Label Uncertainty and Learning Using Partially Available Privileged Information for Clinical Decision Support: Applications in Detection of Acute Respiratory Distress Syndrome

by

Narathip Reamaroon

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Bioinformatics)
in The University of Michigan
2021

Doctoral Committee:

Professor Brian D. Athey, Co-Chair
Professor Kayvan Najarian, Co-Chair
Professor Harm Derksen
Dr. Gilbert S. Omenn
Dr. Michael W. Sjoding

Narathip Reamaroon

nreamaro@umich.edu

ORCID iD: 0000-0002-2731-4291

To my parents, Nipon and Krittalux,

and my significant other, Elaine.

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the support and encouragement of many individuals. First and foremost, I would like to thank my advisors, Professor Kayvan Najarian and Professor Brian D. Athey, for their guidance and mentorship.

In addition to being a brilliant scientist and mentor, Dr. Najarian is a role-model for what the scientific pursuit represents: hard-working, collegiate, and limitless enthusiasm for new discoveries. I am grateful for the opportunity to join his lab and I could not have asked for a better mentor to guide me through the Ph.D. His depth and breadth of knowledge has expanded my horizon in the world of research, and I cannot emphasize enough how much I learned from him in these past years.

I am also very fortunate to have Dr. Athey as a co-mentor. Our countless discussions have led to various chapters in this dissertation and beyond. Besides the vast knowledge and support, I am tremendously impressed by his entrepreneurial spirit and his energetic leadership of our department. His passion and excitement for research and advancing human health is nothing short of remarkable, I am forever inspired.

My dissertation committee members were all invaluable mentors. Dr. Harm Derksen's expertise in mathematics has encouraged me to strengthen my understanding of the math behind the fancy algorithms we use, and numerous applications for the developed methodologies in this thesis would not have been possible without his support. I thank Dr. Gil Omenn for his insight, scientific rigor, and for always encouraging

me to reach higher. He is an inspiring figure to every student in the department, and I am fortunate to have been able to work so closely with him. Finally, I am grateful to Dr. Michael Sjoding for being an amazing mentor, for his endless support, and for believing in me. Dr. Sjoding has supported me from day one, in fact, I even remembering attending my first academic conference based on his encouragement to submit my current work for consideration. Undoubtedly, his invaluable advice has sharpened my skills and given me confidence as scientist. It's been my absolute pleasure to work with Dr. Sjoding.

Furthermore, I want to express my appreciation to the Najarian Lab, the Athey Lab, and the Department of Computational Medicine & Bioinformatics. I really couldn't have done it without the support of the faculty and staff in our department. In particular, I would like to thank Dr. Jonathan Gryak - he is truly the glue to our lab and one of the most knowledgeable person I've encountered. I would also like to thank Julia Eussen for helping me navigate through the academic environment, she is truly the rockstar of our department.

This work was supported in part by the National Science Foundation under Grant 1722801. I also want to acknowledge the Gates Millennium Scholars Program and Rackham Merit Fellowship Program for their support.

During my time here, I have been fortunate to be surrounded by a group of friends who have been some of my biggest supporters. I am so thankful to my Ph.D. cohort, labmates, colleagues, and friends I've met along the way in these past years.

I want to thank my significant other, Elaine, who has been my number one supporter through the ups and downs. Her unrelenting love and support has truly helped me get through some of the most difficult periods of the Ph.D.

Finally, I would like to express my deepest thanks to my parents for their unwavering encouragement. Without their love and support, I would not be the person I am today.

TABLE OF CONTENTS

CHAPTER

LIST OF FIGURES

Figure

LIST OF TABLES

Table

LIST OF APPENDICES

Appendix

LIST OF ABBREVIATIONS

AI Artificial Intelligence

ACM Active Contour Model

ARDS Acute Respiratory Distress Syndrome

CLAHE Contrast-Limited Adaptive Histogram Equalization

EHR Electronic Health Record

FDA U.S. Food & Drug Adminstration

GBDT Gradient Boosted Decision Trees

GLCM Grey Level Co-Occurence Matrices

i.i.d. Independent & Identically Distributed

LUPI Learning Using Privileged Information

LUPAPI Learning Using Partially Available Privileged Information

LULUPI Learning Using Label Uncertainty and Privileged Information

LULUPAPI Learning Using Label Uncertainty and Partially Available Privileged Information

ML Machine Learning

PPG Photoplethysmograph

SVM Support Vector Machines

TVAC Total Variation-based Active Contour

ABSTRACT

Artificial intelligence and machine learning have the potential to transform health care by deriving new and important insights from the vast amount of data generated during routine delivery of healthcare. The digitization of health data provides an important opportunity for new knowledge discovery and improved care delivery through the development of clinical decision support that can leverage this data to support various aspects of healthcare - from early diagnosis to epidemiology, drug development, and robotic-assisted surgery. These diverse efforts share the ultimate goal of improving quality of care and outcome for patients. This thesis aims to tackle long-standing problems in machine learning and healthcare, such as modeling label uncertainty (e.g., from ambiguity in diagnosis or poorly labeled examples) and representation of data that may not be reliably accessible in a live environment.

Label uncertainty hinges on the fact that even clinical experts may have low confidence when assigning a medical diagnosis to some patients due to ambiguity in the case or imperfect reliability of the diagnostic criteria. As a result, some data used for machine training may be mislabeled, hindering the model's ability to learn the complexity of the underlying task and adversely affecting the algorithm's overall performance. In this work, I describe a heuristic approach for physicians to quantify their diagnostic uncertainty. I also propose an implementation of instance-weighted support vector machines to incorporate this information during model training.

To address the issue of unreliable data, this thesis examines the idea of learning using "partially available" privileged information. This paradigm, based on knowledge transfer, allows for models to use additional data available during training but may not be accessible during testing/deployment. This type of data is abundant in healthcare, where much more information about a patient's health status is available in retrospective analysis (e.g., in the training data) but not available in real-time environments (e.g., in the test set). In this thesis, "privileged information" are features extracted from chest x-rays (CXRs) using novel feature engineering algorithms and transfer learning with deep residual networks. This example works well for numerous clinical applications, since CXRs are retrospectively accessible during model training but may not be available in a live environment due to delay from ordering, developing, and processing the request.

This thesis is motivated by improving diagnosis of acute respiratory distress syndrome (ARDS), a life-threatening lung injury associated with high mortality. The diagnosis of ARDS serves as a model for many medical conditions where standard tests are not routinely available and diagnostic uncertainty is common. While this thesis focuses on improving diagnosis of ARDS, the proposed learning methods will generalize across various healthcare settings, allowing for better characterization of patient health status and improving the overall quality of patient care. This thesis also includes development of methods for time-series analysis of longitudinal health data, signal processing techniques for quality assessment, lung segmentation from complex CXRs, and novel feature extraction algorithm for quantification of pulmonary opacification. These algorithms were tested and validated on data obtained from patients at Michigan Medicine and additional external sources. These studies demonstrate that careful, principled use of methodologies in machine learning and artificial intelligence can potentially assist healthcare providers with early detection of ARDS and help make a timely, accurate medical diagnosis to improve outcomes for patients.

CHAPTER I

Introduction

1.1 Motivation

The overarching, long-term goal of my research is to develop the computational methods needed to help organize, process, and transform patient data into actionable knowledge, and to work toward using machine learning and data science to build models that will leverage electronic health records in ways that can help healthcare providers improve patient outcomes. The research efforts of this thesis has applications in modeling disease progression and predicting adverse outcomes - including work on developing approaches that can leverage label uncertainty and partially available data, with the ultimate goal of improving quality of care for patients.

The digitization of health data provides an important opportunity for new knowledge discovery and improved care delivery through the development of clinical decision support that can leverage this vast data to support all aspects of healthcare. The advent and accelerated adoption of electronic health records (EHR), as a result of efforts during the Obama administration [1], facilitated the process of generating datasets containing electronically-stored medical data - creating a strong platform digital health innovation. Although EHRs faced significant barriers to adoption [2], this shift in paradigm positioned healthcare for applications of data-driven analytics and artificial intelligence (AI) in the future of medicine. These advances are expected to enhance the quality of healthcare adminstration, automation, and intelligent decision-making in primary/tertiary patient care and public healthcare systems for years to come.

Although EHRs came with the promise of faster and more efficient care, providers in modern settings continue to be inundated by the vast amounts of clinical data being generated during routine patient care. A seminal report from the Institute of Medicine [3] describes how inaccurate medical diagnosis remains a frequent and critical issue for disparity in the quality of healthcare. My research uses machine learning and methodologies in AI to examine such long-standing problems. Specifically, the works in this thesis investigate mathematical approaches to model label uncertainty (e.g., from ambiguity in diagnosis or poorly labeled examples) and representation of data that may not be reliably accessible in clinical settings (e.g., model testing/deployment).

Label uncertainty hinges on the fact that even clinical experts may have low confidence when assigning a medical diagnosis to some patients due to ambiguity in the case or imperfect reliability of the diagnostic criteria. As a result, some data used for machine training may be mislabeled, hindering the model's ability to learn the complexity of the underlying task and adversely affecting the algorithm's overall performance. Furthermore, substantial electronic health data is currently unused (or under-used) during design and training of medical prediction systems, because this data may not be available in live clinical environments. The works presented in this thesis also examine novel approaches to use this type of data as "privileged information" during learning, should improve the accuracy and efficiency of algorithm training, and allowing the integration of a broader array of health data for precision diagnosis. Integrating these disparate data sources in real-time and effectively assisting clinicians in medical decisions could potentially transform how patient care is delivered. By synthesizing this digital information into actionable knowledge, these systems have the potential to provide important insights into the health status of patients, helping physicians make more timely and accurate medical diagnosis and evidence-based treatment decisions.

To achieve this broad set of goals in my research, I aim to develop theoretically-motivated, principled algorithms by drawing from probabilistic modeling, learning using privileged information, reinforcement learning, computer vision, Bayesian inference, and other structured learning approaches. Furthermore, while conducting the research described in this thesis, I also take into consideration the issues of interpretability, reliability, and trustworthiness from the perspective of a healthcare provider to ensure that that the algorithms developed in this thesis will consistently perform as expected, even in the most rigorous tasks. These computational methodologies can assist physicians with making a timely, accurate medical diagnosis to improve quality of care and outcome for patients.

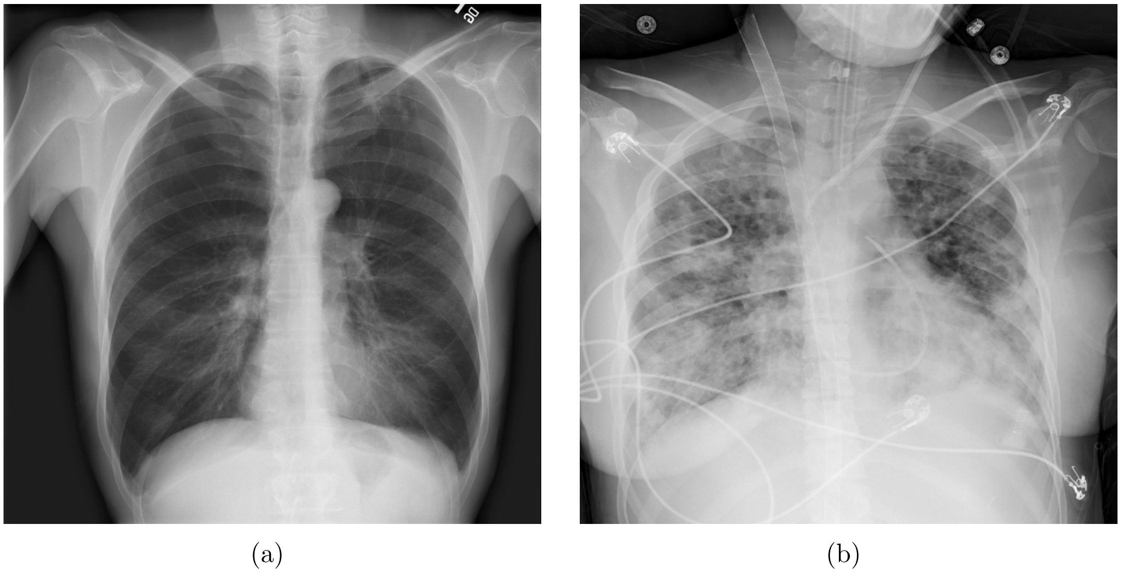
1.2 Background

This thesis is motivated by improving diagnosis of acute respiratory distress syndrome (ARDS), a type of critical respiratory failure. The diagnosis of ARDS serves as a model for many medical conditions where simple, inexpensive gold standard tests are not routinely available and diagnostic uncertainty is common, even among medical experts. While the current proposal focuses on improving diagnosis of ARDS, the proposed learning methods will generalize across various healthcare settings, allowing for better characterization of patient health status and improving the overall quality of patient care.

1.2.1 Acute Respiratory Distress Syndrome

ARDS is life-threatening lung injury characterized by poor oxygenation, diffuse pulmonary infiltrates, and acuity of onset [4]. On the pathological level, this condition is associated with widespread edema and buildup of fluids in the lung. As a result, breathing becomes difficult and organs are deprived of the oxygen needed to function. The clinical corollary is catastrophic and generally associated with poor outcomes, with such risks increasing with age and severity of illness [5].

This condition manifests as an acute injury to the lung, commonly resulting from a prior inciting event such as sepsis, trauma, and severe pulmonary infections [6]. About 50 percent of patients who develop ARDS do so within 24 hours of the inciting event; at 72 hours, 85 percent of patients have clinically apparent ARDS [7]. The clinical presentation of ARDS is characterized by dyspnea, profound hypoxemia, decreased lung compliance, and diffuse bilateral infiltrates on chest radiography.



(a)

(b)

Figure 1.1: Chest x-rays depicting unremarkable presentation and ARDS. The findings of **(a)** shows a negative diagnosis while **(b)** is representative of an ARDS diagnosis.

Imaging findings are variable and depend upon the severity of ARDS. The initial chest radiograph typically depicts diffuse, bilateral alveolar opacities with dependent atelectasis [8]. Example chest x-ray (CXRs) depicting difference between a normal presentation and a diagnosis of ARDS is provided in Figure 1.1. The radiographic appearance of ARDS changes over time, with radiogarphic abnormalities following a perceptible sequence to mirror the rapid histopathological changes. In the first 24 hours following insult, mild alveolar edema is typically present on the CXR [9]. As ARDS progresses, widespread ground-glass opacification becomes more apparent on CT scans. During the next 36 hours, if the condition continues to worsen, the clinical course is characterized by further leakage of inflammatory fluid into the interstitium and air spaces, with evident consolidation on the CXR [10].

ARDS is associated with a high mortality (40% unadjusted mortality, increasing with condition severity [11]), affecting 200,000 patients in the United States and 3 million globally each year [12]. The primary management strategy is life-sustaining supportive care and focuses on reducing shunt fraction, increasing oxygen delivery, decreasing oxygen consumption, and avoiding further injury. Evidence-based practices associated with better outcomes for ARDS includes invasive mechanical ventilation with the use of lower tidal volumes, conservative oxygen therapy, conservative fluid management, lung protective ventilation, prone positioning, positive end-expiratory pressure therapy, and usage of neuromuscular blocking agents until evidence of improvement is observed [13].

1.2.2 Current Approaches & Challenges for Diagnosis

The Berlin Definition of ARDS [4] requires several criteria to be present for clinical diagnosis. First, the onset must be acute, defined as worsening respiratory symptoms occurring within one week of known clinical insult (e.g., sepsis, pneumonia). Bilateral opacities consistent with pulmonary edema (but not fully explained by pleural effusions, lobar collapse, lung collapse, or pulmonary nodules) are required to be present on chest imaging. Furthermore, the patient's respiratory failure must not be fully explained by cardiac failure or fluid overload. Finally, a moderate to severe impairment of oxygenation must be present as defined by the patient's oxygen in arterial blood (PaO_2) to the fraction of the oxygen in the inspired air (FiO_2). These patients have

Table 1.1: The Berlin definition of ARDS.

Timing	Within 1 week of a known clinical insult or new or worsening respiratory symptoms.
Chest Imaging	Bilateral opacities - not fully explained by effusions, lobar/lung collapse, or nodules.
Origin of Edema	Respiratory failure not fully explained by cardiac failure or fluid overload. Need objective assessment[a] to exclude hydrostatic edema if no risk factor present.
Oxygenation	
Mild	200 mmHG $< \mathrm{PaO_2/FiO_2} \leq$ 300 mmHG with PEEP or CAP $\geq$ 5 cmH2O
Moderate	100 mmHG $< \mathrm{PaO_2/FiO_2} \leq$ 200 mmHG with PEEP $\geq$ 5 cmH2O
Severe	$\mathrm{PaO_2/FiO_2} \leq$ 100 mmHG with PEET $\geq$ 5 cmH2O

CPAP = continuous positive airway pressure, $\mathrm{FiO_2}$ = fraction of inspired oxygen, $\mathrm{PaO_2}$ = partial pressure of arterial oxygen, PEEP = positive end-expiratory pressure. [a]e.g., echocardiography.

a $\mathrm{PaO_2/FiO_2}$ ratio of less than 300. These criteria are summarized in Table 1.1.

Patients may present with features specific to ARDS in addition to features of the inciting event. However, the manifestations are so nonspecific that the diagnosis is often missed until the disease progresses [6]. Current ARDS diagnostic criteria also require chest x-ray results as an input, since they provide critical information about whether ARDS is present [4]. However, chest x-rays may not always be available, particularly at the early stages of care. Furthermore, clinicians may be equivocal or even disagree about the diagnosis in some patients using chest x-ray [14], which may result in incorrect labels being provided by an expert.

As ARDS evolves, massive streams of clinical data are collected from their bedside monitoring devices, and recorded in electronic health records as part of routine care. These data are asynchronous - with varying temporal frequency (from millisecond to daily) - and stored in varying formats. Within $24-72$ hours after initial presentation of worsening respiratory symptoms from an inciting event, some patients progress and develop ARDS. However, clinicians will fail to recognize ARDS if they are unable to synthesize the clinical data, obtain chest imaging, and correctly interpret the findings to make an ARDS diagnosis. The ability of healthcare providers to process the massive data-streams generated during the care of critically ill patients has been specifically cited as a potential reason for ARDS miss-diagnosis [15, 11]. Delay in recognition of this condition, followed by delay in the institution of appropriate treatment is associated with increased mortality [16]. Yet, only after a healthcare provider correctly interprets a chest x-ray and synthesize this information with other clinical data will they correctly diagnosis ARDS.

These challenges clinicians face in diagnosing ARDS is analogous to many other clinical diagnoses across acute and chronic care settings. The key problem is the synthesis of longitudinal clinical data to recognize a patient's underlying illness trajectory and the evolution of disease over time. Only with an appropriate level of suspicion will a clinician then obtain the right confirmatory diagnostic test and correctly diagnose a patient's condition. By building electronic clinical decision support systems that models this diagnostic process, the accuracy and reliability of ARDS diagnosis might be greatly enhanced, thereby ensuring patients receive timely treatments that improve clinical outcomes.

1.2.3 Data Driven Clinical Decision Support

Although multiple evidence-based management strategies can be provided to patients with ARDS to improve their outcomes [17], recent evidence suggests that patients with ARDS are not recognized when they develop this syndrome, and consequently, do not receive the evidence-based therapies proven to reduce mortality [11]. The inability of healthcare providers to process the massive streams of clinical data generated while caring for these patients has been specifically cited as a potential reason for poor ARDS recognition [18]. Algorithms that analyze electronic health record (EHR) data and alert providers when patients develop signs of ARDS have been proposed as a potential way to improve early ARDS detection [19, 20].

There have been multiple efforts to develop systems that detect ARDS automatically using routinely collected clinical data. At present, simple rule-based electronic algorithms have been described that analyze EHR data to screen patients for ARDS [21, 22]. Current systems search the text of radiology reports for language consistent with ARDS to identify patients - for example, simple rule-based systems combining arterial blood gas results with keywords searches of radiology reports. However, for these systems to be successful, chest imaging must be obtained at the time when ARDS develops and a radiologist must accurately interpret the radiology image in a timely manner using language that could be interpreted as consistent with ARDS. These dependencies are problematic for successful implementation in clinical practice.

Despite these concerns, if a chest x-ray is available, it is informative and should be integrated into the predictive model's training process - as such, there is considerable potential for using data that is unavailable in live clinical environments but available in training phase. As shown in Figure 1.2, and based on the Berlin Definition, a chest radiograph is the main confirmatory data necessary for a definitive diagnosis of ARDS. In addition, patients known to have early evidence of lung injury on chest x-ray are at high risk for progression to ARDS. However, a chest radiograph may not always be available in real-time, and even the need to perform a chest radiograph may not be recognized at early stages. In addition, the frequency at which chest radiographs are obtained is variable and not amendable to high density collection, in contrast to physiologic monitoring data.

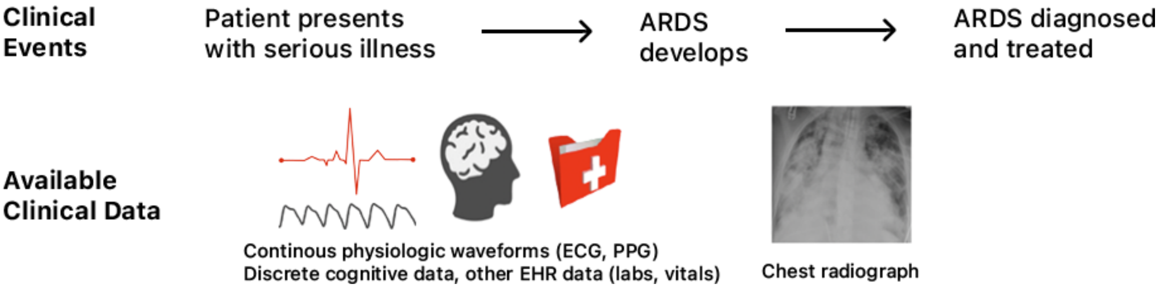

Clinical Events
Patient presents with serious illness
ARDS develops
ARDS diagnosed and treated
Available Clinical Data
Continous physiologic waveforms (ECG, PPG)
Discrete cognitive data, other EHR data (labs, vitals)
Chest radiograph

Figure 1.2: Timeline of ARDS development and clinical data for diagnosis. ECG = electrocardiogram, PPG = photoplethysmography.

Therefore, clinical decision support systems should not rely on chest radiographic data to determine a patient's ARDS risk. However, chest radiographs still can provide critical information about a patient's current state of illness, and this information may be of great value during retrospective training of algorithms that can detect high risk patients. New learning paradigms, specifically Learning Using Privileged Information (LUPI), can leverage information only available during model training - but not required during the testing/deployment phase - to improve the training and predictive performance of machine learning algorithms.

Furthermore, there may also be many cases, particularly at the early stages of diagnosis, where even medical experts may have far less confidence in labeling patients [8]. Computational models that can account for this uncertainty may be more robust to the use of noisy or incomplete health data for medical diagnosis. When training an algorithm to detect ARDS, rather than excluding patients with diagnostic uncertainty, an alternative approach is to use this additional information about diagnostic certainty during training, which could lead to more efficiently learning and better generalize to new patient cases.

Learning with uncertainty is a machine learning paradigm that may be well suited for the task of training an ARDS detection algorithm [23]. The standard machine-learning classification task is to learn a function $f(x) : X \to Y$, which maps input training data $x \in X$ to class $y \in Y$, where X represents a feature space of each patient's covariates and Y is the classification label. The model is trained on well-defined input data of labeled training examples. However, in certain clinical applications, there may be uncertainty in the training labels themselves that could adversely affect model training. In the example of ARDS, there may be challenging cases where the physician has difficulty determining a patient's diagnosis due to clinical ambiguity. As a result, this uncertainty and subsequent mislabeling of training data could adversely affect model training. However, experts may also be able to quantify their diagnostic uncertainty in these cases. The works later described in this thesis examine multiple approaches to incorporate a more realistic representation of uncertainty in real-world applications, avoid discarding uncertain data, and balance the influence of such uncertain inputs in the learning algorithm.

Identifying relevant clinical information from massive data streams is a rapidly advancing field of research in modern medicine. In many healthcare settings, there is an abundance of rich, continuously measured data collected thorough all stages of care. However, there is a lack of reliable systems to: 1) extract clinically relevant features from these data, 2) integrate these features with clinical data to provide quantitative predictions/recommendations about patient's health status, thereby assisting clinicians to improve the quality of decision making. Automated systems that analyze routinely collected data to identify patients with ARDS could improve fidelity with evidence based practice by alerting physicians when patients are not receiving standard care [24]. These advances could support precision medicine by improving rapid yet accurate diagnosis and making just-in-time treatment recommendations that improve care quality and clinical outcomes.

1.3 Outline of Thesis

In this work, I developed and applied novel computational methodologies to integrate electronic health data for early detection of disease. This thesis represents an interdisciplinary research effort encompassing the domains of machine learning, computer vision, and healthcare.

In Chapter II, I present a practical approach to account for label uncertainty for detection of ARDS. First, a heuristic approach for clinical experts to quantify their diagnostic uncertainty is described. I then propose an implementation of instance-weighted support vector machines to incorporate this information during model training and provide validation/testing results on hold-out dataset from Michigan Medicine. Methods to address using highly correlated longitudinal clinical data and handle data imbalanced are also examined. Finally, a signal processing method for quality assessment is presented - this forms the basis of extracting meaningful features from physiological waveforms (e.g., photoplethysmogram).

Chapter III focuses on robust and reliable methods for chest x-ray interpretation. I first discuss the development of an image processing technique for lung segmentation of complex CXRs from hospitalized patients. A novel feature extraction method is then proposed for capturing the notion of diffuse alveolar injury as a mathematical concept. I integrate both of these algorithms and use the extracted features to train multiple machine learning models to detect ARDS. Performance is evaluated with cross-validation and testing on a hold-out test set.

Chapter IV introduces the idea of learning using privileged information, a paradigm that allows for models to use additional data available during training but may not be accessible during testing (i.e., deployment). Chapter V builds upon this foundation and extends the existing algorithm's capabilities to learn from partially available data. I also integrate the previously developed methods of accounting for label uncertainty and CXR interpretation into the privileged information paradigm to train machine learning models for detection of ARDS. The final, comprehensive model is then validated on EHR, CXR, and waveform data from 500 patients at Michigan Medicine.

Chapter VI provides a conclusion to the research and work achieved in this thesis. A discussion is provided to highlight the importance of highly reliable models to ensure healthcare providers that the algorithm will consistently perform as expected, even in the most rigorous tasks. I also address the future direction of between machine learning and healthcare, with emphasis on advancing the discussion of using AI in medical settings. Specifically, I provide insight on the U.S. Food and Drug Administration's perspective on what it will actually take to get these advancements into practice to start making an impact towards better patient care.

CHAPTER II

Accounting for Label Uncertainty

2.1 Introduction

When training a machine learning algorithm for a supervised-learning task in some clinical applications, uncertainty in the correct labels of some patients may adversely affect the performance of the algorithm. For example, even clinical experts may have less confidence when assigning a medical diagnosis to some patients because of ambiguity in the patient's case or imperfect reliability of the diagnostic criteria. As a result, some cases used in algorithm training may be mislabeled, adversely affecting the algorithm's performance. However, experts may also be able to quantify their diagnostic uncertainty in these cases.

The works in this chapter examines a robust method implemented with Support Vector Machines to account for such clinical diagnostic uncertainty when training an algorithm to detect patients who develop ARDS. As previously mentioned in §1.2.1, ARDS is a syndrome of the critically ill that is diagnosed using clinical criteria known to be imperfect. Uncertainty in the diagnosis of ARDS is represented as a graded weight of confidence associated with each training label. The work in this chapter also involves development of a novel time-series sampling method to address the problem of inter-correlation among the longitudinal clinical data from each patient used in model training to limit overfitting. Preliminary results show that we can achieve meaningful improvement in the performance of algorithm to detect patients with ARDS on a hold-out sample, when we compare our method that accounts for the uncertainty of training labels with a conventional SVM algorithm.

Furthermore, this chapter also examines methods for signal quality assessment used for obtaining features to train the machine learning models. Pulsatile physiological signals are often noninvasive recordings of blood-related physiological measurements used in health monitoring. Although these data can be highly informative, the quality of these recordings is a major concern in healthcare [25, 26] as many vital physiological measurements (e.g., respiratory rate, heart rate, and oxygen saturation) are extracted from these signals [27]. The pulsatile nature and similarity of patterns across these signals makes it possible to develop a general algorithm for quality assessment.

The work in this chapter examines six propose morphological features that can be used to determine the quality of the PPG signal and generate a signal quality index. Unlike many similar studies, this approach uses machine learning and does not require a separate signal, such as ECG, for reference. The experiments showed that a cost-sensitive Support Vector Machine (SVM) outperformed other tested methods and was robust to the unbalanced nature of the data. The covariates used as features are provided in Appendix A and Table A.1.

2.2 Time Series Analysis of Health Data

The problem of using highly correlated longitudinal clinical data in model training is often ignored in applications of machine learning in biomedical domains. With the increased use of electronic health records, clinical data are often available in a longitudinal format, where specific metrics of health (e.g., vital signs, or laboratory values) are measured intermittently over time. Analysis of such data requires additional consideration of the stochastic dependency and time-series nature of these data

[28], and they should not be considered independent and identically distributed (i.i.d.) [29], as the data is obviously not. By ignoring the inter-dependency of the time-series data and the i.i.d assumption, training may result in a biased model that overfits to the available data and yield unrealistically large values of specificity, sensitivity, and AUROC [30, 31].

Several techniques, such as dynamic sampling within Markov chain Monte Carlo methods [32] and Bayesian Changepoint Detection [33], are established for analyzing the dependency structure of multivariate time series data. Another standard analytic approach to building prediction models with longitudinal clinical data is using time-to-event models (i.e. survival models) such as the cox-proportional hazards model [34, 35]. Such models have been used in clinical research to estimate a patient's instantaneous hazard of an "absorbing" event such as death based on a patient's current clinical features. However, clinical diagnosis may be better suited as a classification task, where the goal is to determine a patient's status (e.g., ARDS or non-ARDS) based on a set clinical features. Classifying patients at time points prior to their development of a condition, time points after development, and tracking illness recovery, are important for informing treatment decisions—tasks less easily addressed using time-to-event models. In addition, the proportional hazards assumption required of cox models may be overly constraining, while machine learning classifiers like the support vector machine may be more robust.

Building on these techniques, similar methods exist to deal with correlated data in machine learning - such as using a Markov switching process model [36], or partially linear regression model [37] for longitudinal time-series data analysis or a correlation-based fast filter method [38] for choosing among highly correlated features in the model selection process. Beyond the scope of generalized machine learning problems, additional methods to analyze time series properties exist in many domain-specific applications, such as stock market prediction with support vector machine and case-based reasoning [39], or time-delay neural networks [40] and dynamic time warping [41] for speech recognition. However, methods addressing stochastic dependency are largely underdeveloped for applications on longitudinal clinical data.

2.2.1 Stochastic Dependencies in Longitudinal Data

Longitudinal patient data with repeated measurements over time have strong inter-dependency between each instance for a given patient. Ignoring these dependencies during training may lead to a biased estimator and a flawed learning model.

We address the problem by viewing patients' time-series data as a mixing process and consider the data structure as a stationary process with exponentially weakening dependency, and sample instances in a strategic manner to minimize inter-correlation. This approach provides a way to measure the decay in correlation [42] among data on an individual patient over time, and informs a novel sampling strategy to minimize the correlation among data sampled from the same patient for model training.

Inter-dependency among longitudinal data has been previously conceptualized as a system under mixing conditions [36]. For a given stochastic process, "mixing" indicates asymptotically independency – implying that for a stationary process X, the dependency between $X(t_1)$ and $X(t_2)$ becomes negligible as $|t_1 - t_2|$ increases towards infinity [43]. This mixing structure, while assuming that the dependency weakens in time, often exponentially, allows local dependency among the data points, and as such matches the reality of the majority of time-series processed in medicine as well as many other applications [44].

2.2.2 Thresholded Correlation Decay

In order to address the interdependency of the data, we assumed that each patient's time-series data used to develop the ARDS detection algorithm was a mixing stochastic process and we sampled data according to the quantitative assessment of the correlation decay among the data points. This approach limits the degree of inter-correlation on the data points sampled within the same patient and allows a more realistic assessment of model accuracy and reliability.

To implement this sampling strategy, we first calculated pairwise correlation distance matrices to represent dependency over the span of each patient's time-series data. Given an m-by-n matrix for each patient's data, where m is the number of times the patient was observed, and each observation is treated as 1-by-n row vectors, the correlation distance between vectors X_a and X_b for a single pair of observations is defined as:

$$d_{ab} = 1 - \frac{(X_a - \tilde{X}_a)(X_b - \tilde{X}_b)'}{\sqrt{(X_a - \tilde{X}_a)(X_a - \tilde{X}_a)'}\sqrt{(X_b - \tilde{X}_b)(X_b - \tilde{X}_b)'}}$$

where:

$$\tilde{X}_a = \frac{1}{n}\sum_j X_{aj} \text{ and } \tilde{X}_b = \frac{1}{n}\sum_j X_{bj}$$

Using this correlation distance formula, an m-by-m correlation distance matrix can be derived for all observations on the patient, taken pairwise.

The sampling procedure begins by examining the correlation distances between X_t and $\langle X_t \rangle$ was generated, where X_t corresponds to an instance at the start of a patient's time-series data and $\langle X_t \rangle$ is the span of all subsequent time-points. Then a sampling threshold η is set, which represents the point in which the inter-dependency between data becomes more limited. We chose the threshold value of η to be $\frac{1}{\sqrt{2}}$, based on literature that suggests values of approximately $\frac{1}{\sqrt{2}}$ as an estimate of the width of a correlation-type function [45]. We also explored other values of η to understand their effect on the model building process. Figure 2.1 shows the effect of different sampling thresholds on model performance, including the difference in model accuracy in the training to testing set and AUROC of the testing set. With the proposed sampling strategy, SVM performs very well on the training data at any threshold. This empirical analysis confirmed that optimal results are achieved when the sampling threshold is approximately 0.7 and supports the literature suggested value of $\frac{1}{\sqrt{2}}$.

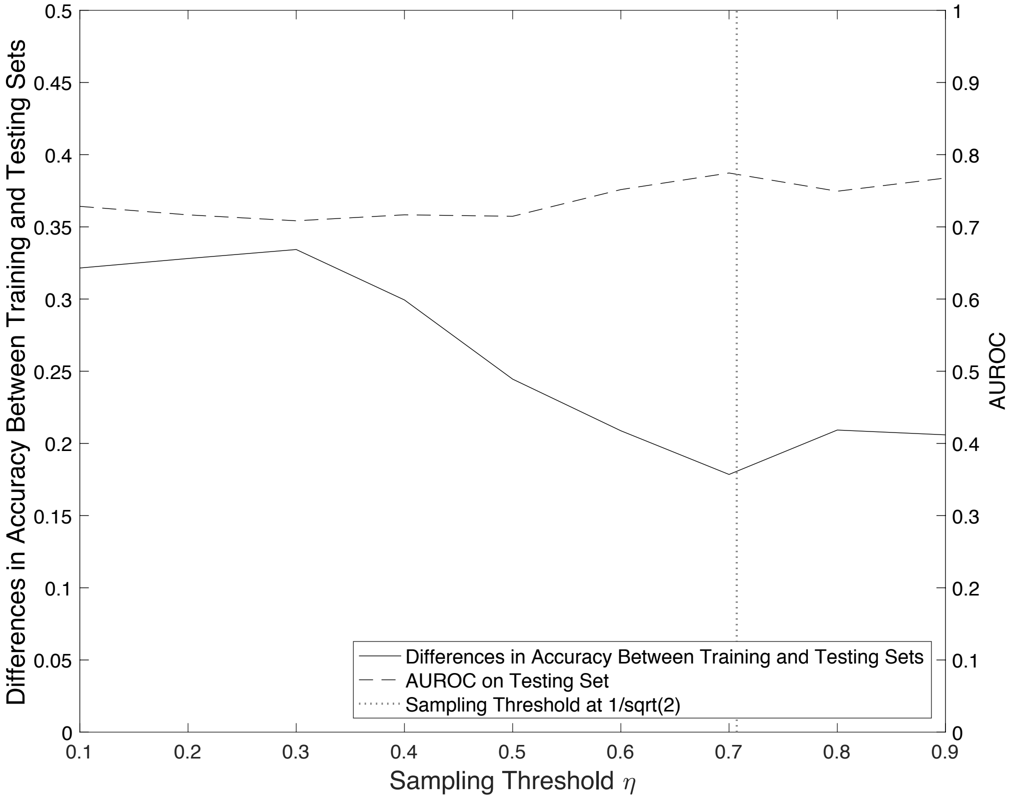


Figure 2.1: Effects of different sampling thresholds on prediction generalizability. Loss in training accuracy is determined when predicting on a hold-out test set to assess the effects of changing the sampling threshold and determining the value for optimal results.

During the data sampling process for each patient, X_t is selected as the start of a patient's time-series data. A pairwise correlation distance matrix is then calculated between X_t and $\langle X_t \rangle$, and a data point X_{t1} is sampled as the first instance with a correlation distance of below η from $\langle X_t \rangle$. This selected point X_{t1} and subsequent time points beyond X_{t1}, $\langle X_{t1} \rangle$, are used to re-calculate a new pairwise correlation distance matrix. A data point X_{t2} is then selected in a similar manner as X_{t1} from data points in $\langle X_{t1} \rangle$ with a correlation distance below the threshold of η. The sampling method is repeated until no further instances of $\langle X_{tn} \rangle$ are below the threshold from X_{tn}.

For this specific dataset, we did not utilize the sampling strategy described above for patient instances with the classification label of ARDS = 1. After inspection of the data, we observed the correlation decay to behave differently according to the label, with the data remained highly correlated over time when ARDS = 1 while correlation decay occurring when ARDS = -1. Therefore, this sampling approach was only performed on the data when ARDS = -1 while all instances were sampled when ARDS = 1. This approach effectively samples all positive examples while undersampling negative examples, which was also necessary given the significant class imbalance of the two labels [46]. The sampling strategy is shown in pseudocode as Algorithm 1 and the average decay of correlation from all patients is shown in Figure 2.11 with error bars representing standard error of the mean.

Algorithm 1: Pseudocode for our algorithm to sample time-series data and reduce inter-dependency.

```
Input  : All available time-series data $\langle X_t \rangle$ from each patient.
for each patient do
    partition data into separate bins according to the classification label;
    if size of either bins is $\leq 4$ then
        sample all available data;
    else
        1) select $X_t$ at the start of the time-series data and sample this instance;
        2) calculate the pairwise correlation distance from $X_t$ to $\langle X_t \rangle$;
        3) sample the first row in $\langle X_t \rangle$ with a correlation distance $< \eta$ and set as the
        new $X_t$;
        repeat
            1) set $\langle X_t \rangle$ as all points subsequent to $X_t$;
            2) calculate the pairwise correlation distance matrix from $X_t$ to $\langle X_t \rangle$;
            3) sample the first row where the correlation distance is $< \eta$ and set as
            the new $X_t$;
        until pairwise distance of $X_t$ to $\langle X_t \rangle > \eta$;
Output: Partial data $\{X_t, X_{t1}, X_{t2}, ..., X_{tn}\}$ with reduced inter-correlation from
        each patient.
```

2.3 Signal Quality Assessment

Some of the features in the longitudinal data were obtained from pulse oximetry - a noninvasive and low-cost physiological monitor that measures blood oxygen levels. While the noninvasive nature of pulse oximetry is advantageous, the estimates of oxygen saturation generated by these devices are prone to motion artifacts and ambient noise, reducing the reliability of such estimations. Clinicians combat this by assessing the quality of oxygen saturation estimation by visual inspection of the photoplethysmograph (PPG), which represents changes in pulsatile blood volume and is also generated by the pulse oximeter. The work described in this section investigates methods to use this data by assessing the quality of the waveforms obtained. Other clinical covariates, in addition to pulse oximetry, used for detection of ARDS are provided in Appendix A and Table A.1

We propose six morphological features that can be used to determine the quality of the PPG signal and generate a signal quality index. Unlike many similar studies, this approach uses machine learning and does not require a separate signal, such as ECG, for reference. Multiple algorithms were tested against 46 30-min PPG segments of patients with cardiovascular and respiratory conditions, including atrial fibrillation, hypoxia, acute heart failure, pneumonia, ARDS, and pulmonary embolism. These signals were independently annotated for signal quality by two clinicians, with the union of their annotations used as the ground-truth. Similar to any physiological signal recorded in a clinical setting, the utilized dataset is also unbalanced in favor of good quality segments.

We acquired data from bedside telemetry monitors of all patients. The PPG recording equipment used in this study is Masimo LNCS DCI adult reusable sensor with GE Medical PDM interface. The sampling frequency of the PPG signals in the dataset is 60 Hz. For the current pulse oximetry quality study, 46 30-min segments of PPG signal from different patients with various cardiovascular and res- piratory con-

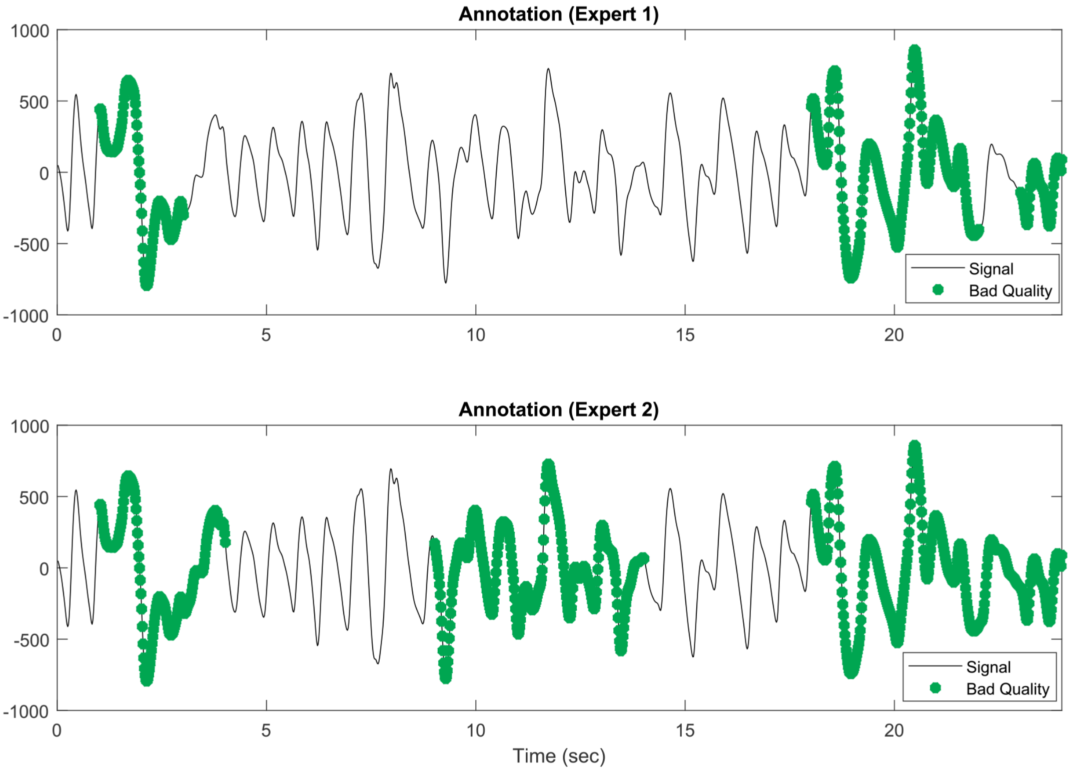


Figure 2.2: Segment of PPG signals designated as "bad" quality from clinicians. A signal segment not annotated as bad quality is assumed to be of good quality.

ditions including atrial fibrillation, hypoxia, acute heart failure, pneumonia, ARDS, and pulmonary embolism were extracted. 27 (out of 46) of these patients are male, the average age of the patients is 57 years old, and 37 (out of 46) are Caucasian. Among these 46 30-min segments, only 12 segments are almost entirely normal, 20 segments contain long episodes of atrial fibrillation and sinus tachycardia, and the rest contain sporadic short-term abnormalities (finger tapping, premature atrial contractions and etc).

Two clinicians independently reviewed PPG signals for uniform, pulsatile changes in the waveform, based on their experience interpreting such waveforms in clinical settings. Waveforms without a clear pulsatile signal (regardless of arrhythmic episodes, only based on morphology) that a clinician would not have trusted as accurate in a clinical setting were annotated as poor quality segments. Certain pulsatile waveforms suspicious for artifact, e.g., finger tapping, were also annotated as poor quality. Figure 2.2 depicts a 24-second segment of PPG signal annotated for signal quality by both clinicians. The union of their labels is used as ground-truth for the algorithm. This cohort is primarily used for development and validation the proposed algorithms.

2.3.1 Preprocessing and Calibration

In the preprocessing phase, a raw PPG signal is first filtered using a band-pass Butterworth filter with a 0.5–5 Hz pass band [47]. The next step is peak detection, wherein potential peaks are only considered if the minimum temporal distance between two consecutive beats is 70% of the mean PPG beat period. Heart rate is adaptively extracted from the power spectrum of the most recent 20 s of PPG signal, as the frequency between 1 Hz and 3 Hz having maximum power spectrum determines the heart rate.

Unlike many algorithms in the literature that use ECG a reference for beat detection [47], the proposed algorithm is independent of any other signal. Moreover, positive and negative peaks are detected independently, resulting in two heart rate signals that should be approximately the same. As described later, the difference between these two heart rate signals is used as a feature of the algorithm, for any significant dissimilarity is due to abnormality in beat morphology. Figure 2.3 depicts examples of raw and filter PPGs with detected positive and negative peaks.

2.3.2 Morphological Features

Six morphological signals/measurements are extracted that are used later to extract features:

1. Beat waveform with positive peak (the interval between two negative peaks).
2. Beat waveform with negative peak (the interval between two positive peaks).

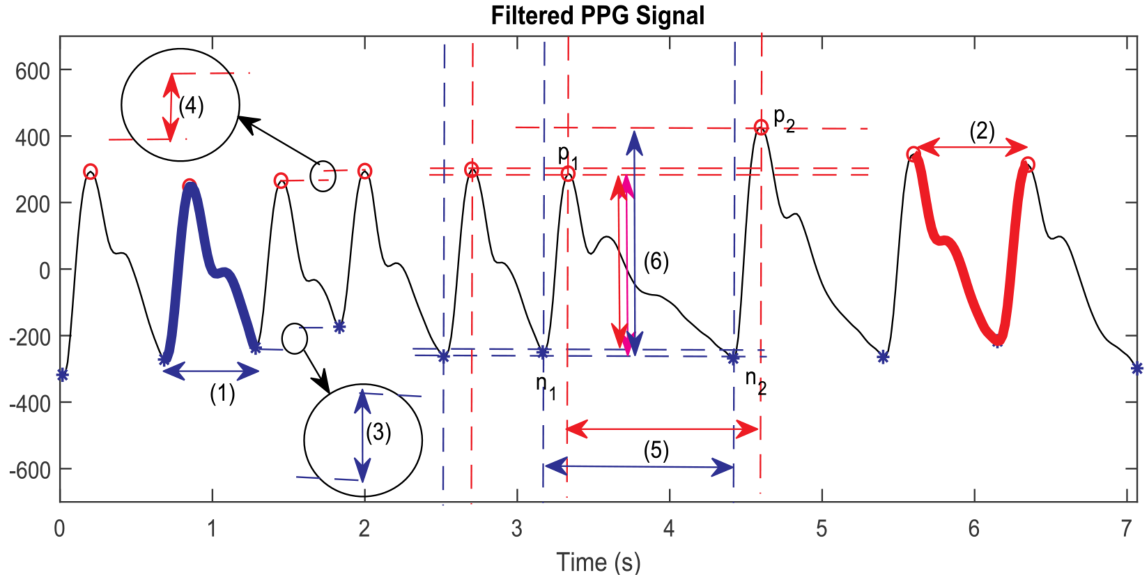


Figure 2.3: Preprocessed PPG signals with morphological measurements: **(1)** beat waveform with positive peak, **(2)** beat waveform with negative peak, **(3)** negative-to-negative peak jump, **(4)** positive-to-positive peak jump, **(5)** positive and negative pulse duration, and **(6)** backward and forward AC components.

3. Change in absolute amplitude between two consecutive negative peaks.

4. Change in absolute amplitude between two consecutive positive peaks.

5. Heart rates extracted from positive peaks and negative peaks (or pulse width).

6. Absolute positive to negative peak amplitude (e.g., the AC component).

These measurements are represented in Figure 2.3. The next step is to use the extracted signals/measurements to calculate morphological features. All of the proposed features are based on some distance or dissimilarity from baseline values or templates. One can think of these templates and baseline values as adaptive averages extracted from normal beats/signals that have already been seen. For now, assume the algorithm is provided with these adaptive averages and focus on the features; later it is described how these averages can be calculated.

Let f_s be the sampling frequency and suppose $\mathscr{T} = \left\{t_k \mid k \in \mathbb{N}, t_k = k\frac{1}{f_s}\right\}$ is the set of time samples in the PPG signal. Assume $f_{\text{PPG}} : \mathscr{T} \to \mathscr{V}$ is the PPG signal amplitude function and $\mathscr{U}$ is the bounded set of these amplitude values, i.e. $\mathscr{V} = \{v_k \mid k \in \mathbb{N}, t_k \in \mathscr{T}, v_k = f_{\text{PPG}}(t_k) \in \mathbb{R}\}$. The features are then extracted as follows:

2.3.2.1 Normalized pulse duration.

Suppose $\mathscr{P}^+$ and $\mathscr{P}^-$ are respectively the set of positive and negative peak locations defined as:

$$\begin{gathered}
\mathscr{P}^+ = \left\{p_i^+ \mid i \in \mathbb{N}, p_i^+ \in \mathscr{T} : \forall t \in \left[p_{i-1}^-, p_i^-\right] \subseteq \mathscr{T}, \right. \\
\left. f_{\text{PPG}}\left(p_i^+\right) \geq f_{\text{PPG}}(t)\right\} \\
\mathscr{P}^- = \left\{p_i^- \mid i \in \mathbb{N}, p_i^- \in \mathscr{T} : \forall t \in \left[p_{i-1}^+, p_i^+\right] \subseteq \mathscr{T}, \right. \\
\left. f_{\text{PPG}}\left(p_i^-\right) \leq f_{\text{PPG}}(t). \right\}
\end{gathered}$$

Then for each consecutive pair of positive peaks $\left(p_{i-1}^+, p_i^+\right) \in \left(\mathscr{P}^+\right)^2$ or negative peaks $\left(p_{i-1}^-, p_i^-\right) \in \left(\mathscr{P}^-\right)^2$ define the normalized pulse duration, $\overline{\nabla p_i}$, as:

$$\overline{\nabla p_i} = \frac{\nabla p_i - \nabla p}{\nabla p}$$

where:

$$\nabla p_i = p_i - p_{i-1} = \begin{cases} p_i^+ - p_{i-1}^+ & (p_{i-1}, p_i) \in \left(\mathscr{P}^+\right)^2 \\ p_i^- - p_{i-1}^- & (p_{i-1}, p_i) \in \left(\mathscr{P}^-\right)^2 \end{cases}$$

and ∇p is the baseline value (as defined in section 2.3.6) of pulse duration. An example of ∇p_i can be seen in Figure 2.3. Given that for every interval between two consecutive positive (negative) peaks there is a negative (positive) peak, each value of $\overline{\nabla p_i}$ is only associated with the interval between the first positive (negative) peak to the next negative (positive) peak.

2.3.2.2 Normalized negative-to-negative peak jump.

Define the set $\mathscr{A}^- = f_{\text{PPG}}\left(\mathscr{P}^-\right) = \left\{P_i^- \mid i \in \mathbb{N}, \forall p_i^- \in \mathscr{P}^-, P_i^- = f_{\text{PPG}}\left(p_i^-\right)\right\}$ as the set of negative peak amplitudes, and let ∇P^- be the baseline value for negative-to-negative peak jump and ∇P be the baseline value for amplitude change from negative to positive (or positive to negative) peaks, i.e., the baseline value for the AC component. For each pair of consecutive negative peaks $\left(p_{i-1}^-, p_i^-\right) \in \left(\mathscr{P}^-\right)^2$, the normalized negative-to-negative peak jump, $\overline{\nabla P_i^-}$, is defined as:

$$\overline{\nabla P_i^-} = \frac{\nabla P_i^- - \nabla P^-}{\nabla P}$$

where $\nabla P_i^- = \left|P_i^- - P_{i-1}^-\right|$.

2.3.2.3 Normalized positive-to-positive peak jump.

Suppose $\mathscr{A}^+ = f_{\text{PPG}}\left(\mathscr{P}^+\right) = \left\{P_i^+ \mid i \in \mathbb{N}, \forall p_i^+ \in \mathscr{P}^+, P_i^+ = f_{\text{PPG}}\left(p_i^+\right)\right\}$ is the set of positive peak amplitudes and ∇P^+ is the baseline value for positive-to-positive peak jump. For each pair of consecutive positive peaks $\left(p_{i-1}^+, p_i^+\right) \in \left(\mathscr{P}^+\right)^2$, the normalized positive-to-positive peak jump, $\overline{\nabla P_i^+}$, is defined as:

$$\overline{\nabla P_i^+} = \frac{\nabla P_i^+ - \nabla P^+}{\nabla P}$$

where $\nabla P_i^+ = \left|P_i^+ - P_{i-1}^+\right|$.

2.3.2.4 Normalized beat amplitude jump.

Suppose $\mathscr{P} = \mathscr{P}^- \cup \mathscr{P}^+$ is the set of positive and negative peak locations and $\mathscr{A} = \mathscr{A}^- \cup \mathscr{A}^+$ is the set of peak amplitudes. Then for any consecutive positive and negative peak $(p_{i-1}, p_i) \in \mathscr{P}^2$, the normalized beat amplitude jump, $\overline{\nabla P_i}$, is defined:

as

$$\overline{\nabla P_i} = \frac{\nabla P_i - \nabla P}{\nabla P}$$

where $\nabla P_i = |P_i - P_{i-1}|$.

2.3.2.5 Dissimilarity measure of positive-peaked beats.

Due to nonlinear and non-stationary changes in beat morphology, a nonlinear time-based stretching or compression of beats is necessary to perform effective template matching [48]. As mentioned earlier in this section, beat waveforms with positive peak (interval between two negative peaks, see Figure 2.3) are extracted and normalized into the range [0,1]. Then, dynamic time warping (DTW) [48] is used to align the PPG with a template as constructed in §2.3.6. Finally, KL divergence [49] is used to measure the difference between the aligned PPG beat and the template, which is formulated as

$$D\left(T^{+} \| B^{+}\right) = \sum_{i=1}^{m} t_i^{+} \log \frac{t_i^{+}}{b_i^{+}}$$

where $B^{+} = \left\{b_k^{+} \mid 1 \leq k \leq m\right\}$ and $T^{+} = \left\{t_k^{+} \mid 1 \leq k \leq m\right\}$ are two aligned time series of beats and template with positive peak, both of which are of length m and normalized such that $\sum_{i=1}^{m} b_i^{+} = \sum_{i=1}^{m} t_i^{+} = 1$. In the proposed algorithm, $D\left(T^{+} \| B^{+}\right)$ is used as the dissimilarity measure of positive-peaked beats feature.

2.3.2.6 Dissimilarity measure of negative-peaked beats.

Applying the same procedure as described above, a dissimilarity measure of negative peaked beats, i.e. $D\left(T^{-} \| B^{-}\right)$, is calculated in which $B^{-} = \left\{b_k^{-} \mid 1 \leq k \leq m\right\}$ and $T^{-} = \left\{t_k^{-} \mid 1 \leq k \leq m\right\}$ are two time series of beat and template with negative peak, both of which are of length k and normalized such that $\sum_{i=1}^{k} b_i^{-} = \sum_{i=1}^{k} t_i^{-} = 1$.

2.3.3 Templates and Baseline Values

As described in §2.3.2, the proposed features $D\left(T^{-} \| B^{-}\right)$ and $D\left(T^{-} \| B^{-}\right)$ require templates, while the features $\overline{\nabla p_i}, \overline{\nabla P_i^{-}}, \overline{\nabla P_i^{+}}$, and $\overline{\nabla P_i}$ need baseline values. A distinct component of the proposed algorithm is the individually generated template and value for each waveform. This section describes how to generate these templates.

2.3.3.1 Initial template and baseline value generation.

The proposed algorithm uses the first T seconds of each waveform in the calibration phase, during which preprocessing and then peak detection is performed on the segment. Based on this segment and the peak locations, the baseline value ∇p is the averaged pulse duration, ∇P^{-} the average negative-to-negative peak jumps, ∇P^{+} the average positive-to-positive peak jumps, and ∇P the average amplitude change from negative to positive peaks. In the results presented in this paper, $T = 20$ seconds is chosen.

Formally, suppose $\mathscr{P}_{0-20}^{+} = \left\{p_i^{+} \mid i \in \mathbb{N}, p_i^{+} \in \mathscr{T}, 0 < p_i^{+} < 20\right\}$ and $\mathscr{P}_{0-20}^{-} = \left\{p_i^{-} \mid i \in \mathbb{N}, p_i^{-} \in \mathscr{T}, 0 < p_i^{-} < 20\right\}$ are the sets of positive and negative peak locations in the [0-20] time interval, and assume there are m positive and m negative peaks in the 20 s segment, i.e., $\left|\mathscr{P}_{0-20}^{+}\right| = \left|\mathscr{P}_{0-20}^{-}\right| = m$ (the procedure is the same if the number of positive and negative peaks are not equal). Similarly, $\mathscr{A}_{0-20}^{+} = f_{\text{PPG}}\left(\mathscr{P}_{0-20}^{+}\right)$ and $\mathscr{A}_{0-20}^{-} = f_{\text{PPG}}\left(\mathscr{P}_{0-20}^{-}\right)$. Also $\mathscr{P}_{0-20} = \mathscr{P}_{0-20}^{-} \cup \mathscr{P}_{0-20}^{+}$ and $\mathscr{A}_{0-20} = \mathscr{A}_{0-20}^{-} \cup \mathscr{A}_{0-20}^{+}$ are the sets of all (positive and negative) peaks and their amplitudes in the 20 -second segment. Then:

$$\nabla P^{+} = \frac{1}{m-1} \sum_{i=2}^{m} \left| f_{\text{PPG}}\left(p_i^{+}\right) - f_{\text{PPG}}\left(p_{i-1}^{+}\right) \right|$$

$$\nabla P^{-} = \frac{1}{m-1} \sum_{i=2}^{m} \left| f_{\text{PPG}}\left(p_i^{-}\right) - f_{\text{PPG}}\left(p_{i-1}^{-}\right) \right|$$

$$\nabla P = \frac{1}{2m-1} \sum_{i=2}^{2m} \mid f_{\mathrm{PPG}}\left(p_i\right) - f_{\mathrm{PPG}}\left(p_{i-1}\right)$$

are the initial baseline values that will be used in the proposed algorithm. The value of ∇p is calculated based on the power spectrum of the 20 s segment, as the frequency between 1 and 3 Hz that has the highest power is the inverse of the heart rate frequency [27], i.e., $\frac{1}{\nabla p}$

In order to extract an initial template with positive peak T^+, first the $m-1$ positive-peaked pulses are sorted with respect to their pulse width. If the template pulse duration is chosen to be the mode of pulse duration (the most frequent pulse duration) in the 20-second segment, then the template T^+ can be calculated as the average of beats that have the same temporal duration as the mode of pulse duration. If the mode of pulse duration is not unique, the median of pulse duration (the middle value for pulse duration) in the 20 -second segment is chosen, and then the beats that have the width closest to the median of pulse duration will be aligned (e.g., by using DTW) or interpolated and then averaged to achieve the template T^4. The same procedure is applied to negativepeaked beats in order to extract the template with negative peak T^-.

2.3.3.2 Updating template and baseline values.

As mentioned above, the first T seconds of each waveform is used as the calibration phase to extract initial individual-specific templates and baseline values, with $T = 20$ seconds chosen for this paper. Since it's possible that the first segment is noisy, the initial baseline values and templates may be invalid. Thus, two criteria for accepting a segment as valid are imposed:

1. The number of positive or negative peaks should be more than $0.95 \times T$; i.e., on average a heartbeat should occur at least every 0.95 s.

2. At least one third of pulse widths (pulse durations) are within 5% of pulse

duration mode/median (as mentioned in the previous section, if the mode of pulse duration is not unique, the median of pulse duration is chosen for template width).

If both of these conditions are satisfied, the first T seconds are accepted for initial baseline values extraction, otherwise the T-second window is iteratively slid for 1 s until both conditions are satisfied (e.g., intervals of [0,20] [1,21], etc.).

Due to the non-stationary nature of the source, after the initial calculation of the baseline values and templates, an adaptive algorithm for updating these baseline values and templates is necessary, particularly if the PPG signal has long duration. As such, after calculating the features of each segment using the baseline values and templates of the previous segment, these baseline values and templates are then updated to be used in the subsequent segment. Similarly, an interval is accepted for updating the baseline values and templates if it also satisfies the two aforementioned criteria.

2.3.4 A Standard Algorithm

Through feature extraction, each sample is represented as $\mathbf{x} \in \mathbb{R}^6$

$$\mathbf{x} = \begin{bmatrix} \overline{\nabla p} \\ \overline{\nabla P^-} \\ \overline{\nabla P^+} \\ \overline{\nabla P} \\ D\left(T^+ \| B^+\right) \\ D\left(T^- \| B^-\right) \end{bmatrix}$$

where $\overline{\nabla p}, \overline{\nabla p^-}, \overline{\nabla p^+}, \overline{\nabla P}, D\left(T^+ \| B^+\right)$ and $D\left(T^- \| B^-\right)$ are the features described in §2.3.1. Using a standard algorithm based on decision rules, these values can be compared with thresholds for classification purposes. The hypothetical thresholds for a simplistic algorithm can be achieved experimentally using training data. After choosing the thresholds, the six features can be used to classify each beat, more specifically each interval between any two peaks (positive to negative peaks or negative to positive peaks), into a good or poor quality interval.

In this case, a standard algorithm assigns the "poor quality" label to each interval between two consecutive peaks if any of the six features are greater than the threshold. Formally speaking, for an interval set $\mathscr{P}^{p_i}_{p_{i-1}} = [p_{i-1}, p_i]$ between any two consecutive peaks (note that $\mathscr{I}^{p_i}_{p_{i-1}} \subseteq \mathscr{T}\Big)$ define the feature function $f_{\text{feature}} : \mathscr{T} \times \mathscr{U} \to \mathbb{R}^6$ as a function from the set of time samples and bounded set of PPG values to the feature space. The set of poor quality intervals is then:

$$\mathscr{T}^{\text{Poor}} = \Big\{ t_j \mid \exists \mathscr{F}^{p_i}_{p_{i-1}} \subseteq \mathscr{T} \; s.t. t_j \in \mathscr{F}^{p_i}_{p_{i-1}},$$

$$f_{\text{feature}} \left(\mathscr{F}^{p_i}_{p_{i-1}}, f_{\text{PPG}} \left(\mathscr{F}^{p_i}_{p_{i-1}} \right) \right) \not< \tau \Big\},$$

where $\tau = [\tau_1, \ldots, \tau_6] \in \mathbb{R}^6$ is a vector of thresholds on the six features, and the inequality is performed component-wise. Obviously the set of good quality signal is the complement of $\mathscr{T}^{\text{Poor}}$, i.e. $\mathscr{T}^{\text{Good}} = \left(\mathscr{T}^{\text{Poor}}\right)^C = \mathscr{T} - \mathscr{T}^{\text{Poor}}$. This algorithm is used later for threshold optimization.

2.3.5 Interval Classification and Signal Quality Index

One of the primary reasons for measuring PPG quality and reliability is that other important signals such as oxygen saturation utilize PPG in their formation. In general, oxygen saturation values are averaged over a moving window of PPG signal. In the ARDS dataset, the pulse oximetry hardware (PPG recording device) calculates every value of oxygen saturation based on the last 8 s of PPG signal. Consequently, having isolated poor quality beats/intervals is insufficient to label a

PPG segment as "poor quality." Thus, for an interval $\mathscr{T}_{t_k}$ of length 8 s such that $\mathscr{T}_{t_k} = \{t_i \mid t_i \in \mathscr{T}, t_k - 8 < t_i \leq t_k\}$, the signal quality index (SQI) for that window is defined as:

$$\mathrm{SQI}\left(\mathscr{T}_{t_k}\right) = 1 - \frac{\left|\mathscr{T}_{t_k} \cap \mathscr{T}^{\mathrm{Poor}}\right|}{\left|\mathscr{T}_{t_k}\right|} = \frac{\left|\mathscr{T}^{\mathrm{Good}}\right|}{\left|\mathscr{T}_{t_k}\right|},$$

which is always a number between zero and one. As discussed in section 3.4, the SQI for any given interval will be compared with a pre-determined rate (threshold) for classification.

2.3.6 Learning Models and Decision Rules

In this paper, two different training/testing frameworks are considered:

1. A standard learning method in which a single model is trained on 6-dimensional samples (Figure 2.4a).

2. Six similar models that are trained on each sample feature separately, followed by a decision rule (Figure 2.4b).

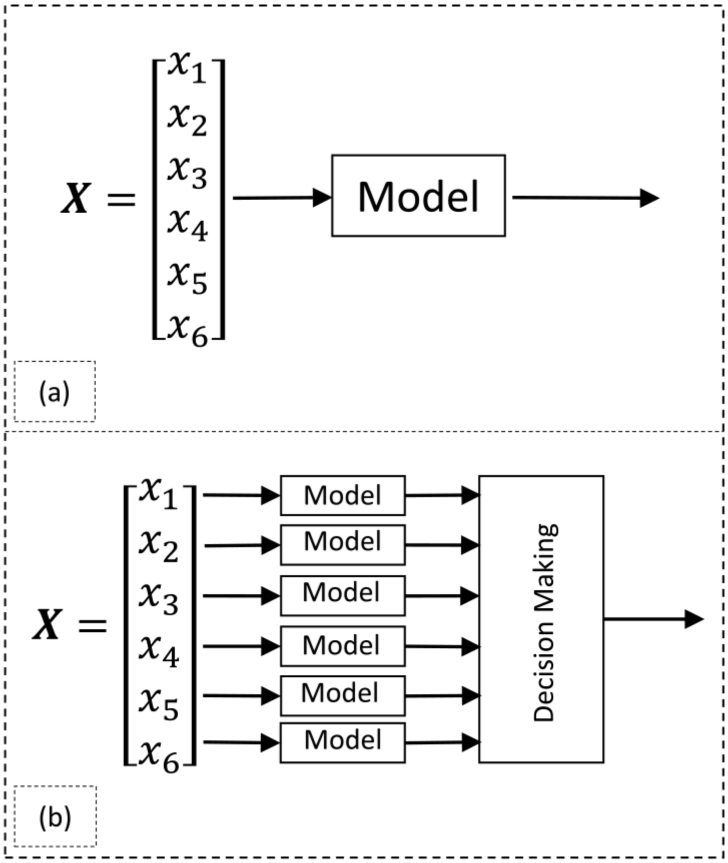


Figure 2.4: Training/testing frameworks used for the initial study: **(a)** a framework for training/testing model on 6-dimensional samples **(b)** a framework for training/ testing six similar models on each 1-dimensional sample feature followed by decision rule, which basically is a logical “or” operation on the six outcomes.

The principle reason for considering the second model is the nature of the proposed normalized features. For a normal PPG beat all the features are expected to be close to zero; while for a poor quality interval, the absolute value of at least one of these features is expected to be greater than a threshold.

To support this argument, Figure 2.5 represents the cumulative distribution function (CDF) of the absolute value of the normalized negative-to-negative peak jump feature $\left|\overline{\nabla P_i^-}\right|$ for both classes. This figure shows that the larger the value of $\left|\overline{\nabla P_i^-}\right|$, the worse the quality, and this is valid for all the features. Thus, in the second framework, the decision rule is simply a logical “or” operation on the outcomes of each trained model on individual features.

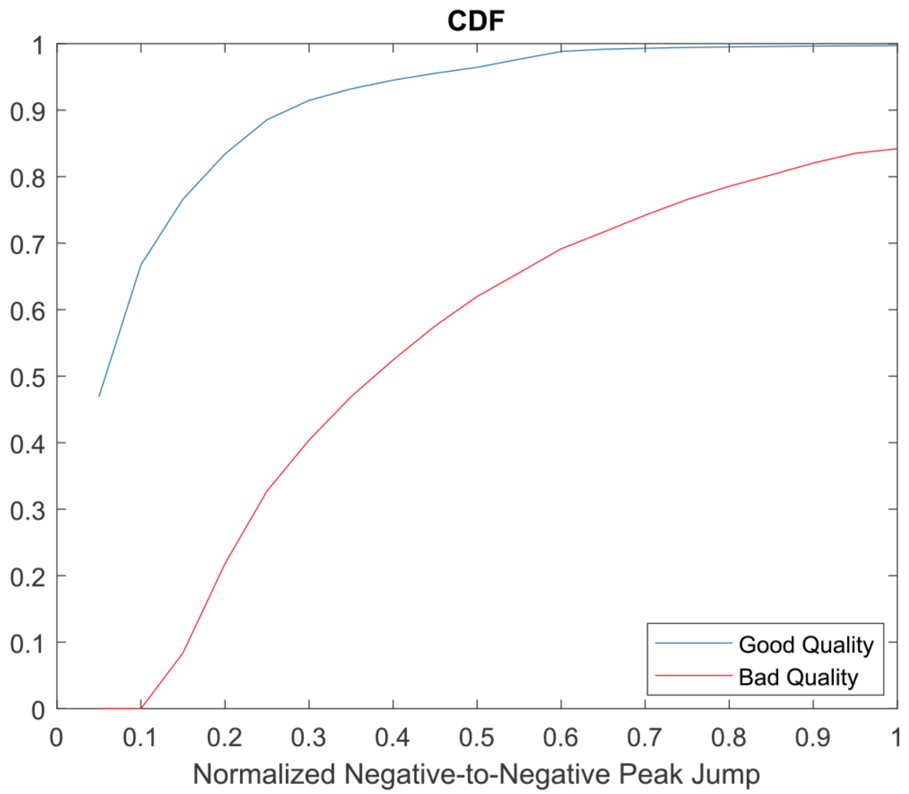


Figure 2.5: Cumulative distribution function of $\overline{\nabla P_i^-}$, the normalized negative-to-negative peak jump.

2.4 Accounting for Label Uncertainty

After developing methods to process the time-series data and performing signal quality assessment, the next step in developing the proposed machine learning model is to account for uncertainty of the classification labels. Data in numerous applications of machine learning may have uncertainty in their training labels or the input data (i.e., features) might be corrupted by uniform/non-uniform noise distribution [50]. For example, in medical applications, precisely determining a prognosis might be difficult due to erroneous lab results or a shirt in baseline perspective as patients may vary in describing their symptoms. As a result, some cases used in algorithm training may be mislabeled, adversely affecting the algorithm's performance.

The diagnosis of ARDS is made based on clinical criteria, but even clinical experts may disagree or have uncertainty about the diagnosis in some patients. However, experts may also be able to quantify their diagnostic uncertainty in these cases. As previously mentioned, label uncertainty can be incorporated as weights within a SVM model. These weights determine the relative penalty of misclassification of training set points. Information about the uncertainty level of an ARDS diagnostic label provided by clinical experts can also prove useful when training a system for ARDS detection. We present a robust method implemented with support vector machines (SVM) to account for such clinical diagnostic uncertainty when training an algorithm to detect patients who develop ARDS.

A group of expert clinicians reviewed all patients for the development of ARDS based on the Berlin definition [4]. As ARDS is a clinical diagnosis without a gold standard, we were unable to "benchmark" expert performance. However, because the inter-rater reliability of ARDS diagnosis is known to be poor in patients with acute hypoxic respiratory failure [51], these patients were reviewed independently by 3 experts, and their ratings were averaged. In addition to determining whether the diagnosis was present (yes or no) and record the time of ARDS onset among positive cases, the experts were also asked to provide their confidence level in the diagnosis label (high, moderate, low, equivocal). This 4-point confidence scale was carefully tested on the experts prior to use in this study, and felt to reasonably capture the range of uncertain that they might have when reviewing patient cases. Their diagnosis label and confidence level could then be converted to a $1-8$ scale, as depicted in Figure 2.6, where $1 =$ no ARDS with high confidence, $8 =$ ARDS with high confidence.

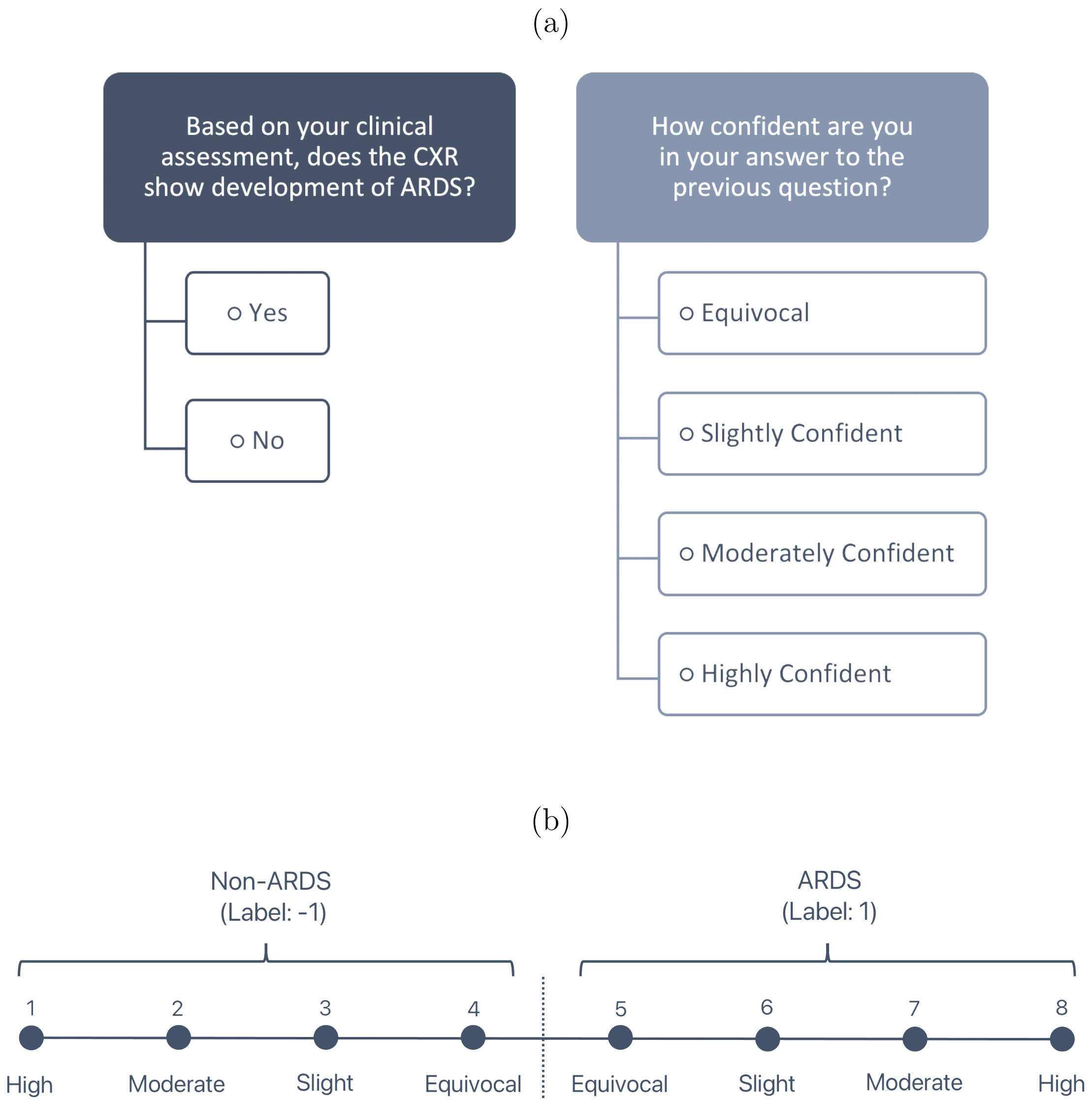


Figure 2.6: Accounting for label uncertainty and label generation. **(a)** Multiple expert clinicians were asked to independently review patients' data and determine if any individuals had ARDS. Clinicians were also rated their confidence of the diagnosis as equivocal, slightly confident, moderately confident, or highly confident. **(b)** The diagnosis and confidence were converted to a scale between 1-8. The final label was generated from aggregating these reviews to ensure correctness and consistency of the diagnosis. A label of -1 (non-ARDS) was assigned if the averaged review was below or equal to 4.5, and a label of 1 (ARDS) was assigned if the averaged review was above 4.5

2.4.1 Support Vector Machines

A support vector machine (SVM) is a supervised machine learning algorithm based on the idea of finding a hyperplane that best divides a dataset into two classes [52].

For a binary classification task, the hyperplane can be represented as a line that linearly separates the two classes in a set of data. The SVM uses a subset of the training data as support vectors to determine the optimal decision boundary, which can be further adjusted to control the acceptable rate of miss-classification. SVMs generally tend to work well on smaller datasets due to efficient usage of a subset of training data as support vectors - reducing the tendency of larger machine learning models to overfit. Furthermore, kernel functions can be used to transform the data input into higher dimensional planes for better separability.

For this study, we train a linear SVM model to discriminate between an ARDS and non-ARDS diagnosis. Given a set of training data:

$$\left(\mathbf{x}_1, y_1\right), \ldots, \left(\mathbf{x}_n, y_n\right) \quad \mathbf{x}_i \in X, y_i \in \{-1, 1\}$$

SVM first maps the training sample vector $\mathbf{x} \in X$ into a vector (space) $\mathbf{z} \in Z$. It then constructs the optimal separating hyperplane by learning the decision rule $f(\mathbf{z}) = \mathbf{w} \cdot \mathbf{z} + b$ where $\mathbf{w}$ and b are the hyperplane parameters and the solution of

$$\begin{gathered} \min_{\mathbf{w}, b, \xi} \tfrac{1}{2}\|\mathbf{w}\|_2^2 + C \textstyle\sum_{i=1}^{n} \xi_i \\ \text{s.t. } \forall 1 \leq i \leq n, y_i\left(\mathbf{w} \cdot \mathbf{z}_i + b\right) \geq 1 - \xi_i \\ \forall 1 \leq i \leq n, \xi_i \geq 0 \end{gathered} \tag{2.1}$$

where $C > 0$ is a hyperparameter and ξ_i is the slack variable. C is a regularization parameter to influence the slack variable, which allow certain constraints to be violated (i.e., perfect linear separability). C essentially controls the level of acceptability for misclassifying each training example. For large values of C, the optimization will choose a smaller-margin hyperplane if that hyperplane does a better job of getting all the training points classified correctly. Conversely, a very small value of C will cause the optimizer to look for a larger-margin separating hyperplane, even if that hyperplane misclassifies more points. Grid search [53] was used to determine the optimal value of C over $\in \{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$.

Using Lagrangian multipliers, the dual optimization problem of 2.1 is:

$$\max_{\boldsymbol{\alpha}} D(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K_{i,j}$$
$$\text{s.t. } \sum_{i=1}^{n} y_i \alpha_i = 0$$
$$\forall 1 \leq i \leq n, 0 \leq \alpha_i \leq C$$

where $K_{i,j} \triangleq K(\mathbf{z}_i, \mathbf{z}_j)$ is the kernel in the decision space with the decision function

$$f(\mathbf{z}) = \mathbf{w} \cdot \mathbf{z} + b = \sum_{i=1}^{n} y_i \alpha_i K(\mathbf{z}_i, \mathbf{z}) + b \tag{2.2}$$

2.4.2 Incorporating Label Uncertainty

To investigate learning with uncertain data, we briefly explore the changes in the learning formulation when moving from the standard machine learning to learning with uncertainty. Almost all existing machine learning paradigms use the following learning scheme: given a set of training examples $(x_i, y_i), i = 1, \ldots, N,)$ in which $x_i \in X$ form the input attributes/features and $y_i \in \{-1, 1\}$ is the output class, find a function $f_\gamma(x) : X \to \{-1, 1\}$ (where $\gamma \in \Lambda$ represents the set of parameters used in the function) to learn / generalize a mapping between the input and the output. In the training phase, the parameters γ that minimizes a cost function, defined over the training examples, identifies the solution $f_\gamma(x)$. Such a cost function often considers both the magnitude of the estimation error on the training data and the complexity of the model (to avoid overfitting).

In learning with uncertainty, the training examples while some examples have certain labels, i.e. $(x_i, y_i), i = 1, \ldots, N_1$, the rest have some level of uncertainty on their labels, i.e. $(x_i, y_i, l_i), i = N_1, \ldots N$, where $0 < l_i \leq 1$ is the level of confidenc (i.e., lack of uncertainty) over sample i. While the exact formulation of learning with uncertainty might vary, the SVM-based formulation might be the most insightful approach. Intuitively speaking, note that in the standa SVM the main objective is to find parameters w and b in a hyperplane $w^T x + b$ that separates positive and negative examples while keeping the value of w small (controlling the complexity of model).

In the formulation of SVM with uncertainty, as described in [35], the following optimization problem is solved:

We implement the following formulation of SVM to account for label uncertainty in the classification model in the following manner:

$$\min_{w,\xi} \frac{1}{2} \parallel w \parallel^2 + C \sum_{i=1}^{N} z_i \xi_i$$

subject to:

$$\begin{cases} y_i(w^T x_i + b) \geq 1 - \xi_i, \quad i = 1, ..., N \\ \quad \xi_i \geq 0 \end{cases} \tag{2.3}$$

where:

$$z_i = (|l_i - \alpha| - \beta) * \gamma + \delta$$

This formulation incorporates the slack variable ξ_i to permit some misclassification and also includes the penalty parameter C to establish soft-margin decision boundaries because ARDS and non-ARDS examples are not linearly separable. In this implementation, support vectors that are based on patients' data with high label confidence are given more weight and influence in the SVM decision boundary. Uncertainty in the label (l_i), as shown in Figure 2.6, is incorporated within (z_i) to directly influence C. The formula for z_i combines two linear transformations for uncertainty in the label annotation (l_i) and generate a scalable weight to that specific observation.

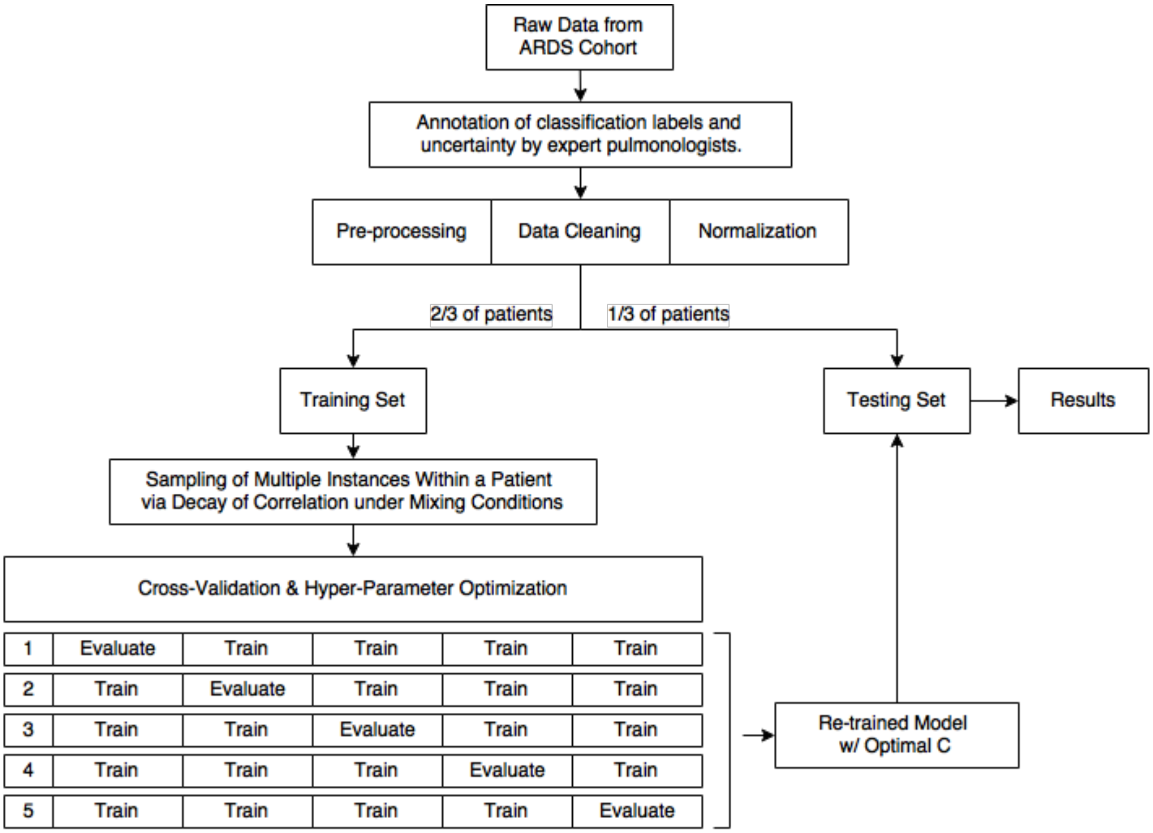
In this application, we set $\alpha = 4.5$, $\beta = 3.0$, $\gamma = 20$, and $\delta = 90$, which scales l_i, with a range of 1-8, into the weighting z_i, with a range between 40-100 in increments of 20. As a result, labels with high confidence (eg. $l_i = 1$ or 8) receive the weight $z_i = 100$, while equivocal labels (eg. $l_i = 4$ or 5) receive the weight $z_i = 40$. z_i is then normalized to 1. This formula for z_i adjusts sample weighting based on l_i and rescales the C parameter as C_i for each observation in a patient's data structure so that the classifier puts more emphasis on points with high confidence.

2.5 Application

In this study, the primary learning algorithms we compare are linear SVM with and without label uncertainty. 5-fold cross validation was performed on the training data to find the optimal value of the hyper-parameter C using grid search [53] over $C \in \{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$. We then re-trained the model on the entire training set using this optimal C parameter. This updated model was then used to classify patients in the hold-out dataset using all their data (e.g., no sampling was performed on the holdout data). The model predictions for each patient in the holdout sample (e.g., ARDS = 1 or -1) are then compared against the label assigned by the majority of experts reviewing the patient. We also compare the performance of our proposed SVM method with Logistic Regression and Random Forest (using the same subsampled training/testing bins and 5-fold cross validation partitions) to determine if the achieved results are equivalent or superior to other state-of-the-art methods.

A simplified protocol of this analysis, including data pre-processing, sampling from the training data to limit inter-correlation, hyper-parameter optimization with 5-fold cross-validation, and hold-out testing is shown in the flowchart of Figure 2.7.

To ensure that our proposed sampling strategy and threshold still maintains for SVM with label uncertainty, we repeat the previous analysis to show the effect of


Raw Data from ARDS Cohort
Annotation of classification labels and uncertainty by expert pulmonologists.
Pre-processing
Data Cleaning
Normalization
2/3 of patients
1/3 of patients
Training Set
Testing Set
Results
Sampling of Multiple Instances Within a Patient via Decay of Correlation under Mixing Conditions
Cross-Validation & Hyper-Parameter Optimization
Re-trained Model w/ Optimal C

1	Evaluate	Train	Train	Train	Train
2	Train	Evaluate	Train	Train	Train
3	Train	Train	Evaluate	Train	Train
4	Train	Train	Train	Evaluate	Train
5	Train	Train	Train	Train	Evaluate

Figure 2.7: Flowchart of the study's experiment design, with 5-fold cross-validation and hyper-parameter optimization using grid search. All samples from the same patient are kept exclusively in either the training or testing set. Hyper-parameter optimization was implemented for separately each model (with and without label uncertainty weight) to give an accurate assessment of performance.

different sampling thresholds on prediction generalizability. Figure 2.8 confirms that optimal results are achieved when the sampling threshold is approximately 0.7, which supports the previous analysis and the literature suggested value of $\frac{1}{\sqrt{2}}$.

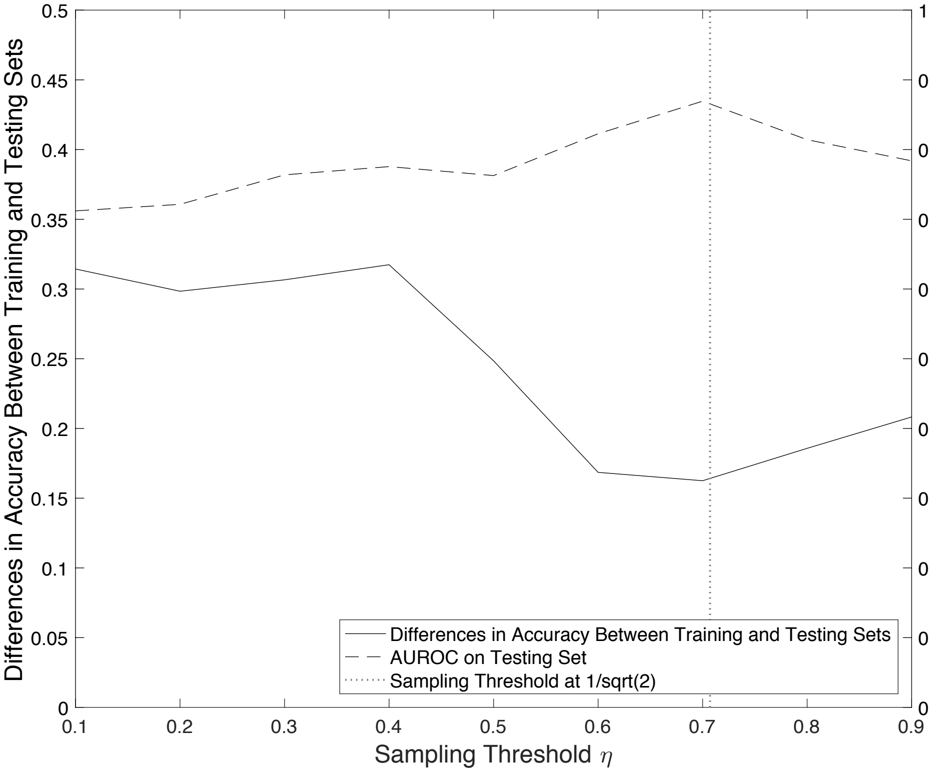


Figure 2.8: Effects of different sampling thresholds on prediction generalizability. We confirm that the sampling strategy and threshold effects observed in Figure 2 is maintained when the SVM model is formulated to account for label uncertainty.

2.5.1 Data

The patient cohort included consecutive adult patients hospitalized in January of 2016 with moderate hypoxia, defined as requiring more than 3 L of supplemental oxygen by nasal cannula for at least 2 hours. The cohort was enriched with additional patients who developed acute hypoxic respiratory failure (PaO_2/FiO_2 ratio of < 300 mm Hg while receiving invasive mechanical ventilation) in February and March of 2016 who are higher risk for developing ARDS.

A total of 401 patient cases were available from the study cohort. Within this dataset, 48 were positive for ARDS and the remaining 353 were negative. Two-thirds of the patients were used in the model training process while the remaining one-third were kept as a hold-out set for testing. All samples from the same patients are kept exclusively in either the training or testing set (not both) to avoid bias in the data.

In patients who developed ARDS, data collected before the time of onset were labeled as "no ARDS," while data collected after the time of onset were labeled as "ARDS." In total, 48 of the patients in the cohort were diagnosed with ARDS with a confidence of 5 or higher after expert review.

Data was first normalized to prevent features with large dynamic ranges from dominating the separating hyperplane. Then the training data was sampled using the proposed sampling method described previously to minimize correlation between data points on the same patient. Prior to sampling, the training set contained 13,722 total instances, 736 of which were positive. After sampling, there were 1,893 total instances, 736 of which were positive.

Time-stamped vital signs and laboratory values were extracted from each patient's electronic health record (EHR) from the first six days of hospitalization and included as clinical features (covariates) to train the ARDS algorithm. Only routinely acquired vital signs and laboratory values with potential for association with ARDS were included, based on guidance from clinical experts. The covariates used as features are provided in Appendix A and Table A.1. This approach minimized the total number of features in the model to 24 variables commonly used in clinical practice and statistical feature selection techniques were not utilized prior to model training. Patients were observed every 2 hours with previous data carried forward until a new value was recorded. If clinical data was missing on a patient because the vital sign or laboratory tests was not performed, it was imputed as a normal value. This is standard approach when developing clinical predictions models and assumes data is not collected because the treating clinician had a low suspicion that it would be abnormal [54, 55].

EHR data also include records such as chief complaint, medications, comorbidities, age, gender, injury scores, and laboratory results. Most of these attributes are measured at relatively fixed time intervals that are much less frequent than the rate at which other physiological signals are recorded. In the database we will use for this project, almost all these records are available for all patients.

2.5.2 Results

Signal quality assessment for beat-scale and fixed interval-scale analysis are presented in Table 2.1, Table 2.2, and Figure 2.9. The decision tree model and threshold optimization both used the second framework in their learning process, while the ensemble of decision trees and SVM used the first framework. The ensemble of decision trees and threshold optimization are the two algorithms that used under-sampling. Both of these methods were also significantly faster to train than those which used the first framework. In comparing model performance on the training and testing datasets, the tree based algorithms overfitted the training data, while SVM and the threshold optimization algorithm have almost the same performance on both datasets.

Table 2.1: Performance comparison of decision tree, the ensemble of decision trees, SVM, and threshold optimization in beat-scale analysis.

		DT	EDT	SVM	TO
Undersampling		No	Yes	No	Yes
Framework		Two	One	One	Two
Running time (sec)		5	450	375	20
Train	Accuracy	96.92	100	85.37	80.02
	Sensitivity	99.92	100	86.05	82.82
	Specificity	96.70	100	85.31	80.67
Test	Accuracy	75.02	88.85	83.02	80.66
	Sensitivity	73.01	70.03	85.45	82.38
	Specificity	75.14	90.04	82.82	80.50

DT = decision trees, EDT = ensemble of decision trees, SVM = support vector machines, TO = threshold optimization. The running time is the average time needed to train the algorithms on 31 30-min PPG signals and test on 15 30-min PPG signals.

Table 2.2: Comparison of decision tree, the ensemble of decision trees, SVM, and threshold optimization in interval-scale analysis.

	DT	EDT	SVM	TO
Sensitivity	88.96	91.56	93.25	90.05
Specificity	86.30	91.97	91.90	89.45
Corresponding Rate	0.45	0.40	0.70	0.65

DT = decision trees, EDT = ensemble of decision trees, SVM = support vector machines, TO = threshold optimization. Please note that the rate in the table corresponds to the best performance.

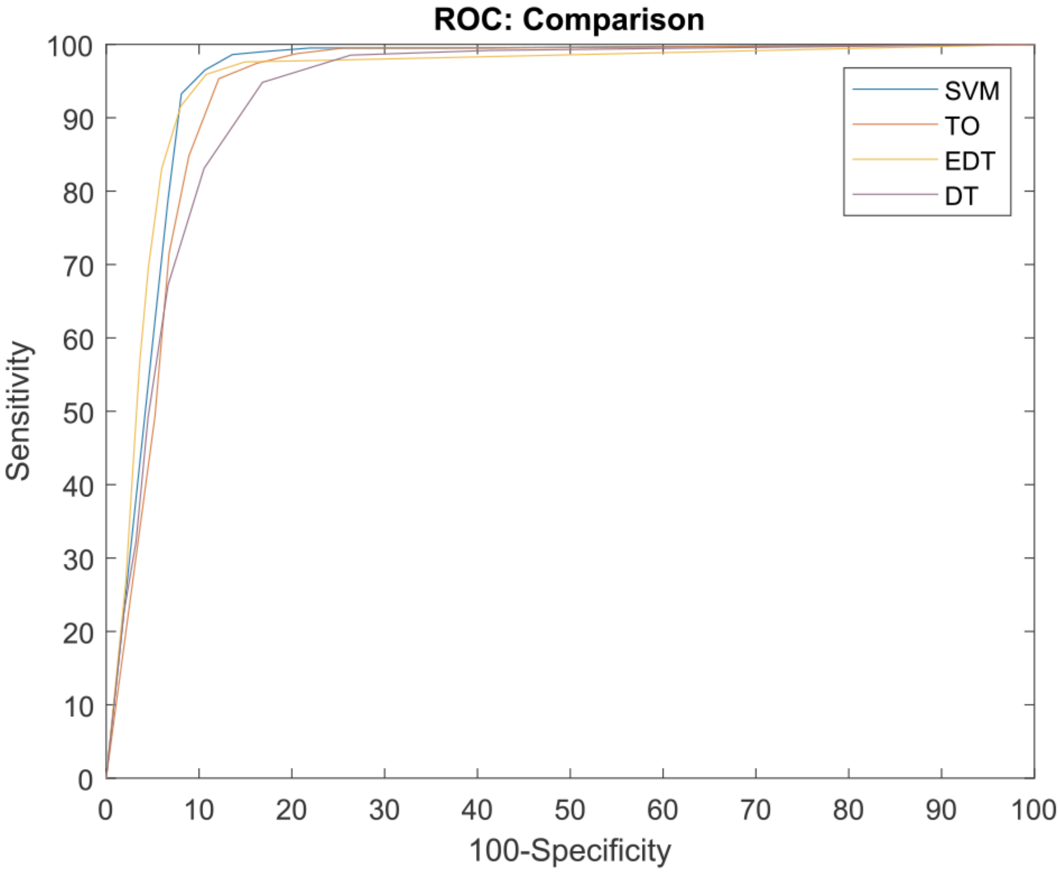


Figure 2.9: ROC curves of the methods used for signal quality assessment.

The best results achieved with SVM yielded a sensitivity of 93.25% and specificity of 91.90% for a rate of 0.7. Figure 2.10 depicts a visual example of the SVM model used for classifying PPG signal quality on the three intervals. In comparison, the best performance of the decision tree model for fixed interval-scale analysis yields a sensitivity of 88.96 and specificity of 86.30 for a rate of 0.45. The best result achieved with an ensemble of decision trees is sensitivity of 91.56 and specificity of 91.97 with rate of 0.40.

In the first interval (0–8 s) both the algorithm and annotation have poor quality segments in beat-scale, which is less than 5.6 (8 × 0.7) seconds; thus, this interval is not considered poor quality by both the algorithm and the annotation. The second interval (8–16 s) had more than 5.6 s of poor quality beat-scale segments using the algorithm, but slightly less than 5.6 s of poor quality beat-scale segments using the annotation, therefore this interval is labeled as poor quality using the algorithm, but not using the annotation. The last interval (16–24 s) has more than 5.6 s poor quality beat-scale segments in both algorithm and annotation.

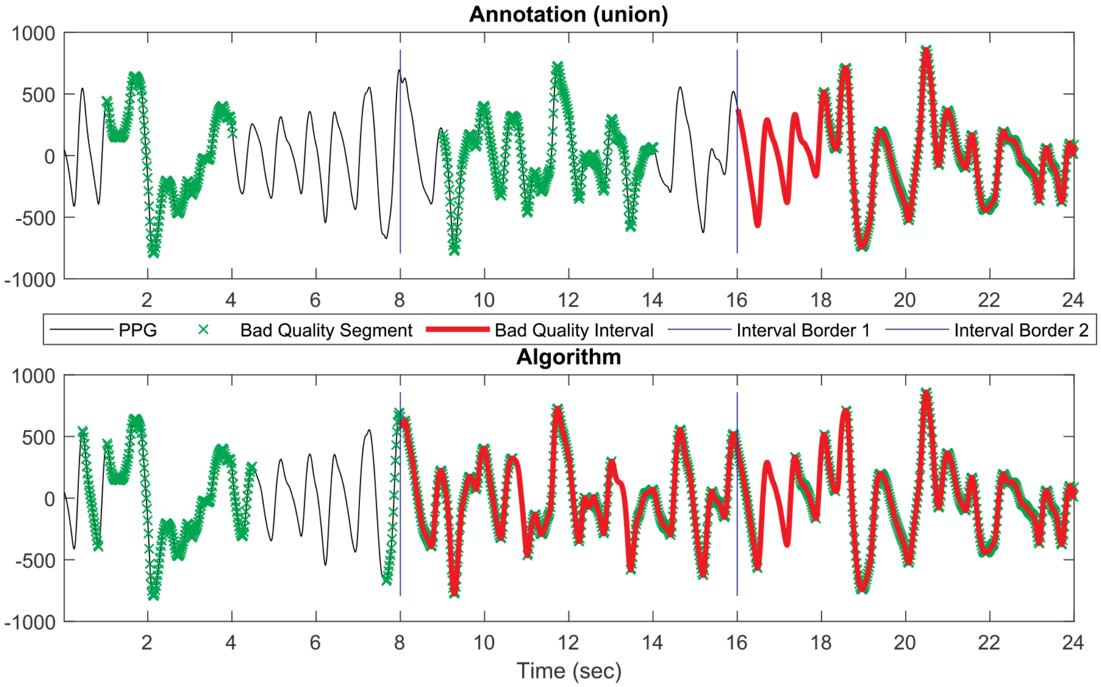


Figure 2.10: Signal quality assessment on fixed interval-scale segments. This example uses an SVM model with rate (threshold on interval SQI) 0.7.

In addition to Table 2.1, the effect of uniform and non-uniform undersampling has been tested on SVM and decision tree: non-uniform undersampling used in threshold optimization reduces the performance of both algorithms, while uniform undersampling has no significant effect on SVM (in its cost-sensitive SVM formulation) and a negative effect on decision tree performance. Based on Figure 2.9 and Table 2.2, SVM and the ensembles of decision trees outperform the other two methods in the fixed interval-scale analysis. Overall, the cost-sensitive SVM with Gaussian kernel outperform the rest, while the proposed threshold optimization is significantly faster.

The proposed thresholded correlation decay method (§2.2.2) is applied to the dataset used for model training. The average correlation decay for each patient's data is shown in Figure 2.11. On average, the correlation between X_t and $\langle X_t \rangle$ drops below η at a distance in time of around 22 hours. Figure 2.12 shows the decay of correlation to be different when the data was analyzed separately according to the classification label: decay of correlation is observed when ARDS = -1 but not observed when ARDS = 1.

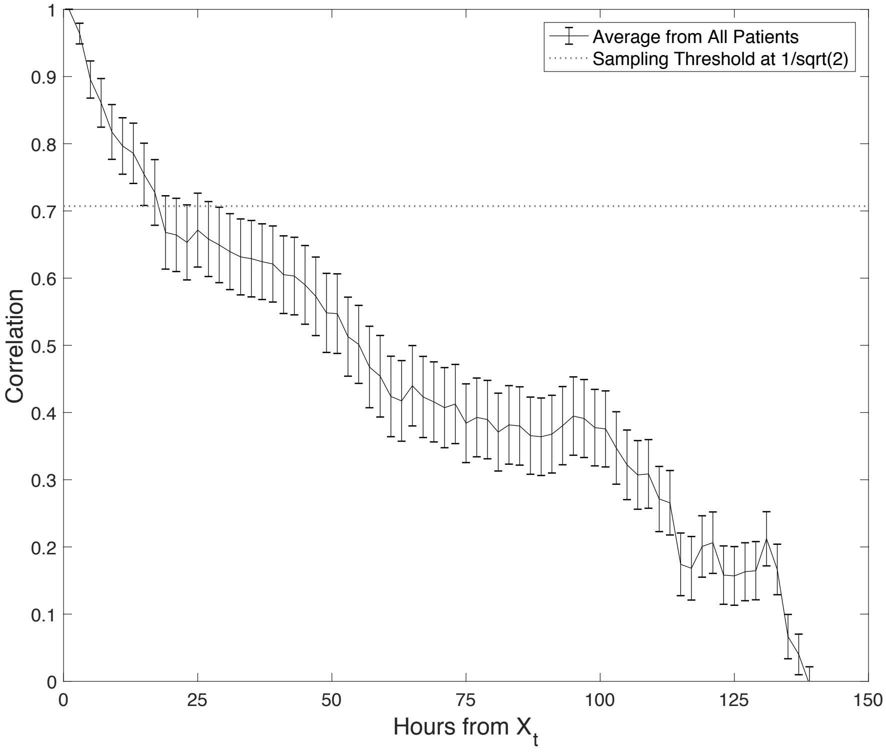
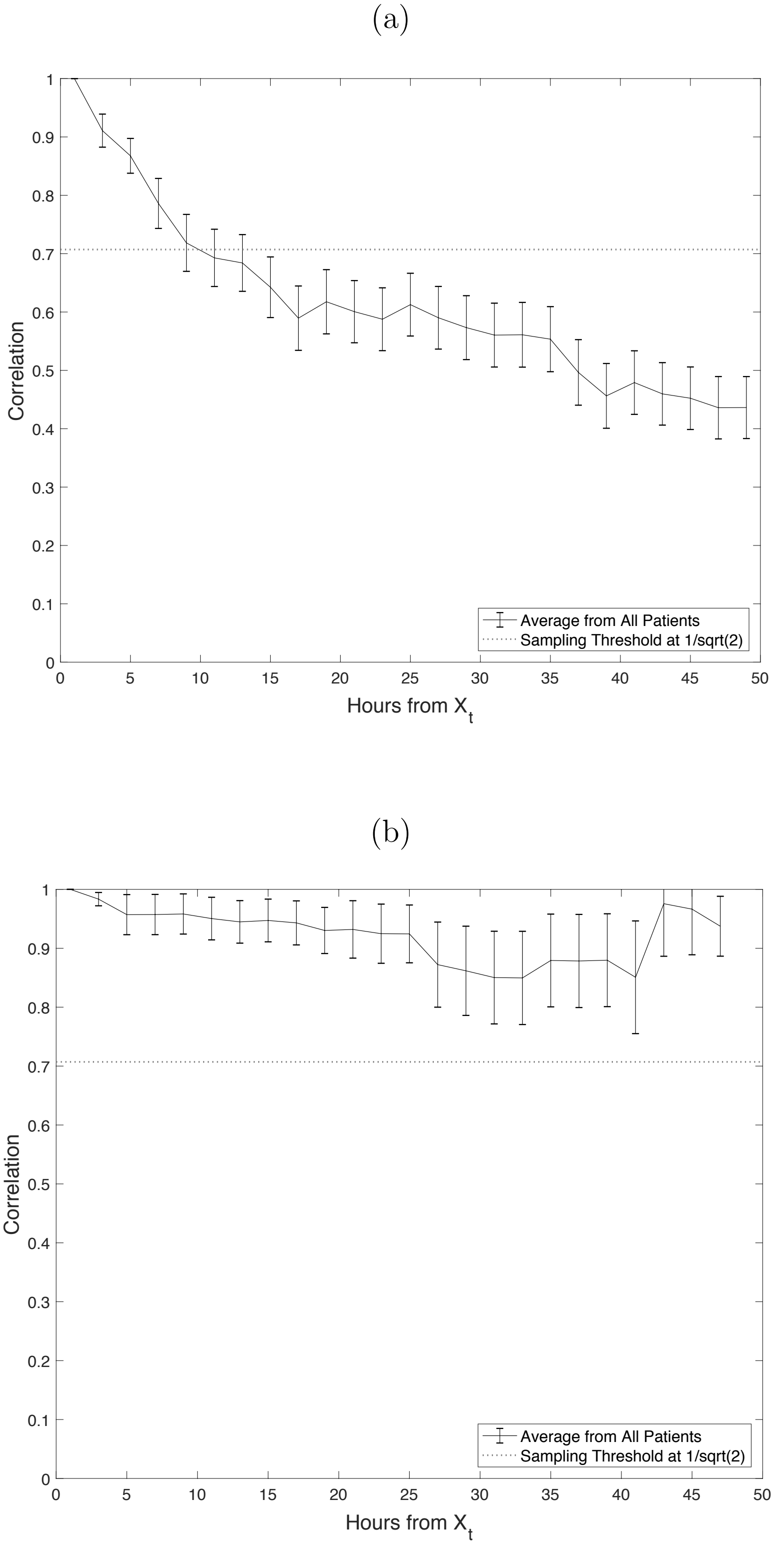


Figure 2.11: Average decay of correlation from all patients. Error bars represent standard error of the mean and each point represents correlation in relation to time (hours) from the initial observation sampled on each patient.



(a)

(b)

Figure 2.12: Decay of correlation from all patients stratified by classification label. **(A)** depicts the average correlation decay during negative diagnosis of ARDS and **(B)** corresponds to the positive diagnosis of ARDS. Error bars represent standard error of the mean and each point represents correlation in relation to time (hours) from the initial observation sampled on each patient.

Therefore, the sampling under η approach was performed on the data when ARDS = -1, which reduce the number of negative examples for model training. Due to the lower number examples, and lack of correlation decay when ARDS = 1, sampling was not performed as it would have further exacerbated the class imbalance.

We benchmarked our proposed SVM method utilizing uncertainty in the label to SVM with a misclassification cost function proportional to the weight of imbalance in the datasets and other standard classification models, such as Random Forest and Logistic Regression, in Table 2.3. We also compared our sampling strategy to an alternative method that utilizes random sampling on negative examples to yield a 2:1 negative to positive ratio from each patient to provide a more balanced dataset - the results for this approach is provided in Table 2.3. In addition, we also examined performance without sampling (using all available data).

Table 2.3: Performance of logistic regression, random forrest, SVM, SVM with a class-weighted cost function, and SVM with label uncertainty.

	Sampling Based on the Proposed Correlation Decay Method				Random Sampling for Balanced (2:1) Training Data		No Sampling	
	Accuracy	AUROC	Specificity at 95% Sensitivity	Specificity at 90% Sensitivity	Accuracy	AUROC	Accuracy	AUROC
Logistic Regression	72.63	72.65	30.07	42.67	69.82	69.79	66.21	64.54
Random Forrest	74.34	74.88	33.92	47.51	71.11	72.54	68.73	69.03
Support Vector Machines	74.92	75.42	37.97	51.14	72.53	73.61	69.20	71.52
SVM w/ Class Weight	78.04	81.13	45.71	59.18	74.78	77.03	70.94	71.22
SVM w/ Label Uncertainty	81.57	85.48	52.85	64.50	76.98	79.89	71.88	74.31

When the SVM was trained to account for uncertainty in the label, we observed over 10% improvement of AUROC (0.8548 versus 0.7542) compared to the conventional SVM learning algorithm (Figure 2.13) when judged in the holdout sample. When the algorithms were benchmarked at a sensitivity of 95% and 90% (to ensure few ARDS cases are missed), the SVM model that accounted for label uncertainty also had improved specificity and outperforms the standard model. These sensitivity levels were set to high levels because it is important clinically for a model to have a high sensitivity and not miss cases of ARDS.

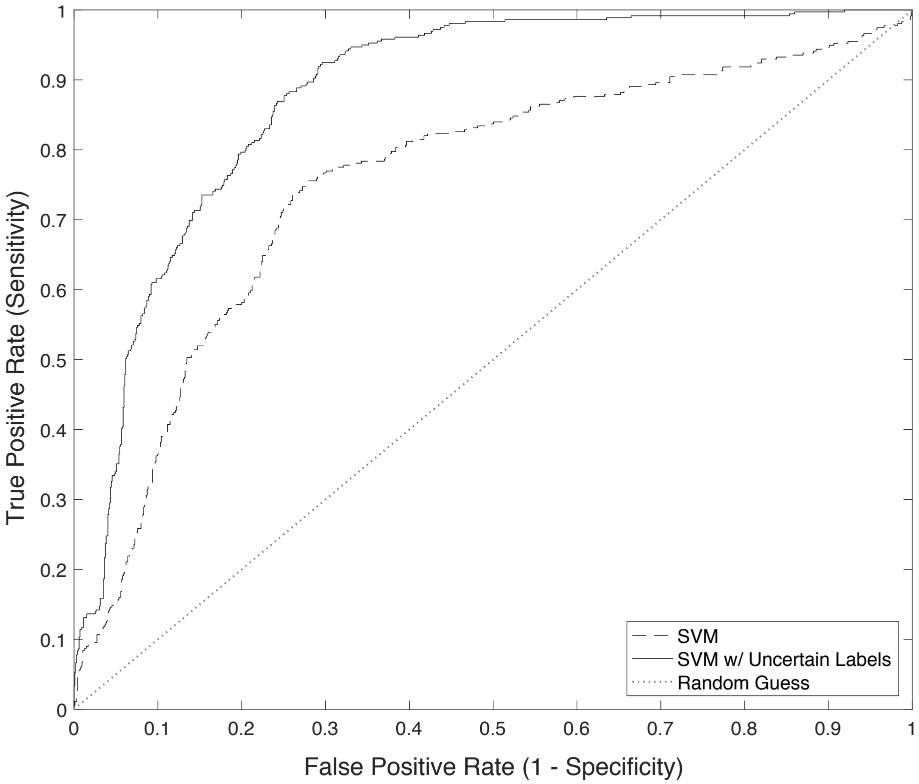


Figure 2.13: ROC curves comparing SVM with and without label uncertainty. Performance metrics are reported in Table 2.3.

2.6 Discussion

We present a robust machine learning algorithm to detect Acute Respiratory Distress Syndrome among hospitalized patients using routinely collected electronic health record data. We report an increase of approximately 10% in AUROC in a hold-out data set when label uncertainty is incorporated in the learning process as a weight on classification penalty, when compared to a conventional SVM learning model.

Our proposed SVM model was trained by incorporating a clinical expert's uncertainty in each patient's classification label as a constraining weight of confidence on the SVM's box constraint. Rather than treating label uncertainty as stochastic noise, this approach leverages information about the degree of uncertainty of each label, as provided by clinical experts, to improve the efficiency of model training. Our implementation of label weighting (z_i) directly influences the C parameter and rescales the cost of misclassification according to uncertainty associated with each label (l_i). Support vectors that are based on the data from patients with high label confidence are given more influence in the SVM decision boundary while instances with more uncertainty are assigned less weight when determining the SVM hyperplane. In future works, alternative mappings between the label uncertainty (l_i) provided by clinical experts and label weighting (z_i) used to find the SVM decision boundary should also be explored.

In addition, we performed a novel time-series sampling method, guided by the theory of mixing in stochastic processes, to limit the amount of correlation among data points on the same patient over time. Due to the time-series structure of a patient's longitudinal health data, each instance is not independent from another. We explored whether the data could be represented under mixing conditions and implemented a novel sampling strategy for minimizing inter-correlation among data points in the training data. For the data to be represented under mixing conditions, the correlation between data on the same patient should decay over time such that

$C_{F,G}(n) \to 0$ as $n \to \infty$. A plot of the correlation function of the data in Figure 6 supported this assumption overall, but not for the data with a classification label of ARDS = 1.

It may not be appropriate to assume all data types can be represented under mixing conditions, therefore, plotting the correlation function of the data is essential prior to utilizing the sampling algorithm. When patients were diagnosed with ARDS, we found their data to have very high inter-correlation with little observable decay – indicating a strong mixing process. Therefore, the proposed sampling method would have been unsuccessful in reducing inter-correlation and would yield very little data instances available for training. This finding made sense when interpreted from a clinical point of view. When a patient is admitted to the emergency room for pulmonary injury (eg. sepsis) and has not yet reached the critical stage of ARDS, their condition rapidly changes as a result of clinical intervention or decline of health, resulting in less stability and inter-correlation in the recorded data. If the patient develops ARDS, less rapid change in the data would be observed since ARDS is recognized as the "final pathway" of pulmonary damage [56].

Since there were significantly more negative than positive examples, we decided against using the sampling strategy when ARDS = 1, which ensured a more balanced number of positive and negative examples in the training data. As minimal correlation decay was observed among the data when ARDS = 1, implementing the sampling strategy for those data instances would have led to further imbalance among positive and negative examples, and limited the model's ability to learn a good decision boundary. Our sampling approach utilized a pairwise correlation distance matrix to quantify dependency within the data structure. There are many ways to quantify the measurement of dependency between X_t to $\langle X_t \rangle$. Bradley et al provides a comprehensive list of mathematical definitions for dependency coefficients to define these mixing conditions [57] and measure decay of correlations [42]. In the future work, we will perform a more comprehensive examination of the data structure using formalized definitions of mixing, such as quantifying dependency with the α-mixing coefficient.

Our sampling method outperforms using all available data (no sampling) from the EHR by producing a much balanced dataset for training and minimizing dependencies in each patient's time series data, making it closer to the state of being i.i.d. We also compared our sampling algorithm to randomly sampling on negative examples to yield a 2:1 negative to positive ratio from each patient. This random sampling method also provides a balanced dataset for training, and as a result, we observed an increase in accuracy and AUROC from all algorithms when compared to training without sampling. However, compared to our proposed sampling strategy, random sampling does not achieve as high performance metrics because it does not account for correlation and may be sampling repeated measurements with strong dependencies, and therefore is not as robust as our method.

This study used a linear SVM for the ARDS model. In preliminary work not shown, we found that an SVM with a non-linear kernel (RBF) had less consistent results. Although the SVM with RBF kernel generally outperformed linear SVM on training dataset, it had inferior performance (accuracy and AUROC) on the hold-out set. Even with 5-fold cross-validation and grid-search hyper-parameter optimization (of C and gamma), we found the performance of the SVM with RBF kernel to be lower on the test set, and standard deviation of the results (after multiple random train-test splits) to be 2-3 fold larger than the linear SVM. We speculate that overfitting possibly occurred because of lower sample size and the number of variables used as features for machine learning. Because linear SVM was more robust, we chose to focus on using label uncertainty in the modeling process using only linear SVM.

With more clinical data, it would be worthwhile to investigate whether incorporating both label uncertainty and a non-linear SVM model would lead to improved model performance. The electronic health record may contain additional data that could be added to our model. Evaluating the performance of the training approach that considers label uncertainty in a higher dimensional space would be of value; however, to limit the possibility of overfitting with our current small dataset size, we have focused on using features that are routinely used for clinical evaluation of ARDS in the current study.

We believe this project makes a significant contribution towards solving traditional classification problems in the context of biomedical and clinical applications. In medicine, there is almost always a degree of uncertainty when assigning a patient to a medical diagnosis. Yet, that diagnosis label may then be used as the classification label or predictive outcome during a machine learning task. Typically, the diagnostic uncertainty associated with the label is not considered during model building. We show how an expert clinicians' confidence in a diagnosis label can be used as vital information in the model training process. Exploiting the known diagnostic uncertainty within a medical domain is a generalizable approach that could be used in many medical applications. For example, sepsis is a clinical condition where early recognition is import for optimal patient care. However, diagnostic uncertainty is common [58], limiting ability to develop robust algorithms for sepsis detection. Incorporating label uncertainty when training an algorithm for sepsis detection may improve algorithm performance in a manner similar to ARDS.

It would also likely be of value to further develop approaches to incorporate label uncertainty into other machine learning frameworks besides SVM, such as random forest and neural networks. Since uncertainty in medical diagnosis occurs so commonly in clinical practice, accounting for label uncertainty with these learning algorithms may be highly applicable in other healthcare applications.

CHAPTER III

Robust Yet Reliable Methods for Analysis of Complex CXRs

3.1 Introduction

Imaging is integral to the care of ARDS patients that are critically ill. As previously mentioned in §1.2.1, CXR features of ARDS usually develop 12-24 hours after initial lung insult. Although appearances in manifestation can vary depending upon the stage of the disease, CXRs of patients typically exhibit characteristics of diffuse bilateral opacities with dense consolidation. Because of this, CXRs are a critical resource that can support an early diagnosis and provide evidence-based management strategies to patients with ARDS to improve their outcomes [56]. Identification of pulmonary opacification is a requirement for diagnosis of ARDS; however, radiological features by themselves are nonspecific and may not be correlated with clinical findings [4, 59]. As a result, inconsistencies in interpretability of chest imaging and poor inter-rater reliability suggest that patients with ARDS are not recognized when they develop this illness. Consequently, they do not receive the therapies proven to reduce mortality [60, 51].

At the time that a CXR is made available, radiology interpretation plays the major role in diagnosis of ARDS. However, due to the noisy nature of these images, the expertise of the interpreter, and other factors, the reliability of chest radiograph interpretation for ARDS is low among clinicians. There is a clear need for image processing algorithms to quantitatively express and extract the changes in the CXR (compared to normal cases) and evaluate the presence and severity of ARDS.

In this chapter, I present the development of an image processing technique for lung segmentation of complex CXRs from hospitalized patients. A novel feature extraction method is then proposed for capturing the notion of diffuse alveolar injury as a mathematical concept. Finally, I integrate both of these algorithms and use the extracted features to train multiple machine learning models to detect ARDS.

3.2 Challenges

Although many methods for lung segmentation exist in the literature [61, 62, 63, 64, 65, 66, 67], they are primarily designed and evaluated on high quality, standardized chest radiographs from controlled studies or outpatient settings that may not be representative of more complex CXRs from hospitalized patients. This is problematic for many patient populations, especially the critically ill, whose CXRs tend to have characteristics of varying image quality (e.g., dynamic range, sharpness), presence of introduced medical devices [68], diverse body habitus [69], and manifestation of disease [70]. As a result, these methods may not generalize and perform as well on CXRs obtained from other clinical settings.

To address this challenge, we collected data for algorithm development in a "stratified by severity" approach represent greater variety in patient population and heterogeneity of disease. The cohort selection criteria were intentionally designed as such, rather than an investigation of ARDS vs. healthy patients, to create a realistic representation of the patient population and clinical settings where these algorithms would be used. Because these data were acquired from hospitalized settings, the CXRs tend to be more complex than a standard chest imaging obtained from controlled stud-

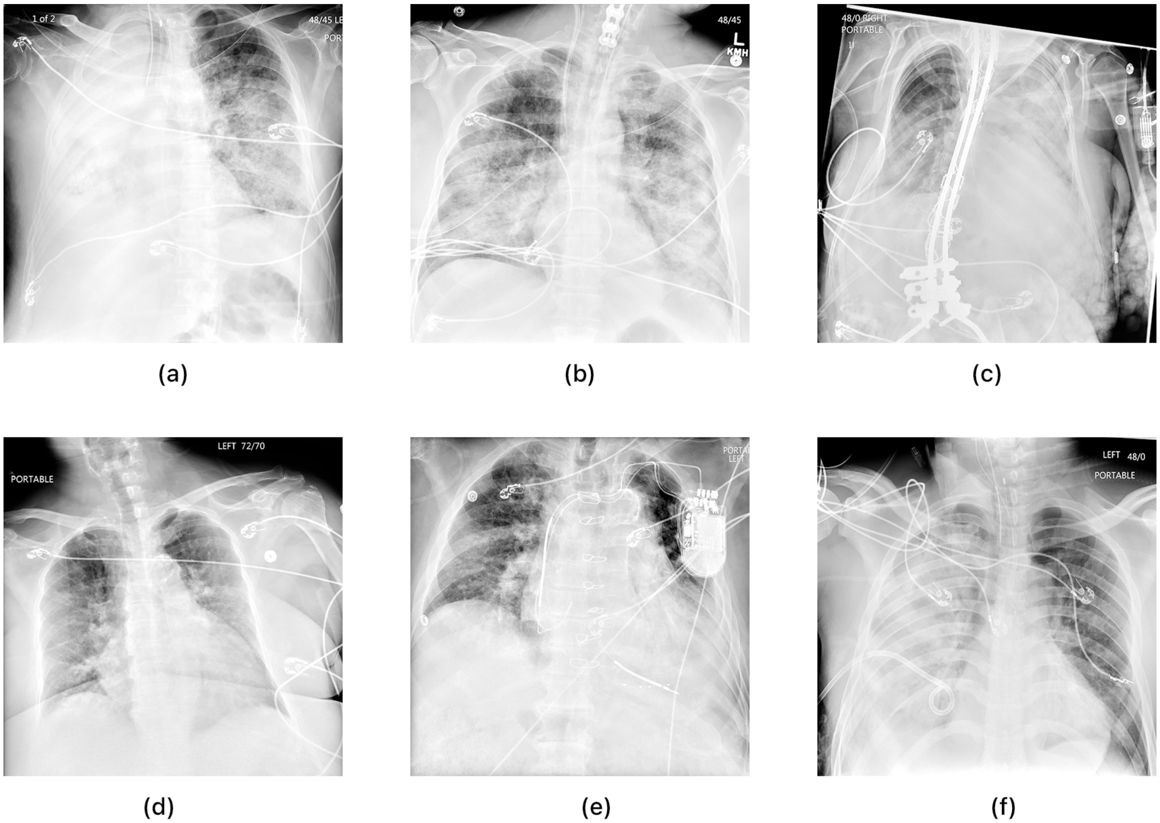


Figure 3.1: Examples of chest x-rays from the Michigan Medicine dataset, annotated as consistent with ARDS or inconsistent with ARDS based on the reviews of multiple clinical experts. Chest x-rays **(a)**, **(b)**, **(c)** demonstrate the findings of ARDS, which are bilateral airspace disease not explained by effusions, lobar/lung collapse, or nodules. These findings may **(b)** or may not **(a, c)** be uniform across both lung fields. Chest x-rays **(d)**, **(e)**, **(f)** do not demonstrate clear findings of ARDS, either because the lung fields lack clear airspace disease **(d)** or the disease that is present is unilateral **(e, f)**.

ies or outpatient settings. Characteristics of these complexities in the CXRs include varying quality (e.g., dynamic range and sharpness), presence of introduced medical devices, diverse body habitus, and manifestation of disease. Additional details about the data used is provided in §3.4.1. Examples of CXRs in this dataset are depicted in Figure 3.1.

3.3 Lung Segmentation of Complex Chest X-Rays

Lung segmentation, the process of accurately identifying regions and boundaries of the lung field from surrounding thoracic tissue, is an essential first step in pulmonary image analysis of many clinical decision support systems. Correct identification of lung fields enables further computational analysis of these anatomical regions [71], such as extraction of clinically relevant features to train a machine learning algorithm for detection of disease and abnormalities. These computational methodologies can assist physicians with making a timely, accurate medical diagnosis to improve quality of care and outcome for patients.

We therefore hypothesize that it may be possible to use image processing techniques to handle heterogeneous characteristics of CXRs to facilitate better, more generalizable lung segmentation. The aim of this study was to develop such an algorithm capable of robust and reliable performance on multiple patient populations, including critically ill patients. Our proposed hierarchical method first uses total variation denoising to remove irrelevant details and artifacts from medical equipment obscuring the lung fields. The image is then binarized with recursive thresholding to identify the left and right lung fields. Finally, a stacked active contour model is used to refine the final shape of the segmentation mask. The proposed method also incorporates systematic quality checks by using various assessment criteria at each step to ensure consistent, successful segmentation. It is especially important that these clinical decision support systems are highly reliable to ensure healthcare providers that the algorithm will consistently perform as expected, even in the most rigorous tasks. A deep learning approaching using U-Net, a convolutional neural network (CNN) architecture designed for biomedical image segmentation [72], was included as a "state-of-the-art" benchmark. A widely-used algorithm based on random walks [73] and another established shape-based "active spline" model [74] were also included for comparison with conventional image processing methods. These algorithms were selected on the basis of having excellent performance results on publicly available databases, being widely cited, and having an available codebase.

3.3.1 Total Variation-based Active Contour

The proposed algorithm, Total Variation-based Active Contour (TVAC), is comprised of three primary steps:

1. Total variation denoising was employed to delineate and remove various medical equipment visually obscuring the lung fields (§3.3.1.1).

2. Recursive binarization was used to systematically identify the lungs (§3.3.1.2).

3. A stacked active contour model was utilized to improve lung boundary formation (§3.3.1.3).

Prior to execution, chest radiographs are first normalized with contrast-limited adaptive histogram equalization (CLAHE) [75] to adjust contrast locally while limiting the amplification of noise to ensure that CXRs in the dataset are generally represented within the same pixel intensity range. A schematic diagram of TVAC is provided in Figure 3.2.

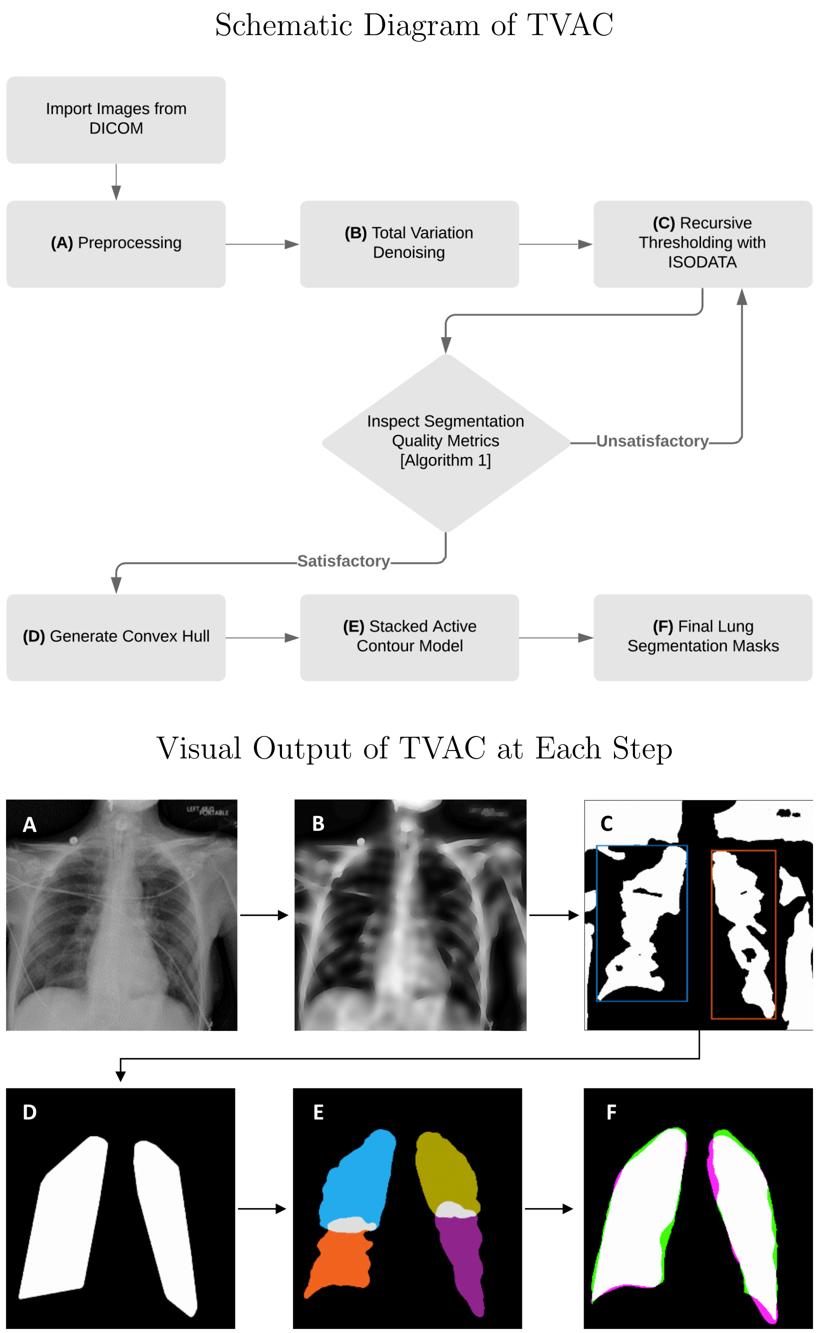

Schematic Diagram of TVAC
Import Images from DICOM
(A) Preprocessing
(B) Total Variation Denoising
(C) Recursive Thresholding with ISODATA
Inspect Segmentation Quality Metrics [Algorithm 1]
Unsatisfactory
Satisfactory
(D) Generate Convex Hull
(E) Stacked Active Contour Model
(F) Final Lung Segmentation Masks
Visual Output of TVAC at Each Step
A
B
C
D
E
F

Figure 3.2: Outline of the proposed TVAC method. **(a)** An example image containing a few wires from a patient diagnosed with acute hypoxic respiratory failure is shown. The image is normalized with CLAHE at this step. **(b)** Total variation denoising is used to diffuse wires while preserving edges of the lungs. **(c)** The denoised image is binarized with recursive thresholding and initial lung segments are extracted. **(d)** Convex hulls are generated from the extracted lung regions to enclose the lung fields and capture regions lost during binarization. **(e)** Lungs are partitioned into quadrants, each is individually processed with the stacked active contour model. **(f)** Final output of the lung segmentation algorithm. Green represents the ground truth, magenta shows the algorithm's segmentation output, and white illustrates overlap of the two – indicating regions that are correctly segmented.

3.3.1.1 Total Variation Denoising

Total Variation Denoising is a method to remove noise from images using a model of Rudin, Osher and Fatemi (ROF) [76]. If $f : \Omega \to \mathbb{R}$ is a a grayscale image, where Ω is a rectangle in $\mathbb{R}^2$, then the total variation of f is:

$$\|f\|_{\mathrm{TV}} = \int_{\Omega} |\nabla f| \tag{3.1}$$

To denoise an image f, we find an approximation u of f for which $\|u\|_{\mathrm{TV}}$ is small by minimizing:

$$\lambda \|u\|_{\mathrm{TV}} + \frac{1}{2} \int_{\Omega} (f - u)^2 \tag{3.2}$$

Here λ is a regularization parameter. The level sets of an optimal solution u have a small perimeter (relative to their area) (see for example [77, §2.2.2]). This means that boundaries of the level sets tend to be smooth and round. ROF denoising removes local details in images, while maintaining and smoothing the boundaries of larger areas.

There are many algorithms for solving the optimization problem in the ROF model. We use an algorithm and implementation of Zhu and Chan [78] that uses the Primal-Dual Hybrid Gradient method (PDHG).

ROF denoising is used to remove irrelevant details and artifacts (e.g., electrocardiographic leads and prosthetic devices) from chest radiographs. Unlike blurring, total variation denoising preserves sharp edges such as the boundary of the lungs. The processed image retains most of the structurally large, well-defined regions of the original image while removing unwanted objects of fine scale and discontinuous variations. This workflow is illustrated in Figure 3.2a and 3.2b.

3.3.1.2 Binarization with Recursive Thresholding and Lung Field Identification

After denoising, the lung fields are localized through binarization of the image with a recursive threshold. Binarization assumes that an image contains two classes of pixels following a bi-modal distribution, where the foreground (region of interest) and background pixels can be distinguished by finding an optimal threshold separating the two classes. To determine this optimal threshold for global binarization, θ_k, the Iterative Self-Organizing Data Analysis Technique (ISODATA) [79] is used.

First, the histogram is initially segmented into two parts using a starting threshold (θ_0) at half the maximum dynamic range. The mean of the values associated with the foreground pixels (μ_{f,θ_0}) and background pixels (μ_{b,θ_0}) is calculated. An updated threshold value θ_1 is calculated as the average of these two sample means. This method is repeated until the updated threshold value does not change anymore. This process is formalized as:

$$\theta_k = \frac{\mu_{f,\theta_k - 1} + \mu_{b,\theta_k - 1}}{2} \text{ until } \theta_k = \theta_{k-1} \tag{3.3}$$

The denoised image is then binarized with threshold θ_k.

After binarization, morphological area opening is performed to remove small objects corresponding to artifacts from binarization. To extract the left lung, an object whose centroid is nearest to the upper right half of the image is selected; to extract the right lung, another object whose centroid is nearest to the upper left half of the image is selected. The binary masks (regions containing the object of interest) extracted from this process are shown in Figure 3.2c. Both objects are assessed for quality of segmentation and similarity comparison, summarized in Algorithm 2, to ensure that they accurately correspond to the two lung fields.

Algorithm 2: Pseudocode for Binarization with Recursive Thresholding and Lung Field Identification

```
Input : Denoised Chest X-Ray
calculate global threshold ($\theta_0$) with ISODATA
repeat
    perform global binarization denoised image at threshold $\theta_0$
    exclude artifacts > imageArea/4 and artifacts < imageArea/100
    identify leftLung as the connected component with minimal Euclidean distance
    from its centroid to the upper left half of the image; check for individual
    assessment:
        check Eccentricity > $\alpha$
        check Equivalent Diameter > $\beta$
        check Filled Area > $\gamma$ of total image
        check Filled Area < $\delta$ of total image
        check ROI is adjacent to image borders
    identify rightLung as the connected component with minimal Euclidean distance
    from its centroid to the upper right half of the image; check for individual
    assessment:
        check Eccentricity > $\alpha$
        check Equivalent Diameter > $\beta$
        check Filled Area > $\gamma$ of total image
        check Filled Area < $\delta$ of total image
        check ROI is adjacent to image borders
    merge the two lung masks; check for similarity assessment:
        check Major Axis Length > 1.5x of each other
        check Convex Area > $\frac{1}{3}$ of total image
    if $\geq$ 2 failure from individual assessment and $\geq$ 1 failure from similarity
    assessment then
        $\theta_0 = \theta_0 * 0.95$
until quality of segmentation is satisfactory;
generate convex hulls from leftLung and rightLung
Output: convex hulls as binary masks for each lung
```

The parameters utilized in Algorithm 2 can be varied as needed to implement this method for similar applications. The specific values used for this experiment are provided in Table 3.1. These values generated reasonable results and demonstrated robustness to variations in analysis. In particular, we increased and decreased these values by 10% and observed that these changes did not have a significant impact on the results. If the masks violate more than 1 of these criteria, threshold θ_k is reduced by 5% and binarization is repeated. This process of recursively reducing threshold θ_k

Table 3.1: Parameters for Algorithm 1.

α	0.98
β	135
γ	1/3
δ	1/100

Although a wide range of parameters were tested, these are the specific values used in this experiment. These values generated reasonable results and demonstrated robustness to variation in analysis - even when the values were increased or decreased by 10%.

is repeated until all but one quality criteria are satisfied.

Convex hulls are then generated from both masks to enclose the lung fields. This geometric representation of the lung fields, shown in Figure 3.2d, is the smallest convex polygon shaped by vertices of the previous mask and is designed to capture interior regions that weren't included during binarization.

3.3.1.3 Stacked Active Contour Model

Following denoising and lung field segmentation the two convex hulls are then further refined to better capture the shape of the lungs. A standard active contour model (ACM) [80] is able to use these templates as a deformable spline, allowing the convex hulls to “stretch” and better fit to the pleural lining of the lung. However, we found that using the lung field as the template yielded unfavorable results and incomplete segmentation, particularly with respect to the costophrenic recess and in peripheral regions. To overcome this obstacle, we developed a stacked active contour model where the lung quadrants, rather than the whole lung field, are used as templates to better capture these peripheral regions such as the apex and costophrenic recess. Standard ACM uses a pre-defined number of consecutive iterations to expand or contract based on minimization of energy and other constraint forces. The proposed ACM model sequentially “stacks” 50 iterations of parameterized contour expansions, followed by 50 iterations of parameterized contractions. This process is repeated 20 times, resulting in a total of 1000 iterations.

The two masks from the upper and lower quadrants of each side are then combined to reconstruct the lung fields. This final step is shown in Figure 3.3c and 3.3d for reconstruction of the right lung field. A smoothing filter is applied to remove jagged edges on the mask boundary.

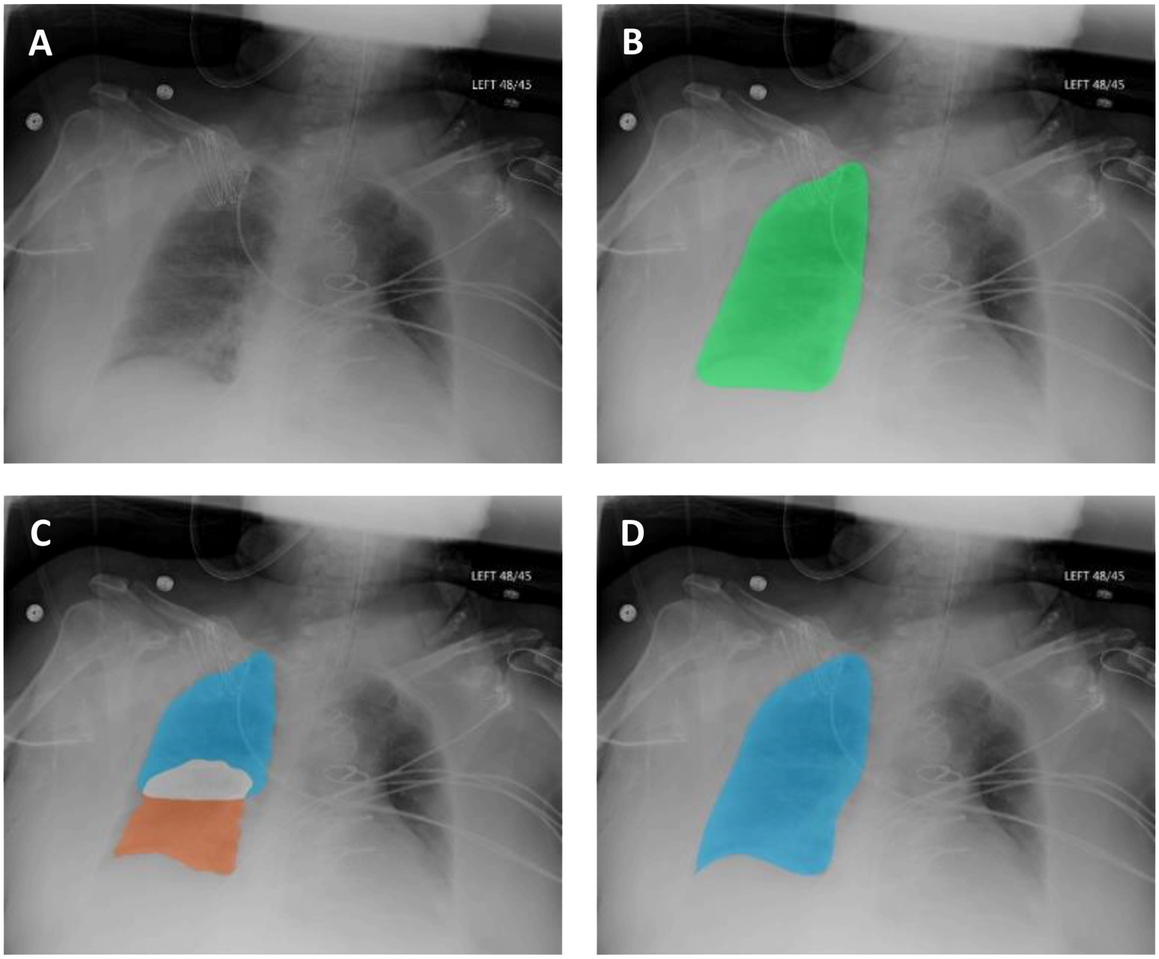


Figure 3.3: Segmentation with the stacked active contour model. **(a)** An example source image is shown for reference. **(b)** When the final segmentation mask is processed with a standard active contour model, areas of incorrect segmentation can be systematically observed – most commonly, at the right lung's costophrenic recess and regions adjacent to the diaphragm. **(c)** Quadrant-based processing with a stacked active contour model shows better deformation and contouring to peripheral boundaries. **(d)** Final output for segmentation of the right lung after combining the upper and lower quadrants and applying a smoothing filter.

3.3.2 Convolutional Neural Networks (U-Net)

U-Net is a convolutional neural network that was developed for biomedical image segmentation [72]. For this study, we've trained the U-Net to perform lung segmentation from CXRs. The network is based on the fully convolutional network and its architecture was modified and extended to work with fewer training images and to yield more precise segmentations. The U-Net CNN was implemented with Keras [81] using the TensorFlow backend and trained on both the JSRT and Montgomery datasets with 5-fold cross-validation. To further extend this analysis, additional experiments were conducted with the U-Net trained on JSRT, Montgomery, and 50% of Michigan Medicine data (including adult ARDS, adult severe ARDS, and pediatric ARDS) to "fine-tune" the model so that it encounters an even greater variety in patient population and heterogeneity of disease in the target dataset. Additional details for these methods are published in previous works [72].

3.3.3 Random Walker

We used a modified implementation of the random walker algorithm designed for unsupervised lung segmentation. This version relies on extracting horizontal intensity profiles to intuitively match a pre-designed template to identify anatomical regions of the CXR and accordingly place seed points for segmentation. First, the algorithm proceeds by extracting 18 intensity profiles running horizontally, each of them equally spacing apart, and in each intensity profile three extreme points denoting the two lungs and the esophagus are determined through profile matching. The algorithm removes profiles that do not intersect with the lung, and the rest of the extreme points are plugged into random walker algorithm to perform segmentation. Additional details for these methods are published in previous works [73].

3.3.4 Active Spline Model

The active spline model used in this study is a combined point distribution model and centripetal-parameterized Catmull-Rom spline for lung segmentation. This "template matching" method uses a fixed set of points resembling a generalized shape of the lungs and adapts this template to capture the lung fields from CXRs. After the lung segmentation boundaries are generated, it can be easily edited to allow for users to interact and refine the segmentation masks. Additional details for these methods are published in previous works [74].

3.4 Application: Lung Segmentation from Complex CXRs

We evaluate the proposed TVAC method for lung segmentation on multiple datasets, including two publicly available CXR repositories and data from Michigan Medicine comprising of critically ill patients with respiratory failure. Furthermore, we compare the proposed algorithm's performance to multiple state-of-the-art lung segmentation methods, including a deep learning approach (§3.3.2), standard computer vision algorithms (§3.3.3), and conventional image processing techniques (§3.3.4).

We used the Sørensen–Dice coefficient, a statistical validation method based on spatial overlap to measure the degree of similarity between the algorithm's segmentation and ground truth reference as annotated by multiple clinicians [82, 83]. Given two sets X and Y representing the segmentation output and ground truth, respectively, the Dice coefficient is defined as:

$$Dice(A, B) = \frac{2TP}{2TP + FP + FN} \tag{3.4}$$

For this study, a Dice coefficient under 0.70 is recognized as failed lung segmentation. This value is determined, through our experience from similar studies, as the lowest acceptable level of segmentation correctness for effective feature extraction and sufficient for machine learning.

In addition to reporting summary statistics, we also present our experimental results with a violin plot generated by a kernel density estimate of all the results [84]. These plots are essentially mirrored density plots and enables a comparison of algorithm performance, in terms of Dice coefficient, across patient populations. The thicker part of a violin plot indicates higher frequency, and the thinner part implies lower frequency. Violin plots with "longer tails" represent algorithms that more often failed to accurately segment a patient's lungs within a population.

3.4.1 Data

The Institutional Review Board approved this study with a waiver of informed consent. We retrospectively identified three cohorts of patients hospitalized in adult and pediatric intensive care units at Michigan Medicine in 2016 and 2017. The first cohort was a random sample of 100 adult patients (mean age 58 years ± 16 [standard deviation], 48% female) with acute hypoxic respiratory failure (PaO_2/FiO_2 ratio of $<$ 300 mm Hg while receiving invasive mechanical ventilation), stratified such that 50 of the patients met the criteria for the Acute Respiratory Distress Syndrome (ARDS) after review by clinical experts. The second cohort included chest radiographs from 25 additional adult patients (mean age 55 years ± 17 [standard deviation], 44% female) with "high confidence ARDS" by multiple physicians [51]. Chest radiographs from this cohort would be expected to have intense, widespread bilateral opacities that would be more difficult for segmentation algorithms. The third cohort included 100 chest x-rays from pediatric patients (mean age 7 years ± 5 [standard deviation], 39% female) hospitalized in the Pediatric Intensive Care Unit. Children age 14 days to 19 years with an endotracheal tube on mechanical ventilation were eligible for inclusion; this cohort was stratified such that 50 of the patients met criteria for pediatric ARDS. Additional details of these patient groups are provided in Table 3.2.

Table 3.2: Patient demographic of Michigan Medicine cohort.

	Adult Cohort				Adult Severe ARDS Cohort				Pediatric Cohort			
	n	Age	Non-ARDS	ARDS	n	Age	Non-ARDS	ARDS	n	Age	Non-ARDS	ARDS
Total	100	58 ± 16	50	50	25	55 ± 17	0	25	100	7 ± 5	50	50
Male	52	60 ± 16	30	22	14	56 ± 16	0	14	61	9 ± 5	31	30
Female	48	64 ± 16	20	28	11	53 ± 19	0	11	39	6 ± 6	19	20

A total of 225 anterior-posterior chest radiographs were exported from Michigan Medicine's picture archiving and communication system then stored in the Digital Imaging and Communications in Medicine format prior to analysis. Annotations for ground truth of the lung regions on the two adult patient groups were performed by a pulmonary critical care physician with 4 years of clinical experience. Annotations for the pediatric cohort were performed by a pediatric critical care intensivist with 5 years of clinical experience.

CXRs from two publicly available datasets were also used to validate the algorithm in other patient populations. The Japanese Society of Radiological Technology (JSRT) [85, 86] is comprised of 247 posterior-anterior CXRs: 154 containing a pulmonary lung nodule and the remaining 93 without any nodules. The second external dataset from Montgomery County, made available by the U.S. National Library of Medicine [87], contains 138 posterior-anterior CXRs: 58 are cases with manifestations of tuberculosis and the remaining 80 are representative of normal, healthy lungs. A summary of these patient groups is provided in Table 3.3.

Table 3.3: Patient demographic of JSRT and Montgomery datasets.

	JSRT			Montgomery		
	n	Normal	Abnormal	n	Normal	Abnormal
Total	247	93	154	138	80	58
Male	119	n/a	n/a	64	n/a	n/a
Female	128	n/a	n/a	74	n/a	n/a

Individual patient age and gender information were not available for these two databases. In the JSRT dataset, "abnormal' refers to the presence of lung nodules. In the Montgomery dataset, "abnormal" refers to the manifestation of tuberculosis.

3.4.2 Results

Summary statistics of lung segmentation performance (mean, min, and standard deviation of Dice coefficient) on the entire Michigan Medicine dataset, stratified by different patient cohorts, from all four algorithms are reported in Table 3.4. The results in Table 3.5 provide summary statistics from 50% of the Michigan Medicine (held-out) dataset, when the other 50% is used for "fine-tuning" the U-Net algorithm. Violin plots are also provided in Figure 3.5 to better visualize the distribution and density of the reported results.

Table 3.4: Lung segmentation accuracy for the Michigan Medicine dataset.

	Adult (n = 100)			Adult Severe ARDS (n = 25)			Pediatric (n = 100)		
	Dice (mean)	Dice (min)	Standard Deviation	Dice (mean)	Dice (min)	Standard Deviation	Dice (mean)	Dice (min)	Standard Deviation
TVAC	86.61	76.16	03.92	84.16	76.14	04.09	85.07	75.26	03.75
U-Net	88.85	00.01	11.77	84.82	43.32	11.81	87.32	56.45	08.53
Random Walker	74.46	14.97	18.82	64.63	15.96	16.19	67.31	18.92	17.78
Active Spline	64.01	20.03	16.70	62.29	15.83	15.83	61.37	14.91	17.60

Data are mean with minimum and standard deviation reported for each algorithm on different patient populations. TVAC = Total Variation-based Active Contour, Dice = Sørensen–Dice coefficient, ARDS = acute respiratory distress syndrome.

On all critically ill patient cohorts, TVAC and U-Net outperformed the random walker and active spline model. Although the TVAC model and U-Net show comparable mean Dice coefficients, the TVAC algorithm maintained more consistency in standard deviation and reliable performance (higher lowest Dice coefficient) across all 3 patient groups.

The TVAC algorithm was able to successfully segment lungs from CXRs of all critically ill patient cohorts; the lowest Dice coefficient reported was 0.75 from the pediatric cohort. Without fine-tuning, the U-Net has a total of 12 lung segmentation failures from the entire Michigan Medicine test set: the algorithm was unable to segment 4% of the adult cohort, 8% of the adult severe ARDS cohort, and 6% of the pediatric cohort. With fine-tuning and exposure to a subset of Michigan Medicine's data in its training set, the U-Net has a total of 12 lung segmentation failures from the 50% held-out test set: the algorithm was unable to segment 8% of the adult cohort, 15% of the adult severe ARDS cohort, and 12% of the pediatric cohort.

In comparison, the random walker algorithm was observed to have 83 unsuccessful lung segmentations, failing 26% of the adult cohort, 44% of the adult severe ARDS cohort, and 46% of the pediatric cohort. The most failures were observed from the active spline model, which reported a total of 130 failures from 55% of the adult cohort, 56% of the adult severe ARDS cohort, and 58% of the pediatric cohort.

Table 3.5: Lung segmentation accuracy with U-Net fine tuning.

	Adult (n = 50)			Adult Severe ARDS (n = 13)			Pediatric (n = 50)		
	Dice (mean)	Dice (min)	Standard Deviation	Dice (mean)	Dice (min)	Standard Deviation	Dice (mean)	Dice (min)	Standard Deviation
TVAC	86.08	77.90	04.39	83.01	76.14	04.23	84.67	78.27	03.64
U-Net	89.53	48.42	09.75	81.88	00.01	26.85	88.29	42.77	14.03
Random Walker	74.22	38.98	12.38	67.11	34.05	14.48	66.81	26.92	16.75
Active Spline	64.75	20.03	17.43	64.31	34.10	12.70	62.03	20.03	18.14

Data are mean with minimum and standard deviation reported for each algorithm on different patient populations. TVAC = Total Variation-based Active Contour, Dice = Sørensen–Dice coefficient, ARDS = acute respiratory distress syndrome.

In Figure 3.4, all four algorithms and their final lung segmentation from chest x-rays of critically ill patients in the Michigan Medicine dataset are shown. These examples were selected to present common pathological findings and characteristics of more complex chest x-rays from hospitalized patients. These visually qualitative results are presented to provide insight into the difficulty of this task and why these algorithms may fail.

Nearly all segmentation methods performed well on lung fields that were clearly defined, unobscured by medical equipment, and absent or with minimal manifestation of any pathological conditions. In the presence of abnormalities, such as pulmonary infiltrate in Figure 3.4a or lung opacities in Figure 3.4b, U-Net and the random walker algorithm both failed to produce acceptable results on these examples. Patients

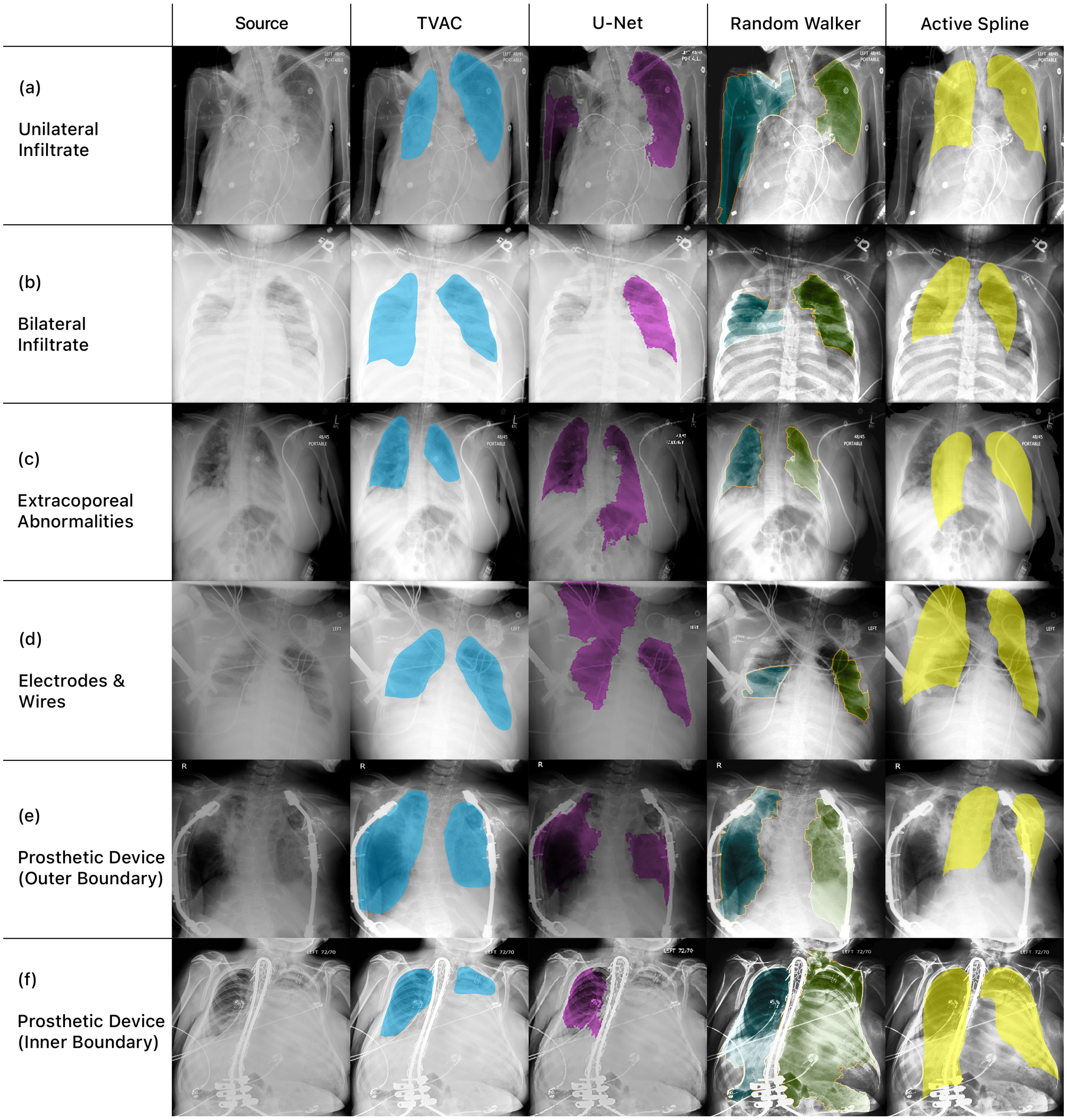


Figure 3.4: Lung segmentation from CXRs of patients at Michigan Medicine. This figure illustrates the qualitative difference among algorithms and focuses on how they fail in different clinical scenarios, including **(a)** manifestations of unilateral infiltrate **(b)** bilateral lung opacities **(c)** extracorporeal abnormality from an unrelated comorbidity in the abdomen **(d)** electrocardiographic leads overlying the lung fields **(e)** a prosthetic device obscuring the outer boundary of the lungs and **(f)** a prosthetic device interfering with the inner boundary of the lung.

suffering from traumatic injury may also present with multiple abnormalities from these comorbidities. An example of extracorporeal abnormality in the abdomen is shown in Figure 3.4c. These types of issues may be problematic for deep learning approaches, which are rigorously trained to identify a specific pattern representation and may struggle when present with an unexpected example outside of what the algorithm has been trained on.

The presence of medical equipment present throughout the chest x-ray is also problematic. Figure 3.4d shows an example with electrocardiographic leads and wires, which have visual characteristics comparable to the lung field boundaries (e.g., edges that are well-defined, bright, and elongated). In this example, U-Net recognizes the wires as an extension of the lung boundary and overextends the final segmentation mask of the right lung field into the patient's shoulder region. The random walker algorithm identifies the wire as lung boundary and produces two lung segmentation masks truncated at where the wires overlay the lung fields. Additional examples of obscuring medical equipment are shown in Figure 3.4e and 3.4f, we observe that both the random walker algorithm and active spline model fails for similar reasons as previously mentioned.

Table 3.6: Lung segmentation accuracy for the JSRT and Montgomery Datasets.

	JSRT ($n = 247$)			Montgomery ($n = 138$)		
	Dice (mean)	Dice (min)	Standard Deviation	Dice (mean)	Dice (min)	Standard Deviation
TVAC	95.01	84.88	02.97	95.69	85.66	02.51
U-Net	98.17	95.00	00.12	96.94	84.42	02.67
Random Walker	88.09	49.73	05.76	87.83	50.84	07.29
Active Spline	87.90	00.01	07.83	86.72	38.35	08.26

Data are mean with minimum and standard deviation reported for each algorithm on different patient populations. TVAC = Total Variation-based Active Contour, Dice = Sørensen–Dice coefficient, ARDS = acute respiratory distress syndrome.

Summary statistics of lung segmentation performance on both the JSRT and Montgomery datasets from our proposed algorithm (TVAC), the U-Net CNN, Ran-

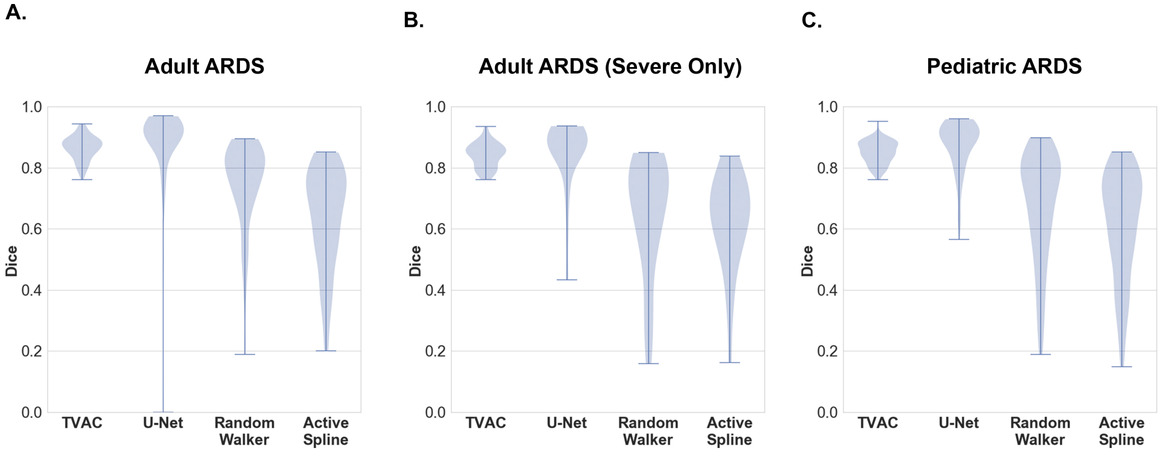


Figure 3.5: Violin plot of segmentation results. Multiple patient cohorts and datasets from Michigan Medicine were analyzed, including **(a)** the adult ARDS dataset, **(b)** the adult ARDS dataset comprising of only severe cases, and **(c)** pediatric ARDS dataset.

dom Walker, and Active Spline Model are reported in Table 3.6. On these two datasets containing standardized chest radiographs from previous studies, all four algorithms perform relatively well. The U-Net CNN reports the best performance (Dice: 0.98 ± 0.01 for JSRT, 0.97 ± 0.03 for Montgomery) of lung segmentation from these two datasets, followed by our proposed TVAC method (Dice: 0.95 ± 0.03 for JSRT, 0.96 ± 0.03 for Montgomery), the Random Walker algorithm (Dice: 0.88 ± 0.06 for JSRT, 0.88 ± 0.07 for Montgomery), and the Active Spline Model (Dice: 0.88 ± 0.08 for JSRT, 0.87 ± 0.08 for JSRT).

3.5 Feature Extraction

After developing a robust method for lung segmentation, the next step in detecting the presence of ARDS is to extract clinically meaningful features from the region of interest (i.e., the lung fields). To do so, it is critical to investigate the lung fields for pulmonary opacification – which manifests as a "cloud-like" appearance on radiographs. We propose a method, Directional Blur, which aims to capture this intuitive notion of diffuse alveolar injury and "cloudiness" as a mathematical concept. The basis of this approach is to strongly blur along areas of the image that exhibits directionality and also in regions where there are few details. Artifacts and peripheral structures in the CXR (e.g., ribs, vasculature, medical equipment) typically have features of directionality, while the detail-rich regions of the lungs and areas with opacification do not. This approach enables quantification of lung injury by capturing a diverse set of properties and measurements that may be suitable indicators for ARDS.

In addition to Directional Blur, this study also examines other features that have been used for similar applications in the detection of related lung diseases. For example, first-order statistics calculated from the histogram are also included in this work. These features have shown promising results in previously published works as a textural descriptor to differentiate a healthy lung field compared to a CXR present with lung injury [88]. Furthermore, we also extracted features from the gray-level co-occurrence matrix (GLCM). These features characterize the texture of an image by considering the spatial relationship and dependencies in the matrix. Although features from the GLCM have been used to train machine learning models for detection of various lung diseases (e.g., pneumonia and atelectasis), we did not find any research or studies applying GLCM features for detection of ARDS [89, 90, 91].

We also investigate the use of transfer learning with pretrained neural networks for extracting additional features that can be used to train machine learning algorithms to detect ARDS. Recent studies have indicated that information extracted from certain layers of convolutional neural networks can be very powerful features for use in classification tasks [92]. For example, neurons in the first layer learn features similar to Gabor filters while those from the last layer are more specific to the given learning task [93]. Initializing a network with transferred features from different layers can yield boosts to generalization even after fine-tuning to the target dataset [94]. Previously published works demonstrate this notion of transfer learning and document the success of using these features extracted from intermediate and higher layers of CNNs for recognition tasks that the network was not trained on [94, 95, 96, 97]. Furthermore, this approach of using deep learning models trained on large scale, non-medical data to extract features for general medical image recognition tasks via transfer learning has been demonstrated with favorable results by multiple research groups [98, 99]. Although several research groups have used pretrained networks to extract features from CXRs, we are not aware of any studies that evaluate the feasibility of using transfer learning in this capacity for detection of ARDS [89, 100].

3.5.1 Directional Blur

Lung infiltrates present with a "cloud-like" appearance on CXRs. We propose Directional Blur, a novel method to capture the intuitive notion of cloudiness as a mathematical concept. For this task, we first exclude normal findings within the CXR – such as ribs, vasculature, and medical equipment (e.g., tubes, cables, prosthetic devices). These artifacts typically have features of directionality, whereas the "clouds" corresponding to lung opacities are non- directional. Therefore, a cloudy lung region without artifacts can be described as follows: a) the average gray value will be above a certain threshold, b) the gray level varies within the region, and c) the gray level is non-directional.

Suppose that $T : [0, a] \times [0, b] \longrightarrow [0, 1]$ is a grayscale CXR. Consider a window $W = [u - \varepsilon, u + \varepsilon] \times [v - \varepsilon, v + \varepsilon]$ of size $2\varepsilon \times 2\varepsilon$ about a point $[u, v]$ in the lung region. Let T_x and T_y be the partial derivatives of T with respect to coordinates x and y. Define:

$$A_{xx} = \int\limits_W T_x^2 dx\ dy\ ,\ A_{xy} = \int\limits_W T_x T_y dx\ dy\ ,\ A_{yy} = \int\limits_W T_y^2 dx\ dy$$

then the function $G : [0, a] \times [0, b] \longrightarrow [0, 1]$ defined as $G(u, v) = \sqrt{A_{xx} + A_{yy}}$ measures the variation in a region. We normalize G such that it has values in [0,1].

The matrix A, defined as:

$$A = \begin{bmatrix} A_{xx} & A_{xy} \\ A_{xy} & A_{yy} \end{bmatrix},$$

is non-negative definite. Let $\lambda_1 < \lambda_2$ be the two eigenvalues of A, and define $H\left(u, v\right) = \frac{\lambda_1}{\lambda_2}$. Note that H has values between 0 and 1.

Our preliminary analysis and observation have shown that the product $G \cdot H$ may already be a suitable indicator for recognition of ARDS. The functions $T, G, H, and G \cdot H$ applied to CXRs for a patient diagnosed with ARDS and for a non-ARDS patient are shown in Figure 3.6a and 3.6b respectively. In both figures, the original CXR is shown in the upper left, G in the lower left, H in the upper right, and $G \cdot H$ in the lower right.

First-order statistics and additional measurements, further described in §3.5.2, are extracted from the product $G \cdot H$ as features to be used in training machine learning algorithms for detection of ARDS. In total, 72 features were extracted from Directional Blur.

3.5.2 Histogram

First-order statistics (mean, max, variance, kurtosis, and skewness) are calculated from the CXR histogram to capture textural properties of the lung fields. Previously published literature has demonstrated that such features extracted from CXRs exhibit significant differences between healthy and injured lung fields [88]. Specifically, higher variance and lower mean values in intensity have been observed in areas with pulmonary opacities when compared to normal, healthy lungs [101]. This evidence suggests that these first-order statistics correspond to the magnitude and the coarseness (or fineness) of the infiltrate [102].

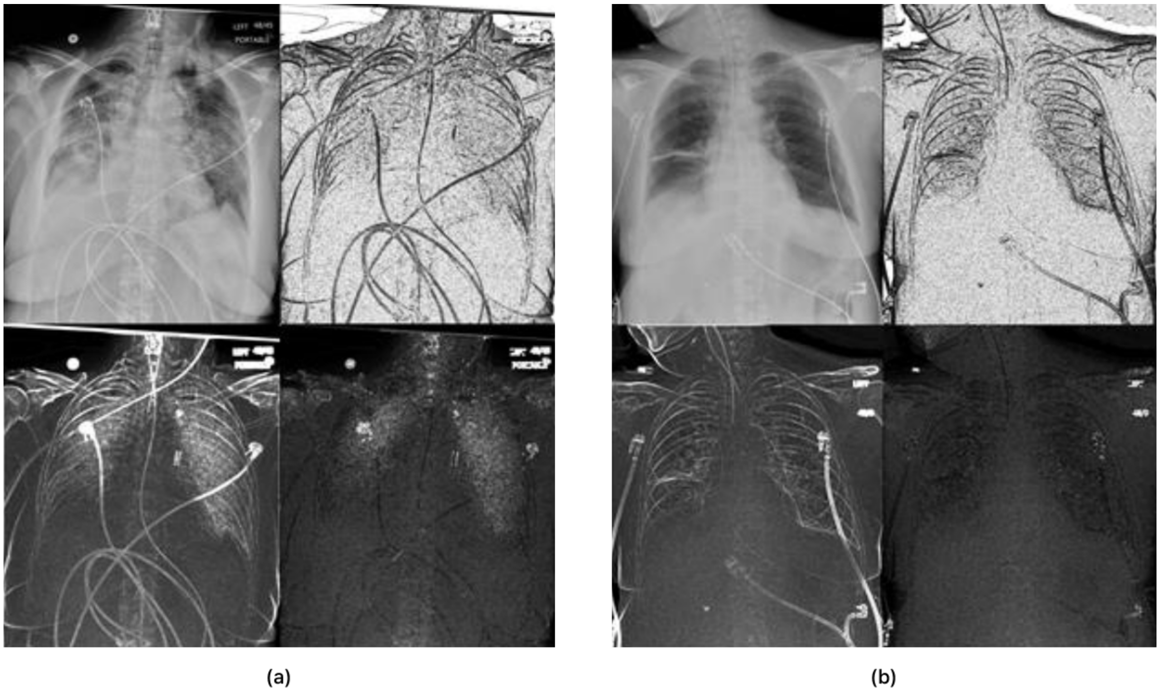


Figure 3.6: Directional blur applied to CXRs. Output is shown **(a)** from a patient diagnosed with ARDS and **(b)** from a non-ARDS diagnosis. The original CXR is shown in the upper left, G in the lower left, H in the upper right, and $G \cdot H$ in the lower right for both figures.

Additional measurements of the histogram are also used as features to capture the density of pulmonary infiltrates by examining local grayscale distribution. These features include standard deviation of the 5 largest local maxima (peaks), width of the largest peak at half-prominence, gray-level value at the first and second largest peaks, median of the maxima distribution and the frequency of that value, and area under the histogram. Features were separately extracted from the lung fields and lung quadrants. In total, 72 features were extracted from the histogram.

3.5.3 Gray-Level Co-Occurence Matrix

The gray-level co-occurrence matrix (GLCM) is a statistical method used to characterize the texture of an image with respect to spatial relationships at the pixel level [103]. The GLCM is defined as a two-dimensional matrix of joint probabilities between pairs of pixels of co-occurring values at specified offsets, which is used to compute second-order statistics [104, 105]. Multiple offsets and angles can be defined to increase the sensitivity of capturing pixel relationships of varying direction and distance [90]. Statistics extracted from the GLCM have demonstrated promising results when used as features to train machine learning models for detection of various lung diseases [89, 91, 106]. The second-order statistics extracted from the GLCMs in this study are contrast, correlation, energy, and homogeneity.

Contrast measures local variation present in an image and returns a measure of the intensity difference between a pixel and its neighbor over the entire image. For example, a high value of this feature may indicate the presence of edges, noise, etc. This property has been demonstrated to be higher in abnormal presentations within chest radiographs as compared to normal findings [107]. Contrast is defined as

$$\sum_{i,j} p_{i,j} |i-j|^2,$$

where i and j represents the x and y coordinates of the GLCM and $p_{i,j}$ is the element i, j of the GLCM.

Correlation measures the linear dependence (joint probability) of pixel pairs and can be interpreted as quantifying the consistency of image textures. A high correlation value indicates the predictability of pixel relationships. We expect that capturing these characteristics of a CXR may be useful as features for this study. Correlation is defined as

$$\sum_{i,j} \frac{p_{i,j}(i-\mu_i)(i-\mu_j)}{\sigma_i \sigma_j}$$

Energy, also referred to as the angular second moment, measures the uniformity of grayscale distribution of the image. Images with a smaller number of gray levels (e.g., when it is considered very uniform in representation) have larger values of energy. Therefore, we expect this measurement to be lower for abnormal findings and useful for distinguishing between a CXR with an ARDS and one from a non-ARDS diagnosis. Energy is defined as:

$$\sum_{i,j} (p_{i,j})^2$$

Homogeneity measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal and can be interpreted as a representation of the scale of local changes in image texture. High values of homogeneity indicate the absence of intra-regional changes and locally homogenous distribution of image textures. Homogeneity is defined as

$$\sum_{i,j} \frac{p_{i,j}}{1 + (i-j)^2}$$

Features were separately extracted from the lung fields and lung quadrants for each of the described GLCM properties. GLCMs are generated with multiple angles (0°, 45°, 90° and 135°) at a pixel distance of 1. A single, "invariant" spatial direction is generated by taking an average of the four directions so that the texture features will not be influenced by the angle of rotation. A total of 24 features were extracted from the GLCMs.

3.5.4 Deep Learning

A pretrained ResNet-50 deep learning model is used as to extract features from chest x-ray scans for detection of ARDS. ResNet-50 is a deep convolutional neural network consisting of 50 layers with skip connections to facilitate training deep networks, specifically for optimizing trainable parameters during backpropagation to mitigate the problem of vanishing gradients [108]. Residual networks such as ResNet50 are composed of multiple building blocks with shortcut connections that skip convolutional layers via identity mapping. Each block is composed of 3 convolutional layers that perform downsampling with a stride of 2, followed by batch normalization and rectified linear unit (ReLU) activation. The architecture of ResNet-50 ends with a global average pooling (GAP) layer and a 1,000 fully connected layer with softmax activation. The network was trained on over a million images from the ImageNet database [109] to learn rich feature representations and is capable of classifying into 1000 object categories.

Based on previously published work on transfer learning, we propagate CXRs (resized to 244 × 244) through the pretrained network. These layers can be reinterpreted as learned feature extraction layers [110] and activations from the GAP layer prior to the fully connected layer are extracted as feature vectors that can be used to train machine learning models to solve the classification task of this study. In total, 2048 features were extracted from the GAP layer of ResNet-50.

3.6 Application: Detection of ARDS from Complex CXRs

In this study, we evaluate these approaches - Directional Blur, first-order statistics from the histogram, GLCM, and pretrained deep neural networks - on CXRs obtained from Michigan Medicine and use the extracted features to train machine learning models for the detection of ARDS. Support vector machine (SVM), random forest, AdaBoost, and RUSBoost models were trained and evaluated with 5-fold cross-validation on these features from 2018 CXRs (data from 70% of patients); the final overall performance was then reported on a hold-out test set comprised of the remaining 1060 CXRs (data from 30% of patients). To better understand the strength and contribution of each technique, we present the results of classification when using each feature set separately and also combining all features when training the machine learning algorithms.

The extracted features were used to train multiple machine learning models for the detection of ARDS. A soft-margin support vector machine (SVM) with a linear kernel was used in this study and utilizes Bayesian optimization for tuning the C parameter (to adjust the penalty of misclassification). Random forest was implemented with 300 decision trees – we determined this was optimal n_{trees} since it provided adequate model complexity and further increasing n_{trees} did not show a significant difference in performance on the validation data. Gradient boosting has often been compared to random forest given the number of similarities between the two techniques. While random forest is known to be more robust to noise and easier to train, boosting techniques maintain a reputation of being more resistant to overfitting, with benchmark results having shown that booting produces better learners than random forests. We implement adaptive boosting (AdaBoost) and random under-sampling boosting (RUSBoost) in this work. Both techniques utilize Bayesian optimization to tune the learning rate and number of learning cycles. Data from 70% of patients (approximately 2018 CXRs) were used for model training and performance evaluated using 5-fold cross-validation ong this training set. The final reported results are from the remaining 30% of patients (1060 CXRs) that were used as the hold-out test set.

3.6.1 Data

This study was approved by the Institutional Review Board with a waiver of informed consent. The patient cohort consists of adult patients hospitalized in intensive care units at Michigan Medicine between 2016 and 2017. We retrospectively identified patients with moderate hypoxia (requiring more than 3 L of supplemental oxygen by nasal cannula for at least 2 hours) and acute hypoxic respiratory failure (PaO2/FiO2 ratio of $<$ 300 mm Hg while receiving invasive mechanical ventilation).

In total, 500 patients were included in this study and 3078 anterior-posterior chest x-rays were obtained. Of this population, 208 patients met the criteria for acute respiratory distress syndrome after being reviewed independently by multiple clinical experts. Labels for this dataset were generated using the same method described in §2.4. Data from 70% of patients (approximately 2018 CXRs) were used for model training and validation while the remaining 30% of patients (1060 CXRs) were used as the hold-out test set. There are 191 females (mean age of 58 years, 32% ARDS

Table 3.7: Data available from the Michigan Medicine ARDS CXR dataset.

	Patients	Chest X-Rays
Positive	151	909
Negative	349	2169
Total	500	3078

Table 3.8: Cohort demographic for the Michigan Medicine ARDS CXR dataset.

	n	Age	ARDS	Non-ARDS
Male	151	57.16 ± 16.72	89	220
Female	349	58.46 ± 15.71	62	129
Total	500	57.65 ± 16.32	151	349

positive) and 309 males in this study cohort (mean age of 57 years, 29% ARDS positive). The cohort demographics are summarized in Tables 3.7 and 3.8.

The chest x-rays in this study were reviewed independently by multiple clinical experts to generate the labels used in training the machine learning algorithms. ARDS is a life-threatening condition without a "gold standard" for diagnosis and the inter-rate reliability for correct diagnosis of the illness is only moderate [8]. Because of this, multiple experts were asked to determine whether each chest x-ray is consistent with ARDS and also to provide a confidence level in their diagnosis as high, moderate, slight, or equivocal. This information was converted to scale between 1-8 as illustrated in Figure 2.6. If the clinical experts' averaged review was below or equal to 4.5, a label of -1 (no ARDS) would be assigned to the chest x-ray. If the averaged review was above 4.5, a label of 1 (ARDS) would be assigned.

3.6.2 Results

CXRs are acquired in DICOM format and converted to an 8-bit grayscale image. Lung segmentation was performed with TVAC (as described in §3.3.1) to identify the region of interest from the CXR. Multiple masks were created for each CXR to represent the two lung fields (e.g., left lung and right lung) and four lung quadrants (e.g., upper-left lung and lower-left lung) to ensure that the extracted features meet one of the clinical criteria for diagnosis of ARDS (bilateral opacities on CXR).

Four distinct feature sets were used to train machine learning models for the detection of ARDS from chest x-ray scans: Directional Blur, first-order statistics from the histogram, information from a gray-level occurrence matrix, and deep learning features extracted with a pre-trained neural network. Based on relevant works in analytical morphomics, we perform a similar approach of normalization with structural physiology (using a ratio of chest width to sternum width) for selected features [111].

Performance metrics (accuracy, Area under the Curve (AUC), and F1 score) on the Michigan Medicine dataset for all four feature sets are reported in Table 3.9. The best performance achieved using an individual feature set was attained with an AdaBoost classifier trained on features derived from Directional Blur (0.78 accuracy and 0.74 AUC), followed by a RUSBoost classifier also trained on the same feature set (0.77 accuracy and 0.74 AUC). These results also indicate that training on multiple feature sets can yield further improvements. The best overall performance was achieved with AdaBoost trained on all available features from Directional Blur, first-order statistics in the histogram, GLCM, and deep learning (0.83 accuracy and 0.79 AUC). The second-best performing model when trained with all features is RUSBoost (0.81 accuracy and 0.77 AUC.) Results for additional permutations of combined features (e.g., Directional Blur and histogram features) are provided in Table 3.10 and in the supplementary materials (Table B.1, B.2, B.3).

Table 3.9: Performance metrics for detection of ARDS using features from Directional Blur, the histogram, GLCM, and deep learning.

	Directional Blur			Histogram			GLCM			Deep Learning (ResNet-50)			All Features Combined		
	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1
SVM	73.31	66.58	58.91	70.83	62.60	57.97	72.47	65.62	59.79	73.34	67.90	64.85	74.63	73.70	64.24
Random Forest	75.97	66.80	56.28	73.86	64.64	54.36	74.98	65.70	54.63	75.91	65.76	59.47	76.81	71.22	63.43
AdaBoost	**78.93**	**74.87**	**61.66**	**75.60**	**73.81**	**58.93**	**76.34**	**73.82**	**59.69**	**77.82**	**73.63**	**62.78**	**83.85**	**79.67**	**65.44**
RUSBoost	**77.68**	**74.78**	**65.90**	**74.62**	**71.55**	**64.62**	**77.70**	**72.12**	**66.38**	**77.88**	**72.46**	**63.93**	**81.03**	**77.68**	**67.29**
Robust Boost	73.98	69.92	55.62	70.48	66.41	55.76	71.88	68.61	57.73	75.69	70.72	56.93	76.79	73.44	63.87
Total Boost	70.68	68.39	55.82	69.67	64.50	54.79	70.36	67.70	55.38	73.79	70.93	53.80	73.50	68.28	59.74

A total of 2216 features were used to generate these results: 72 features from

Table 3.10: Additional performance metrics for classification results when using Directional Blur with additional features.

	Directional Blur + Histogram			Directional Blur + GLCM			Directional Blur + Deep Learning (ResNet-50)			All Features Combined		
	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1
SVM	73.53	68.97	59.70	74.36	69.80	62.87	75.90	70.61	64.75	74.28	73.09	64.75
Random Forest	75.87	66.63	59.74	76.90	68.32	59.64	76.72	67.64	62.80	76.78	71.36	63.91
AdaBoost	**79.88**	**75.70**	**62.46**	**81.80**	**77.46**	**63.73**	**80.98**	**77.35**	**65.46**	**83.18**	**79.61**	**65.80**
RUSBoost	**77.84**	**73.46**	**62.73**	**79.47**	**76.82**	**66.16**	**80.70**	**76.83**	**65.70**	**81.66**	**77.31**	**67.45**
Robust Boost	71.80	70.73	60.58	70.44	70.73	60.82	72.91	69.46	60.80	76.77	73.28	63.67
Total Boost	70.39	69.74	58.58	71.67	68.58	58.37	73.73	68.25	59.96	73.47	68.68	59.97

Directional Blur, 72 features from the histogram, 24 features from the GLCM, and 2048 features from deep learning. To reduce the dimensionality of utilized features, we implemented feature selection with PCA, minimum redundancy maximum relevance (mRMR), and chi-squared test. The first three principal components (98% of variance explained) were used for the PCA approach. The top 100 most important predictors were selected from feature selection with mRMR and chi-squared test. We did not test beyond using the top 100 ranked features since anything beyond that yielded an insubstantial predictor importance score. The results of this analysis are provided in Table 3.11.

Table 3.11: Comparison of methods used for feature selection.

	All Features			PCA (98% Variance Explained)			mRMR (100)			Chi-Squared Test (100)		
	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1
SVM	74.82	73.69	64.31	73.88	67.39	62.44	70.76	66.98	61.71	79.73	74.20	59.47
Random Forest	76.23	71.88	63.91	74.73	65.44	53.09	75.76	67.41	62.82	79.60	73.91	61.70
AdaBoost	**83.37**	**79.84**	**65.90**	**79.75**	**71.64**	**63.82**	**80.79**	**73.82**	**60.14**	**81.73**	**74.55**	**64.87**
RUSBoost	**81.79**	**77.80**	**67.57**	**76.64**	**68.71**	**58.23**	**76.92**	**72.52**	**62.57**	**80.96**	**76.80**	**65.43**
Robust Boost	76.58	73.72	63.83	72.41	67.64	58.20	76.45	68.41	58.34	75.89	70.54	60.82
Total Boost	73.72	68.63	59.47	72.28	66.61	55.87	76.55	67.42	56.03	75.84	70.42	60.33

3.7 Discussion

In this chapter, we developed a lung segmentation algorithm that would perform well on both publicly available datasets from retrospective research studies and on real-world data obtained from hospital and inpatient care, especially from critically ill patients. We demonstrate that our TVAC algorithm is capable of accurate and reliable lung segmentation from chest x-rays in the Michigan Medicine dataset comprising of hospitalized patients, of varying demographics and age groups, diagnosed with moderate hypoxia, acute hypoxic respiratory failure, or ARDS. Furthermore, we also evaluated TVAC on publicly available chest x-rays from the JSRT and Montgomery datasets to benchmark our proposed method with multiple state-of-the-art lung segmentation algorithms.

Many published algorithms and software platforms capable of lung segmentation exist [61, 62, 63, 64, 65, 66, 67]. However, nearly all of them have only been evaluated on chest radiographs where the lungs exhibit minimal or no pathological conditions [112]. Segmentation of normal, healthy lungs can be fairly straightforward, as the black pixels of the lung fields can be readily delineated from the white pixels of peripheral anatomic regions [113]. This task becomes challenging when segmenting lungs from chest x-rays of critically ill patients diagnosed with a lung disease or severe condition, such as ARDS, pneumonia, and pulmonary edema. These injuries tend to manifest with a white diffuse appearance [56, 114, 115] that may be incorrectly recognized by many algorithms as regions outside the lungs, as these attenuating characteristics are similar to the soft tissue of nearby anatomic structures. As a result, consolidation along the pleural margin of the lungs may generate an erroneous delineation and incorrect segmentation. These complications are also present in related applications and orthogonal studies (e.g., detection of consolidation) involving complex chest x-rays from hospitalized patients [70].

Furthermore, medical equipment such as wires, tubes, pacemakers, and various prosthetic devices can obscure lung fields on chest x-rays [68]. These objects are characteristic of CXRs obtained from hospitalized patients or during inpatient care, which may contain a diverse array of medical equipment used to monitor and treat patients [116]. These items appear as connected regions of high pixel intensity with strong edges, often interfering with edge detection of the lung's pleural space and resulting inaccurate boundaries. Because these objects don't typically appear in CXRs obtained from outpatient care or controlled studies, it is therefore essential to include these types of complex data from clinical and hospital settings in the evaluation set of any automatic lung segmentation algorithm. These physiological abnormalities and noise from medical devices can hinder segmentation methods using lung models that have been computed on healthy lungs only [117]. Because of this, we also sought to investigate the efficacy and reliability of these algorithms on our Michigan Medicine dataset.

Despite the high overall performance of the deep learning approach, our experimental results demonstrate that U-Net can be inconsistent and suffers from numerous lung segmentation failures. Based on the violin plots as well as results of segmentation on the JSRT and Montgomery datasets, we can infer that U-Net performs very well on the types of x-rays it has encountered before. However, when new, unseen examples of disease and noise are shown, the CNN is unable to generalize pattern recognition for these challenging lung fields. Even when the U-Net is fine-tuned with 50% of all available Michigan Medicine so that it encounters an even greater variety in patient population and heterogeneity of disease in the target dataset, the same segmentation issues still persist - namely, failure to recognize the lung boundaries due to interference from medical equipment or gross abnormalities present on the image. These results suggest that although the U-Net is very capable of excellent segmentation, robustness of the deep learning approach needs to be improved before it is practical for clinical use.

The “template matching” active spline model suffers from similar generalizability issues as U-Net. On chest x-rays with well-defined lung boundaries, the algorithm is capable of producing excellent masks. However, when lung fields and pleural regions are obscured by injury (e.g. collapsed lung), the template matching attempt usually fails [112, 118]. When developing TVAC, we also take into consideration the issues of deployability, usability, and trustworthiness from the perspective of a healthcare provider. Missing a few pixels is better than missing an entire lung field – especially if the algorithm is extended and applied to subsequent clinical tasks (e.g., using lung segmentation as a preprocessing step for prediction of acute respiratory distress syndrome, pneumonia, or sepsis). Making a clinical decision based on inaccurate information could be extremely dangerous for the patient’s outcome and we believe that healthcare providers would likely opt for a more consistent system in lieu of one with a slightly higher mean performance benchmark but less reliability.

We recognize that there are several limitations to this study. The cohort sample sizes were relatively small, which limited the extent of stratified analysis, such as looking at challenges in segmentation grouped by type of lung injury or in the presence of a specific treatment/medical device. Furthermore, due to the limited amount of data available from Michigan Medicine, we were not able to train the U-Net on this dataset. Therefore, the U-Net was trained on both the JSRT and Montgomery datasets combined (evaluated with 5-fold cross validation) and we thus relied on transfer learning for generalization of this CNN to the Michigan Medicine dataset. The use of significantly larger training databases of CXR with heterogeneous characteristics in future studies may improve the performance of the U-Net CNN. Another limitation to note is that ground truth annotations of the lung fields were provided by critical care physicians instead of radiologists. Although we don’t believe this has affected our study, we do acknowledge that many similar studies involving ground truth from radiographs typically relies on a radiologist, or the supervision of a radiologist, to correctly annotate the image.

To extract features for detection of ARDS from complex CXRs, we propose and describe Directional Blur, a novel feature engineering technique used to capture the "cloud-like" appearance of diffuse alveolar damage as a mathematical concept. This work also examines the effectiveness of using a pretrained deep neural network via transfer learning as a feature extractor in addition to standard features extracted from the histogram and GLCM.

Many published algorithms for the detection of various lung pathologies from chest radiology exists [88, 89, 90, 91, 119, 102]. However, we did not find any studies that particularly focused on acute respiratory distress syndrome. Some of these conditions share similarities in pathology and clinical presentation, but it was unknown whether the features used to detect a particular condition (e.g., sepsis) would also be effective for the detection of ARDS. Our results show that some of these features do in fact work, e.g., first-order statistics from the histogram and GLCM. Furthermore, we demonstrate that the proposed Directional Blur technique is capable of detecting ARDS and outperforms other techniques that have been used for similar applications. We report that the best overall performance is obtained when the machine learning models are trained with all four features sets combined rather than only having access to each individual feature separately.

We also conducted extensive tests with several feature selection methods, including principal component analysis (PCA), minimum reduction maximum relevance (mRMR), and chi-squared test. The outcome of those experiments demonstrated that better classification results were obtained when using all available features compared to only using the most important features selected by these techniques. These results were surprising at first, since we expected redundancy in the feature space, especially from the 2048 features from deep learning. However, after a more comprehensive analysis the data, we concluded that this is reasonable because many of the tree-based and boosting methods already include feature selection in their implementation. Therefore, models that don't already include this process (e.g., SVM) will benefit the most from feature selection – which is exactly what we observe in Table 3.11 with SVM when using all available compared to only using the top ranked features from the chi-squared test.

Although ResNet-50 was used to extract the deep learning features, a number of other pretrained deep neural networks were also considered – including ResNet-18, ResNet-101, Inception-v3, U-Net, and VGG19. These networks were primarily chosen based on their publication record and capability in using arbitrary layers for feature extraction. Preliminary results showed that features extracted with ResNet18, ResNet101, and Inception-v3 did not perform as well as ResNet50. As the architectures for U-Net and VGG19 do not contain a GAP layer features would be extracted from the max pooling or convolutional layers. The activations from VGG19's max pooling layer have a dimensionality of $7 \times 7 \times 512$, while the activations from U-Net's ReLU layer are of size $256 \times 256 \times 32$, resulting in almost 2 million features when flattened. We did try multiple tensor decomposition methods to work with this high-dimensional data, including higher-order singular value decomposition (HOSVD), but did not achieve satisfactory performance.

Intuitively, one could argue that the learned weights from the deeper layers should be more specific to the images of the training dataset and the task it was initially trained for. However, a number of publications have reported promising results with features derived from the GAP layer. Furthermore, our internal testing showed comparable results between extracting features from the GAP layer and a shallower layer. Ultimately, we decided to use ResNet-50's GAP layer – which yielded 2048 features.

We recognize that there are several limitations to this study. With more data, it would be worthwhile to investigate training an end-to-end deep learning model directly from CXRs, comparing the effectiveness of this approach to features extracted with a pretrained deep neural network via transfer learning. Another limitation to note is that our dataset is only labeled for binary classification of ARDS or non-ARDS, even though the patients in the non-ARDS cohort still exhibit a degree of respiratory failure. In future work, we would like to examine the feasibility of multi-label classification to further improve accuracy in diagnosis. We also plan to include additional data, such as signals from physiological waveforms and data from electronic health records, for the detection of ARDS.

CHAPTER IV

Learning Using Privileged Information

4.1 Introduction

The idea of using privileged information was first proposed by V. Vapnik and A. Vashist in, in which they tried to capture the essence of teacher-student based learning and knowledge transfer [120]. Standard machine learning paradigms consider the following scenario: given a set of training examples, find, in a given set of functions, the one that approximates the unknown decision rule in the best possible way. In such a paradigm, the teacher does not play an important role.

Inspired by the way human learning works, they created a new learning paradigm in which the learner is provided with not only a set of training examples, as described above, but also a set of additional "privileged information" that can help significantly improve the learning process. Learning using privileged information (LUPI) accelerates machine learning by more closely mimicking human teacher-student interactions. In human interactions, the teacher provides the student with additional information specific to each example, such as explanations. This allows the student to learn more information from each example, thus learning faster and more effectively [121]. LUPI considers the fact that while the privileged information are not available at the testing stage, the abundance of such information in the training phase can help tune-up and improve the choice of the solution $f_\gamma(x)$. This type of information is abundant in healthcare, where much more information about a patient's health status is available in retrospective databases compared to real-time environments.

In the LUPI architecture, at the training stage some additional information x_i^* about training example x_i are available, while this privileged information will not be available at the test stage. The LUPI paradigm can be formulated as follows: given a set of training examples plus privileged information $(x_i, x_i^* y_i), i = 1, \ldots, N$, in which $x_i \in X$ form the input attributes/features, $x_i^* \in X^*$ are the privileged information on the training sample i, and $y_i \in \{-1, 1\}$ is the output class, the learning task is to find a function $f_\gamma(x) : X \rightarrow \{-1, 1\}$ (where $\gamma \in \Lambda$) to learn/generalize a mapping between the input and the output considering the privileged information. Note that the additional privileged information comes from the space X^*, which is different from the space X and is not needed after the training phase, i.e. the learned function $f_\gamma(x)$ does not need the privileged information to form predictions on the testing data. A depiction of the LUPI paradigm is provided in Figure 4.1.

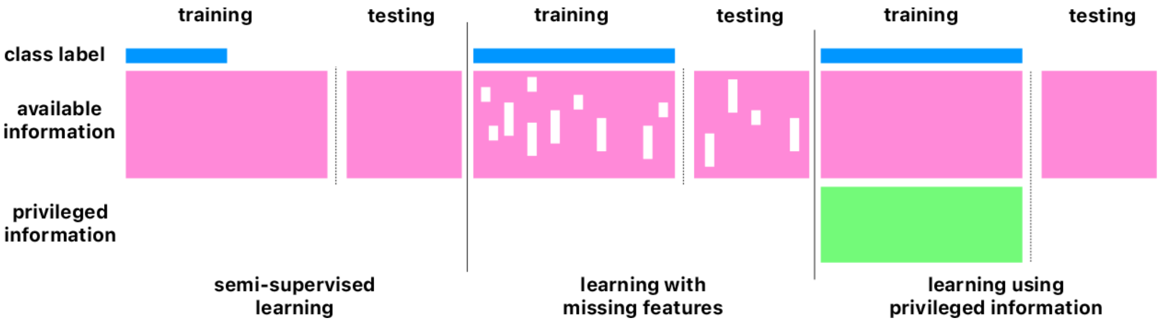


Figure 4.1: Depiction of the LUPI paradigm.

Specifically, in the training phase, a cost function defined over the training examples (that includes both the main input X and the privileged X^* information) is minimized to find the best set of parameters, γ, and therefore the solution $f_\gamma(x)$. In LUPI the cost function is often composed of two sets of terms, one representing in the error in mapping X to the output (as well as the complexity of the function) and another set of terms representing in the error in mapping X^* to the output (as well as the complexity of the function). Again, note that even though X^* is included in the optimization process in the training phase, the result of this optimization is $f_\gamma(x)$ which is defined only over X.

LUPI has proven successful in several applications. Sharmanska et al. found that learning using privileged information aided computer vision tasks [122]. Ribeiro et al. found that SVM+, a modification of SVM that leverages privileged information, improved bankruptcy prediction compared to regular SVM [123]. Liang et al. modified SVM+ to handle multi-task learning and found that it proved more effective than regular SVM [124].

This learning paradigm is highly useful for time-course predictive scenarios such as clinical decisions making, in which the retrospectively training data sets contain highly informative additional information that cannot be used by conventional machine learning. For instance, in the case of the early detection and prediction of ARDS, while the conventional machine learning would only consider the features extracted from clinical/physiological data for early detection of ARDS, i.e. the information in X, an LUPI paradigm can consider the available data, X, along with the chest x-rays that are not available at the time of the actual decision making, i.e. X^*. In other words, since the CXRs are available in the training datasets, they can serve as the privileged information when creating the function $f_\gamma(x)$ that maps the clinical/physiological data to the presence or absence of ARDS, i.e. {-1,1}.

In this project, we implement the LUPI paradigm in several machine learning models for detection patients with ARDS. This initial study uses CXR ratings by clinicians as privileged information. Chapter V provides in-depth details on the optimization algorithms, extension for partially available information, and incorporation with label uncertainty with feature extraction methods as described in §3.5.

4.2 Implementation with Support Vector Machines (SVM+)

The formulation of LUPI for SVM, called SVM+ is provided here. Given a set of training data pairs:

$$(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n) \quad \mathbf{x}_i \in X, y_i \in \{-1, 1\}$$

"Standard" SVM first maps training data vector $\mathbf{x} \in X$ into vector $\mathbf{z} \in Z$ where it constructs the optimal separating hyperplane by learning the decision rule $f(\mathbf{z}) = \mathbf{w} \cdot \mathbf{z} + b$ where $\mathbf{w}$ and b are hyperplane parameters and the solution of:

$$\begin{gathered}\min_{\mathbf{w},b,\xi} \tfrac{1}{2}\|\mathbf{w}\|_2^2 + C \textstyle\sum_{i=1}^{n} \xi_i \\ \text{s.t. } \forall 1 \leq i \leq n, y_i (\mathbf{w} \cdot \mathbf{z}_i + b) \geq 1 - \xi_i \\ \forall 1 \leq i \leq n, \xi_i \geq 0\end{gathered} \tag{4.1}$$

where $C > 0$ is a hyperparameter.

In the LUPI paradigm, the set of training data triplets are:

$$(\mathbf{x}_1, \mathbf{x}_1^*, y_1), \ldots, (\mathbf{x}_n, \mathbf{x}_n^*, y_n) \quad \mathbf{x}_i \in X, \mathbf{x}_i^* \in X^* \quad y_i \in \{-1, 1\}$$

where $\mathbf{x}_i^*$ is the privileged information. Let $\mathbf{z}_i^*$ be a feature map of $\mathbf{x}_i^*$. The SVM formulation for LUPI, i.e. SVM+, can be thought of as the classical SVM with the same decision function $f(\mathbf{z}) = \mathbf{w} \cdot \mathbf{z} + b$ but a correcting function, $\varphi(\mathbf{z}^*) = \mathbf{w}^* \cdot \mathbf{z}^* + b^*$, in lieu of the slack variables. Then the decision rule and the correcting function hyperplane parameters are achieved simultaneously by SVM+ optimization of:

$$\begin{gathered}\min_{\mathbf{w},b,\mathbf{w}^*,b^*} \tfrac{1}{2}\|\mathbf{w}\|_2^2 + \tfrac{\gamma}{2}\|\mathbf{w}^*\|_2^2 + C \textstyle\sum_{i=1}^{n} (\mathbf{w}^* \cdot \mathbf{z}_i^* + b^*) \\ \text{s.t. } \forall 1 \leq i \leq n, y_i (\mathbf{w} \cdot \mathbf{z}_i + b) \geq 1 - (\mathbf{w}^* \cdot \mathbf{z}_i^* + b^*) \\ \forall 1 \leq i \leq n, \mathbf{w}^* \cdot \mathbf{z}_i^* + b^* \geq 0\end{gathered} \tag{4.2}$$

where $C > 0$ and $\gamma > 0$ are hyperparameters. Note that $\frac{\gamma}{2}\left\|\mathbf{w}^*\right\|_2^2$ restricts the VC-dimension of the correcting function space. As implied in [11], replacing the slack variables with the smooth correcting function $\varphi\left(\mathbf{z}^*\right) = \mathbf{w}^* \cdot \mathbf{z}^* + b^*$ may not always be the best choice. Instead we can use a mixture of slacks as:

$$\xi_i' = \left(\mathbf{w}^* \cdot \mathbf{z}_i^* + b^*\right) + \rho\xi_i^* \quad \forall 1 \leq i \leq n \tag{4.3}$$

which results in the following optimization problem:

$$\begin{gathered}
\min_{\mathbf{w},b,\mathbf{w}^*,b^*,\xi^*} \tfrac{1}{2}\|\mathbf{w}\|_2^2 + \tfrac{\gamma}{2}\left\|\mathbf{w}^*\right\|_2^2 + C\textstyle\sum_{i=1}^n \xi_i^* \\
+C^* \textstyle\sum_{i=1}^n \left(\mathbf{w}^* \cdot \mathbf{z}_i^* + b^*\right) \\
\text{s.t. } \forall 1 \leq i \leq n, y_i\left(\mathbf{w} \cdot \mathbf{z}_i + b\right) \geq 1 - \left(\mathbf{w}^* \cdot \mathbf{z}_i^* + b^*\right) - \xi_i^* \\
\forall 1 \leq i \leq n, \mathbf{w}^* \cdot \mathbf{z}_i^* + b^* \geq 0 \quad \xi_i^* \geq 0
\end{gathered} \tag{4.4}$$

where $C = \rho C^*$. If $C \gg C^*$ (i.e. $\rho \gg 1$) this problem behaves similar to (4.4), and if $C \ll C^*$ (i.e. $0 < \rho \ll 1$) it converges to the solution of the conventional SVM with soft-margin (4.1). The dual optimization problem of 4.4 is further discussed in §5.2.

4.3 Implementation with Decision Trees

The proposed method for implementation of LUPI with decision trees is called iterative privileged learning. In the gradient step that is used from step $t-1$ to step t, auxiliary variables are introduced. These encode the privileged information. In other words, auxiliary functions are generated that gives an output corresponding to privileged information.

We want to use a quadratic loss function to determine the “discrepancy” between the label predicted by the model at an iteration t, and the ground truth labels. We use an approach implementing the LUPI method in the theory of Gradient Boosted

Decision Trees (GBDT) [125], modifying the loss function to incorporate privileged information.

Suppose our dataset consists of triplets $(x_i, x{*}_i, y_i)$, where x_i and $x{*}_i$ are vectors with real entries (possibly different lengths, since the number of privileged features might be different from the number of non-privileged features), and y_i are the ground truth labels. In standard GBDT, the loss function used to update the decision trees at each iteration of the algorithm is given by:

$$L = \frac{1}{2} \sum \left(G_t\left(x_i\right) - y_i\right)^2$$

Where $G_t\left(x_i\right)$ is the decision function at step t, evaluated on the vector x_i. At the subsequent step of the algorithm, G_t is updated by $G_t + h_t$, where the latter term is a "small" tree that perturbs the one obtained at step t. The new perturbing decision tree can be determined by minimizing quantity:

$$\sum \left[h_t\left(x_i\right) - \left(G_t\left(x_i\right) - y_i\right)\right]^2$$

over all small trees (for some fixed number of nodes etc.). GBDT with privileged information includes an additional term that takes into account the privileged information, corresponding to:

$$\sum C \left[h_t\left(x_i\right) - \left(G_t\left(x_i\right) - y_i\right) - w_t x_i\right]^2$$

for real numbers w_t that vary during the learning process.

The change of w_t has the meaning of an adjustment of the teacher's instructions during the learning process, which has quite an intuitive meaning as compared to human learning. The number C is a hyper-parameter whose optimal value is set to be determined during cross-validation. We implement the LUPI methodology by including a mixture of slack variables, μ_i, μ_i^* and $a \cdot \mu^* + b$. The coefficients a are updated during learning process, in analogy to the update of w_t, whose significance was previously pointed out. The objective function to minimize is therefore given by:

$$\min \sum \left[h_t\left(x_i\right) \, - \left(G_t\left(x_i\right) - y_i\right)\right]^2 + \sum_t C\left[h_t\left(x_i\right) - \left(G_t\left(x_i\right) - y_i\right) - w_t x_i\right]^2$$

$$+C'\mu_i + C''\mu_i * + C'''(\mathbf{a} \cdot \mu * + b)$$

4.4 Application

In this project, we implement the LUPI paradigm in several machine learning models for detection patients with ARDS. This initial study uses CXR ratings by clinicians as privileged information and features previously described in §2.5.1 and Appendix A. This initial study attempts to address the issue of unreliable data and using the LUPI paradigm to incorporate additional data available during model training but not accessible during testing.

4.4.1 Data

The ARDS dataset used in this study consisted of 485 patients with either moderate hypoxia or acute hypoxic respiratory failure, treated at the University of Michigan Hospital. We received institutional review board from the University of Michigan to collect data for the study (HUM00104714) with a waiver of informed consent among study participants. Each case was independently reviewed by multiple expert clinicians for the diagnosis of ARDS; the labels for this dataset were generated using the same method described in §2.4. Multiple experts reviewed the cases as there can be disagreement between doctors reviewing the same patients for the diagnosis of ARDS [11]. Clinical experts also identified the time of ARDS onset for those patients who were deemed to have developed the condition. Patients who developed ARDS were labeled as negative before the time of onset and positive for ARDS after.

The non-privileged information consisted of 25 clinical variables (features) extracted at two-hour intervals from the patient's EHR. The clinical features are provided in Appendix A. Privileged information for each patient consisted of the average of scores among multiple clinical experts reviewing CXRs performed during the hospitalization. Each clinician gave each CXR a rating of $1 - 8$, scoring their belief that the x-ray was consistent with ARDS (8 for high-confidence ARDS and 1 for high-confidence non-ARDS). As such, privileged information is the average of these scores if the CXRis available.

In the experiment for classifying ARDS, the dataset was first split into training and testing sets as shown in Figure 4.2. In order to avoid bias toward patients, all samples from the same patient were kept exclusively in either training or testing. This yielded 323 patients in the training dataset, and the rest in the testing set. Also, due to the strong inter-dependency between samples of longitudinal patient data, the IID (independent and identically distributed) assumption was invalid. Therefore, the time-series sampling method proposed in §2.2.2 was performed to reduce inter-correlation among the longitudinal clinical data from each patient used in model training. After sampling, there were 4661 samples in the training dataset, with 1298 positive for ARDS. Since there was no sampling in the testing dataset, there were 9362 samples in the test dataset.

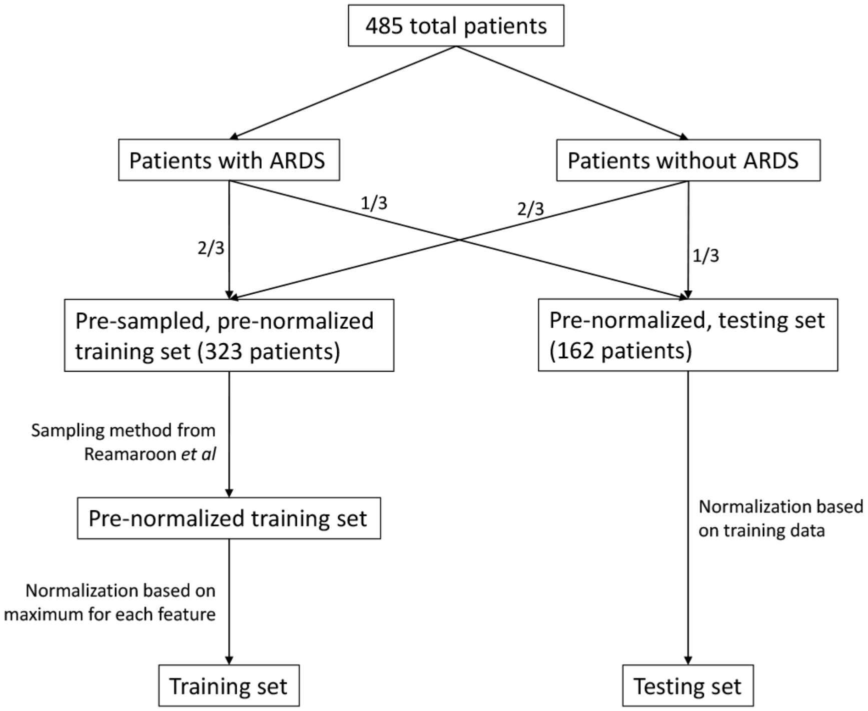

485 total patients
Patients with ARDS
Patients without ARDS
2/3
1/3
2/3
1/3
Pre-sampled, pre-normalized training set (323 patients)
Pre-normalized, testing set (162 patients)
Sampling method from Reamaroon *et al*
Pre-normalized training set
Normalization based on training data
Normalization based on maximum for each feature
Training set
Testing set

Figure 4.2: Flowchart of the study protocol. 5-fold cross-validation was performed as suggested in §2.2.2 and hyperparameter optimization was implemented with grid search.

4.4.2 Results

Table 4.1 summarizes the preliminary results of this experiment. We report that SVM+ achieves an AUROC of 69.85 on the hold-out test set, outperforming standard SVM by 13.9%. Similarly, we observe that DT+ achieves an AUROC of 67.05 and outperforms standard DT by 8.04%.

Table 4.1: Comparison of different learning paradigms in detection of ARDS.

	AUROC	Specificity at 98% Sensitivity	Specificity at 90% Sensitivity
SVM	69.85	55.23	77.90
SVM+	79.61	67.88	91.41
DT	67.05	50.30	73.58
DT+	72.44	64.91	88.47

SVM = support vector machine, SVM+ = LUPI implementation of support vector machine, DT = decision trees, DT+ = LUPI implementation of decision trees.

4.5 Discussion

This preliminary study demonstrates the capability of LUPI-based machine learning models for clinical applications and serves as the foundation for future works presented in Chapter V. From these initial experiments, we gained intuition for how to use LUPI in practical scenarios and for how to improve on the existing paradigm.

In this work, the models utilized CXR ratings by clinicians as privileged information. However, in an actual production environment, it may not be the feasible for physicians to rate each CXR in order to assist with generating the privileged information used by SVM+. A future direction for this research would be to use the methods developed in Chapters II and III to build a fully automated ARDS detection system. In other words, we propose that the previously described lung segmentation method (§3.3.1) and features extracted from the CXRs (§3.5) can serve as the privileged information used in the LUPI models while using the routinely collected EHR data can continue to be used as the standard training feature space.

One addition limitation of LUPI that hasn't been discussed yet is the fact that this learning paradigm assumes that privileged information is available for all training samples during parameter estimation. This motivates us to consider additional modification to the LUPI framework to encode such partial availability of privileged information in the training set. Based on the results presented in Table 4.1, it seems intuitive to move forward with further development of the LUPI framework with an SVM-based model. In Chapter V, we present a re-formulation the SVM model to account for both label uncertainty and "partially available" privileged information to reflect this problem.

CHAPTER V

Learning Using Label Uncertainty and Partially Available Privileged Information

5.1 Introduction

In developing the LUPI-based machine learing models for detection of ARDS, it is important to recognize that privileged information (CXRs) are sometimes only available for a portion of the training data. In other words - not every patient receives an CXR, however, the LUPI paradigm assumes that privileged information is available for all training samples during parameter estimation.

Building from LUPI framework previously described in in Chapter V, we will proposed a learning scheme that provides the capabilities of both learning from uncertain labels and learning using "partially available" privileged information to create a methodology that would better match the clinical reality of ARDS detection and many other healthcare problems. The proposed frameworks use EHR data as regular information, CXRs as partially available privileged information, and clinicians' confidence levels in ARDS diagnosis as a measure of label uncertainty.

We first describe our proposed learning from uncertain labels using privileged information (LULUPI) at a general level and then customize the model towards SVM, whose formulation make the design of such specialized learning paradigms more insightful. In the LULUPI formulation, training examples with uncertain labels and privileged information are available:

1. A subset of data with certain labels, i.e. $(x_i, x_i^*, y_i), i = 1, \ldots, N_1$, in which $x_i \in X$ form the input features, $x_i^* \in X^*$ are the privileged information on the training sample ui'', and $y_i \in \{-1, 1\}$ is the output class.

2. A subset of data with uncertain labels, i.e. $(x_i, x_i^*, y_i, l_i), i = N_1 + 1, N_1 + 2, \ldots, N$, in which $x_i \in X, x_i^* \in X^*, y_i \in \{-1, 1\}$ and $0 < l_i \leq 1$ is the level of confidence (i.e. lack of uncertainty) over sample i.

In LULUPI, the learning task involves minimizing the cost function that includes four sets of terms:

1. Terms representing in the error in mapping X to the output (as well as the complexity of the function) for data with no uncertainty,

2. Terms representing in the error in mapping X to the output (as well as the complexity of the function) for data with uncertainty,

3. Terms representing in the error in mapping X^* to the output (as well as the complexity of the function) for data with no uncertainty,

4. Terms representing in the error in mapping X^* to the output (as well as the complexity of the function) for data with uncertainty.

Another improvement considered here is to incorporate privilege information with label uncertainty, since there are many real-world machine learning scenarios in which the simultaneous use of privileged information and label uncertainty have the potential to improve model performance. Motivated by the potential benefit for such an integration into a unified paradigm, label uncertainty is incorporated into the

LUPAPI paradigm and attendant SVMp+ formulation, resulting in a general framework of learning using label uncertainty and partially available privileged information (LULUPAPI). A depiction of the LUPAPI paradigm is provided in Figure 5.1.

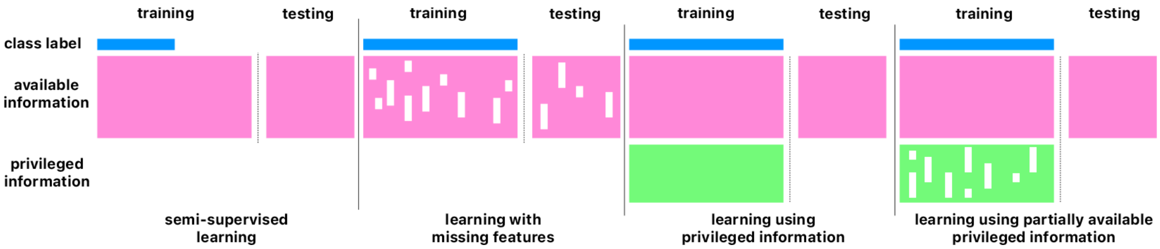


Figure 5.1: Comparison of LUPI and LUPAPI paradigms.

As there are multiple ways in which to incorporate partially available privileged information into SVM, three models are considered:

1. Vapnik's model [120], a natural extension of SVM+.

2. The mixture model, an SVMp+ formulation with a mixture of slack variables and a correcting function.

3. The symmetric mixture model, an SVMp+ formulation with a mixture of slack variables and a correcting function with label coefficients.

The end of this section describes how label uncertainty can be integrated into the SVMp+ formulation (LULUPAPI).

5.2 Implementation with Support Vector Machines (SVM+)

5.2.1 Vapnik's Model: An Initial Model for Partial Availability of Privileged Information

Fundamentally, the problem of partial availability of privileged information can be addressed using a combination of the classical SVM and standard SVM+. In other words, one can consider slack variables for the samples without privileged information and the correcting function for the samples with privileged information. This model was proposed by Vapnik et al. [120] within the LUPI framework, but not explored further.

Suppose the training data has m samples with privileged information and $n-m$ samples without privileged information:

$$\begin{gathered}\left(\mathbf{x}_1,\mathbf{x}_1^*,y_1\right),\ldots,\left(\mathbf{x}_m,\mathbf{x}_m^*,y_m\right),\left(\mathbf{x}_{m+1},y_{m+1}\right),\ldots,\left(\mathbf{x}_n,y_n\right)\\ \mathbf{x}_i\in X,\mathbf{x}_i^*\in X^*,y_i\in\{-1,1\}\end{gathered}$$

The decision rule, the slack variables, and the correcting function hyperplane parameters are achieved simultaneously by the following optimization:

$$\begin{aligned}\min\nolimits_{\mathbf{w},b,\mathbf{w}^*,b^*,\xi}\quad & \tfrac{1}{2}\|\mathbf{w}\|_2^2+\tfrac{\gamma}{2}\left\|\mathbf{w}^*\right\|_2^2+C\textstyle\sum_{i=m+1}^n\xi_i\\ & +C^*\textstyle\sum_{i=1}^m\left(\mathbf{w}^*\cdot\mathbf{z}_i^*+b^*\right)\\ \text{s.t. }\forall 1\le i\le m\quad & y_i\left(\mathbf{w}\cdot\mathbf{z}_i+b\right)\ge 1-\left(\mathbf{w}^*\cdot\mathbf{z}_i^*+b^*\right)\\ \forall 1\le i\le m\quad & \mathbf{w}^*\cdot\mathbf{z}_i^*+b^*\ge 0\\ \forall m+1\le i\le n\quad & y_i\left(\mathbf{w}\cdot\mathbf{z}_i+b\right)\ge 1-\xi_i\\ \forall m+1\le i\le n\quad & \xi_i\ge 0\end{aligned}\tag{5.1}$$

where $C>0,C^*>0$, and $\gamma>0$ are the hyperparameters. This cost function is the most natural extension of the LUPI model. The dual optimization problem of (5.1) can be written as:

$$\begin{gathered}\max\nolimits_{\boldsymbol{\alpha},\boldsymbol{\beta}}D(\boldsymbol{\alpha},\boldsymbol{\beta})=\textstyle\sum_{i=1}^n\alpha_i-\tfrac{1}{2}\sum_{i,j=1}^n\alpha_i\alpha_jy_iy_jK_{i,j}\\ -\tfrac{1}{2\gamma}\textstyle\sum_{i,j=1}^m\left(\alpha_i+\beta_i-C^*\right)\left(\alpha_j+\beta_j-C^*\right)K_{i,j}^*\\ \text{s.t.}\quad\textstyle\sum_{i=1}^n y_i\alpha_i=0\\ \textstyle\sum_{i=1}^m\left(\alpha_i+\beta_i-C^*\right)=0\\ \forall m+1\le i\le n,\quad 0\le\alpha_i\le C\\ \forall 1\le i\le m,\quad 0\le\alpha_i,0\le\beta_i\end{gathered}\tag{5.2}$$

where $K^*_{i,j} \triangleq K^*\left(\mathbf{z}^*_i, \mathbf{z}^*_j\right)$ is a kernel in the correcting space and $K_{i,j} \triangleq K\left(\mathbf{z}_i, \mathbf{z}_j\right)$ is the kernel in the decision space with the decision function:

$$f(\mathbf{z}) = \mathbf{w} \cdot \mathbf{z} + b = \sum_{i=1}^{n} y_i \alpha_i K\left(\mathbf{z}_i, \mathbf{z}\right) + b. \tag{5.3}$$

Since the aforementioned Vapnik's model was never explored further, in this paper an optimization procedure was also developed and tested for this formulation.

5.2.2 The Proposed LUPAPI Framework: SVMp+ Formulations

In this section, two realizations of the SVMp+ formulation of LUPAPI are provided: the mixture model and the symmetric mixture model.

5.2.2.1 Mixture Model

This formulation of SVMp+ can be thought of as SVM+ with the mixture model of slacks as:

$$\xi'_i = \left(\mathbf{w}^* \cdot \mathbf{z}^*_i + b^*\right) + \rho \xi^*_i \quad \forall 1 \leq i \leq n \tag{5.4}$$

In this case, the slack variables are considered for all training samples, and the correcting function only for those samples with privileged information. The decision rule, the slack variables, and the correcting function hyperplane parameters are achieved simultaneously by the following optimization:

$$\begin{aligned}
\min_{\mathbf{w},b,\xi,\mathbf{w}^*,b^*,\xi^*} \quad & \tfrac{1}{2}\|\mathbf{w}\|_2^2 + \tfrac{\gamma}{2}\left\|\mathbf{w}^*\right\|_2^2 + C\textstyle\sum_{i=m+1}^{n}\xi_i \\
& +\rho C^* \textstyle\sum_{i=1}^{m}\xi_i^* + C^*\textstyle\sum_{i=1}^{m}\left(\mathbf{w}^*\cdot\mathbf{z}_i^* + b^*\right) \\
\text{s.t. } \forall 1 \le i \le m \quad & y_i\left(\mathbf{w}\cdot\mathbf{z}_i + b\right) \ge 1 - \left(\mathbf{w}^*\cdot\mathbf{z}_i^* + b^*\right) - \xi_i^* \\
\forall 1 \le i \le m \quad & \mathbf{w}^*\cdot\mathbf{z}_i^* + b^* \ge 0 \\
\forall 1 \le i \le m \quad & \xi_i^* \ge 0 \\
\forall m+1 \le i \le n \quad & y_i\left(\mathbf{w}\cdot\mathbf{z}_i + b\right) \ge 1 - \xi_i \\
\forall m+1 \le i \le n \quad & \xi_i \ge 0
\end{aligned} \tag{5.5}$$

The dual optimization problem of (5.5) can be formulated as:

$$\begin{aligned}
& \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} D(\boldsymbol{\alpha},\boldsymbol{\beta}) = \textstyle\sum_{i=1}^{n}\alpha_i - \tfrac{1}{2}\textstyle\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j K_{i,j} \\
& -\tfrac{1}{2\gamma}\textstyle\sum_{i,j=1}^{m}\left(\alpha_i + \beta_i - C^*\right)\left(\alpha_j + \beta_j - C^*\right)K_{i,j}^* \\
& \text{ s.t.} \qquad \textstyle\sum_{i=1}^{n} y_i\alpha_i = 0 \\
& \quad \textstyle\sum_{i=1}^{m}\left(\alpha_i + \beta_i - C^*\right) = 0 \\
& \forall m+1 \le i \le n, \quad 0 \le \alpha_i \le C \\
& \forall 1 \le i \le m, \quad 0 \le \alpha_i \le \rho C^*, \quad 0 \le \beta_i
\end{aligned} \tag{5.6}$$

5.2.2.2 Symmetric Mixture Model

In this model, the goal is to better transfer the knowledge obtained in the privileged information space to the decision space by allowing the privileged information and the training label to interact, as suggested in [121]. Instead of the mixture model of slacks in (5.4), consider the following mixture for the LUPAPI model:

$$\xi_i' = y_i\left(\mathbf{w}^*\cdot\mathbf{z}_i^* + b^*\right) + \rho\xi_i^* \quad \forall 1 \le i \le m \tag{5.7}$$

The problem can then be written as:

$$\begin{aligned} \min_{\mathbf{w},b,\xi,\mathbf{w}^*,b^*,\xi^*} \quad & \tfrac{1}{2}\|\mathbf{w}\|_2^2 + \tfrac{\gamma}{2}\left\|\mathbf{w}^*\right\|_2^2 + C\textstyle\sum_{i=m+1}^{n}\xi_i \\ & +\rho C^*\textstyle\sum_{i=1}^{m}\xi_i^* + C^*\sum_{i=1}^{m} y_i\left(\mathbf{w}^*\cdot\mathbf{z}_i^* + b^*\right) \\ \text{s.t. } \forall 1\le i\le m \quad & y_i\left(\mathbf{w}\cdot\mathbf{z}_i + b\right) \ge 1 - y_i\left(\mathbf{w}^*\cdot\mathbf{z}_i^* + b^*\right) - \xi_i^* \\ \forall 1\le i\le m \quad & y_i\left(\mathbf{w}^*\cdot\mathbf{z}_i^* + b^*\right) \ge 0 \\ \forall 1\le i\le m \quad & \xi_i^* \ge 0 \\ \forall m+1\le i\le n \quad & y_i\left(\mathbf{w}\cdot\mathbf{z}_i + b\right) \ge 1 - \xi_i \\ \forall m+1\le i\le n \quad & \xi_i \ge 0 \end{aligned} \tag{5.8}$$

The corresponding dual problem can be formulated as:

$$\begin{gathered} \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} D(\boldsymbol{\alpha},\boldsymbol{\beta}) = \textstyle\sum_{i=1}^{n}\alpha_i - \tfrac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j K_{i,j} \\ -\tfrac{1}{2\gamma}\textstyle\sum_{i,j=1}^{m}\left(\alpha_i+\beta_i-C^*\right)\left(\alpha_j+\beta_j-C^*\right) y_i y_j K_{i,j}^* \\ \text{s.t.} \quad \textstyle\sum_{i=1}^{n} y_i\alpha_i = 0 \\ \forall m+1\le i\le n, \quad 0\le\alpha_i\le C \\ \textstyle\sum_{i=1}^{m} y_i\left(\alpha_i+\beta_i-C^*\right) = 0 \\ \forall m+1\le i\le m, \quad 0\le\alpha_i\le\rho C^*, \quad 0\le\beta_i \end{gathered} \tag{5.9}$$

This model is referred to as *symmetric* because the y_i coefficients are considered for the hyperplane in the privileged space as well. This model differs from that proposed [121] in two ways - it incorporates partially available privileged information and allows for two separate sets of slack variables for the training and privileged spaces.

5.2.2.3 LULUPAPI: Incorporating Label Uncertainty within the SVMp+ Formulations

In this section, label uncertainty is integrated into the SVMp+ formulation of LUPAPI, yielding the Learning Using Label Uncertainty and Partially Available Privileged Information (LULUPAPI) model. To avoid repetition, only the mixture model of LUPAPI (described in §5.2.2) is considered. In order to incorporate label uncertainty, one can vary the parameter C for training samples in proportion to their respective label confidence.

As the slack variables ξ_i (or the correcting function) permit some misclassification with penalty parameter C to establish soft-margin decision boundaries, data with high label confidence can be given more weight and subsequent influence on the decision boundary. This yields the LULUPAPI paradigm, which requires the training samples:

$$\begin{gathered}
\left(\mathbf{x}_1, \mathbf{x}_1^*, y_1, \pi_1\right), \ldots, \left(\mathbf{x}_m, \mathbf{x}_m^*, y_m, \pi_m\right), \left(\mathbf{x}_{m+1}, y_{m+1}, \pi_{m+1}\right) \\
\left(\mathbf{x}_{m+2}, y_{m+2}, \pi_{m+2}\right), \ldots, \left(\mathbf{x}_n, y_n, \pi_n\right) \\
\mathbf{x}_i \in X, \mathbf{x}_i^* \in X^*, y_i \in \{-1, 1\}, \pi_i \geq 0
\end{gathered}$$

where π_i is a quantitative measure of uncertainty in the labels. In this case, the LULUPAPI mixture model is:

$$\begin{aligned}
\min_{\mathbf{w}, b, \xi, \mathbf{w}^*, b^*, \xi^*} \quad & \tfrac{1}{2}\|\mathbf{w}\|_2^2 + \tfrac{\gamma}{2}\left\|\mathbf{w}^*\right\|_2^2 + C \textstyle\sum_{i=m+1}^{n} \pi_i \xi_i \\
& + \rho C^* \textstyle\sum_{i=1}^{m} \pi_i \xi_i^* + C^* \sum_{i=1}^{m} y_i \left(\mathbf{w}^* \cdot \mathbf{z}_i^* + b^*\right) \\
\text{s.t. } \forall 1 \leq i \leq m \quad & y_i \left(\mathbf{w} \cdot \mathbf{z}_i + b\right) \geq 1 - y_i \left(\mathbf{w}^* \cdot \mathbf{z}_i^* + b^*\right) - \xi_i^* \\
\forall 1 \leq i \leq m \quad & y_i \left(\mathbf{w}^* \cdot \mathbf{z}_i^* + b^*\right) \geq 0 \\
\forall 1 \leq i \leq m \quad & \xi_i^* \geq 0 \\
\forall m+1 \leq i \leq n \quad & y_i \left(\mathbf{w} \cdot \mathbf{z}_i + b\right) \geq 1 - \xi_i \\
\forall m+1 \leq i \leq n \quad & \xi_i \geq 0
\end{aligned} \tag{5.10}$$

and the dual optimization problem is:

$$\begin{gathered}
\max_{\boldsymbol{\alpha}, \boldsymbol{\beta}} D(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \textstyle\sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K_{i,j} \\
-\tfrac{1}{2\gamma} \textstyle\sum_{i,j=1}^{m} \left(\alpha_i + \beta_i - C^*\right)\left(\alpha_j + \beta_j - C^*\right) K_{i,j}^* \\
\text{s.t.} \qquad \textstyle\sum_{i=1}^{n} y_i \alpha_i = 0 \\
\textstyle\sum_{i=1}^{m} \left(\alpha_i + \beta_i - C^*\right) = 0 \\
\forall 1 \leq i \leq m, \quad 0 \leq \alpha_i \leq \rho \pi_i C^*, \quad 0 \leq \beta_i
\end{gathered} \tag{5.11}$$

5.2.3 An Alternating SMO Algorithm for SVMp+ Formulations of LUPAPI and LULUPAPI Paradigms

A widely used algorithm for solving conventional SVM and SVM+ is Sequential Minimal Optimization (SMO) [126]. SMO-style algorithms iteratively maximize the dual cost function by selecting the best maximally sparse feasible direction in each iteration and updating the corresponding α_i and β_j such that the dual constraints are also satisfied.

A variant of SMO called Alternating SMO for solving SVM+ was previously introduced Pechyony et at. [127, 128]. Inspired by this optimization method, an alternating SMO-style algorithm for SVMp+ is proposed. The SVMp+ dual optimization problems of (5.2), (5.6), (5.9), and (5.11) can be considered as the general form of:

$$\max_{\boldsymbol{\theta} \in \mathcal{F}} D(\boldsymbol{\theta}),$$

where $\boldsymbol{\theta} \in \mathbb{R}^k, D : \mathbb{R}^k \to \mathbb{R}$ is a concave quadratic function, and $\mathcal{F}$ is a convex compact set defined by linear equalities and inequalities.

In order to achieve an alternating SMO algorithm for SVMp+, all feasible directions for each model must be determined. Feasible and maximally sparse feasible directions were defined in [127] as follows:

Definition 1. A direction $\mathbf{u} \in \mathbb{R}^k$ is feasible at the point $\boldsymbol{\theta} \in \mathcal{F}$ if there exists $\lambda > 0$ such that $\boldsymbol{\theta} + \lambda \mathbf{u} \in \mathcal{F}$.

Definition 2. A direction $\mathbf{u}_1 \in \mathbb{R}^k$ with $n_1 < k$ zero elements is maximally sparse feasible if any $\mathbf{u}_2 \in \mathbb{R}^k$ with $n_2 < k$ zero elements such that $n_1 < n_2$ is not feasible.

The cost function in equations (5.2), (5.6), (5.9), and (5.11) have $n + m$ variables: $\{\alpha_i\}_{i=1}^{n}$ and $\{\beta_i\}_{i=1}^{m}$. These can be combined into a single $(n+m)$ variable vector $\boldsymbol{\theta}$ by concatenating the $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ variables: $\boldsymbol{\theta} \triangleq (\boldsymbol{\alpha}, \boldsymbol{\beta})^T$. Thus, each maximally sparse feasible direction is $\mathbf{u} \in \mathbb{R}^{n+m}$. It can be verified that the cost functions in equations (5.2), (5.6), and (5.11) have 9 sets of such directions, and (5.9) has 10. Following [127], each set of feasible directions is denoted by I_i. The detailed descriptions of the feasible directions and other optimization information for Vapnik's model (5.2), the mixture model (5.6), the symmetric mixture model (5.9), and the LULUPAPI mixture model (5.11) formulations can be found in Appendices C, D, E, and F, respectively.

5.2.3.1 Optimization Process

Similar to the SMO algorithm for the LUPI model [127, 128], the recursive step in the proposed optimization for the LUPAPI and LULUPAPI paradigms is finding $\boldsymbol{\theta} = \boldsymbol{\theta}^{\text{old}} + \lambda^*(\mathbf{s})\mathbf{u}_\mathbf{s}$ such that $\mathbf{u}_\mathbf{s} \in \cup I_i$ of the corresponding feasible directions and the step size $\lambda^*(\mathbf{s})$ maximize the corresponding cost function $\psi(\lambda) = D\left(\boldsymbol{\theta}^{\text{old}} + \lambda\mathbf{u}_\mathbf{s}\right)$, while satisfying the constraints. Hence, given the cost function, its constraints, and the corresponding feasible directions, the recursive optimization process of the proposed alternating SMO-style algorithm is the same in both the LUPAPI and LULUPAPI contexts.

Let $\mathrm{g}\left(\theta^{\text{old}}\right)$ and H respectively be the gradient at point θ^{old} and the Hessian of the cost function. Using the Taylor expansion of $\psi(\lambda)$ at point $\lambda = 0$ yields:

$$\begin{aligned}\lambda'\left(\boldsymbol{\theta}^{\text{old}}, \mathbf{s}\right) = \arg\max_{\lambda \geq 0} \psi(\lambda) &= -\left.\frac{\frac{\partial\psi(\lambda)}{\partial\lambda}}{\frac{\partial^2\psi(\lambda)}{\partial\lambda^2}}\right|_{\lambda=0} \\ &= -\frac{\mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T \mathbf{u}_\mathbf{s}}{\mathbf{u}_\mathbf{s}^\mathbf{T} H \mathbf{u}_\mathbf{s}}\end{aligned}$$

Let $\tau > 0$ be small constant. Define: $I = \left\{\mathbf{u}_\mathbf{s} \mid \mathbf{u}_\mathbf{s} \in \bigcup I_i, \mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T \mathbf{u}_\mathbf{s} > \tau\right\}$. If $I = \emptyset$, then the algorithm stops. Suppose $I \neq \emptyset$, define:

$$\widetilde{I}_i = \left\{\mathbf{u}_\mathbf{s} \mid \mathbf{u}_\mathbf{s} \in I_i, \mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T \mathbf{u}_\mathbf{s} > \tau\right\}$$

For each non-empty $\widetilde{I}_i$, find the vector $\mathbf{u}_{\mathbf{s}^{(1)}} \in \widetilde{I}_i$ that has the minimal angle with g $\left(\theta^{\text{old}}\right)$ among all the candidates in $\widetilde{I}_i$:

$$\mathbf{s}^{(i)} = \arg \max_{\mathbf{s}:\mathbf{u}_{\mathbf{s}} \in \widetilde{I}_i} \mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T \mathbf{u}_{\mathbf{s}} \tag{5.12}$$

In the next step, for the directions containing pairs, if $\mathbf{s}^{(i)} = \left(s_1^{(i)}, s_2^{(i)}\right) \neq \emptyset$, fix the value of $s_1^{(i)}$ and find $\mathbf{s}'^{(i)} = \left(s_1^{(i)}, s_2'^{(i)}\right)$ such that $\mathbf{u}_{\mathbf{s}'(\mathbf{i})} \in \widetilde{I}_i$ and

$$\begin{aligned} \mathbf{s}'^{(i)} &= \arg \max_{\mathbf{t}:\mathbf{t}=\left(s_1^{(i)},t_2\right)} D\left(\boldsymbol{\theta}^{\text{old}} + \lambda'\left(\boldsymbol{\theta}^{\text{old}}, \mathbf{t}\right)\mathbf{u}_{\mathbf{t}}\right) - D\left(\boldsymbol{\theta}^{\text{old}}\right) \\ &= \arg \max_{\mathbf{t}:\mathbf{t}=\left(s_1^{(i)},t_2\right)} \frac{-\left(\mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T \mathbf{u}_{\text{t}}\right)^2}{\mathbf{u}_{\mathbf{t}}^{\mathbf{T}} H \mathbf{u}_{\mathbf{t}}} \end{aligned} \tag{5.13}$$

where the last equality is achieved by substituting $\lambda'\left(\theta^{\text{old}}, s\right)$ of equation (24) into $D\left(\boldsymbol{\theta}^{\text{old}} + \lambda'\left(\boldsymbol{\theta}^{\text{old}}, \mathbf{t}\right)\mathbf{u}_{\text{t}}\right) - D\left(\boldsymbol{\theta}^{\text{old}}\right)$ Similarly, for the directions containing triplets, if $\mathbf{s}^{(i)} = \left(s_1^{(i)}, s_2^{(i)}, s_3^{(i)}\right) \neq \emptyset$, fix the value of $s_1^{(i)}$ and $s_3^{(i)}$, and find $\mathbf{s}'^{(i)} = \left(s_1^{(i)}, s_2'^{(i)}, s_3^{(i)}\right)$ such that $\mathbf{u}_{\mathbf{s}'(\mathbf{i})} \in \widetilde{I}_i$ and

$$\mathbf{s}'^{(i)} = \arg \max_{\mathbf{t}:\mathbf{t}=\left(s_1^{(i)},t_2,s_3^{(i)}\right)} \frac{-\left(\mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T \mathbf{u}_{\text{t}}\right)^2}{\mathbf{u}_{\mathbf{t}}^{\text{T}} H \mathbf{u}_{\text{t}}} \tag{5.14}$$

Among all the possible directions from $\mathbf{u}_{\mathbf{s}'(\mathbf{i})} \in \cup \tilde{I}_i$, the optimal direction that maximizes the cost function is chosen:

$$\mathbf{s}^{*(i)} = \arg \max_{\mathbf{s}'(i) \neq \emptyset} \frac{-\left(\mathbf{g}\left(\boldsymbol{\theta}^{\text{old}}\right)^T \mathbf{u}_{\mathbf{t}}\right)^2}{\mathbf{u}_{\mathbf{t}}^{\mathbf{T}} H \mathbf{u}_{\mathbf{t}}} \tag{5.15}$$

Having chosen the optimal direction $\mathbf{s}^{*(i)}$, the value of $\left(\boldsymbol{\theta}^{\text{old}}, \mathbf{s}^{*(i)}\right)$ should be clipped such that it satisfies the upper/lower bound constraints on $\{\alpha_i\}_{i=1}^n$ and $\{\beta_i\}_{i=1}^m$. Clipping functions are specific to the dual problems of each SVMp+ formulation and can be found in the Appendices C, D, E, and F.

5.2.3.2 Algorithm

Having described the framework for the proposed alternating SMO-style algorithm that solves the SVMp+ dual cost functions of the LUPAPI and LULUPAPI models, the resultant algorithm is codified in Algorithm 3. Note that most of the calculations for I_i can be performed once, rather than in each iteration, as they depend largely on the label and indices of the training data samples. One can consider various initial conditions for the α and β variables. Since the feasible directions and clipping function ensure the satisfaction of the dual problem conditions, a satisfactory initial condition guarantees the fulfillment of these conditions in each iteration. In all variants of the LUPAPI and LULUPAPI models, the simplest initial condition that satisfies all of the constraints is $\alpha_i^{(0)} = 0$ and $\beta_i^{(0)} = C^*$.

Algorithm 3: Alternating SMO-style Optimization for SVMp+ formulations of LUPAPI and LULUPAPI

```
Require: Training data, training labels, $\tau > 0, \gamma > 0, C > 0, C^* > 0$, and $0 < \epsilon \ll 1$.
Calculate: Kernels $K$ and $K^*$, Hessian $H$
Initialize: $\theta_i^{(0)} \left(\text{i.e.}, \alpha_i^{(0)} \text{ and } \beta_i^{(0)}\right)$
Initialize: $I_i$ for each feasible direction based on the indexes and training labels.
while exists a maximally sparse feasible direction $u_s$ s.t. $\mathbf{g}\left(\boldsymbol{\theta}^{new}\right)^T \mathbf{u_s} > \tau$ and
  $\left(D\left(\boldsymbol{\theta}^{new}\right) - D\left(\boldsymbol{\theta}^{old}\right)\right) > \epsilon$ do
    $\theta^{old} = \theta^{new}$
    Calculate $g(\theta^{old})$
    Update $I_i$ for all $i$ based on $\theta^{old}$
    Calculate $\tilde{I}_i$ if $I_i \neq \emptyset$
    Calculate $\mathbf{s}^{(i)}$ if $\widetilde{I}_i \neq \emptyset$ using (5.12)
    Calculate $\mathbf{s}'(i)$ if $\widetilde{I}_i \neq \emptyset$ using (5.13) or (5.14)
    Calculate $\mathbf{s}^*(i)$ if $\cup\widetilde{I}_i \neq \emptyset$ using (5.15) Calculate $\lambda^*$ using the corresponding
      clipping function
    Update $\boldsymbol{\theta}^{\text{new}} = \boldsymbol{\theta}^{\text{old}} + \lambda^* u_{\mathbf{s}^*}$
```

As previously mentioned, while the proposed optimization process for the LUPAPI and LULUPAPI models is the same given the dual cost function and the corresponding

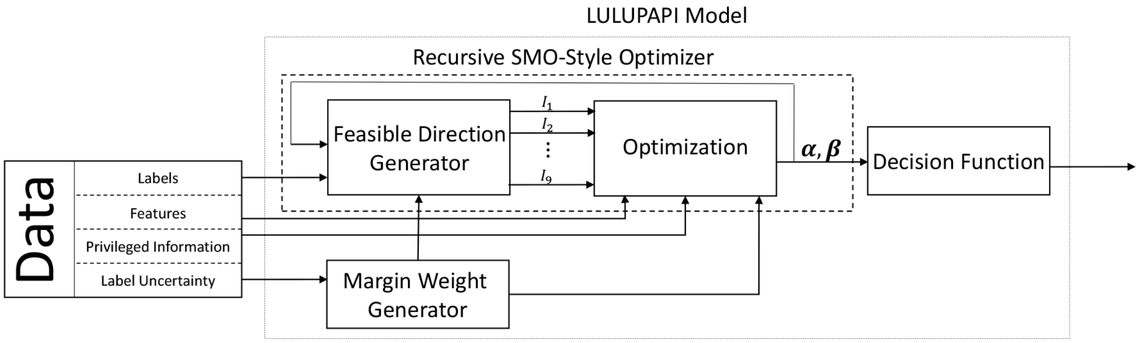


Figure 5.2: Alternating SMO-style optimizer for the LULUPAPI model.

feasible directions, the LULUPAPI model is always the most comprehensive model and any given algorithm for LULUPAPI can easily be modified to realize a LUPAPI version. For instance, any algorithm specifically designed for the LULUPAPI mixture model can be made into a LUPAPI mixture model by simply replacing the uncertainty coefficients with unity. A general schematic diagram of the LULUPAPI iterative optimizer is depicted in Figure 5.2.

5.3 Application

In this work, a unified framework for handling machine learning tasks in which privileged learning is partially available is presented, while simultaneously correcting for label uncertainty. As there are multiple means of incorporating partially available privileged information into SVM, three models were considered: Vapnik's model [120]; and two new SVMp+ formulations, the mixture and symmetric mixture models. An alternating SMO-style optimization algorithm was provided that solves all model formulations.

We implement the proposed LUPAPI and LULUPAPI paradigm in several machine learning models for detection patients with ARDS. The proposed frameworks use EHR data as standard information, CXRs as partially available privileged information, and clinicians' confidence levels in ARDS diagnosis as a measure of label uncertainty.

Unlike the previous study in Chapter IV that relied on CXR ratings from physicians, this work uses feature extracted from the CXR as the privileged information. As previously described, lung segmentation is performed with TVAC (§3.3.1) and features are extracted from the lung fields. In this work, the extracted features are obtained from Directional Blur (§3.5.1), the histogram (§3.5.2), gray-level co-occurence matrix (§3.5.3), and deep learning (§3.5.4).

5.3.1 Data

This work uses the same dataset as previously described in §2.5.1 and §3.6.1.

5.3.2 Results

For the ARDS dataset in the previous section, a LUPAPI or LULUPAPI formulation was the best performing model. In the LUPAPI experiments on the ARDS dataset as depicted in Table 5.1, the mixture model was the best performing model, achieving an AUC of 85.78%, a 2.8% improvement over SVM. Using the same mixture model, but incorporating label uncertainty, the LULUPAPI formulation in Table 5.2 achieved an AUC of 87.01%, a 4.3% improvement over SVM. The statistical tests in Table 5.3 verify the statistical significance of improvements in performance. Additionally, Table 5.5 shows that LULUPAPI formulation achieved 2.39% improvement over the most competitive deep learning method.

Table 5.1: LUPAPI results for ARDS classification.

	Train				Test			
	Accuracy	Sensitivity	Specificity	AUC	Accuracy	Sensitivity	Specificity	AUC
SVM	88.61	80.43	91.76	86.10	89.27	76.59	90.33	83.46
Vapnik's Model	86.01	78.51	88.91	83.71	88.42	75.62	89.50	82.56
Mixture Model	88.09	81.28	90.72	86.00	88.78	82.23	89.34	85.78
Symmetric Mixture Model	88.56	81.59	91.26	86.42	88.94	81.13	89.60	85.37

Table 5.4: Comparison of different LUPI paradigms in detection of ARDS.

	Accuracy	AUROC	Specificity at 98% Sensitivity	Specificity at 90% Sensitivity
SVM	78.04	81.13	45.71	59.18
SVM+	83.81	83.75	51.02	62.83
SVM w/ LU	81.57	85.48	52.85	64.50
DT	73.59	67.05	50.30	73.58
DT+	77.30	72.44	64.91	88.47

SVM = support vector machine, SVM+ = LUPI implementation with support vector machine, SVM w/ LU = support vector machine with label uncertainty, DT = decision tree, DT+ = LUPI implementation with decision trees.

Table 5.2: ARDS classification results using SVM.

	Train				Test				
	Accuracy	Sensitivity	Specificity	AUC	Accuracy	Sensitivity	Specificity	AUC	F1 Score
SVM	88.61	80.43	91.76	86.10	89.27	76.58	90.33	83.46	79.73
SVM with LU	87.56	78.74	90.96	84.85	89.52	80.03	90.32	85.17	79.60
LUPAPI	88.09	81.28	90.72	86.00	88.78	82.23	89.34	85.78	79.17
LULUPAPI	87.96	83.59	89.65	86.62	88.38	85.40	88.63	87.01	79.46

The McNemar test [129] was employed to assess the statistical significance of improvements in performance of the proposed models over SVM. Since this test is insensitive to the proportion of positive versus negative cases [130], the test was applied exclusively to positive cases. Table 5.3 summarizes the results of the McNemar tests and verifies the statistical significance of incorporating both label uncertainty and partial available privileged information in detection of patients with ARDS.

Table 5.3: McNemar $\mathcal{X}^2$ test assessment of statistical significance of performance improvements exclusively among ARDS patients.

	SVM	SVM w/ LU	LUPAPI	LULUPAPI
SVM	0	12.80	39.02	56.70
SVM w/ LU	X	0	10.22	33.58
LUPAPI	X	X	0	15.61
LULUPAPI	X	X	X	0

In this table, each row represents the *null* classifier, and each column represents the *alternative* classifier. For example, LULUPAPI versus SVM has the MeNemar test statistic $\mathcal{X}^2 = 56.70$, which is extremely in favor of LULUPAPI (p-value $\ll 0.001$), If the *null* classifier outperforms the alternative classifier, the value is represented with an X.

Beyond comparisons with SVM-based methods, the LUPAPI and LULUPAPI

Table 5.5: Comparison of ARDS classification using LUPAPI mixture model, LULUPAPI mixture model, and deep learning methods.

	Train				Test			
	Accuracy	Sensitivity	Specificity	AUC	Accuracy	Sensitivity	Specificity	AUC
LUPAPI	88.09	81.82	90.72	86.00	88.78	82.23	89.34	85.78
LULUPAPI	87.96	83.59	89.65	86.62	88.38	85.40	88.63	87.01
Shallow NN (2 layers, 10 nodes)	87.52	86.65	87.77	87.31	79.98	84.99	79.56	82.23
Shallow NN (2 layers, 50 nodes)	87.89	84.81	89.08	86.94	83.41	82.97	89.05	85.39
Shallow NN (2 layers, 100 nodes)	87.41	82.43	89.33	85.88	87.97	66.67	89.76	78.22
LSTM (25 layers, 10 nodes)	90.94	83.10	93.97	88.53	88.39	64.33	90.41	77.34
LSTM (25 layers, 50 nodes)	91.99	84.67	94.82	89.74	88.91	70.39	90.47	80.43
LSTM (25 layers, 100 nodes)	92.62	85.79	95.25	90.52	87.40	71.76	88.71	80.24

models were also benchmarked against multiple popular deep learning methods. A "shallow" neural network (two-layer feedforward network) with one hidden layer of either 10, 50, or 100 nodes was trained to create a less complex neural network more suitable for this type of data. In addition, a long short-term memory (LSTM) network, a specialized type of artificial recurrent neural network for time-series sequential data [131], was also trained to provide a performance comparison to a state-of-the-art deep learning algorithm. The LSTM network was composed of 25 layers with either 10, 50, or 100 hidden units. Both the shallow neural network (Shallow NN) and LSTM models were implemented with the Keras deep learning library using the Adam optimizer algorithm [132] with 500 epochs (mini-batch size of 32) and cross entropy as the loss function. Table 5.5 summarizes the results of this experiment. As can be seen, LUPAPI and LULUPAPI outperformed the deep learning methods.

5.4 Discussion

While Vapnik's model (5.1) is a natural extension of the SVM+ framework, the experiments showed that only the two SVMp+ models (5.5) and (5.8) outperform SVM on the real-world ARDS dataset. Moreover, on the ARDS dataset, which contained both partially available privileged information and label uncertainty, the LULUPAPI model incorporating both outperformed the LUPAPI model that solely considered privileged information. Even though these models were developed with clinical decision support systems in mind, the proposed models can be applied to many other machine learning applications in healthcare and other domains.

Though the LUPAPI and LULUPAPI frameworks improved performance overall, Vapnik's model underperformed the mixture models and SVM on the ARDS dataset. Based on the experiments, the primary reason for such performance is due to the offset parameter b of the decision function (the detailed calculation of which can be found in the Appendices). For the mixture models and SVM, the set N (defined in the Appendices) includes fewer $\alpha_i s$, corresponding to fewer support vectors that would be used to calculate the offset parameter. This is in turn due to the consideration of slack variables for all samples, regardless of privileged information availability. However, for Vapnik's model the set N consists of more α_s that negatively effects the offset parameter precision.

With respect to time complexity, the proposed alternating SMO-style algorithm for the LUPAPI model is $O(n)$. This is similar to the SMO algorithm for conventional SVM and the alternating SMO algorithm for the LUPI model (the SVM+ formulation), and results from the feasible direction vectors having a constant number of nonzero components. However, the experiments showed that given the same parameters and stoppage criterion, Vapnik's model required more iterations for convergence.

The experimental results also support the claim that if the hyperparameter optimization is performed thoroughly, performance of the mixture model (9) is always lower-bounded by SVM. This claim can also be verified using the dual forms, noting the bound on α_i and the inclusion of SVM feasible directions in the feasible directions of (9).

CHAPTER VI

Conclusion

In just the past decade, machine learning has made a profound impact in many areas of science and technology, including life science and medical research. Ongoing research and recent advances demonstrated the potential to transform the medical landscape - from early diagnosis through clinical decision support to epidemiology, drug development, and robotic-assisted surgery. These diverse efforts share the ultimate goal of improving quality of care and outcome for patients.

In this thesis, I propose the integration of label uncertainty and learning using privileged information to develop robust machine learning models for detection of ARDS (§5.2.2). This research also includes development of methods for time-series analysis of longitudinal health data (§2.2), signal processing techniques for quality assessment (§2.3), lung segmentation from complex CXRs (§3.3.1), and novel feature extraction algorithm for quantification of pulmonary opacification (§3.5.1). These algorithms were tested and validated on retrospective study on data obtained from hospitalized patients at Michigan Medicine in addition to data from external sources (e.g. publicly available databases). These studies demonstrate that careful, principled use of methodologies in machine learning and artificial intelligence can potentially assist healthcare providers with early detection of ARDS. However, in light of these advances, it is imperative to ensure that the developed methods are practical, reliable, clinical valid, and interpretable decision making tools.

Using AI in any existing research domain often heralds comparison to a (human) expert's performance for the same task. In research, if the domain expert achieves 90% accuracy for a given task and a machine learning model reports 95% on the same test set - we can celebrate that advances were made and significant milestones have been reached. However, if a CXR diagnosis model outperforms a panel of radiologists, should that mean healthcare providers and hospital administrators can use it for their patients? In reality, physicians and clinical experts care more about safety, reliability, bias, performance on edge cases, and a number of other factors on which can't assess with a simple performance benchmark.

Although AI has potential to improve healthcare, the process of bringing these advances to the practice seems to be the primary current setback to adoption and innovation. For example - if you've put a neural network into production but achieved slightly better results with a re-trained model, do you simply re-deploy the update model for immediate use? In research, and also in certain commercial applications, this is considered standard practice. However, in healthcare, if the updated model misses an edge case, an incorrect medical decision may be made - possibly resulting in an adverse outcome for the patient. This problem also poses significant challenges for agencies like the U.S. Food and Drug Administration (FDA) who are used to regulating products, not systems.

As such, there are is a growing need for a system view to regulate AI/ML-based software as medical devices [133]. The FDA has recently released an action plan [134] to address some of these issues. In this approach, they expressed an expectation for transparency and real-world performance monitoring by manufacturers that could enable FDA and manufacturers to evaluate and monitor a software product from its pre-market development through post-market performance. These new safety challenges would enable FDA to provide a reasonable assurance of safety and effectiveness while embracing the iterative improvement power of artificial intelligence and machine learning-based software as a medical device.

In addition to the action plan, the FDA also highlighted the importance of Good Machine Learning Practice (GMLP) through consensus driven standards and other community initiatives. They provide guidance similar to good software engineering practices (e.g. data management, feature extraction, training, interpretability, evaluation, and documentation) in an effort to facilitate the shift in evaluation through their proposed regulatory approach. Furthermore, to encourage transparency with assessment and adoption, they also provide structured requirements for demonstration of analytical validation (e.g. performance evaluation protocols to minimize data leakage if the data is used in multiple evaluations) and clinical validation (e.g. ensuring that the software is valid on the targeted population in the context of clinical care).

Although these regulatory processes can feel like additional obstacles and challenges to overcome, the updated guidelines should be seen as a starting point to accelerate adoption and improve the quality of machine learning in healthcare. This shift in perspective is crucial to maximizing the safety and efficacy of AI/ML in clinical applications. These discussions at the regulatory level and advances in research, like the works presented in this thesis, gives hope for the community to continue working together and iterating upon these existing frameworks to be fit for purpose.

APPENDICES

APPENDIX A

Data Dictionary for Covariates Used as Features

Table A.1: Covariates used for model training.

Name	Description
temp	Temperature
hr	Heart rate
rr	Respiratory rate
sbp	Systolic blood pressure
dbp	Diastolic blood pressure
PEEP	Positive end-expiratory pressure
plat	Plateau pressure
mAirP	Mean airway pressure
wbc	White blood cell count level
lactate	Lactate acid level obtained by blood gas
bicarb	Bicarbonate level
paco2	Carbon dioxide level obtained by blood gas
pH	pH level obtained by blood gas
bnp	Brain natriuretic peptide level
trop	Troponin level
alb	Albumin level

invasive	If patient is currently receiving invasive mechanical ventilation (1 = yes, 0 = no)
non_invasive	If patient is currently receiving non-invasive mechanical ventilation (1 = yes, 0 = no)
sp02	Pulse oximetry value
fi02	Level of supplemental oxygen (FiO_2 = fraction of inspired oxygen)
pf	Ratio of blood oxygen to supplemental oxygen ($\text{P/F} = \text{Pa0}_2/\text{FiO}_2$)
pressor	If patient is currently receiving vasopressor support for hemodynamic insufficiency
net_in	Net IV fluids: amount of total IV fluids given (total urinary output to date)
shock	Ratio of heart rate over systolic blood pressure

APPENDIX B

Additional Results for Directional Blur

Table B.1: Classification results using the histogram with additional features.

	Histogram + Directional Blur			Histogram + GLCM			Histogram + Deep Learning			All Features Combined		
	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1
SVM	73.55	68.61	59.42	71.65	65.78	58.43	72.64	67.88	60.48	74.86	73.41	64.06
Random Forrest	75.67	66.58	59.44	73.67	65.45	55.12	74.77	65.67	59.63	76.76	71.59	63.34
AdaBoost	**79.55**	**75.61**	**62.43**	**75.76**	**73.32**	**59.19**	**77.83**	**74.50**	**61.44**	**83.51**	**79.93**	**65.47**
RUSBoost	**77.53**	**73.48**	**62.11**	**75.67**	**72.48**	**61.33**	**76.70**	**72.35**	**63.41**	**81.93**	**77.80**	**67.56**
Robust Boost	71.55	70.61	60.13	70.52	68.94	56.37	73.96	68.44	59.52	76.47	73.82	63.42
Total Boost	70.65	69.77	68.91	70.84	67.61	55.38	71.54	66.53	58.11	73.84	68.31	59.60

Table B.2: Classification results using the GLCM with additional features.

	GLCM + Directional Blur			GLCM + Histogram			GLCM + Deep Learning			All Features Combined		
	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1
SVM	74.55	69.60	62.71	72.56	63.32	58.55	72.62	67.55	62.47	74.56	73.27	64.05
Random Forrest	76.56	68.69	59.22	74.58	65.98	56.12	74.79	66.51	59.27	76.58	71.61	63.32
AdaBoost	**81.53**	**77.80**	**63.51**	**76.63**	**74.91**	**59.36**	**77.51**	**75.81**	**61.32**	**83.87**	**79.50**	**65.49**
RUSBoost	**79.53**	**76.60**	**66.48**	**76.92**	**72.51**	**65.40**	**78.55**	**75.61**	**63.29**	**81.53**	**77.91**	**67.70**
Robust Boost	70.51	70.70	60.39	71.90	69.32	57.60	74.55	71.53	57.10	76.59	73.90	63.44
Total Boost	71.53	69.90	58.44	70.53	66.31	55.09	72.31	70.22	55.31	73.57	68.43	59.55

Table B.3: Classification results using deep learning with additional features.

	Deep Learning + Directional Blur			Deep Learning + Histogram			Deep Learning + GLCM			All Features Combined		
	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1
SVM	75.60	70.53	64.90	72.34	65.71	62.34	73.50	65.53	62.19	74.80	73.84	64.77
Random Forrest	76.71	67.95	62.30	74.59	66.91	59.04	74.61	66.10	63.37	76.88	71.30	63.43
AdaBoost	**80.55**	**77.90**	**65.13**	**76.78**	**74.32**	**62.15**	**77.59**	**75.88**	**64.30**	**83.56**	**79.89**	**65.41**
RUSBoost	**80.76**	**76.51**	**65.19**	**75.65**	**72.94**	**61.34**	**78.55**	**75.61**	**65.04**	**81.82**	**77.89**	**67.55**
Robust Boost	72.35	69.54	60.11	72.93	68.59	58.44	73.54	71.65	61.40	76.55	73.90	63.34
Total Boost	73.52	68.44	59.21	72.60	69.54	59.35	73.74	71.90	60.43	73.66	68.90	59.11

APPENDIX C

Vapnik's Model

Let θ be a $(n+m)$-variable vector as the concatenation of the α and β variables: $\theta \triangleq (\alpha, \beta)^T$.

Feasible Directions and the Clipping Function

It can be verified that the cost function in equation 5.2 has 9 sets of maximally sparse feasible directions (defined in §5.2.3) as follows:

Direction 1: $I_1 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), n+1 \leq s_1, s_2 \leq n+m, s_1 \neq s_2; u_{s_1} = 1, u_{s_2} = -1, \theta_{s_2} > 0, \forall i \notin \mathbf{s} u_i = 0\}$

Direction 2: $I_2 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), 1 \leq s_1, s_2 \leq m, s_1 \neq s_2, y_{s_1} = y_{s_2}; u_{s_1} = 1, u_{s_2} = -1, \theta_{s_2} > 0, \forall i \notin \mathbf{s} u_i = 0\}$

Direction 3: $I_3 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), m+1 \leq s_1, s_2 \leq n, s_1 \neq s_2, y_{s_1} = y_{s_2}; u_{s_1} = 1, \theta_{s_1} < C, u_{s_2} = -1, \theta_{s_2} > 0, \forall i \notin \mathbf{s} u_i = 0\}.$

Direction 4: $I_4 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), m+1 \leq s_1, s_2 \leq n, s_1 \neq s_2, y_{s_1} \neq y_{s_2}, \forall i \notin \mathbf{s} u_i = 0; u_{s_1} = u_{s_2} = 1, \theta_{s_1} < C, \theta_{s_2} < C \text{ or } u_{s_1} = u_{s_2} = -1, \theta_{s_1} > 0, \theta_{s_2} > 0\}$

Direction 5: $I_5 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \le s_1, s_2 \le m, n+1 \le s_3 \le n+m, s_1 \ne s_2, y_{s_1} \ne y_{s_2}, \forall i \notin \mathbf{s} u_i = 0; u_{s_1} = u_{s_2} = 1, u_{s_3} = -2, \theta_{s_3} > 0 \text{ or } u_{s_1} = u_{s_2} = -1, \theta_{s_1} > 0, \theta_{s_2} > 0, u_{s_3} = 2\}$

Direction 6: $I_6 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \le s_1 \le m, m+1 \le s_2 \le n, n+1 \le s_3 \le n+m, y_{s_1} = y_{s_2}, \forall i \notin \mathbf{s} u_i = 0; u_{s_1} = 1, u_{s_2} = -1, \theta_{s_2} > 0, u_{s_3} = -1, \theta_{s_3} > 0 \quad \text{or} \;\; u_{s_1} = -1, \theta_{s_1} > 0, u_{s_2} = 1, \theta_{s_2} < C, u_{s_3} = 1\}$

Direction 7: $I_7 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \le s_1 \le m, m+1 \le s_2 \le n, n+1 \le s_3 \le n+m, y_{s_1} \ne y_{s_2}, \forall i \notin \mathbf{s} u_i = 0; u_{s_1} = u_{s_2} = 1, \theta_{s_2} < C, u_{s_3} = -1, \theta_{s_3} > 0 \text{ or } u_{s_1} = u_{s_2} = -1, \theta_{s_1} > 0, \theta_{s_2} > 0, u_{s_3} = 1\}$

Direction 8: $I_8 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \le s_1, s_2 \le m, m+1 \le s_3 \le n, s_1 \ne s_2, y_{s_1} \ne y_{s_2}, y_{s_3} = y_{s_2}, \forall i \notin \mathbf{s} u_i = 0; u_{s_1} = 1, u_{s_2} = -1, \theta_{s_2} > 0, u_{s_3} = 2, \theta_{s_3} < C \text{ or } \; u_{s_1} = -1, \theta_{s_1} > 0, u_{s_2} = 1, u_{s_3} = -2, \theta_{s_3} > 0\}$

Direction 9: $I_9 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \le s_1, s_2 \le m, m+1 \le s_3 \le n, s_1 \ne s_2, y_{s_1} \ne y_{s_2}, y_{s_3} = y_{s_1}, \forall i \notin \mathbf{s} u_i = 0; u_{s_1} = 1, u_{s_2} = -1, \theta_{s_2} > 0, u_{s_3} = -2, \theta_{s_3} > 0 \text{ or } \; u_{s_1} = -1, \theta_{s_1} > 0, u_{s_2} = 1, u_{s_3} = 2, \theta_{s_3} < C\}$

Generally, a move from an old feasible point θ^{old} to a new feasible point $\boldsymbol{\theta}^{\text{old}} = \boldsymbol{\theta}^{\text{old}} + \lambda \mathbf{u_s}$ in the direction of $\mathbf{u_s} \in \cup I_i$ will satisfy all the constraints corresponding to the dual problem if the step size λ fulfills the bounding constraints of equation 5.2. In §5.2.3 it was shown how the best direction and the corresponding step size parameter λ are chosen. After determining the direction and step size, the clipping function (C.1), ensures that the aforementioned boundary conditions of the dual form are satisfied.

$$\lambda^*\left(\boldsymbol{\theta}^{\text{old}}, \mathbf{s}^{*(i)}\right) = \min_{k \in \mathbf{s}^{*(i)}, u_k > 0} \left\{ \frac{C - \theta_k^{\text{old}}}{u_k}, \min_{j \in \mathbf{s}^{*(i)}} \left(\lambda'\left(\boldsymbol{\theta}^{\text{old}}, \mathbf{s}^{*(i)}\right), \left| \frac{\theta_j^{\text{old}}}{u_j} \right| \right) \right\} \tag{C.1}$$

Offset Parameter of the Decision Function

In order to calculate the offset parameter b of the decision function, suppose α and β are the solution of the SVMp+ dual problem (2). Define the two sets $N \triangleq \{i\,|1 \leq i \leq m, \alpha_i\rangle\ 0\}$ and $N' \triangleq \{i \mid m+1 \leq i \leq n, 0 < \alpha_i < C\}$. By the conditions in SVMp+, for the support vectors the KarushKuhn-Tucker (KKT) conditions state [128, 135]:

$$\forall i \in N \quad y_i\left(\mathbf{w} \cdot \mathbf{z}_i + b\right) = 1 - \left(\mathbf{w}^* \cdot \mathbf{z}_i^* + b^*\right) \tag{C.2}$$

$$\forall i \in N' \quad y_i\left(\mathbf{w} \cdot \mathbf{z}_i + b\right) = 1 \tag{C.3}$$

Define:

$$F_i \triangleq \mathbf{w} \cdot \mathbf{z}_i\Big|_{i \in N} = \textstyle\sum_{j=1}^{n} y_j \alpha_j K_{ij}\Big|_{i \in N}$$

$$F'_i \triangleq \mathbf{w} \cdot \mathbf{z}_i\Big|_{i \in N'} = \textstyle\sum_{j=1}^{n} y_j \alpha_j K_{ij}\Big|_{i \in N'}$$

$$f_i \triangleq \gamma \mathbf{w}^* \cdot \mathbf{z}_i^*\Big|_{i \in N} = \textstyle\sum_{j=1}^{m} \left(\alpha_j + \beta_j - C^*\right) K_{ij}^*\Big|_{i \in N}$$

The equalities in equations C.2 and C.3 can be rewritten as:

$$\begin{cases} b + b^* = 1 - \frac{f_i}{\gamma} - F_i & \forall i \in N, y_i = 1 \\ b - b^* = -1 + \frac{f_i}{\gamma} - F_i & \forall i \in N, y_i = -1 \\ b = 1 - F'_i & \forall i \in N', y_i = 1 \\ b = -1 - F'_i & \forall i \in N', y_i = -1 \end{cases}$$

Define $N_+ = \{i \mid i \in N, y_i = 1\}$ and $S_+ = \sum_{i \in N_+}\left(1 - \frac{f_i}{\gamma} - F_i\right), N_- = \{i \mid i \in N, y_i = -1\}$ and $S_- = \sum_{i \in N_-}\left(-1 + \frac{f_i}{\gamma} - F_i\right), N'_+ = \{i \mid i \in N', y_i = 1\}$ and $S'_+ = \sum_{i \in N'_+}\left(1 - F'_i\right), N'_- = \{i \mid i \in N', y_i = -1\}$ and $S'_- = \sum_{i \in N'_-}\left(-1 - F'_i\right)$. Solving the four equations gives two possible answers: $b = \frac{1}{2}\left(\frac{S_+}{|N_+|} + \frac{S_-}{|N_-|}\right)$ and $b = \frac{1}{2}\left(\frac{S'_+}{|N'_+|} + \frac{S'_-}{|N'_-|}\right)$. The following average for the offset parameter was used:

$$\begin{aligned} b = \Bigg[& \frac{|N|}{|N| + |N'|}\left(\frac{S_+}{|N_+|} + \frac{S_-}{|N_-|}\right) \\ & + \frac{|N'|}{|N| + |N'|}\left(\frac{S'_+}{|N'_+|} + \frac{S'_-}{|N'_-|}\right)\Bigg] \end{aligned}$$

APPENDIX D

Mixture Model

Let θ be a $(n+m)$-variable vector as the concatenation of the α and β variables: $\theta \triangleq (\alpha, \beta)^T$.

Feasible Directions and the Clipping Function

For the cost function in equation (5.6) and its constraints, the sets of feasible directions are as follows:

Direction 1: $I_1 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2)\,, n+1 \leq s_1, s_2 \leq n+m, s_1 \neq s_2; u_{s_1} = 1, u_{s_2} = -1, \theta_{s_2} > 0, \forall i \notin \mathbf{s} u_i = 0\}\,.$

Direction 2: $I_2 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2)\,, 1 \leq s_1, s_2 \leq m, s_1 \neq s_2, y_{s_1} = y_{s_2}; u_{s_1} = 1, \theta_{s_1} < \rho C^*, u_{s_2} = -1, \theta_{s_2} > 0, \forall i \notin \mathbf{s} u_i = 0\}\,.$

Direction 3: $I_3 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2)\,, m+1 \leq s_1, s_2 \leq n,$ $s_1 \neq s_2, y_{s_1} = y_{s_2}; u_{s_1} = 1, \theta_{s_1} < C, u_{s_2} = -1, \theta_{s_2} > 0,$ $\forall i \notin \mathbf{s} u_i = 0\}\,.$

Direction 4: $I_4 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2)\,, m+1 \leq s_1, s_2 \leq n,$ $s_1 \neq s_2, y_{s_1} \neq y_{s_2}, \forall i \notin \mathbf{s} u_i = 0; u_{s_1} = u_{s_2} = 1, \theta_{s_1} < C,$ $\theta_{s_2} < C$ or $u_{s_1} = u_{s_2} = -1, \theta_{s_1} > 0, \theta_{s_2} > 0\}\,.$

Direction 5: $I_5 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3)\,, 1 \le s_1, s_2 \le m,$

$n + 1 \le s_3 \le n + m, s_1 \ne s_2, y_{s_1} \ne y_{s_2}, \forall i \notin \mathbf{s} u_i = 0;$

$u_{s_1} = u_{s_2} = 1, \theta_{s_1} < \rho C^*, \theta_{s_2} < \rho C^*, u_{s_3} = -2, \theta_{s_3} > 0$

or $u_{s_1} = u_{s_2} = -1, \theta_{s_1} > 0, \theta_{s_2} > 0, u_{s_3} = 2\}\,.$

Direction 6: $I_6 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3)\,, 1 \le s_1 \le m,$

$m + 1 \le s_2 \le n, n + 1 \le s_3 \le n + m, y_{s_1} = y_{s_2}, \forall i \notin \mathbf{s} u_i = 0;$

$u_{s_1} = 1, \theta_{s_1} < \rho C^*, u_{s_2} = -1, \theta_{s_2} > 0, u_{s_3} = -1, \theta_{s_3} > 0$

or $u_{s_1} = -1, \theta_{s_1} > 0, u_{s_2} = 1, \theta_{s_2} < C, u_{s_3} = 1\}\,.$

Direction 7: $I_7 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3)\,, 1 \le s_1 \le m,$

$m + 1 \le s_2 \le n, n + 1 \le s_3 \le n + m, y_{s_1} \ne y_{s_2}, \forall i \notin \mathbf{s} u_i = 0;$

$u_{s_1} = u_{s_2} = 1, \theta_{s_1} < \rho C^*, \theta_{s_2} < C, u_{s_3} = -1, \theta_{s_3} > 0$

or $u_{s_1} = u_{s_2} = -1, \theta_{s_1} > 0, \theta_{s_2} > 0, u_{s_3} = 1\}\,.$

Direction 8: $I_8 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3)\,, 1 \le s_1, s_2 \le m,$

$m + 1 \le s_3 \le n, s_1 \ne s_2, y_{s_1} \ne y_{s_2}, y_{s_3} = y_{s_2}, \forall i \notin \mathbf{s} u_i = 0;$

$u_{s_1} = 1, \theta_{s_1} < \rho C^*, u_{s_2} = -1, \theta_{s_2} > 0, u_{s_3} = 2, \theta_{s_3} < C$

or $u_{s_1} = -1, \theta_{s_1} > 0, u_{s_2} = 1, \theta_{s_2} < \rho C^*, u_{s_3} = -2, \theta_{s_3} > 0\}.$

Direction 9: $I_9 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3)\,, 1 \le s_1, s_2 \le m,$

$m + 1 \le s_3 \le n, s_1 \ne s_2, y_{s_1} \ne y_{s_2}, y_{s_3} = y_{s_1}, \forall i \notin \mathbf{s} u_i = 0;$

$u_{s_1} = 1, \theta_{s_1} < \rho C^*, u_{s_2} = -1, \theta_{s_2} > 0, u_{s_3} = -2, \theta_{s_3} > 0$

or $u_{s_1} = -1, \theta_{s_1} > 0, u_{s_2} = 1, \theta_{s_2} < \rho C^*, u_{s_3} = 2, \theta_{s_3} < C\}.$

It can be verified that when moving from any feasible point θ^{old} in the direction of $\mathbf{u_s} \in \cup I_i$ and applying the clipping function of equation (30), the constraints corresponding to dual problems are satisfied.

Offset Parameter of the Decision Function

In order to calculate the offset parameter b of the decision function, suppose α and β are the solution of the SVMp+ dual problem (10). Define the two sets $N \triangleq \{i \mid 1 \le i \le m, 0 < \ \alpha_i < \rho C^*\}$ and $N' \triangleq \left\{i \mid m + 1 \le i \le n, \bar{0} < \bar{\alpha}_i < C\right\}$.

The rest of the calculations are similar to the previous case in Appendix C.

APPENDIX E

Symmetric Mixture Model

Let θ be a $(n+m)$-variable vector as the concatenation of the α and β variables: $\theta \triangleq (\alpha, \beta)^T$.

Feasible Directions and the Clipping Function

Using the cost function in equation (5.9), the sets of feasible directions are:

Direction 1: $I_1 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), n+1 \leq s_1, s_2 \leq n+m, s_1 \neq s_2, y_{s_1-n} = y_{s_2-n}; u_{s_1} = 1, u_{s_2} = -1, \theta_{s_2} > 0, \forall i \notin \mathbf{s} u_i = 0\}.$

Direction 2: $I_2 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), n+1 \leq s_1, s_2 \leq n+m, s_1 \neq s_2, y_{s_1-n} \neq y_{s_2-n}, \forall i \notin \mathbf{s} u_i = 0; u_{s_1} = u_{s_2} = 1 \text{ or } u_{s_1} = u_{s_2} = -1, \theta_{s_1} > 0, \theta_{s_2} > 0\}.$

Direction 3: $I_3 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), 1 \leq s_1, s_2 \leq m, s_1 \neq s_2, y_{s_1} = y_{s_2}; u_{s_1} = 1, \theta_{s_1} < \rho C^*, u_{s_2} = -1, \theta_{s_2} > 0, \forall i \notin \mathbf{s} u_i = 0\}.$

Direction 4: $I_4 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), 1 \leq s_1, s_2 \leq m, s_1 \neq s_2, y_{s_1} \neq y_{s_2}, \forall i \notin \mathbf{s} u_i = 0; u_{s_1} = u_{s_2} = 1, \theta_{s_1} < \rho C^*, \theta_{s_2} < \rho C^* \text{ or } u_{s_1} = u_{s_2} = -1, \theta_{s_1} > 0, \theta_{s_2} > 0\}$

Direction 5: $I_5 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), m+1 \leq s_1, s_2 \leq n,$
$s_1 \neq s_2, y_{s_1} = y_{s_2}; u_{s_1} = 1, \theta_{s_1} < C, u_{s_2} = -1, \theta_{s_2} > 0,$
$\forall i \notin \mathbf{s} u_i = 0\}.$

Direction 6: $I_6 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), m+1 \leq s_1, s_2 \leq n,$
$s_1 \neq s_2, y_{s_1} \neq y_{s_2}, \forall i \notin \mathbf{s} u_i = 0; u_{s_1} = u_{s_2} = 1, \theta_{s_1} < C,$
$\theta_{s_2} < C \quad \text{or} \quad u_{s_1} = u_{s_2} = -1, \theta_{s_1} > 0, \theta_{s_2} > 0\}.$

Direction 7: $I_7 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \leq s_1 \leq m,$
$m+1 \leq s_2 \leq n, n+1 \leq s_3 \leq n+m, y_{s_1} = y_{s_2} = y_{s_3} - n,$
$\forall i \notin \mathbf{s} u_i = 0; u_{s_1} = -1, \theta_{s_1} > 0, u_{s_2} = 1, \theta_{s_2} < C, u_{s_3} = 1$
or $u_{s_1} = 1, \theta_{s_1} < \rho C^*, u_{s_2} = -1, \theta_{s_2} > 0, u_{s_3} = -1, \theta_{s_3} > 0\}.$

Direction 8: $I_8 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \leq s_1 \leq m,$
$m+1 \leq s_2 \leq n, n+1 \leq s_3 \leq n+m, y_{s_1} = y_{s_2}, y_{s_3-n} \neq y_{s_1},$
$\forall i \notin \mathbf{s} u_i = 0; u_{s_1} = -1, \theta_{s_1} > 0, u_{s_2} = 1, \theta_{s_2} < C,$
$u_{s_3} = -1, \theta_{s_3} > 0$ or $u_{s_1} = 1, \theta_{s_1} < \rho C^*, u_{s_2} = -1,$
$\theta_{s_2} > 0, u_{s_3} = 1\}.$

Direction 9: $I_9 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \leq s_1 \leq m,$
$m+1 \leq s_2 \leq n, n+1 \leq s_3 \leq n+m, y_{s_1} \neq y_{s_2}, y_{s_3-n} = y_{s_1},$
$\forall i \notin \mathbf{s} u_i = 0; u_{s_1} = 1, \theta_{s_1} < \rho C^*, u_{s_2} = 1, \theta_{s_2} < C,$
$u_{s_3} = -1, \theta_{s_3} > 0$ or $u_{s_1} = -1, \theta_{s_1} > 0, u_{s_2} = -1,$
$\theta_{s_2} > 0, u_{s_3} = 1\}.$

Direction 10: $I_{10} \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \leq s_1 \leq m,$
$m+1 \leq s_2 \leq n, n+1 \leq s_3 \leq n+m, y_{s_1} \neq y_{s_2}, y_{s_3-n} = y_{s_2},$
$\forall i \notin$ s $u_i = 0; u_{s_1} = 1, \theta_{s_1} < \rho C^*, u_{s_2} = 1, \theta_{s_2} < C, u_{s_3} = 1$
or $u_{s_1} = -1, \theta_{s_1} > 0, u_{s_2} = -1, \theta_{s_2} > 0, u_{s_3} = -1, \theta_{s_3} > 0\}.$

As described in the Chapter V, after determining the best feasible direction and the corresponding step size, the clipping function of equation (E.1)ensures that the boundary conditions of the dual form are satisfied.

$$\lambda^{*}\left(\boldsymbol{\theta}^{\text{old}},\mathbf{s}^{*(i)}\right)=\begin{cases}\min\left\{\rho C^{*}-\theta_k^{\text{old}}\,,\min_{j\in\mathbf{s}^{*}(i)}\left(\lambda'\left(\boldsymbol{\theta}^{\text{old}}\,,\mathbf{s}^{*(i)}\right),\left|\frac{\theta_j^{\text{old}}}{u_j}\right|\right)\right\} & 1\le k\le m, k\in\mathbf{s}^{*(i)}, u_k>0\\ \min\left\{C\pi_k-\theta_k^{\text{old}}\,,\min_{j\in\mathbf{s}^{*}(i)}\left(\lambda'\left(\boldsymbol{\theta}^{\text{old}}\,,\mathbf{s}^{*(i)}\right),\left|\frac{\theta_j^{\text{old}}}{u_j}\right|\right)\right\} & m+1\le k\le n, k\in\mathbf{s}^{*(i)}, u_k>0\end{cases}\tag{E.1}$$

Offset Parameter of the Decision Function

If equation (5.9) is considered, the two sets $N \triangleq \{i \mid 1 \le i \le m, 0 < \alpha_i < \rho C^*\}$ and $N' \triangleq \{i \mid m+1 \le i \le n, 0 < \alpha_i < C\}$ are defined. By the conditions for equation (5.8), for the support vectors, the KKT conditions state:

$$\forall i \in N \quad y_i\left(\mathbf{w}\cdot\mathbf{z}_i + b\right) = 1 - y_i\left(\mathbf{w}^*\cdot\mathbf{z}_i^* + b^*\right)$$

$$\forall i \in N' \quad y_i\left(\mathbf{w}\cdot\mathbf{z}_i + b\right) = 1$$

Define (note that f_i is not the same as the previous cases in Appendices C and D):

$$F_i \triangleq \mathbf{w}\cdot\mathbf{z}_i\Big|_{i\in N} = \sum_{j=1}^{n} y_j\alpha_j K_{ij}\Bigg|_{i\in N}$$

$$F_i' \triangleq \mathbf{w}\cdot\mathbf{z}_i\Big|_{i\in N'} = \sum_{j=1}^{n} y_j\alpha_j K_{ij}\Bigg|_{i\in N'}$$

$$f_i \triangleq \gamma\mathbf{w}^*\cdot\mathbf{z}_i^*\Big|_{i\in N} = \sum_{j=1}^{m}\left(\alpha_j+\beta_j-C^*\right)y_j K_{ij}^*\Bigg|_{i\in N}$$

Note that not only is f_i not the same as previous cases, but the second equation has also been changed. Define $N_+ = \{i \mid i \in N, y_i = 1\}$ and $S_+ = \sum_{i\in N_+}\left(1 - \frac{f_i}{\gamma} - F_i\right)$, $N_- = \{i \mid i \in N, y_i = -1\}$ and $S_- = \sum_{i\in N_-}\left(-1 - \frac{f_i}{\gamma} - F_i\right)$ $N_+' = \{i \mid i \in N', y_i = 1\}$ and $S_+' = \sum_{i\in N_+'}\left(1 - F_i'\right)$ $N_-' = \{i \mid i \in N', y_i = -1\}$ and $S_-' = \sum_{i\in N_-'}\left(-1 - F_i'\right)$ Solving the four equations gives two possible answers: $b = \frac{1}{2}\left(\frac{S_+}{|N_+|} + \frac{S_-}{|N_-|}\right)$ and $b = \frac{1}{2}\left(\frac{S_+'}{|N_+'|} + \frac{S_-'}{|N_-'|}\right)$. The following average for the offset parameter was used:

$$\begin{aligned}b = \Bigg[&\frac{|N|}{|N|+|N'|}\left(\frac{S_+}{|N_+|} + \frac{S_-}{|N_-|}\right)\\ &+\frac{|N'|}{|N|+|N'|}\left(\frac{S_+'}{|N_+'|} + \frac{S_-'}{|N_-'|}\right)\Bigg]\end{aligned}$$

APPENDIX F

LULUPAPI Mixture Model

Let θ be a $(n+m)$ -variable vector as the concatenation of the α and β variables: $\theta \triangleq (\alpha, \beta)^T$.

Feasible Directions and the Clipping Function

For the cost function in equation (5.11) and its constraints, the sets of feasible directions are as follows:

Direction 1: $I_1 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), n+1 \le s_1,$
$s_2 \le n+m, s_1 \ne s_2; u_{s_1} = 1, u_{s_2} = -1, \theta_{s_2} > 0, \forall i \notin \mathbf{s} u_i = 0\}.$

Direction 2: $I_2 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), 1 \le s_1, s_2 \le m, s_1 \ne s_2,$
$y_{s_1} = y_{s_2}; u_{s_1} = 1, \theta_{s_1} < \rho C^* \pi_{s_1}, u_{s_2} = -1, \theta_{s_2} > 0, \forall i \notin \mathbf{s} u_i = 0\}.$

Direction 3: $I_3 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), m+1 \le s_1, s_2 \le n,$
$s_1 \ne s_2, y_{s_1} = y_{s_2}; u_{s_1} = 1, \theta_{s_1} < C\pi_{s_1}, u_{s_2} = -1, \theta_{s_2} > 0, \forall i \notin \mathbf{s} u_i = 0\}.$

Direction 4: $I_4 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2), m+1 \le s_1, s_2 \le n,$
$s_1 \ne s_2, y_{s_1} \ne y_{s_2}, \forall i \notin \mathbf{s} u_i = 0; u_{s_1} = u_{s_2} = 1, \theta_{s_1} < C\pi_{s_1},$
$\theta_{s_2} < C\pi_{s_2}$ or $u_{s_1} = u_{s_2} = -1, \theta_{s_1} > 0, \theta_{s_2} > 0\}.$

Direction 5: $I_5 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \leq s_1, s_2 \leq m,$
$n+1 \leq s_3 \leq n+m, s_1 \neq s_2, y_{s_1} \neq y_{s_2}, \forall i \notin \mathbf{s} u_i = 0;$
$u_{s_1} = u_{s_2} = 1, \theta_{s_1} < \rho C^* \pi_{s_1}, \theta_{s_2} < \rho C^* \pi_{s_2}, u_{s_3} = -2,$
$\theta_{s_3} > 0 \quad \text{or} \quad u_{s_1} = u_{s_2} = -1, \theta_{s_1} > 0, \theta_{s_2} > 0, u_{s_3} = 2\}$

Direction 6: $I_6 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \leq s_1 \leq m,$
$m+1 \leq s_2 \leq n, n+1 \leq s_3 \leq n+m, y_{s_1} = y_{s_2}, \forall i \notin \mathbf{s} u_i = 0;$
$u_{s_1} = 1, \theta_{s_1} < \rho C^* \pi_{s_1}, u_{s_2} = -1, \theta_{s_2} > 0, u_{s_3} = -1,$
$\theta_{s_3} > 0$ or $u_{s_1} = -1, \theta_{s_1} > 0, u_{s_2} = 1, \theta_{s_2} < C\pi_{s_2}, u_{s_3} = 1\}.$

Direction 7: $I_7 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \leq s_1 \leq m,$
$m+1 \leq s_2 \leq n, n+1 \leq s_3 \leq n+m, y_{s_1} \neq y_{s_2}, \forall i \notin$
$\mathbf{s} u_i = 0; u_{s_1} = u_{s_2} = 1, \theta_{s_1} < \rho C^* \pi_{s_1}, \theta_{s_2} < C\pi_{s_2}, u_{s_3} = -1,$
$\theta_{s_3} > 0$ or $u_{s_1} = u_{s_2} = -1, \theta_{s_1} > 0, \theta_{s_2} > 0, u_{s_3} = 1\}.$

Direction 8: $I_8 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \leq s_1, s_2 \leq m,$
$m+1 \leq s_3 \leq n, s_1 \neq s_2, y_{s_1} \neq y_{s_2}, y_{s_3} = y_{s_2}, \forall i \notin \mathbf{s} u_i = 0;$
$u_{s_1} = 1, \theta_{s_1} < \rho C^* \pi_{s_1}, u_{s_2} = -1, \theta_{s_2} > 0, u_{s_3} = 2, \theta_{s_3} < C\pi_{s_3}$
or $u_{s_1} = -1, \theta_{s_1} > 0, u_{s_2} = 1, \theta_{s_2} < \rho C^* \pi_{s_2}, u_{s_3} = -2, \theta_{s_3} > 0\}.$

Direction 9: $I_9 \triangleq \{\mathbf{u_s} \mid \mathbf{s} = (s_1, s_2, s_3), 1 \leq s_1, s_2 \leq m,$
$m+1 \leq s_3 \leq n, s_1 \neq s_2, y_{s_1} \neq y_{s_2}, y_{s_3} = y_{s_1}, \forall i \notin \mathbf{s} u_i = 0;$
$u_{s_1} = 1, \theta_{s_1} < \rho C^* \pi_{s_1}, u_{s_2} = -1, \theta_{s_2} > 0, u_{s_3} = -2, \theta_{s_3} > 0$
or $u_{s_1} = -1, \theta_{s_1} > 0, u_{s_2} = 1, \theta_{s_2} < \rho C^* \pi_{s_2}, u_{s_3} = 2, \theta_{s_3} < C\pi_{s_3}\}.$

It can be verified that when moving from any feasible point θ^{old} in the direction of $\mathbf{u_s} \in \cup I_i$ and applying the clipping function of equation (F.1), the constraints corresponding to dual problems are satisfied.

$$\lambda^*\left(\boldsymbol{\theta}^{\text{old}}, \mathbf{s}^{*(i)}\right) = \begin{cases} \min\left\{\frac{\rho\pi_k C^* - \theta_k^{\text{old}}}{u_k}, \min_{j \in \mathbf{s}^{*}(i)} \left(\lambda'\left(\boldsymbol{\theta}^{\text{old}}, \mathbf{s}^{*(i)}\right), \left|\frac{\theta_j^{\text{old}}}{u_j}\right|\right)\right\} & 1 \leq k \leq m, k \in \mathbf{s}^{*(i)}, u_k > 0 \\ \min\left\{\frac{C\pi_k - \theta_k^{\text{old}}}{u_k}, \min_{j \in \mathbf{s}^{*}(i)} \left(\lambda'\left(\boldsymbol{\theta}^{\text{old}}, \mathbf{s}^{*(i)}\right), \left|\frac{\theta_j^{\text{old}}}{u_j}\right|\right)\right\} & m+1 \leq k \leq n, k \in \mathbf{s}^{*(i)}, u_k > 0 \end{cases} \tag{F.1}$$

Offset Parameter of the Decision Function

In order to calculate the offset parameter b of the decision function, suppose α and β are the solution of the SVMp+ dual problem (23). Define two sets $N \triangleq \{i \mid 1 \leq i \leq m, 0 < \alpha_i < \rho\pi_i C^*\}$ and $N' \triangleq \{i \mid m+1 \leq i \leq n, 0 < \alpha_i < C\pi_i\}$ The rest of the calculations are then similar to the previous case in Appendix C.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Paul Christopher Webster. Electronic health records a “strong priority” for us government. *CMAJ*, 182(8):E315–E316, 2010.

[2] Ashish K Jha, Catherine M DesRoches, Eric G Campbell, Karen Donelan, Sowmya R Rao, Timothy G Ferris, Alexandra Shields, Sara Rosenbaum, and David Blumenthal. Use of electronic health records in us hospitals. *New England Journal of Medicine*, 360(16):1628–1638, 2009.

[3] Engineering National Academies of Sciences, Medicine, et al. *Improving diagnosis in health care*. National Academies Press, 2015.

[4] ARDS Definition Task Force, VM Ranieri, GD Rubenfeld, BT Thompson, ND Ferguson, E Caldwell, et al. Acute respiratory distress syndrome. *Jama*, 307(23):2526–2533, 2012.

[5] Sujal R Desai. Acute respiratory distress syndrome: imaging of the injured lung. *Clinical radiology*, 57(1):8–17, 2002.

[6] Leonard D Hudson, John A Milberg, Doreen Anardi, and Richard J Maunder. Clinical risks for development of the acute respiratory distress syndrome. *American journal of respiratory and critical care medicine*, 151(2):293–301, 1995.

[7] Owe R Luhr, Kristian Antonsen, Magnus Karlsson, Sidsel Aardal, Adalbjorn Thorsteinsson, CLAES G FROSTELL, JaN Bonde, and ARF Study Group. Incidence and mortality after acute respiratory failure and acute respiratory distress syndrome in sweden, denmark, and iceland. *American journal of respiratory and critical care medicine*, 159(6):1849–1861, 1999.

[8] Gordon D Rubenfeld, Ellen Caldwell, John Granton, Leonard D Hudson, and Michael A Matthay. Interobserver variability in applying a radiographic definition for ards. *Chest*, 116(5):1347–1353, 1999.

[9] Reginald Greene. Adult respiratory distress syndrome: acute alveolar damage. *Radiology*, 163(1):57–66, 1987.

[10] EN Milne, Massimo Pistolesi, Massimo Miniati, and Carlo Giuntini. The radiologic distinction of cardiogenic and noncardiogenic edema. *American journal of roentgenology*, 144(5):879–894, 1985.

[11] Giacomo Bellani, John G Laffey, Tài Pham, Eddy Fan, Laurent Brochard, Andres Esteban, Luciano Gattinoni, Frank Van Haren, Anders Larsson, Daniel F McAuley, et al. Epidemiology, patterns of care, and mortality for patients with acute respiratory distress syndrome in intensive care units in 50 countries. *Jama*, 315(8):788–800, 2016.

[12] Gordon D Rubenfeld, Ellen Caldwell, Eve Peabody, Jim Weaver, Diane P Martin, Margaret Neff, Eric J Stern, and Leonard D Hudson. Incidence and outcomes of acute lung injury. *New England Journal of Medicine*, 353(16):1685–1693, 2005.

[13] Jennifer N Ervin, Victor C Rentes, Emily R Dibble, Michael W Sjoding, Theodore J Iwashyna, Catherine L Hough, Michelle Ng Gong, and Anne E Sales. Evidence-based practices for acute respiratory failure and acute respiratory distress syndrome: A systematic review of reviews. *Chest*, 158(6):2381–2393, 2020.

[14] Michael W Sjoding, Timothy P Hofer, Ivan Co, Jakob I McSparron, and Theodore J Iwashyna. Differences between patients in whom physicians agree and disagree about the diagnosis of acute respiratory distress syndrome. *Annals of the American Thoracic Society*, 16(2):258–264, 2019.

[15] Giacomo Bellani, Tài Pham, and John G Laffey. Missed or delayed diagnosis of ards: a common and serious problem. *Intensive care medicine*, 46(6):1180–1183, 2020.

[16] Ruyang Zhang, Zhaoxi Wang, Paula Tejera, Angela J Frank, Yongyue Wei, Li Su, Zhaozhong Zhu, Yichen Guo, Feng Chen, Ednan K Bajwa, et al. Late-onset moderate to severe acute respiratory distress syndrome is associated with shorter survival and higher mortality: a two-stage association study. *Intensive care medicine*, 43(3):399–407, 2017.

[17] Rob Mac Sweeney and Daniel F McAuley. Acute respiratory distress syndrome. *The Lancet*, 388(10058):2416–2430, 2016.

[18] Brendan J Clark and Marc Moss. The acute respiratory distress syndrome: Dialing in the evidence? *Jama*, 315(8):759–761, 2016.

[19] Michael W Sjoding and Robert C Hyzy. Recognition and appropriate treatment of the acute respiratory distress syndrome remains unacceptably low. *Critical care medicine*, 44(8):1611, 2016.

[20] Michael W Sjoding. Translating evidence into practice in acute respiratory distress syndrome: teamwork, clinical decision support, and behavioral economic interventions. *Current opinion in critical care*, 23(5):406–411, 2017.

[21] Vitaly Herasevich, Murat Yilmaz, Hasrat Khan, Rolf D Hubmayr, and Ognjen Gajic. Validation of an electronic surveillance system for acute lung injury. *Intensive care medicine*, 35(6):1018–1023, 2009.

[22] Helen C Koenig, Barbara B Finkel, Satjeet S Khalsa, Paul N Lanken, Meeta Prasad, Richard Urbani, and Barry D Fuchs. Performance of an automated electronic acute lung injury screening system in intensive care unit patients. *Critical care medicine*, 39(1):98–104, 2011.

[23] David F Nettleton, Albert Orriols-Puig, and Albert Fornells. A study of the effect of different types of noise on the precision of supervised learning techniques. *Artificial intelligence review*, 33(4):275–306, 2010.

[24] John G Laffey, Tài Pham, and Giacomo Bellani. Continued under-recognition of acute respiratory distress syndrome after the berlin definition: what is the solution? *Current opinion in critical care*, 23(1):10–17, 2017.

[25] Qiao Li and Gari D Clifford. Signal quality and data fusion for false alarm reduction in the intensive care unit. *Journal of electrocardiology*, 45(6):596–603, 2012.

[26] Ikaro Silva, Joon Lee, and Roger G Mark. Signal quality estimation with multichannel adaptive filtering in intensive care settings. *IEEE Transactions on Biomedical Engineering*, 59(9):2476–2485, 2012.

[27] TL Rusch, R Sankar, and JE Scharf. Signal processing methods for pulse oximetry. *Computers in biology and medicine*, 26(2):143–159, 1996.

[28] John Shawe-Taylor, Peter L Bartlett, Robert C Williamson, and Martin Anthony. Structural risk minimization over data-dependent hierarchies. *IEEE transactions on Information Theory*, 44(5):1926–1940, 1998.

[29] Riccardo Bellazzi, Alberto Riva, et al. Learning conditional probabilities with longitudinal data. In *Working Notes of the IJCAI Workshop Building Probabilistic Networks: Where Do the Numbers Come from*, pages 7–15, 1995.

[30] K Najarian, Guy A Dumont, Michael S Davies, and NE Heckman. Pac learning in non-linear fir models. *International Journal of Adaptive Control and Signal Processing*, 15(1):37–52, 2001.

[31] Vladimir N Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999.

[32] Carlo Berzuini, Nicola G Best, Walter R Gilks, and Cristiana Larizza. Dynamic conditional independence models and markov chain monte carlo methods. *Journal of the American Statistical Association*, 92(440):1403–1412, 1997.

[33] Xiang Xuan and Kevin Murphy. Modeling changing dependency structure in multivariate time series. In *Proceedings of the 24th international conference on Machine learning*, pages 1055–1062, 2007.

[34] Danyu Y Lin and Lee-Jen Wei. The robust inference for the cox proportional hazards model. *Journal of the American statistical Association*, 84(408):1074–1078, 1989.

[35] Ralf Bender, Thomas Augustin, and Maria Blettner. Generating survival times to simulate cox proportional hazards models. *Statistics in medicine*, 24(11):1713–1723, 2005.

[36] Drausin Wulsin, Emily Fox, and Brian Litt. Parsing epileptic events using a markov switching process model for correlated time series. In *International Conference on Machine Learning*, pages 356–364, 2013.

[37] Guoliang Fan and Hanying Liang. Empirical likelihood for longitudinal partially linear model with α-mixing errors. *Journal of Systems Science and Complexity*, 26(2):232–248, 2013.

[38] Lei Yu and Huan Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pages 856–863, 2003.

[39] Kyoung-jae Kim. Financial time series forecasting using support vector machines. *Neurocomputing*, 55(1-2):307–319, 2003.

[40] Alex Waibel. Modular construction of time-delay neural networks for speech recognition. *Neural computation*, 1(1):39–46, 1989.

[41] Donald J Berndt and James Clifford. Using dynamic time warping to find patterns in time series. In *KDD workshop*, volume 10, pages 359–370. Seattle, WA, USA:, 1994.

[42] Michael J Kastoryano and Jens Eisert. Rapid mixing implies exponential decay of correlations. *Journal of Mathematical Physics*, 54(10):102201, 2013.

[43] Mathukumalli Vidyasagar. *Learning and generalisation: with applications to neural networks.* Springer Science & Business Media, 2013.

[44] Geert Verbeke. Linear mixed models for longitudinal data. In *Linear mixed models in practice*, pages 63–153. Springer, 1997.

[45] Le Nguyen Binh. Advanced digital: Optical communications, 2015.

[46] Haibo He and Edwardo A Garcia. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284, 2009.

[47] J Abdul Sukor, SJ Redmond, and NH Lovell. Signal quality measures for pulse oximetry through waveform morphology analysis. *Physiological measurement*, 32(3):369, 2011.

[48] Qiao Li and Gari D Clifford. Dynamic time warping and machine learning for signal quality assessment of pulsatile signals. *Physiological measurement*, 33(9):1491, 2012.

[49] Thomas M Cover. *Elements of information theory.* John Wiley & Sons, 1999.

[50] Benoît Frénay, Ata Kabán, et al. A comprehensive introduction to label noise. In *ESANN*, 2014.

[51] Michael W Sjoding, Timothy P Hofer, Ivan Co, Anthony Courey, Colin R Cooke, and Theodore J Iwashyna. Interobserver reliability of the berlin ards definition and strategies to improve the reliability of ards diagnosis. *Chest*, 153(2):361–367, 2018.

[52] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[53] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[54] William A Knaus, Douglas P Wagner, Elizabeth A Draper, Jack E Zimmerman, Marilyn Bergner, Paulo G Bastos, Carl A Sirio, Donald J Murphy, Ted Lotring, Anne Damiano, et al. The apache iii prognostic system: risk prediction of hospital mortality for critically iii hospitalized adults. *Chest*, 100(6):1619–1636, 1991.

[55] Matthew M Churpek, Trevor C Yuen, Christopher Winslow, Ari A Robicsek, David O Meltzer, Robert D Gibbons, and Dana P Edelson. Multicenter development and validation of a risk stratification tool for ward patients. *American journal of respiratory and critical care medicine*, 190(6):649–655, 2014.

[56] Lorraine B Ware and Michael A Matthay. The acute respiratory distress syndrome. *New England Journal of Medicine*, 342(18):1334–1349, 2000.

[57] Richard C Bradley. Basic properties of strong mixing conditions. a survey and some open questions. *arXiv preprint math/0511078*, 2005.

[58] Allan J Walkey. Unreliable syndromes, unreliable studies, 2016.

[59] Sarah Sheard, Praveen Rao, and Anand Devaraj. Imaging of acute respiratory distress syndrome. *Respiratory care*, 57(4):607–612, 2012.

[60] Jin-Min Peng, Chuan-Yun Qian, Xiang-You Yu, Ming-Yan Zhao, Shu-Sheng Li, Xiao-Chun Ma, Yan Kang, Fa-Chun Zhou, Zhen-Yang He, Tie-He Qin, et al. Does training improve diagnostic accuracy and inter-rater agreement in applying the berlin radiographic definition of acute respiratory distress syndrome? a multicenter prospective study. *Critical Care*, 21(1):1–8, 2017.

[61] Ewa Pietka. Lung segmentation in digital radiographs. *Journal of digital imaging*, 7(2):79–84, 1994.

[62] Samuel G Armato III, Maryellen L Giger, and Heber MacMahon. Automated lung segmentation in digitized posteroanterior chest radiographs. *Academic radiology*, 5(4):245–255, 1998.

[63] Matthew S Brown, Laurence S Wilson, Bruce D Doust, Robert W Gill, and Changming Sun. Knowledge-based method for segmentation and analysis of lung boundaries in chest x-ray images. *Computerized medical imaging and graphics*, 22(6):463–477, 1998.

[64] Yonghong Shi, Feihu Qi, Zhong Xue, Liya Chen, Kyoko Ito, Hidenori Matsuo, and Dinggang Shen. Segmenting lung fields in serial chest radiographs using both population-based and patient-specific shape statistics. *IEEE Transactions on medical Imaging*, 27(4):481–494, 2008.

[65] Pavan Annangi, Sheshadri Thiruvenkadam, Anand Raja, Hao Xu, XiWen Sun, and Ling Mao. A region based active contour method for x-ray lung segmentation using prior shape and low level features. In *2010 IEEE international symposium on biomedical imaging: from nano to macro*, pages 892–895. IEEE, 2010.

[66] Ajay Mittal, Rahul Hooda, and Sanjeev Sofat. Lf-segnet: A fully convolutional encoder–decoder network for segmenting lung fields from chest radiographs. *Wireless Personal Communications*, 101(1):511–529, 2018.

[67] Chunliang Wang. Segmentation of multiple structures in chest radiographs using multi-task fully convolutional networks. In *Scandinavian Conference on Image Analysis*, pages 282–289. Springer, 2017.

[68] Tim B Hunter, Mihra S Taljanovic, Pei H Tsau, William G Berger, and James R Standen. Medical devices of the chest. *Radiographics*, 24(6):1725–1746, 2004.

[69] Chamith S Rajapakse and Gregory Chang. Impact of body habitus on radiologic interpretations. *Academic radiology*, 21(1):1–2, 2014.

[70] Hamed Behzadi-khormouji, Habib Rostami, Sana Salehi, Touba Derakhshande-Rishehri, Marzieh Masoumi, Siavash Salemi, Ahmad Keshavarz, Ali Gholamrezanezhad, Majid Assadi, and Ali Batouli. Deep learning, reusable and problem-based architectures for detection of consolidation on chest x-ray images. *Computer methods and programs in biomedicine*, 185:105162, 2020.

[71] Shiying Hu, Eric A Hoffman, and Joseph M Reinhardt. Automatic lung segmentation for accurate quantitation of volumetric x-ray ct images. *IEEE transactions on medical imaging*, 20(6):490–498, 2001.

[72] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[73] Jen Hong Tan, U Rajendra Acharya, Choo Min Lim, and K Thomas Abraham. An interactive lung field segmentation scheme with automated capability. *Digital Signal Processing*, 23(3):1022–1031, 2013.

[74] Jen Hong Tan and U Rajendra Acharya. Active spline model: A shape based model—interactive segmentation. *Digital signal processing*, 35:64–74, 2014.

[75] Karel Zuiderveld. Contrast limited adaptive histogram equalization. In *Graphics gems IV*, pages 474–485. Academic Press Professional, Inc., 1994.

[76] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992.

[77] Antonin Chambolle, Vicent Caselles, Daniel Cremers, Matteo Novaga, and Thomas Pock. An introduction to total variation for image analysis. *Theoretical foundations and numerical methods for sparse recovery*, 9(263-340):227, 2010.

[78] Mingqiang Zhu and Tony Chan. An efficient primal-dual hybrid gradient algorithm for total variation image restoration. *UCLA CAM Report*, 34, 2008.

[79] TW Ridler, S Calvard, et al. Picture thresholding using an iterative selection method. *IEEE trans syst Man Cybern*, 8(8):630–632, 1978.

[80] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988.

[81] François Chollet et al. keras, 2015.

[82] Th A Sorensen. A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on danish commons. *Biol. Skar.*, 5:1–34, 1948.

[83] Kelly H Zou, Simon K Warfield, Aditya Bharatha, Clare MC Tempany, Michael R Kaus, Steven J Haker, William M Wells III, Ferenc A Jolesz, and Ron Kikinis. Statistical validation of image segmentation quality based on a spatial overlap index1: scientific reports. *Academic radiology*, 11(2):178–189, 2004.

[84] Jerry L Hintze and Ray D Nelson. Violin plots: a box plot-density trace synergism. *The American Statistician*, 52(2):181–184, 1998.

[85] Junji Shiraishi, Shigehiko Katsuragawa, Junpei Ikezoe, Tsuneo Matsumoto, Takeshi Kobayashi, Ken-ichi Komatsu, Mitate Matsui, Hiroshi Fujita, Yoshie Kodera, and Kunio Doi. Development of a digital image database for chest radiographs with and without a lung nodule: receiver operating characteristic analysis of radiologists' detection of pulmonary nodules. *American Journal of Roentgenology*, 174(1):71–74, 2000.

[86] Bram Van Ginneken, Mikkel B Stegmann, and Marco Loog. Segmentation of anatomical structures in chest radiographs using supervised methods: a comparative study on a public database. *Medical image analysis*, 10(1):19–40, 2006.

[87] Stefan Jaeger, Sema Candemir, Sameer Antani, Yì-Xiáng J Wáng, Pu-Xuan Lu, and George Thoma. Two public chest x-ray datasets for computer-aided screening of pulmonary diseases. *Quantitative imaging in medicine and surgery*, 4(6):475, 2014.

[88] Jen Hong Tan, U Rajendra Acharya, Collin Tan, K Thomas Abraham, and Choo Min Lim. Computer-assisted diagnosis of tuberculosis: a first order statistical approach to chest radiograph. *Journal of medical systems*, 36(5):2751–2759, 2012.

[89] Tengku Afiah Mardhiah Tengku Zainul Akmal, Joel Chia Ming Than, Haslailee Abdullah, and Norliza Mohd Noor. Chest x-ray image classification on common thorax diseases using glcm and alexnet deep features. *International Journal of Integrated Engineering*, 11(4), 2019.

[90] Nitin S Lingayat and Manoj R Tarambale. A computer based feature extraction of lung nodule in chest x-ray image. *International Journal of Bioscience, Biochemistry and Bioinformatics*, 3(6):624, 2013.

[91] SA Patil. Texture analysis of tb x-ray images using image processing techniques. *Journal of Biomedical and Bioengineering*, 3(1):53–56, 2012.

[92] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 806–813, 2014.

[93] Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Y Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Proceedings of the 26th annual international conference on machine learning*, pages 609–616, 2009.

[94] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.

[95] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

[96] Joe Yue-Hei Ng, Fan Yang, and Larry S Davis. Exploiting local features from deep networks for image retrieval. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 53–61, 2015.

[97] Ali S Razavian, Josephine Sullivan, Stefan Carlsson, and Atsuto Maki. Visual instance retrieval with deep convolutional networks. *ITE Transactions on Media Technology and Applications*, 4(3):251–258, 2016.

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

[99] Dan C Cireşan, Alessandro Giusti, Luca M Gambardella, and Jürgen Schmidhuber. Mitosis detection in breast cancer histology images with deep neural networks. In *International conference on medical image computing and computer-assisted intervention*, pages 411–418. Springer, 2013.

[100] Yaniv Bar, Idit Diamant, Lior Wolf, and Hayit Greenspan. Deep learning with non-medical training used for chest pathology identification. In *Medical Imaging 2015: Computer-Aided Diagnosis*, volume 9414, page 94140V. International Society for Optics and Photonics, 2015.

[101] Bram Van Ginneken, BM Ter Haar Romeny, and Max A Viergever. Computer-aided diagnosis in chest radiography: a survey. *IEEE Transactions on medical imaging*, 20(12):1228–1241, 2001.

[102] Laurence Monnier-Cholley, Heber MacMahon, Shigehiko Katsuragawa, Junji Morishita, Takayuki Ishida, and Kunio Doi. Computer-aided diagnosis for detection of interstitial opacities on chest radiographs. *AJR. American journal of roentgenology*, 171(6):1651–1656, 1998.

[103] Robert M Haralick, Karthikeyan Shanmugam, and Its' Hak Dinstein. Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, (6):610–621, 1973.

[104] Aleksandr Zotin, Yousif Hamad, Konstantin Simonov, and Mikhail Kurako. Lung boundary detection for chest x-ray images classification based on glcm and probabilistic neural networks. *Procedia Computer Science*, 159:1439–1448, 2019.

[105] K Sudarshan Vidya, EYK Ng, U Rajendra Acharya, Siaw Meng Chou, Ru San Tan, and Dhanjoo N Ghista. Computer-aided diagnosis of myocardial infarction using ultrasound images with dwt, glcm and hos methods: a comparative study. *Computers in biology and medicine*, 62:86–93, 2015.

[106] SA Patil and VR Udupi. Geometrical and texture features estimation of lung cancer and tb images using chest x-ray database. *International Journal of Biomedical Engineering and Technology*, 6(1):58–75, 2011.

[107] Qian Zhao, Chang-Zheng Shi, and Liang-Ping Luo. Role of the texture features of images in the diagnosis of solitary pulmonary nodules in different sizes. *Chinese Journal of Cancer Research*, 26(4):451, 2014.

[108] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[109] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[110] Edmar Rezende, Guilherme Ruppert, Tiago Carvalho, Fabio Ramos, and Paulo De Geus. Malicious software classification using transfer learning of resnet-50 deep neural network. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1011–1014. IEEE, 2017.

[111] K Chughtai, Y Song, P Zhang, B Derstine, E Gatza, J Friedman, L Hully, C Inglis, S Goldstein, J Magenau, et al. Analytic morphomics: a novel ct imaging approach to quantify adipose tissue and muscle composition in allogeneic hematopoietic cell transplantation. *Bone Marrow Transplantation*, 51(3):446–450, 2016.

[112] Awais Mansoor, Ulas Bagci, Brent Foster, Ziyue Xu, Georgios Z Papadakis, Les R Folio, Jayaram K Udupa, and Daniel J Mollura. Segmentation and image analysis of abnormal lungs at ct: current approaches, challenges, and future trends. *RadioGraphics*, 35(4):1056–1076, 2015.

[113] Ingrid Sluimer, Mathias Prokop, and Bram Van Ginneken. Toward automated segmentation of the pathological lung in ct. *IEEE transactions on medical imaging*, 24(8):1025–1038, 2005.

[114] William A Goodrich. Pulmonary edema: A correlation of x-ray appearance and physiological chanqes. *Radiology*, 51(1):58–65, 1948.

[115] DENNIS Osborne. Radiologic appearance of viral disease of the lower respiratory tract in infants and children. *American Journal of Roentgenology*, 130(1):29–33, 1978.

[116] Claudia I Henschke, David F Yankelevitz, Austin Wand, Sheila D Davis, and Maria Shiau. Chest radiography in the icu. *Clinical imaging*, 21(2):90–103, 1997.

[117] Sema Candemir and Sameer Antani. A review on lung boundary detection in chest x-rays. *International journal of computer assisted radiology and surgery*, 14(4):563–576, 2019.

[118] Ulaş Bağcı, Mike Bray, Jesus Caban, Jianhua Yao, and Daniel J Mollura. Computer-assisted detection of infectious lung diseases: a review. *Computerized Medical Imaging and Graphics*, 36(1):72–84, 2012.

[119] Shigehiko Katsuragawa and Kunio Doi. Computer-aided diagnosis in chest radiography. *Computerized Medical Imaging and Graphics*, 31(4-5):212–223, 2007.

[120] Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural networks*, 22(5-6):544–557, 2009.

[121] Vladimir Vapnik and Rauf Izmailov. Learning using privileged information: similarity control and knowledge transfer. *J. Mach. Learn. Res.*, 16(1):2023–2049, 2015.

[122] Viktoriia Sharmanska, Novi Quadrianto, and Christoph H Lampert. Learning to rank using privileged information. In *Proceedings of the IEEE international conference on computer vision*, pages 825–832, 2013.

[123] Bernardete Ribeiro, Catarina Silva, Ning Chen, Armando Vieira, and Joao Carvalho das Neves. Enhanced default risk models with svm+. *Expert Systems with Applications*, 39(11):10140–10152, 2012.

[124] Lichen Liang and Vladimir Cherkassky. Connection between svm+ and multi-task learning. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pages 2048–2054. IEEE, 2008.

[125] Xue Li, Bo Du, Yipeng Zhang, Chang Xu, and Dacheng Tao. Iterative privileged learning. *IEEE transactions on neural networks and learning systems*, 31(8):2805–2817, 2019.

[126] John Platt. Sequential minimal optimization: A fast algorithm for training support vector machines. 1998.

[127] Dmitry Pechyony, Rauf Izmailov, Akshay Vashist, and Vladimir Vapnik. Smo-style algorithms for learning using privileged information. In *Dmin*, pages 235–241, 2010.

[128] Dmitry Pechyony and Vladimir Vapnik. Fast optimization algorithms for solving svm+. *Stat. Learning and Data Science*, 1, 2011.

[129] Thomas G Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, 10(7):1895–1923, 1998.

[130] A Trajman and RR Luiz. Mcnemar $\chi 2$ test revisited: comparing sensitivity and specificity of diagnostic examinations. *Scandinavian journal of clinical and laboratory investigation*, 68(1):77–80, 2008.

[131] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[132] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[133] Sara Gerke, Boris Babic, Theodoros Evgeniou, and I Glenn Cohen. The need for a system view to regulate artificial intelligence/machine learning-based software as medical device. *NPJ digital medicine*, 3(1):1–4, 2020.

[134] U.S. Food & Drug Admisntration. Artificial intelligence and machine learning in software as a medical device. *https://www.fda.gov/medical-devices/software-medical-device-samd/artificial-intelligence-and-machine-learning-software-medical-device*, 2021.

[135] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization.* Cambridge university press, 2004.