Prediction, Cross Validation and Classification in the Presence COVID-19 of Indian States and Union Territories using Machine Learning Algorithms

P. Arumugam, V. Kadhirveni, R. Lakshmi Priya, Manimannan G

Abstract: The present study predicts, cross validate and classify the data of COVID-19 based on four machine learning algorithm with four major parameters namely confirmed cases, recoveries, deaths and active cases. The secondary sources of database were collected from Ministry of Health and Family Welfare Department (MHFWD), from Indian State and Union Territories up to March, 2021. Based on these background, the database classified and predicted various machine learning Algorithm, like SVM, kNN, Random Forest and Logistic Regression. Initially, the k-mean clustering analysis is used to perform and identified five meaningful clusters and is labeled as Very Low, Low, Moderate, High and Very High of four major parameters based on their average values. In addition the five clusters are cross validated using four machine learning algorithm and affected states were visualized with help of prediction and probabilities. The different machine learning models achieved cross validation accuracy of 88%, 97%, 91% and 91%. . Delhi, Uttar Pradesh and West Bengal were Moderately Affected States, Assam, Bihar, Chhattisgarh, Haryana, Gujarat, Madhya Pradesh, Odisha, Punjab, Rajasthan and Telangana are Low Affected States, wherein Tamil Nadu, Kerala, Andhra Pradesh and Karnataka are highly affected States. and Maharashtra the Very Highly Affected State. Rest of the States and Union Territories has Very Low affected Covid-19 Cases is clearly identified.

Keywords: COVID-19, Machine Learning Algorithms, Prediction, Cross Validation and Classification.

I. INTRODUCTION

The COVID-19 pandemic disease caused by SARS-CoV-2 virus and this virus was identified from Wuhan, China in the year 2019. World Health Organization declared world pandemic situation on 11th March, 2020, its spread over the world in a short period. Many people were affected by this virus and lost their lives, economy, jobs, Education, etc. In India, the first case is recorded from Kerala and it has spread over the Indian states and union territories. Recently, the second wave of COVID-19 spread is exponentially increasing in all states and union territories. The Indian government launched vaccine camps for the age group of above 45 to upper age people. The vaccine raises immunity in our human bodies and second dosage of vaccine is from 28 to 42 days.

The main objective of this research paper is to identify the classification and visualization of affected persons using various machine learning and statistical algorithm.

Manuscript received on April 10, 2021.
Revised Manuscript received on April 17, 2021.
Manuscript published on May 30, 2021.

*Correspondence Author

P. Arumugam, Professor, Department of Statistics, Manonmaniam Sundaranar University, Tirunelveli (Tamil Nadu), India.

V. Kadhirveni, Research Scholar, Department of Statistics, Manonmaniam Sundaranar University, Tirunelveli (Tamil Nadu), India.

R. Lakshmi Priya, Assistant Professor, Department of Statistics, Dr. Ambedkar Government Arts College, Vyasarpadi, Chennai (Tamil Nadu), India.

Manimannan G*, Assistant Professor. Department of Statistics, TMG College of Arts and Science, Chennai (Tamil Nadu), India.

II. BACKGROUND OF THE STUDY

In Data Mining the KDD (Knowledge Discovery in a Database) is the iterative process that uses to discover novel information and knowledge from large amounts of database. According to Han and Kamber, data mining software allows end-users to analyze data from different dimensions Also, categorize the details and summarize the relationships which are identified during the mining process [1]. Kamber and Han, classification falls into a supervised data mining technique. This process consists of two steps, the first step is learning setup, in which the model is constructed and trained with the help of a predetermined database with class labels. The second step is the place where the trained model is consumed to perform the predictions for given database and measure the accuracy level of the classifier algorithm model [1]. Most of the researches are focusing on behavior of predictive models in data mining (Padmavathi Janardhana). With the help of modern technologies, the researcher can collect a large number of various types of data with different types of parameters in relevant field. Then apply the data mining techniques very correctly and effectively to mine and absorb meaningful interpretations, predictions, etc. when comes to the medical science field, the above said process can be applied to many database like, predicting, classification and visualization of breast cancer, heart attack, oral diseases, diabetes, etc. [2] According to Manimannan G. *et. al.* classification is defined as a process that gives the model to describe and differentiate database classes or concepts to predict the class of objects whose class label is not known. Partitioning clustering algorithm, artificial neural networks, etc. are the major tools used for constructing these models in COVID-19 of Indian States [3]. In recent days, data mining has become an attractive discipline that is used in business, medical science, engineering, text mining and other professional field as well in information technology community.

*Retrieval Number: 100.1/ijrte.A56590510121
DOI:10.35940/ijrte.A5659.0510121
Journal Website: www.ijrte.org*

*Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication
© Copyright: All rights reserved.*

Data mining strategies can provide useful answers to a problem. The methods are Classification, Association, Clustering, Estimation, Novelty detection, sequencing deduction, etc [4]

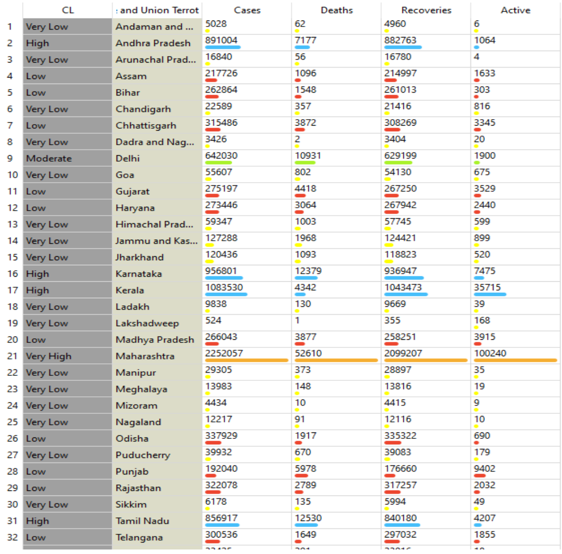
Rajkumar and G.S Reena carried out research using machine learning algorithms (such as K- nearest neighbor, Naive Bayes) for heart disease prediction.

The data set consists of 3000 instances with 14 attributes. Dataset was divided as 70% for training and 30% for testing. According to the test results, Naïve bayes algorithm was selected as the algorithm with better performances when compared with KNN and Decision List [8].Halgurd S. Maghdid *et al*. have projected a new framework to detect corona virus disease using the inboard smartphone sensors. The designed AI (Artificial Intelligence) framework collects data from various sensors to predict the grade of pneumonia as well as predicting the infection of the disease [6]. The proposed framework takes uploaded CT Scan images as the key method to predict COVID-19 [5] Manimannan G. *et. al*. has used Silhouette distance measure for k- means clustering algorithm. It produced effective results and visualized their result in a simple manner. This technique achieved three meaningful groups and is labeled as C1, C2 and C3. C1 represents highly affected, C2 Moderate affected and C1 Low affected States and Union Territories [6].

III. DATABASE

The secondary source of database was collected from Ministry of Health and Family Welfare Department (MHFWD), from Indian State and Union Territories up to March, 2021 (Table 1). The database consists of four parameters namely confirmed cases, recoveries, deaths and active cases. Initially all parameters are classified using k-means algorithm and additionally tested and cross validated 10 folds stratified sampling methods with help of testing database 10 and training database 60 percent. Subsequently, the machine learning algorithm of kNN, SVM, Random Forest classification and Neural network cross validates the original database and gives better results.

Table 1. Sample Data and Parameters



	CL	: and Union Terrot	Cases	Deaths	Recoveries	Active
1	Very Low	Andaman and ...	5028	62	4960	6
2	High	Andhra Pradesh	891004	7177	882763	1064
3	Very Low	Arunachal Prad...	16840	56	16780	4
4	Low	Assam	217726	1096	214997	1633
5	Low	Bihar	262864	1548	261013	303
6	Very Low	Chandigarh	22589	357	21416	816
7	Low	Chhattisgarh	315486	3872	308269	3345
8	Very Low	Dadra and Nag...	3426	2	3404	20
9	Moderate	Delhi	642030	10931	629199	1900
10	Very Low	Goa	55607	802	54130	675
11	Low	Gujarat	275197	4418	267250	3529
12	Low	Haryana	273446	3064	267942	2440
13	Very Low	Himachal Prad...	59347	1003	57745	599
14	Very Low	Jammu and Kas...	127288	1968	124421	899
15	Very Low	Jharkhand	120436	1093	118823	520
16	High	Karnataka	956801	12379	936947	7475
17	High	Kerala	1083530	4342	1043473	35715
18	Very Low	Ladakh	9838	130	9669	39
19	Very Low	Lakshadweep	524	1	355	168
20	Low	Madhya Pradesh	266043	3877	258251	3915
21	Very High	Maharashtra	2252057	52610	2099207	100240
22	Very Low	Manipur	29305	373	28897	35
23	Very Low	Meghalaya	13983	148	13816	19
24	Very Low	Mizoram	4434	10	4415	9
25	Very Low	Nagaland	12217	91	12116	10
26	Low	Odisha	337929	1917	335322	690
27	Very Low	Puducherry	39932	670	39083	179
28	Low	Punjab	192040	5978	176660	9402
29	Low	Rajasthan	322078	2789	317257	2032
30	Very Low	Sikkim	6178	135	5994	49
31	High	Tamil Nadu	856917	12530	840180	4207
32	Low	Telangana	300536	1649	297032	1855

IV. METHODOLOGY

The following section describes various machine learning Data Mining Algorithms and Workflow (Figure 1):

4.1 k-mean Clustering Algorithm

MacQueen [7] suggests the term k-means for describing an algorithm of his that assigns each item to the cluster having the nearest centroids. The process composed of these three steps:

Step 1: Partition the items (input database) into k-initial clusters.

Step 2: Proceed through the list of items, assigning an item to the cluster whose centroid is nearest (using Euclidean distance measure). Recalculate the centroid for the cluster receiving the new item and for the cluster losing the item.

Step 3: Repeat Step 2 until no more reassignments take place.

4.2 Random Forest Classification Algorithm

Random Forest is a classification technique proposed by Breiman, 2001 [8]. When given a set of class-labeled database**,** Random Forest builds a set of classification trees. Each tree is developed from a bootstrap sample from the training data. Classification is based on majority vote from individually developed tree classifiers in the forest.

Step 1: Specify the name of the classifier. The default name is “Random Forest Classification”.

Step 2*:* State how many classification trees will be included in the forest, and how many attributes will be arbitrarily drawn for consideration at each node. If the latter is not specified, this number is equal to the square root of the number of attributes in the data.

Step 3: Brieman’s proposal is to grow the trees without any pre-pruning, but since pre-pruning often works quite well and is faster, the user can set the depth to which the trees will be grown.

Step 4: Produce a report.

Step 5: Tick Apply to communicate the changes to other widgets. Alternatively, click the box on the left side of the Apply button and changes will be communicated automatically.

4.3 Support Vector Machine (SVM)

Statistical learning aims at gaining knowledge, making predictions, making decisions or constructing model from a set of database, Statistical learning theory gained renewed momentum in data mining after the introduction of SVM developed by Vapnik. *et. al* [9]. The Orange Data mining algorithm presents here:

Step 1: The learner can be given a name under which it will appear in other widgets.

Step 2: Classification type with test error settings. C-SVM and v-SVM are based on different minimization of the error function. On the right side, you can set test error bounds, Cost for C-SVM and Complexity bound for v-SVM.

Step 3: The next block of options deals with kernel, a function that transforms attribute space to a new feature space to fit the maximum-margin hyperplane, thus allowing the algorithm to create non-linear classifiers with Polynomial, RBF and Sigmoid kernels.

Step 4: Set permitted deviation from the expected value in Numerical Tolerance. Tick the box next to Iteration Limit to set the maximum number of iterations permitted.

Step 5: Produce a report and Click Apply to commit changes.

4.4 k-Nearest Neibours (k-NN))

This algorithm of k-Nearest Neibours (k-NN) in statistics is a non-parametric classification method first developed by Evelyn Fix et. al. 1951 [10] and later expanded by Thomas Cover [11]. It is used for classification and regression. In both the cases, input consists of the k closest training examples in data set. The output depends on whether k-NN is used for classification or regression: The Orange kNN algorithm is:

Step 1: A name under which it will appear in other widgets. The default name is “kNN”.

Step 2: Set the number of nearest neighbors, the distance parameter and weights as model criteria. In this paper, the researcher used Euclidean distance between two points.

Step 3: The Weights you can use are:Uniform: all points in each neighborhood are weighted equally.

Distance: closer neighbors of a query point have a greater influence than the neighbors further away.

Step 4: Produce a report in the Test and Score window.

4.5 Neural Network Algorithm

McCullough and Walter Pitts first proposed neural network algorithm in the year 1944 [12]. A Multil- Layer Perceptron (MLP) algorithm is used with back propagation in orange data mining:

Step 1: A name under which it will appear in other widgets. The default name is “Neural Network”.

Step 2: Set model parameters:

Step 3 Produce a report and the box is ticked (Apply Automatically), the widget will communicate changes automatically.

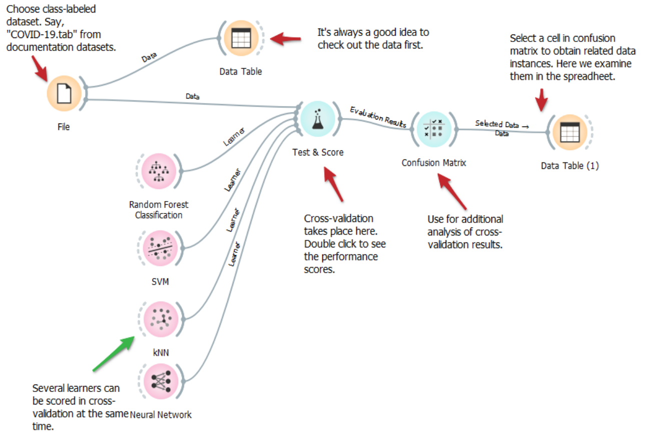

Choose class-labeled dataset. Say, "COVID-19.tab" from documentation datasets.
File
Data
Data Table
It's always a good idea to check out the data first.
Data
Learner
Test & Score
Evaluation Results
Confusion Matrix
Selected Data → Data
Data Table (1)
Select a cell in confusion matrix to obtain related data instances. Here we examine them in the spreadheet.
Random Forest Classification
Learner
SVM
Learner
kNN
Learner
Neural Network
Cross-validation takes place here. Double click to see the performance scores.
Use for additional analysis of cross-validation results.
Several learners can be scored in cross-validation at the same time.

Figure 1. Workflow of various Machine Learning Algorithms

Reddy Prasad suggests a system at his paper to classify the patient with heart disease based on some features. Therefore, the proposed system can predict the presence of heart problems on a person based on the given data. He used logistic regression technology to perform classification and prediction. Further used sigmoid function for the representation processed [13].

V. RESULT AND DISCUSSION

The k-means algorithms achieved five meaningful clusters and are labeled as five categories of States and Union Territories of India affected by COVID-19 (Table 2). And also the same result produced the cross validation machine learning algorithm in Table 4 to 7.

Table 2: Final Cluster of k-means Algoritm

Final Cluster Centers					
	Cluster				
	1 (Low)	2 (High)	3 (Very Low)	4 (Very High)	5 Moderate)
Conformed Cases	36553.44	947063.00	276334.50	2252057.00	607981.67
Deaths	499.39	9107.00	3020.80	52610.00	9984.67
Recoveries	35794.89	925840.75	270399.30	2099207.00	595758.33
Active Cases	259.17	12115.25	2914.40	100240.00	2238.67
Number of Case in each Cluster	10	4	18	1	3

Table 3 shows the various model accuracy of training data. This is a very high value of accuracy and it is reasonable to expect that the model would be useful to predict new, previously unknown, instance of COVID-19 problem. Rest of the data may be outlier due to data variation in the study period.

The confusion matrix can be used to define a number of performance criteria commonly used in model evaluation suggested by Ali, 2005 [14]. The following section describes various measures of the test score:

5.1 AUC (Area under Curve)

AUC stands for "Area under the ROC Curve." That is, AUC measures the entire two-dimensional area underneath the entire ROC curve from (0,0) to (1,1).

An **ROC** curve (Receiver Operating characteristic Curve) is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameter, they are True Positive Rate and False Positive Rate.

True Positive Rate (**TPR**) is a synonym for recall and is therefore defined as follows:

$$TPR = \frac{TP}{TP + FN}$$

False Positive Rate (**FPR**) is defined as follows:

$$FPR = \frac{FP}{FP + TN}$$

5.2 Classification Accuracy (CA)

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions and formally has the following definition:

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions}$$

Accuracy=Number of correct predictions shared by the Total number of predictions. For binary classification, accuracy can also be calculated in terms of positives and negatives as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives.

5.3 F1 Score

The F1 score is calculated by using the following formula:

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall} = \frac{TP}{TP + \frac{1}{2}(FP + FN)}$$

5.4 Precision

Precision is given by:

$$\frac{True\ Positive}{True\ Positive\ + False\ Positive}$$

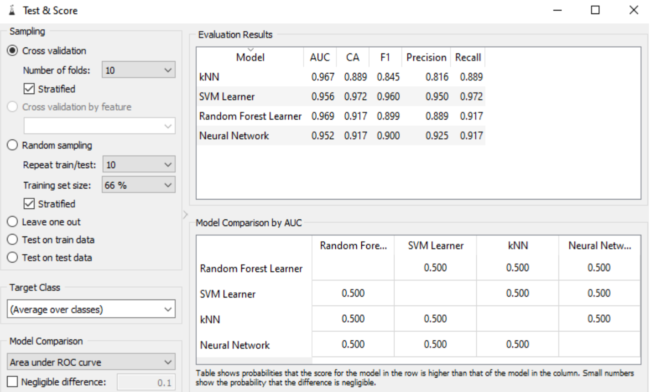
5.5 Recall

Recall Measure is computed by using the formula:

$$\frac{True\ Positive}{True\ Positive\ + False\ Negative}$$

The Models of AUC, CA, F1, Precision and Recall measure accuracy value closer to 1 is the best fitted model and closer to 0 is not a good fitted model . In this study all the measures are closter to 1 and these models are best fitted models using machine learning algorithms.

Table 3: Test Score of Various Algorim


Test & Score

Sampling

● Cross validation

Number of folds: 10

☑ Stratified

○ Cross validation by feature

○ Random sampling

Repeat train/test: 10

Training set size: 66 %

☑ Stratified

○ Leave one out

○ Test on train data

○ Test on test data

Target Class

(Average over classes)

Model Comparison

Area under ROC curve

☐ Negligible difference: 0.1

Evaluation Results

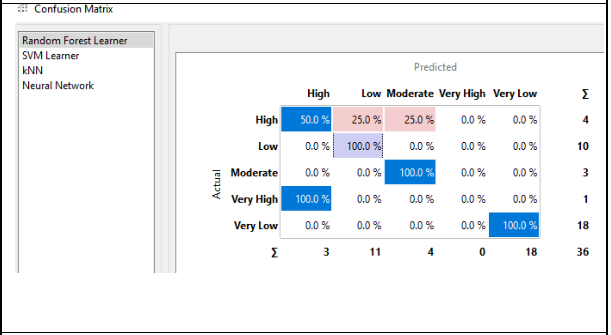
Model	AUC	CA	F1	Precision	Recall
kNN	0.967	0.889	0.845	0.816	0.889
SVM Learner	0.956	0.972	0.960	0.950	0.972
Random Forest Learner	0.969	0.917	0.899	0.889	0.917
Neural Network	0.952	0.917	0.900	0.925	0.917

Model Comparison by AUC

	Random Fore...	SVM Learner	kNN	Neural Netw...
Random Forest Learner		0.500	0.500	0.500
SVM Learner	0.500		0.500	0.500
kNN	0.500	0.500		0.500
Neural Network	0.500	0.500	0.500	

Table shows probabilities that the score for the model in the row is higher than that of the model in the column. Small numbers show the probability that the difference is negligible.

Table 4: Confusion Matrix of Random Forest Learner

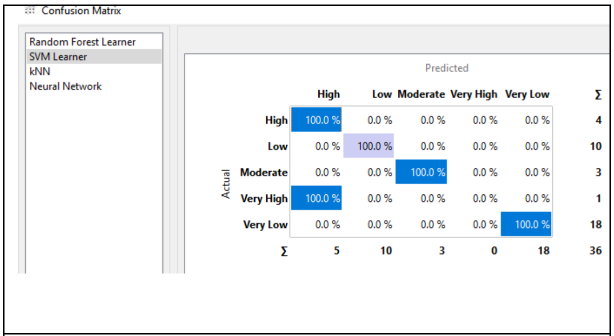

Confusion Matrix

Random Forest Learner
SVM Learner
kNN
Neural Network

Predicted

Actual	High	Low	Moderate	Very High	Very Low	Σ
High	50.0 %	25.0 %	25.0 %	0.0 %	0.0 %	4
Low	0.0 %	100.0 %	0.0 %	0.0 %	0.0 %	10
Moderate	0.0 %	0.0 %	100.0 %	0.0 %	0.0 %	3
Very High	100.0 %	0.0 %	0.0 %	0.0 %	0.0 %	1
Very Low	0.0 %	0.0 %	0.0 %	0.0 %	100.0 %	18
Σ	3	11	4	0	18	36

Table 5: Confusion Matrix of SVM Learner

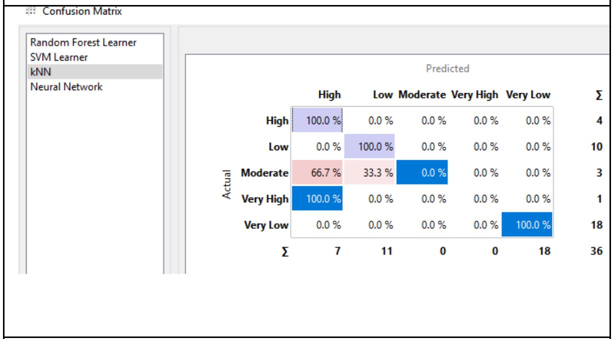

Confusion Matrix

Random Forest Learner
SVM Learner
kNN
Neural Network

Predicted

Actual	High	Low	Moderate	Very High	Very Low	Σ
High	100.0 %	0.0 %	0.0 %	0.0 %	0.0 %	4
Low	0.0 %	100.0 %	0.0 %	0.0 %	0.0 %	10
Moderate	0.0 %	0.0 %	100.0 %	0.0 %	0.0 %	3
Very High	100.0 %	0.0 %	0.0 %	0.0 %	0.0 %	1
Very Low	0.0 %	0.0 %	0.0 %	0.0 %	100.0 %	18
Σ	5	10	3	0	18	36

Table 6. Confusion Matrix of kNN

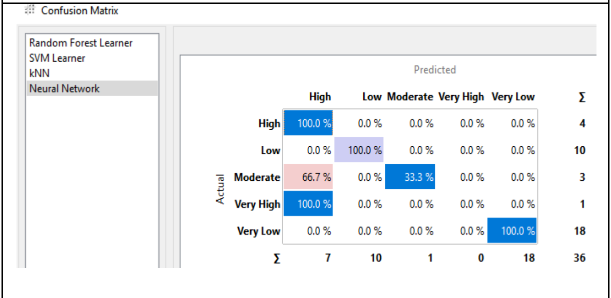

Confusion Matrix

Random Forest Learner
SVM Learner
kNN
Neural Network

Predicted

Actual	High	Low	Moderate	Very High	Very Low	Σ
High	100.0 %	0.0 %	0.0 %	0.0 %	0.0 %	4
Low	0.0 %	100.0 %	0.0 %	0.0 %	0.0 %	10
Moderate	66.7 %	33.3 %	0.0 %	0.0 %	0.0 %	3
Very High	100.0 %	0.0 %	0.0 %	0.0 %	0.0 %	1
Very Low	0.0 %	0.0 %	0.0 %	0.0 %	100.0 %	18
Σ	7	11	0	0	18	36

Table 7. Confussion Matrix of Neural Network

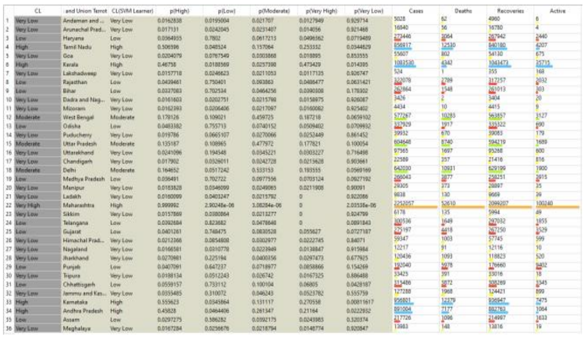

Confusion Matrix

Random Forest Learner
SVM Learner
kNN
Neural Network

Predicted

Actual	High	Low	Moderate	Very High	Very Low	Σ
High	100.0 %	0.0 %	0.0 %	0.0 %	0.0 %	4
Low	0.0 %	100.0 %	0.0 %	0.0 %	0.0 %	10
Moderate	66.7 %	0.0 %	33.3 %	0.0 %	0.0 %	3
Very High	100.0 %	0.0 %	0.0 %	0.0 %	0.0 %	1
Very Low	0.0 %	0.0 %	0.0 %	0.0 %	100.0 %	18
Σ	7	10	1	0	18	36

Table 8. Prediction and Probilities of States and Union Teritories



	CL	and Union Terrot	CL(SVM Learner)	p(High)	p(Low)	p(Moderate)	p(Very High)	p(Very Low)	Cases	Deaths	Recoveries	Active
1	Very Low	Andaman and ...	Very Low	0.0162838	0.0195004	0.021707	0.0127949	0.929714	5028	62	4960	6
2	Very Low	Arunachal Prad...	Very Low	0.017131	0.0242045	0.0231407	0.014056	0.921468	16840	56	16780	4
3	Low	Haryana	Low	0.0564935	0.7802	0.0617215	0.0496362	0.0719489	273446	3064	267942	2440
4	High	Tamil Nadu	High	0.506596	0.048524	0.157084	0.253332	0.0344829	856917	12530	840180	4207
5	Very Low	Goa	Very Low	0.0204079	0.0767549	0.0303868	0.018895	0.853555	55607	802	54130	675
6	High	Kerala	High	0.46738	0.0188569	0.0257398	0.473429	0.014395	1083530	4342	1043473	35715
7	Very Low	Lakshadweep	Very Low	0.0157718	0.0246623	0.0211053	0.0117135	0.926747	524	1	355	168
8	Low	Rajasthan	Low	0.0439461	0.750401	0.093863	0.0486477	0.0631421	322078	2789	317257	2032
9	Low	Bihar	Low	0.0327083	0.702534	0.0464256	0.0390308	0.178302	262864	1548	261013	303
10	Very Low	Dadra and Nag...	Very Low	0.0161603	0.0202751	0.0215798	0.0158975	0.926087	3426	2	3404	20
11	Very Low	Mizoram	Very Low	0.0162393	0.0206406	0.0217097	0.0160082	0.925402	4434	10	4415	9
12	Moderate	West Bengal	Moderate	0.178126	0.109021	0.459725	0.187218	0.0659102	577267	10283	563857	3127
13	Low	Odisha	Low	0.0483382	0.755713	0.0740152	0.0509402	0.0709932	337929	1917	335322	690
14	Very Low	Puducherry	Very Low	0.019786	0.0665107	0.0270066	0.0252449	0.861452	39932	670	39083	179
15	Moderate	Uttar Pradesh	Moderate	0.135187	0.108965	0.477972	0.177821	0.100054	604648	8740	594219	1689
16	Very Low	Uttarakhand	Very Low	0.0241096	0.194548	0.0345221	0.0303227	0.716498	97565	1697	95268	600
17	Very Low	Chandigarh	Very Low	0.017902	0.0326011	0.0242728	0.0215628	0.903661	22589	357	21416	816
18	Moderate	Delhi	Moderate	0.164652	0.0517242	0.533153	0.193555	0.0569169	642030	10931	629199	1900
19	Low	Madhya Pradesh	Low	0.036491	0.702722	0.0977556	0.0703124	0.0927192	266043	3877	258251	3915
20	Very Low	Manipur	Very Low	0.0183828	0.0346099	0.0249065	0.0211908	0.90091	29305	373	28897	35
21	Very Low	Ladakh	Very Low	0.0160099	0.0403247	0.0215792	0	0.922086	9838	130	9669	39
22	Very High	Maharashtra	High	0.999992	2.90248e-06	3.08284e-06	0	2.03538e-06	2252057	52610	2099207	100240
23	Very Low	Sikkim	Very Low	0.0157869	0.0380864	0.0213277	0	0.924799	6178	135	5994	49
24	Low	Telangana	Low	0.0392684	0.823682	0.0478648	0	0.0891843	300536	1649	297032	1855
25	Low	Gujarat	Low	0.0401261	0.748475	0.0830528	0.055627	0.0727187	275197	4418	267250	3529
26	Very Low	Himachal Prad...	Very Low	0.0212368	0.0854808	0.0302977	0.0222745	0.84071	59347	1003	57745	599
27	Very Low	Nagaland	Very Low	0.0166581	0.0310778	0.0223949	0.0138847	0.915984	12217	91	12116	10
28	Very Low	Jharkhand	Very Low	0.0270981	0.225194	0.0400356	0.0297473	0.677925	120436	1093	118823	520
29	Low	Punjab	Low	0.0407091	0.647237	0.0718977	0.0858866	0.154269	192040	5978	176660	9402
30	Very Low	Tripura	Very Low	0.0188134	0.0512243	0.026742	0.0167325	0.886488	33425	391	33016	18
31	Low	Chhattisgarh	Low	0.0559157	0.733112	0.100104	0.06805	0.0428187	315486	3872	308269	3345
32	Very Low	Jammu and Kas...	Very Low	0.0355485	0.310072	0.046243	0.0523782	0.555759	127288	1968	124421	899
33	High	Karnataka	High	0.555623	0.0345864	0.131117	0.270558	0.00811617	956801	12379	936947	7475
34	High	Andhra Pradesh	High	0.45828	0.0464406	0.261347	0.21164	0.0222932	891004	7177	882763	1064
35	Low	Assam	Low	0.0297275	0.586282	0.0392175	0.0243983	0.320374	217726	1096	214997	1633
36	Very Low	Meghalaya	Very Low	0.0167284	0.0256676	0.0218794	0.0148774	0.920847	13983	148	13816	19

Based on the above models, Machine Learning algorithm achieved better prediction and proximities in all the methods. On the whole ten percentages of prediction and proximities are misclassified due to different models and noisy data. The different machine learning models cross validation and classification accuracy are 88%, 97%, 91% and 91%. The Classification of States and Union Territories were named as Very Low Affected (VLA), Low Affected (LA), Moderately Affected (MA), Highly Affected (HA) and Very Highly Affected (VHA) States and Union Territories of India by COVID-19 cases.

Retrieval Number: 100.1/ijrte.A56590510121
DOI:10.35940/ijrte.A5659.0510121
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication

Maharashtra is correctly classified as Very High Affected States, Delhi, Uttar Pradesh and West Bengal falls in Moderately Affected States, Assam, Bihar, Chattisgarh, Haryana, Gujarat, Madhya Pradesh, Odisha, Punjab, Rajasthan and Telangana falls in Low Affected States, and Tamilnadu, Kerala Andhra Pradesh and Karnataka forms a group of highly affected States. Remaining States and Union Territories falls in Very Low affected by Covid-19 Cases (Table 8). The second wave of COVID-19 also started from March 2021. The government is taking necessary action to prevent and control the spread of COVID. Also our government advices our people to take vaccination over and above 45 years of age.

VI. CONCLUSION

In this section the researcher predicts and classifies the data of COVID-19 based on four machine learning algorithm with four major parameters namely confirmed cases, recoveries, deaths and active cases. The secondary sources of database were collected from Ministry of Health and Family Welfare Department (MHFWD), from Indian State and Union Territories up to March, 2021. Based on these background, the database classified and predicted various machine learning Algorithm, like SVM, kNN, Random Forest and Logistic Regression. Initially, k-means clustering analysis is used to perform and identified five meaningful clusters and is labeled as Very Low, Low, Moderate, High and Very High of four major parameters based on their average values. In addition the five clusters are cross validated using four machine algorithm and affected states are visualized in the table with help of prediction and probabilities. The different machine learning models cross validation and classification accuracy are 88%, 97%, 91% and 91%. The Classification of States and Union Territories were named as Very Low Affected (VLA), Low Affected (LA), Moderately Affected (MA), Highly Affected (HA) and Very Highly Affected (VHA) States and Union Territories of India by COVID-19 cases. Maharashtra is correctly classified as Very High Affected States, Delhi, Uttar Pradesh and West Bengal falls in Moderately Affected States, Assam, Bihar, Chattisgarh, Haryana, Gujarat, Madhya Pradesh, Odisha, Punjab, Rajasthan and Telangana falls in Low Affected States and Tamilnadu, Kerala Andhra Pradesh and Karnataka forms a group of highly affected States. Remaining States and Union Territories falls in Very Low affected by Covid-19 Cases (Table 8).

REFERENCES

1. Han J. Pei and M. Kamber (2011), Data Mining: Concepts and Techniques. Elsevier, 2011.
2. Padmavathi Janardhana et.al, (2015), Effectiveness of Support Vector Machines in Medical Data mining Journal Of Communications Software And Systems, Vol. 11, No. 1, Pp. 25-30.
3. Prediction of Affected and Deceased Population Trend of COVID-19 in India using Statistical Analysis, International Journal of Scientific and Innovative Mathematical Research(IJSIMR) Volume 8, Issue 2, 2020, PP 37-43.
4. A.B. M. Shawkat Ali and Saleh A. Wasimi (2016). Data Mining: Methods and Techniques, Cengage Learning India private Imitated, New Delhi.
5. Halgurd S. Maghdid, Kayhan Zrar Ghafoor, Ali Safaa Sadiq, Kevin Curran, and Khaled Rabie (2020). A novel AI-enabled framework to diagnose coronavirus covid 19 using smartphone embedded sensors: Design study. Computer Science, Cornel University. pp.1-7.
6. Manimannan G. et. al (2020), Clustering Study of Indian States and Union Territories by coronavirus (COVID-19) using k-mean algorithm, International Journal of Data Mining and Emerging Technologies, Vol.9 Issue-2, pp.43-51.
7. MacQueen (2015), Applied Multivariate Statistical Analysis, Prentice Hall, New Delhi
8. Breiman, L (2001), Random Forests. Machine Learning **45,** 5–32.
9. H. Drucker, Donghui Wu and V. N. Vapnik (1999), "Support vector machines for spam categorization," in *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 1048-1054
10. Evelyn Fix and Joseph L Hodges Jr (1951). Discriminatory analysis-nonparametric discrimination: consistency properties. Technical report, DTIC Document.
11. Thomas M Cover and Peter E Hart (1967). Nearest neighbor pattern classification. IEEE Transactions on Information Theory, 13(1):21-27.
12. McCulloch, W. S. & Pitts, W. (1943) BulL Math Biophys. 5, 115-133.
13. R. Prasad, P. Anjali, S. Adil, and N. Deepa, “Heart Disease Prediction using Logistic Regression Algorithm using Machine Learning,” Mach. Learn., vol. 8, no. 3, p. 4.
14. S. Ali. And S. Wasimi (2005). Data Mining: Methods and Techniques, Cengage Learning India.

AUTHORS PROFILE

P. Arumugam., Department of Statistics, Monomaniam Sundaranar University, Tirunelveli.

V. Kadhirveni, Department of Statistics, Manonmaniam Sundaranar University, Tirunelveli

R. Lakshmi Priya, Department of Statistics, Dr. Ambedkar Govt. Arts College, Vyasarpadi, Chennai

Manimannan G., Department of Statistics, TMG College of Arts and Science, Chennai