS Section G ("MDS 3.0 Technical Information", n.d.).

Many researchers have attempted to automate the disability assessment or prediction of functional status, including Activity of Daily Livings (ADLs). In a study, machine learning methods linked to biomedical ontologies were used to predict functional status (Min et al., 2017), achieving predictive accuracy of 0.6. Using a Logistic Regression-based method, researchers predicted mortality and disability following injury for the elderly achieving the R2 of 0.86 (Jeffery et al., 2018). In one study, six standard frailty indicators (gait speed, physical activity, hand grip, body mass index, fatigue, and balance) was examined for assessing ADLs, of which only gait speed was predictive of ADL; however, no predictive accuracy was reported (Gobbens & Van Assen, 2014). Wojtusiak et al. also constructed a set of models to predict trajectories of ADL improvement or decline post hospitalization (Wojtusiak et al., 2016). Even though these studies had model performances below ones reported below, but since they were applied different settings, no direct comparison is meaningful.

As an attempt to improve state of the art in ADL prediction, decision support tool was constructed that could automatically predict ADLs 3, 6, and 12 months beyond prediction time based on a patient's demographic information and their diagnosis codes. The tool, called the Computational Barthel Index Tool (CBIT), was designed to allow for assessment of functional status at any moment and up to one year ahead. As mentioned, one novelty of this work was in using *TMMR* method in constructing the CBIT. Specifically, this method was used because many diagnoses in medical records correlate with a patient's functional ability, either temporarily or permanently. For example, some patients who undergo surgery may develop urinary incontinence temporarily, while

amputation affects ability to walk permanently.

Model Construction

Demographic information including age, race and sex, diagnoses, and functional evaluations of patients over a ten-year period were extracted from the Department of Veterans Affairs (VA) Corporate Data Warehouse (CDW). Patients with less than two evaluations were removed from the data. The data consisted of 18,912,553 inpatient and 180,123,710 outpatient diagnosis codes based on ICD-9 codes. These codes were transformed into CCS codes resulting in 281 distinct CCS codes representing health comorbidities. All diagnoses were represented by *TMMR* method (total of 562 codes), and non-present codes were replaced with ±999999.0. The final dataset consisted of total of 578 independent attributes.

Patients' evaluations recorded in Minimum Data Set (Hawes et al., 1995) were mapped to the nine Barthel Index categories. The Barthel Index (or Barthel Score) is a measure of independence in performing 10 ADLs with the total value ranging from 0 to 100 (feeding, bathing, grooming, dressing, bowels, bladder, toilet use, transfers, mobility, and stairs) (Collin et al., 1988; Shah et al., 1989). Nine out of these 10 categories (except for using stairs) were binarized (indicating any levels of disability vs. no disability) to construct nine outcomes. Barthel Score of stairs was eliminated as it was not consistently assessed, making it difficult to standardize among nursing home residents.

The data were split into 90% and 10% training and test sets, respectively. Four ML algorithms including Logistic Regression, Decision Trees, Naïve Bayes, and Random Forest were applied to construct the models with Random Forest achieving the best

performance. Hyperparameters were tuned using 10-fold cross-validation and models were calibrated using 5-fold cross-validated isonomic regression.

A total of 72 models were constructed for predicting functional status using Random Forest corresponding to four time points, nine ADLs and two different models called Evaluation/Re-Evaluation ($M_E{}^d{}_\tau$ / $M_{RE}{}^d{}_\tau$) models. Evaluation/Re-Evaluation models refer to models in which the last known functional status is present and unknown, respectively.

Results

Table 1 reports the performance of the models as the average of the performance of all 9 disabilities for each time point and for both Re-Evaluation ($M_E{}^d{}_\tau$) and Evaluation ($M_{RE}{}^d{}_\tau$) models. In general, the CBIT showed very high accuracy in assessing ADLs at a given time. The models achieved average AUC of 0.94 (0.93-0.95), accuracy of 0.90 (0.89-0.91), recall of 0.90 (0.84-0.95), and precision of 0.91 (0.89-0.92). When predicting ADLs up to one year ahead, the accuracy decreased to average AUC of 0.77 (0.73-0.79), accuracy of 0.73 (0.69-0.80), recall of 0.69 (0.34-0.96), and precision of 0.74 (0.66-0.81). Moreover, for Evaluation models, the performance decreased by about 16% ($p<0.05$) in terms of AUC. The average results of these models were AUC 0.79, accuracy 0.74, precision 0.74, and recall 0.80.

Table 1: Average+/- standard deviation of accuracy, AUC, precision and recall of models in assessing ADLs.

	Re-Evaluation Models ($M_{RE}{}^d{}_\tau$)				**Evaluation Models ($M_E{}^d{}_\tau$)**			
Prediction Time τ	**Accuracy**	**AUC**	**Precision**	**Recall**	**Accuracy**	**AUC**	**Precision**	**Recall**
Current	.900 ± .007	.947 ± .006	.910 ± .011	.907 ± .041	.743 ± .029	.795 ± .010	.743 ± .046	.800 ± .128

3 Months	.815 ± .020	.876 ± .011	.849 ± .019	.816± .094	.727 ± .037	.761 ± .006	.734 ± .049	.783 ± .161
6 Months	.759 ± .029	.808 ± .014	.784 ± .029	.737± .165	.720 ± .038	.746 ± .009	.721 ± .045	.729 ± .238
12 Months	.737 ± .035	.772 ± .022	.742± .049	.699 ± .226	.716 ± .039	.725 ± .016	.696 ± .073	.701 ± 264

Next, the average GINI Index (Breiman, 2001) produced by Random Forest was used to measure the quality of predictors. It was observed that the past functional status are the most predictive attributes, followed by the time since the most recent diagnosis of delirium, dementia, and amnestic and other cognitive disorders (CCS 653) and patient age. These top predictors along with their reported importance (average GINI score over all trees in forest and over all models) are depicted in Table 2 (Note that previous functional disabilities are not shown in this table). Other most predictive diagnoses/administrative codes included: the time since the most recent diagnosis of urinary tract infections (CCS 159); chronic ulcer of skin (CCS 199); other connective tissue disease (CCS 211); paralysis (CCS 82); administrative/social admission (CCS 255); alcohol-related disorders (CCS 660); aspiration pneumonitis; food/vomitus (CCS 129); and schizophrenia and other psychotic disorders (CCS 659). Interestingly, a combination of *Min* and *Max* attributes were among the top predictors of ADLs, emphasizing the importance of first or last occurrence of a disease depending on the type of diagnosis code. As shown in Table 2, the first occurrence of chronic ulcer of skin (CCS 199) and aspiration pneumonitis ulcers (CCS 129) were important, while for the rest of the top diagnosis codes, the last occurrence of the diagnosis seemed more significant.

Table 2: Top ranked predictors of functional status. 'GINI RE-EVAL' indicates score of a variable in Re-Evaluation models ($M_{RE}{}^{d}{}_{\tau}$). 'GINI EVAL' indicates score of a variable in in Evaluation models ($M_{E}{}^{d}{}_{\tau}$). R are potentially reversible or red flag that this person is at risk and needs restorative therapy; Race and Gender variables are included at the bottom of the table for comparison but have very low impact on prediction.

Rank	Attr.	Min/Max	Description	GINI RE-EVAL	GINI EVAL
1	CCS653	*Min*	Delirium, dementia, and amnestic and other cognitive disorders	0.0216	0.0310
2	Age		Age at the time of prediction	0.0133	0.0335
3	CCS159	*Min*	Urinary tract infections	0.0128	0.0217
4	CCS199	*Max*	Chronic ulcer of skin	0.0071	0.0121
5	CCS211	*Min*	Other connective tissue disease	0.0065	0.0091
6	CCS82	*Min*	Paralysis	0.0062	0.0110
7	CCS255	*Min*	Administrative/social admission	0.0061	0.0107
8	CCS660	*Min*	Alcohol-related disorders	0.0058	0.0110
9	CCS129	*Max*	Aspiration pneumonitis; food/vomitus	0.0055	0.0072
10	CCS659	*Min*	Schizophrenia and other psychotic disorders	0.0055	0.0089
	…		……		
337	W		Race White	0.0006	0.0012
341	UR		Unknown Race	0.0006	0.0011
365	B		Race Black	0.0004	0.0009
434	Gender		Gender	0.0002	0.0004
445	A		Race Asian	0.0002	0.0003

The above models were based on the full set of 578 input attributes. Further, a set of simplified models (called $MS_{RE}{}^{d}{}_{\tau}$ and $MS_{E}{}^{d}{}_{\tau}$) was constructed based on top 50 patient characteristics as ranked by feature importance of Random Forest models. These models did not perform statistically significantly different than the original full models.

Properties of Temporal Min-Max Representation in CBIT

One important advancement of the presented CBIT is the way the diagnosis codes were represented. Therefore, a full set of experiments was performed to determine if

TMMR is different from Binary Representation of diagnosis codes. All constructed $MR{E^d}_\tau$, $M{E^d}_\tau$, $MSR{E^d}_\tau$, and $MS{E^d}_\tau$ models were compared in terms of AUC at different time points up to one year ahead. In one experiment, Random Forest was compared with other algorithms including Logistic Regression, Decision Tree, and Naïve Bayes. As described earlier, when Temporal Representation is used, one needs to assign special values to diagnoses that are not present in data. To understand how these special values would impact models, +/- 999999 (6_9) were compared with +/-9999 (4_9), and +/-99999 (5_9) across all models. Temporal Representation (6_9) was also compared with Binary Representation to determine any significant difference in the performance of models. Two-tailed t-test was used to assess all comparisons ($p<0.05$).

As summarized in Table 3, both Random Forest and Logistic Regression showed significant difference in AUC when *TMMR* was applied ($p<0.05$). The results indicated that Random Forest with *TMMR* performs significantly better than Binary Representation. With simplified Evaluation models (AUC of 0.79 vs. 0.76 for Random Forest), where there is no information about patients' previous health status, the pure effect of representation can clearly be seen. Such relationship was, however, opposite for Logistic Regression (the Binary Representation was better). Decision Trees and Naïve Bayes results were also included in the table, but the performance was inferior. It was also observed that these special values did not impact Random Forest, Decision Tree and Naïve Bayes, while the performance of Logistic Regression was affected by these values.

These results inspired the large experimental evaluation and study of diagnoses representation methods that is the main topic of this dissertation. The detailed comparison on four example prediction problems is presented in next chapter.

Table 3: Results of valuation of Temporal and Binary Representation of diagnoses as part of CBIT construction and evaluation. The results are presented in terms of AUC for the current assessment, and prediction up to 12 months ahead. Full models that include 578 attributes and simplified models with 50 attributes are shown. Evaluation (no previous known ADL status) and Re-Evaluation (known previous status) results are presented.

	AUC	Current Assessment				3 Month Prediction				6 Month Prediction				12 Month Prediction			
		RF	LR	DT	NB	RF	LR	DT	NB	RF	LR	DT	NB	RF	LR	DT	NB
$M_{RE^d_\tau}$	***TMMR*** 4_9	0.95*	0.85^{*+}	0.92^{+}	0.87^{*+}	0.88	0.79^{*+}	0.83^{+}	0.83^{+}	0.81	0.77^{*+}	0.74^{+}	0.78^{+}	0.77	0.74^{*+}	0.70^{+}	0.74^{+}
	TMMR 5_9	0.95	0.78^{*+}	0.92^{+}	0.89^{+}	0.88	0.76^{*+}	0.83^{+}	0.83^{+}	0.81	0.74^{+}	0.74^{*+}	0.78^{+}	0.77	0.71^{*+}	0.70^{+}	0.74^{+}
	TMMR 6_9	0.95	0.78^{+}	0.92^{+}	0.90^{+}	0.88	0.75^{+}	0.83^{+}	0.83^{+}	0.81	0.74^{+}	0.74^{+}	0.78^{+}	0.77	0.72^{+}	0.70^{+}	0.74^{+}
	Binary	0.94*	0.94*	0.91^{*+}	0.87^{*+}	0.87*	0.87^{*+}	0.82^{*+}	0.80^{+}	0.81	0.81^{*+}	0.74^{+}	0.77^{*+}	0.77	0.77^{*+}	0.70^{+}	0.74^{*+}
$MS_{RE^d_\tau}$	***TMMR*** 4_9	0.95	0.94^{*+}	0.92^{+}	0.89^{+}	0.88	0.88^{*+}	0.83^{+}	0.82^{+}	0.81*	0.81^{*+}	0.74^{+}	0.76^{+}	0.77	0.77*	0.70^{+}	0.72^{+}
	TMMR 5_9	0.95	0.93^{*+}	0.92^{+}	0.89^{+}	0.88	0.84^{*+}	0.82^{+}	0.82^{+}	0.81*	0.79^{*+}	0.74^{+}	0.76^{+}	0.77	0.75^{*+}	0.70^{+}	0.72^{+}
	TMMR 6_9	0.95	0.76^{+}	0.92^{+}	0.90^{+}	0.88	0.72^{+}	0.83^{+}	0.82^{+}	0.81	0.71^{+}	0.74^{+}	0.76^{+}	0.77	0.69^{+}	0.70^{+}	0.72^{+}
	Binary	0.94*	0.94^{*+}	0.90^{*+}	0.90^{+}	0.88*	0.87^{*+}	0.81^{*+}	0.83^{+}	0.81*	0.81^{*+}	0.74^{*+}	0.78^{*+}	0.77	0.77*	0.69^{+}	0.74^{*+}
$M_{E^d_\tau}$	***TMMR*** 4_9	0.79	0.79^{*+}	0.72^{*+}	0.73^{+}	0.76	0.76*	0.68^{+}	0.68^{+}	0.75*	0.75*	0.66^{+}	0.71^{+}	0.73	0.72*	0.64^{+}	0.69^{+}
	TMMR 5_9	0.79	0.78^{*+}	0.71^{+}	0.73^{+}	0.76	0.75^{+}	0.68^{+}	0.68^{+}	0.75	0.74^{*+}	0.66^{+}	0.71^{+}	0.73	0.71^{*+}	0.64^{+}	0.69^{+}
	TMMR 6_9	0.79	0.78*	0.72^{+}	0.73^{+}	0.76	0.75^{+}	0.68^{+}	0.68^{+}	0.75	0.74	0.66^{+}	0.71^{+}	0.73	0.72^{+}	0.64^{+}	0.69^{+}
	Binary	0.78*	0.78*	0.70^{*+}	0.73^{+}	0.76	0.76*	0.67^{*+}	0.70^{*+}	0.75	0.75*	0.66^{+}	0.71^{*+}	0.72*	0.73^{*+}	0.64^{+}	0.69^{*+}
$MS_{E^d_\tau}$	***TMMR*** 4_9	0.79	0.77^{*+}	0.71^{+}	0.64^{+}	0.76	0.75*$_{+}$	0.68^{+}	0.63^{+}	0.74	0.73^{*+}	0.66^{+}	0.60^{+}	0.72	0.72*	0.63^{+}	0.58^{+}
	TMMR 5_9	0.79	0.76^{*+}	0.71^{+}	0.64^{+}	0.76	0.73^{*+}	0.68^{+}	0.63^{+}	0.74	0.72^{*+}	0.66^{+}	0.60^{+}	0.72	0.71^{*+}	0.63^{+}	0.58^{+}
	TMMR 6_9	0.79	0.75^{+}	0.71^{+}	0.64^{+}	0.76	0.72^{+}	0.68^{+}	0.63^{+}	0.74	0.71^{+}	0.66^{+}	0.60^{+}	0.72	0.69^{+}	0.63^{+}	0.58^{+}
	Binary	0.76*	0.77^{*+}	0.68^{*+}	0.74^{*+}	0.74*	0.74*	0.65^{*+}	0.71^{*+}	0.73*	0.73*	0.64^{*+}	0.71^{*+}	0.71*	0.72^{*+}	0.63^{+}	0.69^{*+}

4_9 indicates encoding of diagnoses not present in patient's history as +/-9999, 5_9 as +/-99999 and 6_9 as +/-999999. * indicates significance ($p<0.05$) of coding systems compared to '6_9' and + indicates significance ($p<0.05$) of different algorithms compared to Random Forest.

Online Decision Support System

An online decision support system was constructed based on the developed CBIT that can automatically assess current and future functional status up to one year (Wojtusiak et al., 2021b). Clinical Decision Support Systems (CDSS) are a key component of health information systems and integral part of clinical workflows (Wasylewicz et al., 2019). While most commercially available CDSS are rule-based with sets of rules manually implemented to support guidelines, there is a growing interest in integrating models created by machine learning (ML) methods as part of CDSS (Peiffer-Smadja et al., 2020). Along with triggering alerts, ML-based models are also used to predict likely outcomes and help with diagnosing patients (Belard et al., 2017).

As stated before, initial models were created using Random Forest algorithm. However, due to large size of models (each being 1GB to 2 GB, totaling about 100 GB across all models), it was infeasible to use them as part of the decision support tool. Therefore, Gradient Boost (GB)-based Models were selected to be used as part of the tool. The GB models were significantly smaller in terms of size and could be easily incorporated in the online tool. Experimental results showed that RF ad GB provide comparable results with an overall R2 =0.92 and Kappa=0.86 across all 72 models, making GB a proper alternative to RF-based models.

The web-based decision support tool (web calculator) was developed using simplified models. The system is publicly available at https://hi.gmu.edu/cbit website. It provides Web interface as well as Application Programming Interface (API). Web requests

are submitted from an HTML form, while API requests are submitted as JSON (JavaScript Object Notation). The tool consists of the following components: the CBIT models, web form to enter data, data consistency check, graphical results presentation, and explanation module.

The Web form (depicted in Figure 10) used to insert data has two sections that correspond to Evaluation and Re-Evaluation models. Previous known functional status is pre-set as fully independent, and age is pre-set to 71 indicating the average value of age in the data. Time from the first and last occurrence of selected diagnosis codes can be entered as number of days or selected from pre-populated list (last week, last two weeks, last month, last three months, last six months, last year, last three years, and more than three years).

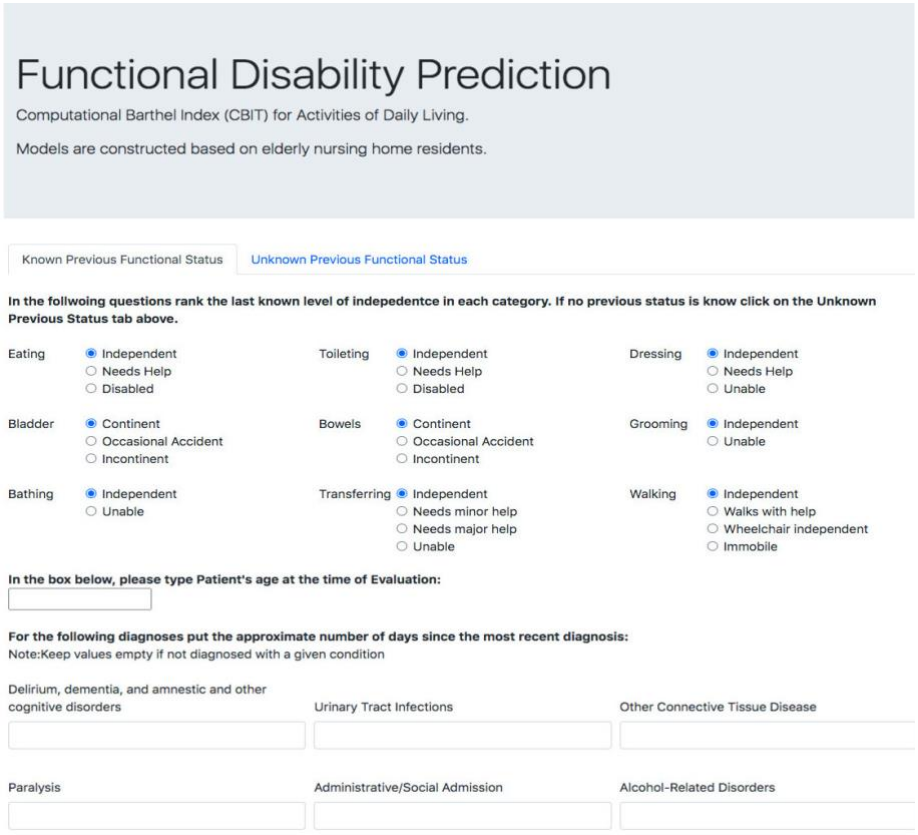

Figure 10: Part of CBIT Web Calculator screen used to enter patient characteristics. The calculator is available at https://hi.gmu.edu/cbit.

The results or application of CBIT models in the decision support system are presented graphically in Figure 11 for a hypothetical patient. The horizontal axis corresponds to time up to one year after the time of prediction. The prediction results are shown as the probabilities of full functional independence vs. any level of disability. The higher the value is, the more likely the patient is to be independent. As shown, the patient is predicted to have a high risk of not being independent in toileting (low probability < 0.3 of full independence). The probability of full independence slightly increases with time, but the risk remains high/medium. In terms of all other ADLs, the patient is predicted to

have low risk of disability (high probability > 0.7 of independence) with the risk slowly increasing with time.

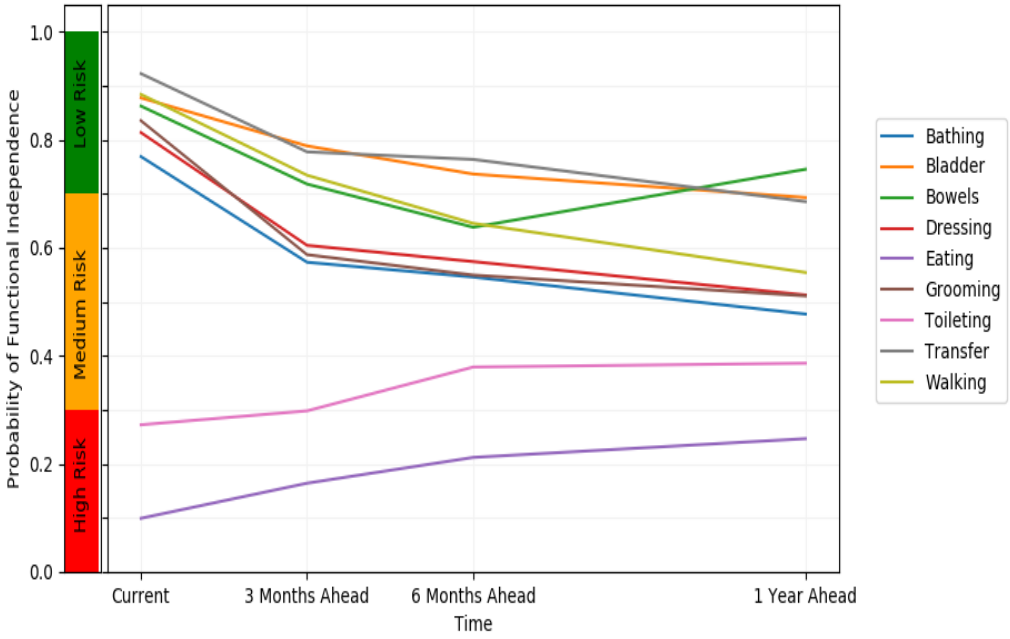

Figure 11: Visualization of the predicted ADL independence trajectories for a hypothetical patient.

The above results were among the first efforts in representing the administrative codes other than standard methods in the literature. The results clearly indicated that *TMMR* outperforms Binary Representation in predicting ADLs. In the next chapter, *TMMR* and simple Binary Representation methods will be compared in detail using a different dataset and on four different classification problems (predicting 1-year mortality, high utilization of medical services, CKD, and CHF). The comparison will be made on both population (standard model accuracy metrics) and on individual level. The choice of

the two representation methods was based on our preference but the applied concepts could be applied on other representation techniques.

EVALUATION OF TEMPORAL MIN-MAX REPRESENTAION

Datasets and Prediction Problems

The comparison of the representation methods is illustrated in terms of four supervised learning problems. Models are constructed to predict patient outcomes using medical claims data. For this purpose, four classification problems were established for predicting one-year mortality (*Problem 1*), predicting high utilization of medical services (*Problem 2*), predicting chronic kidney disease (CKD) (*Problem 3*), and predicting congestive heart failure (CHF) (*Problem 4*). Details of the prediction problems were presented in "Introduction" chapter.

In all models, the outcomes were calculated in year 2013, and all inputs were derived from data prior to 2013. The data used to construct the models were 5% control sample of Medicare beneficiaries between years 1995 and 2013. The Medicare claims collected by the Center for Medicare and Medicaid Services (CMS) provide one of the largest longitudinal datasets for the Medicare eligible population (aging population) in the United States. More specifically, in this study control group of individuals in SEER-Medicare prostate cancer data were used to construct the models. Despite its limitations (i.e., data from SEER regions, male patients only), this dataset is sufficient for the purpose of investigating data representation. The sample should be more representative for the construction of models used in clinical practice.

The patient cohort included those alive and at least 70 years old on January 1st, 2013. Excluding patients younger than 70 years guarantees that there are at least 5 years of data available prior to the prediction time as Medicare eligibility generally starts at 65.

For *Problem 1 (*Mortality*)*, a binary output/outcome attribute was created indicating whether or not the patient died in 2013. In *Problem 2 (*High Utilization*)*, a binary output attribute was created to indicate high utilization of services in 2013. Simple approach was used in which the patients were classified as high utilizers when their total number of claims was above 90-th percentile in 2013. In *Problem 3 (*CKD*)*, binary output attribute was defined based on the first occurrence of ICD-9 codes 585.* (representing CKD) in year 2013. Finally, for *Problem 4 (*CHF*)*, binary output attribute was created to indicate the presence/absence of congestive heart failure based on first occurrence of ICD-9 428.* in 2013. For the last two models, patients who were diagnosed with CKD or CHF prior to 2013 were removed from the cohorts, respectively.

In addition to the diagnosis codes that are the focus of this work, patient age and race were included in the analysis. The ICD-9 diagnosis codes were combined from multiple tables in the dataset: Medicare provider analysis and review (MedPAR), outpatient, durable medical equipment (DME), carrier (NCH), home health agency (HHA) and hospice. The ICD-9 codes were transformed into 282 AHRQ's CCS codes. Binary and Temporal Min-Max Representation methods were applied to the data, resulting in 282 diagnosis codes for Binary Representation and 564 codes for *TMMR*. The codes that were not present in medical history were replaced with ±999999.0 (6_9) in *TMMR*.

The prediction time (t_0) was January 1st, 2013, and a fixed one-year prediction window was defined to construct the output attributes ending on December 31st, 2013.

Table 4 shows the characteristics of the patients in the study population. The unit of analysis in the models was patient (each row in the test datasets corresponding to a patient). The final dataset included 83,590 patients for the first two models and 61,750 and 53,699 patients for *Problem 3* and *Problem 4*, respectively. As shown, about 10% of the population were high healthcare utilizers and about 7% of the cohort died, experienced CKD and CHF in 2013. Most patients were white and the average age in the cohort was about 79 years old.

When the observation window size is set to 18 years, the average number of days between diagnoses and time of prediction across all CCS codes is about 1847. As shown, the average number of days across all CCS codes and CCS^{min} attributes was less among positive classes. Also, this average number of days across CCS^{max} groups was slightly lower among positive class in *Problem 1* and *Problem 2* and higher in *Problem 3* and *4*. In addition, the average number of present distinct CCS codes across four problems is about 44.42 with the total number of codes being higher among positive labels.

Table 4: Characteristics of the study population. Tot, Pos and Neg correspond to all, positive and negative cases, respectively.

	Prob 1			***Prob 2***			***Prob 3***			***Prob 4***		
	Tot	**Pos**	**Neg**	**Tot**	**Pos**	**Neg**	**Tot**	**Pos**	**Neg**	**Tot**	**Pos**	**Neg**
N	83590	6111	77477	83590	8401	75189	61750	4129	57621	53699	3502	50197
%		7.32	92.68		10.05	89.95		6.69	93.31		6.52	93.48
Race% White	82.67	82.06	82.71	82.67	82.68	82.67	84.37	80.72	84.64	83.22	82.72	83.26

	Black	6.96	8.49	6.84	6.96	8.56	6.87	5.58	8.23	5.59	6.51	7.42	6.45
	Asian	4.09	4.02	4.10	4.09	3.58	4.15	3.72	4.31	3.68	3.90	4.11	3.89
	Native	0.39	0.54	0.38	0.39	0.39	0.39	0.39	0.03	0.39	0.41	0.71	0.39
	Hispanic	2.83	2.45	2.86	2.83	2.61	2.85	2.73	3.25	2.70	2.64	2.37	2.66
	Unknown	3.05	2.43	3.10	3.05	2.17	3.15	3.02	3.15	3.00	3.31	2.66	3.35
Age		79.7	82.6	79.5	79.72	80.3	79.7	79.3	80.4	79.23	78.8	80.2	78.7
CCS^{total}		1846.2	1693.9	1858	1846.2	1612.7	1872.6	1886.4	1856.8	1888.4	1810.7	1770.9	1813.3
CCS^{max}		2312.4	2230.8	2318.8	2312.4	2252.5	2319.3	2318.6	2352.7	2316.3	2212	2242.5	2210
CCS^{min}		1379.9	1157	1397.2	1379.9	971.9	1426	1454.2	1361	1460.5	1409.4	1299.3	1416.7
# Code		47.8	57.2	47	47.8	70.8	45.2	42.79	44.37	42.7	39.4	43	39.1

The models were developed using the standard model construction methods. The data were first split into 10% test set and 90% training set with the testing portion set aside for final validation. Ten-fold cross validation was used to tune model hyperparameters. After tuning final models were constructed using the entire 90% training dataset and validated using the 10% test dataset set aside. The quality of the models was measured using the standard machine learning measures including accuracy, area under the curve (AUC; often referred to as C-statistic), recall and precision, as well as sensitivity.

Four machine learning classification algorithms, Random Forest (RF), Gradient Boost (GB), Logistic Regression (LR) and Decision Tree (DT), were used to construct the models. For each algorithm, default parameters provided by scikit-learn (0.21.3) in Python 3 were tuned to develop the models.

Model Performance

The first set of experiments was to compare the performance of the models when diagnoses were constructed using standard Binary and Temporal Min-Max

Representations. For this purpose, the observation window size was set to 18 years. Table 5 presents a summary of the performance of the models in terms of AUC, accuracy (Acc), precision (Prec), and recall (Rec). Two tailed paired t-test was used to determine the level of significance ($p<0.05$). As summarized, *TMMR* performed statistically significantly better than Binary Representation ($p<0.05$) for most of the four criteria, except for LR and DT for *Problem* 2, 3 and 4. Overall, the results suggest that the *TMMR* can improve the quality of the predictive models. In predicting the four outcomes, GB achieved the highest performance with the average AUC ranging between 0.66 and 0.85. As shown, recall was low in the models due to imbalanced data. However, it is important to note that the purpose of this dissertation was not to develop the best models with optimized parameters, but to systematically compare different diagnosis representation methods for supervised learning. It is also possible that adding more attributes to the data (i.e., provider information) could improve the overall accuracies of the models.

Table 5: Average AUC, accuracy, precision and recall of the models for Temporal Min-Max Representation (*TMMR*) vs. Binary Representation (*BIN*).

***Problem 1*-Mortality**								
	TMMR				***BIN***			
Alg	**AUC**	**Acc**	**Prec**	**Rec**	**AUC**	**Acc**	**Prec**	**Rec**
RF	.767*	.927*	.605*	.025*	.735	.926	.312	.003
GB	.794*	.928*	.579*	.084*	.767	.927	.467	.014
LR	.765*	.926	.436	.032*	.759	.926	.415	.021
DT	.575*	.874*	.183*	.208*	.550	.865	.140	.164
***Problem 2*-High Utilization**								
RF	.845*	.911*	.748*	.179*	.787	.902	.656	.060
GB	.853*	.914*	.682*	.263*	.803	.903	.603	.115

LR	.821*	.905*	.595	.160*	.801	.903	.578	.118
DT	.628*	.859*	.315*	.344*	.574	.836	.221	.250
***Problem 3*-CKD**								
RF	.637*	.933*	.383	.010	.619	.932	.324	.011
GB	.673*	.933*	.194*	.003*	.663	.933	.00	.00
LR	.641*	.933*	.167	.001	.637	.933	.00	.00
DT	.538	.866	.089	.109	.535	.865	.085	.103
***Problem 4*-CHF**								
RF	.630*	.934*	.166	.002	.609	.934	.148	.002
GB	.663*	.934*	.233	.003	.644	.935	.15	.001
LR	.622	.935	.100	.000	.622	.935	.000	.000
DT	.528	.869	.082	.098	.529	.870	.083	.100

Method for Detailed Analysis of Models on Individual Level

The following experiments aimed at understanding individual-level differences between the data representations. Following the methodology presented by Wojtusiak (Wojtusiak, 2021) and later extended by Wojtusiak and Asadzadehzanjani (Wojtusiak & Asadzadehzanjani, 2022), it is not sufficient to compare models purely based on their statistical performance measures. A detailed understanding of models' behavior and properties is needed. There is surprisingly little literature that present frameworks for comparing ML models. When searching for published literature on comparison of ML models, all papers that appear are comparing specific models (or algorithms) for solving specific problems at hand. Virtually all of them report only some statistical measures discussed earlier in "Introduction" chapter. Similarly, large number of 'data science' websites discuss practical aspects of comparing models, including examples of source code, but also limit these comparisons to statistical accuracy measures. Some approaches

to comparing models are available in other fields, including work by Lee and Sangiovanni-Vincentelli (Lee & Sangiovanni-Vincentelli, 1998) who presented a general framework for comparing computation methods.

While AUC, accuracy, recall, and precision are good overall metrics of model performance, they are insufficient to understand why models perform differently for individual cases. Comparison models on individual level allows to capture differences in models that are missed in population-level comparison; this would help identify cases for which that a specific model works best. The individual level comparison of models is categorized into two groups: Output and Input Comparison, which will be explained below.

There are three requirements associated with model comparison: (1) the comparison should be applied on the same cases; this means that the cases should be extracted from either the same database or different databases linked by a common identifier; (2) the models should have the same unit of analysis meaning that each row corresponds to the same object; (3) the compared models should have the same output. The third condition can however be relaxed to some degree.

Even though calibration curves allow for output comparison at different levels, like the standard statistical methods, they do not investigate model performance on cases-by-case basis. Model Correlation Plots (MCPs) is an Output Comparison method that allow for visual case-by-case comparison of model outputs (Wojtusiak et al., 2017). The MCPs are scatterplots with axis corresponding to outputs of two models, and points representing individual cases (i.e., patients) for which predictions are made. If two models are identical, all points are located at the diagonal. Further, MCPs encode true class by color or symbol.

The corresponding datapoints in MCPs could also be aggregated and reported as tables in Output Comparison. Finally, comparing the distribution of the correct classified cases for each of the models is another Output Comparison method, which will be discussed in the 'Output Comparison' section.

Comparing model outputs allows for visually or statistically inspecting differences between models on individual cases. However, one needs to understand if there are any patterns within input values that correspond to differences in outputs of the compared models. In other words, are there patterns in input values that correspond to outputs visualized in model correlation plots?

The patterns can be described in terms of attributes present in the data or derived from them. Results can be presented visually or in the form of a data table. Let TS^+ be a set of positive cases in TS and TS^-be a set of negative cases in TS. Let's consider now four subsets of the testing set:

Equation 8: Better vs. Correct Prediction

$$CPM_1 = \{x \in TS^+ : M_1(x) \geq \tau \wedge M_2(x) < \tau\}$$
$$CNM_1 = \{x \in TS^- : M_1(x) < \tau \wedge M_2(x) \geq \tau\}$$
$$SPM_1 = \{x \in TS^+ : M_1(x) \geq M_2(x) + \varepsilon\}$$
$$SNM_1 = \{x \in TS^- : M_1(x) < M_2(x) - \varepsilon\}$$

CPM_1 and CNM_1 are respectively positive and negative cases correctly classified by model M_1 but not model M_2. SPM_1 and SNM_1 are positive and negative cases better classified by model M_1. Superior (Better) prediction is defined based on higher output

probabilities for one model vs. other model on positive cases and lower output probability on negative cases (output probability closer to the correct 0/1 label). Better prediction does not tell us, which model makes correct prediction, but rather if it is performing better in classifying positive and negative labels. These four sets can be compared in terms of values of input attributes. The sets CPM_2, CNM_2, SPM_2, and SPM_2 are defined analogously for results superior by model M_2.

The following experiments gives us case-by-case insight into model's performance. Output comparison of the representation methods will be explained in Comparison of "Output Probability" section and Input Comparison is explained in the rest of the sections.

Output Comparison

Comparison of Output Probabilities

A detailed comparison was made on the output probabilities of the models. First, the average output probability was compared between *TMMR* vs. Binary Representation methods across positive and negative labels. Since the distribution of the output probabilities across the two models were not normally distributed, Wilcoxon signed-rank test was used to compare the results. As shown in Table 6, the average output probability among cases with positive labels was significantly higher across Temporal Min-Max Representation-based models except for GB and DT of *Problem 4* for which the results were not significant. Conversely, *TMMR* had overall lower output probabilities among negative label cases except for RF in predicting renal failure for which the pattern was opposite. (The results were not significant using DT in *Problem 3* and RF and DT in *Problem 4*). Overall, higher probabilities of positive labels and lower probability of

negative labels in Temporal Representation suggest that this method is generally more likely to correctly classify both classes. Also, higher recall in Temporal Representation of diagnoses suggests that the method allows the algorithms to select more positive cases that are missed in Binary Representation method, thus leading to overall higher recall.

Table 6: Output probability comparison for Temporal Min-Max Representation (*TMMR*) vs. Binary Representation (*BIN*) on all cases.

***Problem 1*-Mortality**				
	Positive Label		**Negative Label**	
Alg	***TMMR***	***BIN***	***TMMR***	***BIN***
RF	0.1859*	0.1459	0.0758*	0.0765
GB	0.1883*	0.1452	0.0637*	0.0674
LR	0.1579*	0.1487	0.0665*	0.0673
DT	0.2134*	0.1704	0.0753*	0.0814
***Problem 2*–High Utilization**				
RF	0.3040*	0.2300	0.0879*	0.0946
GB	0.3282*	0.2385	0.0748*	0.0850
LR	0.2680*	0.2414	0.0817*	0.0848
DT	0.3444*	0.2506	0.0843*	0.0990
***Problem 3*-CKD**				
RF	0.1185*	0.1062	0.0773*	0.0750
GB	0.0891*	0.0863	0.0654*	0.0655
LR	0.0866*	0.0847	0.0656*	0.0657
DT	0.1319	0.1197	0.0824	0.0833
***Problem 4*-CHF**				
RF	0.1090*	0.0957	0.0760	0.0731
GB	0.0851	0.0800	0.0638*	0.0643
LR	0.0810*	0.0799	0.0642*	0.0643
DT	0.1194	0.1094	0.0806	0.0801

The data summarized in Table 6 are also shown graphically in the form of Model Correlation Plots (MCP) in Figure 12 depicting 1000 randomly selected patients from the test set. In this example, values on axes represent output probabilities from models (vertical Binary, horizontal Temporal Min-Max). Green points represent negative cases (no death or no high utilization, no CKD and no CHF) and red represent positives (death or high utilization, CKD and CHF) according to real labels. The plots were created for RF, GB, and LR algorithms which give outputs in the form of probabilities, thus are not applicable to DT which is a symbolic classification method.

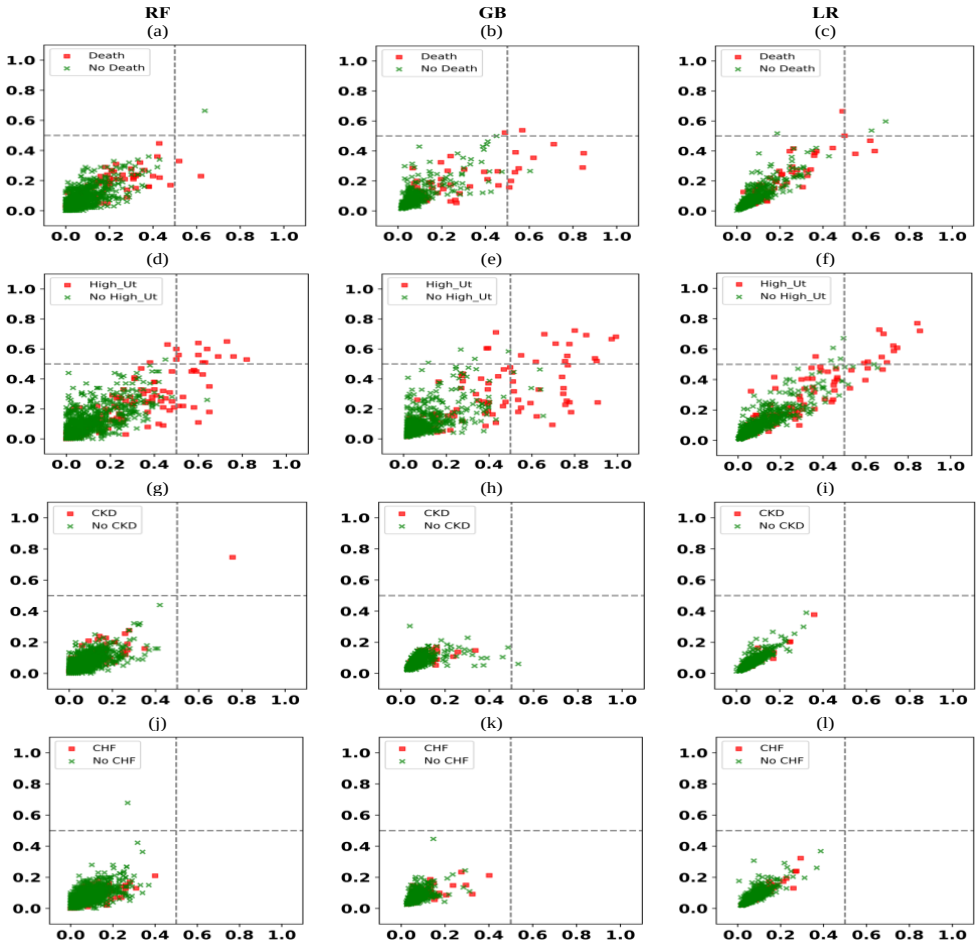

Figure 12: Comparison of the output probability of *TMMR* vs. Binary Representation that shows weak correlation. Vertical and horizontal axes show Binary and Temporal Representation,

respectively. Plots (a), (b) and (c) are for *Problem 1*, plots (d), (e), (f) for *Problem 2*, plots (g), (h), (i) for *Problem 3*, plots (j), (k), (l) for *Problem 4*.

The models showed an overall medium or high agreement between models based on Binary and Temporal Representations. The average R2 across all three algorithms and among all cases and cases with positive and negative labels are as follows: *Problem 1*: (All: 0.61, Positive: 0.60, Negative: 0.58), *Problem 2*: (All: 0.66, Positive: 0.66, Negative: 0.59), *Problem 3*: (All: 0.56, Positive: 0.62, Negative: 0.54), *Problem 4*: (All: 0.50, Positive: 0.50, Negative: 0.49). The overall agreement is highest in predicting high utilization (*Problem 2*) and lowest in predicting CHF (*Problem 4*). Also, the correlation tends to be higher in positive labels.

Despite high correlation coefficient, there are clear differences in the models' predictions. For RF and GB algorithms, there was a visible shift of values to the right of the plots, indicating that the Temporal models output overall higher values. This observation is clearer in predicting mortality and high utilization as the other two models had smaller output probabilities for both *TMMR* and Binary methods as shown in Table 6.

Distribution of Cases Between the Two Representation Methods

To further investigate differences between *TMMR* and Binary methods, the distribution of the cases that are correctly classified or better predicted by one of the representation methods were calculated. The results are reported as percentage in Table 7 and Table 8. Based on correction prediction definition (Table 7), positive cases tend to be correctly captured by Temporal Representation, while negative cases tend to be correctly classified by Binary Representation. Superior prediction table, however, indicates that most cases

were better captured by Temporal Representation method regardless of true labels of cases except for a few algorithms across the four prediction problems. It should be noted that the sum of values in superior comparison for both Temporal and Binary methods (Table 8) may not be always 100% as there are some cases with similar output probability.

Table 7: Comparison of the distribution of cases for Temporal Min-Max Representation (*TMMR*) vs. Binary Representation (*BIN*) on correct prediction.

***Problem 1*-Mortality**				
	Positive Label		**Negative Label**	
Alg	***TMMR***	***BIN***	***TMMR***	***BIN***
RF	94.70%	5.30%	17.58%	82.42%
GB	93.37%	6.63%	16.14%	83.86%
LR	77.95%	22.05%	32.91%	67.09%
DT	57.82%	42.18%	52.38%	47.62%
***Problem 2*-High Utilization**				
RF	90.43%	9.57%	29.44%	70.56%
GB	88.04%	11.96%	34.01%	65.99%
LR	75.83%	24.17%	35.35%	64.65%
DT	63.06%	36.94%	54.94%	45.06%
***Problem 3*-CKD**				
RF	45.45%	54.55%	75.61%	24.39%
GB	100.00%	0.00%	9.68%	90.32%
LR	100.00%	0.00%	8.33%	91.67%
DT	53.55%	46.45%	50.27%	49.73%
***Problem 4*-CHF**				
RF	0.00%	100.00%	81.25%	18.75%
GB	76.92%	23.08%	13.16%	86.84%
LR	100.00%	0.00%	0.00%	100.00%
DT	53.07%	46.93%	49.75%	50.25%

Table 8: Comparison of the distribution of cases for Temporal Min-Max Representation (*TMMR*) vs. Binary Representation (*BIN*) on superior prediction.

***Problem 1*-Mortality**				
	Positive Label		**Negative Label**	
Alg	***TMMR***	***BIN***	***TMMR***	***BIN***
RF	59.25%	34.78%	48.43%	40.16%
GB	56.75%	43.25%	52.70%	47.30%
LR	56.79%	43.21%	46.30%	53.70%
DT	16.12%	11.84%	6.88%	6.26%
***Problem 2*–High Utilization**				
RF	69.15%	27.12%	51.64%	34.69%
GB	65.94%	34.06%	60.20%	39.80%
LR	63.11%	36.89%	56.62%	43.38%
DT	22.65%	13.27%	8.27%	6.78%
***Problem 3*-CKD**				
RF	51.51%	41.44%	46.84%	43.31%
GB	46.79%	53.21%	49.90%	50.10%
LR	54.83%	45.17%	46.44%	53.56%
DT	9.23%	7.94%	7.03%	6.90%
***Problem 4*-CHF**				
RF	50.11%	41.92%	47.36%	42.60%
GB	47.77%	52.23%	53.81%	46.19%
LR	52.17%	47.83%	46.73%	53.27%
DT	9.34%	8.45%	6.83%	6.81%

Input Comparison

Days Between Diagnosis and Prediction Time

The following experiment was conducted as part of *Input Comparison* to understand what patterns within inputs correspond to changes in the outputs based on the two representation methods. The average number of days between diagnosis occurrence and prediction time

across all diagnosis codes (*Min* and *Max* attributes) was compared between Temporal and Binary Representations. In *TMMR*, the diagnosis codes are represented by calculating the time from the first and last occurrence of the diagnosis to the prediction time (represented with *Min* and *Max* attributes of each diagnosis code). Therefore, the average of time to diagnosis across all CCS codes (both *Min* and *Max* attributes) were compared on cases that are either correctly or better predicted by one of the representation methods and the results are shown in

Table 9 and Table 10. Due to non-normal distribution of the data, Mann-Whitney U test was used instead of the t-test for comparing results. A general observation was that the average number of days was smaller for Temporal Representation among cases with positive labels and larger among cases with negative labels in most comparisons. The results were conclusive and significant in *Problem 1* and *Problem 2* for both correct and superior prediction. In terms of superior prediction, similar pattern was observed among GB and LR algorithms in predicting CHF and CKD, while it was opposite for RF in these two outcomes. However, the results were not comparable when comparing correctly classified cases in predicting these outcomes; this is because either there were no correctly predicted cases (represented by N/A in the tables) in one of the representation methods or no significant difference between the values. The results overall suggest that *TMMR* representation tends to better classify positive cases who had health issues for a shorter period, while it tends to better predict true negative cases who were sick longer. A related issue is further investigated in "Model History Length" section in which the relationship

between observation window sizes (history length) and model performance will be examined.

Table 9: Comparison of the average number of days for *TMMR* vs. Binary Representations on correct prediction.

***Problem* 1-Mortality**				
	Positive Label		**Negative Label**	
Alg	***TMMR***	***BIN***	***TMMR***	***BIN***
RF	1259.6*	2599.1	2405.3*	1243.4
GB	1353.6	1413.3	1898.0*	1380.5
LR	1181.7*	1906.9	2608.1*	1193.3
DT	1595.1*	1680.8	1981.4*	1838.6
***Problem* 2–High Utilization**				
RF	1432.5*	1913.5	2070.2*	1555.0
GB	1468.2*	1754.1	1834.7*	1525.5
LR	1184.2*	2195.7	2362.4*	1213.2
DT	1579.9*	1669.6	1944.5*	1754.2
***Problem* 3-CKD**				
RF	3933.5	2426.3	2568.4	1751.9
GB	1340.5	N/A	2182.1	1977.3
LR	1871.6	N/A	1612.9	1572.3
DT	1929.1	1863.2	1959.7	1964.2
***Problem* 4-CHF**				
RF	N/A	2693.0	2444.8	1245.7
GB	1849.8	1983.4	2080.6	1751.1
LR	2248.5	N/A	N/A	1660.8
DT	1794.3	1793.7	1886.8	1868.4

Table 10: Comparison of the average number of days for *TMMR* vs. Binary Representations based on superior prediction.

***Problem* 1-Mortality**

	Positive Label		Negative Label	
Alg	***TMMR***	***BIN***	***TMMR***	***BIN***
RF	1641.26*	1771.76	1920.35*	1816.88
GB	1549.92*	1896.64	2004.06*	1699.10
LR	1510.49*	1933.96	2103.39*	1643.55
DT	1600.81*	1682.96	1977.14*	1842.69
***Problem* 2–High Utilization**				
RF	1538.88*	1795.15	2048.76*	1644.65
GB	1497.31*	1833.94	2055.30*	1587.85
LR	1338.79*	2082.75	2188.95*	1459.63
DT	1579.53*	1669.15	1942.51*	1755.21
***Problem* 3-CKD**				
RF	1876.77*	1829.25	1888.13*	1914.82
GB	1656.12*	2062.27	2118.64*	1665.69
LR	1603.46*	2159.21	2232.56*	1586.16
DT	1915.05	1847.00	1959.98	1963.72
***Problem* 4-CHF**				
RF	1805.54*	1734.04	1815.74*	1835.75
GB	1545.74*	2003.89	2046.07*	1551.52
LR	1477.50*	2083.87	2158.46*	1506.57
DT	1807.58	1815.79	1886.77	1882.89

Number of Present Health Conditions

Similar experiment was conducted to compare the number of present diagnosis codes for Temporal vs. Binary Representation methods. This experiment is intended to determine how the representation methods are impacted by the number of present diagnosis codes resulting in different prediction. For this purpose, the average number of present diagnoses (CCS codes) was compared for the two representation methods based on correct and superior prediction and the results were shown in Table 11 and Table 12, respectively.

In general, the number of present health conditions was larger or smaller depending on what is being predicted, algorithm, and output class when comparing correctly classified cases; this means that no specific pattern was observed for each of the representation methods.

The results were more conclusive based on superior prediction; the average number of codes tends to be larger for Temporal Representations among cases with positive labels and smaller among negative ones for most algorithms across four prediction problems. Therefore, the larger number of present codes (sicker patients) in predicting positive cases suggests that sicker patients (patients with more diagnosis codes) are better predicted with Temporal Representation. Conversely, sicker patients are better captured with Binary Representation in predicting negative labels. These results can be interpreted as when using the Temporal Representation in predicting positive cases, patients need to be more severely sick with larger number of conditions present. In general, in Binary Representation, each of the conditions present in a patient's record provides an incremental increase to the predicted probability. In contrast, in *TMMR*, individual diagnoses can have stronger impact as well as non-linear relationship with the predicted outcomes because of the time information available.

Table 11: Comparison of the number of present codes for *TMMR* vs. Binary Representations on correct prediction

***Problem* 1-Mortality**				
	Positive Label		**Negative Label**	
Alg	***TMMR***	***BIN***	***TMMR***	***BIN***
RF	83.0*	1.1	10.5*	77.3

GB	77.9*	87.0	85.8*	78.5
LR	80.0	86.5	84.9*	78.8
DT	68.9*	65.0	57.8*	59.5
***Problem* 2–High Utilization**				
RF	85.3*	103.7	101.5*	88.9
GB	79.7*	97.0	96.8*	80.8
LR	85.8*	99.7	98.1*	85.6
DT	76.9*	78.6	63.6	63.9
***Problem* 3-CKD**				
RF	0.4*	1.0	1.6	1.9
GB	74.7	N/A	78.0	77.5
LR	76.3	N/A	73.0	60.1*
DT	58.4*	54.2	50.3	53.7*
***Problem* 4-CHF**				
RF	N/A	0.8	0.8	1.3
GB	85.5*	77.7	90.8	72.7*
LR	94.0	N/A	N/A	66.0
DT	56.8*	52.1	47.8	49.8*

Table 12: Comparison of the number of present codes for *TMMR* vs. Binary Representations on superior prediction

***Problem 1*-Mortality**				
	Positive Label		**Negative Label**	
Alg	***TMMR***	***BIN***	***TMMR***	***BIN***
RF	60.68*	52.52	46.25*	50.13
GB	60.52*	52.92	47.71*	46.23
LR	57.35	57.08	49.19*	45.13
DT	68.46*	64.24	56.89*	58.49
***Problem 2*–High Utilization**				
RF	71.74*	70.50	47.24*	50.83
GB	70.70	70.99	49.22*	39.08
LR	69.72*	72.65	44.79*	45.71

DT	76.84*	78.54	62.85	63.19
***Problem 3*-CKD**				
RF	48.86*	38.88	40.58*	46.15
GB	51.73*	37.89	42.72	42.63
LR	44.32	44.42	44.44*	41.15
DT	56.73*	53.05	49.37	53.21
***Problem 4*-CHF**				
RF	47.73*	37.93	37.03*	42.87
GB	48.70*	37.83	37.72*	40.77
LR	42.56	43.52	40.38*	38.04
DT	55.67*	49.83	46.56*	49.08

Model History Length (Back Window Size)

Does the history matter? This set of experiments was to investigate the impact of the observation window size (length of patient history) on the quality of models when applying the Binary vs. Temporal Min-Max Representation methods. Intuitively, longer windows allow for inclusion of more diagnosis codes present in the patients' history. However, when Binary Representation is used, large window size may cause inclusion of diagnoses (codes) that are no longer relevant (i.e., acute conditions). In contrast, when *TMMR* is used, data with irrelevant timeframe can be adjusted by the model itself. The presented discussion assumes fixed window size across all diagnoses. However, initial results have shown that this assumption is an over-simplification since period of relevance may depend on specific diagnoses. In general, an optimal window size should be optimized for each diagnosis separately, yet that may not be practical due to computational complexity of the problem.

For this purpose, diagnosis codes were extracted 1 year, 2 years, … 12 years as well as 18 years prior to the prediction time to allow for investigating the impact of the amount of information on model performance. As illustrated in Figure 13, the grey area indicates the size of observation window ranging from 18 years to 1 year and the fixed-size window (1 year) on the right of each axis shows the prediction window. For each observation window size, the models were developed for the four outcomes, four algorithms and two representation methods.

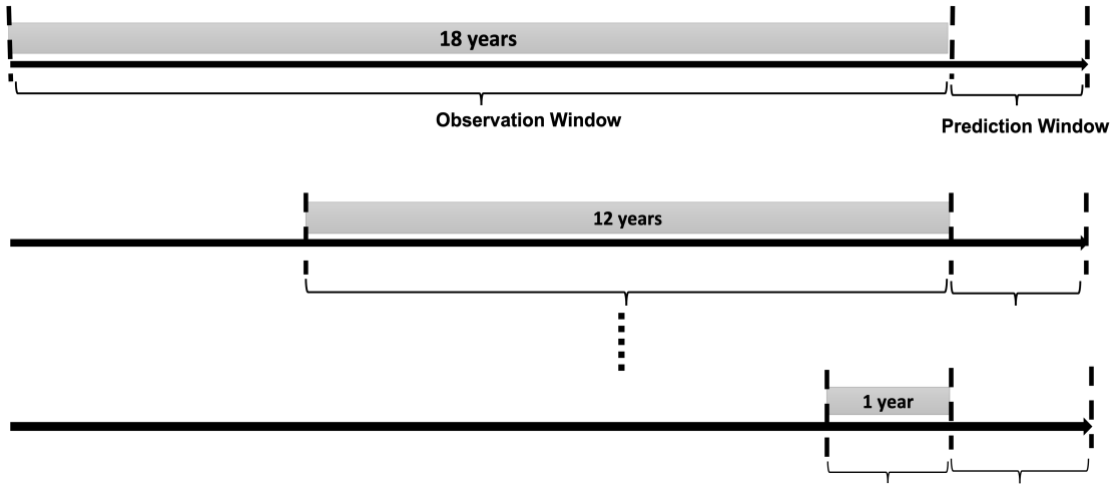

Figure 13: Illustration of temporality in diagnosis codes extraction

The results are summarized on Figure 14 that graphically shows the AUC of the constructed models. The vertical axis refers to the AUC of the models and horizontal axis represents the size of the observation window ranging from 18 years to 1 year. Similar plots were generated for accuracy, precision, and recall, but they are not included here due

to space limitation. Red and black lines correspond to the changes in AUC for Temporal vs. Binary methods. An interesting observation was that the changes in observation window size affect the quality of the models differently across the four models, four algorithms and the two representation methods. The results suggest that one needs to carefully pick the optimal size of observation window with respect to the algorithm, outcome, and the representation method to improve the quality of the models. The AUC ranged between about 0.53 and 0.76 in *Problem 1,* 0.54 and 0.85 in *Problem 2,* 0.50 and 0.67 in *Problem 3,* and between 0.48 and 0.66 in *Problem 4.*

In general, it was observed that *TMMR* outperformed Binary method in most observation window sizes. However, in some algorithms or observation window sizes, Binary Representation outperformed Temporal Representation or have equal performance. The AUC change pattern was similar for RF and GB across *Problem 1, 3* and *4.* In these models, the accuracy of Temporal-based models increased with more amount of data achieving the highest AUC when the window size was 18 years (the longest that can be constructed from available data). This suggests that in predicting these three outcomes, it is important to know what happened in patient's medical history long time ago and that *Max* columns should have high ranking. In *Problem 2,* however, it was observed that the accuracy of the models does not depend on the amount of data as it was almost constant over the course of 18 years; this suggest that the most recent diagnoses or *Min* columns should have high rank in predicting high utilization. As an exception, it was observed that the AUC of DT-based model in predicting high utilization drops with less amount of data. However, since AUC is not a good metric for assessing DT, the results are not reliable for

comparing. Using Binary Representation for RF and GB algorithms, the performance of the models increased with less amount of data in predicting mortality and high utilization achieving the best performance by having about three years of data in *Problem 1* and only one year in *Problem 2*. However, in *Problem 3* and *4*, the performance of Binary-based models dropped by shrinking the size of the observation window. For Decision Tree (DT), the accuracy dropped with less amount of data for Temporal and Binary Representations across four prediction problems. Additionally, when LR was used, the pattern varied based on the prediction problem and the representation method. Finally, it was shown that there was a large difference between the two representations between *Problem 1* and *Problem 2*, while these two methods did not really differ in predicting CKD and CHF.

In summary, it is clear that the relationship between the window size and model performance depends on the specific type of algorithm used as well as the outcome being predicted. When predicting mortality, symbolic methods (RF, GB, DT) performed better with more data available and *TMMR* representation was used. When Binary Representation was used, RF and GB peak at about 3 years of data. LR models on the same data preferred smaller window sizes, with peak at 3 years again. Different shapes were observed for other prediction problems, indicating that amount of data depends on the outcome predicted.

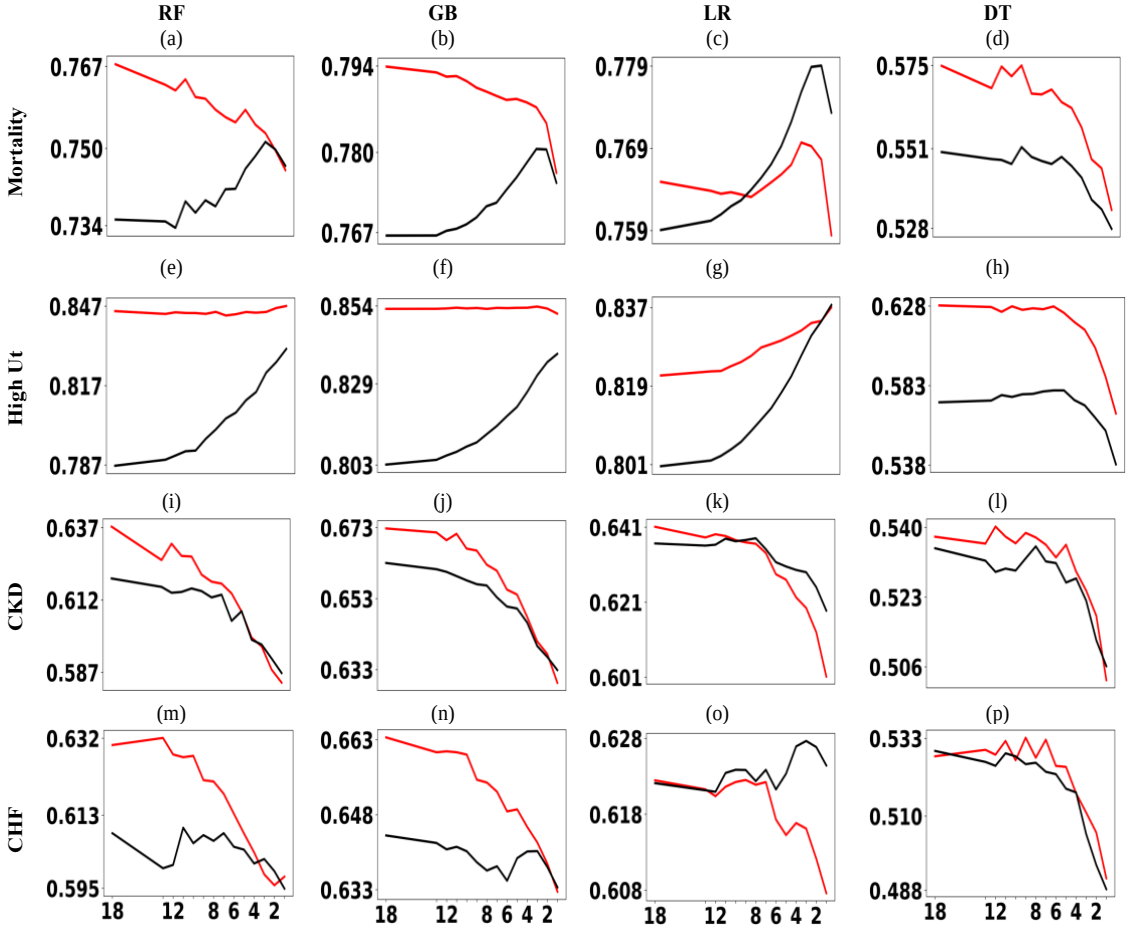

Figure 14: Comparison of the Temporal (red) vs. Binary (black) Representation for the four problems. Vertical and horizontal axes show AUC and observation window size, respectively. Different scales on the sub-plots are irrelevant because the focus is on presenting shapes of the curves.

Distribution Importance of Chronic and Non-chronic Diagnoses

As described along with the definition of *TMMR*, the choice between *Min* and *Max* attributes is to capture the first occurrence of the diagnosis which may typically be important among chronic conditions and last occurrence of the diagnosis, which may be more important for acute conditions. In the previous section, it was observed that RF, GB, and DT had better AUC in predicting mortality, CHF, and CKD in larger observation window size while the change in observation window size did have small impact in high

utilization prediction. In predicting high utilization, the change was constant in RF and GB over the course of 18 years, while the AUC of DT-based model dropped in smaller window sizes. Therefore, it was suggested that time from the first visit (*Max* attributes) tends to have high ranking for the first three models compared to the last model, meaning that there should be more *Max* attributes as top predictors of mortality, CKD and CHF compared to high utilization. To test this observation, average Gini scores across the 10 training sets were calculated to measure the quality of the predictors. Table 13 shows the top CCS^{max} attributes among the top 40 attributes for each of the models. As shown, there were more *Max* attributes in *Problem 1*, *Problem 3*, and *Problem 4*, while there seems to be few *Max* attributes in predicting high utilization. The results agree with what was observed in previous section, that in most models some information about when diagnoses were present for the first time is important, while in predicting high utilization recent data is more important.

In addition, the Chronic Condition Indicator (CCI) from the AHRQ was used to distinguish Chronic vs. Non-chronic CCS conditions. The tool essentially categories the ICD-9 CM diagnosis codes into CCS categories, that were later assigned chronic/non-chronic status. In this experiment, CCS codes with both chronic and chronic/acute definitions were defined as chronic. This mapping resulted in 141 chronic conditions and 92 non-chronic conditions. Condition names associated with each of these CCS codes as well as chronic/non-chronic status could be found in Appendix section (Table 21). By examining the CCS codes (see Table 13), it was observed that most of these codes are chronic conditions, supporting the definition of *Max* attributes in *TMMR* method. It should

be mentioned that since the plots were not consistent for LR-based models, the results could not be generalized to this algorithm.

Table 13: Top *Max* attributes among the top 40 predictors for each problem. The importance was calculated based on the average Gini Score for RF, GB, and DT algorithms. Condition names associated with the CCS codes can be found in in Table 21 (Appendix).

Alg	CCS	CCS%
	***Problem 1*-Mortality**	
RF	CCS98, CCS653, CCS259, CCS10, CCS211, CCS108, CCS133, CCS257, CCS106	22.5
GB	CCS98, CCS108, CCS204	7.5
DT	CCS53, CCS98, CCS211, CCS55, CCS108, CCS133, CCS10, CCS257, CCS259, CCS198	25.0
	***Problem 2*–High Utilization**	
RF	CCS108, CCS158, CCS157	7.5
GB	CCS158, CCS108	5.0
DT	CCS53, CCS257, CCS98	7.5
	***Problem 3*-CKD**	
RF	CCS98, CCS10, CCS53, CCS259, CCS49, CCS211, CCS101, CCS257, CCS164, CCS86, CCS106, CCS133, CCS117, CCS205, CCS102	37.5
GB	CCS98, CCS257, CCS50, CCS53, CCS17, CCS259, CCS49, CCS126, CCS653, CCS2616, CCS221, CCS97, CCS138	32.5
DT	CCS10, CCS259, CCS95, CCS133, CCS134, CCS101, CCS114, CCS126, CCS49, CCS127	25.0
	***Problem 4*-CHF**	
RF	CCS98, CCS53, CCS259, CCS257, CCS133, CCS164, CCS49, CCS106, CCS205, CCS256	25.0
GB	CCS98, CCS256, CCS53, CCS2618, CCS203, CCS50, CCS154, CCS63, CCS158, CCS132	25.0
DT	CS98, CCS53, CCS10, CCS211, CCS257, CCS133, CCS49, CCS259, CCS164, CCS95, CCS106, CCS114, CCS203, CCS205, CCS99	37.5

Diagnoses Groupers

The original diagnoses are stored in claims data as ICD-9 or ICD-10 codes. When modeling, these codes are often grouped to larger categories such as CCS, Elixhauser, or Charlson to reduce dimensionality of the representation space. Such reduction is often needed when limited amount of data are available.

This experiment addresses the question that if the results described above are specific to CCS codes or are generalizable to other coding systems (code groupings). More specifically, a version of Elixhauser (ELIX) code (version 3.0 or AHRQ-web ICD-9-CM Elixhauser code (Quan et al., 2005)) was applied to map ICD-9 codes into 30 categories. Specifically, CCS codes were mapped to ELIX codes resulted in a total of 30 attributes for Binary and 60 attributes for *TMMR*. The models were reconstructed using ELIX codes and compared with CCS-based models. Figure 15 compares the AUC of the four prediction problems for CCS vs. ELIX codes. As shown, the AUC of the models constructed on the ELIX codes was lower than the CCS-based models across all algorithms except for Temporal-based CKD model using DT in which ELIX model was slightly better. This is reasonable as some information is lost when mapping CCS codes to ELIX grouper codes with smaller number of categories and large number of datapoints, resulting in lower performance.

A key observation was that the 'shape' of the plots is the same as when using CCS codes. Thus, one can reason that the change in the AUC is independent of the representation methods in both Temporal and Binary Representations, which suggests potential generalizability of the findings to other groupers. However, similar experiments are needed

to be completed for other groupers. It is also possible that the ‘shape’ of the curves will change when only a limited amount of data is available.

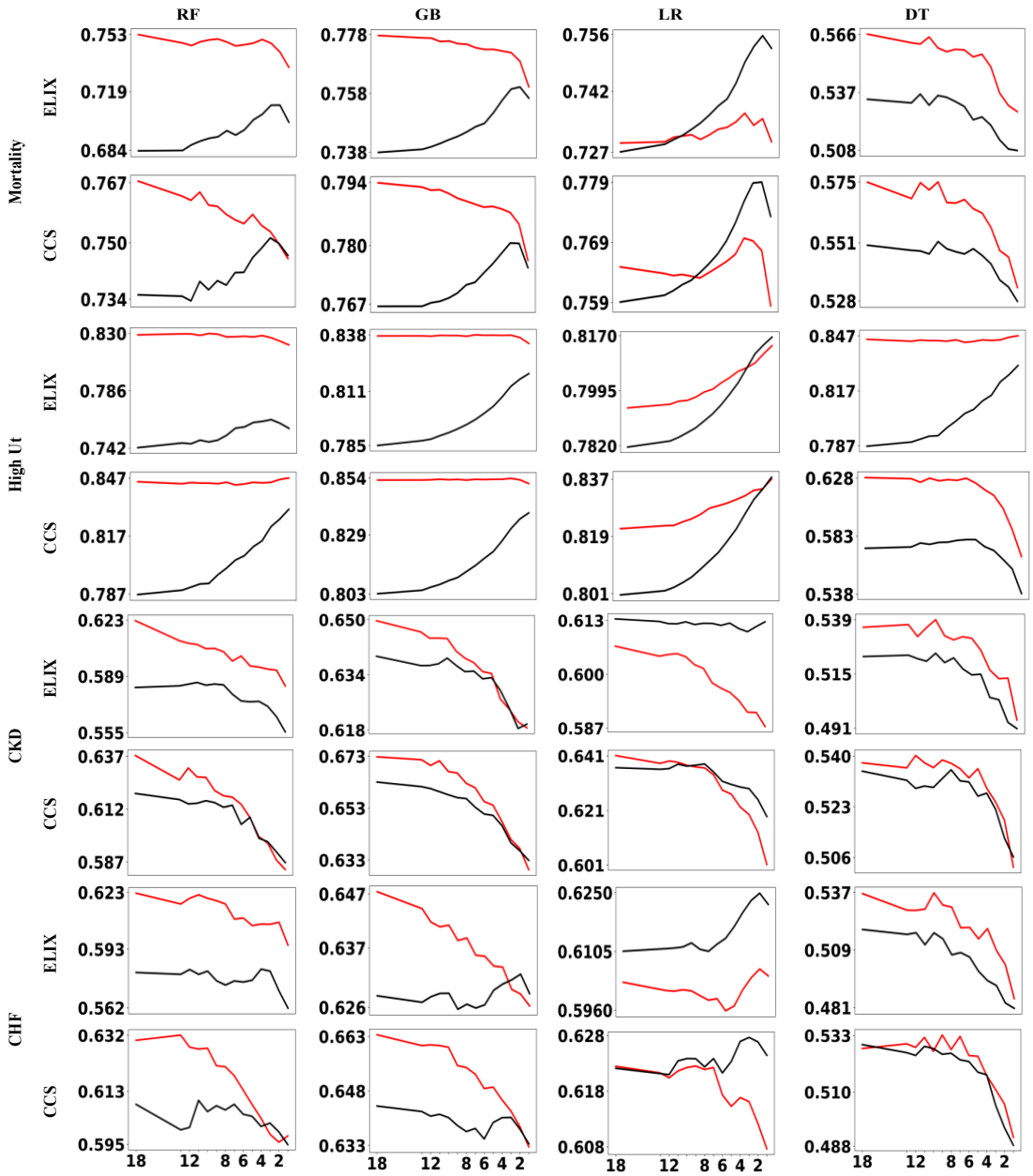

Figure 15: Comparison of the AUC of models on two different representation systems by changing the size of the observation window. Red and black lines indicate Temporal vs. Binary Representations, respectively.

Shape of the curves indicate that model performance depends on data representation but not on diagnosis groupers used.

Representation of Non-present Diagnoses

As described along with the definition of *TMMR*, the non-present administrative codes should take special values. While it may be reasonable to represent missing values with zero in regression-like models (that cancels out a term in regression), it may not work for symbolic models. The initial results in predicting Activities of Daily Living suggested that symbolic representation algorithms including Random Forest, Decision Tree and Gradient Boosting are not impacted by what special value was used, however it mattered for algorithms with numeric representation such as Logistic Regression (Wojtusiak et al., 2021a). The following experiment aims at replicating the results on the four prediction problems by replacing the special values with different n_9 values including +/-999999 (6_9), +/-99999 (5_9), and +/-9999 (4_9) values. Also, these non-present codes were replaced with other values including 365 (representing 1 year), 730 (representing 2 years) and maximum value of each administrative code (Max_Each) to determine how small special values that are within the range of the data would impact the results. In addition, the models were created by changing the observation window size to see how these special values are impacted by the amount of available data.

Figure 16 compares the AUC of the constructed models using different Temporal methods. Each line corresponds to one of the Temporal methods. The AUC corresponding to Binary Representation method was also shown in each plot for better comparison. The results confirmed our initial observation in predicting ADLs; Random Forest, Gradient

Boosting and Decision Tree were not affected by these special values were assigned through different observation window sizes, while the AUC of the Logistic Regression-based models was impacted by the representation method. One explanation for these results is that for symbolic methods, it is irrelevant how not-present values are represented as long as the value is distinct, while parametric models need to find a coefficient for each diagnosis code, which is affected by the representation of the codes. Replacing non-present codes with 365 (1 year) or 730 (2 years), however, lowered the AUC of some models in symbolic representation algorithms. Because these values are small enough, they might be confused by the number of days within the data, thus affecting the AUC. Additionally, it was observed that depending on the size of the observation window, one Temporal method outperformed the other in LR-based model. Finally, and as expected, different Temporal Min-Max methods outperform Binary Representation using RF, GB, and DT algorithms, while it changes depending on the observation window size in LR-based models.

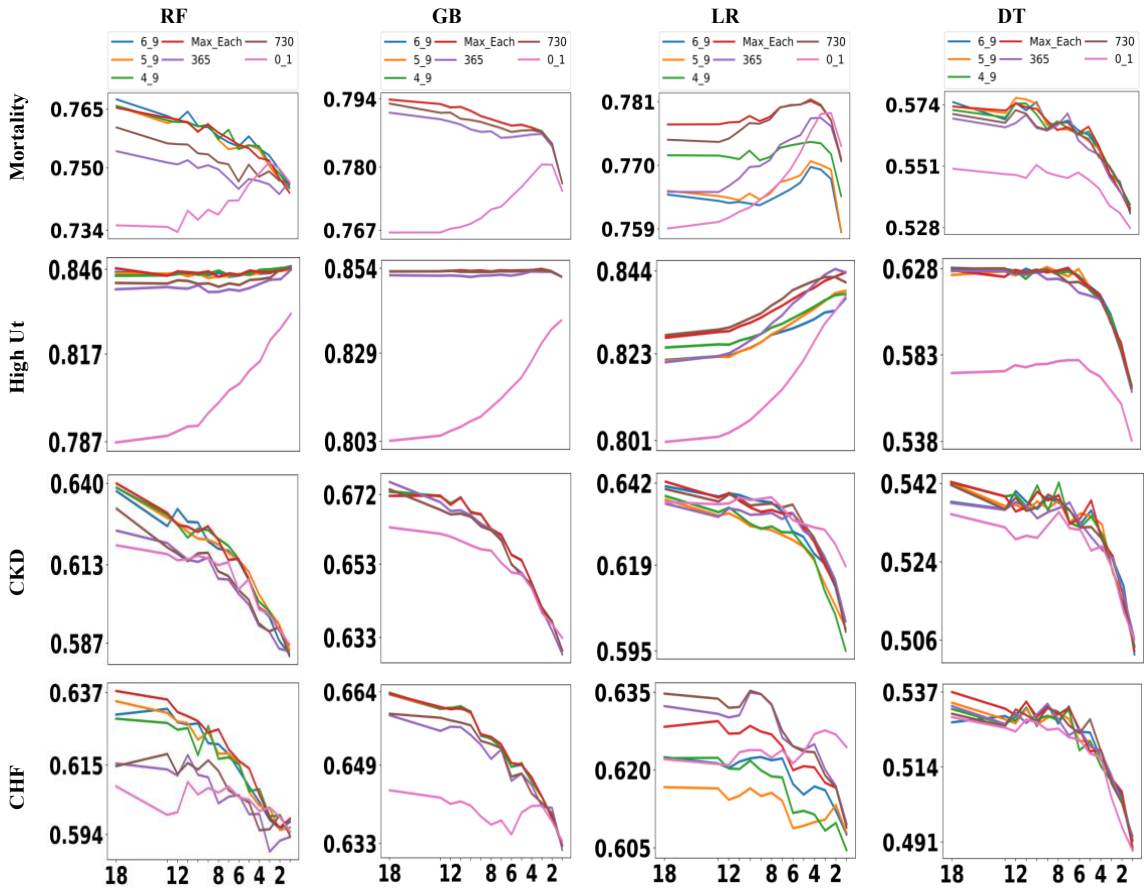

Figure 16: Comparison of the AUC of different Temporal Min-Max Representation across four models including +/-999999 (6_9) , +/-99999 (5_9), and +/-9999 (4_9), 365 (1 year), 730 (2 years), and maximum value of each diagnosis attribute (Max_Each) by varying observation window size; each line correspond to one Temporal method. Binary Representation was also included for better comparison.

Age and Racial Biases

Another important question is how the two representation methods differ with respect to demographic factors? It is essential that machine learning-based models are free of biases that may potentially discriminate against certain groups of individuals. Thus, testing how claims representation methods affect potential biases is of critical importance. To answer this question, the constructed models with the two representation methods were

tested on patients with different age groups as well as races. For age experiment, different age categories were created including: ’70-74’, ’75-80’, ’81-85’, ’86-89’, ’75-80’, and ‘more 90’ age groups. The models were also tested on six race groups including: ‘White’, ‘Black’, ‘Asian’, ‘Native American’, ‘Hispanic’, and ‘Unknown’. The AUC of the models was then calculated for these subcategories.

The plots in Figure 17 demonstrate the AUC values of the models on different age groups for *TMMR* vs. Binary methods. Patients’ distribution for each age group is also included in the figure. It was observed that *TMMR* outperforms Binary Representation method across different age groups for the four prediction problems, suggesting the superiority of *TMMR* across age groups. More interestingly, the results indicated that even though models’ performance becomes worse for older patients (due to smaller sample size), this decrease is smaller for *TMMR* method; larger difference in AUC of the two representation methods among older patients (underrepresented population) can indicate that *TMMR* is potentially more stable than Binary method in predicting the four outcomes. Consequently, this suggests the potential of *TMMR* in minimizing the disparity among different age groups.

Results with race groups (see Figure 18) also confirmed the superiority of *TMMR* over Binary method. Similar patterns were also observed in predicting some of the underrepresented population across the four prediction problems.

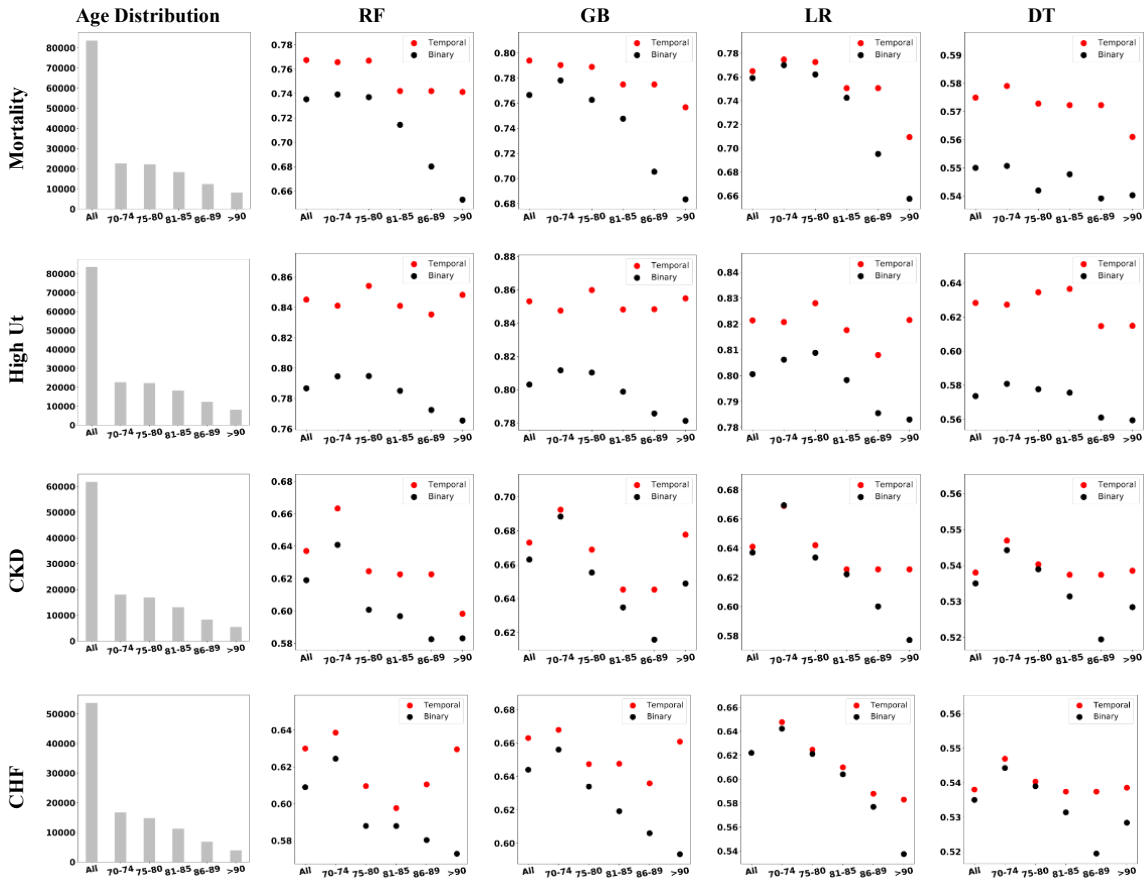

Figure 17: Comparison of the AUC of models for *TMMR* vs Binary methods on different age groups. The age groups include 70-74', '75-80', '81-85', '86-89', '75-80', and 'more 90'. For better comparison, the average AUC of models on all patients is included in the figure. Also, the distribution of the cases for different age groups is shown for each prediction problem.

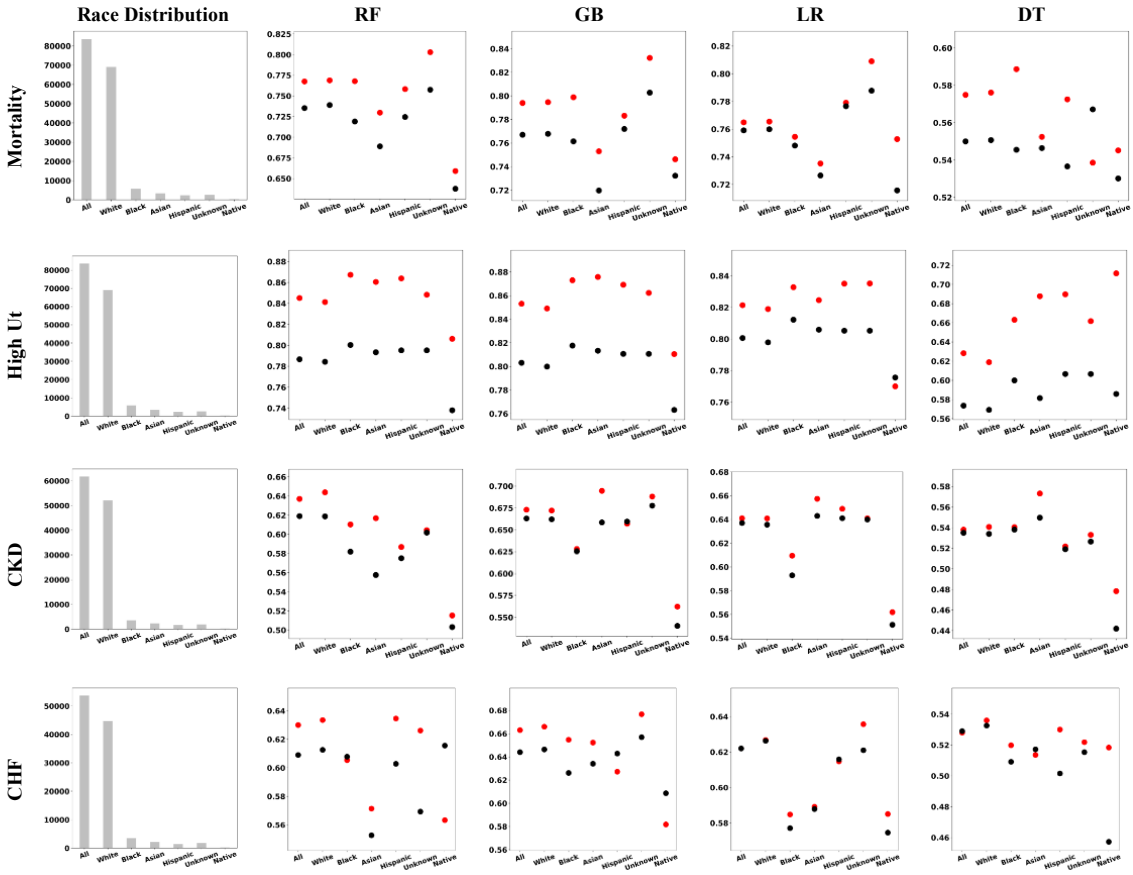

Figure 18: Comparison of the AUC of models for *TMMR* vs Binary methods on different races. The race groups include 'White', 'Black', 'Asian', 'Hispanic', 'Unknown', and 'Native American'. For better comparison, the average AUC of models on all patients is included in the figure. Also, the distribution of the cases for each race group is shown for each prediction problem.

The above analysis illustrated that the potential biases are smaller for *TMMR* than standard Binary Representation. However, potential biases need to be investigated for all constructed models, regardless of representation used. Further analysis could also be done for other characteristics of patients, such as gender that is not present in the analyzed data.

Sensitivity Analysis

Sensitivity analysis is one way to measure the uncertainty of the models in making prediction. In other words, in measures how the changes in input attributes would affect

output. It refers to model's variance in machine learning and statistics domain. The model becomes less stable or more sensitive, if any small changes in inputs results in significant changes on the output (Wojtusiak, 2021). Therefore, an experiment was conducted to determine how the probability of each outcome is affected by changes in each administrative code (CCS code) across the two representation methods. For this purpose, the change in output probability was measured by converting present CCS codes (one code at a time) to non-present one. In Binary Representation, present codes (represented with 1) were converted to zero. In *TMMR method*, present *Min* and *Max* attributes of each code (represented by the number of days from diagnosis to prediction time) were replaced with 999999.0 and -999999, respectively. Figure 19 visually illustrates how sensitivity analysis is applied for both Binary and *TMMR* methods on one diagnosis code. As shown, CCS_1 is converted to zero in Binary Representation, while *Min* and *Max* copies of the same code is converted to +/- 999999 in *TMMR* representation. Then the changes in output probability before and after this conversion is calculated. The sensitivity of the model with respect to each CCS code is the average of the changes for all patients in one of the test sets. The average of the change in output probabilities was then calculated across all CCS codes and compared between the two representation methods. The results were reported across the true label of each outcome and Wilcoxon signed-rank test was also used to determine the significance of the results.

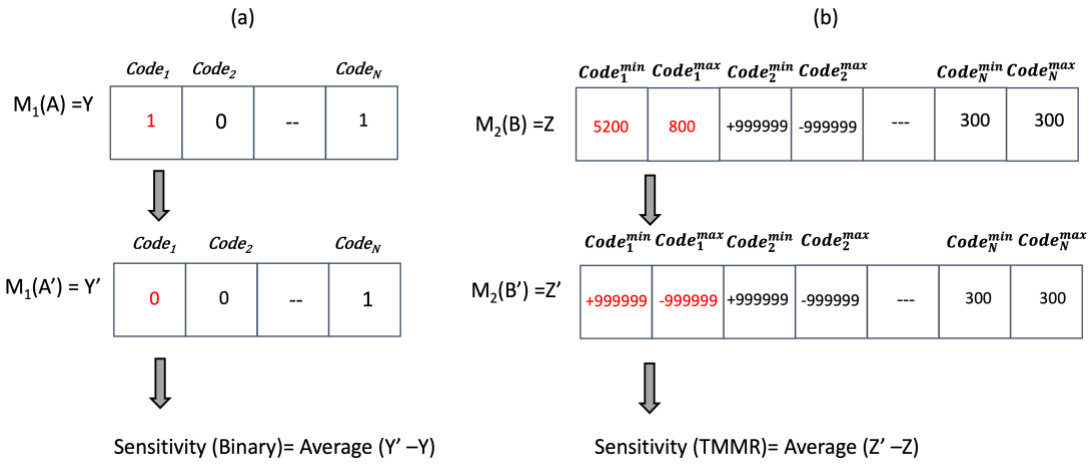

Figure 19: Sensitivity analysis framework for Binary (a) and *TMMR* (b) methods. In Binary Representation, the change in output probability is calculated by converting 1 to 0. In *TMMR*, such change is measured by changing the present codes to +/-999999. Sensitivity of the models with respect to each diagnosis codes is the average of output probability changes for all patients.

As shown in Table 14, both representation methods had small changes in the average output probabilities across all CCS codes suggesting that the models developed with two representation methods are not sensitive to changes in individual diagnoses in general. Nevertheless, there were some cases for which converting present to non-present codes affected the output prediction with large changes in output probability. It was observed that the sensitivity of the models on the two representation methods was impacted by the prediction problem, algorithm, and the true label of each class and the results were significant among most of the comparisons. For example, in predicting mortality using GB, Binary Representation-based model was on average more sensitive than Temporal one, while an opposite pattern was observed when LR was used.

Intuitively, one would expect a decrease in the output probability by converting present CCS codes to non-present ones. Even though this is true for most instances, it is possible for some algorithms and models to have an increase in the output probability,

resulting in the negative change in output probabilities. This could happen due to the correlation among different CCS codes that can impact learning of an instance during model training.

Table 14: Comparison of the changes in output probabilities between the two representation methods.

***Problem 1*-Mortality**				
	Positive Label		**Negative Label**	
Alg	***TMMR***	***BIN***	***TMMR***	***BIN***
RF	0.0001	0.0002	-0.0001*	-0.0001
GB	0.0006*	0.0007	0.0001*	0.0001
LR	0.0005*	0.0004	0.0000*	0.0000
DT	0.0003	-0.0001	-0.0002*	-0.0003
***Problem 2*–High Utilization**				
RF	0.0013	0.0015	0.0003*	0.0004
GB	0.0021	0.0022	0.0004*	0.0006
LR	0.0021*	0.002	0.0006*	0.0006
DT	0.0018	0.0012	0.0002*	0.0001
***Problem 3*-CKD**				
RF	0.0000	0.0000	-0.0001	-0.0001
GB	0.0001*	0.0002	0.0000*	0.0001
LR	-0.0044*	-0.0001	-0.0046*	-0.0001
DT	-0.0319*	-0.0004	-0.0430*	-0.0002
***Problem 4*-CHF**				
RF	0.0000	0.0000	0.0001*	-0.0001
GB	0.0001*	0.0002	0.0000*	0.0001
LR	0.0001*	0.0000	0.0000*	0.0000
DT	0.0001*	-0.0005	-0.0002*	-0.0004

Then, similar experiment was conducted across all CCS codes that were present in patient's records only once in the data (value of *Min* and *Max* attributes are the same indicating only one encounter for a specific diagnosis code) and the results are shown in Table 15. It was observed that the changes in output probability was larger in Binary-based models for most of the algorithms across four prediction problems and the results were significant mostly across negative labels.

Table 15: Comparison of the changes in output probabilities between the two representation methods for codes present once in data.

***Problem 1*-Mortality**				
	Positive Label		**Negative Label**	
Alg	***TMMR***	***BIN***	***TMMR***	***BIN***
RF	0.0035*	0.005	0.0036*	0.005
GB	0.0025	0.0033	0.0005*	0.0024
LR	0.0018	0.0019	0.0041	0.0025
DT	0.0085	0.0081	0.0049*	0.0072
***Problem* 2–High Utilization**				
RF	0.0031*	0.0063	0.004*	0.0064
GB	0.0024*	0.0064	0.0008*	0.0042
LR	0.004*	0.0041	0.0016*	0.0032
DT	0.003*	0.0122	0.0061*	0.0088
***Problem 3*-CKD**				
RF	0.003	0.0045	0.0042*	0.0048
GB	0.0003	0.001	0.0008	0.0018
LR	0.0017	0.0016	0.0001	0.0005
DT	0.0089	0.013	0.0037*	0.0145
***Problem 4*-CHF**				
RF	0.0036*	0.0047	0.0044*	0.0053
GB	0.0004	0.001	0.0013	0.0046

LR	0.002	0.0017	0.0034	0.0018
DT	0.0065	0.017	0.0054	0.0114

Comparison of Model Sensitivity among Min and Max Attributes

To investigate the relationship between *Min* and *Max* attributes and the change in output probabilities of Temporal-based models, scatterplots were created for different CCS codes. Figure 20 and Figure 21 show examples of three codes: CCS108 (congestive heart failure), CCS653 (delirium, dementia, and amnestic and other cognitive disorders) and CCS158 (chronic kidney disease) in predicting high utilization of medical services using Gradient Boosting algorithm across *Min* and *Max* attributes, respectively. These codes were selected from the top 10 predictors of high utilization based on the average GINI Score. The scatterplots were created for all cases as well as those for which then changes in output probability is meaningful (absolute value is greater than 5%) as shown in the second row of each figure. Overall, the change in output probabilities was higher among top predictors whereas the change in less important attributes was smaller or close to zero. As shown in Figure 20, there was a relatively small negative relationship between the last occurrence of diagnosis (*Min* attribute) and changes in output probabilities, meaning that the changes are larger when the last occurrence of these diagnoses is close to the prediction time. The negative coefficient corresponds to the fitted line in these scatterplots confirms such relationship. Also, large changes in output probabilities happens when the last recent occurrence of the diagnosis was less than about 1000 days. The negative correlation was more apparent on cases for which the change in output probability was greater than 5%.

Conversely, there was a small or no correlation between the time from first occurrence of the diagnosis (*Max* attributes) and changes in output probability (See Figure 21, first row). However, among cases with meaningful changes (greater than 5%), there was a clear positive correlation between *Max* attributes and changes in output probabilities (second row); this indicates that the models are more prone to changes when the patient was first diagnosed with a disease long time ago. The results overall suggest that Temporal models are more sensitive to the last occurrence of the diagnosis than time since the first diagnosis. This fact is also consistent with the last occurrence of disease (*Min* attribute) being overall more important.

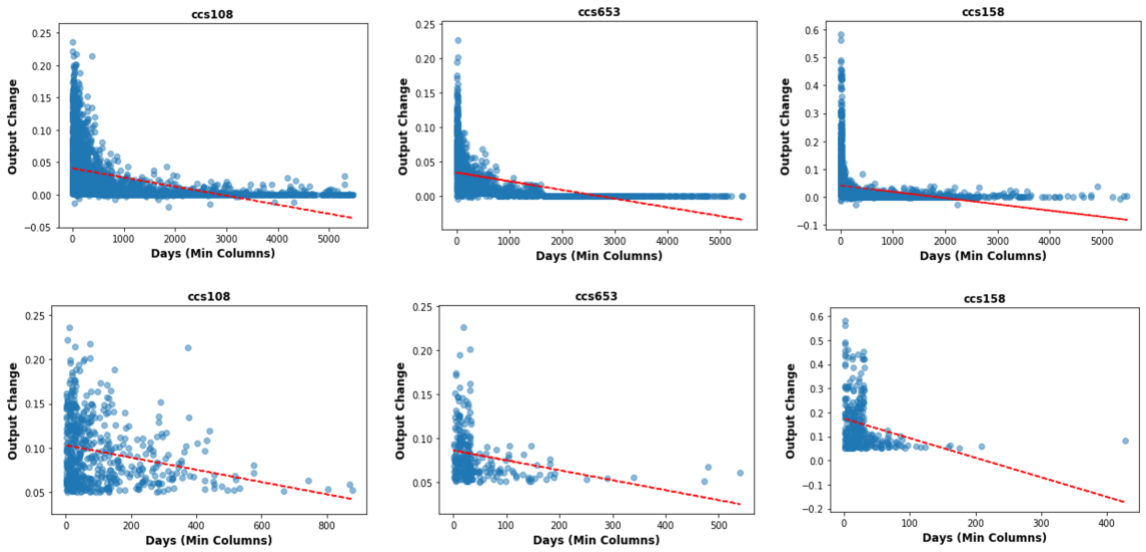

Figure 20: Relationship between changes in output probability and *Min* attributes in predicting high utilization using Temporal method. The vertical and horizontal axes correspond to changes in output probability and *Min* attributes for CCS108, CCS653 and CCS158, respectively. The first row refers to all patients in the test set with mentioned diagnosis codes, while the second row refers to those with meaningful changes in probabilities (greater than 5%).

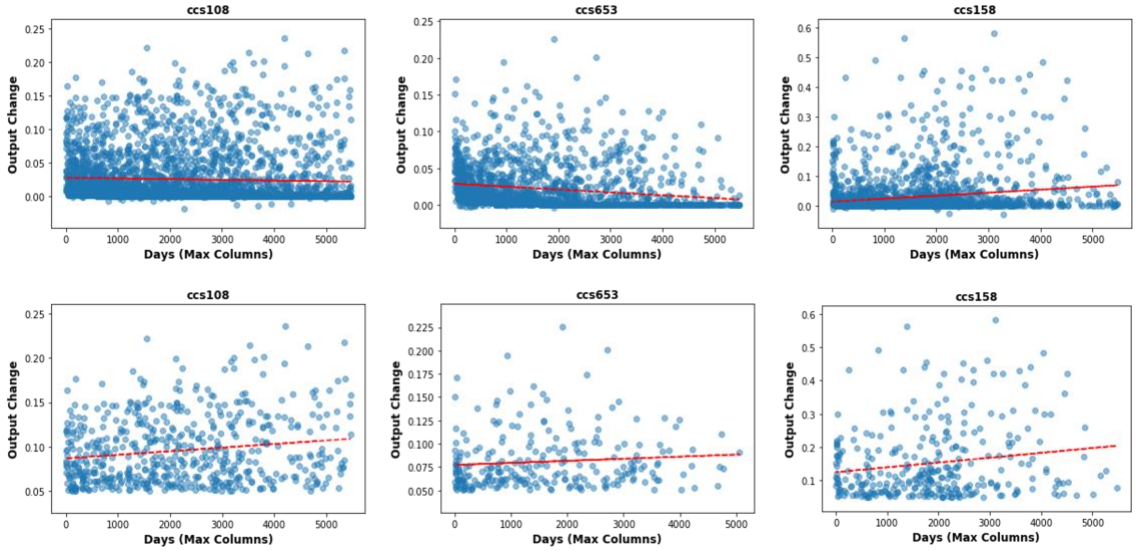

Figure 21: Relationship between changes in output probability and *Max* attributes in predicting high utilization using Temporal method. The vertical and horizontal axes correspond to changes in output probability and *Max* attributes for CCS108, CCS653 and CCS158, respectively. The first row refers to all patients in the test set with mentioned diagnosis codes, while the second row refers to those with meaningful changes in probabilities (greater than 5%).

Sensitivity Comparison between TMMR and Binary Models across Min and Max Attributes

An experiment was then performed to determine how the time from diagnosis (*Min* and *Max* attributes) impacts the sensitivity of models with respect to *TMMR* vs. Binary methods. Scatterplots were created to visualize example results for both *Min* and *Max* attributes (high utilization prediction using GB algorithm) and the results are shown in Figure 22 and Figure 23 for diagnosis codes CCS108, CCS653 and CCS158. The figures were developed on all cases (first row of each figure) and on cases for which the change was greater than 5% (Second row of each figure). Blue and Orange points refer to Temporal and Binary-based models, respectively. As shown, Binary models were less sensitive to time from diagnosis compared to Temporal-based models, meaning that there is a weak

correlation between the time from diagnosis and changes in output probabilities across Binary models. One might argue that since the changes in output probabilities in Binary models was calculated based on converting present code (represented as 1) to non-present (represented as 0), the model should not be impacted by the time to diagnosis and therefore, the correlation should be zero for these models. This weak correlation can be justified by the association between specific CCS codes with other diagnosis codes, which can potentially impact the output probability.

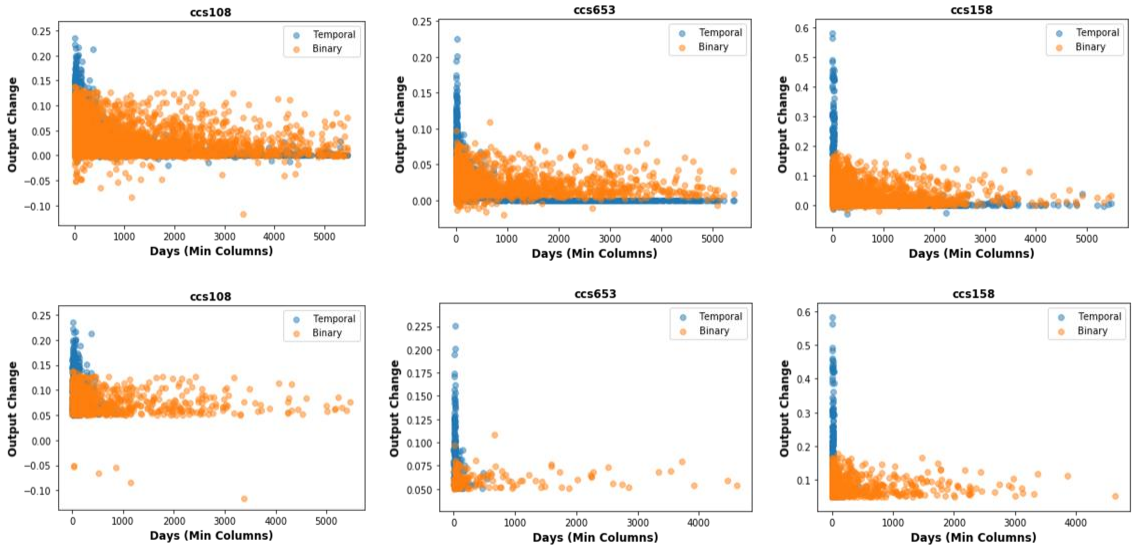

Figure 22: Relationship between changes in output probability and *Min* attributes across both Temporal (blue points) and Binary Representation (orange points) methods. The vertical and horizontal axes correspond to changes in output probability and the number of days (*Min* attributes), respectively. The first row refers to all patients, while the second row refers to those with meaningful changes in probabilities (greater than 5%).

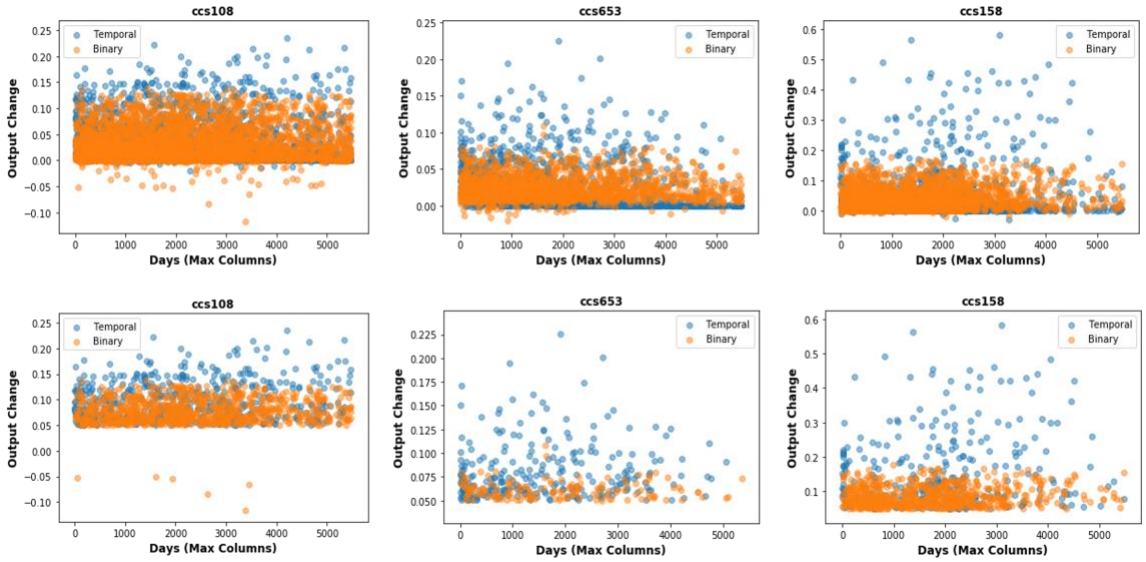

Figure 23: Relationship between changes in output probability and *Max* attributes across both Temporal and Binary Representation methods. The vertical and horizontal axes correspond to changes in output probability and the number of days (*Max* attributes).

Model Sensitivity and Outcome Analysis

To investigate the sensitivity of the models to changes in input attributes, a series of experiments were performed. To illustrate results, example scatterplots were created for the three mentioned diagnosis codes across *Problem 1* (Mortality) and *Problem 2* (High Utilization) using GB algorithm. Figure 24 and Figure 25 show the changes in output probabilities across *Min* and *Max* attributes, respectively. In each figure, first row refers to *Problem 1* and second row refers to *Problem 2*. For better illustration, the plots were only produced when the change in output probability was greater than 5%.

As shown, the negative correlation between *Min* attributes and output probability and the positive/close to zero correlation between *Max* attributes and output probability changes is independent of the outcome. However, the magnitude of the changes is different for each CCS code with models being more sensitive to their top predictors. For example,

the changes were larger for CCS108 (congestive heart failure) and CCS 653 (delirium, dementia, and amnestic and other cognitive disorders) in predicting mortality, whereas the change for CCS158 (chronic kidney disease) was larger in predicting high utilization. The diagnosis codes CCS108 and CCS653 are among the most important features in predicting mortality, while CCS158 is more predictive of high utilization of medical services. Although CCS653 is also a top predictor of high utilization, its higher importance in predicting mortality making it more sensitive in *Problem 1*. Similar analysis for CKD and CHF models also confirmed such observation. In summary, it is advisable to test sensitivity every time a new model is constructed, rather than assuming certain model properties.

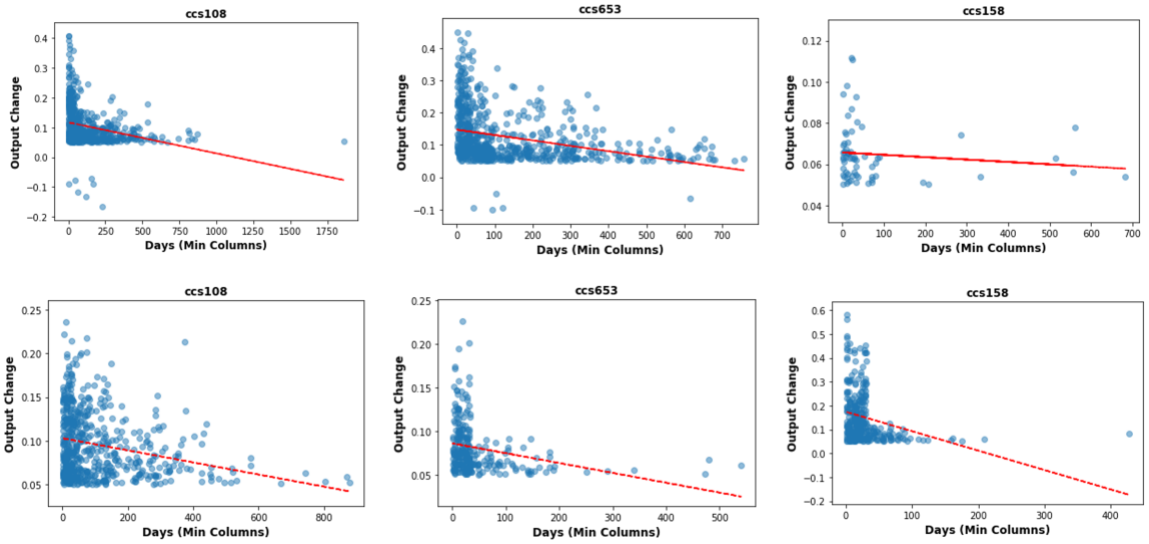

Figure 24: Comparison of the changes in output probability and *Min* attributes across *Problem 1* (Mortality) and *Problem 2* (High Utilization); the first row corresponds to *Problem 1* and the second row refers to *Problem 2*.

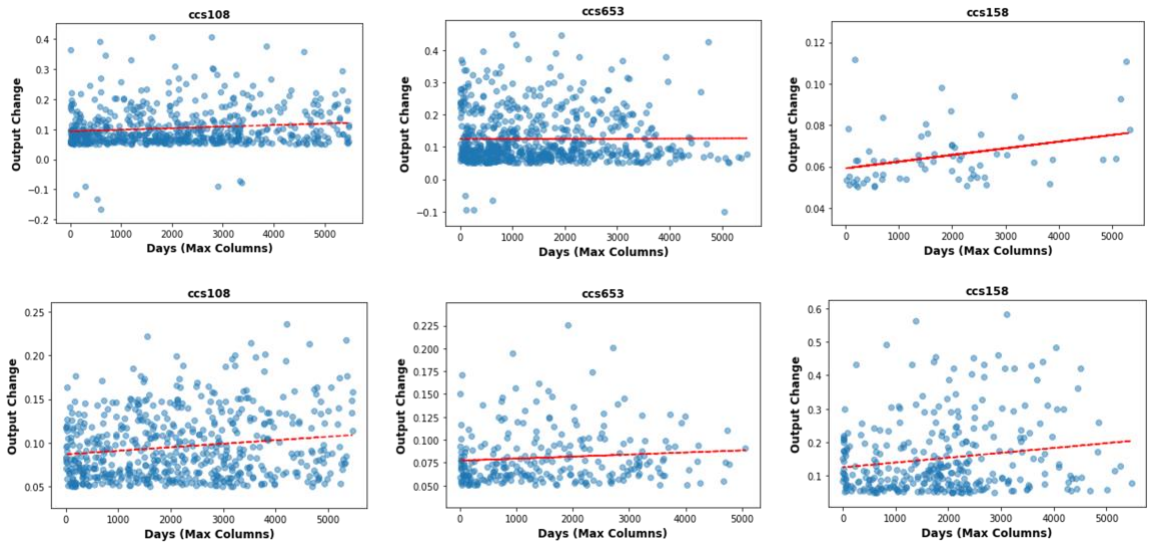

Figure 25: Comparison of the changes in output probability and *Max* attributes across *Problem 1* (Mortality) and *Problem 2* (High Utilization); the first row corresponds to *Problem 1* and the second row refers to *Problem 2*.

Conclusion on Temporal Min-Max Representation

In this chapter, Temporal Min-Max Representation (*TMMR*) was evaluated and compared with Binary Representation on four prediction problems. Population-level comparison of the two representation methods indicated that *TMMR* outperforms Binary Representation in most cases. The results also showed that the representation of these codes is affected by algorithm, outcome, observation window size, characteristics of input attributes including time from diagnosis to prediction time, number of diagnosis codes and demographic factors. Depending on any of these parameters, one representation method outperforms the other. The sensitivity of the models also differs with respect to the representation method. The results indicated that the sensitivity of *TMMR* models is larger compared to Binary models for codes present once in the data. It was also shown that even within *TMMR*-based models, the sensitivity of models is impacted by *Min* vs. *Max*

attributes, with models being generally more sensitive to *Min* attributes. In these models, there is a positive and negative correlation between *Min* and *Max* attributes and changes in output probability, and such correlations are independent of the outcome.

TRAJECTORY REPRESENTATION

Trajectory of Illness

A trajectory of illness indicates the change in health status as a function of time. Modeling illness trajectories helps patients and caregivers understand how the health conditions change over time, as well as in prediction of changes to come. Knowing why and what changes the health of status can help identifying the adverse events and implementing appropriate interventions if needed. Changes in health can be impacted by many factors including genetics, biological, behavioral, social, and environmental factors (Henly et al., 2011). In medical literature, the trajectories of end of life are categorized into four different groups (see Figure 26): 1. Trajectory with steady progress over time followed by a sudden decline in the final few months. This trajectory is typically observed among cancer patients. 2. Trajectory with gradual decline followed by multiple episodes of exacerbations and temporary improvement, resulting in sudden death. These trajectories include heart failure or chronic obstructive pulmonary disorder for which the time of death is less certain. 3. Trajectory with gradual decline progressing over many years. This trajectory is observed during fatality or among patients with dementia. 4. Sudden death or medical disability, which could be due of trauma or cardiopulmonary/neurologic condition (Murray et al., 2005; Barker & Scherer, 2019). The trajectories considered in the above works are limited to the end of life stages, whereas more trajectories (shapes) can be observed in general. One could think of many possible patterns for illness trajectories including improvement, decline, stable, temporary decline, temporary improvement etc.

(Wojtusiak et al., 2016). For example, Wojtusiak et. al identified 7 different trajectories when predicting changes in ADLs post hospitalization; these trajectories included early recovery, delayed recovery, delayed recovery after temporary decline, early decline, delayed decline, delayed decline after temporary recovery, and no change (Wojtusiak et al., 2016). Different mathematical functions can be used to model changes in the health status including linear, quadratic and higher-order polynomial, and exponential functions (Henly et al., 2011).

One complexity in modeling trajectories (which are continuous by nature) is that patients are observed at discrete points in time and these time points are only available when data elements such as diagnosis or procedure codes are reported in medical claims. The diagnosis codes in medical do not represent the real patient status as only discrete billable conditions being reported. Thus, these are only as an approximation of the hidden patient's status as well as its change over time.

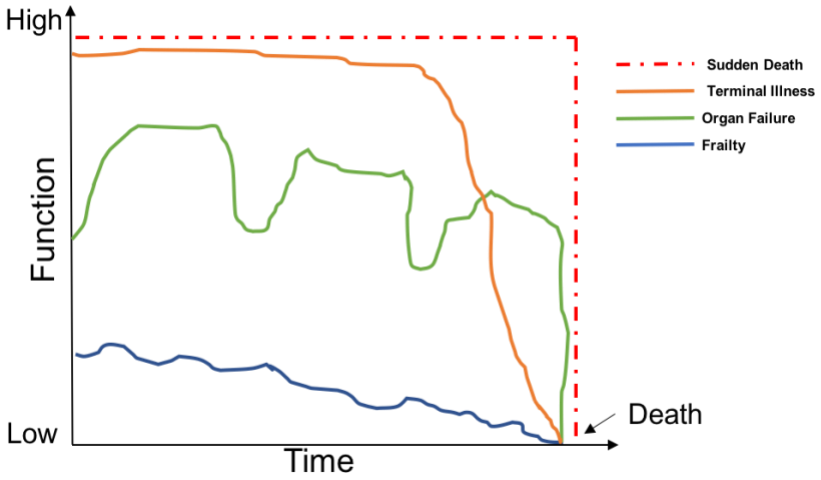

Figure 26: Trajectory of illness for different categories of health conditions. The horizontal axis corresponds to time to death and vertical axis refers to

functionality of individuals according to their health conditions. Adapted from https://www.mypcnow.org/fast-fact/illness-trajectories-description-and-clinical-use/

The trajectory of illness can clearly display the severity of illness (SOI) over time. Severity of illness refers to "the extent of physiologic decompensation or organ system loss of function" (Averill et al., 2003), meaning how serious a medical condition is. The severity of illness (SOI) rating assigns a score to a physiological condition based on a 1 to 4 scale. As part of hospital reimbursement, SOI plays a crucial role in healthcare costs (Spurgeon et al., 2011).

In 1983, CMS introduced the diagnosis-related groups (DRGs) codes to substitute the per-dime method of hospital reimbursement, in which patient stay and resource utilization were used for payment assessment (Averill et al., 2003; Spurgeon et al., 2011). DRG is categorized into three groups: basic DRGs, All Patient DRGs (AP-DRGs), and All Patient Refined DRGs (APR-DRG). The first category is used by CMS for payment assessment of beneficiaries. All Patient DRGs (AP-DRGs) encompasses Medicare and non-Medicare population including pediatric patients. In All Patient Refined DRGs (APR-DRG), four subclasses have been added to represent the severity of illness in order to better match the money received by hospitals and the cost they incur to treat patients (Averill et al., 2003). In this type of code, the severity of illness or the risk of mortality is categorized into minor, moderate, major, and extreme or is numbered from 1-4, with 1 being a minor health condition (Averill et al., 2003). Similar to other types of billing codes, DRGs are static, and do not reflect the changes of patients' health status over time.

Longitudinal claims data store patient's administrative codes along with the date of service (or claim date) in the form of sequence of temporal events, which can occur at regular or irregular time intervals. These sequences also vary across different patients. Sequences of these billing codes can show progression of disease or severity of illness for each condition over time. Consequently, each diagnosis code can be represented in the form of trajectory of disease.

This chapter aims at introducing and evaluating another method of representing diagnosis codes called 'Trajectory Representation' (*TJR*) by using the trajectory of illness concept. This study uses trajectories reconstructed from patients' histories as inputs in predicting patient outcomes. This contrasts with works available in the literature that focus on predicting the future trajectories of patients.

Trajectory Representation

The approach studied in this dissertation is to construct the trajectory of illness representing changes in patient's health from the time between visits/encounters specific to a given condition. The time between visits can represent how severe a health condition is and how it changes over time. Consider a patient with an underlying health condition such as cancer, which progresses over time and has different stages. Such patient could have multiple inpatient and outpatient visits at even or uneven time intervals. In the beginning, the patient might have short intervals between visits when the diagnosis and initial treatment is made. As the treatments continues, the patient has regular intervals between visits. Later, if the patient responds to the treatment, fewer visits at longer time intervals may be needed. On the other hand, if the treatment doesn't work well and more

interventions are needed, the frequency of visits might increase while the time between visits might decrease.

In the method studied here, the trajectory of illness is represented by a regression line fitted into time between encounters, and more specifically the coefficient and intercept of a fitted linear regression. Similar to Temporal Min-Max Representation, each diagnosis code (i.e. ICD code) is represented by its corresponding coefficient and intercept of that regression.

Figure 27 illustrates how the Trajectory Representation *(TJR)* method works. Suppose a patient has *N* visits $V_{1,\dots N}$ for a diagnosis code *C* over time, where V_1 and V_N are the first and most recent encounter of *C*, respectively. The diagnosis C_i and its associated V_j at date ($dt_{i,j}$) is represented by subtracting the time from the previous visit (V_{j-1}). In other words, C_i associated with V_j is represented as follows and is shown as $t_{i,j}$:

Equation 9: Time Between Diagnosis

$$t_{i,j} = dt_{i,j} - dt_{i,j-1}$$

In the above equation, there are *N-1* time-between points for *N* visits associated with a diagnosis C_i. A simple approach such as linear regression can then be used to model change in time between visits for a given diagnosis code, resulting in coefficients (slope) and intercepts associated with that diagnosis. Since one encounter is not sufficient to fit the line, a minimum of three encounters is needed to construct trajectories.

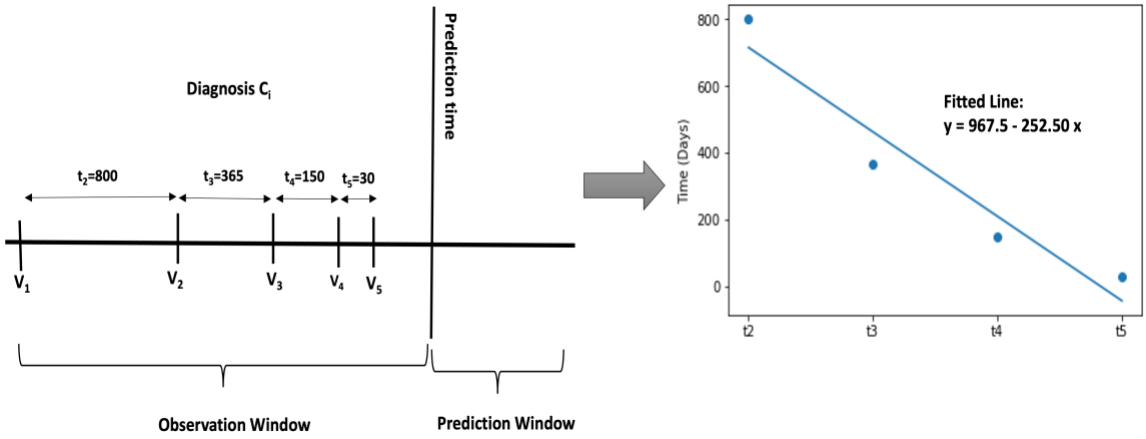

Figure 27: Illustration of constructing trajectory of disease by calculating the time between visits for each diagnosis code.

Interpretation of Trajectory Representation

The coefficient (slope) and intercept of the fitted line can represent how patient's health changes over time. For each diagnosis, these slopes can indicate improvement, decline or stability in health condition. Figure 28 compares the constructed trajectories of two hypothetical patients represented by blue (patient 1) and orange (patient 2) lines. Positive slope indicates that the time between visits is increasing, while negative slope indicates the opposite. Thus, and as shown in Figure 28 (a), the positive slope in the case of patient 2 may indicate possible patient's improvement, and negative slope can indicate patient's decline. The value of the slope can also indicate how quickly patient's health is changing. Figure 28 (b) shows two hypothetical patients with negative slopes with the blue line (patient 1) being steeper than orange line. Shorter intervals between visits result in

steeper slope; this suggests that even though both patients' health is deteriorating, it is faster for blue line.

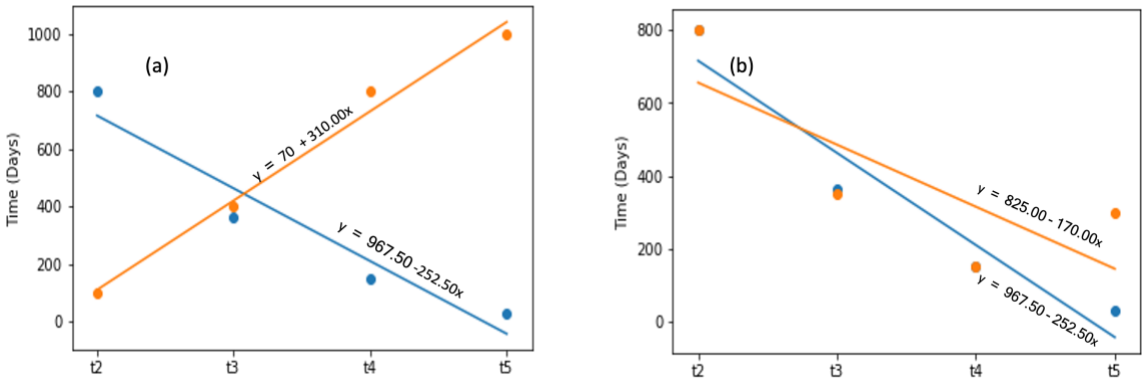

Figure 28: Changes in the coefficient of the fitted lines with different time intervals in visits.

Figure 29 illustrates the translation of time between diagnosis codes to the trajectory of illness for a hypothetical patient with two health conditions (ELIX21 code representing diabetes and ELIX18 representing hypertension) and how this time can be translated to coefficient and intercept of a fitted line. Each datapoint in the figure corresponds to the visit related to that diagnosis code. While the horizontal axis shows the time since the first occurrence of the disease, the time interval between the visits is clearly visible. According to the figure, the time between visits related to diabetes (ELIX21) is decreasing, resulting in a fitted line with a negative coefficient; this suggests that patient's health is declining with regard to diabetes. In terms of hypertension (ELIX18), however, as the interval between visits increases over time, the patient may require fewer visits, indicating that the condition has improved or better managed.

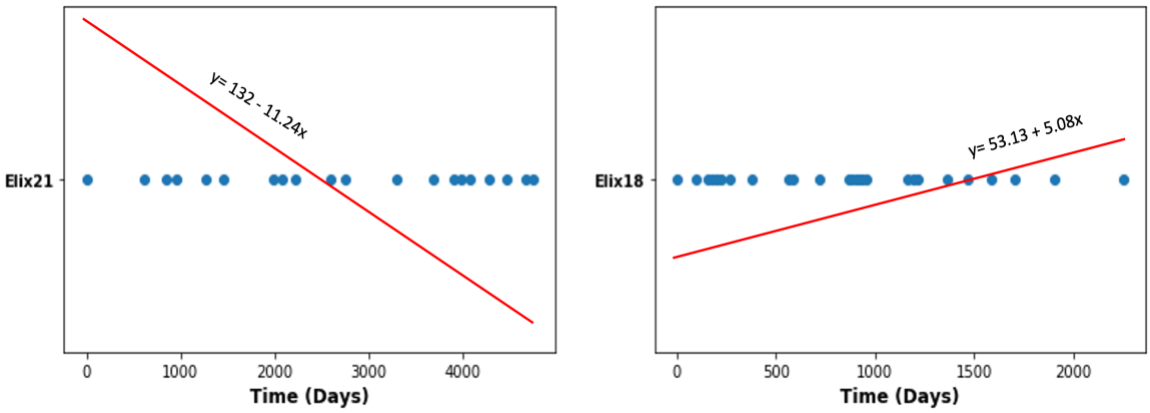

Figure 29: Illustration of the constructed trajectory of illness from time between visits for a hypothetical patient. The diagnosis codes from left to right are ELIX21 (Diabetes) and ELIX18 (Hypertension), respectively.

Trajectory Construction

To construct the trajectories, patients' visits corresponding to diagnosis codes were extracted over 18 years of patient history. These trajectories were created with Exlihauser (ELIX) codes, which reduce dimension and increase frequency of visits. Each diagnosis code with more than 10 associated encounters was represented with two attributes referred as intercept and coefficient of the fitted line. It is important to note that even though a minimum number of 3 visits are required to fit a line, but they may not be enough to reflect how patients' health changes over time. An experimental analysis led to selection of the minimum of 10 data points (encounters) as they are large enough to represent the trajectory but small enough not to miss any important trajectories. On the other hand, the coefficient and intercept of diagnoses with less than 10 visits were represented with a special value of –888888 (the value was selected as they are easily distinguishable in data but have no

specific numeric meaning). With 30 Elixhauser codes, a total of 60 attributes were assigned to trajectories.

Table 16 displays the mean of the coefficient and intercept values for all ELIX codes between positive and negative classes. As shown, the mean values of the calculated coefficients were negative in both positive and negative classes among the four outcomes. However, the absolute value of the coefficients as well as the average value of intercepts were higher for positive labels compared to negative labels. As a result, if a hypothetical line is plotted for positive vs. negative labels, it can be observed that even though patients' health is declining between the two classes, it is declining faster for positive labels (patients with negative health outcome, i.e., death).

Table 16: Distribution of the mean of coefficients and intercepts of all ELX codes among positive and negative labels of the four outcomes.

	Positive Label		Negative Label	
Outcome	**Coefficient**	**Intercept**	**Coefficient**	**Intercept**
Mortality	-5.24	152.69	-2.11	131.65
High Utilization	-5.02	143.38	-1.69	131.61
CKD	-2.56	136.57	-1.29	129.58
CHF	-2.38	130.61	-1.44	124.64

As mentioned, the corresponding intercept is larger for positive labels than for negative ones across the four outcomes. However, since the fitted line was determined based on the time between visits, its intercepts cannot be interpreted directly. To visually compare the difference in value of intercept between positive and negative classes,

scatterplots between the coefficient and intercept of each diagnosis code were created. Figure 30 shows a scatterplot for one Elixauaser code representing congestive heart failure. The vertical and horizontal axes correspond to the intercept and coefficient, respectively. As shown, there was a negative correlation between the coefficient and intercept (the smaller coefficients are the larger the intercepts are) suggesting that the average intercept value is larger for positive classes. Similar plots for other diagnosis codes confirmed this negative correlation across most of the codes.

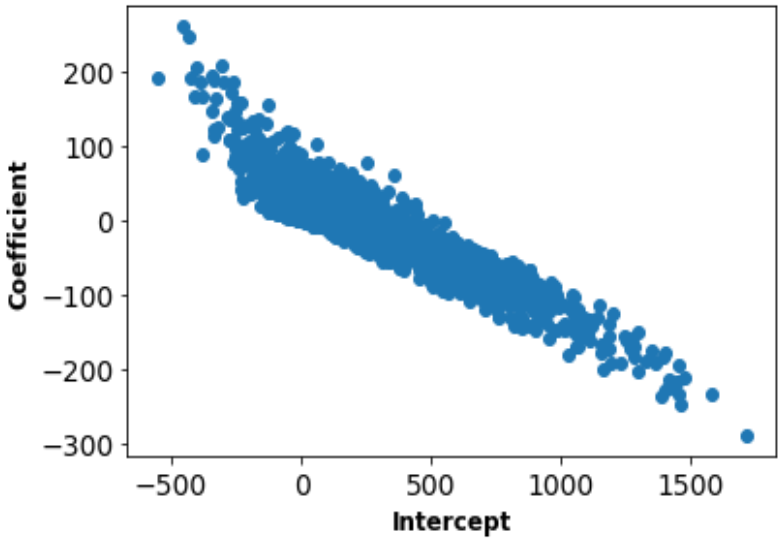

Figure 30: Correlation between the calculated intercept and coefficient for one Elixauaser code representing congestive heart failure. Similar plots were created for other codes but are not shown here due to space limitation.

Also, a scatterplot between the number of visits and the calculated intercept was created for the above Elixauaser code (See Figure 31). The vertical and horizontal axes correspond to the number of visits and the intercept, respectively.

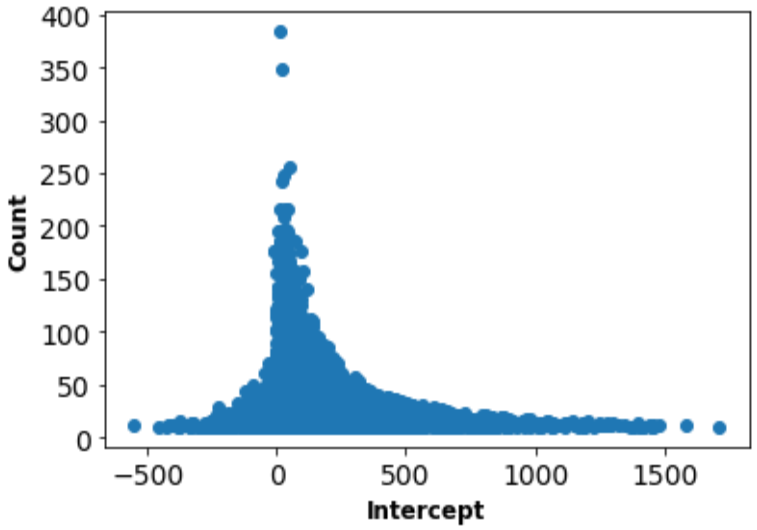

Figure 31: Correlation between the calculated intercept and the number of visits for a Elixauaser code representing congestive heart failure. Similar plots were created for other codes but are not shown here due to space limitation.

The plot clearly showed that the intercept is closer to zero when the number of visits is large, while large value of intercept occurs when there are few visits corresponding to the diagnosis code. One may argue that since the average number of visits was larger among positive cases compared to negative cases (See Table 17), the average intercept should be smaller for cases with positive labels. It should be noted that it is not only the number of visits that correlates with intercept. In fact, large intercept happens when a patient is diagnosed with that health condition long time before prediction time with no or few visits until more recently when he/she starts having visit at shorter time intervals. These cases are more likely to have positive labels.

Table 17: Average number of visits among positive ang negative labels for the four outcomes. This average is reported for diagnoses with more than 10 visits for comparison with the calculated intercept.

Outcome	Positive Label	Negative Label
Mortality	33.88	30.68
High Utilization	36.40	29.19
CKD	33.01	28.37
CHF	30.13	28.51

Trajectory Model Construction

A number of models were created and investigated for the outcomes described in the previous chapter. Models were first created using: 1) patient demographics and Trajectory coefficients only (*Coef*) with a total of 38 attributes; 2) patient demographics, Trajectory coefficients and intercepts (*Coef_Int*) with a total of 68 attributes. Then these Trajectory-based attributes were combined with *Min* and *Max* attributes from *TMMR* resulting in *Com_Coef* models with total of 98 attributes and *Com_Coef_Int* models with total of 128 attributes. In addition, Trajectory-based attributes were constructed using 1 as well as 5 years of data and combined with *TMMR* attributes to determine if the size of observation window impact the constructed trajectories and models. The models were defined as *Com_Coef_1yr*, *Com_Coef_Int_1yr*, *Com_Coef_5yr*, and *Com_Coef_Int_5yr*, with *1yr* and *5yr* representing 1 year and 5 years of data, respectively. These models were compared with the *TMMR*. Two tailed paired t-tests were used when applicable to determine the level of significance ($p<0.05$). For completeness, the performance of Binary-based models was also reported.

Table 18: Comparison of the performance of different Trajectory-based models in predicting mortality, high utilization, CKD, and CHF. The models include *Coef, Coef_Int, Com_Coef, Com_Coef_Int, Com_Coef_1yr, Com_Coef_Int_1yr, Com_Coef_5yr, and Com_Coef_Int_5yr*. The models were compared with Binary and Temporal Min-Max Representation results.

		RF				GB				LR				DT			
		AUC	**Acc**	**Prec**	**Rec**	**AUC**	**Acc**	**Prec**	**Rec**	**AUC**	**Acc**	**Prec**	**Rec**	**AUC**	**Acc**	**Prec**	**Rec**
Mortality	**Binary**	.684	.923	.13	.01	.738	.927	.15	0	.727	.927	.429	.002	.534	.864	.134	.157
	TMMR	.753	.927	.575	.034	.778	.928	.557	.072	.729	.927	.502	.009	.566	.876	.178	.193
	Coef	.673*	.924*	.136*	.006*	.707*	.927*	.368*	.007*	.672*	.927	.416	.006	.519*	.883*	.136*	.111*
	Coef_Int	.678*	.926*	.299*	.004*	.707*	.926*	.379*	.009*	.672*	.927	.413	.006	.520*	.882*	.135*	.114*
	Com_Coef	.754	.927	.56	.027	.778	.928	.553	.07	.739	.926	.405	.017*	.569	.875	.181	.200
	Com_Coef_Int	.753	.928	.62	.031	.777	.928	.551	.068	.739	.926	.408	.016*	.567	.875	.177	.194
	Com_Coef_1yr	.752	.928	.583	.033	.777	.928	.566	.073	.737	.926	.452	.03*	.565	.874	.173	.191
	Com_Coef_Int_1yr	.752	.928	.595	.031	.777	.928	.566	.072	.736	.926	.459	.029*	.569	.875	.180	.200
	Com_Coef_5yr	.755	.927	.591	.029	.778	.928	.569	.072	.74*	.926	.423	.02*	.57	.874	.18	.202
	Com_Coef_Int_5yr	.752	.928	.592	.029	.778	.928	.555	.071	.74	.926	.422	.019*	.564	.874	.173	.192
High Utilization	**Binary**	.742	.894	.381	.086	.785	.901	.561	.065	.782	.901	.55	.059	.557	.833	.213	.246
	TMMR	.829	.91	.699	.189	.838	.911	.662	.231	.794	.901	.566	.074	.611	.854	.294	.322
	Coef	.748*	.902*	.552*	.12*	.785*	.906*	.64*	.136*	.772*	.902*	.569	.108*	.501*	.851*	.256*	.255*
	Coef_Int	.756*	.905*	.631*	.12*	.785*	.906*	.646*	.138*	.772*	.902*	.569	.108*	.509*	.853*	.266*	.265*
	Com_Coef	.829	.911	.714	.187	.838	.911	.668	.236	.812*	.904*	.588	.151*	.614	.854	.297	.328
	Com_Coef_Int	.83	.911	.715	.188	.838	.912	.672	.235	.812*	.904*	.587	.151*	.614	.855	.299	.328
	Com_Coef_1yr	.832	.912	.723	.197	.84	.913	.689	.244	.819*	.909*	.638*	.209*	.617	.857	.307	.332
	Com_Coef_Int_1yr	.834	.912	.714	.204*	.84	.913	.687	.241	.819*	.909*	.639*	.208*	.62*	.859*	.313*	.337*
	Com_Coef_5yr	.831	.911	.717	.19	.838	.912	.67	.235	.816*	.904	.574	.158*	.614	.855	.298	.329
	Com_Coef_Int_5yr	.831	.911	.719	.194	.838	.912	.672	.235	.816*	.904	.575	.158*	.612	.854	.295	.325
CKD	**Binary**	.582	.928	.119	.011	.639	.933	0	0	.607	.933	0	0	.523	.871	.079	.087
	TMMR	.623	.933	.323	.006	.65	.933	.194	.002	.607	.933	0	0	.536	.871	.084	.094
	Coef	.569*	.929*	.085*	.006	.605	.933	.083	0	.573*	.933	0	0	.496*	.89*	.083*	.064*
	Coef_Int	.568*	.933	.02*	0*	.602*	.933	.14	.001	.574*	.933	0	0	.495*	.89*	.081	.062*
	Com_Coef	.617	.933	.343	.006	.65	.933	.203	.002	.615	.933	0	0	.539	.871	.09	.101
	Com_Coef_Int	.617	.933	.367	.006	.649	.933	.289	.003*	.615	.933	0	0	.537	.872	.087	.096
	Com_Coef_1yr	.625	.933	.362	.006	.651	.933	.233	.002	.613	.933	.100	.000	.540	.871	.090	.102
	Com_Coef_Int_1yr	.619	.933	.340	.006	.651	.933	.263	.003	.613	.933	.100	.000	.538	.873	.089	.098

	Com_Coef_5yr	.607	.94*	.362	.011*	.64	.94*	.225	.003*	.608	.94*	0	0	.549*	.887*	.069*	.072*
	Com_Coef_Int_5yr	.608	.94*	.373	.01	.639	.94*	.128	.002*	.609	.94*	0	0	.554*	.89*	.077	.078*
CHF	**Binary**	.581	.929	.075	.008	.628	.935	0	0	.628	.935	0	0	.519	.875	.086	.094
	TMMR	.623	.934	.05	.001	.647	.934	.3	.003	.647	.934	.3	.003	.536	.876	.098	.109
	Coef	.568*	.93*	.051	.004*	.604*	.935	.15	.001	.604*	.935	.15	.001	.49*	.895*	.089	.065*
	Coef_Int	.573*	.934	.064	.001	.6*	.934	.158	.001	.6*	.934	.158	.001	.493*	.894*	.095	.072*
	Com_Coef	.621	.934	.067	.001	.647	.934	.319	.004	.608	.935	0	0	.531	.874	.088	.1
	Com_Coef_Int	.625	.934	.067	.001	.646	.935	.412	.005	.608	.935	0	0	.529	.874	.086	.096*
	Com_Coef_1yr	.624	.934	.067	.001	.646	.934	.229	.004	.605	.935	.000	.000	.536	.874	.096	.110
	Com_Coef_Int_1yr	.624	.934	.098	.001	.645	.934	.227	.004	.605	.935	.000	.000	.533	.875	.091	.103
	Com_Coef_5yr	.62	.934	.05	0	.646	.934	.366	.003	.608	.935	0	0	.534	.876	.094	.104
	Com_Coef_Int_5yr	.622	.934	.05	0	.645	.934	.208	.003	.608	.935	0	0	.536	.876	.097	.11

*Coef and Coef_*Int refer to models constructed on coefficient and coefficient +intercept, respectively. *Com_Coef* and *Com_Coef_Int* refer to models in which coefficient and coefficient +intercept attributes were added to Min-Max models, receptively. Finally, 1yr and 5yr show that Trajectory-based attributes were created on 1 and 5 years of data, respectively. * Indicates significance ($p < 0.05$) of different models compared with Temporal Min-Max models.

As shown in Table 18, models that only used Trajectory-based attributes (either coefficient or in combination with intercept) were on average not performing better than Binary or Temporal Min-Max Representation-based models in terms of AUC. It is likely that Binary and Temporal Mi-Max models provide a more comprehensive representation of diagnosis codes in predicting the four outcomes. In fact, trajectories were constructed only on diagnoses with large numbers of claims, thus limiting their ability to represent the codes. It is found that trajectories slightly improved the recall or precision of some of the models compared to Binary Representation method. It was also observed that the combination of these attributes with *TMMR* models' attributes does not on average improve the performance of the models and the results were consistent on models with trajectories constructed on 1 and 5 years of data. The results however showed that this combination

(addition of coefficient or coefficient+intercept) performs better using Logistic Regression in predicting mortality and high utilization. When Logistic Regression was used to predict mortality, the best results occurred when trajectories were constructed using 5 years of data, while it was 1 year in predicting high utilization.

Even though the results suggested that *TJR* was not on average performing better than *TMMR* and Binary Representation methods according to population-level metrics, individual-level evaluation of the above models indicated that *TJR* can work better for certain patients. Therefore, the next two sections aim at understanding what types of patients are more accurately described with trajectories as compared to *TMMR*.

Comparison of Number and Average Value of Coefficients

One experiment to compare representations on individual-level (Input Comparison) involved determining the value and the number of constructed coefficients between the two representation methods. For this purpose, the average value of coefficients and the average number defined coefficients (diagnosis codes with more than 10 datapoints) were compared across Temporal Min-Max Representation (*TMMR*) and Trajectory Representation (*TJR*). The *Com_Coef_Int* models (combination of *TMMR* and Trajectory attributes) were used for Trajectory Representation models and were called *TJR* models for simplicity. These experiments were applied on both correct and superior prediction as well as superior prediction with the difference in output probability greater than 5%. Due to the similarity of the models, the results were compared only for superior prediction with more than 5% difference in output probabilities. The other comparisons results can be found in Appendix section (Table 27 to Table 30). The average number of coefficients and the

average value of the coefficients across true label of each outcome were compared in Table 19 and Table 20, respectively and Mann-Whitney U test was used to determine the significance of the results.

As shown in Table 19, the number of constructed coefficients was larger for *TJR* among cases with positive labels and smaller among cases with negative labels in most comparisons except for DT of *Problem 1* and GB of *Problem 4*, for which the results were not significant. The results shown in Table 20 also indicated that the average value of coefficients across all diagnoses was negative for both representation methods, with the absolute value being larger for *TJR* representation among positive cases and smaller among negative cases. The results, however, were not significant for most comparison in predicting congestive heart failure (*Problem 4*).

Larger number as well as absolute value of coefficients in predicting positive cases suggests that the addition of Trajectory-based attributes to *TMMR* models works better when the patient has one average more diagnosis codes represented by trajectories and faster decline in health status. It should be noted that the intercept was not compared for this experiment, as it cannot not provide meaningful insight about patient's overall health status.

Table 19: Comparison of the number of coefficients for Temporal Min-Max Representation (*TMMR*) vs. Trajectory Representation (*TJR*) based on superior prediction (difference in probability greater than 5%)

***Problem 1*-Mortality**				
	Positive Label		**Negative Label**	
Alg	***TJR***	***TMMR***	***TJR***	***TMMR***
RF	4.54*	3.64	2.96*	3.59
GB	5.94*	5.05	5.02*	5.69

LR	6.75*	3.65	3.72*	6.09
DT	4.42	4.11	3.19	3.31
***Problem 2*-High Utilization**				
RF	5.89*	4.25	2.74*	4.04
GB	6.92*	4.65	3.94*	6.00
LR	7.30*	3.14	2.35*	6.27
DT	5.23*	4.91	3.29*	3.48
***Problem 3*-CKD**				
RF	3.55*	2.56	2.16*	2.90
GB	5.77*	3.91	3.68*	4.79
LR	6.19*	2.29	2.73*	5.64
DT	3.26*	2.89	2.38*	2.63
***Problem 4*-CHF**				
RF	2.94*	2.50	1.92*	2.58
GB	4.66	4.46	3.49*	4.29
LR	6.09*	2.89	3.06*	5.80
DT	2.91*	2.66	2.09*	2.22

Table 20: Comparison of the coefficient average for *TMMR* vs. *TJR* based on superior prediction (difference in probability greater than 5%).

***Problem 1*-Mortality**				
	Positive Label		**Negative Label**	
Alg	***TJR***	***TMMR***	***TJR***	***TMMR***
RF	-6.87*	-6.05	-3.81*	-3.03
GB	-12.28*	-6.93	-7.90*	-9.34
LR	-8.18*	-6.03	-5.13*	-5.82
DT	-4.31	-6.30	-4.00	-3.69
***Problem 2*–High Utilization**				
RF	-6.86*	-6.30	-3.52*	-5.09
GB	-6.15	-7.89	-5.50*	-5.82
LR	-6.29*	-5.66	-3.07*	-4.85
DT	-6.93*	-5.95	-3.48*	-4.44
***Problem 3*-CKD**				
RF	-6.68*	-3.14	-2.97*	-4.37
GB	-8.60	-5.03	-5.72	-6.84
LR	-4.38*	1.60	-1.84*	-5.41
DT	-5.16	-2.65	-3.33	-3.09
***Problem 4*-CHF**				
RF	-5.77	-4.27	-2.84*	-2.48
GB	-0.31	-6.78	-5.43	-6.46
LR	-6.04	-2.98	2.12*	-5.17
DT	-3.57	-3.20	-3.91	-3.06

Comparison of Mean Absolute Error

Second individual-level comparison experiment examined how the difference in the output probabilities (absolute difference) of the two representation methods affects the Mean Absolute Error (MAE). The rationale behind this comparison is to test which method performs better when there is a larger disagreement between *TMMR* and *TJR*.

For this purpose, MAE was calculated for both *TMMR* and *TJR* by varying the difference in output probability of the two models. Figure 32 compares the MAE of the four prediction problems for both *TMMR* and *TJR* methods. The horizontal axis refers to the difference in the output probability of the two methods and vertical axis refers to the Mean Absolute Error on the selected patients. The results suggested that the change in MAE by varying the difference in output probability is problem dependent. In general, increasing the output probability difference for the two representation methods increases the MAE difference, except for the DT algorithm, where the change in output probability does not impact MAE. However, the magnitude of the difference depended on the algorithm and the outcome. Furthermore, while the plots of LR-based models suggest that the *TJR* models have larger MAE compared to the *TMMR* models specifically on larger differences in output probability, the results were inconclusive for RF and GB-based models. For example, the results of the GB algorithm clearly showed that while the MAE is always smaller for *TJR* in predicting high utilization, it was larger in predicting congestive heart failure (CHF). In this case, *TJR* might be a better representation method for predicting high utilization, but not for predicting CHF.

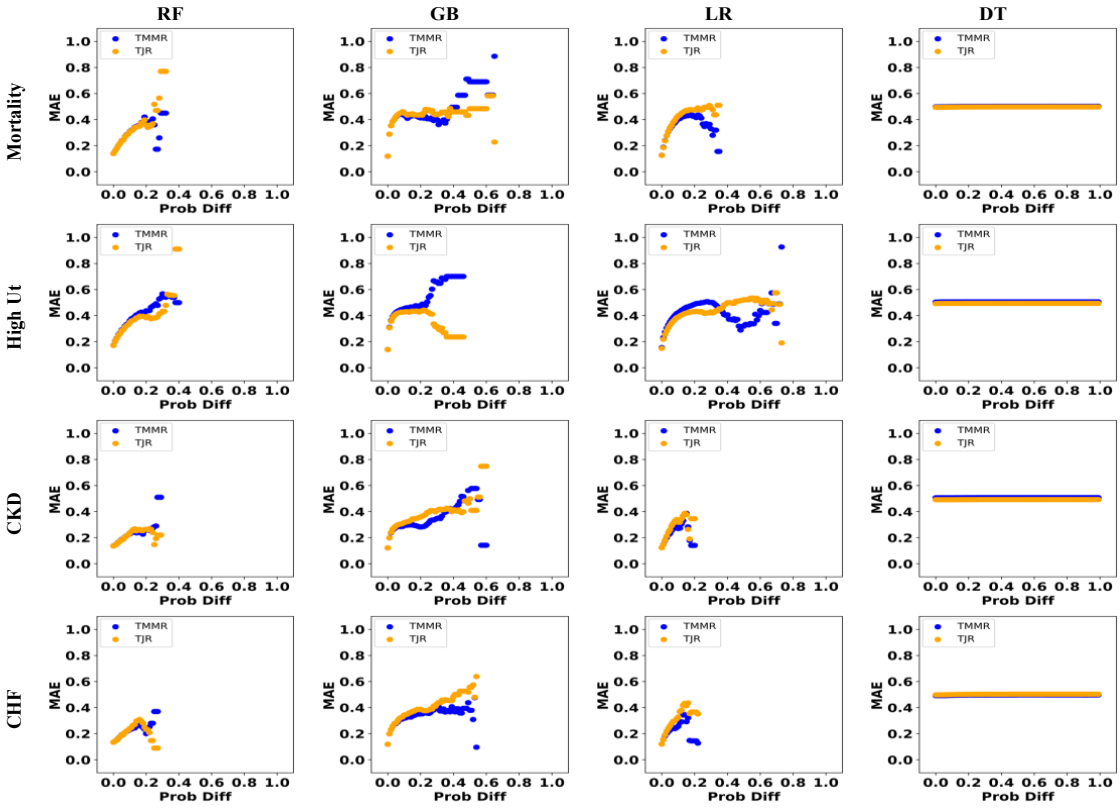

Figure 32: Comparison of the Mean Absolute Error (MAE) by changing the difference in output probability of the two different representation systems. Blue and orange lines indicate *TMMR* vs. *TJR* Representations, respectively. Shape of the plots indicate that changes in MAE by varying the difference in output probability is problem dependent.

Conclusion on Trajectory Representation

In this chapter, another method of the administrative codes' representation called Trajectory Representation was introduced. The assumption for representation was that the time between visits indicate changes in patients' health over time. In this method, each diagnosis code (here Elixhauser codes) was represented with coefficient and intercept of the fitted line to the time between diagnosis data points. The population-level analysis based on standard model accuracy indicated that Trajectory Representation does not

provide a better representation of the codes compared to *TMMR* methods. However, *TJR* worked better for certain patients based on individual-level comparison of *TMMR* and *TJR* models. Based on the average number and mean value of coefficients, as well as the Mean Absolute Error (MAE), the *TJR* method appears to perform better for certain patients than *TMMR*.

In summary, the investigation of the two introduced representation methods (*TMMR* and *TJR*) showed that despite some patterns observed from the experiments, there are no specific rules to indicate which representation method should be used. Therefore, when these codes are used in supervised machine learning, different parameters should be tested for optimal results.

CONCLUSION

Administrative Data Preprocessing and Diagnosis Representation

The problem of representing data before applying machine learning methods has a significant impact on the quality of the models applied in patient care and health system management, as well as health policy and payment. Previous studies have observed similar effects of representation on performance of ML methods in health-related applications, particularly in claims processing (Lynam et al., 2020). A crucial factor in determining model quality is the ability to extract the right information from raw data by applying proper transformations and represent the current knowledge in the data that could allow better prediction. When analyzing complex data such as those from Electronic Medical Records, or medical claims considered here, the process is not trivial and involves construction of flat representation from multi-dimensional and highly temporal databases. In such data preprocessing, the goal is to reduce the complexity of the data and extract/construct the relevant attributes from the data, which can subsequently be used for analysis.

This dissertation considered data preprocessing steps as part of model construction and optimization: *the way data are represented affects specific types of models either on the quality of the models or prediction of specific instances in the data*. The presented study focused on data preprocessing steps applied to transform raw medical claims data into final analytic files before applying ML methods. While the dissertation specifically

addressed representation of medical diagnoses, it can be generalized into other data elements.

The claims data contains information in the form of codes, dates, text, symbols, and so forth. In contrast to other types of healthcare data, in medical claims data much of the information is structured, with little or no information provided as free text. There is an irregularity in claims data that corresponds to when the patient receives services. In fact, many claims datasets don't indicate when the services are actually provided; data are only available about when a claim is recorded. The data are also potentially censored on both sides because of benefit eligibility and events such as death or discontinuation of treatment. Since most ML algorithms cannot handle multiple records within a relational database like claims data, some summary functions including Boolean representation and counting the occurrence of each event are used to aggregate the data. In general, the difficulty in analyzing claims data comes from complexity of coding systems used that potentially leads to extremely high dimensionality, temporality of the data, missing information (i.e., undercoding of data), as well as noisy information (i.e., overcoding of certain data).

Among the types of structured data in claims are administrative codes (medical codes) such as International Classification of Diseases (ICD-9 or ICD-10) codes, Current Procedural Terminology (CPT), Healthcare Common Procedure Coding System (HCPCS) codes and other specialized codes used for specific claim types. These codes are used to classify disease, comorbidities, and procedures when providing services in healthcare settings. They are popular source of information in predicting health outcomes with the common assumption that the presence of the codes in claims is representative of patients'

true status. Claims data can be viewed as a sequence of claims that include one or more of these codes recorded over time. Consequently, this dissertation focused on the representation of the administrative codes specific to diagnoses in data preprocessing step of supervised machine learning.

Methodological Gap in Administrative Codes Representation

Despite wide use of administrative codes in constructing ML-based models, there is still a gap in better representation of administrative codes that could enhance the quality of the models. This dissertation discussed and investigated some of the common administrative codes representation methods including: 1. Binary Representation in which binary indicators (dummies) are used over a timeframe, 2. Binary Representation with Multiple Time bins in which observation window is divided into multiple bins and then Binary Representation is applied for each bin separately, 3. Enumeration Representation in which the number of visits related to a specific health condition (code) is counted within a predefined window. Due to the longitudinal nature of claims data, temporal information could be better captured in representing these codes in data i.e., incorporating the time from the occurrence of the disease or capturing the changes in prognosis of the disease over time and the representation of these codes by using temporal information has gained popularity recently. Therefore, in this dissertation, 'Temporal Min-Max Representation' *(TMMR)* and 'Trajectory Representation' *(TJR)* as two additional representation methods were described in detail, which focus on capturing the heterogeneity and hidden temporal information in the data. Also, a major gap with regard to the use of the administrative codes is the lack of little systematic research on how these codes should be preprocessed (represented) before

being model construction. These standard methods mentioned above are extensively used in supervised machine learning without determining if they are appropriate methods for the problem at hand. It is assumed that the method of representing data could be impacted by many factors including the type of algorithm, outcome, size of observation window, the type of administrative code, characteristics of inputs etc.

Temporal Min-Max Representation

This dissertation focused on two methods in representing administrative codes called Temporal Min-Max Representation *(TMMR)* and Trajectory Representation (*TJR*). In the "Temporal Min-Max Representation" chapter, *TMMR* method was introduced which works by calculating the time from the first ($Code_i^{max}$) and most recent ($Code_i^{min}$) occurrence of the diagnosis to prediction time. The advantage of this representation is the ability to capture long-term effect of chronic health conditions that are present over a long period of time and temporary impact of acute health conditions. The method was first introduced in predicting the Activities of Daily Living (ADLs) and the results on different learning algorithms suggested that this method outperforms Binary Representation method in terms of standard model performance measurement including accuracy, AUC, precision, and recall. The "Evaluation of Temporal Min-Max Representation" chapter focused on comparing *TMMR* and Binary Representation methods using a large-scale experimental evaluation on four classification problems: predicting mortality, high utilization of medical services, chronic kidney disease, and congestive heart failure. The results indicated that *TMMR* outperforms Binary Representation in most cases. However, the optimal data representation is highly dependent on the classification problem, observation window size

(how much historical data are available for patients), model representation, learning algorithm, the predicted outcome, and characteristic of input attributes (number of health conditions and time from diagnosis and demographic forctors). For example, the results suggest that *TMMR* method performed better on positive cases (those with bad predicted outcome) with smaller time between diagnosis and larger number of diagnosis code, while the pattern was opposite on negative cases. The sensitivity analysis of the models also showed the difference between constructed *TMMR* or Binary Representation-based models. Specifically, *TMMR* models were on average more sensitive across codes present once in the data. Furthermore, the sensitivity of the models differed between *Min* and *Max* attributes, with *Min* attributes exhibited negative correlation with changes in output probability and *Max* attributes exhibited small positive or close to zero correlation. It was also shown that even though the negative and positive correlation between *Min* and *Max* attributes was independent of the outcome, the magnitude of sensitivity was larger for the top predictor of each outcome.

Trajectory Representation

In "Trajectory Representation" chapter, *TJR* method was introduced in representing the administrative codes by using the time between visits related to specific diagnosis. Longer intervals between encounters can indicate that a patient's health status is improving for a specific disease, while shorter intervals indicate his health status is deteriorating. In this method, the trajectory of disease was represented by calculating the coefficient and intercept of a fitted line to time between visits data points. Different variants of constructed Trajectory attributes were applied to the four classification problems and were compared

with *TMMR* and Binary Representation methods. These variants included coefficients only, coefficient and intercept, and combination of Trajectory-based attributes with *TMMR* attributes in which Trajectory attributes were constructed using 18, 5 and 1 years of data. The results suggested that using the Trajectory-based attributes alone does not provide a better representation of codes compared to that of *TMMR* and Binary methods. Also, the combination of these attributes with *TMMR* attributes does not on average improve the AUC of models. However, the individual-level comparison of *TMMR* and *TJR* (combination of *TMMR* and Trajectory attributes) methods based on the average number of coefficients, the mean value of coefficients and the Mean Absolute Error (MAE) showed that *TJR* method can perform better for specific patients under specific circumstances.

Representation Methods Evaluation

One of the major focuses of this dissertation is on the systematic and comprehensive comparison of different ML-based models specific to the representation of the administrative codes. In this dissertation, the evaluation and comparison of the representation methods were applied on both population and individual level. The population level comparison uses standard model performance metrics including AUC, accuracy, precision, and recall for classification problems and Mean Square Error (MSE), Mean Absolute Error (MAE), and correlation coefficient in regression problems. With ML field being dominated with statistical methods, researchers often assume that statistical model evaluation and comparison are sufficient. This cannot be further from the truth, especially in the medical or health care fields in which every ‘test case’ is a patient whose treatments and potentially life-altering decisions may be made based on predictions. The

models may be similar or even identical in terms of the statistical metrics but can be very different on individual level. Comparing models on an individual basis can help identify certain groups of patients (even if small) for which a model is most effective. Unfortunately, there are no standard methods for comparing models in detail on an individual level in machine learning.

The methods of individual level comparison of the representation methods used in the dissertation could be divided into two levels: Input and Output comparison. Model Correlation Plots (MCPs), output probability table through aggregating the results, and distribution of correct classified cases, MAE by varying the difference in output probability were used as Output comparison methods to visually compare individual cases based on their output probability corresponding to each representation method. In addition, Input Comparison methods were used to investigate how the representation of input attributes correspond to differences in outputs. The comparison was done by looking into the time between diagnosis and prediction time, number of health conditions, average number of defined coefficients, and average value of coefficients on cases that are correctly classified or better classified (superior prediction) by either of the representation methods. In general, classification-based models are compared based on an accurate prediction on a specific threshold. However, the comparison could be made based on superior prediction meaning that the cases of the two models are compared if they are better predicted in terms of output probabilities. This better (superior) prediction could be made based on any difference in the output probability or within a specific tolerance. In fact, superior prediction comparison allows for better understanding of the nuances between the representation methods. Also,

other than the characteristics of diagnosis codes, demographic factors including age and race were compared between the representation methods. It should be noted that the selection of the specific metrics for comparison should be done carefully to gain maximum insights about the models. In this dissertation, the Input Comparison methods in either the *TMMR* vs. Binary Representation or *TMMR* vs. *TJR* are justified by the problem at hand.

Finally, the methodology of model comparison and experimental evaluations presented in this dissertation focuses on representation of administrative codes. However, they can serve as a general framework in which models are described by their inputs, models, and their corresponding outputs. The framework can be used to study model performance, explainability, fairness, and other factors that may ultimately lead to end users' trust and model adoption.

Contribution

This methodological dissertation focused on pushing the understanding of representation of administrative codes generated in healthcare. Thus, its main contributions are methodological and experimental, but the constructed models also contribute to the considered application areas of mortality, high utilization of medical services, and chronic conditions such as chronic kidney disease and congestive heart failure. More specifically, the main contributions of the current dissertation can be summarized as:

1. A detailed review of methods used for representing administrative codes in supervised machine learning to construct models for predicting patient outcomes.

2. Study of two methods of representing the administrative codes called Temporal Min-Max Representation (*TMMR*) and Trajectory Representation (*TJR*), which are constructed based on complex temporal information in claims data.
    a. Detailed study of *TMMR* method constructed by calculating the time from first and most recent diagnosis and its comparison with Binary Representation method.
    b. Detailed study of *TJR* method, which assumes that the trajectory of illness is proportional to time between encounters for a given condition and its comparison with Binary and *TMMR* methods. The method has been introduced and studied in different variants including coefficient of the fitted line only, coefficient and intercept, the combination of the coefficient/coefficient+intercept attributes with *TMMR*-based attributes and construction of *TJR* with different window sizes.
3. Detailed comparison of representations on both population and individual level. Population comparison used simple standard accuracy measures, while individual level comparisons were made based on models' inputs and their outputs including output probability, time from diagnosis to prediction, number of present health conditions, window size/history, how missing codes are represented, demographic factors, number and average of coefficients, and MAE with respect to output probabilities.
4. Construction of models to solve practical problems in health care including prediction of mortality, high utilization of medical services and chronic kidney

disease (CKD) and congestive heart failure (CHF). The quality of the first two problems was comparable to results published in the literature. While the model quality in predicting CKD and CHF was below what may be needed in clinical practice, the models provide insights into the possibility of using administrative data to predict these chronic conditions.

In summary, the results of this dissertation indicated that there is no best administrative code representation method in supervised machine learning. These methods should be experimentally tested using different parameters to achieve optimal results in predicting health outcomes.

This dissertation is highly experimental and includes a large number of models and their evaluations. During this study, about 30,000 models were constructed in comparing the representation methods, each being used in the process of study and optimization, which accounted for weeks of continuous CPU utilization. The analyses have been done mainly in Python 3.6 and data preprocessing in PostgreSQL.

Limitation and Future Works

This dissertation is intended to be a significant step toward designing a systematic study of using administrative codes in health data. There is still a need to further examine the representation issues for multi-class classification, regression, and unsupervised learning. Even within supervised learning, there is a practically unlimited number of ways to transform raw claims data into flat tables from ML algorithms. Future works include designing' *Perfect Representation'* in which each diagnosis is optimized individually. This

means that a combination of two or more of the representation methods is used, depending on which method is most appropriate for a particular administrative code.

One limitation of the presented work is that it is applicable in settings in which longitudinal information of patients are collected over multiple years. Therefore, large and well-established databases of claims or EHRs data are required. Furthermore, claims data are often expensive to purchase for research purposes, thus are often limited to short periods of time. In fact, this dissertation highlights the importance of longitudinal data to create high quality models. It is the authors' opinion that it is more beneficial to use data collected over longer periods of time when limited resources are available.

Also, the presented work did not completely take into consideration censoring of data based on data availability (multiple payers, insurance eligibility, etc.), as many nuances of payment system need to be accounted for. The other limitation is that some comparisons were not possible due to the lack of cases correctly predicted by one of the representation methods. The issue was specific to predicting CKD and CHF which had relatively smaller sample size. Therefore, more robust conclusion could be achieved by using larger datasets.

Finally, the presented work focused on 'traditional' machine learning algorithms (Gradient Boosting, Random Forest, Logistic Regression and Decision Trees) and results are most likely generalizable to similar methods. While Neural-network approaches were investigated in this dissertaion, they were limited to simple feed-forward perceptron (multi-layered) and did not account for recent advances in deep learning. Deep learning methods and more specifically Recurrent Neural Networks (RNN) such as Long Short-Term

Memory (LSTM) (Hochreiter & Schmidhuber; 1997) and Gated Recurrent Units (GRU) (Cho et al., 2014) are attractive alternatives for analyzing highly temporal claims. RNNs can be trained on the actual sequences of claims rather than aggregated data within selected windows. A future work would apply data representation using RNNs, especially in settings where large amounts of data are available.

APPENDIX

Table 21: Full list of Single-Level CCS diagnosis codes derived from AHRQ website: https://www.hcup-us.ahrq.gov/toolssoftware/ccs/ccs.jsp. The blank values in Chronic/Non-chronic column mean that the code does not belong to any of these categories.

CCS Code	CCS Code Description	Chronic/Non-chronic
1	Tuberculosis	non-chronic
2	Septicemia (except in labor)	non-chronic
3	Bacterial infection; unspecified site	non-chronic
4	Mycoses	non-chronic
5	HIV infection	chronic
6	Hepatitis	chronic
7	Viral infection	chronic
8	Other infections; including parasitic	non-chronic
9	Sexually transmitted infections (not HIV or hepatitis)	chronic
10	Immunizations and screening for infectious disease	non-chronic
11	Cancer of head and neck	non-chronic
12	Cancer of esophagus	non-chronic
13	Cancer of stomach	chronic
14	Cancer of colon	chronic
15	Cancer of rectum and anus	chronic
16	Cancer of liver and intrahepatic bile duct	non-chronic
17	Cancer of pancreas	
18	Cancer of other GI organs; peritoneum	chronic
19	Cancer of bronchus; lung	chronic
20	Cancer; other respiratory and intrathoracic	non-chronic
21	Cancer of bone and connective tissue	
22	Melanomas of skin	non-chronic
23	Other non-epithelial cancer of skin	chronic
24	Cancer of breast	
25	Cancer of uterus	non-chronic
26	Cancer of cervix	non-chronic
27	Cancer of ovary	non-chronic

28	Cancer of other female genital organs	chronic
29	Cancer of prostate	non-chronic
30	Cancer of testis	non-chronic
31	Cancer of other male genital organs	non-chronic
32	Cancer of bladder	non-chronic
33	Cancer of kidney and renal pelvis	chronic
34	Cancer of other urinary organs	non-chronic
35	Cancer of brain and nervous system	non-chronic
36	Cancer of thyroid	chronic
37	Hodgkin`s disease	chronic
38	Non-Hodgkin`s lymphoma	chronic
39	Leukemias	chronic
40	Multiple myeloma	chronic
41	Cancer; other and unspecified primary	chronic
42	Secondary malignancies	chronic
43	Malignant neoplasm without specification of site	chronic
44	Neoplasms of unspecified nature or uncertain behavior	chronic
45	Maintenance chemotherapy; radiotherapy	chronic
46	Benign neoplasm of uterus	
47	Other and unspecified benign neoplasm	non-chronic
48	Thyroid disorders	chronic
49	Diabetes mellitus without complication	chronic
50	Diabetes mellitus with complications	chronic
51	Other endocrine disorders	chronic
52	Nutritional deficiencies	
53	Disorders of lipid metabolism	
54	Gout and other crystal arthropathies	chronic
55	Fluid and electrolyte disorders	non-chronic
56	Cystic fibrosis	chronic
57	Immunity disorders	chronic
58	Other nutritional; endocrine; and metabolic disorders	chronic
59	Deficiency and other anemia	chronic
60	Non-chronic posthemorrhagic anemia	

61	Sickle cell anemia	chronic
62	Coagulation and hemorrhagic disorders	chronic
63	Diseases of white blood cells	chronic
64	Other hematologic conditions	chronic
76	Meningitis (except that caused by tuberculosis or sexually transmitted disease)	non-chronic
77	Encephalitis (except that caused by tuberculosis or sexually transmitted disease)	non-chronic
78	Other CNS infection and poliomyelitis	chronic
79	Parkinson`s disease	
80	Multiple sclerosis	
81	Other hereditary and degenerative nervous system conditions	chronic
82	Paralysis	chronic
83	Epilepsy; convulsions	chronic
84	Headache; including migraine	chronic
85	Coma; stupor; and brain damage	chronic
86	Cataract	chronic
87	Retinal detachments; defects; vascular occlusion; and retinopathy	chronic
88	Glaucoma	chronic
89	Blindness and vision defects	chronic
90	Inflammation; infection of eye (except that caused by tuberculosis or sexually transmitteddisease)	chronic
91	Other eye disorders	chronic
92	Otitis media and related conditions	chronic
93	Conditions associated with dizziness or vertigo	chronic
94	Other ear and sense organ disorders	chronic
95	Other nervous system disorders	chronic
96	Heart valve disorders	chronic
97	Peri-; endo-; and myocarditis; cardiomyopathy (except that caused by tuberculosis or sexually transmitted disease)	chronic
98	Essential hypertension	
99	Hypertension with complications and secondary hypertension	chronic
100	Non-chronic myocardial infarction	chronic
101	Coronary atherosclerosis and other heart disease	chronic
102	Nonspecific chest pain	non-chronic

103	Pulmonary heart disease	chronic
104	Other and ill-defined heart disease	chronic
105	Conduction disorders	chronic
106	Cardiac dysrhythmias	chronic
107	Cardiac arrest and ventricular fibrillation	chronic
108	Congestive heart failure; nonhypertensive	chronic
109	Non-chronic cerebrovascular disease	chronic
110	Occlusion or stenosis of precerebral arteries	chronic
111	Other and ill-defined cerebrovascular disease	
112	Transient cerebral ischemia	
113	Late effects of cerebrovascular disease	chronic
114	Peripheral and visceral atherosclerosis	chronic
115	Aortic; peripheral; and visceral artery aneurysms	chronic
116	Aortic and peripheral arterial embolism or thrombosis	chronic
117	Other circulatory disease	chronic
118	Phlebitis; thrombophlebitis and thromboembolism	chronic
119	Varicose veins of lower extremity	
120	Hemorrhoids	
121	Other diseases of veins and lymphatics	chronic
122	Pneumonia (except that caused by tuberculosis or sexually transmitted disease)	non-chronic
123	Influenza	non-chronic
124	Non-chronic and chronic tonsillitis	chronic
125	Non-chronic bronchitis	non-chronic
126	Other upper respiratory infections	non-chronic
127	Chronic obstructive pulmonary disease and bronchiectasis	chronic
128	Asthma	chronic
129	Aspiration pneumonitis; food/vomitus	
130	Pleurisy; pneumothorax; pulmonary collapse	chronic
131	Respiratory failure; insufficiency; arrest (adult)	chronic
132	Lung disease due to external agents	
133	Other lower respiratory disease	chronic
134	Other upper respiratory disease	chronic

135	Intestinal infection	non-chronic
136	Disorders of teeth and jaw	chronic
137	Diseases of mouth; excluding dental	non-chronic
138	Esophageal disorders	chronic
139	Gastroduodenal ulcer (except hemorrhage)	chronic
140	Gastritis and duodenitis	chronic
141	Other disorders of stomach and duodenum	non-chronic
142	Appendicitis and other appendiceal conditions	
143	Abdominal hernia	non-chronic
144	Regional enteritis and ulcerative colitis	
145	Intestinal obstruction without hernia	non-chronic
146	Diverticulosis and diverticulitis	chronic
147	Anal and rectal conditions	non-chronic
148	Peritonitis and intestinal abscess	non-chronic
149	Biliary tract disease	non-chronic
151	Other liver diseases	non-chronic
152	Pancreatic disorders (not diabetes)	
153	Gastrointestinal hemorrhage	chronic
154	Noninfectious gastroenteritis	chronic
155	Other gastrointestinal disorders	chronic
156	Nephritis; nephrosis; renal sclerosis	chronic
157	Non-chronic and unspecified renal failure	
158	Chronic kidney disease	chronic
159	Urinary tract infections	chronic
160	Calculus of urinary tract	non-chronic
161	Other diseases of kidney and ureters	chronic
162	Other diseases of bladder and urethra	chronic
163	Genitourinary symptoms and ill-defined conditions	chronic
164	Hyperplasia of prostate	chronic
165	Inflammatory conditions of male genital organs	non-chronic
166	Other male genital disorders	chronic
167	Nonmalignant breast conditions	non-chronic
168	Inflammatory diseases of female pelvic organs	chronic

169	Endometriosis	
170	Prolapse of female genital organs	chronic
171	Menstrual disorders	
172	Ovarian cyst	
173	Menopausal disorders	chronic
174	Female infertility	
175	Other female genital disorders	chronic
176	Contraceptive and procreative management	non-chronic
177	Spontaneous abortion	non-chronic
178	Induced abortion	non-chronic
179	Postabortion complications	
180	Ectopic pregnancy	non-chronic
181	Other complications of pregnancy	chronic
182	Hemorrhage during pregnancy; abruptio placenta; placenta previa	non-chronic
183	Hypertension complicating pregnancy; childbirth and the puerperium	chronic
184	Early or threatened labor	non-chronic
185	Prolonged pregnancy	non-chronic
186	Diabetes or abnormal glucose tolerance complicating pregnancy; childbirth; or the puerperium	chronic
187	Malposition; malpresentation	non-chronic
188	Fetopelvic disproportion; obstruction	non-chronic
189	Previous C-section	non-chronic
190	Fetal distress and abnormal forces of labor	non-chronic
191	Polyhydramnios and other problems of amniotic cavity	non-chronic
192	Umbilical cord complication	non-chronic
193	OB-related trauma to perineum and vulva	non-chronic
194	Forceps delivery	non-chronic
195	Other complications of birth; puerperium affecting management of mother	non-chronic
196	Other pregnancy and delivery including normal	non-chronic
197	Skin and subcutaneous tissue infections	non-chronic
198	Other inflammatory condition of skin	chronic
199	Chronic ulcer of skin	chronic
200	Other skin disorders	chronic

201	Infective arthritis and osteomyelitis (except that caused by tuberculosis or sexually transmitted disease)	chronic
202	Rheumatoid arthritis and related disease	chronic
203	Osteoarthritis	chronic
204	Other non-traumatic joint disorders	chronic
205	Spondylosis; intervertebral disc disorders; other back problems	chronic
206	Osteoporosis	chronic
207	Pathological fracture	non-chronic
208	Acquired foot deformities	non-chronic
209	Other acquired deformities	chronic
210	Systemic lupus erythematosus and connective tissue disorders	
211	Other connective tissue disease	chronic
212	Other bone disease and musculoskeletal deformities	chronic
213	Cardiac and circulatory congenital anomalies	chronic
214	Digestive congenital anomalies	chronic
215	Genitourinary congenital anomalies	chronic
216	Nervous system congenital anomalies	chronic
217	Other congenital anomalies	chronic
218	Liveborn	non-chronic
219	Short gestation; low birth weight; and fetal growth retardation	non-chronic
220	Intrauterine hypoxia and birth asphyxia	non-chronic
221	Respiratory distress syndrome	
222	Hemolytic jaundice and perinatal jaundice	non-chronic
223	Birth trauma	non-chronic
224	Other perinatal conditions	chronic
225	Joint disorders and dislocations; trauma-related	chronic
226	Fracture of neck of femur (hip)	non-chronic
227	Spinal cord injury	chronic
228	Skull and face fractures	non-chronic
229	Fracture of upper limb	non-chronic
230	Fracture of lower limb	non-chronic
231	Other fractures	non-chronic
232	Sprains and strains	non-chronic

233	Intracranial injury	non-chronic
234	Crushing injury or internal injury	non-chronic
235	Open wounds of head; neck; and trunk	non-chronic
236	Open wounds of extremities	non-chronic
237	Complication of device; implant or graft	chronic
238	Complications of surgical procedures or medical care	chronic
239	Superficial injury; contusion	non-chronic
240	Burns	non-chronic
241	Poisoning by psychotropic agents	non-chronic
242	Poisoning by other medications and drugs	non-chronic
243	Poisoning by nonmedicinal substances	non-chronic
244	Other injuries and conditions due to external causes	non-chronic
245	Syncope	
246	Fever of unknown origin	non-chronic
247	Lymphadenitis	
248	Gangrene	chronic
249	Shock	non-chronic
250	Nausea and vomiting	non-chronic
251	Abdominal pain	non-chronic
252	Malaise and fatigue	chronic
253	Allergic reactions	non-chronic
254	Rehabilitation care; fitting of prostheses; and adjustment of devices	non-chronic
255	Administrative/social admission	non-chronic
256	Medical examination/evaluation	non-chronic
257	Other aftercare	chronic
258	Other screening for suspected conditions (not mental disorders or infectious disease)	non-chronic
259	Residual codes; unclassified	chronic
650	Adjustment disorders	chronic
651	Anxiety disorders	chronic
652	Attention-deficit, conduct, and disruptive behavior disorders	chronic
653	Delirium, dementia, and amnestic and other cognitive disorders	chronic
654	Developmental disorders	chronic

655	Disorders usually diagnosed in infancy, childhood, or adolescence	chronic
656	Impulse control disorders, NEC	chronic
657	Mood disorders	chronic
658	Personality disorders	chronic
659	Schizophrenia and other psychotic disorders	chronic
660	Alcohol-related disorders	chronic
661	Substance-related disorders	chronic
662	Suicide and intentional self-inflicted injury	non-chronic
663	Screening and history of mental health and substance abuse codes	chronic
670	Miscellaneous mental health disorders	chronic
2601	E Codes: Cut/pierceb	
2602	E Codes: Drowning/submersion	
2603	E Codes: Fall	
2604	E Codes: Fire/burn	
2605	E Codes: Firearm	
2606	E Codes: Machinery	
2607	E Codes: Motor vehicle traffic (MVT)	
2608	E Codes: Pedal cyclist; not MVT	
2609	E Codes: Pedestrian; not MVT	
2610	E Codes: Transport; not MVT	
2611	E Codes: Natural/environment	
2612	E Codes: Overexertion	
2613	E Codes: Poisoning	
2614	E Codes: Struck by; against	
2615	E Codes: Suffocation	
2616	E Codes: Adverse effects of medical care	
2617	E Codes: Adverse effects of medical drugs	
2618	E Codes: Other specified and classifiable	
2619	E Codes: Other specified; NEC	
2620	E Codes: Unspecified	
2621	E Codes: Place of occurrence	

Table 22: Full list of Elixhauser (ELIX) codes. The 3.0 version or AHRQ-web ICD-9-CM Elixhauser code was used in this dissertation, in which Cardiac arrhythmias is removed from the list of comorbidities.

ELIX	ELIX Code Description
1	Valvular disease
2	AIDS
3	Solid tumor without metastasis
4	Alcohol abuse
5	Depression
6	Renal failure
7	Psychoses
8	Fluid and electrolyte disorders
9	Diabetes, uncomplicated
10	Lymphoma
11	Weight loss
12	Hypothyroidism
13	Congestive heart failure
14	Deficiency anemia
15	Other neurological disorders
16	Rheumatoid arthritis
17	Drug abuse
18	Hypertension, complicated
19	Pulmonary circulation disorders
20	Paralysis
21	Diabetes, complicated
22	Metastatic Cancer
23	Peptic ulcer disease excluding bleeding
24	Obesity
25	Blood loss anemia
26	Hypertension, uncomplicated
27	Peripheral vascular disorders
28	Liver disease
29	Chronic pulmonary disease
30	Coagulopathy

Table 23: Average coefficient and intercept for each ELIX code across positive and negative labels of mortality

	Coefficient		Intercept	
Code	**Death**	**No Death**	**Death**	**No Death**
Elix1	-9.671*	-4.937	233.829*	193.015
Elix2	-2.642	3.010	63.144	40.454
Elix3	-1.805*	3.860	98.950	77.220
Elix4	-9.625	-0.610	179.611	118.300
Elix5	-4.746*	-3.507	182.162	171.974
Elix6	-5.430*	-4.370	117.235	114.837
Elix7	-2.072*	-0.571	106.938	87.060
Elix8	-14.593*	-9.518	245.399*	214.351
Elix9	-2.087*	-1.494	112.496*	106.669
Elix10	-0.127	0.591	44.075	52.531
Elix11	-7.851	-6.401	179.580	165.122
Elix12	-7.077*	-5.452	211.299	203.268
Elix13	-7.485*	-3.080	168.925*	143.451
Elix14	-9.422*	-6.792	206.255	200.848
Elix15	-7.745*	-4.290	153.731*	133.033
Elix16	-0.293*	0.910	111.358	91.904
Elix17	-6.714	-0.095	159.165	107.208
Elix18	-7.452*	-3.688	212.415*	180.660
Elix19	-2.151*	3.018	116.574*	53.545
Elix20	2.463*	2.925	95.737*	74.252
Elix21	-6.737	-4.557	170.539	159.962
Elix22	-1.784*	-1.050	59.225	67.539
Elix23	-6.957	6.992	158.695	95.811
Elix24	-9.119	-7.481	247.817	224.553
Elix25	-1.979	-3.831	146.925	154.026
Elix26	-3.734*	-2.994	153.287*	148.007

Elix27	-5.059	-5.402	180.930	176.711
Elix28	-3.700*	0.621	142.039*	114.152
Elix29	-8.848*	-4.357	202.051*	177.389
Elix30	-2.745	-0.774	120.335	101.639

Table 24: Average coefficient and intercept for each ELIX code across positive and negative labels of high utilization

	Coefficient		**Intercept**	
Code	**High Ut**	**No High Ut**	**High Ut**	**No High Ut**
Elix1	-10.148*	-4.246	234.299*	187.955
Elix2	-0.534	3.428	57.672	37.799
Elix3	-1.015*	4.273	86.387	78.073
Elix4	-8.346	-0.125	160.210	117.717
Elix5	-4.398*	-3.425	176.358	172.291
Elix6	-4.819*	-4.386	104.512*	120.759
Elix7	-2.673*	-0.175	103.899	85.105
Elix8	-13.434*	-8.805	231.123*	214.115
Elix9	-3.109*	-1.216	108.383	106.911
Elix10	-1.210*	1.419	52.812	50.410
Elix11	-10.356*	-4.961	178.656	163.268
Elix12	-8.366*	-4.986	212.514	202.114
Elix13	-6.909*	-2.610	162.541*	141.900
Elix14	-8.849*	-6.516	189.763*	205.747
Elix15	-7.597*	-4.055	149.792*	132.552
Elix16	-1.635*	1.445	105.869*	90.529
Elix17	-1.900	-0.520	112.521	115.536
Elix18	-8.065*	-2.823	203.836*	177.842
Elix19	-2.792*	4.039	103.446*	47.218
Elix20	2.818*	2.872	83.985*	75.235
Elix21	-6.218*	-4.295	161.616	161.016
Elix22	-1.680*	-0.948	60.951	68.352
Elix23	-0.164	7.458	99.782	100.565
Elix24	-12.092*	-6.259	242.062	221.802
Elix25	-2.889	-3.787	136.227	161.683
Elix26	-4.050*	-2.887	142.784*	149.358

Elix27	-7.854*	-4.653	198.572*	171.177
Elix28	-2.712*	0.842	128.358	115.281
Elix29	-7.370*	-4.279	190.693*	177.824
Elix30	-2.341	-0.582	121.877	98.027

Table 25: Average coefficient and intercept for each ELIX code across positive and negative labels of CKD

	Coefficient		**Intercept**	
Code	**CKD**	**No CKD**	**CKD**	**No CKD**
Elix1	-6.942*	-4.408	204.502	187.899
Elix2	-2.381	7.023	80.673	20.716
Elix3	0.326	4.441	102.089	70.914
Elix4	7.002	-2.278	75.244	125.895
Elix5	2.532	-3.209	162.699	168.872
Elix6	-1.469	3.260	87.091	96.354
Elix7	-2.693	0.059	92.545	84.131
Elix8	-13.325*	-6.254	225.833	200.659
Elix9	-1.539*	-1.178	109.389	110.432
Elix10	0.231	-0.836	28.415	60.165
Elix11	-6.659	-6.781	186.034	167.054
Elix12	-5.320	-4.776	211.693	202.033
Elix13	-3.851	-2.865	152.213	146.383
Elix14	-8.312	-6.428	226.622	210.108
Elix15	-3.060	-3.673	138.591	127.080
Elix16	0.944	1.623	87.829	86.242
Elix17	-6.978	-0.699	112.519	112.557
Elix18	-1.442	-0.866	179.281*	160.860
Elix19	3.116	3.688	68.785*	45.142
Elix20	4.519	2.441	71.691	72.143
Elix21	-3.097	-3.778	155.416	159.213
Elix22	-0.768	-2.827	55.856	74.448
Elix23	-0.445	11.645	112.395	98.129
Elix24	-4.510	-7.692	217.183	221.541
Elix25	-1.187	-3.440	163.220	166.963
Elix26	-3.417*	-3.048	157.829	154.745
Elix27	-5.609	-5.124	183.796*	171.805

Elix28	-2.986*	1.473	131.962	116.496
Elix29	-7.116*	-3.818	203.973*	172.129
Elix30	-2.273	-0.290	111.879	96.399

Table 26: Average coefficient and intercept for each ELIX code across positive and negative labels of CHF

	Coefficient		Intercept	
Code	**CHF**	**No CHF**	**CHF**	**No CHF**
Elix1	-12.577*	-4.572	251.017*	190.020
Elix2	4.746	3.510	47.918	45.062
Elix3	2.808	3.884	68.457	72.936
Elix4	-2.332	-0.141	106.677	107.966
Elix5	1.765	-3.049	131.116	170.097
Elix6	-5.035	-3.710	119.031	111.049
Elix7	-1.299	0.299	89.209	82.926
Elix8	-9.273	-6.041	205.122	177.997
Elix9	-1.864*	-0.927	111.314	106.307
Elix10	-0.870	-0.878	40.148	47.961
Elix11	-7.517	-6.672	166.923	167.849
Elix12	-5.898	-4.707	204.294	198.741
Elix13	-3.872	3.579	182.598	98.492
Elix14	-8.350	-5.847	207.907	194.756
Elix15	-5.961	-3.168	136.209	123.757
Elix16	2.091	1.093	91.944	83.047
Elix17	-8.165	-0.731	136.461	114.489
Elix18	-0.118	-1.776	165.498	164.605
Elix19	2.730	5.127	61.038	33.716
Elix20	8.312	2.458	62.029	66.657
Elix21	-4.748	-4.410	154.514	161.356
Elix22	-3.382*	-0.773	92.849*	58.250
Elix23	-13.617	5.435	149.467	98.887
Elix24	5.884	-8.921	173.517*	219.768
Elix25	1.195	-2.777	116.301	147.316
Elix26	-3.495	-2.902	157.478*	152.404
Elix27	-2.630	-3.996	167.042	160.725
Elix28	4.779	1.416	49.295*	114.974

Elix29	-6.346	-3.798	195.209	178.735
Elix30	1.741	-0.158	77.786	88.494

Table 27: Comparison of the number of coefficients for *TMMR* vs. *TJR* based on correct prediction

Model 1-Mortality				
	Positive Label		**Negative Label**	
Alg	***TJR***	***TMMR***	***TJR***	***TMMR***
RF	6.62	5.76	4.57	5.00
GB	6.26*	4.95	5.85	6.76
LR	8.19*	3.92	4.35*	8.02
DT	4.45	4.14	3.21*	3.35
Model 2-High Utilization				
RF	6.75*	5.41	5.40*	7.00
GB	6.70*	4.56	4.75*	6.50
LR	8.61*	4.18	3.77*	7.98
DT	5.23*	4.91	3.29*	3.48
Model 3-CKD				
RF	0.00	0.00	0.00	0.00
GB	4.43	4.00	4.65	5.86
LR	N/A	N/A	N/A	N/A
DT	3.26*	2.89	2.38	2.63
Model 4-CHF				
RF	N/A	0	0	0.89
GB	5.233	2.0	3.47	4.53
LR	N/A	N/A	N/A	N/A
DT	2.92	2.67	2.12	2.26

Table 28: Comparison of the coefficient average for *TMMR* vs. *TJR* based on correct prediction

***Problem* 1-Mortality**				
	Positive Label		**Negative Label**	
Alg	***TJR***	***TMMR***	***TJR***	***TMMR***
RF	-7.67	-5.14	-7.68	-10.65
GB	-10.81	-7.69	-9.74	-7.73
LR	-10.91	-7.76	-8.61	-6.11
DT	-4.31	-6.30	-4.00	-3.69
***Problem* 2-High Utilization**				
RF	-6.28	-6.90	-6.29*	-7.21
GB	-6.86	-7.75	-4.50*	-7.95
LR	-6.12	-8.49	-6.54*	-6.38
DT	-6.93*	-5.95	-3.48*	-4.44

		***Problem* 3-CKD**		
RF	N/A	N/A	N/A	N/A
GB	-27.46	-8.94	-2.69*	-17.47
LR	N/A	N/A	N/A	N/A
DT	-5.16	-2.65	-3.33	-3.09
		***Problem* 4-CHF**		
RF	N/A	N/A	N/A	0.62
GB	0.72	-1.21	-7.38	-12.46
LR	N/A	N/A	N/A	N/A
DT	-3.57	-3.06	-3.91	-3.06

Table 29: Comparison of the number of coefficients for *TMMR* vs. *TJR* based on superior prediction

		***Problem* 1-Mortality**		
	Positive Label		**Negative Label**	
Alg	***TJR***	***TMMR***	***TJR***	***TMMR***
RF	3.40	3.16	2.05*	2.26
GB	3.09*	3.26	2.25*	1.82
LR	3.42	2.85	2.34*	1.68
DT	4.42	4.11	3.19	3.30
		***Problem* 2-High Utilization**		
RF	4.95*	3.93	1.88*	2.40
GB	4.72*	3.95	1.86*	1.84
LR	5.41*	2.82	1.54*	2.17
DT	5.23*	4.91	3.28*	3.47
		***Problem* 3-CKD**		
RF	2.26*	1.86	1.50*	1.75
GB	2.23*	1.80	1.51*	1.58
LR	2.32*	1.61	1.19*	1.97
DT	3.26*	2.89	2.38*	2.63
		***Problem* 4-CHF**		
RF	1.96*	1.77	1.32*	1.52
GB	1.87*	1.76	1.36	1.33
LR	2.21*	1.39	1.01*	1.77
DT	2.91*	2.66	2.09*	2.22

Table 30: Comparison of the coefficient average for TMMR vs. *TJR* based on superior prediction

		***Problem* 1-Mortality**		
	Positive Label		**Negative Label**	
Alg	***TJR***	***TMMR***	***TJR***	***TMMR***
RF	-4.55*	-5.19	-3.55*	-1.70
GB	-4.55*	-5.19	-3.55*	-1.70

LR	-6.19*	-3.54	-2.62*	-2.69
DT	-4.31	-6.30	-4.00	-3.69
***Problem* 2-High Utilization**				
RF	-5.96*	-5.69	-2.60*	-2.69
GB	-5.63	-6.00	-1.51*	-2.96
LR	-6.09*	-5.35	-2.00*	-2.72
DT	-6.93*	-5.95	-3.49*	-4.44
***Problem* 3-CKD**				
RF	-3.87*	-2.87	-2.10*	-2.50
GB	-3.35*	-3.38	-2.23*	-2.19
LR	-4.06*	-2.43	-1.82*	-2.54
DT	-5.16	-2.65	-3.33	-3.09
***Problem* 4-CHF**				
RF	-3.56	-3.52	-2.27*	-2.12
GB	-3.13*	-3.48	-2.22*	-2.07
LR	-4.45*	-1.90	-1.83*	-2.43
DT	-3.57	-3.20	-3.91	-3.06

REFERENCES

Abràmoff, M. D., Lavin, P. T., Birch, M., Shah, N., & Folk, J. C. (2018). Pivotal trial of an autonomous AI-based diagnostic system for detection of diabetic retinopathy in primary care offices. *NPJ digital medicine*, *1*(1), 1-8.

Ajami, S., & Bagheri-Tadi, T. (2013). Barriers for adopting electronic health records (EHRs) by physicians. *Acta Informatica Medica*, *21*(2), 129.

Aktuerk, D., McNulty, D., Ray, D., Begaj, I., Howell, N., Freemantle, N., & Pagano, D. (2016). National administrative data produces an accurate and stable risk prediction model for short-term and 1-year mortality following cardiac surgery. *International journal of cardiology*, *203*, 196-203.

Alpaydin, E. (2020). *Introduction to machine learning*. MIT press.

Alzoubi, H., Alzubi, R., Ramzan, N., West, D., Al-Hadhrami, T., & Alazab, M. (2019). A review of automatic phenotyping approaches using electronic health records. *Electronics*, *8*(11), 1235.

Ambale-Venkatesh, B., Yang, X., Wu, C. O., Liu, K., Hundley, W. G., McClelland, R., ... & Lima, J. A. (2017). Cardiovascular event prediction by machine learning: the multi-ethnic study of atherosclerosis. *Circulation research*, *121*(9), 1092-1101.

Atherton, J. (2011). Development of the electronic health record. *AMA Journal of Ethics*, *13*(3), 186-189.

Averill, R. F., Goldfield, N., Hughes, J. S., Bonazelli, J., McCullough, E. C., Steinbeck, B. A., ... & Gay, J. (2003). All patient refined diagnosis related groups (APR-DRGs) version 20.0: methodology overview. *Wallingford, CT: 3M Health Information Systems*, *91*.

Azizi, S., Bayat, S., Yan, P., Tahmasebi, A., Nir, G., Kwak, J. T., ... & Mousavi, P. (2017). Detection and grading of prostate cancer using temporal enhanced ultrasound: combining deep neural networks and tissue mimicking simulations. *International journal of computer assisted radiology and surgery*, *12*(8), 1293-1305.

Baldi, P., Brunak, S., Chauvin, Y., Andersen, C. A., & Nielsen, H. (2000). Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, *16*(5), 412-424.

Barker, P. C., & Scherer, J. S. (2019, March 06). Illness trajectories: Description and clinical use. Retrieved January 10, 2022, from https://www.mypcnow.org/fast-fact/illness-trajectories-description-and-clinical-use/

Belard, A., Buchman, T., Forsberg, J., Potter, B. K., Dente, C. J., Kirk, A., & Elster, E. (2017). Precision diagnosis: a view of the clinical decision support systems (CDSS) landscape through the lens of critical care. *Journal of clinical monitoring and computing*, *31*(2), 261-271.

Belciug, S. (2009). Patients length of stay grouping using the hierarchical clustering algorithm. *Annals of the University of Craiova-Mathematics and Computer Science Series*, *36*(2), 79-84.

Berg, G. D., & Gurley, V. F. (2019). Development and validation of 15-month mortality prediction models: a retrospective observational comparison of machine-learning techniques in a national sample of Medicare recipients. *BMJ open*, *9*(7), e022935.

Berger, J. S., Haskell, L., Ting, W., Lurie, F., Chang, S. C., Mueller, L. A., ... & Alas, V. (2020). Evaluation of machine learning methodology for the prediction of healthcare resource utilization and healthcare costs in patients with critical limb ischemia—is preventive and personalized approach on the horizon?. *EPMA Journal*, *11*(1), 53-64.

Bergquist, S. L., Brooks, G. A., Keating, N. L., Landrum, M. B., & Rose, S. (2017, November). Classifying lung cancer severity with ensemble machine learning in health care claims data. In *Machine Learning for Healthcare Conference* (pp. 25-38). PMLR.

Boyd, K., Eng, K. H., & Page, C. D. (2013, September). Area under the precision-recall curve: point estimates and confidence intervals. In *Joint European conference on machine learning and knowledge discovery in databases* (pp. 451-466). Springer, Berlin, Heidelberg.

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

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5-32.

Cadarette, S. M., & Wong, L. (2015). An introduction to health care administrative data. *The Canadian journal of hospital pharmacy*, *68*(3), 232.

Cai, J., Luo, J., Wang, S., & Yang, S. (2018). Feature selection in machine learning: A new perspective. *Neurocomputing*, *300*, 70-79.

Cartwright, D. J. (2013). ICD-9-CM to ICD-10-CM codes: what? why? how?.

Castillo, S., Gopalacharyulu, P., Yetukuri, L., & Orešič, M. (2011). Algorithms and tools for the preprocessing of LC–MS metabolomics data. *Chemometrics and Intelligent Laboratory Systems*, *108*(1), 23-32.

CCS (Clinical Classifications Software) - Synopsis. (n.d.). Retrieved January 08, 2022, from https://www.nlm.nih.gov/research/umls/sourcereleasedocs/current/CCS/index.html

Centers for Disease Control and Prevention. (2015, November 6). ICD - ICD-10-CM - International Classification of diseases,(icd-10-CM/PCS transition. Centers for Disease Control and Prevention. Retrieved January 3, 2022, from https://www.cdc.gov/nchs/icd/icd10cm_pcs_background.htm

Charlson, M. E., Pompei, P., Ales, K. L., & MacKenzie, C. R. (1987). A new method of classifying prognostic comorbidity in longitudinal studies: development and validation. *Journal of chronic diseases*, *40*(5), 373-383.

Chen, P. H. C., Liu, Y., & Peng, L. (2019). How to develop machine learning models for healthcare. *Nature materials*, *18*(5), 410-414.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Choi, E., Bahadori, M. T., Schuetz, A., Stewart, W. F., & Sun, J. (2016, December). Doctor ai: Predicting clinical events via recurrent neural networks. In *Machine learning for healthcare conference* (pp. 301-318). PMLR.

Choi, E., Schuetz, A., Stewart, W. F., & Sun, J. (2017). Using recurrent neural network models for early detection of heart failure onset. *Journal of the American Medical Informatics Association*, *24*(2), 361-370.

Clinical classifications software (CCS) for ICD-10-PCS (beta version). (n.d.). Retrieved January 24, 2022, from https://www.hcup-us.ahrq.gov/toolssoftware/ccs10/ccs10.jsp

Clinical Classifications Software (CCS) for ICD-9-CM Fact Sheet. (n.d.). Retrieved January 8, 2022, from https://www.hcup-us.ahrq.gov/toolssoftware/ccs/ccsfactsheet.jsp

CMS Forms list. CMS. (n.d.). Retrieved January 30, 2022, from https://www.cms.gov/Medicare/CMS-Forms/CMS-Forms/CMS-Forms-List

Collin, C., Wade, D. T., Davies, S., & Horne, V. (1988). The Barthel ADL Index: a reliability study. *International disability studies*, *10*(2), 61-63.

Concept: Charlson Comorbidity Index: MCHP concept dictionary and glossary for population-based research: Max Rady College of Medicine: University of Manitoba. (n.d.). Retrieved January 03, 2022, from http://mchp-appserv.cpe.umanitoba.ca/viewConcept.php?conceptID=1098

Connelly, R., Playford, C. J., Gayle, V., & Dibben, C. (2016). The role of administrative data in the big data revolution in social science research. *Social science research*, *59*, 1-12.

CPT® Overview and Code Approval. American Medical Association. (n.d.). Retrieved January 3, 2022, from https://www.ama-assn.org/practice-management/cpt/cpt-overview-and-code-approval

Dai, D., Alvarez, P. J., & Woods, S. D. (2021). A Predictive Model for Progression of Chronic Kidney Disease to Kidney Failure Using a Large Administrative Claims Database. *ClinicoEconomics and Outcomes Research: CEOR*, *13*, 475.

Desai, R. J., Wang, S. V., Vaduganathan, M., Evers, T., & Schneeweiss, S. (2020). Comparison of machine learning methods with traditional models for use of administrative claims with electronic medical records to predict heart failure outcomes. *JAMA network open*, *3*(1), e1918962-e1918962.

Ferrao, J. C., Oliveira, M. D., Janela, F., & Martins, H. M. (2016). Preprocessing structured clinical data for predictive modeling and decision support. *Applied clinical informatics*, *7*(04), 1135-1153.

Ferver, K., Burton, B., & Jesilow, P. (2009). The use of claims data in healthcare research. *The Open Public Health Journal*, *2*(1).

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, *55*(1), 119-139.

Fried, T. R., Bradley, E. H., Towle, V. R., & Allore, H. (2002). Understanding the treatment preferences of seriously ill patients. *New England Journal of Medicine*, *346*(14), 1061-1066.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189-1232.

Gasparini, A. (2018). comorbidity: An R package for computing comorbidity scores. *Journal of Open Source Software*, *3*(23), 648.

Ghassemi, M., Naumann, T., Doshi-Velez, F., Brimmer, N., Joshi, R., Rumshisky, A., & Szolovits, P. (2014, August). Unfolding physiological state: Mortality modelling in intensive care units. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 75-84).

Gobbens, R. J., & van Assen, M. A. (2014). The prediction of ADL and IADL disability using six physical indicators of frailty: a longitudinal study in the Netherlands. *Current gerontology and geriatrics research*, *2014*.

Golden, J. A. (2017). Deep learning algorithms for detection of lymph node metastases from breast cancer: helping artificial intelligence be seen. *Jama*, *318*(22), 2184-2186.

Goutte, C., & Gaussier, E. (2005, March). A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. In *European conference on information retrieval* (pp. 345-359). Springer, Berlin, Heidelberg.

Guo, H. W., Huang, Y. S., Lin, C. H., Chien, J. C., Haraikawa, K., & Shieh, J. S. (2016, October). Heart rate variability signal features for emotion recognition by using principal component analysis and support vectors machine. In *2016 IEEE 16th International Conference on Bioinformatics and Bioengineering (BIBE)* (pp. 274-277). IEEE.

Hawes, C., Morris, J. N., Phillips, C. D., Mor, V., Fries, B. E., & Nonemaker, S. (1995).

Reliability estimates for the Minimum Data Set for nursing home resident assessment and care screening (MDS). *The Gerontologist*, *35*(2), 172-178.

He, D., Mathews, S. C., Kalloo, A. N., & Hutfless, S. (2014). Mining high-dimensional administrative claims data to predict early hospital readmissions. *Journal of the American Medical Informatics Association*, *21*(2), 272-279.

Henly, S. J., Wyman, J. F., & Findorff, M. J. (2011). Health and illness over time: The trajectory perspective in nursing science. *Nursing research*, *60*(3 Suppl), S5.

Henry, K. E., Hager, D. N., Pronovost, P. J., & Saria, S. (2015). A targeted real-time early warning score (TREWScore) for septic shock. *Science translational medicine*, *7*(299), 299ra122-299ra122.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

Hu, Z., Hao, S., Jin, B., Shin, A. Y., Zhu, C., Huang, M., ... & Ling, X. (2015). Online prediction of health care utilization in the next six months based on electronic health record information: a cohort and validation study. *Journal of medical Internet research*, *17*(9), e4976.

Hyer, J. M., Paredes, A. Z., White, S., Ejaz, A., & Pawlik, T. M. (2020). Assessment of utilization efficiency using machine learning techniques: A study of heterogeneity in preoperative healthcare utilization among super-utilizers. *The American Journal of Surgery*, *220*(3), 714-720.

Hyer, J. M., Tsilimigras, D. I., Gani, F., Sahara, K., Ejaz, A., White, S., & Pawlik, T. M. (2020). Factors associated with switching between low and super utilization in the surgical population: A study in medicare expenditure. *The American Journal of Surgery*, *219*(1), 1-7.

Ilyas, H., Ali, S., Ponum, M., Hasan, O., Mahmood, M. T., Iftikhar, M., & Malik, M. H. (2021). Chronic kidney disease diagnosis using decision tree algorithms. *BMC nephrology*, *22*(1), 1-11.

Jakobsen, J. C., Gluud, C., Wetterslev, J., & Winkel, P. (2017). When and how should multiple imputation be used for handling missing data in randomised clinical trials–a practical guide with flowcharts. *BMC medical research methodology*, *17*(1), 1-10.

Jeffery, A. D., Dietrich, M. S., & Maxwell, C. A. (2018). Predicting 1-year disability and mortality of injured older adults. *Archives of gerontology and geriatrics*, *75*, 191-196.

Jhajharia, S., Varshney, H. K., Verma, S., & Kumar, R. (2016, September). A neural network based breast cancer prognosis model with PCA processed features. In *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (pp. 1896-1901). IEEE.

Jiang, F., Jiang, Y., Zhi, H., Dong, Y., Li, H., Ma, S., ... & Wang, Y. (2017). Artificial intelligence in healthcare: past, present and future. *Stroke and vascular neurology*, *2*(4).

Jović, A., Brkić, K., & Bogunović, N. (2015, May). A review of feature selection methods with applications. In *2015 38th international convention on information and communication technology, electronics and microelectronics (MIPRO)* (pp. 1200-1205). Ieee.

Kim, S. Y., Kim, S., Cho, J., Kim, Y. S., Sol, I. S., Sung, Y., ... & Sohn, M. H. (2019). A deep learning model for real-time mortality prediction in critically ill children. *Critical care*, *23*(1), 1-10.

Kim, Y. J., & Park, H. (2019). Improving prediction of high-cost health care users with medical check-up data. *Big data*, *7*(3), 163-175.

Kleinbaum, D. G., & Klein, M. (2010). Introduction to logistic regression. In *Logistic regression* (pp. 1-39). Springer, New York, NY.

König, S., Pellissier, V., Hohenstein, S., Bernal, A., Ueberham, L., Meier-Hellmann, A., ... & Bollmann, A. (2021). Machine learning algorithms for claims data-based prediction of in-hospital mortality in patients with heart failure. *ESC Heart Failure*.

Kotsiantis, S. (2011). Feature selection for machine learning classification problems: a recent overview. *Artificial Intelligence Review*, *42*(1), 157-176.

Kotsiantis, S. B., Kanellopoulos, D., & Pintelas, P. E. (2006). Data preprocessing for supervised leaning. *International journal of computer science*, *1*(2), 111-117.

Krishnamurthy, S., Ks, K., Dovgan, E., Luštrek, M., Gradišek Piletič, B., Srinivasan, K., ... & Syed-Abdul, S. (2021, May). Machine learning prediction models for chronic kidney disease using national health insurance claim data in Taiwan. In *Healthcare* (Vol. 9, No. 5, p. 546). Multidisciplinary Digital Publishing Institute.

Lee, E. A., & Sangiovanni-Vincentelli, A. (1998). A framework for comparing models of computation. *IEEE Transactions on computer-aided design of integrated circuits and systems*, *17*(12), 1217-1229.

Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., & Liu, H. (2017). Feature selection: A data perspective. *ACM Computing Surveys (CSUR)*, *50*(6), 1-45.

Liao, M., Li, Y., Kianifard, F., Obi, E., & Arcona, S. (2016). Cluster analysis and its application to healthcare claims data: a study of end-stage renal disease patients who initiated hemodialysis. *BMC nephrology*, *17*(1), 1-14.

Lin, H., Long, E., Ding, X., Diao, H., Chen, Z., Liu, R., ... & Liu, Y. (2018). Prediction of myopia development among Chinese school-aged children using refraction data from electronic medical records: A retrospective, multicentre machine learning study. *PLoS medicine*, *15*(11), e1002674.

List of registries. (2022, January 04). Retrieved January 06, 2022, from https://www.nih.gov/health-information/nih-clinical-research-trials-you/list-registries

Liu, L., Shen, J., Zhang, M., Wang, Z., & Tang, J. (2018, April). Learning the joint representation of heterogeneous temporal events for clinical endpoint prediction. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 32, No. 1).

Luo, W., Phung, D., Tran, T., Gupta, S., Rana, S., Karmakar, C., ... & Berk, M. (2016). Guidelines for developing and reporting machine learning predictive models in biomedical research: a multidisciplinary view. *Journal of medical Internet research*, *18*(12), e323.

Lustgarten, J. L., Gopalakrishnan, V., Grover, H., & Visweswaran, S. (2008). Improving classification performance with discretization on biomedical datasets. In *AMIA annual symposium proceedings* (Vol. 2008, p. 445). American Medical Informatics Association.

Lynam, A. L., Dennis, J. M., Owen, K. R., Oram, R. A., Jones, A. G., Shields, B. M., & Ferrat, L. A. (2020). Logistic regression has similar performance to optimised machine learning algorithms in a clinical setting: application to the discrimination between type 1 and type 2 diabetes in young adults. *Diagnostic and prognostic research*, *4*, 1-10.

Makar, M., Ghassemi, M., Cutler, D. M., & Obermeyer, Z. (2015). Short-term mortality prediction for elderly patients using Medicare claims data. *International journal of machine learning and computing*, *5*(3), 192.

Malley, B., Ramazzotti, D., & Wu, J. T. (2016). Data Pre–processing; Secondary Analysis of Electronic Health Records.

Maslove, D. M., Podchiyska, T., & Lowe, H. J. (2013). Discretization of continuous features in clinical datasets. *Journal of the American Medical Informatics Association*, *20*(3), 544-553.

McHugh, M. L. (2012). Interrater reliability: the kappa statistic. *Biochemia medica*, *22*(3), 276-282.

MDS 3.0 Technical Information. CMS. (n.d.). Retrieved January 30, 2022, from https://www.cms.gov/Medicare/Quality-Initiatives-Patient-Assessment-Instruments/NursingHomeQualityInits/NHQIMDS30TechnicalInformation

Michalski, R. S., & Wojtusiak, J. (2012). Reasoning with unknown, not-applicable and irrelevant meta-values in concept learning and pattern discovery. *Journal of Intelligent Information Systems*, *39*(1), 141-166.

Min, H., Mobahi, H., Irvin, K., Avramovic, S., & Wojtusiak, J. (2017). Predicting activities of daily living for cancer patients using an ontology-guided machine learning methodology. *Journal of biomedical semantics*, *8*(1), 1-8.

Miotto, R., Li, L., Kidd, B. A., & Dudley, J. T. (2016). Deep patient: an unsupervised representation to predict the future of patients from the electronic health records. *Scientific reports*, *6*(1), 1-10.

Mitchell, T. (1997). Machine learning.

Murray, S. A., Kendall, M., Boyd, K., & Sheikh, A. (2005). Illness trajectories and palliative care. *Bmj*, *330*(7498), 1007-1011.

Ngiam, K. Y., & Khor, W. (2019). Big data and machine learning algorithms for health-care delivery. *The Lancet Oncology*, *20*(5), e262-e273.

NIH's definition of a clinical trial. (n.d.). Retrieved January 16, 2022, from https://grants.nih.gov/policy/clinical-trials/definition.htm

Nithya, B., & Ilango, V. (2017, June). Predictive analytics in health care using machine learning tools and techniques. In *2017 International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 492-499). IEEE.

Olson, M. A., & Wyner, A. J. (2018). Making sense of random forest probabilities: a kernel perspective. *arXiv preprint arXiv:1812.05792*.

Peiffer-Smadja, N., Rawson, T. M., Ahmad, R., Buchard, A., Georgiou, P., Lescure, F. X., ... & Holmes, A. H. (2020). Machine learning for clinical decision support in infectious diseases: a narrative review of current applications. *Clinical Microbiology and Infection*, *26*(5), 584-595.

Quan, H., Sundararajan, V., Halfon, P., Fong, A., Burnand, B., Luthi, J. C., ... & Ghali, W. A. (2005). Coding algorithms for defining comorbidities in ICD-9-CM and ICD-10 administrative data. *Medical care*, 1130-1139.

Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, *1*(1), 81-106.

Rajkomar, A., Oren, E., Chen, K., Dai, A. M., Hajaj, N., Hardt, M., ... & Dean, J. (2018). Scalable and accurate deep learning with electronic health records. NPJ Digital Medicine, 1(1), 1-10.

Ren, Y., Fei, H., Liang, X., Ji, D., & Cheng, M. (2019). A hybrid neural network model for predicting kidney disease in hypertension patients based on electronic health records. *BMC medical informatics and decision making*, *19*(2), 131-138.

Renard, F., Guedria, S., De Palma, N., & Vuillerme, N. (2020). Variability and reproducibility in deep learning for medical image segmentation. *Scientific Reports*, *10*(1), 1-16.

Richardson, M. (2009). Principal component analysis. *URL: http://people. maths. ox. ac. uk/richardsonm/SignalProcPCA. pdf (last access: 3.5. 2013). Aleš Hladnik Dr., Ass. Prof., Chair of Information and Graphic Arts Technology, Faculty of Natural Sciences and Engineering, University of Ljubljana, Slovenia ales. hladnik@ ntf. uni-lj. si*, *6*, 16.

Roysden, N., & Wright, A. (2015). Predicting health care utilization after behavioral health referral using natural language processing and machine learning. In *AMIA Annual Symposium Proceedings* (Vol. 2015, p. 2063). American Medical Informatics Association.

Samuel, A. L. (1959). Eight-move opening utilizing generalization learning,(See Appendix B, Game G-43.1 Some Studies in Machine Learning Using the Game of Checkers). *IBM Journal*, 210-229.

Segal, Z., Kalifa, D., Radinsky, K., Ehrenberg, B., Elad, G., Maor, G., ... & Koren, G. (2020). Machine learning algorithm for early detection of end-stage renal disease. *BMC nephrology*, *21*(1), 1-10.

Shah, N. D., Steyerberg, E. W., & Kent, D. M. (2018). Big data and predictive analytics: recalibrating expectations. *Jama*, *320*(1), 27-28.

Shah, S., Vanclay, F., & Cooper, B. (1989). Improving the sensitivity of the Barthel Index for stroke rehabilitation. *Journal of clinical epidemiology*, *42*(8), 703-709.

Shickel, B., Tighe, P. J., Bihorac, A., & Rashidi, P. (2017). Deep EHR: a survey of recent advances in deep learning techniques for electronic health record (EHR) analysis. *IEEE journal of biomedical and health informatics*, *22*(5), 1589-1604.

Sidey-Gibbons, J. A., & Sidey-Gibbons, C. J. (2019). Machine learning in medicine: a practical introduction. *BMC medical research methodology*, *19*(1), 1-18.

Singh, A., Nadkarni, G., Gottesman, O., Ellis, S. B., Bottinger, E. P., & Guttag, J. V. (2015). Incorporating temporal EHR data in predictive models for risk stratification of renal function deterioration. *Journal of biomedical informatics*, *53*, 220-228.

Sondhi, P. (2009). Feature construction methods: a survey. *sifaka. cs. uiuc. edu*, *69*, 70-71.

Spurgeon, A., Hiser, B., Hafley, C., & Litofsky, N. S. (2011). Does improving medical record documentation better reflect severity of illness in neurosurgical patients?. Neurosurgery, 58(CN_suppl_1), 155-163.

Stein, J. D., Lum, F., Lee, P. P., Rich III, W. L., & Coleman, A. L. (2014). Use of health care claims data to study patients with ophthalmologic conditions. *Ophthalmology*, *121*(5), 1134-1141.

Surveys. (n.d.). Retrieved January 14, 2022, from https://www.nlm.nih.gov/nichsr/stats_tutorial/section3/mod1_surveys.html

Torrey, T. (2020, February 26). *How medical codes are used in the healthcare field*. Verywell Health. Retrieved January 3, 2022, from https://www.verywellhealth.com/a-patients-guide-to-medical-codes-2615316

Tran, T., Luo, W., Phung, D., Gupta, S., Rana, S., Kennedy, R. L., ... & Venkatesh, S. (2014). A framework for feature extraction from hospital medical data with applications in risk prediction. *BMC bioinformatics*, *15*(1), 1-9.

U.S. Department of Health and Human Services. (n.d.). *What are clinical trials and studies?* National Institute on Aging. Retrieved January 16, 2022, from https://www.nia.nih.gov/health/what-are-clinical-trials-and-studies

Uddin, S., Khan, A., Hossain, M. E., & Moni, M. A. (2019). Comparing different supervised machine learning algorithms for disease prediction. *BMC medical informatics and decision making*, *19*(1), 1-16.

Wasylewicz, A. T. M., & Scheepers-Hoeks, A. M. J. W. (2019). Clinical decision support systems. *Fundamentals of clinical data science*, 153-169.

Wells, B. J., Chagin, K. M., Nowacki, A. S., & Kattan, M. W. (2013). Strategies for handling missing data in electronic health record derived data. *Egems*, *1*(3).

Wiens, J., & Shenoy, E. S. (2018). Machine learning for healthcare: on the verge of a major shift in healthcare epidemiology. *Clinical Infectious Diseases*, *66*(1), 149-153.

Wilson, J., & Bock, A. (2012). The benefit of using both claims data and electronic medical record data in health care analysis. *Optum Insight*, *1*, 1-4.

Wojtusiak, J. (2008, June). Data-driven constructive induction in the learnable evolution model. In *Proceedings of the 16th International Conference Intelligent Information Systems, Zakopane, Poland*.

Wojtusiak, J. (2021). Reproducibility, Transparency and Evaluation of Machine Learning in Health Applications. In *HEALTHINF* (pp. 685-692).

Wojtusiak, J., & Asadzadehzanjani, N. (2022). A Discussion on Comparing Machine Learning Models for Health Outcome Prediction, In *HEALTHINF*

Wojtusiak, J., Asadzadehzanjani, N., Levy, C., Alemi, F., & Williams, A. E. (2021). Computational Barthel Index: an automated tool for assessing and predicting activities of daily living among nursing home patients. *BMC medical informatics and decision making*, *21*(1), 1-15.

Wojtusiak, J., Asadzaehzanjani, N., Levy, C., Alemi, F., & Williams, A. E. (2021). Online Decision Support Tool that Explains Temporal Prediction of Activities of Daily Living (ADL). In HEALTHINF (pp. 629-636).

Wojtusiak, J., Elashkar, E., & Nia, R. M. (2017, February). C-Lace: Computational Model to Predict 30-Day Post-Hospitalization Mortality. In *HEALTHINF* (pp. 169-177).

Wojtusiak, J., Elashkar, E., & Nia, R. M. (2018). C-LACE2: computational risk assessment tool for 30-day post hospital discharge mortality. *Health and Technology*, *8*(5), 341-351.

Wojtusiak, J., Levy, C. R., Williams, A. E., & Alemi, F. (2016). Predicting functional decline and recovery for residents in veterans affairs nursing homes. *The Gerontologist*, *56*(1), 42-51.

Xie, Y., Schreier, G., Hoy, M., Liu, Y., Neubauer, S., Chang, D. C., ... & Lovell, N. H. (2016). Analyzing health insurance claims on different timescales to predict days in hospital. *Journal of biomedical informatics*, *60*, 187-196.

Yang, L., & Xu, Z. (2019). Feature extraction by PCA and diagnosis of breast tumors using SVM with DE-based parameter tuning. *International Journal of Machine Learning and Cybernetics*, *10*(3), 591-601.

BIOGRAPHY

Negin Asadzadehzanjani was born and grew up in Tehran, Iran. She received her bachelor's degree in materials engineering from Imam Khomeini International University in 2013 and graduated with a Master of Science degree in Biomedical Engineering from University of Tehran in 2016. She then joined the department of Health Administration and Policy at George Mason University and received her Ph.D. in Health Services Research with concentration in Knowledge Discovery and Health Informatics.

Negin was employed as graduate research assistant for 5 years during her PhD and has been a member of George Mason University Machine Learning and Inference (MLI) Laboratory. Her research interest focuses on both methodological work and application of artificial intelligence and machine learning in healthcare. In her Ph.D., Negin investigated different representation methods of administrative codes available in claims data for use in machine learning algorithms. She was involved in various projects including functional status prediction from medical claims, constructing online calculator to predict functional status, opioid abuse trajectory prediction, association of infection and Alzheimer's disease, and methods of machine learning model comparison for health outcome prediction. She had co-authored and authored a number of research papers in the field of machine learning in healthcare.