Predicting COVID-19 in India using Machine Learning

Ameya Laad, Heth Sheth, Kushal Shah, and Nandish Chheda

BITS Pilani, Goa Campus

April 30, 2021

Abstract

Coronavirus, or COVID-19 took the world by storm in 2020. Many countries have had thousands of cases, with hundreds of deaths. All around the worlds, efforts have been undertaken to try and accurately model the spread of COVID-19. In this experiment, we attempt to aid in modelling and predicting the daily number of cases with the help of different machine learning as well as deep learning techniques. We also explore the possibility of lockdown having an impact on the number of cases. We collected data from different sources to create our dataset consisting of mobility data, government policies and daily cases all around the world.

Keywords: COVID-19, Machine Learning, Linear Regression, LSTM

1 Introduction

Machine Learning has been highly effective in solving complex problems in recent times. Problems that require making predictions and forecasting can be solved using ML. Some examples of these problems are: weather forecasting, disease forecasting, stock market forecasting and disease prognosis.

Covid-19 has become a serious threat to humanity and as of April 2021, it will be almost 1.5 years since it has affected the world. The virus originated from Wuhan, China in late 2019 when people started showing symptoms like pneumonia. Almost 150 million have been affected by the virus and thus far it has claimed 3.15 million lives across the globe. 87 million people have also recovered. A lot of damage has been caused by this virus and almost all countries have been or are under attack of a second wave of the virus which has been stronger and lethal than the first wave. The virus spreads through respiratory droplets of an affected person or through touching contaminated surfaces. One of the shocking and daunting aspects of this virus is that an infected individual can be asymptotic for several days and before getting diagnosed the virus may have been spread to other people.

In the last 1.5 years, the whole world had to invest a lot of resources in tackling the pandemic. This included boosting research in the hope of finding an effective vaccine for the virus and also the introduction of lockdowns to curb the spread by reducing social gatherings. There is no fixed medication for treating infected patients therefore it is imperative to search for ways to reduce the spread and to understand how applying various restrictions will affect the spread.

The aim of this project is to analyze the first wave of coronavirus spread in India and to build models which give predictions for a future number of daily cases for the second wave. The models built in this project inspect several aspects of the first wave and attempt to give accurate predictions.

2 Dataset

2.1 Data Collection

We collected data for daily cases as well as other factors such as mobility, vaccination and government policies. Due to the widespread nature of the pandemic, enormous amounts of data has

been collected worldwide and made readily available in csv formats. The features that we decided to collect can be broadly classified into the following:

2.1.1 Confirmed cases, Recoveries, Deaths and vaccination data

The daily and cumulative number of confirmed cases, recoveries and deaths in India were collected from COVID19-India API[1].
For the worldwide data, the data was taken from Our World In Data [2]

2.1.2 Google - COVID-19 Community Mobility Reports:[3]

Google's Community Mobility Reports are broken down by location and displays the change in visits to places like grocery stores and parks. Each value is the percentage change from the baseline (equal to 0). The different features available are:

- Retail & recreational mobility: Mobility towards places like restaurants, cafes, shopping centers, museums, libraries, and picture theaters are named as retail & recreational mobility
- Grocery & pharmacy mobility: Daily or sometimes weekly mobility trends for places viz. grocery, food warehouses, markets, local hats, farmer's markets, specialty food shops, different drug or medicine stores, and pharmacies
- Parks mobility: Mobility trends for places of attraction like local parks, national parks, public beaches, marinas, dog parks, plazas, and public gardens
- Transit stations mobility: This mobility refers to the process by which a person moves from one place to another place like public transport hubs such as subway, bus, and train stations.
- Workplaces mobility: This type of mobility trends for going places of work from a native place.
- Residential mobility: Mobility in the direction of places of residence where a person lived.

2.1.3 Apple - Walking, Driving Mobility: [4]

Apple's mobility data shows a relative volume of directions requests per country/region, sub-region or city compared to a baseline volume on January 13th, 2020. The base value was set to 100.

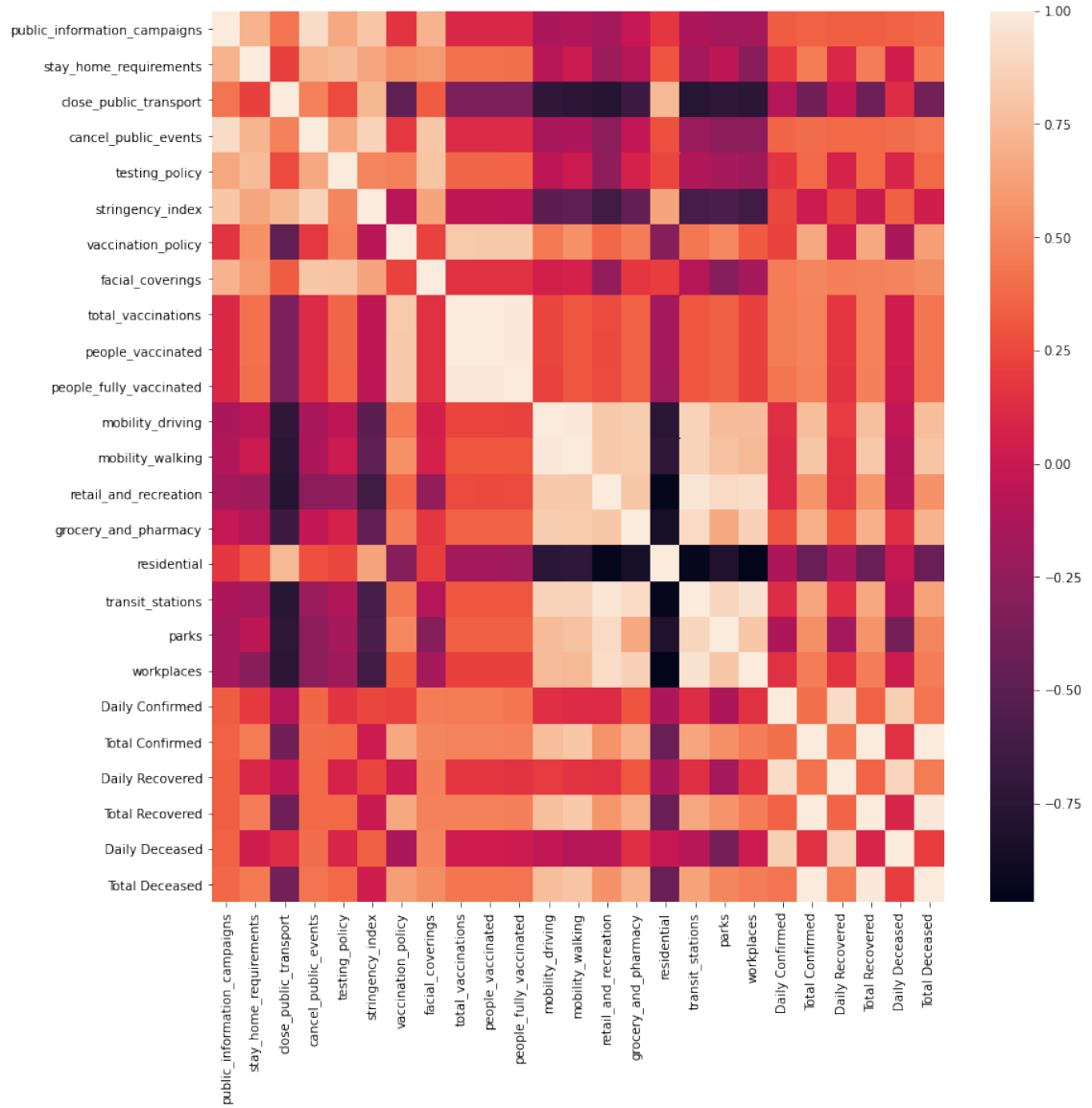
2.1.4 Government Policy Responses:

- Stringency_index: This is a composite measure based on nine response indicators including school closures, workplace closures, and travel bans, rescaled to a value from 0 to 100 (100 = strictest).
- Stay-home-requirements: categorizes nations into following "No requirement" , "Recommended", "Required (except essentials)" and "Required (with few exceptions)".
- Public_information_campaigns: classified into three levels - "None" "Public officials urging caution" "Coordinated information campaign"
- Close_public_transport: three categories based on need for cancellation of public transport
- Cancel_public_events: three categories based on need for cancellation of public events.
- Testing_policy: four categories based on the availability of covid testing
- Vaccination_policy: four categories based on availability of vaccines for the key workers group/ clinically vulnerable group and elderly group.
- Facial_coverings: four levels based on need for facial covering.

2.2 Data Cleaning and Feature Selection

We encountered various issues with our data which needed to be taken care of. Firstly, many data points were missing. Missing values of cases, recoveries or deaths were usually in the starting phase and hence were set to 0. Missing values of all other features were set to the mean of that particular feature with the exception of vaccinations whose values were set to the value of the previous day. The timeline of our data was from 13-01-2020 to 12-04-2021.

The last step related to the data processing was the selection of features to include in our input vectors. As the focus of this study was to predict daily cases and analyze the impact of lockdown on daily cases, we decided to select features which would have a role in these predictions. We decided not to include features which were categorical with respect to time as these would not have a direct impact on daily cases. These included most of the government policy categorical features except stringency index.



Pearson Correlation Matrix

2.3 Data Visualization

2.3.1 Mobility data

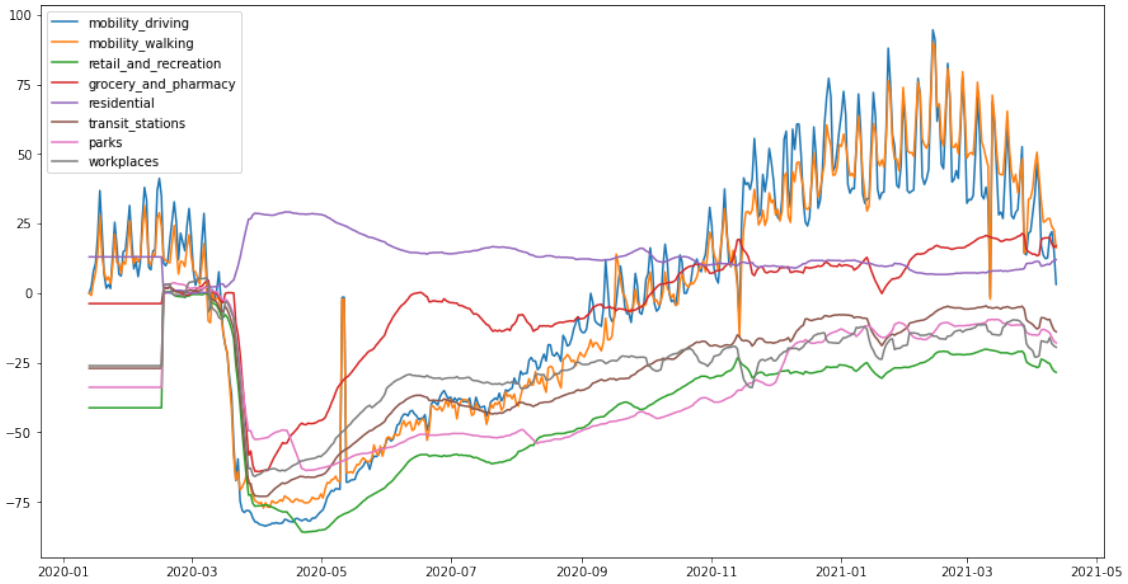


Figure 1: Mobility data

The dates of lockdown are highlighted in the background of the mobility data graphs. It can be clearly observed that lockdown in the country led to a steep decrease in outdoor mobility data. As time went on and lockdown restrictions started being eased, a gradual rise in mobility can be observed.

2.3.2 Confirmed cases

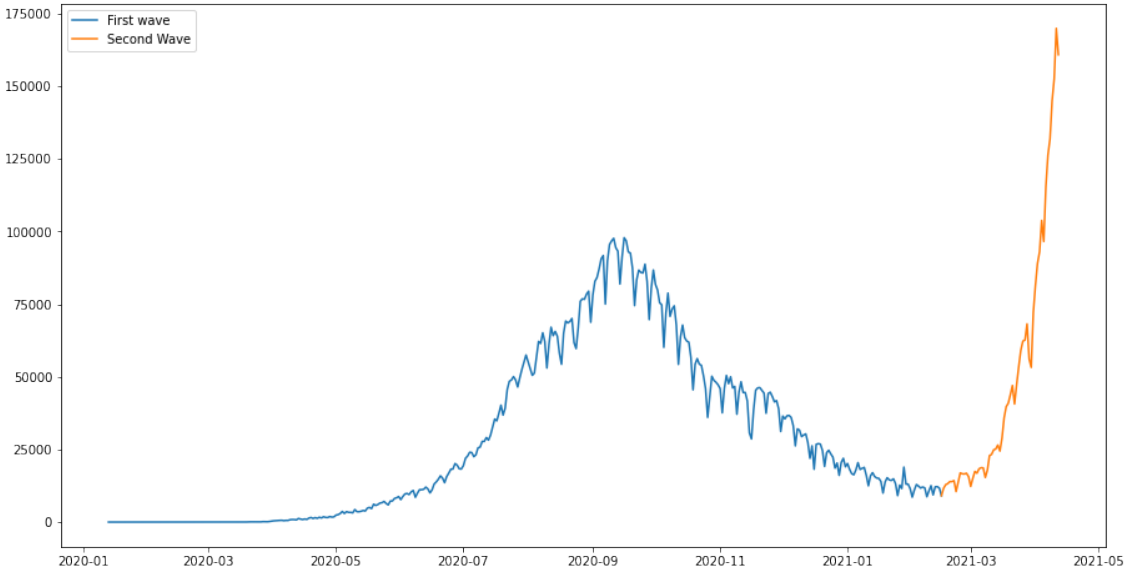


Figure 2: Daily confirmed cases in India

2.3.3 Recovery rate vs Deceased rate

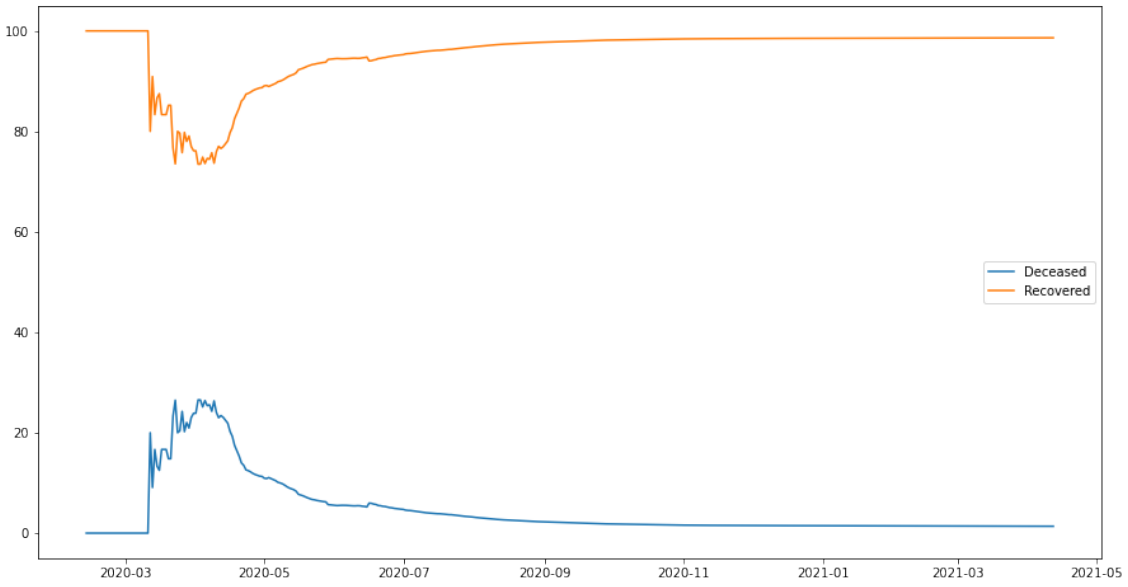


Figure 3: Recovery rate vs Deceased rate

We can observe from the graph that in the initial stages the deceased rate was quite high which can be attributed to limited knowledge on possible medications to reduce the impact of the virus. This gradually started decreasing and remained around 1.4% while the recovery rate increased up to 98.4%.

3 Methodology

Our study mainly involves 2 major areas:

- Prediction of COVID 19 cases:
  For our first task, we tried various regression and deep learning models and testing them on their predictions of the 2nd wave. The classifiers which outshined among the others were:- Linear regression, ARIMA, HMM, LSTM, Encoder-Decoder LSTM.
- Analysis of Impact of Lockdown on COVID 19 cases
  For our 2nd task, we focused on using Linear Regression model to show the impact of increased lockdown on number of cases.

3.1 Linear Regression

To find the relationship between dependent and independent variables in a dataset Regression models are used. Linear regression is a type of a regression model best suited for predictive analysis. There are two factors (x,y) that are involved in linear regression analysis. The equation below shows how y is related to x known as regression:

$$y = b_0 + b_1 x + e$$

Here 'e' represents the variability in the values between x and y, 'b0' is the intercept and 'b1' is the slope vector. Model x is input as the training dataset for the target feature which is y. The job of a linear regression model is to find the best values for b0(intercept) and b1(coefficient) to get the best-fit regression line.

3.2 ARIMA

For time series forecasting AutoRegressive Integrated Moving Average models have been highly accurate. Before building a ARIMA model one has to find out whether the given timeseries is stationary and seasonal.

Three parameters used in summarizing an ARIMA model are the AR parameter p, integration parameter d, and MA parameter q. Parameters p and q denote the order of AR and MA, while d denotes the degree of differencing the series to obtain stationarity. In this project as target feature is a dependent feature we have used a multivariate model in which independent features are passed to model as exogeneous variables. We used the auto-arima library which hypertunes the parameters to find the best fit.

3.3 Hidden Markov Models (HMM)

A Hidden Markov Model (HMM) is a finite state machine. It provides a probabilistic framework for modelling a time series of multivariate observations. HMMs are capable of predicting and analysing time-based phenomena and have found applications in fields such as speech recognition, natural language processing, and financial market prediction. We thus tried to explore the covid time series data with HMMs.

HMMs can model hidden state transitions from "observable" sequential data. These hidden states can be thought of as causal factors in our probabilistic model. They are generative models that model the joint distribution of observations, hidden states or equivalently the prior distributions of hidden states (the transition probabilities) and the conditional distribution of observations given hidden states (the emission probabilities).

A GaussianHMM model (hmmlearn python library) was used with six hidden states and a "full" covariance matrix for the state's Gaussian distributions (manually chosen based on performance on test set). The model uses a sliding window approach to make predictions for a given day. This window-size parameter defines the number of previous observations we want to base our predictions on. Thus, the model takes a series of observations as input and outputs the most likely observation for the following day.

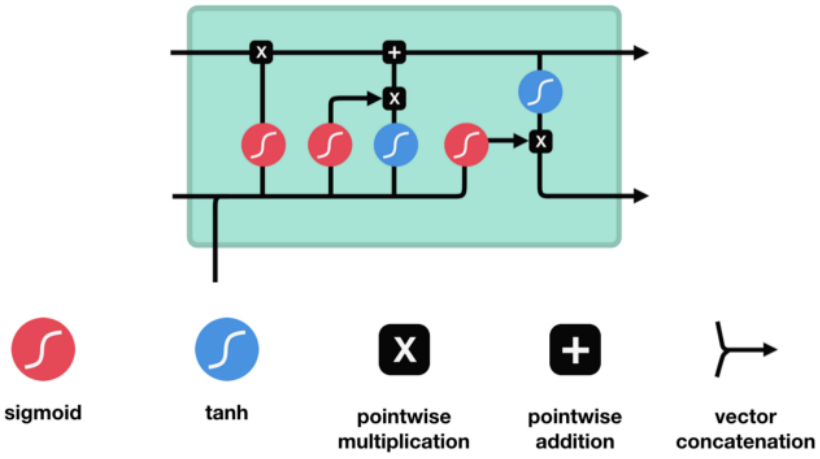


Figure 4: A LSTM cell [5]

3.4 Long-Short Term Memory (LSTM) Models

LSTM networks are well-suited for classifying, processing, and making predictions based on time series data. While traditional RNN's struggle to understand long-term dependencies, LSTM's are

capable of learning these. LSTM's have found huge success wherever a component of time-based memory comes into play, and are now superseded by attention-based models.

One cell of an LSTM layer looks as in Figure 1. The upper line could be said to represent the "long-term" memory of the LSTM. The black boxes represent gates which regulate the flow of information passed between each LSTM block.

Recurrent models also have a timestep parameter. This parameter can be varied to determine the number of steps n time the model sees before it makes a prediction. Similar to the reset of the models, this parameter is varied between 7, 10, and 14 days. In order to ensure that the weekly seasonality of the data does not come into play, the model is trained and tested on a 7-day rolling average.

Multiple runs of the model were tried out, including but not limited to: using mobility data from Apple and Google, using different model architectures, varying the timesteps to consider longer periods of time, using the data for other countries to attempt to predict India's lockdown impact and number of new_cases.
More information on the runs can be found on wandb, which we have used to collect all the runs we have done.

3.5 Seq2Seq LSTM Model

Encoder decoder models such as the Seq2Seq model have been traditionally used to great effect in natural language processing tasks. Recent studies have shown that they can prove highly effective when dealing with time series data as well. We use a similar architecture built with 2 LSTM units in the encoder and the decoder as can be seen in figure 5.

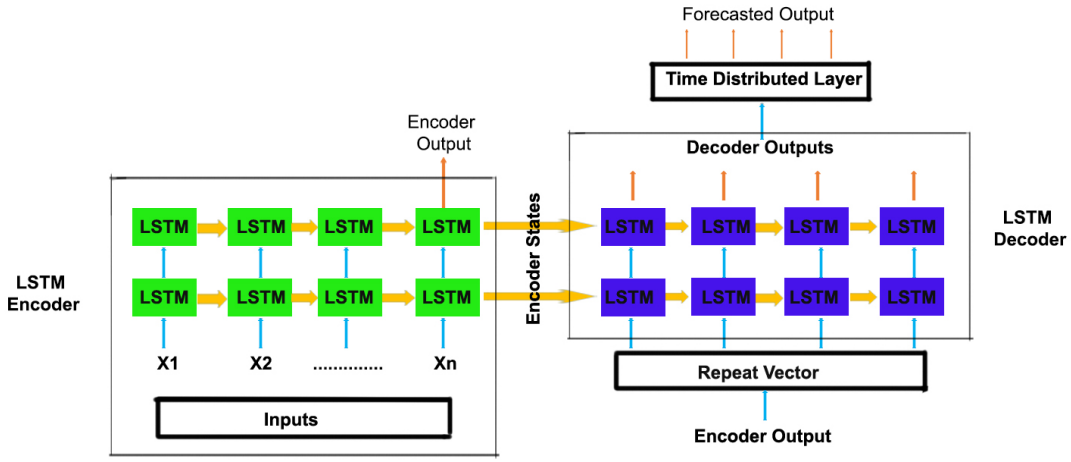


Figure 5: A 2-layered encoder-decoder model [6]

The input data is fed in the format of (samples, time-steps, features) i.e. the training data is transformed into samples having the previous n time-step data for given feaatures. The encoder consists of 2 LSTM layers of 256 units each which convert the given input to a context vector. This context vector is fed as an input to the decoder along with the final encoder state.

In order to forecast m future days, a repeat vector layer repeats the encoder ouput m times. The decoder also has 2 LSTM layers of 256 units each which are followed by the time distributed layer which consists of fully connected layers to seperate the m future outputs. However, we found that as we predict more days in the future, the accuracy decreases. Hence we focus on predicting the next day.

4 Results

We trained on the first wave of COVID-19 in India which we considered to be the first 400 days of our data (13/01/2020 to 15/02/2021) and predicted the results for the second wave (16/02/2021 to 12/04/2021).

4.1 Choice of Metric

The task of prediction of daily confirmed cases can be classified as a time-series prediction task. The aim of the models is to be able to predict the trends in daily cases as close to the actual trends. For this purpose, the 2 metrics used in judging the performance of the models were Mean Absolute Error and Root Mean Square Error.

4.1.1 Mean Absolute Error (MAE)

Mean absolute error is defined as the mean of the absolute differences between the predicted values and true values.

$$mae = (\frac{1}{n}) \sum_{i=1}^{n} |y_i - x_i| \tag{1}$$

4.1.2 Root Mean Squared Error (RMSE)

RMSE is defined as the square-root of the mean of the squared differences between the predicted values and true values.

$$rmse = \sqrt{(\frac{1}{n}) \sum_{i=1}^{n} (y_i - x_i)^2} \tag{2}$$

4.2 Confirmed cases predictions

We first focused on using only daily confirmed cases data to predict the future confirmed cases. The independent variables for inputting in the model were the no. of new cases of the last 'n' days. Here n could be tuned to study (values for n tested were: 7,10,14).

Loss	Mean Absolute Error			Root Mean Squared Error		
Days (n)	7	10	14	7	10	14
ARIMA	27479.17	5202.53	18143.62	41364.85	5640.82	26787.82
Linear Regression	**7167.00**	**3585.20**	**3446.60**	**10893.88**	**5678.58**	**5463.60**
Hidden Markov Model	17719.40	17667.20	17432.01	29438.16	29598.35	29319.00
LSTM	11611.51	14838.06	15328.78	16248.80	20538.42	21760.05
Seq2Seq	9719.56	8972.0	7439.94	15945.44	14672.89	11669.86

Figure 6: Loss values for various models as predicted on the Daily Confirmed Cases

On all values of n, we find that the linear regression model best fits the daily cases curve of India with the best MAE of 3446.60 for n=14. This was closely followed by the Seq2Seq and ARIMA model which performed quite well for n=10. The difference in values of MAE for different n shows that different models rely on varied amounts of previous data to effectively predict cases.

4.3 Second wave prediction of all models

The following graphs show the predictions of all our models on the Second wave i.e. our test set.

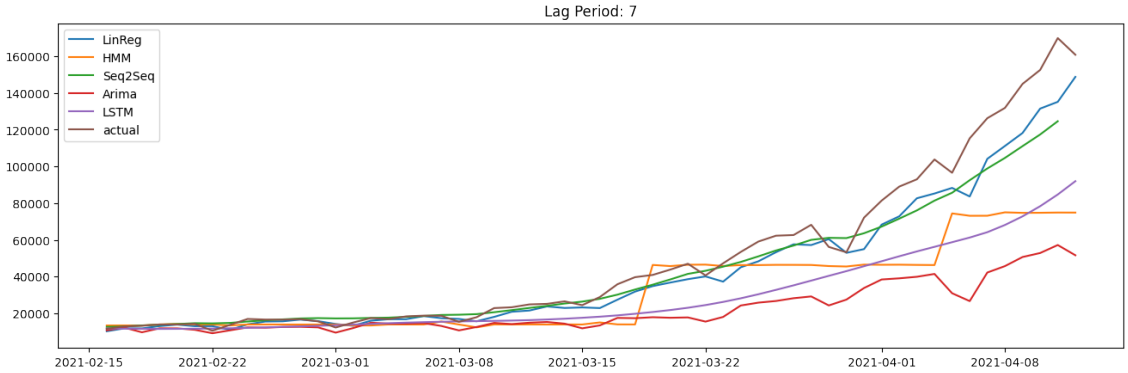


Figure 7: Second wave prediction of all models using lag period of 7 days

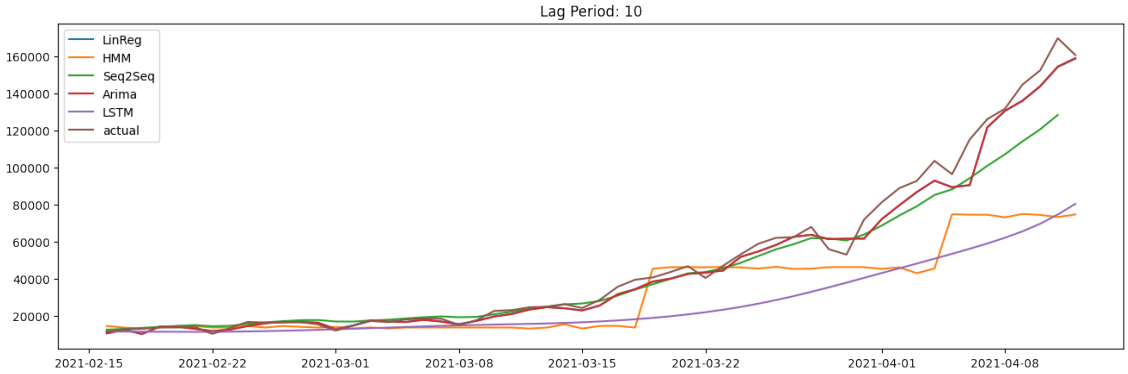


Figure 8: Second wave prediction of all models using lag period of 10 days

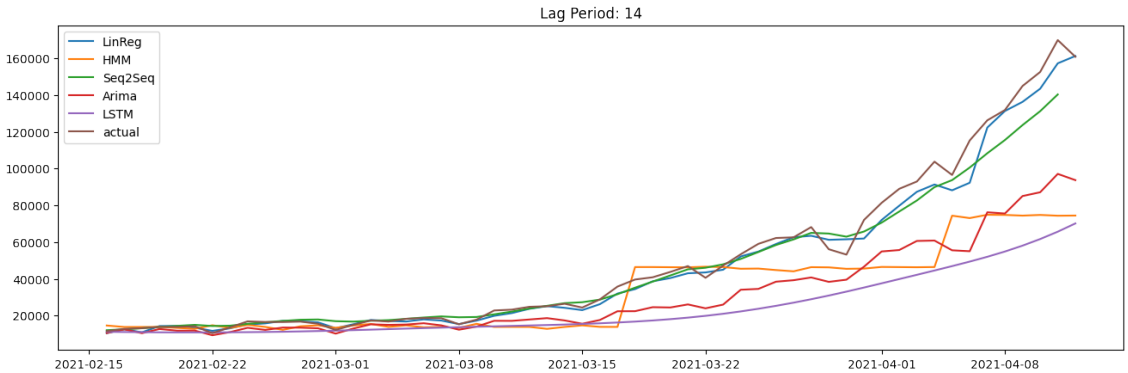


Figure 9: Second wave prediction of all models using lag period of 14 days

4.4 Forecasting using Linear Regression

Figure 10: 20 days forecast using linear regression

4.5 Impact of Lockdown

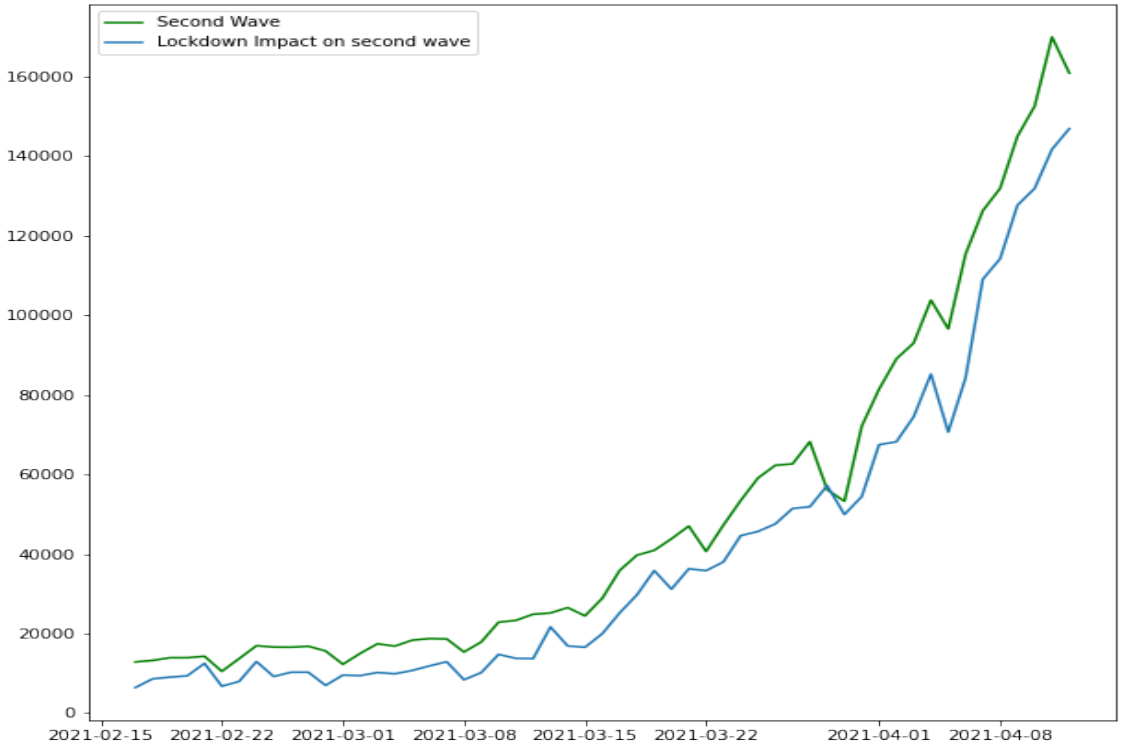


Figure 11: Impact of 10% reduction in mobility

5 Future Work

1. The impact of vaccinations can be studied at a later date, where more data is available, and important features can be observed at a later date.

2. While the deep learning models used in this had limited performance, there are more advanced, "attention" based models that have found huge success in recent NLP. These models can be modified to be used with time series and compared with current LSTM models

6 Conclusions

In this report we assessed the performance of various machine learning and deep learning models in predicting the number of daily cases in India.

From the results we found that different models were giving their best results for various values for last n-days. Overall from the results of testing on various n values we found that maximum accuracy could be achieved by taking into account cases of last 7-14 days which might be the incubation period of the virus.

Overall best accuracy was given by the linear-regression closely followed by the arima model. After altering mobility values in our dataset we passed it to our trained LR model and found that the rate of increase of cases was slowed down, showcasing that introducing stricter social-distancing norms will prove beneficial in reducing rise of cases.

7 References

1. COVID19-India API
`https://api.covid19india.org"/`

2. Data on COVID-19 (coronavirus) by Our World in Data, Our World in Data
`https://github.com/owid/covid-19-data/tree/master/public/data`

3. Community Mobility Reports, Google
`https://www.google.com/covid19/mobility/`

4. Mobility Trends Reports, Apple
`https://covid19.apple.com/mobility`

5. Micheal Phi, "Illustrated Guide to LSTM's and GRU's: A step by step explanation," 2018,
`https://towardsdatascience.com/time-series-forecasting-using-auto-arima-in-python-bb83e49210cd`

6. JAGADEESH23, "Multivariate Multi-step Time Series Forecasting using Stacked LSTM sequence to sequence Autoencoder in Tensorflow 2.0 / Keras" 2020,
`https://www.analyticsvidhya.com/blog/2020/10/multivariate-multi-step-time-series-forecasting-using-stacked-lstm-sequence-to-sequence-autoencoder-in-tensorflow-2-0-keras/`

7. Mohamed Hawas, "Generated time-series prediction data of COVID-19s daily infections in Brazil by using recurrent neural networks," Data in Brief, Volume 32, 2020, 106175, ISSN 2352-3409.
`https://doi.org/10.1016/j.dib.2020.106175.`

8. Shastri S, Singh K, Kumar S, Kour P, Mansotra V., "Time series forecasting of Covid-19 using deep learning models: India-USA comparative case study," Chaos Solitons Fractals. 2020;140:110227. doi:10.1016/j.chaos.2020.110227
https://www.researchgate.net/publication/309457872

9. Eli Shlizerman, "Which models to use for epidemic prediction?"
https://towardsdatascience.com/which-models-to-use-for-epidemic-prediction-25b22932c4ca

10. Sushmitha Pulagam, "Time Series forecasting using Auto ARIMA in python"
https://towardsdatascience.com/time-series-forecasting-using-auto-arima-in-python-bb83e49210cd

11. Tian, Yuan Luthra, Ishika Zhang, Xi. (2020). Forecasting COVID-19 cases using Machine Learning models. 10.1101/2020.07.02.20145474.

12. Jurafsky, Daniel. Martin, James. (2020) Hidden Markov Models
https://web.stanford.edu/~jurafsky/slp3/A.pdf

13. Malki, Zuhair Atlam, El-Sayed Ewis, Ashraf Dagnew, Guesh Alzighaibi, Ahmad ELmarhomy, Ghada El-Hosseini, Mostafa Hassanien, Aboul Ella Gad, Ibrahim. (2021). ARIMA models for predicting the end of COVID-19 pandemic and the risk of second rebound. Neural Computing and Applications. 33. 1-20. 10.1007/s00521-020-05434-0.
https://www.researchgate.net/publication/344881632_ARIMA_models_for_predicting_the_end_of_COVID-19_pandemic_and_the_risk_of_second_rebound