Long COVID Challenge: Predictive Modeling of Noisy Clinical Tabular Data

Mirna Elizondo*, Rasim Musal† June Yu‡ Jelena Tešić§
* ‡ § Department of Computer Science and Engineering
† Department of Information Systems & Analytics
Texas State University, San Marcos TX 78666

***Abstract*—We present an end-to-end machine learning pipeline for aggregating, analyzing, and modeling National COVID Cohort Collaborative (N3C) data on the Enclave system as part of the NIH Long COVID Computational Challenge (L3C). The challenge's goal is to determine the probability of patients who have tested positive for SARS-CoV-2 in an outpatient or hospital setting (ICU or non-ICU) developing PASC/Long COVID. To achieve this, we have utilized state-of-the-art machine learning algorithms to process millions of clinical observations and identify the most impactful attributes that support accurate prediction modeling. The pipeline is optimized for deployment on N3C Enclave and aims to inform clinical decisions for managing and preventing PASC/Long COVID by identifying the most relevant factors. The study implements four state-of-the-art machine learning methods in PySpark on the Enclave for processing noisy tabular data and a novel robust cascaded fusion model. Results show improved modeling performance for high noise levels in clinical data sources and the highest number of true positives and the lowest number of true negatives for the cascaded model. Multiple conditions, observations, and drugs relevant to Long COVID diagnoses and treatment were also identified.**

***Index Terms*—gradient boosting, predictive modeling, noisy data**

I. Introduction

Since the first appearance of acute Coronavirus disease (COVID-19), millions of deaths have occurred. Significant advances have been made in identifying and treating this disease; however, our understanding of the disease has only begun. Scientists have found that in some cases, patients experience post-acute sequelae of SARS-CoV-2 infection (Long COVID, PASC, post-COVID-19 condition), which appears as lasting or new symptoms four weeks before or after being diagnosed with COVID-19. The current definition of Long COVID needs to be clarified due to the need for a greater understanding of the disease. Long COVID is currently reported to have heterogeneous signs and symptoms that can make identifying the most important one difficult; many of these symptoms can appear in other diseases and conditions. Understanding the different conditions or combinations that can lead to Long COVID will allow scientists to develop specific treatments. Patients with autoimmune disorders have shown significant differences in the symptoms they experience; [1] The severity of this disease is drastic for some groups. In this study, we participated in the Long COVID Computational Challenge hosted by the National COVID Cohort Collaborative in the Enclave system (N3C), a National Institute of Health (NIH) National Center for Advancing Translational Sciences (NCATS). In the N3C system, we have access to patient information provided by 75 healthcare centers and 49/50 states in the United States. Data from this challenge represent 15 million patients, including 5.8 million cases of COVID positive and more than 17.5 rows of data. [2].

This study compares and contrasts the multi-step statistical learning pipeline with multiple state-of-the-art decision tree modeling algorithms, including boost and bagging. We used cross-entropy to compare these methods using clinical characteristics selected from demographic, drugs, conditions, and observation information provided by N3C.

Our initial data cleaning, data integration, and data analysis reveal that the nature of the N3C data is typical tabular data from multiple heterogeneous sources. Tabular data in the wild are difficult to model due to the uneven distribution of attributes, missing, overlapping, noisy values, and a mix of categorical and numerical data attributes. We have already shown that the intentional data science pipeline can automatically uncover important attributes, reduce feature space, and model prediction in a robust manner from multi-source tabular data in [3]. We have also shown the effect of socio-demographics on COVID mortality and the importance of their interactions [4].

II. Related Work

Machine Learning The most popular Machine Learning techniques (logistic regression, support vector machines, Bayesian belief network, decision trees, and neural network) for data in the wild generally offer an excellent classification accuracy above 70% for simple classification tasks [5]. From a data science perspective, the modeling approaches evaluated need to be narrower in scope, and feature engineering almost guarantees poor domain/data translation results. Recent findings show that state of the art in machine learning in tabular data outperforms existing approaches and is not as sensitive to input bias and noise as DNN [6]. State-of-the-art *gradient boosted decision trees* (GBDT) models such as XGBoost [7], LightGBM [8], and CatBoost [9] are the most popular models of choice when it comes to tabular data. In recent years, deep-pipeline learning models have

emerged as state-of-the-art techniques on tabular data. TabNet [10], DNF-Net [11], and Neural Obvious Decision Ensembles (NODE) [12]. There is no consensus that deep learning exceeds GBDT in tabular data because standard benchmarks have been absent and open source implementations need to be improved [13], [14]. Recent studies provide competitive examples that compare deep learning models and GBDT in multiple tabular data sets [13], [15], [16]; however, all these benchmarks indicate that there is no dominant winner and GBDT models still outperform deep learning in general. It has been shown that the intentional data science pipeline automatically uncovers important attributes, reduces feature space, and models robustly real tabular data, as demonstrated in [3].

Data Science for Health In a similar study conducted in March 2022 by various institutions, the N3C system contained 2,909,292 patients and 5,645 patients diagnosed with COVID of Long, following the U09.9 code. This provided data set was found and further confirmed that a steeper risk gradient for Long COVID increases depends on the severity of COVID-19 infection [17]. Multi-source clinical tabular data are increasingly challenging to tackle on large scales, and Long COVID data are constantly expanding. This is seen in a study conducted by the N3C Consortium in June 2022. At that time, the data consisted of 1,793,604 patients and 97,995 patients diagnosed with Long COVID, following the U09.9 code. Similarly to the March study, age and sex resulted in high feature importance scores. [18] However, in this study, they developed three XGBoost machine learning models compared to adapting The Phenomizer, which is a web application that generates "a list of clinical characteristics that are most specific for individual diagnoses in a set of selected syndromes and can use this list to guide the further study" [19].

III. N3C Long COVID Computational Challenge (L3C)

Considering the underlying heterogeneity of symptoms in Long COVID and the impact of COVID-19 disease on and predicted by NCATS, NCATS scientists developed the Long COVID Computational Challenge (L3C). The challenge objective was to create "AI/ML models and algorithms that serve as open source tools or use structured medical records to identify which patients infected with SARS-CoV-2 have a high probability of developing PASC / Long COVID" [20]. The challenge started in August and lasted five months. We were assigned to develop, train, and test machine learning algorithms to better understand susceptibility and the probability of developing Long COVID in patients with COVID-19 disease. The *challenge question* we are trying to answer in this work is: "Of patients who have tested positive for SARS-CoV-2 in an outpatient or hospital setting (ICU or non-ICU), what is the probability of developing PASC/Long COVID?" [20].

Some patients in these records have been identified with the U09.9 code of Diagnoses with Long COVID; others could have undiagnosed Long COVID. The N3C Enclave uses the standard data model of the observational medical outcomes partnership (OMOP) (version 5.3) used in various health centers. OMOP models facilitate the understanding of relevant factors and serve as central identity management for all patients in the database [21]. This allows a centralized vocabulary to be used throughout organizations. Observational Health Data Science and Informatics (OHDSI) has created an open source web application, Athena [22], which facilitates the interpretation of the OMOP common model. Currently, health professionals have classified 30 standards *concept_id's* that are relevant to "post-acute COVID-19" [23].

TABLE I
L3C CHALLENGE TRAINING AND TESTING DATA FRAME SOURCES AND PERCENTAGE OF MISSING DATA. WE HAVE USED THE **BOLDED** DATA SOURCES.

Data Set	Rows X Columns (% missing values) Test	Train
care_site	8,367 x 8 (66)	26 x 8 (1)
condition_era	2,484,521 x 8 (0)	13,639 x 8 0
condition_occurrence	**6,316,765 x 21 (37)**	**36,451 x 21 (35)**
condition_to_macro	**1,286,673 x 8 (6)**	**8,388 x 8 (5**
device exposure	422,167 x 19 (44)	2,836 x 19 (45)
drug_era	**2,090,455 x 9 (0)**	**12,698 x 9 (0)**
drug_exposure	**13,611,559 x 28 (42)**	**66,050 x 28 (39)**
location	25,142 x 9 (57)	281 x 9 (60)
long COVID	**57,675 x 2 (13)**	**300 x 2 (0)**
manifest_safe	69 x 6 (23)	300 x 13 (23)
measure	32,569,723 x 29 (33)	198,151 x 30 (33)
measure_to_macro	17,839,906 x 8 (0)	112,243 x 8 (2)
micro_to_macro	3,524,398 x 28 (54)	19,430 x 26 (54)
note	321,151 x 19 (59)	2,710 x 19 (66)
note_nlp	7,580,262 x 21 (38)	60,486 x 21 (45)
observation	**6,869,266 x 25 (49)**	**43,355 x 25 (43)**
observation_period	**45,404 x 7 (0)**	**234 x 7 (0)**
payer_plan_period	1,370,746 x 26 (69)	6,029 x 26 (69)
person	**57,672 x 26 (29)**	**300 x 26 (28)**
procedure_occurrence	**278,981 x 19 (22)**	**14,645 x 19 (23)**
procedures_to_macro	991,579 x 8 (5)	5,247 x 8 (5)
provider	31,664 x 18 (51)	477 x 18 (56)
visit_occurrence	350,934 x 23 (49)	19,411 x 23 (49)

A. N3C requirements

N3C guidelines follow privacy regulations to ensure patient anonymity. Organizations are required to sign a Data Use Agreement (DUA) with N3C, and depending on the level of deidentified clinical data, participants may be required to have Institutional Review Board (IRB) approval. Next, all team participants must register, join N3C and complete the data security training. Finally, to participate in the challenge, each participant in the group must submit a request for access to a project workspace and the challenge. The processing time of the requirements caused delays in development for our team. Challenge data consist of COVID-19 patients' cases and include their demographics, medical conditions, medications prescribed, consultation observations, procedures, laboratory tests, physical measurements, and more. In N3C, there are

three levels of deidentified data; for the challenge, we were granted access to level two data; this means that the dates are algorithmically shifted, patient ZIP codes are limited to three digits or removed if there are fewer than 20,000 individuals, or the location represents tribal land. The challenge has provided a set of *Censored Training* and *Censored Test* sets in which we have 23 data sets in both sets, as described in Table I. In this work, we refer to sets as training and testing sets. The dates of the censored set range from the COVID index date to four weeks after exposure. More details on data analysis and aggregation are provided in Section IV.

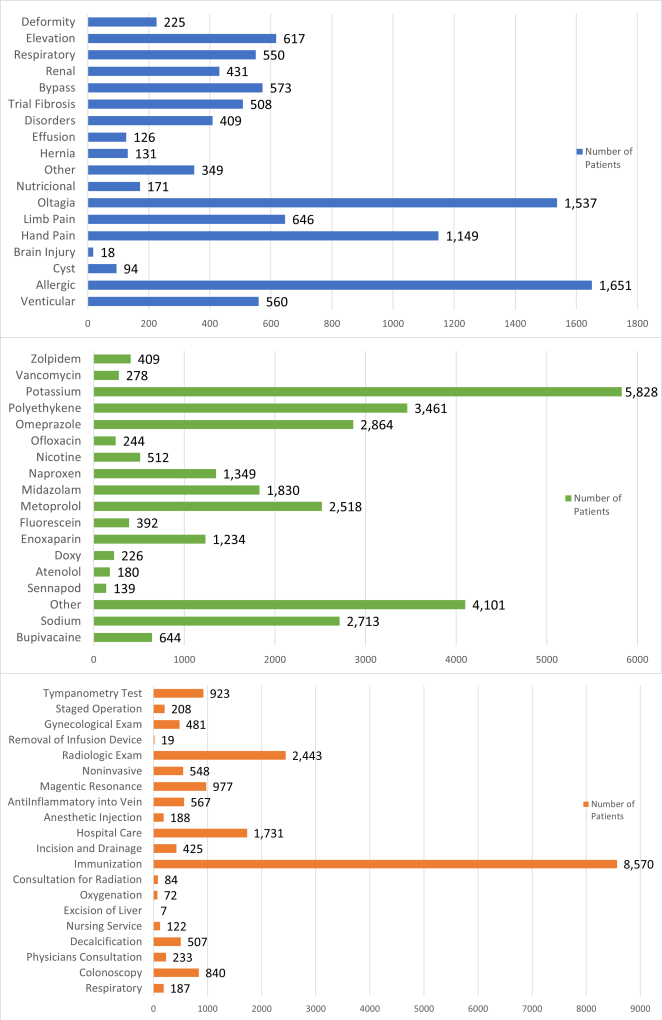

Fig. 1. Top 20 conditions, drugs, and procedures used in the **Early Fusion** frame and the average frequency of the record per patient.

IV. L3C Data Source Aggregation to Data Frames

Our base population is defined as patients with a history of a COVID-19 Diagnoses code or positive post-acute sequelae of SARS-CoV-2 PCR or antigen test. **Long COVID Silver Standard** information provided for the challenge contains the 2 columns in a Long COVID frame described in Table I, *person_id* and *covid_index*. The label provided for the training

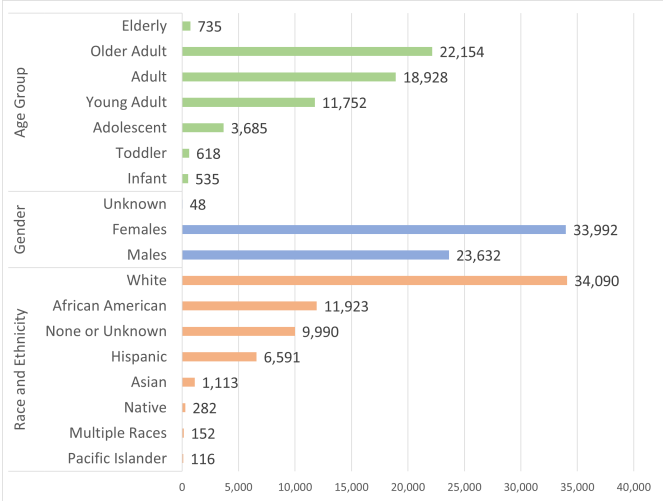

Fig. 2. Distribution over aggregated gender, race, ethnicity, and age group categories in the Demographics data frame

data is the *pasc_after_four_weeks* column is a binary label on who has tested positive again with COVID-19, and of 57,672 patients, of which 9,031 patients were recorded as testing positive for Long COVID after four weeks after infection. We use this column as the prediction label in Section VII to train our models. Next, we describe in detail how we have used and aggregated the information available in the bolded data frames in Table I to produce Table IV data frames. The training set includes 57,672 patient records and 11,446 patients diagnosed with Long COVID before or after four weeks of the COVID index. Since 2,415 records were taken four weeks before the COVID index, we excluded these and used 9,031 to determine prognostic factors leading to diagnoses. All final data frames used in the analysis are listed in IV.

The Early Fusion data frame is our baseline data frame. It was created from the data sources *Long COVID Silver Standard*, *condition_occurrence*, *drug_era*, *procedure_occurrence*, described in Table I. The **Early Fusion** data set uses the original *person_id* as an index and *covid_index* is used to derive the numerical age column as the difference between the record date (computed as *covid_index* OR *year_of_death* OR 2021) and *year_of_birth*. The column of the *year_of_birth* data set has 2,065 missing values and *is_age_90_or_older* has 1,330 missing values. To fill the missing *year_of_birth* values we decided if a patient was recorded as older than 90 years, they were assigned as 1932 since it would be the minimum value possible (2021-89) as their *year_of_birth*. We then predict the following missing *year_of_birth* values using Ordinary Least Squares regression on *gender_concept_name*, *race_concept_name* from person data sets and the Long COVID Diagnoses binary training label. All *gender_source_value* entries are aggregated into 3 binary categories, and all *race_concept_name* into 10 different categories. In total, three attributes were integrated *year_of_birth*, *gender_source_value*, and *race_concept_name*

related to the demographics of the patient. Twenty attributes were integrated for each of the conditions, drugs and procedures selected based on the evaluation of the amount of information each attribute has on the uncertainty of the Long COVID label through Lindley entropy [24]. The distribution of the 60 conditions, procedures, and drugs for the Early Fusion data frame is illustrated in Figure 1.

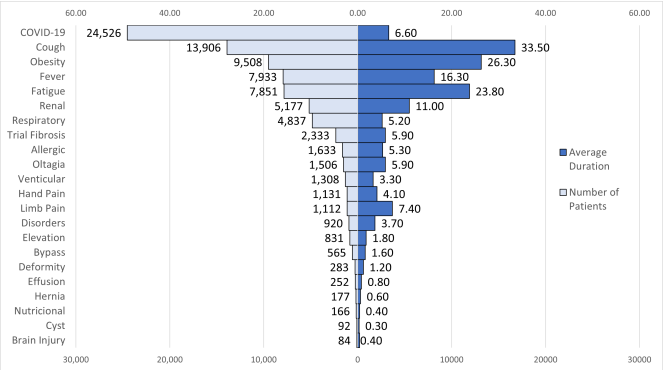

Fig. 3. The most frequent conditions per patient in the training data set (left) and their average duration (right).

The data frame **Demographics** aggregates the information provided in the **person** data set, described in Table I. The aggregated three columns of *gender_concept_name* and one column for age are the same as in the Early Fusion data frame. Next, we aggregated 22 different *race_concept_name* and 7 different *ethnicity_concept_name* into eight different racial binary columns, as illustrated in Figure 2. We also use the age column to create seven additional binary columns for the classification of age groups: infant (less than 2), toddler (more than 2 and less than 4), adolescent (more than 4 and less than 14), young adult (more than 14 and less than 30), adult (more than 30 and less than 50), older adult (greater than 50 and less than 90) and elderly when *is_age_90_or_older* is true. The **Demographics** data frame has 22 attributes: *person_id*, age, seven binary age groups, three binary gender, and 11 binary race and ethnicity attributes, and Figure 2 illustrates the distribution of the attributes in the training set.

Conditions per patient and their occurrence and duration records were obtained from *condition_occurrence*, *condition_era*, and *condition_to_macrovisits* (see Table I) resulting in aggregated 38,044 patient records with at least one condition in the training set, and 200 of 300 patients in the test set. Each patient had at least one condition out of 14,764 unique conditions, lasting from 1 to 409 days. We selected 96 unique conditions from the set of 14,764 as the intersection of unique conditions in 3 sets: (1) 30 conditions previously found to be related to secondary symptoms of COVID-19 [23]; (2) 20 conditions from the Early Fusion set; and (3) 66 most distinct (not trivial) frequent conditions in the training set associated with the second COVID label of the patient. These 96 unique conditions were aggregated into 24 total attributes, for example, all attributes containing the phrase 'brain injury'

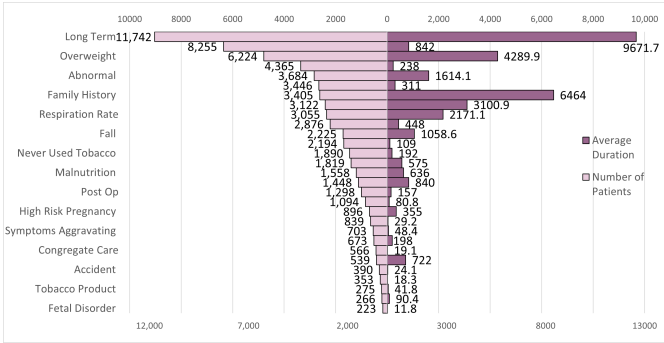

Fig. 4. The most frequent observations per patient in the training data set (left) and their average duration (right).

are integrated into one column, and the distribution of the conditions and their cumulative time per patient is illustrated in Figure 3. Two data frames were created: the **Condition** data frame, whose entries capture the cumulative duration (days) for the patient (row) who experienced the condition (column), and the **ConditionsB** data frame captures the binary relationship between the patient and the condition: 1 if the patient experienced the condition, 0 if they did not).

Observation records are aggregated from two data sources: *observation_period* and *observation* (Table I), and the records were recovered for 38,340 patients in the training set and 208 in the test set (Table IV) for 2,744 unique observations. Each observation can last from 1 to a 'long-term stay' in the hospital, and multiple observations can be observed in a single patient as illustrated in Figure 4. For this feature selection, we consider the 34 observations we found to be most frequent in the patients in training set diagnosed with second COVID. Then, we aggregate the observations by common keyword into 34 groups, as illustrated in Figure 4. Finally, two numerical data frames were created: **Observation** captures the cumulative duration (days) the patient (row) has the observation (column) in their record, and **ObservationB** captures the binary relation between patient and the observation (1 if the patient has the recorded observation, 0 if they did not).

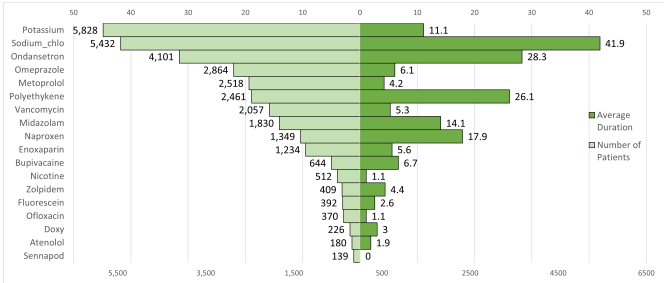

Fig. 5. Most frequent drugs per unique patient in the training data set (left) and their average duration (right).

Drugs *drug_era* and *drug_exposure* data sources, described in Table I. We integrated records for 35,872 patients and

14,159 unique drug values and aggregated the drug values along 23 drug indices about those most frequently prescribed for patients. We integrated all those records that can be observed in a single patient as illustrated in Figure 5. Note two data frames were created: Drugs-prepare captures the binary relation between the patient and the drug (1 if the patient was taking the drug, 0 if they were not) and Drugs-prepare1 captures the cumulative duration (days) the patient (row) was taking the drug (column). The drugs data frame was integrated into the diagnoses data frame and the models were not run on drugs. The **Diagnoses** integrates the 22 Demographics, 24 Conditions, and 18 Drug attributes per 38,340 patients in the training set and 200 in the test set (Table IV). Note two data frames were created: **DiagnosesB** cell value is 1 if the patient (row) ever had the Diagnoses (column), 0 if they never not, and **Diagnoses** frame numerical cell captures the cumulative duration in days that the patient (row) had the Diagnoses (column).

TABLE II
AGGREGATED DATA FRAMES PER PATIENT (57,672 PATIENTS IN TEST AND 300 PATIENTS IN TRAINING), AND THE FINAL COLUMN DIMENSION.

Data Frame	Unique patient record		Attribute
	Train	Test	Count
Early Fusion	57,562	300	64
Demographics	57,562	300	22
Conditions	38,044	200	24
ConditionsB	38,044	200	24
Observations	38,340	208	32
ObservationsB	38,340	208	32
Diagnoses	33,899	200	64
DiagnosesB	33,899	200	64

V. DATA MODELING

In this section we introduce four modeling strategies for the challenge: statistical multi-step logistical regression, random forests, decision trees, and gradient boosting.

A. Multi-Step Logistical Regression

We used three additional data frames for logistic regression modeling: conditions, drugs, and procedures. These involve the patient's history of diagnosed conditions, prescribed drugs, and procedures performed. In the first stage of logistic regression, we fit all the characteristics of the model, *gender_source_value*, *is_age_90_or_older*, *race_concept_name*, age and the top 20 characteristics selected from conditions, drugs, and procedures. The second step uses the first logistic regression model as a starting point. It performs a step-wise search method to eliminate variables that do not improve the Bayesian Information Criterion (BIC) [25]. We used the remainder of the identified features in a second-stage logistic regression model search that involves interactions between these features. The threshold probability for the binary outcomes was simulated for various values and 0.3 was chosen because it subjectively provided the best trade-off between accuracy, sensitivity, and precision. Next, the selected model is used as a base, and we test possible interactions between these features through a step-wise regression using Bayesian information criterion (BIC) [25]. The final model used 64 features with 35 identified interactions. The Akaike information criterion (AIC) decreases from 44737 to 43640 from the second to third logistic regression, which makes up for the loss in degrees of freedom. Logistic regression with interactions takes approximately 17 hours, and Table III has information on the evaluation of Model 3 on the training dataset, a third stage of the pipeline with a cutoff point of the probability of 0.3.

B. Decision Tree Models

Decision Tree Modeling of the L3C data Decision Trees are an easily understood classification approach that closely mirrors human decision-making, they can represent data graphically and are easily interpreted even by a non-expert. Decision trees also handle categorical variables well, and their performance can be improved using ensemble modeling. Taking into considering that a single decision tree improves predictions for a particular data set but poorly for others we implement. They are slower to build and are difficult to interpret at this. Four parameters can be tuned in this model: the depth of the trees, the minimum number of instances required per node, the minimum information gain, and the impurity.

C. Random Forests Models

A Random Forests model is made up of multiple decision trees and is an extension of the bagging method. Using a random selection of decision trees ensures low correlation since the models perform better as a group rather than alone. A single decision tree is likely to produce errors but by selecting the most common predictions we are able to reduce the risk of over-fitting and determine feature importance. Random Forest can also handle estimating missing data. The algorithm is made up of a set of trees, which each ensemble has a data sample from the training set called the bootstrap sample. This is trained using bootstrap aggregation (bagging), which conducts row and feature sampling from the data for the data frames for the model. Random Forest models show the different tree combinations that are seen with only small changes in the data and features. Considering the Enclave system limitations in scaling by using a random forest model we can benefit from their ability to handle large data bases without variable deletion. Two parameters can be tuned in this model: the number of trees and the number of features per node.

D. Gradient Boosting Models

In comparison, Gradient Boosting provides an explainable ensemble of relatively small trees that sequentially model the error, and offer an easy way to retrieve importance scores for each attribute. The decision trees are build one after another, Gradient boosting approaches handle tricky observations well and are optimized for faster and more efficient fitting using a data sparsity-aware histogram-based algorithm. Next, we

constrain the tree structures to reduce the growth of complex and longer trees by optimizing parameters such as the number of trees, the depth of trees, and the number of leaves per tree. The more an attribute is used to make key decisions with decision trees, the higher its relative importance.Three parameters can be tuned in this model: the number of trees, the depth of trees and learning rate. The difference between a Random Forest and Gradient Boosting model lies in how the ensemble of trees are trained, RF trains each tree individually while in GB each tree helps the previously trained trees errors. These models were tested with different parameters and the best performing hyper-parameters can be seen in Section VII-C.

E. Late Fusion Cascade Modeling

Algorithm 1: Cascade Model Fusion

Data: Set of models Mk, $k \in 0, 2, ..K-1$(Tab. VII)
Data: Set of patients P, $p_i \in i \in 1, ...N$
Result: Tuple (L, P) where L is a label set for P

```
k=0; P_0 = P, L = ∅ ;
while k < N OR all |P_k| = 0 do
    Apply model M_k to all N patients ;
    L_k is set of patient labels that have coverage in
      the model M_k ;
    P_{k+1} = P \ P_k ;
    L = L ∪ L_k ;
    k++;
end
```

In this section we propose a cascade fusion of different data models based on their effectiveness and on the coverage. N3C data is clinical data, and we do not have the data on every patient in every frame. We only have the information for every patient in the training and test set in the Demographics data frame, see Table IV. To mitigate for the varying degree of data coverage we propose the late fusion cascade model. First, we rank all the available models in terms of their effectiveness on the training dataset, and assign ranking in order as M1 the most effective, M2 right after, and so on. Next, we evaluate the model in the ranking order on the target dataset. For each patient int he dataset we assign the label of the model with the highest ranking that covers that patient data in one of the data frames. Thus, we guarantee that the best model gets the largest coverage, but also that every patient in the dataset has a prediction label. The algorithm is detailed in Alg. 1. The approach is evaluated in Section VII-C after we rank the models on the training dataset in Table VII.

VI. Attribute Importance Analysis

The L3C data set is sparse and noisy, and over 10,000 unique identifier per frame makes it hard to evaluate the feature selection and aggregations approaches on Enclave. Thus, we seek to validate the data aggregation approaches presented in Section IV by applying eight different feature ranking methods to Demographics (22 binary and one numerical attribute), Conditions (24 binary attributes), Observations (32 numerical attributes) data frames, and Diagnoses (64 binary). and designed in Section IV. Each proposed feature importance method selects a subset of features based on minimum redundancy and maximum relevancy: (1) Variance threshold filtering removes attributes by eliminating all low-variance attributes in the training set. (2) Lasso regularization of logistic regression (penalty L1 term) shrinks the coefficients by minimizing the loss function during training. (3) Random Forest embedding has a built-in feature importance measured by the Gini importance or mean decrease impurity. We propose to set the 50th percentile threshold for the importance of the attribute to include a relevant attribute in the final set. (4) Random forest Recursive Feature Elimination (RFE) with Random forest first fits the full set of attributes in our data set, and we eliminate features with the smallest coefficients if they deteriorate the 10-fold cross-validation score of the models in the training data. (5) Recursive Feature Elimination (RFE) with Ridge Regression eliminates features with the smallest coefficients if they deteriorate the 10-fold cross-validation score of the models in the training data. Permutation Feature Importance(PFI) with (6) Random Forest method and with (7) Ridge Regression methods replace the values of a feature with redundant noise and measures the difference in the accuracy score or other performance metrics between the baseline and the permuted data set. The sequential Feature Selection (SFS) model selects an optimal set of features by searching the feature space of all combinations in a greedy manner. We evaluated each subset of features that add one predictor at a time forward based on the 5-fold cross-validation score of the (8) Ridge regression regularization. We apply the importance attribute rankings to separate frames captioning demographics, conditions, observations and drugs, and to the fused frame of demographics, conditions and drugs attributes, the Diagnoses frame, and analyze the results in Section VII-D.

VII. Experiments

A. Setup

In this section we report on the attribute importance and modeling experiments using data frames described in Section IV and outlined in Table IV. The training set includes 57,672 patient records where 9,031 patients have been diagnosed with Long COVID and 48,531 were not. We report the effectiveness of the modeling using a confusion matrix and derived information retrieval measures such as precision, recall, F1, and accuracy measures [26]. We use the following notations: **TP**-the number of patients who had Long COVID in the set with the ground truth label that the model predicted as positive; **FP**-the number of patients who did not have Long COVID in the set with ground truth label that model predicted as positive; **FN**-the number of patients who had Long COVID in the set with ground truth label that the model predicted as negative; and **TN**-the number of patients that did not have the Long COVID in the set with ground truth label that model

predicted as negative, and precision P, recall R, F1 measure and accuracy Acc, defined as:

$$P = \frac{TP}{TP+FP}, R = \frac{TP}{TP+FN}, F1 = \frac{2*P*R}{P+R} \quad (1)$$

$$Acc = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

If the model is evaluated on the entire training data set, the following equations hold for the model evaluations TP + FN = 9,032 and TN + FP = 48,531 in Table III and for the Demographics and Early Fusion data frame in Table VII. Precision (P) is defined as 'the probability that an object is relevant given that it is returned by the system' [27] and recall (R) is defined as 'the probability that a relevant object is returned' [27]. We use F1-measure to combine our results for precision and recall by taking the (weighted) average of precision and recall. Acc is short for accuracy and is the fraction of predictions our model correctly evaluated. Note that training data is somewhat imbalanced with 9,032 patients with the Long COVID Diagnoses and 48,351 patients with no Long COVID Diagnoses. Thus, we do not use accuracy as a measure to draw conclusions in the paper, as accuracy emphasizes the trivial negative class.

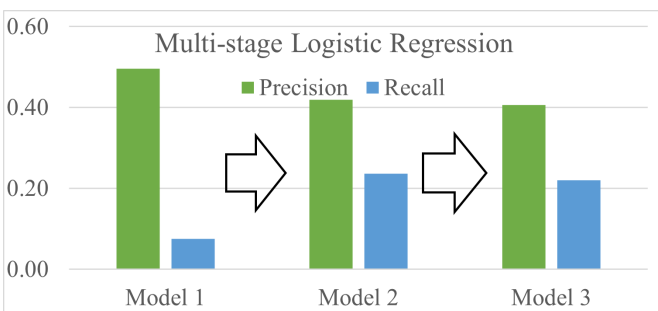

Fig. 6. Precision (green) and Recall (blue) Scores for each of the three stages of the logistic regression modeling on the entire data set.

B. Experiment: Statistical Modeling Baseline

TABLE III
STATISTICAL MODELING SCORES.

	Model 3	**Model 2** threshold selection)				**Model 1**
	0.30	**0.10**	**0.20**	**0.30**	**0.40**	**0.50**
TP	2031	8128	4516	2167	1174	722
FP	2967	28817	8765	2994	1378	723
FN	7000	903	4515	6864	7857	8009
TN	45564	20319	39776	45437	471533	47808
P	0.41	0.22	0.34	0.42	0.46	0.50
R	0.22	0.90	0.50	0.24	0.13	0.08
F1	0.29	0.36	0.41	0.31	0.20	0.14
Acc	0.25	0.50	0.77	0.83	0.84	0.84

The statistical model is a three-stage model trained and tested using the attributes of the Early Fusion data frame. In the first stage, Model 1 uses demographic information from the Early fusion data frame to train the logistic regression model and to identify 33 attribute interactions. The results of the trained model on the entire training data set are outlined in the Model 1 column in Table III. The next step uses the Model 1 scores on the same attribute set and evaluates the effectiveness of the performance with respect to cutoff threshold at lower levels in Table III. We select **0.3** as logistic regression classification threshold for assigning 1 and 0 labels to the outcome. This results in the slight drop in precision with the significant increase in the recall R , as illustrated in Figure 6. The modeling result on Model 3 applies the findings to the full set of 64 features. The L3C benchmark submission is Model 3 coefficients applied to the testing data with 0.3 binary threshold. The model pipeline performance (from right to left) is evaluated on entire data set and described in Table III, and modeling comparison is illustrated in Figure 6.

Statistical modeling uncovers several interesting gender relations in the training data set: (1) Even though the race category is not statistically significant, it is chosen by the Bayesian Information Criterion at the Long logistic regression stage in Model 1; (2) Model 1 coefficient analysis shows that females have a higher risk of Long COVID than males; and (3) being of unknown gender has a rather different coefficient estimate (-1.8) relative to being classified as male (-0.21). The final model, Model 3, identifies 35 interactions on the entire training data set, and the Akaike's Information Criterion decreases from 44,737 to 43,640 from Model 2 to Model 3 which makes up for the loss in degrees of freedom. An African American man who is 40 years old has a probability of 29.5% to get Long COVID whereas for a female this increases to 31.32%. On the other hand, if gender is unknown this probability for the same individual decreases to 7%. This indicates an unobserved effect regarding this value and a deeper investigation is necessary. The largest coefficient that is effecting the log probability is the type of life support system utilized: Extracorporeal Membrane Oxygenation (ECMO) and Extracorporeal Carbon Dioxide Removal (ECLS) which increases the above mentioned female acquiring Long COVID to 86%. If instead, this individual utilizes omeprazole drug it amounts to a probability of 76%. The ECMO is linked to cardiac failure, cardio-respiratory failure, and respiratory failure conditions; the ECLS is linked to CO_2 retention conditions. [28]

Figure 6 and Table III compare all 3 models on the training data set in terms of precision and recall. Model 3 includes the full 64 attributes from Early Fusion data frame, and the precision and recall stay comparable to Model 2 that includes demographics data only in terms of precision, recall, and F1 measure, while the accuracy significantly drops on the account of the increased true *and* false positives in the model.

C. Experiment: Model Tuning, Selection and Fusion

In this experiment, we tune and train three families of models, random forests, decision tree, and gradient boosting on 80% held-out training data set for 7 frames. Decision tree and gradient boosting hyper parameters tuned in the data

TABLE IV
DECISION TREE AND GRADIENT BOOSTING HYPERPARAMETERS PLUS ALL MODELS' IMPURITY IS 'ENTROPY' AND MININFOGAIN IS 0.0. * - MISSING VALUES FROM ENCLAVE.

	Decision Tree		Grad Boost	
data frame	**max Depth**	**min Instance**	**min Weight**	**sub sampling**
Demographics	10	10	0	0.3
Conditions	10	10	0	0.5
ConditionsB	10	5	0	0.5
ObservationsB	10	5	0	*
Observations	10	10	0	*
Diagnoses	5	1	0.1	*
DiagnosesB	5	1	0	*

TABLE V
RANDOM FOREST PARAMETER TUNING OUTCOMES FOR DATA FRAMES. ALL DATA FRAMES SHARE THE FOLLOWING: MAXDEPTH IS 10, MININSTANCE IS 10, MININFOGAIN IS 0.0, MINWEIGHT PER NODE IS 0.0 AND MAXITERATIONS IS 20.

	Random Forest Hyper Parameters			
data frame	**max Depth**	**Impurity**	**min Instance**	**sub sampling**
Demographics	20	gini	1	0
Conditions	10	entropy	5	0
ConditionsB	10	entropy	5	0.3
ObservationsB	20	entropy	10	0
Observations	30	gini	10	0.3
Diagnoses	30	gini	1	0
DiagnosesB	10	gini	1	0

frames are listed in Table IV. For both models and all data frames, the best impurity parameter is 'entropy' and the best minInfoGain is 0.0. The Random Forest parameters are in the Table VII-C, and all data frames share the following tuning parameters maxDepth is 10, minInstance is 10, minInfoGain is 0.0, minWeight Per Node is 0.0 and maxIterations is 20.

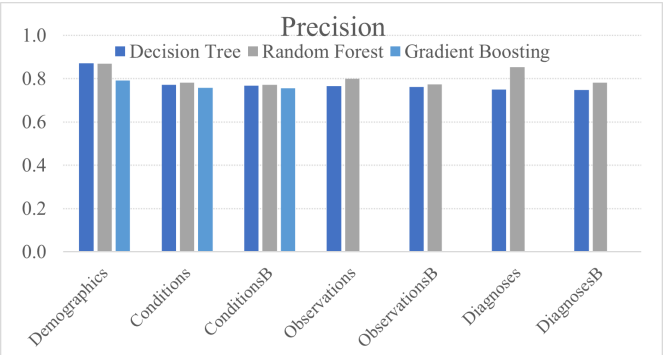

Fig. 7. Precision, Recall and F1 comparisons for the Decision Tree Model, Random Forest and Gradient Boosting Models

Next, we compare the performance of the model on the 20% of the training data set, and Table VI summarizes the effectiveness of the three methods on seven data frames in terms of precision P, recall R, and F1-measure. First, we compare the Gradient Boosting of the Decision Tree with the Decision Tree and Random Forest Modeling due to the time

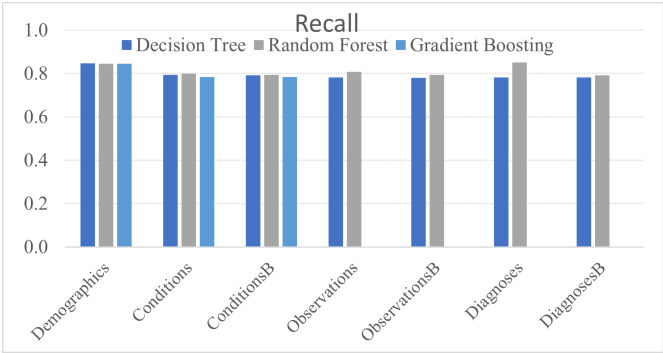

Fig. 8. Recall for Random Forest Model, Random Forest and Gradient Boosting Models

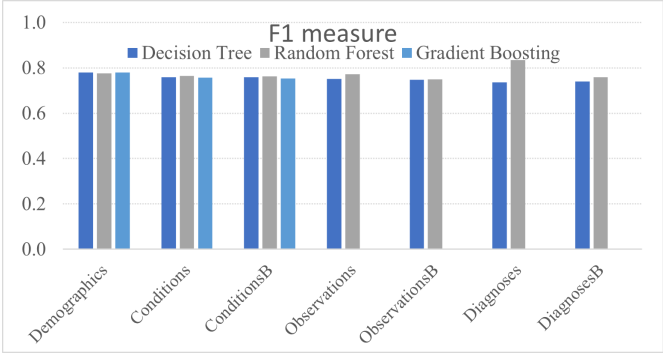

Fig. 9. F1 for Decision Tree Model, Random Forest and Gradient Boosting Models

TABLE VI
EFFECTIVENESS OF THE THREE MODELS WITH TUNED HYPER-PARAMETERS EVALUATED ON A HOLDOUT TRAINING SET FOR SEVEN DATA FRAMES FOR DECISION TREE AND RANDOM FOREST AND ON THREE DATA FRAMES FOR GRADIENT BOOSTING.

method	Gradient Boosting			Random Forest		
data frame	**P**	**R**	**F1**	**P**	**R**	**F1**
Demographics	**0.792**	0.845	**0.78**	**0.869**	0.845	0.78
Conditions	0.75	0.785	0.76	0.78	0.799	0.76
ConditionsB	0.75	0.785	0.75	0.77	0.793	0.76
method	**Decision Tree**			**Random Forest**		
data frame	**P**	**R**	**F1**	**P**	**R**	**F1**
Demographics	**0.871**	**0.848**	**0.780**	**0.87**	0.845	0.78
Conditions	0.77	0.794	0.76	0.78	0.799	0.76
ConditionsB	0.76	0.792	0.76	0.77	0.793	0.76
Observations	0.76	0.782	0.75	0.79	0.808	0.77
ObservationsB	0.76	0.781	0.75	0.77	0.793	0.75
Diagnoses	0.75	0.783	0.74	0.85	**0.850**	**0.84**
DiagnosesB	0.75	0.780	0.74	0.78	0.792	0.76

complexity of running the method on Enclave: the processes never completed for the Gradient Boosting models on our Observations and diagnoses data frames. Random Forest in Table VI demonstrates the same or superior comparison over Gradient Boosting for demographics and conditions data frames. When we expand our analysis for Decision Trees and seven data frames, Random Forest modeling shows a consistently robust superior approach over Decision Trees. Thus, we select Random Forests as our final modeling ap-

proach as they consistently demonstrated the same or superior performance when compared to Decision Tree and Gradient Boosting Decision Tree in terms precision P, recall R, and F1-measure, as illustrated in Figure 7, Figure 8, and Figure 9, respectively.

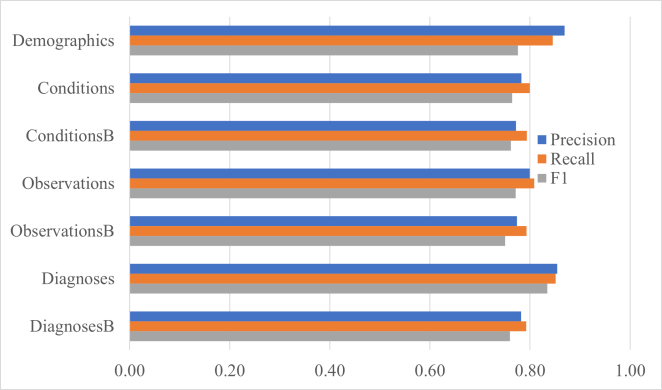

Fig. 10. Selecting the most informative data frames for the Random Forest modeling based on precision P, recall R, and F1-measure on the training holdout set.

Random Forest was selected as the final model across all data frames based on the speed of performance and comparable or superior measures across the board in Table VI. Our next step is to select the most informative data frames for the final submission from all due to the Enclave processing limitations. To this end, we compare the Random Forest performance of all data frames in Table VI per data frame in Figure 10. Condition dataframe is part of the Diagnoses dataframe so we focus on the Demographics, Observations, and Diagnoses data frames going forward.

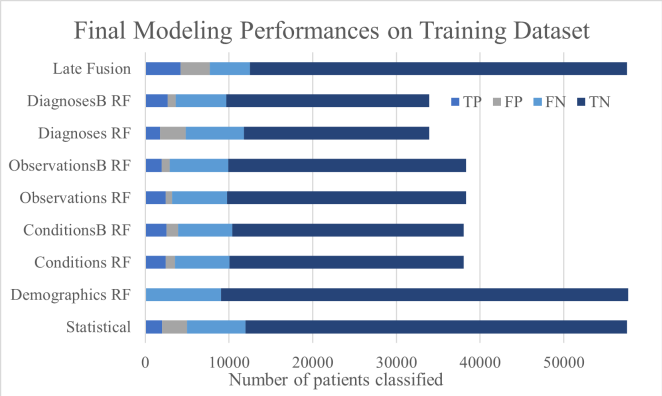

Fig. 11. Final model performances from Table VII on entire training dataset. Late fusion model only returns prediction for 26,923 patients out of 57,562 patients total.

The random forest model was then re-trained on the entire training data set using tuned hyper-parameters from Table VII-C and evaluated on the entire training dataset, as illustrated in Table VII and illustrated in Figure 11. Late Fusion modeling Note that Early Fusion and Demographics datasets are the only datasets in Table IV to have the full train and test coverage. We narrow down the final decision based on Table VII measures on the entire dataset, and the Late Fusion challenge submission combines in cascade all Random Forest model outputs as follows. DiagnosesB Random Forest model output had the highest number of detected true positives (2,659) and the lowest number of detected false negatives (996). The Diagnoses data frame combines the Demographics, Conditions, and Drugs data frames. It provides the most robust precision score of all modeling outcomes in Table VII and in Figure 11, and it has coverage to provide prediction scores for the 33,899 patients in the training set and 125 patients in the test set, as outlined in Table IV. Next, we apply the cascaded model fusion algorithm explained in Section V-E based on the model ranking in Table VII: M1-DiagnosesB -¿ M2-ObservationsB -¿ M3-ConditionsB -¿ M4-Demographics. The ordering of the models was determined based on their specificity, coverage, discrimination, and effectiveness on the training dataset, as analyzed in Table VII. For the training dataset, the algorithm Alg. 1 assigns DiagnosesB modeling output values to 33899 patients in the training set. For the remaining 23,663 patients we use M2 then M3 then M4 prediction label, and final results is the Late Fusion model evaluated in the last row of Table VII with robust 4,216 true positives, 448 false negatives, and 3482 true positives in the training dataset, and highest recall and accuracy of all models.

D. Experiment: Attribute Importance

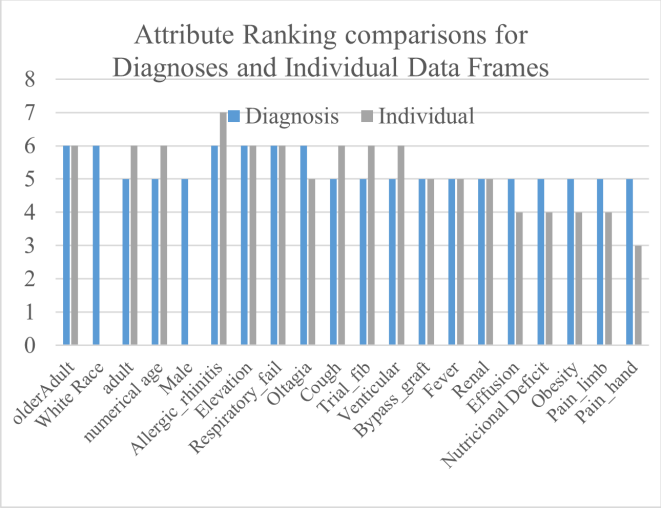

Fig. 12. Feature Importance of Top Demographics and Conditions Attributes selected by y=axis number of methods from Diagnoses dataframe (blue) and Individual dataframes (grey).

In this experiments, we compare the attribute importance ranking the Diagnoses frame for eight selection methods described in section VI. Diagnoses data frame aggregates Demographics, Conditions, and Drugs data frames. Figure 12 illustrates the difference in attribute selection based on importance from eight methods from individual data frames (Demographics and Conditions) (gray) versus when the same

TABLE VII
FINAL MODEL PERFORMANCES FOR THE ENTIRE TRAINING DATA SET. THE ENCODING OF THE TOTAL DAYS FOR CONDITIONS OBSERVATIONS AND DIAGNOSES MADE NO STATISTICAL DIFFERENCE FOR THE MODELING.

Model	**data set**	**TP**	**FP**	**FN**	**TN**	**P**	**R**	**F1**	**Acc**	Model Rank for Alg. 1
Statistical	**Early Fusion**	2031	2967	7000	45564	0.406	0.225	0.290	0.827	
Random Forest	**Demographics**	98	0	8933	48641	**1.000**	0.011	0.021	0.845	M4
Random Forest	**Conditions**	2449	1085	6544	27966	0.693	0.272	0.391	0.799	
Random Forest	**ConditionsB**	2546	1402	6447	27649	0.645	0.283	0.393	0.794	M3
Random Forest	**Observations**	2432	799	6555	28554	0.753	0.271	0.398	0.808	
Random Forest	**ObservationsB**	1980	922	7007	28431	0.682	0.220	0.333	0.793	M2
Random Forest	**Diagnoses**	1764	3070	6935	22130	0.365	0.203	0.261	0.705	
Random Forest	**DiagnosesB**	2659	**991**	6040	24209	0.728	0.306	0.431	0.793	M1
Late Fusion	Alg. 1	**4216**	3482	**4816**	45048	0.548	**0.467**	**0.504**	**0.856**	

approach is applied to the fused Diagnoses frame (blue bar). What is interesting is that white race and male gender were not selected as important from the individual demographics data frame, and six methods agreed that four attributes related to age are important and nothing else is from the entire demographics frame. Allergic Rhinitis, Elevation and Respiratory Failure are ranked as the most impactful condition attributes in the Diagnoses frame where the individual ranking includes Oltagia, Cough, Trial Fibrosis, Ventricular and Cough, also. All eight conditions are the top-ranked conditions by algorithms both from Diagnoses and from Conditions data frames. The interesting finding is that the Loss of Taste conditions were never found as most impactful; see Figure 12 for more detailed comparison of attribute selection. The following **Drugs** in Diagnoses data frame were selected by six methods: Enoxaparin, Ondansetron, and Sennapod as most impactful, and we do not have individual scores to evaluate the robustness of this finding.

Six methods feature importance out of eight introduced in Section VI marked twelve observations documented by provider as most relevant for Long COVID patients: Alcohol, Congregate Care, Drug Indicated, Family History, Health Status, History of Observation, Malignant Disease, Never Smoked, Observation period duration, Severely Obese, Aggravating Symptoms, and Tobacco Product.

E. Experiment: Binary vs. Numerical Attribute Encoding Impact

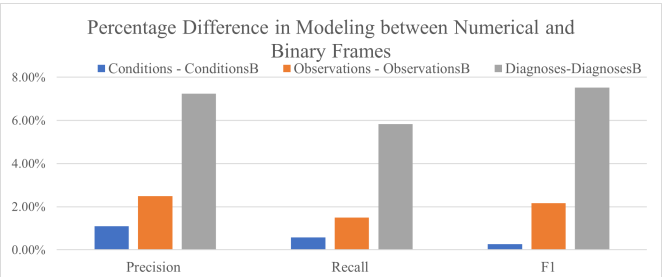

Fig. 13. Numerical vs. Binary feature encoding shows the significant difference in performance on the 20% holdout testset for the diagnoses data frame that integrates condition and drug information.

In this experiment, we also analyzed the modeling performance of numerical versus binary data frames for conditions, observations, and diagnoses. Table VI shows that the performance of the models when the feature value is the cumulative number of days a patient was assigned the condition / diagnoses / observation / drug attribute performs consistently better than simply modeling the presence of the attribute for the patient in the data frame. Performance improvements for conditions, observations, and diagnoses using rich data frame encoding are illustrated in Figure 13 by the difference in percentage points. The diagnoses data frame combines demographic, conditions, and drug data frame, and has been selected as the most informative one in Figure 10, so it is not surprising that the greatest information gain is obtained by encoding the number of days the patient is associated with the attribute. However, the final model evaluation on the entire training dataset favors binary representation as the number of true positives is much higher and the number of false positives much lower where the number of false negatives is decreased. This requires more investigation on the larger dataset.

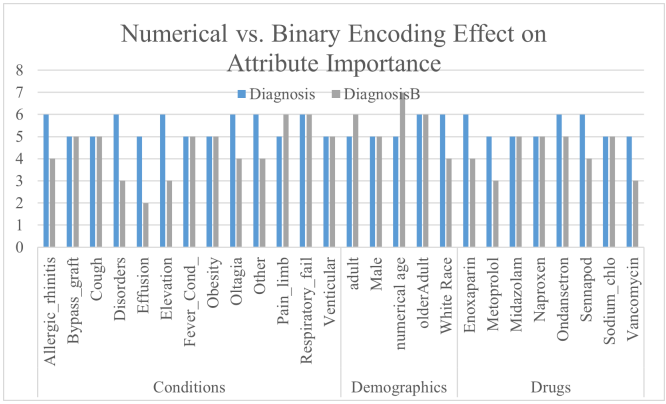

Fig. 14. Diagnoses Feature Importance: Five or more selection methods selected to feature in for DiagnosesB (binary) and Diagnoses (cumulative).

For attribute ranking over integrated Diagnoses vs. DiagnosesB data frame, we have illustrated the most prominent classification differences in Figure 14 for attributes that were selected as the most impactful by most of the eight methods we have applied to the Diagnoses and DiagnosesB

data frames. Five out of eight times our ranking algorithms rank the Effusion condition as important if evaluated on the Diagnoses data frame, and only twice when evaluated on the DiagnosesB data frame using the same procedure. A similar discrepancy is seen for the disorders and elevation conditions. For the demographic attributes, their importance is dampened in the numerical data frame, while drug importance is generally ranked higher for the same drug when numerical frame is used. This experiment demonstrates the importance of encoding rich information, both in terms of modeling performance and in terms of meaningful attribute importance selection.

VIII. Conclusion and Future Work

In this study, we describe the pipeline used on the N3C Enclave system and present the results of its application to the L3C challenge. Our findings indicate that the information in the "Conditions" data frame is complementary to that in the "Drugs" data frame, making the "Diagnoses" data frame a promising source for modeling conditions, drugs, and demographics. The "Observation" data frame was found to have limited discriminatory power with respect to diagnoses. Encoding the fields as binary or numerical had no impact on modeling performance across three data frames. PySpark processing on the Enclave system was not as efficient as on the desktop workstation, requiring early aggregation decisions that impacted the outcome. Our pipeline's chosen data frames and attribute selection resulted in a robust detector The number of false positives and false negatives indicates that we need to include more data sources in our analysis. The modeling performance was similar using random forests, decision trees, or gradient boosting. Given that model selection and hyper-parameter tuning were limited by the inadequate attribute space, we chose Random Forests as the fastest modeling algorithm on the Enclave. Our final model submission includes the novel cascade enhancement to ensure effectiveness as well as the prediction coverage of our modeling pipeline. The late fusion model described in Alg 1 is the most robust model as is evident by the highest number of true positives and lowest number of false negatives among all models. Techniques selected had to be robust to handle with over-fitting: retaining the noisy data allows for each algorithm to integrate its "own noise handling routine to ensure robustness" [29]

In future research, the calculation of entropy should consider multiple variables simultaneously and incorporate a better interaction among variables. This will result in a more robust and informative model. Individual conditions, procedures, and drugs should be incorporated into the model. To determine the most impactful factors, different methods must be used to classify the importance of the features in all data frames. The next steps are to automate feature aggregation and selection for all unique drugs, conditions, observations, and procedures fields and to improve the scalability of the Enclave processing. With access to 17,411,971 patient records with demographic information, 1,092,858 patient records with condition information, 16,908,022 patient records with observation information, and 14,613,563 patient records with drug information, as well as 207 conditions and 6,128 patients labeled as COVID-19-related and death records for 475,085 patients, N3C Enclave data provide ample opportunities to scale the pipeline and identify the most impactful attributes (N3C, 2023). The statistical learning pipeline also reveals interesting feature correlations between Long COVID and demographics, with decision tree and gradient boosting models showing that using clinical data underfits the prediction model, as evidenced by a high number of false positives.

IX. Acknowledgment

The analyses described in this document were carried out with data or tools accessed through the NCATS N3C Data Enclave [2] and N3C Attribution & Publication Policy v 1.2-2020-08-25b supported by NCATS U24 TR002306. This research was possible due to the patients whose information is included in the data and the organizations [30] and the scientists who have contributed to the ongoing development of this community resource [2].

References

[1] Assessment of correlations between risk factors and symptoms, Dec 2021.

[2] Melissa A Haendel, Christopher G Chute, Tellen D Bennett, David A Eichmann, Justin Guinney, Warren A Kibbe, Philip R O Payne, Emily R Pfaff, Peter N Robinson, Joel H Saltz, Heidi Spratt, Christine Suver, John Wilbanks, Adam B Wilcox, Andrew E Williams, Chunlei Wu, Clair Blacketer, Robert L Bradford, James J Cimino, Marshall Clark, Evan W Colmenares, Patricia A Francis, Davera Gabriel, Alexis Graves, Raju Hemadri, Stephanie S Hong, George Hripscak, Dazhi Jiao, Jeffrey G Klann, Kristin Kostka, Adam M Lee, Harold P Lehmann, Lora Lingrey, Robert T Miller, Michele Morris, Shawn N Murphy, Karthik Natarajan, Matvey B Palchuk, Usman Sheikh, Harold Solbrig, Shyam Visweswaran, Anita Walden, Kellie M Walters, Griffin M Weber, Xiaohan Tanner Zhang, Richard L Zhu, Benjamin Amor, Andrew T Girvin, Amin Manna, Nabeel Qureshi, Michael G Kurilla, Sam G Michael, Lili M Portilla, Joni L Rutter, Christopher P Austin, Ken R Gersing, and the N3C Consortium. The National COVID Cohort Collaborative (N3C): Rationale, design, infrastructure, and deployment. *Journal of the American Medical Informatics Association*, 28(3):427–443, 08 2020.

[3] June Yu. Gradient boosting public data modeling for the policy planning in education. Master's thesis, Texas State University, December 2022. Advisor: Jelenqa Tešić.

[4] R M Musal, T Ekin, and T Aktekin. Bayesian spatial analysis of socioeconomic determinants on covid-19 mortality: An application to the state of california. *Journal of Royal Statistical Society A*.

[5] Tatiana Cardona, Elizabeth A Cudney, Roger Hoerl, and Jennifer Snyder. Data mining and machine learning retention models in higher education. *Journal of College Student Retention: Research, Theory & Practice*, page 1521025120964920, 2020.

[6] Kuan Yan. Student performance prediction using xgboost method from a macro perspective. In *2021 2nd International Conference on Computing and Data Science (CDS)*, pages 453–459, 2021.

[7] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. *Information Fusion*, pages 785–794, 2016.

[8] Thomas Finley et al. Guolin Ke, Qi Meng. Lightgbm: A highly efficient gradient boosting decision tree. *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 3149–3157, 2017.

[9] Anna Veronika Dorogush, Vasily Ershov, and Andrey Gulin. Catboost: gradient boosting with categorical features support. *arXiv preprint arXiv:1810.11363*, 2018.

[10] Sercan ö. Arik and Tomas Pfister. Tabnet: Attentive interpretable tabular learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(8):6679–6687, May 2021.
[11] Ami Abutbul, Gal Elidan, Liran Katzir, and Ran El-Yaniv. Dnf-net: A neural architecture for tabular data. *CoRR*, abs/2006.06465, 2020.
[12] Sergei Popov, Stanislav Morozov, and Artem Babenko. Neural oblivious decision ensembles for deep learning on tabular data. *CoRR*, abs/1909.06312, 2019.
[13] Ravid Shwartz-Ziv and Amitai Armon. Tabular data: Deep learning is not all you need. *Information Fusion*, 81:84–90, 2022.
[14] Manu Joseph. Pytorch tabular: A framework for deep learning with tabular data, 2021.
[15] Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci. Deep neural networks and tabular data: A survey, 2021.
[16] Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on tabular data?, 2022.
[17] Justin T. Reese, Hannah Blau, Timothy Bergquist, Johanna J. Loomba, Tiffany Callahan, Bryan Laraway, Corneliu Antonescu, Elena Casiraghi, Ben Coleman, Michael Gargano, and et al. Generalizable long covid subtypes: Findings from the nih n3c and recover programs. 2022.
[18] Emily R Pfaff, Andrew T Girvin, Tellen D Bennett, Abhishek Bhatia, Ian M Brooks, Rachel R Deer, Jonathan P Dekermanjian, Sarah Elizabeth Jolley, Michael G Kahn, Kristin Kostka, and et al. Identifying who has long covid in the usa: A machine learning approach using n3c data. *The Lancet Digital Health*, 4(7), July 2022.
[19] Sebastian Köhler, Marcel H. Schulz, Peter Krawitz, Sebastian Bauer, Sandra Dölken, Claus E. Ott, Christine Mundlos, Denise Horn, Stefan Mundlos, Peter N. Robinson, and et al. Clinical diagnostics in human genetics with semantic similarity searches in ontologies. *The American Journal of Human Genetics*, 85(4):457–464, Oct 2009.
[20] Synapse. Nih long covid computational challenge, 2022. Accessed: 2023-01-12.
[21] OHDSI CDM Working Group. Omop cdm v5.3, Aug 2021.
[22] Inc. Odysseus Data Services. Athena website application. OMOP Vocabulary version: v5.0 16-JAN-23.
[23] Inc. Odysseus Data Services. Athena search 'post-acute covid-19. OMOP Vocabulary version: v5.0 16-JAN-23.
[24] Ehsan S Soofi. Capturing the intangible concept of information. *Journal of the American Statistical Association*, 89(428):1243–1254, 1994.
[25] Andrew A. Neath and Joseph E. Cavanaugh. The bayesian information criterion: Background, derivation, and applications. *WIREs Computational Statistics*, 4(2):199–203, 2011.
[26] M. Mitchell, A. S. Luccioni, N. Lambert, M. Gerchick, A. McMillan-Major, E. Ozoani, N. Rajani, T. Thrush, Y. Jernite, and D. Kiela. Measuring data, Dec 2022.
[27] D. E. Losada and J. M. Fernández-Luna. Precision, recall and f -score. in advances in information retrieval 27th european conference on ir research, ecir 2005, Mar 2005.
[28] S.A. Conrad et. al. The extracorporeal life support organization maastricht treaty for nomenclature in extracorporeal life support. a position paper of the extracorporeal life support organization. *In Advances in information retrieval 27th european conference on IR Research, ECIR 2005*, 198(4):447–451, 2018.
[29] Shivani Gupta and Atul Gupta. Dealing with noise problem in machine learning data-sets: A systematic review, Jan 2020.
[30] National Institute of Health. National covid cohort collaborative (n3c) data transfer agreement signatories, 2022. Accessed: 2023-01-12.