IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Study of COVID-19 Cases in India by Using Python Based Support Vector Regression Model in Machine learning

Ankesh Gupta
Department of Computer science & Engineering
Vivekananda Institute of Technology Jaipur, Rajasthan, India

Madhav sharma
Department of Computer science & Engineering
Vivekananda Institute of Technology Jaipur, Rajasthan, India

Tarun Agrawal
Department of Computer science & Engineering
Vivekananda Global University Jaipur, Rajasthan, India

Jyoti Sharma
Department of Computer science & Engineering
Vivekananda Institute of Technology Jaipur, Rajasthan, India

Abstract- ***The Eruption of the Novel Corona virus or the COVID-19 in various parts of the world has been affected by the epidemic. The world as a whole and caused millions of death number this remains an inauspicious caveat to public health and will be stained as one of the greatest pandemics in world history. The proposed work utilizes a support vector regression model to predict the total number of the case found total deaths and recovered cases, the cumulative number of confirmed cases and several daily cases. The data is collected for the time period of 1st March to 30th April (61 Days). The total number of cases as of 30th April is found to be 35043 confirmed cases with 1147 total deaths and 8889 recovered patients. The model was developed in Python 3.6.3 to obtain the predicted values of the aforementioned cases until 30th June. The proposed methodology is based on a prediction of values using a support vector regression model with Radial Basis Function as the kernel and 10% confidence interval for the curve fitting. The data has been split into train and test set with test size 40% and training 60%. The model performance parameters are calculated as mean square error, root means square error, regression score, and percentage accuracy. The sample has above 97% accuracy in predicting deaths, recovered, the cumulative number of confirmed cases, and 87% accuracy in predicting daily new cases. The results suggest a Gaussian decrease in the number of cases and could take another 4 to 5 months to come down to the minimum level with no new cases being reported. The method is very efficient and has higher accuracy than linear or polynomial regression. In addition, this paper also analyses the current trends or patterns of Covid-19 in India. With the help of the Indian Ministry of Health and Family Welfare dataset, this study proposes different trends and patterns experienced in different parts of the world.***

Keywords – *COVID19, Data analysis, Machine learning, Python, Support vector regression*

I. INTRODUCTION

The spread of coronavirus disease 2019 (COVID-19) has become a global admonition and the World Health Organization (WHO) declared COVID-19 as a global pandemic on March 11, 2020 [1]. As of June 6, 2021, there are 17.4 Cr confirmed cases and 37.4L deaths from COVID19 worldwide[7] (https://coronavirus.jhu.edu/data/new-cases). The COVID-19 pandemic has been very affecting people's lives and the world's economy. Among many infection-related questions, governments and people are major concerned with (i) when will the COVID19 infection rate reach the maximum (ii) how long the pandemic will take to stop spreading and (iii) What could be the total number of individuals that will lastly be infected (iv) what will be the total number of deaths [4]. The questions are of main concern in India also, a country with high population density and economic diversity. The spread of the disease in India is broadly lower than that of China, USA, and other European countries. India is under complete lockdown since 21st March 2020 and experts believe that this could be harmful in slacken the COVID19 spread among its citizens. Currently, the development of vaccines is still in progress and there are no dominant antiviral drugs for treating COVID-19 infections. As of June 6, 2021, the total number of

COVID19 cases in India is 2.9 Cr and 3.51 L have died due to Severe Acute Respiratory Syndrome (SARS) (https://www.mohfw.gov.in/). The total number of COVID19 recovered individuals in India is 2.73 Cr to date.

The lockdown is harshly affecting the poor and migrant labourers. Staying at home may not be a practicable option in the near future since a lot of people may die out of hunger and other ailments. News media reports all over the world are reporting about the crisis and how it is affecting the lives of people. Much research is being put an end to at all levels to quickly gather information, develop mitigation tools and methods to implement the same. Therefore policymakers and authorities want to have an Entire view of the current circumstances and want to visualize the extent to which it can dispersion in the near future for informed policy-making and deciding the next steps of course.

The paper here discusses about the Python Based Support Vector Machine Model of COVID19 spread in India using support vector regression implemented in Python.3.6. The steps of the model are discussed in the methodology section with subsequent analysis. The results are shown and discussed. The authors conclude the overall purpose of the work in Conclusion.

METHODOLOGY

In case of two-dimensional image, after a DWT transform, the image is divided into four corners, upper left corner of the original image, lower left corner of the vertical details, upper right corner of the horizontal details, lower right corner of the component of the original image detail (high frequency). You can then continue to the low frequency components of the same upper left corner of the 2nd, 3rd inferior wavelet transform.

A. Preparation of the dataset

The .csv file of Novel Corona virus 2019 dataset available at https://www.kaggle.com/sudalairajkumar/novel-corona-virus-2019-dataset is downloaded. A separate .csv file is created from the global dataset only for India. The columns include Total Deaths, Total Recovered, and a Total number of confirmed COVID19 patients on day to day basis from 1st March 2021 to 30th April 2021 (61 days). All the data is in Accumulative form. From the Accumulative dataset, we have computed the disparity time series to get the values based on a daily new case basis. So we have now extended our dataset to have six columns 3 for Accumulative cases and 3 for respective daily new cases of deaths, recovery, or confirmed COVID19 individuals.

B. Data preprocessing

In the data preprocessing section, we have set the columns created above as the dependent variable column (y) and the number of days starting from 1st March as the independent variable (X). X column is basically a numpy array of elements 1 to 61. The X and y is then reshaped to be a column vector of size 61 (i.e. 61 rows, 1 column)

The dataset is split for Training (60%) and Test (40%) using train_test_split() function imported from class model selection of sklearn python library. The training and testing variables are saved for further appraisal.

The training and testing variables of both X and y are standardized using StandardScaler() object imported from class pre-processing of sklearn python library. Separate objects have been created for standardization of X and y data. The fit_transform() function is used to fit the object into the data and transform the values of X and y in standard form ranging from -3 to +3. The scaled data is now fit for regression application.

C. Support vector regression

Support vector regression is a popular choice for prediction and curve fitting for both linear and non linear regression types. SVR is based on the elements of Support vector machine (SVM), where support vectors are basically closer points towards the generated hyperplane in an n-dimensional feature space that distinctly seggregates the data points about the hyperplane. More discussions on the SVR and SVM can be found on [[3,2,6]]. The SVR model performs the fitting as shown in Fig. 1 . The generalized equation for hyperplane may be represented as $y = wX + b$, where w is weights and b is the intercept at $X = 0$. The margin of tolerance is represented by epsilon ε. The SVR regression madel is imported from SVM class of sklearn python library. The regressor is fit on the training dataset. The model parameter as chosen here for analysis is shown below.

Support vector regression is a popular preference for prediction and curve fitting for both linear and non-linear regression types. SVR is based on the elements of the Support vector machine (SVM), where support vectors are basically closer points towards the generated hyperplane in an n-dimensional feature space that distinctly disassemble the data points about the hyperplane. More discussions on the SVR and SVM can be found on [3,2,6]. The SVR model performs the fitting as shown in Fig. 1. The generalized equation for hyperplane may be represented as $y = wX + b$, where w is for weights and b is for the intercept at $X = 0$. The margin of tolerance is represented by epsilon ε. The SVR regression model is imported from the SVM class of sklearn python library. The regressor is fit on the training dataset. The model parameters chosen here for analysis are shown below.

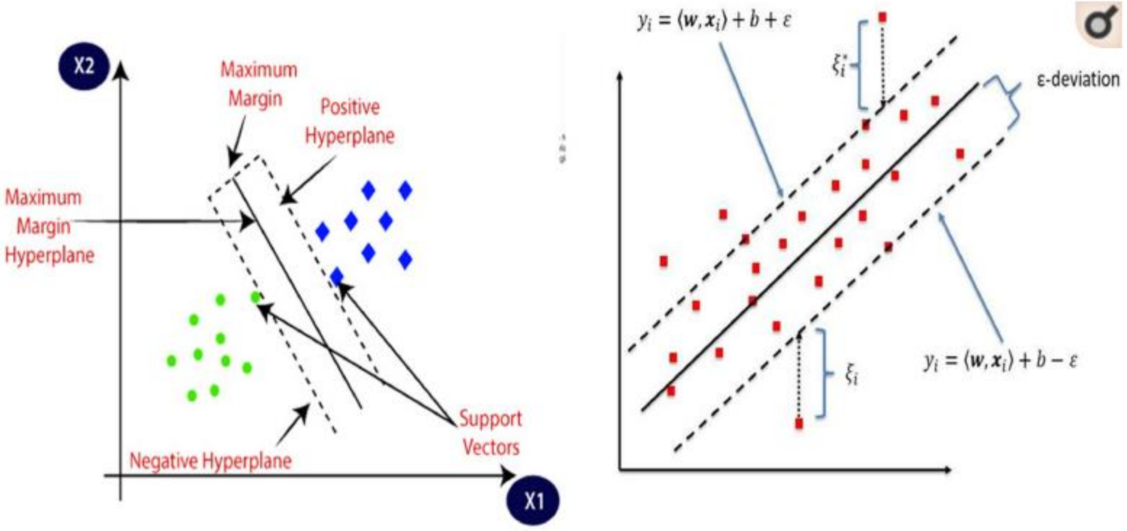
X2
Maximum Margin
Positive Hyperplane
Maximum Margin Hyperplane
Negative Hyperplane
Support Vectors
X1
$y_i = \langle w, x_i \rangle + b + \varepsilon$
ξ_i^*
ε-deviation
ξ_i
$y_i = \langle w, x_i \rangle + b - \varepsilon$

Fig. 1: Support vector regression model for linear regression fitting where X1= X and X2 = y are the features and label in our case. [Image credit: https://www.researchgate.net/figure/Schematic-of-the-one-dimensional-support-vector-regression-SVR-model-Only-the-points_fig5_320916953]

SVR(C=1.0, cache_size=200, coef0=0.0, degree=3, epsilon=0.1, gamma='auto', kernel='rbf', max_iter=-1, shrinking=True, tol=0.001, verbose=False)

D. Visulization

The regression fitting of the data with predicted values of the test data is plotted using the scatter plot function imported from matplotlib python library. The actual points and the predicted points are shown in Fig. 2 for all the respective conditions. [5]

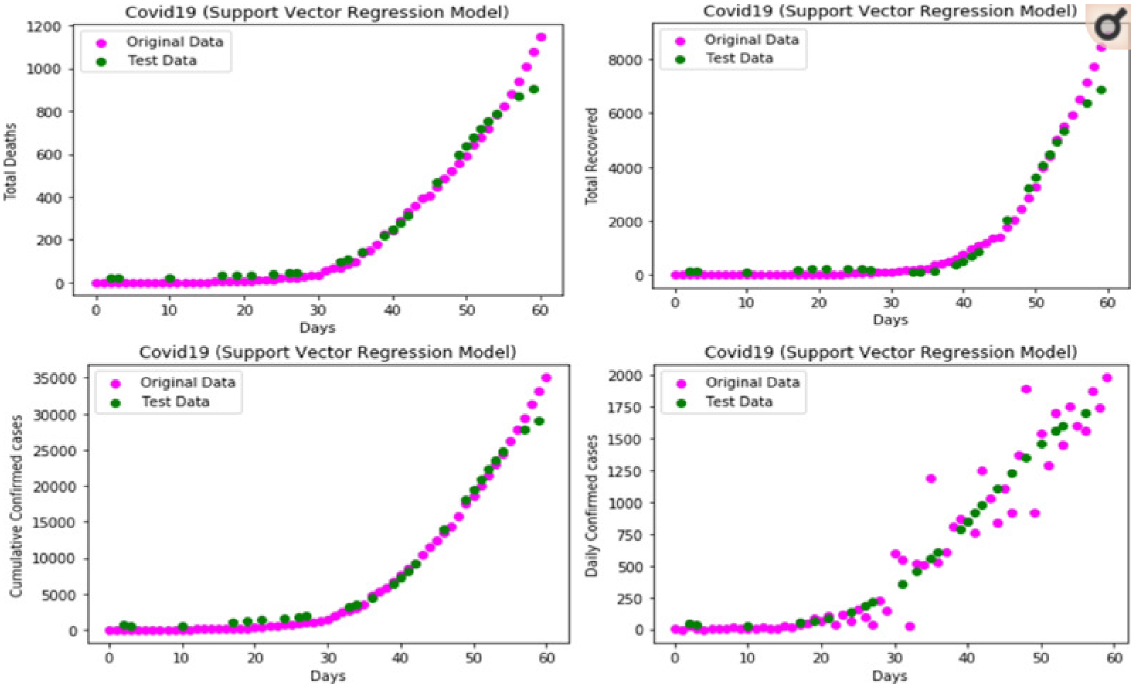

Fig. 2 : The figures shown here are the plots of regression fit with the data for total deaths, total recovered, cumulative confirmed cases and daily confirmed cases (in clockwise direction) [5]

E. Model performance evaluation

The model performance parameters are then evaluated to check for the reliability in predicting the outcome. The mean square error (MSE), root mean square error (RMSE), R^2 score and percentage accuracy are calculated and shown in Table 1 .

Table 1: The support vector regression model performance parameters with RBF kernel and 10 % fitting confidence interval [5]

Data	MSE	RMSE	Reg. score	% Accuracy
Total deaths	0.00849	0.09214	0.98681	99%
Total recovered	0.03028	0.17403	0.97343	97%
Daily confirmed	0.1094	0.33083	0.87490	87%
Cumulative confirmed	0.01285	0.11338	0.98861	99%
Daily deaths	0.13087	0.36172	0.82182	82%

F. Prediction

The prediction of the future values of the time series involves some steps of data manipulation to get the cumulative trend so as to match the original dataset trend of the past. The past dataset is in Accumulative form, but since we have implemented the RBF kernel in our model, it is quite obvious that the predicted time series would be decreasing the gaussian trend. The decreasing trend can be preserved by a modification as discussed below. We have implemented some steps in the algorithm that could help us reach our objective.

Here we have obtained the predicted time series for each case separately for 60 more days that start just after 30th April or 61st day from the starting. Therefore, we wish to merge the 60 days prediction with the past 61 days. The predicted column consists of decreasing values. So, we have computed the difference of the time series and then used absolute values of the difference time series. The difference time series gets inverted and gives us a rising trend, which saturates after certain values. Then we performed an Accumulative sum of the elements of the time series and added the max value of the past time series to it. This helps us in preserving the trend and visualizing it in cumulative form. The plots of the past and forecasting values are shown in Fig. 3 and Fig. 4

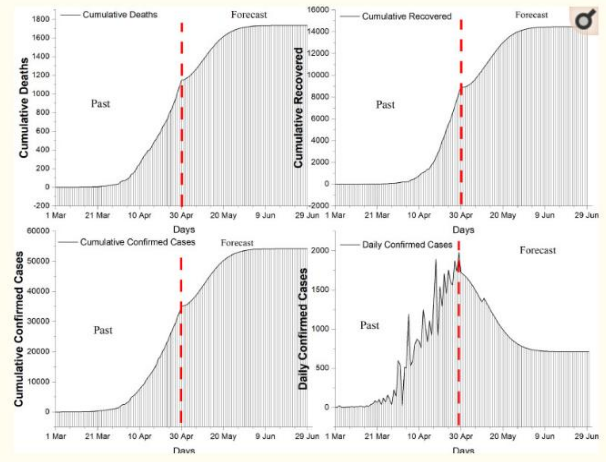

Fig 3: The past and forecast of the total deaths, total recovered, cumulative confirmed and daily confirmed cases of COVID19 patients in India. [Past: 1st Mar to 30th April; Forecast*: 1st* May to 30th June] [6]

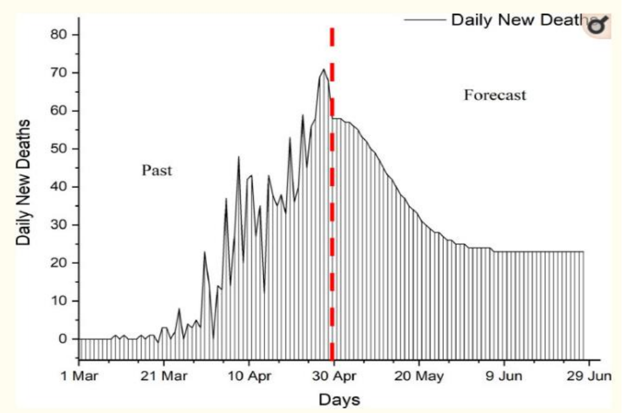

Fig 4: The past and forecast of the daily number of deaths [6]

This variation is not required for the prediction of time series of daily new cases analysis. All the necessary codes used in the appraisal of the above-mentioned steps are uploaded to the GitHub repository for further use and improvisation. The link is https://github.com/DebanjanParbat/Support-Vector-Regression

RESULTS AND DISCUSSION

The results show that the model performed well in fitting the accumulative cases while poor-fitting is observed in the case of a daily number of cases. The daily data show that there are many spikes that reduce the accuracy of the predictability of the model. The model predicts that the total number of infected persons may cross the 2.91 Cr in India. If the current rate of daily new cases prevails, by the second week of June. The total number of people that can die based on the recent trends predict that it can surpass the 2600 mark within the second week of June. [7]
Moreover, if more spikes are in daily new cases and daily deaths then the total number of infected persons may rise and there could be more delay in attaining flattening of the curve. The spikes induce non-stationary in the dataset making it difficult for regression models to accurately predict. But we can say, that if in near future the spikes are controlled with strict physical distancing and containment measures then the flattening of the curve can be achieved by the end of 2nd week of June. [4]

CONCLUSION

The proposed methodology predicts the total number of COVID19 infected cases, total number of daily new cases, total number of deaths, and the total number of daily new deaths. The total number of recovered individuals is also predicted. Based on the recent trends, the future trends have been predicted using a robust machine learning model, the support vector regression. The SVR has been reported to outperform the consistency in predictability with respect to other linear, polynomial and logistic regression models. The variability in the dataset is addressed by the proposed methodology. The model has above 97% accuracy in predicting deaths, recovered, cumulative number of confirmed cases, and 87% accuracy in predicting daily new cases. The disease spread is significantly high and if proper containment measures with physical distancing and hygienity is maintained then we can reduce the spikes in the dataset and hence lower the rate of progression.

REFERENCES

1. Boccaletti S. Modeling and forecasting of epidemic spreading: the case of COVID-19 and beyond. *Chaos Solitons Fractals.* 2020 [PMC free article] [PubMed] [Google Scholar]

2. Drucker H. Advances in neural information processing systems. MIT Press; 1997. Support vector regression machines; pp. 155–161. [Google Scholar]
3. Hastie T.J. Springer; New York: 2008. The elements of statistical learning: data mining, inference, and prediction. [Google Scholar]
4. Li L. Propagation analysis and prediction of the COVID-19. Infect Dis Model. 2020:282–292. [PMC free article] [PubMed] [Google Scholar]
5. Matplotlib Documentation. 2020 [Google Scholar]
6. Scikitlearn.(2020). https://scikitlearn.org/stable/auto_examples/svm/plot_svm_regression.html