Global Scientific JOURNALS

GSJ: Volume 8, Issue 10, October 2020, Online: ISSN 2320-9186
www.globalscientificjournal.com

COVID-19 Prediction: A Comparative Analysis of Machine Learning Models

[1]Etebong Isong, *[2]Edward Udo, [3]Emmanuel Nyoho

[1]Department of Computer Science, Akwa Ibom State University, Ikot Akpaden, Nigeria

[2, 3] Department of Computer Science, University of Uyo, Uyo, Nigeria

Email: [1]etebongisong@gmail.com,*[2]edwardudo@yahoo.com, [3]javaandme4ever@gmail.com

ABSTRACT

The effect of the novel coronavirus disease across the globe is ravaging both the economy and social well-being of the people. The disease seems not to be over as World Health Organization (WHO) is reporting new cases on daily basis, indicating that the affected countries need to learn to live with the disease. It therefore behooves on researchers to develop prediction models to predict the trend of the infection. This work analyses comparatively four (4) machine learning (ML) models; One-Class Support Vector Machine (OC-SVM), Isolation Forest (I-Forest), Minimum Covariance Determinant (MCD) and Local Outlier factor (LOF) for the prediction of COVID-19 cases using dataset from kaggle.com. The dataset is unbalanced as class distribution of the training (70%) and test (30%) sets were computed to be 91% positive 9% negative cases and 96% positive 4% negative cases respectively, which makes the dataset suitable for use in one class classification, hence the choice of the predictive models used in this work. The dataset was preprocessed using One-Hot encoding to convert categorical data such as fever, cough, chills, fatigue, body pain, malaise, diarrhea, nausea, weakness, sneezing, runny-nose, breathing-difficulty, headache, and sore-throat to numerical data. Principal Component Analysis (PCA) was employed for dimension reduction. After the training, the performances of the four models were evaluated using Accuracy, Precision, Recall, and F1-Score on the training and test dataset. Finally, the F1-Score was used as the bases for best model selection (model with the highest F1-Score) since it takes into account, both the negative and positive classes. Isolation Forest (I-Forest) with F1-Score of 0.822133 for training and 0.918464 for testing turned out to be the best model among others for the prediction of COVID-19. The model is therefore capable of predicting COVID-19 cases with higher accuracy thereby helping to drastically curb the spread of COVID-19. The system was implemented using the python programming language on a Pycharm integrated development environment.

Keywords: COVID-19, Prediction, Machine Learning Models, One-Hot Encoding, One-Class Support Vector Machine, One Class Classification, Isolation Forest, Model Evaluation.

1 INTRODUCTION

The novel Coronavirus Disease (COVID-19) is an infectious disease caused by a virus, a member of the Betacoronavirus family called severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). The virus was first reported in Wuhan, Hubei Province, China on December 31, 2019 and was declared by World Health Organization (WHO) as a Public Health Emergency of International Concern on January 31, 2020 [Wang et al., 2020a]. Its rapid spread across the world also necessitated WHO on March 11, 2020 to recognize COVID-19 as a pandemic. Since then, the disease has developed into a global public health crisis [Wang et al., 2020b].

The rate of COVID-19 infection and its spread is such that it covers the whole world within a very short time when compared to other viral infections that were encountered before now [Kirbas et al., 2020]. Different studies show that COVID-19 has clinical characteristics similar to that of SARS-CoV with dominant symptoms of fever and cough while gastrointestinal symptoms are uncommon [Chen et al. 2020; Huang et al. 2020 and Li et al. 2020]. COVID-19 spread primarily through close physical contacts with an infected person, respiratory droplets or by touching contaminated surfaces [Rustam et al., 2020]. WHO has continued to identify and report new cases across the 216 affected countries and territories around the world. As at August 31, 2020, 02:00 GMT+2, there were 25,118,689 confirmed cases and 844,312 confirmed deaths [WHO 2020; JHU 2020].

In the absence of proper vaccination and curative drug to arrest the spread and curtail the number of infected people, the best option to evade the effect of the virus and save the lives of people is to adhere to government and WHO guidelines regarding washing of hands, use of facemask, methods of sneezing, public gatherings, physical distancing, travel restrictions and even lockdowns [Waqas et al., 2020; Tuli et al., 2020; Arora et al., 2020]. It is important to note that the extent of the virus infection varies from country to country and the strategies for its control also vary depending on some national conditions (Wang et al., 2020a). Implementing some of these measures impose great cost to local economies and social well being of the people thereby resulting in devastating economic crises, loses and damaging social impact as well as the compromise of strength and morals of heavily infected nations [Shinde et al., 2020].

It behooves on researchers to therefore develop prediction models to predict the trend of the infection as this is an extremely important challenge which need to be solved so that vital and significant insights regarding the likely spread and consequences of the virus can be revealed and anticipate outcomes to improve the decision making on the future course of actions [Ardabili et al., 2020].

Machine Learning (ML) and Data Science communities are striving to improve the forecast of epidemiological models and analyze the information available in social media platforms for the development of management strategies and impact assessment of policies in order to curb the spread of diseases [Tuli et al., 2020]. Over the last decade, ML has proved itself to be a prominent field of study because of its ability to solve many complex and sophisticated real-world problems [Rustam et al., 2020]; its methods for outbreak prediction modeling demonstrate a better advancement over time-series approaches and improvement over SIR and SEIR models [Ardabili et al., 2020; Rustam et al., 2020].

Various ML algorithms have been used in several forecasting application areas to give adequate guide regarding necessary course of action needed. Some of such areas include weather forecasting, stock market forecasting as well as diseases prediction. ML techniques have been employed to predict cardiovascular disease [Anderson et al., 1991], coronary artery disease [Lapuerta et al., 1995], breast cancer [Asri et al., 2016], H1N1 flu [Koike and Morimoto, 2018], dangue fever [Anno et al., 2019], influenza [Papak et al., 2019], swine fever [Liang et al., 2020].

Health care industries and clinicians worldwide have employed various ML technologies to tackle COVID-19 pandemic and addresses the challenges of the outbreak [Lalmuanawma et al., 2020]. Rustam et al., (2020) demonstrated the capability of ML models to forecast the number of future patients to be affected by COVID-19 using four standard forecasting models: linear regression (LR), least absolute shrinkage and selection operator (LASSO), support vector machine (SVM), and exponential smoothing

(ES) . Predictions made by each of the models were the number of newly infected cases, the number of deaths, and the number of recoveries for the next 10 days. The results proved that it a promising to apply ML approaches for prediction of COVOD-19 cases. Among the models used, ES performed best followed by LR and LASSO while SVM performs poorly

Iwendi et al. (2020) proposed a fine-tuned Random Forest model boosted by AdaBoost algorithm using COVID-19 patient's geographical, travel, health, and demographic data to predict the severity of the case and the possible outcome, recovery, or death.. The result revealed a positive correlation between patients' gender and deaths, and also indicated that the majority of patients are aged between 20 and 70 years.

Car et al., (2020) presented a machine learning solution, a multilayer perceptron (MLP) artificial neural network (ANN) to model the spread of COVID-19 which predicts the maximal number of people who contracted the disease per location, maximal number of people who recovered per location and maximal number of deaths per location within a given time unit.

Lieu et al. (2020) applied ML to process internet activity, news reports, health organization reports, and media activity to predict the spread of the outbreak on the providence level in China; [Pinter et al., 2020] proposed a hybrid machine learning approach of adaptive network-based fuzzy inference system (ANFIS) and multi-layered perceptron-imperialist competitive algorithm (MLP-ICA) to predict time series of infected individuals and mortality rate of COVID-19 and demonstrated its potential using data from Hungary. The validation was performed for 9 days with promising results, which confirmed the model accuracy.

Nemati et al. (2020) used survival analysis techniques including statistical analysis and ML approaches to predict survival times and to examine the effect of basic risk factors on hospital discharge time probabilities; Ardabili et al., (2020) presented a comparative analysis of Machine Learning and Soft Computing models for prediction of COVID-19 outbreak. Machine learning models (MLP and ANFIS) were considered for two data scenarios and comparison between analytical and machine learning models was done. The results of MLP and ANFIS had high generalization ability for long-term prediction; [Ribeiro et al., 2020] used support vector regression and stacking ensemble on clinical data to predict COVID-19; [Khanday et al., 2020] applied ML approaches on clinical text data for detecting COVID-19 patients.

This work seeks to develop a ML model for prediction of COVID-19 using One Class Support Vector Machine (OC-SVM), Isolation Forest (I-Forest), Minimum Covariance Determinant (MCD) and Local Outlier Factor (LOF) on COVID-19 datasets from Kaggle data repository containing data items such as case reported data, location, gender, age, symptoms, hospital, recovered, death, exposure and traveling history. The dataset is unbalanced and suitable for use in one class classification, hence the choice of our ML predictive models. The performance of these models will be evaluated using Accuracy, Precision, Recall, and F1-Score on the training and test dataset.

2 PROPOSED COVID-19 PREDICTION MODEL

This work proposes a comparative analysis of machine learning models for COVID-19 prediction. The proposed model is shown in Figure 1.

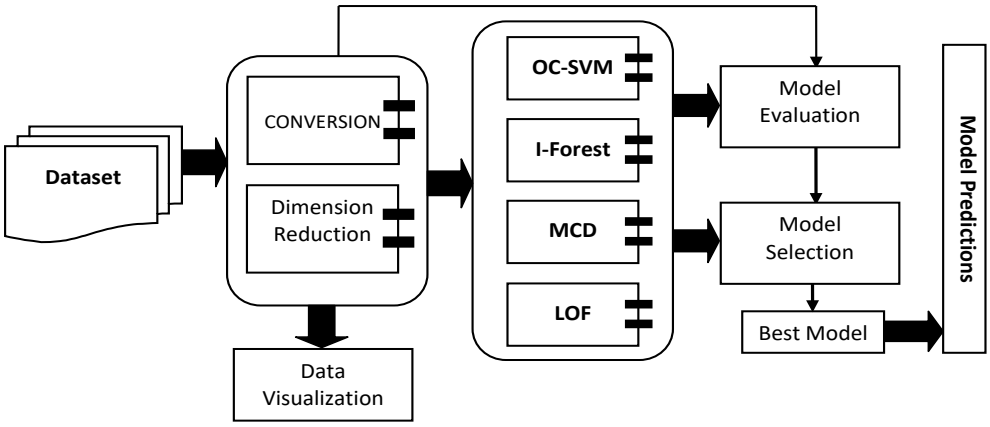


Figure 1: Proposed COVID-19 Prediction Model

The model takes COVID-19 dataset gotten from kaggle.com as input. The dataset consists of input parameters (features) and output parameters (labels). The dataset is a collection of COVID-19 symptoms and values used in training and testing the machine learning models. The symptoms are listed in Figure 2.

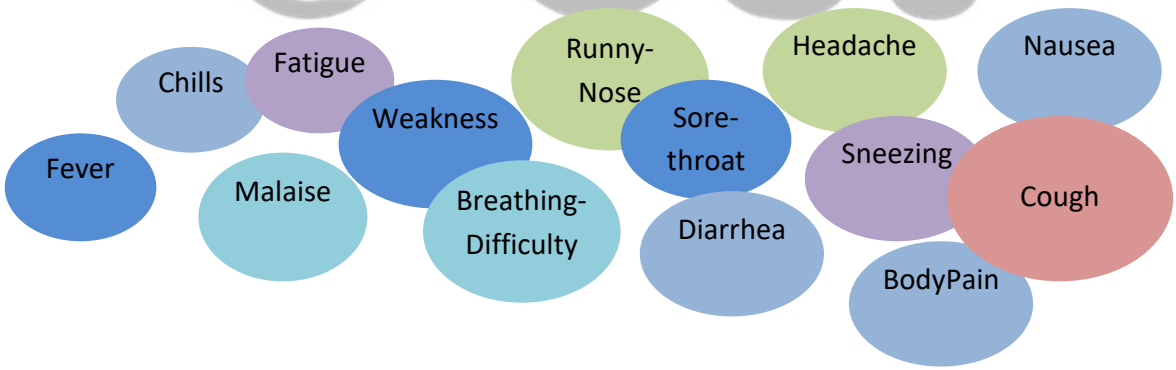


Figure 1: Symptoms (features) of COVID-19

One-Hot Encoding and Principal component Analysis (PCA) were used to clean the dataset. One-Hot Encoding converts the dataset from categorical to numeric [Machine Learning Mastery, 2020] and the result subjected to PCA for elimination of redundant features and reduction of the number of dimensions for ease of visualization and performance improvement. The resulting dataset is normalized and reduced which is used by the visualization module of our proposed framework as well as in the model training. The predictive models used in this work are OC-SVM, I-Forest, MCD and LOF as they are suitable for

prediction with an unbalanced dataset (i.e higher number of inliers (positive classes) and lower number of outliers (negative classes)) as the case is with the dataset used in this work. The models are trained to segregate COVID-19 positive cases, their performance evaluated and the result used to select the best model that will succeed as the sole model for prediction of COVID-19 cases.

3 DATA PRE-PROCESSING

A section of the dataset used in this work is depicted in Figure 3.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	
1	id	case_in_c	reporting date		summary	location	country	gender	age	symptom	If_onset_a	hosp_visit	exposure_	exposure_	visiting W	from Wuh	death	recovered	symptom	so
2	241	5	########		new confi	Aichi Pref	Japan	male	45	########	0	########	NA	########	0	1	0	0	fever	Ky
3	242	6	########		new confi	Nara Pref	Japan	male	65	########	0	########	1/8/2020	########	0	0	0	0	cough, chills, joint pain	Ky
4	243	7	########		new confi	Hokkaido	Japan	female	45	########	0	########	NA	########	1	1	0	0	fever, cough	Ky
5	244	8	########		new confi	Osaka Pre	Japan	female	45	########	0	########	########	########	0	0	0	0	fever, cough	NH
6	245	9	########		new confi	Tokyo	Japan	male	55	########	0	NA	NA	########	0	1	0	0	throat pain, fever	Ba
7	246	10	########		new confi	Mie	Japan	male	55	########	0	########	########	########	1	0	0	0	fever	NH
8	247	11	########		new confi	Japan	Japan	female	35	########	0	########	########	########	1	0	0	0	fever, cough	NH
9	248	12	########		new confi	Kyoto	Japan	female	25	########	0	########	########	########	1	0	0	0	fever	NH
10	249	13	########		new confi	Haneda	Japan	female	25	########	0	########	########	########	1	0	0	0		Ni
11	250	14	2/1/2020		new confi	Japan	Japan	male	45	NA	NA	NA	NA	########	1	0	0	0		Re
12	251	15	2/1/2020		new confi	Japan	Japan	male	45	########	0	########	NA	########	1	0	0	0	fever, cough	Re
13	252	16	2/1/2020		new confi	Japan	Japan	male	35	NA	NA	NA	NA	########	1	0	0	0		Mi
14	253	17	2/4/2020		new confi	Chiba Pre	Japan	female	35	########	0	########	NA	NA	0	1	0	0	fever, runny nose	Mi
15	254	18	2/4/2020		new confi	Chiba Pre	Japan	female	55	########	0	########	########	########	1	0	0	0	fever, cough	Mi
16	255	19	2/4/2020		new confi	Japan	Japan	male	55	########	0	########	NA	########	0	1	0	0	fever	Mi
17	256	20	2/5/2020		new confi	Chiba Pre	Japan	male	45	########	0	########	NA	NA	0	1	0	0		Mi
18	257	21	2/5/2020		new confi	Kyoto	Japan	male	25	########	0	########	NA	NA	0	0	0	0	fever	Mi
19	258	22	2/5/2020		new confi	Japan	Japan	male	55	NA	NA	NA	NA	NA	0	1	0	0		Mi
20	259	23	2/8/2020		new confi	Japan	Japan	male	25	NA	NA	NA	NA	NA	0	1	0	0		Mi
21	260	24	########		new confi	Japan	Japan	male	45	2/8/2020	0	########	NA	########	1	0	0	0	fever	Mi
22	261	25	########		new confi	Japan	Japan	male	55	2/7/2020	0	2/7/2020	NA	########	0	1	0	0	fever, cough	Mi
23	262	26	NA		NA	Japan	Japan	NA	NA	NA	NA	NA	NA	NA	0	0	0	0		Mi
24	263	27	########		new confi	Kanagawa	Japan	female	85	########	0	########	NA	NA	0	0	########	0	fatigue	Mi
25	264	28	########		new confi	Tokyo	Japan	male	75	########	0	########	NA	NA	0	0	0	0	fever	Mi

Figure 3: COVID-19 Dataset (Source: https://www.kaggle.com/datasets)

Data points with missing data values were eliminated. Categorical data in the symptom column such as fever, cough, chills, fatigue, body pain, malaise, diarrhea, nausea, weakness, sneezing, runny-nose, breathing-difficulty, headache, sore-throat became column names while their data values were gotten as 1 (existence of the variable in the category) or 0 (non-existence). Eqt (1) is the Microsoft Excel equation that searches for features in the symptom column of the dataset.

$$IF(ISNUMBER(SEARCH(\text{feature}, symptom - column - ref)), 1, 0) \quad (1)$$

A subset of the dataset after One-Hot Encoding processing is shown in Figure 4 and the frequency of the features as they contributed to COVID-19 positive case is shown in Figure 5 while the percentage of the features in the dataset is shown in Figure 6.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	fever	cough	chills	fatigue	bodypain	malaise	diarrhea	nausea	weakness	sneezing	runny-nos	breathing-difficulty	headache	sore-throat	LABEL	
2	1	1	0	0	0	0	0	0	1	0	0	0	1	0	1	
3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	
4	1	1	0	1	0	0	0	0	1	0	0	0	0	0	1	
5	1	1	0	0	0	0	0	0	1	0	0	0	0	0	1	
6	1	1	0	0	0	0	1	0	0	1	0	0	0	0	1	
7	1	1	0	0	0	0	0	0	1	0	0	0	0	0	1	
8	1	0	0	0	1	0	1	0	0	0	0	0	0	1	1	
9	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	
10	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
11	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
12	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	
13	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
14	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	
15	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	
16	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	
17	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
18	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	
19	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	
20	1	0	0	0	0	0	0	0	0	1	0	0	0	0	1	

Figure 4: Subset of Dataset after One Hot Encoding

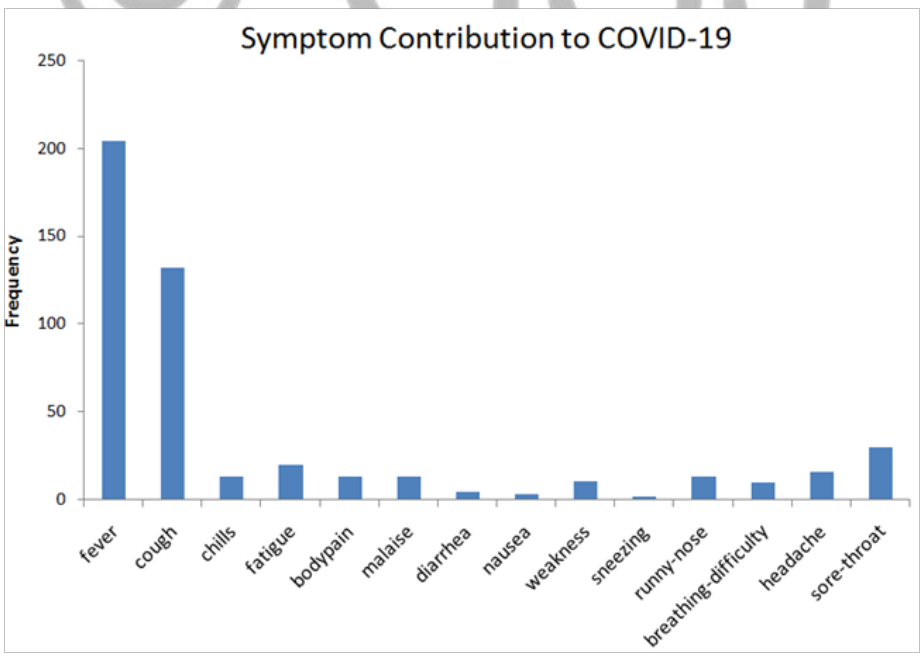


Figure 5: Frequency of Symptoms Contribution to COVID-19

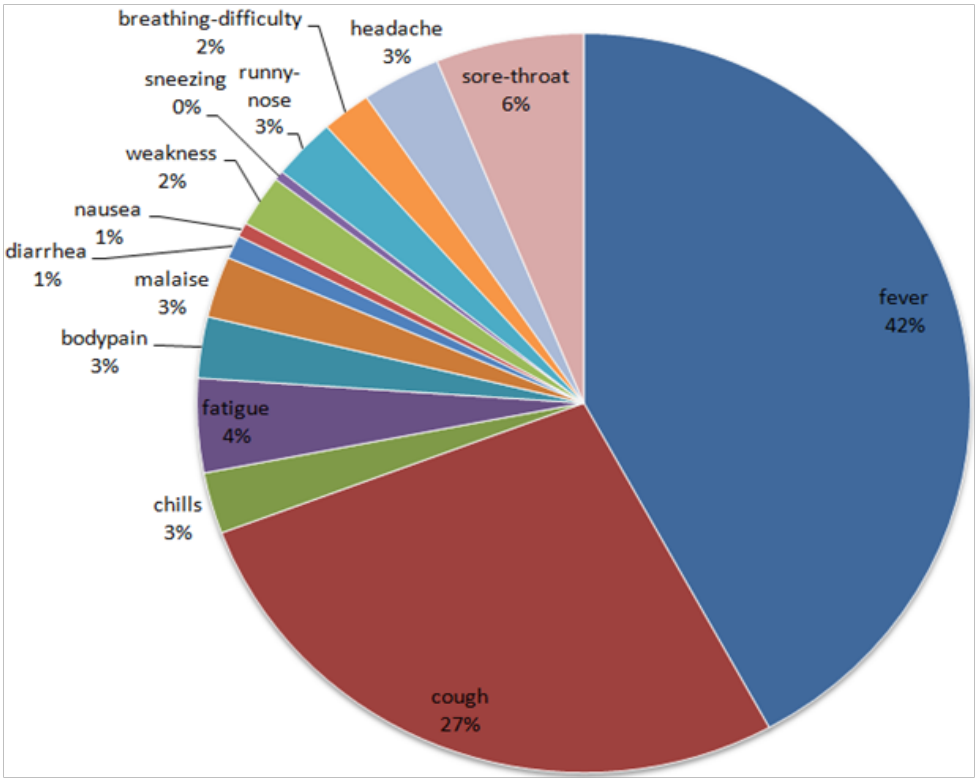


Figure 6: Percentages of feature contribution to COVID-19

Fever, cough, sore-throat and fatigue are the major symptoms exhibited by COVID-19 patients as indicated in Figure 5 and 6.

The encoded dataset (Figure 4) was subjected to PCA:

STEP 1: Calculate the covariance matrix
STEP 2: Calculate the eigenvectors and eigenvalues of the covariance matrix to identify the principal components.
STEP 3: Choose K eigenvectors that corresponds to the largest K eigenvalues to be the principal components of the dataset.
STEP 4: Project the data as $Y = Xv$; (2)

$\boldsymbol{v} = [\boldsymbol{v_1} \ldots \boldsymbol{v_K}]$ is a $\boldsymbol{d\ x\ K}$ matrix where columns $\boldsymbol{v_i}$ are the eigenvectors corresponding to the largest K eigenvalues.

Percentage of information retained by PCA and the eigenvalue plot of COVID-19 features is shown in Figures 7 and 8 respectively.

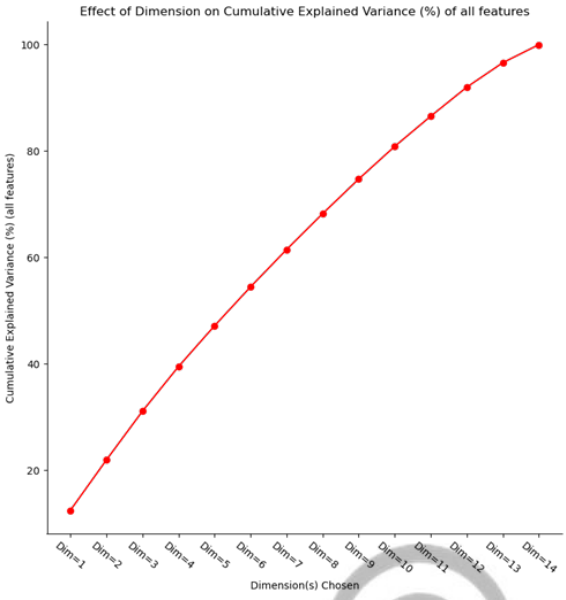


Figure 7: % of information retained by PCA

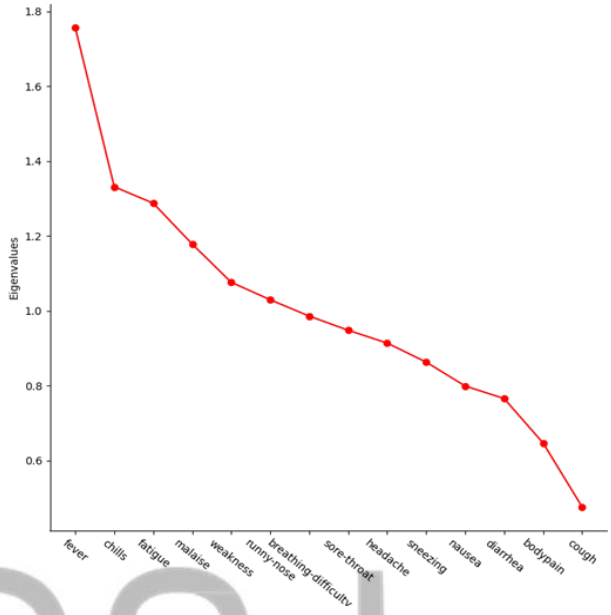


Figure 8: Eigenvalue plot of the features

After the PCA, 8 features - fever, chills, fatigue, malaise, weakness, runny-nose, breathing-difficulty and sore-throat were chosen which satisfies the condition that at least 60% percent of the original information must be retained. Hence the dataset was suitable for use in modeling COVID-19 prediction.

Figure 10 shows the correlation plot to check the reduced dataset against redundancy. The PCA broke the redundancy in the dataset and encouraged feature to label relationship. The lag plot in Figure 11 checks the randomness in the dataset and also identifies outliers and lack of pattern. There exists a unique pattern in the dataset, hence the dataset suitable to use.

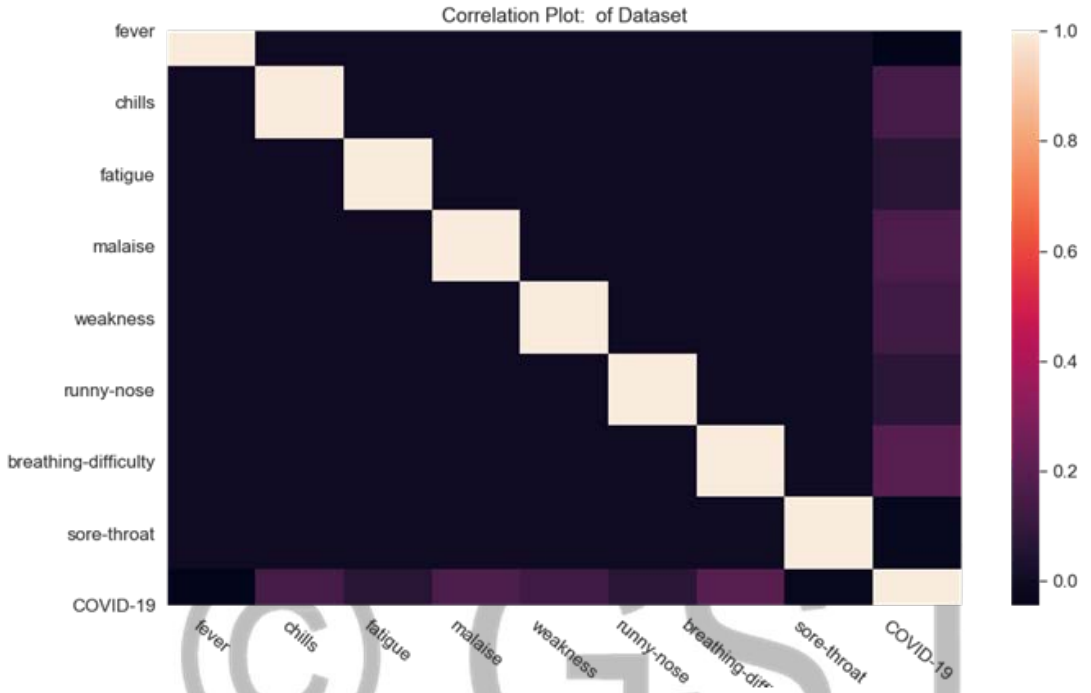


Figure 10: Correlation Plot of the Reduced Dataset

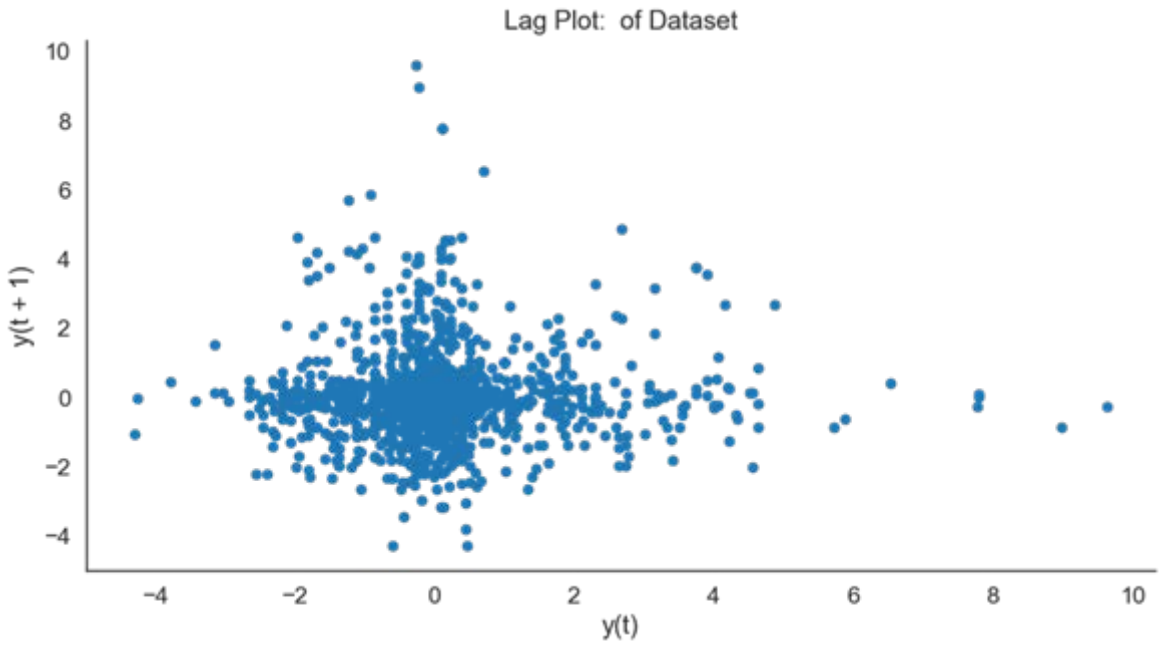


Figure 11: Lag Plot of the Reduced Dataset

The separation of the inliers (+1) and the outliers (-1) into their distinct classes is shown in the Andrew's curve in Figure 12.

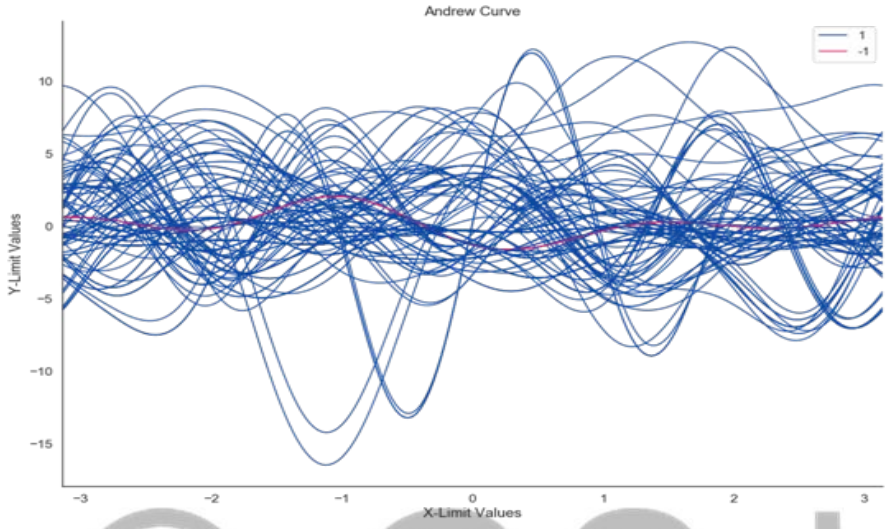


Figure 12: Andrew's Curve for the Reduced Dataset

The histogram plot and the pattern (line plot) of the 8 features of the reduced dataset is depicted respectively in Figures 13 and 14.

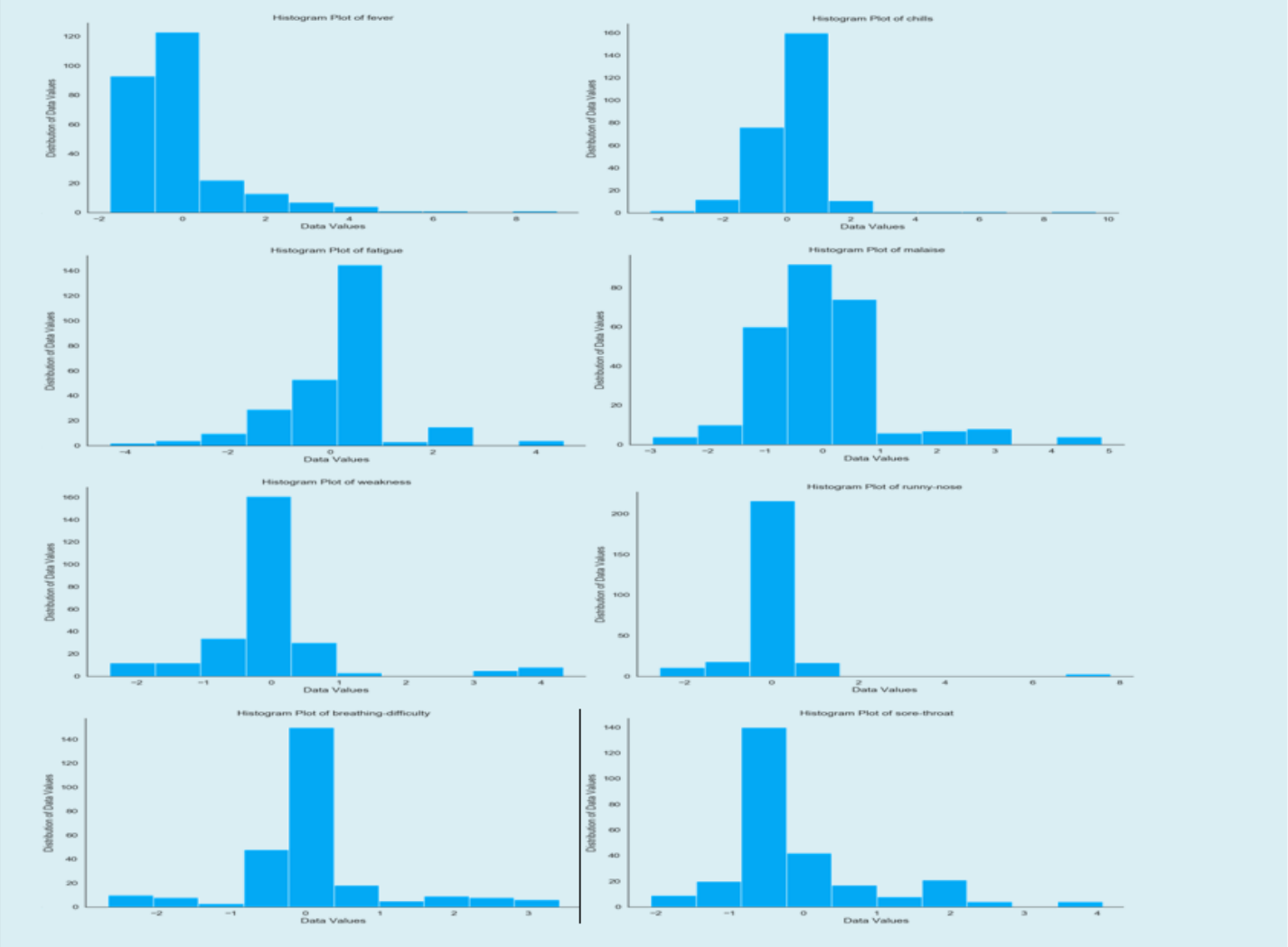


Figure 13: Histogram Plot of the Reduced Dataset

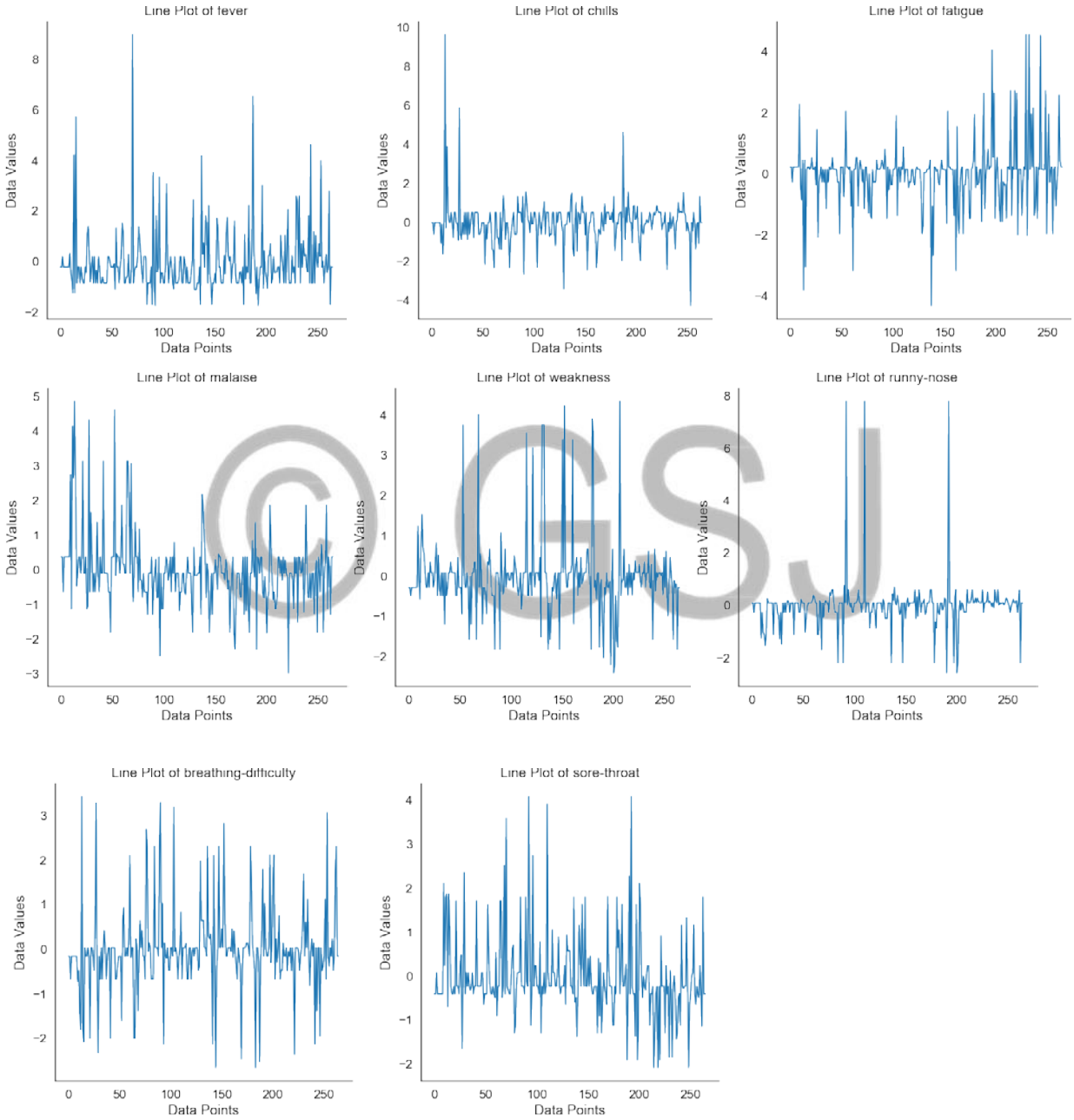


Figure 14: Line Plot of the Reduced Dataset

3.1 One-Class Support Vector Machine (OC-SVM)

In this work, one class support vector machine is used to identify corona virus patients from all other patients, by primarily learning from a training set containing a majority of corona virus case [Oliveri, 2017]. Support vector machine separates all the data points from the origin (in feature space F) and maximizes the distance from the hyperplane to the origin. The result is a binary function which captures regions in the input space where the probability density of the data lives. Thus the function returns +1 in a "small" region (capturing the training data points) and -1 elsewhere.

The quadratic programming minimization function for OC-SVM algorithm is presented as:

$$\begin{aligned} &\min_{w,\xi_i,\rho} \frac{1}{2}\|W\|^2 + \frac{1}{vn}\sum_{i=1}^{n}\xi_i - \rho \\ &\left(w.\Phi(X_i)\right) \geq \rho - \xi_i \; for\; all\; i = 1,\dots,n \\ &\xi_i \geq 0 \qquad\qquad for\; all\; i = 1,\dots,n \end{aligned} \tag{3}$$

The parameter v decides the smoothness of the function. The functions of the parameter v are as follows;

1. It sets an upper bound on the fraction of outliers (training examples regarded out-of-class).
2. It is a lower bound on the number of training examples used as Support Vector.

By using Lagrange techniques and kernel function for the dot-product calculations, the decision function becomes:

$$f(x) = sgn\left(\left(w.\Phi(x_i)\right) - \rho\right) = sgn(\sum_{i=1}^{n}\alpha_i K(x,x_i) - \rho) \tag{4}$$

The OC-SVM thus creates a hyperplane characterized by w and ρ which has maximal distance from the origin in feature space F and separates all the data points from the origin.

3.2 Isolation Forest (I-Forest)

I-Forest separates anomalous samples (instances in the dataset that do not conform to the normal profile [Chandola et al., 2009]) from the rest of the sample by recursively generating partitions on the sample by randomly selecting an attribute and selecting a split value for the attribute, between the minimum and maximum values allowed for that attribute. The number of partitions required to isolate a point is the length of the path, within the tree, to reach a terminating node starting from the root.

Anomaly detection with I-Forest is a process composed of two main stages [Liu et al., 2008a]:

1. A training dataset is used to build Isolation Trees (iTrees).
2. Each instance in the test set is passed through the iTrees built in the first stage and "anomaly score" is assigned to the instance. Once all the instances have been assigned an anomaly score, it is possible to mark as "anomaly" any point whose score is greater than a predefined threshold.

3.2.1 Isolation Trees (iTrees) Defined

Let $X = x_1,\dots,x_n$ be a set of d-dimensional points and $X' \subset X$ a subset of X. An iTree is defined as a data structure with the following properties:

1. For each node T in the Tree, T is either an external-node with no child, or an internal-node with one "test" and exactly two daughter nodes (T_l, T_r)
2. A test at node T consists of an attribute q and a split value p such that the test q < p determines the traversal of a data point to either T_l or T_r.

In order to build an iTree, the algorithm recursively divides X' by randomly selecting an attribute q and a split value p, until either (i) the node has only one instance or (ii) all data at the node have the same values.

When the iTree is fully grown, each point in X is isolated at one of the external nodes. Intuitively, the anomalous points are those (easier to isolate, hence) with the smaller *path length* in the tree, where the path length $h(x_i)$ of point $x_i \in X$ is defined as the number of edges x_i traverses from the root node to get to an external node.

3.2.2 Algorithm for Computing Anomaly Score

The algorithm for computing the anomaly score of a data point is based on the observation that the structure of iTrees is equivalent to that of Binary Search Tree (BST): a termination to an external node of the iTree corresponds to an unsuccessful search in the BST [Liu et al., 2008b]. As a consequence, the estimation of average h(x) for external node terminations is the same as that of the unsuccessful searches in BST [Shaffer, 2011].

$$c(m) = \begin{cases} 2H(m-1) - \frac{2(m-1)}{m} \; for \; m > 2 \\ 1 \; for \; m = 2 \\ 0 \; otherwise \end{cases} \quad (5)$$

Where n is the test data size, m is the size of the sample set and H is the harmonic number, which can be estimated by $H(i) = \ln(i) + \gamma$, where $\gamma = 0.5772156649$ is the Euler-Mascheroni constant.

The value of c(m) represents the average of h(x) given m, so we can use it to normalize h(x) and get an estimation of the anomaly score for a given instance x:

$$s(x, m) = 2^{\frac{-E(h(x))}{c(m)}} \quad (6)$$

where $E(h(x))$ is the average value of h(x) from a collection of iTrees. It is interesting to note that for any given instance *x*:

1. if *s* is close to 1 then *x* is very likely to be an anomaly
2. if *s* is smaller than 0.5 then *x* is likely to be a normal value

If for a given sample, all instances are assigned an anomaly score of around 0.5, then it is safe to assume that the sample does not have any anomaly.

3.3 Minimum Covariance Determinant (MCD)

MCD is a highly robust estimator of multivariate location and scatter, for which a fast algorithm is available. Since estimating the covariance matrix is the cornerstone of many multivariate statistical methods, the MCD is an important building block when developing robust multivariate techniques. It also serves as a convenient and efficient tool for outlier detection [Hubert et al., 2018]. Suppose we take a random sample of size **h**. We can evaluate the similarity between data points in the full set and our randomly sampled subset. In particular the Mahalanobis distance is used.

Let M be the mean of the random subset and **S** be the standard covariance of the random subset. The Mahalanobis distance is defined as:

$$D = [(x - M)S^{-1}(x - M)]^{\frac{1}{2}} \quad (7)$$

The Minimum Covariance Algorithm is as follows:

STEP 1: choose a random subset of H0 of X, with size h
STEP 2: repeat

a. Determine covariance S and mean M of the subset H0
b. Determine distances d(Xi) for all Xi relative to H with the Mahalanobis distance
c. Choose the h smallest distances and create a new subset H1
d. repeat with Ho <- H1, until Ho and H1 are equal or 0

STEP 3: Evaluate from 1 for K times (maybe 500) and determine the selection that had the smallest volume.

3.4 Local Outlier Factor (LOF)

This is an algorithm that was proposed by [Breunig et al., 2000] for finding anomalous data points by measuring the local deviation of a given data point with respect to its neighbours. LOF is based on a concept of a local density, where locality is given by *k* nearest neighbors, whose distance is used to estimate the density. By comparing the local density of an object to the local densities of its neighbors, one can identify regions of similar density, and points that have a substantially lower density than their neighbors. These are considered to be outlier.

The local density is estimated by the typical distance at which a point can be "reached" from its neighbors. The definition of "reachability distance" used in LOF is an additional measure to produce more stable results within clusters.

The steps used in LOF are as follows:
STEP 1: Distance Calculation
STEP 2: Kth-Nearest Neighbor Distance Calculation
STEP 3: K-Nearest Neighbor Calculation
STEP 4: Local Reachability Density (LRD) Calculation
STEP 5: LOF calculation
STEP 6: Analysis

The Reachability distance of Local Outlier Factor algorithm thus:

$$reachability - distance_k(o \leftarrow o') = max\{dist_k(o), dist(o, o')\} \quad (8)$$

Where:
k is specified by the user.
$dist_k(o)$: is the distance between o and its k-th NN(k-th nearest neighbor)
The k-distance neighborhood of o is defined as;

$$N_k(o) = \{o' | o' \, in \, D, dist(o, o') \leq dist_k(o)\} \quad (9)$$

The local reachability density of o is:

$$lrd_k(o) = \frac{\|N_k(o)\|}{\sum_{o' \in N_k(o)} reac\,hability\, -distance\,_k(o' \leftarrow o)} \quad (10)$$

The LOF of an object o is the average of the ratio of local reachability of o and those of o's k-th nearest neighbors. LOF is presented as:

$$LOF_k(o) = \frac{\sum_{o' \in N_k(o)} \frac{lrd\,_k(o')}{lrd\,_k(o)}}{\|N_k(o)\|} = \sum_{o' \in N_k(o)} lrd_k(o') \, . \sum_{o' \in N_k(o)} reachability - distance_k(o' \leftarrow o) \quad (11)$$

The lower the local reachability density of o, and the higher the local reachability density of the kNN of o, the higher LOF. Every LOF above a given threshold is considered an outlier.

4 PERFORMANCE EVALUATION

The performance of the models (One-Class SVM, Isolation Forest, Minimum Covariance Determinant and Local Outlier Factor) used in predicting COVID-19 cases was evaluated using Accuracy, Precision, Recall, and F1-score.
Table 1 is the confusion matrix components used in the evaluation:

Table 1: Confusion Matrix

		Predicted Class	
		+1	-1
Actual Class	+1	True Positive (TP)	False Negative (FN)
	-1	False Positive (FP)	True Negative (TN)

1. **Accuracy** – This is the ratio of correctly predicted observation to the total observations. Accuracy is computed as:

$$Accuracy = \frac{True\ Positive + True\ Negative}{Total\ Sample} \equiv \frac{TP + TN}{N}$$

2. **Precision** – Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \equiv \frac{TP}{TP + FP}$$

3. **Recall (Sensitivity)** – Recall is the ratio of correctly predicted positive observations to the all observations in actual class that should have been identified as positive (i.e COVID-19 cases)

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \equiv \frac{TP}{TP + FN}$$

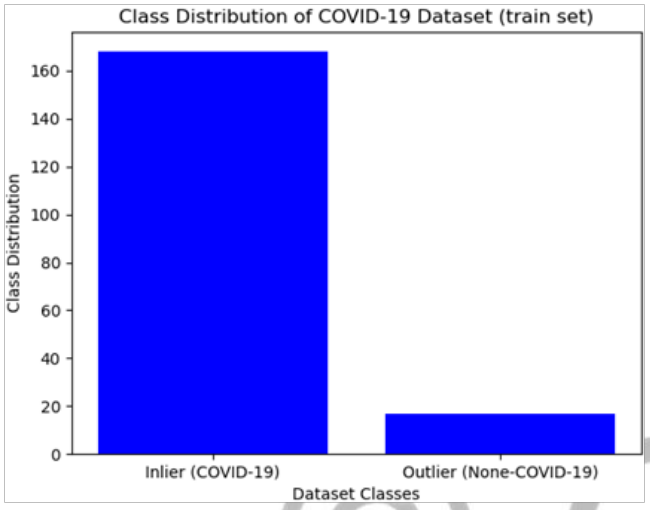
4. **F1 Score** – F1-Score is the harmonic mean of precision and recall. Its value ranges from 0 to 1. The F1-Score is computed as follows. F1 is usually more useful than accuracy, especially in the case of our COVID-19 prediction in which our dataset is unbalanced.

$$F1 - Score = 2 * \frac{Precision * recall}{Precision + recall}$$

5 RESULTS AND DISCUSSION

The COVID-19 dataset was partitioned into training set (70%) and test set (30%) for model training and testing respectively. The models trained were One-Class Support Vector Machine, Isolation Forest, Minimum Covariance Determinant, and Local Outlier Factor. Both the training and test sets shared the same set of features - fever, chills, fatigue, malaise, weakness, runny-nose, breathing-difficulty, and sore-throat. In this work, the output variable represents COVID-19 positive or negative case.

The class distribution that reveals the unbalanced nature of the dataset for both the training and test set is shown in Figure 15 and Figure 16 respectively.



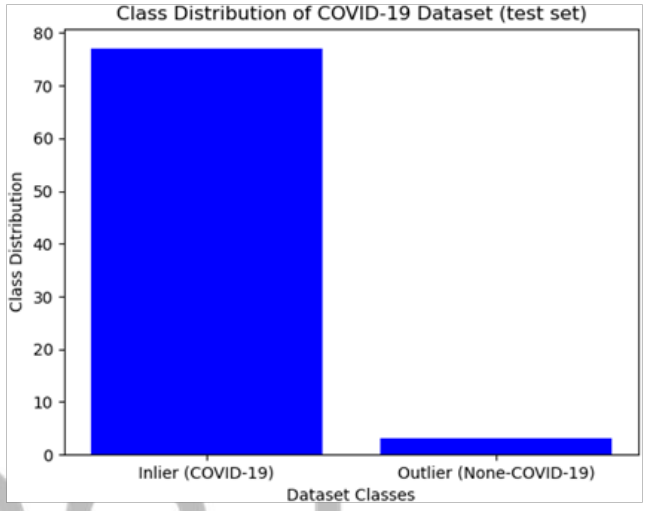


Figure 15: Class Distribution of Training Set **Figure 16: Class Distribution of Test Set**

From the training dataset class distribution in Figure 15, 91% of the dataset represents COVID-19 positive cases while 9% represents COVID-19 negative cases. Also, from the test dataset class distribution in Figure 16, 96% represents COVID-19 positive cases while 4% represents COVID-19 negative cases. Higher percentage value of COVID-19 positive cases points to the fact that our dataset for the prediction of COVID-19 cases is unbalanced; hence this dataset is only suitable for use in one class classification (outlier or anomaly detection) algorithms which gives rise to the choice of algorithms used in this work.

5.1 Model Training

Our model training is a case of outlier detection (or anomaly detection) since the COVID-19 dataset is unbalanced. The model's training predictions (predictions gotten by passing the training set to the trained models) are presented in Figure 17 where each dot is a data point in the training dataset. The blue dots represent positive cases of COVID-19 (inliers) while the red dots represent negative cases (outliers).

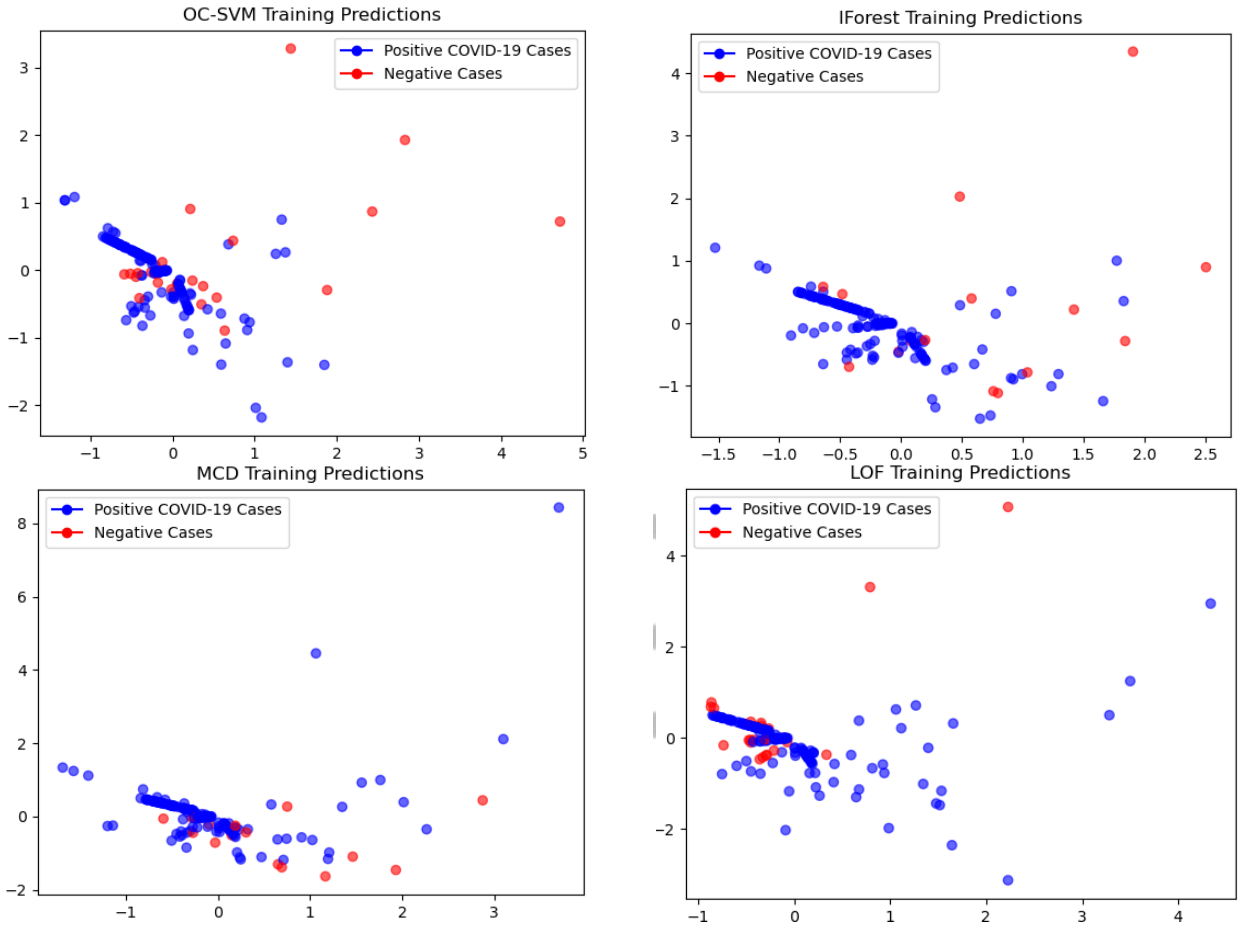


Figure 17: Model's Training Predictions

To visualize the effectiveness of the trained models on the training dataset, the plot of correctly and incorrectly classified points is used and is shown in Figure 18. The blue dots are the correctly classified points (True Positive or True Negative) and the red dots are the incorrectly classified points (False Positive or False Negative). As observed, Isolation Forest has the least number of incorrectly classified points, which implies that it has the best accuracy.

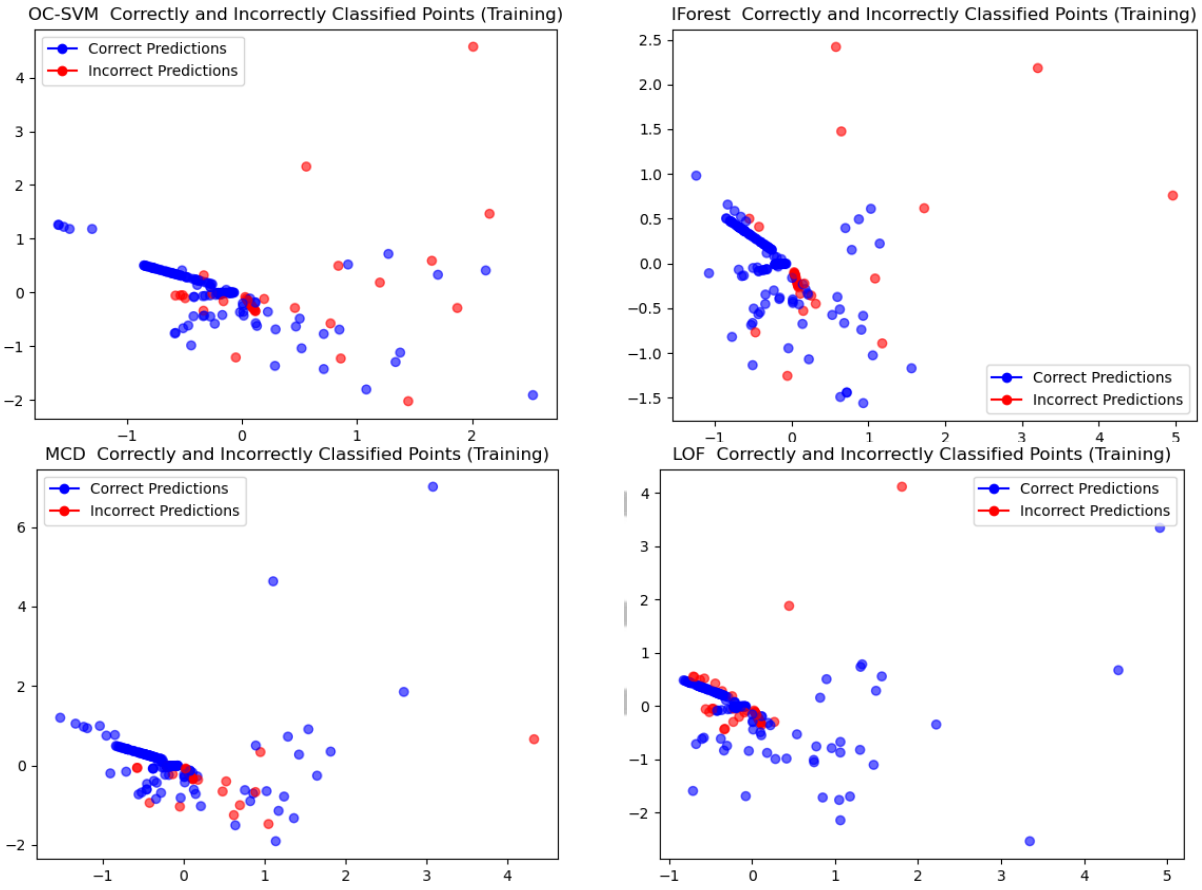


Figure 18: Correctly and Incorrectly Classified Training Points

The performance of our models was evaluated using Accuracy, Precision, Recall, and F1-Score. The scores are shown in Table 2 while the bar chart comparing the prediction models is depicted in Figure 19.

Table 2: Model's Training Performance

Models	Training Performance Metrics			
	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
One-Class SVM	0.783784	0.812813	0.783784	0.798034
Isolation Forest	0.827027	0.817297	0.827027	0.822133

Minimum Covariance Determinant	0.816216	0.816216	0.816216	0.816216
Local Outlier Factor	0.789189	0.813397	0.789189	0.801111

Figure 19: Performance of the Trained Models

From Table 2, Isolation Forest gives a better performance on the training dataset with an accuracy of 0.827027, precision of 0.817297, recall of 0.827027 and F1-Score of 0.822133. The bar chart in Figure 19 also confirmed Isolation Forest as the best.

5.2 Model Testing

30% of the dataset was used for model testing. The trained models were also evaluated using the same set of performance metrics. The test predictions are presented in Figure 20. Isolation Forest predicted more of the positive cases than the negative cases while one-class SVM predicted equal number of positive and negative cases.

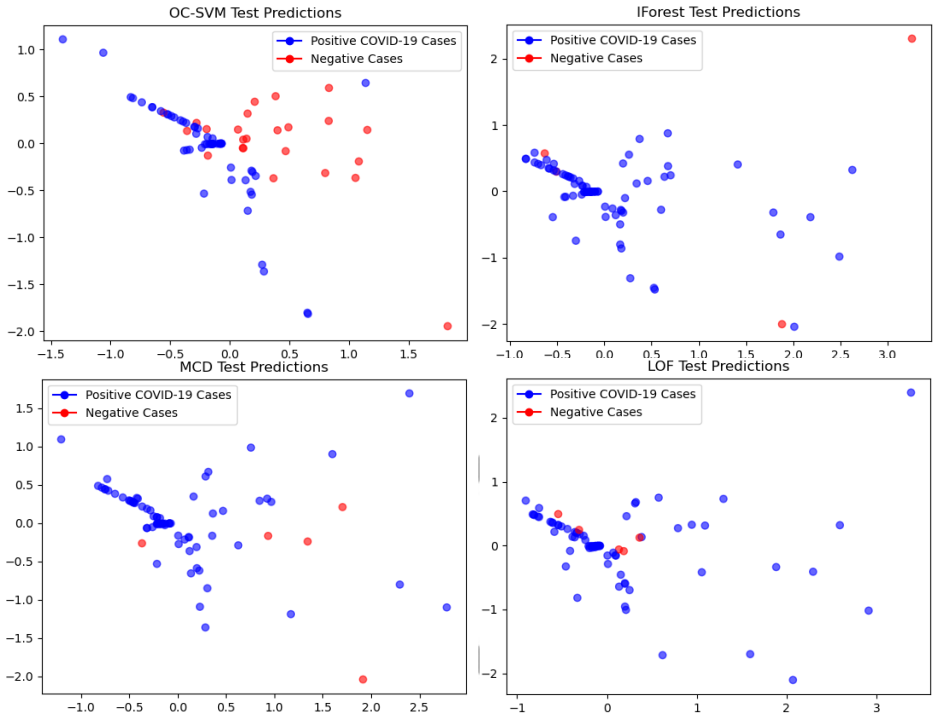


Figure 20: Model's Test Predictions

A scatter plot of correct and incorrect predictions is presented in Figure 21 with blue and red dots representing correctly classified points and incorrectly classified points. Isolation Forest performed better with minimum amount of incorrectly classified points while One-Class SVM performed poorly with high number of incorrectly classified points

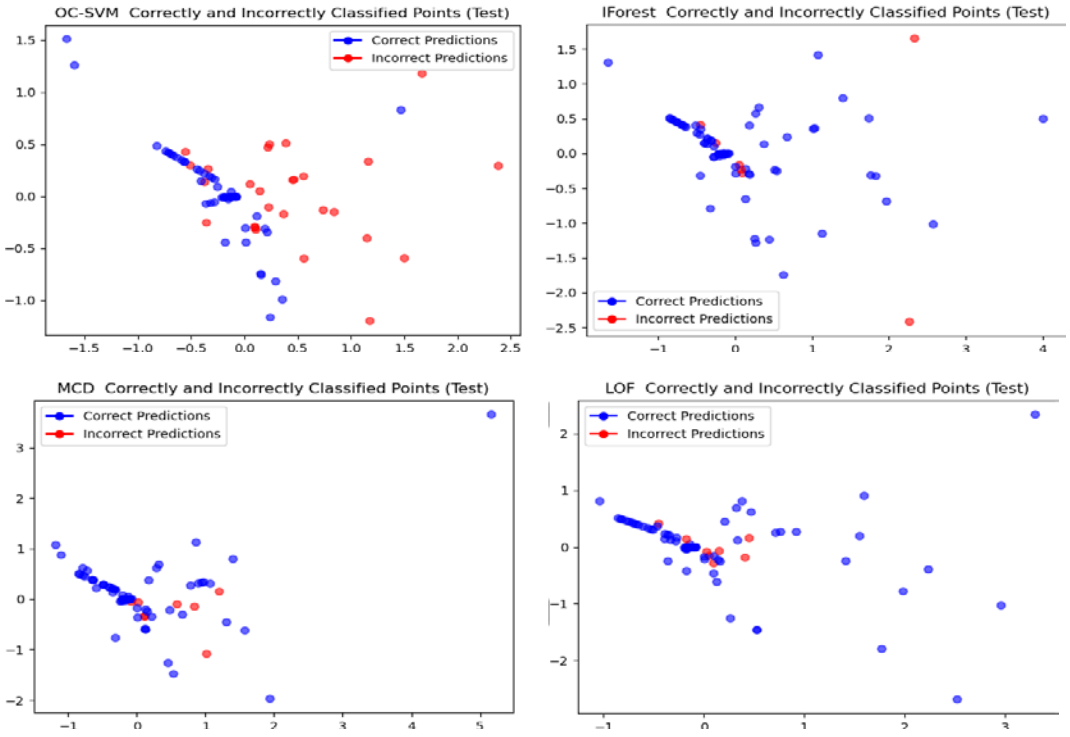


Figure 21: Correct and Incorrect Test Prediction

The performance evaluation result of the models on the test dataset is shown in Table 3 and the bar chart comparing the prediction models is depicted in Figure 22.

Table 3: Model's Test Performance

Models	Test Performance Metrics			
	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
One-Class SVM	0.6625	0.910938	0.6625	0.767105
Isolation Forest	0.9125	0.924507	0.9125	0.918464
Minimum Covariance	0.9	0.924	0.9	0.911842

Determinant				
Local Outlier Factor	0.9	0.924	0.9	0.911842

Figure 22: Test Performance of the Trained Model

From Table 3 Isolation Forest was the best model with accuracy of 0.9125, precision of 0.924507, recall of 0.9125, and F1-Score of 0.918464. The bar chart in Figure 24 also confirmed that. One-Class SVM lagged behind while Minimum Covariance Determinant and Local Outlier Factor are the same in performance when tested with the 30% test dataset.

5.3 Model Selection

From the comparative analysis of the predictive models considering the result of the F1-score metrics shown in Table 4, a Model with the highest training and test performance is selected as the best model. The performance analysis of the predictive models is also presented in a bar chart in Figure 23.

Table 4: Comparative analysis of Predictive Models for COVID-19 prediction

	Predictive Models			
	One-Class SVM	Isolation Forest	Minimum Covariance Determinant	Local Outlier Factor
Training Performance (%)	0.798034	0.822133	0.816216	0.80111

Test Performance (%)	0.767105	0.918464	0.911842	0.911842

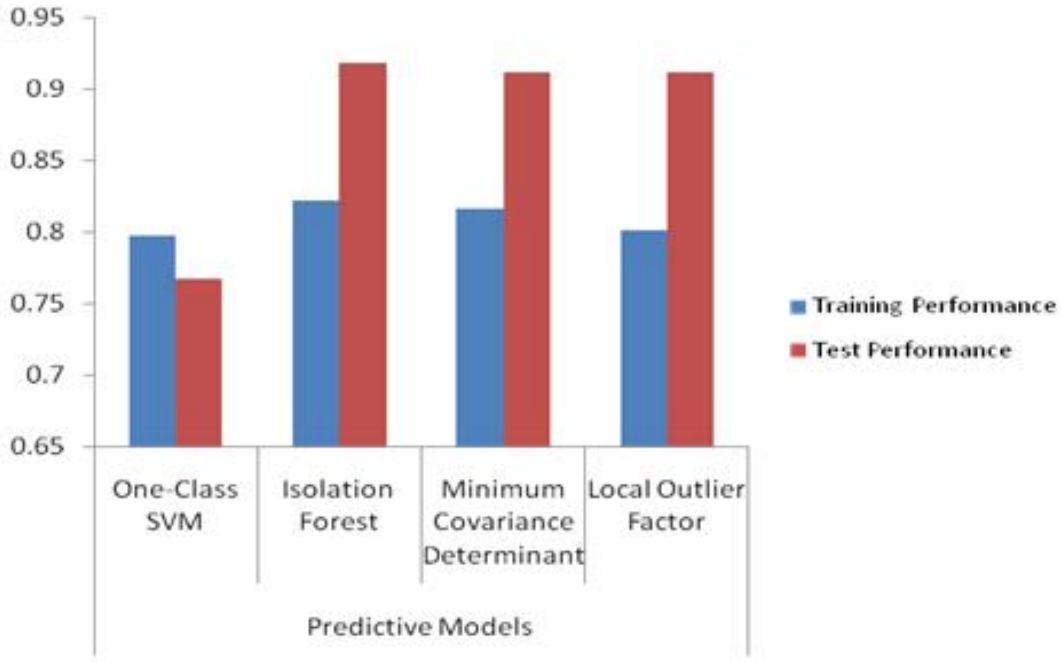


Figure 23: Performance Analysis of COVID-19 Predictive Models

From Table 4 and Figure 23, Isolation Forest performs better in Predicting cases of COVID-19 both in the training and test dataset with the F1-Score of 0.822133% for training and 0.918464% for testing respectively. Hence Isolation Forest is selected as the best, among others used, for the prediction of COVID-19 cases.

6 CONCLUSION

This work analyses comparatively four (4) ML models (OC-SVM, I-Forest, MCD and LOF) for the prediction of COVID-19 cases using dataset from kaggle.com. The dataset is unbalanced and suitable for use in one class classification, hence the choice of the predictive models.

The Framework comparative analysis framework comprised the input data used in training and testing the COVID-19 predictive models, the predictive models trained to predict cases of COVID-19 and model evaluation and selection that chooses the best performing model based on its F1-Score.

The work successfully trained the COVID-19 predictive models using a version of the dataset preprocessed using One-Hot encoding and Principal component analysis for converting the categorical dataset to numerical dataset and reducing the dataset's dimension respectively. The work performed a comparative analysis of the various models for the prediction of COVID-19. The models where capable of predicting cases of COVID-19 in the dataset with a F1-Score performance ranging from 0.798034 to 0.822133 for training and 0.767105 to 0.918464 for testing with Isolation Forest giving the best F1-Score; accuracy of 0.822133 and 0.918464 for training and testing respectively.

The model is capable of predicting COVID-19 cases with higher accuracy thereby helping to drastically curb the spread of COVID-19.

We hope to apply the best performing model (I-Forest) in our ongoing work in software aided contact tracing for corona virus cases.

References

1. Anderson, K., Odell, P., Wilson, P. and Kannel, W. (1991). Cardiovascular Disease risk Profiles. American Heart Journal, 121(1), 293 – 298.

2. Anno, S.; Hara, T.; Kai, H.; Lee, M.A.; Chang, Y.; Oyoshi, K.; Mizukami, Y.; Tadono, T. (2019). Spatiotemporal Dengue Fever Hotspots Associated with Climatic factors in Taiwan including Outbreak Predictions Based on Machine-Learning. Geospatial Health, 14, 183-194, doi:10.4081/gh.2019.771.

3. Ardabili, S.F.; Mosavi, A.; Ghamisi, P.; Ferdinand, F.; Varkonyi-Koczy, A.R.; Reuter, U.; Rabczuk, T.; Atkinson, P.M. (2020) COVID-19 Outbreak Prediction with Machine Learning. Preprints 2020, 2020040311 (doi: 10.20944/preprints202004.0311.v1).

4. Arora, P., Kumar, H. and Panigrahi, B. (2020). Prediction and Analysis of COVID-19 Positive Cases using Deep Learning Models: A Descriptive Case Study of India. Chaos, Spolitons and Fractals, 139, 1-9.

5. Asri, H., Mousannif, H., Moatassime, H. and Noel, T. (2016).`Using Machine Learning Algorithms for Breast Cancer risk Prediction and Diagnosis. Procedia Computer Science, Vol. 83, 1064 – 1069.

6. Breunig, M. M.; Kriegel, H.-P.; Ng, R. T.; Sander, J. (2000). LOF: Identifying Density-based Local Outliers. Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data. SIGMOD. 93–104.

7. Car, Z., Segota, S., Andelic, N., Lorencin, I. and Mrzljak, V. (2020). Modeling the Spread of COVID-19 Infection using Multilayer Perceptron. Computational and mathematical Methods in Medicine, doi:10.1155/2020/5714714

8. Chandola, Varun; Banerjee, Arindam; Kumar, Kumar (2009). Anomaly Detection: A Survey. ACM Computing Surveys. 41. doi:10.1145/1541880.1541882.

9. Chen N., Zhou M., Dong X., Qu J., Gong F., Han Y., et al. (2020). Epidemiological and Clinical Characteristics of 99 cases of 2019 novel Coronavirus Pneumonia in Wuhan, China: A Descriptive study, Lancet, 395:507–513. doi: 10.1016/S0140-6736(20)30211-7

10. Huang C, Wang Y, Li X, Ren L, Zhao J, Hu Y, et al (2020). Clinical Features of Patients infected with 2019 novel Coronavirus in Wuhan, China. Lancet, 395:497–506. doi: 10.1016/S0140-6736(20)30183-5

11. Hubert M, Debruyne M, Rousseeuw PJ.(2018) Minimum Covariance Determinant and Extensions. WIREs Comput Stat.;10:e1421. https://doi.org/10.1002/wics.1421

12. Iwendi, C., Bashir, A., Peshkar, A., Sujatha, R., Chatterjee, J., Pasupuleti, S., Mishra, R., Pillai, S. and Jo, O. (2020). COVID-19 Patients Health Prediction using Boosted Random Forest Algorithm. Frontiers in Public Health, Vol. 8, Article 357. Doi: 10.3389/fpubh.2020.0035

13. John Hopkins University (JHU), 2020. COVID-19 Dashboard by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University (JHU). https://www.coronavirus.jhu.edu/map.html (accessed August 31, 2020).

14. Khanday, A., Rabani, S., Khan, Q., Rouf, N. and Din, M. (2020). Machine Learning Based Approaches for Detecting COVID-19 using Clinical Text Data. International Journal of Information Technology, doi: 10.1007/s41870-202-00494-9

15. Kirbas, I., Sozen, A., Tuncer, A. and Kazancioghu, F. (2020). Comparative Analysis of Forcasting of COVID-19 Cases in Various European Countries with ARIMA, NARIN and LSTM approaches. Chaos, Solitons and Fractals, 138.

16. Koike, F.; Morimoto, N. (2018). Supervised Forecasting of the Range Expansion of Novel Non-indigenous Organisms: Alien Pest Organisms and the 2009 H1N1 flu pandemic. Global Ecology and Biogeography, 27, 991-1000, doi:10.1111/geb.12754

17. Lalmuanawma, S., Hussain, J. and Chhakchhuak, L. (2020). Applications of Machine Learning and Artificial Intelligence for COVID-19 (SARS-COV-2) Pandemic: A Review. Chaos, Solitons and Fractals, doi: 10.1016/j.chaos.2020.110059

18. Lapuerta, P., Azen, S. and Labree, L. (1995). Use of Neural Networks in Predicting the risk of Coronary Artery Disease. Computers and Biomedical Research, 28(1) 38 – 52.

19. Liang, R.; Lu, Y.; Qu, X.; Su, Q.; Li, C.; Xia, S.; Liu, Y.; Zhang, Q.; Cao, X.; Chen, Q., et al. (2020). Prediction for Global African Swine Fever Outbreaks Based on a Combination of Random Forest Algorithms and Meteorological Data. Transboundary Emerging Diseases, 67(2), 935-946, doi:10.1111/tbed.13424.

20. Li Q, Guan X, Wu P, Wang X, Zhou L, Tong Y, et al. (2020). Early Transmission Dynamics in Wuhan, China, of novel Coronavirus–infected Pneumonia, New England Journal of Medicine, 382:1199–1207. doi: 10.1056/NEJMoa2001316

21. Liu D, Clemente L, Poirier C, Ding X, Chinazzi M, Davis JT, et al. (2020). A machine learning methodology for real-time forecasting of the 2019–2020 COVID-19 outbreak using Internet searches, news alerts, and estimates from mechanistic models. arXiv. 2004.04019. Available online at: https://arxiv.org/abs/2004.04019

22. Liu, Fei Tony; Ting, Kai Ming; Zhou, Zhi-Hua (2008a). Isolation Forest. In Proceeding of Eighth IEEE International Conference on Data Mining: 413–422. doi:10.1109/ICDM.2008.17.

23. Liu, Fei Tony; Ting, Kai Ming; Zhou, Zhi-Hua (2008b). Isolation-based Anomaly Detection. ACM Transactions on Knowledge Discovery from Data. 6: 1–39.

24. Nemati, M., Ansang, J. and Nemati, N. (2020). Machine Learning Approaches in COVID-19 Survival Analysis and Discharge Time Likelihood Prediction using Clinical Data Pattern, doi:https://doi.org/10:1016/j.patter.2020.100074

25. Oliveri, P. (2017). Class-modelling in food Analytical Chemistry: Development, Sampling, Optimisation and Nalidation Issues - A tutorial. Analytica Chimica Acta, 982: 9 – 19.

26. Pinter, G., Felde, I., Mosavi, A., Ghamisi, P. and Gloaguen, R. (2020). COVID-19 Pandemic Prediction for Hungary: A Hybrid Machine Learning Approach. Mathematics, 8, 890, doi: 10.3390/math80608990.

27. Ribeiro, M. H. D. M., da Silva, R. G., Mariani, V. C., & Coelho, L. dos S. (2020). Short-term forecasting COVID-19 cumulative confirmed cases: Perspectives for Brazil. Chaos, Solitons and Fractals, 109853, doi:10.1016/j.chaos.2020.109853

28. Rustam, F., Reshi, A., Mehmood, A., Ullah, S., On, B., Aslam, W. and Choi, G. (2020). COVID-19 Future Forecasting using Surpervised Machine Learning Models. IEEE Access, Vol. 8, 101489 – 101499.

29. Shaffer, Clifford A. (2011). Data Structures & Algorithm Analysis in Java (3rd Dover ed.). Mineola, NY: Dover Publications. ISBN 9780486485812. OCLC 721884651.

30. Shinde, G., Kalamkar, A., Mahalle, P., Dey, N., Chaki, J. and Hassanien, A. (2020). Forecasting Models for Coronavirus Disease (COVID-19): A Survey of the State-of-the-Art. SN Computer Science. 1:197. Doi: 10.1007/s42979-020-00209-9

31. Tapak, L.; Hamidi, O.; Fathian, M.; Karami, M. (2019). Comparative evaluation of time series models for predicting influenza outbreaks: Application of influenza-like illness data from sentinel sites of healthcare centers in Iran. BMC Research Notes, 12, doi:10.1186/s13104-019-4393-y.

32. Tuli, S., Tuli, S., Tuli, r. and Gill, S. (2020). Predicting the Growth and Trend of COVID-19 Pandemic using Machine Learning and Cloud Computing. Internet of Things, doi: https://doi.org/10.1016/j.iot.2020.100222.

33. Wang, P., Zheng, X., Li, J. and Zhu, B. (2020a). Prediction of Epidemic Trends in COVID-19 with Logistic Model and Machine Learning Technigues. Chaos, Solitons and Fractals, doi: https://doi.org/10.1016/j.chaos.2020.110058

34. Wang, S., Ding, S. and Xiong, L. (2020b). A New System for Surveillance and Digital Contact Tracing for COVID-19: Spatiotemporal Reporting Over Network and GPS. JMIR Mhealth and Uhealth, 8(6), doi: 10.2196/19457.

35. Waqas, M., Farooq, M., Ahmad, R. and Ahmed, A. (2020). Analysis and Prediction of COVID-19 Pandemic in Pakistan using Time-dependent SIR Model. arXiv:2005.02353v2

36. World Health Organization (WHO), 2020. Coronavirus disease (COVID-2019) situation Reports .https://www.who.int/emergencies/diseases/novel-coronavirus-2019/situation- reports/ (accessed August 31, 2020).