MSc Data Science

Module: R for Data Science (DSM110)

Session: April 2022

Student Number: 200199830

End Term Coursework: Data Modelling
Predicting COVID-19 outcomes in a hospital setting

Word count (excluding title, figures, tables, references, and the code): 4961 words

Table of content

Abstract

Accurately predicting the disease outcome of COVID-19 can help in more precise allocation of healthcare resources to ultimately improve patient outcomes. This study investigates different statistical modelling approaches based on a real-world dataset from patients admitted to hospital because of COVID-19. Logistic regression, LASSO regression, k-nearest neighbour, gradient boosting machine, random forest, support vector machine, neural network, and a meta-predictor consisting of several of the other models are being compared.

Laboratory data is more expensive to collect than clinical baseline information. The models are trained and evaluated with only clinical information (e.g., age, sex) or only laboratory measurements (e.g., inflammatory markers) or both clinical and laboratory data to compare the informativeness of these variable classes for prediction. Classifiers that are trained with laboratory measurements outperform classifiers that only rely on clinical information.

After preparing the data and establishing a standardised modelling pipeline, all of the classifiers outperform the common sense baseline. Overall, the gradient boosting machine is the most successful classifier (f1-score 70%) showing the promise of statistical modelling for COVID-19 outcomes. However, even the best classifier misses a significant proportion of deceased patients. Possibilities to further improve the classifiers' predictions are being discussed.

1. Introduction

1.1 Overview

The COVID-19 pandemic has disrupted normal life and resulted in significant mortality and disease burden worldwide with more than approx. 600 million cases and 6.4 million deaths as of the 10th september 2022 ("WHO Coronavirus (COVID-19) Dashboard" n.d.). The disease has been especially fatal for certain patients at the beginning of the pandemic when no vaccines or efficient antiviral medications were available. COVID-19 causes a wide clinical spectrum and it is not trivial to predict the disease outcome especially in the early stages of the disease (Hu et al. 2021).

This study investigates the merits of different statistical models for prediction of death from COVID-19 in patients immediately after admission to a hospital with the diagnosis of COVID-19. Such models can help in better allocating the limited resources of the health system to focus on the patients that have the highest probability of succumbing to the disease.

First, this work compares different models and their potential for outcome prediction with the goal to find the best model for this unbalanced classification task as only a minority of patients will die because of COVID-19, making it an unbalanced classification task. A successful classifier must identify patients that are most likely to die (true positives) while also not predicting a fatal outcome for too many patients that will eventually survive (false positives) as a high false positivity rate will make a model futile in clinical practice.

The data consists of clinical information and laboratory measurements. While clinical information is available in patient records or can be obtained by interviewing a patient, laboratory measurements require expensive machinery and an infrastructure to conduct blood taking and analysis. The second objective of this project is to analyse benefits of combining clinical data and laboratory data and if clinical data alone achieves similar prediction results compared to utilising more expensive laboratory data. Thus, the models are trained with different subsets of the data, i.e. with only clinical, only laboratory and both clinical and laboratory data.

1.2 Research in context

Using tabular data in medicine for a classification task has been traditionally modelled with logistic regression, a linear model (Shipe et al. 2019). More modern and complex non-linear models such as decision tree ensembles with bagging, random forest (Breiman 2001), or boosting, gradient boosting machine (J. Friedman, Hastie, and Tibshirani 2000) have been increasingly deployed in the medical domain.

While several modelling approaches have been undertaken for different aspects of the COVID-19 pandemic, the investigated dataset for this study has not been used previously for modelling. Thus, no benchmarking or particular modelling results for this dataset have been published before.

There are other attempts in predicting COVID-19 outcomes based on clinical or laboratory information. For example, Assaf et al. reported improved prediction by utilising machine learning when compared to a more basic clinical score such as the APACHE score (Assaf et al. 2020). The APACHE score (Knaus et al. 1985) uses a wide array of clinical and laboratory measurements to predict death in severe disease. APACHE is not tailored to any specific disease and uses a rigid additive model. Unfortunately, the APACHE score uses information that is not readily available when patients are admitted to a hospital but instead only become available when patients are transferred to an intensive care unit (e.g., blood oxygenation, arterial pH, rectal temperature). As this study investigates data immediately after hospital admission and is not limited to patients that require intensive care treatment, the results cannot be compared to the APACHE score and an APACHE score cannot be calculated for a large proportion of the patients within this dataset.

Attempts to predict COVID-19 outcomes with laboratory data have been published before (Domínguez-Olmedo et al. 2021), yet these datasets mostly use limited clinical variables and lack information about preconditions and specific medication.

1.3 Statistical modelling approach

This work will compare linear and non-linear statistical models with different selections of features (clinical or laboratory data, all data). Linear models include logistic regression and least absolute shrinkage and selection operator (LASSO) logistic regression. Logistic regression is a widely used popular model for classification while LASSO modifies the logistic regression formula by applying a penalty for the size and amount of coefficients in the model. This shrinks coefficients up to zero ultimately resulting in a sparser model. This can be beneficial in medicine where sparse models allow for derivation of risk scores with only a few input variables (Khanji et al. 2019), (Heinze, Wallisch, and Dunkler 2018).

Although support vector machines use hyperplanes to separate the data and are thus considered linear classifiers, they can also classify nonlinear datasets. This is achieved by utilising kernels (e.g., radial kernel) that maps the input data into a different space where the nonlinear data becomes linearly separable (James et al. 2021).

In addition, k-nearest neighbour, gradient boosting machine, random forest and neural networks are used as a nonlinear modelling approach for this dataset as non-linear models might achieve higher prediction performance by being able to capture complex interaction effects.

Decision tree ensembles such as gradient boosting machine and random forest (Breiman 2001) are especially popular for medical tabular data (Shwartz-Ziv and Armon 2022).
K-nearest neighbour calculates distances by using e.g., Euclidean distance between data points and uses the nearest k data points from the training data to predict the label of a new testing data point. The number of training data points that are being considered for classification is called “k” and is a hyperparameter. The distance function is sensitive to the scale of the data thus, it requires the data to be standardised (James et al. 2021).
In random forests, a decision tree ensemble of trees is constructed. The decision trees are built with random subsets of the predictor variables and with random subsets of the training data with replacement (“bagging”). A similar technique is gradient boosting machines, where decision trees, weak learners, are sequentially added to a model while focusing on the misclassified data points of the previous decision trees by using functional gradient descent (J. H. Friedman 2001). Gradient boosting machines belong to the most successful modelling techniques for tabular data (Howard and Gugger 2020).
Neural networks have become increasingly popular as a modelling technique because of their flexible nature that can model highly complicated and interdependent feature interactions and theoretically approximate any function (Hornik 1991). When choosing a non-linear activation function, they become highly non-linear classifiers that have been successfully applied to unstructured data (e.g., images, text) and structured (tabular) datasets (Chollet 2017).
Lastly, a meta-predictor is built that consists of an ensemble of different linear and nonlinear models, i.e. logistic regression, support vector machine with radial kernel, random forest, gradient boosting machine, k-nearest neighbour and averages the weighted vote of each classifier.
This study does not only compare different models but also the input data used to train these models. Specifically, subsets of the data (clinical, laboratory and all data) are being used to investigate the merits of combining clinical information (patient characteristics such as age, sex) with laboratory measurements (e.g., blood count, inflammatory markers) or only using laboratory or clinical predictors.

2. Methods

2.1 Overview of selected dataset and preprocessing

The selected dataset consists of pseudonymized data from patients that were treated in a university hospital in Germany during the COVID-19 pandemic before vaccines became widely available (1st Match 2020 to 29th January 2021). The clinical investigators gave permission for the use of this data for a student coursework project with the University of London. The data was provided in three files, consisting of patient outcomes, patient baseline characteristics and laboratory measurements that all had to be cleaned and merged. Duplicated rows were dropped and laboratory data was filtered by patient IDs and restricted to the first set of measurements that were conducted when the patient was admitted to the hospital. Unintelligible column names in German were translated and cut-off points of laboratory measurements (containing symbols such as “<”, “>”) were replaced with numeric cutoff values according to consultation with clinical experts. Columns that exceeded a defined threshold (10%) of missing values were dropped. For other missing values imputation with median and multiple imputation by chained equations (van Buuren and Groothuis-Oudshoorn 2011) were performed.

2.2 Feature engineering, selection of predictors

An important pitfall in predictive modelling is the use of information of the future to predict outcomes of the past, i.e. using data during model training that in fact only becomes available after the time point when the classifier is supposed to be used. Instead, the flow of time must be respected if one aims to build classifiers for real-world applications that have a more realistic assessment of model performance (Hernan and Robins 2020). In this dataset, several columns contain information that only becomes available during the course of hospitalisation and not at the time point of hospital admission when the prediction should take place. This includes information on mechanical ventilation (intubation) and organ replacement therapy with extracorporeal membrane oxygenation (ECMO) or dialysis to replace the function of the lung or kidneys, respectively. Consequently, these variables are removed except for the column “intubated_referral” which contains information about which patient was admitted being intubated as this information is available at admission .

The date of hospital admission with the cut-off of the 16th June 2020 is used to derive a new boolean variable (“dexa_standard_period”) that codes for a change of standard of care of COVID-19: After this cut-off date severe cases of COVID-19 routinely received an anti-inflammatory drug (dexamethasone) in their second week of illness which significantly improved patient outcomes (RECOVERY Collaborative Group et al. 2021).

Non-informative columns (e.g., patient ID) and columns that were previously used for derived features (e.g., height, weight for body mass index) are removed.

A derived feature that sums up all preconditions of patients is built.

Variables that have (almost) no variation (near zero variance) might not be helpful for building a classifier and can even become problematic for e.g., logistic regression (Kuhn 2008). Testing for near zero variance variables is done with a cutoff of 90 to 10 between the most common and second most common value and consequently 9 columns are identified. All of these columns are removed after a discussion with a clinical expert except for “DNR/DNI” and “COPD” which are informative albeit only a subset of patients vary within these categories. “DNR/DNI” ("do not resuscitate/ do not intubate") codes for patients that do not want maximum possible care and “COPD” (chronic obstructive pulmonary disease) is a known risk factor for severe outcomes of COVID-19.

Additionally, the clinical and laboratory predictors are defined (see **Supplementary table S1**) that will consequently be used for model building.

2.3 Training and testing data, dummy encoding and data normalisation

The data is randomly split into two subsets that are stratified by the outcome variable (patients that succumbed to the disease) thus retaining the percentage of different outcomes of this dataset within the subsets. The training data consists of 70% of the original data and the remaining 30% are used for testing.

The mean and variance of each predictor in the training data is calculated and used to normalise the data to zero mean and unit variance. This is required for convergence of e.g., k-nearest neighbour and neural networks. Importantly, to avoid information leakage the mean and variance is only calculated from the training set and subsequently applied to both training and testing subsets (Kuhn and Johnson 2018).

Categorical variables are dummy encoded which means that for n levels within a categorical column n-1 columns are generated and the original column is dropped. To avoid perfect collinearity, only n-1 columns are generated. Otherwise it would be possible to use n-1 columns to predict the n-th column which leads to a singular design matrix that cannot be inverted and e.g., regression modelling with least squares estimation fails (Suits 1957). Columns are renamed to avoid illegal characters (“make.names” function in R).

Finally, the target column “deceased” is encoded as a factor as caret’s model training pipeline expects the outcome in classification tasks to be a factor (Kuhn 2008).

2.4 Evaluate model performance

This dataset is unbalanced with only 14% of the patients having died (deceased). Using accuracy for evaluation is deceptive as by always predicting the majority class, one can already achieve 86% accuracy. After taking several evaluation metrics into consideration (Ferri, Hernández-Orallo, and Modroiu 2009), the F1-score will be used as the primary metric for assessment. The F1 score is the harmonic mean of the precision and recall of a classifier and it will be evaluated together with precision and recall. The precision and recall (see formulas below) are both important as neither a classifier for prediction of COVID-19 outcomes should neither miss many patients at high risk of dying nor should it overestimate the risk of dying for patients that will survive as healthcare resources are limited and the classifier should improve allocation of them.
In addition, area under the receiver operating curve (ROC) will be considered but only as an adjunct as ROC has limitations in imbalanced datasets (Saito and Rehmsmeier 2015).

$$Precision \ = \frac{True\ positive}{True\ positive\ + False\ positive}$$

$$Recall \ = \frac{True\ positive}{True\ positive\ + False\ negative}$$

$$F1\ score \ = \frac{2 * recall * precision}{recall + precision}$$

2.5 Benchmarking and common sense baseline

For this particular dataset no benchmarks are available as this dataset has not been previously used in this configuration for predictive tasks.
As a common sense baseline a custom built dummy classifier is used that outputs predictions while reflecting the distribution of the outcome in the training data without using any further information. This “stratified” strategy reflects the prior probabilities of each class inspired by a scikit-learn implementation of a dummy classifier (Pedregosa et al. 2011). The ratio of negative outcomes in the target column “deceased” is used as the probability for a binomial distribution to predict patient outcomes without taking into account any further information of a specific row (“dummy” approach). Surely, any advanced statistical model must beat this common sense baseline to be of real use.

Code snippet 1: excerpt of dummy classifier

```
set.seed(123)  #to ensure reproducible results
prediction_dummy <- rbinom(number_predictions, #number of observations
                    size=1, #how many trials are conducted
                    prob= proportion_target_class #probability of
success per trial
  )
```

Secondly, logistic regression is used as a baseline classifier. This widely-utilised linear classifier has interpretable regression coefficients and will be compared to more advanced and algorithmically intricate models. It will be evaluated by which margin more complex models can outperform a simpler logistic regression approach.

2.6 Model building

This work utilises the Classification And REgression Training (caret) package framework (Kuhn 2008) and extends the caret training pipeline with custom functions that ensure a streamlined and equal sequence of steps for all models. Furthermore, an implementation of LASSO regression (Hastie T 2022) and a neural network (Fritsch S 2019) outside of the caret framework are used.
Each model is trained with 5-fold cross-validation for hyperparameter optimization (James et al. 2021) and finally evaluated on a hold-out test set. Other resampling methods such as "Jackknife" ("leave one out") and bootstrap resampling (resampling with replacement) were considered. K-fold cross-validation was chosen as it is considered advantageous by some authors (Witten, Frank, and Hall 2011).
By using hyperparameter optimization it is ensured that the competing models all have been tuned to this particular classification task. Each model is trained only on clinical data (1), only laboratory data (2), both clinical and laboratory data (3) and both clinical and laboratory data with additional oversampling of the minority class in the target column (4).
The oversampling of the target column will probably aid in tipping the classifier towards a more balanced prediction without excessive prediction of the majority class (Kuhn and Johnson 2018).

Code snippet 2: parameters for caret's train control model training

```
#define parameters that are being used during training
train_control_cv <- trainControl(
  method = 'cv', number = 5,       #as resampling method, perform k-fold
cross validation with 5 folds
  savePredictions = 'final',       #predictions are saved
  classProbs = T,                  #class probabilities will be returned
  summaryFunction=twoClassSummary #defines summary
)
```

```
#define modified training parameters where minority class oversampling
is used
train_control_cv_upsampling <- trainControl(
  method = 'cv', number = 5, savePredictions = 'final', classProbs = T,
summaryFunction=twoClassSummary,
  #oversampling - minority class is sampled multiple times but
train/validation split of cross-validation is respected, so no
information leak during training occurs
  sampling = "up")
```

The final evaluation on previously unseen test data ensures that no information leakage from the test to the training phase during hyperparameter optimization can occur while the oversampling of the minority class strictly occurs after splitting of training and test set and also only after selecting a hold-out validation set in k-fold cross-validation to avoid biassed model performance (Kapoor and Narayanan 2022).

LASSO logistic regression

The employed LASSO model uses coordinate descent (J. Friedman, Hastie, and Tibshirani 2010). It has a hyperparameter, lambda, that determines the scale of applied penalty for the coefficients. The optimal value is chosen after cross-validation by choosing the value that minimizes mean cross-validated error. For a binary classification task the objective function is defined as a penalized negative binomial log-likelihood function (Hastie T 2022).

Code snippet 3: details of LASSO logistic regression

```
#cross validation is used to optimise the lambda hyperparameter (amount
of coefficient shrinkage)
cv_lasso_model <- cv.glmnet(x = X_train, y = y_train,
                            alpha = 1, #alpha of 1 equals lasso regression
                            family = "binomial") #for binary response

#Build model with lambda value that minimises mean cross-validated error
lasso_model <- glmnet(X_train, y_train, family = "binomial", alpha = 1,
                     lambda = cv_lasso_model$lambda.1se) #result from CV
```

Random forest

After training and tuning of a random forest model to optimise the number of random features that are considered at each split of the decision tree, the random forest is tested against the test set. Variable importance is calculated, scaled and displayed with the “varImp” function.

Neural network

No hyperparameter tuning is used for the neural network, instead the number of hidden neurons per layer is determined by empirical guidance using approx. ⅔ of the input variables as the number of neurons for the first hidden layer (Heaton 2008). Logistic activation function of the last layer is used as required for a binary classification task and linear output is set to False to bound output of the neural network to probabilities. Input data has been normalised before with zero mean and unit variance. This improves stability of the learning process and accelerates convergence (Chollet 2017).

Code snippet: 4: neural network architecture

```
nn_model <- neuralnet::neuralnet(formula = nn_formula,
                      data=df_train_dummy,
                      hidden=c(32,12), #neurons in hidden layers
                      act.fct = "logistic", #activation function for
final layer (binary)
                      linear.output = FALSE) #classification task,
should output probabilities (bound between 0 and 1
```

Meta-predictor (ensemble model)

Lastly, a meta-predictor that consists of different previously utilised classifiers is built using the package caretEnsemble. After training all of the individual models that make up the ensemble, a generalised linear model is trained on top of the ensemble that can average and weight the different predictions of each individual model. As the model building is computationally more expensive than the other models, only the full dataset and no subset of predictor variables is assessed.

Code snippet 5: meta-predictor

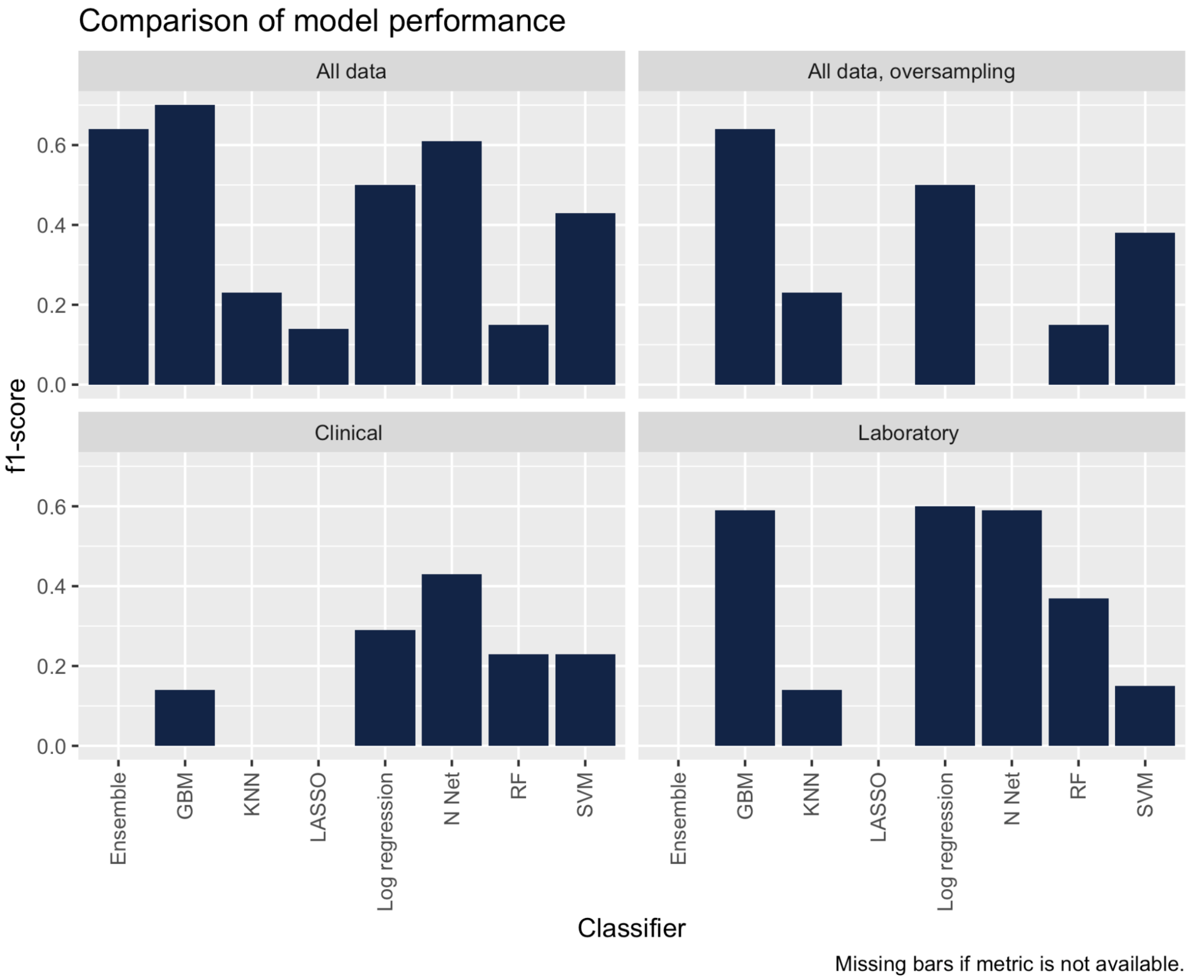
```
#train all models that are subsequently used as an ensemble
ensemble_models <- caretList(deceased ~ ., data=df_train,
                             methodList=ensemble_vector,
                             metric="ROC",    #metric for optimization
                             trControl=train_control_cv) #use CV

#training ensemble model
ensemble_glm <- caretStack(all.models=ensemble_models, method="glm",
metric="ROC", trControl= train_control_cv) #see above
```

3. Results

3.1 Comparison of performance

Figure 1: Comparison of the f1-score of different models
Abbreviations in figure legend: GBM - gradient boosting machine, KNN - k nearest neighbour, Log regression - logistic regression, N Net - Neural network, RF - random forest, SVM - support vector machine



The dummy classifier achieves recall and precision and f1-score of approx. 12%. It was trained on all data but the results are identical when using only parts of the data as the dummy classifier ignores all information in the predictor variables and instead predicts according to a “stratified strategy” that only relies on the prior probability of each class in the training data. The more complex modelling approaches should yield superior performance as they utilise information contained in the predictor variables.

A logistic regression classifier is trained that outperforms the dummy classifier by a large margin when using only laboratory data (F1-score: 60%, AUC: 89%). Using laboratory data is superior to only clinical data (F1-score: 29%). Oversampling of the minority class does not improve performance.

A penalised logistic regression approach with LASSO shows poor results when reflecting f1-score, in fact, when using laboratory or clinical data alone, the model does not predict a

single positive case, therefore precision and recall are both zero. Coefficient shrinkage has led to always predicting the majority class. “Deceased” as outcome prediction is only observed when training the model with all data simultaneously albeit this leads to a very low recall of 8%. Because of the implementation without the caret pipeline no oversampling of the minority class is performed.
The next predictor is k-nearest neighbour which generally has lower recall values than other classification approaches. Again, restricting the data to clinical variables leads to the worst performance. The best f1-score (23%) is achieved when using all of the data.
Radial kernel support vector machines achieve very high AUC (up to 93%) but the f1-score remains lower with a best performance (f1-score: 43%) when using all of the data. Contrary to the previous predictors it performs the worst when using only laboratory data (f1-score: 15%).

Gradient boosting is the best performing classifier on this task with an f1-score of 70% (precision: 80%, recall: 62%, AUC: 93%) when using all of the data. Worse performance is observed when using only clinical data or using oversampling of minority class. Gradient boosting is known to perform well on tabular data (Shwartz-Ziv and Armon 2022).
Although also an ensemble method, random forest does not perform as well as gradient boosting when only considering f1-score. With a perfect precision of 100% but only limited recall, f1-scores are between 15% and 37%. Interestingly, the area under the receiver operating curve is 94% when training on all data without oversampling. This indicates that using a different threshold instead of 0.5 for label prediction might lead to better f1-scores.

A neural network belongs to the best performing model with a maximum f1-score of 61% when using all data. Performance is worse when using only laboratory or clinical data. The AUC is consistently high ranging between 90-92%. As this model is implemented outside of the regular caret pipeline no oversampling is performed.
Lastly, an ensemble model (meta-predictor) consisting of diverse previously utilised predictors (logistic regression, k-nearest neighbour, support vector machine, gradient boosting, random forest) is constructed and in addition a linear classifier that weighs and averages the individual votes of the stacked model is trained. This complex approach is only trained on all data to avoid excessive computation time and achieves 64% f1-score (precision: 67% and recall: 62%). As only labels are predicted, AUC is not calculated.

For a graphical overview about all model performances please refer to **figure 1**, for a detailed tabular presentation please refer to **table 1**.

Table 1: Overview of performance of different classifiers

Classifier	Predictor set	Precision	Recall	f1-score	AUC
Dummy classifier (stratified)	All data	0.12	0.13	0.12	NA
Logistic regression	Clinical	0.38	0.23	0.29	0.69
	Laboratory	0.53	0.69	0.6	0.89
	All data	0.47	0.54	0.5	0.78
	All data, oversampling	0.47	0.54	0.5	0.78
Penalised Logistic regression: LASSO	Clinical	0	0	NA	0.71
	Laboratory	0	0	NA	0.84
	All data	0.5	0.08	0.14	0.87
K nearest neighbour	Clinical	0	0	NA	0.69
	Laboratory	0.5	0.08	0.14	0.88
	All data	0.5	0.15	0.23	0.84
	All data, oversampling	0.5	0.15	0.23	0.84
Radial kernel support vector machines	Clinical	0.5	0.15	0.23	0.73
	Laboratory	1	0.08	0.15	0.93
	All data	0.5	0.38	0.43	0.91
	All data, oversampling	0.5	0.31	0.38	0.91
Gradient boosting machine	Clinical	0.5	0.08	0.14	0.71
	Laboratory	0.64	0.54	0.59	0.91
	All data	0.8	0.62	0.7	0.93
	All data, oversampling	0.67	0.62	0.64	0.92

Table 1: continuing from previous page

Classifier	Predictor set	Precision	Recall	f1-score	AUC
Random forest	Clinical	0.5	0.15	0.23	0.78
	Laboratory	1	0.23	0.37	0.93
	All data	1	0.08	0.15	0.94
	All data, oversampling	1	0.08	0.15	0.93
Neural network	Clinical	0.5	0.38	0.43	0.91
	Laboratory	0.57	0.62	0.59	0.92
	All data	0.7	0.54	0.61	0.9
Meta-predict (ensemble with averaging)	All data	0.67	0.62	0.64	NA

All classifiers perform worse when only trained on clinical data and most classifiers have their best performance when using both clinical and laboratory data except for random forest that performs better when only using laboratory data.
Oversampling of the minority class does not lead to large improvements in f1-scores.
This study uses seeds to improve reproducibility, however, results will differ slightly on each new run of the models as e.g. the initialization of weights in a neural network will contain additional random elements.

3.2 Feature importance of predictor variables

Based on the random forest classifiers overall feature importance is estimated once for clinical variables and once for laboratory variables.
In clinical variables (see **figure 2**), age and body mass index (BMI, defines obesity) are the most important predictors that intriguingly have also been identified in other studies (Jimenez-Solem et al. 2021).

In laboratory variables mainly proinflammatory markers come up on top (interleukin 6, procalcitonin, CRP, monocytes, see **figure 3**). Interleukin 6 (IL-6) is by far the most important predictor variable dwarfing the other predictors. Also other studies have determined IL-6 as an important predictor of worse outcome and severe disease (Liu et al. 2020). Il-6 is also specifically targeted by specialised medications in cases of severe COVID-19 which is marked by a dysregulated immune system (Ghosn et al. 2021).

Figure 2: Scaled feature importance of clinical variables

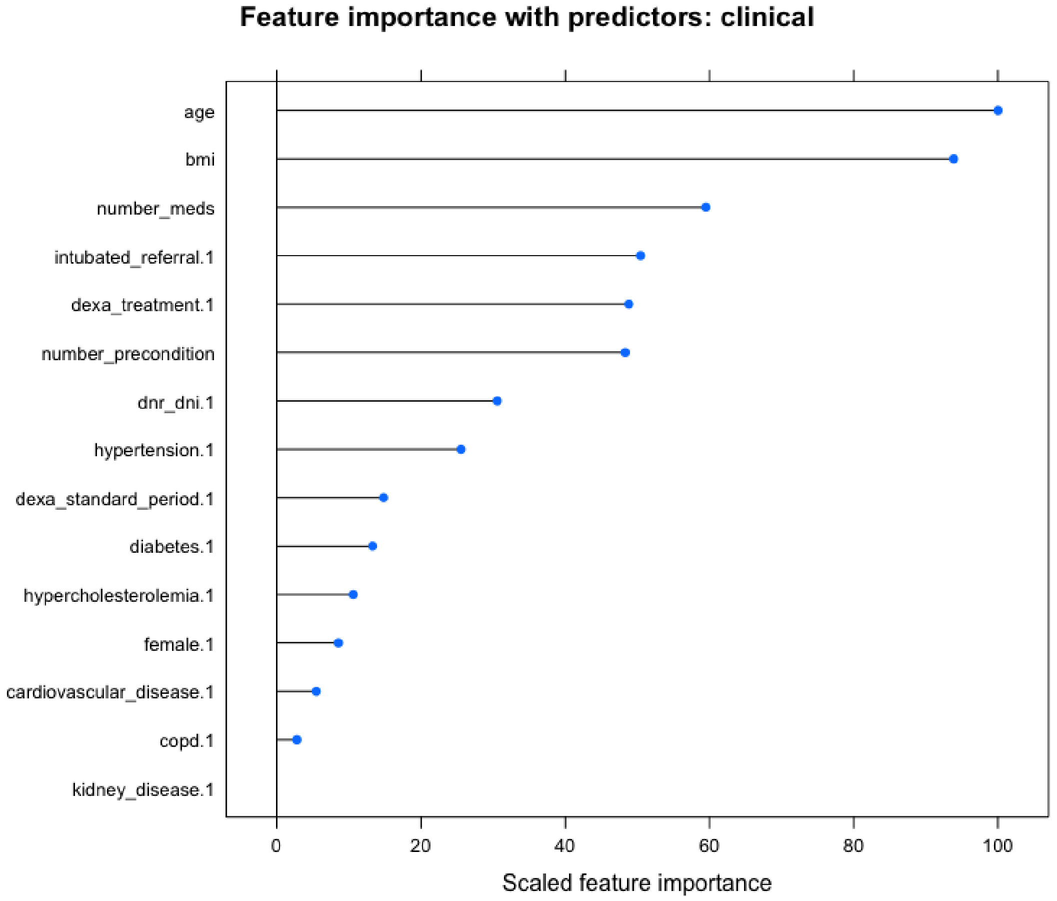
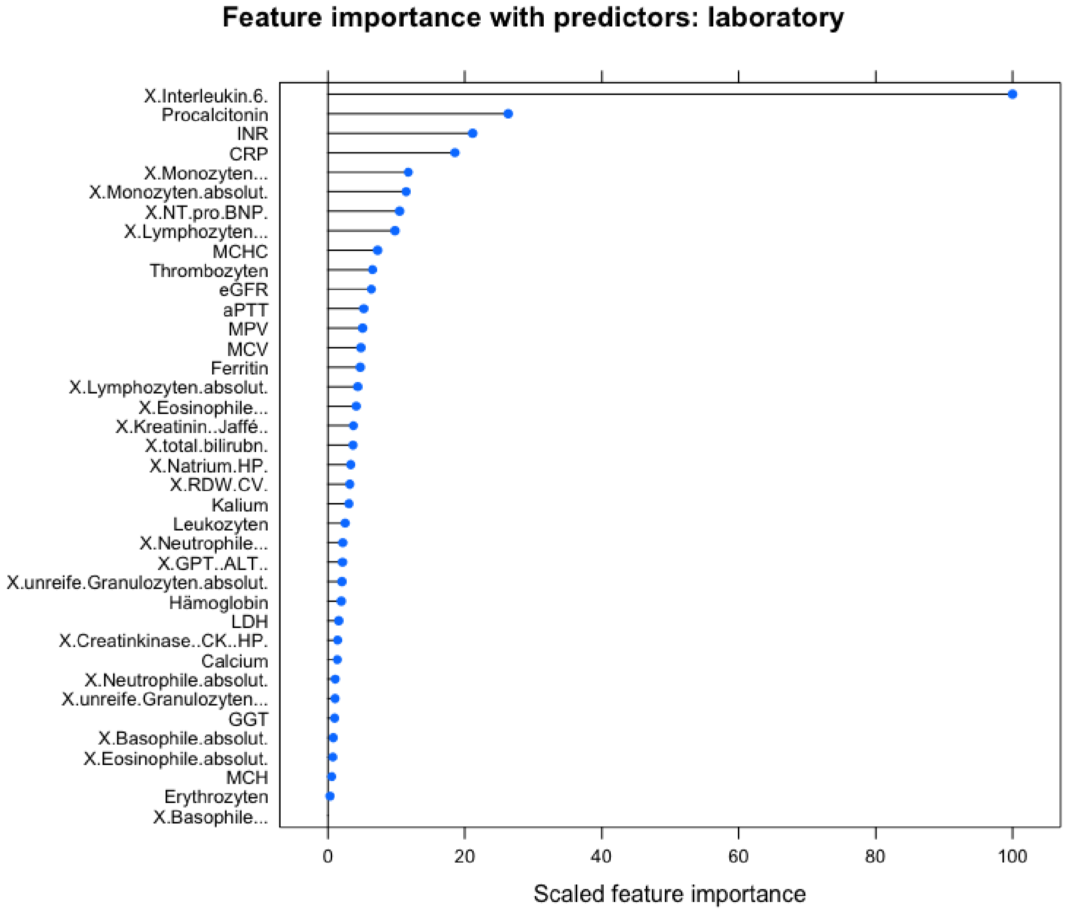


Figure 3: Scaled feature importance of laboratory variables



4. Discussion and Conclusions

All different classifiers outperformed the dummy classifier. This indicates that helpful information for predicting disease outcome is contained in the predictor variables. Logistic regression, a linear classifier, is already able to predict disease outcome with an f1-score of up to 60% which is mostly not improved by more complex modelling approaches except for gradient boosting machines. This is interesting as the complexity of logistic regression is lower when compared to random forests or neural networks. It illustrates that for tabular datasets a more traditional linear modelling approach with logistic regression is a good baseline.

The gradient boosting machine achieves an f1-score of up to 70% when using both clinical and laboratory data and is therefore the most successful classifier for this prediction task. It accurately identifies 62% of deceased patients in the test set (recall) while having a precision of 80%. While the recall is not exceptionally high, the high precision is important as otherwise the surviving patients, which are the majority of cases, will be falsely classified as having a high risk of dying. Still, a recall of only 62% means that almost half of the patients that will ultimately die are not accurately predicted of dying. Such a model may help in clinical settings but clinicians cannot rely on it. Further improvements of predictive modelling are needed for such a model to play a more central role in clinical treatment.

Combining clinical and laboratory data most often leads to the best performance. However, solely relying on clinical data shows worse results and when only using laboratory data the performance is in most cases almost equal to the combined use of clinical and laboratory data. In conclusion, a separation of outcomes by clinical variables is inferior to laboratory data. Although requiring expensive machinery and analysis pipelines, laboratory data seems advantageous for this prediction task.

Reasons for the limited predictive capabilities of clinical variables in this dataset may be the boolean encoding that might lead to a significant loss of information (Harrell 2015). In binary encoding (present/absent) of a disease, the disease severity is not reflected and instead mild disease and severe disease with multiple complications are put into the same category. A good illustrative example is the binary encoding of presence/absence of COPD (chronic bronchitis) which encompasses mild COPD where people use inhalers when needed to severe COPD that requires daily inhalation or even constant supplemental oxygen therapy (Voelkel, Mizuno, and Cool 2017).

The derived variable importance based on random forest is informative. Interestingly, it gives similar results to identified risk factors for severe COVID-19 of previous studies. Although variable importance estimation from random forests might not be as robust (Strobl et al. 2007), in this dataset derived variable importance is in accordance with other publications.

This study has several limitations: Firstly, the data has a relatively small sample size and cannot capture the wide clinical picture of COVID-19. Albeit not atypical for datasets from clinical studies that are often composed of only a few hundred patients, a small dataset can become problematic for complex statistical modelling that relies on large data (James et al. 2021).

Generalizability of the predictive models may be limited due to selection bias of the specific patients within the dataset. Collection of the data depended on the specific circumstances that drove the pandemic situation in Germany at this particular hospital at the time of data collection. Generally, selection bias is an important concern in COVID-19 research (Griffith et al. 2020).

The chosen preprocessing with a fixed splitting into training and a hold-out test set might be problematic as the test set may coincidentally contain a diverse set of patients than the training set. An alternative would be to use nested cross validation where performance of classifiers on all of the data can be assessed which requires more computational resources than k-fold cross validation.

To better compare neural networks to the other modelling approaches additional hyperparameter tuning with different neural architectures and learning rates as well as regularisation (L1 or L2 regularisation) should be tested (Chollet 2017).

To limit computational costs, nested cross validation as well as hyperparameter tuning of neural networks was not conducted in this study but may be beneficial in future studies.

It is challenging to model real world clinical data and to predict COVID-19 outcomes. Modelling of the data requires exchange with domain experts and data cleaning. Despite the early identification of risk factors for severe courses of COVID-19, the coherent prediction of outcomes remains challenging in multiple studies (Barish et al. 2020). This work also showed the challenge of modelling unbalanced data.

Adding more granular clinical measurements that accurately reflect severity of underlying medical conditions in patients could be beneficial.

To test the generalizability of the derived models, it would be useful to assess external validity by using a dataset from a different hospital. Here, a challenge lies in finding a dataset that collected the same clinical variables and used a standardised reporting of laboratory measurements with the same laboratory panel.

Future directions of this work include implementing interpretable machine learning with the aim to make the highly complex non-linear models such as gradient boosting machines and neural networks more interpretable. Particularly, post-hoc model-agnostic interpretation of predictions to derive overall feature importance with shapley values (Lundberg et al. 2020) seems promising but requires comparison of different implementations in R packages. Additionally recurrent architectures that can handle a sequential flow of clinical information seem promising. While computationally expensive such methods may give real time probabilities of patient outcome allowing for timely allocation of health resources (Meyer et al. 2018).

In summary, laboratory measurements and clinical variables show potential for COVID-19 disease outcome prediction. Laboratory measurements show better results than clinical variables in separating disease outcome classes. Gradient boosting machine is overall the best performing classifier in this study and outperforms a common sense baseline (dummy classifier) as well as a baseline classifier (logistic regression). Still, the achieved f1-scores need further improvement to apply such a model in clinical practice. Directions to improve the predictive classifiers in future studies were discussed.

6. Bibliography

Assaf, Dan, Ya ’ara Gutman, Yair Neuman, Gad Segal, Sharon Amit, Shiraz Gefen-Halevi, Noya Shilo, et al. 2020. “Utilization of Machine-Learning Models to Accurately Predict the Risk for Critical COVID-19.” *Internal and Emergency Medicine* 15 (8): 1435–43.

Barish, Matthew, Siavash Bolourani, Lawrence F. Lau, Sareen Shah, and Theodoros P. Zanos. 2020. “External Validation Demonstrates Limited Clinical Utility of the Interpretable Mortality Prediction Model for Patients with COVID-19.” *Nature Machine Intelligence* 3 (1): 25–27.

Breiman, Leo. 2001. “Random Forests.” *Machine Learning* 45 (1): 5–32.

Buuren, Stef van, and Karin Groothuis-Oudshoorn. 2011. “Mice: Multivariate Imputation by Chained Equations in R.” *Journal of Statistical Software* 45 (December): 1–67.

Chollet, Francois. 2017. *Deep Learning with Python*. 1st ed. USA: Manning Publications Co.

Domínguez-Olmedo, Juan L., Álvaro Gragera-Martínez, Jacinto Mata, and Victoria Pachón Álvarez. 2021. “Machine Learning Applied to Clinical Laboratory Data in Spain for COVID-19 Outcome Prediction: Model Development and Validation.” *Journal of Medical Internet Research* 23 (4): e26211.

Ferri, C., J. Hernández-Orallo, and R. Modroiu. 2009. “An Experimental Comparison of Performance Measures for Classification.” *Pattern Recognition Letters* 30 (1): 27–38.

Friedman, Jerome H. 2001. “Greedy Function Approximation: A Gradient Boosting Machine.” *The Annals of Statistics* 29 (5): 1189–1232.

Friedman, Jerome, Trevor Hastie, and Rob Tibshirani. 2010. “Regularization Paths for Generalized Linear Models via Coordinate Descent.” *Journal of Statistical Software* 33 (1): 1–22.

Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. 2000. “Additive Logistic Regression: A Statistical View of Boosting (With Discussion and a Rejoinder by the Authors).” *The Annals of Statistics* 28 (2): 337–407.

Fritsch S, Guenther F. 2019. “Package ‘neuralnet.’” Training of Neural Networks. February 7, 2019. https://cran.r-project.org/web/packages/neuralnet/neuralnet.pdf.

Ghosn, Lina, Anna Chaimani, Theodoros Evrenoglou, Mauricia Davidson, Carolina Graña, Christine Schmucker, Claudia Bollig, et al. 2021. “Interleukin‐6 Blocking Agents for Treating COVID‐19: A Living Systematic Review.” *Cochrane Database of Systematic Reviews* , no. 3. https://doi.org/10.1002/14651858.CD013881.

Griffith, Gareth J., Tim T. Morris, Matthew J. Tudball, Annie Herbert, Giulia Mancano, Lindsey Pike, Gemma C. Sharp, et al. 2020. “Collider Bias Undermines Our Understanding of COVID-19 Disease Risk and Severity.” *Nature Communications* 11 (1): 5749.

Harrell, Frank E. 2015. *Regression Modeling Strategies*. Springer International Publishing.

Hastie T, Quian J. 2022. “An Introduction to `glmnet`.” Glmnet Stanford. 2022. https://glmnet.stanford.edu/articles/glmnet.html.

Heaton, Jeff. 2008. *Introduction to Neural Networks with Java*. Heaton Research, Inc.

Heinze, Georg, Christine Wallisch, and Daniela Dunkler. 2018. “Variable Selection - A Review and Recommendations for the Practicing Statistician.” *Biometrical Journal. Biometrische Zeitschrift* 60 (3): 431–49.

Hernan, Miquel A., and James M. Robins. 2020. *Causal Inference*. Taylor & Francis.

Hornik, Kurt. 1991. “Approximation Capabilities of Multilayer Feedforward Networks.” *Neural Networks: The Official Journal of the International Neural Network Society* 4 (2): 251–57.

Howard, Jeremy, and Sylvain Gugger. 2020. *Deep Learning for Coders With Fastai and Pytorch: AI Applications Without a Phd*. O’Reilly Media, Inc, USA.

Hu, Ben, Hua Guo, Peng Zhou, and Zheng-Li Shi. 2021. “Characteristics of SARS-CoV-2 and COVID-19.” *Nature Reviews. Microbiology* 19 (3): 141–54.

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2021. *An Introduction to Statistical Learning: With Applications in R (Springer Texts in Statistics)*. 2nd ed. Springer.

Jimenez-Solem, Espen, Tonny S. Petersen, Casper Hansen, Christian Hansen, Christina Lioma, Christian Igel, Wouter Boomsma, et al. 2021. “Developing and Validating COVID-19 Adverse Outcome Risk Prediction Models from a Bi-National European Cohort of 5594 Patients.” *Scientific Reports* 11 (1): 3246.

Kapoor, Sayash, and Arvind Narayanan. 2022. “Leakage and the Reproducibility Crisis in ML-Based Science.” *arXiv [cs.LG]*. arXiv. http://arxiv.org/abs/2207.07048.

Khanji, Cynthia, Lyne Lalonde, Céline Bareil, Marie-Thérèse Lussier, Sylvie Perreault, and Mireille E. Schnitzer. 2019. “Lasso Regression for the Prediction of Intermediate Outcomes Related to Cardiovascular Disease Prevention Using the TRANSIT Quality Indicators.” *Medical Care* 57 (1): 63–72.

Knaus, W. A., E. A. Draper, D. P. Wagner, and J. E. Zimmerman. 1985. “APACHE II: A Severity of Disease Classification System.” *Critical Care Medicine* 13 (10): 818–29.

Kuhn, Max. 2008. “Building Predictive Models in R Using the Caret Package.” *Journal of Statistical Software* 28 (November): 1–26.

Kuhn, Max, and Kjell Johnson. 2018. *Applied Predictive Modeling*. Springer New York.

Liu, Tao, Jieying Zhang, Yuhui Yang, Hong Ma, Zhenyu Li, Jiaoyue Zhang, Ji Cheng, et al. 2020. “The Role of Interleukin-6 in Monitoring Severe Case of Coronavirus Disease 2019.” *EMBO Molecular Medicine* 12 (7): e12421.

Lundberg, Scott M., Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. 2020. “From Local Explanations to Global Understanding with Explainable AI for Trees.” *Nature Machine Intelligence* 2 (1): 56–67.

Meyer, Alexander, Dina Zverinski, Boris Pfahringer, Jörg Kempfert, Titus Kuehne, Simon H. Sündermann, Christof Stamm, Thomas Hofmann, Volkmar Falk, and Carsten Eickhoff. 2018. “Machine Learning for Real-Time Prediction of Complications in Critical Care: A Retrospective Study.” *The Lancet. Respiratory Medicine* 6 (12): 905–14.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, et al. 2011. “Scikit-Learn: Machine Learning in Python.” *Journal of Machine Learning Research: JMLR* 12 (85): 2825–30.

RECOVERY Collaborative Group, Peter Horby, Wei Shen Lim, Jonathan R. Emberson, Marion Mafham, Jennifer L. Bell, Louise Linsell, et al. 2021. “Dexamethasone in Hospitalized Patients with Covid-19.” *The New England Journal of Medicine* 384 (8): 693–704.

Saito, Takaya, and Marc Rehmsmeier. 2015. “The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets.” *PloS One* 10 (3): e0118432.

Shipe, Maren E., Stephen A. Deppen, Farhood Farjah, and Eric L. Grogan. 2019. “Developing Prediction Models for Clinical Use Using Logistic Regression: An Overview.” *Journal of Thoracic Disease* 11 (Suppl 4): S574–84.

Shwartz-Ziv, Ravid, and Amitai Armon. 2022. “Tabular Data: Deep Learning Is Not All You Need.” *An International Journal on Information Fusion* 81 (May): 84–90.

Strobl, Carolin, Anne-Laure Boulesteix, Achim Zeileis, and Torsten Hothorn. 2007. “Bias in Random Forest Variable Importance Measures: Illustrations, Sources and a Solution.” *BMC Bioinformatics* 8 (January): 25.

Suits, Daniel B. 1957. “Use of Dummy Variables in Regression Equations.” *Journal of the American Statistical Association* 52 (280): 548–51.

Voelkel, Norbert F., Shiro Mizuno, and Carlyne D. Cool. 2017. “The Spectrum of Pulmonary Disease in COPD.” In *COPD: Heterogeneity and Personalized Treatment*, edited by Sang-Do Lee, 195–207. Berlin, Heidelberg: Springer Berlin Heidelberg.

“WHO Coronavirus (COVID-19) Dashboard.” n.d. Accessed September 10, 2022. https://covid19.who.int/.

Witten, Ian H., Eibe Frank, and Mark A. Hall. 2011. *Data Mining: Practical Machine Learning Tools and Techniques*. Elsevier.

Appendix (Supplementary material)

Supplement Table 1: Definition of clinical and laboratory variables
While all of the laboratory data is of numeric (continuous) datatype, the clinical predictors are mostly categorical except certain continuous features (e.g., age, number of medication).

Predictor type	Name of predictor variables	Datatype
Clinical variables	● age ● number_meds ● hypertension ● cardiovascular_disease ● copd ● diabetes ● hypercholesterolemia ● kidney_disease ● intubated_referral ● deceased ● dnr_dni ● female ● dexa_treatment ● bmi ● dexa_standard_period ● number_precondition	Mostly categorical
Laboratory variables	● Monozyten % ● MCH ● Neutrophile % ● Thrombozyten ● unreife Granulozyten % ● Basophile % ● MPV ● Lymphozyten % ● Erythrozyten ● MCV ● Leukozyten ● Eosinophile % ● Neutrophile absolut ● MCHC ● Hämoglobin ● unreife Granulozyten absolut ● Eosinophile absolut ● Basophile absolut ● RDW-CV ● Monozyten absolut ● Lymphozyten absolut ● INR ● aPTT ● Kalium ● Natrium HP ● CRP ● Creatinkinase (CK) HP ● LDH ● GPT (ALT) ● GGT ● total bilirubn	Numeric

	● Calcium ● Kreatinin (Jaffé) ● eGFR ● NT-pro BNP ● Procalcitonin ● Interleukin-6 ● Ferritin	