2017

Centralized and distributed learning methods for predictive health analytics

BOSTON UNIVERSITY

COLLEGE OF ENGINEERING

Dissertation

CENTRALIZED AND DISTRIBUTED LEARNING METHODS FOR PREDICTIVE HEALTH ANALYTICS

by

THEODORA S. BRISIMI

Diploma, National Technical University of Athens, 2011

Submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

2017

Approved by

First Reader

Ioannis Ch. Paschalidis, PhD
Professor of Electrical and Computer Engineering
Professor of Systems Engineering
Professor of Biomedical Engineering

Second Reader

Christos G. Cassandras, PhD
Professor of Electrical and Computer Engineering
Professor and Head of Systems Engineering

Third Reader

Prakash Ishwar, PhD
Professor of Electrical and Computer Engineering
Professor of Systems Engineering

Fourth Reader

Evimaria Terzi, PhD
Associate Professor of Computer Science

Acknowledgments

I am indebted to a number of people, whose support and advice over the course of my PhD studies have been invaluable to the success of my efforts.

First and foremost, I would like to thank my advisor Ioannis (Yannis) Paschalidis for his constant encouragement and informed guidance. Yannis through his advising style has given me the space to grow, develop my ideas and analytical skills, while at the same time he helped me open up my perspective on how to conduct research and enhance my academic skills. His suggestions have always served as a source of motivation for me to aim for novelty and excellence. Working with him has been a great pleasure and a significant influence to my life after the PhD.

I am grateful to Christos Cassandras, who was a member of my committee and followed my thesis work at all stages. I also had the chance to collaborate extensively with him on a research project that is not included in the current thesis. In this context, I have benefited a lot from his constructive feedback and guidance. Moreover, I am deeply thankful to Prakash Ishwar and Evimaria Terzi, also members of my thesis committee, whose excellent classes have considerably built up my knowledge in the research area. They both guided me through asking the right research questions during discussions and helped me advance my research.

I would like to acknowledge the funding agency: the material in this dissertation is based upon work supported by the National Science Foundation (NSF). Moreover, I would also like to acknowledge with gratitude the financial support of the Alexander S. Onassis Public Benefit Foundation during most of the period of my studies and research at Boston University.

Special thanks are due to Wuyang Dai and Wilbur (Wei) Shi, two close collaborators. Together with Wuyang, we set the basis of the work presented in this thesis, while with Wei we worked on "completing the story" by developing the distributed

setting. I am grateful for our everyday discussions, from which I kept learning. During the process of developing the ideas presented in this thesis, I have also collaborated with excellent colleagues and published together papers: William (Bill) Adams, James Seong-Cheol Kang, Venkatesh Saligrama, Alex Olshevsky, Tingting Xu, Taiyao Wang, to all of whom I am grateful.

The encouragement and support of my colleagues and friends has been tremendous throughout all the PhD years. I would like the thank in particular the "quals gang": Jenny B., John G., Michael S.F., Morteza H., and Wenbo H., my ever first office mate Fuzhuo H., my good "neighbors": Amanda G.B., Berkin C., Eran S., Liangxiao X. and Limor M. and our "Boston tea party": Henghui Z. and Ruidi C. for our fun times at BU and outside BU. Being part of the ISS lab has been a great experience. I wish to thank my BU friends who have offered me their invaluable friendship but also advice on research issues: Ahmet T., Andrew C., Aydan U., Cem A., Emir K., Jing W., Jing Z., Jonathan W., Qi Z., Tolga B., as well as friends outside BU that have supported me and shared all my good and bad PhD moments: Alex K., Bertan C., Christina P., Constantina C., Dionysia M., Katerina P., Kelly P., Manolis Z., Maria T., Marieta P., Vasilis S. and Yannis M..

I would like to express my sincere and deep love to my sister, Vasiliki, for her endless love and support throughout this journey. There has not been a single day that she was not there to help me in any way I needed.

Last, my deepest gratitude together with my love and appreciation goes to my wonderful parents, Sophocles and Dimitra. Their unconditional love, their lifelong commitment to education and their continuous encouragement in every stage of my life have all been reasons without which I wouldn't have been where I am today.

This thesis is dedicated to my family.

CENTRALIZED AND DISTRIBUTED LEARNING METHODS FOR PREDICTIVE HEALTH ANALYTICS

THEODORA S. BRISIMI

Boston University, College of Engineering, 2017

Major Professor: Ioannis Ch. Paschalidis, PhD
Professor of Electrical and Computer Engineering
Professor of Systems Engineering
Professor of Biomedical Engineering

ABSTRACT

The U.S. health care system is considered costly and highly inefficient, devoting substantial resources to the treatment of acute conditions in a hospital setting rather than focusing on prevention and keeping patients out of the hospital. The potential for cost savings is large; in the U.S. more than $30 billion are spent each year on hospitalizations deemed preventable, 31% of which is attributed to heart diseases and 20% to diabetes. Motivated by this, our work focuses on developing centralized and distributed learning methods to predict future heart- or diabetes- related hospitalizations based on patient Electronic Health Records (EHRs).

We explore a variety of supervised classification methods and we present a novel likelihood ratio based method (K-LRT) that predicts hospitalizations and offers interpretability by identifying the K most significant features that lead to a positive prediction for each patient. Next, assuming that the positive class consists of multiple clusters (hospitalized patients due to different reasons), while the negative class is drawn from a single cluster (non-hospitalized patients healthy in every aspect), we present an alternating optimization approach, which jointly discovers the clusters in

the positive class and optimizes the classifiers that separate each positive cluster from the negative samples. We establish the convergence of the method and characterize its VC dimension. Last, we develop a decentralized cluster Primal-Dual Splitting (cPDS) method for large-scale problems, that is computationally efficient and privacy-aware. Such a distributed learning scheme is relevant for multi-institutional collaborations or peer-to-peer applications, allowing the agents to collaborate, while keeping every participant's data private. cPDS is proved to have an improved convergence rate compared to existing centralized and decentralized methods. We test all methods on real EHR data from the Boston Medical Center and compare results in terms of prediction accuracy and interpretability.

Contents

List of Tables

List of Figures

List of Abbreviations

ACC	.....	Alternating Clustering and Classification
AMI	.....	Acute Myocardial Infarction
AUC	.....	Area Under the ROC Curve
BMC	.....	Boston Medical Center
CART	.....	Classification And Regression Trees
CDSS	.....	Clinical Decision Support System
CPK	.....	Creatine PhosphoKinase
CPT	.....	Current Procedural Terminology
CRP	.....	C-Reactive Protein
CT-LSVM	.....	Cluster Then- Linear Support Vector Machines
CT-SLSVM	.....	Cluster Then- Sparse Linear Support Vector
	.....	Machines
DADMM	.....	Distributed Alternating Direction Method of
	.....	Multipliers
DBP	.....	Diastolic Blood Pressure
ECG	.....	Electrocardiography
EHR	.....	Electronic Health Record
EMR	.....	Electronic Medical Record
ER	.....	Emergency Room
EXTRA	.....	EXact first order Algorithm
FHS	.....	Framingham Heart Study
FRF	.....	Framingham Risk Factors
GDP	.....	Gross Domestic Product
GIC	.....	Group Insurance Commission
HDL	.....	High-Density Lipoprotein
HIPAA	.....	Health Insurance Portability and Accountability
	.....	Act of 1996
ICD9	.....	International Classification of Diseases- Ninth
	.....	Revision
IS	.....	Importance Score
IncrSub	.....	Incremental Subgradient
KKT	.....	Karush-Kuhn-Tucker
LAC	.....	Linear time Average Consensus

LDL	Low-Density Lipoprotein
LRT	Likelihood Ratio Test
MDDSS	Medical Diagnostic Decision Support System
ML	Machine Learning
MRI	Magnetic Resonance Imaging
PHI	Personal Health Identifiers/Information
PHR	Personal Health Record
RBF	Radial Basis Function
ROC	Receiver Operating Characteristic
SBP	Systolic Blood Pressure
SLSVM	Sparse Linear Support Vector Machines
SPDC	Stochastic Primal Dual Coordinate
SVM	Support Vector Machines
SubGD	Subgradient Descent
VC	Vapnik-Chervonenkis
cPDS	cluster Primal Dual Splitting
sSVM	sparse Support Vector Machines

Chapter 1

Introduction

1.1 Machine Learning in Healthcare

1.1.1 The Significance: An Underlying Motive

Healthcare has adapted to the recent advancements in machine learning and the unprecedented explosion of data. Mainly economic, if not other quality-of-service related, reasons play a key role in the transformation that healthcare is undergoing. In 2006, health-care expenses represented 15.5% of the United States (U.S.) gross domestic product (GDP) [Jiang et al., 2009], while in 2013, this percentage was increased to 17.6% [Groves et al., 2013], that is nearly $600 billion more than what was expected of a country with the U.S.'s size and wealth.[1] The World Health Organization estimates that healthcare costs will grow to 20% of U.S.'s GDP (nearly $5 trillion) by 2021 [Adler and Hoagland, 2012], especially with civilization diseases (or else called lifestyle diseases), like diabetes, coronary heart disease and obesity, growing. At the same time, machine learning (ML), whose aim is to develop algorithmic systems that can learn from data, improve through experience and be used for predictions, provides the tools needed to address a variety of diagnostic and prognostic medical problems.

[1] This estimate is based on McKinsey's regression analysis of income and spending data from other countries in the Organization of Economic Co-operation and Development. This measure estimates how much a country is expected to spend on healthcare based on per capita GDP [Groves et al., 2013].

1.1.2 Healthcare Big Data

Traditionally, healthcare data have been static, paper-based and used for record keeping and information sharing. The digitalization of patients' records started more than two decades ago. Widespread adoption of electronic health records (EHRs) (else called electronic medical records EMRs, or personal health records PHRs), has generated massive data sets. 87% of US office-based physicians were using them by the end of 2015, up from 42% in 2008 [Office of the National Coordinator for Health Information Technology, 2016]. EHRs have found diverse uses [Ludwick and Doucette, 2009], e.g., in assisting the quality management in hospitals [Takeda et al., 2003], in detecting adverse drug reactions [Hannan, 1999], and in general primary care [Wang et al., 2003]. These early application uses of EHRs merely scratch the surface of what may be possible.

Healthcare data nowadays, are being collected and stored massively and cheaply in data warehouses. They come from various sources and can be of different types: insurance claims and cost data, research and development data developed over the years and aggregated into medical databases by pharmaceutical companies, electronic health records, genomic sequences, human genetics, population data, clinical trial data, patients' behavior data etc. Most recent forms include biometric sensor readings, 3D imaging and more. EHRs form a rich set that might include demographics, hospital admissions, the patient's admitting, primary and secondary diagnoses, procedures, the physician's name, the hospital's name, nurse and doctor's notes, treatment reimbursement codes, discharge records, MRI (magnetic resonance imaging) and other images, ECGs (electrocardiograms) etc. EHRs also contain quantitative data (e.g., laboratory values, blood pressure measurements), qualitative data (e.g., text-based notes, demographic information) and transactional data (e.g., records of drug deliveries) [Murdoch and Detsky, 2013].

In 2001, D. Laney introduced "the three V" dimensions along which big data are expanding [Laney, 2001]: *volume*, *velocity* and *variety*. In [Raghupathi and Raghupathi, 2014], it is reported that the U.S. healthcare system data reached, in 2011, 150 exabytes and that at this rate of growth they will soon be on the order of zettabytes or yottabytes. Two perhaps even greater challenges than volume, are the variety of healthcare data, that come from diverse sources, all of which need to be utilized in order to fully leverage the potential of analytics, and the velocity. Velocity refers to the fact that healthcare data do not come only in static formats anymore. Nowadays real-time measurements are available by regular monitoring, e.g., diabetic glucose measurements, blood pressure readings, ECGs, data from operating room monitoring or new data streams, such as heart rate measurements, from fitness trackers. Oftentimes, critical outcomes, such as life or death, depend on mining those real-time streams. For example, in [Convertino et al., 2011] monitoring noninvasively measured hemodynamic signals detects early indicators of blood volume loss and impending circulatory failure in conscious, healthy humans who experience reduced central blood volume. In [Jin et al., 2009] real-time electrocardiographic monitoring can predict the cardiovascular disease, since heart rhythm irregularities cannot always be detected on a standard resting ECG machine.

Since 2011, a fourth dimension has been added to the description of big data and that is *veracity*. [Meyfroidt et al., 2009] discuss the problem of data accuracy and how it is hard to be measured in retrospect. Accuracy is understood as the ability of collected data to properly describe the clinical continuum during the time they were collected. It has two aspects to it: completeness, referring to how much data were actually collected and how much are missing, and correctness, which is capturing both whether the code or amount recorded in the system has the correct value and whether the diagnoses or prescriptions recorded capture what has truly happened.

Most recently, people consider *value* as the fifth "V" big data dimension, which refers to the business potential of achieving greater value through insights from superior analytics. [Chen et al., 2012] highlight the increasing demand for individuals that know how to manage the three perspectives of business decision making: descriptive, predictive and prescriptive analytics.

1.1.3 How Can Big Data Benefit the Medical Domain?

The big data transformation of healthcare can potentially have a positive impact on all key components of the system, i.e., provider, payer, patient and management. Essentially, the benefits originate from being able to provide the most suitable intervention at the most appropriate time for each individual patient. [Murdoch and Detsky, 2013] present examples of illustrative beneficial directions.

First, big data have a great potential on generating new knowledge and advancing medical research: (i) clinical data could be used for discovering phenotypes and treatment of patients; (ii) EHRs could potentially establish new patient-stratification techniques and reveal unknown disease correlations; and (iii) integrating EHRs with genetic data could provide insights into the genotype /phenotype relationships [Jensen et al., 2012].

Second, big data may support knowledge dissemination in clinical care. Standard medical practice is gradually becoming more evidence-based shifting away from subjective decision making. However, most physicians struggle to stay updated with the latest evidence that is guiding clinical practice. The digitalized format of medical literature articles facilitates access to knowledge, however the large number of available information hinders it. A medical diagnostic decision support system (MDDSS or else called clinical decision support system CDSS) that analyzes real-time data and provides recommendations could not only be a helpful tool to physicians, but reduce costs and contribute to standardization of care as well.

Third, big data slowly shift healthcare towards a model in which the patient is empowered with a more active role. Patients will no longer always have to visit the doctor and passively receive advice and treatment. By using smartphone devices, fitness trackers, or health applications, they could take some measurements on their own and also improve their health-related data (e.g., medication list, family history) by linking them to other personal data (e.g., income, education, dietary habits, exercise). By maintaining a digital medical history that always resides with them, patients obtain greater control over their health. Based on the individual's history, big data offer the capability of creating an indicator in a patient-directed way of whether the patient could be targeted and participate in public health initiatives, to reduce for example smoking or obesity. Taking it one step further, big data could also contribute to the medicine development process. People will be able to better evaluate in a more direct way a drug, since analyzing large patient populations data and making results available is feasible.

1.1.4 Applications of Machine Learning to Medical Problems

There is a number of papers in the literature that examine the application of data analytics techniques to healthcare problems. [Raghupathi and Raghupathi, 2014] mention as examples: detecting diseases at earlier stages (e.g., [Li et al., 2007], [Moore et al., 2013]; managing population health by detecting vulnerabilities within patient populations during disease outbreaks or disasters (e.g., [Wong et al., 2002]), predicting outcomes, such as length of stay, based on historical data (e.g., [Hachesu et al., 2013], [Frost et al., 2017], [Pathak et al., 2013], [Hrovat et al., 2014]); forecasting illness/disease progression (e.g., [Rizk-Jackson et al., 2011]); discovering causal factors for co-morbid conditions (e.g., [Diamond and Sekhon, 2013]); reducing readmissions (e.g., [Bayati et al., 2014]); improving outcomes by examining at-home health monitors (e.g., [Costa et al., 2012], [Rodríguez-Martín et al., 2017], [Forkan and Khalil,

2017]); combining clinical, financial and operational data to analyze resource utilization productively and in real time (e.g., [IBM, 2013]).

[Obenshain, 2004] discuss the mining applications in the drug discovery process (e.g., [Burbidge et al., 2001]), infection control surveillance [Brossette et al., 2000], ranking hospitals and healthcare plans [Cerrito et al., 2002] and identifying high-risk patients [Ridinger, 2002]. [Koh et al., 2011] refer to how machine learning can help healthcare insurers detect fraud and abuse (e.g., [Tomar and Agarwal, 2013], [Boxwala et al., 2011]), healthcare organizations make customer relationship management decisions (e.g., [Paddison, 2000]), physicians identify effective treatments and best practices, and patients receive better and more affordable healthcare services (e.g., [Groves et al., 2013]).

[Magoulas and Prentza, 2001] discuss the use of ML for extracting medical knowledge for outcome research (e.g., [Phan et al., 2017]), therapy planning and individualized support (e.g., [Tuarob et al., 2017]), interpretation of continuous data used in the Intensive Care Unit (e.g., [Kim et al., 2011]), and medical diagnostic reasoning (e.g., [Stausberg and Person, 1999]). Even more examples of applications can be found in [Meyfroidt et al., 2009], [Yoo et al., 2012] and [Dua et al., 2014].

1.1.5 Challenges and Concerns

The challenges encountered in using analytics with big healthcare data do not only concern volume, heterogeneity, complexity and uncertainty [Fan et al., 2014]. There are major concerns especially tied to the medical domain.

First, there is the problem of data privacy, i.e., who has access to patients personal information and under what conditions. In the U.S., the law that provides a standard for medical privacy is the Health Insurance Portability and Accountability Act (HIPAA), voted in 1996. Removing a set of protected health information (PHI) ensures HIPAA compliance and allows querying the data. Examples of PHI identifiers

are names, social security numbers, medical records numbers, dates (other than year) and geographical identifiers smaller than a state. Even though there is regulation in place and security measures are taken in health institutions to protect medical data, the healthcare industry is experiencing a number of data breaches. According to the Identity Theft Resource Center's reports, in 2013 the health sector breaches represented 43% of the total overall breaches, with the percent increasing to 44.1% in 2014 and dropping to 35.5% in 2015. The 2013 health sector data breaches impacted 1.84 million Americans and the average victim held liable for more than $\$18,600$ in medical services, according to the Ponemon Institutes 2013 Survey on Medical Identity Theft. Clearly, even with the patients identities blinded, the risk of a central data repository being compromised renders data privacy an important concern and medical data sharing challenging.

Another impeding factor for healthcare to embrace the benefits of big data is that all stakeholders must recognize the value of analytics and be willing to act on its insights and this is a fundamental change in mentality. Diagnostic decision support systems (DDSS) have been developed for and used by physicians. Studies have indicated that although physicians acknowledge the need for diagnostic support, they rarely alter their diagnostic judgment based on recommendations by the system [Ridderikhoff and van Herk, 1997], [Ridderikhoff and van Herk, 1999], [Rosenbloom et al., 2004]. Moreover, there are entities within the health sector that do not deliver direct individual patient care, such as health service researchers, pharmaceutical companies, public health or other government organizations, who yet do not exploit big data due to a lack of coherent policies and standard good practices for secondary use of health data [Safran et al., 2007]. While at first, there might have not been strong enough incentives for the adoption of algorithmic approaches in practice, the healthcare industry nowadays has been transitioning from a fee-for-service to a value-

based reimbursement model, which is mandating better care at a lower cost. Health providers that cannot achieve the required quality-of-care scores, face significant financial penalties. This, together with the realization of the business benefits of big data, are driving the required mindset shift. The combination of physicians and data-driven assistance in clinical practice deserves further and thorough exploration. To that end, it is of critical importance for the proposed algorithmic big data solutions to be as much as possible accessible to and comprehensible by the physicians.

1.1.6 Methodologies: What is Needed, What Works, What Matters

There is a number of established and emerging machine learning paradigms for healthcare informatics. Successful application of ML in healthcare problems, requires accuracy, transparency, ability to deal with complex data (often imbalanced, of low or very large sample size, of high dimensionality, including missing data, of varying time intervals), ability to incorporate background knowledge to the model, and time efficiency. More than that, algorithms need to be robust, since they often deal with life or death decisions.

As nicely suggested in [Kohavi et al., 1997], it is useful to "take each algorithm for a test drive" and keep in mind the criteria of classification accuracy, comprehensibility, compactness, training and classification time. For the medical domain in particular, accuracy is important, but also interpretability is a paramount quality that machine learning methods should aim to achieve [Vellido et al., 2012]. Sparse classifiers are interpretable, since they provide feedback on how important each feature is. Based on this feedback, one can often remove many non-predictive variables from a model without any significant loss in accuracy, highlighting at the same time features that most affect the classification decision. We will extensively experiment with and show the advantages of sparse classifiers in this thesis. While harder to interpret than linear and sparse algorithms, ensemble methods that build collections of classifiers,

such as boosting and random forests, can model non-linear relationships, have proven to provide more accurate models for common healthcare problems, including the one we study in this thesis, and run quite quickly. We will also explore other models, such as Naive-Bayes classifiers, which seem to perform well in the medical domain [Kononenko, 1993], and logistic regression [Wu et al., 2010].

Last, it is our belief that the increasing volume of available healthcare data and the concerns about data privacy will cause a paradigm shift in data sharing and healthcare analytics towards a distributed computing model. Such a scheme is more scalable compared to a single or centralized computing site and, given security measures are in place, it has many key benefits: greater statistical power due to the large number of samples and ability to (a) study occurrences of rare outcomes, uptake or usage of new drugs or therapies, and diverse populations of individuals, (b) combine sources of data to develop novel analytic and statistical methods, and (c) alleviate data holders concerns over data security, patient privacy and proprietary interests [Popovic, 2015]. This thesis explores this research direction too.

1.2 Thesis Goal: Predict Chronic Disease Hospitalizations

The key problem we will address in this thesis is to *explore and develop centralized and distributed methods to predict hospitalizations (i.e., admissions to the hospital) during a target year for patients with heart-related diseases or diabetes based on their medical history as described in their Electronic Health Records.*

Diseases of the heart have been consistently among the top causes of death. In the U.S., heart disease is yearly the cause of one in every four deaths, which translates to 610,000 people, while every year, about 735,000 Americans have a heart attack [Centers for Disease Control and Prevention, 2015]. At the same time, diabetes is recognized as the worlds fastest growing chronic condition. One in eleven adults has

diabetes worldwide (415 million) and 12% of global health expenditures is spent on diabetes ($673 billion) [International Diabetes Federation, 2015]. In the U.S. alone, 29.1 million people or 9.3% of the population had diabetes in 2012 [Centers for Disease Control and Prevention, 2014]. Given its impact, medical and health services studies have been tracking the prevalence and trends in diabetes among adults [Menke et al., 2015,King et al., 1998,Rathmann and Giani, 2004]. While heart diseases and diabetes affect primarily the patients at many levels (physical, financial, etc.), they also pose an economic burden to states influencing healthcare costs and GDP/productivity metrics.

According to [Jiang et al., 2009], nearly $30.8 billion in hospital care cost during 2006 was preventable. Heart diseases and diabetes were the leading contributors accounting correspondingly for more than $9 billion, or about 31% and for almost $6 billion, or about 20%. Clearly, even modest percentage reductions in these amounts matter. This motivates our research to predict heart and diabetes-related hospitalizations. Two key enablers to such research are the availability of patient EHRs and the existence of sophisticated (machine learning) algorithms that can process and learn from the data.

Our algorithms consider the history of a patient's records and predict whether each individual patient will be hospitalized in the following year, thereby, alerting the health care system and potentially triggering preventive actions. An obvious advantage of our algorithmic approach is that it can easily scale to a very large number of monitored patients; such scale is not possible with human monitors.

In Chapter 2, we provide a detailed description of the two datasets we will be experimenting with in this thesis, namely the dataset of patients with heart-related diseases and the dataset of patients with diabetes. The EHRs come from Boston Medical Center, the largest safety-net hospital in Boston. We formulate the problem

as a binary classification problem and seek to differentiate between patients that will be hospitalized in a target year and those who will not. Chapters 3 and 4 explore methods that separate the two classes of samples (patients) using a single classifier. Specifically, in Chapter 3 we explore a set of baseline methods and we evaluate their performance in terms of prediction accuracy and interpretability of the model and the results. Baseline methods include well-established methods, such as Support Vector Machines, AdaBoost with trees as the weak learner, logistic regression, and also a novel likelihood ratio based method we develop, K-LRT, that identifies the K most significant features for each patient that lead to hospitalization.

We continue the analysis in Chapter 4 by developing a distributed cluster Primal Dual Splitting (cPDS) method, that is computationally efficient and privacy-aware. Such a distributed leaning scheme is relevant for multi-institutional (e.g., hospitals) collaborations or peer-to-peer (e.g., patients' smartphones) applications, allowing the"agents" to collaborate, while keeping every participant's data private. We also show that cPDS has an improved convergence rate compared to existing centralized and decentralized methods. We test all methods in Chapters 3 and 4 in the heart disease dataset.

In Chapter 5, assuming that the positive class consists of multiple clusters (hospitalized patients due to different reasons), while the negative class is drawn from a single cluster (non-hospitalized patients healthy in every aspect), we present an alternating optimization approach, which jointly discovers the clusters in the positive class and optimizes the classifiers that separate each positive cluster from the negative samples. We also establish the convergence of the method, characterize its VC dimension and present experimental results on both the heart disease and the diabetes datasets. Lastly, we conclude in Chapter 6 with our key findings in this thesis and we discuss potential future research directions for this work.

1.3 Position in the Literature

To the best of our knowledge, the problem of chronic disease hospitalization prediction using ML methods is novel. A closely related problem, which has received a lot of attention in the literature, is the re-hospitalization prediction, since around 20% of all hospital admissions occur within 30 days of previous discharge. Medicare penalizes hospitals that have high rates of readmissions, especially among patients with heart failure, heart attack and pneumonia. Identifying patients at risk of readmission can guide efficient resource utilization and can potentially save millions of healthcare dollars each year. Examples of work on this problem include [Hosseinzadeh et al., 2013], [Zolfaghar et al., 2013], [Strack et al., 2014] and [Caruana et al., 2015].

Other related problems are: predicting the onset of diabetes using artificial neural networks [Pradhan and Sahu, 2011], developing an intelligent system that predicts, using data mining techniques, which patients are likely to be diagnosed with heart disease [Palaniappan and Awang, 2008] (it can answer complex queries for diagnosing heart disease and thus assist healthcare practitioners to make intelligent clinical decisions, which traditional decision support systems cannot), and using data mining techniques to predict length of stay in cardiac patients (with decision trees, support vector machines and artificial neural networks) [Hachesu et al., 2013] or in acute pancreatitis (with artificial neural networks) [Pofahl et al., 1998].

As the last related problem, let us mention the Heritage Health Prize, a competition by Kaggle, whose goal was to predict the length of stay for patients who will be admitted to a hospital within the next year, using historical claims (insurance) data and data mining techniques [Heritage Provider Network, 2011].

1.4 Our Contributions

- In terms of the application field, we explore a novel, to the best of our knowledge, problem, that of predicting chronic disease hospitalizations, using centralized or distributed machine learning methods that we develop. We analyze the results of the various methods in terms of accuracy and interpretability.

- We show that the accuracy rates achieved by our computational methods surpass what is possible with more empirical but well accepted risk metrics, such as a heart disease risk factor that emerged out of the Framingham study [D'Agostino et al., 2008]. We show that even a more sophisticated use of the features used in the Framingham risk factor, still leads to results inferior to our approaches. This suggests that the entirety of a patient's EHR is useful in the prediction and this can only be achieved with a systematic algorithmic approach.

- We develop a likelihood ratio based method, K-LRT, that is able to identify the K most significant features for each patient that lead to hospitalization. K-LRT is proven to achieve high accuracy providing at the same time interpretability to the results.

- We propose a decentralized optimization scheme (cPDS) for solving the sparse Support Vector Machines problem. Advantages of this scheme include the scalability and the fact that it avoids raw data exchange, which are crucial for applications in domains like healthcare. We also prove that cPDS has improved convergence rate compared to alternatives. The cPDS framework is general and can be applied to solve other problems that follow the "nonsmooth+nonsmooth" minimization structure. Such problems can be found in machine learning, where we aim to minimize functions with non-smooth reguralizers, or in distributed model predictive control.

- We apply cPDS to the sparse SVM problem and illustrate its efficiency on the real healthcare problem we are studying at this thesis, that is aiming to differentiate between patients that are likely or not likely to be hospitalized within a target year.

- We propose a novel method, an alternating optimization approach, which jointly discovers the clusters in the class of hospitalized patients and optimizes the classifiers that separate each cluster of hospitalized patients (positive class) from the non-hospitalized patients (negative class). We establish the convergence of this joint clustering/ classification process and characterize its Vapnik-Chervonenkis (VC) dimension; a metric of complexity of the classification function that can lead to generalization guarantees.

1.5 Bibliographic Notes

Large part of the thesis appears in published or working research papers: [Dai et al., 2014], [Dai, 2015], [Dai et al., 2015], [Xu et al., 2016], [Brisimi et al., 2016] and [Brisimi et al., 2017] .

Chapter 2

Datasets: Electronic Health Records of patients with heart diseases or diabetes

The purpose of this chapter is to describe in detail the two datasets that will be used throughout the thesis. The first dataset contains the medical history of patients who have at least one heart-disease diagnosis, while the second dataset contains the medical history of patients who have at least one diabetes mellitus diagnosis. Both datasets are extracted from the Boston Medical Center. A survey by the American Hospital Association showed that adoption of EHRs has doubled from 2009 to 2011, partly a result of funding provided by the Health Information Technology for Economic and Clinical Health Act of 2009 [Charles et al., 2012]. Indeed, in the two datasets we compile, we observe an abundance of EHRs after 2007, with the number of EHRs increasing every year.

2.1 Heart Diseases Dataset

2.1.1 Detailed Data Description

The data we used are from the Boston Medical Center (BMC) - the largest safety-net hospital in New England. The study is focused on patients with at least one heart-related diagnosis or procedure record in the period 01/01/2005–12/31/2010. For each patient in the above set, we extract the medical history (demographics, visit history, problems, medications, labs, procedures and limited clinical observations) for the period 01/01/2001–12/31/2010, which contains relevant *medical factors* and

from which the features of the dataset will be formed. Data were available from the hospital EHR and billing systems. The ontologies, along with the number of factors and some examples corresponding to each, are shown in Table 2.1[1]. We note that some of the Diagnoses and Admissions are not directly heart-related, but may be good indicators of a heart problem. Overall, our data set contains 45,579 patients (60% of which compose the training set and the rest the test set).

Our objective is to leverage past medical factors for each patient to predict whether she/he will be hospitalized or not during a *target* year which could be different for each patient.

Ontology	**Number of Factors**	**Examples**
Demographics	4	Sex, Age, Race, Zip Code
Diagnoses	22	e.g., Acute Myocardial Infarction (ICD9: 410), Cardiac Dysrhythmias (ICD9: 427), Heart Failure (ICD9: 428), Acute Pulmonary Heart Disease (ICD9: 415), Diabetes Mellitus with Complications (ICD9: 250.1-250.4, 250.6-250.9), Obesity (ICD9: 278.0)

[1]ICD9 (International Classification of Diseases, 9th revision), CPT (Current Procedural Terminology), LOINC (Logical Observation Identifiers Names and Codes), and MSDRG (Medicare Severity-Diagnosis Related Group) are commonly used medical coding systems for diseases, procedures, laboratory observations, and diagnoses, respectively.

Procedures CPT	3	Cardiovascular Procedures (including CPT 93501, 93503, 93505, etc.), Surgical Procedures on the Arteries and Vein (including CPT 35686, 35501, 35509, etc.), Surgical Procedures on the Heart and Pericardium (including CPT 33533, 33534, 33535)
Procedures ICD9	4	Operations on the Cardiovascular System (ICD9: 35-39.99), Cardiac Stress Test and pacemaker checks (ICD9: 89.4), Angiocardiography and Aortography (ICD9: 88.5), Diagnostic Ultrasound of Heart (ICD9: 88.72)
Vitals	2	Diastolic Blood Pressure, Systolic Blood Pressure
Lab Tests	4	CPK (Creatine phosphokinase) (LOINC:2157-6), CRP Cardio (C-reactive protein) (LOINC:30522-7), Direct LDL (Low-density lipo-protein) (LOINC:2574-2), HDL (High-density lipoprotein) (LOINC:9830-1)
Tobacco	2	Current Cigarette Use, Ever Cigarette Use
Visits to the ER	1	Visits to the Emergency Room

Admissions	17	e.g., Heart Transplant or Implant of Heart Assist System (MSDRG: 001, 002), Cardiac Valve and Other Major Cardiothoracic procedures (MSDRG: 216-221), Coronary Bypass(MSDRG: 231-234), Acute Myocardial Infarction (MSDRG: 280-285), Heart Failure and Shock (MSDRG: 291-293), Cardiac Arrest (MSDRG: 296-298), Chest Pain (MSDRG: 313), Respiratory System related admissions (MSDRG: 175-176, 190-192)

Table 2.1: Medical Factors in the Heart Diseases Dataset.

In order to organize all the available information in some uniform way for all patients, some preprocessing of the data is needed to summarize the information over a time interval. Details will be discussed in the next subsection. We will refer to the summarized information of the medical factors over a specific time interval as *features*.

Each feature related to Diagnoses, Procedures CPT [American Medical Association, 2014], Procedures ICD9 [World Health Organization, 1999] and Visits to the Emergency Room is an integer count of such records for a specific patient during the specific time interval. Zero indicates absence of any record. Blood pressure and lab tests features are continuous valued. Missing values are replaced by the average of values of patients with a record at the same time interval. Features related to tobacco use are indicators of current- or past-smoker in the specific time interval. Admission features contain the total number of days of hospitalization over the specific time interval the feature corresponds to. Admission records are used both to form the Admission features (past admission records) and in order to calculate the prediction

variable (existence of admission records in the target year). We treat our problem as a classification problem and each patient is assigned a *label*: 1 if there is a heart-related hospitalization in the target year and 0 otherwise.

2.1.2 Data Preprocessing

In this section we discuss several data organization and preprocessing choices we make. For each patient, a target year is fixed (the year in which a hospitalization prediction is sought) and all past patient records are organized as follows.

Summarization of the medical factors in the history of a patient: Based on experimentation, an effective way to summarize each patient's medical history is to form four time blocks for each medical factor with all corresponding records summarized over one, two, and three years before the target year and all earlier records being summarized in a fourth block. For blood pressure and tobacco use, only the year before the target year is kept. This process results to a vector of 212 features for each patient.

Selection of the target year: As a result of the nature of the data, the two classes are highly imbalanced. When we fix the target year for all patients to be 2010, the number of hospitalized patients is about 2% of the total number of patients, which makes the classification problem much more challenging. Thus, and to increase the number of hospitalized patient examples, if a patient had only one hospitalization throughout 2007–2010, the year of hospitalization is set as the target year for that patient. If a patient had multiple hospitalizations, a target year between the first and the last hospitalization is randomly selected.

Setting the target time interval to be a year: Based on experimentation, a year has been proven to be an appropriate time interval for prediction. Moreover, given that hospitalization occurs roughly uniformly within a year, we take the prediction time interval to be a calendar year.

Removing noisy samples: Patients who have no records before the target year are impossible to predict and are thus removed.

2.1.3 Correlation Between Features

The correlation coefficient matrix of all features is shown in Fig. 2·1. Each point (i, j) corresponds to the correlation coefficient between feature i and feature j. There are a few features with zero variance (shown as white stripes) that are later removed from the features set. Most of the features are weakly correlated. There is moderate correlation between features that refer to the same medical factor but correspond to different time blocks (near- diagonal elements) and between few other pairs of features including: Diagnosis of Chronic Ischemic Heart Disease with Diagnosis of Diabetes, Diagnosis of Ischemic Heart Disease with Diagnosis of Old Myocardial Infarction, Diagnosis of Heart Failure with Admission due to Heart Failure, and Operations on Cardiovascular System with Ultrasound of the Heart.

2.2 Diabetes Dataset

2.2.1 Detailed Data Description

The data in this dataset also come from the Boston Medical Center (BMC). The population of the study consists of patients with a Diagnosis record of Diabetes Mellitus between 01/01/2007–12/31/2012. For each patient in the above set, we extract their medical history (demographics, visit history, problems, procedures and department information) for the period 01/01/2001–12/31/2012, which contains relevant *medical factors* and from which the features of the dataset will be formed. Data are available from the hospital EHR and billing systems. The ontologies, along with some examples corresponding to each, are shown in Table 2.2. As expected, many of the diagnoses and procedures are direct complications due to diabetes. Diabetes-related admissions

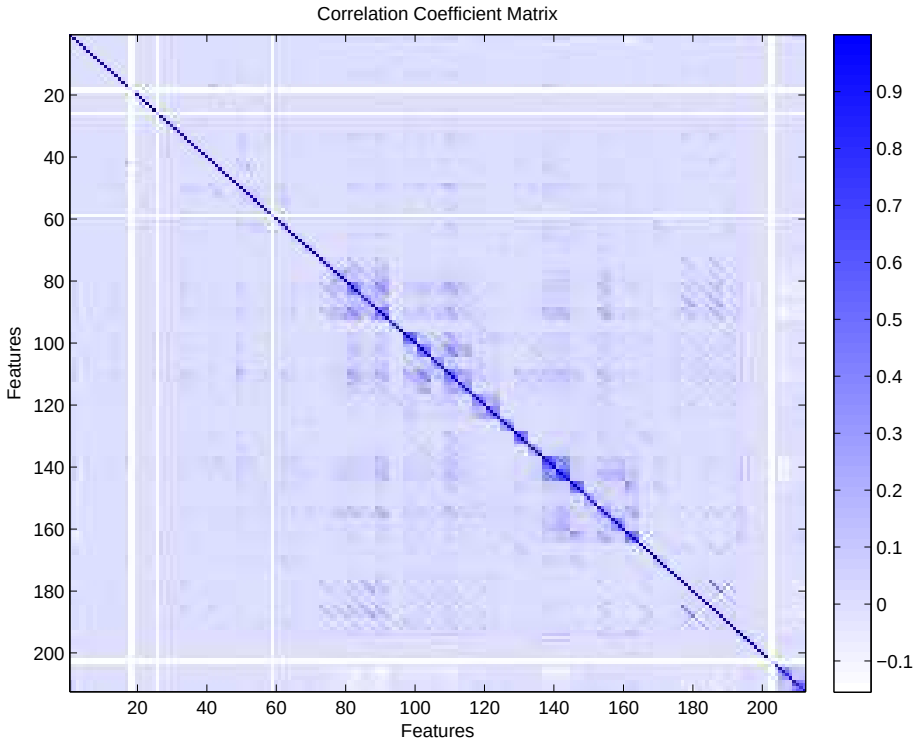

Figure 2·1: Correlation coefficient between features in the heart diseases dataset.

are not trivially identifiable, and are revealed through the procedure described in the next subsection. Overall, our data set consists of 40,921 patients (60% of which form the training set and the rest the test set).

Similarly with the heart-related dataset, our objective is to leverage past medical factors for each patient to predict whether she/he will be hospitalized or not during a *target* year which could be different for each patient. The target year for the never-hospitalized patients is set to be 2012, so that there is as much possible available history for them. Patients that have a single hospitalization in a year between 2007-2012, have this year as their target year. For patients with multiple hospitalizations, the target year is randomly selected between their first and last hospitalization. The fraction of hospitalized patients is 16.97%.

Ontology	Examples
Demographics	Sex, Age, Race, Zip Code
Diagnoses	e.g., Diabetes mellitus with complications, Thyroid disorders, Hypertensive disease, Pulmonary heart disease, Heart failure, Aneurysm, Skin infections, Abnormal glycose tolerance test, Family history of diabetes mellitus
Procedures (CPT or ICD9)	e.g., Procedure on single vessel, Insertion of intraocular lens prosthesis at time of cataract extraction, Venous catheterization, Hemodialysis, Transfusion of packed cells
Admissions	e.g., Diabetes (with and without) complications, Heart failure and shock, Deep Vein Thrombophlebitis, Renal failure, Chest pain, Chronic obstructive pulmonary disease, Nutritional. & misc metabolic disorders, Bone Diseases & Arthropathies, Kidney & urinary tract infections, Acute myocardial infarction, O.R. procedures for obesity, Hypertension
Service by Department	Inpatient (admit), Inpatient (observe), Outpatient, Emergency Room

Table 2.2: Medical Factors in the Diabetes Dataset.

2.2.2 Identifying the Diabetes-Related Hospitalizations

Identifying the hospitalizations that occur mainly due to diabetes is not a trivial task, the reason being that many diabetes- related hospitalizations are recorded in the system as other types of admissions, e.g., heart-related, mostly because of financial reasons (the billing system charges more for other diseases than diabetes). Therefore, as a first step we aim to separate the diabetes-related admissions -including diabetes-related admissions that are labeled otherwise- from all the rest. For that, we consider all patients with at least one admission record (that indicates hospitalization) between 1-1-2007 and 12-31- 2012. From this set, patients with at least one Diabetes Mellitus record during 1-1-2007 up to 12-31-2012 are assigned to the *diabetic population*, while the rest are assigned to the *non-diabetic population*.

We list the union of all the unique admission types for both populations (732

unique types). The total number of admission samples for the diabetic and non-diabetic populations are $N_1 = 47,352$ and $N_2 = 116,934$ correspondingly. For each type of admission d, each admission sample can be viewed as the outcome of a binary random variable, that takes the value 1, if the sample hospitalization occurs because of this type of admission, or 0 otherwise. Thus, we can transform the two sets of admission records for the two populations into 0/1 sequences. By comparing in the way described below the proportions of d in the two populations, we can infer whether admission d was caused mainly by diabetes or not.

At this point, we will elaborate on the statistical hypothesis test used that involves sample differences of proportions [Sprinthall and Fisk, 1990]. Let P_1 and P_2 be the sample proportions obtained in large samples of sizes N_1 and N_2 drawn from respective populations p_1 and p_2. Consider the null hypothesis that the population parameters are the same ($p_1 = p_2$) and thus that the samples are drawn from the same distribution. The sampling distribution of differences in proportions is approximately normally distributed, with its mean and standard deviation given by

$$\mu_{P_1-P_2} = 0 \quad \text{and} \quad \sigma_{P_1-P_2} = \sqrt{pq\left(\frac{1}{N_1} + \frac{1}{N_2}\right)}, \tag{2.1}$$

where $p = (N_1P_1 + N_2P_2)/(N_1 + N_2)$ is used as an estimate of the population proportion and where $q = 1 - p$. By using the standardized variable $z = (P_1 - P_2)/(\sigma_{P_1-P_2})$ we can check if the results observed in the samples differ markedly from the results expected under the null hypothesis. We do that using the *single sided p-value* of the statistic z. The p-value is the probability of observing a sample statistic as more extreme than the one observed under the assumption that the null hypothesis is true. To interpret that, a small p-value (typically ≤ 0.05) indicates strong evidence against the null hypothesis. Thus, the smaller the p-value is, the higher the confidence we

have in the alternative hypothesis or equivalently in the fact that the diabetic patients have higher chance of getting admission records of type d than the non-diabetic ones (since we consider the difference $P_1 - P_2$). After producing the list of increasing order p-values for each one of the admission types d, we infer that the ones with p-value $\leq \alpha = 1E-4$[2] are caused by diabetes. Examples of diabetes-related admissions are shown in Table 2.2.

2.2.3 Data Preprocessing

The features are formed as combinations of different medical factors (instead of considering the factors as separate features) that better describe what happened to the patients during their visits to the hospital. Specifically, we formulate triplets that consist of a diagnosis, a procedure (or the information that no procedure was done) and the service department. An example of a complex feature (a triplet) is the diagnosis of ischemic heart disease that lead to an adjunct vascular system procedure (procedure on single vessel) while the patient was admitted to the inpatient care. Clearly, since each category can take one of several discrete values, a huge number of combinations should be considered. Naturally, not all possible combinations occur, which reduces significantly the total number of potential features that describe each patient. Also for each patient, we extract information about the diabetes type over their history and demographics including age, gender and race. Next, we present several data organization and pre-processing steps we take. For each patient, a target year is fixed and all past patient records are organized as follows.

Forming the complex features. We create a diagnoses-procedures indicator matrix to keep track of which diagnosis occurs with which procedure. The procedures that are not associated with any diabetes-related diagnosis are removed. Procedures

[2]Apart from selecting a small-value α, we also check the cumulative fraction of patients that are potentially labeled as belonging to the hospitalized class not to be too small, so that the dataset is not highly imbalanced.

in the dataset are listed in the most detailed level of the ICD9 coding system [World Health Organization, 1999] or the CPT coding system [American Medical Association, 2014]. We group together procedures that belong to the same ICD/CPT family, resulting in 31 categories (out of 2004 in total).

Summarization of the complex features in the history of a patient. Based on experimentation, an effective way to summarize each patient's medical history with a fixed target year is to form four time blocks for each medical factor with all corresponding records summarized over one, two, three years before the target year and a fourth time block containing averages of all the earlier records. This produces a 9402-dimensional vector of features characterizing each patient.

Reducing the number of features. We remove all the features that do not contain enough information for a significant amount of the population (less than 1% of the patients), as they could not help us generalize. This leaves 320 medical and 3 demographical features.

Identifying the diabetes type. The ICD9 code for diabetes is assigned to category 250 (diabetes mellitus). The fifth digit of the diagnosis code determines the type of diabetes and whether it is uncontrolled or not stated as uncontrolled. Thus, we have four types of diabetes diagnoses: type II, not stated as uncontrolled (fifth digit 0), type I, not stated as uncontrolled (fifth digit 1), type II or unspecified type, uncontrolled (fifth digit 2) and type I, uncontrolled (fifth digit 3). Based on these four types, we count how many records of each type each patient had in the four time blocks before the target year, thus adding 16 new features for each patient.

Setting the target time interval to a calendar year. Based on some preliminary experiments we conducted, we observed that there is greater variability in the results when trying to predict hospitalizations in periods of time shorter than a year (e.g., predicting hospitalization in the next 1, 3 or 6 months). Thus, we have designed our

experiment to predict hospitalizations in the target time interval of a year starting on the 1st of January and ending on the 31st of December.

Selection of the target year. As a result of the nature of the data, the two classes are highly imbalanced. To increase the number of hospitalized patient examples, if a patient had only one hospitalization throughout 2007-2012, the year of hospitalization will be set as the target year. If a patient had multiple hospitalizations, a target year between the first and the last hospitalizations will be randomly selected. 2012 is set as the target year for patients with no hospitalization, so that there is as much available history for them as possible. By this policy, the ratio of hospitalized patients in the data set is 16.97%.

Removing patients with no record. Patients who have no records before the target year are removed, since there is nothing on which a prediction can be based. The total number of patients left is 33,122.

Splitting the data into a training set and a test set randomly. As is common in supervised machine learning, the population is randomly split into a training and a test set. Since from a statistical point of view, all the data points (patients features) are drawn from the same distribution, we do not differentiate between patients whose records appear earlier in time than others with later time stamps. A retrospective/prospective approach appears more often in the medical literature and is more relevant in a clinical trial setting, rather than in our algorithmic approach. What is critical in our setting is that for each patient prediction we make (hospitalization/non-hospitalization in a target year), we only use that patients' information before the target year (cf. summarization of patient history above).

Chapter 3

Baseline Methods; Performance and Interpretability

To predict whether patients are going to be hospitalized in the target year given their medical history, we experiment with five different methods. All five are typical examples of supervised machine learning. We adapt the last one to better fit the specific application we examine. The first three methods fall into the category of discriminative learning algorithms, while the latter two are generative algorithms. Discriminative algorithms directly partition the input space into label regions without modeling how the data are generated, while generative algorithms assume a model that generates the data, estimate the model's parameters and use it to make classifications. Discriminative methods are likely to give higher accuracy, but generative methods provide more interpretable models and results. This is the reason we experiment with methods from both families and the trade-off between accuracy and interpretability is observed in our results.

3.1 Support Vector Machines (SVM)

An SVM is a very efficient two-category classifier [Cortes and Vapnik, 1995]. Intuitively, the SVM algorithm attempts to find a separating hyperplane in the feature space, so that data points from the two different classes reside on the different sides of that hyperplane. We can calculate the distance of each input data point from the hyperplane. The minimum over all these distances is called margin. The goal of SVM

is to find the hyperplane that has the maximum margin. In many cases data points are neither linearly nor perfectly separable. To that end, one can make the classifier tolerant to some misclassification errors (*soft-margin SVMs*) and leverage kernel functions to "elevate" the features into a higher dimensional space where linear separability is possible (*kernelized SVMs*) [Cortes and Vapnik, 1995]. Given training data $\mathbf{x}_i \in \mathbb{R}^d$ and labels $y_i \in \{-1, 1\}$, $i = 1, \ldots, n$, soft-margin SVMs find the classifier $(\boldsymbol{\beta}, \beta_0)$, $\boldsymbol{\beta} \in \mathbb{R}^d, \beta_0 \in \mathbb{R}$ by solving:

$$\begin{array}{ll} \min\limits_{\boldsymbol{\beta},\beta_0,\xi_i} & 0.5\|\boldsymbol{\beta}\|^2 + C\sum_{i=1}^n \xi_i \\ \text{s. t.} & \xi_i \geq 0, \quad \forall i \\ & y_i(\mathbf{x}_i^\intercal \boldsymbol{\beta} + \beta_0) \geq 1 - \xi_i, \quad \forall i \end{array} \tag{3.1}$$

where C is a tunable parameter and ξ_i is the penalty that will be imposed if point $(\mathbf{x}_i, y_i)$ is misclassified. Kernelized SVMs use $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_i)$ as a kernel for some feature mapping function ϕ and solve an optimization problem that is based on the dual of (3.1) to find the optimal $(\boldsymbol{\beta}, \beta_0)$. In our application, we employ the widely used Radial Basis Function (RBF) $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/2\sigma^2)$ [Scholkopf et al., 1997] as the kernel function in our experiment settings. Tuning parameters are the misclassification penalty coefficient C and the kernel parameter σ; we used the values [0.3, 1, 3] and [0.5, 1, 2, 7 15, 25, 35, 50, 70, 100], respectively. Optimal values of 1 and 7, respectively, were selected by cross-validation.

3.2 AdaBoost with Trees

Boosting [Freund et al., 1999] provides an effective way of combining decisions of not necessarily strong classifiers to produce highly accurate predictions. The AdaBoost algorithm iteratively adjusts the weights of various training data points through an exponential up-weighting or down-weighting procedure. Specifically, starting with equal weights, the algorithm generates in every iteration a new base classifier to best

fit the current weighted samples. Then, the weights are updated so that the misclassified samples are assigned higher weights so as to influence the training of the next base classifier. At termination, a weighted combination of the base classifiers is the output of AdaBoost. In our study we use stumps, which are two-level Classification and Regression Trees (CART), as the base classifier [Hastie et al., 2009]. This method recursively partitions the space into a set of rectangles and then fits a prediction within each partition. There is an extra preprocessing step applied to the data. The zip code values are clustered into 4 clusters using the k-means algorithm [Hastie et al., 2009] and this feature is treated as a categorical one. Moreover, we used cross-validation to set to 100,000 the number of Adaboost iterations.

3.3 Logistic Regression

Logistic Regression [Bishop et al., 2006] is a popular classification method in real applications. This method models the posterior probability that a sample falls into a certain class (e.g. positive class) as a logistic function and the input of this logistic function is the linear combination of the input features. Under this model, the log-likelihood ratio of the posterior probabilities of the two classes is a linear function of the input features. Therefore, the decision boundary that separates the two classes is still linear. However, beyond the classification decision, the prediction on a certain sample point naturally comes with a probability value, which could be meaningful in many applications. Thus, logistic regression is widely used.

3.4 Naïve Bayes Event Model

Naïve Bayes models are generative models that assume the features or "events" to be generated independently (naïve Bayes assumption [McCallum et al., 1998]). Naïve Bayes classifiers are among the simplest models in machine learning, but despite their

simplicity, they work quite well in real applications. There are two types of naïve Bayes models [McCallum et al., 1998]. The first one will be presented extensively in the next method. The second one, referred to as the Naïve Bayes Event Model, works as follows. To generate a new patient from the model, a label y will first be generated (hospitalized or non-hospitalized) based on a prior distribution $p(y)$. Then, for this patient, a sequence of events ($\mathbf{x}_t$'s) is generated by choosing each event independently from certain multinomial conditional distributions $p(\mathbf{x}|y)$. An event can appear many times for a patient and the overall probability of this newly generated patient is the product of the class prior with the product of the probabilities of each event. In our problem, an event is a specific combination of the medical factors. We consider only the medical factors from the following six ontologies: Diagnoses, Admissions, Emergency, Procedures CPT, Procedures ICD9, and Lab Tests. To generate such a data set, we aggregate the medical factors that belong to each one of these types and count the total number of records of the same type in each of the four time blocks discussed earlier that represent a patient's history. Thus, each patient is represented as a sequence of four events. To make events more intuitive and to reduce the total number of possible events, the data just formed are quantized into binary values and then the tuples of the six binary values (one for each ontology) are encoded into single values. We estimate the prior distribution of labels $p(y)$ and the conditional distributions $p(\mathbf{x}|y)$ from the training set and make predictions for the test set based on the likelihoods calculated from these distributions.

3.5 K-Likelihood Ratio Test

The Likelihood Ratio Test (LRT) is a Naïve Bayes classifier and, as described before, assumes that features x_i are independent. For this method as well, we quantize the data as shown in Table 3.1. In the quantized data set, the LRT algorithm em-

Features	**Levels of quantization**	**Comments**
Sex	3	0 represents missing information
Age	6	Thresholds at 40, 55, 65, 75 and 85 years old
Race	10	
Zip Code	0	Removed due to its vast variation
Tobacco (Current and Ever Cigarette Use)	2	Indicators of tobacco use
Diastolic Blood Pressure (DBP)	3	Level 1 if DBP $<$ 60mmHg, Level 2 if 60mmHg $\leq$ DBP $\leq$ 90mmHg and Level 3 if DBP $>$ 90mmHg
Systolic Blood Pressure (SBP)	3	Level 1 if SBP $<$ 90mmHg, Level 2 if 90mmHg $\leq$ SBP $\leq$ 140mmHg and Level 3 if SBP $>$ 140mmHg
Lab Tests	2	Existing lab record or Non-Existing lab record in the specific time period
All other dimensions	7	Thresholds are set to 0.01%, 5%, 10%, 20%, 40% and 70% of the maximum value of each dimension

Table 3.1: Quantization of Features.

pirically estimates the distribution $p(x_i|y)$ of each feature for the hospitalized and the non-hospitalized class. Given a new test sample $\mathbf{z} = (z_1, z_2, \cdots, z_n)$, LRT calculates the two likelihoods $p(\mathbf{z}|y = 1)$ and $p(\mathbf{z}|y = 0)$ ($y = 0$ corresponds to non-hospitalized and $y = 1$ to hospitalized) and then classifies the sample based on the ratio $p(\mathbf{z}|y = 1)/p(\mathbf{z}|y = 0)$. Due to independence, $p(\mathbf{z}|y = 1)/p(\mathbf{z}|y = 0) = \prod_{i=1}^{n} p(z_i|y = 1)/p(z_i|y = 0)$. In our variation of the method, which we will call K-LRT, instead of taking into account the ratios of the likelihoods of all features, we consider only the K features with the largest ratios. This type of method is closely related to the anomaly detection methods in [Saligrama and Zhao, 2012]. The purpose of this "feature selection" is to identify the K most significant features for each individual patient. Thus, each patient is actually treated differently. After experimentation, the best performance is achieved by setting $K = 4$. The prediction accuracy for $K = 1$ is also reported in the experimental results section.

3.6 Experimental Results on the Heart Disease Dataset

Typically, the primary goal of learning algorithms is to maximize the prediction accuracy or equivalently minimize the error rate. However, in the specific medical application problem we study, the ultimate goal is to alert and assist doctors in taking further actions to prevent hospitalizations before they occur, whenever possible. Thus, our models and results should be accessible and easily explainable to doctors and not only machine learning experts. With that in mind, we examine our models from two aspects: prediction accuracy and interpretability.

3.6.1 Prediction Accuracy

The prediction accuracy is captured in two metrics: the *False Alarm Rate* (the fraction of false positives out of the negatives) and the *Detection Rate* (the fraction of true positives out of the positives). Note that in the medical literature, the detection rate

is often referred to as *sensitivity* and the term *specificity* is used for one minus the false alarm rate. For a binary classification system, the evaluation of the performance using these two metrics is typically illustrated with the *Receiver Operating Characteristic (ROC)* curve, which plots the Detection Rate versus the False Alarm Rate at various threshold settings.

We first compare the performance of LRT using all features and K-LRT under different values of K. Fig. 3·1 shows the prediction accuracy for LRT, 1-LRT and 4-LRT. In Fig. 3·2, a comparison of the performance of all five methods we presented is illustrated. We also generate the ROC curve based on patients' 10-year risk of General Cardiovascular Disease defined in the Framingham Heart Study (FHS) [D'Agostino et al., 2008]. FHS is a seminal study on heart diseases that has developed a set of risk factors for various heart problems. The 10-years risk we are using is the closest to our purpose and has been widely used. We calculate this risk value (which we call the *Framingham Risk Factor-FRF*) for every patient and make the classification based on this risk factor only. We also generate an ROC by applying the AdaBoost with trees method just to the features involved in FRF. The generated ROC serves as a baseline for comparison.

Based on the experimental results, we draw the following conclusions:

1. LRT, 1-LRT and 4-LRT achieve very similar performance. This indicates that using only the most significant or several significant features with the largest likelihood ratios, is sufficient in making an accurate prediction. It also suggests that our problem is close to an "anomaly detection" problem and identifying the most anomalous feature captures most of the information that is useful for classification.

2. From the comparison of all five methods in Fig. 3·1, it can be seen that AdaBoost is the most powerful one and performs the best except for situations that require

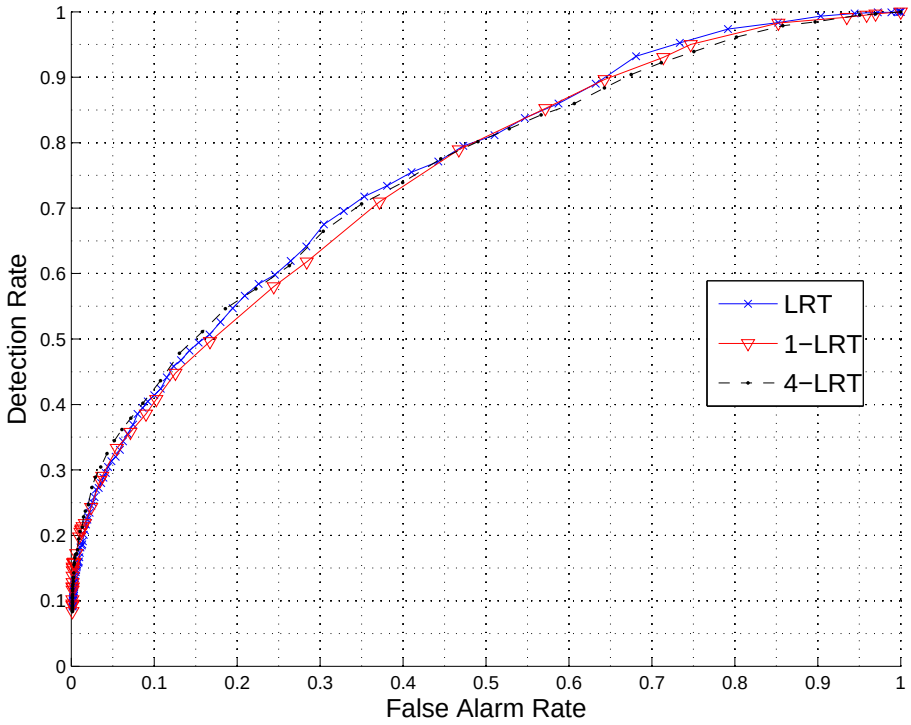

Figure 3·1: Comparison of LRT, 1-LRT and 4-LRT.

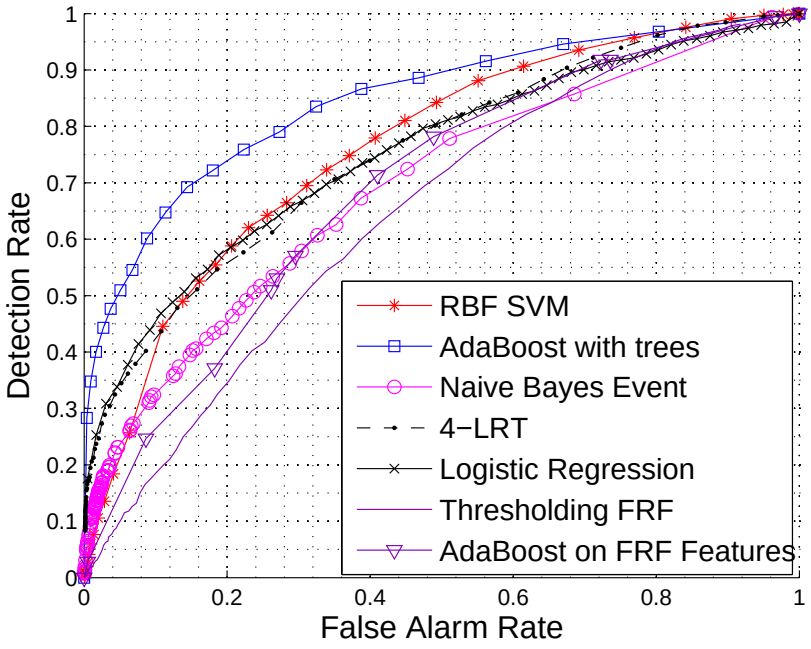

Figure 3·2: Comparison of all five methods and the methods based on the Framingham Heart Study.

very low False Alarm Rates. On the other hand, the Naïve Bayes Event classifier generally performs the worst due to its simplicity.

3. The performance of RBF SVM, Logistic Regression, AdaBoost with trees, and 4-LRT is quite similar in general. However, these methods have very different assumptions and underlying math formulation. Based on this observation, we conjecture that we have approached the limit of the prediction accuracy that could be achieved with the available data.

4. All of our proposed methods perform better than utilizing the FRF, except for the naïve Bayes event classifier for high false alarms rates. Even by applying AdaBoost with Trees (the best method so far) to the features involved in calculating the FRF, does not seem to help a lot. This suggests that it is valuable to have and leverage a multitude of patient-specific features obtained from EHRs. Using these data, however, necessitates the use of the algorithmic approach we advocate.

3.6.2 Interpretability

With SVM, the features are mapped through a kernel function from the original space into a higher-dimensional space. This, however, makes the features in the new space not interpretable. In AdaBoost with trees, while a single tree classifier which is used as the base learner is explainable, the weighted sum of a large number of trees makes it relatively complicated to find the direct attribution of each feature to the final decision. The naïve Bayes Event model is in general interpretable, but in our specific problem each patient has a relatively small sequence of events (four) and each event is a composition of medical factors. Thus, again, to find the direct attribution of each feature to the final decision is hard. LRT itself and Logistic Regression still lack interpretability, because we have more than 200 features for each sample and there

is no direct relationship between prediction of hospitalization and the reasons that led to it. The most interpretable method is K-LRT. K-LRT highlights the top K features that lead to the classification decision. These features could be of help in assisting the physicians reviewing the patient's EHR profile.

Below we present the features highlighted by two of our methods: 1-LRT and Adaboost. We remind the reader that in 1-LRT, each test patient is essentially associated with a single feature. For all features, we count how many times they were selected as the primary feature and we report in Table 3.2 the 10 features that were the most popular as primary. Adaboost, on the other hand, yields a linear combination of decision trees and is hard to interpret. However, we can calculate a variable Importance Score (IS) [Hastie et al., 2009] for each feature, which highlights the most significant features. Table 3.3 lists the top 10 important features indicated by their importance score. Features that appear in both tables are in bold.

The two sets of features highlighted from the two methods have several features in common, indicating that the results from the different methods are consistent. This consistency supports the validity of our methods from a stability/sensitivity perspective as well.

From a medical point of view, the features listed in Table 3.2 and Table 3.3 are reasonably highlighted. ER visits, a diagnosis of heart failure, and chest pain or other respiratory symptoms are often precursors of a major heart episode. The CPK test is also viewed as one of the most important tests for diagnosing Acute Myocardial Infarction (AMI) and AMI, among all heart diseases, is the most probable to lead to hospitalization.

To provide additional insight into the algorithms, Table 3.4 presents five more medically significant features highlighted by each method and two interesting features with low significance in both methods. For 1-LRT, features with low significance are

1-**LRT Counts**	1-**LRT Feature Name**
1591	**Age**
548	**Visit to the Emergency Room, 1 year before the target year**
525	Diagnosis of hematologic disease, 1 year before the target year
523	**Diagnosis of heart failure, 1 year before the target year**
514	**Symptoms involving respiratory system and other chest symptoms, 1 year before the target year**
486	**Diagnosis of diabetes mellitus w/o complications, 1 year before the target year**
474	Lab test CPK, 1 year before the target year
451	Lab test CPK, 4 years before the target year and the rest of the history
408	**Diagnosis of heart failure, 2 years before the target year**
356	Diagnosis of diabetes mellitus w/o complications, 2 years before the target year

Table 3.2: Top 10 significant features for 1-LRT.

the ones with a likelihood ratio $p(z_i|y = 1)/p(z_i|y = 0)$ close to 1. For Adaboost, non-significant features have a low IS. It is interesting that Hypertensive heart disease is considered non-significant by both methods. This is probably due to the fact that, once diagnosed, it is usually well-treated and the patient's blood pressure is well-controlled.

AdaBoost IS ($\times 10^{-4}$)	**AdaBoost Feature Name**
0.6462	Diagnosis of diabetes mellitus w/o complications, 1 year before the target year
0.5498	**Diagnosis of heart failure, 1 year before the target year**
0.4139	**Age**
0.3187	**Symptoms involving respiratory system and other chest symptoms, 1 year before the target year**
0.2470	Admission due to other circulatory system diagnoses, 1 year before the target year
0.2240	Visit to the Emergency Room, 4 years before the target year and the rest of the history
0.1957	Operations on cardiovascular system (heart and septa OR vessels of heart OR heart and pericardium), 4 years before the target year and the rest of the history
0.1578	**Visit to the Emergency Room, 1 year before the target year**
0.1543	Symptoms involving respiratory system and other chest symptoms, 4 years before the target year and the rest of the history
0.1124	**Diagnosis of heart failure, 2 year before the target year**

Table 3.3: Top 10 significant features for AdaBoost with Trees.

Another 5 significant features in 1-LRT	**Another 5 significant features in 1-LRT**
Lab Test High-density lipoprotein (HDL)	Lab Test High-density lipoprotein (HDL), 1 year before the target year
Lab Test Low-density lipoprotein (LDL)	Angiography and Aortography procedures, 4 years before the target year and the rest of the history
Systolic Blood Pressure	Cardiac Catheterization Procedures, 4 years before the target year and the rest of the history
Diagnosis of Heart Failure	Race
Diagnosis of Other Forms of Chronic Ischemic Heart Diseases	Cardiac Dysrhythmias, 1 year before the target year
2 non-significant features in 1-LRT	**2 non-significant features in AdaBoost with Trees**
Sex	Sex
Hypertensive Heart Disease, 1 year before the target year	Hypertensive Heart Disease, 1 year before the target year

Table 3.4: Other significant and non-significant features with 1-LRT and AdaBoost with Trees.

Chapter 4

A Distributed Cluster Primal Dual Splitting Method for Large-Scale Sparse Support Vector Machines

As the volume, variety, velocity and veracity (the four V's) of the clinical data are growing, there is greater need for efficient computational models to mine these data. Insights from these techniques could help design efficient healthcare policies, detect disease causes, provide medical resolutions that are personalized and less costly and finally, improve the quality of hospital care for the patients. We are motivated by problems in the medical domain that can be formulated as binary supervised classification problems and solved using Support Vector Machines; the applications range from prediction of diabetes disease [Kumari and Chitra, 2013] [Yu et al., 2010], prediction of medication adherence in heart failure patients [Son et al., 2010], automated recognition of the obstructive sleep apnea syndrome [Khandoker et al., 2009], to our problem of predicting heart-related hospitalizations. Results in the literature suggest that sparse classifiers, i.e., relying on few informative features, have strong predictive power and generalize well out-of-sample, providing at the same time interpretability in both models and results, which is crucial in order for healthcare practitioners to trust the computed solutions. Another major concern, especially in the medical domain, is the privacy of the data, attracting recent research efforts, including work on the field of differential privacy [Narayanan and Shmatikov, 2008, Dwork, 2011, Brown et al., 2013]. Two well-known examples of privacy breaches are the Net-

flix Prize and the Massachusetts Group Insurance Commission (GIC) medical records database. In both cases, individuals were identified even though the data had been through a de-identification process. This demonstrated that one's identity and other sensitive information could be compromised once a single center has access and processes all data. Especially, under the Precision Medicine Initiative, in the near future, these data could include individuals' genome information, which is too sensitive to be shared.

In this part of the thesis, we are particularly interested in addressing three challenges tied to healthcare data: (1) data reside in different sources (hospitals, health centers, patients' smartphones), which implies the need to build a collective intelligence system; (2) there is a growing availability of data, which makes scalable frameworks rather important; and (3) data privacy is a critical concern. A decentralized computational scheme that collaboratively utilizes all the data in the network, while avoiding centralized data collection and coordination, may meet the requirements.

Notational Conventions: All vectors are assumed to be column vectors. $\mathbf{x}^\intercal$ represents the transpose of $\mathbf{x}$. For any real matrices (vectors) with appropriate dimensions, $\mathbf{A}$ and $\mathbf{B}$, the inner product of them is defined as $\langle \mathbf{A}, \mathbf{B} \rangle = \text{Trace}\{\mathbf{A}^\intercal \mathbf{B}\}$. The Frobenius norm of matrix $\mathbf{B} = (B_{ij})$ is denoted by $\|\mathbf{B}\|_{\text{Fro}} = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |B_{ij}|^2}$. Suppose $\mathbf{M}$ is a real matrix with appropriate dimensions, then an $\mathbf{M}$-weighted norm of $\mathbf{B}$ is defined as $\|\mathbf{B}\|_{\mathbf{M}} = \sqrt{\langle \mathbf{B}, \mathbf{MB} \rangle}$. Suppose $\mathbf{B}$ is also a square matrix, then $\sqrt{\mathbf{B}} = \mathbf{L}\sqrt{\mathbf{S}}\mathbf{R}^\intercal$ where $\mathbf{B} = \mathbf{LSR}^\intercal$ is the singular value decomposition of $\mathbf{B}$. Given an $m \times n$ matrix $\mathbf{A}$ and a $p \times q$ matrix $\mathbf{B}$, their Kronecker product $\mathbf{C} = (C_{ij}) = \mathbf{A} \otimes \mathbf{B}$ is an $(mp) \times (nq)$ matrix with elements defined by $C_{\alpha\beta} = A_{ij} B_{kl}$, where $\alpha = p(i-1) + k$ and $\beta = q(j-1) + l$. The largest eigenvalue of $\mathbf{B}$ is denoted by $\lambda_{\max}\{\mathbf{B}\}$. The m-by-m identity matrix is denoted by $\mathbf{I}_m$. The relation $\mathbf{A} \preccurlyeq \mathbf{B}$ means $(\mathbf{B} - \mathbf{A})$ is positive semidefinite while $\mathbf{A} \prec \mathbf{B}$ means $(\mathbf{B} - \mathbf{A})$ is positive definite ($\succcurlyeq$ and $\succ$

are similarly defined). For any proper, closed, and convex function (could be nonsmooth/nondifferentiable) $f : \mathbb{R}^p \to \mathbb{R}$, we use $\partial f(\mathbf{x})$ to denote the subdifferential of f at $\mathbf{x}$ which is actually a convex set. A subgradient of f at $\mathbf{x}$, denoted as $\widetilde{\nabla} f(\mathbf{x})$, is an element of $\partial f(\mathbf{x})$. Which subgradient we will use in the algorithm will be clear from the context.

Aim: The focus of this section is to solve in a decentralized manner the soft-margin ℓ_1-regularized (sparse) Support Vector Machines (sSVM) problem [Friedman et al., 2001]. Given training data $\boldsymbol{\phi}_i \in \mathbb{R}^d$ and labels $l_i \in \{-1, 1\}$, $i = 1, \ldots, n$, we would like to find the classifier $(\boldsymbol{\beta}, \beta_0)$, $\boldsymbol{\beta} \in \mathbb{R}^d, \beta_0 \in \mathbb{R}$ by solving:

$$\begin{array}{ll} \min\limits_{\boldsymbol{\beta},\beta_0,\xi_i} & 0.5\|\boldsymbol{\beta}\|^2 + C\sum_{i=1}^n \xi_i + \kappa\|\boldsymbol{\beta}\|_1 \\ \text{s. t.} & \xi_i \geq 0, \quad \forall i \\ & l_i(\boldsymbol{\phi}_i^\intercal \boldsymbol{\beta} + \beta_0) \geq 1 - \xi_i, \quad \forall i \end{array} \tag{4.1}$$

where C and κ are tunable parameters. If $\xi_i = 0$, then $1 - l_i(\boldsymbol{\phi}_i^\intercal \boldsymbol{\beta} + \beta_0) \leq 0$ and the corresponding term in the objective function is 0. If $\xi_i > 0$, then $1 - l_i(\boldsymbol{\phi}_i^\intercal \boldsymbol{\beta} + \beta_0) \leq \xi_i$ and minimizing ξ_i in the objective function is equivalent to minimizing $1 - l_i(\boldsymbol{\phi}_i^\intercal \boldsymbol{\beta} + \beta_0)$. Thus, formulation (4.1) is equivalent to the following with the tunable parameters being $\tau = 1/C$ and $\rho = \kappa/C$:

$$\min_{\boldsymbol{\beta},\beta_0} \quad \sum_{i=1}^n h_i(\boldsymbol{\beta}, \beta_0) + 0.5\tau\|\boldsymbol{\beta}\|^2 + \rho\|\boldsymbol{\beta}\|_1 \tag{4.2}$$

where each $h_i(\boldsymbol{\beta}, \beta_0) = [1 - l_i(\boldsymbol{\phi}_i^\intercal \boldsymbol{\beta} + \beta_0)]_+ = \max\{0, 1 - l_i(\boldsymbol{\phi}_i^\intercal \boldsymbol{\beta} + \beta_0)\}$ is a hinge loss function corresponding to sample i and the $\|\boldsymbol{\beta}\|_2$ and $\|\boldsymbol{\beta}\|_1$ terms are regularizers for the model parameters.

In the distributed context we are interested in a setting where each agent[1] holds a part of the data/samples, namely, a subset of $\{\boldsymbol{\phi}_i, \forall i\}$ and $\{l_i, \forall i\}$, and would like to collaborate with the others to obtain a better estimate on $\boldsymbol{\beta}$ and β_0. Due to privacy

[1]We will use the generic term "agent" to represent each data/computation center. The term could refer to institutions, or even individuals.

concerns, these agents are not willing to share their original data with each other or with a processing center. Thus, we will pursue a decentralized solution that avoids raw data exchange.

Related literature: Problem (4.2) involves minimization of the sum of two convex but non-smooth terms that have the form of a sum: $\sum_{i=1}^{n}[1-l_i(\boldsymbol{\phi}_i^\intercal\boldsymbol{\beta}+\beta_0)]_+$ and $\sum_{i=1}^{n}\{0.5\tau_i\|\boldsymbol{\beta}\|^2+\rho_i\|\boldsymbol{\beta}\|_1\}$ where $\sum_i\tau_i=\tau$ and $\sum_i\rho_i=\rho$. When all the data are stored and computations are executed in a centralized unit, we can solve the problem using the interior point (also referred to as barrier) method [Bertsekas, 1999] or the classical subgradient method (SubGD) [Bertsekas, 1999] .

A recent method for solving regularized empirical risk minimization problem that features a master-slave type (star network) distributed computing scheme is proposed in [Zhang and Lin, 2015] (SPDC). For the sSVM problem, such scheme is shown to have an $O(1/k)$ convergence rate[2] with constant step size and supports mini-batch coordinate updates. However, it cannot be performed in a general decentralized network. When the objectives are strongly convex and smooth (though not applicable to sSVM), it is proved to have geometric convergence rate. Another approach with $O(1/\sqrt{k})$ convergence rate that can solve the sSVM and allows a decentralized implementation is the incremental subgradient method (IncrSub). However, IncrSub needs to deploy vanishing step size to reach exact convergence and only works over networks containing a ring structure [Nedic and Bertsekas, 2001]. Besides the above methods that only work in networks with some specific topology, the following optimization schemes can be applied to arbitrary undirected communication networks. The subgradient method introduced in [Nedic and Ozdaglar, 2009] can be directly used for solving the sSVM. Although this algorithm features an elegant update rule, its convergence is typically slow due to the use of diminishing step sizes. A re-

[2]A nonnegative sequence $\{a_k\}$ is said to be convergent to 0 at an $O(1/k)$ rate if $\limsup_{k\to\infty}ka_k<+\infty$. In contrast, it is said to have an $o(1/k)$ rate if $\limsup_{k\to\infty}ka_k=0$.

cent fully decentralized scheme that has made a significant improvement over the subgradient method is the Linear time Average Consensus optimization algorithm (LAC) [Olshevsky, 2014]. The LAC algorithm utilizes a fixed but small step size and is shown to have $O(1/\sqrt{k})$ convergence rate. A good feature of the LAC is that it improves the algorithmic scalability in the size of the network through utilizing Nesterov's acceleration technique [Nesterov, 1983]. There exist other decentralized optimization frameworks that allow the objective to be nondifferentiable, such as the Proximal-EXact first order Algorithm (Proximal-EXTRA) algorithm [Shi et al., 2015b] and the Distributed Alternating Direction Method of Multipliers (DADMM) algorithm [Bertsekas and Tsitsiklis, 1989]. Under strong convexity and smoothness assumptions, these methods are proved to have geometric convergence rate, whereas in the presence of nonsmooth objectives, the convergence rate of these methods is $O(1/k)$. But since these (proximal) methods are not tailored for problems with "nonsmooth+nonsmooth" structure, their per-iteration cost will be rather expensive if we apply them to the sSVM directly.

We list in Table 4.1 comparative results that illustrate the trade-offs between different methods, including our proposed cluster Primal Dual Splitting (cPDS) framework, when applied to the sparse SVM problem.[3]

Table 4.1: Theoretical performance results for all methods for the sSVM problem

Method	Decentralized?	Per iteration complexity	ϵ-accuracy iterations
SubGD	$\times$	$O(nd)$	$O(1/\epsilon^2)$
IncrSub	$\times$	$O(d)$	$O(1/\epsilon^2)$
SPDC	$\times$	$O(nd)$	$O(1/\epsilon)$
LAC	$\surd$	$O(n^2d)$	$O(1/\epsilon^2)$
cPDS	$\surd$	$O((n+m^2)d)$	$o(1/\epsilon)$

[3] "Per iteration complexity" measures how many scalar multiplications are needed per iteration when applied for solving sSVM. "ϵ-accuracy iterations" measure how many iterations are needed to reach ϵ-accuracy.

4.1 The Cluster Primal Dual Splitting Method

In this section, we first introduce the general decentralized primal dual spliting scheme that we have designed for solving "nonsmooth+nonsmooth" optimization problems. Then we provide convergence analysis for the scheme we have proposed.

Let us assume there is a network of agents, each of which is holding part of the data and they all collectively would like to solve (4.2) utilizing all data. We consider two cases: each agent is holding (1) multiple samples (semi-centralized) or (2) one sample (fully-decentralized) of the data. In the healthcare context, for the first scenario agents can be hospitals that process the data of their patients only and exchange messages in order to optimize globally some function, while the second scenario may correspond to each patient having their data stored and processed in their smartphone and messages to be exchanged between patients' phones. In both cases, the m agents are connected through an underlying communication network, which is modeled by an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, 2, \ldots, m\}$ is the vertex set and $\mathcal{E}$ is the edge set. Throughout the paper, we make the following Assumption 1.

Assumption 1. *The graph $\mathcal{G}$ is connected.*

Let $\mathbf{W} = [w_{ij}]$ be a doubly stochastic matrix generated following the Metropolis rule on $\mathcal{G}$, i.e.,

$$w_{ij} = \begin{cases} \frac{1}{\max\{\text{degree}(i),\text{degree}(j)\}+1}, & \text{if } (i,j) \in \mathcal{E}, \\ 0, & \text{if } (i,j) \notin \mathcal{E} \text{ and } i \neq j, \\ 1 - \sum\limits_{k \in \mathcal{V}} w_{ik}, & \text{if } i = j. \end{cases}$$

Such rule allows each agent i to generate $w_{ij}, \forall j$, by only using local information (its own and neighbors' degree information). Note that we always have $-\mathbf{I}_m \prec \mathbf{W} \preccurlyeq \mathbf{I}_m$ [Shi et al., 2015a]. Let us also define $\mathbf{Ł} \triangleq (\mathbf{I}_m - \mathbf{W}) \otimes \mathbf{I}_{d+1}$ and $\mathbf{U} \triangleq \sqrt{\mathbf{Ł}}$. We note that $\mathbf{U}$ has the same null space as $\mathbf{Ł}$.

Clearly, in the decentralized environment, problem (4.2) can be reformulated into

the following m-cluster splitting formulation:

$$\begin{array}{ll} \min\limits_{\boldsymbol{\beta},\beta_0} & \sum\limits_{j=1}^{m}\left\{\sum\limits_{i=1}^{n_j}[1-y_{ji}]_+ + 0.5\tau_j\|\boldsymbol{\beta}_j\|^2 + \rho_j\|\boldsymbol{\beta}_j\|_1\right\} \\ \text{s.t.} & \gamma_{ji}(l_{ji}(\boldsymbol{\phi}_{ji}^{\mathsf{T}}\boldsymbol{\beta}_j + \boldsymbol{\beta}_{j0}) - y_{ji}) = 0, \forall j, i; \\ & \boldsymbol{\beta}_1 = \boldsymbol{\beta}_2 = \ldots = \boldsymbol{\beta}_m; \\ & \boldsymbol{\beta}_{1,0} = \boldsymbol{\beta}_{2,0} = \ldots = \boldsymbol{\beta}_{m0}, \end{array} \tag{4.3}$$

where each agent (hospital) j holds n_j samples (note that $n = \sum_{j=1}^{m} n_j$) and maintains its own copy of the model parameters to be estimated $\mathbf{x}_j = (\boldsymbol{\beta}_j, \beta_{j0}) \in \mathbb{R}^{d+1}$ and τ_j and ρ_j are tunable model parameters. The parameters γ_{ji}'s are arbitrary nonzero scalar constants and will serve as algorithmic parameters later in the designed algorithm. Let us define a long vector $\mathbf{x} = [\mathbf{x}_1; \ldots; \mathbf{x}_m] \in \mathbb{R}^{m(d+1)}$ to compactly represent all the local copies; a long vector variable $\mathbf{y} = [\mathbf{y}_1; \ldots; \mathbf{y}_m] \in \mathbb{R}^n$ where each block $\mathbf{y}_j = [y_{j1}; \ldots; y_{jn_j}] \in \mathbb{R}^{n_j}$ is handled by agent j. We also prepare $\mathbf{q} \in \mathbb{R}^n$ and $\boldsymbol{\lambda} \in \mathbb{R}^{m(d+1)}$ which will serve as auxiliary (dual) variables later in the algorithm. Their blocks are handled in parallel by agents and the operations on them will be clear from the context.

With the above notation, problem (4.3) can be represented in a more compact form as follows:

$$\begin{array}{ll} \min\limits_{\{\mathbf{x}_j,\mathbf{y}_j\},\ \forall j} & \sum\limits_{j=1}^{m}\{\mathbf{g}_j(\mathbf{x}_j) + \mathbf{f}_j(\mathbf{y}_j)\} \\ \text{s.t.} & \boldsymbol{\Gamma}_j(\mathbf{A}_j\mathbf{x}_j - \mathbf{y}_j) = 0,\ \forall j, \\ & \mathbf{x}_1 = \mathbf{x}_2 = \ldots = \mathbf{x}_m, \end{array} \tag{4.4}$$

where the function $\mathbf{f}_j(\mathbf{y}_j)$ contains all the hinge loss functions $\sum\limits_{i=1}^{n_j}[1-y_{ji}]_+$ for agent j, while the function $\mathbf{g}_j(\mathbf{x}_j)$ includes the regularizers $0.5\tau_j\|\boldsymbol{\beta}_j\|^2 + \rho_j\|\boldsymbol{\beta}_j\|_1$ over agent j. Each $\mathbf{A}_j$ contains all the data samples for agent j (c.f. the corresponding first group of constraints in (4.3)). Each matrix $\boldsymbol{\Gamma}_j \in \mathbb{R}^{n_j \times n_j}$ is a diagonal full rank matrix containing $\gamma_{ji}, \forall i$. Each $\boldsymbol{\Gamma}_j$ is locally produced/tuned by agent j to possibly precondition/scale $\mathbf{A}_j$ so as to achieve better performance. To solve the cluster sSVM, or its more general form (4.4), in a decentralized fashion, we propose Algorithm 1.

Algorithm 1 Cluster PDS Method

Input: $\forall j$, Prepare data/objectives $\mathbf{f}_j$ and $\mathbf{g}_j$; Set parameters $\mathbf{\Gamma}_j$ and $\mathbf{\Theta}_j$.
Initialize: $\forall j$, $\mathbf{x}_j^0 \in \mathbb{R}^{d+1}$, $\mathbf{y}_j^0 \in \mathbb{R}^{n_j}$, $\mathbf{q}_j^{-1} = \mathbf{0}$, $\mathbf{q}_j^0 = \mathbf{\Gamma}_j(\mathbf{A}_j^\intercal \mathbf{x}_j^0 - \mathbf{y}_j^0)$, $\boldsymbol{\lambda}_j^{-1} = \mathbf{0}$, and $\boldsymbol{\lambda}_j^0 = \mathbf{x}_j^0 - \sum_{i\in\mathcal{N}_j\cup\{j\}} w_{ji}\mathbf{x}_i^0$.
repeat

x-update (locally): $\forall j$

$$\begin{aligned}\mathbf{x}_j^{k+1} = \ & \arg\min_{\mathbf{x}_j} \Big\{ (2\mathbf{q}_j^k - \mathbf{q}_j^{k-1})^T(\mathbf{\Gamma}_j\mathbf{A}_j\mathbf{x}_j) + \mathbf{g}_j(\mathbf{x}_j) \\ & +(2\boldsymbol{\lambda}_j^k - \boldsymbol{\lambda}_j^{k-1})^T\mathbf{x}_j + 0.5\|\mathbf{x}_j - \mathbf{x}_j^k\|^2_{\mathbf{\Theta}_j} \Big\}\end{aligned}$$

y-update (locally): $\forall j$

$$\mathbf{y}_j^{k+1} = \ \arg\min_{\mathbf{y}_j} \Big\{ \mathbf{f}_j(\mathbf{y}_j) + (\mathbf{q}_j^k)^T(-\mathbf{\Gamma}_j\mathbf{y}_j) + 0.5\|\mathbf{y}_j - \mathbf{A}_j\mathbf{x}_j^{k+1}\|^2_{\mathbf{\Gamma}_j^\intercal\mathbf{\Gamma}_j} \Big\}$$

q-update (locally): $\forall j$

$$\mathbf{q}_j^{k+1} = \mathbf{q}_j^k + \mathbf{\Gamma}_j(\mathbf{A}_j\mathbf{x}_j^{k+1} - \mathbf{y}_j^{k+1})$$

$\boldsymbol{\lambda}$-update (requires information exchange): $\forall j$

$$\boldsymbol{\lambda}_j^{k+1} = \boldsymbol{\lambda}_j^k + \mathbf{x}_j^{k+1} - \sum_{i\in\mathcal{N}_j\cup\{j\}} w_{ji}\mathbf{x}_i^{k+1}$$

until specific criteria are met.

In the algorithm, $\mathbf{\Theta}_j \in \mathbb{R}^{(d+1)\times(d+1)}$ is a diagonal positive definite matrix that serves as a part of the algorithmic parameters maintained by agent j. Later after the analysis, we will remark that the algorithmic parameters $\mathbf{\Gamma}_j$ and $\mathbf{\Theta}_j$ can be determined by agent j fully locally and independently from the network topology.

To facilitate our convergence analysis, let us further write (4.4) into an even more compact form:

$$\begin{array}{ll}\min\limits_{\mathbf{x},\mathbf{y}} & \{\mathbf{g}(\mathbf{x}) + \mathbf{f}(\mathbf{y})\} \\ \text{s.t.} & \mathbf{\Gamma}(\mathbf{A}\mathbf{x} - \mathbf{y}) = 0, \\ & \mathbf{U}\mathbf{x} = 0,\end{array} \tag{4.5}$$

where $\mathbf{g}(\mathbf{x}) = \sum_{j=1}^{m} \mathbf{g}_j(\mathbf{x}_j)$, $\mathbf{f}(\mathbf{y}) = \sum_{j=1}^{m} \mathbf{f}_j(\mathbf{y}_j)$ and $\mathbf{\Gamma}$, $\mathbf{A}$ are the matrices/tensors that contain $\mathbf{\Gamma}_j$ and $\mathbf{A}_j$ correspondingly $\forall j = 1, \ldots, m$. We note that $\mathbf{U}\mathbf{x} = 0$, where $\mathbf{U} \triangleq \sqrt{\mathbf{L}}$, is equivalent to $\mathbf{x}_1 = \mathbf{x}_2 = \ldots = \mathbf{x}_m$, as long as the graph is connected, since the null space of $\mathbf{U}$ is $\{\mathbf{x} = [\mathbf{x}_1; \mathbf{x}_2; \ldots; \mathbf{x}_m] \in \mathbb{R}^{m(d+1)} \big| \mathbf{x}_1 = \mathbf{x}_2 = \ldots = \mathbf{x}_m\}$.

Before starting the convergence analysis, let us make two more basic assumptions.

Assumption 2. *The functions* $\mathbf{g} : \mathbb{R}^{m(d+1)} \to \mathbb{R}$ *and* $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}$ *are both proper,*[4] *closed, and convex.*

Assumption 3. *The solution set* χ^* *of (4.5) is nonempty and bounded.*

Assumption 2 imposes a minimal requirement on the objectives to conduct convex analysis. Assumption 3 is obviously satisfied by the sSVM problem.

The augmented Lagrangian function of (4.5) is as follows

$$\mathcal{L} = \mathbf{g}(\mathbf{x}) + \mathbf{f}(\mathbf{y}) + \langle \mathbf{r}, \mathbf{U}\mathbf{x} \rangle + \langle \mathbf{q}, \mathbf{\Gamma}(\mathbf{A}\mathbf{x} - \mathbf{y}) \rangle \tag{4.6}$$

where $\mathbf{r} \in \mathbb{R}^{m(d+1)}$ and $\mathbf{q} \in \mathbb{R}^n$ contain the dual variables. The first-order conditions for the saddle point of (4.6) are

$$\widetilde{\nabla}\mathbf{g}(\mathbf{x}^*) + \mathbf{U}\mathbf{r}^* + A^\intercal \mathbf{\Gamma}^\intercal \mathbf{q}^* = 0; \tag{4.7a}$$

$$\mathbf{U}\mathbf{x}^* = 0; \tag{4.7b}$$

$$\widetilde{\nabla}\mathbf{f}(\mathbf{y}^*) - \mathbf{\Gamma}^\intercal \mathbf{q}^* = 0; \tag{4.7c}$$

$$\mathbf{A}\mathbf{x}^* - \mathbf{y}^* = 0. \tag{4.7d}$$

Clearly, $\mathbf{x}^* \in \chi^*$. The update relations of Algorithm 1 can be recast into the following:

[4] A function $f : X \to Y$ is proper if $\exists\, \mathbf{x}$ in the domain X of f such that $f(\mathbf{x})$ is finite.

$$\widetilde{\nabla}\mathbf{g}(\mathbf{x}^{k+1}) + \mathbf{A}^\intercal\boldsymbol{\Gamma}^\intercal(2\mathbf{q}^k - \mathbf{q}^{k-1}) + \mathbf{U}(2\mathbf{r}^k - \mathbf{r}^{k-1}) + \boldsymbol{\Theta}(\mathbf{x}^{k+1} - \mathbf{x}^k) = 0; \tag{4.8a}$$

$$\widetilde{\nabla}\mathbf{f}(\mathbf{y}^{k+1}) - \boldsymbol{\Gamma}^\intercal\mathbf{r}^k + \boldsymbol{\Gamma}^\intercal\boldsymbol{\Gamma}(\mathbf{y}^{k+1} - \mathbf{A}\mathbf{x}^{k+1}) = 0; \tag{4.8b}$$

$$\mathbf{q}^{k+1} = \mathbf{q}^k + \boldsymbol{\Gamma}(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{y}^{k+1}); \tag{4.8c}$$

$$\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + \text{Ł}\mathbf{x}^{k+1}. \tag{4.8d}$$

By reorganisation of the updates and by using the optimality conditions, we derive the following Lemma.

Lemma 1. *The recursion of the proposed algorithm obeys*

$$(\boldsymbol{\Theta} - \mathbf{A}^\intercal\boldsymbol{\Gamma}^\intercal\boldsymbol{\Gamma}\mathbf{A} - \text{Ł})(\mathbf{x}^{k+1} - \mathbf{x}^k) + \mathbf{A}^\intercal\boldsymbol{\Gamma}^\intercal\boldsymbol{\Gamma}(\mathbf{y}^{k+1} - \mathbf{y}^k) = -\widetilde{\nabla}\mathbf{g}(\mathbf{x}^{k+1}) - \mathbf{U}\mathbf{r}^{k+1} - \mathbf{A}^\intercal\boldsymbol{\Gamma}^\intercal\mathbf{q}^{k+1}; \tag{4.9a}$$

$$(\boldsymbol{\Theta} - \mathbf{A}^\intercal\boldsymbol{\Gamma}^\intercal\boldsymbol{\Gamma}\mathbf{A} - \text{Ł})(\mathbf{x}^{k+1} - \mathbf{x}^k) + \mathbf{A}^\intercal\boldsymbol{\Gamma}^\intercal\boldsymbol{\Gamma}(\mathbf{y}^{k+1} - \mathbf{y}^k) = \widetilde{\nabla}\mathbf{g}(\mathbf{x}^*) - \widetilde{\nabla}\mathbf{g}(\mathbf{x}^{k+1}) + \mathbf{U}(\mathbf{r}^* - \mathbf{r}^{k+1}) + \mathbf{A}^\intercal\boldsymbol{\Gamma}^\intercal(\mathbf{q}^* - \mathbf{q}^{k+1}); \tag{4.9b}$$

$$\widetilde{\nabla}\mathbf{f}(\mathbf{y}^{k+1}) - \boldsymbol{\Gamma}^\intercal\mathbf{q}^{k+1} = 0; \tag{4.9c}$$

$$\widetilde{\nabla}\mathbf{f}(\mathbf{y}^{k+1}) - \widetilde{\nabla}\mathbf{f}(\mathbf{y}^*) - \boldsymbol{\Gamma}^\intercal(\mathbf{q}^{k+1} - \mathbf{q}^*) = 0; \tag{4.9d}$$

$$\mathbf{q}^{k+1} = \mathbf{q}^k + \boldsymbol{\Gamma}(\mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^*) - (\mathbf{y}^{k+1} - \mathbf{y}^*)); \tag{4.9e}$$

$$\mathbf{r}^{k+1} = \mathbf{r}^k + \mathbf{U}\mathbf{x}^{k+1}; \tag{4.9f}$$

$$\boldsymbol{\lambda}^k = \mathbf{U}\mathbf{r}^k;$$

$$\mathbf{r}^{k+1} = \mathbf{r}^k + \mathbf{U}(\mathbf{x}^{k+1} - \mathbf{x}^*). \tag{4.9g}$$

Before stating our main results, let us define a long vector $\mathbf{z}^k \triangleq \left[\mathbf{x}^k; \mathbf{y}^k; \mathbf{r}^k; \mathbf{q}^k\right] \in \mathbb{R}^{2m(d+1)+2n}$ and a $(2m(d+1)+2n) \times (2m(d+1)+2n)$ block diagonal matrix[5] $\mathbf{M} \triangleq \text{blkdiag}\{\boldsymbol{\Theta} - \mathbf{A}^\intercal\boldsymbol{\Gamma}^\intercal\boldsymbol{\Gamma}\mathbf{A} - \text{Ł}, \boldsymbol{\Gamma}^\intercal\boldsymbol{\Gamma}, \mathbf{I}_{m(d+1)}, \mathbf{I}_n\}$. Correspondingly, we also define $\mathbf{z}^* \triangleq$

[5]We use blkdiag(A,B,C,D), where A, B, C, D are matrices, to denote a block diagonal matrix of the form $\begin{bmatrix} A & 0 & 0 & 0 \\ 0 & B & 0 & 0 \\ 0 & 0 & C & 0 \\ 0 & 0 & 0 & D \end{bmatrix}$.

$[\mathbf{x}^*; \mathbf{y}^*; \mathbf{r}^*; \mathbf{q}^*] \in \mathbb{R}^{2m(d+1)+2n}$.

Theorem 1 (Convergence)**.** *If the scaling parameters are chosen such that*

$$\boldsymbol{\Theta} - \mathbf{A}^\intercal \boldsymbol{\Gamma}^\intercal \boldsymbol{\Gamma} \mathbf{A} - \boldsymbol{Ł} \succ 0, \tag{4.10}$$

then $\|\widetilde{\nabla}\mathbf{f}(\mathbf{y}^{k+1}) - \boldsymbol{\Gamma}^\intercal \mathbf{q}^{k+1}\|_{\text{Fro}}^2 = 0$ *and the sequences* $\{\|\mathbf{z}^k - \mathbf{z}^{k+1}\|_{\mathbf{M}}^2\}$, $\{\|\widetilde{\nabla}\mathbf{g}(\mathbf{x}^{k+1}) + \mathbf{U}\mathbf{r}^{k+1} + \mathbf{A}^\intercal \boldsymbol{\Gamma}^\intercal \mathbf{q}^{k+1}\|_{\text{Fro}}^2\}$, $\{\|\boldsymbol{\Gamma}(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{y}^{k+1})\|_{\text{Fro}}^2\}$, *and* $\{\|\mathbf{U}\mathbf{x}^{k+1}\|_{\text{Fro}}^2\}$ *are all infinitely summable over* $k \geq 0$*. Consequently, the sequence* $\{\mathbf{x}^{k+1}\}$ *generated by the proposed algorithm converges to a consensual*[6] *and optimal solution of problem* (4.5) *(*$\mathbf{x}$ *part).*

Proof. By the convexity of $\mathbf{g}$, we have

$$0 \leq 2\langle \mathbf{x}^{k+1} - \mathbf{x}^*, \widetilde{\nabla}\mathbf{g}(\mathbf{x}^{k+1}) - \widetilde{\nabla}\mathbf{g}(\mathbf{x}^*)\rangle. \tag{4.11}$$

Substituting (4.9b) into the second term of the right-hand-side of (4.11) for $(\widetilde{\nabla}\mathbf{g}(\mathbf{x}^{k+1}) - \widetilde{\nabla}\mathbf{g}(\mathbf{x}^*))$ gives

$$\begin{array}{rl} & 0 \\ \leq & 2\langle \mathbf{x}^{k+1} - \mathbf{x}^*, \widetilde{\nabla}\mathbf{g}(\mathbf{x}^{k+1}) - \widetilde{\nabla}\mathbf{g}(\mathbf{x}^*)\rangle \\ = & 2\langle \mathbf{x}^{k+1} - \mathbf{x}^*, -(\boldsymbol{\Theta} - \mathbf{A}^\intercal \boldsymbol{\Gamma}^\intercal \boldsymbol{\Gamma} \mathbf{A} - Ł)(\mathbf{x}^{k+1} - \mathbf{x}^k) \\ & -\mathbf{U}(\mathbf{r}^{k+1} - \mathbf{r}^*)\rangle + 2\langle \mathbf{x}^{k+1} - \mathbf{x}^*, \\ & -\mathbf{A}^\intercal \boldsymbol{\Gamma}^\intercal \boldsymbol{\Gamma}(\mathbf{y}^{k+1} - \mathbf{y}^k) - \mathbf{A}^\intercal \boldsymbol{\Gamma}^\intercal (\mathbf{q}^{k+1} - \mathbf{q}^*)\rangle \\ = & 2\langle \mathbf{x}^{k+1} - \mathbf{x}^*, (\boldsymbol{\Theta} - \mathbf{A}^\intercal \boldsymbol{\Gamma}^\intercal \boldsymbol{\Gamma} \mathbf{A} - Ł)(\mathbf{x}^k - \mathbf{x}^{k+1})\rangle \\ & +2\langle \mathbf{U}(\mathbf{x}^{k+1} - \mathbf{x}^*), \mathbf{r}^* - \mathbf{r}^{k+1}\rangle \\ & +2\langle \mathbf{x}^{k+1} - \mathbf{x}^*, \mathbf{A}^\intercal \boldsymbol{\Gamma}^\intercal \boldsymbol{\Gamma}(\mathbf{y}^k - \mathbf{y}^{k+1})\rangle \\ & +2\langle \mathbf{x}^{k+1} - \mathbf{x}^*, \mathbf{A}^\intercal \boldsymbol{\Gamma}^\intercal (\mathbf{q}^* - \mathbf{q}^{k+1})\rangle. \end{array} \tag{4.12}$$

Next, let us look into the four terms at the right-hand-side of (4.12). The first term has an appropriate form; For the second term, by (4.9g), we have

$$\begin{array}{rl} & 2\langle \mathbf{U}(\mathbf{x}^{k+1} - \mathbf{x}^*), \mathbf{r}^* - \mathbf{r}^{k+1}\rangle \\ = & 2\langle \mathbf{r}^{k+1} - \mathbf{r}^k, \mathbf{r}^* - \mathbf{r}^{k+1}\rangle; \end{array} \tag{4.13}$$

For the third term, we have

$$\begin{array}{rl} & 2\langle \mathbf{x}^{k+1} - \mathbf{x}^*, \mathbf{A}^\intercal \boldsymbol{\Gamma}^\intercal \boldsymbol{\Gamma}(\mathbf{y}^k - \mathbf{y}^{k+1})\rangle \text{ (by (4.9e))} \\ = & 2\langle \boldsymbol{\Gamma}(\mathbf{y}^{k+1} - \mathbf{y}^*) + \mathbf{q}^{k+1} - \mathbf{q}^k, \boldsymbol{\Gamma}(\mathbf{y}^k - \mathbf{y}^{k+1})\rangle \text{ (by (4.9c))} \\ = & 2\langle \mathbf{y}^{k+1} - \mathbf{y}^*, \boldsymbol{\Gamma}^\intercal \boldsymbol{\Gamma}(\mathbf{y}^k - \mathbf{y}^{k+1})\rangle \\ & -2\langle \widetilde{\nabla}\mathbf{f}(\mathbf{y}^k) - \widetilde{\nabla}\mathbf{f}(\mathbf{y}^{k+1}), \mathbf{y}^k - \mathbf{y}^{k+1}\rangle \\ & \text{(by the convexity of } \mathbf{f}) \\ \leq & 2\langle \mathbf{y}^{k+1} - \mathbf{y}^*, \boldsymbol{\Gamma}^\intercal \boldsymbol{\Gamma}(\mathbf{y}^k - \mathbf{y}^{k+1})\rangle; \end{array} \tag{4.14}$$

[6] All rows of $\mathbf{x}$ are equal to each other.

For the last term, we have

$$\begin{array}{rl} & 2\langle \mathbf{x}^{k+1}-\mathbf{x}^*, \mathbf{A}^\intercal\boldsymbol{\Gamma}^\intercal(\mathbf{q}^*-\mathbf{q}^{k+1})\rangle \text{ (by (4.9e))} \\ = & 2\langle \boldsymbol{\Gamma}(\mathbf{y}^{k+1}-\mathbf{y}^*)+\mathbf{q}^{k+1}-\mathbf{q}^k, \mathbf{q}^*-\mathbf{q}^{k+1}\rangle \text{ (by (4.9d))} \\ = & -2\langle \mathbf{y}^{k+1}-\mathbf{y}^*, \widetilde{\nabla}\mathbf{f}(\mathbf{y}^{k+1})-\widetilde{\nabla}\mathbf{f}(\mathbf{y}^*)\rangle + 2\langle \mathbf{q}^{k+1}-\mathbf{q}^k, \\ & \mathbf{q}^*-\mathbf{q}^{k+1}\rangle \text{ (by the convexity of } \mathbf{f}) \\ \leq & 2\langle \mathbf{q}^{k+1}-\mathbf{q}^k, \mathbf{q}^*-\mathbf{q}^{k+1}\rangle. \end{array} \tag{4.15}$$

Combining (4.12)–(4.15) and using the definition of $\mathbf{M}$ and $\mathbf{z}$, we get

$$\begin{array}{rl} & 0 \\ \leq & 2\langle \mathbf{x}^{k+1}-\mathbf{x}^*, (\boldsymbol{\Theta}-\mathbf{A}^\intercal\boldsymbol{\Gamma}^\intercal\boldsymbol{\Gamma}\mathbf{A}-\boldsymbol{Ł})(\mathbf{x}^k-\mathbf{x}^{k+1})\rangle \\ & +2\langle \mathbf{y}^{k+1}-\mathbf{y}^*, \boldsymbol{\Gamma}^\intercal\boldsymbol{\Gamma}(\mathbf{y}^k-\mathbf{y}^{k+1})\rangle \\ & +2\langle \mathbf{r}^{k+1}-\mathbf{r}^k, \mathbf{r}^*-\mathbf{r}^{k+1}\rangle \\ & +2\langle \mathbf{q}^{k+1}-\mathbf{q}^k, \mathbf{q}^*-\mathbf{q}^{k+1}\rangle \\ = & 2\langle \mathbf{z}^{k+1}-\mathbf{z}^*, \mathbf{M}(\mathbf{z}^k-\mathbf{z}^{k+1})\rangle \\ = & \|\mathbf{z}^k-\mathbf{z}^*\|_{\mathbf{M}}^2-\|\mathbf{z}^{k+1}-\mathbf{z}^*\|_{\mathbf{M}}^2-\|\mathbf{z}^k-\mathbf{z}^{k+1}\|_{\mathbf{M}}^2. \end{array} \tag{4.16}$$

It shows from (4.16) that $\|\mathbf{z}^k-\mathbf{z}^*\|_{\mathbf{M}}^2$ is monotonically non-increasing and thus is bounded. Since $\mathbf{z}^0$ and $\mathbf{z}^*$ are bounded, we also have that $\mathbf{z}^k$ is bounded and thus $\|\mathbf{z}^k-\mathbf{z}^{k+1}\|_{\mathbf{M}}^2$ is bounded. It then follows from (4.16) that

$$\begin{array}{rl} & \sum_{k=0}^{\infty}\|\mathbf{z}^k-\mathbf{z}^{k+1}\|_{\mathbf{M}}^2 \\ \leq & \sum_{k=0}^{\infty}(\|\mathbf{z}^k-\mathbf{z}^*\|_{\mathbf{M}}^2-\|\mathbf{z}^{k+1}-\mathbf{z}^*\|_{\mathbf{M}}^2) \\ & \text{(Telescoping cancellation)} \\ = & \|\mathbf{z}^0-\mathbf{z}^*\|_{\mathbf{M}}^2-\|\mathbf{z}^\infty-\mathbf{z}^*\|_{\mathbf{M}}^2<+\infty, \end{array} \tag{4.17}$$

thus, we conclude that $\{\|\mathbf{z}^k-\mathbf{z}^{k+1}\|_{\mathbf{M}}^2\}$ is infinitely summable over k thus

$$\lim_{k\to+\infty}\|\mathbf{z}^k-\mathbf{z}^{k+1}\|_{\mathbf{M}}^2=0. \tag{4.18}$$

Below we will show that $\mathbf{z}^k$ converges to a point $\mathbf{z}^\infty$ that satisfies the first-order optimality condition. Combining (4.9f), (4.17), and (4.18) gives that $\{\|\mathbf{U}\mathbf{x}^k\|_{\text{Fro}}^2\}$ is infinitely summable and

$$\lim_{k\to+\infty}\|\mathbf{U}\mathbf{x}^{k+1}\|_{\text{Fro}}^2=\lim_{k\to+\infty}\|\mathbf{r}^k-\mathbf{r}^{k+1}\|_{\text{Fro}}^2=0; \tag{4.19}$$

Combining (4.8c), (4.17), and (4.18) gives that $\{\|\boldsymbol{\Gamma}(\mathbf{A}\mathbf{x}^k-\mathbf{y}^k)\|_{\text{Fro}}^2\}$ is infinitely

summable and

$$\begin{array}{rl} & \lim_{k\to+\infty} \|\mathbf{\Gamma}(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{y}^{k+1})\|_{\mathrm{Fro}}^2 \\ = & \lim_{k\to+\infty} \|\mathbf{q}^{k+1} - \mathbf{q}^k\|_{\mathrm{Fro}}^2 = 0; \end{array} \tag{4.20}$$

Combining (4.9a), (4.17), and (4.18) gives that $\{\|\widetilde{\nabla}\mathbf{g}(\mathbf{x}^k) + \mathbf{U}\mathbf{r}^k + \mathbf{A}^\intercal\mathbf{\Gamma}^\intercal\mathbf{q}^k\|_{\mathrm{Fro}}^2\}$ is infinitely summable and

$$\begin{array}{rl} & \lim_{k\to+\infty} \|\widetilde{\nabla}\mathbf{g}(\mathbf{x}^{k+1}) + \mathbf{U}\mathbf{r}^{k+1} + \mathbf{A}^\intercal\mathbf{\Gamma}^\intercal\mathbf{q}^{k+1}\|_{\mathrm{Fro}}^2 \\ = & \lim_{k\to+\infty} \|(\mathbf{\Theta} - \mathbf{A}^\intercal\mathbf{\Gamma}^\intercal\mathbf{\Gamma}\mathbf{A} - \mathbf{Ł})(\mathbf{x}^{k+1} - \mathbf{x}^k) \\ & +\mathbf{A}^\intercal\mathbf{\Gamma}^\intercal\mathbf{\Gamma}(\mathbf{y}^{k+1} - \mathbf{y}^k)\|_{\mathrm{Fro}}^2 \\ \leq & \lim_{k\to+\infty} 2(\lambda_{\max}\{\mathbf{\Theta}^\intercal\mathbf{\Theta}\} \\ & +\lambda_{\max}\{\mathbf{\Gamma}\mathbf{A}\mathbf{A}^\intercal\mathbf{\Gamma}\})\|\mathbf{z}^k - \mathbf{z}^{k+1}\|_{\mathbf{M}}^2 \\ = & 0; \end{array} \tag{4.21}$$

The relation (4.9c) ensures that we always have

$$\|\widetilde{\nabla}\mathbf{f}(\mathbf{y}^{k+1}) - \mathbf{\Gamma}^\intercal\mathbf{q}^{k+1}\|_{\mathrm{Fro}}^2 = 0 \tag{4.22}$$

By comparing (4.19)–(4.22) with the first-order optimality condition (4.7), we can see that the limit point $\mathbf{z}^\infty$ satisfies the KKT system (4.7) thus is an optimal solution to the KKT system. Finally, we conclude that all blocks of $\mathbf{x}^\infty$ are equal to each other and every block of $\mathbf{x}^\infty$ is an optimal solution to problem (4.5) ($\mathbf{x}$ part). This concludes the proof. □

In addition to the infinite summability of $\{\|\mathbf{z}^k - \mathbf{z}^{k+1}\|_{\mathbf{M}}^2\}$ shown above, in the following Lemma 2, we will show that this sequence is also monotonic non-increasing.

Lemma 2 (Monotonic successive difference)**.** *Under the same settings as those in Theorem 1, the sequence* $\{\mathbf{z}^k\}$ *generated by the proposed algorithm satisfies*

$$\|\mathbf{z}^{k+1} - \mathbf{z}^{k+2}\|_{\mathbf{M}}^2 \leq \|\mathbf{z}^k - \mathbf{z}^{k+1}\|_{\mathbf{M}}^2, \tag{4.23}$$

for any $k = 0, 1, \ldots$.

Proof. To ease the description of the proof, let us define $\Delta\mathbf{x}^{k+1} \triangleq \mathbf{x}^k - \mathbf{x}^{k+1}$, $\Delta\mathbf{r}^{k+1} \triangleq \mathbf{r}^k - \mathbf{r}^{k+1}$, $\Delta\mathbf{y}^{k+1} \triangleq \mathbf{y}^k - \mathbf{y}^{k+1}$, $\Delta\mathbf{q}^{k+1} \triangleq \mathbf{q}^k - \mathbf{q}^{k+1}$, $\Delta\mathbf{z}^{k+1} \triangleq \mathbf{z}^k - \mathbf{z}^{k+1}$, $\Delta\widetilde{\nabla}\mathbf{g}(\mathbf{x}^{k+1}) \triangleq \widetilde{\nabla}\mathbf{g}(\mathbf{x}^k) - \widetilde{\nabla}\mathbf{g}(\mathbf{x}^{k+1})$, and $\Delta\widetilde{\nabla}\mathbf{f}(\mathbf{y}^{k+1}) \triangleq \widetilde{\nabla}\mathbf{f}(\mathbf{y}^k) - \widetilde{\nabla}\mathbf{f}(\mathbf{y}^{k+1})$.

By the convexity of $\widetilde{\nabla}\mathbf{g}$, we have

$$0 \leq 2\langle\Delta\mathbf{x}^{k+1}, \Delta\widetilde{\nabla}\mathbf{g}(\mathbf{x}^{k+1})\rangle. \tag{4.24}$$

Taking the the successive difference of (4.9a) gives

$$\begin{array}{rl} & (\mathbf{\Theta} - \mathbf{A}^\intercal\mathbf{\Gamma}^\intercal\mathbf{\Gamma}\mathbf{A} - \mathbf{Ł})(\Delta\mathbf{x}^{k+1} - \Delta\mathbf{x}^k) \\ & +\mathbf{A}^\intercal\mathbf{\Gamma}^\intercal\mathbf{\Gamma}(\Delta\mathbf{y}^{k+1} - \Delta\mathbf{y}^k) \\ = & -\Delta\widetilde{\nabla}\mathbf{g}(\mathbf{x}^{k+1}) - \mathbf{U}\Delta\mathbf{r}^{k+1} - \mathbf{A}^\intercal\mathbf{\Gamma}^\intercal\Delta\mathbf{q}^{k+1}. \end{array} \tag{4.25}$$

Substituting (4.25) into the second term of the right-hand-side of (4.24) for $\Delta\widetilde{\nabla}\mathbf{g}(\mathbf{x}^{k+1})$ yields

$$\begin{array}{rl} & 0 \\ \leq & 2\langle\Delta\mathbf{x}^{k+1}, -(\mathbf{\Theta} - \mathbf{A}^\intercal\mathbf{\Gamma}^\intercal\mathbf{\Gamma}\mathbf{A} - \mathbf{Ł})(\Delta\mathbf{x}^{k+1} - \Delta\mathbf{x}^k)\rangle \\ & +2\langle\Delta\mathbf{x}^{k+1}, -\mathbf{A}^\intercal\mathbf{\Gamma}^\intercal\mathbf{\Gamma}(\Delta\mathbf{y}^{k+1} - \Delta\mathbf{y}^k)\rangle \\ & +2\langle\mathbf{U}\Delta\mathbf{x}^{k+1}, -\Delta\mathbf{r}^{k+1}\rangle + 2\langle\Delta\mathbf{x}^{k+1}, -\mathbf{A}^\intercal\mathbf{\Gamma}^\intercal\Delta\mathbf{q}^{k+1}\rangle. \end{array} \tag{4.26}$$

Next, let us look into the four terms at the right-hand-side of (4.26). The first term has an appropriate form; For the second term, we have

$$\begin{array}{rl} & 2\langle\Delta\mathbf{x}^{k+1}, -\mathbf{A}^\intercal\mathbf{\Gamma}^\intercal\mathbf{\Gamma}(\Delta\mathbf{y}^{k+1} - \Delta\mathbf{y}^k)\rangle \text{ (by (4.8c))} \\ = & 2\langle\mathbf{\Gamma}\Delta\mathbf{y}^{k+1} + \Delta\mathbf{q}^{k+1} - \Delta\mathbf{q}^k, -\mathbf{\Gamma}(\Delta\mathbf{y}^{k+1} - \Delta\mathbf{y}^k)\rangle \\ = & 2\langle\Delta\mathbf{y}^{k+1}, -\mathbf{\Gamma}^\intercal\mathbf{\Gamma}(\Delta\mathbf{y}^{k+1} - \Delta\mathbf{y}^k)\rangle \\ & -2\langle\Delta\mathbf{q}^{k+1} - \Delta\mathbf{q}^k, \mathbf{\Gamma}(\Delta\mathbf{y}^{k+1} - \Delta\mathbf{y}^k)\rangle. \end{array} \tag{4.27}$$

For the third term, by (4.9f), we have

$$2\langle\mathbf{U}\Delta\mathbf{x}^{k+1}, -\Delta\mathbf{r}^{k+1}\rangle = 2\langle\Delta\mathbf{r}^{k+1} - \Delta\mathbf{r}^k, -\Delta\mathbf{r}^{k+1}\rangle. \tag{4.28}$$

For the last term, we have

$$\begin{array}{rl} & 2\langle\Delta\mathbf{x}^{k+1}, -\mathbf{A}^\intercal\mathbf{\Gamma}^\intercal\Delta\mathbf{q}^{k+1}\rangle \text{ (by (4.8c))} \\ = & 2\langle\mathbf{\Gamma}\Delta\mathbf{y}^{k+1} + \Delta\mathbf{q}^{k+1} - \Delta\mathbf{q}^k, -\Delta\mathbf{q}^{k+1}\rangle \text{ (by (4.9c))} \\ = & -2\langle\mathbf{y}^k - \mathbf{y}^{k+1}, \tilde{\nabla}\mathbf{f}(\mathbf{y}^k) - \tilde{\nabla}\mathbf{f}(\mathbf{y}^{k+1})\rangle \\ & +2\langle\Delta\mathbf{q}^{k+1} - \Delta\mathbf{q}^k, -\Delta\mathbf{q}^{k+1}\rangle \text{ (by the convexity of } \mathbf{f}) \\ \leq & 2\langle\Delta\mathbf{q}^{k+1} - \Delta\mathbf{q}^k, -\Delta\mathbf{q}^{k+1}\rangle. \end{array} \tag{4.29}$$

Combining (4.26)–(4.29) gives

$$\begin{array}{rl} 0 \leq & 2\langle\Delta\mathbf{x}^{k+1}, -(\mathbf{\Theta} - \mathbf{A}^\intercal\mathbf{\Gamma}^\intercal\mathbf{\Gamma}\mathbf{A} - \mathbf{Ł})(\Delta\mathbf{x}^{k+1} - \Delta\mathbf{x}^k)\rangle \\ & +2\langle\Delta\mathbf{y}^{k+1}, -\mathbf{\Gamma}^\intercal\mathbf{\Gamma}(\Delta\mathbf{y}^{k+1} - \Delta\mathbf{y}^k)\rangle \\ & -2\langle\Delta\mathbf{q}^{k+1} - \Delta\mathbf{q}^k, \mathbf{\Gamma}(\Delta\mathbf{y}^{k+1} - \Delta\mathbf{y}^k)\rangle \\ & +2\langle\Delta\mathbf{r}^{k+1} - \Delta\mathbf{r}^k, -\Delta\mathbf{r}^{k+1}\rangle \\ & +2\langle\Delta\mathbf{q}^{k+1} - \Delta\mathbf{q}^k, -\Delta\mathbf{q}^{k+1}\rangle \\ = & 2\langle\Delta\mathbf{z}^k - \Delta\mathbf{z}^{k+1}, \mathbf{M}\Delta\mathbf{z}^{k+1}\rangle \\ & -2\langle\Delta\mathbf{q}^{k+1} - \Delta\mathbf{q}^k, \mathbf{\Gamma}(\Delta\mathbf{y}^{k+1} - \Delta\mathbf{y}^k)\rangle \\ = & \|\Delta\mathbf{z}^k\|^2_{\mathbf{M}} - \|\Delta\mathbf{z}^{k+1}\|^2_{\mathbf{M}} - \|\Delta\mathbf{z}^k - \Delta\mathbf{z}^{k+1}\|^2_{\mathbf{M}} \\ & -2\langle\Delta\mathbf{q}^{k+1} - \Delta\mathbf{q}^k, \Delta\mathbf{y}^{k+1} - \Delta\mathbf{y}^k\rangle, \end{array} \tag{4.30}$$

and thus we eventually have

$$\begin{array}{rl} & \|\Delta\mathbf{z}^k\|_{\mathbf{M}}^2 - \|\Delta\mathbf{z}^{k+1}\|_{\mathbf{M}}^2 \\ \geq & \|\Delta\mathbf{z}^k - \Delta\mathbf{z}^{k+1}\|_{\mathbf{M}}^2 \\ & +2\langle\Delta\mathbf{q}^{k+1} - \Delta\mathbf{q}^k, \boldsymbol{\Gamma}(\Delta\mathbf{y}^{k+1} - \Delta\mathbf{y}^k)\rangle \\ = & \|\Delta\mathbf{x}^k - \Delta\mathbf{x}^{k+1}\|_{\boldsymbol{\Theta}-\mathbf{A}^\intercal\boldsymbol{\Gamma}^\intercal\boldsymbol{\Gamma}\mathbf{A}-\boldsymbol{Ł}}^2 + \|\Delta\mathbf{r}^k - \Delta\mathbf{r}^{k+1}\|_{\mathrm{Fro}}^2 \\ & +\|(\Delta\mathbf{q}^{k+1} - \Delta\mathbf{q}^k) + \boldsymbol{\Gamma}(\Delta\mathbf{y}^{k+1} - \Delta\mathbf{y}^k)\|_{\mathrm{Fro}}^2 \\ \geq & 0, \end{array} \tag{4.31}$$

which completes the proof. □

We will use Theorem 1 and Lemma 2 to establish the rate of convergence. Before that, we need an interlude on the convergence property of a nonnegative monotonic scalar sequence. This is stated as a proposition bellow which has also appeared in recent works [Deng et al., 2017, Davis and Yin, 2017, Shi et al., 2015b].

Proposition 1. *If a sequence* $\{a_k\} \subset \mathbb{R}$ *is:* (i) *nonnegative,* $a_k \geq 0$, (ii) *summable,* $\sum_{t=1}^{\infty} a_t < \infty$, *and* (iii) *monotonically non-increasing,* $a_{k+1} \leq a_k$, *then we have:* $a_k = o(1/k)$.

Proof. Since a_k is monotonically non-increasing, we have $ka_{2k} \leq \sum_{t=k+1}^{2k} a_t$. By this, along with the fact that $\lim_{k\to\infty} \sum_{t=k+1}^{2k} a_t \to 0$ which is given by the summability, we get $a_k = o(1/k)$. □

Theorem 2 (Sublinear rate)**.** *Under the same settings as those in Theorem 1, the first-order optimality residuals (violation to the KKT system* (4.7)*):*

(i) Consensus violation $\|\mathbf{U}\mathbf{x}^k\|_{\mathrm{Fro}}^2$*;*

(ii) Local replication error $\|\boldsymbol{\Gamma}(\mathbf{A}\mathbf{x}^k - \mathbf{y}^k)\|_{\mathrm{Fro}}^2$*;*

(iii) Dual/Gradient span space error $\|\widetilde{\nabla}\mathbf{g}(\mathbf{x}^k) + \mathbf{U}\mathbf{r}^k + \mathbf{A}^\intercal\widetilde{\nabla}\mathbf{f}(\mathbf{y}^k)\|_{\mathrm{Fro}}^2$

all converge at an $o(1/k)$ *rate.*

Proof. By applying Proposition 1 to the results of Theorem 1 and Lemma 2, we have that $\|\mathbf{z}^k - \mathbf{z}^{k+1}\|_{\mathbf{M}}^2 = o(1/k)$. By using the definition of $\mathbf{z}$ and the recursive relation for $\mathbf{z}^{k+1}$ of the Algorithm 1, we derive the statement of this theorem. □

Remark 1 (Fully localized parameter settings)**.** *We have obtained that under the parameter settings* $\boldsymbol{\Theta} - \mathbf{A}^\intercal\boldsymbol{\Gamma}^\intercal\boldsymbol{\Gamma}\mathbf{A} - \boldsymbol{Ł} \succ 0$*, the algorithm converges at an* $o(1/k)$ *rate.*

Here we show that the parameter selection can be fully localized. As presented in Section 4.1, we choose $\boldsymbol{Ł} = (\mathbf{I}_m - \mathbf{W}) \otimes \mathbf{I}_{d+1}$ *where* $\mathbf{W}$ *is an* $m \times m$ *doubly stochastic matrix whose non-zero pattern meets the topology of the underlying communication network. The agents in the network can determine/implement a* $\mathbf{W}$ *matrix only using local information in a purely decentralized manner. Such matrix* $\mathbf{W}$ *satisfies* $-\mathbf{I}_m \prec \mathbf{W} \preccurlyeq \mathbf{I}_m$ *thus* $0 \preccurlyeq \boldsymbol{Ł} \prec 2\mathbf{I}_{m(d+1)}$*. To fulfill* $\boldsymbol{\Theta} - \mathbf{A}^\intercal \boldsymbol{\Gamma}^\intercal \boldsymbol{\Gamma} \mathbf{A} - \boldsymbol{Ł} \succ 0$*, we only need* $\boldsymbol{\Theta} - \mathbf{A}^\intercal \boldsymbol{\Gamma}^\intercal \boldsymbol{\Gamma} \mathbf{A} \succ 2\mathbf{I}_{m(d+1)}$*, which can be broken down to* $\boldsymbol{\Theta}_j - \mathbf{A}_j^\intercal \boldsymbol{\Gamma}_j^\intercal \boldsymbol{\Gamma}_j \mathbf{A}_j \succ 2\mathbf{I}_{d+1},\ \forall j$*. Each agent* j *has the freedom of how it would like to choose its own* $\boldsymbol{\Theta}_j$ *and* $\boldsymbol{\Gamma}_j$ *to satisfy such requirement. We recommend that one chooses* $\boldsymbol{\Gamma}_j$ *to make the largest and smallest nonzero eigenvalues of* $\mathbf{A}_j^\intercal \boldsymbol{\Gamma}_j^\intercal \boldsymbol{\Gamma}_j \mathbf{A}_j$ *close to* 2 *and then* $\boldsymbol{\Theta}$ *can be chosen as* $4\mathbf{I}_{m(d+1)}$*.*

4.2 Application of cPDS on ℓ_1-Regularized Support Vector Machines

In this section we will apply cPDS to the large-scale distributed sSVM problem. Assume that n samples of data (patients EHRs) are distributed among m agents (hospitals) that want to collectively agree on a global classifier to separate the two classes. Each agent is holding n_j samples and maintains a copy $(\boldsymbol{\beta}_j, \beta_{j0})$ of the classifier parameters to be estimated. $(\boldsymbol{\beta}_j, \beta_{j0})$ are updated in each iteration of the method, using data locally stored in the agent as well as information that the agent receives from its neighbors. Let $\boldsymbol{\phi}_{ji} \in \mathbb{R}^d$ and $l_{ji} \in \mathbb{R}$ be the features and the label of sample i in agent j accordingly and f_{ji} be the corresponding hinge loss for that sample. g_j contains the regularizers of parameters $(\boldsymbol{\beta}_j, \beta_{j0})$ for each agent j. Define $\mathbf{a}_{ji} = (l_{ji}\boldsymbol{\phi}_{ji}, l_{ji})$, which we will use later. In every iteration each agent updates $\mathbf{x}_j = (\boldsymbol{\beta}_j, \beta_{j0}) \in \mathbb{R}^{d+1}$, $\mathbf{y}_j \in \mathbb{R}^{n_j}$, $\mathbf{q}_j \in \mathbb{R}^{n_j}$ and $\boldsymbol{\lambda}_j \in \mathbb{R}^{d+1}$. Let us illustrate below the cPDS updates that each agent is performing. For simplicity in the implementation, we use $\boldsymbol{\Theta}_j = \theta_j \mathbf{I}_{d+1}$, where θ_j is a positive scalar maintained by agent j locally. Next we describe the updates over each agent j.

x-update

$$(\boldsymbol{\beta}_j^{k+1}, \beta_{j0}^{k+1}) = \underset{\boldsymbol{\beta}_j, \beta_{0j}}{\arg\min} \sum_{i=1}^{n_j} \gamma_{ji}(2q_{ji}^k - q_{ji}^{k-1})\mathbf{a}_{ji}^\intercal(\boldsymbol{\beta}_j, \beta_{j0}) \\ + \frac{\tau}{2}\|\boldsymbol{\beta}_j\|^2 + \rho\|\boldsymbol{\beta}_j\|_1 + (2\boldsymbol{\lambda}_j^k - \boldsymbol{\lambda}_j^{k-1})^\intercal(\boldsymbol{\beta}_j, \beta_{j0}) + 0.5\|(\boldsymbol{\beta}_j, \beta_{j0}) - (\boldsymbol{\beta}_j^k, \beta_{j0}^k)\|^2\theta_j \tag{4.32}$$

The simple form of the non-smooth g_j allows us to get a closed form solution for this problem. Problem (4.32) can be decoupled into two problems, one that finds $\boldsymbol{\beta}_j^{k+1}$, whose solution is given by the soft thresholding function, i.e., $\forall\ t = 1, \ldots, d$,

$$\beta_{jt}^* = \text{sgn}(u_{jt})(|u_{jt}| - \mu)_+ = \begin{cases} u_{jt} - \mu, & \text{if} \quad u_{jt} > \mu, \\ 0, & \text{if} \quad |u_{jt}| \leq \mu, \\ u_{jt} + \mu, & \text{if} \quad u_{jt} < -\mu, \end{cases}$$

with $\mu = (2\rho)/(\tau + \theta_j)$ and $\mathbf{u}_j = -1/(\tau + \theta_j)\Big[\sum_{i=1}^{n_j} \gamma_{ji} l_{ji}(2q_{ji}^k - q_{ji}^{k-1})\boldsymbol{\phi}_{ji} + (2\boldsymbol{\lambda}_{j,1:d}^k - \boldsymbol{\lambda}_{j,1:d}^{k-1}) - \theta_j\boldsymbol{\beta}_j^k\Big]$, and one that finds β_{j0}^{k+1}, which has as an optimal solution:

$$\beta_{j0}^* = \frac{-\sum_{i=1}^{n_j} \gamma_{ji} l_{ji}(2q_{ji}^k - q_{ji}^{k-1}) - (2\boldsymbol{\lambda}_{i,d+1}^k - \boldsymbol{\lambda}_{i,d+1}^{k-1}) + \theta_j\beta_{j0}^k}{\theta_j} \tag{4.33}$$

y-update

$$\mathbf{y}_j^{k+1} = \underset{\mathbf{y}_j}{\arg\min} \sum_{i=1}^{n_j} \Big\{ \max\{0, 1 - y_{ji}\} - \gamma_{ji} q_{ji}^k y_{ji} \\ + \frac{1}{2}\|\gamma_{ji}(l_{ji}\boldsymbol{\phi}_{ji}^\intercal\boldsymbol{\beta}_j^{k+1} + l_{ji}\beta_{j0}^{k+1} - y_{ji})\|^2 \Big\}. \tag{4.34}$$

To deal with the second non-smooth term, the hinge loss function, we consider three cases for each term: $1 - y_{ji} > 0$, $1 - y_{ji} < 0$ and $1 - y_{ji} = 0$. For each agent j, we can obtain every entry of $\mathbf{y}_j$ in parallel, i.e., for all i:

- Solve $\tilde{y}_{ji}^{k+1} = \underset{\mathbf{y}_{ji}}{\arg\min} \Big\{ (\gamma_{ji}^2 y_{ji}^2)/2 + (-1 - \gamma_{ji} q_{ji}^k - \gamma_{ji}^2 l_{ji}\boldsymbol{\phi}_{ji}^\intercal\boldsymbol{\beta}_j^{k+1} - \gamma_{ji}^2 l_{ji}\beta_{j0}^{k+1}) \Big\}$, which yields $\tilde{y}_{ji}^{k+1} = 1/\gamma_{ji}^2 \left(1 + \gamma_{ji} q_{ji}^k + \gamma_{ji}^2 l_{ji}\boldsymbol{\phi}_{ji}^\intercal\boldsymbol{\beta}_j^{k+1} + \gamma_{ji}^2 l_{ji}\beta_{j0}^{k+1}\right)$. If $1 - \tilde{y}_{ji} > 0$, then $y_{ji}^{k+1} = \tilde{y}_{ji}^{k+1}$; otherwise proceed to the next step.

- Solve $\tilde{y}_{ji}^{k+1} = \arg\min_{\mathbf{y}_{ji}}\{\gamma_{ji}q_{ji}^k, -y_{ji} + 1/2\|\gamma_{ji}(l_{ji}\boldsymbol{\phi}_{ji}^\intercal\boldsymbol{\beta}_j^{k+1} + l_{ji}\beta_{j0}^{k+1} - y_{ji})\|^2\}$, which yields $\tilde{y}_{ji} = 1/\gamma_{ji}^2\left(\gamma_{ji}q_{ji}^k + \gamma_{ji}^2 l_{ji}\boldsymbol{\phi}_{ji}^\intercal\boldsymbol{\beta}_j^{k+1} + \gamma_{ji}^2 l_{ji}\beta_{j0}^{k+1}\right)$. If $1 - \tilde{y}_{ji}^{k+1} < 0$, then $y_{ji}^{k+1} = \tilde{y}_{ji}^{k+1}$; otherwise proceed to the next step.

- $y_{ji}^{k+1} = 1$.

q-update $\forall i$

$$q_{ji}^{k+1} = q_{ji}^k + \gamma_{ji}(l_{ji}\boldsymbol{\phi}_{ji}^\intercal\boldsymbol{\beta}_j^{k+1} + l_{ji}\beta_{j0}^{k+1} - y_{ji}^{k+1}) \tag{4.35}$$

$\boldsymbol{\lambda}$-update

$$\boldsymbol{\lambda}_j^{k+1} = \boldsymbol{\lambda}_j^k + \sum_{i\in\mathcal{N}_j\cup\{j\}} w_{ji}\mathbf{x}_j^{k+1} \tag{4.36}$$

Last, let us mention that the storage needed to operate cPDS for sSVM following the updates in this section is $O(nd)$, which is the same as the other methods listed in Table 4.1 (also see Table 4.1 for other comparisons).

4.3 Experimental Results on the Heart Disease Dataset

We will now apply our methodology to solve the heart diseases hospitalization prediction problem. We measure the performance of cPDS in terms of the Area Under the Receiver Operator Characteristic (ROC) curve (AUC), which plots the detection rate (i.e., out of the hospitalized patients how many were correctly predicted as hospitalized) versus the false alarm rate (i.e., out of the non-hospitalized patients how many were wrongly predicted to be hospitalized). We also consider for comparison the barrier method, the SubGD, the IncrSub descent and the LAC scheme we have mentioned in the Introduction. For SubGD and IncrSub, we use the steplength rule for the diminishing stepsize.[7] The parameters for all methods are selected via cross-validation.

[7]Following the steplength rule, the diminishing stepsize in k-th iteration is set as $a_k = a_0/(\|\widetilde{\nabla}\mathbf{g}(\mathbf{x}^k)\| + \epsilon)$, where a_0 is an initial value of the stepsize and ϵ a very small number.

In cPDS the data are distributed between $m = 10$ hospitals connected through a random graph generated by the Erdős–Rényi model. Using this model, a graph is constructed by connecting nodes randomly. Each edge is included in the graph with probability p independent from every other edge. In our experiments, we have used $p = 0.2$. Table 4.2 shows the AUC and the total running time of the algorithms to perform the maximum number of iterations reported. In the experiment, cPDS gives a comparable AUC as IncrSub does, both outperforming the others. Although cPDS requires more time than IncrSub does, it can work over general decentralized networks.

Table 4.2: Numerical performance of different methods for solving the sSVM problem: AUC, maximum number of iterations, total running time (in secs).

Method	Distributed?	AUC	max iters	total time
SubGD	$\times$	0.7667	1500	2055
Barrier	$\times$	0.7688	32	40.174
IncrSub	$\times$	0.7734	554	6.3485
LAC	$\surd$	0.7520	1000	147,090
cPDS	$\surd$	0.7711	2000	11,176

Chapter 5

An Alternating Clustering and Classification Framework

We seek to predict hospitalizations associated with heart diseases or diabetes within one year from the time the EHR of a patient is examined. We treat hospitalization prediction as a classification problem, distinguishing between patients likely to be hospitalized or not. Intuitively, however, patients belong to different clusters depending on their demographics and ailments that are likely to cause a future hospitalization. The supervised learning methods we have explored so far can certainly make classifications without considering these hidden clusters; yet, identifying the clusters can potentially improve classification performance. An additional key benefit of hidden cluster identification is that results become more interpretable. Patients in the same cluster, especially if the cluster is identified based on a low-dimensional subspace of "diagnostic" features, share key characteristics and their cluster membership offers an explanation as to why they have been flagged for a future hospitalization. In the medical setting, interpretability has an essential role in persuading physicians to trust the learning outputs and rely on them for their decision making. EHRs exhibit interesting special structure in that for each patient only a very low-dimensional subset of features is important in predicting a future hospitalization. This subset is different for each cluster and, typically, there is no universal set of irrelevant features that can be eliminated. This suggests that it is useful to consider sparse classifiers for each cluster.

Related literature: In the literature, there are generally two types of assumptions about hidden clusters in a classification problem, implicit or explicit. The implicit approach is more prevalent, which is implied in piecewise linear techniques [Toussaint and Vijayakumar, 2005, Dai et al., 2006, Yujian et al., 2011, Pele et al., 2013]. The purpose of a piecewise linear classifier is to approximate nonlinear boundaries with a union of local linear classifiers. Therefore samples are implicitly assumed to lie in local regions/clusters and classified by the local classifiers there. A more obvious assumption of hidden clusters (even though still implicit) is in feature space partitioning methods. Tree-based methods [Breiman et al., 1984] partition the whole feature space into sub-regions and each sub-region can be viewed as a cluster. Different from the greedy approach tree methods took, [Wang and Saligrama, 2012] utilize an iterative way of partitioning the feature space and train classifiers inside each subregion. All these methods do not have clustering as their goal and clusters are simply a byproduct in their classification models.

An explicit assumption of clusters within a classification problem is proposed in [He et al., 2006,Gu and Han, 2013], where training samples are first put into clusters and then separate classifiers are trained. They both do clustering once and [He et al., 2006] trains classifiers in parallel while [Gu and Han, 2013] trains classifiers jointly. Due to the sequential procedure, the clustering does not take label information into account and thus these methods' advantage mostly lie in boosting the speed of model training.

A special feature of our problem is that the two classes are asymmetric in the sense that only the positive samples are assumed to have hidden clusters. A concrete example can be drawn again from medical diagnosis, where the positive class represents the unhealthy people and the negative class represents the opposite. It is very intuitive that people get sick for various reasons (viewed as different clusters) while

the healthy people should be healthy in every aspect (thus forming only one cluster). A similar asymmetric setting is also proposed in [Zhao and Shrivastava, 2013] where the data are assumed to be imbalanced and the larger class contains hidden clusters. Their solution is to solely cluster the larger class and train classifiers with copies of the samples from the other class. We design two methods along this direction which serve as our baseline for comparison.

From all the related literature, the most similar problem is mentioned in [Filipovych et al., 2012], also with a medical application. There, they try to maximize the margin between hidden clusters and, thus, are generally suitable for cases with only two hidden clusters. Besides, they use mixed integer programming to represent the cluster tags, which makes the problem intractable for large instances.

5.1 Problem Definition

We consider a classification problem that has multiple hidden clusters in the positive class, while the negative class is assumed to be drawn from a single distribution. For different clusters in the positive class, we assume that the discriminative dimensions, with respect to the negative class, are different and sparse. We could think of these clusters as "local opponents" to the whole negative set (see Fig. 5·1) and therefore, the "local boundary" (classifier) could naturally be assumed to be different and lying in a lower-dimensional subspace of the feature vector.

In summary, the classification problem satisfies the following assumptions:

- The negative class samples are assumed to be i.i.d. and drawn from a single cluster with distribution P_0.
- The positive class samples belong to L clusters, with distributions $P_1^1, \cdots, P_1^L$.
- Different positive clusters have different features that separate them from the

negative samples.

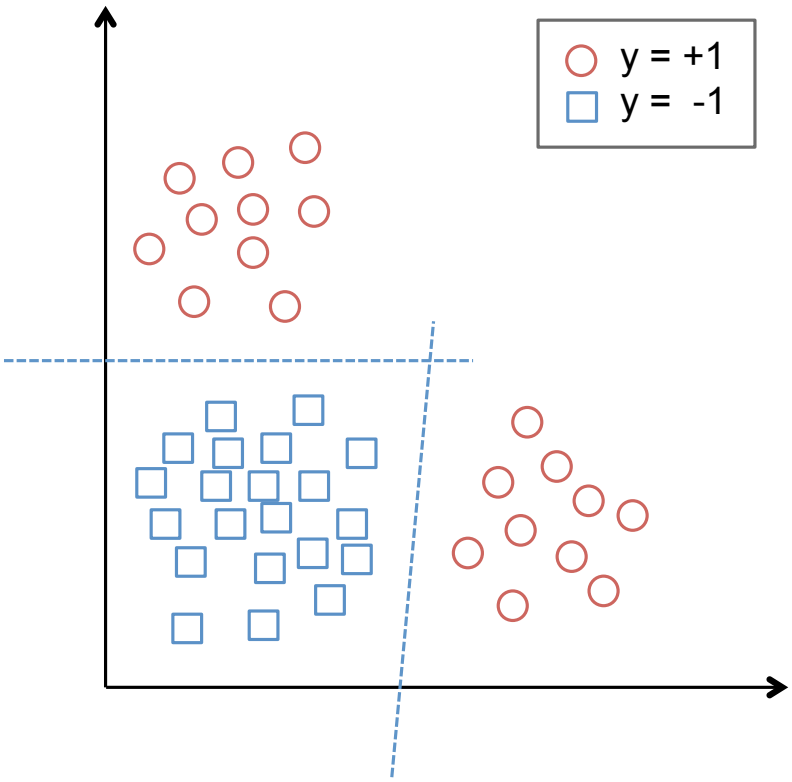

Figure 5·1: Positive clusters as "local opponents". The positive class contains two clusters and each cluster is linearly separable from the negative class, denoted by dashed lines.

We propose a joint cluster detection and classification problem under the SVM framework. Let $(\mathbf{x}_i^+, y_i^+)$ and $(\mathbf{x}_j^-, y_j^-)$ be the $D+1$ dimensional positive and negative samples, where $i \in \{1, 2, \ldots, N^+\}$ and $j \in \{1, 2, \ldots, N^-\}$. Let T be the parameter controlling the local sparsity. Assuming L hidden clusters in the positive class, we try to discover the L hidden clusters (denoted by a mapping function $l(i)$) and L

classifiers ($\boldsymbol{\beta}^l, \beta_0^l$'s, one for each cluster) as the solution of

$$\begin{aligned}
\min_{\boldsymbol{\beta}^l,\beta_0^l,l(i)} \quad & \sum_{l=1}^{L}\left(\tfrac{1}{2}||\boldsymbol{\beta}^l||^2 + \lambda^+ \sum_{i:l(i)=l} \xi_i^{l(i)} + \lambda^- \sum_{j=1}^{N^-} \zeta_j^l\right) \\
\text{s.t.} \quad & \sum_{d=1}^{D} |\beta_d^l| \le T, \\
& \xi_i^{l(i)} \ge 1 - y_i^+ \beta_0^{l(i)} - \sum_{d=1}^{D} y_i^+ \beta_d^{l(i)} x_{i,d}^+, \\
& \zeta_i^l \ge 1 - y_j^- \beta_0^l - \sum_{d=1}^{D} y_j^- \beta_d^l x_{j,d}^-, \\
& \xi_i^{l(i)}, \zeta_j^l \ge 0, \\
\text{where} \quad & y_i^+ = 1, \text{ and } y_j^- = -1.
\end{aligned} \tag{5.1}$$

The negative samples are not clustered but simply copied into each cluster. So their empirical costs are counted L times as shown in (5.1). The relative weight of costs from negative samples compared to that of the positive samples is controlled by λ^- and λ^+. The constraint $\sum_{d=1}^{D} |\beta_d^l| \le T$ is an ℓ_1-relaxation of the sparsity requirement to the local classifiers.

5.2 Alternating Clustering and Classification (ACC)

Problem (5.1) involves two sets of decision variables: $(\boldsymbol{\beta}^l, \beta_0^l)$ for the classifiers and $l(i)$ for cluster assignment. The problem is a mixed integer programming problem, but given $l(i)$, it reduces to L quadratic optimization problems. This motivates the alternating clustering and classification (ACC) optimization approach we present next.

The process starts with a random cluster assignment of the positive clusters and then alternates between two modules: (i) training a classifier for each cluster and (ii) re-clustering samples given all the estimated classifiers using a subset of "diagnostic" features $\mathcal{C}$. Note that since only positive samples belong to different clusters, only these samples need to be re-clustered. During the training phase, we alternate between (i) –training L sparse classifiers– and (ii) –re-clustering the positive samples

given the classifiers– until convergence. Algorithm 2 describes the training process, while Algorithm 3 provides details on how we re-cluster the positive samples given the classifiers learnt in (i).

The re-clustering step concentrates only on a subset of "discriminative" features $\mathcal{C} \subseteq \{1, 2, \ldots, D\}$, which adds more flexibility to the model and allows us to incorporate prior knowledge, so that the discovered clusters are more intuitive. Also note that the re-clustering step in ACC does not need to assume any cluster centers, as is common in other clustering methods (e.g., in k-means [Lloyd, 1982]). The reason is that samples labels are available during training (in contrast with a typical clustering problem) and the goal of clustering is to assist classification. Samples are being assigned into a cluster, where they lie as far away as possible from the classification boundary, i.e., the projection $< \mathbf{x}_{i,\mathcal{C}}, \boldsymbol{\beta}^l_{\mathcal{C}} >$ is the maximum. When checking for the maximum projection, an extra constraint (5.3) needs to be imposed to guarantee the global convergence of the alternating optimization process. Intuitively, the terms in (5.3) are associated with the slack variables in the sparse linear SVM (SLSVM) problem, as presented in (5.9), and imposing this constraint will guarantee that the alternating process moves in a monotonic direction, such that the costs from the slack variables are non-increasing.

$$
\begin{aligned}
O^l = \min_{\boldsymbol{\beta}^l, \beta^l_0} \quad & \tfrac{1}{2}||\boldsymbol{\beta}^l||^2 + \lambda^+ \sum_{i=1}^{N_l^+} \xi^l_i + \lambda^- \sum_{j=1}^{N_l^-} \zeta^l_j \\
\text{s.t.} \quad & \sum_{d=1}^{D} |\beta^l_d| \leq T, \\
& \xi^l_i, \zeta^l_j \geq 0, \\
& \xi^l_i \geq 1 - y^+_i \beta^l_0 - \sum_{d=1}^{D} y^+_i \beta^l_d x^+_{i,d}, \\
& \zeta^l_i \geq 1 - y^-_j \beta^l_0 - \sum_{d=1}^{D} y^-_j \beta^l_d x^-_{j,d}, \\
\text{where} \quad & y^+_i = 1, \quad \forall i \in \{1, \ldots, N_l^+\} \text{ and} \\
& y^-_j = -1 \quad \forall j \in \{1, \ldots, N_l^-\}.
\end{aligned}
\tag{5.2}
$$

Algorithm 2 Alternating Clustering and Classification Training

Initialization:
Randomly assign positive class sample i to cluster $l(i)$. $i \in \{1, \ldots, N^+\}$ and $l(i) \in \{1, \ldots, L\}$.

repeat

Classification Step:
Train an SLSVM classifier for each cluster of positive samples combined with all negative samples. Each classifier is the outcome of a quadratic optimization problem (5.9), that provides $\boldsymbol{\beta}^l$ and O^l.

Re-clustering Step:
Re-cluster the positive samples based on the classifiers $\boldsymbol{\beta}^l$ and update $l(i)$'s.

until no $l(i)$ is changed or $\sum_l O^l$ (the sum of the objective values in training classifiers) is not decreasing.

Once training has been performed with Algorithm 2, we can classify a newly presented sample not seen during training using Algorithm 4. Specifically, we compute the projections on each classifier and assign the new sample to the cluster with the largest projection value. We use the classifier of this cluster to classify the corresponding samples. We note that tuning λ^+ and λ^- in ACC should be done globally, i.e., λ^+ and λ^- should be fixed across all clusters to guarantee convergence.

Two Hierarchical Methods for Clustering and Classification. To demonstrate the superiority of ACC, we will compare it against regural SVM that use a linear or an RBF kernel and with other binary supervised learning methods, such as ℓ_1-regularized logistic regression and random forests [Friedman et al., 2001]. We will also compare ACC with two hierarchical methods which naturally arise from our assumptions regarding the data.

Since we assume that only the positive class contains clusters, during the model training phase we could first cluster the positive samples (still based on the feature set C), then copy negative samples into each cluster, and finally optimize classifiers (linear SVMs) for each cluster. For clustering we adopt the widely used k-means

Algorithm 3 Re-clustering procedure given classifiers

Input: positive samples $\mathbf{x}_i^+$, classifiers $\boldsymbol{\beta}^l$, current cluster assignments which assigns sample i to cluster $l(i)$.

for all $i \in \{1, \ldots, N^+\}$ **do**

for all $l \in \{1, \ldots, L\}$ **do**
calculate projection a_i^l from positive sample i onto the classifier for cluster l with only desired dimensions $\mathcal{C}$. $a_i^l =< \mathbf{x}_{i,\mathcal{C}}^+, \boldsymbol{\beta}_{\mathcal{C}}^l >$;
end for

update cluster assignment of sample i from $l(i)$ to
$l^*(i) = \arg\max_l a_i^l$,
subject to

$$< \mathbf{x}_i^+, \boldsymbol{\beta}^{l^*(i)} > +\beta_0^{l^*(i)} \ \geq < \mathbf{x}_i^+, \boldsymbol{\beta}^{l(i)} > +\beta_0^{l(i)}. \tag{5.3}$$

end for

Algorithm 4 Alternating Clustering and Classification Testing

for each test sample $\mathbf{x}$ **do**
Assign it to cluster $l^* = \arg\max_l < \mathbf{x}_{\mathcal{C}}, \boldsymbol{\beta}_{\mathcal{C}}^l >$.
Classify $\mathbf{x}$ with $\boldsymbol{\beta}^{l^*}$.
end for

method [Lloyd, 1982]. To classify new (test) samples we can use an approach just like the ACC method. We name this algorithm Cluster Then Linear SVM (CT-LSVM).

The second hierarchical method we introduce is very similar to CT-LSVM but instead of training a linear SVM, we train a sparse linear SVM, calling this method Cluster-Then- Sparse-Linear-SVM (CT-SLSVM). Notice that an important difference between CT-LSVM, CT-SLSVM and ACC is that ACC has an alternating procedure while the other two do not. With only one-time clustering, CT-LSVM and CT-SLSVM create unsupervised clusters without making use of the negative samples, whereas ACC is taking class information and classifiers under consideration so that the clusters also help the classification.

5.3 ACC Theoretical Performance Guarantees

We begin by presenting a result that suggests a favorable sample complexity for SLSVM compared to the standard linear SVM. Suppose that SLSVM for the l-th cluster yields $Q^l < D$ non-zero elements of $\boldsymbol{\beta}^l$, thus, identifying a Q^l-dimensional feature subspace used for classification. The value of Q^l is controlled by T^l. Assume we draw a training set with N^- negative samples from P_0 and N_l^+ positive samples from P_1^l, where $N^l = N_l^+ + N^-$. Let R_N^l denote the expected training error rate and R^l the expected test error under these distributions.

Theorem 3. *For a sparse linear SVM lying in a Q-dimensional subspace of the original D-dimensional space, for any $\epsilon > 0$ and $\delta \in (0,1)$, if the sample size N^l satisfies*

$$N^l \geq \frac{8}{\epsilon^2}\left((Q^l+1)log\frac{2eN^l}{Q^l+1} + Q^l log\frac{eD}{Q^l} + log\frac{2}{\delta}\right), \tag{5.4}$$

then with probability no smaller than $1-\delta$, $R^l - R_N^l \leq \epsilon$.

Proof. To simplify notation we drop the cluster index l. We will use a result from [Bousquet et al., 2004]. We note that the family of linear classifiers in a D-dimensional space has VC-dimension $D+1$ ([Vapnik, 1998]). Let $\mathcal{G}$ be a function family with VC-dimension $D+1$. For ease of notation we will drop the reference to the l-th cluster as the result applies to all clusters. Let $R_N(g)$ denote the training error rate of classifier g on N training samples randomly drawn from an underlying distribution $\mathcal{P}$. Let $R(g)$ denote the expected test error of g with respect to $\mathcal{P}$. The following theorem from [Bousquet et al., 2004] is useful in establishing our result.

Theorem 4 ([Bousquet et al., 2004])**.** *If the function family $\mathcal{G}$ has VC-dimension $D+1$, then*

$$P\left[R(g) - R_N(g) \leq 2\sqrt{2\frac{(D+1)\log\frac{2eN}{D+1} + \log\frac{2}{\rho}}{N}}\right] \geq 1-\rho \tag{5.5}$$

for any function $g \in \mathcal{G}$ and $\rho \in (0,1)$.

For the given ϵ select large enough N such that

$$\epsilon \geq 2\sqrt{2((D+1)\log(2eN/D+1) + \log(2/\rho))/N}.$$

or

$$2/\rho \leq \exp\left\{(N\epsilon^2)/8 - (D+1)\log(2eN/(D+1))\right\}. \tag{5.6}$$

It follows from Theorem 4

$$P\left(R(g) - R_N(g) \geq \epsilon\right) \leq \rho. \tag{5.7}$$

In our setting, the classifier g is restricted to a Q-dimensional feature subspace of the D-dimensional feature space. Thus, the bound in (5.6) holds by replacing D with Q in the right hand side and the bound in 5.7 holds for any such Q-dimensional subspace selected by the ℓ_1-penalized optimization. Since there are $\binom{D}{Q}$ possible choices for the subspace, using the union bound and the inequality $\binom{D}{Q} \leq (\frac{eD}{Q})^Q = \exp(Q\log\frac{eD}{Q})$, we obtain:

$$P\left(R(g) - R_N(g) \geq \epsilon\right) \leq \rho \exp\left\{Q\log\frac{eD}{Q}\right\} \tag{5.8}$$

For the given $\delta \in (0,1)$ in the statement of Theorem 3, select small enough ρ such that

$$\delta \geq \rho \exp\left\{Q\log\frac{eD}{Q}\right\}$$

or equivalently

$$\frac{1}{\delta} \geq \frac{1}{\rho}\exp\left\{-Q\log\frac{eD}{Q}\right\}$$

Using 5.6 (with Q replacing D), we obtain

$$log\frac{2}{\delta} \leq \frac{N\epsilon^2}{8} - (Q+1)log\frac{2eN}{Q+1} - Qlog\frac{eD}{Q}$$

which implies that N must be large enough to satisfy

$$N \geq \frac{8}{\epsilon^2}\left((Q+1)log\frac{2eN}{Q+1} + Qlog\frac{eD}{Q} + log\frac{2}{\delta}\right)$$

This establishes $P\left(R(g) - R_N(g) \geq \epsilon\right) \leq \rho$, which is equivalent to Theorem 3 and concludes the proof. □

Theorem 5. *The ACC algorithm converges for any set $\mathcal{C}$.*

Proof. At each alternating cycle, for each cluster l we train a SLSVM with positive samples of that cluster combined with all negative samples. This produces an optimal

value O^l and the corresponding classifier $(\boldsymbol{\beta}^l, \beta_0^l)$. Specifically, the formulation is:

$$O^l = \min_{\substack{\boldsymbol{\beta}^l, \beta_0^l, \\ \zeta_j^l, \xi_i^l}} \frac{1}{2}||\boldsymbol{\beta}^l||^2 + \lambda^+ \sum_{i=1}^{N_l^+} \xi_i^l + \lambda^- \sum_{j=1}^{N^-} \zeta_j^l \tag{5.9}$$

$$\begin{gathered} \text{s.t. } \xi_i^l \geq 1 - y_i^+ \beta_0^l - \textstyle\sum_{d=1}^D y_i^+ \beta_d^l x_{i,d}^+, \ \forall i; \\ \zeta_j^l \geq 1 - y_j^- \beta_0^l - \textstyle\sum_{d=1}^D y_j^- \beta_d^l x_{j,d}^-, \ \forall j; \\ \textstyle\sum_{d=1}^D |\beta_d^l| \leq T^l; \ \xi_i^l, \zeta_j^l \geq 0, \ \forall i, j. \end{gathered}$$

We use the sum of the optimal objective function values in (5.9) across different clusters to prove the convergence. We have

$$Z = \sum_{l=1}^{L} O^l = \sum_{l=1}^{L} \left(\frac{1}{2}||\boldsymbol{\beta}^l||^2 + \lambda^- \sum_{j=1}^{N^-} \zeta_j^l \right) + \lambda^+ \sum_{i=1}^{N^+} \xi_i^{l(i)},$$

where $l(i)$ maps sample i to cluster $l(i)$, $\sum_{l=1}^{L} N_l^+ = N^+$, and $\boldsymbol{\beta}^l$, β_0^l, ζ_j^l, and $\xi_i^{l(i)}$ are optimal solutions of (5.9) for each l. Now, let us consider the change of Z at each iteration of the ACC procedure.

First, we consider the re-clustering step given SLSVMs. During the re-clustering step, the classifier and slack variables for negative samples are not modified. Only the $\xi_i^{l(i)}$ get modified since the assignment functions $l(i)$ change. When we switch positive sample i from cluster $l(i)$ to $l^*(i)$, we can simply assign value $\xi_i^{l(i)}$ to $\xi_i^{l^*(i)}$. Therefore, the value of Z does not change during the re-clustering phase and takes the form

$$Z = \sum_{l=1}^{L} \left(\frac{1}{2}||\boldsymbol{\beta}^l||^2 + \lambda^+ \sum_{\{i: l^*(i)=l\}} \xi_i^l + \lambda^- \sum_{j=1}^{N^-} \zeta_j^l \right).$$

Next, given new cluster assignments we re-train the local classifiers by resolving problem (5.9) for each cluster l. Notice that re-clustering was done subject to the constraint in Eq. (5.3) (see Alg. 3). Since $y_i^+ = 1$, we have

$$\xi_i^{l(i)} \geq 1 - \beta_0^{l(i)} - \sum_{d=1}^{D} \beta_d^{l(i)} x_{i,d}^+ \geq 1 - \beta_0^{l^*(i)} - \sum_{d=1}^{D} \beta_d^{l^*(i)} x_{i,d}^+.$$

The first inequality is due to $\xi_i^{l(i)}$ being feasible for (5.9). The second inequality is due to $y_i^+ = 1$ and Eq. (5.3) in Alg. 3. Thus, by assigning $\xi_i^{l(i)}$ to $\xi_i^{l^*(i)}$ it follows

that the $\xi_i^{l^*(i)}$ remain feasible for problem (5.9). Given that the remaining decision variables do not change, $(\boldsymbol{\beta}^l, \beta_0^l, \zeta_j^l, \xi_i^{l^*(i)},\ \forall i = 1, \ldots, N_l^+,\ \forall j = 1, \ldots, N^-)$ forms a feasible solution of problem (5.9). This solution has a cost equal to O^l. Re-optimizing can produce an optimal value that is no worse. It follows that in every iteration of ACC, Z is monotonically non-increasing. Given that Z is bounded below by zero, we establish the convergence of ACC. □

As a remark on convergence, it is worth mentioning that the values λ^+ and λ^- should be fixed across all clusters to guarantee convergence.

Let $\mathcal{H}$ denote the family of clustering/classification functions produced by ACC.

Theorem 6. *The VC-dimension of* $\mathcal{H}$ *is bounded by*

$$V_{ACC} \triangleq (L+1)L(D+1)\log\left(e\frac{(L+1)L}{2}\right). \tag{5.10}$$

Proof. The proof is based on Lemma 2 of [Sontag, 1998]. Given the L functions for clustering, named $g_1, g_2, \ldots, g_L$, the final cluster of a sample is determined by the maximum of g_1 to g_L. This clustering process could be viewed as the output of $(L-1)L/2$ comparisons between pairs of g_i and g_j, where $1 \leq i < j \leq L$. The pairwise comparison could be further transformed into a boolean function (i.e., $\text{sign}(g_i - g_j)$). Then together with the L classifiers for each cluster, we have a total of $(L+1)L/2$ boolean functions to make the final classification. Among all these boolean functions, the maximum VC-dimension is $D+1$. □

From Theorem 6, we draw the observation that the VC-dimension of ACC grows linearly with the dimension of data samples and polynomially (between quadratic and cubic) with the number of clusters. Since the local classifier is trained under an ℓ_1 constraint, it is defined in a lower dimensional subspace. At the same time, the clustering function also lies in a lower dimensional space $\mathcal{C}$. Thus, the bound in Theorem 6 could be tighter in practice.

An immediate consequence of Theorem 6 is the following corollary which establishes out-of-sample generalization guarantees for ACC-based classification and is based on a result in [Bousquet et al., 2004].

Corollary 1. *For any $\rho \in (0,1)$, with probability at least $1-\rho$ it holds:*

$$R \leq R_N + 2\sqrt{2\frac{V_{ACC}\log\frac{2eN}{V_{ACC}} + log\frac{2}{\rho}}{N}}$$

5.4 Experimental Results on the Heart Disease Dataset

For ACC, parameter tuning was done by 3-fold cross-validation with only training data. Some preliminary experiments led us to set $T^l = 6$. L explicitly varies in $(2,3,4,5,6)$ for all methods involving clustering. ACC uses a subset of "diagnostic" features for clustering to better delineate across various types of heart disease. We use the Area Under the ROC Curve (AUC) again as the performance criterion. In Table 5.1, only the best results for CT-LSVM and CT-SLSVM are presented ($L = 2$). The last column of Table 5.1 counts the number of times (out of 10 runs) that each method's AUC outperforms RBF SVM. It can be seen from the table that ACC outperforms the alternatives by anywhere between 0.97% and 5.75% in average AUC. Under $L = 3$, ACC outperforms RBF SVM in 10 our of 10 repetitions.

Settings	avg. AUC	std. AUC	#
ACC, $L = 2$	76.83%	0.87%	10
ACC, $L = 3$	77.06%	1.04%	10
ACC, $L = 4$	75.14%	0.92%	10
ACC, $L = 5$	75.14%	1.00%	9
ACC, $L = 6$	74.32%	0.87%	6
Lin. SVM	72.83%	0.51%	3
RBF SVM	73.35%	1.07%	-
CT-LSVM ($L = 2$)	71.31%	0.76%	0
CT-SLSVM ($L = 2$)	71.97%	0.73%	1

Table 5.1: Average (avg) and standard deviation (std) of the Prediction Accuracy (AUC) of various methods on Heart Disease Data.

To interpret the clusters generated by ACC, we plot in Figure 5·2 the mean value over each cluster of each element in the feature vector $x_{\mathcal{C}}$. The 3 clusters are well-separated. Cluster 2 contains patients with other forms of chronic ischemic disease

(mainly coronary atherosclerosis) and old myocardial infarction. Cluster 3 contains patients with dysrhythmias and heart failure. Cardiologists would agree that these clusters contain patients with very different types of heart disease. Finally, Cluster 1 contains all other cases with some peaks corresponding to endocardium/pericardium disease.

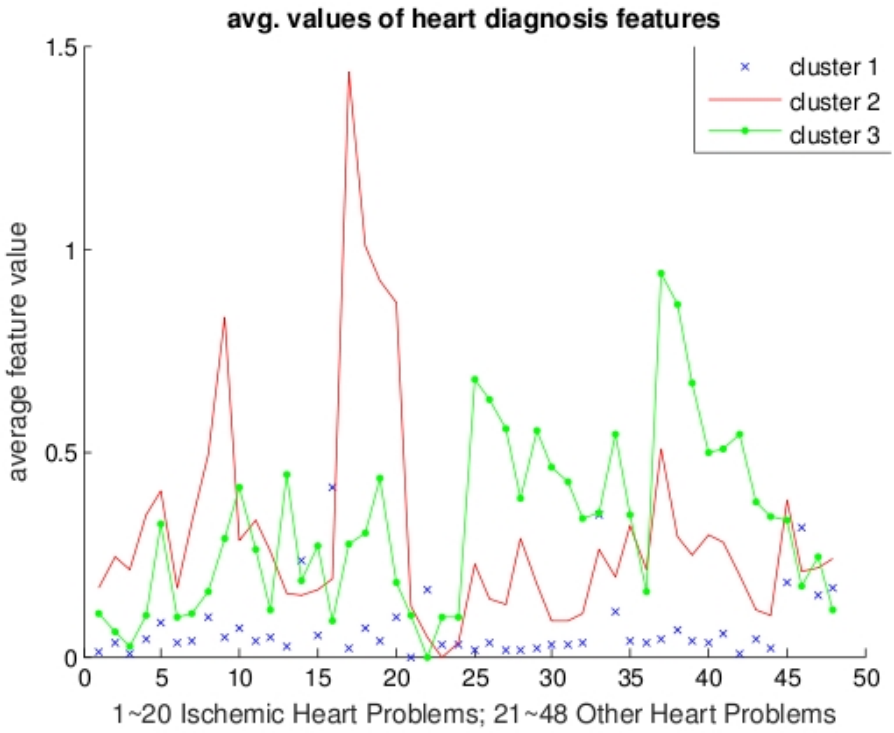

Figure 5·2: Average feature values in each cluster ($L = 3$) for the heart diseases dataset.

5.5 Experimental Results on the Diabetes Dataset

Figure 2 plots ROC curves for a variety of classification methods and Table III lists the corresponding AUCs (average and standard deviation of AUC over 10 runs with different training and test sets). Parameter tuning was done for all methods using k-fold cross validation. For ACC, the initial assignment of the positive samples to the

clusters is random. The parameters as used in 5.1 are set as follows. Some preliminary experiments led us to set the sparsity-controlling parameter T^l to 8. The number of clusters L explicitly takes its values from $\{2, 3, 4, 5, 6\}$ for all methods involving clustering, the soft-margin parameter for the negative class λ^- takes its values from $\{100, 10, 1, 0.1\}$ and the soft-margin parameter for the positive class λ^+ is set equal to $\mathrm{L}\lambda^-$. For all methods 40% of the data are used for training and the rest for testing. The training data are normalized to have zero mean and unit standard deviation and are balanced by down-sampling the negative population. We also compare ACC with CT-LSVM and CT-SLSVM. Only the best results for CT-LSVM (obtained under $L = 2$) and CT-SLSVM (obtained under $L = 2$) are presented.

Clustering with ACC can use a subset of "diagnostic" features (subset $\mathcal{C}$), since these are the features that better delineate across different types of diabetes complications. We base, however, the clustering in these experiments on all features due to the fact that almost all triplet features are related to "diagnostic" features. In Table 5.2, random forests perform best, perhaps not surprisingly since they produce very complex classifiers. Still, and most importantly, they lack interpretability, because of the complexity of the models they produce. ACC performs the best among the remaining alternative methods, and is able to detect the hidden positive clusters and identify why a specific patient is labeled as hospitalized. It is interesting that ACC performs quite well even though the resulting classifiers are relatively sparse and do not use many features. This also makes them easy to implement. Notice that ACC utilizes sparse linear SVM as the base classifier. According to Theorem 4, sparsity (i.e., small D) leads to smaller generalization error. ACC also proved to be efficient from a computational point of view, since in our implementation, it is faster than random forests by a factor of 3.

In an attempt to interpret the ACC clusters we plot in Figure 3 the mean value

Settings	avg. AUC	std. AUC
ACC, $L = 1$ (SLSVM)	0.7924	0.0052
ACC, $L = 2$	0.7855	0.0041
ACC, $L = 3$	0.7853	0.0041
ACC, $L = 4$	0.7846	0.0035
ACC, $L = 5$	0.7836	0.0036
ACC, $L = 6$	0.7818	0.0050
Lin. SVM	0.7687	0.0048
RBF SVM	0.7796	0.0027
CT-LSVM ($L = 2$)	0.7563	0.0050
CT-SLSVM ($L = 2$)	0.7799	0.0049
sparse logistic regression	0.7891	0.0038
random forests	0.8454	0.0026

Table 5.2: Average (avg) and standard deviation (std) of the Prediction Accuracy (AUC) of various methods on Diabetes Data.

over each cluster of each element in the feature vector, using as diagnostic features the subset of features which have a correlation larger than 0.01 with the labels in the training set. This is done for a single repetition of the experiment and L = 3, yielding interesting clusters and highlighting the interpretative power of ACC. We observe that Cluster 1 contains diabetes patients with chronic skin ulcers, hypertension, and an abnormal glucose tolerance test. Cluster 2 contains patients with more severe complications including cerebrovascular disease, hypertension, and heart diseases. Cluster 3 contains patients with less acute disease, combining diabetes with hypertension. The feature values of these three clusters clearly separate from the feature values in the negative class.

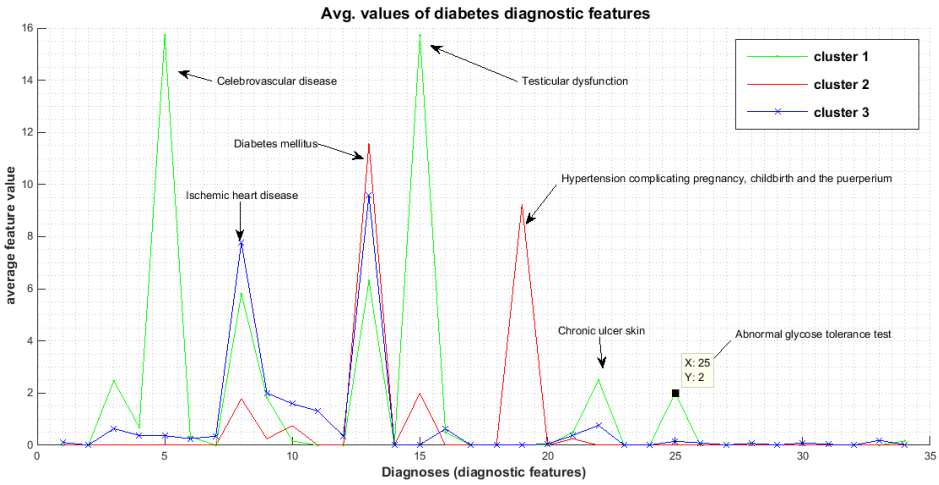

Figure 5·3: Average feature values in each cluster ($L = 3$) for the diabetes dataset.

Chapter 6

Conclusions

6.1 Key Findings

In this thesis, we studied and developed a variety of centralized and distributed methods for predictive health analytics and we showcased them for the novel, to the best of our knowledge, problem of predicting heart- and diabetes- related hospitalizations based on patients Electronic Health Records (EHRs). Below, we summarize our methods and key findings, in the order they were presented:

- We explored a variety of supervised classification methods, such as Support Vector Machines with various kernels, AdaBoost using trees as the weak learner, logistic regression, naive bayes methods etc. We also developed a likelihood ratio based method, K-LRT, that is able to identify the k most important features for each patient. Our results show that with a 30% false alarm rate, we can successfully predict 82% of the patients with heart diseases that are going to be hospitalized in the following year. We have examined methods that have high prediction accuracy (Adaboost with trees), as well as methods that can help doctors identify features to help them when examining patients (K-LRT). One could choose which one to use depending on the ultimate goal and the desirable target for detection and false alarm rates. Our methods also produce a set of significant features of the patients that lead to hospitalization. Most of these features are well-known precursors of heart problems, a fact which highlights

the validity of our models and analysis. The methods are general enough and can easily handle new predictive variables as they become available in EHRs, to refine and potentially improve the accuracy of our predictions.

- We showed that these accuracy rates surpass what is possible with more empirical but well accepted risk metrics, such as the heart disease risk factor that emerged out of the Framingham study. Even a more sophisticated use of the features used in the Framingham risk factor, still leads to results inferior to our approaches. This suggests that the entirety of a patient's EHR is useful in the prediction and this can only be achieved with a systematic algorithmic approach.

- We developed a decentralized method, the cluster Primal Dual Splitting (cPDS) method, to solve the problem of ℓ_1-regularized Support Vector Machines. We proved that cPDS has improved $o(1/k)$ convergence rate compared to the alternatives we have considered. Our formulation has the flexibility to address a range of problems from fully-centralized to fully-decentralized. Information processing can happen either at the level of the patient e.g., in their smartphones or at the level of the hospitals that process data of their own patients. cPDS is a general framework and can be applied to any problem that has the structure of minimizing two nonsmooth terms. For the heart-related hospitalization prediction problem, cPDS achieves AUC as high as 77%.

- We introduced a statistical procedure to identify the diabetes-related admissions and we experimented with a number of machine learning methods that predict hospitalizations in a target year for diabetic patients. With a 20% false alarm rate, we can correctly predict almost 75% of the hospitalized patients.

- We developed a novel clustering and classification framework (ACC) that jointly

discriminates between hospitalized and non-hospitalized patients and discovers clusters of patients with key factors, different in each cluster, that lead to hospitalization. The identification of the clusters has the significant advantage of interpretability, which is crucial in the medical domain. We proved convergence of the new algorithm and established theoretical generalization guarantees.

- If coupled with case management and preventive interventions, our methods have the potential to prevent a significant number of hospitalizations by identifying patients at greatest risk and enhancing their outpatient care before they are hospitalized.

6.2 A Cost-Benefit Analysis

We next assess the potential financial benefits of implementing a predictive model, such as the ones we have presented in this thesis. As an illustrative example, let us consider the Alternating Clustering and Classification scheme for predicting diabetes-related hospitalizations and focus on year 2012; our dataset has $N_H = 916$ hospitalized and $N_{NH} = 27,025$ non-hospitalized patients that year. According to [Clancy et al., 2011], the average cost per hospitalization due to diabetes with complications is \$9,500. Thus, assuming no spending on the non-hospitalized patients and a single hospitalization for the hospitalized, the expected cost per patient if no prevention measures are implemented is:

$$\frac{9500 N_H}{N_H + N_{NH}} = \$311.$$

Suppose now we elect to utilize the predictive model and operate at a point on the ROC curve corresponding to a roughly $P_D = 75\%$ detection rate and a $P_FA = 20\%$ false alarm rate (see Figure 2). We bring each patient predicted to be hospitalized to

the clinic, at a cost of \$220 for a visit according to [Clancy et al., 2011], and prescribe an 1-year-supply of drugs at an average cost of \$100. Thus, the cost of preventive measures is \$320 per patient. Notice that this overestimates the cost because for some patients predicted to be hospitalized, the physician may decide that additional drugs are not needed. For patients the predictive model misses, there is no action and they would receive their normal care. Let P_S the probability that prevention is effective and averts the hospitalization. It follows that the cost per patient becomes:

$$\frac{9500N_H(1-P_D)+320N_{NH}P_FA+320N_HP_DP_S+(9500+320)N_HP_D(1-P_S)}{N_H+N_{NH}}.$$

A simple calculation implies that the above quantity is less than \$311 for $P_S > 0.3$. Taking $P_S = 0.5$ leads to an expected cost per patient equal to \$264, resulting in savings of \$47 per patient. If such a model was used for each patient with diabetes in the U.S. during 2002 (29.1 million), the overall savings amount to \$1.3 billion for the year! This is about 22% of the overall amount spent on preventable diabetes-related hospitalizations each year.

6.3 Future Directions

An immediate possible extension of the cPDS framework could be the analysis of cPDS when the graph that connects the agents is time-varying. Another possible direction is based on the observation that IncrSub is very fast. IncrSub however can only be a decentralized method, when the network follows a ring structure. We would like to consider the possibility of combining cPDS and IncrSub with the goal of generating a distributed method that is as general as cPDS and as fast as IncrSub.

Another interesting extension of this work is to move from *predictive analytics*, which addresses the question of what is likely to happen, to *prescriptive analytics*, which also specifies the actions that are necessary to be taken in order achieve the

predicted outcomes.

References

Adler, L. and Hoagland, G. (2012). What is driving US health care spending? America's unsustainable health care cost growth. https://bipartisanpolicy.org/library/what-driving-us-health-care-spending-americas-unsustainable-health-care-cost-growth/.

American Medical Association (2014). Current Procedural Terminology (CPT). www.ama-assn.org/ama/pub/physician-resources/solutions-managing-your-practice/coding-billing-insurance/cpt.page.

Bayati, M., Braverman, M., Gillam, M., Mack, K. M., Ruiz, G., Smith, M. S., and Horvitz, E. (2014). Data-driven decisions for reducing readmissions for heart failure: General methodology and case study. *PloS one*, 9(10):e109264.

Bertsekas, D. P. (1999). *Nonlinear programming*. Athena scientific Belmont.

Bertsekas, D. P. and Tsitsiklis, J. N. (1989). *Parallel and distributed computation: numerical methods*. Prentice hall Englewood Cliffs, NJ.

Bishop, C. M. et al. (2006). *Pattern recognition and machine learning*. Springer New York.

Bousquet, O., Boucheron, S., and Lugosi, G. (2004). Introduction to statistical learning theory. In *Advanced lectures on machine learning (Part of the Lecture Notes in Computer Science book series)*, volume 3176, pages 169–207. Springer.

Boxwala, A. A., Kim, J., Grillo, J. M., and Ohno-Machado, L. (2011). Using statistical and machine learning to help institutions detect suspicious access to electronic health records. *Journal of the American Medical Informatics Association*, 18(4):498–505.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees*. Wadsworth International Group.

Brisimi, T. S., Olshevsky, A., Paschalidis, I. C., and Shi, W. (2017). A Decentralized Cluster Primal Dual Splitting Method for Large-Scale Sparse Support Vector Machines with An Application to Hospitalization Prediction. Submitted.

Brisimi, T. S., Xu, T., Dai, W., Wang, T., and Paschalidis, I. C. (2016). Predicting diabetes-related hospitalization based on Electronic Health Records. Submitted.

Brossette, S., Sprague, A., Jones, W., Moser, S., et al. (2000). A data mining system for infection control surveillance. *Methods of Information in Medicine*, 39(4-5):303–310.

Brown, J., Kahn, M., and Toh, S. (2013). Data quality assessment for comparative effectiveness research in distributed data networks. *Medical care*, 51(8 Suppl, 3):S22–S29.

Burbidge, R., Trotter, M., Buxton, B., and Holden, S. (2001). Drug design by machine learning: support vector machines for pharmaceutical data analysis. *Computers & chemistry*, 26(1):5–14.

Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., and Elhadad, N. (2015). Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1721–1730. ACM.

Centers for Disease Control and Prevention (2014). National diabetes statistics report: estimates of diabetes and its burden in the United States. `www.cdc.gov/diabetes/pdfs/data/2014-report-acknowledgments.pdf`.

Centers for Disease Control and Prevention (2015). Heart Disease Facts. `www.cdc.gov/heartdisease/facts.htm`.

Cerrito, P. B., Cox, J. A., Mayes, M., and Thompson, W. (2002). Using text analysis to examine icd-9 codes to determine uniformity in the reporting of medpar® data. In *Proceedings of the AMIA Symposium*, page 992. American Medical Informatics Association.

Charles, D., Furukawa, M., and Hufstader, M. (2012). Electronic health record systems and intent to attest to meaningful use among non-federal acute care hospitals in the united states: 2008–2011. *ONC Data Brief*, 1:1–7.

Chen, H., Chiang, R. H., and Storey, V. C. (2012). Business intelligence and analytics: From big data to big impact. *MIS quarterly*, 36(4):1165–1188.

Clancy, C., Munier, W., Crosson, K., Moy, E., Ho, K., Freeman, W., and Bonnett, D. (2011). 2010 national healthcare quality & disparities reports. `http://healthequity.lib.umd.edu/2650/`.

Convertino, V. A., Moulton, S. L., Grudic, G. Z., Rickards, C. A., Hinojosa-Laborde, C., Gerhardt, R. T., Blackbourne, L. H., and Ryan, K. L. (2011). Use of advanced machine-learning techniques for noninvasive monitoring of hemorrhage. *Journal of Trauma and Acute Care Surgery*, 71(1):S25–S32.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

Costa, Â., Castillo, J. C., Novais, P., Fernández-Caballero, A., and Simoes, R. (2012). Sensor-driven agenda for intelligent home care of the elderly. *Expert Systems with Applications*, 39(15):12192–12204.

D'Agostino, R. B., Vasan, R. S., Pencina, M. J., Wolf, P. A., Cobain, M., Massaro, J. M., and Kannel, W. B. (2008). General cardiovascular risk profile for use in primary care the Framingham Heart Study. *Circulation*, 117(6):743–753.

Dai, J., Yan, S., Tang, X., and Kwok, J. (2006). Locally adaptive classification piloted by uncertainty. In *Proceedings of The 23rd International Conference on Machine Learning*, pages 225–232.

Dai, W. (2015). *Detection and prediction problems with applications in personalized health care*. PhD thesis, Boston University.

Dai, W., Brisimi, T. S., Adams, W. G., Mela, T., Saligrama, V., and Paschalidis, I. C. (2014). Prediction of hospitalization due to heart diseases by supervised learning methods. *International Journal of Medical Informatics*, 84(3):189–197.

Dai, W., Brisimi, T. S., Xu, T., Saligrama, V., and Paschalidis, I. C. (2015). Joint clustering and classification approach for healthcare predictive analytics. In *2nd Workshop on Data Mining for Medical Informatics (DMMI): Predictive Analytics*. AMIA.

Davis, D. and Yin, W. (2017). Faster convergence rates of relaxed peaceman-rachford and admm under regularity assumptions. *Mathematics of Operations Research*.

Deng, W., Lai, M.-J., Peng, Z., and Yin, W. (2017). Parallel multi-block admm with o (1/k) convergence. *Journal of Scientific Computing*, 71(2):712–736.

Diamond, A. and Sekhon, J. S. (2013). Genetic matching for estimating causal effects: A general multivariate matching method for achieving balance in observational studies. *Review of Economics and Statistics*, 95(3):932–945.

Dua, S., Acharya, U. R., and Dua, P. (2014). *Machine learning in healthcare informatics*. Springer.

Dwork, C. (2011). Differential privacy. In *Encyclopedia of Cryptography and Security*, pages 338–340. Springer.

Fan, J., Han, F., and Liu, H. (2014). Challenges of big data analysis. *National science review*, 1(2):293–314.

Filipovych, R., Resnick, S., and Davatzikos, C. (2012). Jointmmcc: Joint maximum-margin classification and clustering of imaging data. *IEEE Transactions on Medical Imaging*, 31(5):1124–1140.

Forkan, A. R. M. and Khalil, I. (2017). A clinical decision-making mechanism for context-aware and patient-specific remote monitoring systems using the correlations of multiple vital signs. *Computer Methods and Programs in Biomedicine*, 139:1–16.

Freund, Y., Schapire, R., and Abe, N. (1999). A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612.

Friedman, J., Hastie, T., and Tibshirani, R. (2001). *The elements of statistical learning*, volume 1. Springer series in statistics Springer, Berlin.

Frost, D. W., Vembu, S., Wang, J., Tu, K., Morris, Q., and Abrams, H. B. (2017). Using the electronic medical record to identify patients at high risk for frequent ed visits and high system costs. *The American Journal of Medicine*, 130(5):601.

Groves, P., Kayyali, B., Knott, D., and Van Kuiken, S. (2013). The 'big data' revolution in healthcare: Accelerating value and innovation. `http://www.pharmatalents.es/assets/files/Big_Data_Revolution.pdf`. McKinsey & Company, Center for US Health System Reform.

Gu, Q. and Han, J. (2013). Clustered support vector machines. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, pages 307–315.

Hachesu, P. R., Ahmadi, M., Alizadeh, S., and Sadoughi, F. (2013). Use of data mining techniques to determine and predict length of stay of cardiac patients. *Healthcare informatics research*, 19(2):121–129.

Hannan, T. J. (1999). Detecting adverse drug reactions to improve patient outcomes. *International Journal of Medical Informatics*, 55(1):61–64.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction*. Springer, 2nd edition.

He, J., Zhong, W., Harrison, R., Tai, P., and Pan, Y. (2006). Clustering support vector machines and its application to local protein tertiary structure prediction. *International Conference on Computational Science*, 3993:710–717.

Heritage Provider Network, I. (2011). Heritage health prize.

Hosseinzadeh, A., Izadi, M. T., Verma, A., Precup, D., and Buckeridge, D. L. (2013). Assessing the predictability of hospital readmission using machine learning. In *Proceedings of the Twnety-Seventh AAAI Conference on Artificial Intelligence*

and the Twenty-Fifth Innovative Applications of Artificial Intelligence Conference. https://www.aaai.org/ocs/index.php/IAAI/IAAI13/paper/view/6475.

Hrovat, G., Stiglic, G., Kokol, P., and Ojsteršek, M. (2014). Contrasting temporal trend discovery for large healthcare databases. *Computer methods and programs in biomedicine*, 113(1):251–257.

IBM (2013). Data-driven healthcare organizations use big data analytics for big gains.

International Diabetes Federation (2015). Diabetes Atlas. `www.diabetesatlas.org/component/attachments/?task=download&id=116.`

Jensen, P. B., Jensen, L. J., and Brunak, S. (2012). Mining electronic health records: towards better research applications and clinical care. *Nature Reviews Genetics*, 13(6):395–405.

Jiang, H., Russo, C., and Barrett, M. (2009). Nationwide frequency and costs of potentially preventable hospitalizations, 2006: Statistical brief# 72. Agency for Health Care Policy and Research (US), Rockville (MD).

Jin, Z., Sun, Y., and Cheng, A. C. (2009). Predicting cardiovascular disease from real-time electrocardiographic monitoring: An adaptive machine learning approach on a cell phone. In *Annual International Conference of the IEEE, Engineering in Medicine and Biology Society, 2009. EMBC 2009.*, pages 6889–6892. IEEE.

Khandoker, A. H., Palaniswami, M., and Karmakar, C. K. (2009). Support vector machines for automated recognition of obstructive sleep apnea syndrome from ecg recordings. *IEEE transactions on information technology in biomedicine*, 13(1):37–48.

Kim, S., Kim, W., and Park, R. W. (2011). A comparison of intensive care unit mortality prediction models through the use of data mining techniques. *Healthcare informatics research*, 17(4):232–243.

King, H., Aubert, R. E., and Herman, W. H. (1998). Global burden of diabetes, 1995–2025: prevalence, numerical estimates, and projections. *Diabetes care*, 21(9):1414–1431.

Koh, H. C., Tan, G., et al. (2011). Data mining applications in healthcare. *Journal of healthcare information management*, 19(2):65.

Kohavi, R., Sommerfield, D., and Dougherty, J. (1997). Data Mining Using a Machine Learning Library in C++. *International Journal on Artificial Intelligence Tools*, 6(04):537–566.

Kononenko, I. (1993). Inductive and Bayesian learning in medical diagnosis. *Applied Artificial Intelligence an International Journal*, 7(4):317–337.

Kumari, V. A. and Chitra, R. (2013). Classification of diabetes disease using support vector machine. *International Journal of Engineering Research and Applications*, 3(2):1797–1801.

Laney, D. (2001). 3d data management: Controlling data volume, velocity and variety. *META Group Research Note*, 6.

Li, S., Shi, F., Pu, F., Li, X., Jiang, T., Xie, S., and Wang, Y. (2007). Hippocampal shape analysis of alzheimer disease based on machine learning methods. *American Journal of Neuroradiology*, 28(7):1339–1345.

Lloyd, S. (1982). Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28:129–137.

Ludwick, D. A. and Doucette, J. (2009). Adopting electronic medical records in primary care: lessons learned from health information systems implementation experience in seven countries. *International Journal of Medical Informatics*, 78(1):22–31.

Magoulas, G. D. and Prentza, A. (2001). Machine learning in medical applications. In *Lecture Notes in Computer Science, Machine Learning and its applications*, volume 2049, pages 300–307. Springer.

McCallum, A., Nigam, K., et al. (1998). A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization*, volume 752, pages 41–48. Citeseer.

Menke, A., Casagrande, S., Geiss, L., and Cowie, C. C. (2015). Prevalence of and trends in diabetes among adults in the united states, 1988-2012. *Journal of the American Medical Association*, 314(10):1021–1029.

Meyfroidt, G., Güiza, F., Ramon, J., and Bruynooghe, M. (2009). Machine learning techniques to examine large patient databases. *Best Practice & Research Clinical Anaesthesiology*, 23(1):127–143.

Moore, P., Thomas, A., Tadros, G., Xhafa, F., and Barolli, L. (2013). Detection of the onset of agitation in patients with dementia: real-time monitoring and the application of big-data solutions. *International Journal of Space-Based and Situated Computing*, 3(3):136–154.

Murdoch, T. B. and Detsky, A. S. (2013). The inevitable application of big data to health care. *Journal of the American Medical Association*, 309(13):1351–1352.

Narayanan, A. and Shmatikov, V. (2008). Robust de-anonymization of large sparse datasets. In *IEEE Symposium on Security and Privacy, 2008. SP 2008.*, pages 111–125. IEEE.

Nedic, A. and Bertsekas, D. P. (2001). Incremental subgradient methods for nondifferentiable optimization. *SIAM Journal on Optimization*, 12(1):109–138.

Nedic, A. and Ozdaglar, A. (2009). Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61.

Nesterov, Y. (1983). A method of solving a convex programming problem with convergence rate o (1/k2). In *Soviet Mathematics Doklady*, volume 27, pages 372–376.

Obenshain, M. K. (2004). Application of data mining techniques to healthcare data. *Infection Control & Hospital Epidemiology*, 25(08):690–695.

Office of the National Coordinator for Health Information Technology (2016). Office-based Physician Electronic Health Record Adoption, Health IT Quick-Stat# 50.

Olshevsky, A. (2014). Linear time average consensus on fixed graphs and implications for decentralized optimization and multi-agent control. *arXiv preprint arXiv:1411.4186*.

Paddison, N. V. (2000). Index predicts individual service use. *Health management technology*, 21(2):14.

Palaniappan, S. and Awang, R. (2008). Intelligent heart disease prediction system using data mining techniques. In *Computer Systems and Applications, 2008. AICCSA 2008. IEEE/ACS International Conference on*, pages 108–115. IEEE.

Pathak, J., Kho, A. N., and Denny, J. C. (2013). Electronic health records-driven phenotyping: challenges, recent advances, and perspectives.

Pele, O., Taskar, B., Globerson, A., and Werman, M. (2013). The pairwise piecewise-linear embedding for efficient non-linear classification. In *Proceedings of The 30th International Conference on Machine Learning*, pages 205–213.

Phan, N., Dou, D., Wang, H., Kil, D., and Piniewski, B. (2017). Ontology-based deep learning for human behavior prediction with explanations in health social networks. *Information Sciences*, 384:298–313.

Pofahl, W. E., Walczak, S. M., Rhone, E., and Izenberg, S. D. (1998). Use of an artificial neural network to predict length of stay in acute pancreatitis. *The American Surgeon*, 64(9):868.

Popovic, J. R. (2015). Distributed data networks: A paradigm shift in data sharing and healthcare analytics. In *PharmaSUG 2015*.

Pradhan, M. and Sahu, R. K. (2011). Predict the onset of diabetes disease using artificial neural network (ann). *International Journal of Computer Science & Emerging Technologies (E-ISSN: 2044-6004)*, 2(2).

Raghupathi, W. and Raghupathi, V. (2014). Big data analytics in healthcare: promise and potential. *Health Information Science and Systems*, 2(1):3.

Rathmann, W. and Giani, G. (2004). Global prevalence of diabetes: estimates for the year 2000 and projections for 2030. *Diabetes care*, 27(10):2568–2569.

Ridderikhoff, J. and van Herk, B. (1999). Who is afraid of the system? doctors attitude towards diagnostic systems. *International journal of medical informatics*, 53(1):91–100.

Ridderikhoff, J. and van Herk, E. (1997). A diagnostic support system in general practice: is it feasible? *International journal of medical informatics*, 45(3):133–143.

Ridinger, M. (2002). American healthways uses sas to improve patient care. *DM Review*, 12(139).

Rizk-Jackson, A., Stoffers, D., Sheldon, S., Kuperman, J., Dale, A., Goldstein, J., Corey-Bloom, J., Poldrack, R. A., and Aron, A. R. (2011). Evaluating imaging biomarkers for neurodegeneration in pre-symptomatic huntington's disease using machine learning techniques. *Neuroimage*, 56(2):788–796.

Rodríguez-Martín, D., Samà, A., Pérez-López, C., Català, A., Arostegui, J. M. M., Cabestany, J., Bayés, À., Alcaine, S., Mestre, B., Prats, A., et al. (2017). Home detection of freezing of gait using support vector machines through a single waist-worn triaxial accelerometer. *PloS one*, 12(2):e0171764.

Rosenbloom, S. T., Talbert, D., and Aronsky, D. (2004). Clinicians perceptions of clinical decision support integrated into computerized provider order entry. *International journal of medical informatics*, 73(5):433–441.

Safran, C., Bloomrosen, M., Hammond, W. E., Labkoff, S., Markel-Fox, S., Tang, P. C., and Detmer, D. E. (2007). Toward a national framework for the secondary use of health data: an American Medical Informatics Association White Paper. *Journal of the American Medical Informatics Association*, 14(1):1–9.

Saligrama, V. and Zhao, M. (2012). Local anomaly detection. In *International Conference on Artificial Intelligence and Statistics*, pages 969–983.

Scholkopf, B., Sung, K.-K., Burges, C. J., Girosi, F., Niyogi, P., Poggio, T., and Vapnik, V. (1997). Comparing support vector machines with gaussian kernels to radial basis function classifiers. *IEEE Transactions on Signal Processing*, 45(11):2758–2765.

Shi, W., Ling, Q., Wu, G., and Yin, W. (2015a). Extra: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2):944–966.

Shi, W., Ling, Q., Wu, G., and Yin, W. (2015b). A proximal gradient algorithm for decentralized composite optimization. *IEEE Transactions on Signal Processing*, 63(22):6013–6023.

Son, Y.-J., Kim, H.-G., Kim, E.-H., Choi, S., and Lee, S.-K. (2010). Application of support vector machine for prediction of medication adherence in heart failure patients. *Healthcare informatics research*, 16(4):253–259.

Sontag, E. D. (1998). VC dimension of neural networks. In *Neural Networks and Machine Learning*, pages 69–95. Springer.

Sprinthall, R. C. and Fisk, S. T. (1990). *Basic statistical analysis*. Prentice Hall Englewood Cliffs, NJ.

Stausberg, J. and Person, M. (1999). A process model of diagnostic reasoning in medicine. *International Journal of Medical Informatics*, 54(1):9–23.

Strack, B., DeShazo, J. P., Gennings, C., Olmo, J. L., Ventura, S., Cios, K. J., and Clore, J. N. (2014). Impact of hba1c measurement on hospital readmission rates: analysis of 70,000 clinical database patient records. *BioMed research international*, 2014.

Takeda, H., Matsumura, Y., Nakajima, K., Kuwata, S., Zhenjun, Y., Shanmai, J., Qiyan, Z., Yufen, C., Kusuoka, H., and Inoue, M. (2003). Health care quality management by means of an incident report system and an electronic patient record system. *International Journal of Medical Informatics*, 69(2):285–293.

Tomar, D. and Agarwal, S. (2013). A survey on data mining approaches for healthcare. *International Journal of Bio-Science and Bio-Technology*, 5(5):241–266.

Toussaint, M. and Vijayakumar, S. (2005). Learning discontinuities with products-of-sigmoids for switching between local models. In *Proceedings of The 22rd International Conference on Machine Learning*, pages 904–911.

Tuarob, S., Tucker, C. S., Kumara, S., Giles, C. L., Pincus, A. L., Conroy, D. E., and Ram, N. (2017). How are you feeling?: A personalized methodology for predicting mental states from temporally observable physical and behavioral information. *Journal of Biomedical Informatics*.

Vapnik, V. (1998). *Statistical learning theory*. Wiley, New York.

Vellido, A., Martín-Guerrero, J. D., and Lisboa, P. J. (2012). Making machine learning models interpretable. In *ESANN 2012: 20th European Symposium on Artificial Neural Networks*, volume 12, pages 163–172. Citeseer.

Wang, J. and Saligrama, V. (2012). Local supervised learning through space partitioning. In *Advances in Neural Information Processing Systems*, pages 91–99.

Wang, S. J., Middleton, B., Prosser, L. A., Bardon, C. G., Spurr, C. D., Carchidi, P. J., Kittler, A. F., Goldszer, R. C., Fairchild, D. G., Sussman, A. J., et al. (2003). A cost-benefit analysis of electronic medical records in primary care. *The American Journal of Medicine*, 114(5):397–403.

Wong, W.-K., Moore, A., Cooper, G., and Wagner, M. (2002). Rule-based anomaly pattern detection for detecting disease outbreaks. In *AAAI - 2002: Proceedings of the Eighteenth National Conference on Artificial Intelligence and the Fourteenth Annual Conference on Innovative Applications of Artificial Intelligence*, pages 217–223.

World Health Organization (1999). International Classification of Diseases (ICD). `www.who.int/classifications/icd/en/`.

Wu, J., Roy, J., and Stewart, W. F. (2010). Prediction modeling using EHR data: challenges, strategies, and a comparison of machine learning approaches. *Medical care*, 48(6):S106–S113.

Xu, T., Brisimi, T. S., Wang, T., Dai, W., and Paschalidis, I. C. (2016). A joint sparse clustering and classification approach with applications to hospitalization prediction. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 4566–4571. IEEE.

Yoo, I., Alafaireet, P., Marinov, M., Pena-Hernandez, K., Gopidi, R., Chang, J.-F., and Hua, L. (2012). Data mining in healthcare and biomedicine: a survey of the literature. *Journal of medical systems*, 36(4):2431–2448.

Yu, W., Liu, T., Valdez, R., Gwinn, M., and Khoury, M. J. (2010). Application of support vector machine modeling for prediction of common diseases: the case of diabetes and pre-diabetes. *BMC Medical Informatics and Decision Making*, 10(1):16.

Yujian, L., Bo, L., Xinwu, Y., Yaozong, F., and Houjun, L. (2011). Multiconlitron: a general piecewise linear classier. *IEEE Transactions on Neural Networks*, 22(2):276–289.

Zhang, Y. and Lin, X. (2015). Stochastic primal-dual coordinate method for regularized empirical risk minimization. In *32nd International Conference on Machine Learning (ICML 2015)*, pages 353–361.

Zhao, Y. and Shrivastava, A. (2013). Combating sub-clusters effect in imbalanced classification. In *IEEE 13th International Conference on Data Mining (ICDM)*, pages 1295–1300.

Zolfaghar, K., Meadem, N., Teredesai, A., Roy, S. B., Chin, S.-C., and Muckian, B. (2013). Big data solutions for predicting risk-of-readmission for congestive heart failure patients. In *Big Data, 2013 IEEE International Conference on*, pages 64–71. IEEE.

CURRICULUM VITAE