International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Impact of Data Pre-Processing on Covid-19 Diagnosis Using Machine Learning Algorithms

Dina A. Salem[1], Esraa.M.Hashim[2]

Submitted: 15/10/2022 **Revised**: 14/12/2022 **Accepted**: 03/01/2023

Abstract: Human coronaviruses present a significant disease burden. Identifying infected coronavirus patients using artificial intelligence draws researchers' attention all over the world. Blood test is a striking element that can significantly contribute to provide a reliable, accurate, and quick automated detection tool of covid-19 diagnosis. Medical datasets are known to be associated with different data problems mainly, unbalancing, missing values, and amplitude variations. Performance of classifiers cannot be correctly assessed without handling those problems. For this, the paper at hand proposed multiple solutions that merge several data pre-processing techniques with three dominant classifiers namely Deep Learning (DL), K-Nearest Neighbors (KNN), and Support Vector Machines (SVM). After detailed dataset treatment, all three classifiers achieved good performance according to the gold standard with SVM scoring the highest accuracy and sensitivity of 86% and 95% respectively. This study showed the clinical soundness and feasibility of utilizing blood test analysis and machine learning as a replacement to rRT-PCR for detecting COVID-19-positive cases.

Keywords*: COVID-19, Deep Learning, Machine Learning, K-Nearest Neighbours, Support Vector Machine*

1. Introduction

The severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) of the genus Beta coronavirus is the main cause of COVID-19, a contagious condition [1]. It has a linkage to zoonotic viruses, which might spread the infection to certain mammalian or bird species [1]. Within a few months of its initial appearance in late 2019 in Wuhan, China [2], this deadly virus has extended rapidly around the world. The COVID-19 outbreak has had a severe overall result, damaging the healthcare system, the business, education, and social sectors [3].

Although SARS, MERS, and COVID-19 clinical symptoms appear to be similar, differential diagnoses have been noted to date [4]. Epidemiological data, clinical symptoms, Computed Tomography (CT) or positive chest X-ray, and positive pathogenic assessment based on blood tests are some of the criteria used to determine the diagnosis of COVID-19 and are also important in improving reliable evaluation [5]. Understanding the specific variations of the COVID-19 prognosis may be aided by distinguishing between COVID-19 individuals with severe illness and those with mild symptoms. The knowledge might make it easier to establish an early diagnosis of COVID-19 severity [6].

Reverse transcription polymerase chain reaction (RT-PCR) was initially used as the COVID-19 diagnosis toolkit, but later, several antigens and antibody testing tool kits received widespread approval. However, rapid testing kits have lesser accuracy and are more expensive than RT-PCR tests, which take several hours to get findings [7]. Additionally, this test may not detect patients with COVID-19 who are fully symptomatic [8]. Therefore, more accessible, alternative, and reliable solutions are required.

Along with radiological images, machine learning (ML) approaches have been applied in infectious disease diagnosis and medical imaging to develop alternate methods of quickly and accurately detecting COVID-19. ML algorithms generally create a numerical algorithm based on the data available, known as "training data", to make predictions without being specifically designed to do so. ML techniques are generally utilized for applications where it is challenging or impractical to implementing or creating models using common programming algorithms [9].

Deep learning (DL) is a subfield of ML that addresses artificial neural network (ANN) techniques that simulate the function and structure of the human mind [10]. Recently, DL techniques have attracted significant attention as a result of their exceptional capability to learn features and structural patterns from image databases and subsequently make estimations on unobserved data. Feature learning is a key component of DL which aims to extract feature automatically from raw data and aims to learn feature hierarchies from grater levels of the hierarchy produced by the assembly of lower-level features [11]. Several ML-based techniques have recently been used with routine blood testing results to improve on shortcomings of RT-PCR tests. Table 1 lists some of the research articles for COVID-19 diagnosis using ML and DL classifiers stating classifier associated with each.

1Computer Engineering Department, MUST University, Giza, Egypt, dina.almahdy@must.edu.eg
2Biomedical Engineering Department, MUST University, Giza, Egypt, esraa.shebib@must.edu.eg

Table 1: List of Results of Similar Research Studies on COVID-19 with the Employed Classifiers

Ref.	Method	Data Types	No. of Patients	Results
11	Classification (Multilayer perceptron)	Clinical	5644	Accuracy: 93.13%, Recall:93%, Precision:93%
12	SVM	laboratory features	336	Accuracy: 77.5%, Specificity: 78.4%
13	XGBoost classifier	Blood samples	485	Accuracy: 90%
14	Random forest	Clinical	49	Accuracy: 95.95% Specificity: 96.95%
15	SVM	Blood samples	18	Accuracy: 93% Specificity: 93.33%
16	XGBoost classifier	Laboratory features	5644	Accuracy: 75% Specificity: 49%
17	KNN SVM	Blood samples	279	Accuracy: 66% Accuracy: 69%

In [18], COVID-19 pathogen detection based on data from the blood samples of patients has been presented. The results of COVID-19 patients have been predicted using a variety of supervised ML models, including neural networks, SVM, RF, KNN, and variants of gradient boosting machines. It has been demonstrated that this method could be utilized to identify contaminated patients with COVID-19 who are at high mortality risk, allowing hospitals to use their resources more effectively for COVID-19 therapy [19]. Due to their properties, preprocessing is extremely significant for medical datasets. No preprocessing technique is best for all datasets because they are all distinct. Without trial and comparison, it is impossible to select the ideal set of preprocessing techniques for a specific dataset [20]. For this reason, it is thought that treating incomplete values adequately during preprocessing is a step that is needed to get a high-performance classification method. In this study, we will be dealing with three main key principles of data preprocessing methods: data balancing, data imputation, and data normalization. To construct different datasets extracted from the original dataset based on the proposed processing steps applied, three classification approaches are considered to produce the prediction models after utilizing the preprocessing procedures.

2. Material and Methods

The purpose of this study is to create a prediction method based on ML and DL approaches that will be able to predict if a suspicious COVID-19 patient blood test sample is positive or negative by putting forward three distinct approaches for filling in the missing data. The dataset and data analysis process utilized for model training is both covered in the remaining paragraphs of this section.

a. Dataset

The IRCCS [17] provided the dataset that was used in this investigation, which was made up of 279 cases that were randomly selected among patients admitted to that hospital between the end of February and the middle of March 2020. The patient's age, gender, and results from regular blood tests were included in each case, along with the COVID-19 RT-PCR test result from nasopharyngeal swabs. The parameters collected by the blood test are classified into numerical and categorial (gender and swab).

b. Methods

Figure 1 present the workflow of the proposed models. The study at hand started with finding a suitable dataset, then preparing it to be embedded in the proposed model by classifying the data into positive (Class 1), and negative (Class 0). Dataset is then preprocessed using data balancing, data imputation, and data normalization techniques. Preprocessing using one, two or the three of the listed methods opens into getting ten different datasets. The next step was to employ and tune different predictive models using ML and DL techniques and evaluate each.

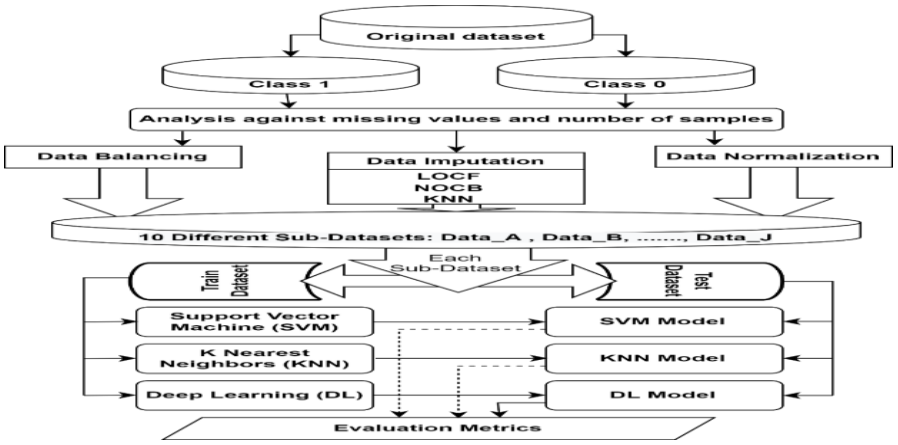

Fig. 1: Workflow of the proposed methodology.

c. Data Pre-processing

Before evaluating the data, missing values must be dealt with because ignoring or removing them could lead to biased or inaccurate analysis. Some methods, such as deleting instances and replacing possible or approximated values (a method known as imputation) [21–22], can be used to address missing values [23].

The presence of missing values is one of the main issues with medical datasets, particularly those involving blood testing. The used dataset's missing values have been carefully analyzed in the following sections. The first dataset is split into 2 sub-datasets; dataset 1 which includes all positive patients and dataset 0 which contains all negative patients. Following that, number of missing values in each of the 15 attributes is calculated in each sub-dataset and reported in table 2.

Table 2: Number and Percentage of Missing Features in each Dataset

Features	Dataset 1		Dataset 0	
	No of missing	% Of Missing	No of missing	% Of Missing
Gender	0	0	0	0
Age	0	0	0	0
Leukocyte Count	2	1	0	0
Platelets	2	1	0	0
Neutrophils	30	17	40	39
Lymphocytes	30	17	40	39
Monocytes	30	17	40	39
Eosinophils	30	17	40	39
Basophils	31	17.5	40	39
C-reactive Protein (CRP)	5	3	1	1
Transaminases (AST)	2	1	0	0
Transaminases (ALT)	13	7	0	0
alkaline phosphatase (ALP)	88	50	60	59
Gamma Glutamil Transferasi (GGT)	91	51	52	51
Lactate dehydrogenase (LDH)	46	26	39	38

Each of the ALP and GGT features are found to contain samples with missing values that constitute more than 50% of the total number of samples. Imputing such amount of missing values will not reflect the actual nature of the data. Consequently, the dataset should then be cleared of those two features. By referring to the original dataset it is found that it contains 177 samples of positive class and 102 samples of negative class with a ratio of 63.5% to 36.5% respectively. This indicates that the dataset is imbalanced and biased to the positive class. Working with such imbalanced data leads to a big difference between sensitivity and specificity performance measurements. And the outcomes of the [17] support this conclusion. Total number of missing features in each sample is counted after removing ALP and GGT features from datasets. Table 3 contains the number and percentage of samples with missing features = 0 to 13 for each dataset.

Table 3: Number and Percentage of Samples with Missing Features in each Dataset

No of missing Features	Dataset 1		Dataset 0	
	No of samples	% of samples	No of samples	% of samples
0	98	55	40	39
1	43	25	21	21
2	6	3	1	1
3	1	0.6	0	0
4	0	0	0	0
5	20	11	23	23
6	6	3	17	17
7	1	0.6	0	0
8	0	0	0	0
9	1	0.6	0	0
10	0	0	0	0
11	1	0.6	0	0
12	0	0	0	0
13	0	0	0	0

From the previous table, it is seen that dataset1 contains 98 samples without any missing values. Then, a new preprocessed dataset that only contains those 98 samples from the positive class will be created as a suggested method to solve the

problem of unbalanced data and in the same way, decrease the number of missing values to make the data more accurate and reliable. With 98 samples from the positive class and 102 samples from the negative class, the new preprocessed dataset is now balanced. While preprocessed dataset1 does not have any missing values, dataset0 contains many. Imputation of data is thus required for dataset 0.

Handling missing values is crucial and can be done by many conventional statistical and ML imputation algorithms including ensemble-based, regression, and k nearest neighbor [24]. Three imputation methods are applied to the used dataset to solve missing values problem: Next Observation Carried Backward (NOCB), Last Observation Carried Forward (LOCF), and Imputation by KNN. First, one-hot encoding was used to convert the categorical feature Gender into two binary features that the classifiers can handle. To study the impact of different data preprocessing methods applied, 10 different datasets are composed according to the proposed method of data balancing, normalization, and imputation as shown in table 4.

Table 4: Details of the 10 Datasets Constructed Using Pre-processing Methods

Data	*Data specifications*
Data A	Unbalanced, non-normalized dataset that contains all samples of dataset1 and all samples of dataset0 after data imputation of each dataset separately using NOCB.
Data B	Data A, after being normalized using min-max criteria.
Data C	Balanced, non-normalized dataset contains samples of dataset1 that has no missing values and all samples of dataset0 after data imputation using NOCB.
Data D	Data C, after being normalized using min-max criteria.
Data E	Balanced, non-normalized dataset contains samples of dataset1 that have no missing values and all samples of dataset0 after data imputation using LOCF.
Data F	Data E,after being normalized using min-max criteria.
Data G	unbalanced, non-normalized dataset that contains all samples of dataset1 and all samples of dataset0 after data imputation of each dataset separately using K-Nearest neighbors imputation method with K=3.
Data H	Data G, after being normalized using min-max criteria.
Data I	balanced, non-normalized dataset contains samples of dataset1 that have no missing values and all samples of dataset0 after data imputation using K-Nearest neighbors imputation method with K=3.
Data J	Data I, after being normalized using min-max criteria.

d. Training model and classifiers selection

1-K-Nearest Neighbor (KNN)

ML algorithms are successfully used in a wide range of applications across different sectors. The non-parametric algorithm is called KNN. Based on the problem or dataset that has been provided, learning and prediction analysis is carried out. Without making any assumptions about the dataset, the KNN classification method's projection is only dependent on values for neighboring data [25]. 'K' represents the quantity of data values from the nearest neighbour. The KNN algorithm is chosen how to classify the provided dataset based on "K [26]. The training dataset is immediately classified by the KNN model. It implies that the identification of a new instance is done by finding the similar 'K' neighbor instances in the whole training set and identifying based on the class of highest instances. According to equation 1, the Euclidean distance is the square root of the total of the squared differences between the new instance (xi) and the present instance (yj) [26].

$$\text{Euclideani, j} = \sqrt{\sum_{k=1}^{n} \left(x_{ik} - y_{jk}\right)^2} \qquad (1)$$

2-Suport vector machine (SVM)

The goal behind the SVM is to create a hyperplane between the two classes. It can handle categorical and continuous data. The maximum margin hyperplane, which has the longest distance to the closest points from the two classes, is then discovered using an optimization solution [27], The SVM aims at maximizing margins between different classes by separating hyperplanes as best it can. The hyperplane is a data instance of the provided dataset utilized by the support vectors. The margin is the distance at which the support vector and the hyperplane can divide most. [28].

3-Deep learning

The expression of "Deep Learning" is commonly utilized when studying multilayer Artificial Neural Networks (ANN). The multilayer perceptron (MLP) is the first type of multi-layer ANN design. Three layers form the simplest type: one hidden layer, one input layer, and one output layer [29]. The multilayered or hierarchical network structure is what gives neural networks their predictive value. To create higher-order features, the data structure can select (learn to represent) features at various resolutions or scales. For example, from lines to collections of lines to shapes [29]. The output layer is completely connected to the hidden layer, and every unit in the hidden layer relates to every unit in the input layer. The number of layers represents just one of the "hyper-parameters" of a Deep Neural network (DNN). The complexity of a network is given also by the number of neurons, connections, and weight. Every individual weight represents a parameter that needs to be learned.

This study designs a DL model to increase the accuracy of reported instance and to accurately predict the infection from blood samples. DNN was implemented by Python language using "Keras" [29], trying to optimize the classification results in terms of accuracy. First, we load the data, then define the model using a fully connected network structure with five different types of layers with output shapes. we initialize the network weights between zero and 0.05 because that is the default uniform weight settings in Keras. To assess a number of weights we must specify the loss function, which is logarithmic loss, this loss is for binary classification problems, it self-tunes automatically and performs well with a variety of problems. The process for the DL technique is demonstrated in Figure 2.

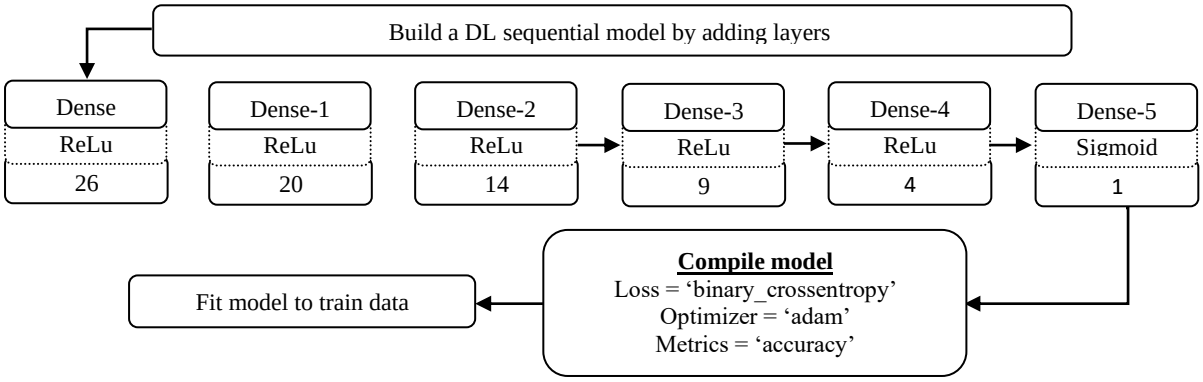

Fig. 2: Details of the applied deep learning model.

e. Evaluation matrices

Several evaluation parameters are used to evaluate the performance of the classification. The most common metrics include accuracy (ACC), precision (PREC), sensitivity (recall) (REC), specificity, and f-score (F1) [24]. All the listed five metrics are recorded for the three classifiers on each dataset.

3. Results

The current global outbreak of COVID-19 has increased the requirement for efficient and accurate automatic detection technologies. To study the impact of different data preprocessing methods applied 10 different datasets are composed according to the proposed methods by applying KNN, SVM, and DL algorithms to each dataset. The performance of the three classification algorithms is tested and evaluated by calculating the evaluation parameters. As shown in Figures 3,4,5 for 10 tested constructed datasets to KNN, SVM, and DL techniques respectively. Figure 6 represents the accuracy of 10 trained constructed datasets to the three classification techniques.

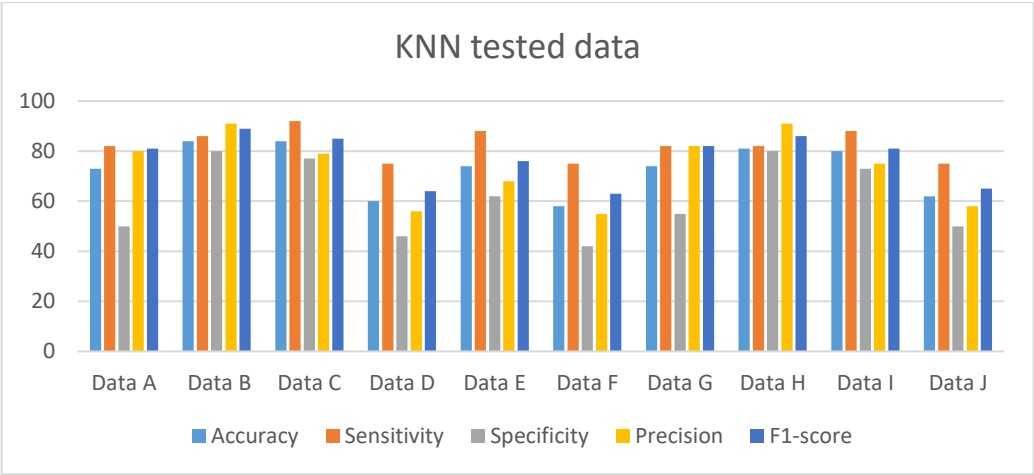

Fig. 3: Performance of KNN (test dataset) on the 10 constructed datasets.

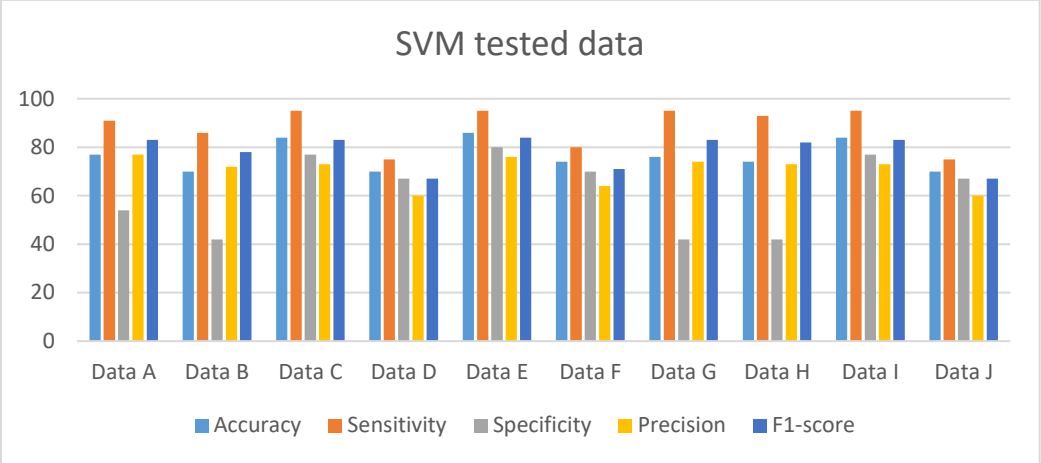

Fig. 4: Performance of SVN (test dataset) on the 10 constructed datasets.

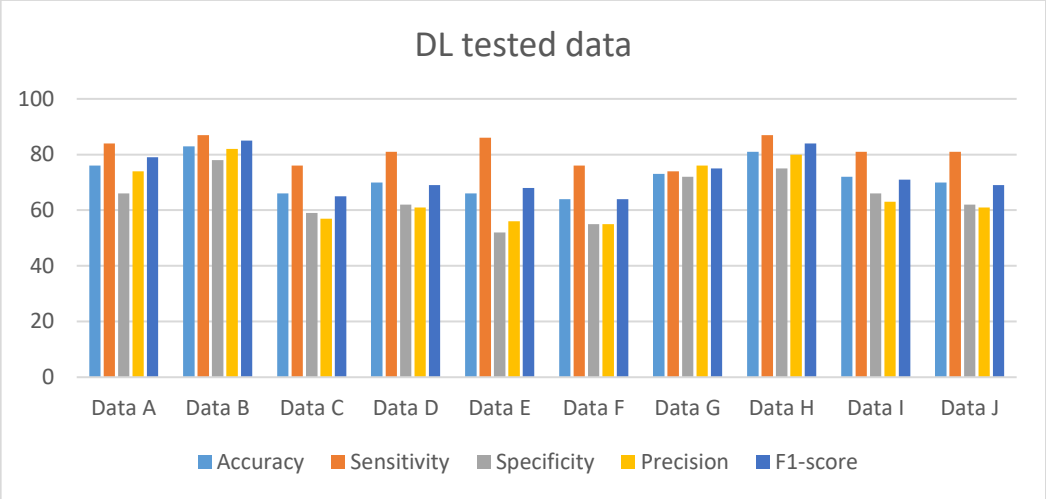

Fig. 5: Performance of Deep Learning (test dataset) on the 10 constructed datasets.

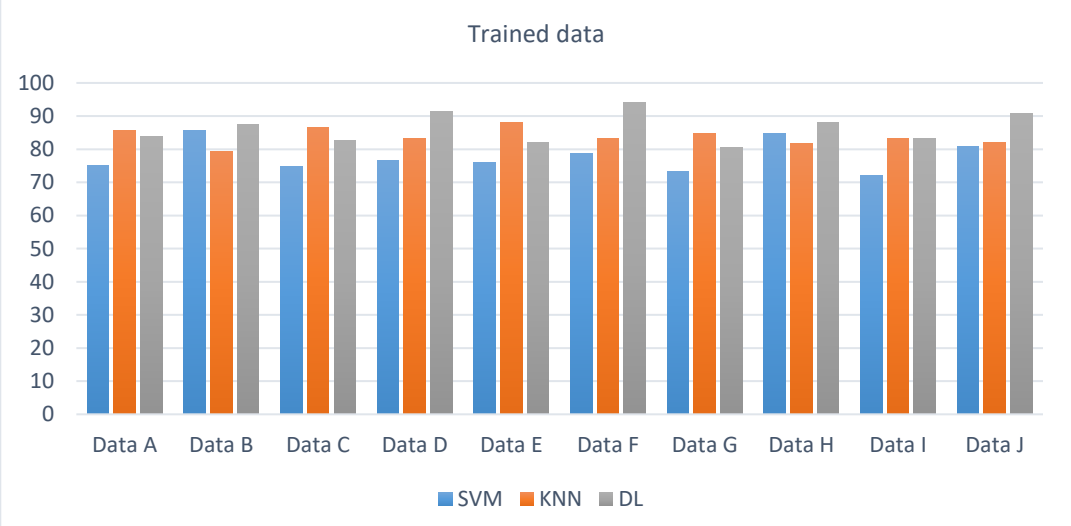

Fig. 6: Train accuracy of the three classifiers on each dataset.

It can be seen from Figure 3,4,5 that our proposed models achieved promising results compared with other related studies as shown in Table 1. The KNN classifier has obtained the best sensitivity of 92%, the accuracy of 84%, and the specificity of 77% for data C, which was balanced, non-normalized, and imputed. Also, it obtained the best sensitivity of 88%, the accuracy of 80%, and the specificity of 73% for data I, which was balanced, non-normalized, and imputed. The DL classifier has obtained, a sensitivity of 87%, an accuracy of 83%, and a specificity of 78% for data B, which is normalized data. Also, it obtained a sensitivity of 87%, an accuracy of 81%, and a specificity of 75% for data H, which was normalized.

The best performing model is SVM, the SVM classifier has obtained the best sensitivity of 95%, the accuracy of 84%, and the specificity of 77% for data C, which was balanced, non-normalized, and imputed. Also, it obtained the best sensitivity

of 95%, the accuracy of 86%, and specificity of 80% for data E, which was balanced, non-normalized, and imputed, so close to the gold standard. For the trained data in figure 6, the percentage of accuracy is similar for the three techniques.

The impact of various combinations of data preprocessing methods was examined for each classifier and results show that for the KNN technique, the best results were for balanced, non-normalized, and imputed by NOCB and imputed by KNN method. For the DL technique, the best results were for normalized data and balanced, non-normalized, and imputed by the KNN method. But as for the results of the SVM classifier which give the best accuracy, sensitivity, and precision, the best preprocessing type of data was for balanced, non-normalized and imputed by NOCB although for data that normalized by min-max criteria, and data balanced, non-normalized, and imputed by the KNN method.

4. Conclusion

A sample blood test has lately emerged as an essential tool to aid in the detection of false-positive/negative rRT-PCR tests, duo to the fact that it is an affordable and convenient method to identify potential COVID-19 patients. Given its potential as a tool for supported decision-making, a machine learning model that has been trained on a complete and correct dataset could appear to be a significant resource for primary care doctors. ML and DL models being the core of data analysis and information extraction from data needs appropriate and precise data pre-processing to fulfill their remarkable classification role.

In this paper, the goal was to develop a reliable prediction model to determine whether a COVID-19 sample is positive or negative. Available datasets encounter major pitfalls that is known to negatively affect the performance of ML algorithms. To solve this problem, dataset is pre-processed using min-max criteria for data normalization, data under-sampling for data balancing along with three different techniques for data imputation (Next observation carried backward, Last observation carried forward and KNN imputation). Ten different datasets are constructed and introduced to three classifiers (DL, SVM and KNN) and different evaluation metrics are recorded after accurately tuning classifiers' potential parameters. The results have shown that SVM exhibited very high sensitivity (95%) with an accuracy of (~86%) on balanced datasets imputed using LOCF which surpassed other related studies used the same dataset. DL achieves its best training accuracy (~ 91%) on balanced normalized datasets. Highest precision resulted from KNN classifier (91%) was recorded on dataset that is normalized and imputed by KNN-imputation.

Data preprocessing directly affects the performance of the learner. Handling missing data, data normalization and data balancing are keys to preprocessing activities, so the impact of various combinations of data preprocessing methods was examined. The normalization techniques and additional widely used approaches for addressing incomplete values were taken into consideration. From the reported evaluation, It's interesting to note that the effects of the procedural preparation methods differ between various classification techniques. More particularly, the effect of the data preprocessing methods was more obvious when the SVM classifier was used to impute the classification models. The limitations of this study: The most noticeable is the DL algorithm's use of a small number of blood samples.

References

[1] S. Yang, L. Jiang, Z. Cao, L. Wang, J. Cao, R. Feng, Z. Zhang, X. Xue, Y. Shi, and F. Shan, "Deep learning for detecting coronavirus disease (COVID-19) on high-resolution computed tomography: a pilot study," *Ann Transl. Med.*, vol. 8(7):450, Apr. 2020.

[2] E. M. Hashim, and M. S. Mabrouk, "Protein-ligand In-silico molecular docking model for discovering potential drugs of covid-19," *Advanced Engineering Trends*, vol. 42(1), Jan. 2022.

[3] L. Wynants et al., "Prediction models for diagnosis and prognosis of COVID-19: systematic review and critical appraisal," *BMJ*, vol. 369, Mar. 2020.

[4] N. Chen, M. Zhou, X. Dong, J. Qu, F. Gong, Y. Han, Y. Qiu, J. Wang, Y. Liu, Y. Wei, J. Xia, T. Yu, X. Zhang, and L. Zhang, "Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in Wuhan, China: a descriptive study," *Lancet*, vol. 395, pp. 507–513, Feb. 2020.

[5] A. M. Karim, H. Kaya, V. Alcan, and B. Sen, "New optimized deep learning application for COVID-19 detection in chest X-ray images," *Symmetry*, vol. 14(1003), May2022.

[6] Y. Haochen, Z. Nan, Z. Ruochi, D. Meiyu, X. Tianqi, P. Jiahui, P. Ejun, H. Juanjuan, Z. Yingli, X. Xiaoming, X. Hong, Z. Fengfeng, and W. Guoqing, "Severity detection for the coronavirus disease 2019 (COVID-19) patients using a machine learning model based on the blood and urine tests," *Frontiers in Cell and Developmental Biology*, vol. 8, July 2020.

[7] M. Ahishali, A. Degerli, M. Yamac, S. Kiranyaz, M. E. H. Chowdhury, K. Hameed, T. Hamid, R. Mazhar, and M. Gabbouj, "Warning methodologies for COVID-19 using chest x-ray images," *IEEE Access*, vol. 9, pp. 41052–41065, Mar. 2021.

[8] D. Li, D. Wang, J. Dong, N. Wang, H. Huang, H. Xu, and C. Xia, "False-Negative results of real-time reverse-transcriptase polymerase chain reaction for severe acute respiratory syndrome coronavirus 2: role of deep-learning-based CT diagnosis and insights from two Cases," *Korean Journal of Radiology*, vol. 21(4), pp. 505–508, Apr. 2020.

[9] P. Chatterjee, M. Biswas, and A. K. Das, "Specialized covid-19 detection techniques with machine learning," *J. Phys.: Conf. Ser*, vol. 1797(1), pp. 012–033, Feb. 2021.

[10] L. Deng, "A tutorial survey of architectures, algorithms, and applications for deep learning," *APSIPA Transactions on Signal and Information Processing*, vol. 3, Jan. 2014.

[11] M. R. H. Mondal, S. Bharati, P. Podder, and P. Podder, "Data analytics for novel coronavirus disease," *Informatics in Medicine Unlocked*, vol. 20, June 2020.

[12] L. Sun, F. Song, N. Shi, et al., "Combination of four clinical indicators predicts the severe/critical symptom of patients infected COVID-19," *Journal of Clinical Virology*, vol.128, July 2020.

[13] L. Yan, H. T. Zhang, J. Goncalves, et al., "An interpretable mortality prediction model for COVID-19

patients," *Nat Mach Intell*, vol. 2(5), ppt. 283-288, May 2020.

[14] F. Ucar, and D. Korkmaz, "COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease 2019 (COVID-19) from X-ray images," *Med Hypotheses*, vol. 140, July 2020.

[15] K. H. Abdulkareem et al., "Realizing an Effective COVID-19 Diagnosis System Based on Machine Learning and IoT in Smart Hospital Environment," *in IEEE Internet of Things Journal*, vol. 8, no. 21, pp. 15919-15928, Nov. 2021.

[16] P. Schwab, A. D. Schütte, B. Dietz, and S. Bauer, "Clinical Predictive Models for COVID-19: Systematic Study," *J. Med. Internet Res*, vol. 22(60), Oct. 2020.

[17] D. Brinati, A. Campagner, D. Ferrari, M. Locatelli, G. Banfi, and F. Cabitza, "Detection of COVID-19 Infection from Routine Blood Exams with Machine Learning: A Feasibility Study," *J Med Syst*, vol. 44(8):135, July 2020.

[18] S. Aktar, M. M. Ahamad, M. Rashed-Al-Mahfuz, A. Azad, S. Uddin, A. Kamal, et al., "Machine learning approach to predicting covid-19 disease severity based on clinical blood test data: Statistical analysis and model development," *JMIR Medical Informatics*, vol. 9 (4), Apr. 2021.

[19] A. Dairi, F. Harrou and Y. Sun, "Deep Generative Learning-Based 1-SVM Detectors for Unsupervised COVID-19 Infection Detection Using Blood Tests," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1-11, Nov. 2021.

[20] S. Almuhaideb, M. E. B Menai, "Impact of preprocessing on medical data classification," *Front. Comput. Sci*., vol.10(6), pp. 1082–1102, Oct. 2016.

[21] Z. Zhang, "Missing values in big data research: some basic skills," *Ann Transl Med*., vol. 3(21), Dec. 2015.

[22] D. L. Langkamp, A. Lehman, and S. Lemeshow, "Techniques for handling missing data in secondary analyses of large surveys," *Acad Pediatr*., vol. 10(3), pp. 205–210. May-Jun 2010.

[23] A. R. Donders, G. j. Heijden, T. Stijnen, and k. G. Moons, "Review: a gentle introduction to imputation of missing values," *J Clin Epidemiol*, vol. 59(10), pp. 1087–1091, Oct. 2006.

[24] T. Emmanuel, T. Maupong, D. Mpoeleng, et al, "A survey on missing data in machine learning," *J Big Data*, vol. 8(140), Oct. 2021.

[25] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp.175-185, Aug. 1992.

[26] O. Altay, and M. Ulas, "Prediction of the autism spectrum disorder diagnosis with linear discriminant analysis classifier and K-nearest neighbor in children," *ISDFS*, pp. 1-4, March 2018.

[27] D. A. Salem, R. A. Abul Seoud, and Y. Kadah, "Conformational B-cell epitopes classification using machine learning techniques," *Journal of Engineering and Applied Science*, Jul. 2013.

[28] B. Schoslkopf, A. Smola, "Learning with Kernels, Support Vector Machines," *MIT*, Mar. 2002.

[29] J. Brownlee, *Deep Learning with Python*, 1st Ed., 2016.