Predicting COVID-19 Severity Using a Cut-and-Solve Feature Selection Approach

Kenneth Smith
Department of Computer Science
University of Missouri - St. Louis
Saint Louis, USA
kpsc59@umsystem.edu

Michael Chan
Optimization
GotSport
Jacksonville Beach, USA
0000-0002-7991-2074

John Brandenburg
Stord
brandenburgjohn@gmail.com

Katarina A Jones
Department of Chemistry
University of Tennessee Knoxville
Knoxville, USA
kjone166@vols.utk.edu

Shawn R. Campagna
Department of Chemistry
The University of Tennessee
Knoxville, USA
campagna@utk.edu

Michael Garvin
Computational Predictive Biology
Oak Ridge National Laboratory
Oak Ridge, USA
garvinmr@ornl.gov

Alan R. Templeton
Department of Biology
Washington University
Saint Louis, USA
temple_a@wustl.edu

Daniel Jacobson
Biosciences Division
Oak Ridge National Laboratory
Oak Ridge, USA
jacobsonda@ornl.gov

Carlos Cruchaga
Department of Psychiatry
Washington University School of Medicine
Saint Louis, USA
cruchagac@wustl.edu

Sharlee Climer
Department of Computer Science
University of Missouri - St. Louis
Saint Louis, USA
climer@umsl.edu

***Abstract*—Individuals with coronavirus disease 2019 (COVID-19) infection present in a variety of ways, ranging from asymptomatic or mild cough, to organ failure or death. One of the major challenges for the medical community is the quick and accurate determination of how COVID-19 will progress in an individual. Herein, we introduce a new Cut-and-Solve based feature selection program for identifying predictive feature sets in heterogeneous data. We analyze proteomics data from Washington University to identify models ranging in size from a single feature up to five. Validation of logistic regression models using area under the curve (AUC) were applied for both a hold-out data set and an independent data set from Massachusetts General Hospital. A variety of known and novel biomarkers for COVID-19 severity were identified. The best model for predicting severe (ventilation or death) vs. non-severe infection is achieved for *CALCOCO2* and *STC1*, with an average AUC=0.81. Based on the known severity markers, several different proteomic pathways are identified. Enrichment analysis indicates activity associated with inflammatory response, as well as myelination and cardiac function.**

***Index Terms*—Feature Selection, COVID-19, Mixed Integer Programming**

I. Introduction

Due to coronavirus disease 2019 (COVID-19) prevalence, two of the major challenges placed on the medical community are identifying COVID-19 infected individuals and predicting the severity of symptoms one would develop. These challenges are exasperated by the variety of symptoms caused by the disease, ranging from asymptomatic individuals, to mild cases (e.g. dry cough, body aches, and olfactory dysfunction), all the way to critical cases (e.g. renal failure, blood clotting, and cytokine storm) [1]. Several research studies focused on identification of COVID-19 infection using plasma based proteomics [2, 3], while others used plasma data to predict disease severity [2, 4, 5, 6, 7] or cardiovascular complications [8, 9]. In addition to providing tools for precision treatment, analysis of omics data may help identify drug targets for new therapies [10].

A wide variety of techniques have been used to study plasma data for COVID-19, including regression [4, 6], linear discriminant analysis [5], random forest [2], genetic algorithms [2], differential analysis [8], along with an ensemble of techniques [3]. Most of these approaches aim to find a single model that accurately predicts severity or infection. However, due to the high level of heterogeneity in COVID-19 symptoms, a single model may not capture all of the sub-types of symptoms present in the population.

Our recent feature selection algorithm, called *Frequency Based Pruning* (FBP), identifies multiple sets of continuous valued features associated with a disease [11]. These feature sets are initially identified based on the Youden J statistic and validated based on the Area Under the Receiver/Operator Curve (AUC). By identifying multiple feature sets associated with a disease, instead of constructing a single model, FBP aims to capture the heterogeneous pathologies of a disease. FBP searches for combinations of features associated with a

National Science Foundation (CNS-1429294)

979-8-3503-3748-8/23/$31.00 ©2023 IEEE

disease by pruning away sections of the search space based on the frequency of smaller combinations. This approach is highly distributed and efficient for small combinations, but quickly increases in run time with the cardinality of the combination.

A similar program, called Operations Research for Combinatorial Associations (ORCA) [12, 13], was developed to find genetic patterns associated with a disease. ORCA finds the pattern which optimizes the Youden J statistic by solving a Mixed Integer Program (MIP) using a Cut-and-Solve (CNS) approach [14]. CNS is an iterative search strategy that solves a MIP by repeating four steps until convergence. First, a relaxed version of the MIP is solved. Second, based on the solution to the relaxation problem, a cut is generated, which bifurcates the solution space into a large partition and small partition. Third, the small partition (sparse problem) is optimally solved, providing solutions for maintaining an incumbent solution. Fourth, a constraint is added to the model, removing the small partition from future searches. Once the objective value of the relaxation solution is less than the incumbent solution, the programs has converged and the incumbent solution is guaranteed to be optimal. ORCA has two variants, one in which the sparse problem is solved as a MIP, and the other in which all feasible solutions are enumerated.

In this manuscript we present a CNS feature selection (CSFS) approach for continuous valued data. We expand upon ORCA by incorporating the discretization and solution pool methods of FBP, along with adding additional constraints to remove sub-optimal portions of the search space. We evaluate our method by predicting COVID-19 severity using plasma proteomics, and replicate our results twice, first with held-out data and second with an independent data set.

II. Methods

A. Pre-processing

We performed discovery and initial validation for COVID-19 severity using a cohort of 324 COVID-19 patients from Barnes-Jewish Hospital and data produced by Washington University in St. Louis (WU). Independent validation was performed using a cohort of 306 COVID-19 patients from Massachusetts General Hospital (MGH) [6]. WU proteomics were generated using the SomaScan v4.1 7K panel, while MGH proteomics were generated using the SomaScan 5K panel. Both data sets were cleaned using DataRetainer [15, 16], until each remaining individual and feature had at most 10% missing data. After quality control, 4634 proteins remained in common between the two panels. Since multiple samples were collected for several patients in both cohorts, the earliest samples, labeled 'Day 0' were used.

To test for COVID-19 severity, we categorized the WU cases as severe if they were placed in the ICU, ventilated, or died due to COVID-19. All other cases were considered non-severe. Any MGH case which was ventilated, intubated, or died withing 28 days was considered severe, and all other cases were non-severe. Both groups of individuals were classified based on their most severe status during the study.

After cleaning the data sets, we split the WU data into discovery (70%) and hold-out (30%) data sets. The discovery and hold-out data sets were scaled using the mean and interquartile range (IQR) of each feature in the WU discovery data set. Since the MGH data was already scaled, we selected a random subset of individuals from MGH that matched the class distribution of the WU discovery. Using the mean and IQR of this subset, we rescaled the entire MGH set.

To select features, the continuous data was converted to discrete variables. We performed quantile discretization on each feature, categorizing the top 30% of samples as high, the bottom 30% as low, and the remaining 40% as neutral. From these discretized values, we created a binary matrix according to [11]. Table I shows how each categorical value (high, neutral, low, or missing) was encoded using two rows.

TABLE I
Analyte Encoding

Binned Analyte Level	High	Neutral	Low	Missing
High Expression	1	0	0	1
Low Expression	0	0	1	1

B. Formulating the MIP

Like FBP and ORCA, we measure the association strength of a pattern based on the Youden J statistic. The J statistic measures the difference in the classification rate between two groups, G_1 and G_2. We use the vector ind to indicate how the individuals are classified for each pattern. Using n_1 and n_2 to indicate the number of individuals in G_1 and G_2, respectively, we write J in terms of the ind variables as:

$$J = \frac{1}{n_1} \sum_{j \epsilon G_1} ind_j - \frac{1}{n_2} \sum_{j \epsilon G_2} ind_j \qquad (1)$$

The binary encoded values from Section II-A are called marker states. We indicate the marker states in a pattern with the vector m, where $m_i = 1$ indicates that marker state i is in the pattern and $m_i = 0$ indicates that marker state i is not in the pattern. Each pattern is made up of a combination of marker states and the number of marker states in a pattern is called the pattern size (PS). We treat the PS as a constant and solve the MIP repeatedly for different PS values. By adding the following constraint, where M is the number of marker states in the problem, we restrict our search to a specific PS:

$$\sum_{i=1}^{M} m_i = PS \qquad (2)$$

The optimizer should set $ind_j = 1$ if the individual contains all of the marker states in the pattern, and to 0 otherwise. Let $\mathbf{B}_{M \times N}$ be the binary matrix encoded in Section II-A, where M is the number of marker states and N is the number of individuals. To force individuals with the full pattern to 1, a lower bound constraint is added for each ind_j.

$$ind_j \geq \sum_{i=1}^{M} \mathbf{B}_{ij} m_i - PS + 1 \quad \forall j \qquad (3)$$

From Eq. (2), we know the maximum value of the sum in Eq. (3) is PS. This occurs when individual j contains all of the marker states in the pattern. The resulting *rhs* will equal 1 and the individual will be correctly set. If individual j is missing any of the marker states in the pattern, and the marker states are integer values, the *rhs* will be non-positive, allowing ind_j to be set to 0.

An individual should be assigned a value of 1 only if it contains all of the marker states in the pattern. To accomplish this, we add an upper bound constraint on each ind_j.

$$ind_j \leq \frac{1}{PS}\sum_{i=1}^{M}\mathbf{B_{ij}}m_i \quad \forall j \tag{4}$$

The maximum value of the sum in Eq. (4) is PS, which results in a *rhs* value of 1. If individual j is missing one or more of the marker states in the pattern, the *rhs* is strictly less than 1. Finally, we want ind and m to be binary vectors. By adding integer constraints to both vectors we arrive at the MIP formulation below. As noted in [13], the lower bound constraint need only be applied to $j\epsilon G_2$ and upper bound constraints need only be applied to $j\epsilon G_1$.

$$\underset{ind,m}{\text{maximize}} \quad J = \frac{1}{n_1}\sum_{j\epsilon G_1} ind_j - \frac{1}{n_2}\sum_{j\epsilon G_2} ind_j \tag{5a}$$

$$\text{subject to} \quad \sum_{i=1}^{M} m_i = PS, \tag{5b}$$

$$ind_j \geq \sum_{i=1}^{M}\mathbf{B}_{ij}m_i - PS + 1 \quad \forall j\epsilon G_2, \tag{5c}$$

$$ind_j \leq \frac{1}{PS}\sum_{i=1}^{M}\mathbf{B_{ij}}m_i \quad \forall j\epsilon G_1, \tag{5d}$$

$$m_i, ind_j \;\epsilon\; \{0,1\} \;\forall\; i,j \tag{5e}$$

C. Implementing Sync Cut-and-Solve

CSFS can be operated in to modes: 1) find an optimal solution, and 2) create a solution pool. The solution pool is a collection of optimal and near optimal solutions. When operating in solution pool mode, a lower bound (LB) needs to be provided and is not updated during the CSFS execution. In this manuscript, we operate in the solution pool mode and use the greedy algorithm [11] to provide a LB for all PS≥3 trials.

CSFS uses a controller-worker model. The controller is responsible for solving the relaxation problem, generating the cut, sending the sparse problem to the worker, and receiving solutions from the worker. The worker receives and solves the spare problem, sending back any solutions. Since solving the relaxation problem and generating a cut typically takes much less time than a solving sparse problem, CSFS is able to distribute the multiple sparse problems to decrease run time. Next, we detail the relaxation problem, cut generation, and sparse problem.

1) The Relaxation Problem: To create the relaxation problem, we begin by dropping the integer constraints, Eq. (5e), resulting in a linear program. Additional constraints will be added based on the cuts from Section II-C2, which remove small regions of the search space from the relaxation problem as they are searched in sparse problems. The basic MIP can also be improved upon by removing unnecessary variables.

The number of variables in the relaxation problem can be reduced in three ways: 1) a marker state is removed, 2) an individual is removed, 3) and individual is set equal to another individual. As shown in [11], the J value for any pattern is bounded above by the percentage of G_1 members that contain any subset of marker states in the pattern. Any marker state, for which the percentage of G_1 members containing it is less than LB, can be removed. This check is performed every time a new LB is found or an individual is removed. An individual is removed whenever the number of remaining marker states it contains is less than PS or when all of the remaining marker states are contained in a cut from Section II-C2. Finally, when two individuals contain the same remaining marker states, we can force the two ind variables to take on the same value. If any of these variable reductions occur in the relaxation problem, they can also be applied to any subsequent sparse problems. After performing all variable reduction checks, the relaxation problem is solved and a piercing cut, that bifurcates the solution space, is generated

2) Generating Cuts: Three types of piercing cuts were identified in [13]: 1) from a relaxed solution, 2) from an individual, and 3) from merging previous cuts. All three cuts can be defined by a set of marker states, K. To create a cut from the relaxed solution, let x^* be an optimal relaxation solution and let $K = \{i|x^*_i > 0\}$. A cut from individual j is generated by letting $K = \{i|\mathbf{B}_{ij} = 1\}$. As the set of cuts generated from the relaxation solutions or individuals grows, it was useful to manage them by merging or deleting cuts. A cut is deleted whenever all of the marker states are contained within another cut. Two cuts are merged by taking the union of the marker states in the two cuts. When cuts are merged, the old cuts can be deleted from the problem. Regardless of how a cut is generated, the following constraint is added to the relaxation problem after using the cut to create a sparse problem. This constraint removes all patterns constructed entirely from markers in the cut.

$$\sum_{k\epsilon K} x_k \leq PS - 1$$

3) The Sparse Problems: The sparse problem begins by adding all of the constraints from the relaxation problem, including ones that removed variables and ones generated from cuts. Next the integer constraints are added, along with the following constraint generated from the most recent piercing cut.

$$\sum_{k\epsilon K} x_k \geq PS$$

Any marker states not in K are forced to 0. We also remove any individuals that contain less than PS of the marker states in

K. Finally, we limit our search to solutions with an objective value $\geq$ LB. In CPLEX, a collection of optimal and near optimal solutions is generated using a solution pool. We limited the solution pool size to 1000 for each sparse problem.

D. Evaluating Models

To measure the quality of CSFS feature sets, we fit a logistic regression model on the real valued WU discovery data, for each pattern in the solution pool. These models were then subject to the same criteria as [11], including AUC and p-value filters, 3-fold cross-validation, and 100 permutation trials. Models which passed these checks were then evaluated on the WU hold-out and MGH data sets separately, and needed an $AUC \geq 0.7$ and Benjamini–Hochberg corrected p-values < 0.05 in both sets to be considered validated.

Models that validated in both sets were analyzed further to find hub analytes. A maker state is either a high or low expression of an analyte. For each PS, the unique analytes in validated models were determined. Then, using Fisher's exact test and Benjamini–Hochberg correction, we determined which analytes appeared more often than expected. These were considered the hub analytes.

III. Results

The greedy algorithm from [11] was used to construct the PS=1 and PS=2 solution pools and provided LBs for the PS$\geq$ 3 CSFS trails. Both algorithms were executed on the UM Lewis Cluster using 12 processors. The CPU time of the main processor is shown in Table II. CSFS was implemented using IBM ILOG CPLEX 12.7.0 and Open MPI 3.1.3.

TABLE II
Run Time

PS	Greedy Run Time (s)	CSFS Run Time (s)
1	< 1	NA
2	194	NA
3	10	9322
4	13	40678
5	15	74592

Table III lists the number of valid models, unique analytes, and hub analytes, for each PS. The number of validated models for PS ≥ 3 was smaller than expected, due in large part to how CPLEX populates the solution pool. The pool for each sparse problem was filled with many fractional solutions. To account for CPLEX's integer tolerance, we applied a rounding step to the sparse solutions. This reduced the full solution pool down to a few dozen unique integer solutions. We tried enlarging the solution pool limit to 10000 for each sparse problem, but observed only a few additional patterns with a large increase in run time. However, much can still be gleaned from the CSFS results.

TABLE III
Number of Validated Models and Analytes

PS	# Models	# Analytes	# Enriched Analytes
1	22	22	NA
2	1822	411	46
3	128	92	10
4	101	97	11
5	59	77	8

Proteins from validated PS=1 patterns include *IL1RL1*, *SF1*, *RBFOX2*, *TEAD4*, *ARHGAP36*, *RBP1*, *DLL4*, *STC1*, *ANGPTL4*, *EDA2R*, *KRT1*, *SFN*, *PLA2G2A*, *TNFSF15*, *PCDHAC2*, *TNNT2*, *LST1*, *NBL1*, *CALCOCO2*, *CHCHD10*, *TMX3*, and *H3C1*. *RBFOX2* proved to be the best single-feature predictor with an average AUC=0.79. Previous research indicates associations between COVID-19 severity and *ANGPTL4* [17], *IL1RL1* [18], and *PLA2G2A* [19]. Based on the findings of Li [20], CSFS identified several distinct COVID-19 proteomic pathways. Fold Change (FC) analysis identified 16 differential expressed gene (DEGs) in the WU data set, 6 of which present in validated CSFS models. The remaining 10 DEGs were identified by CSFS, but failed the AUC test in either the discovery of hold-out data sets.

Table IV shows the 67 PS=2 models that achieved higher AUCs than the best PS=1 model. The top performing model of *CALCOCO2* and *STC1* averaged an AUC=0.81 between the 3 data sets. Both of the analytes identified in this model were present in validated PS=1 models. Most of the PS=2 models (47) contained one analyte from a PS=1 model, while 16 contained no PS=1 analyte. In fact, the second best PS=2 model of *IGFALS* and *PCDHGA10*, contained no PS=1 analytes. Some of the new proteins from PS=2 patterns include known prognostic markers *MB* [21] and *IGFALS* [21], along with *TTN*, and *SOD2*. Enrichment analysis for the 46 hub proteins using STRING-DB indicated activity in the regulation of myelination, cardiac function, death receptors, and a large number of TNFs. Additional analysis using STRING-DB, on the analytes in Table IV, indicate a strong presence of proteins associated with cardiovascular disease [22, 23, 24].

AUC values for PS=3, PS=4, PS=5 models dropped slightly compared to PS=2, indicating possible over fitting. However, a core group of hub nodes emerged in these three trials: *SF1*, *LGALS2*, *EFNA4*, *PCDHGA12*, *TEAD4*, *EVPL*, *PCDHAC2*, *TNFSF15*. These analytes are often up-regulated together in severe patients, indicated a high level of correlation. They are also decent predictors of severity themselves. Intestingly, STRING-DB indicated no know association among these proteins.

IV. Conclusion

In this manuscript, we introduce a new efficiently distributed Cut-and-Solve based feature selection algorithm called CSFS. Our program identifies candidate feature sets by collecting a pool of optimal and near optimal solutions, based on maximizing Youden J. These candidate sets are used to train logistic regression models that were validated using AUC on a hold-out set and an independent data set. Validated models were used to identify hub proteins and perform enrichment analysis.

TABLE IV
PS=2 PATTERNS WITH AUC ≥ 0.79

Feature 1	Feature 2	AUC	# Features in PS=1
CALCOCO2	STC1	0.81	2
IGFALS	PCDHGA10	0.81	0
TTN	PLA2G2A	0.80	1
CALCOCO2	FSTL3	0.80	1
TFF3	PCDHAC2	0.80	1
RBP1	ANGPTL4	0.80	2
P4HA1	ANGPTL4	0.80	1
IL1RL1	DAG1	0.80	1
PARVA	ANGPTL4	0.80	1
BAG3	TNFSF15	0.80	1
LGALS2	ANGPTL4	0.80	1
TMX3	S100A12	0.80	1
TNFRSF1B	PCDHAC2	0.80	1
TTN	PSMA1	0.80	0
PTH1R	PCDHAC2	0.80	1
BMP10	ASGR1	0.80	0
MYL3	ANGPTL4	0.80	1
TNNT2	AFP	0.80	1
ARL5A	PCDHGA12	0.79	0
DAG1	PCDHAC2	0.79	1
CALCOCO2	MYL3	0.79	1
TTN	P4HA1	0.79	0
NID2	ANGPTL4	0.79	1
NET1	HTRA1	0.79	0
TFPI	ANGPTL4	0.79	1
BRD1	SF1	0.79	1
S100A2	ANGPTL4	0.79	1
CALCOCO2	IL1RN	0.79	1
RBFOX2	TFF3	0.79	1
DMBT1	TNFSF15	0.79	1
FABP3	TNNT2	0.79	1
PUF60	ANGPTL4	0.79	1
SF1	FSTL3	0.79	1
MDM1	PCDHAC2	0.79	1
HOPX	NBL1	0.79	1
SF1	CCN1	0.79	1
NBL1	CLSTN3	0.79	1
TNFSF15	TMX3	0.79	2
IGFALS	PCDHAC2	0.79	1
STC1	TFF3	0.79	1
CALCOCO2	KLK11	0.79	1
AHSG	TMX3	0.79	1
H3C1	ASGR1	0.79	0
FGFBP3	TMX3	0.79	1
DLL4	CPD	0.79	1
TMX3	NTN1	0.79	1
SMOC1	PCDHAC2	0.79	1
RBP1	ROR2	0.79	1
FSTL3	NLGN4X	0.79	0
PCSK2	PCDHAC2	0.79	1
HS6ST2	PCDHAC2	0.79	1
SF1	GDF15	0.79	1
RBL2	MB	0.79	0
NECTIN4	PCDHAC2	0.79	1
TTN	DCN	0.79	0
PARVA	TMX3	0.79	1
TTN	PLAAT3	0.79	0
TNFSF15	MB	0.79	1
RBFOX2	PCDH10	0.79	1
NTN1	PCDHAC2	0.79	1
SOD2	DAG1	0.79	0
SPP1	MB	0.79	0
SOD2	PGF	0.79	0
TNFRSF4	ANGPTL4	0.79	1
IGFALS	NEFL	0.79	0
TNFSF15	DLL4	0.79	2
TTN	SLC26A7	0.79	0

CSFS identified 22 analytes, in the PS=1 models, which predicted COVID-19 severity with an AUC>0.7 in all three data sets, including known severity biomarkers *ANGPTL4*,*IL1RL1*, and *PLA2G2A*, which are involved in different proteomic pathways. Only six of these proteins were identified through FC analysis. The best model for predicting severe (ventilation or death) vs. non-severe infection was achieved for *CALCOCO2* and *STC1*, with an average AUC=0.81. Additionally, 16 PS=2 models were constructed from proteins which were neither identified through FC analysis, nor had a J value > 0.15, indicating only the co-expression of these proteins was associated with severity. These findings illustrate the ability of CSFS to identify proteins associated with different pathways and the ability to identify proteins whose co-expression is associated with severity, even when each individual protein is not.

One major drawback of CSFS involves the CPLEX populate feature. Some of the MIP solutions above the LB were not identified due to the allocated solution pool size and duplicate solution produced by populate. Improvements to this approach may uncover better feature sets. Despite this limitation, CSFS was able to identify strong predictors in the heterogeneous COVID-19 severity data set, and replicate the results using an independent data set.

V. SOURCE CODE

Source code for the greedy algorithm and CSFS are available on GitHub. Greedy: https://github.com/ClimerLab/sync-greedy. CSFS: https://github.com/ClimerLab/CSFS.

VI. ACKNOWLEDGEMENTS

The computation for this work was performed on the high performance computing infrastructure provided by Research Computing Support Services and in part by the National Science Foundation under grant number CNS-1429294 at the University of Missouri, Columbia MO. DOI: https://doi.org/10.32469/10355/69802

REFERENCES

[1] Meredith Wadman et al. “A rampage through the body”. In: *Science* 368 (6489 Apr. 2020), pp. 356–360. ISSN: 0036-8075. DOI: 10.1126/science.368.6489.356.

[2] Rumi Iqbal Doewes, Rajit Nair, and Tripti Sharma. “Diagnosis of COVID-19 through blood sample using ensemble genetic algorithms and machine learning classifier”. In: *World Journal of Engineering* (2021). ISSN: 17085284. DOI: 10.1108/WJE-03-2021-0174.

[3] Lei Chen et al. “Identifying COVID-19-Specific Transcriptomic Biomarkers with Machine Learning Methods”. In: *BioMed Research International* 2021 (2021), pp. 1–11. ISSN: 2314-6133. DOI: 10 . 1155 / 2021 / 9939134.

[4] Álvaro Tamayo-Velasco et al. “Clinical Medicine HGF, IL-1α, and IL-27 Are Robust Biomarkers in Early Severity Stratification of COVID-19 Patients”. In: *J. Clin. Med* 10 (2021), p. 10. DOI: 10.3390/jcm10092017.

[5] Auriel A. Willette et al. “Using machine learning to predict COVID-19 infection and severity risk among 4510 aged adults: a UK Biobank cohort study”. In: *Scientific Reports* 12 (1 Dec. 2022). ISSN: 20452322. DOI: 10.1038/s41598-022-07307-z.

[6] Michael R. Filbin et al. “Longitudinal proteomic analysis of severe COVID-19 reveals survival-associated signatures, tissue-specific cell death, and cell-cell interactions”. In: *Cell Reports Medicine* 2 (5 May 2021). ISSN: 26663791. DOI: 10.1016/J.XCRM.2021.100287. URL: /pmc/articles/PMC8091031/%20/pmc/articles/PMC8091031/?report=abstract%20https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8091031/.

[7] Lihua Wang et al. “Plasma proteomics of SARS-CoV-2 infection and severity reveals impact on Alzheimer’s and coronary disease pathways”. In: *iScience* 26 (4 Apr. 2023), p. 106408. ISSN: 25890042. DOI: 10.1016/j.isci.2023.106408.

[8] Jason Roh et al. “Plasma Proteomics of COVID-19 Associated Cardiovascular Complications: Implications for Pathophysiology and Therapeutics”. In: *Research Square* (2021). DOI: 10.21203/RS.3.RS-539712/V1. URL: /pmc/articles/PMC8202429/%20/pmc/articles/PMC8202429/?report=abstract%20https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8202429/.

[9] Tao Guo et al. “Cardiovascular Implications of Fatal Outcomes of Patients with Coronavirus Disease 2019 (COVID-19)”. In: *JAMA Cardiology* 5 (7 July 2020), pp. 811–818. ISSN: 23806591. DOI: 10.1001/JAMACARDIO.2020.1017.

[10] Sharlee Climer. “COVID-19 and the differential dilemma”. In: *Patterns* 2 (5 May 2021). ISSN: 26663899. DOI: 10.1016/j.patter.2021.100260.

[11] Kenneth Smith, Jamie Lea, and Sharlee Climer. “Finding Single and Multi-Gene Expression Patterns for Psoriasis Using Sub-Pattern Frequency Pruning”. In: Institute of Electrical and Electronics Engineers Inc., 2021, pp. 2322–2329. ISBN: 9781665401265. DOI: 10.1109/BIBM52615.2021.9669803.

[12] John Brandenburg. “A Parallelized Method for Solving Large Scale Integer Linear Optimization Problems using Cut-and-Solve with Applications to cGWAS”. In: (2017).

[13] Michael Yip-hin Chan. “A Parallelized Implementation of Cut-and-Solve and a Streamlined Mixed-Integer Linear Programming Model for Finding Genetic Patterns Optimally Associated with Complex Diseases Diseases”. In: (2018), p. 56. URL: https://irl.umsl.edu/thesis/343.

[14] Sharlee Climer and Weixiong Zhang. “Cut-and-solve: An iterative search strategy for combinatorial optimization problems”. In: *Artificial Intelligence* 170 (8-9 2006), pp. 714–738. ISSN: 00043702. DOI: 10.1016/j.artint.2006.02.005.

[15] Sharlee Climer, Alan R. Templeton, and Weixiong Zhang. “Allele-Specific Network Reveals Combinatorial Interaction That Transcends Small Effects in Psoriasis GWAS”. In: *PLoS Computational Biology* 10 (9 2014). ISSN: 15537358. DOI: 10.1371/journal.pcbi.1003766.

[16] Sharlee Climer. “Connecting the dots: The boons and banes of network modeling”. In: *Patterns* 2 (12 Dec. 2021). ISSN: 26663899. DOI: 10.1016/j.patter.2021.100374.

[17] Pavan K. Bhatraju et al. “Angiopoietin-Like4 Is a Novel Marker of COVID-19 Severity”. In: *Critical Care Explorations* 5 (1 Dec. 2022), E0827. ISSN: 26398028. DOI: 10.1097/CCE.0000000000000827.

[18] Karsten Suhre et al. “Identification of Robust Protein Associations With COVID-19 Disease Based on Five Clinical Studies”. In: *Frontiers in Immunology* 12 (Jan. 2022). ISSN: 16643224. DOI: 10.3389/fimmu.2021.781100.

[19] Abdulrahman Mujalli et al. “Bioinformatics insights into the genes and pathways on severe COVID-19 pathology in patients with comorbidities”. In: *Frontiers in Physiology* 13 (Dec. 2022). ISSN: 1664-042X. DOI: 10.3389/fphys.2022.1045469.

[20] Yijia Li et al. “SARS-CoV-2 viremia is associated with distinct proteomic pathways and predicts COVID-19 outcomes”. In: *Journal of Clinical Investigation* 131 (13 July 2021). ISSN: 15588238. DOI: 10.1172/JCI148635.

[21] Yayoi Kimura et al. “Identification of serum prognostic biomarkers of severe COVID-19 using a quantitative proteomic approach”. In: *Scientific Reports* 11 (1 Dec. 2021). ISSN: 20452322. DOI: 10.1038/s41598-021-98253-9.

[22] Hui Ma, Arabelle Cassedy, and Richard O’Kennedy. “The role of antibody-based troponin detection in cardiovascular disease: A critical assessment”. In: *Journal of Immunological Methods* 497 (Oct. 2021). ISSN: 18727905. DOI: 10.1016/j.jim.2021.113108.

[23] Peter Jirak et al. “Dynamic changes of heart failure biomarkers in response to parabolic flight”. In: *International Journal of Molecular Sciences* 21 (10 May 2020). ISSN: 14220067. DOI: 10.3390/ijms21103467.

[24] Gulseren Sagcan et al. “Impact of Promising Biomarkers on Severity and Outcome of Acute Pulmonary Embolism”. In: *International Journal of General Medicine* Volume 16 (Aug. 2023), pp. 3301–3309. DOI: 10.2147/ijgm.s416541.