

TECHNOLOGICAL UNIVERSITY DUBLIN

SCHOOL OF COMPUTER SCIENCE

COHORT 2 MACHINE LEARNING DUBLIN

Data Mining – Assignment 2

Author

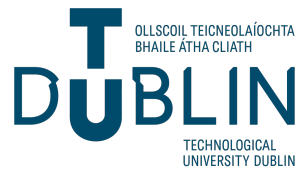
António SILVA

Iain HULL

Supervisor

Dr. Abubakr SIDDIG

April 28, 2024



Data Mining – Assignment 2

Silva, António	Hull, Iain
D23129331@mytudublin.ie	D23129341@tudublin.com

April 28, 2024

Contents

1	Definition of the problem	1
2	Data Exploration & Descriptive Analytics	2
2.1	Data Exploration	2
2.2	Features Description	2
2.3	Insights	2
2.3.1	Age	2
2.3.2	Pneumonia	3
2.4	Intubation and ICU	4
2.5	Other observations	4
3	Data Preparation	5
3.1	Defining the Target Variable	5
3.2	Data Normalization	5
3.3	Feature Removal	6
3.4	Outliers	7
3.5	Missing Values	7
3.6	Feature Selection	8
3.7	Dimension Reduction	8
3.8	Dataset Isolation	9
4	Details of Algorithms & Configuration	10
4.1	Model Selection	10
4.2	Custom Stacked Model	11
4.3	Feature Specific Models	11
5	Model Performance Metrics & Evaluation of Results	12
5.1	Plain Models	12
5.2	Stacked Feature Specific Models	15
5.3	Applying the Model	15
6	Identification of the most important variables	17
6.1	Pneumonia	18
6.2	Age	18
6.3	Classification Final	19
6.4	Diabetes	19
7	Reflections	19
8	Actions	20
A	Appendix	22
A.1	Features Analysis in Depth	i
A.2	Feature Statistics	x
A.3	Feature Insights	xi
A.4	Feature Correlation Matrix	xiii
A.5	Feature Ranking	xiv
A.6	Feature Importance per Metric	xv
A.6.1	AUC	xv
A.6.2	Precision	xvi
A.6.3	Recall	xvii
A.7	Feature Normalization Using Orange Data Mining	xviii
A.8	Code Listings	xix

A.9 Feature Analysis Correlation Matrix	xix
A.9.1 Feature Normalization	xx

List of Figures

1	Age by Disease	2
2	Age by Outcome	3
3	Pneumonia by Outcome	4
4	Intubation and ICU vs Death	4
5	Age \geq 100 Years old	7
6	Top 5 Ranked Features for predicting if a Patient needs Hospitalization	8
7	Orange Data Mining Data Splitting	10
8	Confusion Matrix Per Model	13
9	ROC Curves for the Outcomes	14
10	Triage Process	17
11	Feature Importance per Model	18
12	COVID-19 Feature Statistics	x
13	Age by Disease (Probability Histograms)	xi
14	Feature Correlation Matrix	xiii
15	Feature Ranking	xiv
16	Feature Importance for AUC	xv
17	Feature Importance for Precision	xvi
18	Feature Importance for Recall	xvii
19	Data Preparation using Edit Domain Widget	xviii
20	Data Preparation using Using Formula Widget	xix

List of Tables

1	Hospitalization Patient Required Condition Rules	5
2	Feature Normalization	6
3	Splitting the Data	9
4	Model Parametrization	11
5	Feature Categories	12
6	Original dataset Target Variable Balance	12
7	Evaluation Model Scores	13
8	Evaluation Scores for Stacked Models	16
9	Feature Analysis	i

Listings

1	Correlation Matrix For Feature Analysis	xix
2	Feature Normalization For COVID-19 Dataset - Functions	xx
3	Feature Normalization For COVID-19 Dataset - Process	xxi

1 Definition of the problem

The COVID-19 pandemic placed a strain in the health services in most countries. The disease is very infectious, resulting in sudden spikes in case numbers which quickly overwhelmed hospitals. The doctors, nurses and other front line health professionals were critical for responding to the pandemic, however their work placed them in extreme risk of infection. The health services ability to respond and ensure the best possible outcomes to patients depended on the health of front line staff.

Every effort was made with PPE equipment and operational procedures to protect staff. A critical aspect of this was to reduce contact between patients and staff and ensure that as many patients could be treated at home as possible.

The Mexican government have sponsored a data mining study to examine how simple health data could be used to automate the triage of COVID-19 patients. They hope this will improve their response to new waves of COVID-19 or similar respiratory illness. The goal is to improve their triage process and safely reduce the number of patients that require contact with a health professional.

Initially patients are triaged outside the hospital to limit the chance of cross infection. Normally patients are triaged by a health professional as soon as they present, this person decides if they need to be admitted to hospital for treatment. However before they are triaged patients will be asked to complete a simple questionnaire. This data together with a COVID-19 test result will enable a non-health professional to efficient route them as follows:

- Low risk patients will be sent home without further contact with front line staff;
- High risk patients will be streamlined for hospital care with the least possible interaction;
- If the risk assessment is undetermined patients will triaged by a health professional who will make the final decision on admission.

While the goal is to minimise the number of patients requiring triage it is very important to do so safely and with the utmost care for the patients. Therefore false negatives where patients are sent home in error must be minimised. Likewise false positives of where patients are hospitalised in error must be minimised to preserve capacity. If there is any doubt the patient must be triaged by a health professional.

The government has provided a dataset (Nizri 2023) with details and test results for 1,048,575 previous COVID-19 patients. This is used to develop a predictive model to identify patients who require admission to hospital, no admission and those with require further triage.

A well trained model can aid medical staff in making quick data-driven decisions helping:

- Medical resources prioritization;
- Enhance patient care;
- Increase the survival rate.

We can also correlate some key conditions with some patients conditions such as:

- Age;
- Obesity;
- Diabetes;
- Hypertension.

2 Data Exploration & Descriptive Analytics

2.1 Data Exploration

The COVID-19 Dataset (Nizri 2023), provides a comprehensive view of the impact of the COVID-19 pandemic focusing on various aspects of the patients such as:

- Hospitalization;
- Symptoms;
- Medical Records.

This dataset is useful to *identify high risk factors and predict hospitalizations*. It also contains extensive anonymized data related to patients with their conditions. It features *21 different attributes* and *1,048,575 unique patient records*.

Boolean features are identified by 1 meaning ‘yes’ and 2 meaning ‘no’. Missing data is specified out-of-range values like 97 to 99 or 9999.

2.2 Features Description

So the features provided from the dataset provides a comprehensive profile of patient health status and the medical care received. You can find a deep analysis per feature in the appendix section [Features Analysis in Depth](#).

In terms of null values, we have a minimal number of empty entries (always close to 0%), except when the data does not make sense. For example, in the case of pregnancy, we have 50% missing data. This is reasonable because half of the dataset consists of men. In the case of intubation or ICU data, there is 82% missing data. This is likely because the majority of the patients were not hospitalized.

You can get all the statistics resume in the Figure 12.

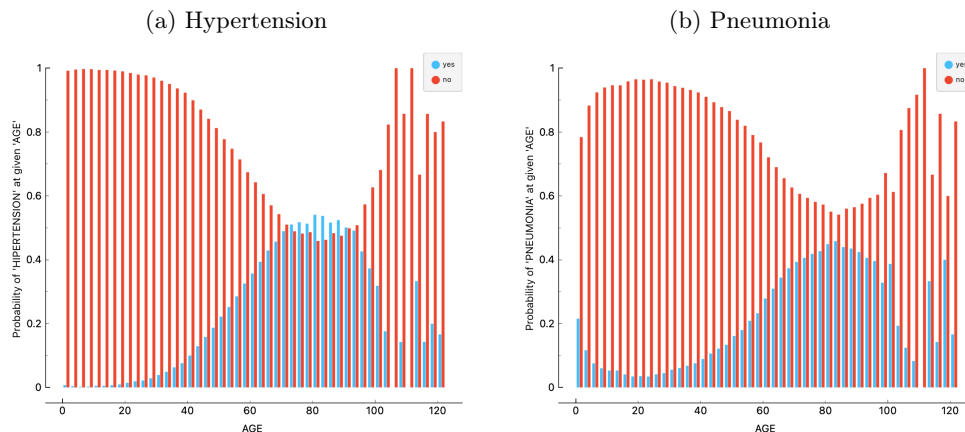
2.3 Insights

2.3.1 Age

Age appears to be one of the primary factors in determining whether a patient will be hospitalized or survive COVID-19.

As you age, you become more susceptible to various diseases, and your immune system weakens, increasing the likelihood of experiencing severe complications if you contract COVID-19. (Figure 1 and Figure 13).

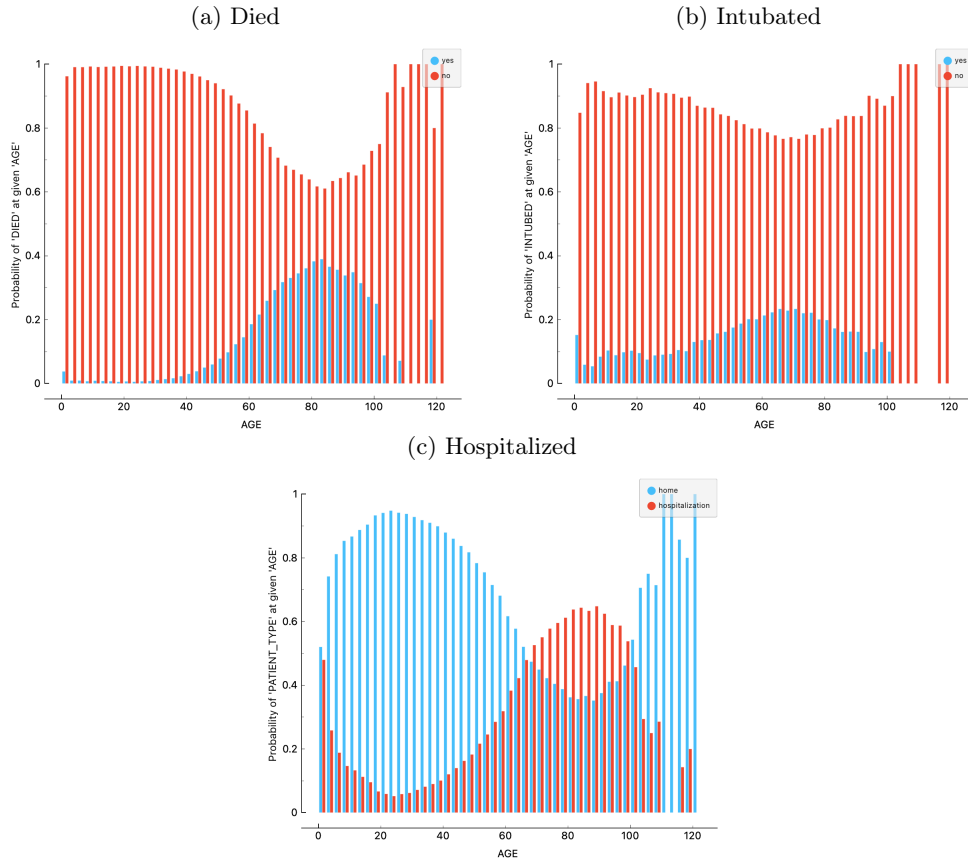
Figure 1: Age by Disease



Additionally, older individuals are more likely to experience the following outcomes if they contract COVID-19:

- Being Hospitalized;
- Being Intubated;
- Dying.

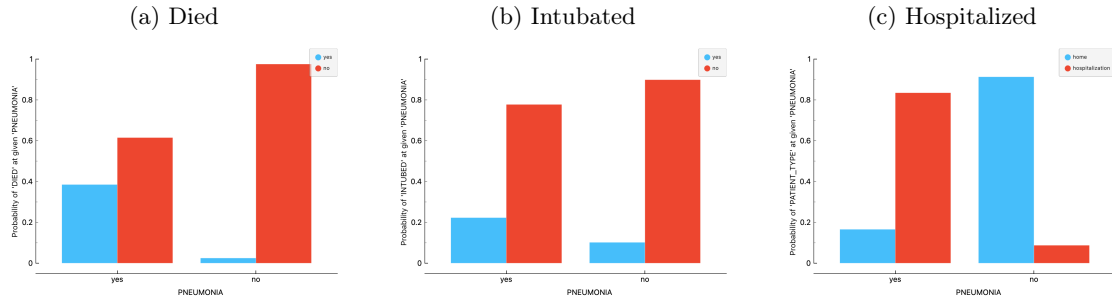
Figure 2: Age by Outcome



2.3.2 Pneumonia

Pneumonia significantly complicates the condition of COVID-19 patients, often leading to more severe symptoms and outcomes. Effective management and early intervention are crucial to mitigate the adverse effects associated with pneumonia in these patients.

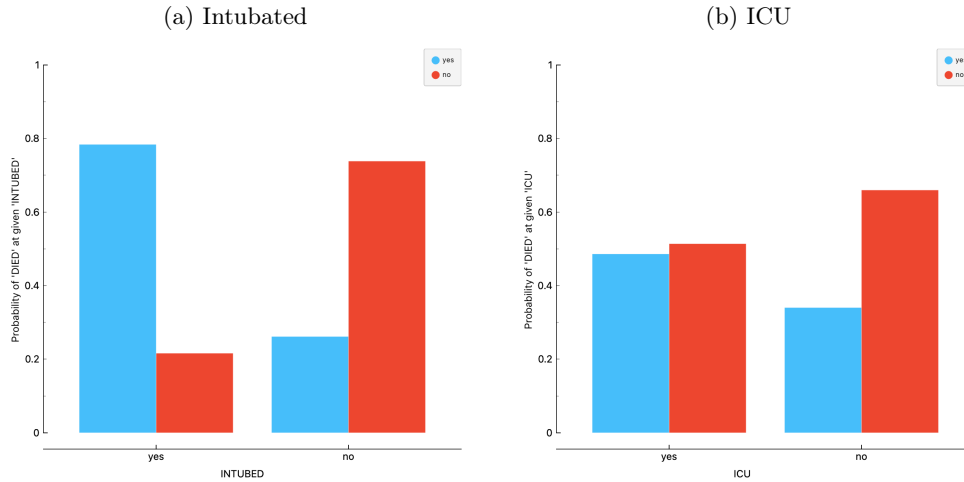
Figure 3: Pneumonia by Outcome



2.4 Intubation and ICU

Patients with COVID-19 who require intubation or admission to an intensive care unit (ICU) are generally at a higher risk of mortality compared to those who do not require such intensive interventions.

Figure 4: Intubation and ICU vs Death



2.5 Other observations

You can discover insights by creating and examining the correlation matrix among features. An intriguing observation from the matrix suggests that if you have one condition, such as renal disease or diabetes, you are also likely to have other related conditions.

Observing the correlation matrix, these are the most significant (Figure 14):

- Only women are pregnant;
- Only if you are hospitalized and in ICU you can be intubated;
- Only if you are hospitalized we can be in ICU;
- There are some correlation between the conditions: (Obesity, COPD and Diabetes) and others.

3 Data Preparation

3.1 Defining the Target Variable

Our goal is to enhance and automate the triage process for patients arriving at the hospital diagnosed with COVID-19. Implementing an automated triage system, we aim to streamline the initial assessment and quickly determine the appropriate level of care required for each patient. The outcome of this automated system will categorize patients into one of three categories:

Maybe This outcome is indeterminate, suggesting that the case isn't clear-cut. Such patients might need further analysis either through a follow-up assessment after some time or an immediate review by a medical professional.

No Indicates that the patient does not require hospitalization. These patients are stable enough to be sent home with instructions for self-care and possibly follow-up recommendations.

Yes Signifies that the patient requires hospital admission due to the severity of their symptoms or underlying health conditions that pose increased risk.

By classifying incoming patients into these categories, the hospital can reduce contact with front line medical staff as well as optimize resource allocation and reduce wait times. This system not only enhances operational efficiency but also improves patient outcomes by facilitating immediate and accurate decision-making at the point of entry ¹.

The target variable prediction is true if the following conditions are true in the dataset:

Table 1: Hospitalization Patient Required Condition Rules

Patient Type	ICU	Intubed	Died	Required Treatment
NA	NA	NA	Yes (\neq 9999 – 99 – 99)	Yes
NA	NA	Yes (1)	No (= 9999 – 99 – 99)	Yes
NA	Yes (1)	NA	No (= 9999 – 99 – 99)	Yes
Home (1)	No	NA	No (= 9999 – 99 – 99)	No
Hospitalized (1)	NA	NA	No (= 9999 – 99 – 99)	Maybe

3.2 Data Normalization

This dataset includes numerous binary variables, each encoded with distinct meanings. We found these hard to reason about when analysing the data so we converted them to short text values. While we considered using the *Edit Domain Widget* (Mining 2015) from *Orange Data Mining* as cited in , we found the task to be cumbersome, involving considerable repetition, and challenging in terms of replicating the logic applied previously (Figure 5 and Figure 20).

Instead, we utilized a Scala Script (Listing 2 and Listing 3) that reads the original dataset and transforms the data as follows:

¹You can find the transformation details in the function `makeRequiredTreatment` in the Listing 2

Table 2: Feature Normalization

Feature	Transformation
sex	$1 \rightarrow \text{female}, 2 \rightarrow \text{male}$
age	
classification	
patientType	$1 \rightarrow \text{home}, 2 \rightarrow \text{hospital}$
pneumonia	$1 \rightarrow \text{yes}, 2 \rightarrow \text{no}$
pregnancy	$\text{sex} = \text{female} \wedge 1 \rightarrow \text{yes}, \text{sex} = \text{female} \wedge 2 \rightarrow \text{no}, \text{sex} = \text{male} \rightarrow \text{na}$
diabetes	$1 \rightarrow \text{yes}, 2 \rightarrow \text{no}$
copd	$1 \rightarrow \text{yes}, 2 \rightarrow \text{no}$
asthma	$1 \rightarrow \text{yes}, 2 \rightarrow \text{no}$
inmsupr	$1 \rightarrow \text{yes}, 2 \rightarrow \text{no}$
hypertension	$1 \rightarrow \text{yes}, 2 \rightarrow \text{no}$
cardiovascular	$1 \rightarrow \text{yes}, 2 \rightarrow \text{no}$
renal_chronic	$1 \rightarrow \text{yes}, 2 \rightarrow \text{no}$
other_disease	$1 \rightarrow \text{yes}, 2 \rightarrow \text{no}$
obesity	$1 \rightarrow \text{yes}, 2 \rightarrow \text{no}$
tobacco	$1 \rightarrow \text{yes}, 2 \rightarrow \text{no}$
usmer	
medical_unit	
intubed	$1 \rightarrow \text{yes}, 2 \rightarrow \text{no}$
icu	$1 \rightarrow \text{yes}, 2 \rightarrow \text{no}$
date_died	$9999 - 99 - 99 \rightarrow \text{no}, \text{otherwise} \rightarrow \text{yes}$

3.3 Feature Removal

To streamline our model for predicting hospitalization requirements, we have decided to create a target variable named ‘Required Hospitalization.’ Consequently, any features used to construct ‘Required Hospitalization’ will be excluded from the predictive modeling phase and reserved solely for the verification process:

- Patient Type;
- Intubed;
- ICU;
- Date Died.

By removing these features from the modeling stage, we ensure that the model develops the ability to predict hospitalization based on less direct indicators, thereby enhancing its applicability and effectiveness in real-world triage scenarios.

A medical unit is typically composed of a team of doctors and nurses operating within a larger entity, such as the armed forces, a prison, or a hospital. Given this definition, we conclude that the presence of a medical unit does not directly influence the decision regarding a patient’s need for hospitalization. Therefore, we have decided to exclude the `MEDICAL_UNIT` feature from our analysis.

The variable `USMER`² was removed from our analysis because it indicates whether a patient was hospitalized, making it a consequence rather than a predictor. Including it could potentially bias our predictive modeling efforts, as it does not contribute to determining the factors leading to hospitalization but rather describes an outcome. Thus, to maintain the integrity of our model and ensure it accurately identifies true predictive variables, `USMER` has been excluded.

So in summary we also remove the following features:

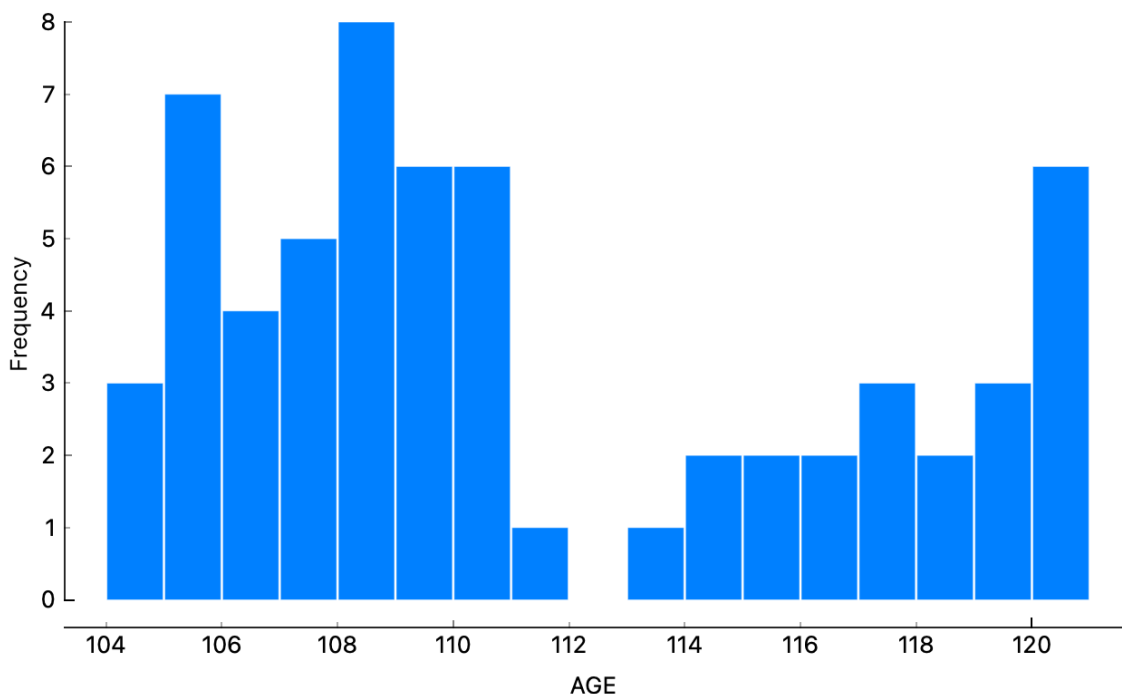
²`USMER` refers at which type a patient received in a medical unit.

- Medical Unit;
- USMER.

3.4 Outliers

We have 208 cases of patients aged between 100 and 121 years. It seems improbable to have such a high number of patients over 104 years old. However, since these cases only represent 208 out of 1 million and are likely to be older patients, we believe this data will not significantly impact the final calculations.

Figure 5: Age ≥ 100 Years old



3.5 Missing Values

In the dataset we're discussing, most of the features are quite complete, with each having about 3,000 missing entries. Despite this, these missing entries only account for a negligible portion of the entire dataset—essentially 0% of all the data, indicating that the overall impact on the dataset's integrity is minimal.

However, one particular feature, 'pregnancy status' stands out due to a higher rate of missing data. This feature is missing 50% of its values. The reason for this significant number of missing values is linked to the data normalization process. Specifically, the data entries related to males are marked as missing in the pregnancy status field, because pregnancy is not applicable to them.

Additionally, the feature for pneumonia has a 2% rate of missing values from the sample data. Given this relatively low proportion of missing data, we plan to impute these missing entries with the most common response, which is 'no'. This approach is justified by the high likelihood – 87% – that a given entry would not have pneumonia, based on the data we have.

3.6 Feature Selection

Because we have only 14 Features we won't apply any dimension reduction but we can still identify the most relevant features for the target variable. This is an important step because we can verify if our training models will pick the most important variables to classify the patients that need to be hospitalized.

In the article by Jason Brownlee (Brownlee 2020), it is suggested that when dealing with a large number of categorical variables, the most effective metrics to assess their importance are *Chi-Squared Statistics* (χ^2) and *Information Gain*. These metrics are particularly useful for identifying which variables have the strongest relationships with the target variable, thereby helping to refine and improve the performance of predictive models. The Figure 6 shows the top χ^2 ranked variables³.

Figure 6: Top 5 Ranked Features for predicting if a Patient needs Hospitalization

		#	Info. gain	Gain ratio	χ^2
1	N AGE		0.095	0.047	94278.454
2	C PNEUMONIA	2	0.258	0.451	62754.060
3	C CLASIFFICATION_FINAL	7	0.039	0.024	40450.411
4	C HIPERTENSION	2	0.038	0.061	10210.767
5	C DIABETES	2	0.044	0.083	9352.530

3.7 Dimension Reduction

We have decided not to apply any dimension reduction techniques to train the data due to several key reasons:

Categorical Variables : Our dataset consists predominantly of categorical variables, which do not lend themselves well to traditional dimension reduction techniques like PCA that are better suited for continuous data;

Interpretability : Dimension reduction can often obscure the meaning of the original variables. Given the importance of each variable in our dataset for clinical decision-making and understanding patient outcomes, maintaining the interpretability of the data is crucial. Reducing variables like Tobacco, Obesity, Diabetes or Pneumonia or Age will contribute for lack of understanding of the outcomes;

Complex Relationships The relationships between variables in our dataset are complex and may not be linear. Dimension reduction techniques, particularly those based on linear assumptions, might not capture these nuances effectively, leading to sub-optimal model performance.

Considering we only have 15 features that are crucial for the training and explainability of the model's outcomes, we have decided not to reduce any features for training. This decision stems from a few key points:

Importance Each feature has been identified as significant in impacting the model's predictions. Removing any of these could jeopardize the accuracy and reliability of the model;

Dataset With just 15 features, our dataset is relatively concise. This manageable number of features facilitates a straightforward training process and clear interpretation of results, making further reduction unnecessary.

³The full ranked variables ranking are in Figure 15.

3.8 Dataset Isolation

To enhance our model’s effectiveness we have divided the dataset as detailed in the Table 3. This division is aimed at preventing the model from merely training the data avoiding overfitting. Isolating the data also ensures that the model remains robust across various data samples (Not merely the ones on which it was trained.). Splitting the data also facilitates a fair evaluation. As assessing the model on previously unseen data yields more accurate results and better reflects real-world scenarios.

Table 3: Splitting the Data

Dataset	Size	Comment
Training	692060 (66%)	This is the largest portion of the data and is used to train the model. The model learns to recognize patterns and make decisions based on this dataset.
Verification	235300 (22%)	The validation set is used to tune the hyperparameters of the model and prevent overfitting. It acts as a check during training to ensure that the model not only learns the training data well but can also apply this learning to new data. It is used to evaluate model performance during the training process.
Testing	121215 (12%)	After the model has been trained and validated, the test set is used to assess its performance. This data set is never used in the training phase and serves as a new, unbiased platform to evaluate how well the model can generalize its learning to data it has never seen before.

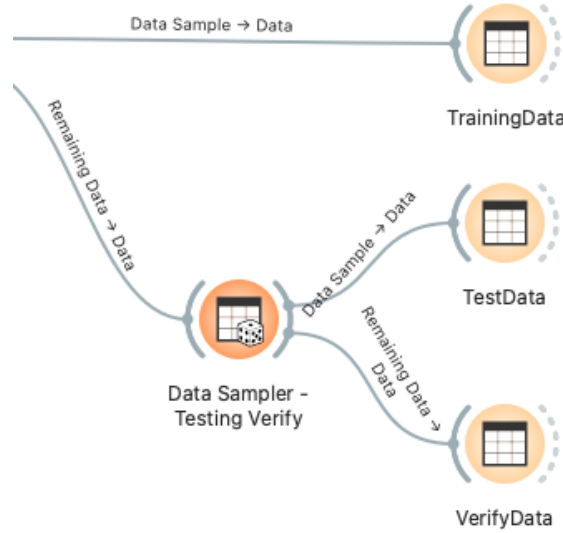
To split the data we used the *Orange Data Mining Data - Sampler Widget* (Figure 7) with the options:

- Replicable (deterministic) sampling;
- Stratify sample (when possible).

The Replicable sampling ensures that sampling patterns can be consistently applied across different users.

Stratified splitting ensures that each division of the dataset accurately reflects the overall dataset in terms of key characteristics’ proportions. This approach is particularly crucial for handling imbalanced datasets.

Figure 7: Orange Data Mining Data Splitting



4 Details of Algorithms & Configuration

4.1 Model Selection

Given that 13 out of the 14 features in our dataset are categorical⁴, we will primarily focus on models that are particularly well-suited for handling categorical variables. This approach ensures that we leverage the characteristics of our dataset effectively, optimizing the model's performance by utilizing algorithms designed to manage and interpret categorical data efficiently.

Decision Trees are also a popular choice for handling categorical data. It also handles well in conjunction with numerical and categorical data. The rules derived from decision trees are straightforward and easy to interpret. This can be very useful in our case when the decisions need to be justified.

Starting with *Gradient Boosting - CatBoost*⁵ might be particularly beneficial due to its ease of use with categorical variables and strong default performance (Yandex 2024).

We will also employ another decision tree algorithm, *Random Forest*. This algorithm not only enhances accuracy beyond that of a traditional decision tree but also offers greater resistance to overfitting (ibm 2024b).

The last is *Neural Networks*. This one can learn complex patterns and relationships between variables. With the size of our dataset 1M this could be a useful approach (ibm 2024a).

In terms of parametrization we choose the most recommended parameters for our dataset based on the documentation provided in Orange Data Mining and scikit-learn⁶.

⁴With only the 'Age' feature being numerical

⁵Cat Boosting is based on Decision Trees.

⁶Orange Data Mining employs scikit-learn's classification and prediction algorithms behind the scenes.

Table 4: Model Parametrization

Model	Parameter	Value
Gradient Boosting (catboost)	Number of trees	1000
	Learning rate	0.03
	Replicable training	true
	Regularization (Lambda)	3
	Limit depth of individual trees	16
	Fraction of features for each tree	1
Random Forest	Number of trees value	100
	Number of attributes considered at each split	4
Neural Network	Neurons in hidden layers	100,100
	Activation	ReLu
	Solver	Adam
	Regularization	$\alpha = 0.0001$
	Maximal number of iterations	200
	Replicate training	true

4.2 Custom Stacked Model

The models above performed well however we wanted better results given the nature of the predictions we are trying to make and the consequences of false positives or false negatives.

Initially we tried stacking these models with in Orange Stacking widget (used with Logistic Regression). This trains a model to react to the outputs of the other models. This model had a number of issues it was very slow to train and it complicates the explainability of the model. It improved on the score over the other models but it did not make a significant improvement to false positives or false negatives. As a result the Stacking widget was discarded and a custom stacking approach was followed instead.

Our primary concerns were:

- Only recommend sending a patient home when they did not need hospital treatment;
- Only recommend admitting a patient when they needed hospital treatment;
- Triage when ever there is a doubt;
- Ability to explain or justify our predictions.

To this end we combined the predictions from multiple models as follows:

- yes if and only if all models predict yes;
- no if and only if all models predict no;
- maybe otherwise.

These rules are easy to explain and justify and are well aligned with our goals.

4.3 Feature Specific Models

Next we decided to train our models to consider specific features.

Table 5: Feature Categories

Category	Features
Respiratory Illness	age, pneumonia, copd, asthma, cardiovascular, tobacco, classification
Other Illness	age, diabetes, hypertension, other_disease, renal_chronic, obesity, cardiovascular, classification
Sex & Pregnancy	age, sex, pregnancy, classification

These feature categories were then used to train Random Forest and Gradient Boosting models as above. Finally we examined how combining these models together with the rules above improved the results. We kept the Neural Network model trained on all features in case this discovered some patterns across features that the feature categories would miss.

We were not able to implement the rules above in Orange, so we evaluated the results in a python notebook. First we used Orange to generate predictions using our base models and the test data. The results were then imported into Collab where the prediction columns were combined and added to the dataframe. Finally python was used to score different combinations of predictions below.

5 Model Performance Metrics & Evaluation of Results

5.1 Plain Models

When examining the distribution of our dataset by the target variable, it is evident that the majority of the results are ‘no’, followed by ‘maybe’ and ‘yes’. This distribution can be confirmed in Table 6.

Table 6: Original dataset Target Variable Balance

Required Treatment	Size	Comment
no	841668 (80%)	Patients with COVID 19 that does not required hospitalization.
maybe	117155 (11%)	Patients with COVID 19 that required further analysis (Inconclusive by the data).
yes	89752 (9%)	Patients with COVID 19 that requires hospitalization.

Due to the presence of an *imbalanced target variable* in our dataset, we must exercise caution when analyzing the performance and reliability of our model. The predominance of the ‘no’ class could introduce a bias, skewing predictions toward this outcome.

Furthermore, it is crucial to monitor the model’s performance regarding the minority classes, ‘maybe’ and ‘yes’, which are pivotal to our predictive objectives. These outcomes are critical as they relate directly to our goal of accurately identifying severe cases of COVID-19, ensuring that these individuals receive hospital care rather than being inappropriately sent home. Therefore, a thorough evaluation of the model’s accuracy for these classes is essential to avoid potentially grave errors in clinical decision-making.

So the most important metrics to analyze a Imbalanced Dataset in a classification model are:

Precision Precision measures the positive predictions made⁷ (Equation 1). Precision is particularly important where the cost of a false positive is high;

Recall On the other hand Recall measures the model’s ability to identify all positives in the dataset⁸. This metric is specially important in our case because In a medical diagnosis

⁷Off all the instances classified as positive, how many are positive?

⁸Off all the actual positives, how many were identified correctly by the model?

for COVID-19 we need to reduce the false negatives. This means reducing the risk of not treating patients that need. The Equation 2 shows that there's a trade-off between precision and recall. Improving the precision will reduce the recall and vice-versa;

AUC AUC provides an aggregate performance across all possible classification thresholds. AUC needs to be closer to 1 for a good classifier. Any value ≤ 1 is a worthless classifier with no better accuracy than a random chance.

$$Precision = \frac{TruePositives(TP)}{TruePositives(TP) + FalsePositives(FP)} \quad (1)$$

$$Recall = \frac{TruePositives(TP)}{TruePositives(TP) + FalseNegatives(FN)} \quad (2)$$

After following running the models we got the results presented in the Table 7.

Table 7: Evaluation Model Scores

Model	Recall	Prec	AUC	F1	CA
Catboost	0.867	0.850	0.913	0.856	0.867
Neural Network	0.866	0.848	0.912	0.855	0.866
Random Forest	0.857	0.840	0.886	0.847	0.857

We also have the confusion in the Figure 8 and the Roc Curves for every output in Figure 9.

Figure 8: Confusion Matrix Per Model

(a) Catboost

		Predicted			Σ
		maybe	no	yes	
Actual	maybe	52.4 %	5.7 %	31.0 %	26290
	no	18.2 %	92.3 %	9.4 %	188870
	yes	29.4 %	2.0 %	59.6 %	20140
	Σ	17446	199418	18436	235300

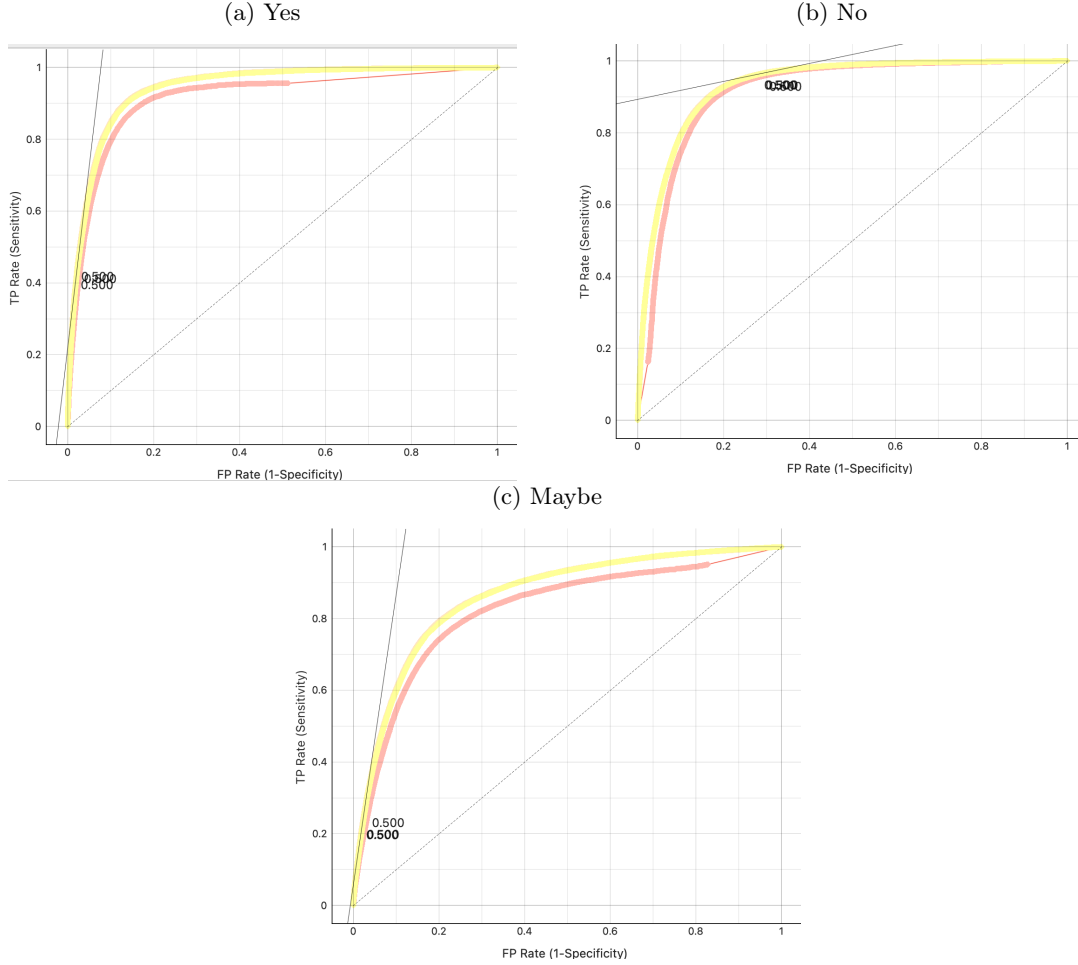
(b) Neural Networks

		Predicted			Σ
		maybe	no	yes	
Actual	maybe	51.8 %	5.8 %	31.1 %	26290
	no	17.9 %	92.1 %	9.3 %	188870
	yes	30.3 %	2.1 %	59.6 %	20140
	Σ	17752	199805	17743	235300

(c) Random Forest

		Predicted			Σ
		maybe	no	yes	
Actual	maybe	47.4 %	5.7 %	33.4 %	26290
	no	21.7 %	92.1 %	11.3 %	188870
	yes	30.9 %	2.1 %	55.3 %	20140
	Σ	18420	198412	18468	235300

Figure 9: ROC Curves for the Outcomes



As anticipated, given that our dataset predominantly consists of categorical variables, the most effective classification model for this scenario is *Catboost*. Unexpectedly, Neural Networks ranked second, outperforming Random Forests, which came in last. The superior performance of *Neural Networks* over Random Forests could be attributed to the non-linear relationships among the features, or possibly because the Random Forest model became overfitted.

Regarding our results, it is apparent that predicting when a patient does not need hospitalization is both straightforward and accurate. We achieve a commendable accuracy rate of 90.7% in this scenario, with only a 2.7% error rate⁹.

However, the same cannot be said for predicting the necessity of hospitalization. Predicting whether a patient requires treatment proved to be challenging, with an accuracy of only 61.1% in confidently determining the need for hospitalization. An additional 32.2% of cases required further analysis, and unfortunately, 6.7% of patients were incorrectly sent home. This 6.7% represents a significant proportion of misclassified cases. To improve accuracy, the algorithm might benefit from training on a larger dataset of hospitalized patients. It's possible that even a dataset size of one million might not suffice, or perhaps the model requires additional features or data to enhance the reliability of these predictions.

As for predicting cases that fall into the 'maybe' category, which as anticipated, is the most challenging to predict, our model achieves less than 49.2% accuracy. On a positive note, 30.7% of these cases were escalated to hospitalization, and 20.1% were not, which suggests some level

⁹Here, we only consider the 'yes' value because 'maybe' indicates a need for further analysis.

of effective triage. Since ‘maybe’ inherently implies the need for human intervention and further analysis, the implications of these predictions are not immediately dire. Nevertheless, similar to the ‘yes’ predictions, enhancing model accuracy in this area would likely require more data for training.

5.2 Stacked Feature Specific Models

Once we discovered the strengths and limitations of the standard models, we tested a number of custom stacked models. We want our stacked models to make conservative predictions, so they only send a patient home or straight to hospital if all sub-models agree. If any sub-model does not agree the patient will be sent for triage. The sub-models were trained on three specific feature sets, these were deliberately chosen to make unanimous predictions harder and the stacked model safer.

The model returns ‘yes’ when a patient should be treated in hospital and ‘no’ when the patient should recover at home. It returns ‘maybe’ when a patient should be triaged or when it cannot decide. The three way nature of this decision is very important, saying ‘maybe’ when a patient should recover at home is a true positive result, likewise saying ‘maybe’ when a patient should be treated in hospital is also a true positive. When evaluating the effectiveness of these models we should compare them with whether the patient needs hospitalisation rather than the target variable used to train the model. This results in an adjusted confusion matrix as seen in Table 8. We treat the home and hospital like two different questions and then measure the precision and recall for both. Given the nature of the decision the model is making, saying ‘no’ when the patient should be treated in hospital has the worst outcome for the patient. As a results we want this value to be as small as possible. A corollary of this is that we want the precision for this decision to be as high as possible.

The first observation is that the Stacked Gradient Boosting performs better than Stacked Random Forest. This is to be expected given the categorical nature of the data.

Secondly including the Neural Network in the stack improves both the Stacked Gradient Boosting and the Stacked Random Forest. It reduced the false positives for Stacked Gradient Boosting from 5% to 4.8%. And it reduced the false positives for Stacked Random Forest from 5.1% to 4.7%. This indicates the Neural Network has found other characteristics in the data not found in the other models. When this is included in our conservative stacking algorithm it reduces the ‘yes’ and ‘no’ results in favour of more ‘maybe’s’.

Thirdly it is interesting that the Stacked Random Forest with Neural Network performs better than Stacked Gradient Boosting (Catboost) with Neural Network. This shows that the results generated from Gradient Boosting are close to the results generated by the Neural Network

Finally when we stack all models together we see a much larger improvement. The false positives are 4.1% and the precision is 0.786. Ideally the precision should be higher for a medical application, however there are some times when this could be acceptable as discussed in Section 5.3.

5.3 Applying the Model

The goal of applying this model is to reduce contact between patients and frontline health professionals. Ordinarily they would triage all patients however since COVID-19 is so infectious and the Personal Protective Equipment (PPE) required to protect staff is in short supply in the pandemic, the health system must protect its staff and conserve PPE to remain functional. This is an emergency situation and not a normal application of a medical model.

Compare the flow of patients in Figure 10, with the normal triage process 100% of patients need to be triaged, when the patients are screened first by the model only 20% of patients have to be triaged. We can clearly see 4% of the population are sent home in error. Not all patents required the same level of treatment in hospital, when we cross reference the 4% with the sickest patients requiring ICU we see this grave error is only 0.8% of the population. Hopefully any patients sent home by the screening process would be able to return to the hospital as their symptoms worsen.

Table 8: Evaluation Scores for Stacked Models

(a) All Models Stacked

	maybe	no	yes	Precision	Recall
home	0.067	0.741	0.001	0.998	0.952
hospital	0.134	0.041	0.016	0.786	0.991

(b) Stacked Gradient Boosting (Catboost) with Neural Network

	maybe	no	yes	Precision	Recall
home	0.041	0.764	0.004	0.995	0.943
hospital	0.115	0.048	0.027	0.746	0.971

(c) Stacked Gradient Boosting (Catboost)

	maybe	no	yes	Precision	Recall
home	0.039	0.765	0.004	0.993	0.941
hospital	0.111	0.050	0.029	0.736	0.963

(d) Stacked Random Forest with Neural Network

	maybe	no	yes	Precision	Recall
home	0.049	0.756	0.004	0.995	0.945
hospital	0.120	0.047	0.024	0.754	0.975

(e) Stacked Random Forest

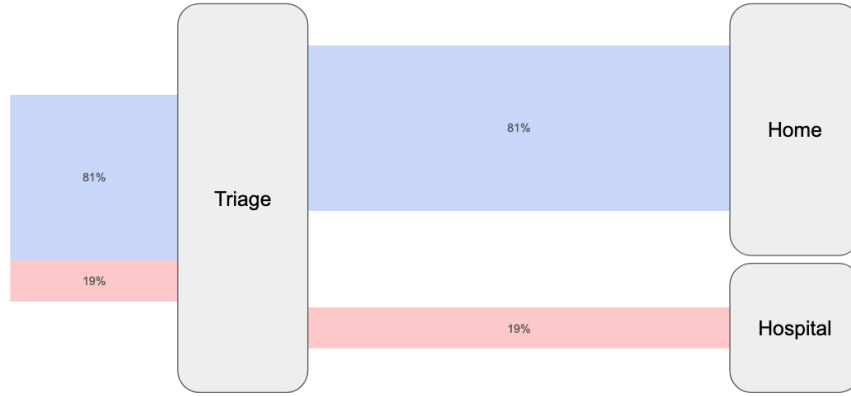
	maybe	no	yes	Precision	Recall
home	0.046	0.758	0.005	0.994	0.941
hospital	0.115	0.051	0.025	0.735	0.967

When we examine the patients sent straight to hospital we see this accounts for 1.7% of the population, included in that is 0.1% who do not require hospitalisation. Given the population routed straight to hospital is so small, it probably makes sense to include this group in the triage process. As it would only change the triage reduction from 80% to 78%.

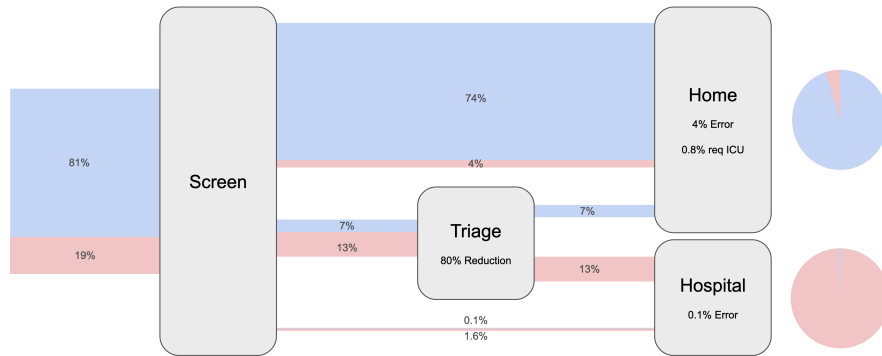
Based on this evaluation we could only recommend applying this model and the screening process in the most grave emergency situation like the COVID-19 pandemic, where frontline health professionals were risking their lives to treat patients and their infection rates risked the efficacy of the health system as a whole.

Figure 10: Triage Process

(a) Normal Triage Process



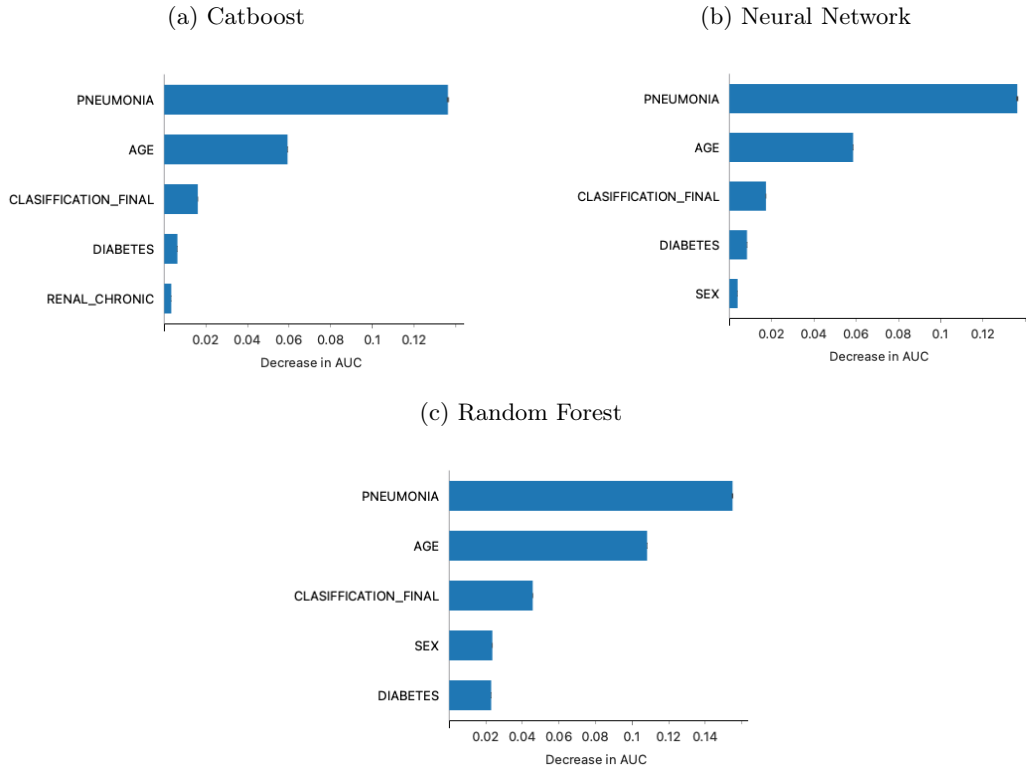
(b) Triage Process with our Model



6 Identification of the most important variables

To determine the most significant features, we employed the *Feature Importance Widget* from Orange Data Mining (Mining 2015). This tool analyzes the provided data to calculate the contribution of each feature towards the prediction, thereby assessing its impact on the target variable.

Figure 11: Feature Importance per Model



Based on the variable rankings presented in Figure 6, we can observe that the models have identified the variables in alignment with our expectations. This comparison confirms that the models are accurately prioritizing the variables as anticipated. We identified 4 main variables that could affect if a patient with COVID-19 needs to be hospitalized or not:

- Pneumonia;
- Age;
- Classification Final;
- Diabetes.

6.1 Pneumonia

Pneumonia is a serious and common complication of COVID-19, especially in severe cases. When someone is infected with the virus that causes COVID-19, the infection can spread to the lungs, leading to pneumonia. This is the main indicator the a Patient needs to be hospitalized urgently.

6.2 Age

Age is a significant factor in the impact and outcomes of COVID-19 infection. The risk of severe illness from COVID-19 increases with age, with older adults being at the highest risk for severe illness, complications, and mortality.

6.3 Classification Final

The term ‘Classification Final’ pertains to the COVID testing result of a patient. As expected, this is a crucial factor because a positive test result is necessary for a patient to be treated for COVID-19.

6.4 Diabetes

People with both type 1 and type 2 diabetes are more likely to have severe illness and complications from COVID-19 compared to people without diabetes. This increased risk can be attributed to several factors, such as:

Immune System Impairment Diabetes can weaken the immune system, making it less able to fight off infections;

Coexisting Conditions people with diabetes often have other conditions such as hypertension, cardiovascular disease, or obesity, which are themselves risk factors for severe COVID-19

7 Reflections

Our study highlights the complex relationship between COVID-19, infectious diseases, and chronic health conditions such as diabetes and pneumonia.

The link between diabetes and COVID-19 is particularly noteworthy, as individuals with diabetes face a heightened risk of severe outcomes when infected with the virus. Similarly, pneumonia is recognized as one of the most severe complications associated with COVID-19, often leading to critical cases.

Age also plays a crucial role and is a strong predictor of the need for hospitalization. Patients aged over 50 are significantly more likely to require hospital treatment. This increased vulnerability is partly due to the diminished immune response seen in older individuals, coupled with a higher incidence of comorbid conditions that exacerbate the severity of COVID-19 outcomes.

We were dissatisfied with the model’s inability to accurately predict hospitalization needs, achieving only a 61.1% accuracy rate. Our primary concern is the 6.2% of patients who require treatment but were mistakenly not recommended for hospitalization by our model. This issue largely stems from the imbalanced nature of our dataset.

To reduce errors in patient triage, we employed a stacking technique to enhance prediction accuracy by integrating multiple algorithms. We organized the features by category as outlined in Section 5.2. This approach successfully decreased the error margin from 6.2% to 4%, marking a substantial improvement.

Improving the dataset is another effective strategy for enhancing the accuracy of predictions in patient triage. To enhance the quality of our data and potentially improve the model’s performance, we propose the following steps:

Increase the dataset size Given the complexity of the scenario, a dataset of 1 million patients may not be sufficient. Increasing the dataset size is essential, particularly to better represent minority scenarios such as patients requiring hospitalization. This expansion will enable the model to learn more effectively from these critical cases;

International Expansion Extending the study to include data from countries beyond Mexico could enhance the robustness of the model. Incorporating international datasets would provide a more diverse range of patient profiles and health dynamics, contributing to improved model accuracy;

Balance the Dataset There is a need to adjust the dataset to include more cases that involve hospitalization. A more balanced dataset will help mitigate the current model’s bias toward the majority class and improve its predictive accuracy for critical care scenarios;

Enrich the Feature Set Adding more variables could significantly enhance the model’s understanding of the patients. Features such as socioeconomic status, living conditions, and vaccination status are vital as they can influence the likelihood of hospitalization and the overall course of treatment. These additional features will provide a more comprehensive view of each patient, leading to better-informed predictions.

Implementing these strategies, we can enhance our model’s ability to accurately predict which patients require hospitalization, thereby improving patient outcomes and optimizing resource allocation.

8 Actions

Our model has pinpointed the primary factors contributing to severe COVID-19 cases. It appears that severe scenarios are more common among older patients, particularly those with comorbidities such as diabetes or pneumonia. Based on our findings, it is crucial to focus protective measures on the elderly, as they are the most vulnerable group. There are existing measures that can be enhanced to safeguard this population in the event of a pandemic:

Vaccination & Education Ensure vaccines are easily accessible to the vulnerable groups. Educate the population on the benefits of vaccination;

Rapid Testing Deploy mobile testing units to areas with outbreaks;

Contact Tracing Leverage digital tools and apps to improve the speed of contact tracing;

Distancing Using masks in crowded spaces, avoiding public spaces and limit the contact for the older population;

Hospital Infrastructure Augment resources in hospitals: ICU beds, ventilators and trained healthcare personal.

These actions when implemented effectively can improve the response to COVID-19, helping to reduce the transmission rates, support the healthcare systems, and ultimately save lives.

Triage for COVID-19 involves prioritizing medical care and resources based on the severity of patients’ symptoms and their potential risk for serious complications. Effective triage is crucial during a pandemic to ensure that healthcare systems can manage the influx of patients without becoming overwhelmed.

To ensure effective triage, we should continue to refine and expand our model, enabling it to provide rapid feedback across various scenarios and assist with resource allocation. Quick response and feedback are essential in every situation, particularly during a pandemic. Relying solely on human analysis to manage all cases is impractical and unsustainable. Therefore, developing classification models is crucial to aid in the decision-making process, helping determine which patients require immediate medical attention in urgent care settings.

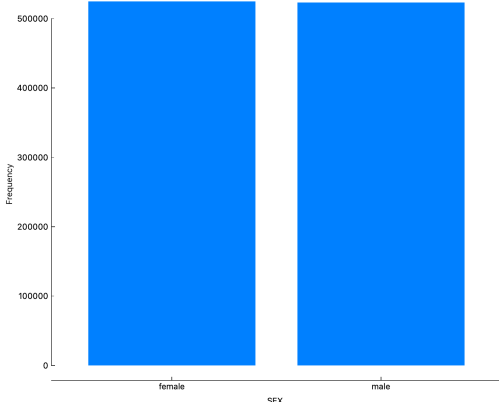
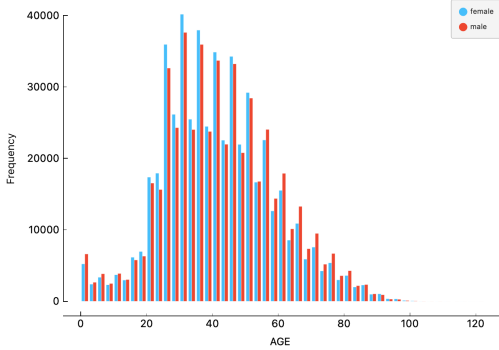
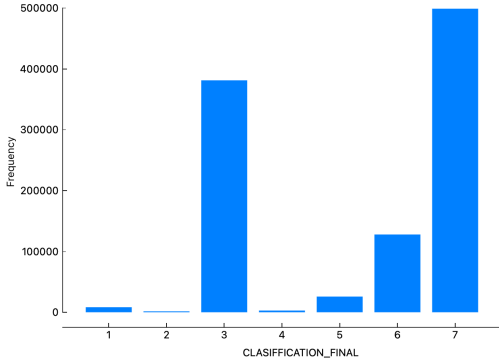
References

- Brownlee, Jason (2020). *How to Perform Feature Selection with Categorical Data*. URL: <https://machinelearningmastery.com/feature-selection-with-categorical-data/> (visited on 04/24/2024).
- ibm (2024a). *What is a neural network?* URL: <https://www.ibm.com/topics/neural-networks> (visited on 04/22/2024).
- (2024b). *What is random forest?* URL: <https://www.ibm.com/topics/random-forest> (visited on 04/22/2024).
- Ljubljana, University of (2024). *Orange Data Mining*. URL: <https://orangedatamining.com> (visited on 04/23/2024).
- Mining, Orange Data (2015). *Orange data mining*. URL: <https://orangedatamining.com/> (visited on 04/13/2024).
- Nizri, Meir (2023). *COVID-19 patient's symptoms, status, and medical history*. URL: <https://www.kaggle.com/datasets/meir nizri/covid19-dataset> (visited on 04/13/2024).
- scikit-learn (2024). *SkiKit Learn*. URL: <https://scikit-learn.org/> (visited on 04/23/2024).
- Thais Mayumi, Oshiro, Perez Pedro Santoro, and Baranauskas José Augusto (2012). “How Many Trees in a Random Forest?” In: *Lecture Notes in Computer Science* 7376.
- Yandex (2024). *CatBoost*. URL: <https://catboost.ai/> (visited on 04/22/2024).

A Appendix

A.1 Features Analysis in Depth

Table 9: Feature Analysis

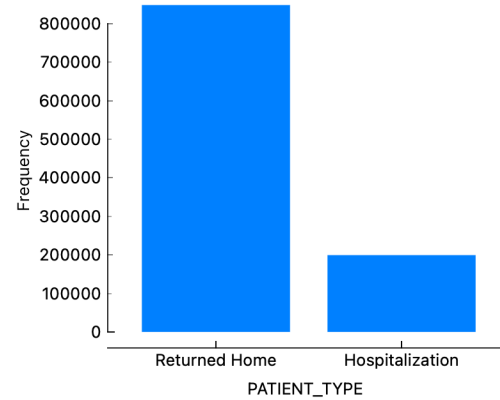
Feature Analysis	Frequency Distribution
<p>Sex</p> <p>Sex is the sex of the patient. It's a binary variable with the following meaning:</p> <p>1 Female;</p> <p>2 Male.</p> <p>The distribution is almost 50-50 (50.07% females and 49.93% male).</p>	
<p>Age</p> <p>Age is the age of the patient. It's a numeric variable and we can tell that is composed by mainly adults. The dataset is fairly distributed by age and sex.</p>	
<p>Classification</p> <p>Classification is the diagnosis based on the COVID-19 test result. Has a range of values 1-7. Where 1 means that the patient has 1 degree COVID-19 and 7 it has no COVID-19/inconclusive. So we have:</p> <p>1-3 The patient has COVID-19 (degrees 1 to 3);</p> <p>4-7 The patient is not a carrier of COVID-19 or the test is inconclusive.</p> <p>The most predominant values is the level 3 and 7 representing almost 70% of the data.</p>	

Patient Type

Type of care of the patient received in the unit:

- 1 Returned Home;
- 2 Hospitalization.

The most predominant are the patients that returned home.

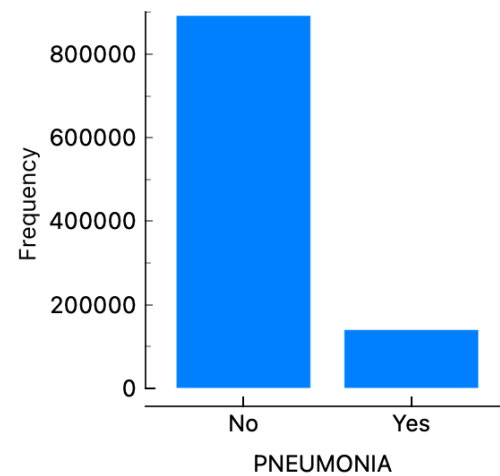


Pneumonia

Pneumonia is a binary variable whether the patient has the inflammation or not.

- 1 Has Pneumonia;
- 2 Does not have Pneumonia;
- 99 Missing value.

We have around 2% of missing data and the predominant value is No.

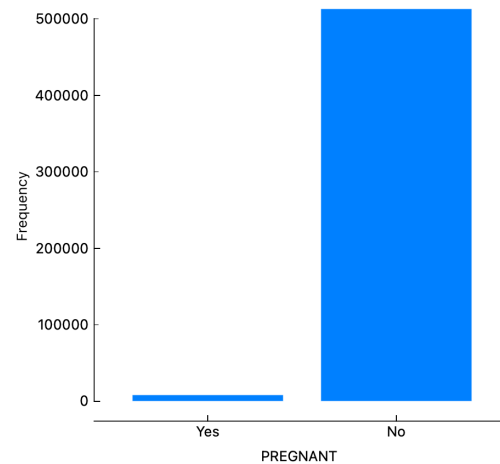


Pregnancy

Pregnancy is a binary variable whether the patient has the inflammation or not.

- 1 Pregnant;
- 2 Non Pregnant;
- 99 Missing value.

We have around 50% of missing data (related with the fact that 50% are men) and the predominant value is No. We only have around 1.5% pregnant.



Diabetes

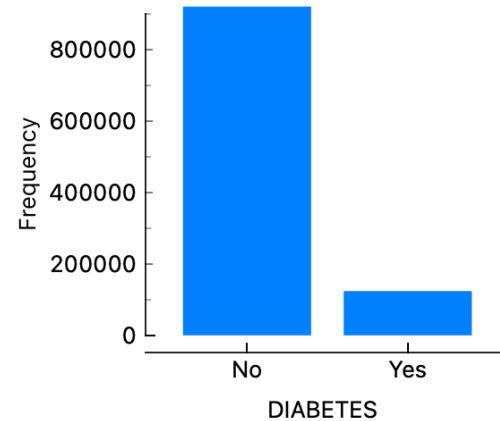
Whether the patient has diabetes or not.

1 Has Diabetes;

2 Does not have Diabetes;

98 Missing value.

We have only 3338 cases with missing data.
And we have 12% of population with diabetes.



copd

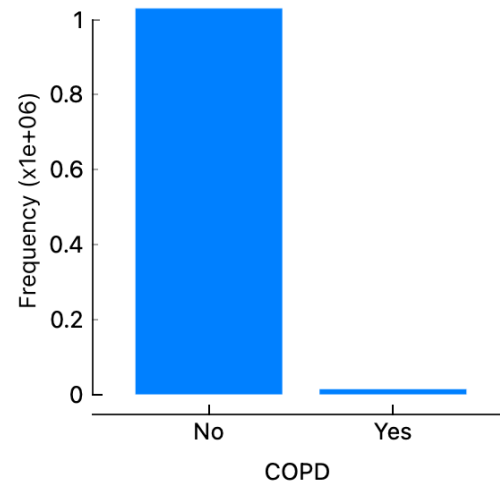
copd indicated whether the patient has *Chronic Obstructive Pulmonary Disease* or not

1 Yes;

2 No;

98 Missing value.

We have only 3003 cases with missing data.
And we have 1.44% of population with copd.



asthma

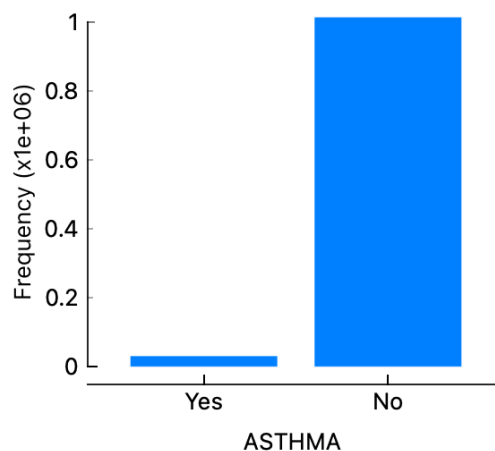
Whether the patient has asthma or not. Values:

1 Yes;

2 No;

98 Missing value.

We have 2979 cases with missing data. And we have 3% of population with asthma.



inmsupr

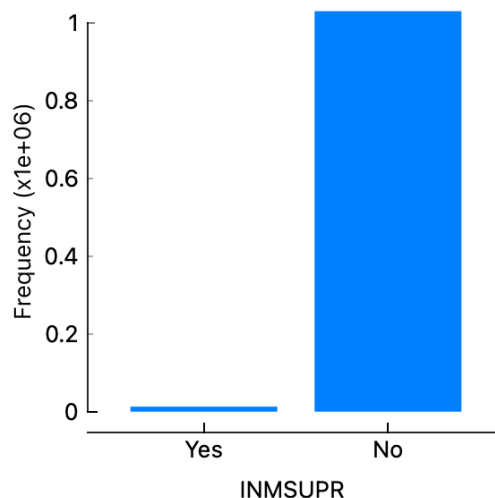
Whether the patient is immunosuppressed or not. Being immunosuppressed means that a person's immune system is less effective at fighting off diseases and infections. Values:

1 Yes;

2 No;

98 Missing value.

We have 3404 cases with missing data. And we have 1% of population immunosuppressed.



Hypertension

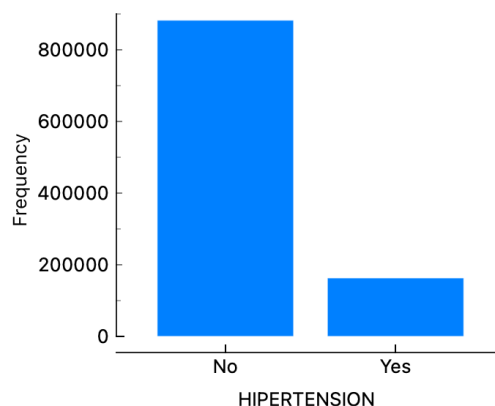
Whether the patient has hypertension or not. Values:

1 Yes;

2 No;

98 Missing value.

We have 3104 cases with missing data. And we have 16% of population with Hypertension.



Cardiovascular

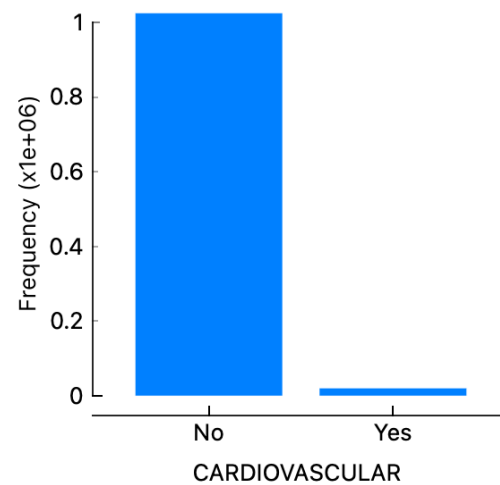
Whether the patient has heart or blood vessels related disease. Values:

1 Yes;

2 No;

98 Missing value.

We have 3076 cases with missing data. And we have 2% of population has Cardiovascular disease.



renal chronic

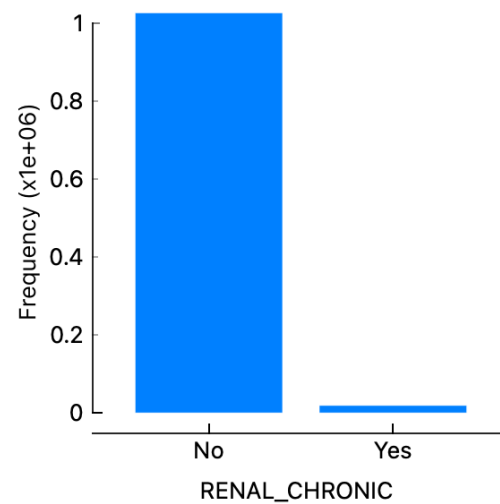
Whether the patient has chronic renal disease or not. Values:

1 Yes;

2 No;

98 Missing value.

We have 3006 cases with missing data. And we have 1.81% of population has renal disease.



other disease

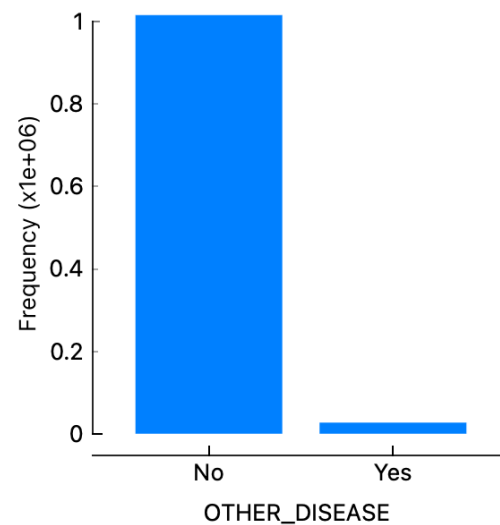
Whether the patient has other disease or not. Values:

1 Yes;

2 No;

98 Missing value.

We have 5045 cases with missing data. And we have 2.7% of population has other disease.



Obesity

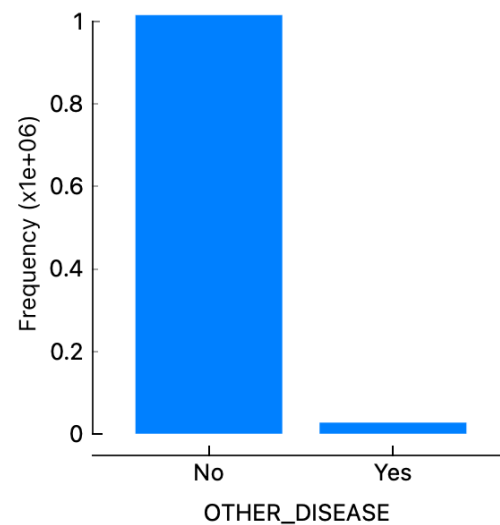
Whether the patient is obese or not. Values:

1 Yes;

2 No;

98 Missing value.

We have 3032 cases with missing data. And we have 15% of population is obese.



tobacco

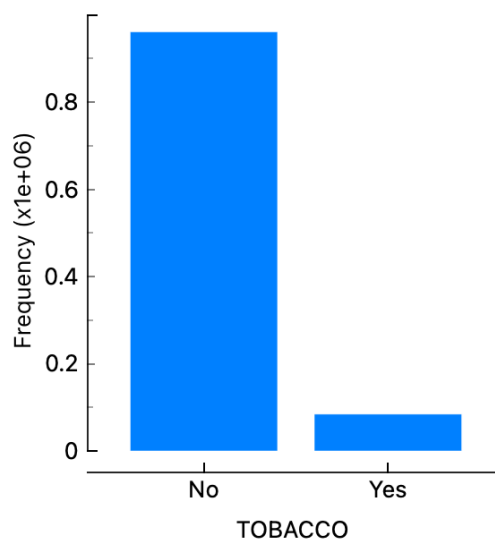
Whether the patient is a tobacco user. Values:

1 Yes;

2 No;

98 Missing value.

We have 3220 cases with missing data. And we have 8% of population is obese.



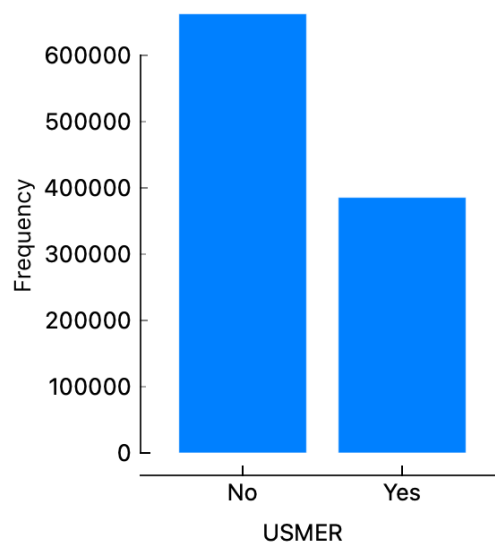
usmer

Indicates whether the patient treated medical units of the first, second or third level. Values:

1 Yes;

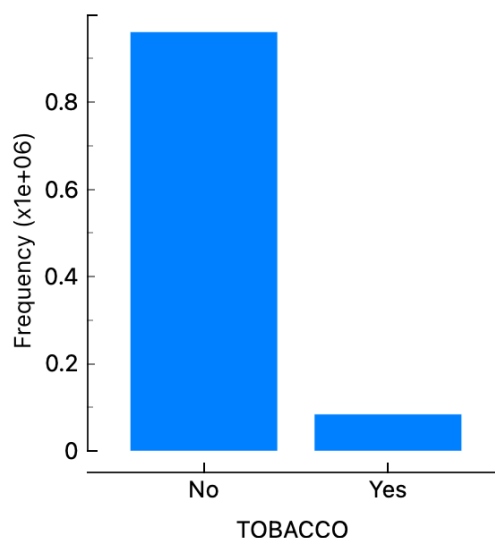
2 No.

We have no missing data. And 37% of population was treated in a medical unit.



medical unit

Type of institution of the National Health System that provided the care were the patients were treated. It is a categorical value with 13 Medical Unit Types (Numeric values ranging between 1 to 13). We have no missing data. The predominant data is 12 and 4.



intubed

Whether the patient was connected to the ventilator. Values:

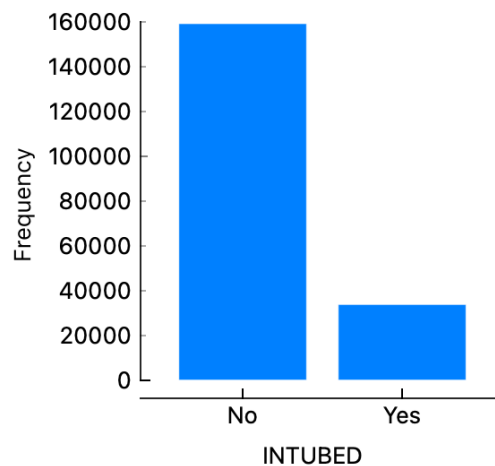
1 Yes;

2 No;

97 Missing value;

99 Missing value.

We have 82% cases with missing data (Probably patients that were not hospitalized or not followed by clinics). And we have 17% of population was intubed (Around 40000 in one million. The 17% is measured based on the populated data (18% of the population)).



icu

Indicates whether the patient had been admitted to an Intensive Care Unit. Values:

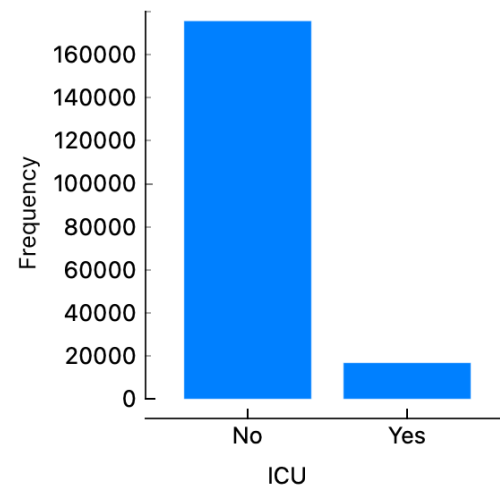
1 Yes;

2 No;

97 Missing value;

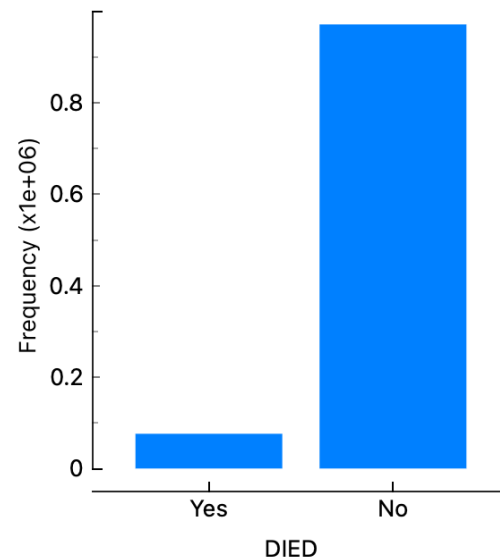
99 Missing value.

We have 82% cases with missing data (Probably patients that were not hospitalized or not followed by clinics). And we have 8% of population was in ICU (Around 17000 in one million. The 17% is measured based on the populated data (18% of the population)).
























date died

If the patient died indicate the date of death, and 9999-99-99 otherwise. And we have 8% death.



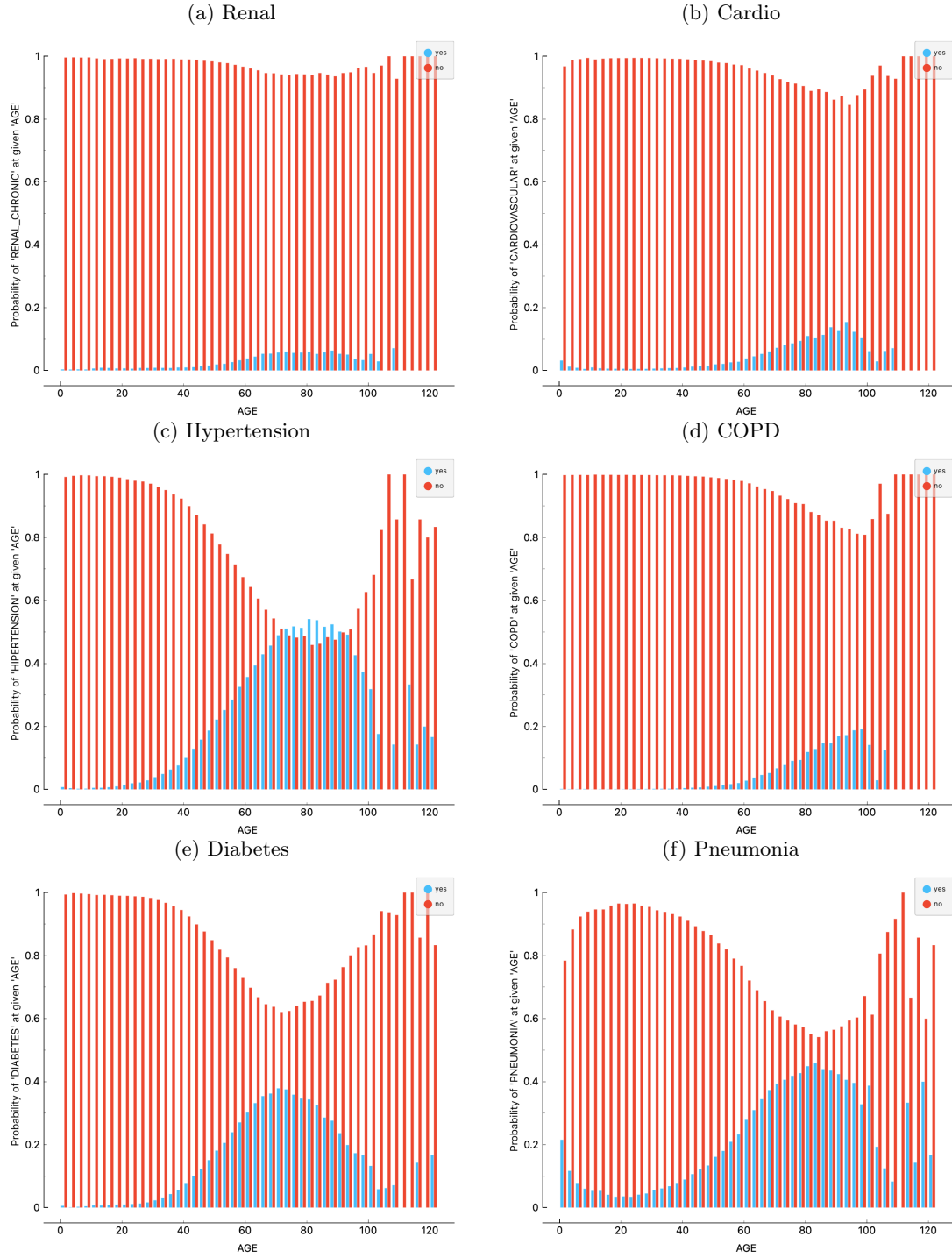
A.2 Feature Statistics

Figure 12: COVID-19 Feature Statistics

Feature Statistics									
Sun Apr 14 24, 18:49:23									
Name	Distribution	Mean	Mode	Median	Dispersion	Min.	Max.	Missing	
N AGE		41.79	30	40	0.40	0	121	0 (0 %)	
C ASTHMA			no		0.135			2979 (0 %)	
C CARDIOVASCULAR			no		0.0975			3076 (0 %)	
C CLASIFFICATION_FINAL			7		1.14			0 (0 %)	
C COPD			no		0.0754			3003 (0 %)	
C DIABETES			no		0.366			3338 (0 %)	
C DIED			no		0.262			0 (0 %)	
C HIPERTENSION			no		0.432			3104 (0 %)	
C ICU					0.297			856032 (82 %)	
C INMSUPR			no		0.0718			3404 (0 %)	
C INTUBED					0.463			855869 (82 %)	
C MEDICAL_UNIT			12		1.16			0 (0 %)	
C OBESITY			no		0.428			3032 (0 %)	
C OTHER_DISEASE			no		0.124			5045 (0 %)	
C PATIENT_TYPE			home		0.487			0 (0 %)	
C PNEUMONIA			no		0.397			16003 (2 %)	
C PREGNANT					0.0804			527265 (50 %)	
C RENAL_CHRONIC			no		0.0905			3006 (0 %)	
C SEX			female		0.693			0 (0 %)	
C TOBACCO			no		0.281			3220 (0 %)	
C USMER			non-treated		0.658			0 (0 %)	

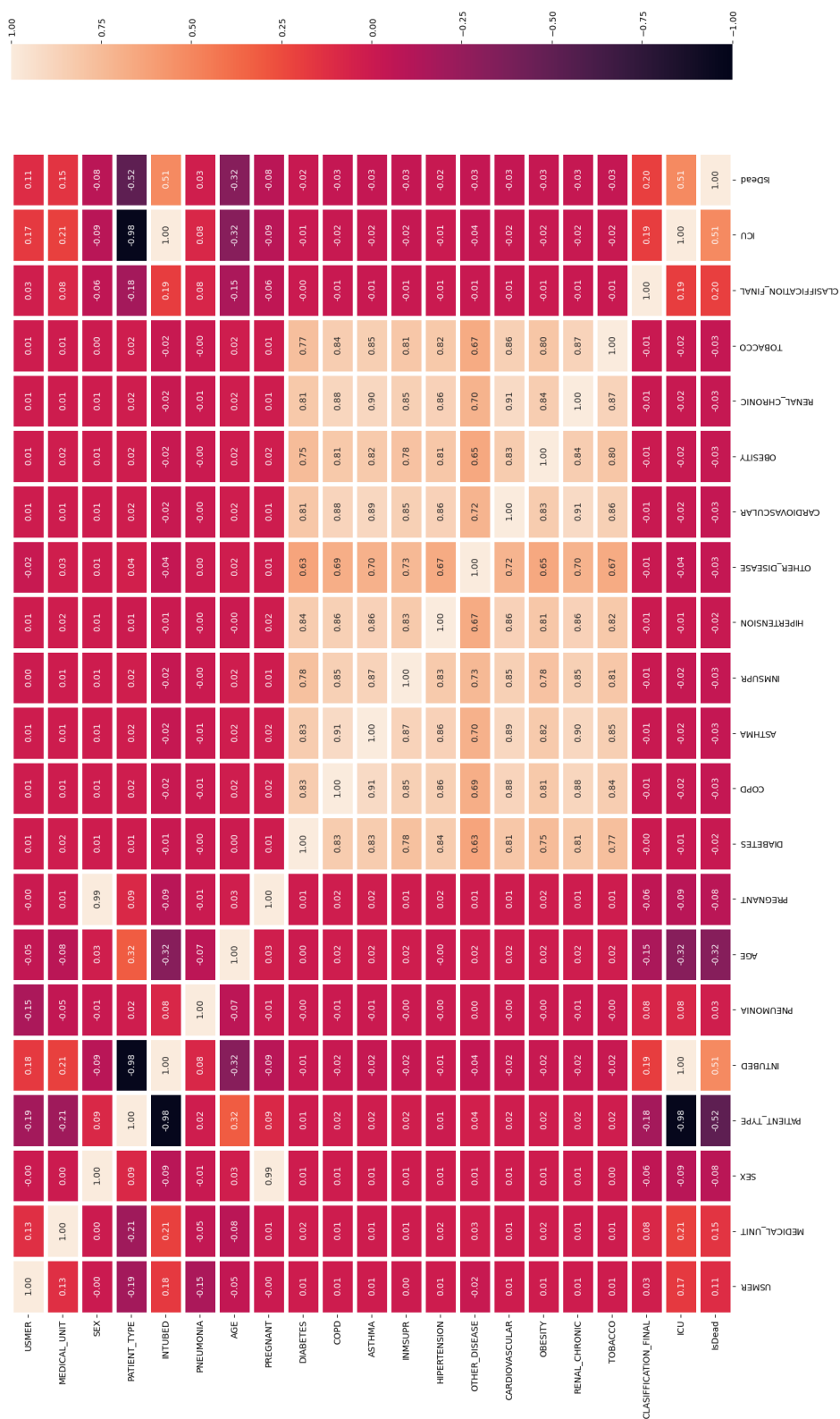
A.3 Feature Insights

Figure 13: Age by Disease (Probability Histograms)



A.4 Feature Correlation Matrix

Figure 14: Feature Correlation Matrix



A.5 Feature Ranking

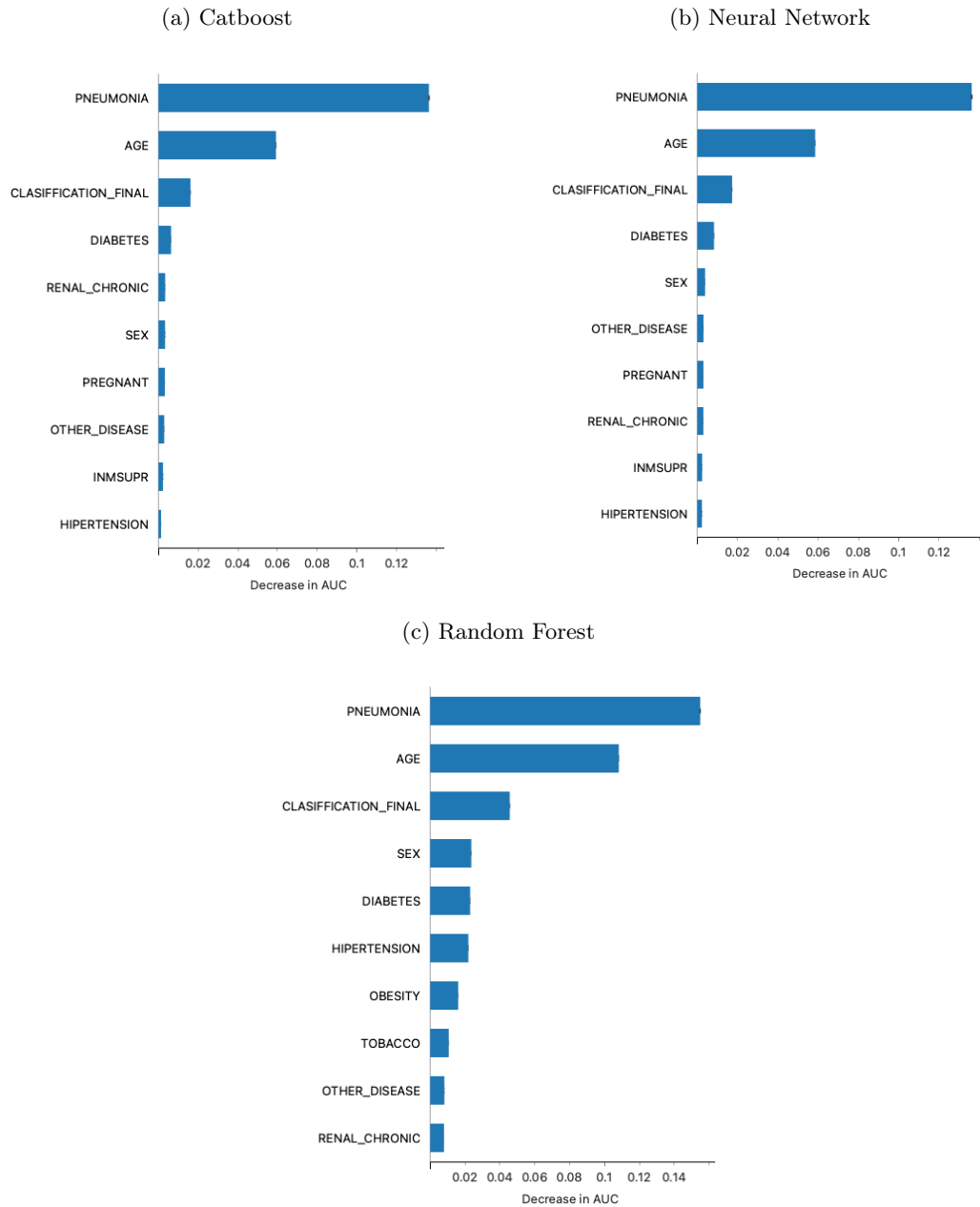
Figure 15: Feature Ranking

		#	Info. gain	Gain ratio	χ^2	▼
1	N AGE		0.095	0.047	94278.454	
2	C PNEUMONIA	2	0.258	0.451	62754.060	
3	C CLASIFFICATION_FINAL	7	0.039	0.024	40450.411	
4	C HIPERTENSION	2	0.038	0.061	10210.767	
5	C DIABETES	2	0.044	0.083	9352.530	
6	C SEX	2	0.007	0.007	5275.072	
7	C OBESITY	2	0.003	0.006	829.012	
8	C PREGNANT	2	0.001	0.013	572.537	
9	C RENAL_CHRONIC	2	0.013	0.099	461.117	
10	C OTHER_DISEASE	2	0.005	0.030	248.374	
11	C CARDIOVASCULAR	2	0.007	0.047	244.875	
12	C COPD	2	0.008	0.077	231.528	
13	C INMSUPR	2	0.005	0.049	126.825	
14	C ASTHMA	2	0.000	0.001	10.294	
15	C TOBACCO	2	0.000	0.000	3.544	

A.6 Feature Importance per Metric

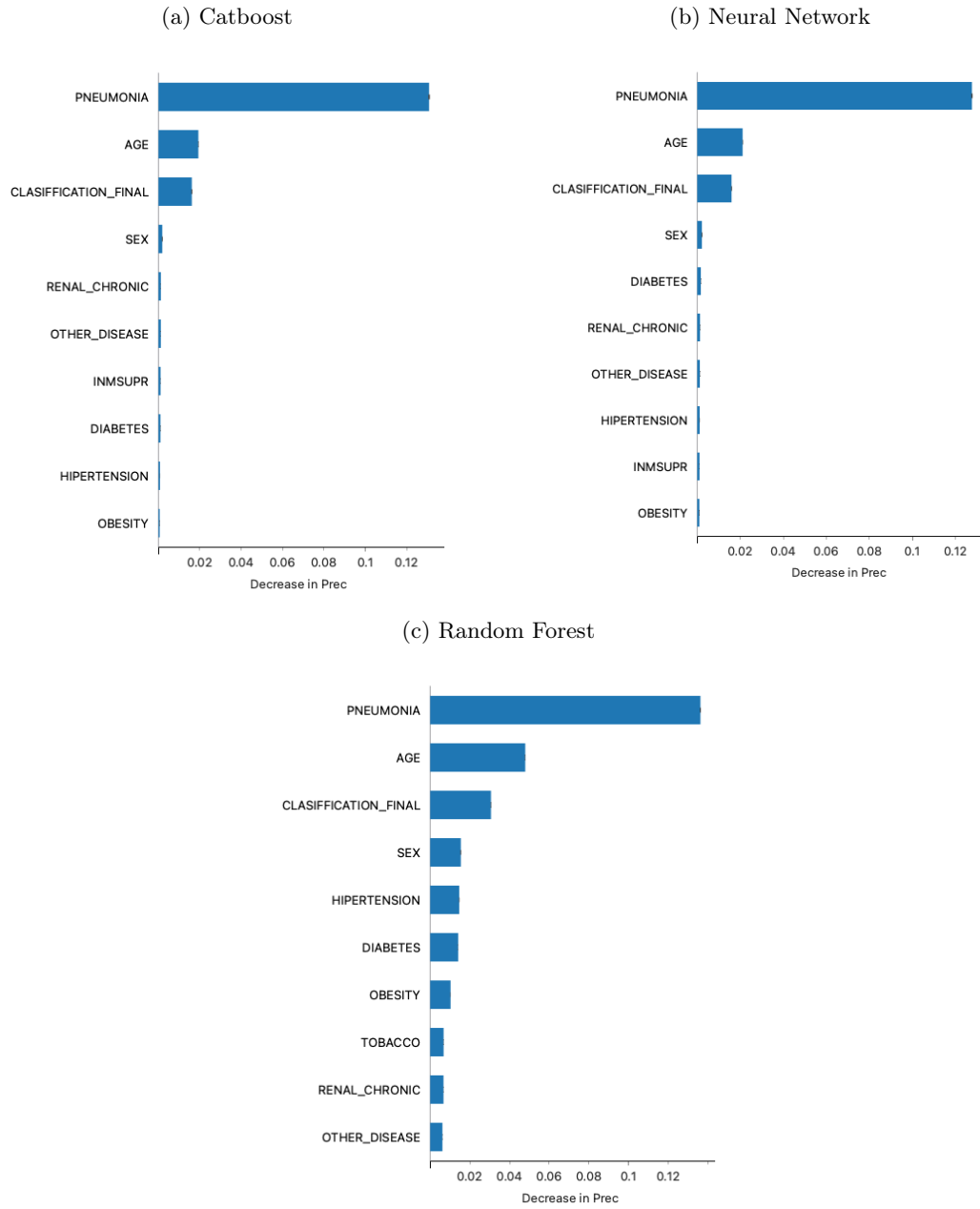
A.6.1 AUC

Figure 16: Feature Importance for AUC



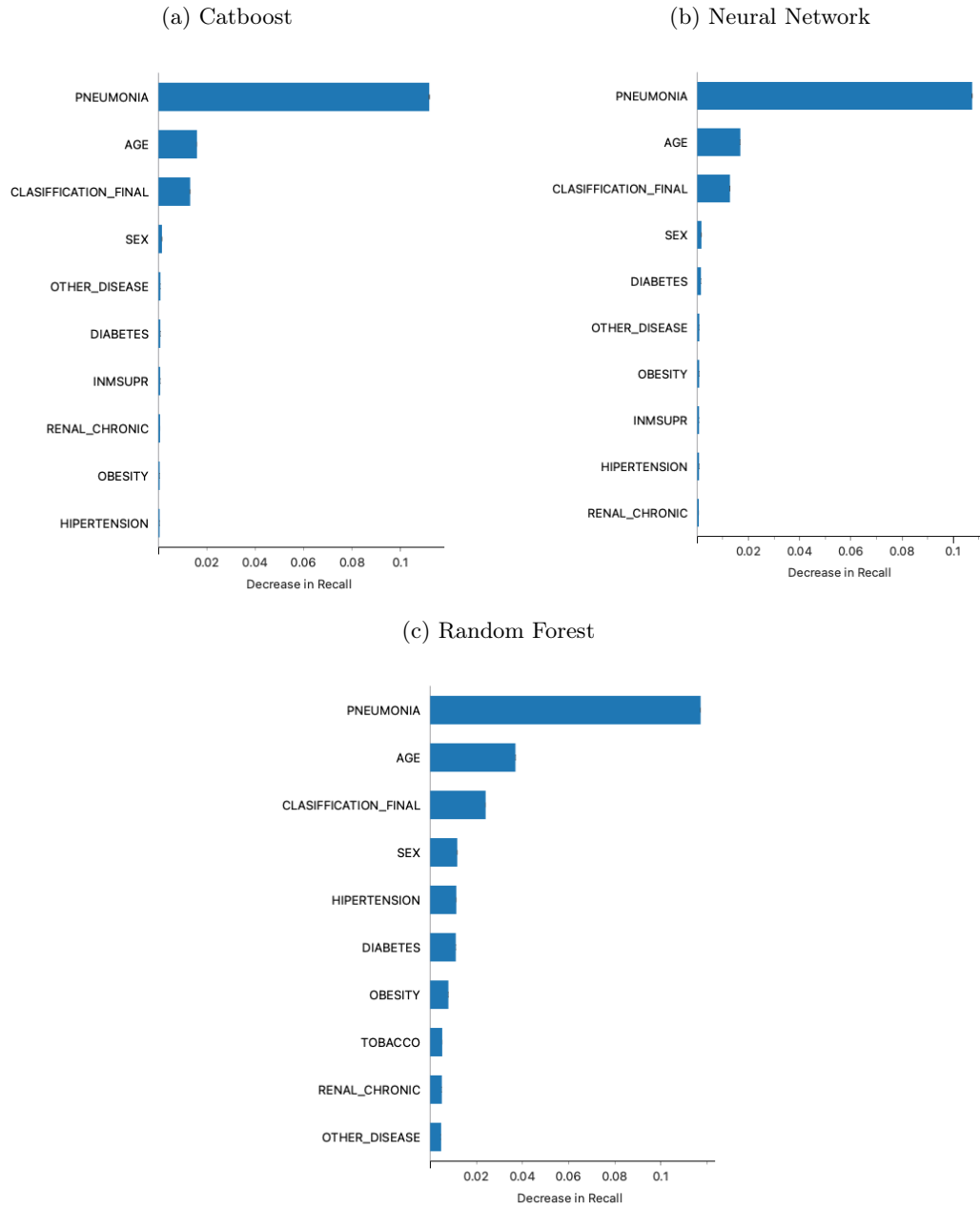
A.6.2 Precision

Figure 17: Feature Importance for Precision



A.6.3 Recall

Figure 18: Feature Importance for Recall



A.7 Feature Normalization Using Orange Data Mining

Figure 19: Data Preparation using Edit Domain Widget

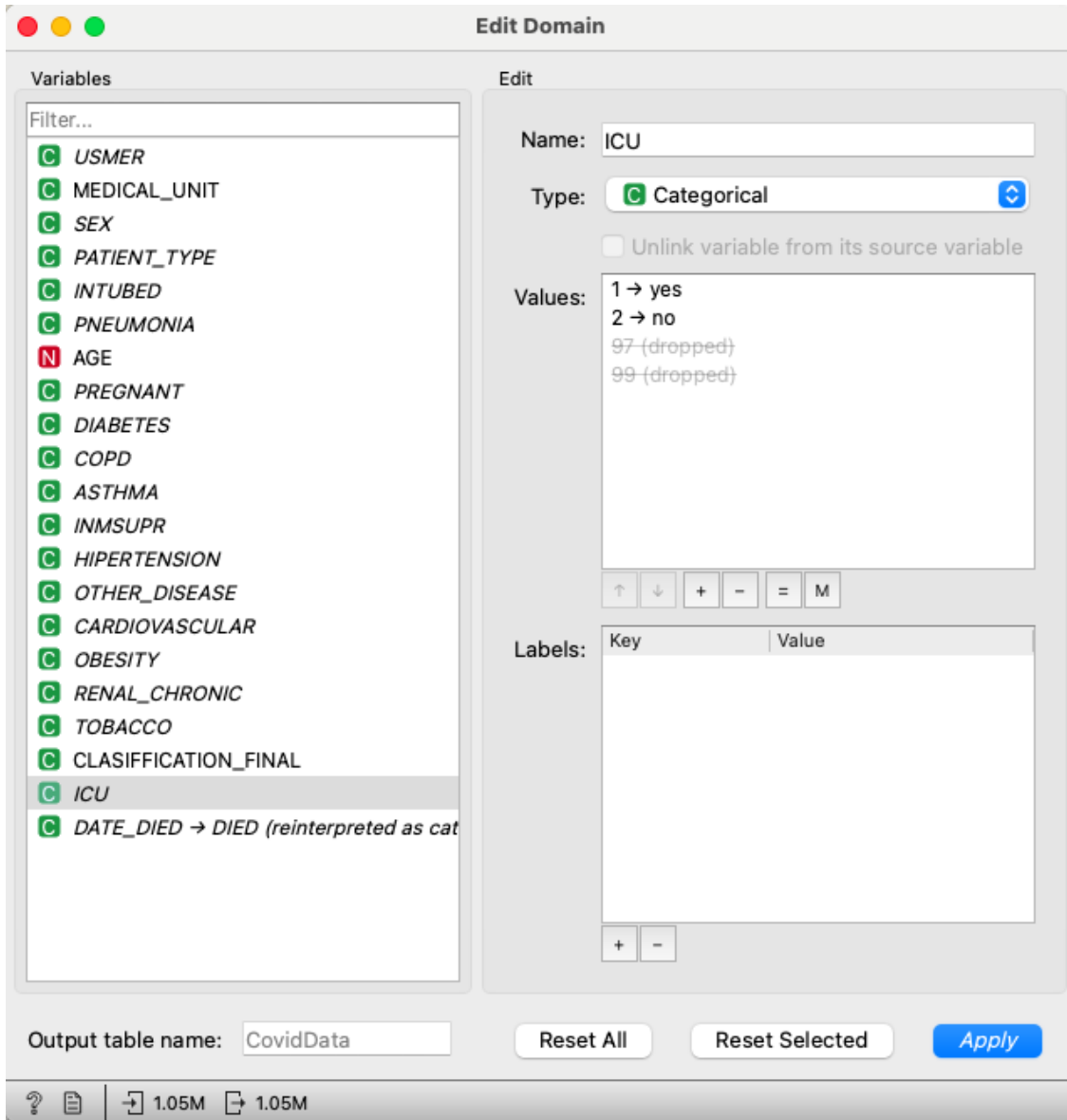
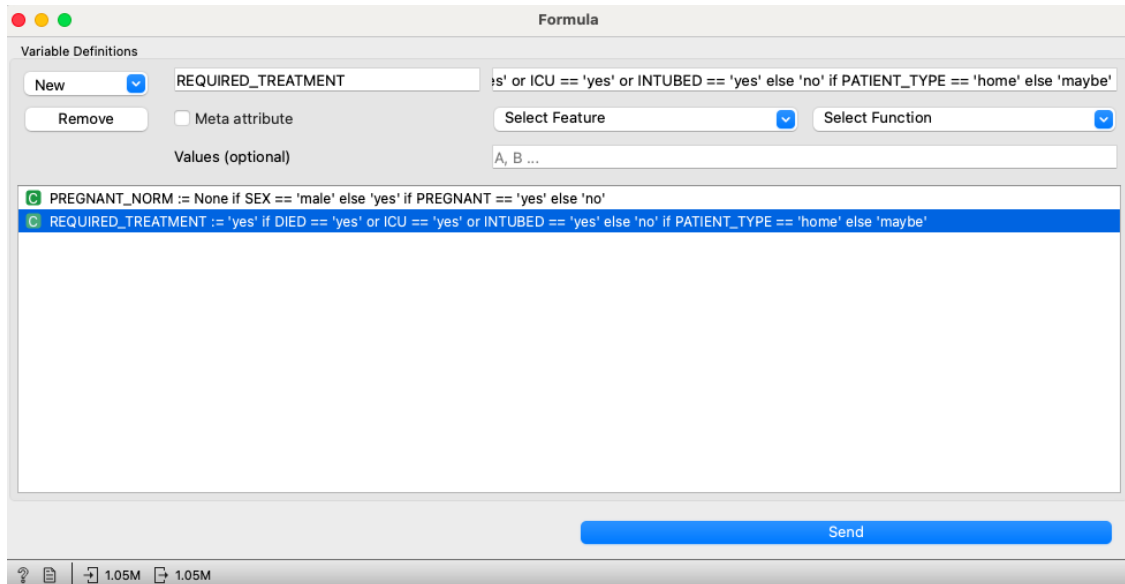


Figure 20: Data Preparation using Using Formula Widget



A.8 Code Listings

A.9 Feature Analysis Correlation Matrix

```

1 import pandas as pd
2 import matplotlib.pyplot as plt
3 import seaborn as sns
4 data = pd.read_csv("CovidData.csv")
5 data.describe().T
6 data['IsDead'] = data['DATE_DIED'].apply(lambda x: 1 if x == '9999-99-99' else 0)
7 data = data.drop(["DATE_DIED"], axis = 1)
8
9 plt.figure(figsize=(30, 15))
10 sns.heatmap(data.corr(), vmin=-1, vmax=1, cbar = True,
11             linewidths = 5,
12             annot = True,
13             fmt=".2f")
14 plt.show()

```

Listing 1: Correlation Matrix For Feature Analysis

A.9.1 Feature Normalization

```
1  def makeBool(value: String): String = value match {
2      case "1" => "yes"
3      case "2" => "no"
4      case _ => ""
5  }
6
7  def makeDate(value: String): String = value match {
8      case "9999-99-99" => ""
9      case _ => value
10 }
11
12 def makeSex(value: String): String = value match {
13     case "1" => "female"
14     case "2" => "male"
15 }
16
17 def makeDied(value: String): String = value match {
18     case "9999-99-99" => "no"
19     case _ => "yes"
20 }
21
22 def makeRequiredTreatment(dateDied: String, icu: String,
23                           intubed: String, patientType: String)
24   : String = {
25     if (dateDied != "9999-99-99") "yes"
26     else if (icu == "1") "yes"
27     else if (intubed == "1") "yes"
28     else if (patientType == "1") "no"
29     else "maybe"
30 }
31
32 def makePatientType(patientType: String): String = patientType match {
33     case "1" => "home"
34     case "2" => "hospital"
35 }
```

Listing 2: Feature Normalization For COVID-19 Dataset - Functions

```

1  Array(usmer,
2      medicalUnit,
3      makeSex(sex),
4      makePatientType(patientType),
5      makeDate(dateDied),
6      makeBool(intubed),
7      makeBool(pneumonia),
8      age,
9      if (makeSex(sex) == "male") "na" else makeBool(pregnant),
10     makeBool(diabetes),
11     makeBool(copd),
12     makeBool(asthma),
13     makeBool(inmsupr),
14     makeBool(hipertension),
15     makeBool(other_disease),
16     makeBool(cardiovascular),
17     makeBool(obesity),
18     makeBool(renal_chronic),
19     makeBool(tobacco),
20     clasiffication_final,
21     makeBool(icu),
22     makeDied(dateDied),
23     makeRequiredTreatment(dateDied, icu, intubed, patientType)).mkString(",")
24 }

```

Listing 3: Feature Normalization For COVID-19 Dataset - Process