1-1-2018

Data-Driven Modeling For Decision Support Systems And Treatment Management In Personalized Healthcare

Milad Zafar Nezhad
Wayne State University,

DATA-DRIVEN MODELING FOR DECISION SUPPORT SYSTEMS AND TREATMENT MANAGEMENT IN PERSONALIZED HEALTHCARE

by

MILAD ZAFAR NEZHAD

DISSERTATION

Submitted to the Graduate School

of Wayne State University,

Detroit, Michigan

in partial fulfillment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

2018

MAJOR: INDUSTRIAL ENGINEERING

Approved By:

_____

Advisor Date

_____

_____

_____

_____

DEDICATION

To my beloved wife and my great family for their endless love, encouragement and support.

ACKNOWLEDGEMENTS

I would first like to express my gratitude to my advisor Prof. Kai Yang for his continuous support, motivation and academic advises during my research. I am also very thankful to Prof. Dongxiao Zhu for his scientific guidances in each step of this dissertation. Besides them, I would like to thank the rest of committee members: Prof. Joseph Kim and Prof. Qingyu Yang for their helpful comments and supports.

I am grateful to Dr. Jennifer Beebe-Dimmer and Julie Ruterbusch from Barbara Ann Karmanos Cancer Institute in Detroit, Michigan who helped us to access to the SEERMedicare datasets. My thanks also go to Dr. Philip Levi from department of emergency medicine and cardiovascular research institute, Medical School, Wayne State University for helping us to access cardiovascular datasets in Detroit Medical Center.

I am thankful my fellow officemates in Healthcare System Engineering group at Wayne state University: Hossein Badri, Mohammad Abdollahi, Hessam Olya and Ali Asadi for useful comments, advices and fun time.

Finally, I must express my gratitude to Nasim (my wife) and my family for their love, encouragement and endless support. Without their help this accomplishment could have never been possible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1 INTRODUCTION

Healthcare is transforming from a disease-centered model to a patient-centered model [106], in a disease-centered model, medical decisions is made based on the clinical knowledge and expertise, data from medical tests and different evidences. In a patient-centered model, patients actively are considered in their own care plan and treated focused on individual needs and preferences. In another word, patient-center model revolves around the patients rather than physicians and providers [22]. The explosive increase of Electronic Medical Records (EMR) provides many opportunities to carry out data science research by applying data mining and machine learning tools and techniques. EMR contains massive and a wide range of information of patients concerning different aspects of healthcare, such as patient conditions, diagnostic tests, lab results, imaging exams, genomics, proteomics, treatments and financial records [57]. Particularly, the extensive and powerful patient-centered data enables data scientists and medical researchers to conduct their research in the field of personalized (precision) medicine or healthcare. There are several definitions about personalized medicine (healthcare) in the literature. It has been defined as: 1) "A medical model that proposes the customization of healthcare, with decisions and practices being tailored to the individual patient by use of genetic or other information.", [108]; 2) "The tailoring of medical treatment to the specific characteristics of each patient. It does not literally mean the creation of drugs or medical devices that are unique to a patient. Rather, it involves the ability to classify individuals into subpopulations that are uniquely or disproportionately susceptible to a particular disease or responsive to a specific treatment", [97]; and 3) "The use of combined knowledge (genetic or otherwise) about a person to predict disease susceptibility, disease prognosis, or treatment response and thereby improve that person's health ", [97].

In general, the goal of precision medicine or personalized healthcare is to provide the right treatment to the right patient at the right time. Personalized medicine is a multi-disciplinary area that combines data science tools and statistics techniques with medical knowledge to develop tailor-made treatment, prevention and intervention plans for individual patients [97]. By emerge of huge amount of biomedical data, data-driven and networks-driven thinking and methods can play a significant role in proceeding of personalized healthcare [22]. In recent years, many researchers from different area focus on specific disease such as hypertension, diabetes and several cancer types to discover individual preventable disease risk factors, precision diagnosis and personalized treatment policy [121]. Therefore, personalized medicine (healthcare) needs a computing and integrated framework to aggregate and analyze big datasets, realize deep knowledge about patient network and their similarities, and prepare personalized disease risk profiles for each individual patient [22]. With this end, much recent research efforts have been provided to applying machine learning, data analytics and business intelligent methodologies which can be used to derive real-world medical and medicine data for designing personalized decision support systems in healthcare delivery and treatment management [57].

In sum, integrating medical and medicine knowledge by applying data analytics tools and methods on huge electronic medical records (EMR) leads to achieve smart clinical decision support systems which can assist physicians in providing precision and personalized clinical recommendations [12].

1.1 Research Framework

In recent years, several studies have been conducted in personalized medicine to answer the questions such as: How to deeply use big data from EMRs, patients' medical history and personal

information to select and predict the specific disease risk factors for individual patient?, How to detect a subgroup of patient that are more similar with each other and then assign special treatment policy?, How to produce personalized drugs based on different patients' characteristics? And how to develop an intelligent system to optimize targeted therapy? [22]. Based on appropriate responses to these questions in the literature; we can categorize the research works accomplished in the field of data-driven personalized medicine (healthcare) as three main groups [57]:

1. **Predictive modeling and risk factor identification for high dimensional data:** Building accurate prediction models for different healthcarre purposes and extracting the most important features (Risk factors) is a key challenge in developing risk prediction models from high dimensional (thousands to tens of thousands features) observational healthcare data. There are several large-scale algorithms have been developed to come up this challenge in the field of precision medicine.

2. **Patient similarity analytics:** Discovering similar subgroup of patients by applying data analytics methods according to their disease condition risks; is an important component in personalized decision support systems and effective care management because for patients with similar risks and behavior, we may assign similar treatment plans.

3. **Mining care pathways and Personalized treatment optimization:** Clinical pathways, a sequence of medical treatment, traditionally devised by a physician after patient diagnosis based on physician's education, experience, and intuition. Recently, data analysis of rich longitudinal data obtained from EMRs empowers clinicians with a data-driven precision care pathway and based on patient' similarities, the doctors can understand the pattern of treatment and make an optimized decision.

In Figure1 a data-driven personalized healthcare platform obtained from IBM research demon-

strates how the whole recommendation system works.

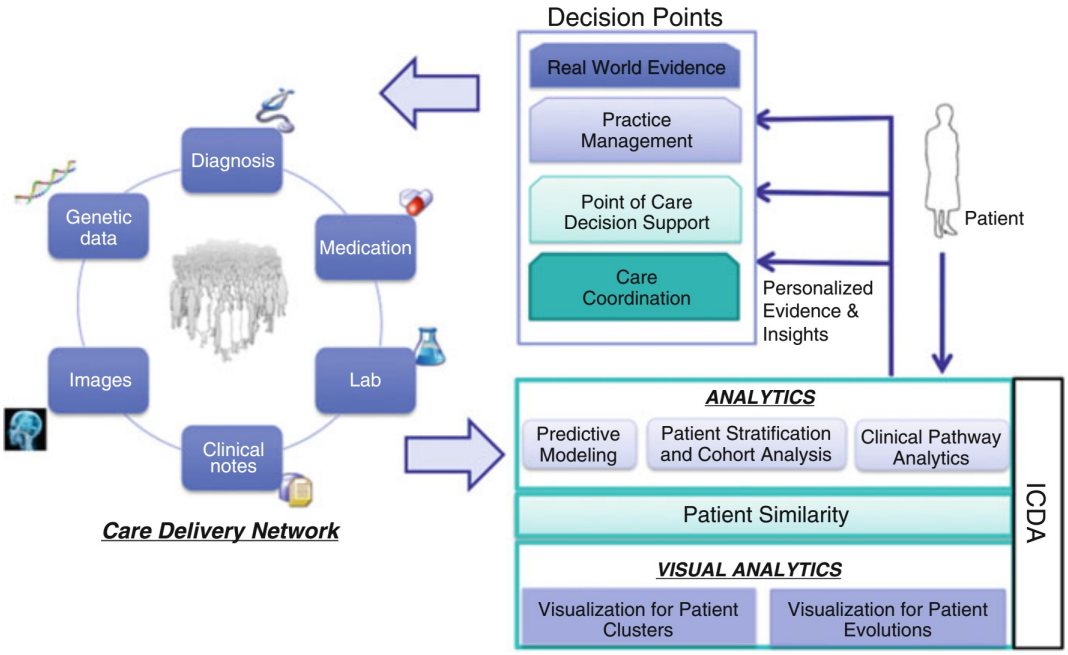


Figure 1: Intelligent Care Delivery Analytics (ICDA)–the data driven personalized healthcare analytics platform at IBM research [57]

In this research we focus on these three categories as our research framework to develop some novel personalized data-driven algorithms with competitive performance, then we apply our proposed methods on some specific diseases such as hypertension, cardiovascular disease and cancers.

1.2 Motivations and Objectives

The goal of this research is to develop data-driven algorithms and methodologies for carrying out precision medicine and personalized healthcare in some specific disease. With this motivation we design our research framework as illustrated in Figure 2 to address three main objectives in this study as following:

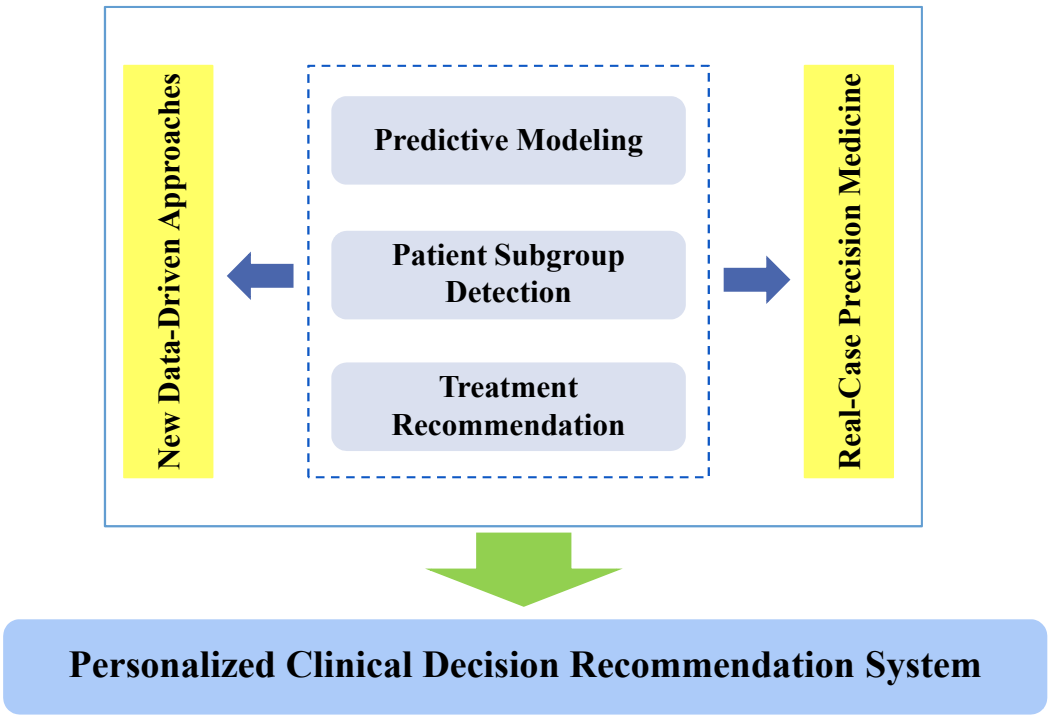


Figure 2: Our Research Framework

1. **Predictive Modeling:** Since in personalized healthcare applications, the clinical datasets are usually high-dimensional, sparse, complex and noisy, learning an accurate model for predictive analytics and patient risk monitoring is hard and challenging. To overcome this challenge, the first goal of our study is to provide a predictive model which can handle complex medical data and provide precise prediction in different clinical applications such as disease risk forecasting, drug response discovering and health condition monitoring. Our method outperforms rather than the well-known state of arts machine learning methods in the literature.

2. **Patient Subgroup Detection:** Discovering subgroup of patient who are similar in terms of specific characteristic is highly useful in many clinical purposes such as finding the pattern of treatment, evaluating treatment effects and discovering important risk factors. There are several methods developed in the literature such as tree-based methods or clustering methods to detect the

subgroup of patients from high-dimensional data. The other goal of this study is to propose a novel patient subgroup detection method which considers similarities among risk factors and response variable simultaneously.

3. Treatment Recommendation: There exist several approaches for treatment recommendation in the healthcare domain such as clustering based models or collaborative filtering. Most of these approach are not appropriate for high dimensional clinical data specially in the case that labeled data is not enough. The goal of this step is developing a treatment recommendation approach using an accurate novel survival analysis model from high-dimensional and personalized data.

The framework explained above can be considered as an integrated approach to make a decision recommendation system for patients and healthcare providers. In other words, this data driven approach can suggest optimal treatment policy for patients individually based on their personalized data and can recommend important risk factors for individual patient and treatment policy for healthcare providers.

CHAPTER 2 PREDICTIVE APPROACH USING DEEP FEATURE LEARNING FOR PERSONALIZED HEALTHCARE

In this chapter, we propose a new predictive approach based on feature representation using deep feature learning and word embedding techniques. Our method uses different deep architectures (Stacked autoencoders, Deep belief network and Variational autoencoders) for feature representation in higher-level abstraction to obtain effective and more robust features from EMRs, and then build prediction models on the top of them. Our approach is particularly useful when the unlabeled data is abundant whereas labeled one is scarce. We investigate the performance of representation learning through a supervised approach. First, we perform our method on a small dataset related to a precision medicine application, which concentrates on prediction of cardiovascular risk level measured by left ventricular mass indexed to body surface area termed LVMI among African-Americans. Then we use two large datasets from eICU collaborative research database to predict the length of stay in Cardiac-ICU and Neuro-ICU based on high dimensional features. Finally we provide a comparative study and show that our predictive approach leads to better results in comparison with others.

2.1 Problem Definition

Recently, data-driven modeling and optimization has been applied in different domains such as manufacturing [99], healthcare [87], quality assessment [43] and chemical processes sustainability [84]. In healthcare domain, the explosive growing of Electronic Medical Records (EMRs) creates huge opportunity to accomplish data science research by applying machine learning and data analytics tools and techniques [86]. EMRs includes vast and wide range of information on patients related to several aspects of healthcare, such as patient information, health conditions, lab results, diagnostic tests, imaging data, genomics, proteomics, treatments and medication records [57, 87].

Particularly, the massive and powerful patient-centered data encourages medical researchers and data scientists to carry out their research in the field of personalized/precision medicine. Personalized/precision medicine is a multi-domain research area which use data science methods and medical knowledge to recommend the right treatment to the right patient at the right time [97].

Since EMRs are complex, sparse, heterogeneous and time-dependent; using EMRs for personalized medicine is challenging and complicated to interpret. Representation learning or feature learning provides the opportunity to overcome this problem by transforming medical features to a higher level abstraction, which can provide more robust features. On the other side, labeling of clinical data is expensive, difficult and time-consuming in several cases such as special disease where unlabeled data (features) may be abundant. Representation learning through unsupervised approach is a very beneficial way to extract strong feature learning from both labeled and unlabeled data and improve training models performance made based on labeled data.

Representation learning [8] includes a set of techniques that learn a feature via transformation of input data to a representation that can improve machine learning tasks such as classification and regression. In the other words, representation learning helps to provide more useful information. Despite the success of feature learning in several domains such as text mining, multimedia, and marketing, these techniques have not been applied widely for Electronic Health Records (EHRs) [81]. In this way, many research have been developed in recent years and those are growing up very fast specially in the field of precision medicine and health informatics. The main challenges exist in processing of EHRs listed as following [26]: 1) High-Dimensionality, 2) Temporality which refers to the sequentiality of clinical events, 3) Sparsity, 4) Irregularity which means the high variabilities exist in the EHRs and 5) Bias including systematic errors in the medical data.

Representation learning can overcome those challenges and the choice of data representation

or feature representation plays a significant role in success of machine learning algorithms [8]. For this reason, many efforts in developing machine learning algorithms focus on designing preprocessing mechanisms and data transformations for representation learning that would enable more efficient machine learning algorithms [8]. There are several approaches for feature learning such as K-means clustering, Principal component analysis (PCA), Local linear embedding, Independent component analysis (ICA) and Deep learning.

Deep learning methods with multiple layers of transformation are representation learning algorithms, composing by simple but nonlinear transformations which represent the raw data at higher level abstraction [51]. Deep learning models demonstrated promising performance and potential in computer vision, speech recognition and natural language processing tasks. The rising popularity of using deep learning in healthcare informatics is remarkable for different purposes. For instance deep learning was recently employed to medicine and genomics to rebuilding brain circuits, performance prediction of drug molecules, identifying the effects of mutations on gene expressions, personalized prescriptions, treatment recommendations, and clinical trial recruitment [82]. Applying deep learning through unsupervised way on EHRs addressed in many recent research works for feature representation in order to achieve specific or general goals [104]. For instance "Deep patient" [82] and "Doctor AI" [28] approaches are good examples of these recent works which used unsupervised learning via deep learning before supervised learning.

In this study we focus on two specific healthcare informatics problems using high dimensional electronic medical records. The first one is related to a African-Americans cohort at high risk of heart failure. In this case study, we use left ventricular mass indexed to body surface area (LVMI) as a measure of heart failure risk. The capability to precisely predict LVMI could improve the treatment and reduce the cost of LVMI measurement for patients and hospitals. In the second

case study, we use eICU collaborative research database with several personalized factors to predict patient length of stay (LOS) in ICU for two different patient types. The more accurate LOS prediction can lead to better scheduling in hospital which reduce the cost and increase the patient satisfaction.

According to individual medical data with several features such as demographic information, patient clinical history, individual health condition, laboratory test results, diagnosis and treatment data, we first use feature representation by applying deep learning to transforms current features to higher level abstraction and then, we implement machine learning methods to predict our target of interest (LVMI and LOS) through a supervised approach. This prediction framework can be applied as a decision support system to assist physicians and health systems managers.

Figure 3 demonstrates our integrated approach in three consecutive steps; first we start by preprocessing raw data to overcome some popular issues such as missing values, outliers and data quality, in the second step we apply unsupervised deep learning for producing higher-level abstraction of input data and in the final step, supervised learning method is implemented for forecasting the target value and model evaluation. Based on the model evaluation results, steps B and C are applied iteratively to finalize and select the best deep architecture for feature learning.

Representation by deep learning is different from traditional feature learning techniques. In fact, deep learning with multiple hidden layers provides meaningful and higher level abstractions of the input data [82]. A completely unsupervised representation from raw data can be applied to other unsupervised or supervised tasks such as patient subgroup analysis, treatment clustering and disease risk prediction. Therefore we can infer our approach as a semi-supervised learning framework where we apply the benefits obtained from unsupervised tasks to the different tasks as well as risk prediction.

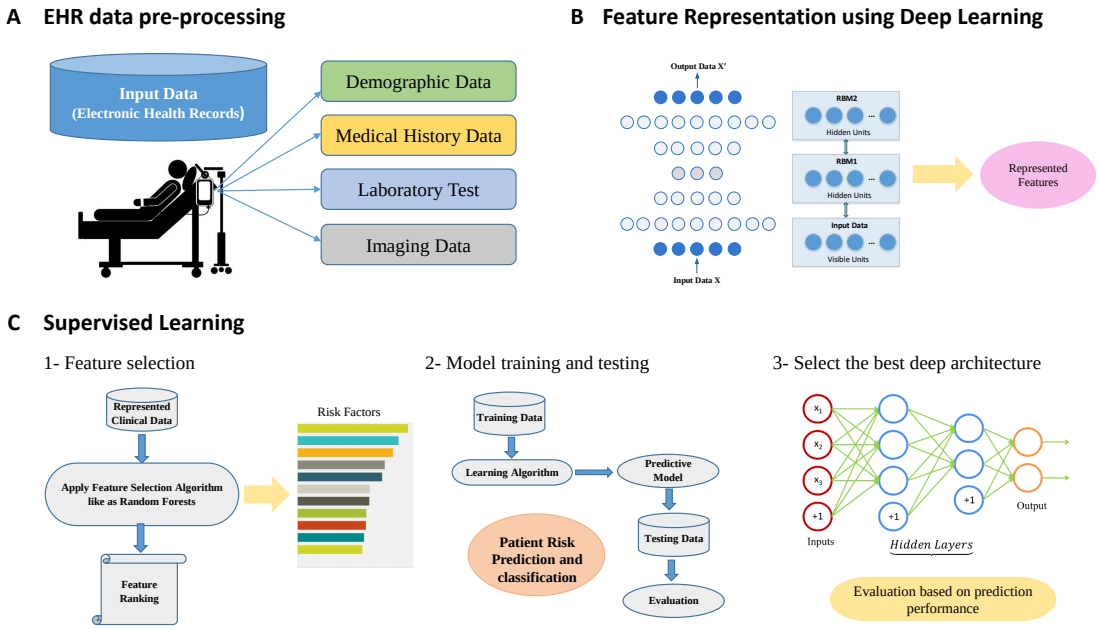


Figure 3: An illustration of the three consecutive steps for our approach

We use unsupervised learning before supervised learning because the success of predictive machine learning algorithms highly depends on feature representation and extraction [81]. Since in several situation, data is sparse, noisy, high dimensional and repetitive, supervised learning and feature selection approaches cannot identify the pattern of data which makes them inappropriate for modeling the hierarchical and complex data. To overcome this shortcoming, unsupervised feature learning or representation learning attempts automatically to discover complexity and dependencies in the data to learn a compact and high-level representation which provides better features to extract useful information when applying classifiers and predictive models.

In this chapter, we develop a new predictive approach using deep learning and data representation for EMRs. In our method, we apply three deep architectures for feature representation in higher levels abstraction: Stacked autoencoders, Deep belief network and Variational autoencoders. Our contributions in this chapter lie into three folds: 1) To our knowledge, it is one of the first methods that uses Variational Autoencoders (VAE) for feature representation on EHRs where

the advantage of VAE over traditional autoencoders is learning the true distribution of the training data as opposed to just remembering the particular training dataset, hence it can improve the representation performance significantly, 2) It is the first work that provides a comparative study to investigate the choice of deep representation among small and large datasets, and 3) Our proposed framework is highly useful for exploiting a large amount of unlabeled medical records for extracting high level representation of labeled data for supervised learning tasks.

2.2 Literature Review

Deep learning, including predictive modeling and feature representation, has been developed and applied in a different areas, such as natural language processing, computer vision, remote sensing, and healthcare informatics. The main causes for this wide range applications are improving the prediction performance, ability to model of complex informations and providing high-level features representation [71].

Deep learning with multiple hidden layers provides meaningful and higher level abstractions of the input data [82]. Among several applications of deep learning in different domains, we focus on the healthcare and bioinformatics applications. In this domain, deep learning have been applied in different areas using EHRs, clinical imaging and genomics data [80].

In terms of research purpose and different applications, we categorize the current related works in three following categories: 1) Research works applied deep learning to predict and classify disease risk levels. For instances, Cheng et al. [26] represented the EHRs for every patient as a temporal matrix with two dimension (i.e., time and event). The authors applied a four-layer Convolutional Neural Network (CNN) to predict congestive heart failure and chronic disease and demonstrated that method outperforms over the baseline. In the other study, Choi et al. [28]

developed a predictive approach called Doctor AI for clinical events using Recurrent Neural Network (RNN) and applied to longitudinal large EMR data to predict the diagnosis and treatment categories for the following visit. Miotto et al. [81], used stack of denoising autoencoders for unsupervised feature representation of EHRs of about 700,000 patients for different diseases such as severe diabetes, schizophrenia, and various cancers. Their approach improved clinical prediction which could provide a machine learning framework for medical decision systems. 2) Studies used deep learning for feature representation in purpose of feature selection and discovering disease phenotypes. Li et al. [71] proposed a deep feature selection method using regularized regression idea for selecting important input features in a deep network. They added an one-to-one linear layer right after the visible layer and connected it to the first hidden layer of a deep network with an elastic-net regularization. After training of deep network, the significant features are selected based on their weights in the input layer. Finally, the authors performed their model in a clinical problem using genomics data. In the other research work [85], a new feature selection approach is developed using a five-layers stacked autoencoders deep network. Authors applied their method on a precision medicine application to discover risk factors among African-Americans at the high risk of heart failure. 3) Research works applied deep learning for clinical image processing with the goal of disease diagnosis and image segmentation. It is appropriate to mention that the first application of deep learning to medical data is on clinacal image processing, especially on the analysis of brain Magnetic Resonance Imaging (MRI) scans [82].

Cheng et al. [24] used deep learning for computer-assisted diagnosis for the diagnosis classification of benign and malignant nodules. They applied stacked denoising autoencoder on the two applications for the classification of lung CT nodules and breast ultrasound lesions using clinical images. In another research, Gulshan et al. [46] used CNN to identify diabetic retinopathy and di-

Table 1: Summary of research works developed and applied deep learning approach in healthcare domain

Category	Example	Model	Ref
Prediction and classification of disease risk level	Predict unplanned readmission after discharge using EHR	CNN	[88]
	Multi-task prediction of disease onset from lab test results	RNN, CNN	[96]
	Predict future clinical events using EHR	SDA	[81]
	Predict chromatin marks from genomics data	CNN	[124]
	Prediction of protein backbones using genomics data	SAE	[78]
	Classification of cancer from gene expression profiles	SAE	[40]
Discovering of important disease risk factors and phenotype	Risk factor prioritization using multi-task deep learning	FDNN	[70]
	Risk factors selection for cardiovascular disease	SAE	[85]
	Deep feature selection approach using genomics data	MLNN	[71]
	Discovering of characteristic patterns of physiology	SAE	[23]
	A semisupervised learning method for EHR phenotype extraction	SDA	[6]
Diagnosis detection and segmentation by image processing	Risk classification for skin cancer	CNN	[39]
	Diagnosis of breast cancer using clinical images	SDA	[24]
	Diagnosis of Alzheimer disease using brain MRIs	SAE	[75]
	Deep feature learning for knee cartilage segmentation	CNN	[92]
	Identifying modes of variations in Alzheimer disease	RBM	[16]

abetic macular edema in retinal fundus images. They applied CNN to classify those images using a retrospective large datasets of nearly 128,000 retinal images.

The summary of our review based on above three categories demonstrated in Table 1. Readers for more comprehensive review about applications of deep learning in health informatics can refer to recent review papers provided by Miotto et al. [82], Shickel et al. [104] and Ravi et al. [95].

2.3 Introduction to Deep Architectures

Deep Learning is a subfield of machine learning algorithms that model raw data to higher-level abstraction by training a deep network consisting several hidden layers with linear and non-linear transformations [9, 65, 31]. In another word, deep learning applies computational techniques, which include multiple processing layers to learn feature representation with several levels of abstraction [65].

Deep learning applications include many areas. The major ones are speech recognition, image

processing, object detection and bio informatics or bio medicine [65]. In biomedical and health science, improvements in information systems, technological development and research laboratory equipments have created huge amount of data with many characteristics. Since deep feature learning outperformed some traditional methods such as singular value decomposition (SVD) or principal component analysis (PCA) in handling of high-dimensional clinical data, it has great potential for feature representation and dimensionality reduction in biomedical and biomedicine research [80].

Among all different deep architectures, four deep architectures are more popular in clinical data analysis [80]. 1) The Convolutional neural network (CNN), 2) Stacked Autoencoders (SAE), 3) Restricted Boltzmann Machine (RBM) and 4) Deep Belief Network (DBN). In this research, we use three different deep architectures including Stacked Autoencoders, Deep Belief Network and Variational Autoencoders for representation learning of continuous features. In this section we review each architecture briefly as following.

2.3.1 Introduction to Stacked Autoencoders (SAE)

Training process for deep neural networks with several hidden layers is known to be hard and challenging. Standard approach for learning neural network uses gradient-based optimization with back-propagation method by initializing random weights in network concludes poor training results empirically when there exist three or more hidden layers in deep network [63]. Hinton et al. [52] developed a greedy layer-wise unsupervised learning algorithm for training DBN parameters by using a RBM in the top of deep architecture. Bengio et al. [10] used greedy layer-wise unsupervised learning to train deep neural network when the building block of deep architecture is an autoencoder instead of the RBM. Stacked Autoencoders shown in Figure 4 is constructed by stacking multiple layers of autoencoder.

An autoencoder is trained to reconstruct its own inputs by encoding and decoding processes. Let us define $w^{(h,l)}$, $w^{(h,2)}$, $b^{(h,l)}$, $b^{(h,2)}$ as the parameters of h^{th} autoencoder for weights and biases in encoding and decoding processes respectively. Encoding process of each layer is a forward process and mathematically described as follows:

$$a^{(h)} = f(z^{(h)}), \tag{2.1}$$

$$z^{(h+1)} = w^{(h,1)}a^{(h)} + b^{(h,1)} \tag{2.2}$$

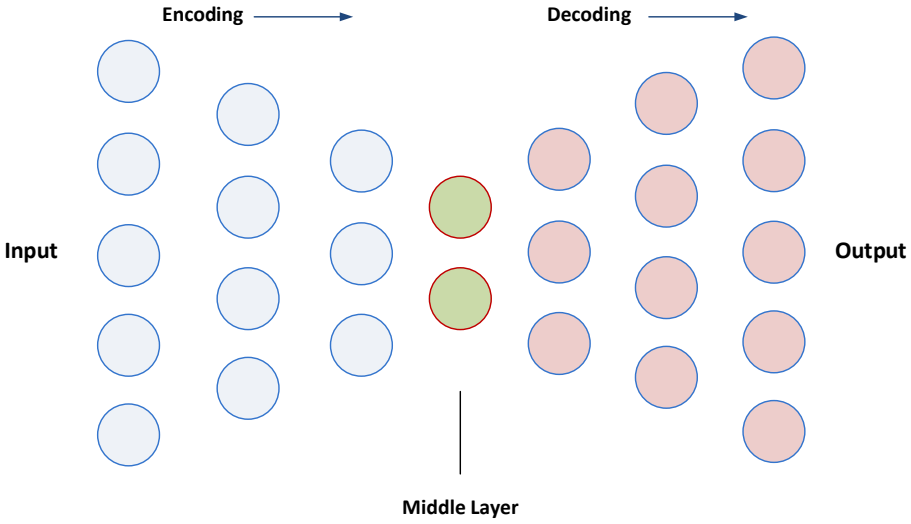


Figure 4: Stacked Autoencoders

$f(x)$ is an activation function such as sigmoid or hyperbolic tangent function for transforming data. If n represents the location of middle (latent) layer in stacked autoencoders, the decoding

process is to implement the decoding stack of each autoencoder below [64]:

$$a^{(n+h)} = f(z^{(n+h)}), \tag{2.3}$$

$$z^{(n+h+1)} = w^{(n+h,2)}a^{(n+h)} + b^{(n+h,2)}. \tag{2.4}$$

Training algorithm for estimating parameters of stacked autoencoders is based on a greedy layer-wise approach [10]. It means that each autoencoder should be trained by encoding and decoding process one by one. By training this deep network, $a^{(n)}$ (middle layer) demonstrates the highest representation of the input data [64]. In the simplest case, when an autoencoder with sigmoid activation function has only one hidden layer and takes input x, the output of encoding process will be :

$$z = Sigmoid_1(wx + b). \tag{2.5}$$

Therefore z is the vector of transformed input in the middle layer. In the second step (decoding process), z is transformed into the reconstruction $x^{'}$, i.e.,

$$x^{'} = Sigmoid_2(w^{'}z + b^{'}). \tag{2.6}$$

In the final step, autoencoder is trained by minimizing the reconstruction errors as follows:

$$\begin{aligned} &\text{Loss}(x, x^{'}) = \|x - x^{'}\| = \\ &\|x - \text{Sigmoid}_2(w^{'}(\text{Sigmoid}_1(wx + b)) + b^{'})\|. \end{aligned} \tag{2.7}$$

2.3.2 Introduction to Deep Belief Network (DBN)

Deep Belief Networks are graphical models that are constructed by stacking of several RBMs to get better performance rather than individual RBM. Hinton and Salakhutdinov [53] showed that DBNs can be trained in greedy layer-wise unsupervised learning approach. They defined the joint probability distribution between visible and hidden layers as follows:

$$P(x, h^1, ..., h^l) = \prod_{k=0}^{l-2} P(h^k|h^{k+1}) P(h^{l-1}, h^l) \tag{2.8}$$

Where, x= h^0, $P(h^{k-1}|h^k)$ is a conditional distribution for the visible units conditioned on the hidden units of the RBM at level k, and $P(h^{l-1}, h^l)$ is the visible-hidden joint distribution in the top-level RBM. This is illustrated in the figure below.

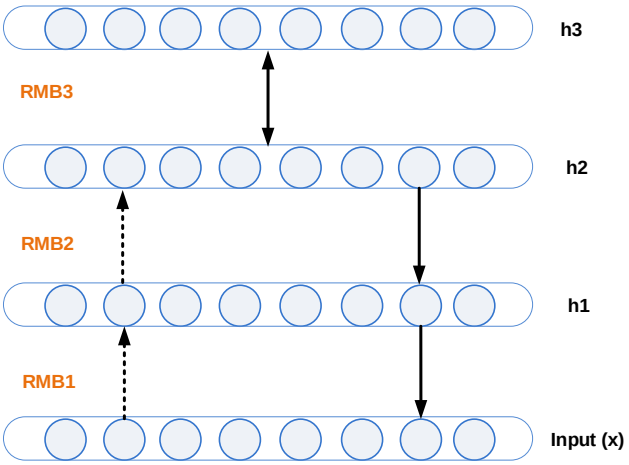


Figure 5: Deep Belief Network

In the layer-wised training, the input layer (visible unit) is trained as a RBM and transformed into the hidden layer, then the representation in hidden units will be considered as input data (visible units) for the second layer and this process continues. Readers for more detail about the

training process can refer to Hinton et al. [52] and Bengio et al. [10].

2.3.3 Introduction to Variational Autoencoders (VAE)

Variational Autoencoders has been developed as one of the most useful approaches to representation learning of complex data in recent years. VAE have already demonstrated promising performance in complicated data including handwritten digits, faces, house numbers, speech and physical models of scenes [34]. VAE has the structure of autoencoders including encoders, decoders and latent layer. Variational autoencoders are probabilistic generative models. Assume X is our input data and z is the latent variable, based on the total probability law we have:

$$P(x) = \int P(X, z)dz = \int P(X|z)P(z)dz \tag{2.9}$$

VAE tries to maximize the probability of each X in the training set according to the Eq.(2.9) under the generative process. $P(X|z)$ is the probability function of the observed data given to latent variable, which means how can find the distribution of input data based on distribution of sample of latent variable. The main idea in variational autoencoder is to attempt to sample values of latent variables (z) that are likely produce X, and construct $P(X)$ from those. In this way, we need a new function $Q(z|X)$ which can describe the distribution of z based on value of X. In the other words, z is sampled from an arbitrary distribution and Q can be any distribution such as standard normal distribution and help to compute $E_{z\sim Q}P(X|z)$. For doing that, we start to match $P(z|X)$ to $Q(z)$ using Kullback-Leibler divergence between $P(z|X)$ and $Q(z)$, for some arbitrary Q:

$$D[Q(z) \parallel P(z|X)] = E_{z\sim Q}[log^{Q(z)} - log^{P(z|X)}]) \tag{2.10}$$

The objective function of variational autoencoders can be formulated as following which maximizes $log^{P(X)}$ minus an error term:

$$log^{P(X)} - D[Q(z|X) \parallel P(z|X)] \tag{2.11}$$

We can infer $P(X)$ and $P(X|z)$ into Eq. (2.10) by applying Bayes rule to $P(z|X)$ and reformulate Eq. (2.11):

$$\begin{aligned} &log^{P(X)} - D[Q(z|X) \parallel P(z|X)] = \\ &E_{z\sim Q}[log^{P(X|z)}] - D[Q(z|X) \parallel P(z|X)] \end{aligned} \tag{2.12}$$

This equation known as the core of the variational autoencoder. In particular, the right hand side acts as an autoencoder, since Q is encoding X into z, and P is decoding it to reconstruct X.

2.4 Methodology

The method proposed in this research is a predictive approach using deep learning, which is called Deep Integrated Prediction (DIP) approach. The work flow of DIP approach is illustrated in Figure 6 that encompasses three main steps as following:

2.4.1 Features Partitioning

First, we separate categorical features from continuous features (if both exist in the dataset). Since the representation learning algorithms are different for continuous and categorical features we partition them in our framework.

2.4.2 Features Representation

The second step is feature representation section. Continuous features are transformed in higher-level abstraction by using deep network and categorical features are represented as vectors by a well-known word-to-vector algorithm:

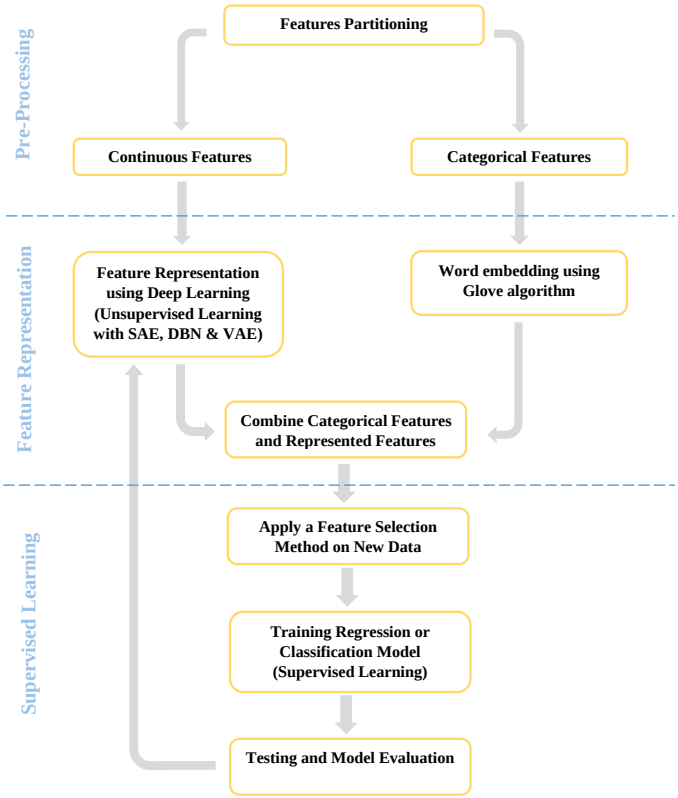


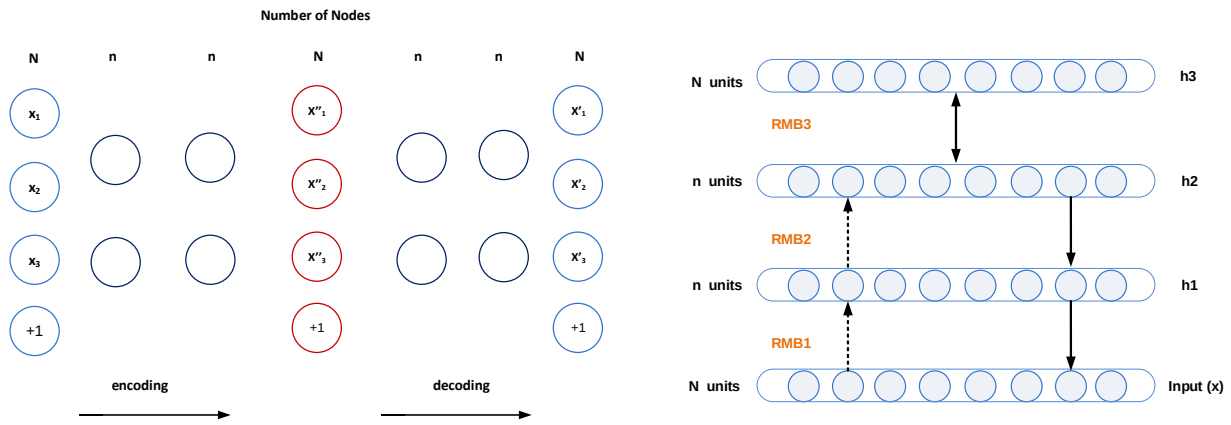
Figure 6: The Proposed DIP Workflow

1. **Categorical Features Representation using Word Embedding:** Discovering efficient representations of discrete categorical features has been a key challenge in a variety of applications as well as bioinformatics [29]. Word Embedding algorithms are developed to map the categorical features (words) to vectors of real numbers. Among several approaches for word embedding in the literature such as Matrix Factorization methods and Shallow Window-Based methods, we use Glove algorithm [90] as a well-known algorithm for word representation. GloVe algorithm uses

the global word co-occurrence matrix to learn the word representations.

2. Continuous Features Representation using Deep Learning: This step is the key step of our framework where we apply unsupervised learning using deep architecture to represent continuous features in order to achieve more robust features with less complexity. We do feature representation by three different deep architectures: stacked autoencoders, variational autoencoders and deep belief network.

The deep architecture of stacked autoencoders and variational autoencoders are considered with 5 hidden layers (two hidden layers of encoders, two hidden layers of decoders and one latent/middle layer) as shown in Figure 7(a).



(a) SAE and VAE Architecture **(b)** DBN Architecture

Figure 7: Deep architectures used for feature representation

In this deep architecture, N is the number of continuous variables in the dataset and n is a parameter. The middle hidden layer has N units, same as input and output layers, and the other four hidden layers have n units which is variant. The represented features are obtained from latent/middle layer and n is selected in an iterative process through unsupervised and supervised learning steps.

For deep belief network architecture, we choose a DBN with 3 hidden layers as depicted in

figure 7(b). In this architecture N refers to the number of continuous features and n is a parameter similar to SAE and VAE network.

The choice of deep architectures affects the performance of feature representation strongly. In our deep architectures, we consider different amount of n (hidden units) which can be less or higher than the number of original features. It means we not only try to transform data in lower dimensions (under-complete representation) but also we try to represent data in higher dimensions as well (over-complete representation) while an over-complete representations can be considered as an alternative of "compressed" or under-complete representation [114].

2.4.3 Supervised Learning

In the final step, the represented continuous and categorical features are combined with each others and then supervised learning to be performed on the top of new dataset. It begins with feature selection (if needed), which can apply any feature selection approach (e.g. random forests). Significant features from represented data are used to train a supervised learner (regression or classification) and after training step, model should be evaluated by some specific measures/indicators in testing process (e.g. in the regression problem this measure can be Mean Squared Error (MSE) or R-Squared). If the stop criteria is reached then we stop, if not, model captures the other deep architecture by changing the number of hidden units (n) and evaluates the new results. This iterative process will be repeated until model converges to some specific criteria or given number of iterations.

2.5 Implementation on Electronic Medical Records (EMRs)

In our experimental study, we implement our methodology on three different EMRs datasets. First we use a small datasets related to cardiovascular disease with high dimensional features, then

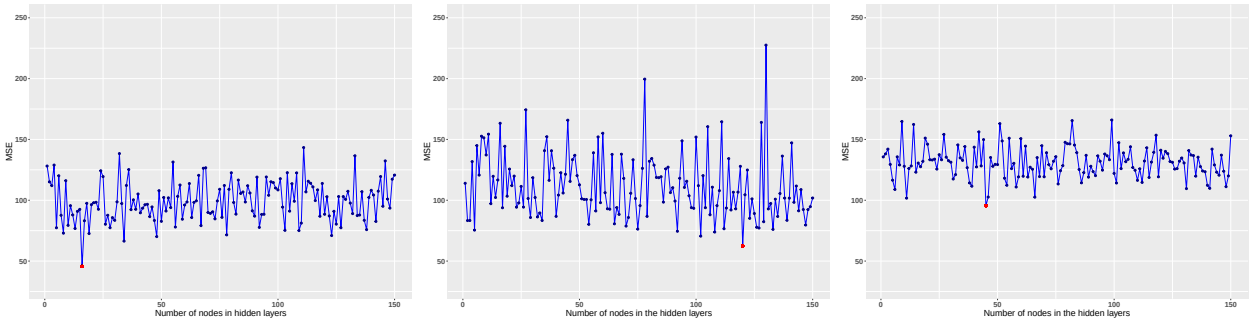
we apply our method on two large datasets from eICU collaborative research database. This study design (considering small and large datasets) helps us to discover the performance of our method in different scenarios and compare the choice of representation learning for each one.

2.5.1 Case study 1 (Small Dataset): DMC dataset

Cardiovascular disease (CVD) is the leading cause of death in the United States. Among different race groups, African-Americans are at higher risk of dying from CVD and have a worse risk factor profile. Left ventricular hypertrophy is an important risk factor in cardiovascular disease and echocardiography has been widely used for diagnosis. The data used in our first case study is belong to a subgroup of African-Americans with hypertension who are at high risk of cardiovascular/heart failure disease. Data are captured from patients admitted in the emergency department of Detroit Receiving Hospital in Detroit Medical Center (DMC). Across several features consisting demographic information, patient clinical history, individual health condition, laboratory tests, and cardiovascular magnetic resonance imaging results, 172 features remained after preprocessing step for data analysis related to 91 patients. As mentioned before, the goal is to predict value of heart damage risk level based on high-dimensional features.

We implemented all deep networks for feature representation using TensorFlow library in Python and applied word embedding in R using "text2vec" package. According to figure 6; we applied our approach for different deep architectures including SAE, DBN and VAE with different number of hidden units. For the supervised learning step we consider four well-known supervised classifiers: Random Forests, Lasso Regression, Decision Trees and Support Vector Machine (SVM). We used Mean Squared Error (MSE) as our evaluation measure for performance validation in testing process.

Figure 8 shows the performance of different deep architectures (SAE, DBN and VAE) across



(a) SAE performance **(b)** DBN performance **(c)** VAE performance

Figure 8: Performance of SAE, DBN and VAE based on different architectures

different number of nodes in the hidden layers (Random Forests used for supervised learning). We applied 150 different networks for each deep architecture and their performance for SAE, DBN and VAE is demonstrated in Figure 8. It is obvious the performance of each deep network is fluctuated across different number of hidden units. For instance SAE with n=16 nodes in all hidden layers (except latent layer) yields least error (MSE= 45.26) among all different architectures and for DBN and VAE; the best performance is achieved by 120 and 45 nodes respectively.

We performed our approach for different combinations of deep architectures (represented data) and supervised classifiers as well as original data (unrepresented data), and compared their performance based on average Mean Squared Errors (MSE) obtained from testing process with 5-folds cross validation. This comparison has been shown in Table 15. According to this results, our approach with representation learning reduces the prediction error and achieves a better accuracy rather than using the original features. Among different combinations, using stacked autoencoders for feature learning and Random Forests for supervised learning lead to the least MSE for this small dataset (DMC dataset). Figure 9 demonstrates the MSE for different deep architectures when we use random forests based on different number of trees. It is clear that SAE representation provides better feature learning across different number of trees in comparison with DBN, VAE and original

data.

Table 2: Performance comparison among represented data and original features

	Random Forests	**Lasso**	**SVM**	**Regression Trees**
SAE	**45.56**	**81.74**	**75.73**	**63.72**
DBN	62.54	96.06	100.49	74.04
VAE	75.41	103.55	98.82	101.05
Original	122.84	192.31	75.73	265.75

2.5.2 Case study 2 (Large Datasets): eICU dataset

In the second case study, we consider two large datasets from eICU collaborative research database. This database is populated with data from a combination of many critical care units in the Unites States. The data in the eICU database covers patients who were admitted to critical care units in 2014 and 2015. Among different care units, we select cardiovascular intensive care unit (eICU Cardiac) and Neurological intensive care unit (eICU Neuro). By integrating different features including demographics data, hospital and administration information, diagnosis and laboratory data, treatment and drugs information, monitored invasive vital sign data and clinical patient history data, we finalize more than 150 features for each dataset with approximately 7000 and 8000 records related to eICU Cardiac and eICU Neuro respectively. In this case study, our goal is to predict the patient length of stay in these ICU units based on personalized features. The ability to predict the LOS can improve the scheduling process which leads to patient waiting time and hospital cost reduction.

Table 3: Performance comparison among represented data and original features (ICU-Cardiac)

	RF	Lasso	SVM	RT
SAE	2.51	18.63	5.22	6.32
DBN	0.79	16.21	4.11	4.57
VAE	**0.08**	**6.31**	**2.62**	**2.41**
Original	2.71	17.21	6.35	7.32

Table 4: Performance comparison among represented data and original features (ICU-Neuro)

	RF	Lasso	SVM	RT
SAE	1.63	6.37	11.25	4.36
DBN	0.47	4.06	3.25	3.61
VAE	**0.02**	**0.54**	**1.99**	**1.88**
Original	1.92	8.71	12.36	5.73

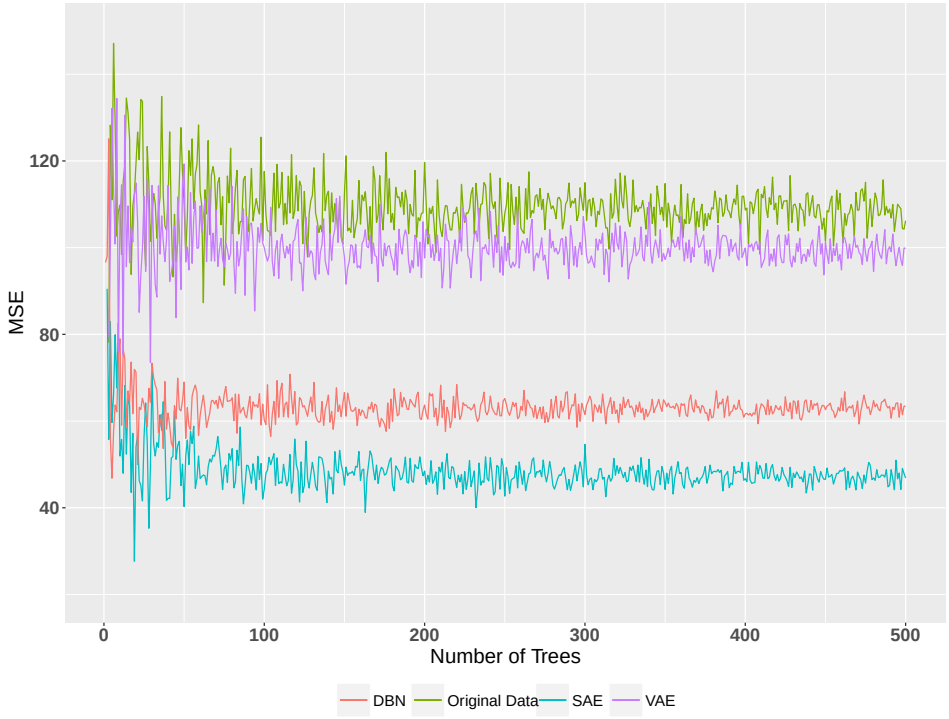
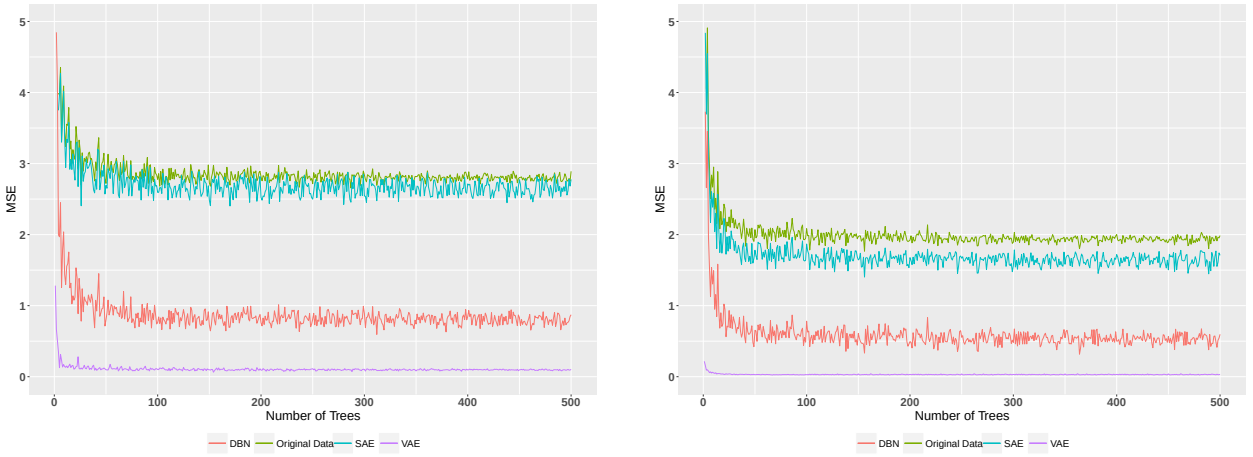


Figure 9: Performance of Random Forests across represented data and original features

We applied our DIP approach on both datasets. We trained different deep architectures (SAE, VAE and DBN) with different number of hidden unites and networks parameters (batch size, epoch number and learning rate) to find the best feature representation. Similar to the first case study, we used four different classifiers in supervised learning step on the top of both represented and original data. The results has been demonstrated in Table 3 and Table 4 for each dataset.

According to these results, using representation learning based on different deep architectures improved the accuracy of model (error reduction) for both datasets. Similar to the DMC dataset



(a) ICU-Cardiac **(b)** ICU-Neuro

Figure 10: Performance of Random Forests across represented data and original features

(Small dataset), Random Forests outperforms the other supervised learners in general, but against DMC dataset, variational autoencoders leads to significantly better results in comparison with SAE and DBN.

Although using original features achieves good results, representation learning using VAE provides impressive accuracy while the average of MSE in testing process with 5-folds cross validation are 0.08 and 0.02 for Cardiac ICU and Neuro ICU datasets respectively when we use random forests in supervised learning step. Also our model increases the R-squared from 93% to 98% and from 95% to 99% for the first and second large datasets respectively. In the other words, our model using VAE representation provides a perfect predictive approach for the second case study. Figures 10a and 10b demonstrate MSEs comparison for different deep networks when we use random forests with different number of trees.

2.6 Discussion and Conclusion

In this research, we developed a novel predictive approach using deep feature learning for applications of Electronic Medical Records (EMRs). Our Deep Integrated Prediction (DIP) approach

discovers the complexity and dependencies in the EMRs using unsupervised learning (feature representation) which improves the clinical prediction performance significantly. First, we applied our model on a small datasets obtained from Detroit Medical Center related to cardiovascular disease to predict the heart failure risk level (LVMI) and then we captured two large datasets from eICU collaborative research database to predict the patient length of stay in ICU units based on personalized features including demographics, diagnosis, medication and laboratory results information.

In both case study we applied four well-known supervised learning algorithms consisting of Random Forests, Lasso Regression, Decision Tree and SVM on the top of clinical represented features and original features. Our results indicate that feature learning using appropriate deep network improves the accuracy of all supervised learners. We used three different deep architectures (SAE, DBN and VAE) and considering different training parameters in each network (including number of hidden units, bach size, number of epochs and learning rate).

The results emphasize that the choice of representation learning plays an effective rule in the performance of clinical prediction. While in the first case study (small datasets), SAE has a better accuracy in comparison with DBN and VAE, for large datasets (eICU database), VAE outperforms the other deep architectures and SAE cannot improve the prediction results significantly. In other words, we can conclude that feature representation using deep learning would be effective for both small and large datasets and choice of deep network achieves different results. The advantage of VAE in learning true distribution of input features based on distribution of sample from latent variables makes it different and it seems that VAE achieves better representation in the case of large and more complex data in comparison with traditional autoencoders such as SAE and DBN.

In summary, we present a novel data-driven approach for predictive modeling of clinical data with high dimensional, complex and sparse features. Our model is the first model which use the ad-

vantages of variational autoencoders in clinical feature representation and compare its performance with two other traditional autoencoder deep architectures. We demonstrated that deep learning could be effective for small datasets as well as large data and our comparative study between small and large clinical datasets provides some new insights in the choice of deep representation. We believe that our model with great EHRs feature learning has potential to be applied in different clinical and health informatics aspects including treatment planning, risk factor identification, personalized recommendation and survival analysis. Also, our proposed framework is highly useful for exploiting a large amount of unlabeled data in the feature learning (unsupervised learning) step to extract high level abstraction of features when the labeled data are limited and expensive.

For further directions, we plan to apply our method to the other small, large and big datasets for different clinical predictive purposes like as personalized recommendations. We will involve the other deep architectures including Stacked Denoising Autoencoders and compare their performance with each others. Finally we will consider clustering task in the last step of our approach (instead of supervised learning) to discover important clinical patterns such as treatment schemes among patients.

CHAPTER 3 PATIENT SUBGROUP DETECTION APPROACH FOR PERSONALIZED HEALTHCARE

Traditional medicine typically applies one-size-fits-all treatment for the entire patient population whereas precision medicine develops tailored treatment schemes for different patient subgroups. One of the current focus of precision medicine emphasizes health disparities because health in populations is driven by biologic, environmental, social, and economic factors. The fact that some factors may be more significant for a specific patient subgroup motivates clinicians and medical researchers to develop new approaches to subgroup detection and analysis, which is an effective strategy to personalize treatment. In this chapter, we propose a novel patient subgroup detection method, called Supervised Biclustring (SUBIC) using convex optimization and apply our approach to detect patient subgroups and prioritize risk factors for hypertension (HTN) in a vulnerable demographic subgroup (African-American). Our approach not only finds patient subgroups with guidance of a clinically relevant target variable but also identifies and prioritizes risk factors by pursuing sparsity of the input variables and encouraging similarity among the input variables and between the input and target variables.

3.1 Problem Definition

The explosive increase of Electronic Medical Records (EMR) and emerge of precision (personalized) medicine in recent years holds a great promise for greatly improving quality of healthcare. In fact, the paradigm in medicine and healthcare is transferring from disease-centered (empirical) to patient-centered, the latter is called Personalized Medicine. The extensive and rich patient-centered data enables data scientists and medical researchers to carry out their research in the field of personalized medicine [85].

A crucial step in personalized medicine is to discover the most important input variables (dis-

ease risk factors) related to each patient. Since identification of risk factors needs multi-disciplinary knowledge including data science tools, statistics techniques and medical knowledge, many machine learning and data mining methods have been proposed to identify, select and prioritize risk factors. Some popular methods such as linear model with shrinkage [112] and random forest [15] effectively select significant risk factors for the entire patient population. However, these approaches are not capable of detecting risk factors for each patient subgroup because they are developed based on an assumption that the patient population is homogeneous with a common set of risk factors.

While the point of input variable selection is well taken, the association with small subgroups, a key notion in personalized medicine, is often neglected. As mentioned, personalized healthcare aims to identify subgroup of patients who are similar with each other according to both target variables and input variables. Discovering potential subgroups plays a significant role in designing personalized treatment schemes for each subgroup. Therefore, it is essential to develop a core systematic approach for patient subgroup detection based on both input and target variables [41]. A number of data-driven approaches have been developed for subgroup identification. The more popular methods can be divided in two categories: 1) Tree-based approaches [35] (or so called recursive partitioning), and 2) Biclustering approaches [91]. Tree based methods in subgroup analysis are greatly developed in recent years, such as Model-based recursive partitioning [122], Interaction Trees [107], Simultaneous Threshold Interaction Modeling Algorithm (STIMA) [36], Subgroup Identification based on Differential Effect Search (SIDES) [73], Virtual Twins [42], Qualitative Interaction Tree (QUINT)[37] and Subgroup Detection Tree [69]. The second approaches (Biclustering) have been extensively developed and applied to analyze gene expression data. Most of the biclustering algorithms developed up-to-date are based on optimization

procedures as the search heuristics to find the subgroup of genes or patients.

Tree-based methods detect patient subgroups using the relationship between input and target variables whereas biclustering methods just focus on clustering rows and columns of the input variables simultaneously to identify different subgroups with specific risk factors (prioritized input variables). The former employs a target variable to guide subgroup detection by selecting a common set of input variables. The latter selects subgroup of specific input variables without guidance of a target variable. Moreover, both approaches are heuristic in nature that subgroup detection and risk factor identification are sensitive to choices of data sets and initializations hence has a poor generalization performance. Our proposed method combines the strength of the both approaches by using a target variable to guide the subgroup detection and selecting subgroup of specific risk factors. Meanwhile, our systematic approach overcomes the stability limitation of both approaches by casting the problem into a stable and mature convex optimization framework. Figure11 demonstrates consecutive steps of our approach.

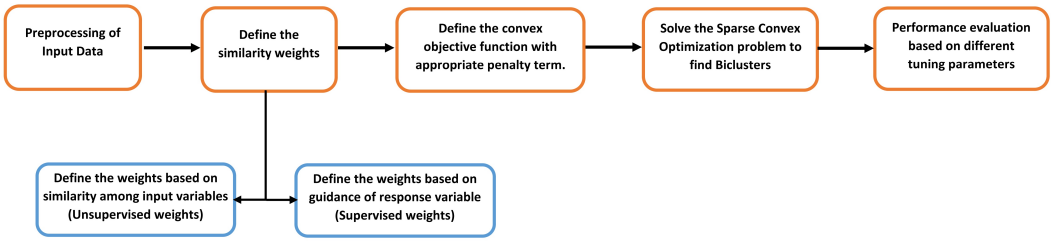


Figure 11: The consecutive steps of our approach

In this study, we propose a new supervised biclustering approach, called SUBIC, for solving patient subgroup detection problem. Our approach is the generalized (supervised) version of convex biclustering [27], which enables prediction of target variables for new input variables. Moreover, we employ the elastic-net penalty [126] (both l_1 and l_2 regularization terms) that encourages sparsity of the correlated input variable groups (X) with the guidance of a target value (Y). Our

model is specifically designed for patient subgroup detection and target variable prediction from high dimension data. To the best of our knowledge, our model is the first supervised biclustering approach that can be applied in many domains such as personalized medicine. To demonstrate the performance of SUBIC approach, we apply it to detect subgroups among hypertension (HTN) patients with guidance of left ventricular mass indexed to body surface area (LVMI), a clinically important target variable.

3.2 Related Works

Biclustering is defined as simultaneous clustering of both rows and columns in the input data matrix. Such clusters are important since they not only discover the correlated rows, but also identify the group of rows that do not behave similarly in all columns [38]. In the context of precision medicine, rows correspond to patients and columns correspond to input variables measured in each patient. Biclustering was originally introduced in 1972 [47], and Cheng and Church [25] were the first to develop a biclustering algorithm and applied it to gene expression data analysis. There exist a wide range of biclustering methods developed using different mathematical and algorithmic approaches. Tanay et al. [110] proved that biclustering is a NP-hard problem, and much more complicated than clustering problem [33]. Therefore, most of methods are developed based on heuristic optimization procedures [91]. Madeira and Oliveira [79], Busygin et al.[18], Eren et al. [38] and Pontes et al.[18] provided four comprehensive reviews about biclustering methods in 2004, 2008, 2012 and 2015 respectively. Based on the most recent review [91], biclustering approaches can be divided in two main groups. The first one refers to methods based on evaluation measures, which means some heuristic methods are developed using a measure of quality to reduce the solution space and complexity of biclustering problem. Table 5 demonstrates different

algorithmic categories within this group:

Table 5: Biclustering methods based on evaluation measure.

Algorithm	**Description**	**Prosperous Methods**
Iterative greedy search	These methods follow a greedy strategy to find an approximate solution. They improve the measure of evaluation in each step and construct a set of objects from the smallest possible solution space recursively or iteratively.	Direct Clustering [47], Cheng and Church [25], HARP Algorithm [120], Maximum Similarity Bicluster[76]
Stochastic iterative greedy search	These methods use a stochastic strategy by adding a random variable to the iterative greedy search in order to speed up the biclustering algorithm.	Flexible Overlapped Biclustering [117], Random Walk Biclustering [2], Reactive GRASP Biclustering [32], Pattern-Driven Neighborhood Search[5]
Nature-inspired meta-heuristics	These methods are developed based on a nature-inspired meta-heuristic, such as simulated annealing, ants colony and swarm optimization.	Simulated Annealing Biclustering [17], Evolutionary Algorithms for Biclustering [13], SEBI (Sequential Evolutionary Biclustering) [33], Multi-objective Evolutionary Algorithms for Biclustering[83]
Clustering-based approach	These methods carry out their search based on traditional clustering methods in one dimension and then use an additional approach to cluster second dimension.	Possibilistic Spectral Biclustering. [19], Biclustering with SVD and Hierarchical Clustering.[118]

The second group of approaches is called non metric-based biclustering methods that do not use any measure of quality (evaluation measure) for guiding the search. These methods use graph-based or probabilistic algorithms to identify the patterns of biclusters in data matrix. Table 6 summarizes different algorithms of non metric-based group:

Table 6: Biclustering method based on non-metric.

Algorithm	**Description**	**Prosperous Methods**
Graph-based approaches	These methods are developed based on the graph theory. They use nodes for either genes, samples or both gene and sample representations, or refer to nodes as representing the whole biclusters.	Statistical-Algorithmic Method for Bicluster Analysis (SAMBA)[110], Qualitative Biclustering algorithm (QUBIC)[68], Pattern-based Co-Regulated Biclustering (QoBi) [98], MicroCluster [123]
One-way clustering-based approaches	These methods are developed based on the same concept of clustering-based approached, but they do not use any measure of quality in their search path.	Coupled Two-way Clustering [44], Interrelated Two-way Clustering [111]
Probabilistic search	These methods are created using statistical modeling and probability theory.	Plaid Models [58], Rich Probabilistic Models [100], Gibbs Sampling [103], Bayesian Biclustering Model [45]
Linear algebra	These methods use linear algebra to apply linear mapping between vector spaces for describing and identifying the most correlated submatrices from the original dataset.	Spectral Biclustering [61], Iterative Signature Algorithm [11], Non-smooth Non-negative Matrix Factorization (nsNMF) [20]
Optimal reordering rows and columns	These methods are based on the strategy of performing permutations of the original rows and columns in the data matrix, to achieve a better arrangement and make biclusters.	Pattern-based Biclustering [50], order-preserving sub-matrices (OPSMs)[7]

One of the important aspects of bicluster structure is overlapping, which means several biclusters share rows and columns with each other. Because of the characteristic of search strategy in biclustering methods, overlapping may or may not be allowed among the biclusters. Most of the algorithms mentioned in Table 5 and Table 6 allow overlapping biclusters [91]. Since these algorithms use heuristic approach for guiding search, final biclusters may vary depending on how the algorithm is initialized. Therefore, they don't guarantee a global optimum nor are they robust against even small perturbations [27].

Recently, Chi et al.[27] formulated biclustering problem as a convex optimization problem and solved it with an iterative algorithm. Their convex biclustering model corresponds to checkerboard mean model, which means each data matrix component is assigned to one bicluster. They used the

concept of fused lasso [113] and generalized it with a new sparsity penalty term corresponding to the problem of convex biclustering. This method has some important advantages over the previous heuristic-based methods, that is, it created a unique global minimizer for biclustering problem, which maps data to one biclustering structure, therefore the solution is stable and unique. Also it used a single tuning parameter to control the number of biclusters. Authors performed simulation studies to compare their algorithm with two other biclustering algorithms, dynamic tree cutting algorithm [62] and sparse biclustering algorithm [109], which assume the checkerboard mean structure. Results showed that convex biclustering outperforms the competing approaches in terms of Rand index [27].

Despite the improved performance, the convex biclustering method, like other biclustering methods, does not exploit a target variable on subgroup detection and risk factor selection. As a result, the detected biclusters do not link to target variables of interest. Hence, it is unable to predict the target variable for future input variables. Clearly, the target variable such as LVMI provides a critical guidance for detection and selection of the meaningful biclusters (patient subgroups). To overcome this limitation, we develop a new supervised biclustering algorithm which uses a target variable to guide the patient subgroup detection and risk factor selection.

Moreover, the l_1 penalty term alone in convex biclustering encourages the sparsity of individual input variables but overlooks the fact that they are also correlated within variable groups. To overcome both limitations, we introduce a new elastic-net regularization term that seeks sparsity of the correlated variable groups and employs a target variable to supervise the biclustering optimization process. Consequently, our model is truly a predictive model that is capable of predicting value of the target variable for new patients. In the next section, we describe our method in detail.

3.3 Method

3.3.1 The object function of the SUBIC method

The goal of convex biclustering is to identify biclusters using convex optimization. Chi et al. [27] formulated biclustering problem as a regularized regression problem where their convex biclustering approach can be seen as a generalization of the Fused Lasso. They developed this model for checkerboard mean structure. We generalized and extend this concept and propose a novel sparse supervised convex biclustering that is capable of using a target variable to guide optimization. (you have mentioned it in the previous paragraph)

Let's assume that the input data matrix $X_{n\times p}$ represents n instances with different p input variables and Y_n is the continues target variable (e.g. LVMI), corresponds to n^{th} instance (patients). According to the checkerboard mean structure, we assume R and C are the sets of rows and columns of the bicluster B respectively, and $x_{i,j}$ refers to elements belong to the bicluster B, the observed value of $x_{i,j}$ can be defined as [27]: $x_{i,j} = \mu_0 + \mu_{RC} + \varepsilon_{i,j}$, where μ_0 is a baseline mean for all elements, μ_{RC} is the mean of bicluster corresponds to R and C, and $\varepsilon_{i,j}$ refers to error that is i.i.d with $N(0, \sigma)$. With considering non-overlapping biclusters, this structure corresponds to a checkerboard mean model.[61] Without loss of generality, we ignore μ_0 from all elements. The goal of biclustering is to find the partition indices with regard to R and C then estimate the mean of each corresponding bicluster (B). To achieve this goal, we minimize the following convex objective function:

$$F_{\lambda_1,\lambda_2} = \frac{1}{2}\|X - T\|_F^2 + P(T), \tag{3.1}$$

where matrix $T \in R^{n \times p}$ includes our optimization parameters, which are the estimate of means matrix. The first term is frobenius norm of matrix $X - T$ refers to error term and $P(T) = P_1(T) + P_2(T)$ is the elastic-net regularization penalty term formulated as follows:

$$P_1(T) = \lambda_1[\Sigma_{i<j} w_{i,j} \|T_{.i} - T_{.j}\|_2^2 + \Sigma_{i<j} h_{i,j} \|T_{i.} - T_{j.}\|_2^2], \tag{3.2}$$

and

$$P_2(T) = \lambda_2[\Sigma_{i<j} w_{i,j} \|T_{.i} - T_{.j}\|_1 + \Sigma_{i<j} h_{i,j} \|T_{i.} - T_{j.}\|_1]. \tag{3.3}$$

It is clear that this objective function is similar to subset selection problem in regularized regression [112]. In the penalty function λ_1 and λ_2 are tunning parameters. The first term penalized by λ_1 is a l_2-norm regularization term and the second term penalized by λ_2 is a l_1-norm regularization term. Therefore the penalty term $P(T)$ acts as regression elastic-net penalty [126]. $T_{i.}$ and $T_{.i}$ refer to ith row and column of matrix T, which can be considered as a cluster center (centroid) of ith row and column respectively.

By minimizing the objective function defined in Eq.3.1 with sparsity based regularization, the cluster centroids are shrunk together when the tunning parameters increase. It means that sparse optimization tries to unify the similar rows and columns to specific centroid simultaneously. Finding the similarity between rows and columns is guided by different weights ($w_{i,j}$, $h_{i,j}$), which are included in objective function. These weights has been defined based on distance between input variables ($X_{.i} - X_{.j}$ and $X_{i.} - X_{j.}$), distance between target variables ($Y_i - Y_j$) and correlation between input variables and target variable ($X_{.i}, Y_{.j}$). Therefore both input variables and

target variable play significant rule in guiding of sparsity to find the best centroids. The first kind of weights ($w_{i,j}$) proceeds the columns convergence and the second one ($h_{i,j}$) proceeds the rows convergence. The weights are constructed from un-supervised and supervised parts, where:

$$w_{i,j} = w^1_{i,j} + w^2_{i,j} \quad \text{and} \quad h_{i,j} = h^1_{i,j} + h^2_{i,j}. \tag{3.4}$$

The unsupervised part ($w^1_{i,j}, h^1_{i,j}$) attempts to converge rows(columns) based on the similarity exists among input variables, and the supervised part ($w^2_{i,j}, h^2_{i,j}$) converges rows and columns according to the similarity of input and target variables. Since the rows and columns are in R^n and R^p spaces respectively, it is required to normalize the weights (recommended the sum of row weights and column weights to be $\frac{1}{\sqrt{n}}$ and $\frac{1}{\sqrt{p}}$ respectively). We used the idea of sparse Gaussian kernel weights [27] for defining $w^1_{i,j}, w^2_{i,j}, h^1_{i,j}, h^2_{i,j}$. Table 7 demonstrates the mathematical description of weights:

Table 7: Description of the weights formula.

#	**Weight Formula**	**Description**
1	$w^1_{i,j} = l^k_{i,j} \exp^{(-\varphi \|X_{.i} - X_{.j}\|_2^2)}$	This weight is to converge the similar columns in terms of distance similarity measure. $l^k_{i,j}$ is 1 when j^{th} column is among the k-nearest neighbor of i^{th} column, otherwise it is zero. Therefore it guarantees the weights are sparse. $(0 \leq \varphi \leq 1)$
2	$w^2_{i,j} = l^k_{i,j} \exp^{(-\varphi \|corr(x_{.i},Y) - corr(x_{.j},Y)\|)}$	This weight is the supervised part of $w_{i,j}$, the goal is to converge the columns that have similar correlation with target variable. It means that the features which behave similarly with target variable should be converged. In our model we used Pearson correlation that assumes a linear relationship between input variable and target variables. $(0 \leq \varphi \leq 1)$
3	$h^1_{i,j} = l^k_{i,j} \exp^{(-\varphi \|X_{i.} - X_{j.}\|_2^2)}$	This weight is the same as $w^1_{i,j}$, which attempts to converge the similar rows with lower distance from each other.$(0 \leq \varphi \leq 1)$
4	$h^2_{i,j} = l^k_{i,j} \exp^{(-\varphi \sqrt{\|(Y_i - Y_j)\|})}$	This weight is supervised part of $h_{i,j}$, and it converges the rows that are similar in term of target variable value. This weight considers the role of target variable in clustering of similar rows.$(0 \leq \varphi \leq 1)$

The way to define the weights has a substantial impact on the quality of biclustering. The weights described above guarantee the sparsity of the problem and employ the similarity of all input and target variables in supervised and unsupervised manner. According to defined weights, the two columns (rows) that are more similar with each other will get larger weight in the convex penalty function, therefore in minimization process, those columns (rows) should be in higher priority, and it means that convex minimizer attempts to cluster the similar columns (rows). The choice of elastic-net penalty term can overcome the lasso limitations. While the l_1-norm can generates a sparse model, the quadratic part of the penalty term encourages grouping effect and stabilizes the l_1-norm regularization path. Also the elastic-net regularization term performs very suitable for high dimensional data with correlated input variables and would be a better model when $p \gg n$ specially in the case of gene expression data and precision medicine problems [126].

3.3.2 The algorithm to train the SUBIC model

It can be proved easily that the objective function in Eq.3.1 is a convex function. Therefore we need to develop appropriate algorithm to solve this unconstrained convex optimization. Since the second part of penalty function, $P_2(T)$ is undifferentiated we use Split Bregman method [119] developed for large-scale Fused Lasso. It can be shown that this method is equivalent to the alternating direction method of multipliers (ADMM) [14]. Readers can refer to Split Bregman method [119] or ADMM algorithm [14] for more comprehensive explanation. According to both methods we need to use splitting variable and Lagrangian multiplier and then apply augmented Lagrangian for undifferentiated part ($P_2(T)$) of objective function. First we need to transform our problem to the equality-constrained convex optimization problem by defining two new variables (V,S) and adding two constraints correspond to $P_2(T)$ and then use Lagrangian multipliers:

$$\begin{aligned}
\min \quad & F_{\lambda_1,\lambda_2} = \frac{1}{2}\|X-T\|_F^2 + \lambda_1[\Sigma_{i<j} w_{i,j}\|T_{.i} - T_{.j}\|_2^2 + \Sigma_{i<j} h_{i,j}\|T_{i.} - T_{j.}\|_2^2] + \\
& \lambda_2[\Sigma_{i<j} w_{i,j}\|T_{.i} - T_{.j}\|_1 + \Sigma_{i<j} h_{i,j}\|T_{i.} - T_{j.}\|_1], \\
\text{subject to :} \quad & w_{i,j}(T_{.i} - T_{.j}) = V_{i,j} \quad \forall i,j; i<j, \\
& h_{i,j}(T_{i.} - T_{j.}) = S_{i,j} \quad \forall i,j; i<j,
\end{aligned} \tag{3.5}$$

where V and S are matrices in $R^{n\times p}$. Assuming the differentiated part of objective function in (1) is F'_{λ_1,λ_2}, the Lagrangian Multiplier for the above problem is:

$$\begin{aligned}
& \tilde{L}(T,M,N,V,S) = F'_{\lambda_1,\lambda_2} + \lambda_2[\Sigma_{i<j} w_{i,j}\|V_{i,j}\|_1 + \Sigma_{i<j} h_{i,j}\|S_{i,j}\|_1] + \Sigma_{i<j}\langle M_{i,j}, w_{i,j}(T_{.i} - T_{.j}) - V_{i,j}\rangle + \\
& \Sigma_{i<j}\langle N_{i,j}, h_{i,j}(T_{i.} - T_{j.}) - S_{i,j}\rangle,
\end{aligned} \tag{3.6}$$

where M and N are the vectors of dual variables (Lagrangian Multipliers) corresponding with each constraints in Eq.3.5 (totally there are $\binom{n}{2} + \binom{p}{2}$ constraints). Finally the Augmented Lagrangian function of Eq.3.5 is as following:

$$L(T, M, N, V, S) = F'_{\lambda_1,\lambda_2} + \lambda_2[\Sigma_{i<j} w_{i,j}\|V_{i,j}\|_1 + \Sigma_{i<j} h_{i,j}\|S_{i,j}\|_1] + \Sigma_{i<j} \langle M_{i,j}, w_{i,j}(T_{.i} - T_{.j}) - V_{i,j}\rangle +$$

$$\Sigma_{i<j} \langle N_{i,j}, h_{i,j}(T_{i.} - T_{j.}) - S_{i,j}\rangle + \frac{\mu_1}{2}[\Sigma_{i<j}\|w_{i,j}(T_{.i} - T_{.j}) - V_{i,j}\|_2^2] + \frac{\mu_2}{2}[\Sigma_{i<j}\|h_{i,j}(T_{i.} - T_{j.}) - S_{i,j}\|_2^2], \tag{3.7}$$

where $\mu_1 > 0$ and $\mu_2 > 0$ are two parameters. The Split Bregman algorithm for supervised convex biclustering problem described below:

Algorithm 1 Split Bregman algorithm for Training the SUBIC model

1: Initialize T^0, V^0,S^0, M^0, N^0

2: **repeat**

3: $$T^{k+1} = \text{argmin}_T \quad \frac{1}{2}\|X - T\|_F^2 + \lambda_1[\Sigma_{i<j} w_{i,j}\|T_{.i} - T_{.j}\|_2^2 + \Sigma_{i<j} h_{i,j}\|T_{i.} - T_{j.}\|_2^2]$$
$$+ \Sigma_{i<j} \left\langle M_{i,j}^k, w_{i,j}(T_{.i} - T_{.j}) - V_{i,j}^k \right\rangle + \Sigma_{i<j} \left\langle N_{i,j}^k, h_{i,j}(T_{i.} - T_{j.}) - S_{i,j}^k \right\rangle$$
$$+ \frac{\mu_1}{2}[\Sigma_{i<j}\|w_{i,j}(T_{.i} - T_{.j}) - V_{i,j}^k\|_2^2] + \frac{\mu_2}{2}[\Sigma_{i<j}\|h_{i,j}(T_{i.} - T_{j.}) - S_{i,j}^k\|_2^2]$$

4: $$V_{i,j}^{k+1} = \tau_{\frac{\lambda_2}{\mu_1}}(w_{i,j}(T_{.i}^{k+1} - T_{.j}^{k+1}) + \mu_1^{-1} M_{i,j}^k) \quad \forall i,j; \quad i < j$$

5: $$S_{i,j}^{k+1} = \tau_{\frac{\lambda_2}{\mu_2}}(h_{i,j}(T_{i.}^{k+1} - T_{j.}^{k+1}) + \mu_2^{-1} N_{i,j}^k) \quad \forall i,j; \quad i < j$$

6: $$M_{i,j}^{k+1} = M_{i,j}^k + \delta_1(w_{i,j}(T_{.i}^{k+1} - T_{.j}^{k+1}) - V_{i,j}^{k+1}) \quad \forall i,j; i < j; \quad 0 < \delta_1 \leq \mu_1$$

7: $$N_{i,j}^{k+1} = N_{i,j}^k + \delta_2(h_{i,j}(T_{i.}^{k+1} - T_{j.}^{k+1}) - S_{i,j}^{k+1}) \quad \forall i,j; i < j; \quad 0 < \delta_2 \leq \mu_2$$

8: **until**

9: Convergence

τ acts as a soft thresholding operator defined on vector space and satisfying the following equation:

$$\tau_\lambda(w) = [t_\lambda(w_1), t_\lambda(w_2), ...]^T, \quad t_\lambda(w_i) = \text{sgn}(w_i)\max\{0, |wi - \lambda|\}. \tag{3.8}$$

3.3.3 The SUBIC based prediction approach

For prediction of the target variable based on supervised biclustering framework, we introduce a simple yet effective approach based on generalized additive model (GAM) [48]. Assuming that

K biclusters {BC_1,BC_2,..., BC_K } are detected by training the SUBIC model, we consider K classifiers corresponding to each biclusters, i.e., $f_k(y|x_{bc_k}, x_{\text{new}}) = \overline{y}_{bc_k}$. It means that each classifier predicts the target value as an average of the target variables of the corresponding bicluster. The proposed GAM model is as follows:

$$g(E(y)) = R_1(x_{bc_1}) + R_2(x_{bc_2}) + ... + R_k(x_{bc_k}), \text{ where } \quad R_k(x_{bc_k}) = q_k f_k(y|x_{bc}, x_{\text{new}}). \tag{3.9}$$

q_k is defined as normalized weight based on posterior probabilities. Assuming that each bicluster follows a Gaussian distribution as $N(\mu_i, \sigma)$ and $P(bc_k|x_{new})$ is the posterior probability which refers to the probability of each bicluster given a new instance , we can define q_k as below:

$$q_k = \frac{P(bc_k|x_{\text{new}})}{\sum_{i=1}^{k} P(bc_i|x_{\text{new}})}, \text{ where } \quad P(bc_k|x_{\text{new}}) = P(x_{\text{new}}|bc_k) \times P(bc_k). \tag{3.10}$$

$P(x_{\text{new}}|bc_k)$ is conveniently calculated based on Gaussian distribution assuming equal variance and zero covariance and $P(bc_k)$ is the prior that can be calculated by counting the number of instances in each bicluster.

3.4 Experimental Study and Model Evaluation

Evaluating the quality of clustering (biclustering) algorithms has been known very hard in the literature. For assessing the performance of our approach, we carry out simulation studies and use Rand index (RI) [93]and Adjusted rand index (ARI) [59] as two popular measures for evaluating the quality of clustering. Since our biclustering method is supervised, we simulate data for input and target variables based on a checkerboard mean structure. We used normal distribution with

different means to generate simulated data. Figure 12 illustrates an example simulation study.

As shown below, data was simulated in 20×20 matrix. Data in each segment has different size and were created based on a different normal distribution, all sections are generated with low-noisy data ($\sigma = 1.5$). Input data in segments (2, 3, 4 and 5) are in high positive correlation with the target variable and input data in segment (6, 7, 8 and 9) are in high negative correlation with the target variable. Segments 1 and 10 in general, are similar with very low correlation with target variable. Segments 1 and 3 of the target variable are positive and the other two sections have negative values.

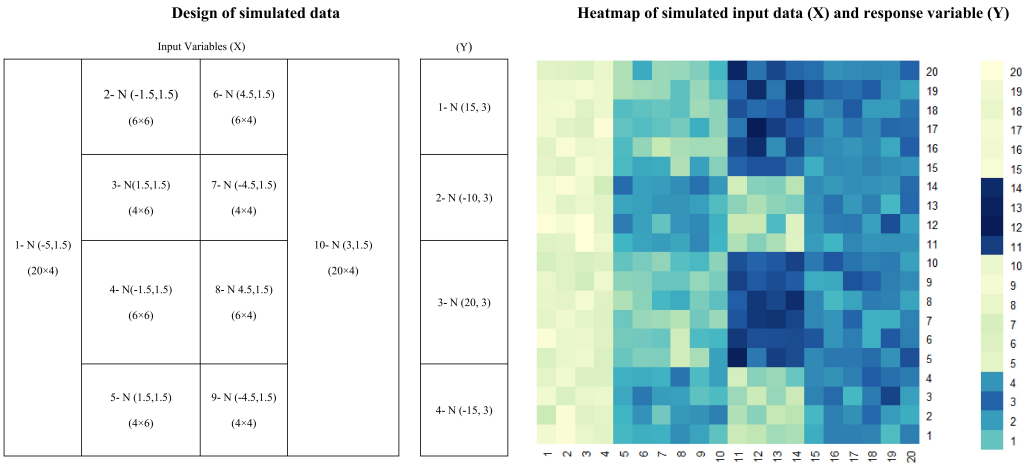


Figure 12: The chessboard structure (left panel) and the simulated data (right panel).

According to this assumptions and consider the effect of target variable, it is clear that the true number of biclusters should be 16 (not 10). It means that segments 1 and 10 include 4 biclusters within each. The results of SUBIC implementation for different tuning parameters are displayed in Figure 13.

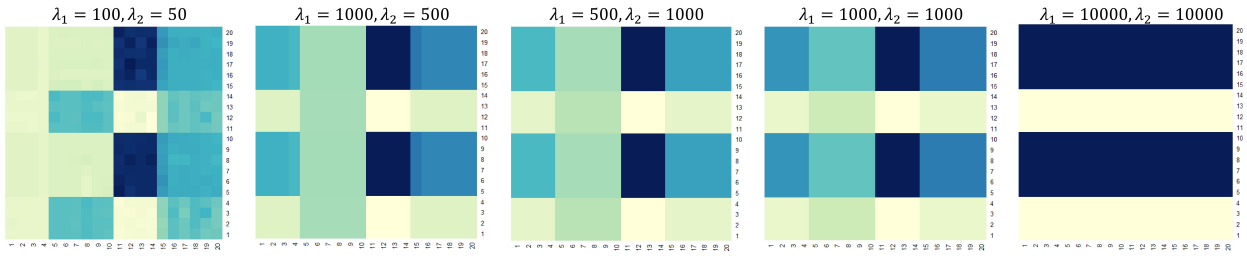


Figure 13: Results of SUBIC method implementation on the simulated data for different tuning parameters

According to Figure 13, tuning parameters provide a flexible mechanism to analyze data with both high and low variances. It is obvious that by increasing λ_1 and λ_2, rows and columns are unified to mean in each bicluster but when λ_1 and λ_2 get larger values such as 10000, bicluster patterns are “smoothed out" and the number of biclusters reduces.

We consider different scenarios in Figure 14 to show that the flexibility and generalization of our method. Panel *a* shows our supervised biclustering approach, SUBIC, with elastic-net penalty (l_1 and l_2) as the most general case. By zeroing out λ_1, the l_2 penalty (special case 1), SUBIC becomes the extended (supervised) version of the convex biclustering approach [27] (Panel *b*). If we instead zero out the supervised weight components $w_{i,j}^2$ and $h_{i,j}^2$ (special case 2), SUBIC becomes extended unsupervised convex biclsutering with elastic-net penalty (Panel *c*). Finally, if we zero out both the l_2 penalty and the supervised weight components $w_{i,j}^2$ and $h_{i,j}^2$ (special case 3), SUBIC becomes the *bona fide* convex biclustering method reported in [27]. Therefore, our SUBIC approach is sufficiently general and flexible that employs a target value to guide the subgroup detection by encouraging sparsity of the number of variable groups and variables within each group. Correspondingly, our SUBIC approach most accurately detect the biclusters given in the ground truth. Panel *a* and *b* in Figure 14 confirm that the impact of supervised weights (target value guidance) in identifying of true biclusters in comparison with convex biclustering approach

[27] (Panels c and d). Also in both cases the elastic-net regularization appears more accurate in detecting true biclusters.

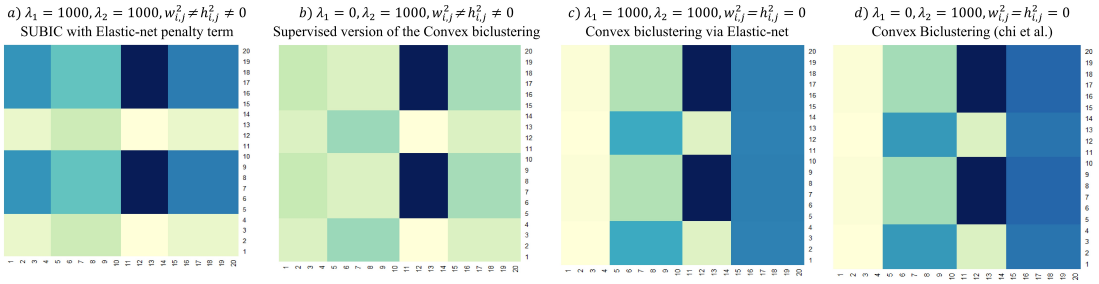


Figure 14: Different scenarios which show the flexibility of SUBIC method

We extend the above idea to 80×80 matrix and consider different design (true biclusters) with two noise levels (low and high) for assessment of our model. We use different tuning parameters in each design and evaluate SUBIC method with rand index and adjusted rand index. The results of average RI and ARI over 10 replicates are depicted in table 8 and 9 for low-noisy and high-noisy data respectively.

Table 8: Evaluation results based on RI and ARI for different designs with low noisy simulated data

Row	Design	σ	$\lambda_1 = \lambda_2 = 100$		$\lambda_1 = \lambda_2 = 1000$		$\lambda_1 = \lambda_2 = 10000$	
			RI	ARI	RI	ARI	RI	ARI
1	2×4	1.5	0.85	0.71	0.99	0.96	0.79	0.65
2	4×4	1.5	0.79	0.62	0.98	0.95	0.76	0.64
3	4×8	1.5	0.73	0.56	0.98	0.97	0.68	0.59
4	8×8	1.5	0.82	0.69	0.96	0.93	0.72	0.61

Table 9: Evaluation results based on RI and ARI for different designs with high noisy simulated data

Row	Design	σ	$\lambda_1 = \lambda_2 = 100$		$\lambda_1 = \lambda_2 = 1000$		$\lambda_1 = \lambda_2 = 10000$	
			RI	ARI	RI	ARI	RI	ARI
1	2×4	3	0.65	0.53	0.90	0.88	0.99	0.96
2	4×4	3	0.68	0.58	0.85	0.81	0.99	0.97
3	4×8	3	0.59	0.49	0.93	0.90	0.98	0.93
4	8×8	3	0.55	0.43	0.87	0.82	0.99	0.95

As shown above, the performance of SUBIC is fully tunable using the pair of tuning parameters in response to data with different levels of variances. From Table 8, it is clear that SUBIC's superior performance is very stable for both low and high variance data. In particular, the robust performance against high-variance data is achieved by setting larger values of tuning parameters.

3.5 Application in Personalized Medicine

In this section we demonstrate how SUBIC method is capable of identifying patient subgroups with guidance of the target variable LVMI. We study the population of African-Americans with hypertension and poor blood pressure control who have high risk of cardiovascular disease.

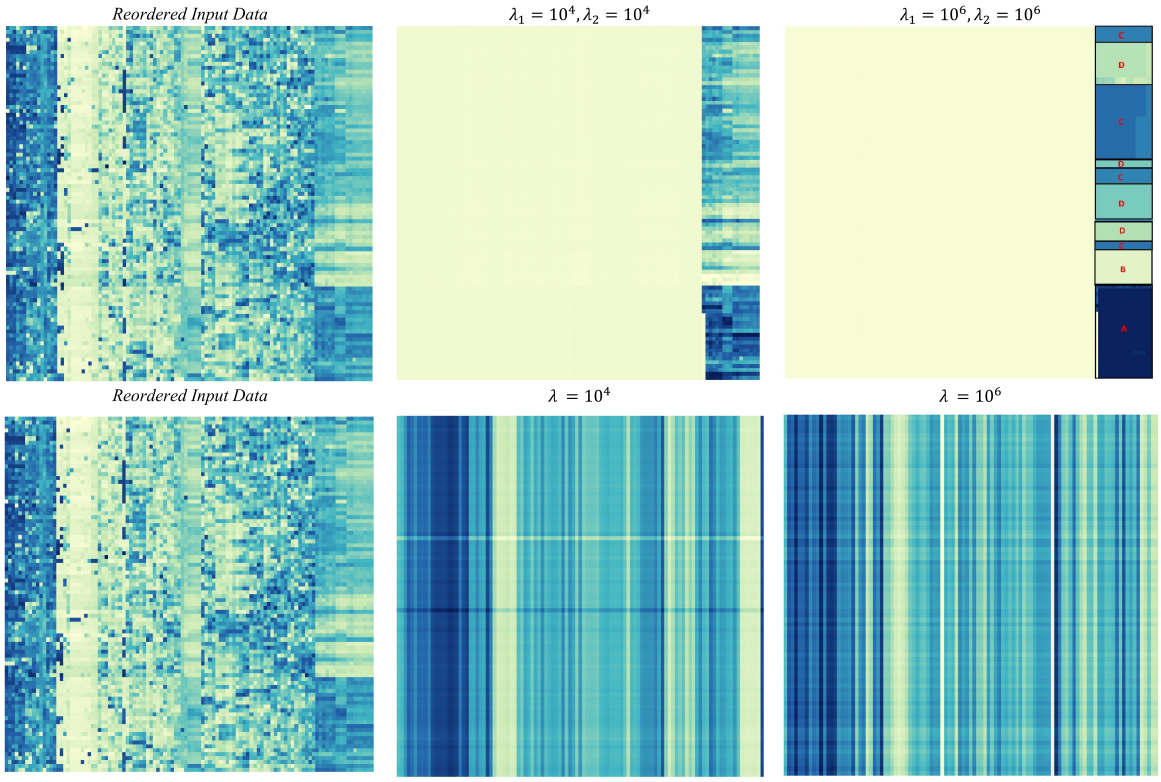


Figure 15: Results of SUBIC implementation (top panel) and COBRA method (bottom panel) on the data related to African-American patients at high risk of cardiovascular disease.

Data are obtained from patients enrolled in the emergency department of Detroit Receiving Hospital. After preprocessing step, our data consists of 107 features including demographic char-

acteristics, previous medical history, patient medical condition, laboratory test result, and CMR results related to 90 patients. To achieve a checkerboard pattern, we reorder rows and columns (original data) at first [27] using hierarchical clustering and then apply SUBIC method. The results are shown in the top panel of Figure 15. In addition, we implemented convex biclustering method (COBRA) developed by Chi et al. [27] using package "cvxbiclustr" in R for comparing with our SUBIC method. Results obtained using different tuning parameters (λ) are shown in the bottom panel of Figure 15.

In Figure 15, our SUBIC method detects 4 subgroups using 15 features for $\lambda_1 = \lambda_2 = 10^4$. These 15 features belong to 3 major groups of features including: 1) Waist Circumference Levels (mm); 2) Average Weight (kg) and 3) Calculated BMI. The statistics related to these risk factors based on 4 groups of patients is summarized in Table 10. It is worth mentioning that other potential risk factors such as "Troponin Level" or "Plasma Aldosterone" can be also significant but these three groups of features are sufficient to describe the disparity among patients based on guidance of the target variable LVMI. On the contrary, COBRA method fails to find any patient subgroups for this data set.

Table 10: Average of three disparity factors and LVMI (along with standard deviation) for subgroups detected by SUBIC

Subgroup	size	Waist Circumference Levels (mm)	Average Weight (kg)	Calculated BMI	LVMI
A	24	1248.86 (104.73)	125.17 (13.16)	41.65 (5.20)	85.78 (11.95)
B	28	1092.65 (74.55)	99.73 (11.01)	35.18 (3.81)	82.74 (13.77)
C	29	972.83 (89.67)	84.88 (10.25)	30.01 (4.97)	80.97 (13.70)
D	9	813.33 (123.79)	64.46 (10.65)	23.83 (4.39)	79.38 (11.8)
Total	90	1067.76 (163.59)	98.20 (22.24)	34.10 (7.28)	82.64 (12.98)

3.6 Discussion and Conclusion

In this chapter, we have developed a novel supervised subgroup detection method called SUBIC based on convex optimization. We used the idea of convex biclustering approach [27] and proposed a new supervised biclutering approach which overcomes the limitation of previous works when we have a target variable.

SUBIC is a predictive model that combines the strength of biclustering and tree-based methods. We introduced a new elastic-net penalty term in our model and defined two new weights in our objective function to enable the supervised training under the guidance of a clinically relevant target variable in detecting biclusters. We further presented a generalized additive model for predicting target variables for new patients. We evaluated our SUBIC approach using simulation studies and applied our approach to identify disparities among African-American patients who are at high risk of cardiovascular disease. Future directions include extending our SUBIC approach to predict categorical target variables, such as stages and subtypes of heart diseases.

CHAPTER 4 TREATMENT RECOMMENDATION USING SURVIVAL ANALYSIS FOR PERSONALIZED HEALTHCARE

Survival analysis has been developed and applied in the number of areas including manufacturing, finance, economics and healthcare. In healthcare domain, usually clinical data are high-dimensional, sparse and complex and sometimes there exists few amount of time-to-event (labeled) instances. Therefore building an accurate survival model from electronic health records is challenging. With this motivation, we address this issue and provide a new survival analysis framework using deep learning and active learning with a novel sampling strategy. First, our approach provides better representation with lower dimensions from clinical features using labeled (time-to-event) and unlabeled (censored) instances and then actively trains the survival model by labeling the censored data using an oracle. As a clinical assistive tool, we introduce a simple effective treatment recommendation approach based on our survival model. In the experimental study, we apply our approach on SEER-Medicare data related to prostate cancer among African-Americans and white patients. The results indicate that our approach outperforms significantly than baseline models.

4.1 Problem Statement

Survival analysis has been applied in several real-world applications such as healthcare, manufacturing and engineering in order to model time until the occurrence of an future event of interest (e.g. biological death or mechanical failure) [54]. Censoring attribute of survival data makes survival analysis different from the other prediction approaches. One popular survival model is the Cox Proportional Hazards model (CPH) [30] which model the risk of an event happening based on linear combination of the covariates (risk factors). The major problem of Cox-based models is linear relationship assumption between covariates and the time of event occurrence. Hence, there

have been developed several model to handle non-linear relationship in survival analysis like as survival neural network and survival random forest models.

In the healthcare area, medical researchers apply survival analysis on EHRs to evaluate the significance of many risk factors in outcomes such as survival rates or cancer recurrence and subsequently recommend treatment schemes. There exist two specific challenges in survival analysis from EHRs: 1) Clinical data are usually high dimensional, sparse and time-dependent where applying traditional survival approaches do not perform well enough to estimate the risk of a medical event of interest accurately, 2) In many health survival applications, labeled data (time-to-event instances) are small, time-consuming and expensive to collect. In this situation, it is hard to learn a survival model based on traditional approaches which able to predict the relative risk of patients precisely.

To address the first challenge, recently, semi-supervised learning using deep feature representation has been applied in number of areas and could improve the performance of different machine learning tasks as well as survival analysis. In the other word, applying unsupervised learning using deep learning can reduce the complexity of raw data and provide robust features with lower dimensions. Using this represented features in the supervised learning algorithms (e.g. survival models) establishes a semi-supervised learning framework which achieve higher performance.

To overcome the second challenge, active learning is well suited to get high accuracy when the labeled instances are small or labeling is expensive and time-consuming. Active learning approach from censored data has been rarely addressed in the literature. However it has been widely used in the other aspects of health informatics where the labeled data are scarce.

In this chapter, first, we propose a novel survival analysis approach using deep learning and active learning termed DASA. Our model is capable to learn more accurate survival model using

high dimensional and small size EHRs in comparison with some baseline survival models. Second, we introduce a personalized treatment recommendation approach based on our survival analysis model which can compare the relative risk (or survival time) associate with different treatment plans and assign better one. We evaluate our approach using SEER-Medicare dataset related to prostate cancer. We consider two racial subgroup of patients (African-American and whites) in our analysis and apply our model on each dataset separately.

Our contributions in this chapter lie into three folds: 1) To best of our knowledge, we propose the first deep active survival analysis approach with promising performance, 2) In our active learning framework we develop a new sampling strategy specifically for survival analysis and 3) Our model with proposed treatment recommendation approach is highly potential to apply for evaluation of new treatment effect on new patients where the labeled data is scarce.

4.2 Background

In this section, we review some basic concepts and approaches for modeling of survival analysis and active learning. The background related to deep learning has been discussed in the chapter 1.

4.2.1 Introduction to Survival Analysis

Survival analysis is a kind of statistical modeling where the main goal is to analyze and model time until the occurrence of an event of interest, such as death in biological systems and failure in mechanical machines. The challenging characteristics of survival data is the fact that time-to-event of interest for many instances is unknown because the event might not have happened during the period of study or missing tracking occurred caused by other events. This concept is called censoring which makes the survival analysis is different. The special case of censoring is where the observed survival time is less than or equal to the true event time called right-censoring the

main focus of our study.

Since the censored data is present in survival analysis, the standard statistical and machine learning approaches are not appropriate to analyze and predict time-to-event outcome because those approaches miss the censored/right-censored instances. Survival modeling provides different statistical approaches to analyze such censored data in many real-world applications.

In survival analysis, a given instance i, represented by a triplet (X_i, δ_i, T_i) where X_i refers to the instance characteristics and Ti indicates time-to-event of the instance. If the event of interest is observed, Ti corresponds to the time between baseline time and the time of event happening, in this case δ_i = 1. If the instance event is not observed and its time to event is greater than the observation time, Ti corresponds to the time between baseline time and end of the observation, and the event indicator is δ_i = 0. The goal of survival analysis is to estimate the time to the event of interest (T) for a new instance X_j.

Survival and hazard functions are the two main functions in survival modeling. The survival function indicates to the probability that the time to the event of interest is not less than a determined time (t). This functions (S) denoted by following formula:

$$S(t) = Pr(T > t) \tag{4.1}$$

The initial value of survival function is 1 when $t = 0$ and it monotonically decreases with t. The second function, hazard function indicates the rate of occurrence of the event at time t given that no event occurred earlier. It describes the risk of failure (dying) changing over time. The

hazard function (or hazard rate or failure rate) is defined as following:

$$h(t) = \lim_{\delta(t)\to 0} \frac{Pr(t \leq T \leq t + \delta(t) | T \geq t)}{\delta(t)} \tag{4.2}$$

Survival and hazard function are non-negative functions. While all the survival function decreases over time, The shape of a hazard function can be in different forms: increasing, decreasing, constant, or U-shaped.

There exist several models for survival analysis in the literature. Among all, Cox Proportional Hazards (CPH) model [30] is the most popular model for survival analysis. CPH estimates the hazard function $h(x)$ as a regression formulation:

$$h(t, X_i) = h_0 \, exp(X_i \beta) \tag{4.3}$$

where h_0 is the baseline hazard function which can be an arbitrary nonnegative function of time and X_i refers to covariate vector for instance i, and β is the coefficient vector estimated after survival model training by maximizing the cox partial likelihood. Because the baseline hazard function $h_0(t)$ in CPH is not determined, we cannot use the standard likelihood function in training process [30]. The partial likelihood is the product of the probability of each instance i at event time T_i that the event has happened for that instance, over the summation of instances (R_j) probability who are still at risk in this time (T_i):

$$L(\beta) = \prod_{i=,\delta_i=1} \frac{exp(X_i \beta)}{\sum_{j \in R_j} exp(X_j \beta)} \tag{4.4}$$

Since the censored instances exist in survival data, the standard evaluation metrics such as

mean squared error and R-squared are not appropriate for evaluating the performance of survival analysis [49]. In survival analysis, the most popular evaluation metric is based on the relative risk of an event for different instances called concordance index or c-index. This measure defines as following formula:

$$\frac{1}{N}\sum_{i,\delta_i=1}\sum_{j,y_i<y_j} I[S(\hat{y_i}|X_i) < S(\hat{y_j}|X_j)] \tag{4.5}$$

Where N refers to the all comparable instance pairs and S is the survival function. The main motivation for using c-index in survival analysis is originated from the fact that the medical doctors and researchers are often more interested in measuring the relative risk of a disease among patients with different risk factors, than the survival times of patients.

In general, the survival analysis models can be divided into two main categories: 1) statistical methods including non-parametric, semi-parametric and parametric and 2) machine learning based methods such survival trees, bayesian methods, neural networks and random survival forests. Readers for more comprehensive review can refer to the recent review provided by wang et al. [116].

4.2.2 Introduction to Active Learning

Active learning is a subfield of machine learning and statistical modeling. The goal of an active learner is the same as a passive learner but the key idea behind active learning is that a machine learning algorithm can lead to better performance with fewer training labels if it can select the data for learning. An active learner chooses queries, usually in the form of unlabeled data instances to be labeled by an oracle which can be a human annotator. Active learning is very efficient in many data-driven applications, where there exist numerous unlabeled data but labels are rare,

time-consuming, or expensive to be labeled [102].

Since large amounts of unlabeled data is nowadays often available and can be easily collected by automatic processes, active learning would be demanding in modern applications in order to reduce the cost of labeling. The active learning framework overcomes the challenge of insufficient labeled data by efficiently modeling the process of obtaining labels for unlabeled data. The advantage is that the active learner just requires to query the labels of just a few, carefully selected instances during the iterative process in order to achieve more accurate learner [56].

There exist several approaches/scenarios in which active learners ask queries. The three main approaches widely used in the literature are [102]: 1) membership query synthesis [3], 2) stream-based selective sampling [4], and 3) pool-based sampling [67]. For all approaches, there are also several different query strategies that have been developed to decide which unlabeled instances should be selected. Among above three approaches, pool-based sampling is most popular in many real-world applications. This approach has been demonstrated in Figure 16:

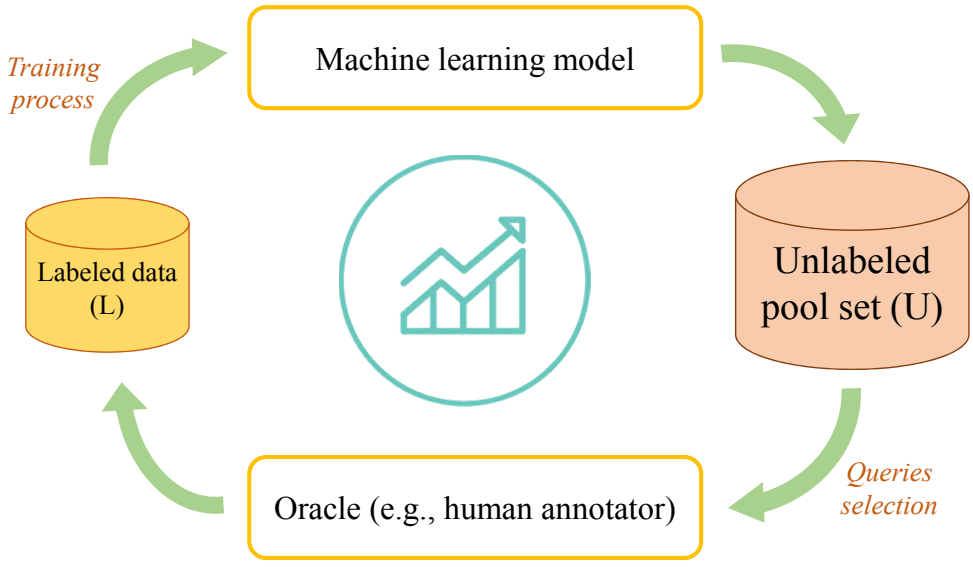


Figure 16: The pool-based active learning approach [102]

According to Figure 16, in pool-based sampling approach, A learner may start to be trained with a few number of labeled instances (L), then requests labels for one or more carefully selected unlabeled instances (U) using an oracle. After labeling, the new instance is simply added to the labeled set(L), and the learner proceeds training process in a standard supervised way. This process continues until some specified iterations or achieved desired accuracy.

4.3 Related Works

Deep learning and active learning as two advanced machine learning methods have been applied in different areas but there exist a few research in the literature that use the benefit of deep learning or active learning in survival analysis. In this section we review the research works which use any of those methods in survival analysis.

Vinzamuri et al. [115] provided the first ever active learning framework for survival analysis. They developed this approach just for regularized Cox regression survival models. Authors proposed a novel sampling strategy based on discriminative gradient for selecting the best candidate from the unlabeled pool set. Finally, they evaluated their model performance using public EHRs datasets and compared with some state of the art survival regression methods.

In the deep learning domain, there exist few studies which developed survival analysis framework using deep learning recently. In 2016, Ranganath et al. [94] proposed a new survival model using deep learning termed deep survival analysis. They used Deep Exponential Family (DEF) for capturing complex dependencies from clinical features including laboratory measurements, diagnosis, and medications codes. They applied their model on a large EHR dataset related to coronary heart disease. In the other research [77], authors introduced a new deep learning approach which can directly predict the survival times for graft patients using foundations of multi-task learning.

They demonstrated that their model outperforms usual survival analysis models such as cox proportional hazard model in terms of prediction quality and concordance index.

Katzman et al. [60] proposed a cox proportional hazards deep multi-layer perceptron called DeepSurv to predict risk of event occurrence for patient and provided personalized treatment recommendations. They performed their approach on simulated and real-world datasets for testing and evaluation. Finally, They used DeepSurv on real medical studies to illustrate how it can provide treatment recommendations. In the other research, Lee et al. [66] introduced a different approach called DeepHit which employs deep architecture to estimate the survival times distribution. They used neural network including two types of sub-networks: 1) a single shared sub-network and 2) family of cause-specific sub-networks. They evaluated their method based on real and synthetic datasets which illustrate that DeepHit leads to better performance in comparison with state of the art methods.

Based on our review, there exist no study to develop a survival analysis approach using both deep learning and active learning. We address this gap in the literature to propose a deep active learning framework for survival analysis. However, There are some studies that develop deep active learning methods for other machine learning tasks. For example, Zhou et al. [125] developed a semi-supervised learning framework termed active deep network (ADN) for sentiment analysis. They used restricted Boltzmann machines (RBM) for feature learning based on labeled reviews and large amount of unlabeled reviews, then applied gradient-descent based supervised learning for fine tuning and constructing semi-supervised framework. Finally they used active learning in their framework to improve model performance. In the other study, Liu et al. [74] proposed a deep active learning approach using Deep Belief Network (DBN) for classifying hyperspectral images in remote sensing application. A summary of our review has been illustrated in Table 11 which

Table 11: Summary of research works used deep learning or active learning in survival analysis

Authors	Research	DL	AL	SA	Ref
Zhou et al. (2013)	proposed semi-supervised sentiment classification algorithm	✓	✓		[125]
Vinzamuri et al. (2014)	developed survival regression for censored data for electronic health records		✓	✓	[115]
Ranganath et al. (2016)	introduced a deep hierarchical generative approach for survival analysis in heart disease	✓		✓	[94]
Nei et al. (2016)	proposed a survival analysis model applied on high-dimensional multi-modal brain images	✓		✓	[89]
Liao et al. (2016)	proposed a survival analysis framework using a LSTM model	✓		✓	[72]
Huang et al. (2017)	developed a survival model using CNN-based and one FCN-based sub-network and applied on pathological images and molecular profiles	✓		✓	[58]
Chaudhary1 et al. (2017)	introduced a DL based, survival model on hepatocellular carcinoma patients using genomic data	✓		✓	[21]
Liu et al. (2017)	proposed an active learning approach using DBN for classification of hyperspectral images	✓	✓		[74]
Luck et al. (2017)	developed a patient-specific kidney graft survival model using principle of multi-task learning	✓		✓	[77]
Sener&Savarese. (2017)	developed an active learning framework using CNN for image processing applications	✓	✓		[101]
Katzman et al. (2018)	proposed a Cox proportional hazards deep neural network for personalized treatment recommendations	✓		✓	[60]
Lee et al. (2018)	developed a survival model using deep learning which trained based on a loss function that uses both risks factors and survival times	✓		✓	[66]

Note: DL, AL and SA refer to Deep Learning, Active Learning and Survival Analysis.

indicates no research have been developed yet to address a survival approach using deep learning and active learning.

4.4 Methodology

The method developed in this research is an active learning based survival analysis uses a novel sampling strategy. In our model, we apply deep learning for feature reduction and extraction, when data is high-dimensional, complex and sparse. Since in survival analysis we deal with censored and uncensored instances, the active learning design should be different from the regular approach. In our framework, we consider censored and uncensored instances in the training set as survival analysis needs both instances in the training process and we consider uncensored data as unlabeled

instances in the pool set which their labels (time to event) are unknown.

The general framework in our survival analysis approach includes two main steps: 1) Deep feature learning for survival data and 2) Active learning based survival analysis. In the first step we do unsupervised learning using deep learning to represent features in higher level abstractions and extract data into lower dimensions. We represent both labeled (time to event) and unlabeled (censored) instances with together ($X_{train} \bigcup X_{pool}$) to obtain strong representation using pool of unlabeled data. In the other words, our framework uses the advantages of abundant unlabeled data to provide less complex and more robust features (labeled and unlabeled) for survival analysis.

In the second step, we apply our novel active learning based survival analysis on the represented/lower dimensions features obtained from the first step. This process demonstrates in Figure 17:

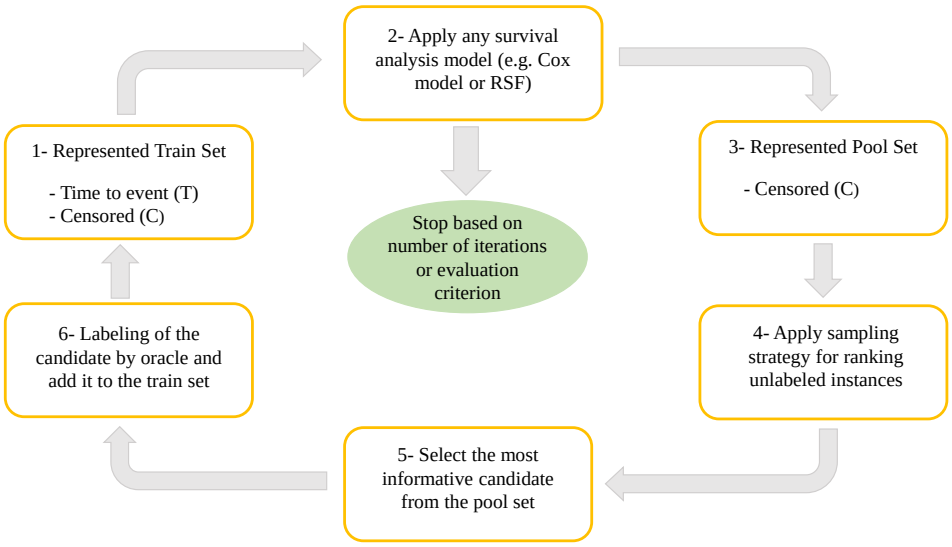


Figure 17: Active Survival Analysis Approach

According to this Figure, we start by applying a survival analysis method such as Cox-based regression or Random survival forest on represented train set. In the next step we use our novel sampling strategy (explained in the next section) to rank the unlabeled data based on their infor-

mativeness level. Then we select the most informative candidate from the pool and add it to the train set and repeat the process untill the stop criteria happens.

4.4.1 Expected Performance Improvement (EPI) Sampling (Query) Strategy

All active learning scenarios as well as pool-based active learning use the informativeness measure for evaluation of unlabeled instances to select the best query (the most informative unlabeled instance). There exist several proposed approach which formulate such query strategies in the literature which can be categorized in general frameworks [102]:1- uncertainty sampling, 2- query by committee, 3- expected model change, 4- expected error reduction, 5- variance reduction and 6- density weighted methods.

In this research we developed a new sampling (query) strategy based on properties of survival analysis. In our strategy, we select the unlabeled instance as the most informative instance (the best query) when it has the greatest performance change to the current survival model if we knew its label. Our sampling model use concordance index (C-index) to define the informative measure to query the unlabeled data. The survival model is trained again by adding a new instance (X^+) from the pool to the training set: $Train_{new}$ = $Train \bigcup X^+$ and the performance change is formulated based on the c-index difference as follows:

$$\Delta C_{X^+} = C_{new\ model} - C_{current\ model} \tag{4.6}$$

Similar to the other active learning sampling strategy, Our goal is to select the most informative instance which could maximally improve the current model performance. This selection can be

formulated as follows:

$$X^* = \underset{X^+ \in pool}{\operatorname{argmax}} \Delta C_{X^+} \tag{4.7}$$

Since in the real-world applications, We do not know the true label (time to event) of the instances in the pool, We should calculated the expected performance change over all possible time to events (T_s) for each unlabeled records as follows:

$$X^* = \underset{X^+ \in pool}{\operatorname{argmax}} \frac{\sum_{s=1}^{S} h(T_s|X^+)\, \Delta C_{X^+}}{\sum_{s=1}^{S} h(T_s|X^+)} \tag{4.8}$$

Our sampling strategy works for all survival analysis approaches such as cox-based models, parametric models and random survival forests. As an example for the cox regression, ΔC_{X^+} can be formulated as following equation and X^* is chosen based on Eq. 4.8.

$$\Delta C_{X^+} = \frac{1}{N}[\sum_{\delta_i=1} \sum_{T_i<T_j} (\hat{\beta}_2^s X_i > \hat{\beta}_2^s X_j) - \sum_{\delta_i=1} \sum_{T_i<T_j} (\hat{\beta}_1 X_i > \hat{\beta}_1 X_j)] \tag{4.9}$$

Where $\hat{\beta}_1$ and $\hat{\beta}_2$ are the estimated cox model coefficients trained based on the current and new training set ($Train_{new}$). N refers to the comparable (permissible) pairs in validation set for calculating c-index.

4.4.2 Proposed Deep Active Survival Analysis (DASA) Algorithm

Algorithm 1 describes our deep active survival analysis approach called DASA in detail. First, we apply deep feature learning on both train and pool sets. In this step we need to keep the weights of deep network for representation learning of new instances. In line 6, we apply survival analysis on deep represented features ($Deep\_Survival$). This framework is flexible and all survival models

can be used in this step. Then we start active learning iterations using EPI sampling strategy and update the pool and train sets until convergence.

Algorithm 2 Deep Active Survival Analysis (DASA) Algorithm

Require: Training set (X_T), Pool set (X_P), Survival status (δ), Time to event (T), Deep architecture parameters (hidden layers, hidden units, ...), Active learning maximum iteration (max_iter)

Round = 1

Training deep network for feature reduction on ($X_T \bigcup X_P$)

$Train\ set \longleftarrow X_T^{'}$

$Pool\ set \longleftarrow X_P^{'}$

repeat

    Model = $Deep\_Survival\ (X_T^{'}, \delta, T)$

    for each record in the pool ($x \in X_P^{'}$) **do**

        Apply EPI sampling strategy and calculate the expected performance improvement for each instance

    end for

    X^* = $argmax_{x \in X_P^{'}} \frac{\sum_{s=1}^{S} h(T_s|x)\ \Delta C_x}{\sum_{s=1}^{S} h(T_s|x)}$

    Labeling (time-to-event) of X^* by an Oracle based on original features

    $X_P^{'} \longleftarrow X_P^{'} - \{X^*\}$

    $X_T^{'} \longleftarrow X_T^{'} \bigcup \{X^*\}$

    $\delta_{X^*} \longleftarrow 1$

    $Round \longleftarrow Round + 1$

until $Round \neq max\_iter$

4.4.3 Treatment Recommendations Using Proposed DASA Approach

In this section, we propose a simple yet effective approach to discover treatment patterns and treatment recommendations using DASA. Our method is highly useful when EHRs are high-dimensional and small size. Suppose $X_T = \{X_1^T, X_2^T, ..., X_n^T\}$ is the treatment set and $X_A = \{X_1^A, X_2^A, ..., X_N^A\}$ refers to all other personalized features related to each patient where $N >> n$. Therefore, the input features is the union of these two sets ($X_T \bigcup X_A$). Since in the case of high-dimensional features, traditional approaches such as cox proportional hazard or random survival forests cannot find the pattern of specific features (e.g. small treatment set), we first represent X_A using deep learning to a lower dimension set ($X_A^{'}$) and then combine this represented set with the treatment set (X_T) to build the new feature set ($X_{new} = X_A^{'} \bigcup X_T$). In the second phase, we apply our active learning framework to train an accurate survival model based on new features and then find the pattern of treatment sets and interpret the results (e.g. comparison the coefficient of treatment options using Cox model or finding the importance of different treatment plan using random survival forests).

In our treatment recommendation approach, we transform many clinical features to a small feature set with higher level abstraction and more robust features. While we represent patient information to lower dimension using deep learning we combine non-represented treatment options (as features of interest) to the represented set and then perform survival analysis using active learning framework. In the next section, we demonstrate how our approach discover the treatment patterns better than traditional approaches.

4.5 Experimental Study: Survival Analysis for Prostate Cancer (SEER-Medicare Data)

In this section, we evaluate the performance of our approach (DASA) through experimental study. We use the Surveillance, Epidemiology and End Results (SEER)-Medicare linked database from SEER program of the National Cancer Institute (NCI). SEER-Medicare data is a powerful and unique source of epidemiological data on the occurrence and survival rates of cancer in the United States. In our study, we use prostate cancer SEER-Medicare data to evaluate our survival analysis approach and provide some insights by treatment recommendation.

4.5.1 Datasets: SEER-Medicare Prostate Cancer Data

Prostate cancer is the most popular diagnosed invasive cancer among men, with approximately 56% of all prostate cancer patients diagnosed in men with age 65 years and older [105]. Fortunately, a wide range of men (nearly 90%) diagnosed with non-metastatic prostate cancer and 5-year relative survival rate is very high for them. The death rate for prostate cancer is different among different populations. A good example of this racial disparity is the death rate for African-American men which is 2.5 times higher than white men. there exists a critical need to develop precision survival analysis for each cohort and discover the pattern of treatment.

In this study, we consider the SEER-Medicare data into two racial groups: 1) African-American patients and 2) White patient. Both groups are including many features (more than 300 features) such as demographic data, socioeconomic variables, tumor information and assigned treatment with approximately 1000 and 5000 patients respectively.

Since SEER-Medicare data is high-dimensional, sparse and complex, feature representation using deep learning can build more robust features when we use pool of unlabeled data (censored

instances) in the representation process. In the other hand, our method using active learning has highly potential to improve the performance of survival models when we deal with small sample size (including time-to-event and censored instances). In this way, in experimental study, we consider small samples in training of survival model and show that how our approach can improve the prediction performance in comparison with baseline.

For labeling of the censored instances (unlabeled data) in active learning framework we used some scientific reports such as SEER cancer statistics review from National Cancer Institute (NCI) [55] which acts as a prior knowledge to establish an oracle. One of these statistics is illustrated in Table 12:

Table 12: 5-Year conditional relative prostate cancer survival and 95% confidence intervals

Stage at Diagnosis	**Survival Time Since Diagnosis**	**Percent Surviving Next 5 years**	
		Percent	Confidence Interval
Local	0-Year	100%	(100, 100)
	1-year	100%	(100, 100)
	3-year	100%	(100, 100)
Regional	0-Year	100%	(100,100)
	1-year	99.3%	(98.9, 99.5)
	3-year	98.9%	(98.4, 99.2)
Distant	0-Year	29.2%	(28.4, 29.9)
	1-year	34.1%	(33.1, 35.1)
	3-year	45.6%	(43.9, 47.2)
Unstaged	0-Year	76.6%	(75.6, 77.5)
	1-year	81.1%	(79.8, 82.1)
	3-year	82.8%	(81.4, 84.1)

To evaluate the performance of our approach, we first employ CPH regression model (as a well-known survival analysis approach) and demonstrate how DASA can improve its performance based

on different training sample size. For deep feature representation we used Stacked Autoencoder (SAE) deep architectures with 5 hidden layers. Figure 18 shows the average performance of our approach for 20 iterations in comparison with baseline on the test data. We sampled training set with 25 instances from African-American patients over 10 runs and calculated the average performance in each iteration.

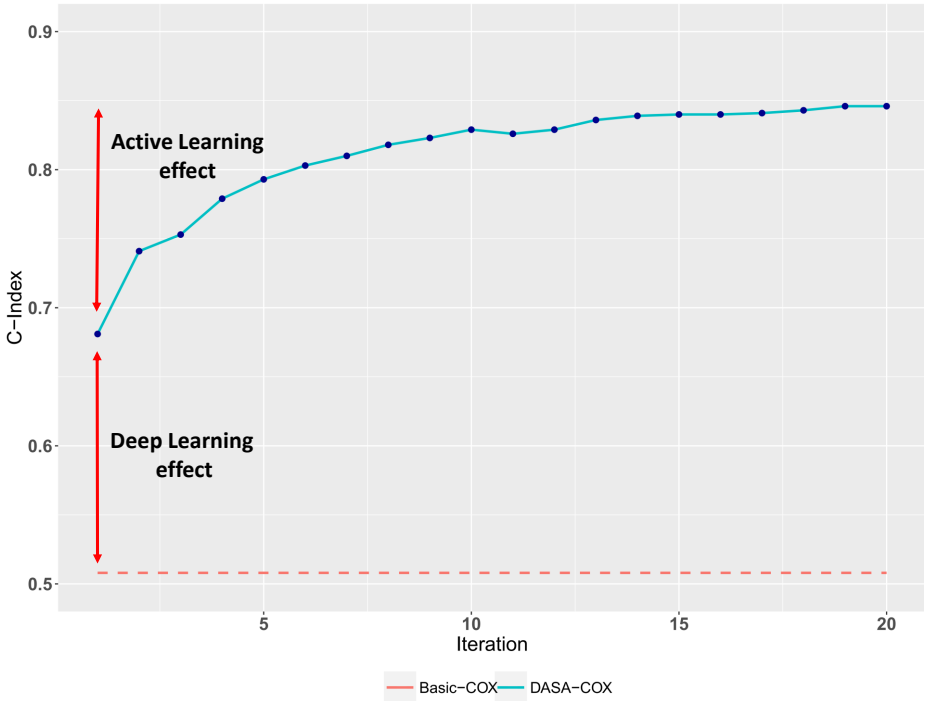
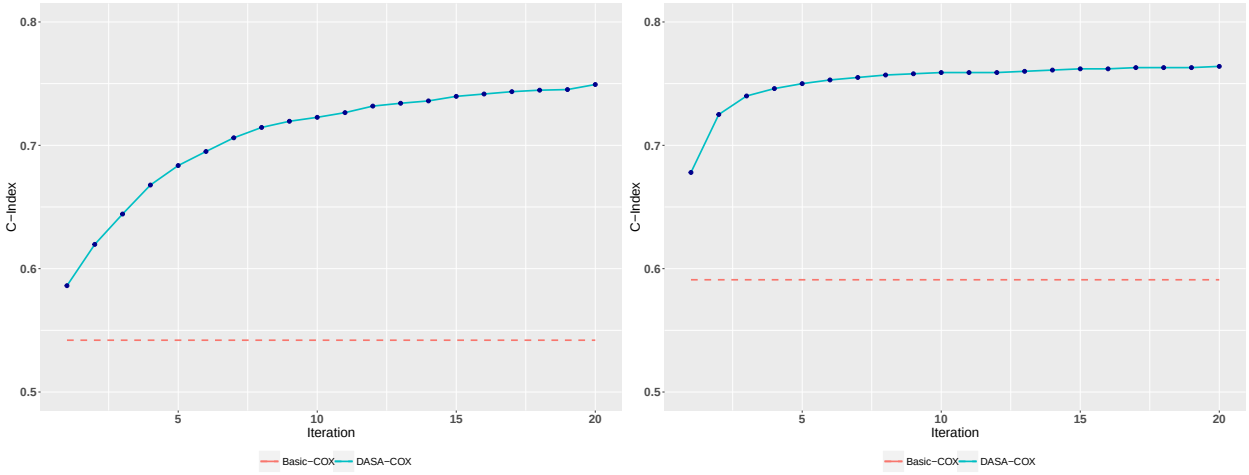


Figure 18: Performance of proposed approach in comparison with baseline (training size =25)

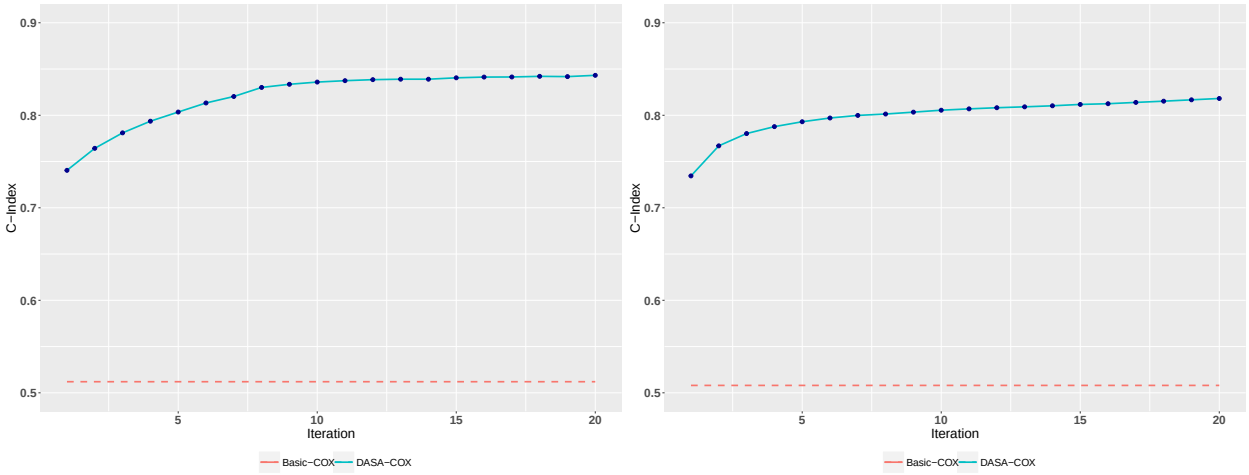
As demonstrated in Figure 18, our method (DASA-COX) improves the performance of Basic-COX significantly in terms of concordance index. This improvement caused by two effects: 1) Deep learning effect which improve the model performance by features representation using labeled and unlabeled instances, and 2) Active learning effect which increase the model performance by involving the best labeled censored instance from the pool set in training process across all iterations.

Figure 19 shows our approach performance for training size of 50 and 100 instances. Top panel belongs to African-American patients and bottom panel is related to white patients. It is clear DASA-COX outperforms baseline approach in all cases. The effect of deep learning in improving model performance is higher at the bottom panel which can be caused by larger amount of pool set related to white patients that provide better feature learning.



(a) African-American (50 instances)

(b) African-American (100 instances)



(c) White (50 instances)

(d) White (100 instances)

Figure 19: Performance of proposed approach in comparison with baseline for different training size

As mentioned before, our approach is flexible enough and can employ any survival analysis

model in its framework to improve the baseline. Hence, we perform Random Survival Forests (RFS) model as a well-know non-linear survival model along with CPH model and evaluate our approach across different training sizes. The results are shown in Table 13 and 14 for African-Americans and white patients respectively.

Table 13: Performance comparison (C-index) between DASA and baseline models (African-Americans)

Training Size	CPH	DASA-CPH	RSF	DASA-RSF
25 instances	55.2%	84.7%	16.3%	57.6%
50 instances	54.2%	74.9%	17.6%	54.5%
100 instances	59.1%	76.6%	21.4%	48.2%
200 instances	58.6%	72.6%	22.3%	47.9%

Table 14: Performance comparison (C-index) between DASA and baseline models (Whites)

Training Size	CPH	DASA-CPH	RSF	DASA-RSF
25 instances	52.4%	87.9%	13.3%	62.1%
50 instances	51.2%	84.4%	15.5%	58.3%
100 instances	50.8%	82.3%	15.7%	49.7%
200 instances	53.6%	77.1%	18.2%	46.4%

The results confirm that our method can improve the concordance index significantly for cox proportional hazard model and random survival forests in each datasets. According to above results, we can conclude that DASA leads to larger performance improvement in smaller training size caused by active learning effect.

In the second step, we demonstrate how our treatment recommendation approach works. we considered three well-known treatment options for prostate cancer: chemotherapy, radiotherapy

and surgery as three binary variables in our dataset. Our goal is to discover the importance of each therapy using DASA approach for each subgroup of patients (African-Americans and white patients). Since in the experimental study CPH illustrated a great performance, we performed survival analysis using CPH. We do feature representation by deep stacked autoencoder network with 150, 100 and 5 hidden unites in encoder, decoder and latent layers respectively. We used small sample size with 50 instances in training process. Before training process, we combined chemotherapy, radiotherapy and surgery variables (features of interest) to the represented features came from deep learning performed on other features in training instances combined with unlabeled pool set and then trained the cox survival model using active learning framework with 20 iterations over all features. The results for average exponential of coefficients (hazard ratios) over 10 runs shown in Table 15 for African-Americans and white patients:

Table 15: Average Hazard Ratio among different treatment plans

	Method	**Chemotherapy**	**Radiotherapy**	**Surgery**
African-Americans	COX-Base	1	1	1
	COX-DASA	0.74	1.04	1.38
White Patients	COX-Base	1	1	1
	COX-DASA	0.96	1.08	2.23

As shown above, traditional CPH model could not differentiate between treatment plans where their hazard ratios are one. Since the data is high-dimensional traditional CPH leads to zero coefficients for these three treatment variables. On the other side, our approach using Cox model can discover the risk associated to each treatment. Based on our results, surgery has the highest risk in the both subgroup of patients, radiotherapy is associate with a decline in the survival rate while chemotherapy increases the survival rate with lowest risk. It is obvious that the pattern of

hazard ratios among treatment plans are different between African-American and white patients. For example the risk related to surgery is significantly higher than the other two therapies in white patients (more than 2 times) while in the African-Americans the pattern is different.

This experimental treatment recommendation was a simple example to show how our method works. This approach is highly useful for comparing the risk associated with new treatment in comparison with current treatment plans where the labeled data is rare and expensive.

4.6 Discussion and Conclusion

In this chapter, we proposed a novel survival analysis framework using deep learning and active learning called Deep Active Survival Analysis (DASA). Our approach is able to improve the survival analysis performance significantly and provides treatment recommendations. DASA is highly applicable when the labeled data is scarce and high-dimensional. Our approach encompasses two main phases: 1) deep feature learning and 2) active learning process. We do feature representation using deep learning to produce robust features from high-dimensional, sparse and complex EHRs. We used the advantage of pool of unlabeled data (censored instances) to provide better representation of labeled instances from deep learning implementation. In the active learning process, we developed a new sampling strategy specifically for survival analysis which can be used for any survival analysis models such as Cox-based approaches and random survival forests.

In experimental study, we used SEER-Medicare data related to prostate cancer among African-Americans and white patients to demonstrate how our model can enhance the performance of survival analysis in comparison of traditional approach. Empirically we showed that deep learning has greater effect on survival performance improvement in the case that we have larger pool of unlabeled data and active learning effect is higher when we deal with smaller training sample size.

We apply our treatment recommendation approach to find hazard ratio of three common treatment plan (chemotherapy, radiotherapy and surgery) for prostate cancer based on Cox model. While traditional CPH model fails to find the hazard ratios among high dimensional data, our approach discovers them and provides some racial treatment insights.

In sum, our method leads to more accurate survival analysis for risk prediction, survival time estimation and treatment recommendation. Our approach is flexible enough to capture any survival analysis model and improve its performance. Our model can be applied on different areas especially in the case of testing and comparing the risk (impact) of new treatment (e.g. in healthcare) or new technology (e.g. in the manufacturing process) where the amount of labeled instances are small and labeling is expensive. For the future works, we will implement DASA on the other datasets and introduce some new sampling strategy with better performance.

CHAPTER 5 CONCLUSION AND FUTURE STEPS

5.1 Conclusion

In this dissertation, we introduced an integrated framework to develop data-driven approaches for different aspects of precision (personalized) healthcare. First we proposed a novel predictive approach using deep feature learning which can be applied in many domains as well as healthcare informatics. Second we introduced a new biclustering approach using convex optimization for patient subgroup analysis which can discover the groups of patient with similar risk factors. Our method has potential to use in cohort analysis and treatment planning. Finally in the last chapter, we developed a novel treatment recommender model using survival analysis, deep learning and active learning which has capable to improve the performance of traditional survival analysis models significantly and provides better interpretation for treatment recommendations. In each work, we provided some new insights based on our theoretical and empirical contributions which is summarized as following:

In predictive modeling using deep learning, We used unsupervised learning before supervised learning because the success of predictive machine learning algorithms highly depends on feature representation and extraction [81]. Since in several situation, data is sparse, noisy, high dimensional and repetitive, supervised learning and feature selection approaches cannot identify the pattern of data which makes them inappropriate for modeling the hierarchical and complex data. To overcome this shortcoming, unsupervised feature learning or representation learning attempts automatically to discover complexity and dependencies in the data to learn a compact and high-level representation which provides better features to extract useful information when applying classifiers and predictive models.

We demonstrated that deep learning could be effective for small datasets as well as large data and our comparative study between small and large clinical datasets provides some new insights in the choice of deep representation. We believe that our model with great EHRs feature learning has potential to be applied in different clinical and health informatics aspects including treatment planning, risk factor identification, personalized recommendation and survival analysis. Also, our proposed framework is highly useful for exploiting a large amount of unlabeled data in the feature learning (unsupervised learning) step to extract high level abstraction of features when the labeled data are limited and expensive.

In the second work (Biclustering approach), we have developed a novel supervised subgroup detection method called SUBIC based on convex optimization. We used the idea of convex biclustering approach [27] and proposed a new supervised biclutering approach which overcomes the limitation of previous works when we have a target variable. Biclustering methods in the literature do not exploit a target variable on subgroup detection and risk factor selection. As a result, the detected biclusters do not link to target variables of interest. Hence, it is unable to predict the target variable for future input variables. Clearly, the target variable such as LVMI provides a critical guidance for detection and selection of the meaningful biclusters (patient subgroups).

SUBIC is a predictive model that combines the strength of biclustering and tree-based methods. We introduced a new elastic-net penalty term in our model and defined two new weights in our objective function to enable the supervised training under the guidance of a clinically relevant target variable in detecting biclusters. The choice of elastic-net penalty term can overcome the lasso limitations. While the l_1-norm can generates a sparse model, the quadratic part of the penalty term encourages grouping effect and stabilizes the l_1-norm regularization path. Also the elastic-net regularization term performs very suitable for high dimensional data with correlated input

variables and would be a better model when $p \gg n$ specially in the case of gene expression data and precision medicine problems [126].

In the last chapter we discussed about two specific challenges existed in survival analysis from EHRs: 1) Clinical data are usually high dimensional, sparse and time-dependent where applying traditional survival approaches do not perform well enough to estimate the risk of a medical event of interest accurately, 2) In many health survival applications, labeled data (time-to-event instances) are small, time-consuming and expensive to collect. In this situation, it is hard to learn a survival model based on traditional approaches which able to predict the relative risk of patients precisely. To overcome these challenges, first, we proposed a novel survival analysis approach using deep learning and active learning termed DASA. Our model is capable to learn more accurate survival model using high dimensional and small size EHRs in comparison with some baseline survival models. Second, we introduced a treatment recommendation approach based on our survival analysis model which can compare the relative risk (or survival time) associate with different treatment plans and assign better one.

Based on our experimental study, empirically we showed that deep learning has greater effect on survival performance improvement in the case that we have larger pool of unlabeled data and active learning effect is higher when we deal with smaller training sample size. We showed that Our approach is flexible enough to capture any survival analysis model and improve its performance. We discussed that our model can be applied on different areas especially in the case of testing and comparing the risk (impact) of new treatment (e.g. in healthcare) or new technology (e.g. in the manufacturing process) where the amount of labeled instances are small and labeling is expensive.

5.2 Future Steps

In this research we focus to develop a data driven framework for precision/personalized medicine. As explained in the previous chapters, we developed three novel analytics approaches for predictive modeling, subgroup detection and survival analysis which can be applied for various precision medicine problems. Our results show that all methods have a competitive performance based on specific measures in comparison with baseline models. In spite of our significant contributions in each chapter, there exist several opportunities to extent each work as future extension. Remarkable current and future works have been described as follows:

1- Medical Data: In this study, we implemented our developed approaches on different medical data sets including cardiovascular disease data related to the subgroup of African-Americans, e-ICU collaborative research datasets and SEER-Medicare prostate cancer data. As a further direction, we can use different datasets in each method and evaluate their performance. For example using microarray (gene-expression) data and various cancer databases have high potential for our method's evaluation.

2- Predictive Modeling: In chapter 2, we presented a new predictive approach using deep learning. For future works, 1) we need to apply more deep architectures like as stacked denoising autoencoders and adversarial autoencoders for representation learning, 2) we can employ different machine learning tasks in the last step of our approach. For instance, we can apply clustering instead of supervised learning to discover treatment schemes among high-dimensional EHRs and finally 3) we can work on deep features interpretations. The key issue in using deep feature representation is difficulties in naturally interpretation of main features. Since many deep learning approaches use several hidden layers for multiple non-linear transformation on input features, they

often known as black boxes where only the input and output of framework is meaningful and there is lack of enough transparency in the model. Therefore, beside of our predictive approach, we can develop feature selection algorithms for interpreting of the main features through deep prediction.

3- Subgroup Analysis: In chapter 3, we developed a novel biclustering approach using convex optimization. We believe that our method is the first supervised biclustering approach which can take the benefit of target variable's guidance to find biclusters. The proposed prediction approach is a simple yet effective approach and needed to be improved in the future work. Also the similarity weights defined in our model play a key role in the final biclusters, hence there is a remarkable opportunity to redefine them based on linear and non-linear relationship between covariates and target variable. Finally, it is necessary to reformulate the proposed model for the case that we have multiple target variables or the response variable is categorical.

4- Survival analysis model with treatment recommendation: In the last chapter of this dissertation, we introduced a new survival analysis model for accurate risk prediction and treatment recommendation. There exist needs to improve the sampling strategy in active learning framework. On the other side we need to implement our approach using different deep architectures as well. Another direction for the future work can be transforming of the treatment recommendation approach to a personalized framework.

Figure 20 shows the summary of the current works and future works in this research:

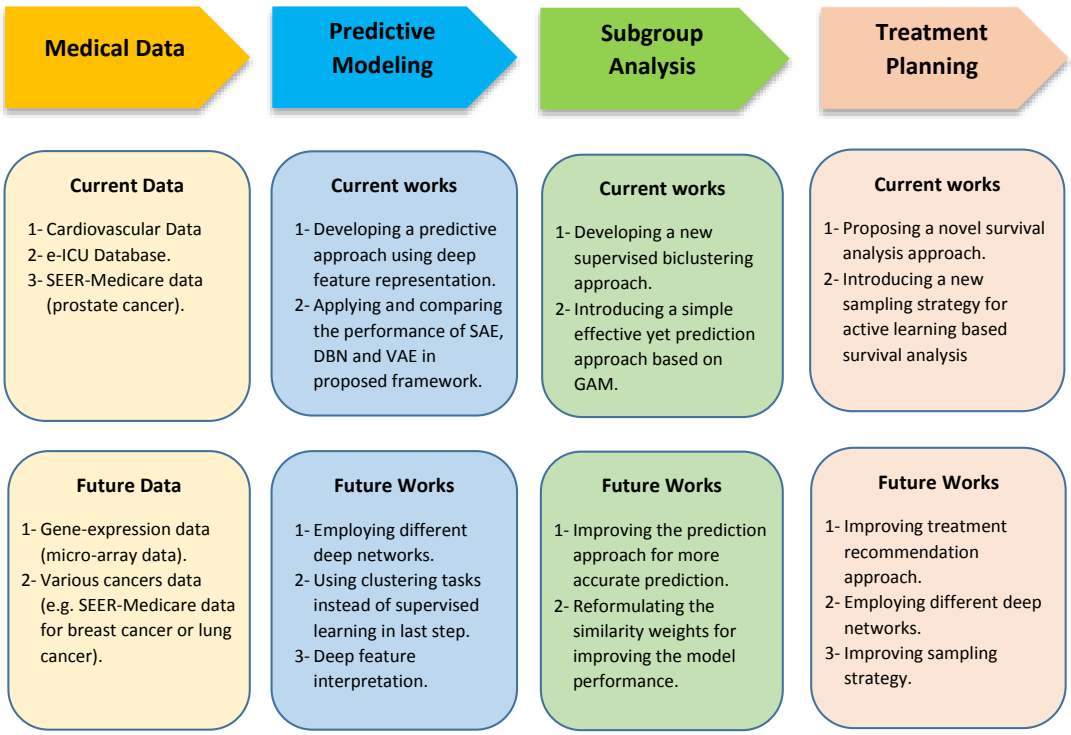


Figure 20: Current works and Future works in this research

5.3 Novelties and Contributions

According to what discussed in the chapter 1, 2, 3 and 4; This research develops and applies data-driven methods for healthcare informatics and precision medicine. In this path it contributes to a number of fields as follows:

1. Developing a new predictive model using deep learning for high dimensional data which can predict the target of interest better than baseline models.
2. Providing a predictive framework which is highly useful for exploiting a large amount of un-labeled medical records for extracting high level representation of labeled data for supervised learning tasks.
3. Providing new insights about choice of deep architectures for feature representation among small and large datasets.

4. Developing a novel subgroup detection approach using biclustering and convex optimization which can be applied on different precision medicine problem.
5. Developing a solution for proposed sparse supervised biclustering approach based on split bregman method.
6. Developing prediction approach based on supervised biclustering framework using generalized additive model.
7. Developing a new survival analysis framework using deep learning and active learning for patient risk prediction or survival time estimation.
8. Proposing a new sampling strategy in active learning framework for survival analysis based on model performance improvement which can select the most informative candidate from the unlabeled pool according to concordance index.
9. Introducing a new approach for discovering of treatment pattern among high-dimensional medical data based on our proposed survival approach.
10. Providing new insights about deep learning and active learning effects on proposed survival model performance based on small/large unlabeled pool set and size of training set respectively.
11. Discovering new medical insights by applying our proposed methods on specific precision medicine problems such as cardiovascular disease and prostate cancer.

REFERENCES

[1] Celestine Aguwa, Mohammad Hessam Olya, and Leslie Monplaisir, *Modeling of fuzzy-based voice of customer for business decision analytics*, Knowledge-Based Systems **125** (2017), 136–145.

[2] Fabrizio Angiulli, Eugenio Cesario, and Clara Pizzuti, *Random walk biclustering for microarray data*, Information Sciences **178** (2008), no. 6, 1479–1497.

[3] Dana Angluin, *Queries and concept learning*, Machine learning **2** (1988), no. 4, 319–342.

[4] Les E Atlas, David A Cohn, and Richard E Ladner, *Training connectionist networks with queries and selective sampling*, Advances in neural information processing systems, 1990, pp. 566–573.

[5] Wassim Ayadi, Mourad Elloumi, and Jin-Kao Hao, *Pattern-driven neighborhood search for biclustering of microarray data*, BMC bioinformatics **13** (2012), no. 7, S11.

[6] BK Beaulieu and CS Greene, *Semi-supervised learning of the electronic health record with denoising autoencoders for phenotype stratification. biorxiv*, 2016.

[7] Amir Ben-Dor, Benny Chor, Richard Karp, and Zohar Yakhini, *Discovering local structure in gene expression data: the order-preserving submatrix problem*, Journal of computational biology **10** (2003), no. 3-4, 373–384.

[8] Yoshua Bengio, Aaron Courville, and Pascal Vincent, *Representation learning: A review and new perspectives*, IEEE transactions on pattern analysis and machine intelligence **35** (2013), no. 8, 1798–1828.

[9] Yoshua Bengio et al., *Learning deep architectures for ai*, Foundations and trends® in Machine Learning **2** (2009), no. 1, 1–127.

[10] Yoshua Bengio, Pascal Lamblin, Dan Popovici, Hugo Larochelle, et al., *Greedy layer-wise training of deep networks*, Advances in neural information processing systems **19** (2007), 153.

[11] Sven Bergmann, Jan Ihmels, and Naama Barkai, *Iterative signature algorithm for the analysis of large-scale gene expression data*, Physical review E **67** (2003), no. 3, 031902.

[12] Eta S Berner, *Clinical decision support systems*, Springer, 2007.

[13] Stefan Bleuler, Amela Prelic, and Eckart Zitzler, *An ea framework for biclustering of gene expression data*, Evolutionary Computation, 2004. CEC2004. Congress on, vol. 1, IEEE, 2004, pp. 166–173.

[14] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Foundations and Trends® in Machine Learning **3** (2011), no. 1, 1–122.

[15] Leo Breiman, *Random forests*, Machine learning **45** (2001), no. 1, 5–32.

[16] Tom Brosch, Roger Tam, AlzheimerâĂŹs Disease Neuroimaging Initiative, et al., *Manifold learning of brain mris by deep learning*, International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2013, pp. 633–640.

[17] Kenneth Bryan, Pádraig Cunningham, and Nadia Bolshakova, *Application of simulated annealing to the biclustering of gene expression data*, IEEE transactions on information technology in biomedicine **10** (2006), no. 3, 519–525.

[18] Stanislav Busygin, Oleg Prokopyev, and Panos M Pardalos, *Biclustering in data mining*, Computers & Operations Research **35** (2008), no. 9, 2964–2987.

[19] Carlos Cano, L Adarve, J López, and Armando Blanco, *Possibilistic approach for biclustering microarray data*, Computers in biology and medicine **37** (2007), no. 10, 1426–1436.

[20] Pedro Carmona-Saez, Roberto D Pascual-Marqui, Francisco Tirado, Jose M Carazo, and Alberto Pascual-Montano, *Biclustering of gene expression data by non-smooth non-negative matrix factorization*, BMC bioinformatics **7** (2006), no. 1, 78.

[21] Kumardeep Chaudhary, Olivier B Poirion, Liangqun Lu, and Lana X Garmire, *Deep learning based multi-omics integration robustly predicts survival in liver cancer*, Clinical Cancer Research (2017), clincanres–0853.

[22] Nitesh V Chawla and Darcy A Davis, *Bringing big data to personalized healthcare: a patient-centered framework*, Journal of general internal medicine **28** (2013), no. 3, 660–665.

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

[24] Jie-Zhi Cheng, Dong Ni, Yi-Hong Chou, Jing Qin, Chui-Mei Tiu, Yeun-Chung Chang, Chiun-Sheng Huang, Dinggang Shen, and Chung-Ming Chen, *Computer-aided diagnosis with deep learning architecture: applications to breast lesions in us images and pulmonary nodules in ct scans*, Scientific reports **6** (2016), 24454.

[25] Yizong Cheng and George M Church, *Biclustering of expression data.*, Ismb, vol. 8, 2000, pp. 93–103.

[26] Yu Cheng, Fei Wang, Ping Zhang, and Jianying Hu, *Risk prediction with electronic health records: A deep learning approach*, Proceedings of the 2016 SIAM International Conference on Data Mining, SIAM, 2016, pp. 432–440.

[27] Eric C Chi, Genevera I Allen, and Richard G Baraniuk, *Convex biclustering*, Biometrics (2016).

[28] Edward Choi, Mohammad Taha Bahadori, Andy Schuetz, Walter F Stewart, and Jimeng Sun, *Doctor ai: Predicting clinical events via recurrent neural networks*, Machine Learning for Healthcare Conference, 2016, pp. 301–318.

[29] Edward Choi, Mohammad Taha Bahadori, Elizabeth Searles, Catherine Coffey, Michael Thompson, James Bost, Javier Tejedor-Sojo, and Jimeng Sun, *Multi-layer representation learning for medical concepts*, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2016, pp. 1495–1504.

[30] David R Cox, *Regression models and life-tables*, Breakthroughs in statistics, Springer, 1992, pp. 527–541.

[31] Li Deng, Dong Yu, et al., *Deep learning: methods and applications*, Foundations and Trends® in Signal Processing **7** (2014), no. 3–4, 197–387.

[32] Smitha Dharan and Achuthsankar S Nair, *Biclustering of gene expression data using reactive greedy randomized adaptive search procedure*, BMC bioinformatics **10** (2009), no. 1, S27.

[33] Federico Divina and Jesus S Aguilar-Ruiz, *Biclustering of expression data with evolutionary computation*, IEEE transactions on knowledge and data engineering **18** (2006), no. 5, 590–602.

[34] Carl Doersch, *Tutorial on variational autoencoders*, arXiv preprint arXiv:1606.05908 (2016).

[35] LL Doove, Elise Dusseldorp, Katrijn Van Deun, and Iven Van Mechelen, *A comparison of five recursive partitioning methods to find person subgroups involved in meaningful treatment–subgroup interactions*, Advances in Data Analysis and Classification **8** (2014), no. 4, 403–425.

[36] Elise Dusseldorp, Claudio Conversano, and Bart Jan Van Os, *Combining an additive and tree-based regression model simultaneously: Stima*, Journal of Computational and Graphical Statistics **19** (2010), no. 3, 514–530.

[37] Elise Dusseldorp and Iven Van Mechelen, *Qualitative interaction trees: a tool to identify qualitative treatment–subgroup interactions*, Statistics in medicine **33** (2014), no. 2, 219–237.

[38] Kemal Eren, Mehmet Deveci, Onur Küçüktunç, and Ümit V Çatalyürek, *A comparative analysis of biclustering algorithms for gene expression data*, Briefings in bioinformatics **14** (2013), no. 3, 279–292.

[39] Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun, *Dermatologist-level classification of skin cancer with deep neural networks*, Nature **542** (2017), no. 7639, 115–118.

[40] Rasool Fakoor, Faisal Ladhak, Azade Nazi, and Manfred Huber, *Using deep learning to enhance cancer diagnosis and classification*, Proceedings of the International Conference on Machine Learning, 2013.

[41] Ailin Fan et al., *New statistical methods for precision medicine: Variable selection for optimal dynamic treatment regimes and subgroup detection.*, (2016).

[42] Jared C Foster, Jeremy MG Taylor, and Stephen J Ruberg, *Subgroup identification from randomized clinical trial data*, Statistics in medicine **30** (2011), no. 24, 2867–2880.

[43] Saeed Z Gavidel and Jeremy L Rickli, *Quality assessment of used-products under uncertain age and usage conditions*, International Journal of Production Research **55** (2017), no. 23, 7153–7167.

[44] Gad Getz, Erel Levine, and Eytan Domany, *Coupled two-way clustering analysis of gene microarray data*, Proceedings of the National Academy of Sciences **97** (2000), no. 22, 12079–12084.

[45] Jiajun Gu and Jun S Liu, *Bayesian biclustering of gene expression data*, BMC genomics **9** (2008), no. 1, S4.

[46] Varun Gulshan, Lily Peng, Marc Coram, Martin C Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, Jorge Cuadros, et al., *Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs*, Jama **316** (2016), no. 22, 2402–2410.

[47] John A Hartigan, *Direct clustering of a data matrix*, Journal of the american statistical association **67** (1972), no. 337, 123–129.

[48] Trevor J Hastie and Robert J Tibshirani, *Generalized additive models*, vol. 43, CRC press, 1990.

[49] Patrick J Heagerty and Yingye Zheng, *Survival model predictive accuracy and roc curves*, Biometrics **61** (2005), no. 1, 92–105.

[50] Rui Henriques and Sara C Madeira, *Bicpam: Pattern-based biclustering for biomedical data analysis*, Algorithms for Molecular Biology **9** (2014), no. 1, 27.

[51] Geoffrey E Hinton, *Deep belief networks*, Scholarpedia **4** (2009), no. 5, 5947.

[52] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh, *A fast learning algorithm for deep belief nets*, Neural computation **18** (2006), no. 7, 1527–1554.

[53] Geoffrey E Hinton and Ruslan R Salakhutdinov, *Reducing the dimensionality of data with neural networks*, science **313** (2006), no. 5786, 504–507.

[54] David W Hosmer, Stanley Lemeshow, and Susanne May, *Applied survival analysis*, Wiley Blackwell, 2011.

[55] N Howlader, AM Noone, M Krapcho, J Garshell, N Neyman, SF Altekruse, et al., *Seer cancer statistics review (csr) 1975–2011. bethesda, md: National cancer institute; 2014*, 2014.

[56] Daniel Joseph Hsu, *Algorithms for active learning*, (2010).

[57] Jianying Hu, Adam Perer, and Fei Wang, *Data driven analytics for personalized healthcare*, Healthcare Information Management Systems, Springer, 2016, pp. 529–554.

[58] Chenglong Huang, Albert Zhang, and Guanghua Xiao, *Deep integrative analysis for survival prediction*, (2017).

[59] Lawrence Hubert and Phipps Arabie, *Comparing partitions*, Journal of classification **2** (1985), no. 1, 193–218.

[60] Jared L Katzman, Uri Shaham, Alexander Cloninger, Jonathan Bates, Tingting Jiang, and Yuval Kluger, *Deepsurv: personalized treatment recommender system using a cox proportional hazards deep neural network*, BMC Medical Research Methodology **18** (2018), no. 1, 24.

[61] Yuval Kluger, Ronen Basri, Joseph T Chang, and Mark Gerstein, *Spectral biclustering of microarray data: coclustering genes and conditions*, Genome research **13** (2003), no. 4, 703–716.

[62] Peter Langfelder, Bin Zhang, and Steve Horvath, *Defining clusters from a hierarchical cluster tree: the dynamic tree cut package for r*, Bioinformatics **24** (2008), no. 5, 719–720.

[63] Hugo Larochelle, Yoshua Bengio, Jérôme Louradour, and Pascal Lamblin, *Exploring strategies for training deep neural networks*, Journal of Machine Learning Research **10** (2009), no. Jan, 1–40.

[64] Deep Learning, *Computer science department*, Stanford University. http://ufldl. stanford. edu/tutorial **20** (2013), 21–22.

[65] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, *Deep learning*, Nature **521** (2015), no. 7553, 436–444.

[66] Changhee Lee, William R Zame, Jinsung Yoon, and Mihaela van der Schaar, *Deephit: A deep learning approach to survival analysis with competing risks*, (2018).

[67] David D Lewis and William A Gale, *A sequential algorithm for training text classifiers*, Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval, Springer-Verlag New York, Inc., 1994, pp. 3–12.

[68] Guojun Li, Qin Ma, Haibao Tang, Andrew H Paterson, and Ying Xu, *Qubic: a qualitative biclustering algorithm for analyses of gene expression data*, Nucleic acids research (2009), gkp491.

[69] Xiangrui Li, Dongxiao Zhu, Ming Dong, Milad Zafar Nezhad, Alexander Janke, and Phillip Levy, *Sdt: A tree method for detecting patient subgroups with personalized risk factors*, Submmit on Clinical Research Informatics, AMIA Conference, March 2017.

[70] Xiangrui Li, Dongxiao Zhu, and Phillip Levy, *Predictive deep network with leveraging clinical measure as auxiliary task*, Bioinformatics and Biomedicine (BIBM), 2017 IEEE International Conference on, IEEE, 2017.

[71] Yifeng Li, Chih-Yu Chen, and Wyeth W Wasserman, *Deep feature selection: theory and application to identify enhancers and promoters*, Journal of Computational Biology **23** (2016), no. 5, 322–336.

[72] Linxia Liao and Hyung-il Ahn, *Combining deep learning and survival analysis for asset health management*, International Journal of Prognostics and Health Management (2016).

[73] Ilya Lipkovich, Alex Dmitrienko, Jonathan Denne, and Gregory Enas, *Subgroup identification based on differential effect searchâĂŤa recursive partitioning method for establishing response to treatment in patient subpopulations*, Statistics in medicine **30** (2011), no. 21, 2601–2621.

[74] Peng Liu, Hui Zhang, and Kie B Eom, *Active deep learning for classification of hyperspectral images*, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **10** (2017), no. 2, 712–724.

[75] Siqi Liu, Sidong Liu, Weidong Cai, Sonia Pujol, Ron Kikinis, and Dagan Feng, *Early diagnosis of*

alzheimer's disease with deep learning, Biomedical Imaging (ISBI), 2014 IEEE 11th International Symposium on, IEEE, 2014, pp. 1015–1018.

[76] Xiaowen Liu and Lusheng Wang, *Computing the maximum similarity bi-clusters of gene expression data*, Bioinformatics **23** (2007), no. 1, 50–56.

[77] Margaux Luck, Tristan Sylvain, Héloïse Cardinal, Andrea Lodi, and Yoshua Bengio, *Deep learning for patient-specific kidney graft survival analysis*, arXiv preprint arXiv:1705.10245 (2017).

[78] James Lyons, Abdollah Dehzangi, Rhys Heffernan, Alok Sharma, Kuldip Paliwal, Abdul Sattar, Yaoqi Zhou, and Yuedong Yang, *Predicting backbone c angles and dihedrals from protein sequences by stacked sparse auto-encoder deep neural network*, Journal of computational chemistry **35** (2014), no. 28, 2040–2046.

[79] Sara C Madeira and Arlindo L Oliveira, *Biclustering algorithms for biological data analysis: a survey*, IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) **1** (2004), no. 1, 24–45.

[80] Polina Mamoshina, Armando Vieira, Evgeny Putin, and Alex Zhavoronkov, *Applications of deep learning in biomedicine*, Molecular pharmaceutics **13** (2016), no. 5, 1445–1454.

[81] Riccardo Miotto, Li Li, Brian A Kidd, and Joel T Dudley, *Deep patient: An unsupervised representation to predict the future of patients from the electronic health records*, Scientific reports **6** (2016), 26094.

[82] Riccardo Miotto, Fei Wang, Shuang Wang, Xiaoqian Jiang, and Joel T Dudley, *Deep learning for healthcare: review, opportunities and challenges*, Briefings in Bioinformatics (2017), bbx044.

[83] Sushmita Mitra and Haider Banka, *Multi-objective evolutionary biclustering of gene expression data*, Pattern Recognition **39** (2006), no. 12, 2464–2477.

[84] Majid Moradi-Aliabadi and Yinlun Huang, *Decision support for enhancement of manufacturing sustainability: A hierarchical control approach*, ACS Sustainable Chemistry & Engineering **6** (2018), no. 4, 4809–4820.

[85] Milad Zafar Nezhad, Dongxiao Zhu, Xiangrui Li, Kai Yang, and Phillip Levy, *Safs: A deep feature selection approach for precision medicine*, Bioinformatics and Biomedicine (BIBM), 2016 IEEE International Conference on, IEEE, 2016, pp. 501–506.

[86] Milad Zafar Nezhad, Dongxiao Zhu, Najibesadat Sadati, and Kai Yang, *A predictive approach using deep feature learning for electronic medical records: A comparative study*, arXiv preprint arXiv:1801.02961 (2018).

[87] Milad Zafar Nezhad, Dongxiao Zhu, Najibesadat Sadati, Kai Yang, and Phillip Levy, *Subic: A supervised bi-clustering approach for precision medicine*, Machine Learning and Applications (ICMLA), 2017 16th IEEE International Conference on, IEEE, 2017, pp. 755–760.

[88] Phuoc Nguyen, Truyen Tran, Nilmini Wickramasinghe, and Svetha Venkatesh, *Deepr: A convolutional net for medical records*, IEEE journal of biomedical and health informatics **21** (2017), no. 1, 22–30.

[89] Dong Nie, Han Zhang, Ehsan Adeli, Luyan Liu, and Dinggang Shen, *3d deep learning for multi-modal imaging-guided survival time prediction of brain tumor patients*, International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2016, pp. 212–220.

[90] Jeffrey Pennington, Richard Socher, and Christopher Manning, *Glove: Global vectors for word representation*, Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1532–1543.

[91] Beatriz Pontes, Raúl Giráldez, and Jesús S Aguilar-Ruiz, *Biclustering on expression data: A review*, Journal of biomedical informatics **57** (2015), 163–180.

[92] Adhish Prasoon, Kersten Petersen, Christian Igel, François Lauze, Erik Dam, and Mads Nielsen, *Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network*, International conference on medical image computing and computer-assisted intervention, Springer, 2013, pp. 246–253.

[93] William M Rand, *Objective criteria for the evaluation of clustering methods*, Journal of the American Statistical association **66** (1971), no. 336, 846–850.

[94] Rajesh Ranganath, Adler Perotte, Noémie Elhadad, and David Blei, *Deep survival analysis*, arXiv preprint arXiv:1608.02158 (2016).

[95] Daniele Ravi, Charence Wong, Fani Deligianni, Melissa Berthelot, Javier Andreu-Perez, Benny Lo, and Guang-Zhong Yang, *Deep learning for health informatics*, IEEE journal of biomedical and health informatics **21** (2017), no. 1, 4–21.

[96] Narges Razavian, Jake Marcus, and David Sontag, *Multi-task prediction of disease onsets from longitudinal laboratory tests*, Machine Learning for Healthcare Conference, 2016, pp. 73–100.

[97] W Ken Redekop and Deirdre Mladsi, *The faces of personalized medicine: a framework for understanding its meaning and scope*, Value in Health **16** (2013), no. 6, S4–S9.

[98] Swarup Roy, Dhruba K Bhattacharyya, and Jugal K Kalita, *Cobi: pattern based co-regulated biclustering of gene expression data*, Pattern Recognition Letters **34** (2013), no. 14, 1669–1678.

[99] Najibesadat Sadati, Ratna Babu Chinnam, and Milad Zafar Nezhad, *Observational data-driven modeling and optimization of manufacturing processes*, Expert Systems with Applications **93** (2018), 456–464.

[100] Eran Segal, Ben Taskar, Audrey Gasch, Nir Friedman, and Daphne Koller, *Rich probabilistic models for gene expression*, Bioinformatics **17** (2001), no. suppl 1, S243–S252.

[101] Ozan Sener and Silvio Savarese, *A geometric approach to active learning for convolutional neural networks*, arXiv preprint arXiv:1708.00489 (2017).

[102] Burr Settles, *Active learning literature survey*, University of Wisconsin, Madison **52** (2010), no. 55-66, 11.

[103] Qizheng Sheng, Yves Moreau, and Bart De Moor, *Biclustering microarray data by gibbs sampling*, Bioinformatics **19** (2003), no. suppl 2, ii196–ii205.

[104] Benjamin Shickel, Patrick Tighe, Azra Bihorac, and Parisa Rashidi, *Deep ehr: A survey of recent advances on deep learning techniques for electronic health record (ehr) analysis*, arXiv preprint arXiv:1706.03446 (2017).

[105] Rebecca L Siegel, Kimberly D Miller, and Ahmedin Jemal, *Cancer statistics, 2015*, CA: a cancer journal for clinicians **65** (2015), no. 1, 5–29.

[106] Mark W Stanton, *Expanding patient-centered care to empower patients and assist providers*, 2002.

[107] Xiaogang Su, Tianni Zhou, Xin Yan, Juanjuan Fan, and Song Yang, *The international journal of biostatistics*, (2008).

[108] Gayle Swenson, *President's council of advisors on science and technology*, (2015).

[109] Kean Ming Tan and Daniela M Witten, *Sparse biclustering of transposable data*, Journal of Computational and Graphical Statistics **23** (2014), no. 4, 985–1008.

[110] Amos Tanay, Roded Sharan, and Ron Shamir, *Discovering statistically significant biclusters in gene expression data*, Bioinformatics **18** (2002), no. suppl 1, S136–S144.

[111] Chun Tang and Aidong Zhang, *Interrelated two-way clustering and its application on gene expression data*, International Journal on Artificial Intelligence Tools **14** (2005), no. 04, 577–597.

[112] Robert Tibshirani, *Regression shrinkage and selection via the lasso*, Journal of the Royal Statistical Society. Series B (Methodological) (1996), 267–288.

[113] Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight, *Sparsity and smoothness via the fused lasso*, Journal of the Royal Statistical Society: Series B (Statistical Methodology) **67** (2005), no. 1, 91–108.

[114] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol, *Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion*, Journal of Machine Learning Research **11** (2010), no. Dec, 3371–3408.

[115] Bhanukiran Vinzamuri, Yan Li, and Chandan K Reddy, *Active learning based survival regression for censored data*, Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, ACM, 2014, pp. 241–250.

[116] Ping Wang, Yan Li, and Chandan K Reddy, *Machine learning for survival analysis: A survey*, arXiv preprint arXiv:1708.04649 (2017).

[117] Jiong Yang, Haixun Wang, Wei Wang, and Philip S Yu, *An improved biclustering method for analyzing gene expression profiles*, International Journal on Artificial Intelligence Tools **14** (2005), no. 05, 771–789.

[118] Wen-Hui Yang, Dao-Qing Dai, and Hong Yan, *Finding correlated biclusters from gene expression data*, IEEE Transactions on Knowledge and Data Engineering **23** (2011), no. 4, 568–584.

[119] Gui-Bo Ye and Xiaohui Xie, *Split bregman method for large scale fused lasso*, Computational Statistics & Data Analysis **55** (2011), no. 4, 1552–1569.

[120] Kevin Y Yip, David W Cheung, and Michael K Ng, *Harp: A practical projected clustering algorithm*, IEEE Transactions on knowledge and data engineering **16** (2004), no. 11, 1387–1397.

[121] Jinsung Yoon, Camelia Davtyan, and Mihaela van der Schaar, *Discovery and clinical decision support for personalized healthcare*, IEEE journal of biomedical and health informatics (2016).

[122] Achim Zeileis, Torsten Hothorn, and Kurt Hornik, *Model-based recursive partitioning*, Journal of Computational and Graphical Statistics **17** (2008), no. 2, 492–514.

[123] Lizhuang Zhao and Mohammed J Zaki, *Microcluster: efficient deterministic biclustering of microarray data*, IEEE Intelligent Systems **20** (2005), no. 6, 40–49.

[124] Jian Zhou and Troyanskaya Olga, *Predicting effects of noncoding variants with deep learning–based sequence model*, Nature methods **12** (2015), no. 10, 931.

[125] Shusen Zhou, Qingcai Chen, and Xiaolong Wang, *Active deep learning method for semi-supervised sentiment classification*, Neurocomputing **120** (2013), 536–546.

[126] Hui Zou and Trevor Hastie, *Regularization and variable selection via the elastic net*, Journal of the Royal Statistical Society: Series B (Statistical Methodology) **67** (2005), no. 2, 301–320.

ABSTRACT

DATA-DRIVEN MODELING FOR DECISION SUPPORT SYSTEMS AND TREATMENT MANAGEMENT IN PERSONALIZED HEALTHCARE

by

MILAD ZAFAR NEZHAD

August 2018

Advisor: Dr. Kai Yang

Major: Industrial Engineering

Degree: Doctor of Philosophy

Massive amount of electronic medical records (EMRs) accumulating from patients and populations motivates clinicians and data scientists to collaborate for the advanced analytics to create essential knowledge for providing personalized insights. Learning from large and complicated data is using extensively in marketing and commercial enterprises to generate personalized recommendations. Recently the medical research community focuses to take the benefits of big data analytic approaches and moves to personalized (precision) medicine. So, it is a significant period in healthcare and medicine for transferring to a new paradigm. There is a noticeable opportunity to implement a data-driven healthcare system to make better medical decisions, better personalized predictions; and more precise discovering of risk factors and their interactions. In this research we focus on data-driven approaches for personalized healthcare. We propose a research framework which emphasizes on three main phases: 1) Predictive modeling, 2) Patient subgroup analysis and 3) Treatment recommendation. Our goal is to develop novel methods for each phase and apply them in real-world applications.

In the first phase, we develop a new predictive approach based on feature representation using deep feature learning and word embedding techniques. Our method uses different deep architectures (Stacked autoencoders, Deep belief network and Variational autoencoders) for feature representation in higher-level abstractions to obtain effective and more robust features from EMRs,

and then builds prediction models on the top of them. Our approach is particularly useful when the unlabeled data is abundant whereas labeled one is scarce. We investigate the performance of representation learning through a supervised approach. We perform our method on different small and large datasets. Finally we provide a comparative study and show that our predictive approach leads to better results in comparison with others.

In the second phase, we propose a novel patient subgroup detection method, called Supervised Biclustring (SUBIC) using convex optimization and apply our approach to detect patient subgroups and prioritize risk factors for hypertension (HTN) in a vulnerable demographic subgroup (African-Americans). Our approach not only finds patient subgroups with guidance of a clinically relevant target variable but also identifies and prioritizes risk factors by pursuing sparsity of the input variables and encouraging similarity among the input variables and between the input and target variables. Also, we introduce a predictive approach based on generalized additive model (GAM) to predict the target variable based on supervised biclustering framework.

Finally, in the third phase, we introduce a new survival analysis framework using deep learning and active learning with a novel sampling strategy. First, our approach provides better representation with lower dimensions from clinical features using labeled (time-to-event) and unlabeled (censored) instances and then actively trains the survival model by labeling the censored data using an oracle. As a clinical assistive tool, we propose a simple yet effective treatment recommendation approach based on our survival model. In the experimental study, we apply our approach on SEER-Medicare data related to prostate cancer among African-Americans and white patients. The results indicate that our approach outperforms significantly than baseline models.

The insights and results provided in each step of this study could be applied by data scientists, medical researchers and health policy makers to carry out precision medicine and personalized healthcare in different diseases.

AUTOBIOGRAPHICAL STATEMENT

Milad Zafar Nezhad is a Ph.D student of Industrial and Systems Engineering at Wayne State University, Detroit, Michigan. He received his second Master's degree in Computer Science at the same University in 2017 and got his first Master's degree in the Industrial Engineering at AmirKabir University of Technology, Tehran, Iran in 2011. He has a bachelors' degree in Industrial Engineering from Sharif University of Technology, Tehran, Iran. His major research interests include Data Analytics, Machine Learning, Helathcare Informatics and Operation Management. His papers have been published in top conferences and journals and received different prestiges awards.