A Multi-Instance Support Vector Machine with Incomplete Data for Clinical Outcome Prediction of COVID-19

Lodewijk Brand
lbrand@mymail.mines.edu
Colorado School of Mines
Golden, Colorado, USA

Lauren Zoe Baker
laurenzoebaker@mymail.mines.edu
Colorado School of Mines
Golden, Colorado, USA

Hua Wang
huawangcs@gmail.com
Colorado School of Mines
Golden, Colorado, USA

ABSTRACT

In order to manage the public health crisis associated with COVID-19, it is critically important that healthcare workers can quickly identify high-risk patients in order to provide effective treatment with limited resources. Statistical learning tools have the potential to help predict serious infection early-on in the progression of the disease. However, many of these techniques are unable to take full advantage of temporal data on a per-patient basis as they handle the problem as a *single-instance* classification. Furthermore, these algorithms rely on complete data to make their predictions. In this work, we present a novel approach to handle the temporal and missing data problems, simultaneously; our proposed *Simultaneous Imputation-Multi Instance Support Vector Machine* method illustrates how multiple instance learning techniques and low-rank data imputation can be utilized to accurately predict clinical outcomes of COVID-19 patients. We compare our approach against recent methods used to predict outcomes on a public dataset with a cohort of 361 COVID-19 positive patients. In addition to improved prediction performance early on in the progression of the disease, our method identifies a collection of biomarkers associated with the liver, immune system, and blood, that deserve additional study and may provide additional insight into causes of patient mortality due to COVID-19. We publish the source code for our method online.[1]

CCS CONCEPTS

• **Computing methodologies** → **Instance-based learning**; • **Applied computing** → *Health care information systems.*

KEYWORDS

classification, alternating direction method of multipliers, multiple instance learning, missing data, COVID-19

ACM Reference Format:
Lodewijk Brand, Lauren Zoe Baker, and Hua Wang. 2021. A Multi-Instance Support Vector Machine with Incomplete Data for Clinical Outcome Prediction of COVID-19. In *12th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (BCB '21), August 1–4, 2021, Gainesville, FL, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3459930.3469552

[1]Code is provided at: https://github.com/minds-mines/SimMISVM.jl

BCB '21, August 1–4, 2021, Gainesville, FL, USA

ACM ISBN 978-1-4503-8450-6/21/08…$15.00
https://doi.org/10.1145/3459930.3469552

1 INTRODUCTION

Predicting mortality of a COVID-19 patient early in their hospital stay would allow adequate time and resources to care for high-risk patients. However, this prediction problem presents two unique challenges. First, the clinical data provided is not necessarily the same size for each patient. For example, a patient that has been in critical care for many days will have more data available than another patient who has recently been admitted. We refer to this type of data as *multi-instance* data where a single patient can contain multiple clinical observations observed over time. Second, these clinical data inevitably contain many missing entries [11] due to the physical constraints of a caregiver in a hospital setting. The variable size of the input data and incompleteness are significant challenges for the modern statistical learning toolbox and are solved using a variety of pre-processing methods. For example, Ma *et al.* [16], utilized random forests to identify clinical outcomes of COVID-19 patients by aggregating data into a single vector per-patient before the algorithm is applied. These techniques may miss-out on the temporal changes evident across the clinical data. Another recent approach, proposed by Yan *et al.* [23], drops instances with missing records to ensure that the algorithms operate on dense data; although, this approach may inadvertently lead to the removal of valuable information.

In this work we propose a *Simultaneous Imputation-Multi Instance Support Vector Machine* method that handles the temporal prediction and incomplete data challenges at the same time for clinical outcome prediction. Our approach relies on combining techniques from multi-instance learning [6, 15, 18–20], as well as matrix completion [5], to handle missing data across an entire patient cohort. In this work we present the following scientific contributions:

- A detailed derivation of a novel multi-instance support vector machine, in its primal form, that is explicitly designed to handle temporal and missing data at the same time.
- Experimental results illustrating how multi-instance learning techniques can identify serious COVID-19 cases earlier than traditional single-instance learning methods.
- Biomarkers, validated by current literature and identified by our approach, that may be predictive of serious outcomes related to COVID-19.

2 METHODS

In this manuscript we represent matrices $\mathbf{M}$ as bold uppercase letters, vectors $\mathbf{m}$ as bold lowercase letters, and scalars m as lowercase

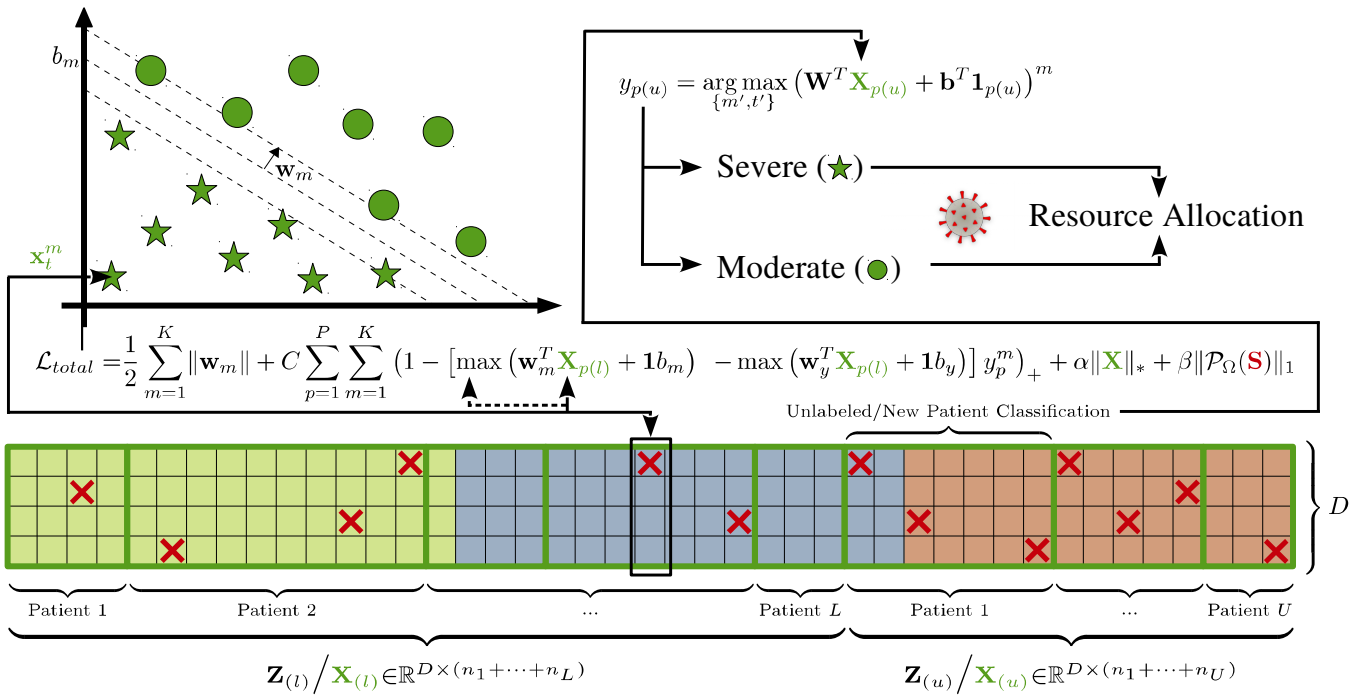


Figure 1: A visualization of the *Simultaneous Imputation-Multi Instance Support Vector Machine* method applied to temporal clinical data. Our method takes as input incomplete temporal data in Z of which L patients have known clinical outcomes. The model is jointly optimized to perform both a classification and imputation task to learn a dense matrix X (green). The trace-norm regularization, $\|\mathrm{X}\|_*$, ensures that the completed data matrix captures patterns across all labeled and unlabeled clinical data with possible corruptions/outliers captured by S (red ×'s). The unlabeled patient data, $\mathrm{X}_{p(u)} \in \mathrm{X}$, which is imputed from the original data, $\mathrm{Z}_{(u)}$, is classified once the joint optimization has finished.

letters. The i-th row and j-th column of a matrix $\mathbf{M}$ are denoted as $\mathbf{m}^i$ and $\mathbf{m}_j$, respectively. Similarly, m_j^i is the scalar value indexed by the i-th row and j-th column of the matrix $\mathbf{M}$. The matrix $\mathbf{M}_p$ corresponds to the p-th column-block of the matrix $\mathbf{M}$. Given the $K \times T$ matrix $\mathbf{M}$, $\{m, t\} = \arg\max_{m',t'}(\mathbf{M})$ gives the row-by-column coordinates for the maximum element in the matrix $\mathbf{M}$. The row and column indices are given by $\arg\max_{m',t'}(\mathbf{M})^m$ and $\arg\max_{m',t'}(\mathbf{M})_t$, respectively.

2.0.1 Building the Objective. We begin with the following general loss function:

$$\mathcal{L}_{total} = \mathcal{L}_{classification} + \mathcal{L}_{impute} \ , \tag{1}$$

which, when optimized, is designed to jointly perform a clinical outcome prediction when provided with incomplete/missing data. Following the multi-task learning paradigm, we expect that the data imputation task should guide the classification task and vice versa, thus we optimize them simultaneously.

In a hospital setting, clinical data will usually contain a varied number of temporal records per-patient (*e.g.* some patients may have been in the hospital longer than other patients). Thus, a classification problem using clinical data can naturally be formulated as a multi-instance classification. In order to classify a patient represented by multi-instance data, and/or to learn from patients who have been discharged, we define a decision function that operates on a patient matrix $\mathbf{X}_p \in \mathbb{R}^{D\times n_p}$ representing n_p clinical measurements with D features as

$$y_p = \arg\max_{\{m',t'\}} \left(\mathbf{W}^T\mathbf{X}_p + \mathbf{b}^T\mathbf{1}_p\right)^m \ , \tag{2}$$

where $\mathbf{W} \in \mathbb{R}^{D\times K}$ and $\mathbf{b} \in \mathbb{R}^K$ are the predictors and intercepts for K classes and $y_p \in K$ is the predicted class index.[2] Intuitively, Eq. (2) returns the class index $m \in K$ from which the t'-th instance in $\mathbf{X}_p$ provides the largest output from the parameterized model. Note that Eq. (2) is defined even if n_p is different for each patient. In order to learn a model parameterized by $\mathbf{W}$ and $\mathbf{b}$ we turn to the multi-instance classifier and support vector machine formulations presented in [1], and [22] and propose the following ℓ_2-regularized multi-instance support vector machine (MISVM)

$$\begin{aligned}\min_{\mathbf{W},\mathbf{b}} \frac{1}{2}\sum_{m=1}^{K}\|\mathbf{w}_m\|_2^2 + C\sum_{p=1}^{P}\sum_{m=1}^{K}\big(1 - [\max(\mathbf{w}_m^T\mathbf{X}_p \\ + \mathbf{1}b_m) - \max(\mathbf{w}_y^T\mathbf{X}_p + \mathbf{1}b_y)]y_p^m\big)_+ \ ,\end{aligned} \tag{3}$$

where C is a tuning parameter, P is the total number of patients and $(\cdot)_+ = \max(\cdot, 0)$. The m-th class label for the p-th patient is captured by $y_p^m \in \{-1, 1\}$. The hyperplane $\mathbf{w}_y$ and intercept b_y are

[2] In Eq. (2) we explicitly provide the size of the row-vector $\mathbf{1}_p$ as the number of instances in $\mathbf{X}_p$, for clarity. Although, for the remainder of this manuscript, we will omit this subscript to simplify notation.

associated with the positive class label for $\mathbf{X}_p$. We refer to Eq. (3) as the "Weston & Watkins MISVM." This formulation allows us to handle the K-classification problem as a single optimization instead of one-vs-all approaches. The SVM proposed by Weston & Watkins has been shown to provide [10] higher accuracies on benchmark datasets than other formulations.

Frequently, the temporal data collected for a given patient during a hospital stay is incomplete. Since data is collected across multiple features at different times it is practically impossible to collect a complete data matrix. Thus, we propose a data imputation approach that can ensure that the multi-instance classification task in Eq. (3) is well defined even when provided with incomplete data. Motivated by [5], we formulate the data imputation task as

$$\min_{\mathbf{X},\mathbf{S}} \|\mathbf{X}\|_* + \beta \|\mathcal{P}_\Omega(\mathbf{S})\|_1 \quad \text{subject to} \quad \mathbf{X}+\mathbf{S}=\mathbf{Z}\ , \tag{4}$$

where β is a tuning parameter and $\mathbf{Z} \in \mathbb{R}^{D\times(n_1+\cdots+n_P)}$ is a horizontal concatenation of the patient data with missing entries. The function $\mathcal{P}_\Omega(\cdot)$ is an orthogonal projection onto the data available in $\mathbf{Z}$. By optimizing over Eq. (4), we intend to uncover a complete data matrix $\mathbf{X}$ that captures a low-rank representation across all patient observations within our cohort. The trace norm regularization on $\mathbf{X}$ will discover underlying patterns across all clinical observations. The matrix $\mathbf{S}$ in the second term, allows for possible outliers present in the original data matrix $\mathbf{Z}$.

Combining Eq. (3) and Eq. (4) together gives us our proposed objective

$$\begin{aligned}\min_{\substack{\mathbf{X},\mathbf{S},\\ \mathbf{W},\mathbf{b}}} \ & \frac{1}{2}\sum_{m=1}^{K}\|\mathbf{w}_m\|_2^2 + C\sum_{p=1}^{P}\sum_{m=1}^{K}(1-[\max(\mathbf{w}_m^T\mathbf{X}_{p(l)} \\ & + \mathbf{1}b_m) - \max(\mathbf{w}_y^T\mathbf{X}_{p(l)}+\mathbf{1}b_y)]y_p^m)_+ + \alpha\|\mathbf{X}\|_* \\ & + \beta\|\mathcal{P}_\Omega(\mathbf{S})\|_1 \quad \text{subject to} \quad \mathbf{X}+\mathbf{S}=\mathbf{Z}\ ,\end{aligned} \tag{5}$$

where $\mathbf{Z}$ and $\mathbf{X}$ are explicitly separated into labeled-unlabeled pairs by $\mathbf{Z}_{(l)}, \mathbf{Z}_{(u)}, \mathbf{X}_{(l)}$ and $\mathbf{X}_{(u)}$ (see Figure 1). We call Eq. (5) the *Simultaneous Imputation-Multi Instance Support Vector Machine* objective. We note that although we only handle the *binary* clinical outcome prediction task in this manuscript, our formulation allows us to classify any number ($K \geq 2$) of case severities. While our final objective is clearly motivated, it is difficult to solve efficiently due to term coupling by $\mathbf{X}$. For this, we use the alternating direction method of multipliers (ADMM) framework to design an algorithm. The idea of the ADMM is to decouple a larger problem that is difficult to solve into collection of smaller parts

$$\begin{aligned}&\min_{x_i} f_1(x_1)+f_2(x_2)+\cdots+f_N(x_N) \\ &\text{subject to} \quad \mathbf{E}_1x_1+\mathbf{E}_2x_2+\cdots+\mathbf{E}_Nx_N=c\ ,\end{aligned} \tag{6}$$

that are easier to solve. Once an appropriately decoupled problem has been defined, the constraints in the decoupled form are incorporated into the objective via an augmented Lagrangian. The updates for the primal variables x_i follow a Douglas-Rachford splitting strategy followed by a dual variable update. See [3] for further details on the ADMM.

Inspired by [17], [21], and the multiblock extension of the ADMM [3] we introduce constraints $e_p^m = y_p^m - q_p^m + r_p^m$, $q_p^m = \max(\mathbf{t}_p^m)$, $\mathbf{t}_p^m = \mathbf{w}_m^T\mathbf{X}_i + \mathbf{1}b_m$, $r_p^m = \max\left(\mathbf{u}_p^m\right)$, $\mathbf{u}_p^m = \mathbf{w}_y^T\mathbf{X}_i + \mathbf{1}b_y$, and $\mathbf{F} = \mathbf{X}$ to decouple the optimization variables in Eq. (5). Then, the augmented Lagrangian can be written as

$$\begin{aligned}\mathcal{L}(\mathrm{p}_{vars}, \mathrm{d}_{vars}) = & \frac{1}{2}\sum_{m=1}^{K}\|\mathbf{w}_m\|_2^2 + \sum_{p=1}^{P}\sum_{m=1}^{K} C\left(y_p^m e_p^m\right)_+ \\ & + \alpha\|\mathbf{F}\|_* + \beta\|\mathcal{P}_\Omega(\mathbf{S})\|_1 + \frac{\mu}{2}\sum_{p=1}^{P}\sum_{m=1}^{K}\Bigg[\left(e_p^m - n_p^m\right)^2 \\ & + \left(q_p^m - \max\left(\mathbf{t}_p^m\right) + \sigma_p^m/\mu\right)^2 + \left(r_p^m - \max\left(\mathbf{u}_p^m\right)\right. \\ & \left. + \omega_p^m/\mu\right)^2 + \left\|\mathbf{t}_p^m - \left(\mathbf{w}_m^T\mathbf{X}_p + \mathbf{1}b_m\right) + \boldsymbol{\theta}_p^m/\mu\right\|_2^2 \\ & + \left\|\mathbf{u}_p^m - \left(\mathbf{w}_y^T\mathbf{X}_p + \mathbf{1}b_y\right) + \boldsymbol{\xi}_p^m/\mu\right\|_2^2\Bigg] \\ & + \frac{\mu}{2}\|\mathbf{F} - \mathbf{X} + \boldsymbol{\Pi}/\mu\|_F^2 + \frac{\mu}{2}\|\mathbf{Z} - (\mathbf{X}+\mathbf{S}) + \boldsymbol{\Delta}/\mu\|_F^2 \\ & \text{where} \quad n_p^m = y_p^m - q_p^m + r_p^m - \lambda_p^m/\mu\ ,\end{aligned} \tag{7}$$

$\mathrm{p}_{vars} = \{\mathbf{W}, \mathbf{b}, \mathbf{X}, \mathbf{E}, \mathbf{Q}, \mathbf{T}, \mathbf{R}, \mathbf{U}, \mathbf{F}, \mathbf{S}\}$ are the primal variables, $\mathrm{d}_{vars} = \{\boldsymbol{\Lambda}, \Sigma, \boldsymbol{\Theta}, \Omega, \Xi, \boldsymbol{\Pi}, \Delta\}$ are the dual variables and $\mu > 0$ is a tuning parameter. Given the augmented Lagrangian, we derive an algorithm by differentiating Eq. (7) with respect to each primal variable, setting the derivative equal to zero, and solving for the differentiating variable; this process is repeated for each primal variable in p_{vars}. After each primal variables has been updated the dual variables are updated accordingly and μ is increased by a factor $\rho > 1$ for the next round. The algorithm terminates when the residuals of the constraints introduced before Eq. (7) are less than a predefined tolerance, which equivalently solves the original problem in Eq. (5).

For the remainder of this section we derive the primal variable updates for optimizing the Weston & Watkins MISVM followed by the proposed *Simultaneous Imputation-Multi Instance Support Vector Machine* method; this is done to increase the clarity of our derivation as Eq. (3) is a subset (*e.g.* without data imputation) of Eq. (5). Finally, we provide Algorithm 1 and Algorithm 2 which clearly specifies the initializations, assorted hyperparameters, dual updates, and the sequence in which each primal variable is updated in the associated code.

W update Removing all terms from Eq. (7) that do not include $\mathbf{W}$ and decoupling across columns of $\mathbf{W}$ gives the following K problems to solve

$$\begin{aligned}\mathbf{w}_m = \arg\min_{\mathbf{w}_m} \ & \frac{1}{2}\|\mathbf{w}_m\|_2^2 + \frac{\mu}{2}\sum_{p=1}^{P}\Bigg[\left\|\mathbf{t}_p^m - \left(\mathbf{w}_m^T\mathbf{X}_p + b_m\right)\right. \\ & \left. + \theta_p^m/\mu\right\|_2^2\Bigg] + \sum_{p'=1}^{P'}\sum_{m=1}^{K}\left[\frac{\mu}{2}\left\|\mathbf{u}_{p'}^m - \left(\mathbf{w}_m^T\mathbf{X}_{p'} + b_m\right) + \xi_{p'}^m/\mu\right\|_2^2\right]\ ,\end{aligned} \tag{8}$$

where p' indicates the column blocks in $\mathbf{X}$ (and the corresponding columns of $\mathbf{U}$ and Ξ) that belong to the m-th class. Taking the derivative of Eq. (8) with respect to $\mathbf{w}_k$ and setting it equal to zero gives the closed form solution

$$\begin{aligned}\mathbf{w}_m^T = & \left(\sum_{p=1}^{P}\left[\left(\mathbf{t}_p^m - \mathbf{1}b_m + \boldsymbol{\theta}_p^m/\mu\right)\mathbf{X}_p^T\right]\right. \\ & \left. + \sum_{p'=1}^{P'}\sum_{m=1}^{K}\left[\left(\mathbf{u}_{p'}^m - \mathbf{1}b_m + \boldsymbol{\xi}_{p'}^m/\mu\right)\mathbf{X}_{p'}^T\right]\right) \\ & * \left(\mathbf{I}/\mu + \sum_{p=1}^{P}\mathbf{X}_p\mathbf{X}_p^T + K\sum_{p'=1}^{P'}\mathbf{X}_{p'}\mathbf{X}_{p'}^T\right)^{-1}\ .\end{aligned} \tag{9}$$

Algorithm 1 Multiblock ADMM for Optimizing Eq. (3)

Data: $\mathbf{X} \in \mathbb{R}^{D\times(n_1+\cdots+n_P)}$ and $\mathbf{Y} \in \{-1,1\}^{K\times P}$.
Hyperparameters: $C > 0$, $\mu > 0$, $\rho > 1$ and $tolerance > 0$.
Initialize: primal $\mathbf{W}, \mathbf{b}, \mathbf{E}, \mathbf{Q}, \mathbf{R}, \mathbf{T}, \mathbf{U}$ and dual variables $\mathbf{\Lambda}, \mathbf{\Sigma}, \mathbf{\Theta}, \mathbf{\Omega}, \mathbf{\Xi}$.
while residual $> tolerance$ **do**
  for $m \in M$ **do**
    Update $\mathbf{w}_m \in \mathbf{W}$ **by** Eq. (9)
    Update $b_m \in \mathbf{b}$ **by** Eq. (11)
  end for
  for $(p, m) \in \{P, M\}$ **do**
    Update $e_p^m \in \mathbf{E}$ **by** Eq. (13)
    Update $q_p^m \in \mathbf{Q}$ **by** Eq. (15)
    Update $r_p^m \in \mathbf{R}$ **by** Eq. (16)
    for $j \in n_p$ **do**
      Update $t_{p,j}^m \in \mathbf{T}$ **by** Eq. (19)
      Update $u_{p,j}^m \in \mathbf{U}$ **by** Eq. (20)
    end for
    Update $\lambda_p^m, \sigma_p^m, \omega_p^m, \boldsymbol{\theta}_p^m, \boldsymbol{\xi}_p^m$ **by** $\lambda_p^m = \lambda_p^m + \mu(e_p^m - (y_p^m - q_p^m + r_p^m))$; $\sigma_p^m = \sigma_p^m + \mu(q_p^m - \max(\mathbf{t}_p^m))$; $\omega_p^m = \omega_p^m + \mu(r_p^m - \max(\mathbf{u}_p^m))$; $\boldsymbol{\theta}_p^m = \boldsymbol{\theta}_p^m + \mu(\mathbf{t}_p^m - (\mathbf{w}_m^T\mathbf{X}_p + \mathbf{1}b_m))$; $\boldsymbol{\xi}_p^m = \boldsymbol{\xi}_p^m + \mu(\mathbf{u}_p^m - (\mathbf{w}_y^T\mathbf{X}_p + \mathbf{1}b_y))$.
  end for
  Update $\mu = \rho\mu$
end while
return $(\mathbf{w}_m, \ldots, \mathbf{w}_K) \in \mathbf{W}$ and $(b_1, \ldots, b_K) \in \mathbf{b}$.

Algorithm 2 Multiblock ADMM for Optimizing Eq. (5)

Data: $\mathbf{Z}_{(l)} \in \mathbb{R}^{D\times(n_1+\cdots+n_L)}$, $\mathbf{Y} \in \{-1,1\}^{K\times L}$, $\mathbf{Z}_{(u)} \in \mathbb{R}^{D\times(n_1+\cdots+n_U)}$, and a masking function $\mathcal{P}_\Omega$ indicating whether an entry in $\mathbf{Z} \in \mathbb{R}^{D\times(n_1+\cdots+n_P)}$ is available/missing.
Hyperparameters: $C > 0$, $\alpha > 0$, $\beta > 0$, $\mu > 0$, $\rho > 1$ and $tolerance > 0$.
Initialize: primal $\mathbf{W}, \mathbf{b}, \mathbf{X}, \mathbf{E}, \mathbf{Q}, \mathbf{R}, \mathbf{T}, \mathbf{U}, \mathbf{F}, \mathbf{S}$ and dual variables $\mathbf{\Lambda}, \mathbf{\Sigma}, \mathbf{\Theta}, \mathbf{\Omega}, \mathbf{\Xi}, \mathbf{\Pi}, \mathbf{\Delta}$.
while residual $> tolerance$ **do**
  for $m \in M$ **do**
    Update $\mathbf{w}_m$ **and** b_m **by** line 6 and 7 in Alg. 1
  end for
  for $p \in P$ **do**
    Update $\mathbf{X}_{p(l)} \in \mathbf{X}$ **by** Eq. (23)
    Update $\mathbf{X}_{p(u)} \in \mathbf{X}$ **by** Eq. (24)
    Update $e_p^m, q_p^m, r_p^m, t_{p,j}^m, u_{p,j}^m$ **by** lines 10-15 in Alg. 1
    Update $\lambda_p^m, \sigma_p^m, \omega_p^m, \boldsymbol{\theta}_p^m, \boldsymbol{\xi}_p^m$ **by** line 17 in Alg. 1
  end for
  Update $\mathbf{F}$ **by** Eq. (26)
  Update $s_n^d \in \mathbf{S}$ **by** Eq. (28)
  Update $\mathbf{\Pi}, \mathbf{\Delta}$ **by** $\mathbf{\Pi} = \mathbf{\Pi} + \mu(\mathbf{F} - \mathbf{X})$; $\mathbf{\Delta} = \mathbf{\Delta} + \mu(\mathbf{Z} - (\mathbf{X} + \mathbf{S}))$
  Update $\mu = \rho\mu$
end while
return $\left[y_{p(u)} = \arg\max_{m',t'}(\mathbf{W}^T\mathbf{X}_{p(u)} + \mathbf{b}^T\mathbf{1}_{p(u)})^m : u \in \{1, 2, \ldots, U\}\right]$.

b update Removing terms that do not include $\mathbf{b}$ from Eq. (7) and decoupling across each element of $\mathbf{b}$ gives K problems to solve

$$b_m = \arg\min_{b_m} \sum_{p=1}^{P}\left[\left\|\mathbf{t}_p^m - \left(\mathbf{w}_m^T\mathbf{X}_p + b_m\right) + \theta_p^m/\mu\right\|_2^2\right] + \sum_{p'=1}^{P'}\sum_{m=1}^{K}\left[\left\|\mathbf{u}_{p'}^m - \left(\mathbf{w}_m^T\mathbf{X}_{p'} + b_m\right) + \xi_{p'}^m/\mu\right\|_2^2\right] . \tag{10}$$

Once again, p' indicates the column blocks that belong to the m-th class are chosen from $\mathbf{X}$. Taking the derivative of Eq. (10) with respect to b_m, setting the derivative equal to zero, and solving for b_m gives

$$b_m = \left(\sum_{p=1}^{P}\left[\mathbf{t}_p^m - \mathbf{w}_m^T\mathbf{X}_p + \boldsymbol{\theta}_p^m/\mu\right] + \sum_{p'=1}^{P'}\sum_{m=1}^{K}\left[\mathbf{u}_{p'}^m - \mathbf{w}_m^T\mathbf{X}_{p'} + \boldsymbol{\xi}_{p'}^m/\mu\right]\right)/\left(P + KP'\right) , \tag{11}$$

where P' is the total number of patients belonging to the m-th class.

E update Dropping terms from Eq. (7), that do not contain $\mathbf{E}$ and decoupling element-wise gives $K \times P$ problems

$$e_p^m = \arg\min_{e_p^m} C\left(y_p^m e_p^m\right)_+ + \frac{\mu}{2}\left(e_p^m - n_p^m\right)^2 , \tag{12}$$

where $n_p^m = y_p^m - q_p^m + r_p^m - \frac{\lambda_p^m}{\mu}$. Equation (12) can be differentiated with respect to e_p^m, set equal to zero, and solved in three cases

$$e_p^m = \begin{cases} n_p^m - \frac{C}{\mu}y_p^m & \text{when} \quad y_p^m n_p^m > \frac{C}{\mu} \\ 0 & \text{when} \quad 0 \le y_p^m n_p^m \le \frac{C}{\mu} \\ n_p^m & \text{when} \quad y_p^m n_p^m < 0 \end{cases} . \tag{13}$$

Q update Keeping only terms with $\mathbf{Q}$ in Eq. (7) and decoupling element-wise gives $K \times P$ problems

$$q_p^m = \arg\min_{q_p^m}\left(e_p^m - y_p^m + q_p^m - r_p^m + \lambda_p^m/\mu\right)^2 + \left(q_p^m - \max\left(\mathbf{t}_p^m\right) + \sigma_p^m/\mu\right)^2 . \tag{14}$$

Taking the derivative of Eq. (14) with respect to q_p^m, setting the result equal to zero, and solving for q_p^m gives the update

$$q_p^m = \frac{\left(y_p^m - e_p^m + r_p^m - \lambda_p^m/\mu + \max\left(\mathbf{t}_p^m\right) - \sigma_p^m/\mu\right)}{2} . \tag{15}$$

R update Following a similar strategy to Eq. (15) the element-wise updates for $\mathbf{R}$ are derived as

$$r_p^m = \frac{\left(e_p^m - y_p^m + q_p^m + \lambda_p^m/\mu + \max\left(\mathbf{u}_p^m\right) - \omega_p^m/\mu\right)}{2} . \tag{16}$$

T update Keeping terms in Eq. (7) containing $\mathbf{T}$ and decoupling across K and P gives the following

$$\mathbf{t}_p^m = \arg\min_{\mathbf{t}_p^m}\left(q_p^m - \max\left(\mathbf{t}_p^m\right) + \sigma_p^m/\mu\right)^2 + \left\|\mathbf{t}_p^m - \left(\mathbf{w}_m^T\mathbf{X}_p + \mathbf{1}b_m\right) + \theta_p^m/\mu\right\|_2^2 , \tag{17}$$

which can be further decoupled element-wise for each $t_{p,j}^m \in \mathbf{t}_p^m$ giving $K \times P \times (n_1 + \cdots + n_P)$ problems

$$t_{p,j}^m = \arg\min_{t_{p,j}^m}\begin{cases}\left(q_p^m - t_{p,j}^m + \sigma_p^m/\mu\right)^2 + \left(t_{p,j}^m - \phi_{p,j}^m\right)^2 \\ \text{when } t_{p,j}^m = \max\left(\mathbf{t}_p^m\right), \\ \left(t_{p,j}^m - \phi_{p,j}^m\right)^2 \text{ else} ,\end{cases} \tag{18}$$

Model	Precision	Recall	F_1-score	Accuracy
k-NN	0.846±0.027	0.936±0.060	0.872±0.023	0.875±0.017
XGBoost	0.769±0.068	0.932±0.069	0.819±0.068	0.814±0.064
LightGBM	0.728±0.033	**0.945±0.038**	0.800±0.036	0.784±0.035
SVM	0.837±0.025	0.941±0.039	0.867±0.048	0.873±0.029
MISVM	0.854±0.078	0.827±0.085	0.822±0.052	0.837±0.049
SimMISVM	**0.900±0.032**	0.862±0.034	**0.872±0.012**	**0.884±0.015**

Table 1: Identifying COVID-19 clinical outcomes within the first twenty-four hours of patient admission. Average performance and standard deviations for each metric are calculated across a six-fold cross validation experiment.

where $\boldsymbol{\phi}_p^m = \mathbf{w}_m^T \mathbf{X}_p + \mathbf{1}b_m - \boldsymbol{\theta}_p^m/\mu$. Taking the derivative of Eq. (18) with respect to $t_{p,j}^m$, setting the result equal to zero, and solving for $t_{p,j}^m$, gives the updates

$$t_{p,j}^m = \begin{cases} \frac{\max(\boldsymbol{\phi}_p^m) + q_p^m + \sigma_p^m/\mu}{2} & \text{if } j = \arg\max(\boldsymbol{\phi}_p^m) \\ \phi_{p,j}^m & \text{else} \end{cases} . \quad (19)$$

U update Following the steps used to derive Eq. (19) the element-wise updates of $\mathbf{U}$ are derived as

$$u_{p,j}^m = \begin{cases} \frac{\max(\boldsymbol{\psi}_p^m) + r_p^m + \omega_p^m/\mu}{2} & \text{if } j = \arg\max(\boldsymbol{\psi}_p^m) \\ \psi_{p,j}^m & \text{else} \end{cases} , \quad (20)$$

where $\boldsymbol{\psi}_p^m = \mathbf{w}_y^T \mathbf{X}_p + \mathbf{1}b_y - \boldsymbol{\xi}_p^m/\mu$. *This completes the primal updates for Algorithm 1.* The final three primal updates are for Algorithm 2.

X update The update for $\mathbf{X}$ is decoupled across column blocks associated with the p-th patient. Since some patients have labels and others do not we have two sets of minimization problems. First, are the L sub-problems for each patient with labels

$$\begin{aligned} \mathbf{X}_{p(l)} = \arg\min_{\mathbf{X}_{p(l)}} & \left\| \mathbf{F}_p - \mathbf{X}_p + \boldsymbol{\Pi}_p/\mu \right\|_F^2 \\ & + \sum_{m=1}^{K} \left[\left\| \mathbf{t}_p^m - \left(\mathbf{w}_m^T \mathbf{X}_p + \mathbf{1}b_m \right) + \boldsymbol{\theta}_p^m/\mu \right\|_2^2 \right. \\ & \left. + \left\| \mathbf{u}_p^m - \left(\mathbf{w}_y^T \mathbf{X}_p + \mathbf{1}b_y \right) + \boldsymbol{\xi}_p^m/\mu \right\|_2^2 \right] \\ & + \left\| \mathbf{Z}_p - \left(\mathbf{X}_p + \mathbf{S}_p \right) + \boldsymbol{\Delta}_p/\mu \right\|_F^2 . \end{aligned} \quad (21)$$

Second, are the U problems associated with the unlabeled patients in $\mathbf{Z}$

$$\begin{aligned} \mathbf{X}_{p(u)} = \arg\min_{\mathbf{X}_{p(u)}} & \left\| \mathbf{F}_p - \mathbf{X}_p + \boldsymbol{\Pi}_p/\mu \right\|_F^2 \\ & + \left\| \mathbf{Z}_p - \left(\mathbf{X}_p + \mathbf{S}_p \right) + \boldsymbol{\Delta}_p/\mu \right\|_F^2 . \end{aligned} \quad (22)$$

Taking the derivatives of Eq. (21) and Eq. (22) with respect to $\mathbf{X}_p$, setting the result equal to zero, and solving for the corresponding $\mathbf{X}_p$ gives the updates

$$\begin{aligned} \mathbf{X}_{p(l)} = & \left(2\mathbf{I} + \textstyle\sum_{m=1}^{K} \mathbf{w}_m \mathbf{w}_m^T + K \mathbf{w}_y \mathbf{w}_y^T \right)^{-1} * \left(\mathbf{F}_p + \boldsymbol{\Pi}_p/\mu \right. \\ & + \mathbf{Z}_p - \mathbf{S}_p + \boldsymbol{\Delta}_p/\mu + \textstyle\sum_{m=1}^{K} [\mathbf{w}_m (\mathbf{t}_p^m - \mathbf{1}b_m + \boldsymbol{\theta}_p^m/\mu) \\ & \left. + \mathbf{w}_y (\mathbf{u}_p^m - \mathbf{1}b_y + \boldsymbol{\xi}_p^m/\mu)] \right) , \end{aligned} \quad (23)$$

for the patients with labels and

$$\mathbf{X}_{p(u)} = \frac{\mathbf{F}_p + \boldsymbol{\Pi}_p/\mu + \mathbf{Z}_p - \mathbf{S}_p + \boldsymbol{\Delta}_p/\mu}{2} , \quad (24)$$

for patients without labels.

F Update Keeping terms in Eq. (7) that contain $\mathbf{F}$ gives

$$\min_{\mathbf{F}} \ \alpha \left\| \mathbf{F} \right\|_* + \frac{\mu}{2} \left\| \mathbf{F} - \mathbf{X} + \boldsymbol{\Pi}/\mu \right\|_F^2 , \quad (25)$$

which can be solved via the soft-thresholding operation [4] on the singular values

$$\mathbf{F} = \hat{\mathbf{U}} \operatorname{diag}\left((\hat{\sigma} - \alpha/\mu)_+ \right) \hat{\mathbf{V}}^T , \quad (26)$$

where $svd(\mathbf{X} - \boldsymbol{\Pi}/\mu) = \{\hat{\mathbf{U}}, \hat{\Sigma}, \hat{\mathbf{V}}^T\}$ and $\hat{\sigma}$ are the singular values along $\hat{\Sigma}$.

S Update Keeping terms in Eq. (7) that contain $\mathbf{S}$ gives

$$\mathbf{S} = \arg\min_{\mathbf{S}} \ \beta \left\| \mathcal{P}_\Omega(\mathbf{S}) \right\|_1 + \frac{\mu}{2} \left\| \mathbf{Z} - \mathbf{X} - \mathbf{S} + \boldsymbol{\Delta}/\mu \right\|_F^2 \quad (27)$$

which, following [5] is updated by

$$s_n^d = \begin{cases} m_n^d & \text{if } \mathcal{P}_\Omega(z_n^d) \text{ is missing} \\ m_n^d - \frac{\beta}{\mu} & \text{if } \mathcal{P}_\Omega(z_n^d) \text{ is available and } m_n^d > \frac{\beta}{\mu} \\ 0 & \text{if } \mathcal{P}_\Omega(z_n^d) \text{ is available and } \left| m_n^d \right| \leq \frac{\beta}{\mu} \\ m_n^d + \frac{\beta}{\mu} & \text{if } \mathcal{P}_\Omega(z_n^d) \text{ is available and } m_n^d < -\frac{\beta}{\mu} \end{cases} \quad (28)$$

where $m_n^d \in \mathbf{M} = \mathbf{Z} - \mathbf{X} + \boldsymbol{\Delta}/\mu$.

3 EXPERIMENTS & RESULTS

We compare our method against an array of statistical learning techniques that have recently been used to predict COVID-19 clinical outcomes followed by a discussion of identified biomarkers.

3.0.1 Data. We obtained the clinical data and associated outcomes for 375 COVID-19 cases included in Yan *et al.* [23]. Patients without timestamped clinical observations were removed. The remaining data were then normalized by feature and a missing data mask was calculated for each patient. The final dataset included 73 features derived from blood tests across an average of ≈ 16.9 observations for 361 patients of which 195 survived and 166 died. The average age of patients in our dataset was ≈ 58.9 years where 205 patients were between 33-65 years and 156 patients were 65 years and older. The proportion of missing data was ≈ 87.6%.

3.0.2 Experiment settings. We compared our method to k-nearest neighbors (*k-NN*), gradient boosted trees with the *XGBoost* [8], *LightGBM* [12] libraries, a linear support vector machine *SVM* implemented in LIBSVM [7], and our implementation in Algorithm 1 of a multi-instance support vector machine (*MISVM*) as a baseline. For the compared methods we handled missing data by following a similar approach to [23] the most recent observation available was used at prediction time. The hyperparameters for the *SimMISVM* method are $C = 10, \alpha = 10^{-2}, \beta = 10^{-2}, \mu = 10^{-4}$, searched over $[10^p : p \in \{-5, \ldots, 5\}]$ for each parameter. Competing methods and grid-search codes and implemented using the MLJ library [2].

3.0.3 Classification performance. In Table 1, we report the performance of our method in predicting COVID-19 clinical outcomes on the Tongji Hospital data. In each case, the models are provided with all clinical data available during training, while at test-time the models were only provided with clinical data from the first twenty-four hours. Table 1 shows that our method has higher precision, F_1-score, and accuracy than the compared methods. Our method also shows improvement over the baseline *MISVM* method. In Figure 2,

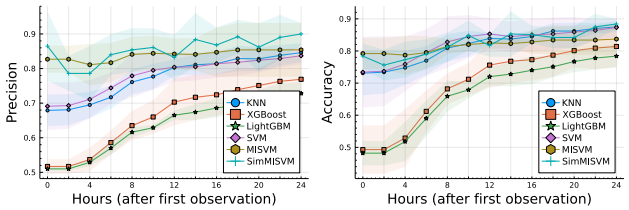
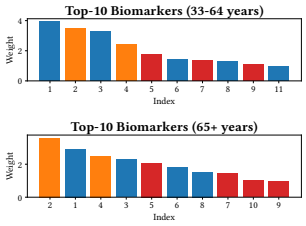


Figure 2: Precision and accuracy results of the compared methods when provided with patient readings every two hours after the first patient data is collected. The width of the ribbons for each method represent the standard deviations across the six-fold cross validation experiment at that time.



Index	Biomarker	*Function*
1	total bilirubin	liver
2	neutrophils(%)	immune
3	direct bilirubin	liver
4	lymphocyte(%)	immune
5	total protein	blood
6	albumin	liver
7	mean corpuscular hemoglobin	blood
8	globulin	liver
9	mean corpuscular volume	blood
10	mean corpuscular hemoglobin conc.	blood
11	indirect bilirubin	liver

Figure 3: Top-10 biomarkers predictive of a clinical COVID-19 outcome identified by the proposed *SimMISVM* method. Weight (y-axis) is derived from the absolute *row*-sum of W.

we show how the performance metrics of the compared methods change as the number clinical observations provided increase. The far-left side of the two panels in Figure 2 highlight that both multi-instance approaches provide increased performance with limited clinical data. This may be due to the fact that our method, since it operates on the *instance* level, can identify trends in previous clinical data which can generalize to new patients early in their hospital stay.

3.0.4 Biomarker identification. In addition to improved predictive performance our method can be analyzed to identify biomarkers from the Tongji hospital data that is discriminative of a fatal COVID-19 outcome. In Figure 3, we show the top-10 biomarkers identified by our approach across two patient cohorts. Liver function, including bilirubin, albumin, and globulin are studied in [13] and were found to be predictive of a serious COVID-19 infection by our approach. Additionally, a high neutrophil to lymphocyte ratio, two biomarkers also identified by our model, have been found to predict mortality [14] for critically ill COVID-19 patients. Finally, [9] also report that higher levels of mean corpuscular volume and hemoglobin were higher in general COVID-19 cases. These identified biomarkers may provide additional insights into COVID-19 mortality and warrant further investigation.

4 CONCLUSION

This work presents a novel *Simultaneous Imputation-Multi Instance Support Vector Machine* approach applied to COVID-19 clinical outcome prediction. Our method shows improved prediction early in the progression of the disease and identifies clinical biomarkers that are validated in current literature; this demonstrates the utility of multi-instance learning techniques for clinical outcome prediction.

ACKNOWLEDGMENTS

Corresponding author: Hua Wang (huawangcs@gmail.com). This work was supported in part by the National Science Foundation under the grant of CCF 2029543.

REFERENCES

[1] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann. 2003. Support vector machines for multiple-instance learning. In *Advances in neural information processing systems*. 577–584.
[2] Anthony D. Blaom, Franz Kiraly, Thibaut Lienart, Yiannis Simillides, Diego Arenas, and Sebastian J. Vollmer. 2020. MLJ: A Julia package for composable machine learning. *Journal of Open Source Software* 5, 55 (2020), 2704. https://doi.org/10.21105/joss.02704
[3] Stephen Boyd, Neal Parikh, and Eric Chu. 2011. *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc.
[4] Jian-Feng Cai, Emmanuel J Candès, and Zuowei Shen. 2010. A singular value thresholding algorithm for matrix completion. *SIAM Journal on optimization* 20, 4 (2010), 1956–1982.
[5] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. 2011. Robust principal component analysis? *Journal of the ACM (JACM)* 58, 3 (2011), 1–37.
[6] Marc-André Carbonneau, Veronika Cheplygina, Eric Granger, and Ghyslain Gagnon. 2018. Multiple instance learning: A survey of problem characteristics and applications. *Pattern Recognition* 77 (2018), 329–353.
[7] Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)* 2, 3 (2011), 1–27.
[8] Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 785–794.
[9] Dodji Kossi Djakpo, Zhiquan Wang, Rong Zhang, Xin Chen, Peng Chen, and Malyn Martha Lilac Ketisha Antoine. 2020. Blood routine test in mild and common 2019 coronavirus (COVID-19) patients. *Bioscience Reports* 40, 8 (2020).
[10] Ürün Dogan, Tobias Glasmachers, and Christian Igel. 2016. A Unified View on Multi-class Support Vector Classification. *J. Mach. Learn. Res.* 17, 45 (2016), 1–32.
[11] James D Dziura, Lori A Post, Qing Zhao, Zhixuan Fu, and Peter Peduzzi. 2013. Strategies for dealing with missing data in clinical trials: from design to analysis. *The Yale journal of biology and medicine* 86, 3 (2013), 343.
[12] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in neural information processing systems*. 3146–3154.
[13] Praveen Kumar-M, Shubhra Mishra, Daya Krishna Jha, Jayendra Shukla, Arup Choudhury, Ritin Mohindra, Harshal S Mandavdhare, Usha Dutta, and Vishal Sharma. 2020. Coronavirus disease (COVID-19) and the liver: a comprehensive systematic review and meta-analysis. *Hepatology international* (2020), 1–12.
[14] Mireille Laforge, Carole Elbim, Corinne Frère, Miryana Hémadi, Charbel Massaad, Philippe Nuss, Jean-Jacques Benoliel, and Chrystel Becker. 2020. Tissue damage from neutrophil-induced oxidative stress in COVID-19. *Nature Reviews Immunology* 20, 9 (2020), 515–516.
[15] Kai Liu, Hua Wang, Feiping Nie, and Hao Zhang. 2018. Learning multi-instance enriched image representations via non-greedy ratio maximization of the L1-norm distances. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7727–7735.
[16] Xuedi Ma, Michael Ng, Shuang Xu, Zhouming Xu, Hui Qiu, Yuwei Liu, Jiayou Lyu, Jiwen You, Peng Zhao, Shihao Wang, et al. 2020. Development and validation of prognosis model of mortality risk in patients with COVID-19. *Epidemiology & Infection* 148 (2020).
[17] Feiping Nie, Yizhen Huang, and Heng Huang. 2014. Linear time solver for primal SVM. In *International Conference on Machine Learning*. 505–513.
[18] Hua Wang, Heng Huang, Farhad Kamangar, Feiping Nie, and Chris Ding. 2011. Maximum margin multi-instance learning. *Advances in neural information processing systems* 24 (2011), 1–9.
[19] Hua Wang, Feiping Nie, and Heng Huang. 2011. Learning instance specific distance for multi-instance classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 25.
[20] Hua Wang, Feiping Nie, and Heng Huang. 2012. Robust and discriminative distance for multi-instance learning. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2919–2924.
[21] Junxiang Wang and Liang Zhao. 2017. Nonconvex Generalization of ADMM for Nonlinear Equality Constrained Problems. *arXiv preprint arXiv:1705.03412* (2017).
[22] Jason Weston, Chris Watkins, et al. 1999. Support vector machines for multi-class pattern recognition.. In *Esann*, Vol. 99. 219–224.
[23] Li Yan, Hai-Tao Zhang, Jorge Goncalves, Yang Xiao, Maolin Wang, Yuqi Guo, Chuan Sun, Xiuchuan Tang, Liang Jing, et al. 2020. An interpretable mortality prediction model for COVID-19 patients. *Nature Machine Intelligence* (2020), 1–6.