https://doi.org/10.31891/csit-2022-2-12
UDC 004.8

Andrii POPOVYCH, Vitalii YAKOVYNA
Lviv Polytechnic National University

COVID-19 MORTALITY PREDICTION USING MACHINE LEARNING METHODS

The paper reports the use of machine learning methods for COVID-19 mortality prediction. An open dataset with large number of features and records was used for research. The goal of the research is to create the efficient model for mortality prediction which is based on large number of factors and enables the authorities to take actions to avoid mass spread of virus to and reduce the number of cases and deaths. Feature selection was conducted in order to remove potentially irrelevant input variables and improve performance of machine learning models. The classic machine learning models (both linear and non-linear), ensemble methods such as bagging, stacking and boosting, as well as neural networks, is used. Comparison of efficiency of ensemble methods and neural networks compared to classic ML methods such as linear regression, Support Vector Machines, K-nearest neighbors etc. is conducted. Ensemble methods and neural networks show much greater efficiency than classical ones. Feature selection does not significantly affect the prediction accuracy.

The scientific novelty of this paper is the large number of machine learning models trained on the large-scale dataset with significant number of features related to different factors that can potentially affect COVID-19 mortality, as well as further analysis of their efficiency. This will assist to select the most valuable features and to become a basis for creating a software designed for tracking the dynamics of the pandemic.

The practical significance of this paper is that present study can be useful for authorities and international organizations in prevention of COVID-19 mortality increase by taking proper preventive measures.

Keywords: machine learning, COVID-19, mortality prediction, ensemble methods, neural networks, feature selection.

Андрій ПОПОВИЧ, Віталій ЯКОВИНА
Національний університет «Львівська політехніка»

ПРОГНОЗУВАННЯ СМЕРТНОСТІ ВІД COVID-19 МЕТОДАМИ МАШИННОГО НАВЧАННЯ

Дана стаття описує використання методів машинного навчання для передбачення рівня смертності від COVID-19. Для дослідження було використано відкритий набір даних з великою кількістю ознак та записів. Метою даного дослідження є створення ефективної моделі для передбачення рівня смертності, що базується на великій кількості чинників та дозволить компетентним органам вжити превентивні заходи для запобігання масовому поширенню COVID-19 та зменшення кількості хворих та померлих від хвороби. Проведено відбір ознак з метою усунення потенційно нерелевантних вхідних змінних та покращення продуктивності моделей машинного навчання. Було використано класичні моделі машинного навчання (як лінійні, так і нелінійні), ансамблеві методи, зокрема беггінг, стекінг та бустинг, а також нейронні мережі. Виконано порівняння ефективності ансамблевих методів порівняно з класичними методами машинного навчання, такими як лінійна регресія, методи опорних векторів, K найближчих сусідів та інші. Ансамблеві методи та нейронні мережі показують значно більшу ефективність, ніж класичні. Відбір ознак не має значного впливу на точність передбачення.

Наукова новизна даної роботи полягає в великій кількості моделей машинного навчання, натренованих на великому наборі даних, що містить значну кількість ознак, які стосуються різноманітних чинників, які потенційно можуть вплинути на смертність від COVID-19, та в подальшому аналізі їх ефективності. Це може допомогти відібрати найбільш значущі ознаки та стати основою у створенні програмних засобів, призначених для відстеження динаміки хвороби.

Практичне значення даної роботи полягає в тому, що наявні в ній дослідження можуть бути корисні для дослідників, закладів охорони здоров'я, державних органів та міжнародних організацій в запобіганні зростання смертності від COVID-19 шляхом вжиття відповідних запобіжних заходів.

Ключові слова: машинне навчання, прогнозування смертності від COVID-19, ансамблеві методи, нейронні мережі, відбір ознак.

Introduction

The COVID-19 pandemic caused by SARS-CoV-2 strain, which started in December 2019 in Wuhan (Hubei province, China), triggered severe global social and economic outcomes around the world. As of May 29, 2022, more than 528 million cases have been registered worldwide, including more than 6.28 million deaths. By the late 2020 - early 2021 when the mass production of vaccines and the mass vaccination started, in order to reduce morbidity and mortality the governments were forced to take strict preventive measures such as lockdowns, social distancing, travel restrictions, wearing masks, quarantines, curfews, workplace hazard controls, postponing or cancelling the events, testing systems, etc.

To mitigate the effects of pandemic and reduce the number of casualties it is crucial to have an instrument which considers different factors that can significantly affect the course of the pandemic, in particular demographic, economic, geographical, etc. This will enable researchers and authorities to better understand dynamics of the pandemic and take proper preventive actions.

The paper describes research and efficiency comparison of different machine learning models using large-size dataset with many features which will potentially improve mortality prediction accuracy.

Related works

In more than two years since the outbreak of the pandemic, a large number of studies have been conducted to predict the COVID-19 mortality rate. Most of them use the clinical and laboratory results of hospitalized patients as input data. These studies used different models of machine learning, feature selection methods, as well as metrics and indicators, which assessed the effectiveness of the models and the quality of their predictions.

Early mortality prediction using machine learning based on based on typical laboratory results and clinical data registered on the day of intensive care unit admission is considered in [1]. Such machine learning algorithms as Random Forest, logistic regression, gradient boosting classifier, Support Vector Machine classifier, and artificial neural network algorithms were used to build classification models. The impact of each marker on the RF model predictions was studied by implementing the LIME-SP technique. The study [2] aimed to compare several ML algorithms to predict the COVID-19 mortality using the patient's data at the first time of admission. An Information GainRatio Attribute evaluation (GA) method was used to select the features. Seven ML algorithms including the J48 decision tree, Random Forest, K-nearest neighborhood, multi-layer perceptron, Naïve Bayes, eXtreme gradient boosting (XGBoost), and logistic regression were applied. Random Forest had better performance than other ML algorithms.

In the study [3] inspired modification of partial least square (SIMPLS)-based model was developed to predict hospital mortality. Latent class analysis (LCA) was carried to cluster the patients with COVID-19 to identify low- and high-risk patients. SIMPLS-based model was able to predict hospital mortality with moderate predictive power and high accuracy. Clustering analysis identified high- and low-risk patients among COVID-19 survivors. The aim of the next study [4] was the development and prospective validation of a state-of-the-art machine learning model to provide mortality prediction within 72 hours after confirmation of SARS-CoV-2 infection. Traditional machine learning models were evaluated independently as well as in a stacked learner and various recurrent neural network architectures were considered. The GRU-D recurrent neural network achieved peak cross-validation performance.

The study [5] aims to train several ML algorithms to predict the COVID-19 in-hospital mortality and compare their performance to choose the best performing algorithm. Six feature scoring techniques and nine well-known ML algorithms were used. To evaluate the models' performances, the metrics derived from the confusion matrix calculated. Experimental results indicated that the Bayesian network algorithm has been more successful in predicting mortality. This study [6] was conducted to develop a machine learning model to predict prognosis based on sociodemographic and medical information. The least absolute shrinkage and selection operator (LASSO), linear Support Vector Machine, SVM with radial basis function kernel, Random Forest and K-nearest neighbors were tested. LASSO and linear SVM demonstrated high sensitivities and specificities while maintaining high specificities, as well as high area under the receiver operating characteristics curves.

Prediction of in-hospital mortality for COVID-19 patients treated with steroid and remdesivir was conducted in [7]. The important variables associated with in-hospital mortality were identified using LASSO and SHAP (SHapley Additive exPlanations) through the light gradient boosting model (GBM). Six important variables were selected. Additionally, the light GBM had high predictability for the latest data (AUC: 0.881). This study [8] aimed to develop a predictive model to predict patients' mortality from the basic medical data on the first day of admission. From different ML models the naive Bayes demonstrated the best performance with an AUC of 0.85. The ensemble model from the naive Bayes and neural network combination had slightly better performance.

The study [9] aimed to develop and compare prognosis prediction machine learning models based on invasive laboratory and noninvasive clinical and demographic data from patients' day of admission. Three SVM models were developed and compared using invasive, non-invasive, and both groups. The results suggested that non-invasive features could provide mortality predictions that are similar to the invasive. The next study [10] experimentally verified that some anti-cancer drugs can be regarded as potential treatments against COVID-19. A broad panel of time-to-event machine learning models was implemented and compared, such as Elastic net penalized Cox proportional hazards regression and Weibull accelerated failure time regression, DeepSurv neural network approach, Random Survival Forests and XGBoost Survival Embeddings.

The purpose of study [11] is to predict new cases and deaths rate one, three and seven-day ahead during the next 100 days. Three methods (LSTM, Convolutional LSTM, and GRU) and their bidirectional variants were used. The results show that the bidirectional models have lower errors than other models. The next study [12] is about development and testing of machine learning-based models for COVID-19 severity prediction. In this research, a new feature engineering method based on topological data analysis called Uniform Manifold Approximation and Projection (UMAP) were used. UMAP has 100% accuracy, specificity, sensitivity, and ROC curve in conducting a prognostic prediction using different machine learning classifiers.

In the study [13] authors developed, verified, and deployed a stacked generalization model to predict mortality by combining 5 previously validated scores and additional novel variables reported to be associated with COVID-19-specific mortality. A ridge regularized logistic regression was chosen as the top-level model to limit overfitting and to address correlation between the component models. The objective of the next study [14] was to develop and validate models that predict mortality of patients diagnosed with COVID-19 admitted to the hospital. A

linear logistic regression and non-linear tree-based gradient boosting algorithm were used. Both models outperformed age-based decision rules used in practice.

The objective of study [15] was to identify prognostic serum biomarkers in patients at greatest risk of mortality. The developed Support Vector Machine model achieved 91% sensitivity and 91% specificity (AUC 0.93) for predicting patient expiration status on held-out testing data. The next study [16] aimed to develop risk scores based on clinical characteristics at presentation to predict ICU admission and mortality in COVID-19 patients. Logistic regression was used to identify independent clinical variables predicting the two outcomes. The risk score model yielded good accuracy for predicting ICU admission and for predicting mortality for the testing dataset.

The next study [17] leverages a database of blood samples to identify crucial predictive biomarkers of disease mortality. For this purpose, multi-tree XGBoost classifier selected three biomarkers that predict the mortality of individual patients more than 10 days in advance with more than 90% accuracy. The aim of next study [18] was to develop an accurate model for predicting COVID-19 mortality using epidemiolocal and clinical variables and for identifying a high-risk group of confirmed patients. Risk scores for COVID-19 mortality prediction model were developed by logistic regression analysis.

This study [19] seeks to develop and validate a data-driven personalized mortality risk calculator for hospitalized COVID-19 patients. The COVID-19 Mortality Risk tool was developed using the XGBoost algorithm to predict mortality. In the last study [20] a bootstrap averaged ensemble of Bayesian networks was also learned to construct an explainable model for discovering actionable influences on mortality and days to outcome. XGboost and logistic regression model yielded the best performance on risk stratification and mortality prediction respectively.

As we can see, the vast majority of studies related to the COVID-19 mortality prediction of from focus on predicting the survival of individual patients who have been hospitalized with a confirmed diagnosis. These studies are based on data provided by health facilities. So, the aim of this study to predict the COVID-19 mortality rate among the population on the basis of a large number of potentially relevant factors that may affect the pandemic. This task involves the selection of the appropriate set of input data, as well as the selection of the optimal prediction method and the factors influencing its results.

Dataset description and exploratory data analysis

An open dataset [21] which contains data related to COVID-19 outbreak in the US, including data from 3142 counties of 49 US states from the beginning of the outbreak (January 2020) to June 2021, was used for study given in this paper.

This data was collected from many public scientific, governmental and other online databases and include daily number of COVID 19 confirmed cases and deaths and features, as well as features that may be relevant to the dynamics of the pandemic: demographic, geographic, climatic, social, etc.

The dataset consists of 992266 records and 64 features. The target variable is daily number of COVID-19 deaths in each county.

The dataset is essentially an aggregation of big amount of data collected from large number of open sources. The data in the dataset were preliminarily prepared by its authors. In particular, KNNimputer was used to impute missing data, and the records about counties with values of both fixed features and temporal features missed for all dates were deleted.

The correlation matrices for some features are presented in Fig. 1. We can see that significant correlation between them and target variable is absent.

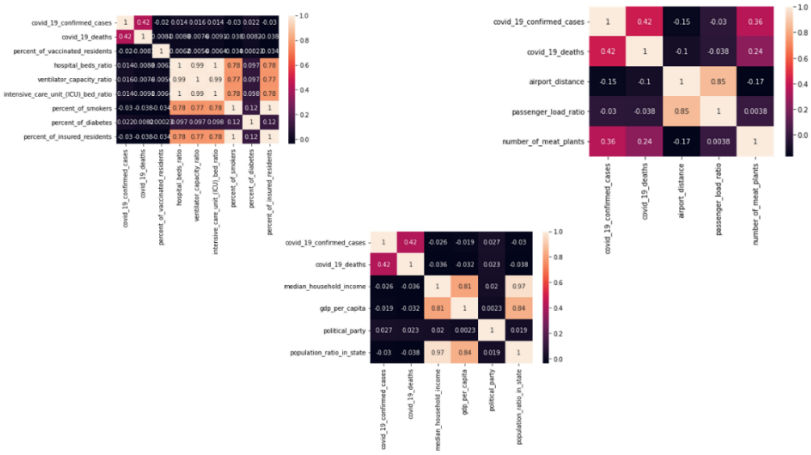

Fig.1. Pearson correlation coefficients matrices for some features

Feature selection

As the dataset contains large number of features, it is necessary conduct feature selection to select a set of input variables that are the relevant to target variable. This will potentially reduce the dimensionality of the training

set, improve model performance and reduce it fitting time. As it is unknown what set of features will be optimal, the following algorithms were used:

1) Boruta [22]. This algorithm based on Random Forest creates random shuffled shadow copies for each feature and determines their Z-scores. Feature is removed if its score is lower than maximum score of its shadow copies. 6 features were selected by this algorithm (9.52% of total number of features).

2) Recursive Feature Elimination (RFE) [23]. This algorithm uses an external estimator to assign some weight coefficients to initial set of features, then features with the lowest weights are pruned. Procedure is recursively repeated until the desired number of features is reached. 32 features were selected by this algorithm (50.7% of total number of features).

3) Recursive Feature Elimination with cross-validation (RFECV) [24] which allows to get the optimal set of features. 22 features were selected by this algorithm (34.9% of total number of features.

Comparison of efficiency of different machine learning models

The first step is applying linear machine learning models to both the entire dataset and the selected features. Such models as linear [25], logistic [26], ridge [27] and ElasticNet [28] regression, as well as stochastic gradient descent [29], were used. For model evaluation, metrics such as mean absolute error (MAE), mean squared error (MSE), its root (RMSE) and coefficient of determination (R^2 score) were used. Data was split with ratio: 75% - training set, 25% - test set. Results are presented in Table 1.

Table 1.

Comparison of efficiency of linear models for different sets of features

Model/metric	MAE	MSE	R^2 score	RMSE
For all features				
Linear	0.674	8.265	0.329	2.875
Logistic	0.480	10.163	0.174	3.188
Ridge	0.674	8.265	0.329	2.875
ElasticNet	0.691	9.198	0.253	3.033
SGD	0.628	8.366	0.320	2.892
For features selected by Boruta algorithm				
Linear	0.637	8.367	0.320	2.893
Logistic	0.480	11.052	0.102	3.324
Ridge	0.637	8.367	0.320	2.893
ElasticNet	0.691	9.198	0.253	3.033
SGD	0.641	8.502	0.309	2.916
For features selected by RFE algorithm				
Linear	0.674	8.265	0.329	2.875
Logistic	0.480	10.163	0.174	3.188
Ridge	0.674	8.265	0.329	2.875
ElasticNet	0.691	9.198	0.253	3.033
SGD	0.628	8.366	0.320	2.892
For features selected by RFECV algorithm				
Linear	0.675	8.293	0.326	2.880
Logistic	0.479	10.545	0.143	3.247
Ridge	0.674	8.293	0.326	2.880
ElasticNet	0.693	9.267	0.247	3.044
SGD	0.660	8.709	0.292	2.951

The next step is the analysis of efficiency of some non-linear machine learning models. The following methods were used: K-nearest neighbors [30], Support Vector Machine [31], decision tree [32]. Results are presented in Table 2.

Table 2.

Comparison of efficiency of non-linear models for different sets of features

Model/metric	MAE	MSE	R^2 score	RMSE
For all features				
DecisionTree	0.606	11.942	0.030	3.456
SVR	0.487	9.589	0.221	3.097
KNeighbors	0.602	9.820	0.202	3.137
For features selected by Boruta algorithm				
DecisionTree	0.630	14.910	0.021	3.860
SVR	0.484	9.733	0.210	3.120
KNeighbors	0.612	10.010	0.187	3.164
For features selected by RFE algorithm				
DecisionTree	0.606	11.942	0.030	3.456
SVR	0.487	9.589	0.221	3.097
KNeighbors	0.604	10.846	0.193	3.293
For features selected by RFECV algorithm				
DecisionTree	0.611	14.930	0.021	3.864
SVR	0.486	9.659	0.215	3.108
KNeighbors	0.612	9.905	0.206	3.147

In general, non-linear models with selected features show slightly worse results than with entire dataset.

The next step is to compare ensemble methods, in particular:

1) Bootstrap aggregation (bagging) [33] - algorithm is trained on random data subsets several times, then the results are averaged. In this study decision tree and Random Forest [34] are used.

2) Boosting [35] - several algorithms are trained consistently; each subsequent algorithm focuses on samples misclassified by previous ones. Gradient boosting [36] (based on decision tree), AdaBoost [37] and XGBoost [38] were used.

3) Stacked generalization (stacking) [39] - several algorithms are trained using the available data, then the results are used as inputs by final estimator which makes the final decision. Gradient boosting, decision tree and Random Forest were used to create ensemble. Results are presented in Table 3.

Table 3.

Comparison of efficiency of ensemble models for different sets of features

Model/metric	MAE	MSE	R^2 score	RMSE
For all features				
AdaBoost	0.451	6.280	0.490	2.506
Bagging	0.505	5.390	0.562	2.322
Gradient Boosting	0.549	5.716	0.536	2.391
XGB	0.508	5.390	0.562	2.322
Random Forest	0.505	5.419	0.560	2.328
Stacking	0.497	5.145	0.582	2.268I
For features selected by Boruta algorithm				
AdaBoost	0.456	6.286	0.489	2.507
Bagging	0.512	5.583	0.546	2.362
Gradient Boosting	0.551	5.906	0.520	2.430
XGB	0.516	5.700	0.537	2.387
Random Forest	0.511	5.182	0.579	2.276
Stacking	0.506	5.091	0.586	2.256
For features selected by RFE algorithm				
AdaBoost	0.451	6.280	0.490	2.506
Bagging	0.505	5.390	0.562	2.321
Gradient Boosting	0.549	5.716	0.536	2.391
XGB	0.508	5.390	0.562	2.322
Random Forest	0.505	5.419	0.560	2.328
Stacking	0.497	5.145	0.582	2.268
For features selected by RFECV algorithm				
AdaBoost	0.465	6.845	0.444	2.616
Bagging	0.508	6.140	0.501	2.478
Gradient Boosting	0.550	5.890	0.521	2.427
XGB	0.514	6.284	0.489	2.507
Random Forest	0.508	6.168	0.499	2.484
Stacking	0.516	5.559	0.548	2.358

We can see that results of ensemble models are much better than results of models mentioned above.

Finally, let's compare efficiency of some deep learning models. For comparison, two neural networks with experimentally selected topologies were created.

The first one is multilayer perceptron [40] neural network, it has four fully connected layers (one input layer and three hidden ones), each of then consists of 256, 128, 64 and 32 nodes respectively. A Rectified Linear Unit (ReLU) activation function is applied to each layer. After every layer we use Dropout layer, which is used for network regularization using neurons exclusion with certain rate (0.2 in our case) to prevent overfitting. Adam optimizer was selected and number of epochs is 100.

The second one is convolutional neural network [41], which contains one input layer with 64 nodes and one hidden layer with 32 nodes. The Flatten layer designed for converting input data into one-dimensional vector, as well as ReLU activation function and Adam optimizer is used.

Table 4.

Comparison of efficiency of neural networks for different sets of features

Model/metric	MAE	MSE	R^2 score	RMSE
For all features				
MLP	0.588	5.641	0.542	2.375
CNN	0.522	5.782	0.530	2.405
For features selected by Boruta algorithm				
MLP	0.528	5.526	0.551	2.351
CNN	0.608	5.975	0.514	2.444
For features selected by RFE algorithm				
MLP	0.567	6.072	0.507	2.464
CNN	0.603	6.171	0.499	2.484
For features selected by RFECV algorithm				
MLP	0.558	6.723	0.454	2.593
CNN	0.515	5.741	0.534	2.396

As it is shown above, the performance of neural networks is slightly lower than ensembles.

Discussion

The comparison graphs, where efficiency of studied machine learning models for both all and selected features is displayed, are shown in Fig. 2-5.

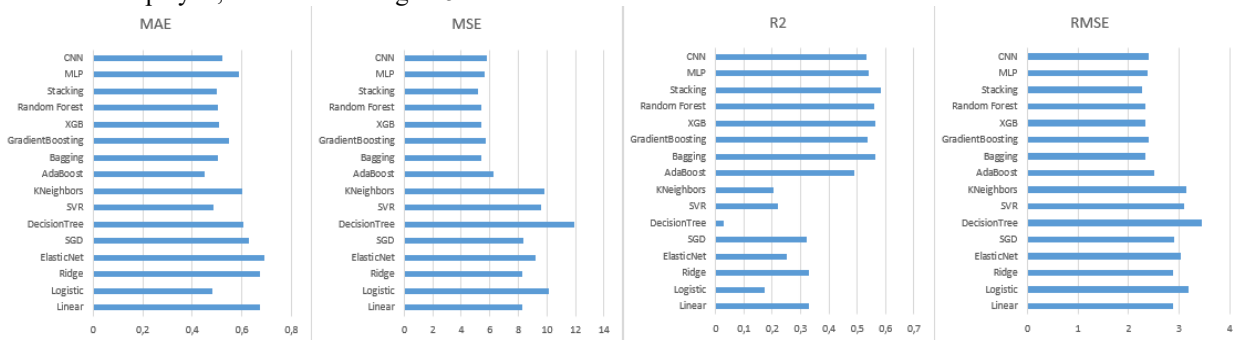

Fig.2. Comparison of efficiency of models (all features)

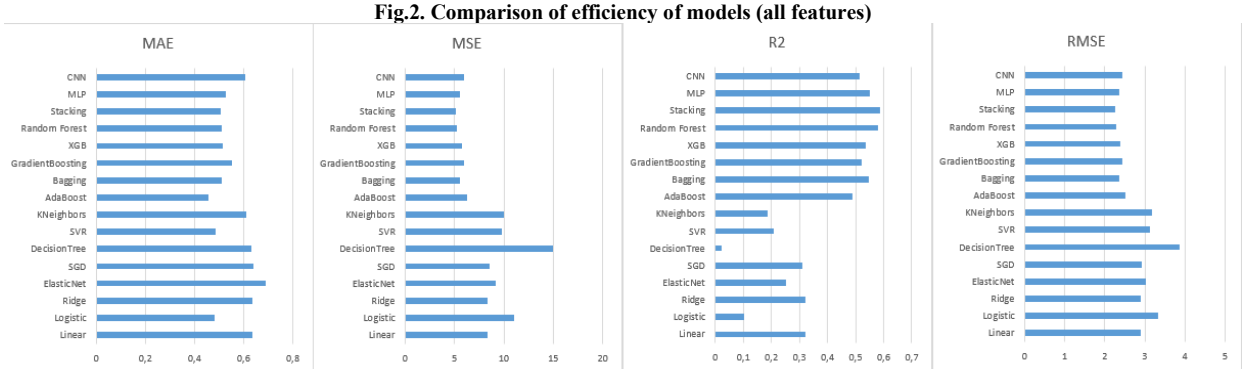

Fig.3. Comparison of efficiency of models (features selected by Boruta algorithm)

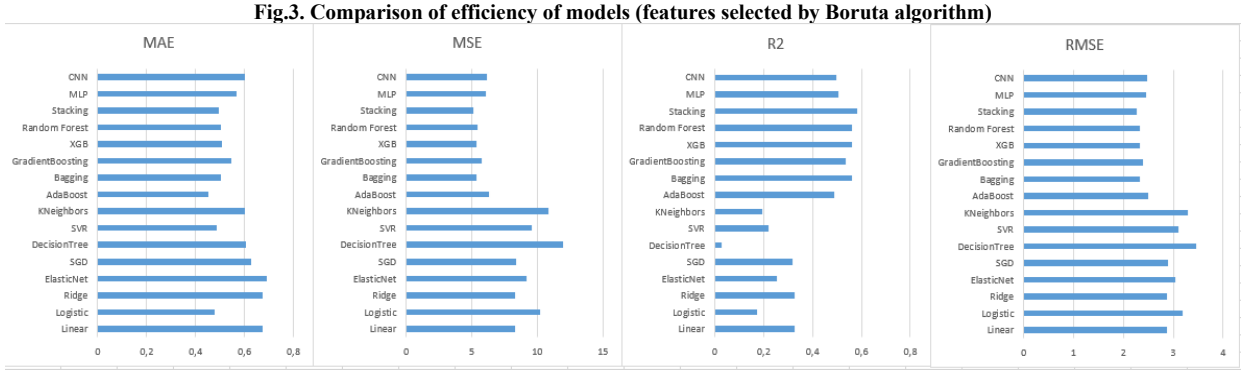

Fig.4. Comparison of efficiency of models (features selected by RFE algorithm)

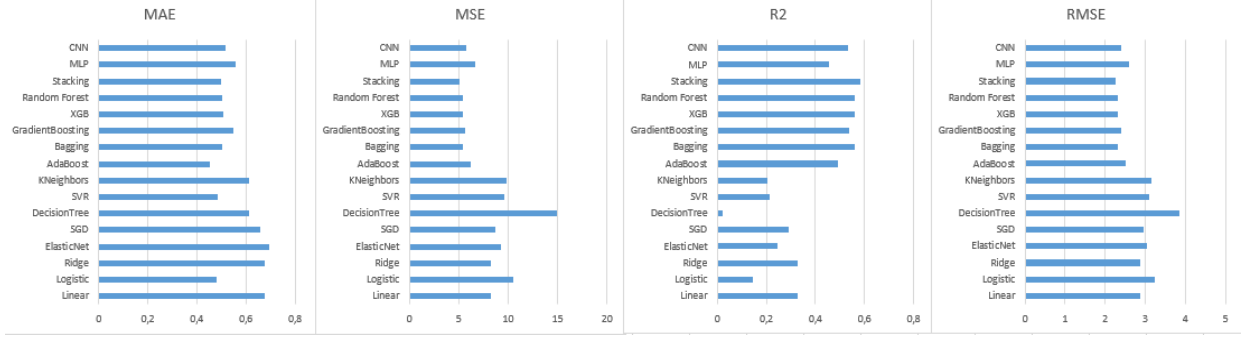

Fig.5. Comparison of efficiency of models (features selected by RFECV algorithm)

Ensemble methods and neural networks give better results compared to classic methods. Developed method improved generalization abilities.

Ensemble methods combine predictions of multiple trained models. The drawback of this approach is that contribution every model makes to ensemble is the same and does not depend on performance of model. The modification of this approach is a weighted average ensemble [42] that weighs contribution of every ensemble member by the expected performance of the model on a holdout dataset. This means that model contribution depends on its performance. This improves average weighted ensemble over average model ensemble.

The main problem related to usage of neural networks is impossibility to select architecture optimal to solve specific task in advance. Selection of suitable configuration is conducted experimentally, such methods as random search, heuristic search, grid search, etc. is often used.

Developed methods for solving the COVID-19 mortality prediction showed significant increase of accuracy compared to existing approaches (decision trees, K-nearest neighbors, Support Vector Machines, linear regression, etc.).

The results are presented both for the entire dataset and selected features, and the results of the metrics in all cases differ slightly.

Conclusions

The subject of this paper is creation of optimal machine learning designed for COVID-19 mortality prediction task, which can be useful for researchers, governments and international organizations to take preventive actions.

The dataset used for study was analyzed, feature selection was conducted, selected models were trained and their efficiency was compared.

Ensemble methods (stacking, bagging and boosting) as well as neural networks were found to be the most efficient. Prediction accuracy may be improved in future studies.

It was discovered that addition of a new predictor can increase the accuracy of prediction, because the output data of the base predictors are input data for the final predictor. In this case, these features are probably correlated, as all basic predictors try to predict the same result.

References

1. Jamshidi E, Asgary A, Tavakoli N, Zali A, Setareh S, Esmaily H, Jamaldini SH, Daaee A, Babajani A, Sendani Kashi MA, Jamshidi M, Jamal Rahi S and Mansouri N (2022) Using Machine Learning to Predict Mortality for COVID-19 Patients on Day 0 in the ICU. Front. Digit. Health 3:681608. doi: 10.3389/fdgth.2021.681608.

2. Moulaei, K., Shanbehzadeh, M., Mohammadi-Taghiabad, Z. et al. Comparing machine learning algorithms for predicting COVID-19 mortality. BMC Med Inform Decis Mak 22, 2 (2022). https://doi.org/10.1186/s12911-021-01742-0.

3. Banoei, M.M., Dinparastisaleh, R., Zadeh, A.V. et al. Machine-learning-based COVID-19 mortality prediction model and identification of patients at low and high risk of dying. Crit Care 25, 328 (2021). https://doi.org/10.1186/s13054-021-03749-5.

4. Sankaranarayanan S, Balan J, Walsh JR, Wu Y, Minnich S, Piazza A, Osborne C, Oliver GR, Lesko J, Bates KL, Khezeli K, Block DR, DiGuardo M, Kreuter J, O'Horo JC, Kalantari J, Klee EW, Salama ME, Kipp B, Morice WG, Jenkinson G COVID-19 Mortality Prediction From Deep Learning in a Large Multistate Electronic Health Record and Laboratory Information System Data Set: Algorithm Development and Validation J Med Internet Res 2021;23(9):e30157 doi: 10.2196/30157 PMID: 34449401 PMCID: 8480399.

5. Shanbehzadeh M, Orooji A, Kazemi-Arpanahi H. Comparing of Data Mining Techniques for Predicting in-Hospital Mortality Among Patients with COVID-19. JBE. 2021;7(2):154-173.

6. An, C., Lim, H., Kim, DW. et al. Machine learning prediction for mortality of patients diagnosed with COVID-19: a nationwide Korean cohort study. Sci Rep 10, 18716 (2020). https://doi.org/10.1038/s41598-020-75767-2.

7. Kuno, T, Sahashi, Y, Kawahito, S, Takahashi, M, Iwagami, M, Egorova, NN. Prediction of in-hospital mortality with machine learning for COVID-19 patients treated with steroid and remdesivir. J Med Virol. 2022; 94: 958- 964. doi:10.1002/jmv.27393.

8. Tabatabaie M, Sarrami A, Didehdar M, et al. (October 14, 2021) Accuracy of Machine Learning Models to Predict Mortality in COVID-19 Infection Using the Clinical and Laboratory Data at the Time of Admission. Cureus 13(10): e18768. doi:10.7759/cureus.18768.

9. Mahdavi M, Choubdar H, Zabeh E, Rieder M, Safavi-Naeini S, et al. (2021) A machine learning based exploration of COVID-19 mortality risk. PLOS ONE 16(7): e0252384. https://doi.org/10.1371/journal.pone.0252384.

10. Thomas Linden, Frank Hanses, Daniel Domingo-Fernández, Lauren Nicole DeLong, Alpha Tom Kodamullil, Jochen Schneider, Maria J.G.T. Vehreschild, Julia Lanznaster, Maria Madeleine Ruethrich, Stefan Borgmann, Martin Hower, Kai Wille, Torsten Feldt, Siegbert Rieg, Bernd Hertenstein, Christoph Wyen, Christoph Roemmele, Jörg Janne Vehreschild, Carolin E.M. Jakob, Melanie Stecher, Maria Kuzikov, Andrea Zaliani, Holger Fröhlich, Machine Learning Based Prediction of COVID-19 Mortality Suggests Repositioning of Anticancer Drug for Treating Severe Cases, Artificial Intelligence in the Life Sciences, Volume 1, 2021, 100020, ISSN 2667-3185, https://doi.org/10.1016/j.ailsci.2021.100020

11. Nooshin Ayoobi, Danial Sharifrazi, Roohallah Alizadehsani, Afshin Shoeibi, Juan M. Gorriz, Hossein Moosaei, Abbas Khosravi, Saeid Nahavandi, Abdoulmohammad Gholamzadeh Chofreh, Feybi Ariani Goni, Jiří Jaromír Klemeš, Amir Mosavi, Time series forecasting of new cases and new deaths rate for COVID-19 using deep learning methods, Results in Physics, Volume 27, 2021, 104495, ISSN 2211-3797, https://doi.org/10.1016/j.rinp.2021.104495.

12. Laatifi, M., Douzi, S., Bouklouz, A. et al. Machine learning approaches in Covid-19 severity risk prediction in Morocco. J Big Data 9, 5 (2022). https://doi.org/10.1186/s40537-021-00557-0.

13. Peter D Sottile, David Albers, Peter E DeWitt, Seth Russell, J N Stroh, David P Kao, Bonnie Adrian, Matthew E Levine, Ryan Mooney, Lenny Larchick, Jean S Kutner, Matthew K Wynia, Jeffrey J Glasheen, Tellen D Bennett, Real-time electronic health record mortality prediction during the COVID-19 pandemic: a prospective cohort study, Journal of the American Medical Informatics Association, Volume 28, Issue 11, November 2021, Pages 2354–2365, https://doi.org/10.1093/jamia/ocab100.

14. Ottenhoff MC, Ramos LA, Potters W on behalf of The Dutch COVID-PREDICT research group, et alPredicting mortality of individual patients with COVID-19: a multicentre Dutch cohortBMJ Open 2021;11:e047347. doi: 10.1136/bmjopen-2020-047347.

15. Booth, A.L., Abels, E. & McCaffrey, P. Development of a prognostic model for mortality in COVID-19 infection using machine learning. Mod Pathol 34, 522–531 (2021). https://doi.org/10.1038/s41379-020-00700-x.

16. Zhao Z, Chen A, Hou W, Graham JM, Li H, et al. (2020) Prediction model and risk scores of ICU admission and mortality in COVID-19. PLOS ONE 15(7): e0236618. https://doi.org/10.1371/journal.pone.0236618.

17. Yan, L., Zhang, HT., Goncalves, J. et al. An interpretable mortality prediction model for COVID-19 patients. Nat Mach Intell 2, 283–288 (2020). https://doi.org/10.1038/s42256-020-0180-7.

18. Jee, Y., Kim, YJ., Oh, J. et al. A COVID-19 mortality prediction model for Korean patients using nationwide Korean disease control and prevention agency database. Sci Rep 12, 3311 (2022). https://doi.org/10.1038/s41598-022-07051-4.

19. Bertsimas D, Lukin G, Mingardi L, Nohadani O, Orfanoudaki A, et al. (2020) COVID-19 mortality risk assessment: An international multi-center study. PLOS ONE 15(12): e0243262. https://doi.org/10.1371/journal.pone.0243262.

20. Alle S, Kanakan A, Siddiqui S, Garg A, Karthikeyan A, et al. (2022) COVID-19 Risk Stratification and Mortality Prediction in Hospitalized Indian Patients: Harnessing clinical data for public health benefits. PLOS ONE 17(3): e0264785. https://doi.org/10.1371/journal.pone.0264785.

21. Arezoo Haratian, Hadi Fazelinia, Zeinab Maleki, Pouria Ramazi, Hao Wang, Mark A. Lewis, Russell Greiner, David Wishart, Dataset of COVID-19 outbreak and potential predictive features in the USA, Data in Brief, Volume 38, 2021, 107360, ISSN 2352-3409, https://doi.org/10.1016/j.dib.2021.107360.

22. Boruta Feature Selection (an Example in Python) | by Aaron Lee | Towards Data Science. URL: https://towardsdatascience.com/simple-example-using-boruta-feature-selection-in-python-8b96925d5d7a (дата звернення: 29.05.2022).

23. Recursive Feature Elimination (RFE) for Feature Selection in Python. URL: https://machinelearningmastery.com/rfe-feature-selection-in-python/ (дата звернення: 29.05.2022).

24. Recursive feature elimination with cross-validation — scikit-learn 1.1.1 documentation. URL: https://scikit-learn.org/stable/auto_examples/feature_selection/plot_rfe_with_cross_validation.html (дата звернення: 29.05.2022).

25. ML | Linear Regression - GeeksforGeeks. URL: https://www.geeksforgeeks.org/ml-linear-regression/ (дата звернення: 29.05.2022).

26. Logistic Regression — Detailed Overview | by Saishruthi Swaminathan | Towards Data Science. URL: https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc. (дата звернення: 29.05.2022).

27. Ridge Regression Definition & Examples | What is Ridge Regression? URL: https://www.mygreatlearning.com/blog/what-is-ridge-regression/ . (дата звернення: 29.05.2022).

28. How to Develop Elastic Net Regression Models in Python. URL: https://machinelearningmastery.com/elastic-net-regression-in-python/ (дата звернення: 29.05.2022).

29. Stochastic Gradient Descent — Clearly Explained !! | by Aishwarya V Srinivasan | Towards Data Science. URL: https://towardsdatascience.com/stochastic-gradient-descent-clearly-explained-53d239905d31 (дата звернення: 29.05.2022).

30. Machine Learning Basics with the K-Nearest Neighbors Algorithm | by Onel Harrison | Towards Data Science. URL: https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761 (дата звернення: 29.05.2022).

31. Support Vector Machine Algorithm - GeeksforGeeks. URL: https://www.geeksforgeeks.org/support-vector-machine-algorithm/ (дата звернення: 29.05.2022).

32. Decision Tree - GeeksforGeeks. URL: https://www.geeksforgeeks.org/decision-tree/ (дата звернення: 29.05.2022).

33. Bagging algorithms in Python | Engineering Education (EngEd) Program | Section. URL: https://www.section.io/engineering-education/implementing-bagging-algorithms-in-python/ (дата звернення: 29.05.2022).

34. Understanding Random Forest. How the Algorithm Works and Why it Is… | by Tony Yiu | Towards Data Science. URL: https://towardsdatascience.com/understanding-random-forest-58381e0602d2 (дата звернення: 29.05.2022).

35. A Quick Guide to Boosting in ML. This post will guide you through an… | by Jocelyn D'Souza | GreyAtom | Medium. URL: https://medium.com/greyatom/a-quick-guide-to-boosting-in-ml-acf7c1585cb5 (дата звернення: 29.05.2022).

36. A Gentle Introduction to the Gradient Boosting Algorithm for Machine Learning. URL: https://machinelearningmastery.com/gentle-introduction-gradient-boosting-algorithm-machine-learning/ (дата звернення: 29.05.2022).

37. Understanding AdaBoost. Anyone starting to learn Boosting… | by Akash Desarda | Towards Data Science. URL: https://towardsdatascience.com/understanding-adaboost-2f94f22d5bfe (дата звернення: 29.05.2022).

38. A Gentle Introduction to XGBoost for Applied Machine Learning. URL: https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/ (дата звернення: 29.05.2022).

39. Stacking in Machine Learning - GeeksforGeeks. URL: https://www.geeksforgeeks.org/stacking-in-machine-learning/ (дата звернення: 29.05.2022).

40. Multilayer Perceptron Explained with a Real-Life Example and Python Code: Sentiment Analysis | by Carolina Bento | Towards Data Science. URL: https://towardsdatascience.com/multilayer-perceptron-explained-with-a-real-life-example-and-python-code-sentiment-analysis-cb408ee93141 (дата звернення: 29.05.2022).

41. A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way | by Sumit Saha | Towards Data Science. URL: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53 (дата звернення: 29.05.2022).

42. Simple Weighted Average Ensemble | Machine Learning | by Jinhang Jiang | Analytics Vidhya | Medium. URL: https://medium.com/analytics-vidhya/simple-weighted-average-ensemble-machine-learning-777824852426 (дата звернення: 29.05.2022).