BITS F464 - ML
MAJOR PROJECT

2020-2021 Semester 2

Submitted by

Name	ID No
Savio Jomton	2018A7PS0227G
Sharanya Ranka	2018A7PS0215G
Sumit Kumar	2018A7PS0223G
Pranjal Srivastava	2018A7PS0211G

ABSTRACT

This work aims to forecast the number of deaths caused by the COVID-19 pandemic in the UK based on data about cases, deaths, government stringency measures etc. We train and compare two ML models for the same. We also attempt a “what if?” analysis that tries to examine what effect high stringency measures (or low stringency measures) right from the early days of the pandemic would have had on the number of deaths.

1. INTRODUCTION

COVID-19 is one of the worst pandemics to have ever occurred in human history. The pandemic started on a small scale in China, but within no time, it blew to international proportions. Even with the fatality rate of the virus being low, it has brought the world to a halt. This highlights how unprepared we, as a civilisation, are for such diseases.

The natural questions that arise out of this pitiful situation are:

1) Whether something can be done to curb the spread of the virus now
2) How to better handle such situations in the future

Thus, it is very important to analyse the current situation around the world and to find out where we have gone wrong in managing the disease, and what can be done better. If good analysis of the situation is done, we can prevent the countries' healthcare systems from being so suddenly overwhelmed.

This pandemic is different from the ones before, because the general public has access to a lot of pandemic-data via the internet. Also, the data is being generated and logged in a much more systematic way, allowing for machine-assisted analysis. Machine Learning techniques can comb through enormous amounts of data and extract insights in a way that manual inspection simply cannot match.

Thus, we have decided to apply machine learning techniques to address two important questions about the Covid 19 crises. The questions we will try to answer are:

1) Can we predict the number of deaths due to Covid 19 in the next k days (k=14) given information for the last n days (n=30).
2) Can we predict the change in the number of deaths if the lockdown or stringency measures were different (more or less strict).

In this report, we focus specifically on one geographical location (i.e. UK) as a demonstration of our methods. Similar analysis can be done on other regions as well.

2. RELATED WORK

We surveyed past work on of COVID-19 modeling. There were many kinds of models including ones that predicted deaths, predicted cases etc. Based on the underlying principle of the models, we observed by-and-large two kinds of models. First, the typical ML models like linear regression, neural networks etc. Second, typical ML techniques combined with an underlying epidemiological model (like the SEIR epidemic model).

We observed that the models that use an epidemic model under the hood perform far better than simple ML models. Some examples of such hybrid models are [YYG] and [GOG]. We think these models perform so well because the underlying SEIR model is able to capture the disease mechanics effectively. In a sense the SEIR model captures the manner in which the data is being generated and provides a strong bias (which is empirically shown to be correct) of how an epidemic progresses hence helping the model make better predictions.

3. METHODOLOGY

The data we are using for our modeling comes from two reliable sources. Data on the number of new cases, new deaths, icu patients, hospital patients etc. comes from “Our World In Data” [OWID], while the Stringency Index, and its breakdown into different components comes from “Oxford COVID-19 Government Response Tracker” [OXCGRT].

The machine learning problem can be stated as a “Time-series regression problem”. ‘Time-Series’ because similar data is logged along a time axis, and information about past time units is required to predict variables in the future. ‘Regression’ because we will be attempting to predict the number of deaths for a given number of days in the future.

3.1 DATA COLLECTION

While trying to find reliable data sources online, we found that some reputable organisations (like the World Health Organisation [WHO]) used data from the same two data-stores [OWID] and [OXCGRT].

[OWID] further uses several more authoritative sources for the data, and prepares a “.csv” file. As this is very convenient for us, and the sources for the data are mentioned, we decided to use the csv files prepared by [OWID]. The data is available from 31st January 2020 and is updated daily. Yet, as mentioned in more detail later, there were several null values in the dataset, so after doing all the preprocessing, we settled with 31st January 2020 to 6th April 2021 as the time-frame that we considered for this problem.

[OXCGRT] makes a similar csv file available regarding Government Measures used in tackling the pandemic. The most important of these measures for us are the Stringency Index, and its components (containment and closure policies). The severity of the issues has been made objective by assigning scores in each policy domain. Higher the scores, more stringent is the policy.

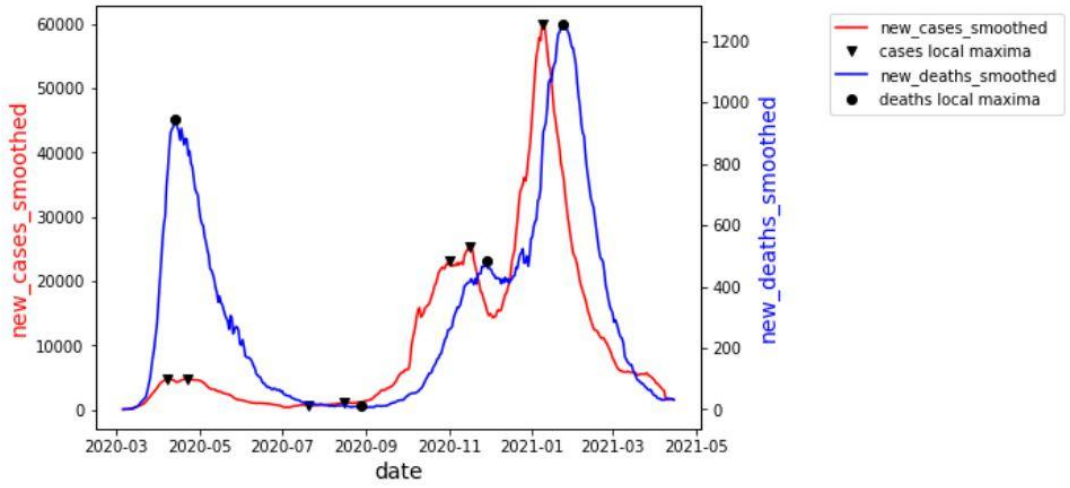
For example consider the policy domain of school closings (“C1_school_closings”). No restrictions applied will be scored a 0(minimum), while compulsory closing of all schools, public and private, will get a score of 3(maximum). A breakdown of all the scores and their meanings is given at [OXCGRT].

It is important to mention here that the effective number of days for which we have data (to train our model) is only 432 days (31st January 2020 to 6th April 2021). Machine learning techniques usually learn patterns after being trained on thousands of data points. Unfortunately, because of the nature of the pandemic, and the non-availability of data, we have very few data points. We will still try to do our best with the data we have got.

3.2 EXPLORATORY DATA ANALYSIS

Using the data, we will now try to obtain some insights about the pandemic and how various factors relate to one another. We will be presenting several graphs and mentioning a few interesting points and hypotheses about each graph. Only a few graphs out of all generated (in ml_major_project_visualisations.ipynb) will be shown as these graphs highlight the interesting points we want to make.

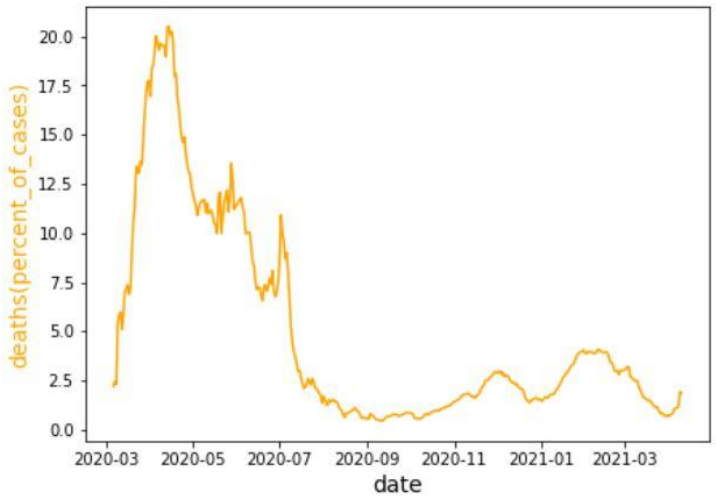
First, let us consider the relationship between the new cases, and new deaths. We will use the smoothed values (7-day rolling averages) to remove artifacts that creep in (as less data may be logged on some days like weekends).



We immediately can notice a shift between the red and blue lines. We can see that a spike in the number of cases is closely followed by a spike in the number of deaths. The number of new cases is highly correlated with the number of deaths a few days later. The black markers specify the local maximum points. We can calculate the number of days required between a spike in cases and deaths. For the first pair (,●) the difference

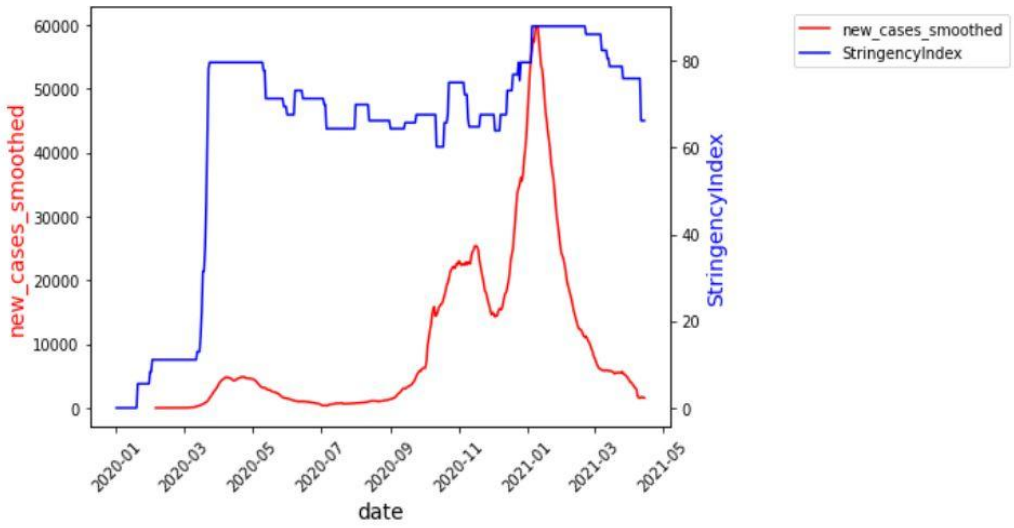
is 5 days, while for the last 2 pairs, it is 12 and 14 days respectively. This suggests that the quality of healthcare has become better now compared to last year (possibly because we know more about the disease) and it takes a longer time on an average for a person to die from the disease.

Next we will show the new deaths as a percentage of new cases.



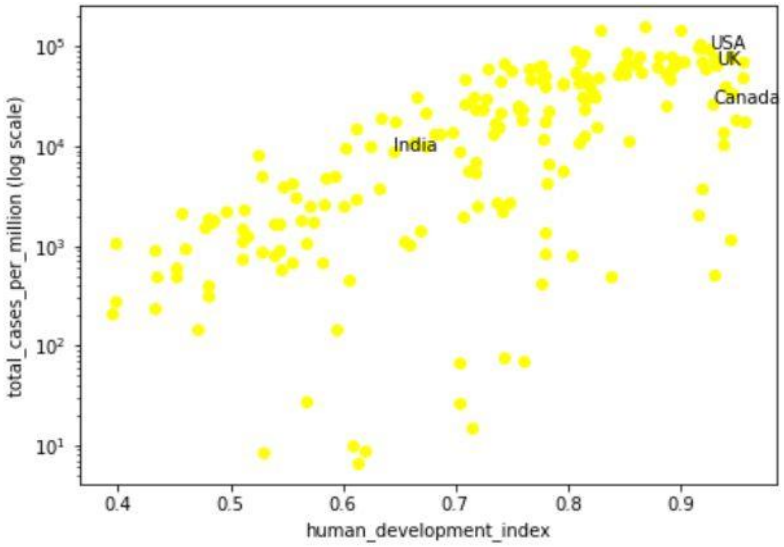
We can see here that the ratio was quite high in the beginning(~20%), and has settled down(~2.5%) towards the end. We can say that during the early days of the pandemic, either the case reporting was low (as widespread distribution of testing supplies occurred later) or that the healthcare system was suddenly overwhelmed and proper care wasn't given to the patients. There are some small peaks in the middle as well. These may be present because the spikes in the deaths and spikes in cases do not correspond perfectly (we get the deaths maxima a few days after the new cases maxima).

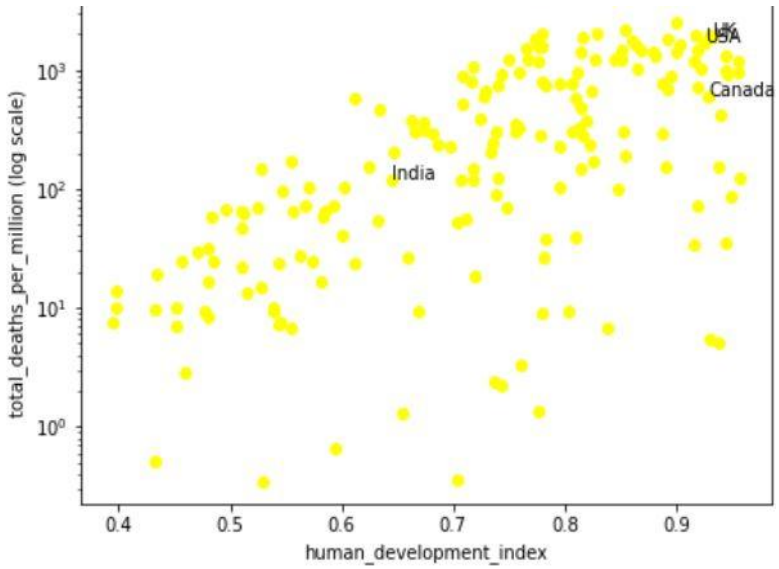
Next we will see the Stringency Index overlaid with the new cases:



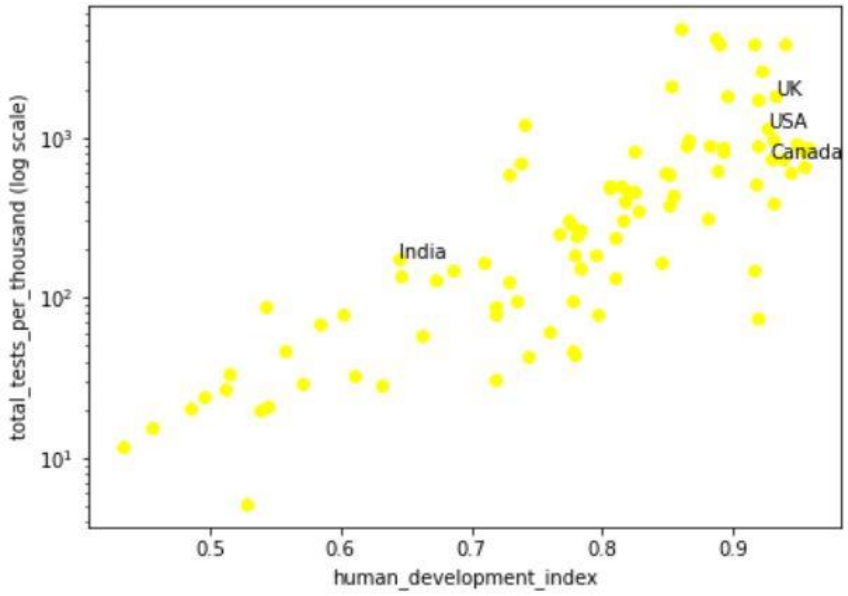
One important fact that we noticed is that the second spike in cases occurred in spite of the stringency being quite high. As lockdown measures are usually said to be quite effective, we can only make one conclusion: The implementation of the stringency measures was not carried out properly. This may have been due to complacency of people (after months of following the policies). But the only way to verify this would be to look for an increase in policy violations.

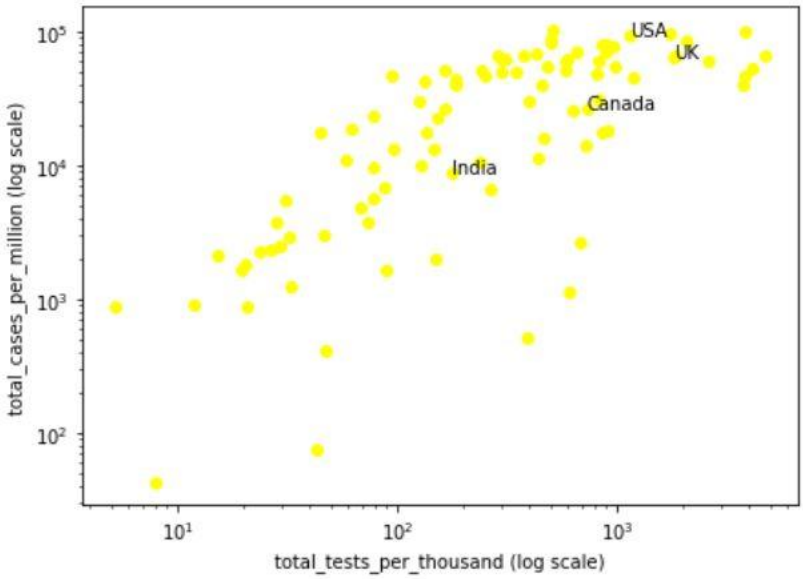
Next we will see how countries around the world have fared compared to each other. The countries we have considered are: UK, India, USA and Canada. We have also segregated the countries using the Human Development Index (HDI) on the x-axis.





This is a scatter graph of all the countries along 2 axes as mentioned in the graph. It is bewildering to see that countries with a lower human_development_index fared better than the others. Why is that? If we plot a few more graphs, we can see that countries who have better HDI also in general have better testing capabilities, thus, they report a greater percentage of the actual cases and deaths due to COVID-19.



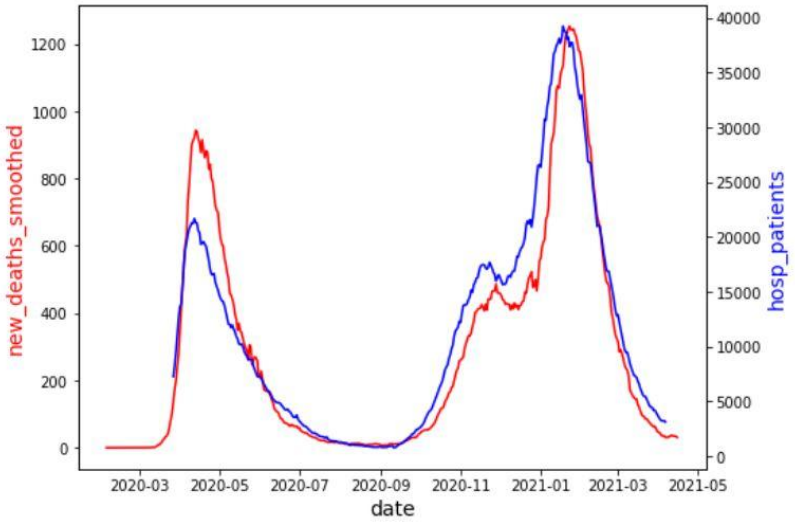


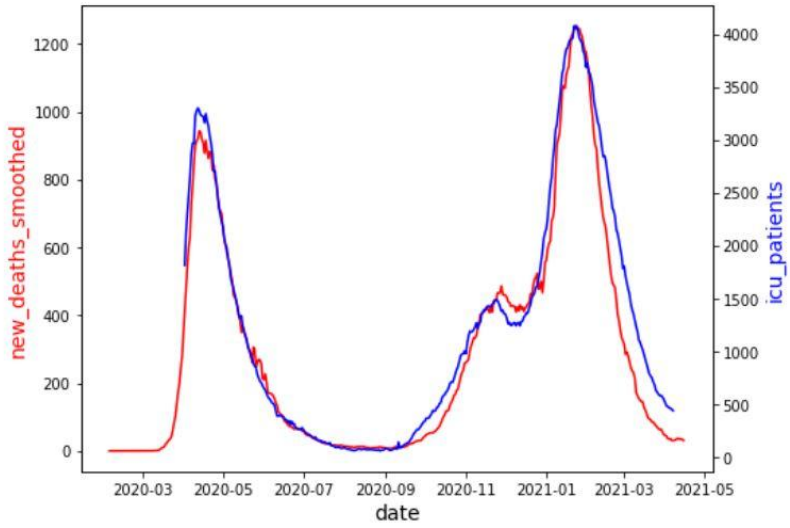
These graphs clearly show that as more tests are performed, more cases of COVID-19 are found. Thus we can confidently say that there is an underreporting of COVID-19 cases. The ratio of actual cases to reported cases would depend on the country, with the Human Development Index(HDI) playing a significant factor.

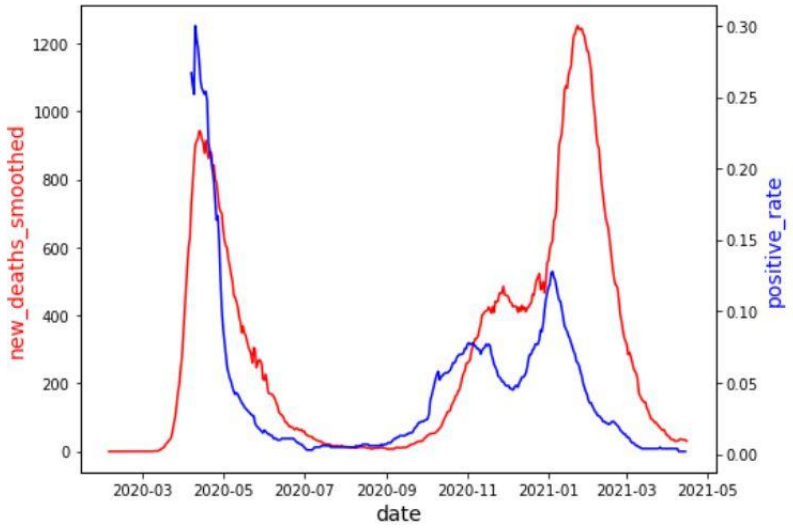
Next, we will try to find features that will help our models predict the number of deaths for the next few days given information of some previous days.

We have already seen that new_cases is a very good predictor of new_deaths a few days later, so this would be a good feature to train our model on.

Other features that show a good (shifted) correlation with our model are icu_patients, hospital_patients, and positive_rate (i.e. the ratio of number of cases and number of tests done). We should therefore include these features in our training as well.


new_deaths_smoothed
hosp_patients
date
0
200
400
600
800
1000
1200
0
5000
10000
15000
20000
25000
30000
35000
40000
2020-03
2020-05
2020-07
2020-09
2020-11
2021-01
2021-03
2021-05


new_deaths_smoothed
icu_patients
date
0
200
400
600
800
1000
1200
0
500
1000
1500
2000
2500
3000
3500
4000
2020-03
2020-05
2020-07
2020-09
2020-11
2021-01
2021-03
2021-05


new_deaths_smoothed
positive_rate
date
0
200
400
600
800
1000
1200
0.00
0.05
0.10
0.15
0.20
0.25
0.30
2020-03
2020-05
2020-07
2020-09
2020-11
2021-01
2021-03
2021-05

We have found several features that may help us to predict the number of deaths a few days into the future. In our notebooks we have also included several extra features like the containment and closure policy scores in order to do the What if analysis based on lockdown measures. The complete list of features we have used is present in the notebooks themselves. Model specific usage/dropping of other features is mentioned in the Modelling section.

3.3 DATA PREPROCESSING

Both our datasets contain data (mostly daywise) for the entire world, we just needed the data specific to the UK, so we removed everything except that.

The [OXCGRT] dataset had very few missing values, however the [OWID] dataset had quite a lot of missing data, especially during the early days of the pandemic. We managed this in the following way:

1. Certain kinds of data can easily be filled in based on press reports. For example: We know from press reports that public vaccinations did not begin in the UK till 8th Dec 2020. Any vaccinations done before that would be an insignificant number, primarily, they would have been given to frontline medical workers or vaccine trial participants. Hence we can fill all prior vaccination counts with 0. Similarly the number of cases prior to the day that the first case was reported can be filled with 0 cases.

2. Other missing data is much harder to obtain, like the number of ICU patients and number of hospital patients, which both increase and decrease with time.

Some data that was missing for a couple of weeks in between, we interpolated based on the surrounding weeks. Missing data for features like daily 'new tests' was also interpolated on the assumption that testing was not so prevalent at least until the first case was reported.

Other data in (2) was obtained by taking a scaled value of some highly correlated feature. Example: icu_patients were highly correlated with new_deaths_smoothed. Since data about the new_deaths_smoothed was available throughout, we took the mean ratio of icu_patients/new_deaths_smoothed and imputed values to icu_patients based on that ratio and the value of new_deaths_smoothed on that date.

The reason why we decided to do such an interpolation is because we wanted to make sure that the model learnt from the first wave of COVID-19 that hit the UK from March 2020 onwards.

Finally the data we used started from 31 Jan 2020. All the null values in between were filled based on the methods described above.

The data at the end of the date range had a couple of weeks of missing data as well (because some sources only update their data bi-weekly) so we decided to exclude that data as well. The last date we included was 6 April 2021.

We preprocessed and combined the two datasets. More details on the data preprocessing can be found in the corresponding ipython notebook.

3.4 CONCERNS ABOUT DATA

Before we used the data we wanted to be sure that the data was reliable. After some research on the Internet, we gained some valuable insights on the data and the nature of its collection. We highlight some of these below:

1. Multiple, independent investigative journal reports as well as scientific studies have revealed that the number of deaths due to COVID-19 is underreported due to various reasons both accidental and intentional. This is proven to be the case even in developed countries like the USA and UK [CD1] as well.

2. Similarly the actual case counts in most countries are estimated to be much higher than reported. This again might be due to various reasons but one common reason might be insufficient testing resources. More information on the undercounting of cases/deaths can be found in [BG1].

3. The stringency measures and the efficacy too is highly dependent on how well they are enforced/followed and their efficacy is not directly apparent from the data.

3.5 FEATURE ENGINEERING

We engineered the features described below:

1. susceptible_population = population - people_vaccinated - total_cases : We think this is a good measure of how many people can still be infected (this interpretation of course assumes that people are unlikely to get reinfected and that people who are vaccinated(even with a single shot) are unlikely to get infected)

2. icu_fatality = new_deaths_smoothed / icu_patients : Gives a rough measure of how many critical cases died (assuming such critical cases managed to get to an ICU)

3. icu_hosp_ratio = icu_patients / hosp_patients gives a rough measure of relative seriousness of the hospitalizations.
4. hosp_beds_left = hospital_beds - hosp_patients : Gives a measure of how many hospital beds are left. Intuitively we expect that as the number of hospital beds left decrease, deaths are likely to increase.
5. potential_cases = susceptible_population*positive_rate : Rough number of the actual number of cases in the population. This is probably an overestimate because only people who have symptoms are likely to get tested.

Later we will see that the decision tree model has used some of these engineered features.

4. MODELLING

4.1 CHALLENGES

We realised that modelling COVID-19 is challenging due to a few reasons.

1. The challenges arising due to the nature of the data and its collection. Other models we saw online try to overcome this issue by scaling those values which are underreported by some constant value. Typically this is done using the advice of domain experts or by taking the average estimated factor from other statistical studies.

2. We don't have a large amount of data, compared to other problems that we have dealt with before. We could try to help our model understand/learn from the data better using the domain knowledge and data that we have gained from dealing with past pandemics. This is exactly what the SEIR based models do.

3. There are many factors that are varying with time, that we may not have complete information about but which may change the disease dynamics rapidly. For example the emergence of more infectious/lethal strains of the virus, as it mutates over time.

4.2 PRELIMINARIES

We used data from 31 Jan 2020 till 31 Dec 2020 as the training data and reserved the data from 1 Jan 2021 to 6 Apr 2021 for testing purposes. The reason we made such a split at the date 1 Jan 2021 is because we need to see how well the model performs on that second wave of COVID-19 that peaked in the UK during the month of January 2021. Both models were fed-in data samples each consisting of 30 days of past data and trying to predict the new_deaths_smoothed for the next 14 days. This is how we converted this problem into a supervised learning task.

Note: We only trained our models on the train set and never on the test set even while making forecasts in the test set date range. That is we did not do a walk forward evaluation of the model.

4.3 METRICS

The metric that our models were trying to optimise was Mean Absolute Error(MAE). We felt that this was a good choice for this problem because comparing absolute death numbers made more sense. We have additionally compared the models based on the Mean Squared Error(MSE) and Mean Absolute Percentage Error(MAPE) as well.

4.4 DECISION TREE MODEL

We choose to try a decision tree model because it is highly interpretable. The features the model considers important may give us insights into the factors that affect the deaths. We used the sklearn implementation of the decision tree, sklearn.tree.DecisionTreeRegressor with the following parameters (based on our hyperparameter search):

max_depth = 5, criterion = 'mae', min_samples_split = 8

Some additional code was used to drop a given number of contiguous days of past features (while keeping a given number of the immediately previous days) so as to reduce the number of features the model has to comb through. The best values were found to be to drop contiguous 18 past days of which we keep the immediate 5 past days data. We trained the model using the features of the immediate 30 days (of which the 18-5 days were dropped) and had it forecast 14 days of new_deaths_smoothed into the future.

We notice that the decision tree picks up on some of our engineered features as well as picks features from about 20 days in the past, which is about the median time a person who is likely to die of COVID-19 will last once infected (estimated by medical experts).

The feature ‘C6_Stay at home requirements’ was dropped because the decision tree gave undue importance to it (half of the total gini feature importance). We hypothesize that this is driven by the extremely high number of deaths that occur during the first peak which is about the same time that this feature becomes high. We observed that if this feature is included the model seems to correlate high values of this feature with high deaths.

The full decision tree that was learnt has been visualised in the notebook.

4.5 LSTM MODEL

In addition to a decision tree, we also decided and fit an LSTM regression model to our data. Long Short Term Memory (LSTM) networks are neural network architectures with feedback connections. Although RNN’s share this characteristic as well, LSTMs have been shown to be better than RNNs at learning long-term dependencies due to the particular nature of their connections.

The LSTM cell consists of 3 main categories of connections:

1. Forget Gate : To “forget” portions of previous information based on current input and past output.

2. Input Gate : To filter information from the input that is added to the cell state.

3. Output Gate : To filter specific information from the cell state to be sent as output of the current cell.

We have used the Keras implementation of the LSTM for our model. Our model consists of 1 LSTM layer consisting of 64 units (this will give a 64-vector as output), and 2 Dense layers. The first dense layer has 2*(number of values to predict) neurons and the last Dense layer has (number of values to predict) neurons. We have 2 Dense layers to help the model create a dependency between successive values in the prediction. We also have dropout layers between the previously mentioned layers. This helps in reducing the overfitting of training data.

The output prediction is a vector of length =(number of values to predict), currently set to 14 as we want next 14 days' predictions. The input to the model is given as a 3-dimensional numpy ndarray (num_of_samples, timesteps_per_sample, features_per_timestep). As we look at the past 30 days data, timesteps_per_sample = 30. 15 features in total (original+engineered) are used to train the model. These features have been selected based on the exploratory data analysis.

In addition to this, the data is scaled for better performance of the LSTM model. We have used a MinMaxScaler with a feature range of 0 to 1. We have only looked at the training data to set the scalers, therefore after scaling the entire dataset there will still be values beyond the feature range. Notably, the vaccination data has feature values above 1 even after scaling because vaccinations started after 1st January 2021. We decided to drop these features as well.

4.6 EFFECT OF LOCKDOWN (WHAT-IF ANALYSIS)

We have considered the effect of lockdown on the number of deaths in the UK because it has more non-zero values in the time frame considered (as compared to vaccination data). This will give us more relevant data points to train our model on. The time frame we have chosen is the same as the time frame for the train dataset. We have considered two scenarios:

1) Predict deaths if the containment and closure policies were at maximum strictness

2) Predict deaths if the containment and closure policies were at minimum strictness

To accomplish this, we copied the train dataset, and changed the values of the containment and closure policies scores. The minimum and maximum values of these scores are provided in [OXCGRT]. Thus, we end up with the two situations mentioned above. We finally get the predictions of the model on max_stringency and

min_stringency and plot the predictions along with the original predictions, and the actual death values.

In order to get quantitative results, we simply took the mean of the differences of the predictions for the max (or min) stringency case and the original predictions. This provides us with one value, that is the average difference in number of deaths predicted if the policies were more strict (or less strict) respectively. We have only considered differences in the predictions (and not prediction vs actual deaths) because we want to see how the model is using the containment and closure policy scores.

5. RESULTS

Here we will report and explain (qualitatively and quantitatively) the predictions made by the models, compare their error scores and also explore the results of the what-if scenario predictions of the models.

Neither of the models did very well on predicting the new deaths for the next 14 days. We can attribute it to three main reasons:

1) Time-series prediction problems are usually quite difficult problems to solve. The usual assumption that individual data points are independent does not hold.

2) A large number of variables would affect the target variable. Apart from that, we are not guaranteed that the reporting of cases, deaths etc. is done correctly (see Challenges subsection in Modelling and discussion on underreporting in exploratory data analysis).

3) We have only around 430 days of data to learn the underlying patterns. More data would have helped the model to observe more clearly the effects of different features on the target.

FORECASTING

The performance of the models on predicting new_deaths_smoothed 14 days into the future on a daily basis is given below. These values are for the testing data only which is common to both the models.

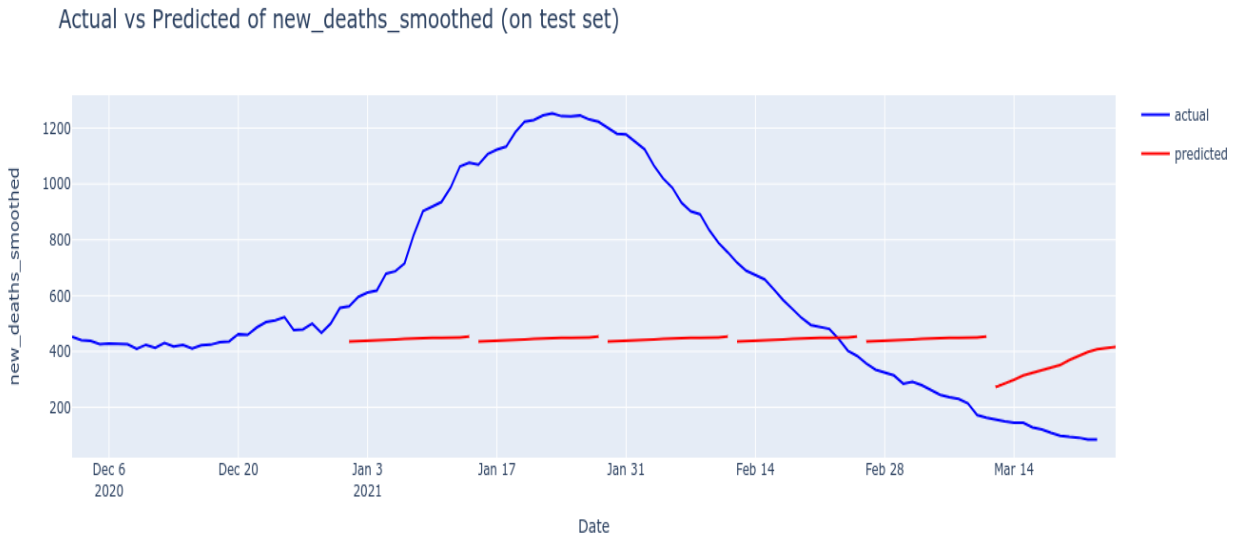
Model\Metric	MSE	MAE	MAPE
Decision Tree	191984.40	354.95	0.99
LSTM	20819.43	125.89	0.55

We can see from the table above that an LSTM Regression model fares better than a Decision Tree Model. The LSTM model predicts ~230 deaths (on average) closer to the actual death count. This may be because an LSTM is more naturally suited for time-series and sequence prediction problems. Nevertheless, the Decision Tree model is

human-interpretable, and the feature importances and hierarchy can be shown more naturally in such a model.

Now, we will present graphical representations of the above. Training and validation graphs are given in the notebooks.

Decision Tree



LSTM

WHAT IF SCENARIO

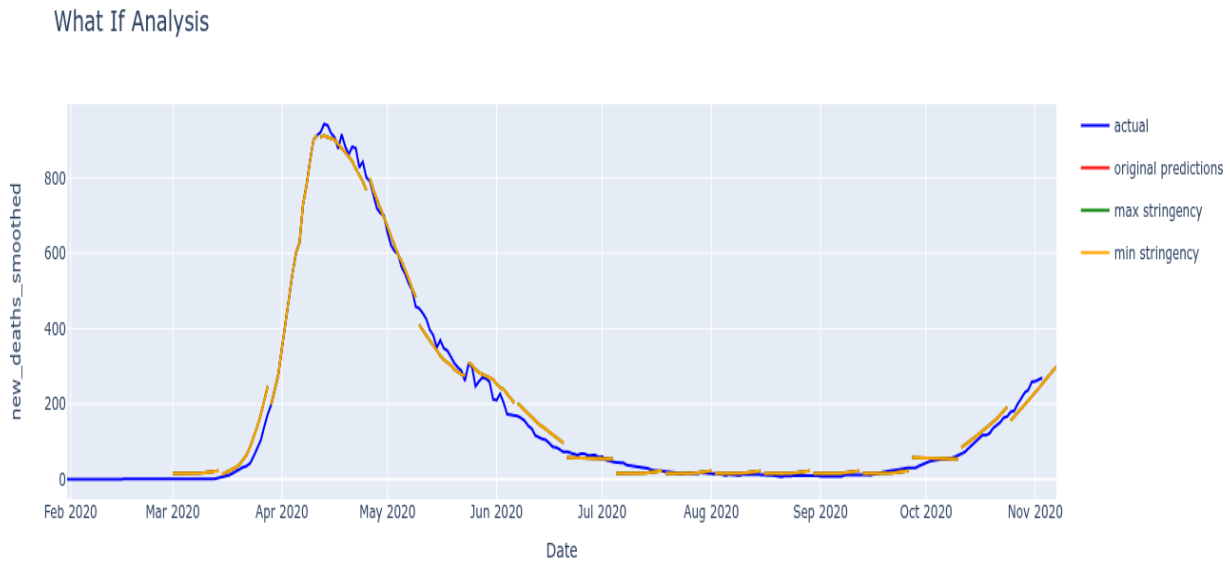
The following table summarises the difference in predicted new_deaths_smoothed (what-if prediction - original prediction) for both the models.

Model\What if?	All stringency measures were at maximum?	All stringency measures were at minimum?
Decision Tree	+1.11	-0.69
LSTM	+16.31	-37.32

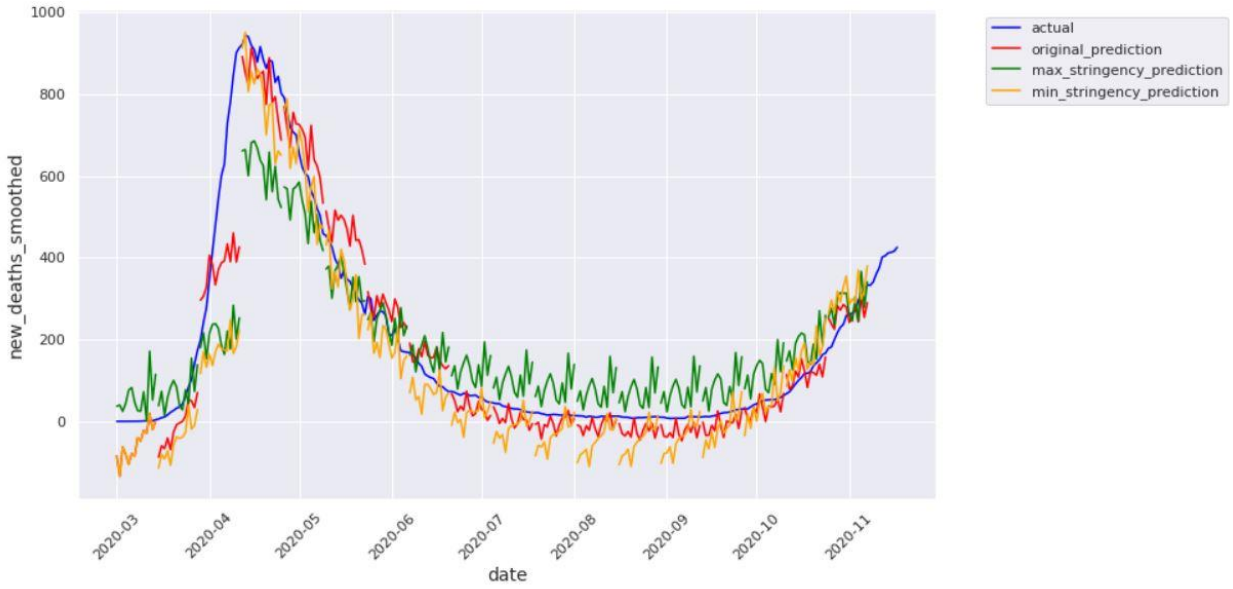
It seems as though the models claim that higher stringency measures lead to more deaths and vice versa!! Either our models are doing something wrong, or we must share this insight with all governments immediately, and all stringency measures should be promptly removed!
There is one way we can explain this lapse in learning of our models. As seen in the exploratory data analysis (also check visualisations notebook), we can see that when the deaths per day were high, the stringency measures were also high. We can explain this as a government's reaction to increasing deaths, and that stringency measures help in quickly flattening the peaks in deaths. Unfortunately, it seems that our models have only learnt the high stringency-high deaths correlation, and thus are giving us opposite results.

Below are the predictions in graphical format.

Decision Tree



LSTM


actual
original_prediction
max_stringency_prediction
min_stringency_prediction
new_deaths_smoothed
1000
800
600
400
200
0
2020-03
2020-04
2020-05
2020-06
2020-07
2020-08
2020-09
2020-10
2020-11
date

6. CONCLUSIONS

From all the discussions and results presented above, we can conclude that

1) An LSTM model is likely to do better than a Decision Tree model on predicting the number of deaths in the future given some information about the past. This may be because LSTMs can more naturally work on time-series data. On the other hand, a Decision Tree can tell us about feature importances in a human-interpretable way.

2) Both the models do much worse compared to mathematical models like SEIR.. A hybrid SEIR and ML model may be able to perform even better, with the parameters in the SEIR model being determined by the ML model, given some appropriate evaluation function.

3) There are several reasons why the ML models do not perform well. Presence of false reporting (accidental or intentional) and less data are two significant reasons.

4) Machine Learning, although a useful tool in such a situation, can do much better with the presence of effective biases to model the problem better.

7. REFERENCES

1. [YYG] https://covid19-projections.com/about/
2. [GOG] https://arxiv.org/abs/2008.00646
3. [OWID] https://github.com/owid/covid-19-data/tree/master/public/data
4. [OXCGRT] https://github.com/OxCGRT/covid-policy-tracker
5. [CD1] https://www.medrxiv.org/content/10.1101/2021.01.15.21249885v1
6. [BG1] https://fivethirtyeight.com/features/coronavirus-case-counts-are-meaningless/
7. [COLAH] https://colah.github.io/posts/2015-08-Understanding-LSTMs/
8. [WHO]https://www.who.int/emergencies/diseases/novel-coronavirus-2019