Learning to Rank Patients Severity of Illness with Partially Observable Data

Jianzhun Du, Yuanheng (Henry) Wang
John A. Paulson School of Engineering and Applied Sciences
Harvard University
{jzdu,yuanheng_wang}@g.harvard.edu

Abstract

Medical care resources are always scarce, especially in the settings of emergent and critical care. How to accurately evaluate and rank severity of illness of patients at any point of time during hospital stay and effectively allocate limited medical resources is a challenging task. Also, in many situations, practitioners have to make decisions when some critical physiological indices are not able to be measured in real time, i.e., only partial data is observable. In this paper, we formulate the ranking of severity of illness of ICU patients into *Maximum Coverage* problem with cardinality constraint, and leverage the framework of *Distributional Optimization from Samples* (DOPS) to solve this optimization problem. Our method achieves about 80 percent ranking accuracy when only 9 basic measurements are accessible, which indicates its potential application in healthcare domain.

1 Introduction

Intensive Unit Care (ICU) has been hugely costly for the people and government of United States. Just in 2005, cost of ICU was roughly $82 billion, which takes up 0.66% of the Gross Domestic Product (GDP) [Halpern and Pastores, 2010]. To reduce the cost, ICU resources will become limited. Hence, many efforts have been poured into effectively allocating the limited beds, medicine and medical staff, and there have been many discussions on the principles and fairness of these allocation tasks [Lanken *et al.*, 1997; Abu Al-Saad *et al.*, 2017]. A fundamental solution is to improve the efficacy of ICU treatment and shortened the average duration of stay. For example, certain hospitals changed their ICU patients' drug and care protocol and have seen cut of stay length and improved survivorship [Betbeze, 2018]. Deciding who should enter/exit ICU and how should a patient's treatment strategy change requires scrutiny and careful examination of patients' conditions.

In order to do this, several scoring systems for measuring patients severity of illness are developed [Rapsang and Shyam, 2014]. Without exception, all of them require certain physiological measurements to be collected and examined. Nevertheless, some lab results, such as lactate and creatinine, can take more than a couple hours to obtain and is impossible to measure in real time. This presents challenges for clinicians to make informative decisions since real-time conditions of patients cannot be accessed consistently.

In this paper, based on the principles of severity scoring system, we transform a part of demographic and physiological data of patients to discrete values with binary or multiple levels, and formulate severity evaluation into *Maximum Coverage* problem with cardinality constraint. With the spirit of *learning to rank* [Burges *et al.*, 2005], we utilize the framework of *Distributional Optimization from Samples* (DOPS) [Rosenfeld *et al.*, 2018] to optimize ranking of patient severity.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

2 Background and Related Work

2.1 Measuring Patient Severity Illness under Healthcare Settings

Machine learning has become increasingly popular in Healtcare and has been widely used in tasks such as patients mortality estimation and patients responsiveness to treatment prediction [Taylor *et al.*, 2016; Girkar *et al.*, 2018]. However, little work has been done to use machine learning to approximate patients severity with partially available information in order to decide how to most effectively allocate medical resources.

2.2 Learning to Rank

Learning to Rank (LtR) is a task to automatically construct a ranking model using training data, such that the model can sort new objects according to their degrees of relevance, preference, or importance. It's useful for many applications, such as document retrieval for search engines, sentiment analysis [Liu and others, 2009] and online advertising [Tagami *et al.*, 2013]. Recent healthcare work has suggested that LtR can improve search results of clinicians looking for relevant patients information [Alsulmi and Carterette, 2018]. Nevertheless, no work has been shown to rank the urgency of patients using LtR approach.

Ranking Support Vector Machine (Ranking SVM) [Herbrich *et al.*, 1999] is a powerful model for address LtR problem in information retrieval area. Ranking SVM states that we can learn a SVM for *classifying the order of pairs of objects* and utilize the classifier in the ranking task. In general, consider two sets of feature vectors $X^{(1)} = \{\mathbf{x}_1^{(1)}, \mathbf{x}_2^{(1)}, \dots, \mathbf{x}_m^{(1)}\}, X^{(2)} = \{\mathbf{x}_1^{(2)}, \mathbf{x}_2^{(2)}, \dots, \mathbf{x}_m^{(2)}\}$ and labels $Y = \{y_1, y_2, \dots, y_m\} \in \{+1, -1\}^m$ denoting feature vector should be ranked ahead, then the learning of Ranking SVM is formalized as the following *quadratic programming* problem:

$$\begin{aligned} \min_{\mathbf{w},\xi} \quad & \frac{1}{2}||\mathbf{w}||^2 + \lambda \sum_{i=1}^{m} \xi_i \\ \text{s.t. } & y_i\langle \mathbf{w}, \mathbf{x}_i^{(1)}\rangle - y_i\langle \mathbf{w}, \mathbf{x}_i^{(2)}\rangle + \xi_i \geq 1, \ \forall i \in [m] \\ & \xi_i \geq 0, \ \forall i \in [m] \end{aligned} \tag{1}$$

where $\langle\cdot,\cdot\rangle$ is the inner product of two vectors and $\mathbf{w}$ represents the linear classifier.

Structured Support Vector Machine (Structured SVM) [Tsochantaridis *et al.*, 2004] generalizes SVM classifier for structured output labels, which also have been used for ranking [Joachims, 2006; Mittal *et al.*, 2012]. In general, for a set of training instances $(X, Y) = \{\mathbf{x}_i, y_i\}_{i=1}^m \in \mathcal{X} \times \mathcal{Y}$, the structured SVM minimizes the following regularized loss function:

$$\begin{aligned} \min_{\mathbf{w},\xi} \quad & \frac{1}{2}||\mathbf{w}||^2 + \lambda \sum_{i=1}^{m} \xi_i \\ \text{s.t. } & \langle \mathbf{w}, \Phi(\mathbf{x}_i, y_i)\rangle - \langle \mathbf{w}, \Phi(\mathbf{x}_i, y)\rangle + \xi_i \geq \Delta(y_i, y), \ \forall i \in [m], \forall y \in \mathcal{Y} \\ & \xi_i \geq 0, \ \forall i \in [N] \end{aligned} \tag{2}$$

where $\Delta(\cdot,\cdot)$ is a function measuring a distance in label space and is an arbitrary function satisfying $\Delta(y, y') \geq 0$ and $\Delta(y, y) = 0, \forall y, z \in \mathcal{Y}$; $\Phi(\cdot,\cdot)$ is a feature function extracting some feature vector from a given sample and label.

2.3 Submodularity and Optimization from Samples

When independent variables are discrete quantities in machine learning, a structure for the model called *submodularity* naturally emerges. Formally, a function $f : 2^N \to \mathbb{R}$ defined over a ground set $N = \{e_1, e_2, \dots, e_n\}$ of elements is submodular if it demonstrates a diminishing marginal returns property, i.e., $f_S(e) \geq f_T(e)$ for all sets $S \subseteq T \subseteq N$ and element $e \notin T$ where $f_S(e) = f(S \cup e) - f(S)$ is the *marginal contribution* of element e to set $S \in N$. Thanks to theoretical guarantees on optimization based on this diminishing marginal returns property, submodular optimization is fast becoming a primary tool in machine learning, such as document summarization [Lin and Bilmes, 2011], optimal pricing in auctions [Maehara *et al.*, 2017], active learning [Golovin and Krause, 2011], sensor array design [Shulkind *et al.*, 2019].

However, in numerous cases, the loss function is submodular yet inaccessible. Instead, a surrogate function can be learned from data, with techniques such as PMAC [Balcan and Harvey, 2011] and *Optimization from Samples* (OPS) [Balkanski *et al.*, 2016]. Unfortunately, these techniques have been proven that for maximizing a submodular function under a cardinality constraint, no algorithm can obtain a constant factor approximation guarantee given polynomially-many samples from any distribution [Balkanski *et al.*, 2017]. Rosenfeld *et al.* [2018] propose a modified version of OPS, *Distributional Optimization from Samples* (DOPS), which provides reliable, efficient and scalable submodular optimization by learning from data within same distribution. In general, a function class $\mathcal{F}$ is in α-DOPS if an α-approximation of the empirical argmax can be found with arbitrarily high probability using polynomially many samples, for any distribution $\mathcal{D}$ and for any $f \in \mathcal{F}$. Formally:

Definition 1 *α-DOPS. Let $\mathcal{F} = \{f : 2^{[n]} \to \mathbb{R}^+\}$ be a class of set functions over n elements. We say that $\mathcal{F}$ is α-distributionally optimizable from samples if there is an algorithm $\mathcal{A}$ that, for every distribution $\mathcal{D}$ over $2^{[n]}$, every $f \in \mathcal{F}$, and every $\epsilon, \delta \in [0,1]$, when $\mathcal{A}$ is given as input a sample set $\mathcal{S} = \{(S^i, f(S^i))\}_{i=1}^M$ where $S^i \overset{iid}{\sim} \mathcal{D}$, with probability of at least $1-\delta$ over $\mathcal{S}$ it holds that*

$$\mathbb{P}_{\mathcal{T}\sim\mathcal{D}^m}\left[f(\mathcal{A}(\mathcal{T})) \geq \frac{1}{\alpha} \max_{S\in\mathcal{T}} f(S) \right] \geq 1-\epsilon \tag{3}$$

where $\mathcal{T} = \{(S^j)\}_{j=1}^m, \mathcal{A}(\mathcal{T}) \in \mathcal{T}$ is the output of the algorithm, and $\mathcal{S}$ is of size $M \in ploy(m, n, \frac{1}{\epsilon}, \frac{1}{\delta}, \alpha)$.

2.4 Maximum Coverage Problem

The maximum coverage problem is a classical question in computer science with widespread application [Dughmi and Vondrák, 2015; Sipos *et al.*, 2012], which is usually solved with approximation algorithm due to its NP-hardness. The corresponding coverage function is a simple, yet important and widely used class of *submodular* functions. We extend the classical Maximum Coverage problem to the *weighted version* formally:

Definition 2 *Weighted Maximum Coverage. Let $\mathcal{U}$ be a ground truth set containing d items with corresponding non-negative weights Θ and $\mathcal{C}$ be a collection of n subsets of $\mathcal{U}$, i.e., $\mathcal{U} = \{u_i\}_{i=1}^d, \Theta = \{\theta_i\}_{i=1}^d, \mathcal{C} = \{C_j\}_{j=1}^n$, where $\forall i \in [d], \theta_i \geq 0$ and $\forall j \in [n], C_j \subseteq \mathcal{U}$. Given S which is a subset of $\mathcal{C}$, a function $f_\theta : 2^{[n]} \to \mathbb{R}$ is a coverage function if:*

$$f_\Theta(S) = \sum_{u_i \in C(S)} \theta_i, \quad C(S) = \bigcup_{C_j \in S} C_j, \quad |S| \leq k \tag{4}$$

The objective is to find S with cardinality constraint such that $f_\theta(S)$ is maximized.

Equivalently, function 4 can be reformulated into:

$$\begin{aligned} f_\Theta(S) = \langle \Theta, \phi(S) \rangle, \quad \phi(S) = [t_1, t_2, \ldots, t_d]^T, \quad |S| \leq k \\ t_j = \mathbb{1}_{\{i \in S \text{ s.t. } u_j \in C_i\}}, \quad \theta_j \geq 0, \quad \forall j \in [d] \end{aligned} \tag{5}$$

3 Formulating to Rank Patients Severity of Illness

3.1 Motivation

Demographic data and physiological indices are innately connected. Readings of certain measurements can give us clues about results of related measurement. For example, low blood pressure and heart rate can indicate a low lactate level for patients undergoing septic shock. Obesity is also highly correlated with abnormal glucose level. Sometimes, when certain bio features cannot be observed, we can infer their values from observable yet correlated features. Hence, with the goal of approximating severity score, mutual correlations between different demographic data and physiological indices form "coverage" in our setting. Further, due to the difficulty of obtaining some measurements in real time, the partially observable physiological indices become "cardinality constriant" for coverage maximization. This is why we are motivated to fit the problem into maximum coverage framework.

3.2 Measurement Transformation

Targeting at Maximum Coverage framework, we have to properly convert our continuous measurements into discrete values. In order to do so, we consult domain experts and refer to official clinician guidelines. We attempted two types of conversion:

- *Two-level transformation*, i.e., converting measurements of patients into binary variables. An appearance of 1 indicates the measurement of the patient is abnormal, whereas a 0 means that it falls in the normal range.
- *Multi-level transformation*. In order to capture different levels of severity of the certain measurements, the data was one-hot encoded based on certain measurements of patients with multiple ranges. Considering a measurement with n levels, we construct a vector $\mathbf{v} = [l_1, l_2, \ldots, l_n]$. One of entries of $\mathbf{v}$ turns 1 according to the observed measurement, while the rest of entries remains 0. For instance, one of the most important demographic measurements, age, is categorized to 6 levels – under 40, 40-59, 60-69, 70-74, 75-79, above 80. If a patient is 50 years old, the corresponding vector is $v = [0, 1, 0, 0, 0, 0]^T$.

3.3 Formulation

For better illustration, we introduce the following notations for later formulation.

- $S_i, \mathcal{S}$: S_i is the historical records of i-th patient, including all types of demographic and physiological data. Let $\mathcal{S} = \{S_i\}_{i=1}^n$.
- $z_i, \mathcal{Z}$: z_i is the desired score of the i-th patient for evaluating severity of illness. Let $\mathcal{Z} = \{z_i\}_{i=1}^n$.
- $u_i, \mathcal{U}$: u_i is the i-th type of demographic data or physiological measurement of a patient, e.g, age, heart rate, etc. Let $\mathcal{U} = \{u_i\}_{i=1}^d$.
- $C_i, \mathcal{C}$: C_i is the set of correlated measurements (also include itself) of a patient with i-th type of demographic data or physiological measurement, e.g., potassium level and sodium level tend to rise and drop together. Let $\mathcal{C} = \{C_i\}_{i=1}^d$.
- θ_i, Θ: θ_i is importance weight of the i-th abnormal measurement contributing to a patient, i.e., $z_j = \sum_{i=1}^d \theta_i$. Let $\Theta = \{\theta_i\}_{i=1}^d$.

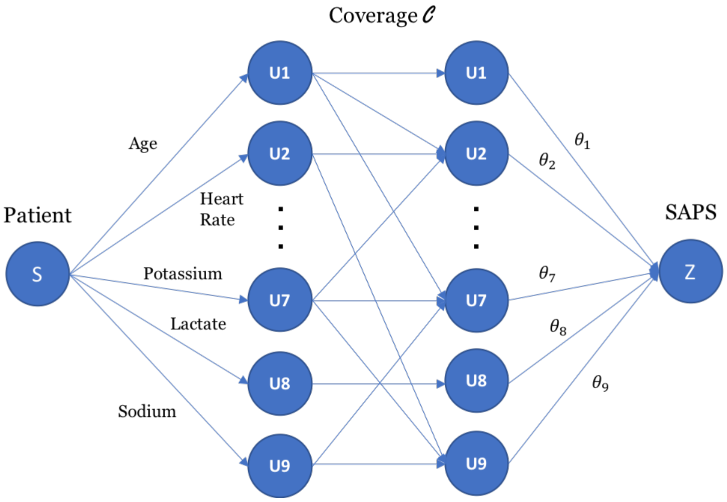

Figure 1: Illustration of maximum coverage framework. The historical records S of a patient contains multiple measurements, and each of them has a importance weights θ contributing to the evaluation score of severity of illness. Also, each measurement is correlated with some other measurements, which is represented by a bipartite.

Figure 1 illustrates maximum coverage framework in our setting. Based on these defined terms and after discretizing the measurements, we are able to propose our formulation of approximating patients severity of illness within maximum coverage framework:

Proposition 1 State-Cover-State. In this model, we assume only demographic data and physiological indices contribute to the evaluation score of severity. Not only does the observed abnormal measurements contribute to the severity, but also the correlated measurements which are unobserved play a part. Thus, given the set S containing k observed measurements of a patient, the coverage function f we learn is exactly function 4.

3.4 Optimization with DOPS and Structured SVM

So far, we have converted the approximation of patient severity of illness to the following learnable maximization problem: Given the training set $\mathcal{S} = \{S_i\}_{i=1}^{n}$ and test set $\mathcal{T} = \{T_i\}_{i=1}^{m}$ (also historical records of patients), we would like to utilize DOPS to learn f, i.e., Θ from $\mathcal{S}$, and find $\text{argmax}_{T_i \in \mathcal{T}} f_\Theta(T_i)$.

Denote that α-approximation or α-quantile solution for our maximization problem by

$$\alpha(\mathbf{z}) = \{y \in [m] : z_y \geq \alpha \max \mathbf{z}\} \tag{6}$$

$$\alpha(\mathbf{z}) = \{y \in [m] : z_y \in \text{top-}\alpha \text{ quantile of } \mathbf{z}\} \tag{7}$$

Given $\mathcal{S}$ and $\mathcal{T}$, DOPS requires minimizing the following loss function:

$$\mathcal{L} = \Delta_\alpha(\mathbf{z}, \hat{y}) = \mathbb{1}_{\{\hat{y} \notin \alpha(\mathbf{z})\}}, \quad \hat{y} = h(\mathcal{T}) = \text{argmax}_{T_i \in \mathcal{T}} f_\Theta(T_i) \tag{8}$$

where f_Θ is learned from $\mathcal{S}$. While $\Delta_\alpha(\mathbf{z}, \hat{y})$ is defined over m-tuples, we also have to divide $\mathcal{S}$ into $N = \lfloor \frac{n}{m} \rfloor$ m-tuples pieces $\{(\mathbf{S}^i, \mathbf{z}^i)\}_{i=1}^{N}$, and minimize the average empirical loss. Now the learning loss is

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \Delta_\alpha(\mathbf{z}^i, \hat{y}^i), \quad \hat{y}^i = h(\mathbf{S}^i) \tag{9}$$

This loss function represents that the learning-to-rank problem is approximated by a classification problem, which can be solved using Ranking SVM since f_Θ is linear. Specifically, we want to find a SVM separating $\mathcal{S}^{(1)} = \{S_y : y \in \alpha(\mathbf{z})\}$ and $\mathcal{S}^{(2)} = \{S_y : y \notin \alpha(\mathbf{z})\}$ ($\mathcal{S}^{(1)}$ should be ranked ahead of $\mathcal{S}^{(2)}$). By introducing program 2 and function 5, we obtain the following quadratic program:

$$\begin{aligned} \min_{\Theta \in \mathbb{R}^d, \xi_{iy'} \in \mathbb{R}} \quad & \frac{1}{2}||\Theta||^2 + \lambda \sum_{i=1}^{N} \sum_{y'=1}^{m} \xi_{iy'} \\ \text{s.t. } & f_\Theta(S_{y'}^i) - f_\Theta(S_y^i) + \xi_{iy'} \geq \Delta(y', y), \ \forall i \in [N], \forall y \in [m], \forall y' \in [m] \\ & \xi_{iy'} \geq 0, \ \forall i \in [N], \forall y' \in [m] \\ & \theta_i \geq 0, \ \forall i \in [d] \end{aligned} \tag{10}$$

Then the constraints in program 10 can be relaxed by substituting $\Delta(y', y)$ with $\Delta_\alpha(\mathbf{z}^i, y)$ in Eq. 9 and calculating the average loss over y':

$$\begin{aligned} \min_{\Theta \in \mathbb{R}^d, \xi_i \in \mathbb{R}} \quad & \frac{1}{2}||\Theta||^2 + \frac{\lambda}{N} \sum_{i=1}^{N} \xi_i \\ \text{s.t. } & \phi_\Theta(\mathbf{S}^i, \mathbf{z}^i) - f_\Theta(S_y^i) + \xi_i \geq \Delta_\alpha(\mathbf{z}^i, y), \ \forall i \in [N], \forall y \in [m] \\ & \xi_i \geq 0, \ \forall i \in [N] \\ & \theta_i \geq 0, \ \forall i \in [d] \end{aligned} \tag{11}$$

where $\phi_\Theta(\mathbf{S}^i, \mathbf{z}^i) = \frac{1}{|\alpha(\mathbf{z}^i)|} \sum_{y \in \alpha(\mathbf{z}^i)} f_\Theta(S_y^i)$. Further, program 11 can be reformulated into the following program by introduce $\max[\cdot]$ term:

$$\begin{aligned}
\min_{\Theta \in \mathbb{R}^d, \xi_i \in \mathbb{R}} \quad & \frac{1}{2}||\Theta||^2 + \frac{\lambda}{N}\sum_{i=1}^{N} \xi_i \\
\text{s.t. } & \xi_i \geq \max_y [\Delta_\alpha(\mathbf{z}^i, y) + f_\Theta(S_y^i) - \phi_\Theta(\mathbf{S}^i, \mathbf{z}^i)], \ \forall i \in [N], \forall y \in [m] \\
& \xi_i \geq 0, \ \forall i \in [N] \\
& \theta_i \geq 0, \ \forall i \in [d]
\end{aligned} \tag{12}$$

Finally, program 12 is equivalent to the following unconstrained quadratic program:

$$\min_{\Theta \in \mathbb{R}^d} \quad \frac{1}{2}||\Theta||^2 + \frac{\lambda}{N}\sum_{i=1}^{N} \max_y [\Delta_\alpha(\mathbf{z}^i, y) + f_\Theta(S_y^i) - \phi_\Theta(\mathbf{S}^i, \mathbf{z}^i)]_+ + \mu \mathbf{1}^T\Theta \tag{13}$$

where $[x]_+ = \max\{0, x\}$. Note that program 13 can be easily solved with a variety of gradient-based algorithms.

4 Experiments

4.1 Data Description

We use the hypotension patients data from MIMIC-III critical care dataset [Johnson *et al.*, 2016], which contains about 13,500 patients record. For each patient, the record contains a dataframe of baseline covariates such as age and weight, time series of vital sign measurements such as blood pressure and bicarbonate. For our problem, we took the record of each patient upon their admission into ICU, including ID, 26 demographic data and physiological indices and *Simplified Acute Physiology Score* (SAPS) [Le *et al.*, 1984]. SAPS is widely adopted for the scoring system that measures the severity of illness for patients admitted to ICU aged 15 or more. The calculation of SAPS is completed in first 24 hours starting from the admission of ICU, and no new score will be calculated during the patients' stay. This introduces another potential problem that while the patient's vital status changes throughout the duration of the stay resulting in different morbidity, not all physiological indices can be tracked and measured in real time, it becomes hard to calculate accurate SAPS, which requires an approach to approximate it. Therefore, our goal is to estimate SAPS using more frequently/easily obtainable vital data, so that hospitals and healthcare facilities can adapt their treatment strategy even without access to latest lab results.

Algorithm 1: Subgradient Descent-DOPS($\{S_i\}_{i=1}^n, \{z_i\}_{i=1}^n, \mathcal{T}, \alpha \in (0,1), \Theta, \lambda, \mu, \eta, E$, loss)

```
$m \leftarrow |\mathcal{T}|$;
Randomly partition $\lfloor \frac{n}{m} \rfloor$ sets $A_1, A_2, \ldots, A_N$;
Create $m$-tuple sample set $\mathcal{O} = \{(\mathbf{S}^i, \mathbf{z}^i)\}_{i=1}^N$, where $\mathbf{S}^i = \{S_j\}_{j \in A_i}$ and $\mathbf{z}^i = \{z_j\}_{z \in A_i}$;
if loss is “approxmation” then
    Compute $\alpha(\mathbf{z}^i) = \{y \in [m] : z_y^i \geq \alpha \max \mathbf{z}^i\}$;
else
    Compute $\alpha(\mathbf{z}^i) = \{y \in [m] : z_y^i \in$ top-$\alpha$ quantile of $\mathbf{z}^i\}$;
end
for $j = 1$ to $E$ do
    Compute $\mathcal{L}(\Theta) = \frac{1}{2}||\Theta||^2 + \frac{\lambda}{N}\sum_{i=1}^N \max_{y \in [m]}[\mathbb{1}_{\{y \notin \alpha(\mathbf{z}^i)\}} + f_\Theta(S_y^i) - \phi_\Theta(\mathbf{S}^i, \mathbf{z}^i)]_+ - \mu \mathbf{1}^T\Theta$;
    $\Theta \leftarrow \Theta - \eta \frac{\partial \mathcal{L}}{\partial \Theta}$;
end
return $argmax_{T \in \mathcal{T}} f_\Theta(T)$
```

4.2 Measurement Masking

To validate the idea of approximating illness severity of patients when certain lab results are not observed, we mask a part of physiological indices which are grossly hard to measured and collected in real time, i.e., “unobservable” in our setting. First, we rank all physiological indices according to their frequency of measurements. The lesser frequently a measurement is conducted, the more difficult to estimate its actual value at real time. Afterwards, we start “masking” physiological indices one by one by turning their values into zero.

4.3 Implementation Details

We solve program 13 using *subgradient descent* shown in Algorithm 1. For each measurement conversion method (two-level and multi-level) and each loss function (Eq. 6 and Eq. 7), we get the actual severity ranks of patients that the algorithm approximates to be the most severe, by averaging over 20 `DOPS` runs with 0.8-approximation and 0.8-top quantile solutions. We carry out this process with various test size m and run Linear Regression with/without intercept for baselines. The metrics for measuring performance are ranking accuracy (the predicted ranking of the most urgent patient compared with the actual maximum) and *Normalized Discounted Cumulative Gain* (NDCG) for top-n results.

4.4 Results

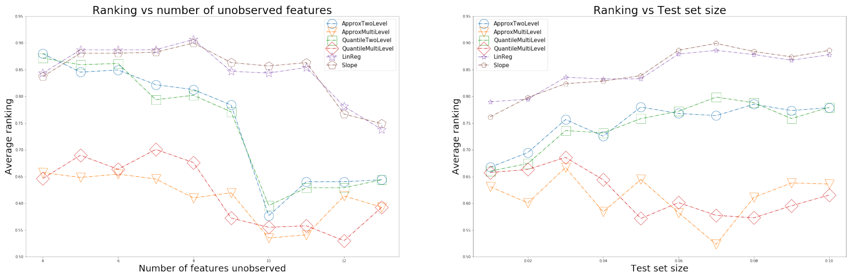

Figure 2: The comparison of performance measured by the predicted ranking of the most urgent patient, with the actual maximum, using different data transformations methods and loss functions. Note that “slope” means linear regression without intercept. *Left*: the different number of unobservable measurements (the cardinality constraint k); *Right*: the different test size m.

The results of ranking accuracy (the predicted ranking of the most urgent patient compared with the actual maximum) are shown in Figure 2, and the results of NDCG are shown in Figure 3.

- The left plots in Figure 2 and 3 indicate that as more measurements become unobservable, it becomes harder for `DOPS` models to learn the coverage function f and ranking performance suffers (also for linear regression). This is expected as the more measurements become inaccessible, the less information does the model know about the patients. However, linear regression outperforms both `DOPS` algorithms, which is unexpected. Also, the two-level coverage model gives superior performance than the multi-level one.
- The right plot in Figure 2 shows the model performance with a series of test set sizes. For linear regression and multi-level coverage model, the performance doesn’t improve a lot as the test size increases; but the ranking accuracy of two-level coverage model improve. As test size becomes larger, its performance stay stable. This is partly because when the number of test samples increases, the information for classification becomes abundant, so the selected argmax may move forward for higher ranking. We also think multi-level model should also follow this trend, but the result is unexpected.
- The right plot in Figure 3 shows the model performance with different rank position p. We can see that while selecting a subset of most urgent patient, the performance relation is still “linear regression > two-level coverage model > multi-level coverage model”. As the size of

the group of urgent patients we want to select increase, NDCG shows a downward trend, because `DOPS` algorithm may make more mistakes since the SVM classifier cannot separate desired “maximum” group with the rest of candidates well. Choosing a smaller α may help.

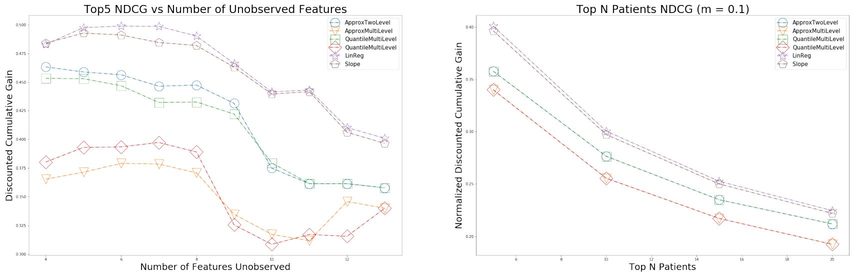

Figure 3: The comparison of performance measured by NDCG, using different data transformations methods and loss functions. *Left*: the different number of unobservable measurements (the cardinality constraint k); *Right*: the different rank position p when test size $m = 0.1$.

We think the there are couple potential reasons for the undesirable results.

- Due to the nature of data, the correlations between different physiological indices are very noisy. Current implementation cannot accurately capture the "coverage" relationships between each feature.
- The multi-level model has the worst performance. Since the level cutoffs are manually designed, our one-hot encoding method might potentially introduce additional noise. We can also adjust our imputation strategy. Currently, for unobserved features, all levels for the corresponding features are set to 0. An alternative approach is to set the level of mean/median value to 1 first, then overwrite the level encoding when that feature is covered by other observable features.
- Due to time constraint, we could not run the models for the full range of measurements. However, as we can see, two-level model is actually slightly better than linear regression when 4 measurements are inaccessible. There is a good chance that our model outperforms linear regression with fewer unobservable features, which can still be beneficial on behalf of clinician’s knowledge on patients in real world.

5 Conclusion and Discussion

In this work, we demonstrate the possibility of approximating and ranking patients severity of illness without observing the necessary physiological indices for the exact calculation of risk score. The main contribution and novelty of our work mainly consists of 3 parts:

- The inspiration of applying learning-to-rank framework for practical healthcare problems.
- The discovery and formulation of maximum coverage problem under healthcare settings.
- Solving this learning-to-rank problem with `DOPS` algorithm.

There are certain limitations to our current work. Firstly, current coverage relation $\mathcal{C}$ extracted from correlations of demographic and physiological indices is not rigorous and bulletproof. Although we attempt to make the coverage as reasonable as possible, a great deal of domain knowledge is required. Thus, the validation from domain experts of our coverage relation will make the results more trustworthy. Secondly, the performance of our `DOPS` algorithm partly depends on the size of training data due to its theoretical guarantee. In order to achieve higher accuracy, much more training data is needed, yet currently this is beyond our current capability to acquire. Further, converting continuous features into discrete values must give rise to the information loss, which contributes to the low ranking accuracy.

Nonetheless, we think this framework can lead to many interesting future directions. An immediate extension we can take is to learn the set representation of the inputs. Instead of coming up with conversion criteria manually, we can use neural networks with appropriate structure to learn the binary latent space, then feed this binary data to DOPS model. Another possible work is to considering treatment effect within maximum coverage framework. From a real world perspective, we know treatments usually have side effects, which means they are not only going to change targeted physiological indices but also induce other undesirable physiological transitions. For instance, Fluid Bolus Therapy (FBT) is a treatment to boost the blood pressure of hypotensive patients. However, excessive FBT can also introduce problems such as slowed heart rate and breathing difficulty. If we can correctly understand the mechanism of drugs and know about their corresponding potential physiological changes, we can use similar approach to approximate the risk level of patients after undergoing treatment, which will become invaluable guideline for ICU clinicians. Last but not least, DOPS is applicable for a variety of combinatorial optimization problems, including Graph k-cuts, Unite Demands, etc. It is worthwhile exploring these combinatorial optimization problems under healthcare settings.

References

Najwan Abu Al-Saad, Chris Skedgel, and Jurgens Nortje. Principles of resource allocation in critical care. *Bja Education*, 2017.

Mohammad Alsulmi and Ben Carterette. Improving medical search tasks using learning to rank. In *2018 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 1–8. IEEE, 2018.

Maria-Florina Balcan and Nicholas JA Harvey. Learning submodular functions. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 793–802. ACM, 2011.

Eric Balkanski, Aviad Rubinstein, and Yaron Singer. The power of optimization from samples. In *Advances in Neural Information Processing Systems*, pages 4017–4025, 2016.

Eric Balkanski, Aviad Rubinstein, and Yaron Singer. The limitations of optimization from samples. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1016–1027. ACM, 2017.

Philip Betbeze. How to cut costs and improve outcomes in the icu, 2018.

Christopher Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Gregory N Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd International Conference on Machine learning (ICML-05)*, pages 89–96, 2005.

Shaddin Dughmi and Jan Vondrák. Limitations of randomized mechanisms for combinatorial auctions. *Games and Economic Behavior*, 92:370–400, 2015.

Uma M Girkar, Ryo Uchimido, Li-wei H Lehman, Peter Szolovits, Leo Celi, and Wei-Hung Weng. Predicting blood pressure response to fluid bolus therapy using attention-based neural networks for clinical interpretability. *arXiv preprint arXiv:1812.00699*, 2018.

Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research*, 42:427–486, 2011.

Neil A Halpern and Stephen M Pastores. Critical care medicine in the united states 2000–2005: an analysis of bed numbers, occupancy rates, payer mix, and costs. *Critical care medicine*, 38(1):65–71, 2010.

Ralf Herbrich, Thore Graepel, and Klaus Obermayer. Support vector learning for ordinal regression. 1999.

Thorsten Joachims. Training linear svms in linear time. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 217–226. ACM, 2006.

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

PN Lanken, Peter Browne Terry, DC Adler, JA Brooks-Brunn, SW Crawford, M Danis, AJ Fedullo, JE Gottlieb, J Hansen-Flaschen, MH Kollef, et al. Fair allocation of intensive care unit resources. *American journal of respiratory and critical care medicine*, 156(4 I):1282–1301, 1997.

JR Gall Le, Philippe Loirat, Annick Alperovitch, Paul Glaser, Claude Granthil, Daniel Mathieu, Philippe Mercier, Remi Thomas, and Daniel Villers. A simplified acute physiology score for icu patients. *Critical care medicine*, 12(11):975–977, 1984.

Hui Lin and Jeff Bilmes. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 510–520. Association for Computational Linguistics, 2011.

Tie-Yan Liu et al. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331, 2009.

Takanori Maehara, Yasushi Kawase, Hanna Sumita, Katsuya Tono, and Ken-ichi Kawarabayashi. Optimal pricing for submodular valuations with bounded curvature. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

A. Mittal, M. B. Blaschko, A. Zisserman, and P. H. S. Torr. Taxonomic multi-class prediction and person layout using efficient structured ranking. In *European Conference on Computer Vision*, 2012.

Amy Grace Rapsang and Devajit C Shyam. Scoring systems in the intensive care unit: a compendium. *Indian journal of critical care medicine: peer-reviewed, official publication of Indian Society of Critical Care Medicine*, 18(4):220, 2014.

Nir Rosenfeld, Eric Balkanski, Amir Globerson, and Yaron Singer. Learning to optimize combinatorial functions. In *International Conference on Machine Learning*, pages 4371–4380, 2018.

Gal Shulkind, Stefanie Jegelka, and Gregory W Wornell. Sensor array design through submodular optimization. *IEEE Transactions on Information Theory*, 65(1):664–675, 2019.

Ruben Sipos, Pannaga Shivaswamy, and Thorsten Joachims. Large-margin learning of submodular summarization models. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 224–233. Association for Computational Linguistics, 2012.

Yukihiro Tagami, Shingo Ono, Koji Yamamoto, Koji Tsukamoto, and Akira Tajima. Ctr prediction for contextual advertising: Learning-to-rank approach. In *Proceedings of the Seventh International Workshop on Data Mining for Online Advertising*, page 4. ACM, 2013.

R Andrew Taylor, Joseph R Pare, Arjun K Venkatesh, Hani Mowafi, Edward R Melnick, William Fleischman, and M Kennedy Hall. Prediction of in-hospital mortality in emergency department patients with sepsis: A local big data–driven, machine learning approach. *Academic emergency medicine*, 23(3):269–278, 2016.

Ioannis Tsochantaridis, Thomas Hofmann, Thorsten Joachims, and Yasemin Altun. Support vector machine learning for interdependent and structured output spaces. In *Proceedings of the twenty-first international conference on Machine learning*, page 104. ACM, 2004.