

The University of Sussex
School of Engineering & Information
Master's Dissertation

Prophet modelling of Covid-19 cases in the United Kingdom

Candidate Number:276270

Supervisor: Maxine Sherman

Word Count: 9676



Computer Science MSc

Informatics Department

The Proclamation

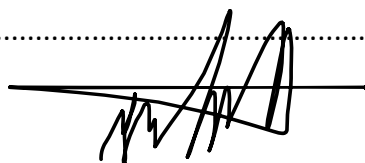
I confirm the reliability of the data on this cover sheet.

I confirm that the ideas presented in this dissertation are original to me and that my work

does not include any instances of misconduct, such as plagiarism, collusion, or result fabrication.

Enes Karaduman

Candidate's signature

A handwritten signature in black ink, consisting of a series of loops and strokes, positioned over a horizontal dotted line.

Acknowledgments

I want to sincerely thank my supervisor, Maxine Sherman, for her support and guidance throughout this project. Her advice and encouragement have been incredibly helpful. I'm very grateful for her time and dedication.

Table of Contents

The proclamation	2
Acknowledgements	3
Abstract	5
1. Introduction	6
2.1. Overview of the Dissertation	6
2. Literature Review	7
2.1. Introduction to Predictive Models	7
2.2. Logistic Growth Model for Epidemic ...	7
2.3. Prophet Model	8
2.4. Comparative Studies with Other Models ...	8
3. Methodology	9
3.1. Ethical Considerations	9
3.2. Dataset Resource	9
3.3. Logistic Growth Model	10
3.4. Prophet Model	11
3.5 Cap Value Estimation	13
3.6 Data Split and Model Training	15
4. Data Collection, Preprocessing and Exploratory Data Analysis.....	16
5. Seasonality and Holiday Effect Exploratory Analysis. 22	
6.1. Seasonality Analysis	23
6.2. Holiday Analysis	24
6. Logistic Growth and Prophet Model Integration: Methodology and Block Diagram ...	26
7. Results and Performance Evaluation	27
7.1. Performance Metrics	28
7.2. Observations	29
8. Discussion	39
9. Conclusion	40
10. References	41
List of Figures	43
List of Figures	44
Appendix	45

Abstract

In this study, Prophet and logistic model approaches were used to predict the number of COVID-19 according to a provided period. epidemic diseases show a tendency to reduce when seasons are getting warmer, which is why we can talk about seasonality as an important factor, which needs to be considered. Prophet's seasonality feature is quite helpful in providing more accurate results. The same thing is also valid for holiday effects. Therefore, prophets can capture the cyclic nature of COVID-19 transmission by using its seasonality and holiday effects.

On the other hand, the predictions were made using data from the United Kingdom's National Health Service. The primary objective of this research is to obtain a model that provides high accuracy while predicting the number of COVID-19 for given periods. The main reason why the Prophet model was picked is that the model can utilize daily, weekly, and seasonality effects in addition to its holiday effects, which is why the model is quite useful and able to reach more accurate prediction outcomes. At the same time, a logistic cap, a value limiting a maximum number of cases, is integrated into the model so that the model will not be able to exceed certain cap values, which will be quite beneficial to reach more accurate results by minimizing errors on predicted values.

The logistic model provides have more robust framework to the machine learning model because it will not exceed thanks to feeding with cap value, which will provide reliable predictions. This dual model approach might be quite beneficial to understanding potential case increases for public health authorities. The outcomes emphasize the importance of temporal and external dynamics in epidemiological forecasting. Overall, it is possible to adapt this study to any epidemiological study like COVID-19, Spanish Flu, Swine Flu, etc. Most of the epidemic diseases have the same increase trend, which is a logistic trend. The study with the help of true adjustments like a holiday, and seasonality effects can be integrated into another epidemiological study as long as they have logistic trends. In addition to that, It is possible to adapt the project the number of countries can be increased as long as they have the same features involved in the model.

1. Introduction

A severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) outbreak, which started in late December 2019, has significantly impacted the lives of individuals and professionals. The viral infection, also called COVID-19, can trigger severe respiratory conditions, including atypical pneumonia, and is characterized by strong human-to-human transmission through airborne particles. It is believed that the virus may have spread from bats to humans through another intermediate host. On January 31, 2020, the World Health Organization (WHO) announced an international emergency

The virus has reached more than 100 countries globally since its initial discovery and, despite stringent control measures, was considered a global pandemic, becoming a serious threat and challenge for both the world's healthcare system and the worldwide economy. The pandemic was declared over after three years of challenge, which resulted in significant damage to the economy and adversely had an impact on people's lives.

As of August 2024, the United Kingdom government published ongoing reports and statistical analyses about COVID-19 cases. The Office for National Statistics (ONS) indicates that the global epidemic has killed a significant number of people; since the outbreak's start, over 7,057,145 million deaths have been associated with COVID-19[11]. When it comes to the United Kingdom, the fatality rate was shown as 1.74% but when you consider the fatality rate all around the world is 5.72% [11]. The main reason for the changes originates from the emergence of new variants like omicron, the rollout of vaccination programs, and changes in public health measures. The United Kingdom government took necessary precautions to minimize those deaths and spread the epidemic like social distancing, lockdown, the closure of businesses and schools, and bans on travel and outdoor activities. The application of precautions shows a change tendency from country to country, which is why the epidemic's peak point, or swings are going to be different for each country. epidemics have the same trend, which is the logistic trend when it comes to cumulative death, cases, etc [1]. The important thing about an epidemic is when it will reach a steadiness level for the trend since all epidemics are the same. The same thing is also valid for Covid-19 too just like any other epidemic ailments. Therefore, prediction of the pandemic and where it will stop is quite important regardless of covid-19.

The primary objective of this research is to develop a predictive model forecasting the number of COVID-19 cases in the United Kingdom over a given period. The ultimate purpose of the research is to utilize and compare the Prophet model and logistic growth model to comprehend their effectiveness in predicting the trajectory of COVID-19 cases, which consider a certain number of factors such as seasonality, holidays, and external events like the occurrence of variants. The main research for this study is: "How long and accurately can the combination of the Prophet and logistic growth models predict the future spread of COVID-19 in the United Kingdom, considering seasonal, external effects and public health interventions?"

1.1 Overview of the Dissertation

This dissertation will contain the use of COVID-19 case data from the National Health Service (NHS) in England and Scotland, which provides quite reliable and comprehensive datasets. The

data will undergo preprocessing due to some issues about the data such as getting rid of missing values and shifting vaccination data over time. Later on, the study will implement the Prophet model with the integration of logistic growth to cap predictions and will also explore how seasonality and holiday effects can affect the spread of the virus over the country. Furthermore, the dissertation has a detailed literature review on predictive models in epidemiology, which is a methodology section demonstrating the implementation of the models, and an analysis of the outputs to be able to assess model performance on the given dataset. Discussions associated with the implications of the results and recommendations for additional study will round out the dissertation.

2. Literature Review

COVID-19 is an illness caused by the SARS-CoV-2 virus, which occurred in late 2019. The virus spreads rapidly and easily, leading to widespread health crises and significant social and economic disruptions worldwide. The pandemic has negatively affected daily life and posed considerable troubles on a global scale including overwhelming healthcare systems due to high spread rates, causing economic downturns due to lockdowns, and changing education and social structures. After a long lockdown, companies, schools, and the government found a way to adapt themselves by remote working to this catastrophic term.

2.1 Introduction to Predictive Models in Epidemiology

Predictive modelling has always been a crucial tool in epidemiology to be able to predict the spread ratio of infectious diseases. Traditional prediction models, such as the Susceptible-Infected-Recovered (SIR) model, have historically provided a basic comprehension of the dynamics of disease.

The SIR model is a basic predictive model in epidemiology used to comprehend and forecast the spread of infectious diseases, which divides a population facing an epidemic into three parts: susceptible (S), infectious (I), and recovered (R), allowing for the estimation of significant epidemiological parameters such as the fundamental reproduction number (R_0) [3]. To illustrate, the SIR model implemented during the COVID-19 pandemic in Ecuador estimated an R_0 of 2.2, meaning that each infected individual could spread the virus to 2.2 other people. It underscores the degree of interventions like quarantine and immunization strategies [3]. The value shown reflects the likelihood of an infection passing on throughout a totally vulnerable population. However, COVID-19's unique traits—such as its high transmissibility and variable incubation period—call for the growth of more advanced models that are capable of handling complex patterns and non-linear trends. Therefore, the use of old-fashioned models like SIR for COVID-19 cases does not have any meaning because of the mentioned reasons.

2.2 Logistic Growth Model for Epidemic

Because of its simplicity of use and ability to precisely show the initial rapid growth followed by a plateau as resources become scarce, the logistic growth model has been used to model epidemics. The model is especially useful for epidemic forecasting as it shows the natural course of many infectious diseases such as COVID-19. the logistic model is often used to have a prediction on the cumulative number of cases over time, capturing the entire epidemic curve from the initial outbreak to the point where the spread of the disease stabilizes. The model's S-shaped curve, which has been effectively used in previous epidemics like SARS, accurately demonstrates the progression of an epidemic by modelling the cumulative number of people suffering from COVID-19. The logistic model is also quite efficient in terms of predicting peak cases, which is the maximum number of active cases at any given time and understanding the potential effect of interventions such as lockdowns and social distancing. According to Chang et al. (2020), because of its straightforward theory and effective computation, the logistic model is

often used in regression analysis fitting of time series data, making it a helpful instrument in predicting COVID-19 trends for different nations [1]. The model has three phases when it comes to epidemic growth: a slow increase at the beginning, a fast growth duration while it approaches the peak, and a slow growth process as the outbreak reaches its maximum [1]. Due to this structure, significant turning points in the epidemic curve can be detected, including the peak infection ratio and the effects of public health initiatives [1].

2.3 Prophet Model

The Prophet model, developed by Facebook, is a machine learning model designed for time series predictions, particularly effective for datasets with strong seasonal effects. Unlike traditional models, Prophet is specifically developed to handle missing data points and incorporate external features such as holidays and seasonal patterns, making it highly useful for COVID-19 forecasting. The trend component models show the non-periodic variations in the data, like general rises or decreases. On the other hand, Seasonality holds the information of periodic patterns, such as weekly or yearly cycles by using the Fourier series. The holiday component serves as an external regressor to adjust for the effects of holidays. Those iterative features enable the Prophet model to make reliable predictions even if data is incomplete or influenced by external factors.

Prophet has been used in a variety of studies to model and forecast pandemics and epidemics. For example, It is also used Prophet in order to predict the spread of COVID-19 in the United States, Brazil, Russia, and India by comparing with other machine learning time series models like ARIMA and SARIMA [9]. Another study by Hassani et al. (2020) implemented Prophet to predict COVID-19 cases in Morocco, demonstrating how well it captures the effects of government actions on the epidemic curve [2]. These examples emphasize. These examples show how Prophet is quite a common model in terms of predicting epidemic cases.

2.3.1 Seasonality Effect:

It refers to regularly recurring fluctuations in data over a certain period. For instance, in the retail sector, there is an increase in sales during certain months of the year (for example, in December due to Christmas shopping) and in the agricultural sector, harvest times of seasonal products affecting production are examples of seasonality, which can be observed not only on a monthly or annual basis but also on a weekly, daily or even hourly basis. Analysis of seasonality is important for businesses to be able to predict these regular fluctuations and plan accordingly.

2.3.2 Holiday Effect

The holiday effect basically creates an impact for a certain period of time, which results in some deviations in seasonality patterns. As a result of the issue, it is likely to affect predictions. For instance, public holidays may encourage people to go outside. This circumstance creates an impact on consumer behaviours and spend more money than any usual day. It is important to note that holidays like Bank Holidays and Easter Sundays are not used as holidays. only the holidays included in Prophet's default UK settings were used. All holidays were treated equally without differentiation.

2.4 Comparative Studies with Other Time Series Machine Learning Models

Numerous comparative analyses evaluated the efficiency of different Covid-19 models for prediction, highlighting the benefits and drawbacks of each approach. When it comes to adapting to changes in the seasons and irregular schedules, like public holidays, the Prophet model is quite successful. A study compared three different time series machine learning models, which are ARIMA (Autoregressive Integrated Moving Average), SARIMA (Seasonal ARIMA), and Prophet models, to predict COVID-19 cases in the USA, Brazil, and India [9]. The Prophet model was quite effective when it came to predicting daily new and cumulative cases in the USA, demonstrating strong predictive accuracy and reliability, particularly in handling seasonal effects and missing data, and providing predictions that were consistently closer to actual outcomes when it is compared with other models [9]. On the other hand, ARIMA is more successful in predicting cumulative cases with a constant positive growth rate. When it comes to SARIMA, it shows better performance in the aspect of predicting weekly cases.[9]. When it comes to COVID-19 prediction for the United Kingdom, the prophet might be more appropriate than these models because the model must deal with seasonality effects and holiday effects, which can make an impact on the virus spread rate. Seasonality provides a piece of information about the recurring patterns in data because of seasonal alterations, such as the increased number of cases throughout colder months when people are more likely to prefer indoor places. Holiday effects tell sudden increases or drops in cases due to changes in social behaviours during public holidays, such as gatherings or travel. It is quite difficult to model the effect because there is an irregular fluctuation significantly affecting the growth rate. Due to this fact, picking Prophet seems the safest option over other models thanks to seasonality and holiday effect customizations.

Identified Gaps and Research Contribution

Despite the comprehensive use of these models, there are some issues and gaps in the literature, particularly associated with integrating logistic growth and Prophet models in order to increase predictive accuracy. When you examine other literature reviews, they only consider short-term effects rather than focusing long term effects. They pretend as if conditions will be the same. Especially, when a variant with a high infectious rate like omicron can occur, it is likely to have an impact on those predictions, a serious issue for epidemic predictions, which needs to be considered.

This study is not only focused on short-term effects but also considers long-term and potential variant effects. Then, those impacts shape model design in addition to seasonal effects.

Conclusion of Literature Review

The importance of accurate and adaptable tools for prediction is highlighted in the literature on COVID-19 predictive modelling.

While both the logistic growth and Prophet models present robust frameworks, their integrations provide more reliable outcomes. The study is going to make a considerable contribution to the field by trying to address current gaps and providing a reliable model adapting to the unique challenges like holidays, and variants of COVID-19 case prediction. Then, the model design is completed according to seasonality, holiday, and other external effect for the United Kingdom.

3. Methodology

3.1 Ethical Considerations

This research is dependent on the BCS Code of Conduct stressing integrity, transparency, and the protection of public interest, especially when handling sensitive data. The study released the use of anonymized COVID-19 data ensuring it had been made publicly available via the NHS, ensuring that no personally identifying information was obtained.

All analyses adhere to ethical responsibility and aim to contribute to public health understanding while it respects to privacy and ethical standards.

3.2 Dataset Resource

The most recent COVID-19 data was obtained from the NHS, which utilizes datasets from the national health services of England and Scotland for the model. Wales and Northern Ireland had a myriad of missing and filling data, which is why they were not involved in the NHS dataset. The main reason why the National Health Service data was chosen is the data is provided by the government since the National Health Service is a sub-institution of the United Kingdom government. The dataset contains daily records from 3rd January 2020 to 2nd October 2023, which involves various features such as daily and cumulative COVID-19 case numbers, new and cumulative first episodes, reinfections, PCR testing data, vaccination doses (first, second, and third), and hospital admissions. The dataset also includes flags for significant events such as the first, second, and third lockdowns, as well as the Omicron variant date flags. This dataset provides comprehensive information on the COVID-19 pandemic's progression in England and Scotland. It will be mentioned to further details on data processing and cleaning and will be discussed in further sections. The dataset used can be accessed via [Kaggle](https://www.kaggle.com/datasets/albertovidalrod/uk-daily-covid-data-countries-and-regions/data) at this <https://www.kaggle.com/datasets/albertovidalrod/uk-daily-covid-data-countries-and-regions/data>, which is the raw version of dataset without applying any preprocessing operation.

3.3 Logistic Growth Model

The logistic growth model is a growth type beginning with a fast increase and then started getting slow down because of limited resources, such as healthcare facilities, or government restrictions for epidemics like lockdowns and the use of masks. Therefore, the model can be integrated into various areas, including ecology, economy, and particularly in modeling the spread of infectious diseases. Resources can be critical because of the above reasons for the epidemic. The model is called the "Verhulst Equation." One of the main reasons why the logistic growth model was chosen for this study is they have the same pattern with epidemic growth, where the first rapid increase of cases is followed by a decrease when these resources become limited. This model will be more fitted to COVID-19 data to predict the epidemic's projection and to comprehend how resource limitations have an impact on the spread of the disease when it comes to the epidemic. The mathematical expression of this model is:

$$\frac{dt}{dP} = r * P \left(1 - \frac{P}{K}\right)$$

Logistic Growth Formula with Differential Equation

P: The initial cumulative number of COVID-19 cases at the start of the model

R: Spread Rate of the virus

K: Cap value (Carrying Capacity)

$$Q(t) = \frac{K}{1 + \exp(-b * (t - a))}$$

Logistic Growth Formula

K: Cap Value (Carrying Capacity)

b: Shows the growth rate of epidemic

a: Shows fastest growth rate. After this point, curve will have a S shape.

After solving the differential equation, we obtain the following equation, which is described in terms of the "time" parameter. Let's say K is the maximum cap value representing the upper limit for cumulative cases. When time goes on and the number of cumulative cases for $Q(\text{time})$ rises, the growth will keep increasing. After time starts to get closer and closer to the t value, the growth will eventually slow down. This slowing growth results in an S-shaped curve, which is shown by the formula. The growth is going to be slower and slower, which means it will obtain an S-shaped Curve according to the formula. Hsieh YH, Lee JY, and Chang [1] obtained a graph about Sars by using the Richard model (Logistic Growth Model) for Taiwan 2003. According to Figure 1. While Graph "A" represents confirmed cases for Sars, Graph "B" provides information about estimated by using truncated data. All in all, the obtained shape for the Sars epidemic was a logistic growth model. It can be generalized every epidemic did not pay attention to governments. After necessary precautions are taken by the government, the growth shows a slow tendency in the long term, which is why an S-shaped curve is obtained. The same issue has also happened for Sars, as well, which means the same approach is also applicable to COVID-19, but we cannot know where the cap is going to limit the epidemic since it changes from epidemic to epidemic depending on its severity and spread ratio. All in all, the obtained inflection points that decide when the growth rate trend will change is an important factor and common for every epidemic when it comes to the logistic growth model.

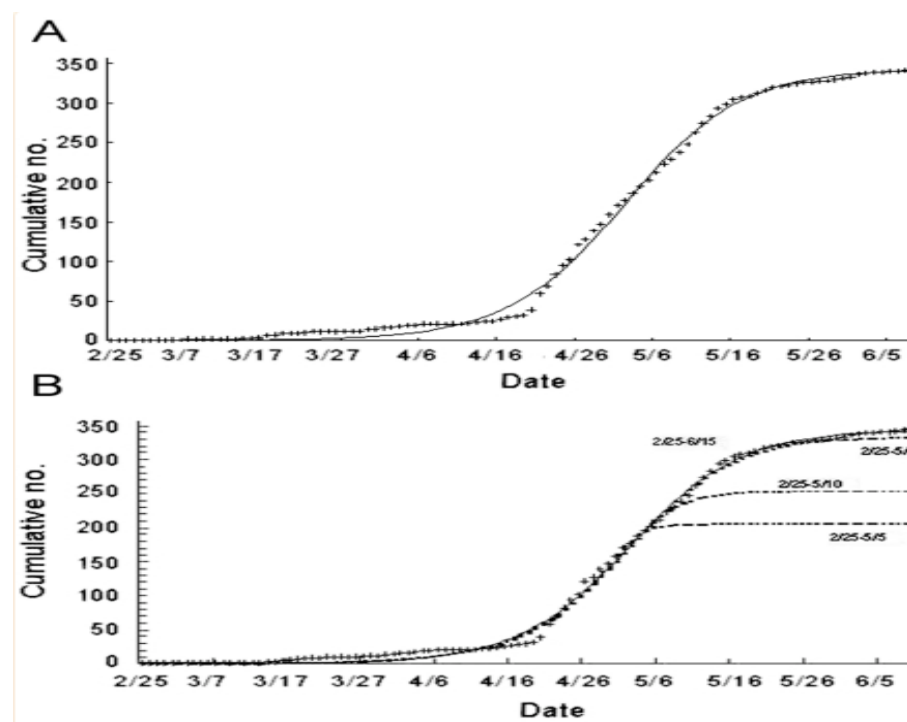


Figure 1. Cumulative Cases in Taiwan, 2003[1]

3.4 Prophet Model

Prophet is a machine learning model that predicts from time series data. Machine learning techniques are used to train models with existing data. After training is completed, the model can have a prediction about future values. In order to check the model's prediction accuracy, you can compare it with test data and predicted values so that you will have an idea about the accuracy rate of the model, which helps how perform on providing historical data. When it comes to "prophet", it is a machine learning model developed by Facebook, which is suited to work on time series data, which is basically an additive machine learning model. It is more suitable to work with data having strong seasonality effects. When you consider COVID-19's circumstances, the seasonal effect is so vital since most epidemics show a tendency to reduce when the weather starts warmer. Holiday effects are also another thing, which might be correlated with weekly effects. That's why, the prophet's sensitivity to seasonality effects is the reason why the model is picked to be able to predict future cases of Covid19. This model will be fitted to the COVID-19 data to assess its prediction performance about future trends and understand the impact of seasonality and holiday effects on the virus's spread.

Mohammed O, El Hassani I, Benabbou F, and Ouhbi B. declare prophet derives from 3 main functions then it creates the prophet itself.[2]. Briefly, The Prophet model is defined as:

$$y(t) = g(t) + s(t) + h(t) + \epsilon t$$

formula depicting prophet consists of function chains

G(t): The trend function is represented by g(t), which is piecewise linear or logical growth to deal with non-periodic variations in the value of the temporal sequence. The given formula for model was

$$Q(t) = \frac{K}{1 + \exp(-b * (t - a))}$$

Logistic Growth Formula

K: Cap Value(Carrying Capacity)

b: Shows the growth rate of epidemic

a: Shows fastest growth rate. After this point, curve will have a S shape.

The formula given above, which basically is used for linear growth model.

S(t): represents the periodic alterations as a week and/or year seasonality. Fourier series are used to be able to comprehend seasonality effects.

$$s(t) = \sum_{n=1}^N \left(a_n \sin\left(\frac{2\pi nt}{P}\right) + b_n \cos\left(\frac{2\pi nt}{P}\right) \right)$$

Fourier series

The formula given above is basically a Fourier series representing the weekly seasonality impact on machine learning, which can be also converted to monthly or annual seasonality effects, as well. the formula is used in the Prophet model to detect repeating patterns in the data.

t: Represents the time variable, typically measured in days.

P: it can decide how long it will remain those seasonality effects, which might be a year so in that case P should be equal to 365.

N: It shows how many Fourier terms are utilized in the series. Larger N values means you can detect complex pattern. If N is too high, it is likely to deal with overfitting issue.

An an Bn: they are the coefficients of sine and cosine. In training process, the model decides value of those coefficients.

H(t): represents the effects of holidays that occur on irregular schedules over a day or more. You can add them to the model and can even create your irregular holidays too like weekend holidays.

$$h(t) = \sum_j H_j \cdot I_j(t)$$

Holiday Indicator Function

Hj: Holiday impact

Ij: Shows existence of the holiday with "1" and "0"

While Hj illustrates holiday impact, Ij is basically associated with whether there is a holiday for a given time at a given holiday on Hj, which exactly works like one hot encoding transformation. Ij has the value '0' if there is no holiday on that day. if there is a holiday, it takes '1' value. This binary format is consistent for every holiday included in the series.

- **εt:** represents any unusual change not accommodated by the model, which basically a part of prophet function holding white noises (random errors).

The parameters of the model are calculated with the help of Bayesian alteration points and optimization algorithms. Bayesian algorithms are used to detect changes in trends. Then, these points are included in the model.

All in all, Prophet is an iterative machine-learning model. The final prophet module consists of all of those functions. The following mathematical formula is one example of how prophet function is going to look like

$$Y(t) = \frac{K}{1 + a * e^{(b*t)}} + s(t) + \sum_j H_j \cdot I_j(t) + \epsilon t$$

Basic Representation of Prophet Formula

3.5 Cap Value Estimation of Logistic Growth Model by using formulas

$$Q(t) = \frac{K}{1 + \exp(-b * (t - a))}$$

Logistic Growth Formula

K: Cap Value (Carrying Capacity)

b: Shows the growth rate of epidemic

a: Shows fastest growth rate. After this point, curve will have a S shape.

The formula is another derivation of the logistic growth model. “a” is constant, K maximum number of cases, and “b” is the incubation rate. The parameter that will be estimated is K(Cap Value) since it represents the cap value. When tfast=a(fastest time of the growth), the curve starts to have an S shape. After this point, the growth rate at cumulative commences to get slower and slower. Parameter estimation starts by assigning random values for a, b, and K. Then, it will be decided by using the Nonlinear Least Squares method and will be updated. “t=a” can also be the fastest growth rate since the curve is going to change its trend over time. If the current day is lower than tfast, which means the growth rate raises exponentially like t_max_train(time is going to be used to estimate cap value) = t_fast_train(fastest growth on time) * M (a number deciding how many times will increase). Otherwise, it can be understood that the growth is controlled. When the maximum number of cases is decided as t_max_train = current_day_train + N (number of days). After cap values are determined, it is presented with % a 95-interval width to the user. Later on, calculated Qtop(Cap Value or K) values are used to be able to feed the machine learning model. Regressor, holiday, or seasonality effects do not be included in Qtop’s calculations.

$$Q(t_{max}) = \frac{K}{1 + \exp(-b * (t_{max}))}$$

Logistic Growth Formula on tmax implementation

3.5.1 Nonlinear Least Square Method

When it comes to nonlinear least square method, the main purpose is to minimize difference between predicted value and observed value.

$$S(\theta) = \sum_{i=1}^n (y_i - \hat{y}(t_i, \theta))^2$$

Nonlinear Least Square formula

n: observed values

y: predicted value

“n” is the number of observed values. Yi is the value of observed values. Other “y” is the predicted value. For the whole formula, the initial chosen values for a, b, and k might be three

different random values in the beginning. initial of “a” might be the maximum of observed values or might be close to that. When it comes to “b”, it is likely to be time series mean. On the other hand, the initial value of k can be chosen as the typical growth value.

It is mentioned that the main purpose is to minimize the value obtained from the function. Therefore, some optimization algorithms can be used such as “Levenberg-Marquardt” algorithm.

$$S(a, b, k) = \sum_{i=1}^n \left(y_i - \frac{a}{1 + e^{-k(t_i - b)}} \right)^2$$

Logistic Growth Formula in Non-linear Least Square Formula

Derivation mathematical operation is implemented to the function to make it “0” but the whole process is complicated. Instead of implementing this, this problem can be solved with the help of “`scipy.optimize.curve_fit`”. It basically finds us values we looked for “a”, “b”, and “k” instead of solving complicated mathematical equations. All in all, the curve fit function in Python is so useful to be able to reach the best parameter for a, b, and k. Then, the obtained values provide an opportunity to find the right Qcap (Cap Value or K) value. Then, we will be able to acquire the tfast (fastest moment of growth) by using these values. That’s why, the nonlinear least square method has critical importance to be able to find the most appropriate Qcap value for the model.

3.6 Data Split and Model Training

3.6.1 Data Split

In this section, it will be mentioned how data is split into training and testing sets and how data is decided to time intervals for model training.

The training data is selected from January 3, 2020, to June 30, 2022, which includes a period slightly beyond the emergence of the Omicron variant. The main reason why the time interval is picked there is a significant increase in the number of COVID-19 cases so that the model has a better understanding of the impact of such variants like omicron and other seasonal effects (the same seasonal effect will be repeated for 2 years).

The testing data was chosen from July 1, 2022, to May 31, 2023. This time frame was chosen to assess how effectively the model performed with data that was affected by the Omicron variation and data that was not, which helps the model to distinguish the existence and non-existence of variants. It will play an important role in the feature weight used on the model.

Data was not selected randomly. Time-based split is used as it is mentioned above. This approach guarantees that the model can accurately comprehend and reflect the impact of seasonality and holidays because training data involved 2 2-year period of time.

3.6.2 Model Training

After the data split is completed and the model is fed with Cap value, the model training phase starts, which involves operations like cross-validation and hyper parameter tuning to obtain better results and also ensures whether the model deal with overfitting or not.

3.6.3 Cross Validation

According to the Facebook Prophet documentation, Prophet provides a solid foundation which makes it easier to assess the performance of the model over different periods. In order to evaluate the predictive accuracy of the model, the data is divided into training and testing sets over a range of periods using the cross-validation process. This methodology allows the evaluation of the model's generalizability to unseen data and identifies potential opportunities for enhancement [5]. It basically helps to understand if the model memorizes the provided dataset. Overfitting is a serious issue that needs to be avoided in terms of the generalization ability of the model. If the model does not have sufficient generalization ability, there is no way the model has an accurate prediction for future

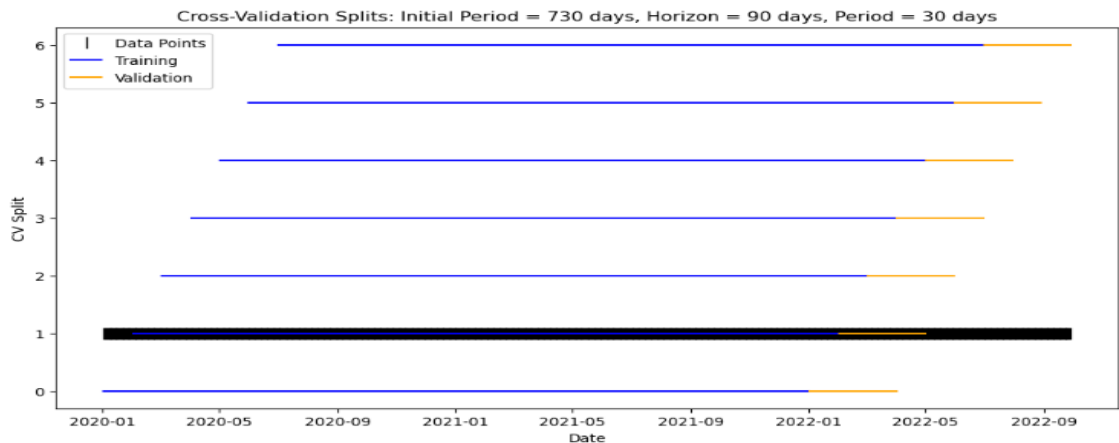


Figure 2. Cross Validation Working Principle on Plot, x-axis date, y-axis number of splits

There is a figure 2. indicating how cross-validation works over time. The black area shows the whole period, which is 730 days. The initial period involves a time interval of training datasets. Period for cross-validation decides how long data will shift over time. Just as the starting point for the initial is shifted forward by the given period, the end of the initial must be shifted forward by the given period so that the model will be able to come across data that will have never seen, which enables it to understand whether the built model is robust. When it comes to the “horizon” parameter for cross-validation, it basically shows how long the future the model will forecast.

3.6.4 Hyperparameter Tunning

According to Facebook Prophet documentation, Prophet is a powerful time series forecasting tool, and adjusting its hyperparameters can significantly improve its performance. Prophet's hyperparameters, such as `seasonality_prior_scale` and `changepoint_prior_scale`, control how flexible the model is when fitting data. For instance, the `changepoint_prior_scale` parameter affects how sensitive the trend is to changes, allowing different levels of flexibility in response to sudden shifts in data patterns [5]. Before using grid search, the model automatically used their default values. The main purpose of grid search is to find better parameters than default values used on model. Therefore, it should be defined in the list. Then, it must be tried every combination of internal parameters while changing their value on the defined list used for each parameter.

4. Data Collection, Preprocessing and Exploratory Data Analysis

To begin with, daily official UK data was collected using the National Health Service. The initial dataset provides information about daily and cumulative number of cases, first, second, and third dose vaccination numbers, testing numbers and positivity rates, reinfections, and hospital

admissions. Key variables in the dataset include `daily_cases`, `cum_cases`, `new_first_episode`, `cum_first_episode`, `new_reinfections`, `cum_reinfections`, `new_pcr_test`, `cum_pcr_test`, `test_roll_pos_pct`, `test_roll_people`, `new_first_dose`, `cum_first_dose`, `new_second_dose`, `cum_second_dose`, `new_third_dose`, `cum_third_dose`, `new_admissions`, and `cum_admissions`.

It is noted that in handling the impact of lockdowns, specific events like the omicron variant were flagged and included as regressors in the model. These flags are seen as part of the involved future of the model. Additionally, vaccination data is directly used as a regressor, which helps to model how increasing the vaccination rate affected the spread of the virus. Using a regressor means selecting features influencing model prediction. In your main dataset, there is no importance to what you involved on the main dataset. The only important thing is which features are involved as regressors from the main dataset.

The main focus for prediction in this project is to forecast a cumulative number of cases because cap value or logistic models are concepts that suit better cumulative cases. The dataset consists of 2,754 rows and 18 columns and has an overall missing data rate of approximately 25.25%. The missing data primarily originates from the unavailability of data on certain dates, rather than errors in the data collection phase. The collected data had data from 4 different regions which are England, Scotland, Wales, and Northern Ireland. However, countries like Wales and Northern Ireland had a myriad of data pieces filled with forward fill and backfill, which means most of the data for these countries was far away from real values. Therefore, these countries' data was not involved in the main dataset used for training model.

After Scotland and England's National Health Service data were merged. Then, their missing values were checked. The main reason for missing values on merged data is there was actually no available data. Due to this fact, there was no point filling them with any method requiring "backfill", "forward fill", "mean filling", "mode filling" etc. `daily_cases`, `cum_cases`, are filled, `new_pcr_test`, `cum_pcr_test`, `test_roll_pos_pct`, `test_roll_people`, `new_first_dose`, `cum_first_dose`, `new_second_dose`, `cum_second_dose`, `new_third_dose`, `cum_third_dose` with "0" since there were no actual values for these dates. The collected data is in a time interval which is between "03/01/2020" and "02/10/2023". There was also another issue about the collected data. Upon examination of the vaccination data, it was seen that the `first_dose`, `second_dose`, and `total_booster` vaccination data's starting date is in the same starting date. Logically, the first dose, second dose, and booster dose cannot have the same start date, which creates an issue not logical. It was basically requiring data shifting. After the detection of starting points of the "first dose", and "second dose" via the NHS website [13], time shifting is decided for the second dose according to NHS information [13]. It is also available information on when the total booster campaign started [14].

In the beginning, the starting date of the second dose and booster dose features is decided and saved as a mini dataset after date adjustments were done according to NHS. Then, shifted second dose and booster datasets were replaced with the original dataset's second dose and booster dose features. Then, a more reliable dataset was obtained with the help of the operation. At the same time, critical dates and events were examined for "United Kingdom". It was found out Lockdown dates were significant dates for COVID-19.

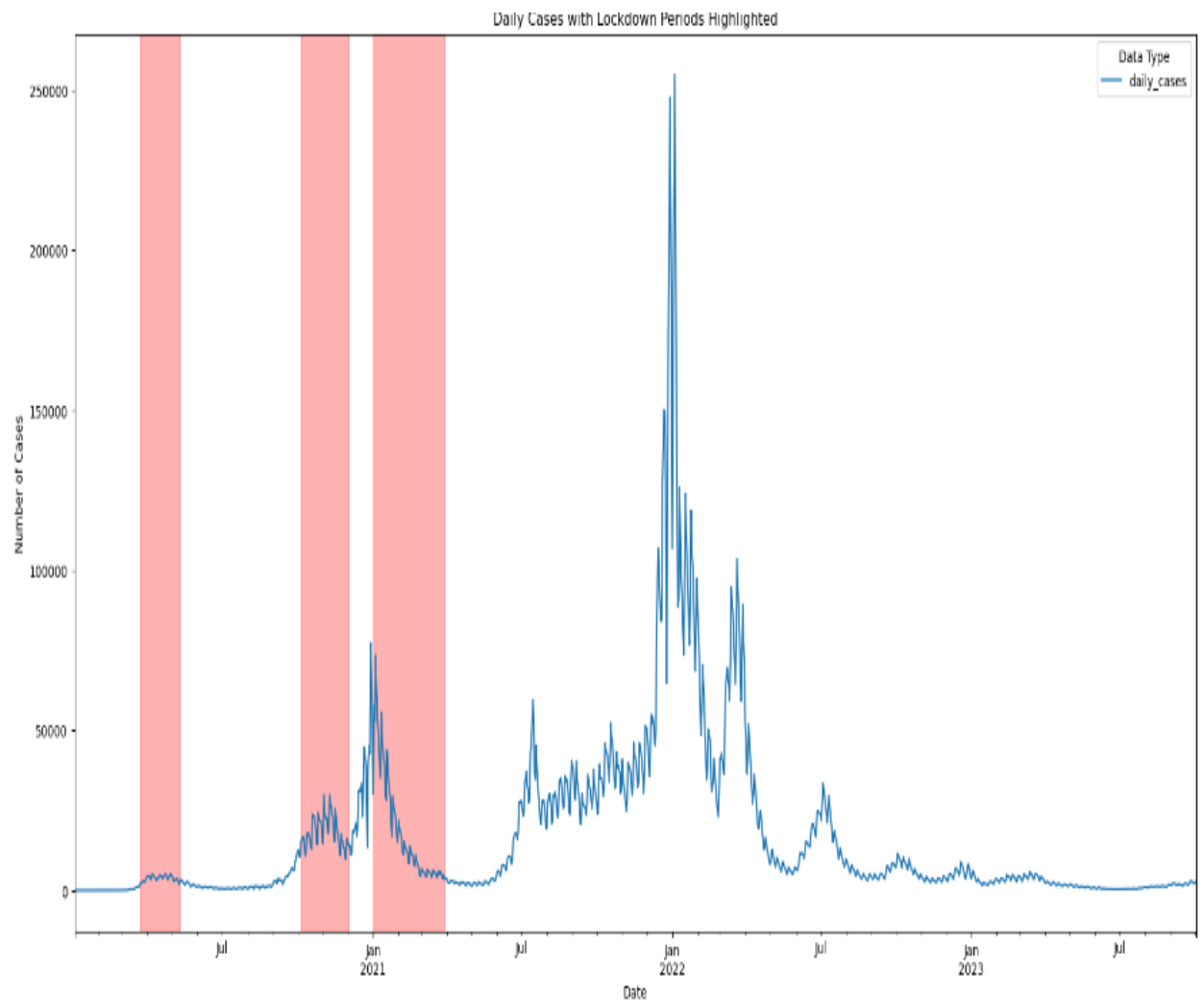


Figure 3. via pre-processed data Covid 19 number of daily cases in United Kingdom, 2021-2023

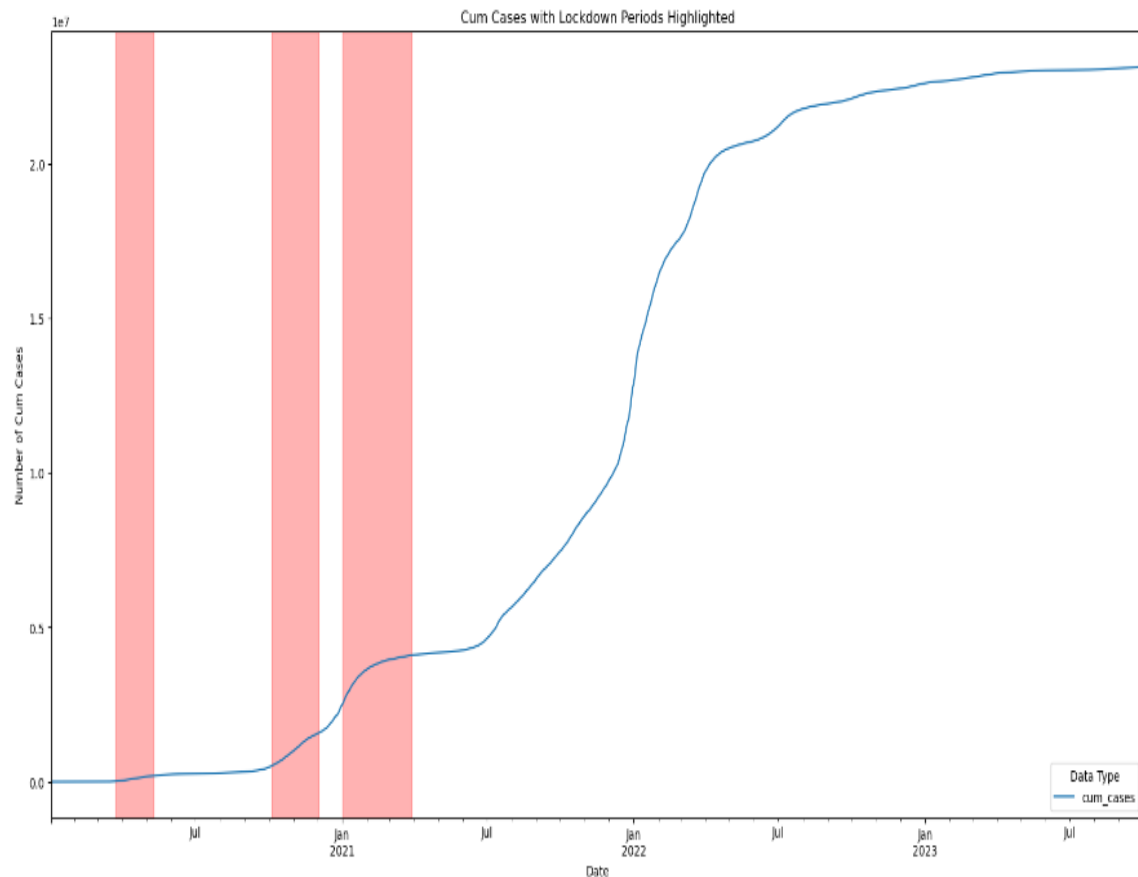


Figure 4. via pre-processed data Cumulative Cases in United Kingdom, 2021-2023

They depict the effect of lockdown dates on both cumulative and daily cases (in blue), which shows whenever a lockdown (red shaded areas) decision has been made by the government over the period 03/01/2020 to 02/10/2023, the number of cases indicates a decreased tendency on growth trend on COVID-19 cases. Lockdown means people cannot go out unless they go shopping, which is highly advised to people to stay in their homes. The information is basically a great sign of why lockdown was important for COVID-19 in order to prevent future cases. Therefore, the dates when lockdowns were implemented by the United Kingdom are added to the main dataset by flagging these crucial dates as “First Lockdown”, “Second Lockdown”, and “Third Lockdown”.

The figure also shows a dramatic rise in the number of cases of covid-19 in January 2022. This significant increase can be attributed to the Omicron variant, which was highly contagious.

That’s why, the period involving omicron is flagged again just like Lockdown dates to be able to be used on the Machine Learning Model again. Flag means time interval involving omicron will be shown on the dataset as “1”. On other dates the omicron is not active shows as “0” on the new omicron feature of the dataset.

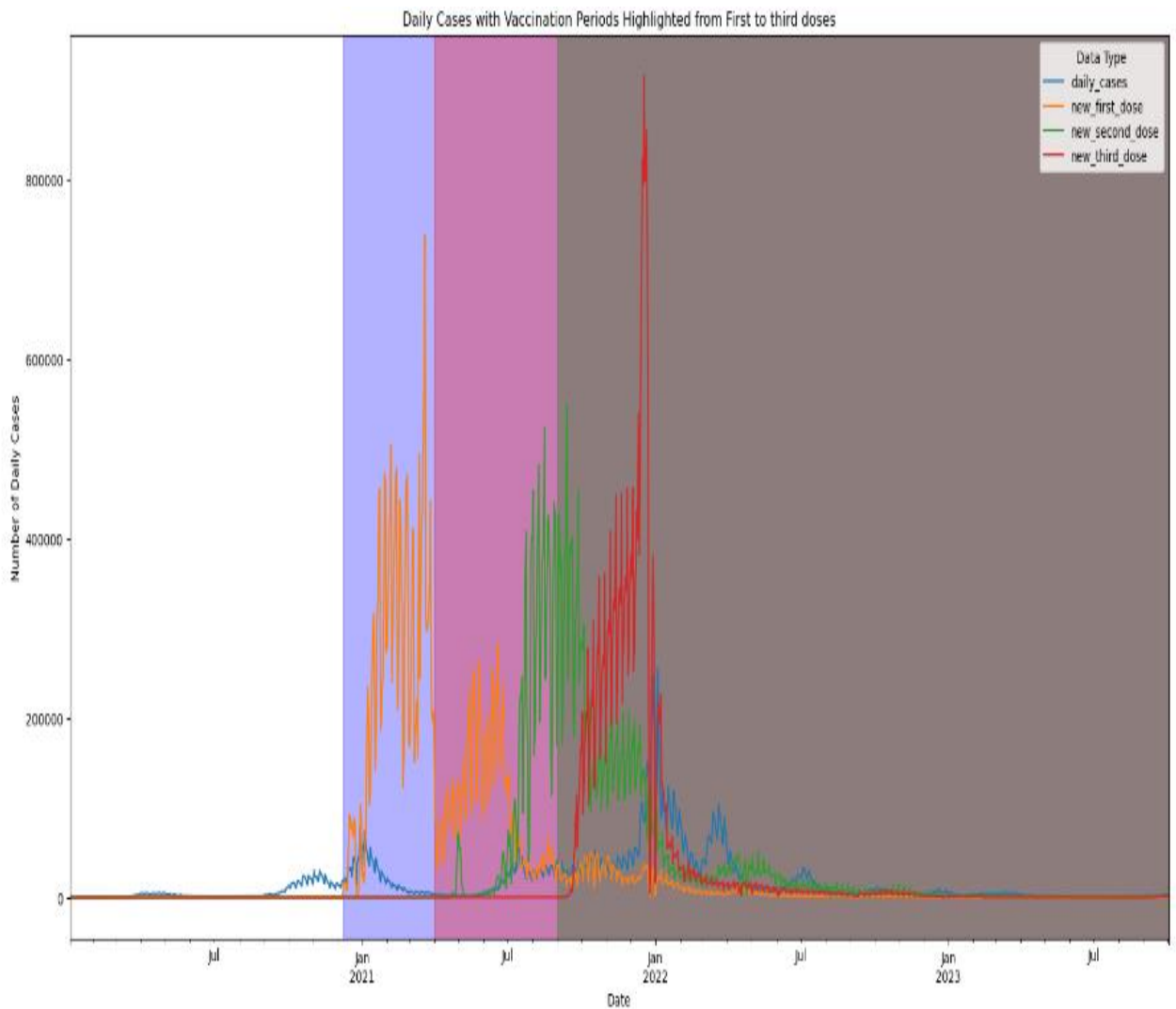


Figure 5. Covid-19 daily number of vaccinated people from the first dose to booster dose in UK, 21-

23

Once you look at the figure, you might notice when the number of vaccinations is increased, there is always a decrease after a certain period of time in daily cases, which basically shows the importance of vaccination. There is also another crucial thing in the figure. You can see there is a huge vaccination number for booster doses, which is in almost the same period as the omicron variant. It is the reason why omicron suddenly decreased despite the number of cases being increased significantly, which is another benefit of vaccination to be able to prevent the spreading of COVID-19 in the population.

These illustrations show why vaccination data should be shifted correctly and involved in the main training dataset.

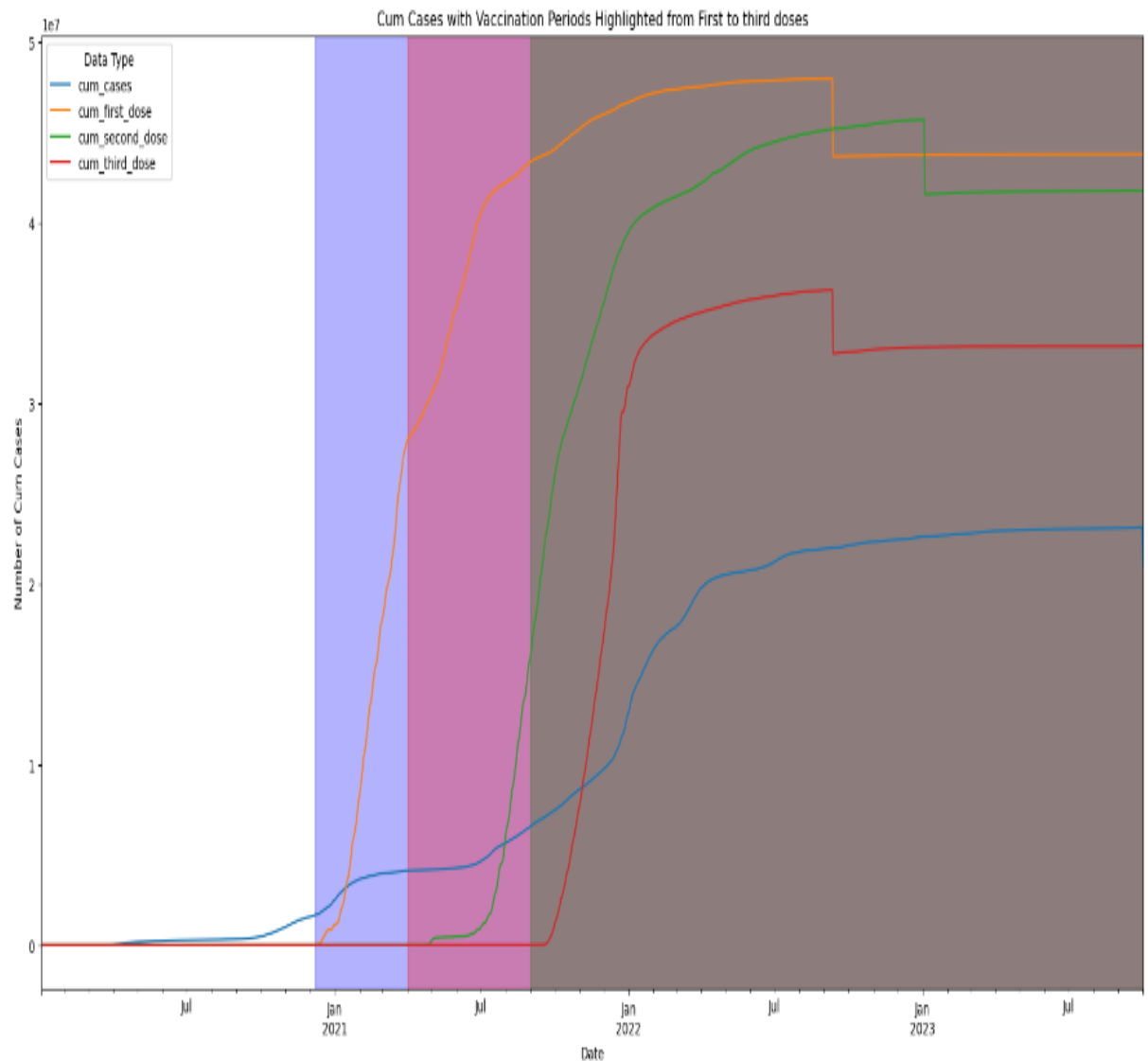


Figure 6. Covid 19 number of cumulative vaccinated people in United Kingdom, 2021-2023

Findings in the figure show the effect of vaccination cumulative cases (orange, blue, red, and green lines) from 03/01/2020 to 02/10/2023.

It is noted that all vaccination periods go up to 02-10-2023, but since the shaded areas overlap, the End date of the first dose and second dose are not clear. As a result, starting points can be distinguished for all shaded areas.

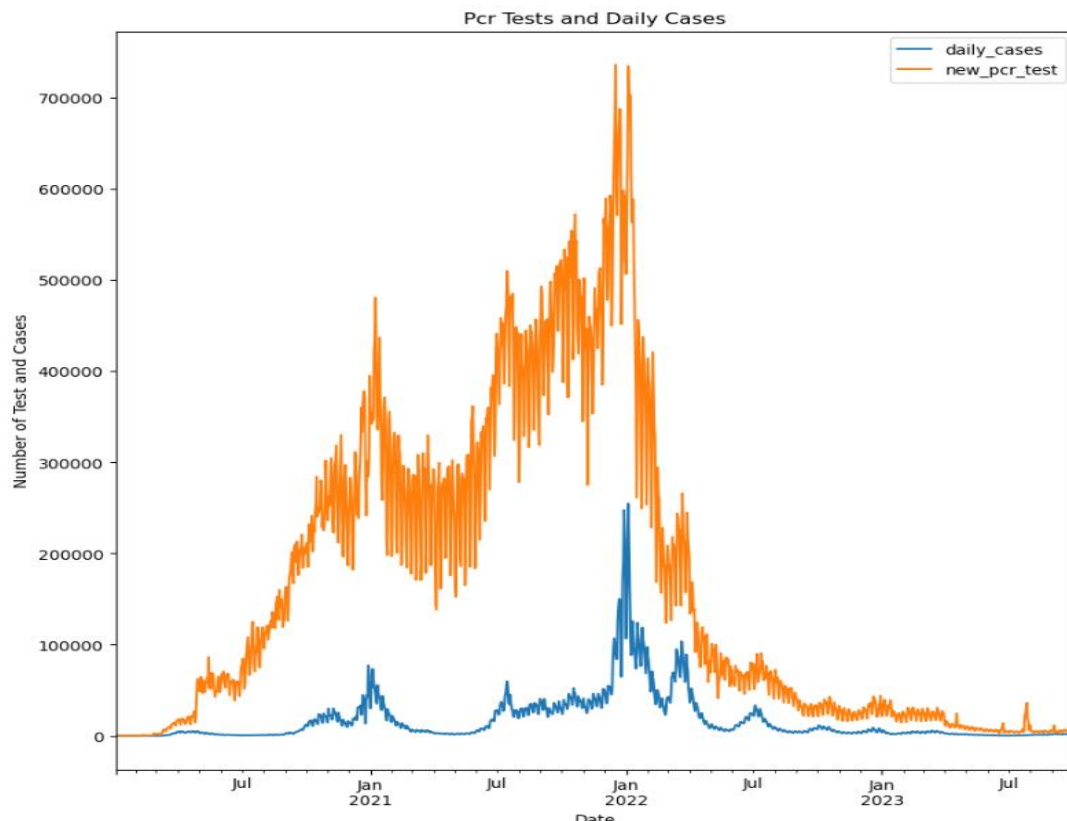


Figure 7. Covid 19 number of daily cases and tests in United Kingdom, 2021-2023

Figure 7. Covid 19 number of daily cases and tests (blue and orange lines respectively) in United Kingdom, 2021-2023. You can also see number of PCR tests (orange line) is highly associated with the number of daily cases (blue line). According to these figures, the number of tests is also involved in the main dataset, as well. In addition to that, COVID-19 positivity rate data were also derived by dividing daily cases and PCR tests to each other so that the rates were acquired and involved in the dataset too. There are also cumulative tests seven-day averages were added to the main dataset. All in all, Omicron flagged dates, Lockdown flagged dates, positivity rate, vaccination numbers, positivity, and a number of PCR tests are the most crucial data on the dataset, which is pretty important to adapt to dates that are flagged correctly to the dataset since those data make an important impact on the number of daily cases in the United Kingdom.

5. Seasonality and Holiday effect with Exploratory Analysis

Time series analysis is a method of examining changes in time and providing a prediction about the future. Two of the most important assets for this analysis are the holiday and seasonality effect. In general, time series tend to show fluctuations due to seasonal changes. Therefore, it is quite vital to consider their effects on the analysis so that it can significantly help to increase model accuracy for its future predictions.

In the next section, I will analyse seasonality and holiday effects in the data. First, it will cover the definition and importance of seasonality, then examine the effects of holiday periods. These

analyses will make a contribution to be able to understand regular and irregular patterns in the data and provide insights.

5.3 Seasonality Analysis

The daily change in the number of cases of the COVID-19 outbreak is of great importance to health authorities and the public. Therefore, how to analyse the number of cases by day can provide critical information for the control and protection of the outbreak. In this analysis, a graph showing the change in the average daily number of cases over a week has been

examined. This analysis will help us understand that weeks of COVID-19 cases do not show specific temperature increases or losses.

5.3.1 Weekly Trends

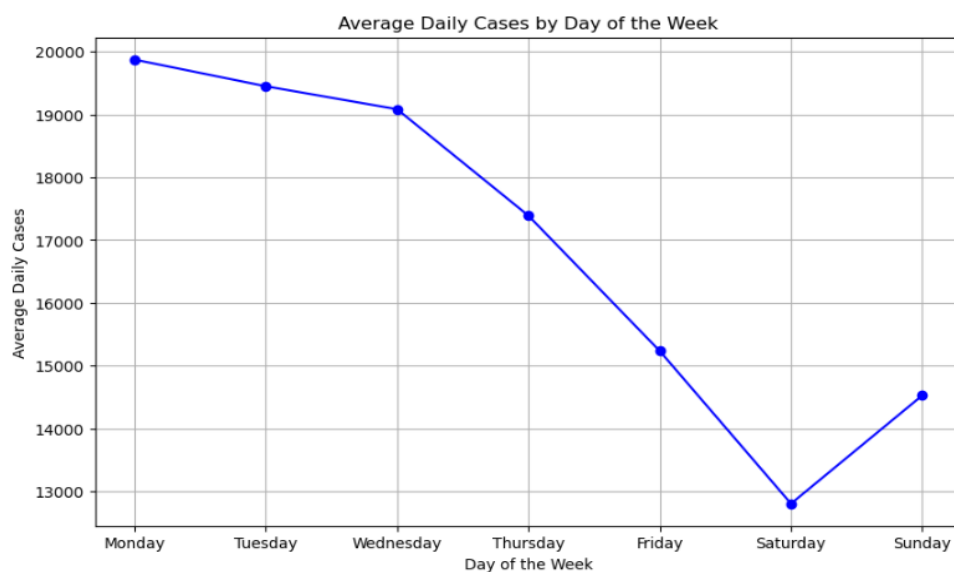


Figure 8. Weekly Number of Daily Cases Trend in United Kingdom, 2021-2023

When it comes to the figure while the y-axis shows the day of the week and, the x-axis depicts average daily cases. For each day, a number of average cases is given with blue dots. As you can see the number of cases shows a decreasing trend from Monday to Saturday. Then, increases again from Saturday to Sunday. This can be explained by the fact that the results of the tests performed over the weekend were reported on Monday. Tuesday and Wednesday shows a slight decrease but the number of cases is still considerably higher than other days. The high number of cases these days may be due to the ongoing intensive testing and reporting activities at the beginning of the week. Despite Saturday having the lowest number of cases, it can be explained that This can be explained by the fact that most of the testing centres are closed, or the number of tests has decreased, or people do not go outside since they work so that the virus spread rate is less than any other days.

To sum up, according to the figure, it reveals that the number of COVID-19 cases shows a significant seasonality on a weekly basis. The number of cases decreases from Monday, reaches its lowest level on Saturday, and then increases slightly on Sunday. It can be said that this seasonality pattern is related to two things which are “the intensity of testing and reporting activities varying according to

the days of the week” and “people show a tendency to go outside on weekends then it triggers the virus spread more”. That kind of pattern is involved in the machine-learning model at Prophet.

5.3.2 Fluctuations over months

The annual change in the number of cases during the COVID-19 pandemic is of great importance in terms of the management and control of the outbreak. Understanding the fluctuations in the number of cases on an annual basis can provide critical information about the measures and strategies to be taken during different periods of the outbreak. In this section, the graph showing the change in the number of COVID-19 cases over a year will be analysed.

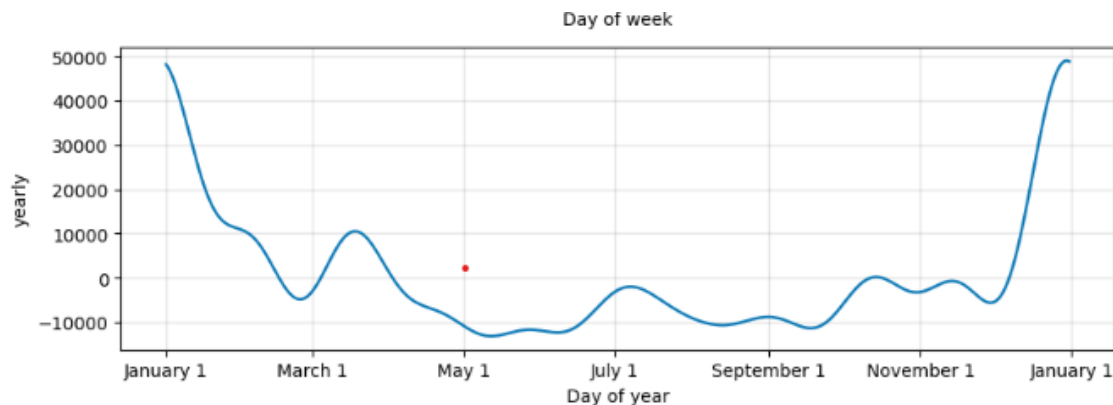


Figure 9. Daily Case Fluctuation over year in United Kingdom, 2021-2023

Firstly, the horizontal axis provides information “the day of the year. When it comes to the vertical axis, it shows a number of daily cases annually. Fluctuation and changes in the figure are shown with a blue line. Those changes can be computed from the beginning to the end of the year on the figure.

The reason why January has the highest number of cases originates from increased contact and mobility due to New Year celebrations and the holiday seasons. In addition to that, there is a spike on Good Friday at the end of March. When it comes to the period from May to August, the number of cases reaches the lowest number of cases. The ultimate reason why it happens is people would rather join outside activities rather than limit themselves to any inside activities. The situation prevents the spread of the virus, which results in coming across less cases. After the weather starts to get colder and colder. Then, people opt for more indoor activities, not the outside. The same loop happens again and again. As a result of the situation, the yearly seasonality effect is a factor that is needed to be involved in the model.

5.4 Holiday Analysis

The impact of holiday periods on the case is of great importance in terms of sustaining and controlling the dynamics of outbreaks. Holiday periods are times when people's social interactions increase and therefore the risk of infection increases. Therefore, understanding and analysing of holiday effect has the critical importance for time series.

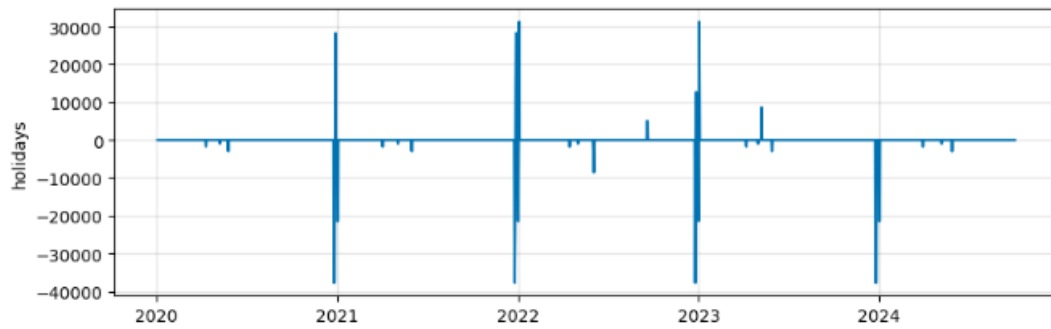


Figure 10. Holiday Effect on Daily Cases in United Kingdom, 2021-2023

While the y-axis shows the year, the x-axis shows how daily cases have changed over time for given holidays. These days are chosen according to the United Kingdom. Every January for each year shows a dramatic increase in the number of cases. “New Year (long break)” can encourage people to go outside and spend their time in indoor areas due to the weather conditions. Because it is likely to see more cases at the beginning of the year. The graph proves it. There is also a special function at prophet. It provides an opportunity to add these days for specific countries to the model automatically with the help of its library. Holidays for the United Kingdom are given in the below figure.

```

0          Good Friday
1          May Day
2          Spring Bank Holiday
3          New Year's Day
4          Christmas Day
5          Boxing Day
6          Boxing Day (observed)
7          Christmas Day (observed)
8          Platinum Jubilee of Elizabeth II
9          State Funeral of Queen Elizabeth II
10         New Year's Day (observed)
11         Coronation of Charles III
dtype: object

```

Figure 11. Prophet default Holiday List in United Kingdom

These days are flagged with holiday effect function of prophet. You can even add your own holidays to the list and can expand it. Figure 11. shows the formal holiday for United Kingdom. These special days were used on the model as a part of holiday effect.

6. Logistic Growth and Prophet Model Integration: Methodology and Block Diagram

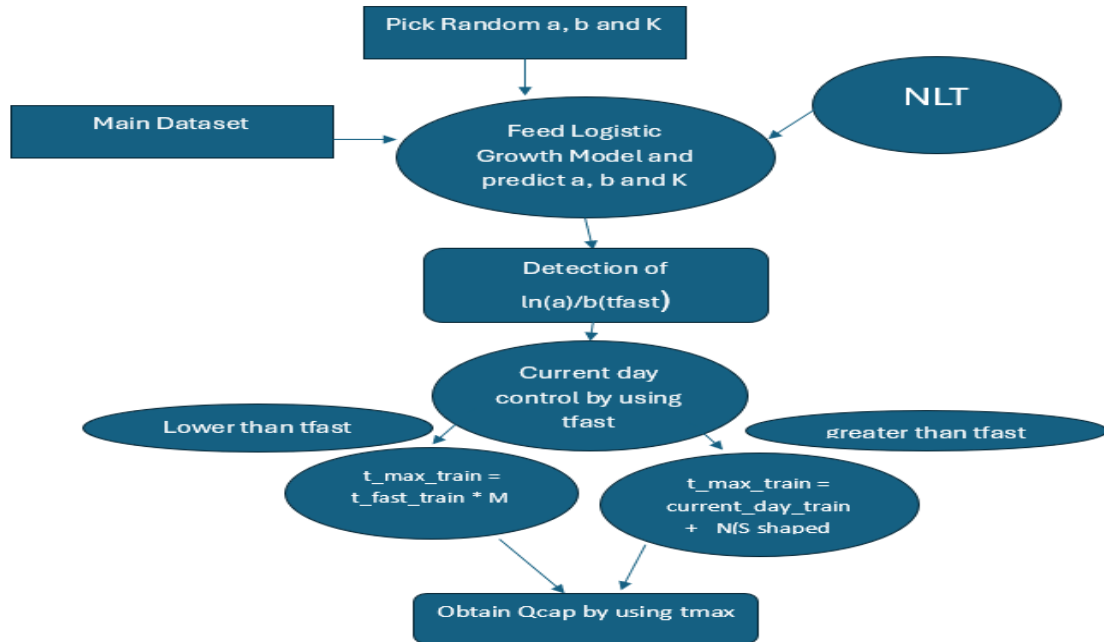


Figure 12. Block Diagram for Logistic Growth

Figure 12. is a block diagram of the logistic growth model used on the machine learning model. The main purpose of the block diagram is to show how the Qcap value is obtained. If the block diagram is summarized, a, b, and k are selected randomly on the dataset with the help of the target feature. Later on, logistic growth model is fed the Nonlinear least square method to be able to determine tfast value which will be used in the current comparison with tfast day. If tfast occurs in the time before the current day than the “lower than tfast” method is going to be executed. Otherwise, “greater than tfast” will be a function which is going to be executed. After, the tmax value is adjusted with the aid of these methods. Then, tmax will be used for Qcap calculation. In the end, the obtained Qcap will be part of machine learning model when it comes to the model training phase. To sum up, the whole block diagram can be simplified as a “logistic growth model determining Qcap”, which is the name of the figure.

$$Q(tmax) = \frac{K}{1 + a * e^{(-b*tmax)}}$$

Logistic Growth Formula on tmax implementation

Above equation shows how Qcap or Qtmax will provide ultimate feeding value for machine learning model.

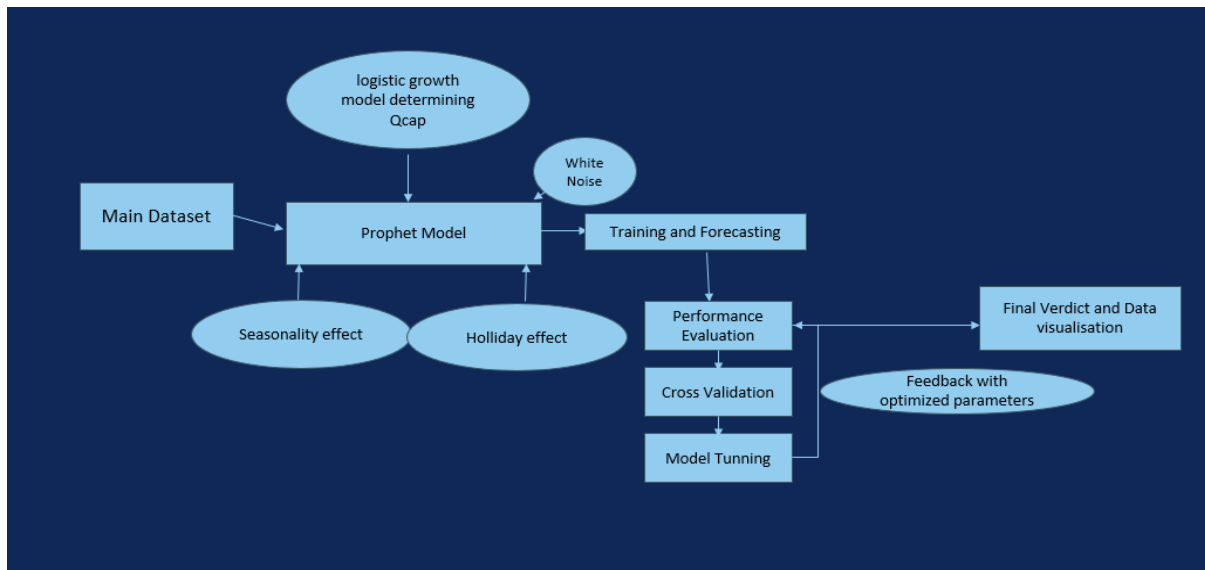


Figure 13. Prophet Model Block Diagram with Logistic Growth Cap Feeding

Figure 13 shows the working principle of the whole project. As mentioned above the model is going to be fed with the appropriate Cap value. At the same time, the main dataset should be divided into two parts which are train and test. The model is also fed by “seasonality”, “holiday”, and “white noise” as well as fed by the “logistic growth model”. It will also add regressors to the model. Regressor is basically featured acquiring on the dataset. Then, those features are used for the training process on the machine learning model. It is critical to pick features that do not provide a direct answer to models like daily cases. Otherwise, it can trigger overfitting, which is why regressor selection has critical importance for the prophet model. They are integral parts of the model and be explained in their mathematical expression, as well. After completion of the feeding process, the model is trained, and predicted future values are according to the given forecasted period. Then, the models’ performance is evaluated. Then, cross-validation starts and decides which period is the best for the model in terms of obtaining high accuracy and performance. After these periods are decided with the help of the cross-validation process, the Model Tuning or Hyperparameter Tuning process commences. The hyperparameter tuning process should be matched with the best timeline provided by cross-validation. Model tuning will provide the best hyperparameter values for the model according to the timeline obtained during “the cross-validation process. Then, the model performance is evaluated again. The point is it should be seen model performance is better than its previous model performance. If the model performance is satisfying, the model prediction and actual values can be visualized, and their performance is evaluated again so that the whole process will help to find the time interval in which the model performs well and the required hyperparameter values. That’s why, it is so vital to implement processes shown on the block diagram for the sake of the machine learning model. It is important that repeated cross validation can result in overfitting. That’s why, it should be avoided repeated cross validation process.

7. Results and Performance Evaluation

First, the model training phase is completed, as detailed in the Methods section. The Cross-Validation method is applied to detect the best period for prediction, which is followed by

Hyperparameter Tuning to optimize model performance. The outcome of these processes show an improvement in the model's accuracy, which indicate how successful model tuning and validation process.

7.1 Performance Metrics

It is an important factor what are performance metrics, and they measure exactly. Performance metrics are important tools for evaluating the accuracy and performance of time series models. Horizon indicates the time period over which the model makes future predictions, and this evaluation is made for periods such as "2" days and "3" days. The performance metrics used in this study are MAPE (Mean Absolute Percentage Error) and Coverage, as they are particularly relevant for the goals of this research.

MSE (Mean Squared Error) expresses the average of the square of the prediction errors; a lower MSE indicates that the model fits the data better.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

Mean Squared Error Formula

y_i : real value

\hat{y}_i : predicted value

N : number of values predicted

RMSE (Root Mean Squared Error) is the square root of MSE and measures the average magnitude of the model's prediction errors. This metric is a more understandable measurement than MSE because its unit is the same as the unit of the predicted values.

$$RMSE = \sqrt{MSE}$$

Root Mean Squared Error Formula

MAE (Mean Absolute Error) shows the average of the absolute values of the prediction errors and indicates the average difference between the values predicted by the model and the actual values.

$$MAE = |MSE|$$

Mean Absolute Error Formula

MAPE (Mean Absolute Percentage Error) gives the average of the absolute percentage values of the prediction errors and allows comparison between data sets at different scales by being expressed as a percentage. MAPE is important for the project since it allows for clear interpretation of prediction accuracy.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100$$

Mean Absolute Percentage Error Formula

MdAPE (Median Absolute Percentage Error) indicates the median of the absolute percentage values of the prediction errors and is not affected by outliers, so it can be a more reliable measure when there are extreme values in the data set.

SMAPE (Symmetric Mean Absolute Percentage Error) is similar to MAPE but normalizes the difference between the predicted and true values, making the results symmetric.

Finally, Coverage indicates how often the model's predicted values fall within the prediction range that covers the true values and is usually expressed as a confidence interval percentage. Coverage is quite crucial in this study in the aspect of the reliability of predictions within a specified confidence interval. R squared data metric is not involved for performance metrics. The reason is R-squared performance evaluation method can be deceptive because of seasonality, and holiday effects on the dataset.

7.2 Observations

horizon	mse	rmse	mae	mape	mdape	smape	coverage
3 days	92305660000	303818.459	253460.4314	0.012578	0.015115	0.012543	0.555556
4 days	1.03154E+11	321176.514	268606.003	0.013313	0.015474	0.01327	0.444444
5 days	1.0271E+11	320484.596	272588.5146	0.013544	0.015474	0.013502	0.333333
6 days	1.08773E+11	329806.893	282770.8401	0.014069	0.015469	0.014019	0.333333
7 days	1.12895E+11	335997.93	287852.0251	0.014318	0.014895	0.014258	0.444444
8 days	1.1856E+11	344325.449	292573.2255	0.014524	0.014396	0.014455	0.555556
9 days	1.25108E+11	353706.581	299635.6089	0.014861	0.014389	0.014783	0.666667
10 days	1.34486E+11	366722.603	313744.6838	0.01559	0.014396	0.015502	0.555556
11 days	1.43686E+11	379058.933	330653.7064	0.016505	0.01452	0.016405	0.444444
12 days	1.52713E+11	390784.884	348362.5424	0.017485	0.01452	0.017372	0.444444
13 days	1.60591E+11	400738.449	362559.0453	0.018281	0.01414	0.018154	0.444444
14 days	1.67266E+11	408982.072	372865.5384	0.018875	0.01414	0.018731	0.444444
15 days	1.76607E+11	420246.115	384225.9065	0.019525	0.016225	0.019359	0.333333
16 days	1.893E+11	435085.965	398595.9933	0.02034	0.019209	0.020148	0.333333
17 days	2.05648E+11	453484.698	415200.4947	0.021271	0.022189	0.021048	0.333333
18 days	2.24758E+11	474085.962	433410.4172	0.022292	0.024762	0.022036	0.333333
19 days	2.42346E+11	492286.038	447984.9878	0.023125	0.028491	0.022838	0.333333
20 days	2.55442E+11	505412.868	457031.2616	0.023649	0.03093	0.023337	0.333333
21 days	2.6251E+11	512356.799	458440.3016	0.023746	0.031251	0.023418	0.333333
22 days	2.7027E+11	519875.07	459903.1601	0.023828	0.032087	0.023484	0.333333
23 days	2.82672E+11	531668.81	465501.2037	0.024135	0.032823	0.023771	0.333333
24 days	3.01641E+11	549218.38	475825.964	0.024698	0.033636	0.024304	0.333333
25 days	3.21293E+11	566827.392	485394.1018	0.025228	0.034565	0.024805	0.333333
26 days	3.37979E+11	581359.922	490031.82	0.025506	0.034925	0.025056	0.333333
27 days	3.49763E+11	591407.919	493025.105	0.025675	0.034971	0.025207	0.333333
28 days	3.58097E+11	598412.402	499703.6694	0.025998	0.035273	0.025518	0.333333
29 days	3.66706E+11	605562.32	512574.7492	0.026601	0.035372	0.026112	0.333333
30 days	3.75655E+11	612906.996	526937.6488	0.027264	0.035969	0.026765	0.333333

Best Period	Best Horizon	Best Coverage
60	30	0.39

Figure 14. Cross Validation Table with Default Internal Parameters with Train Data

The below figure shows the result of the model before hyperparameters are defined. The model has a 0.39 coverage rate in the beginning, which means the model covered %39 of the real values for the train dataset. That kind of rate is not enough when it comes to epidemic cases. Therefore, higher coverage has to be obtained.

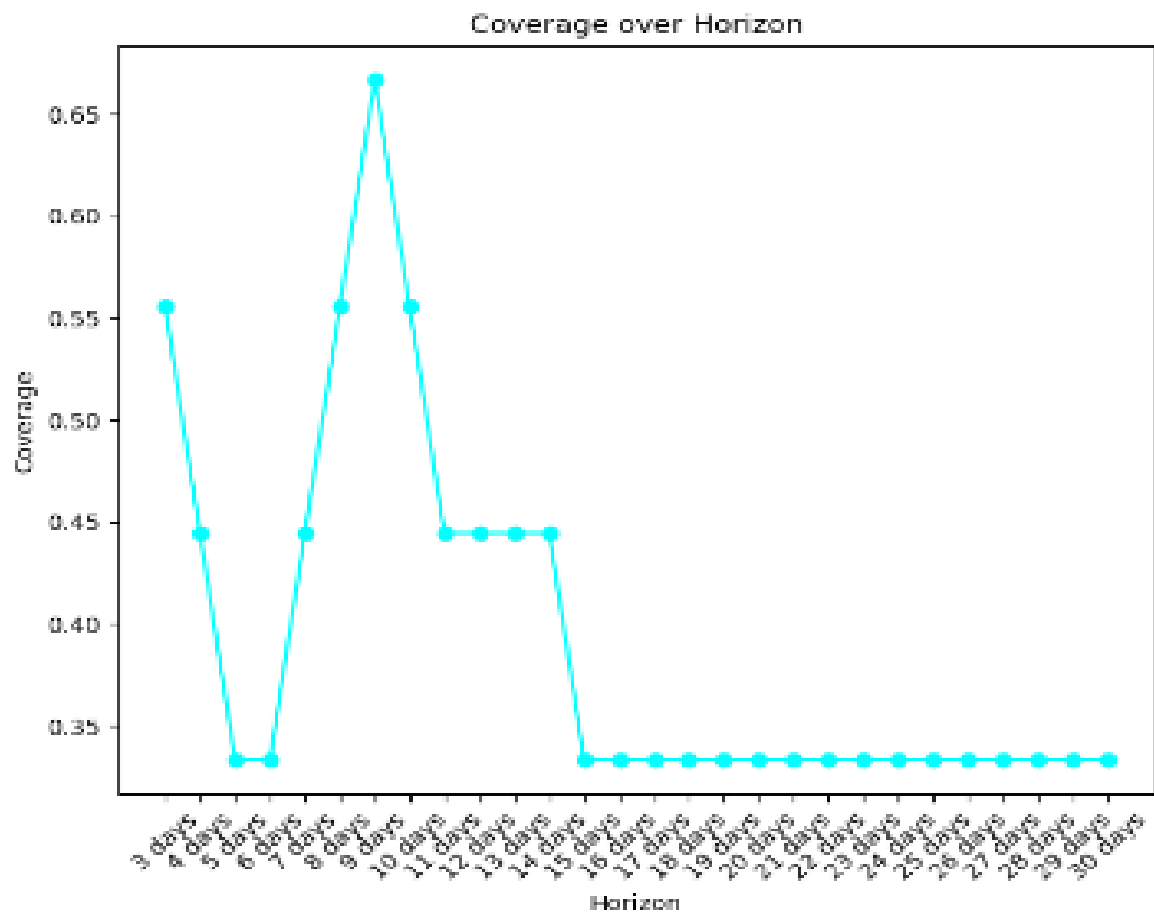


Figure 15. Coverage Metric on the best Horizon with Default Parameters with Train data

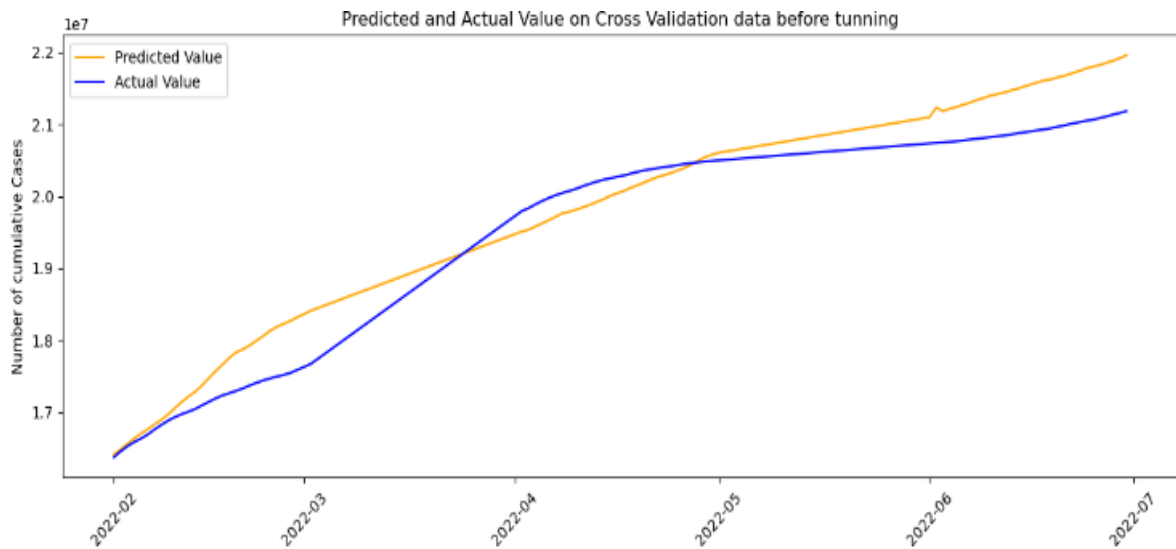


Figure 16. Comparison plot between predicted and actual values with default setting for Cros. Val. With Train data

Figure 16. provides a comparison between predicted and actual values for the given years on test data values. While X-axis depicts date from 2022-02 to 2022-07, the y axis shows the number of cumulative cases in the United Kingdom

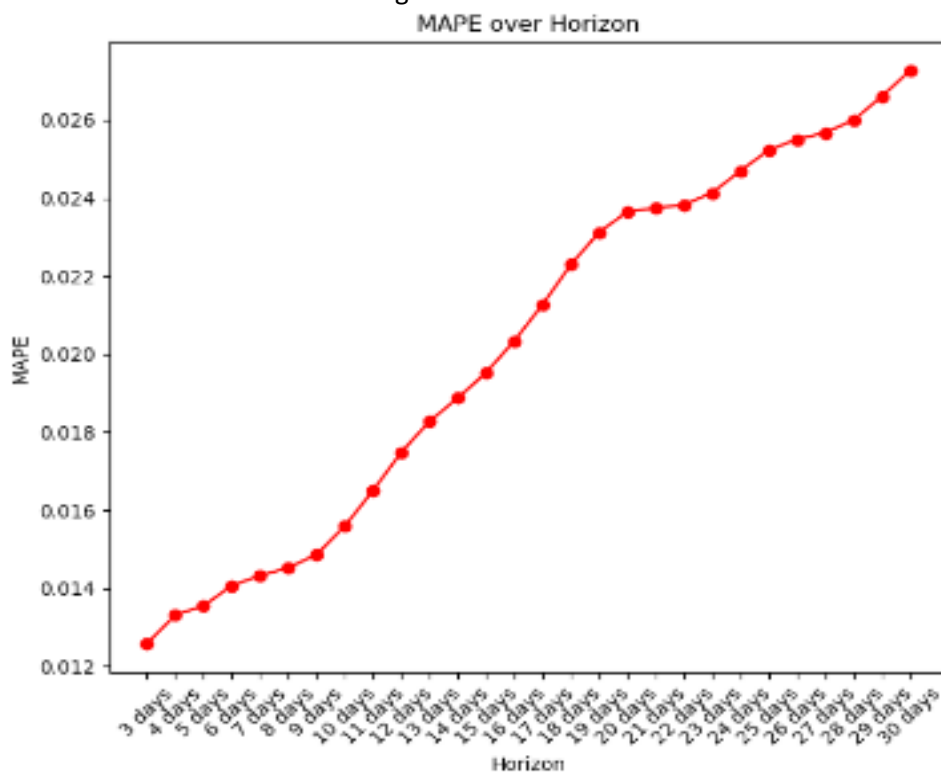


Figure 17. MAPE Metric on the best Horizon with Default Parameters on Train Data

the implemented model uses default hyperparameters instead of specified ones, the reason why a low coverage rate was obtained. Then, hyperparameter tuning is implemented to the model to be able to detect the best hyperparameters for the model in order to improve the outcome for

coverage or any other performance metrics like MAPE. Figure 17. shows mape and when you look at the figure you will see that mape always increases, which is totally expected since as time goes on number of errors is increasing, which means mape error is going to increase. Therefore, we obtain a graph continuously rising, which is why using a coverage error metric plot for further analysis is better in terms of the right interpretations.

Changepoint_prior_scale	Holiday_prior_scale	Seasonality mode	Seasonality_prior_scale	The lowest MAPE
0.5	20.0	multiplicative	30.0	0.007906

Figure 18. Hyperparameter tuning table on the best internal parameter values for Prophet on Train Data

The figure shows the best hyperparameters after completion of grid search. The lowest mape value has also been given.

horizon	mse	rmse	mae	mape	mdape	smape	coverage
3 days	725986900	26944.14402	19329.06766	0.001049	0.000769	0.001049	1
4 days	632450200	25148.56177	20854.44809	0.001077	0.000997	0.001077	1
5 days	699885000	26455.34066	23345.4986	0.001208	0.001027	0.001208	1
6 days	1182534000	34387.99341	33047.02864	0.001724	0.001971	0.001724	1
7 days	1732199000	41619.69459	40860.3185	0.002133	0.002089	0.002134	1
8 days	2652024000	51497.80735	49784.9937	0.002611	0.002233	0.002612	1
9 days	4336834000	65854.64241	62685.40493	0.003306	0.002891	0.003306	0.888889
10 days	6769241000	82275.39835	78876.22968	0.004175	0.004115	0.004175	0.777778
11 days	9113750000	95465.96213	92474.60514	0.004879	0.005301	0.00488	0.666667
12 days	11280860000	106211.4132	103951.7682	0.005446	0.006071	0.005449	0.666667
13 days	12625270000	112362.2369	109596.4837	0.005681	0.006071	0.005687	0.777778
14 days	11322500000	106407.255	102691.7961	0.005276	0.004688	0.005283	0.888889
15 days	13298800000	115320.4082	105298.862	0.005342	0.004688	0.005354	1
16 days	16538450000	128601.8943	109965.2765	0.005516	0.004688	0.005534	1
17 days	23393260000	152948.5378	126141.7291	0.00626	0.005015	0.006287	1
18 days	27786900000	166694.0173	134465.9161	0.006629	0.00572	0.006663	1
19 days	32772000000	181030.39	149802.8462	0.007405	0.006461	0.007445	1
20 days	38506720000	196231.2805	170964.9647	0.008539	0.007166	0.008587	1
21 days	45671130000	213708.0501	194327.2969	0.009815	0.007171	0.009873	0.888889
22 days	53488610000	231276.0409	214064.333	0.010888	0.009764	0.010956	0.777778
23 days	62491630000	249983.2678	231883.227	0.011839	0.010945	0.011919	0.666667
24 days	71463620000	267326.8102	248475.7234	0.012709	0.012539	0.012802	0.666667
25 days	81186190000	284931.9114	266407.885	0.013664	0.01417	0.013771	0.666667
26 days	92013970000	303338.0412	285678.8531	0.014698	0.016686	0.014821	0.666667
27 days	1.06041E+11	325639.7005	306927.0256	0.015856	0.019253	0.016	0.666667
28 days	1.2285E+11	350499.6845	328110.1348	0.017012	0.020242	0.017182	0.666667
29 days	1.38813E+11	372576.2208	345553.2115	0.017958	0.022077	0.018152	0.666667
30 days	1.52994E+11	391144.8874	359368.7645	0.01869	0.023056	0.018905	0.666667
Best Period		Best Horizon	Best Coverage				
60		30	0.845238095				

Figure 19. Cross Validation Table with Hyperparameter Tunning Parameters on Train Data

As you can see the model performance has been developed more after hyperparameters are applied to the model. The coverage increases from %39 to %84.5, which is a considerable improvement in the aspect of the model performance.

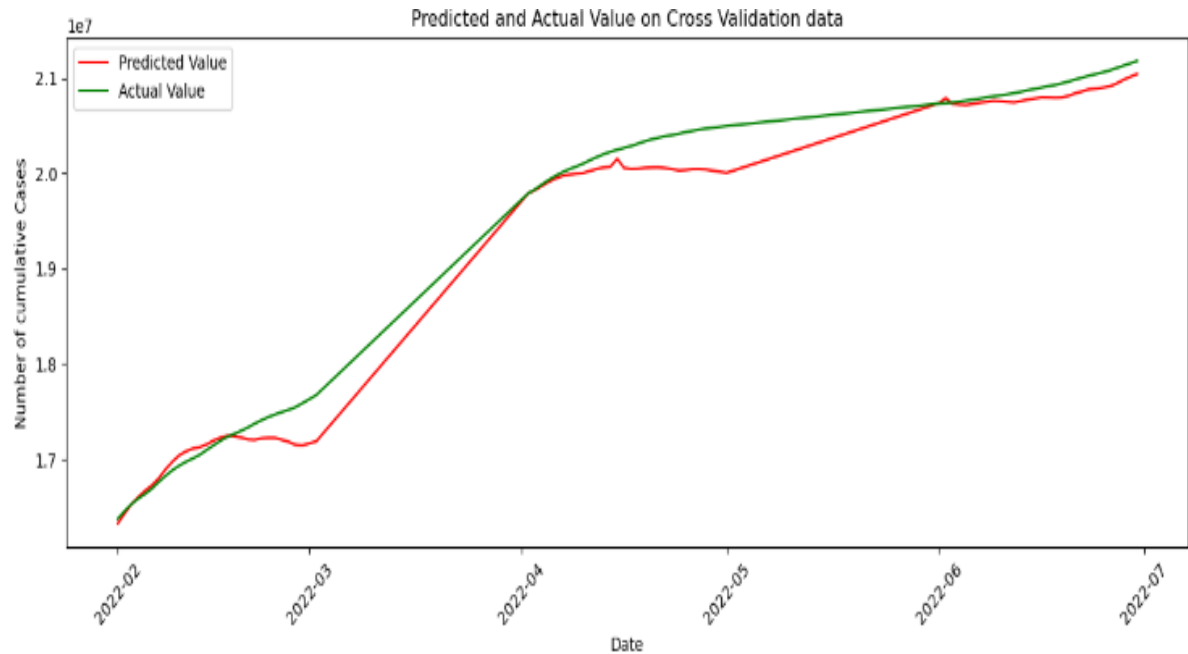


Figure 20. Comparison plot between predicted and actual values with Model Tuning settings for Cros. Val.

Figure 16. and Figure 20. Depict comparison between predicted and actual values based on real data from 2022-02 to 2022-07 by updating for every 60 periods on train dataset.

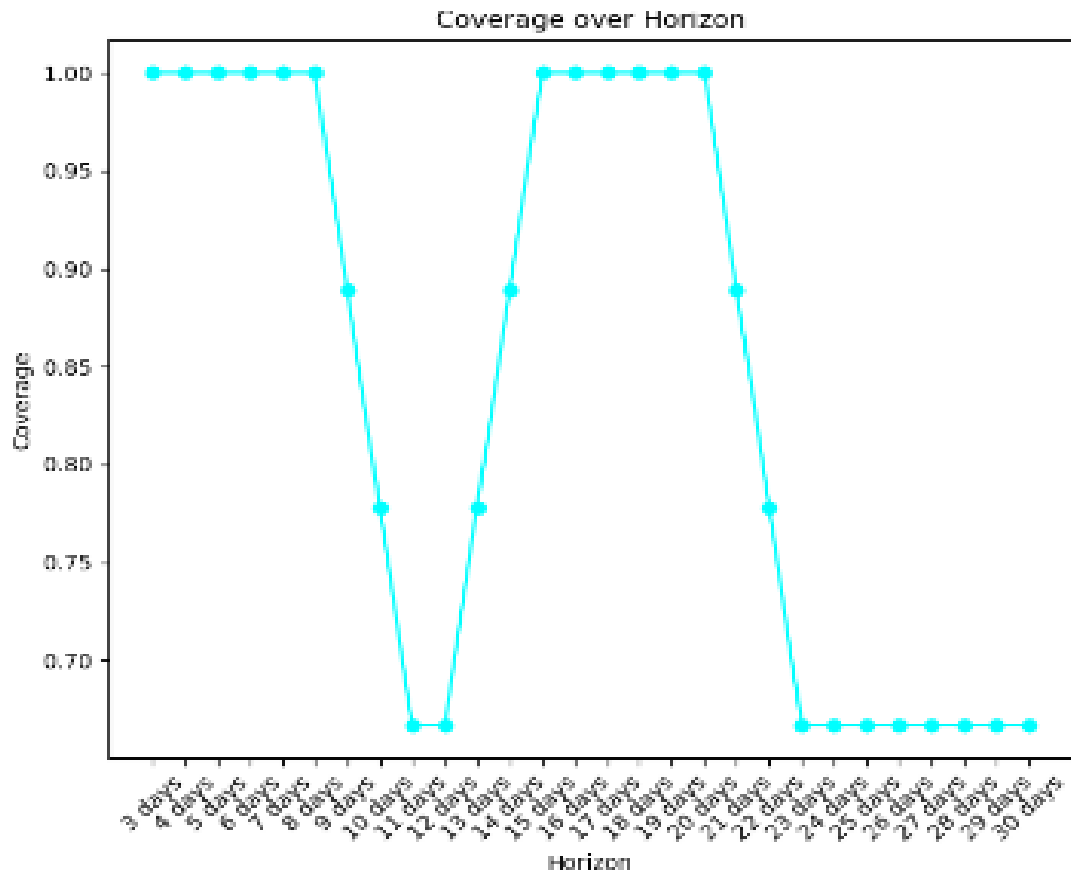


Figure 21. Coverage Metric on the best Horizon with Hyperparameter Tunning Parameters

The figure shows how the model's coverage behaves at different horizons. In the context of cross-validation, the "horizon" refers to the length of the further period of time when the model has predictions after the model is trained on the training dataset. The model performance is considered acceptable because of the high coverage rate on the test set for a specific period of time. Despite model coverage decreasing from 1.00 to 0.5, the model has a 0.845 coverage rate for a month when other days are included. Briefly, an 85% coverage rate is an indication of how well the model performs on the training data. Cross-validation on the training data tests how well the model generalizes the knowledge learned during the training process. 85% coverage of the training data indicates that the model can make reliable predictions on this dataset. This means that the model does not overfit during the training process and performs well on the data overall.

horizon	mse	rmse	mae	mape	mdape	smape	coverage
2 days	1.30212E+11	360849.1467	281208.6361	0.012551	0.008877	0.012432	0.727273
3 days	1.41042E+11	375555.7594	293318.38	0.013089	0.009181	0.01296	0.727273
4 days	1.5199E+11	389858.3269	305989.4366	0.013652	0.009658	0.013514	0.727273
5 days	1.62674E+11	403329.0393	318471.5226	0.014207	0.010433	0.01406	0.681818
6 days	1.73586E+11	416636.1493	330482.9035	0.014741	0.011064	0.014585	0.590909
7 days	1.84891E+11	429989.7059	342477.2343	0.015273	0.011389	0.015107	0.545455
8 days	1.97236E+11	444112.8158	354547.8982	0.015808	0.011725	0.015632	0.5
9 days	2.09845E+11	458088.659	365401.6965	0.016289	0.011998	0.016104	0.5

10 days	2.21454E+11	470589.2905	374433.6264	0.016689	0.012532	0.016495	0.545455
11 days	2.33852E+11	483582.0488	383728.8775	0.017101	0.013257	0.016897	0.545455
12 days	2.46987E+11	496977.4854	393192.8826	0.01752	0.013967	0.017306	0.545455
13 days	2.51695E+11	501691.866	397489.5367	0.01771	0.014502	0.017492	0.5
14 days	2.53393E+11	503381.5578	400414.361	0.017838	0.014828	0.017621	0.454545
15 days	2.5511E+11	505083.9482	403096.6566	0.017954	0.015309	0.017738	0.454545
16 days	2.87387E+11	536084.5288	423678.758	0.018862	0.015846	0.018619	0.454545
17 days	3.30603E+11	574981.2032	449801.1812	0.020015	0.016353	0.019734	0.409091
18 days	3.50778E+11	592265.0639	461150.0509	0.020514	0.017104	0.020219	0.363636
19 days	3.69681E+11	608014.2398	472048.4614	0.020992	0.017847	0.020681	0.409091
20 days	3.6006E+11	600050.0021	470544.9841	0.020923	0.018461	0.02062	0.409091
Best Period		Best Horizon	Best Coverage				
30		20	0.5311				

Figure 22. Cross Validation Table with Default Internal Parameters with Test Data

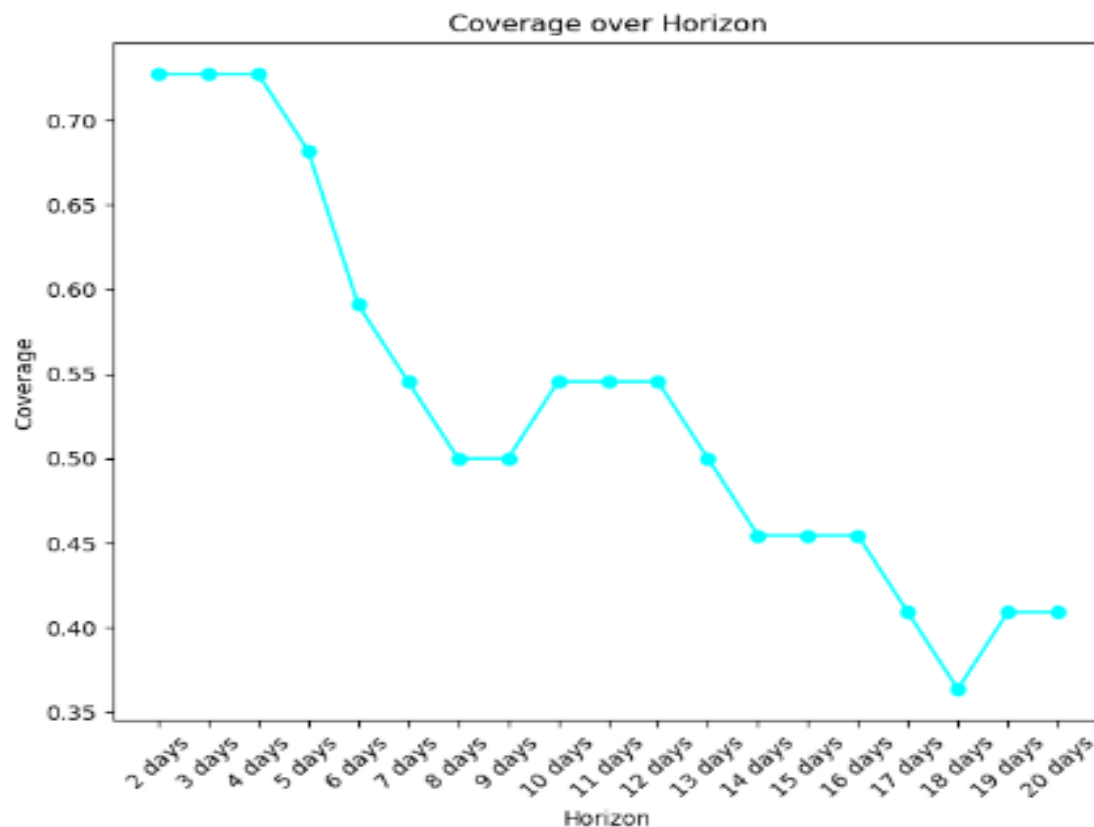


Figure 23. Coverage Metric on the best Horizon with Default Parameters with Test data

Cross Validation has also been applied to test datasets for model. how the model is going to perform on unseen data so that it will help if the model really understands the given input or not. When you look at the results, coverage “0.53” was obtained for the test dataset, which needs to be improved. That’s why, model tuning was applied to acquire better performance on the test dataset. The model training process is updated every 30 days and has a prediction for 20 days.

Changepoint_prior_scale	Holiday_prior_scale	Seasonality mode	Seasonality_prior_scale	The lowest MAPE
0.5	10.0	additive	30.0	0.005667

Figure 24. Hyperparameter tuning table on the best internal parameter values for Prophet on Test Data

The figure shows the best hyperparameters after completion of grid search (by trying every combination of given internal parameter elements). The lowest mape value has also been given.

horizon	mse	rmse	mae	mape	mdape	smape	coverage
2 days	14264930000	119435.88	96787.54348	0.004296	0.003527	0.00429	0.818182
3 days	15126340000	122989.1903	98859.43755	0.004386	0.003542	0.004379	0.818182
4 days	15661670000	125146.5977	99556.11912	0.004415	0.003485	0.004406	0.818182
5 days	16284530000	127610.8597	100024.8323	0.004434	0.003687	0.004424	0.818182
6 days	17238210000	131294.3777	101397.6517	0.004493	0.004002	0.004482	0.818182
7 days	18200780000	134910.2705	102428.8295	0.004537	0.003985	0.004524	0.818182
8 days	19207390000	138590.7184	103020.0172	0.004561	0.003393	0.004547	0.818182
9 days	20314880000	142530.2638	103851.3677	0.004598	0.003034	0.004583	0.727273
10 days	21205930000	145622.5597	105369.2085	0.004666	0.003234	0.004651	0.636364
11 days	22015480000	148376.1557	110816.4788	0.004909	0.003592	0.004893	0.681818
12 days	23219720000	152380.1674	119069.2495	0.005276	0.004047	0.005259	0.681818
13 days	24452890000	156374.196	124941.0638	0.005537	0.004461	0.00552	0.636364
14 days	25644460000	160138.8747	130203.9425	0.005772	0.004903	0.005754	0.636364
15 days	27635450000	166239.1322	139124.207	0.006168	0.005124	0.006149	0.590909
16 days	32922510000	181445.618	155584.6737	0.006897	0.005343	0.006875	0.545455
17 days	39030200000	197560.6207	171546.9905	0.007605	0.005815	0.007578	0.545455
18 days	43922420000	209576.7564	181676.6262	0.008054	0.005927	0.008024	0.5
19 days	48862180000	221047.9137	189988.3759	0.008423	0.006612	0.008389	0.454545
20 days	50666540000	225092.2823	195080.7015	0.00865	0.00725	0.008614	0.454545
Best Period	Best Horizon	Best Coverage					
30	20	0.67					

Figure 25. Cross Validation Table with Hyperparameter Tunning Parameters on Test Data

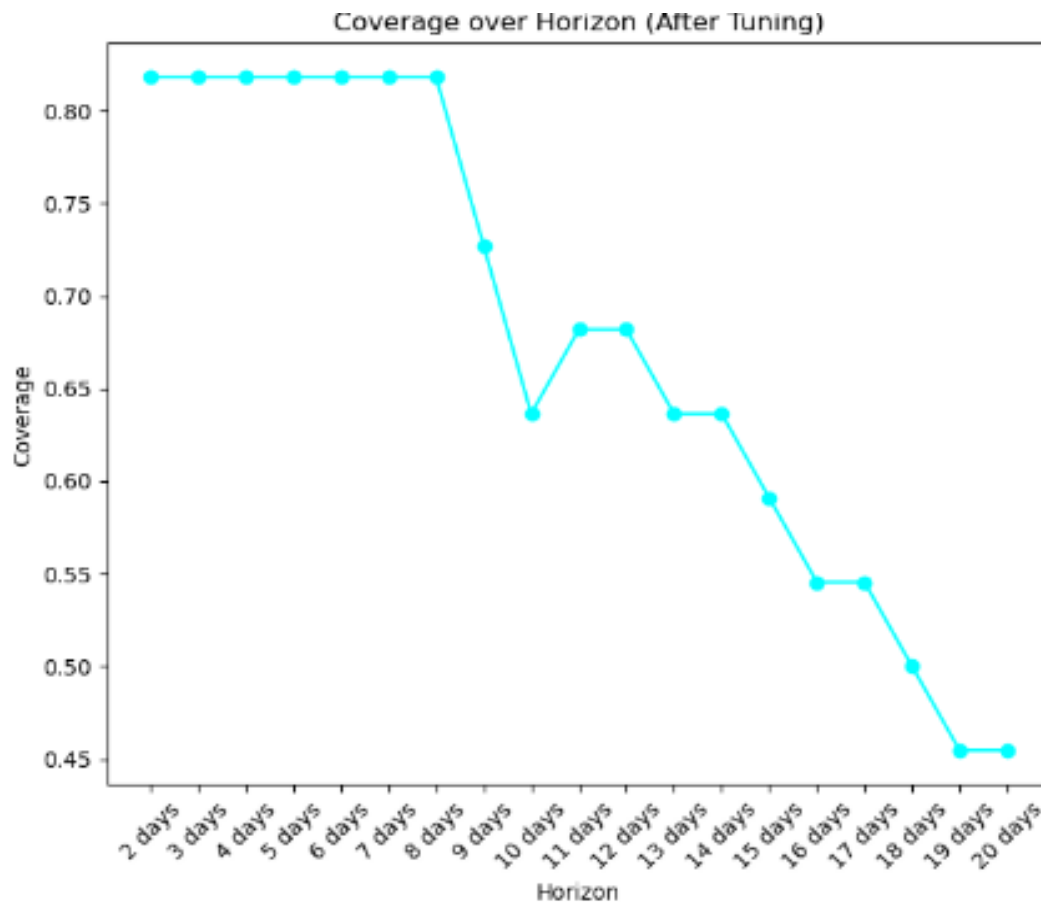


Figure 26. Coverage Metric on the best Horizon with Hyperparameter Tunning Parameters

The difference in performance on training and testing data is usually inevitable, as the model tends to produce better outcomes on training data. However, the coverage rate dropping from 85% to 67% close to almost %70 indicates that the generalization ability of the model is not perfect, but still provides acceptable performance due to the nature of pandemic.

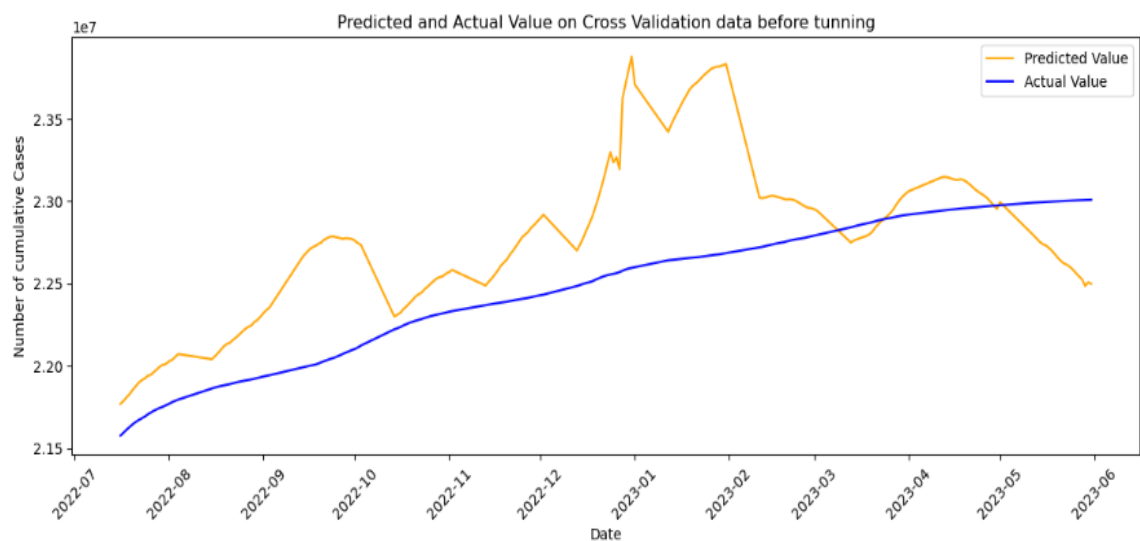


Figure 27. Comparison plot between predicted and actual values with default setting for Cros. Val. With Test data

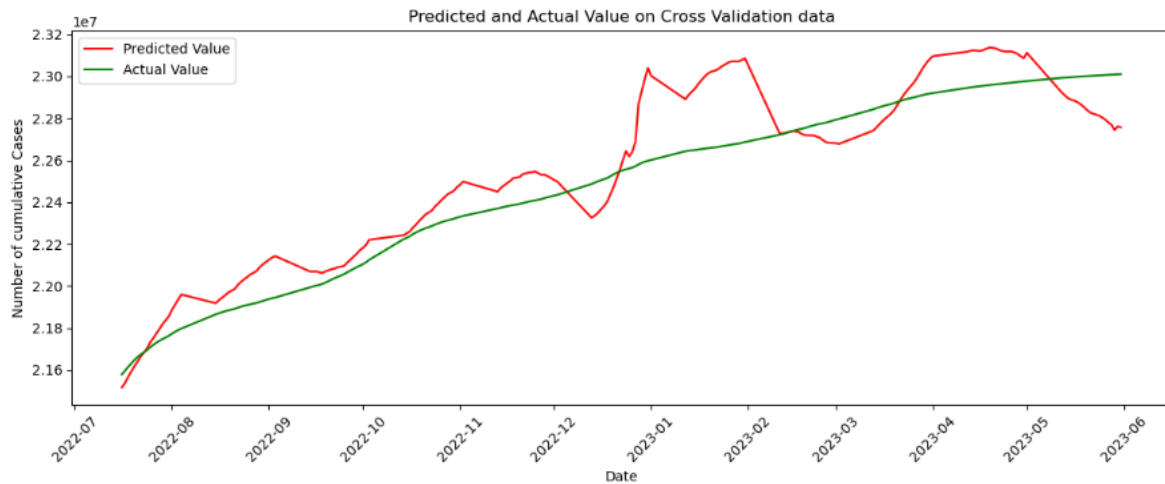


Figure 28. Comparison plot between predicted and actual values with tuning setting for Cros. Val. With Test data

If you look at “Figure 27. and Figure 28.”, you will see after the implementation of cross-validation how predicted data was fitted better than default settings, which shows that the model is found a way to fit itself over the given period for the test dataset. Despite there being some deviations from real-life data, having a performance that is close % to 70 coverage is completely fine on unseen data.

To summarize, while obtaining %70 coverage for training data, train data has %85 coverage for a 30-day prediction period, which means that the model is reliable in terms of medium- or short-term predictions. Having a high coverage rate on training data is an indicator of how the model learns well training data. On the other hand, the %67.5~%70 ratio for the test set shows model has a meaningful performance. Those acquired results prove the model has a good ability to manage uncertainty and provides reliable results in medium- or short-term forecasts. This circumstance supports the generalization ability of the model and its usability in practical

applications.

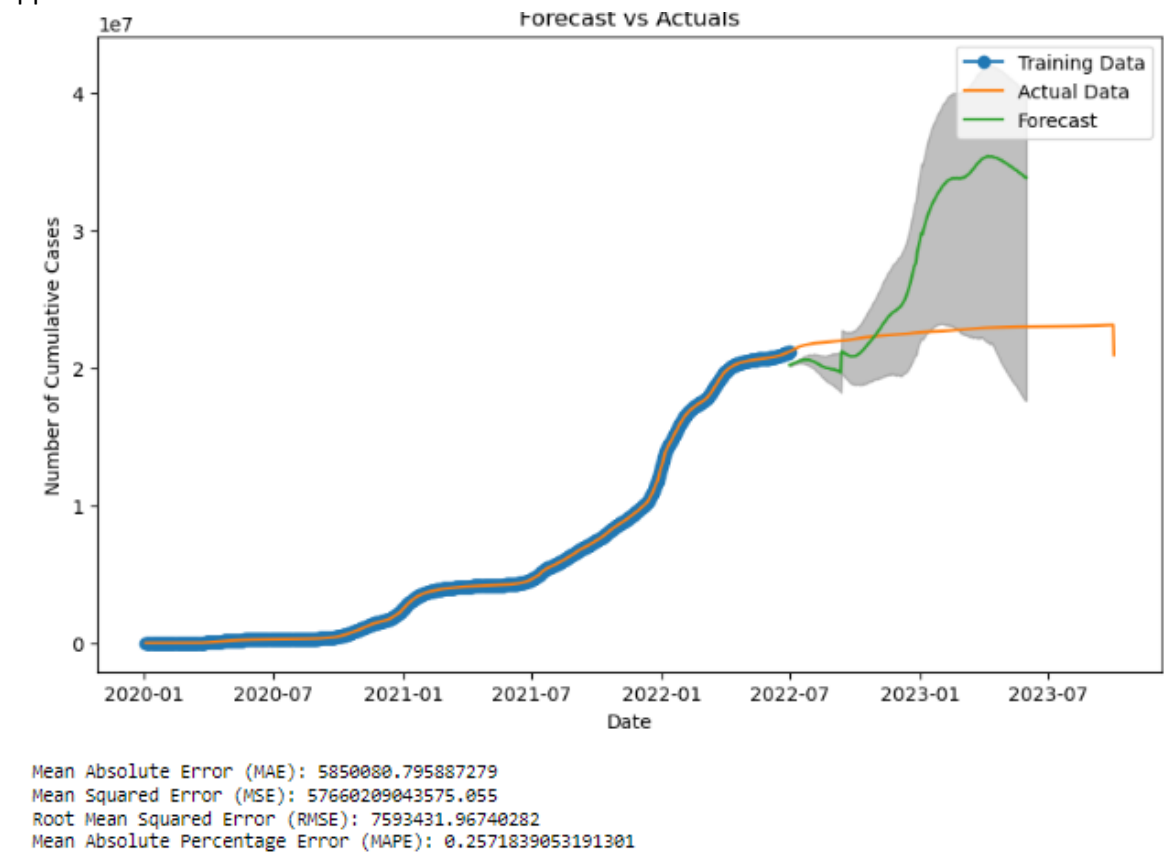


Figure 29. Implementation Prophet Model with train CV Parameters on Longer Time Period in UK, 20-23

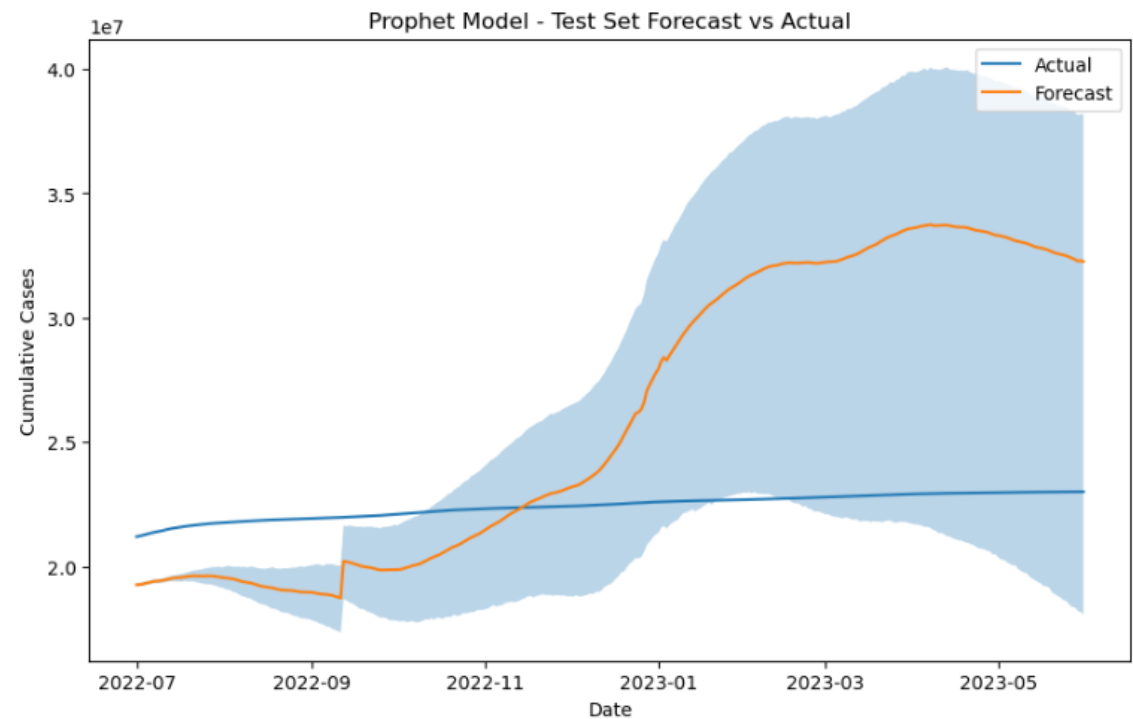


Figure 30. closer look to "Figure 28."

The figure shows the whole time on the used dataset. If you look at the previous figure, the model has % an 87.5 coverage rate for 30 days at different horizons, which is decent. When it comes to “Figure 29.”, mape has 0.25, which means the model's predictions deviate from the actual values by 25% on average. The graph consists of three main components that evaluate the model’s predictive capabilities and its agreement with real data. The blue line shows the training data on which the model was trained, while the orange line represents the real data observed during the period the model is predicting. The green line shows the model’s predictions for future data points, with the Gray area around these predictions indicating the uncertainty range of the predictions. The model has a quite accurate prediction in the short run until the end of 2022. After this point, the uncertainty of predictions starts to diverge from actual data. That’s why, having a lower mape is totally expected because the figure contained a longer period. To sum up, the model is quite reliable from “2022-07” to “2022-11” in the aspect of prediction outcomes.

8. Discussion

In this research, the number of COVID-19 cases was predicted by using Prophet. The logistic Growth Model is basically a mathematical equation or tool telling how number of cumulative cases is going to increase. Instead, the logistic growth model is integrated into the prophet model by providing a Cap value representing upper limit of the number of cases to prophet model. The integrated cap value with the aid of the logistic growth model will help to Prophet model in order to enhance its predictive accuracy. Those predictions were used to evaluate the output. Therefore, the logistic model has a crucial support role in improving model performance.

The built prophet model demonstrated acceptable performance in the short run, particularly capturing the seasonality and holiday effects. However, it did not show the same performance when the horizon was extended due to reduced accuracy on predictions, which is likely due to rising uncertainties such as the emergence of new variants. Briefly, the obtained outcome indicates that while the model is quite for short-term predictions, it does not have the same performance when the time horizon extends.

According to the results, the prophet model depicted quite effective performance by considering seasonality and holiday effects, which also shows the importance of those effects. Despite the model having accurate predictions in the short term, the model does not have the same performance owing to increasing uncertainties. It is seen logistic model helped the model with realistic limitations. In addition to that, It is seen that hyperparameter tuning is quite effective in terms of boosting model performance.

In hindsight, several aspects of the model could have been improved. For instance, it might be missing covid-19 variant that could be included model. More spike periods and their reason could somehow be included as features like as use of flag or feature derivation etc. In addition to that, train time period and holiday selection (adding or reducing the number of holidays on models) could be optimized in a comprehensive way. To illustrate, different holidays and the inclusion of additional features, such as social behaviour data during these periods, could have improved the model's accuracy. Furthermore, some sort of optimization could also be done on the logistic model to be able to reach better cap value so that it might optimize even more reliability of predictions.

Grid search method involving a higher frequency list for each internal parameter can even help more in terms of acquiring better internal parameters. All in all, these optimizations is not only increase the duration of the forecasting period as well as it will increase the accuracy of predictions, which will help to obtain predictions having longer periods with higher accuracy.

The results of this study are largely in line with previous research, especially those that have utilized the Prophet model for time series forecasting in epidemiology. Similar studies trying to make a prediction by using the Prophet obtained similar outcomes for long-term prediction. For example, "Khayyat (2021) had the research by using a prophet in order to predict future cases of COVID-19. He saw when the period increases, coverage also shows a reduced tendency. It might happen due to the dynamic nature of pandemics and the emergence of new variants, which results in some spikes in cases. Then, coverage or mape shows a reduced tendency when the predicted horizon extends.

When it comes to the limitation of the research, The model performance is created and evaluated for the United Kingdom. That's why, the model cannot show the same performance for other countries but as long as you have the same regressors and dataset with the same features, the model can be effective for other countries which is similar to the United Kingdom. The tricky point is to apply flags to critical dates for evaluated countries. Basically, this issue can bring difficulty in adapting datasets for used countries.

9. Conclusion

The research is looking for an answer to build a machine learning model predicting future cases for the United Kingdom. The logistic model and prophet are used for the purpose and are fed to the prophet with its cap value. Then, the model is customized around Cap, seasonality, and holiday effects. After the first evaluation with cross-validation operation, it is seen model performance is enough, which is why hyperparameter tuning operation is applied to the model to acquire better accuracy. Later on, obtained hyperparameters are implemented into the model in order to see the model performance on the longer timeline for the given dataset. Like it is mentioned in the discussion, the model has still some improvement spaces. For instance, finding the most optimized "M" and "N" values for the logistic model is likely to improve the model performance. Deeply examining swings on the dataset might also be another factor in the aspect of developing the model performance in the long term. More frequent grid searches on hyperparameters can also be another positive factor for the model's improvement. All in all, the model shows quite robust performance for 20-30 days at least on cross Validation operation. After those points are fixed, the best performance period detected by the cross-validation method can provide an opportunity to look for a longer period thanks to those algorithm developments. In addition to that, the model performance on test data for cross validation process is still needs an improvement in order to be more accurate on its predictions.

References

- [1] Hsieh, Y. H., Lee, J. Y., & Chang, H. L. (2004). SARS epidemiology modeling. *Emerging Infectious Diseases*, 10(6), 1165-1167. doi: 10.3201/eid1006.031023
- [2] Mohammed, O., El Hassani, I., Benabbou, F., & Ouhbi, B. (2020, October). Covid-19 forecasting in Morocco using FBprophet (Facebook's Framework in Python). *ResearchGate*. Retrieved from https://www.researchgate.net/profile/Quadoud-Mohammed-2/publication/344866592_Covid-19_forecasting_in_Morocco_using_FBprophet_Facebook's_Framework_in_Python/links/5f94bc88a6fdccfd7b7d5d43/Covid-19-forecasting-in-Morocco-using-FBprophet-Facebooks-Framework-in-Python.pdf
- [3] Frontiers in Applied Mathematics and Statistics. (2020). Retrieved from <https://www.frontiersin.org/journals/applied-mathematics-and-statistics/articles/10.3389/fams.2020.571544/full>
- [4] Wang, P., Zheng, X., Li, J., & Zhu, B. (2020). Prediction of epidemic trends in COVID-19 with logistic model and machine learning techniques. **Chaos, Solitons & Fractals**, 139, 110058. <https://doi.org/10.1016/j.chaos.2020.110058>
- [5] Facebook. (n.d.). Prophet Diagnostics. Retrieved from <https://facebook.github.io/prophet/docs/diagnostics.html>
- [6] Khayyat, M. (2021). Time Series Facebook Prophet Model and Python for COVID-19 Outbreak Prediction. *ResearchGate*. Retrieved from https://www.researchgate.net/profile/Masha-el-Khayyat-2/publication/349833934_Time_Series_Facebook_Prophet_Model_and_Python_for_COVID-19_Outbreak_Prediction/links/604319e292851c077f1c8b58/Time-Series-Facebook-Prophet-Model-and-Python-for-COVID-19-Outbreak-Prediction.pdf
- [7] Author(s). (2022). Title of the article. *Journal Name*, doi: 10.1155/2022/1556025. Retrieved from <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9070409/pdf/CMMM2022-1556025.pdf>
- [8] Institute for Government. (2022). Timeline of coronavirus lockdown in December 2021. Retrieved from <https://www.instituteforgovernment.org.uk/sites/default/files/2022-12/timeline-coronavirus-lockdown-december-2021.pdf>
- [9] Emerald Insight. (2020). Forecasting of COVID-19 epidemic size in four high-hitting nations (USA, Brazil, India, and Russia) by FB Prophet machine learning model. *Journal of Applied Clinical Informatics*, doi: 10.1108/ACI-09-2020-0059. Retrieved from <https://www.emerald.com/insight/content/doi/10.1108/ACI-09-2020-0059/full/pdf>
- [10] Vidal Rod, A. (2020). UK Daily COVID Data: Countries and Regions. *Kaggle*. Retrieved from <https://www.kaggle.com/datasets/albertovidalrod/uk-daily-covid-data-countries-and-regions>
- [11] World Health Organization (2024). COVID-19 Dashboard: Deaths. Retrieved from <https://data.who.int/dashboards/covid19/deaths?n=c>

[12]<https://www.kaggle.com/datasets/albertovidalrod/uk-daily-covid-data-countries-and-regions/data>

[13] NHS England. (2021, June). NHS COVID vaccine first dose drive enters the home straight as final cohort called forward. Retrieved from <https://www.england.nhs.uk/2021/06/nhs-covid-vaccine-first-dose-drive-enters-the-home-straight-as-final-cohort-called-forward/>

[14] NHS England. (2021, September). NHS begins COVID-19 booster vaccination campaign. Retrieved from <https://www.england.nhs.uk/2021/09/nhs-begins-covid-19-booster-vaccination-campaign/>

List of Figures

Figure 7. Covid 19 number of daily cases and tests in United Kingdom 2021-2023

Figure 8. Covid 19 number of daily cases and tests (blue and orange lines respectively) in United Kingdom 2021-2023

Figure 10. Holiday Effect on Daily Cases in United Kingdom 2021-2023

Figure 11. Prophet default Holiday List in United Kingdom

Figure 12. Block Diagram for Logistic Growth

Figure 13. Prophet Model Block Diagram with Logistic Growth Cap Feeding

Figure 14. Cross Validation Table with Default Internal Parameters with Train Data

Figure 15. Coverage Metric on the best Horizon with Default Parameters with Train data

Figure 16. Comparison plot between predicted and actual values with default setting for Cros. Val. With Train data

Figure 17. MAPE Metric on the best Horizon with Default Parameters on Train Data

Figure 18. Hyperparameter tuning table on the best internal parameter values for Prophet on Train Data

Figure 19. Cross Validation Table with Hyperparameter Tuning Parameters on Train Data

Figure 20. Comparison plot between predicted and actual values with Model Tuning settings for Cros. Val.

Figure 21. Coverage Metric on the best Horizon with Hyperparameter Tuning Parameters

Figure 25. Cross Validation Table with Hyperparameter Tuning Parameters on Test Data

Figure 26. Coverage Metric on the best Horizon with Hyperparameter Tuning Parameters

Figure 27. Comparison plot between predicted and actual values with default setting for Cros. Val. With Test data

Figure 28. Comparison plot between predicted and actual values with tuning setting for Cros. Val. With Test data

Figure 29. Implementation Prophet Model with train CV Parameters on Longer Time Period in UK 20-23

Figure 30. Closer look to “Figure 28.”

List of Tables

Table for **Figure 14.** Cross Validation Table with Default Internal Parameters with Train Data

Table for **Figure 19.** Cross Validation Table with Hyperparameter Tuning Parameters on Train Data

Table for **Figure 23.** Cross Validation Table with Default Internal Parameters with Test Data

Table for **Figure 25.** Hyperparameter tuning table on the best internal parameter values for Prophet on Test Data

Appendix

Figure 16. Output Table for Train(default)

ds	yhat	yhat_lower	yhat_upper	y	
0	01/02/2022	1.63E+07	1.62E+07	1.64E+07	16375978
1	02/02/2022	1.64E+07	1.64E+07	1.65E+07	16457859
2	03/02/2022	1.65E+07	1.65E+07	1.66E+07	16529071
3	04/02/2022	1.66E+07	1.65E+07	1.67E+07	16587599
4	05/02/2022	1.67E+07	1.66E+07	1.67E+07	16636120
5	06/02/2022	1.67E+07	1.67E+07	1.68E+07	16692332
6	07/02/2022	1.68E+07	1.67E+07	1.69E+07	16762787
7	08/02/2022	1.69E+07	1.68E+07	1.70E+07	16826796
8	09/02/2022	1.70E+07	1.69E+07	1.71E+07	16886027
9	10/02/2022	1.70E+07	1.70E+07	1.71E+07	16934127
10	11/02/2022	1.71E+07	1.70E+07	1.72E+07	16975178
11	12/02/2022	1.71E+07	1.70E+07	1.72E+07	17009732
12	13/02/2022	1.71E+07	1.70E+07	1.72E+07	17048122
13	14/02/2022	1.72E+07	1.71E+07	1.73E+07	17098418
14	15/02/2022	1.72E+07	1.71E+07	1.73E+07	17146779
15	16/02/2022	1.72E+07	1.71E+07	1.73E+07	17194169
16	17/02/2022	1.72E+07	1.71E+07	1.74E+07	17235585
17	18/02/2022	1.72E+07	1.71E+07	1.74E+07	17266452
18	19/02/2022	1.72E+07	1.71E+07	1.73E+07	17298363
19	20/02/2022	1.72E+07	1.71E+07	1.73E+07	17333088
20	21/02/2022	1.72E+07	1.71E+07	1.73E+07	17374268
21	22/02/2022	1.72E+07	1.71E+07	1.74E+07	17409954
22	23/02/2022	1.72E+07	1.71E+07	1.74E+07	17443202
23	24/02/2022	1.72E+07	1.70E+07	1.74E+07	17471238
24	25/02/2022	1.72E+07	1.70E+07	1.74E+07	17496749
25	26/02/2022	1.72E+07	1.70E+07	1.73E+07	17519736
26	27/02/2022	1.72E+07	1.69E+07	1.73E+07	17548417
27	28/02/2022	1.71E+07	1.69E+07	1.73E+07	17589547
28	01/03/2022	1.72E+07	1.70E+07	1.74E+07	17630861
29	02/03/2022	1.72E+07	1.70E+07	1.74E+07	17673589
30	02/04/2022	1.98E+07	1.97E+07	1.99E+07	19792942
31	03/04/2022	1.98E+07	1.98E+07	1.99E+07	19833868
32	04/04/2022	1.99E+07	1.98E+07	1.99E+07	19885801
33	05/04/2022	1.99E+07	1.99E+07	2.00E+07	19933208
34	06/04/2022	1.99E+07	1.99E+07	2.00E+07	19976520
35	07/04/2022	2.00E+07	1.99E+07	2.00E+07	20013939
36	08/04/2022	2.00E+07	1.99E+07	2.01E+07	20045935
37	09/04/2022	2.00E+07	1.99E+07	2.01E+07	20073036
38	10/04/2022	2.00E+07	1.98E+07	2.01E+07	20103066

39	11/04/2022	2.00E+07	1.98E+07	2.02E+07	20139209
40	12/04/2022	2.00E+07	1.98E+07	2.02E+07	20171895
41	13/04/2022	2.01E+07	1.98E+07	2.03E+07	20202542
42	14/04/2022	2.01E+07	1.98E+07	2.03E+07	20228296
43	15/04/2022	2.02E+07	1.98E+07	2.04E+07	20249235
44	16/04/2022	2.01E+07	1.96E+07	2.04E+07	20268550
45	17/04/2022	2.00E+07	1.96E+07	2.04E+07	20288114
46	18/04/2022	2.00E+07	1.95E+07	2.04E+07	20312562
47	19/04/2022	2.01E+07	1.95E+07	2.05E+07	20337401
48	20/04/2022	2.01E+07	1.95E+07	2.05E+07	20358013
49	21/04/2022	2.01E+07	1.94E+07	2.06E+07	20375719
50	22/04/2022	2.01E+07	1.94E+07	2.06E+07	20390793
51	23/04/2022	2.00E+07	1.93E+07	2.06E+07	20403506
52	24/04/2022	2.00E+07	1.93E+07	2.07E+07	20417674
53	25/04/2022	2.00E+07	1.92E+07	2.07E+07	20434181
54	26/04/2022	2.00E+07	1.91E+07	2.08E+07	20448220
55	27/04/2022	2.00E+07	1.90E+07	2.08E+07	20460800
56	28/04/2022	2.00E+07	1.90E+07	2.09E+07	20471973
57	29/04/2022	2.00E+07	1.89E+07	2.09E+07	20481612
58	30/04/2022	2.00E+07	1.89E+07	2.09E+07	20489690
59	01/05/2022	2.00E+07	1.88E+07	2.09E+07	20498405
60	01/06/2022	2.07E+07	2.07E+07	2.08E+07	20734075
61	02/06/2022	2.08E+07	2.07E+07	2.09E+07	20740257
62	03/06/2022	2.07E+07	2.07E+07	2.08E+07	20746841
63	04/06/2022	2.07E+07	2.07E+07	2.08E+07	20754298
64	05/06/2022	2.07E+07	2.07E+07	2.08E+07	20764076
65	06/06/2022	2.07E+07	2.07E+07	2.08E+07	20775749
66	07/06/2022	2.07E+07	2.07E+07	2.08E+07	20787090
67	08/06/2022	2.08E+07	2.07E+07	2.08E+07	20798808
68	09/06/2022	2.08E+07	2.07E+07	2.08E+07	20810078
69	10/06/2022	2.08E+07	2.07E+07	2.09E+07	20820801
70	11/06/2022	2.08E+07	2.06E+07	2.09E+07	20830822
71	12/06/2022	2.07E+07	2.06E+07	2.09E+07	20843459
72	13/06/2022	2.08E+07	2.06E+07	2.09E+07	20858626
73	14/06/2022	2.08E+07	2.06E+07	2.10E+07	20873675
74	15/06/2022	2.08E+07	2.06E+07	2.10E+07	20888419
75	16/06/2022	2.08E+07	2.06E+07	2.10E+07	20902551
76	17/06/2022	2.08E+07	2.05E+07	2.11E+07	20916353
77	18/06/2022	2.08E+07	2.05E+07	2.11E+07	20929892
78	19/06/2022	2.08E+07	2.05E+07	2.11E+07	20946810
79	20/06/2022	2.08E+07	2.05E+07	2.12E+07	20966579
80	21/06/2022	2.08E+07	2.05E+07	2.12E+07	20987298
81	22/06/2022	2.09E+07	2.05E+07	2.12E+07	21008007
82	23/06/2022	2.09E+07	2.05E+07	2.13E+07	21027430
83	24/06/2022	2.09E+07	2.05E+07	2.14E+07	21045602
84	25/06/2022	2.09E+07	2.05E+07	2.14E+07	21062294

85	26/06/2022	2.09E+07	2.04E+07	2.14E+07	21082937
86	27/06/2022	2.09E+07	2.04E+07	2.15E+07	21107814
87	28/06/2022	2.10E+07	2.05E+07	2.16E+07	21132648
88	29/06/2022	2.10E+07	2.05E+07	2.16E+07	21157032
89	30/06/2022	2.10E+07	2.05E+07	2.17E+07	21181062

Figure 20. Output Table for Train (Model Tunning)

ds	yhat	yhat_lower	yhat_upper	y	
0	01/02/2022	1.63E+07	1.62E+07	1.64E+07	16375978
1	02/02/2022	1.64E+07	1.64E+07	1.65E+07	16457859
2	03/02/2022	1.65E+07	1.65E+07	1.66E+07	16529071
3	04/02/2022	1.66E+07	1.65E+07	1.67E+07	16587599
4	05/02/2022	1.67E+07	1.66E+07	1.67E+07	16636120
5	06/02/2022	1.67E+07	1.67E+07	1.68E+07	16692332
6	07/02/2022	1.68E+07	1.67E+07	1.69E+07	16762787
7	08/02/2022	1.69E+07	1.68E+07	1.70E+07	16826796
8	09/02/2022	1.70E+07	1.69E+07	1.71E+07	16886027
9	10/02/2022	1.70E+07	1.70E+07	1.71E+07	16934127
10	11/02/2022	1.71E+07	1.70E+07	1.72E+07	16975178
11	12/02/2022	1.71E+07	1.70E+07	1.72E+07	17009732
12	13/02/2022	1.71E+07	1.70E+07	1.72E+07	17048122
13	14/02/2022	1.72E+07	1.71E+07	1.73E+07	17098418
14	15/02/2022	1.72E+07	1.71E+07	1.73E+07	17146779
15	16/02/2022	1.72E+07	1.71E+07	1.73E+07	17194169
16	17/02/2022	1.72E+07	1.71E+07	1.74E+07	17235585
17	18/02/2022	1.72E+07	1.71E+07	1.74E+07	17266452
18	19/02/2022	1.72E+07	1.71E+07	1.73E+07	17298363
19	20/02/2022	1.72E+07	1.71E+07	1.73E+07	17333088
20	21/02/2022	1.72E+07	1.71E+07	1.73E+07	17374268
21	22/02/2022	1.72E+07	1.71E+07	1.74E+07	17409954
22	23/02/2022	1.72E+07	1.71E+07	1.74E+07	17443202
23	24/02/2022	1.72E+07	1.70E+07	1.74E+07	17471238
24	25/02/2022	1.72E+07	1.70E+07	1.74E+07	17496749
25	26/02/2022	1.72E+07	1.70E+07	1.73E+07	17519736
26	27/02/2022	1.72E+07	1.69E+07	1.73E+07	17548417
27	28/02/2022	1.71E+07	1.69E+07	1.73E+07	17589547
28	01/03/2022	1.72E+07	1.70E+07	1.74E+07	17630861
29	02/03/2022	1.72E+07	1.70E+07	1.74E+07	17673589
30	02/04/2022	1.98E+07	1.97E+07	1.99E+07	19792942
31	03/04/2022	1.98E+07	1.98E+07	1.99E+07	19833868
32	04/04/2022	1.99E+07	1.98E+07	1.99E+07	19885801
33	05/04/2022	1.99E+07	1.99E+07	2.00E+07	19933208
34	06/04/2022	1.99E+07	1.99E+07	2.00E+07	19976520
35	07/04/2022	2.00E+07	1.99E+07	2.00E+07	20013939

36	08/04/2022	2.00E+07	1.99E+07	2.01E+07	20045935
37	09/04/2022	2.00E+07	1.99E+07	2.01E+07	20073036
38	10/04/2022	2.00E+07	1.98E+07	2.01E+07	20103066
39	11/04/2022	2.00E+07	1.98E+07	2.02E+07	20139209
40	12/04/2022	2.00E+07	1.98E+07	2.02E+07	20171895
41	13/04/2022	2.01E+07	1.98E+07	2.03E+07	20202542
42	14/04/2022	2.01E+07	1.98E+07	2.03E+07	20228296
43	15/04/2022	2.02E+07	1.98E+07	2.04E+07	20249235
44	16/04/2022	2.01E+07	1.96E+07	2.04E+07	20268550
45	17/04/2022	2.00E+07	1.96E+07	2.04E+07	20288114
46	18/04/2022	2.00E+07	1.95E+07	2.04E+07	20312562
47	19/04/2022	2.01E+07	1.95E+07	2.05E+07	20337401
48	20/04/2022	2.01E+07	1.95E+07	2.05E+07	20358013
49	21/04/2022	2.01E+07	1.94E+07	2.06E+07	20375719
50	22/04/2022	2.01E+07	1.94E+07	2.06E+07	20390793
51	23/04/2022	2.00E+07	1.93E+07	2.06E+07	20403506
52	24/04/2022	2.00E+07	1.93E+07	2.07E+07	20417674
53	25/04/2022	2.00E+07	1.92E+07	2.07E+07	20434181
54	26/04/2022	2.00E+07	1.91E+07	2.08E+07	20448220
55	27/04/2022	2.00E+07	1.90E+07	2.08E+07	20460800
56	28/04/2022	2.00E+07	1.90E+07	2.09E+07	20471973
57	29/04/2022	2.00E+07	1.89E+07	2.09E+07	20481612
58	30/04/2022	2.00E+07	1.89E+07	2.09E+07	20489690
59	01/05/2022	2.00E+07	1.88E+07	2.09E+07	20498405
60	01/06/2022	2.07E+07	2.07E+07	2.08E+07	20734075
61	02/06/2022	2.08E+07	2.07E+07	2.09E+07	20740257
62	03/06/2022	2.07E+07	2.07E+07	2.08E+07	20746841
63	04/06/2022	2.07E+07	2.07E+07	2.08E+07	20754298
64	05/06/2022	2.07E+07	2.07E+07	2.08E+07	20764076
65	06/06/2022	2.07E+07	2.07E+07	2.08E+07	20775749
66	07/06/2022	2.07E+07	2.07E+07	2.08E+07	20787090
67	08/06/2022	2.08E+07	2.07E+07	2.08E+07	20798808
68	09/06/2022	2.08E+07	2.07E+07	2.08E+07	20810078
69	10/06/2022	2.08E+07	2.07E+07	2.09E+07	20820801
70	11/06/2022	2.08E+07	2.06E+07	2.09E+07	20830822
71	12/06/2022	2.07E+07	2.06E+07	2.09E+07	20843459
72	13/06/2022	2.08E+07	2.06E+07	2.09E+07	20858626
73	14/06/2022	2.08E+07	2.06E+07	2.10E+07	20873675
74	15/06/2022	2.08E+07	2.06E+07	2.10E+07	20888419
75	16/06/2022	2.08E+07	2.06E+07	2.10E+07	20902551
76	17/06/2022	2.08E+07	2.05E+07	2.11E+07	20916353
77	18/06/2022	2.08E+07	2.05E+07	2.11E+07	20929892
78	19/06/2022	2.08E+07	2.05E+07	2.11E+07	20946810
79	20/06/2022	2.08E+07	2.05E+07	2.12E+07	20966579
80	21/06/2022	2.08E+07	2.05E+07	2.12E+07	20987298
81	22/06/2022	2.09E+07	2.05E+07	2.12E+07	21008007

82	23/06/2022	2.09E+07	2.05E+07	2.13E+07	21027430
83	24/06/2022	2.09E+07	2.05E+07	2.14E+07	21045602
84	25/06/2022	2.09E+07	2.05E+07	2.14E+07	21062294
85	26/06/2022	2.09E+07	2.04E+07	2.14E+07	21082937
86	27/06/2022	2.09E+07	2.04E+07	2.15E+07	21107814
87	28/06/2022	2.10E+07	2.05E+07	2.16E+07	21132648
88	29/06/2022	2.10E+07	2.05E+07	2.16E+07	21157032
89	30/06/2022	2.10E+07	2.05E+07	2.17E+07	21181062

Output Table for Test(Default)

ds	yhat	yhat_lower	yhat_upper	y	
0	16/07/2022	2.18E+07	2.14E+07	2.21E+07	21578797
1	17/07/2022	2.18E+07	2.14E+07	2.21E+07	21595940
2	18/07/2022	2.18E+07	2.15E+07	2.21E+07	21614510
3	19/07/2022	2.18E+07	2.15E+07	2.22E+07	21631048
4	20/07/2022	2.19E+07	2.15E+07	2.22E+07	21647080
5	21/07/2022	2.19E+07	2.16E+07	2.22E+07	21661129
6	22/07/2022	2.19E+07	2.16E+07	2.22E+07	21673400
7	23/07/2022	2.19E+07	2.16E+07	2.23E+07	21683265
8	24/07/2022	2.19E+07	2.16E+07	2.22E+07	21694468
9	25/07/2022	2.19E+07	2.16E+07	2.23E+07	21707640
10	26/07/2022	2.19E+07	2.16E+07	2.22E+07	21719275
11	27/07/2022	2.20E+07	2.16E+07	2.23E+07	21729746
12	28/07/2022	2.20E+07	2.16E+07	2.23E+07	21738996
13	29/07/2022	2.20E+07	2.17E+07	2.23E+07	21747422
14	30/07/2022	2.20E+07	2.17E+07	2.24E+07	21754679
15	31/07/2022	2.20E+07	2.17E+07	2.23E+07	21763059
16	01/08/2022	2.20E+07	2.17E+07	2.24E+07	21772788
17	02/08/2022	2.20E+07	2.17E+07	2.24E+07	21781865
18	03/08/2022	2.21E+07	2.17E+07	2.24E+07	21790157
19	04/08/2022	2.21E+07	2.17E+07	2.24E+07	21797381
20	15/08/2022	2.20E+07	2.17E+07	2.24E+07	21865083
21	16/08/2022	2.21E+07	2.17E+07	2.24E+07	21870549
22	17/08/2022	2.21E+07	2.18E+07	2.24E+07	21875539
23	18/08/2022	2.21E+07	2.17E+07	2.24E+07	21880060
24	19/08/2022	2.21E+07	2.18E+07	2.24E+07	21884016
25	20/08/2022	2.21E+07	2.18E+07	2.25E+07	21887489
26	21/08/2022	2.21E+07	2.18E+07	2.24E+07	21891529
27	22/08/2022	2.22E+07	2.19E+07	2.25E+07	21896695
28	23/08/2022	2.22E+07	2.19E+07	2.25E+07	21901421
29	24/08/2022	2.22E+07	2.19E+07	2.25E+07	21905925
30	25/08/2022	2.22E+07	2.19E+07	2.25E+07	21909679
31	26/08/2022	2.22E+07	2.19E+07	2.26E+07	21913099
32	27/08/2022	2.22E+07	2.19E+07	2.26E+07	21916177

33	28/08/2022	2.22E+07	2.19E+07	2.26E+07	21919586
34	29/08/2022	2.23E+07	2.19E+07	2.26E+07	21924217
35	30/08/2022	2.23E+07	2.20E+07	2.26E+07	21929383
36	31/08/2022	2.23E+07	2.19E+07	2.26E+07	21933935
37	01/09/2022	2.23E+07	2.20E+07	2.27E+07	21938232
38	02/09/2022	2.23E+07	2.20E+07	2.27E+07	21941995
39	03/09/2022	2.24E+07	2.20E+07	2.27E+07	21945364
40	14/09/2022	2.27E+07	2.23E+07	2.30E+07	21993451
41	15/09/2022	2.27E+07	2.24E+07	2.30E+07	21998043
42	16/09/2022	2.27E+07	2.24E+07	2.31E+07	22002138
43	17/09/2022	2.27E+07	2.24E+07	2.31E+07	22005631
44	18/09/2022	2.27E+07	2.24E+07	2.31E+07	22009932
45	19/09/2022	2.27E+07	2.24E+07	2.31E+07	22016044
46	20/09/2022	2.28E+07	2.24E+07	2.31E+07	22023780
47	21/09/2022	2.28E+07	2.24E+07	2.31E+07	22031251
48	22/09/2022	2.28E+07	2.25E+07	2.31E+07	22037990
49	23/09/2022	2.28E+07	2.25E+07	2.31E+07	22044397
50	24/09/2022	2.28E+07	2.25E+07	2.31E+07	22050045
51	25/09/2022	2.28E+07	2.24E+07	2.31E+07	22057218
52	26/09/2022	2.28E+07	2.25E+07	2.31E+07	22065745
53	27/09/2022	2.28E+07	2.25E+07	2.31E+07	22073886
54	28/09/2022	2.28E+07	2.25E+07	2.31E+07	22082183
55	29/09/2022	2.28E+07	2.24E+07	2.31E+07	22089983
56	30/09/2022	2.28E+07	2.24E+07	2.31E+07	22097487
57	01/10/2022	2.28E+07	2.24E+07	2.31E+07	22104476
58	02/10/2022	2.27E+07	2.24E+07	2.31E+07	22113584
59	03/10/2022	2.27E+07	2.24E+07	2.31E+07	22125130
60	14/10/2022	2.23E+07	2.21E+07	2.25E+07	22223867
61	15/10/2022	2.23E+07	2.21E+07	2.25E+07	22230211
62	16/10/2022	2.23E+07	2.21E+07	2.25E+07	22237917
63	17/10/2022	2.23E+07	2.21E+07	2.26E+07	22247797
64	18/10/2022	2.24E+07	2.21E+07	2.26E+07	22255938
65	19/10/2022	2.24E+07	2.22E+07	2.26E+07	22263426
66	20/10/2022	2.24E+07	2.22E+07	2.26E+07	22269906
67	21/10/2022	2.24E+07	2.22E+07	2.26E+07	22276118
68	22/10/2022	2.24E+07	2.22E+07	2.26E+07	22281026
69	23/10/2022	2.24E+07	2.22E+07	2.27E+07	22286495
70	24/10/2022	2.25E+07	2.22E+07	2.27E+07	22293202
71	25/10/2022	2.25E+07	2.23E+07	2.27E+07	22298791
72	26/10/2022	2.25E+07	2.23E+07	2.27E+07	22304157
73	27/10/2022	2.25E+07	2.23E+07	2.27E+07	22308932
74	28/10/2022	2.25E+07	2.23E+07	2.27E+07	22313254
75	29/10/2022	2.25E+07	2.23E+07	2.28E+07	22316788
76	30/10/2022	2.25E+07	2.23E+07	2.28E+07	22320994
77	31/10/2022	2.26E+07	2.23E+07	2.28E+07	22326061
78	01/11/2022	2.26E+07	2.23E+07	2.28E+07	22330558

79	02/11/2022	2.26E+07	2.23E+07	2.28E+07	22334513
80	13/11/2022	2.25E+07	2.23E+07	2.27E+07	22369955
81	14/11/2022	2.25E+07	2.23E+07	2.27E+07	22373821
82	15/11/2022	2.25E+07	2.23E+07	2.28E+07	22377309
83	16/11/2022	2.26E+07	2.23E+07	2.28E+07	22380533
84	17/11/2022	2.26E+07	2.24E+07	2.28E+07	22383518
85	18/11/2022	2.26E+07	2.24E+07	2.28E+07	22386251
86	19/11/2022	2.26E+07	2.24E+07	2.29E+07	22388655
87	20/11/2022	2.26E+07	2.24E+07	2.28E+07	22391511
88	21/11/2022	2.27E+07	2.25E+07	2.29E+07	22395397
89	22/11/2022	2.27E+07	2.25E+07	2.29E+07	22399146
90	23/11/2022	2.27E+07	2.25E+07	2.29E+07	22402715
91	24/11/2022	2.28E+07	2.25E+07	2.30E+07	22405999
92	25/11/2022	2.28E+07	2.26E+07	2.30E+07	22408867
93	26/11/2022	2.28E+07	2.26E+07	2.30E+07	22411370
94	27/11/2022	2.28E+07	2.26E+07	2.30E+07	22414790
95	28/11/2022	2.28E+07	2.26E+07	2.31E+07	22419358
96	29/11/2022	2.29E+07	2.26E+07	2.31E+07	22423655
97	30/11/2022	2.29E+07	2.27E+07	2.31E+07	22427546
98	01/12/2022	2.29E+07	2.27E+07	2.31E+07	22430989
99	02/12/2022	2.29E+07	2.27E+07	2.31E+07	22434533
100	13/12/2022	2.27E+07	2.24E+07	2.30E+07	22486694
101	14/12/2022	2.27E+07	2.25E+07	2.30E+07	22493064
102	15/12/2022	2.28E+07	2.25E+07	2.30E+07	22499017
103	16/12/2022	2.28E+07	2.26E+07	2.31E+07	22504916
104	17/12/2022	2.29E+07	2.26E+07	2.31E+07	22509541
105	18/12/2022	2.29E+07	2.26E+07	2.32E+07	22515183
106	19/12/2022	2.30E+07	2.27E+07	2.32E+07	22524022
107	20/12/2022	2.30E+07	2.28E+07	2.33E+07	22532446
108	21/12/2022	2.31E+07	2.28E+07	2.34E+07	22540058
109	22/12/2022	2.32E+07	2.29E+07	2.34E+07	22546756
110	23/12/2022	2.32E+07	2.30E+07	2.35E+07	22552385
111	24/12/2022	2.33E+07	2.30E+07	2.36E+07	22556448
112	25/12/2022	2.32E+07	2.30E+07	2.35E+07	22560110
113	26/12/2022	2.33E+07	2.30E+07	2.35E+07	22564959
114	27/12/2022	2.32E+07	2.29E+07	2.35E+07	22571411
115	28/12/2022	2.36E+07	2.34E+07	2.39E+07	22579654
116	29/12/2022	2.37E+07	2.34E+07	2.40E+07	22587176
117	30/12/2022	2.38E+07	2.35E+07	2.41E+07	22593153
118	31/12/2022	2.39E+07	2.36E+07	2.41E+07	22597351
119	01/01/2023	2.37E+07	2.35E+07	2.40E+07	22600954
120	12/01/2023	2.34E+07	2.31E+07	2.37E+07	22643010
121	13/01/2023	2.35E+07	2.32E+07	2.38E+07	22645104
122	14/01/2023	2.35E+07	2.32E+07	2.38E+07	22646571
123	15/01/2023	2.35E+07	2.33E+07	2.39E+07	22648400
124	16/01/2023	2.36E+07	2.33E+07	2.39E+07	22650917

125	17/01/2023	2.36E+07	2.33E+07	2.39E+07	22653083
126	18/01/2023	2.36E+07	2.33E+07	2.39E+07	22655156
127	19/01/2023	2.37E+07	2.34E+07	2.40E+07	22657125
128	20/01/2023	2.37E+07	2.34E+07	2.40E+07	22658955
129	21/01/2023	2.37E+07	2.34E+07	2.40E+07	22660462
130	22/01/2023	2.37E+07	2.34E+07	2.40E+07	22662344
131	23/01/2023	2.38E+07	2.34E+07	2.40E+07	22664953
132	24/01/2023	2.38E+07	2.35E+07	2.41E+07	22667686
133	25/01/2023	2.38E+07	2.35E+07	2.41E+07	22670285
134	26/01/2023	2.38E+07	2.35E+07	2.41E+07	22672782
135	27/01/2023	2.38E+07	2.35E+07	2.41E+07	22675197
136	28/01/2023	2.38E+07	2.35E+07	2.41E+07	22677139
137	29/01/2023	2.38E+07	2.35E+07	2.41E+07	22679568
138	30/01/2023	2.38E+07	2.35E+07	2.41E+07	22683058
139	31/01/2023	2.38E+07	2.35E+07	2.42E+07	22686548
140	11/02/2023	2.30E+07	2.27E+07	2.33E+07	22721450
141	12/02/2023	2.30E+07	2.28E+07	2.33E+07	22724592
142	13/02/2023	2.30E+07	2.27E+07	2.33E+07	22729428
143	14/02/2023	2.30E+07	2.27E+07	2.33E+07	22734049
144	15/02/2023	2.30E+07	2.27E+07	2.33E+07	22738208
145	16/02/2023	2.30E+07	2.27E+07	2.33E+07	22742431
146	17/02/2023	2.30E+07	2.27E+07	2.33E+07	22746427
147	18/02/2023	2.30E+07	2.28E+07	2.33E+07	22749523
148	19/02/2023	2.30E+07	2.27E+07	2.33E+07	22753174
149	20/02/2023	2.30E+07	2.27E+07	2.33E+07	22758012
150	21/02/2023	2.30E+07	2.27E+07	2.33E+07	22762646
151	22/02/2023	2.30E+07	2.27E+07	2.33E+07	22766794
152	23/02/2023	2.30E+07	2.27E+07	2.33E+07	22770497
153	24/02/2023	2.30E+07	2.27E+07	2.33E+07	22773836
154	25/02/2023	2.30E+07	2.27E+07	2.33E+07	22776533
155	26/02/2023	2.30E+07	2.27E+07	2.33E+07	22779659
156	27/02/2023	2.30E+07	2.27E+07	2.33E+07	22784305
157	28/02/2023	2.30E+07	2.27E+07	2.32E+07	22788789
158	01/03/2023	2.30E+07	2.27E+07	2.33E+07	22793078
159	02/03/2023	2.29E+07	2.26E+07	2.32E+07	22796935
160	13/03/2023	2.28E+07	2.25E+07	2.30E+07	22842838
161	14/03/2023	2.28E+07	2.25E+07	2.30E+07	22848334
162	15/03/2023	2.28E+07	2.25E+07	2.31E+07	22853108
163	16/03/2023	2.28E+07	2.25E+07	2.30E+07	22857906
164	17/03/2023	2.28E+07	2.25E+07	2.30E+07	22862005
165	18/03/2023	2.28E+07	2.25E+07	2.30E+07	22865406
166	19/03/2023	2.28E+07	2.25E+07	2.31E+07	22869473
167	20/03/2023	2.28E+07	2.25E+07	2.31E+07	22874677
168	21/03/2023	2.28E+07	2.26E+07	2.31E+07	22879879
169	22/03/2023	2.29E+07	2.26E+07	2.31E+07	22884670
170	23/03/2023	2.29E+07	2.26E+07	2.31E+07	22889039

171	24/03/2023	2.29E+07	2.26E+07	2.32E+07	22893152
172	25/03/2023	2.29E+07	2.26E+07	2.32E+07	22896162
173	26/03/2023	2.29E+07	2.27E+07	2.32E+07	22899634
174	27/03/2023	2.30E+07	2.27E+07	2.32E+07	22903965
175	28/03/2023	2.30E+07	2.27E+07	2.33E+07	22907892
176	29/03/2023	2.30E+07	2.27E+07	2.32E+07	22911674
177	30/03/2023	2.30E+07	2.28E+07	2.33E+07	22915061
178	31/03/2023	2.30E+07	2.28E+07	2.33E+07	22917843
179	01/04/2023	2.31E+07	2.28E+07	2.33E+07	22920158
180	12/04/2023	2.31E+07	2.28E+07	2.35E+07	22945177
181	13/04/2023	2.31E+07	2.28E+07	2.35E+07	22947273
182	14/04/2023	2.31E+07	2.28E+07	2.35E+07	22949002
183	15/04/2023	2.31E+07	2.28E+07	2.35E+07	22950411
184	16/04/2023	2.31E+07	2.28E+07	2.35E+07	22952369
185	17/04/2023	2.31E+07	2.28E+07	2.35E+07	22954663
186	18/04/2023	2.31E+07	2.28E+07	2.35E+07	22956629
187	19/04/2023	2.31E+07	2.28E+07	2.34E+07	22958484
188	20/04/2023	2.31E+07	2.28E+07	2.34E+07	22960112
189	21/04/2023	2.31E+07	2.28E+07	2.34E+07	22961579
190	22/04/2023	2.31E+07	2.28E+07	2.34E+07	22962865
191	23/04/2023	2.31E+07	2.27E+07	2.34E+07	22964416
192	24/04/2023	2.31E+07	2.27E+07	2.34E+07	22966275
193	25/04/2023	2.30E+07	2.27E+07	2.33E+07	22968035
194	26/04/2023	2.30E+07	2.27E+07	2.34E+07	22969730
195	27/04/2023	2.30E+07	2.27E+07	2.33E+07	22971355
196	28/04/2023	2.30E+07	2.27E+07	2.33E+07	22972819
197	29/04/2023	2.30E+07	2.26E+07	2.33E+07	22974138
198	30/04/2023	2.30E+07	2.27E+07	2.33E+07	22975507
199	01/05/2023	2.30E+07	2.27E+07	2.33E+07	22977274
200	12/05/2023	2.28E+07	2.25E+07	2.31E+07	22992568
201	13/05/2023	2.28E+07	2.25E+07	2.31E+07	22993419
202	14/05/2023	2.28E+07	2.25E+07	2.30E+07	22994522
203	15/05/2023	2.27E+07	2.25E+07	2.30E+07	22995753
204	16/05/2023	2.27E+07	2.25E+07	2.30E+07	22996899
205	17/05/2023	2.27E+07	2.24E+07	2.30E+07	22997989
206	18/05/2023	2.27E+07	2.24E+07	2.30E+07	22999031
207	19/05/2023	2.27E+07	2.24E+07	2.30E+07	22999958
208	20/05/2023	2.27E+07	2.24E+07	2.29E+07	23000729
209	21/05/2023	2.26E+07	2.24E+07	2.29E+07	23001676
210	22/05/2023	2.26E+07	2.23E+07	2.29E+07	23002845
211	23/05/2023	2.26E+07	2.23E+07	2.29E+07	23003958
212	24/05/2023	2.26E+07	2.23E+07	2.29E+07	23004969
213	25/05/2023	2.26E+07	2.23E+07	2.29E+07	23005841
214	26/05/2023	2.26E+07	2.23E+07	2.29E+07	23006602
215	27/05/2023	2.25E+07	2.23E+07	2.28E+07	23007215
216	28/05/2023	2.25E+07	2.22E+07	2.28E+07	23007863

217	29/05/2023	2.25E+07	2.22E+07	2.28E+07	23008668
218	30/05/2023	2.25E+07	2.22E+07	2.28E+07	23009513
219	31/05/2023	2.25E+07	2.22E+07	2.28E+07	23010402

Figure 28. Output Table for Test (Model Tunning)

ds	yhat	yhat_lower	yhat_upper	y	
0	16/07/2022	2.15E+07	2.14E+07	2.16E+07	21578797
1	17/07/2022	2.15E+07	2.14E+07	2.17E+07	21595940
2	18/07/2022	2.16E+07	2.14E+07	2.17E+07	21614510
3	19/07/2022	2.16E+07	2.14E+07	2.17E+07	21631048
4	20/07/2022	2.16E+07	2.15E+07	2.18E+07	21647080
5	21/07/2022	2.16E+07	2.15E+07	2.18E+07	21661129
6	22/07/2022	2.17E+07	2.15E+07	2.18E+07	21673400
7	23/07/2022	2.17E+07	2.15E+07	2.18E+07	21683265
8	24/07/2022	2.17E+07	2.16E+07	2.18E+07	21694468
9	25/07/2022	2.17E+07	2.16E+07	2.19E+07	21707640
10	26/07/2022	2.18E+07	2.16E+07	2.19E+07	21719275
11	27/07/2022	2.18E+07	2.16E+07	2.19E+07	21729746
12	28/07/2022	2.18E+07	2.17E+07	2.19E+07	21738996
13	29/07/2022	2.18E+07	2.17E+07	2.20E+07	21747422
14	30/07/2022	2.18E+07	2.17E+07	2.20E+07	21754679
15	31/07/2022	2.19E+07	2.17E+07	2.20E+07	21763059
16	01/08/2022	2.19E+07	2.17E+07	2.20E+07	21772788
17	02/08/2022	2.19E+07	2.17E+07	2.21E+07	21781865
18	03/08/2022	2.19E+07	2.17E+07	2.21E+07	21790157
19	04/08/2022	2.20E+07	2.18E+07	2.21E+07	21797381
20	15/08/2022	2.19E+07	2.18E+07	2.21E+07	21865083
21	16/08/2022	2.19E+07	2.18E+07	2.21E+07	21870549
22	17/08/2022	2.19E+07	2.18E+07	2.21E+07	21875539
23	18/08/2022	2.20E+07	2.18E+07	2.21E+07	21880060
24	19/08/2022	2.20E+07	2.18E+07	2.21E+07	21884016
25	20/08/2022	2.20E+07	2.19E+07	2.21E+07	21887489
26	21/08/2022	2.20E+07	2.19E+07	2.21E+07	21891529
27	22/08/2022	2.20E+07	2.19E+07	2.22E+07	21896695
28	23/08/2022	2.20E+07	2.19E+07	2.22E+07	21901421
29	24/08/2022	2.20E+07	2.19E+07	2.22E+07	21905925
30	25/08/2022	2.20E+07	2.19E+07	2.22E+07	21909679
31	26/08/2022	2.21E+07	2.19E+07	2.22E+07	21913099
32	27/08/2022	2.21E+07	2.19E+07	2.22E+07	21916177
33	28/08/2022	2.21E+07	2.19E+07	2.22E+07	21919586
34	29/08/2022	2.21E+07	2.19E+07	2.22E+07	21924217
35	30/08/2022	2.21E+07	2.20E+07	2.23E+07	21929383

36	31/08/2022	2.21E+07	2.20E+07	2.23E+07	21933935
37	01/09/2022	2.21E+07	2.20E+07	2.23E+07	21938232
38	02/09/2022	2.21E+07	2.20E+07	2.23E+07	21941995
39	03/09/2022	2.21E+07	2.20E+07	2.23E+07	21945364
40	14/09/2022	2.21E+07	2.19E+07	2.22E+07	21993451
41	15/09/2022	2.21E+07	2.19E+07	2.22E+07	21998043
42	16/09/2022	2.21E+07	2.19E+07	2.22E+07	22002138
43	17/09/2022	2.21E+07	2.19E+07	2.22E+07	22005631
44	18/09/2022	2.21E+07	2.19E+07	2.22E+07	22009932
45	19/09/2022	2.21E+07	2.19E+07	2.22E+07	22016044
46	20/09/2022	2.21E+07	2.19E+07	2.22E+07	22023780
47	21/09/2022	2.21E+07	2.19E+07	2.22E+07	22031251
48	22/09/2022	2.21E+07	2.19E+07	2.22E+07	22037990
49	23/09/2022	2.21E+07	2.19E+07	2.22E+07	22044397
50	24/09/2022	2.21E+07	2.19E+07	2.22E+07	22050045
51	25/09/2022	2.21E+07	2.19E+07	2.22E+07	22057218
52	26/09/2022	2.21E+07	2.20E+07	2.23E+07	22065745
53	27/09/2022	2.21E+07	2.20E+07	2.23E+07	22073886
54	28/09/2022	2.21E+07	2.20E+07	2.23E+07	22082183
55	29/09/2022	2.22E+07	2.20E+07	2.23E+07	22089983
56	30/09/2022	2.22E+07	2.20E+07	2.23E+07	22097487
57	01/10/2022	2.22E+07	2.20E+07	2.23E+07	22104476
58	02/10/2022	2.22E+07	2.20E+07	2.23E+07	22113584
59	03/10/2022	2.22E+07	2.21E+07	2.24E+07	22125130
60	14/10/2022	2.22E+07	2.21E+07	2.24E+07	22223867
61	15/10/2022	2.23E+07	2.21E+07	2.24E+07	22230211
62	16/10/2022	2.23E+07	2.21E+07	2.24E+07	22237917
63	17/10/2022	2.23E+07	2.22E+07	2.24E+07	22247797
64	18/10/2022	2.23E+07	2.22E+07	2.24E+07	22255938
65	19/10/2022	2.23E+07	2.22E+07	2.24E+07	22263426
66	20/10/2022	2.23E+07	2.22E+07	2.24E+07	22269906
67	21/10/2022	2.23E+07	2.22E+07	2.25E+07	22276118
68	22/10/2022	2.24E+07	2.22E+07	2.25E+07	22281026
69	23/10/2022	2.24E+07	2.22E+07	2.25E+07	22286495
70	24/10/2022	2.24E+07	2.23E+07	2.25E+07	22293202
71	25/10/2022	2.24E+07	2.23E+07	2.25E+07	22298791
72	26/10/2022	2.24E+07	2.23E+07	2.25E+07	22304157
73	27/10/2022	2.24E+07	2.23E+07	2.25E+07	22308932
74	28/10/2022	2.24E+07	2.23E+07	2.26E+07	22313254
75	29/10/2022	2.24E+07	2.23E+07	2.26E+07	22316788
76	30/10/2022	2.25E+07	2.23E+07	2.26E+07	22320994
77	31/10/2022	2.25E+07	2.23E+07	2.26E+07	22326061
78	01/11/2022	2.25E+07	2.24E+07	2.26E+07	22330558
79	02/11/2022	2.25E+07	2.24E+07	2.26E+07	22334513
80	13/11/2022	2.24E+07	2.23E+07	2.26E+07	22369955
81	14/11/2022	2.25E+07	2.24E+07	2.26E+07	22373821

82	15/11/2022	2.25E+07	2.24E+07	2.26E+07	22377309
83	16/11/2022	2.25E+07	2.24E+07	2.26E+07	22380533
84	17/11/2022	2.25E+07	2.24E+07	2.26E+07	22383518
85	18/11/2022	2.25E+07	2.24E+07	2.26E+07	22386251
86	19/11/2022	2.25E+07	2.24E+07	2.26E+07	22388655
87	20/11/2022	2.25E+07	2.24E+07	2.26E+07	22391511
88	21/11/2022	2.25E+07	2.24E+07	2.26E+07	22395397
89	22/11/2022	2.25E+07	2.24E+07	2.27E+07	22399146
90	23/11/2022	2.25E+07	2.24E+07	2.27E+07	22402715
91	24/11/2022	2.25E+07	2.24E+07	2.27E+07	22405999
92	25/11/2022	2.25E+07	2.24E+07	2.27E+07	22408867
93	26/11/2022	2.25E+07	2.24E+07	2.27E+07	22411370
94	27/11/2022	2.25E+07	2.24E+07	2.27E+07	22414790
95	28/11/2022	2.25E+07	2.24E+07	2.27E+07	22419358
96	29/11/2022	2.25E+07	2.24E+07	2.27E+07	22423655
97	30/11/2022	2.25E+07	2.24E+07	2.27E+07	22427546
98	01/12/2022	2.25E+07	2.24E+07	2.26E+07	22430989
99	02/12/2022	2.25E+07	2.24E+07	2.26E+07	22434533
100	13/12/2022	2.23E+07	2.22E+07	2.24E+07	22486694
101	14/12/2022	2.23E+07	2.22E+07	2.24E+07	22493064
102	15/12/2022	2.23E+07	2.22E+07	2.24E+07	22499017
103	16/12/2022	2.24E+07	2.23E+07	2.25E+07	22504916
104	17/12/2022	2.24E+07	2.23E+07	2.25E+07	22509541
105	18/12/2022	2.24E+07	2.23E+07	2.25E+07	22515183
106	19/12/2022	2.24E+07	2.23E+07	2.25E+07	22524022
107	20/12/2022	2.25E+07	2.24E+07	2.26E+07	22532446
108	21/12/2022	2.25E+07	2.24E+07	2.26E+07	22540058
109	22/12/2022	2.26E+07	2.24E+07	2.27E+07	22546756
110	23/12/2022	2.26E+07	2.25E+07	2.27E+07	22552385
111	24/12/2022	2.26E+07	2.25E+07	2.28E+07	22556448
112	25/12/2022	2.26E+07	2.25E+07	2.27E+07	22560110
113	26/12/2022	2.26E+07	2.25E+07	2.27E+07	22564959
114	27/12/2022	2.27E+07	2.26E+07	2.28E+07	22571411
115	28/12/2022	2.29E+07	2.28E+07	2.30E+07	22579654
116	29/12/2022	2.29E+07	2.28E+07	2.30E+07	22587176
117	30/12/2022	2.30E+07	2.29E+07	2.31E+07	22593153
118	31/12/2022	2.30E+07	2.29E+07	2.32E+07	22597351
119	01/01/2023	2.30E+07	2.29E+07	2.31E+07	22600954
120	12/01/2023	2.29E+07	2.28E+07	2.30E+07	22643010
121	13/01/2023	2.29E+07	2.28E+07	2.30E+07	22645104
122	14/01/2023	2.29E+07	2.28E+07	2.31E+07	22646571
123	15/01/2023	2.29E+07	2.28E+07	2.31E+07	22648400
124	16/01/2023	2.30E+07	2.28E+07	2.31E+07	22650917
125	17/01/2023	2.30E+07	2.29E+07	2.31E+07	22653083
126	18/01/2023	2.30E+07	2.29E+07	2.31E+07	22655156
127	19/01/2023	2.30E+07	2.29E+07	2.31E+07	22657125

128	20/01/2023	2.30E+07	2.29E+07	2.31E+07	22658955
129	21/01/2023	2.30E+07	2.29E+07	2.32E+07	22660462
130	22/01/2023	2.30E+07	2.29E+07	2.32E+07	22662344
131	23/01/2023	2.30E+07	2.29E+07	2.32E+07	22664953
132	24/01/2023	2.31E+07	2.29E+07	2.32E+07	22667686
133	25/01/2023	2.31E+07	2.29E+07	2.32E+07	22670285
134	26/01/2023	2.31E+07	2.29E+07	2.32E+07	22672782
135	27/01/2023	2.31E+07	2.29E+07	2.32E+07	22675197
136	28/01/2023	2.31E+07	2.29E+07	2.32E+07	22677139
137	29/01/2023	2.31E+07	2.29E+07	2.32E+07	22679568
138	30/01/2023	2.31E+07	2.30E+07	2.32E+07	22683058
139	31/01/2023	2.31E+07	2.29E+07	2.32E+07	22686548
140	11/02/2023	2.27E+07	2.26E+07	2.29E+07	22721450
141	12/02/2023	2.27E+07	2.26E+07	2.29E+07	22724592
142	13/02/2023	2.27E+07	2.26E+07	2.29E+07	22729428
143	14/02/2023	2.27E+07	2.26E+07	2.29E+07	22734049
144	15/02/2023	2.27E+07	2.26E+07	2.29E+07	22738208
145	16/02/2023	2.27E+07	2.26E+07	2.29E+07	22742431
146	17/02/2023	2.27E+07	2.26E+07	2.29E+07	22746427
147	18/02/2023	2.27E+07	2.26E+07	2.29E+07	22749523
148	19/02/2023	2.27E+07	2.25E+07	2.29E+07	22753174
149	20/02/2023	2.27E+07	2.26E+07	2.29E+07	22758012
150	21/02/2023	2.27E+07	2.26E+07	2.29E+07	22762646
151	22/02/2023	2.27E+07	2.26E+07	2.29E+07	22766794
152	23/02/2023	2.27E+07	2.25E+07	2.29E+07	22770497
153	24/02/2023	2.27E+07	2.26E+07	2.29E+07	22773836
154	25/02/2023	2.27E+07	2.25E+07	2.29E+07	22776533
155	26/02/2023	2.27E+07	2.25E+07	2.28E+07	22779659
156	27/02/2023	2.27E+07	2.25E+07	2.29E+07	22784305
157	28/02/2023	2.27E+07	2.25E+07	2.29E+07	22788789
158	01/03/2023	2.27E+07	2.25E+07	2.28E+07	22793078
159	02/03/2023	2.27E+07	2.25E+07	2.28E+07	22796935
160	13/03/2023	2.27E+07	2.26E+07	2.29E+07	22842838
161	14/03/2023	2.28E+07	2.26E+07	2.29E+07	22848334
162	15/03/2023	2.28E+07	2.26E+07	2.29E+07	22853108
163	16/03/2023	2.28E+07	2.27E+07	2.29E+07	22857906
164	17/03/2023	2.28E+07	2.27E+07	2.29E+07	22862005
165	18/03/2023	2.28E+07	2.27E+07	2.30E+07	22865406
166	19/03/2023	2.28E+07	2.27E+07	2.30E+07	22869473
167	20/03/2023	2.28E+07	2.27E+07	2.30E+07	22874677
168	21/03/2023	2.29E+07	2.27E+07	2.30E+07	22879879
169	22/03/2023	2.29E+07	2.28E+07	2.30E+07	22884670
170	23/03/2023	2.29E+07	2.28E+07	2.31E+07	22889039
171	24/03/2023	2.29E+07	2.28E+07	2.31E+07	22893152
172	25/03/2023	2.30E+07	2.28E+07	2.31E+07	22896162
173	26/03/2023	2.30E+07	2.28E+07	2.31E+07	22899634

174	27/03/2023	2.30E+07	2.28E+07	2.31E+07	22903965
175	28/03/2023	2.30E+07	2.29E+07	2.32E+07	22907892
176	29/03/2023	2.30E+07	2.29E+07	2.32E+07	22911674
177	30/03/2023	2.31E+07	2.29E+07	2.32E+07	22915061
178	31/03/2023	2.31E+07	2.29E+07	2.32E+07	22917843
179	01/04/2023	2.31E+07	2.29E+07	2.32E+07	22920158
180	12/04/2023	2.31E+07	2.29E+07	2.33E+07	22945177
181	13/04/2023	2.31E+07	2.29E+07	2.33E+07	22947273
182	14/04/2023	2.31E+07	2.29E+07	2.33E+07	22949002
183	15/04/2023	2.31E+07	2.29E+07	2.33E+07	22950411
184	16/04/2023	2.31E+07	2.29E+07	2.33E+07	22952369
185	17/04/2023	2.31E+07	2.29E+07	2.33E+07	22954663
186	18/04/2023	2.31E+07	2.29E+07	2.33E+07	22956629
187	19/04/2023	2.31E+07	2.29E+07	2.33E+07	22958484
188	20/04/2023	2.31E+07	2.30E+07	2.33E+07	22960112
189	21/04/2023	2.31E+07	2.30E+07	2.33E+07	22961579
190	22/04/2023	2.31E+07	2.29E+07	2.33E+07	22962865
191	23/04/2023	2.31E+07	2.29E+07	2.33E+07	22964416
192	24/04/2023	2.31E+07	2.29E+07	2.33E+07	22966275
193	25/04/2023	2.31E+07	2.29E+07	2.33E+07	22968035
194	26/04/2023	2.31E+07	2.29E+07	2.33E+07	22969730
195	27/04/2023	2.31E+07	2.29E+07	2.33E+07	22971355
196	28/04/2023	2.31E+07	2.29E+07	2.33E+07	22972819
197	29/04/2023	2.31E+07	2.29E+07	2.33E+07	22974138
198	30/04/2023	2.31E+07	2.29E+07	2.33E+07	22975507
199	01/05/2023	2.31E+07	2.29E+07	2.33E+07	22977274
200	12/05/2023	2.29E+07	2.28E+07	2.31E+07	22992568
201	13/05/2023	2.29E+07	2.28E+07	2.31E+07	22993419
202	14/05/2023	2.29E+07	2.27E+07	2.31E+07	22994522
203	15/05/2023	2.29E+07	2.27E+07	2.30E+07	22995753
204	16/05/2023	2.29E+07	2.27E+07	2.30E+07	22996899
205	17/05/2023	2.29E+07	2.27E+07	2.30E+07	22997989
206	18/05/2023	2.29E+07	2.27E+07	2.30E+07	22999031
207	19/05/2023	2.29E+07	2.27E+07	2.30E+07	22999958
208	20/05/2023	2.28E+07	2.27E+07	2.30E+07	23000729
209	21/05/2023	2.28E+07	2.27E+07	2.30E+07	23001676
210	22/05/2023	2.28E+07	2.27E+07	2.30E+07	23002845
211	23/05/2023	2.28E+07	2.27E+07	2.30E+07	23003958
212	24/05/2023	2.28E+07	2.27E+07	2.30E+07	23004969
213	25/05/2023	2.28E+07	2.27E+07	2.30E+07	23005841
214	26/05/2023	2.28E+07	2.26E+07	2.29E+07	23006602
215	27/05/2023	2.28E+07	2.26E+07	2.29E+07	23007215
216	28/05/2023	2.28E+07	2.26E+07	2.29E+07	23007863
217	29/05/2023	2.27E+07	2.26E+07	2.29E+07	23008668
218	30/05/2023	2.28E+07	2.26E+07	2.29E+07	23009513
219	31/05/2023	2.28E+07	2.26E+07	2.29E+07	23010402

