Department of Applied Informatics

MSc in Artificial Intelligence and Data Analytics

Master's Thesis

PREDICTION OF CASES AND TRANSMISSION RATE OF COVID-19 WITH MACHINE LEARNING TECHNIQUES

Vazakidis Dimitrios

Supervising professor: Nikolaos Samaras

Submitted as required for obtaining the master's degree in Artificial Intelligence and Data Analytics

August 2023

Abstract

The COVID-19 pandemic emerged in 2019 and has since evolved into a global crisis of paramount concern for governments and people worldwide. Its initial appearance marked the onset of significant disruptions, affecting both economies and daily lives. The repercussions have been profound, leading to the loss of lives and livelihoods, and even now, four years later, its impact continues to be felt.

Comprehending the patterns and trends within the foundational data related to the pandemic holds immense importance. This knowledge is crucial for governments as they formulate policies and strategies to manage the crisis effectively. It is equally vital for individuals to stay informed and make informed decisions regarding their health and safety in these challenging times.

The pandemic's enduring impact underscores the need for continued vigilance, research, and cooperation on a global scale to mitigate its effects and prevent similar crises in the future.

Contents

Abstract..... i
Contents..... ii
1 INTRODUCTION 1
1.1 Introductory Analysis 1
1.2 Purpose of the thesis 2
1.3 Work Structure 3
PART ONE 4
2.THEORETICAL ANALYSIS 4
2.2. Research Objectives and Questions 4
2.3. Previous studies on COVID-19 prediction using machine learning in Python 5
2.3.1. Overview of relevant machine learning algorithms for case and transmission rate prediction 9
2.3.1.1. Linear Regression 9
2.3.1.2. Support Vector Regression (SVR).....9
2.3.1.3. Random Forest Regression.....10
2.3.1.4. Gradient Boosting Regression 12
2.3.2. Neural Networks 12
2.3.3. Gaussian Processes 15
2.4. K-Nearest Neighbours (KNN).....17
2.5. Ethical Considerations 19
2.5.1. Addressing Ethical Implications of using Machine Learning in Public Health 19
2.5.2. Ensuring Privacy and Security of Data Used in the Thesis 20
3. REVIEW OF THE LITERATURE 21
3.1. Introduction 21
3.2. Long Short-Term Memory-Attention (LSTM-Attention) in COVID-19 prediction 21
3.3. XGBoost in COVID-19 Prediction 22
3.4. Data sources and pre-processing techniques in Python 24
3.4.1 Decision Tree Classification 25
3.4.2. Logistic Regression 26
3.4.3. Random Forest Classification 27
3.4.4. Data Collection and Pre-processing 28
3.4.4.1. Description of Data Sources and Datasets Obtained.....28
3.5. Data Pre-Processing using Python (data cleaning, normalization, feature engineering) 29

3.5.1. Data Profiling 30
3.5.2. Data Cleansing 30
3.5.3. Data Reduction 31
3.5.4. Data Transformation 31
3.5.5. Data Enrichment 31
3.5.6 Data Validation 32
3.5.7 Handling Missing Data and Outliers in Python 32
3.6 Geographical and population data related to Mexico 34
PART TWO 35
4. THE PROGRAM 35
4.1 Machine Learning Models for Transmission Rate Prediction 35
4.2 Visualizations and insights gained from the data using Python libraries 35
4.2.1 Identification of key features and patterns in Python 43
4.2.2 Feature Engineering and Selection 44
4.2.3 Selection of relevant features for prediction 44
4.2.4 Feature engineering techniques applied in Python 45
4.2.4.1 Domain-Knowledge-based feature generation 46
4.2.4.2 Categorical Encoding 47
4.2.4.3 Data reduction 47
4.2.5 Feature selection methods used in Python (Such as Recursive Feature Elimination, Feature Importance) 48
4.2.5.1 Recursive feature elimination 48
4.2.5.2 Feature importance 49
4.2.5.3 Correlation analysis 49
4.3 Machine Learning Models for Cases Prediction 49
4.3.1 Description and implementation of regression algorithms in Python 50
4.3.1.1 Linear Regression 50
4.3.1.2 Simple linear regression 50
4.3.1.3 Multiple Linear Regressions 50
4.3.1.4 Random Forest Regression 51
4.3.1.4.1 Model selection and evaluation using Python (metrics like Mean Squared Error, R-squared) 51
4.3.1.4.1.1 Mean square error 51
4.3.1.4.1.2 R-squared 52

4.3.2 The visualization 55
4.3.2 Decision Tree Model.....56
4.4 Machine Learning Models for Transmission Rate Prediction.....57
4.4.1 Description and implementation of regression algorithms in Python 57
4.4.2 Model selection and evaluation for transmission rate prediction 57
4.4.3 Hyperparameter tuning for transmission rate prediction models.....60
4.4.3.1 Model Interpretability 60
4.4.3.2 SHAP 61
4.4.3.3 LIME.....61
4.4.3.4 Anchors 61
4.4.4 Interpretation of results and insights into key factors affecting cases and transmission rate 61
5. CONCLUSION 64
6. FURTHER RESEARCH 68
7. BIBLIOGRAPHY 71

1 INTRODUCTION

1.1. Introductory Analysis

Coronavirus belongs to the genus of coronaviruses, which are zoonotic. The word "corona" is from a Latin word, meaning "crown" and this alludes to the outer spikes of the virus which are crown-like and club-shaped. The first records of coronaviruses were first discovered in the 1930s after the respiration which affected the chicken. Human coronaviruses were first discovered in the 1960s, and they can be transmitted through respiratory tract infections, which range from those that are minor to ones that are fatal. The purpose of this thesis is to use data provided on Kaggle to identify and understand patterns in the data. The findings from this thesis could help governments and service providers to have proper planning and allocation of resources, which could have a huge impact in terms of minimizing loss of lives. In this thesis, Python is relied upon to clean, process, and generate models from the data. I seek to unearth hidden trends using machine learning and other data analysis methods.

In December 2019, COVID-19 was detected for the first time in the world and spread to all countries around the world and this made it to be declared as a pandemic in January 2020. The clinical result of the condition ranged from asymptomatic to mild condition to very serious conditions, to consequently, death in various cases. The condition is contagious and viral, hence spreading aggressively worldwide. Therefore, it is necessary to determine the transmission rate and cases of COVID-19.

Lack of sufficient information concerning the early-stage symptoms is among the factors that result in the spread of the condition. The citizens in most cases are not aware that they are infected, hence travel without understanding the potential disease transmission.

Throughout the last decade, there has been an expansion in the digital approaches to solving different difficulties and challenges in the health sector, as well as for the prevention of the spread of transmissible diseases. The digital machinery used to deal with COVID-19 is tracked as part of the methods of dealing with COVID-19 since the prognosis of the patient outcome is a complicated topic. Therefore, predicting as well as diagnosing disease is easier when machine learning is used. The self-learning machine learning is able to learn from the machines and use the learning to predict the spread accurately. Several conditions have been predicted by the use of Machine learning and these include the human immunodeficiency virus (virus), Ebola virus, and hepatitis.

1.2. Purpose of the thesis

This thesis aims to identify and understand patterns in the data that represent the rate and cases of COVID-19. This can be used to predict COVID-19 cases in the world. When the pattern is mastered and can be predicted, then it will be possible for the public health personnel to get involved in putting measures in place that can be used to prevent the spread of this condition. The findings from this thesis could help governments and service providers to have proper planning and allocation of resources, which could have a huge impact in terms of minimizing loss of lives.

Figure 1: Total Cases of COVID-19 worldwide, these figures include the recovered cases and the deaths.

1.3.Work Structure

A. Gathering critical document

a. Gathering the critical document to be used for the analysis

b. Identifying the content that is related to the thesis

B. Identifying Key terms

a. Analyse the documents and identify the key terms and deliverables

b. Identify appropriate thesis contents

PART ONE

2.THEORETICAL ANALYSIS

2.1 Introduction

COVID-19 is a serious disease whose occurrence has the potential to lead to multiple losses of lives over a short period. As stated earlier, COVID-19 is transmitted through respiration, and therefore controlling its spread is usually a hard task for the people involved, leading to a high rate of victims. In most cases, during an outbreak of diseases such as COVID-19, one of the main challenges is the shortage of available resources to help manage and control its spread. The lack of resources is attributed to the lack of preparation, hence most of the already available resources are allocated to different areas since no indication of the disease occurrence was predicted before. Therefore, it is important to have a method that can be used to predict both the cases and the transmission rate. This way, policymakers, the institutions involved, and the public that is known to be directly affected can prepare.

2.2. Research Objectives and Questions

There are two important research topics that are associated with this thesis. They include

- To determine the pattern of COVID-19 spread
- To determine the pattern of COVID-19 spread by use of Machine learning method

In this section, all the previous literature that is related to this research is reviewed to establish a connection between the variables as well as to identify a research gap to help with the development of the current research. All the articles that are reviewed in this section are peer-reviewed and are considered to be reliable.

2.3. Previous studies on COVID-19 prediction using machine learning in Python

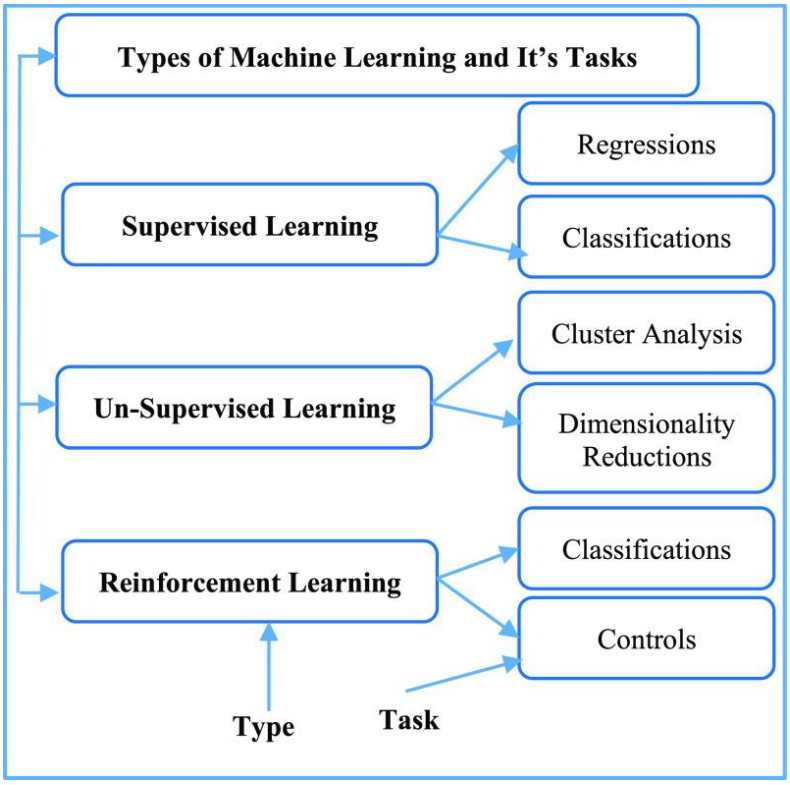
Coronaviruses belong to a family of viruses which affect both humans and animals. The virus can cause mild diseases such as the common cold, while other serotypes can cause severe diseases such as the Middle East respiratory syndrome coronavirus (MERS-CoV). The disease was first reported in the Wuhan region of China but spread globally. The earliest date of the beginning of the persistence of the condition was on 1st December 2019. The symptomatology associated with the patients included fever, Malaise, and dry cough, and the patients were diagnosed with viral pneumonia. Initially, the International Committee on Taxonomy of Viruses named COVID-19 based on the area where the first case was reported and the associated pneumonia symptoms. On December 31, 2019, the World Health Organization was told about a pneumonia outbreak in Wuhan, a city with up to eleven million people. Wuhan was important for China's culture and economy, so the illness and deaths would affect not only health but also economic and social well-being. By 5th January, 59 Data cases had been reported, and the numbers accelerated. Ten days later, the World Health Organization confirmed 282 new cases, whereby the conditions have spread to other countries such as Japan, South Korea, and Thailand (Duffin, 2022). Additionally, within the days, there were six deaths and fifty-one individuals in critical condition.

Human coronaviruses spread by use of droplets and close personal contact when there is no protection. The novel coronavirus was first identified in Wuhan, Hubei Province, in the country of China and the continent of Asia, and it resulted in the outbreak of pneumonia and numerous cases have been identified in different parts of the world. The COVID-19 pandemic has become a major concern for governments and people from all over the world. Since when it first appeared in the year 2019, it brought great disruptions to economies and lives. Loss of life and livelihoods were experienced as a result, and the impact is still evident, 4 years later. Understanding the trends in the underlying data is important for governments and individuals. The outbreaks of novel virus infections such as that in the case of coronavirus are of public health concern. The human coronavirus disease (COVID-19) became the fifth documented global pandemic after the 1918 flu pandemic (Duffin, 2022). The COVID-19 pandemic has become a major concern for governments and people from all over the world. Since when it first appeared in the year 2019, it brought great disruptions to economies and lives. Loss of life and livelihoods were experienced as a result, and the impact is still evident, 4 years later. Understanding the trends in the underlying data is important for governments and individuals.

COVID-19 is believed to be a spill-over of an animal coronavirus that evolved and obtained the adaptability of transmissions from one human being to another (McCarther, 2022). The virus was highly contagious and spread to almost all countries globally, affecting the human population while negatively impacting social interactions. Based on the emergence of the condition, scientists have embarked on various studies to enhance the comprehension of its prevalence, including antiviral designs and vaccine developments. The condition, however, ceased to be a health threat globally.

Machine learning is critical in leveraging technology, especially artificial intelligence. Machine learning is a subdivision of artificial intelligence but is often called AI based on its decision-making ability. Machine learning evolved from mathematical modelling that involved neural networks in 1943 after an attempt to mathematically eradicate thought processes and decision-making, especially in human cognitions (Gilliam, 2018). In 1950, a Turing test was proposed by Alan Turning, and the aim was to discover machines that were considered either intelligent or unintelligent (Roberge & Castelle, 2020). This criterion aimed at ensuring machines receive statuses as intelligent, which implied the ability to convince a human being that it could also reason just like them. After the attempt, Machine learning was conceived at Dartmouth College through a research program. At this point, the intelligent machine learning algorithm and computer programs became a reality, performing activities initially done by human beings, such as planning travel routes, especially for salespeople and tourists, and playing board games with human beings. It is worth noting that machine learning has gradually advanced, doing everything a human would do, such as adopting speech recognition, hence realizing a growth from a mathematical model to a sophisticated technology. Machine learning has greatly changed over time, and the growth has been facilitated by the growth of the internet and the increased availability of useable data, a shift from a knowledge-driven approach to a data-driven form.

A.



B. Figure 2: Different Types of Machine Learning and Tasks

Python with Sklearn is one of the useful libraries in machine learning since it offers a selection of the most effective tools, therefore, it was used in this case. Python with Sklearn offers statistical modelling which includes clustering, classification, regression, and dimension reduction by use of a consistency interface. Except for a focus on loading, summarizing, and manipulating data, Python with Sklearn allows the modeling of data. This includes the use of supervised learning algorithms, unsupervised learning algorithms, clustering, cross-validation, dimensionality reduction, ensemble methods, feature extraction, feature selection, and open source.

2.3.1. Overview of relevant machine learning algorithms for case and transmission rate prediction

The process of predicting COVID-19 cases and the transmission rate can be done using different machine-learning algorithms. Each of the algorithms has its advantages and disadvantages, hence the choice of the algorithm relies on the data as well as the requirements for the prediction rate. Some relevant machine learning that can be used includes linear regression analysis, Support Vector Regression (SVR), Random Forest Regression, Gradient Boosting Regression, Neutral Networks, Long Short-Term Memory, Gaussian Processing, K-Nearest Neighbours, Long Short Term Memory –Attention, XGBoost.

2.3.1.1. Linear Regression

Linear regression is used for modelling the relationship between the dependent variable and another independent variable. The dependent variables in this case are known as the response or target variable, while the independent variables are known as features. In the context of COVID-19 cases and rate of transmission prediction, the linear regression method can be used in the estimation of the relationship between some variables and the transmission rate as well as number of cases. This algorithm assumes a linear form of relationship existing between the target variables and the input features. Even though linear regression is straightforward and simple, it is not suitable for the forms of relationships that are non-linear, hence in this case, more advanced once is to be used.

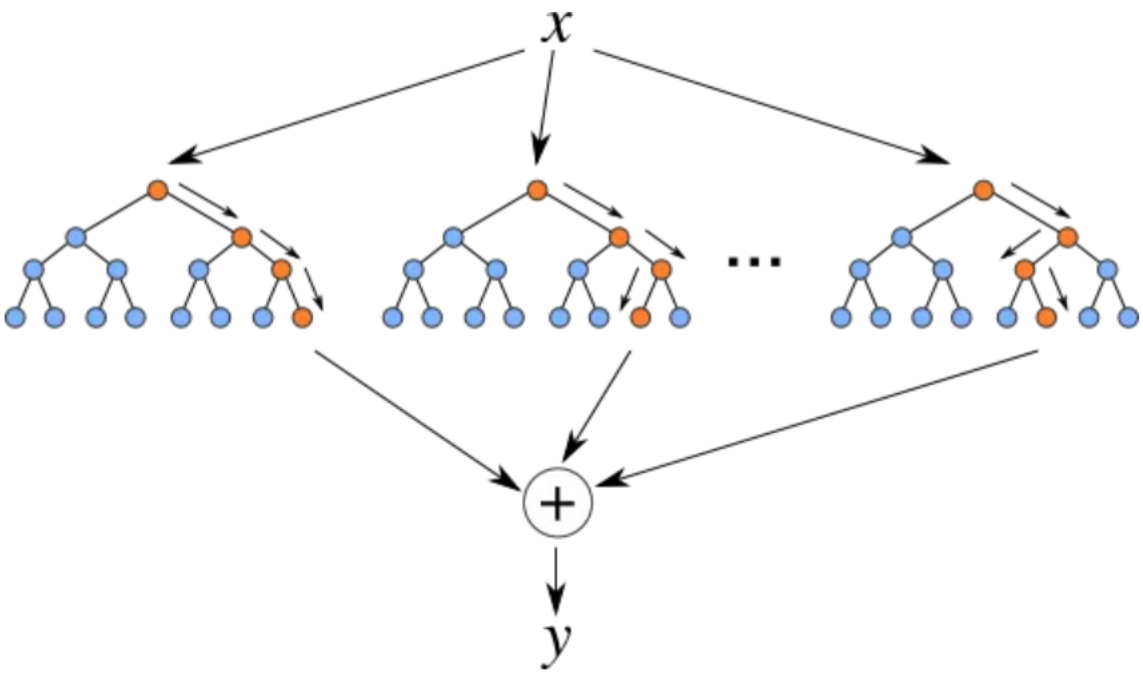
2.3.1.2. Support Vector Regression (SVR)

Support Vector Regression (SVR) is designed for regression-related tasks since it is a versatile algorithm that is used in the case of predicting continuous values. This makes it to

be applicable for different tasks such as COVID-19 rate of transmission and case prediction. The method aims to find a hyperplane in high-dimensional feature space, and this can be used to approximate the relationship existing between the target variables and the input features. In this case, the hyperplane is defined by different support vectors, which are subsets of the training data points that have an influence on the construction of the hyperplane. Support Vector Regression (SVR) is of help when it comes to dealing with the dataset where the relationship between the target variable and the features is non-linear and in cases where there are possibilities of having potential outliers.

2.3.1.3. Random Forest Regression

Random Forest regression is a method that builds different decision trees and then gets the average of their predictions as a method of reducing over-lifting. It is also used to handle non-linearity. The method uses the function of the kernel to map the input data into a higher dimension space where the relationship is better presented. It is important to use the appropriate kernel and tune hyperparameters. Therefore, Random Forest Regression can be used to predict the cases and rate of transmission of COVID-19. A random forest fits a number of classifying decision trees on different subsamples of the dataset, and it uses the averaging method for improving the control over-fitting and predictive accuracy. The ensemble learning algorithm uses the ensemble learning method for regression. It combines predictions from multiple machine learning algorithms to make a more accurate method than just having a single model.



The above diagram shows the structure of a Random Forest, and the trees run in parallel without interaction taking place among them. Random Forest Regression is an extension of the decision tree algorithm, which is mainly used in the numeric value. This makes it also a tool that can be used in the prediction of transmission rates and cases of COVID-19 cases. Random Forest Regression ensures there is creation of multiple decision trees which are used in the process of training, and it combines their prediction for improving accuracy as well as to reduce over-fitting. Random Forest Decision ensemble decision trees and each of the trees are trained on a particular subset of data and a random subset of features to reduce risks associated with overfitting and also to increase its robustness. There is also bagging, which involves training every tree on a bootstrapped subset of training data. It ensures diversity among the trees. It also has feature randomness, since at each node of the decision tree, a random subset is the one considered in the case of splitting. Such a form of randomness reduces variances among the dataset.

2.3.1.4. Gradient Boosting Regression

Another method that can be used is gradient boosting regression, and this is a machine learning algorithm that belongs to the ensemble learning techniques. It is mainly used for regression tasks, and it is more effective in the cases of predicting continuous numeric values. It combines multiple weak learners which form a strong predictive model. In this method, weak learner, which are the decision trees, builds up through sequential training to correct errors made in the previous trees. Gradient boosting uses the decision trees which are of fixed size, and they have learners as the base, in some cases, the improvement can be made to lift the quality of its fit. The gradient boosting has high predictive accuracy even in the cases of complex as well as noisy datasets. It also has an algorithm that can capture the relationships that are nonlinear between features, as well as on the target variable. The other strength of this method is that it is not prone to overfitting compared to the individual tree's method. The drawback of this is associated with the training of multiple trees sequentially which can be computationally demanding, more so in the large dataset as in the case of that of COVID-19. The second drawback is that the algorithm has numerous hyperparameters which must be tuned to realize the optimal performance.

2.3.2. Neural Networks

Neutral Networks (NNs) have been used as a model of the human nervous system by use of different physical and mathematical processes. Neural networks assist in making decisions and in prediction of the trends as it is used in regressions and other methods. It is used for tasks with a large amount of data with a complex pattern. Neurons are the basic building blocks in the case of neural networks as they receive input data, then perform the necessary computation and also for producing the output. The neurons are connected

through weighted connections. The neurons are organized in layers, It is the input layer that receives raw data, the output layers are used to produce the output, which is the final production and the hidden layers are for processing the information.

Long Short-Term Memory (LSTM) is a form of recurrent neural network (RNN) construction built up intending to combat the challenges existing within the initial RNNs, especially when obtaining long-term dependencies and arrangements within the sequential information (Sabry, 2023). In most instances, LSTM is utilized in tasks revolving around sequential data, including time series. Processing of natural language and speech recognition are other demanding tasks for LSTM. LSTM effectively manages long-range dependencies while keeping data longer.

They have memory cells that enhance their chances of keeping data for longer durations (Sabry, 2023). Memory cells are further applicable in obtaining long-term dependencies based on their abilities to store and update data. LSTM has three gates that manage information movement in and out of the memory cells. The forget gates determine the data to reject from the initial state (Ly et al., 2021). On the other hand, the input gate determines the new information to be stored while the output gate manages data output from the existing state. The hidden state reveals the processed data at specific stages and is therefore utilized as input for the subsequent time step (Ly et al., 2021). Notably, it is referred to as short-term memory.

Back Propagation Through Time (BPTT) is another LSTM feature whose algorithm is used in both trainings, especially in the feedforward neural networks (Gers, 2001). BPTT focuses on the accumulated errors while updating the model's parameters across time

steps. Additionally, LSTM is associated with the vanishing gradient problem as its feature since it addresses the vanishing gradient problem, which may occur in traditional RNNs when gradients reduce in size during backpropagation, which may challenge learning long-term dependencies.

LSTMs are used in time series forecasting for the future values within the time series data while focusing on stock prices, weather data, and case counts associated with the COVID-19 virus (Sabry, 2023). It is used in natural language processing for language generation, text classification, sentiment analysis, and machine translation. LSTMs are also used in speech recognition in modelling sequential arrangements in audio information (Chan, 2015). Lastly, they are used in video analysis to perform action recognition and video captioning.

They can be used in tasks that demand extended durations since they capture a long range of dependencies (Chan, 2015). Robustness is their advantage since they are not prone to vanishing gradient problems like the traditional RNNs (Byeon, 2016). This move enables them to acquire a better representation of sequential patterns.

They are associated with computational complexities since their models are expensive when handling huge sequences. In addition, they involve complex data requirements since only an adequate amount of training data generalizes them effectively (Brownlee, 2017). Finally, LSTMs have effectively enhanced deep learning and are applicable in various fields. Therefore, they are subject to future research to improve efficiency and performance.

2.3.3. Gaussian Processes

Gaussian processes (GPs) are non-parametric probabilistic that are robust, flexible, and applicable to supervised learning tasks. Such tasks revolve around classifications and regression (Dym &McKean, 2008). GPs is one of the machine learning algorithms involved in learning fixed numbers of parameters characterized by an infinite-dimensional function space. This feature enables them effectively perform tasks in scenarios where uncertainty estimation is critical and when handling limited data.

The critical characteristics of GPs include the probabilistic model, which models the connection between input features and target variables while handling it as a probability distribution over function (Ibragimov & Rozanov, 2012). Instead of availing point predictions, it presents possible functions that align with the available information. In addition, it has a function space where both mean function and covariance functions define the model. The covariance function is vital in obtaining the similarity between input information stages (Ibragimov & Rozanov, 2012). In addition, it enhances the chances of the model generalizing from observed information to unseen information stages.

Flexibility is another GPs feature associated with its ability to obtain complex nonlinear connections between features and target variables (Lifshits, 2012). This move makes the connections versatile for a series of data arrangements. Bayesian inference is a GPs feature that allows for the incorporation of initial data that define function space and the capacity to update projections as increased data are presented. Finally, uncertainty estimation is a great feature where the GPs avail both projections and uncertain estimations to inform of predictive variances (Lifshits, 2012). The variances are essential since they provide

information about the strength of the projections, which is essential in assessing risks and making decisions.

GPs are applicable in various fields, including regression, where they thesis continuous numeric values. They are further used in time series forecasting to project future values in time series information while working together with uncertainty estimations (Lifshits, 2012). Furthermore, they are used in Bayesian optimization to present the optimal value associated with either parameters or hyperparameters while focusing on the uncertainty. Lastly, they are utilized in reinforcement learning to model the value function and use uncertainty concepts to make decisions.

GPs have various advantages, including uncertain qualifications. They provide a straightforward technique for estimating uncertainties essential in decision-making and critical applications (Marcus & Rosen, 2006). Having few hyperparameters is its other advantage, unlike other machine learning models. Finally, it is non-parametric hence it does not overlook given parametric models for the underlying function, and is more flexible and easily adapts to information patterns.

Computational complexity is its disadvantage, making it expensive for larger datasets. In addition, it is associated with comprehensive storage requirements since storing covariance data is memory intensive (Marcus & Rosen, 2006). They further need to gain the interpretability of simpler models such as linear regression. It is worth noting that the Gaussian Process is an essential tool within the machine learning toolbox, especially when uncertainty projection is a factor to consider while handling limited information. Their

concepts are widely applicable in various areas of research revolving around healthcare and finance, among others.

2.4. K-Nearest Neighbours (KNN)

K-Nearest Neighbours (KNN) is a modest and spontaneous machine learning algorithm applicable to regression and classification activities (Sabry, 2023). The model is both a no-parametric and sluggish learning algorithm. That is, during the training phase, it fails to capture an explicit model. Instead, it captures the training data and bases its projections on the proximity of the new information stages to the existing training information.

KNN works in different stages and contexts, including data storage, whereby during the training phase, it captures all the datasets in its memory (Sabry, 2023). It further works in metrics such as Euclidean distance to evaluate the similarity between information stages. Hamming distance is, in most instances, applicable to categorical features, while Euclidean distance is relevant to continuous features (Roberts, 1990). It is applicable in choosing K, where K is a hyperparameter representing the number of nearest neighbours to focus on during projections. Model evaluation techniques are adopted in selecting the value of K. It is, however, worth noting that the smaller the value of K, the messy the predictions and vice versa. KNN is further used to predict classifications, directing a class label to a new data point according to the majority class existing within the K nearest neighbours (Roberts, 1990). Voting mechanisms are further used to evaluate class labels. It is also used in predictions for regression to thesis the value of new data points as the average of target values of the K nearest neighbours (Roberts, 1990). Finally, KNN works in the weighted KNN where the role of the nearest neighbours is weighted according to their

distance from new information stages, implying that the farther the neighbours, the weaker their influence on the projections and vice versa.

KNN has a series of advantages. It is spontaneous and easier to comprehend, hence a suitable starting point for those joining machine learning, since it is also straightforward to implement (Osipov, 2012). In addition, it is non-parametric, hence avoiding assumptions in the underlying distribution. It is, therefore, flexible and can quickly adapt to any data. Finally, it is appropriate to solve multiclass problems without relying on modifications.

It, however, has limitations, including the fact that it is computationally expensive since the growth in the training data leads to the projections process becoming expensive, especially where large datasets are involved. It is further sensitive to irrelevant features degrading its performance. It further requires a proper feature scale based on its sensitivity to sales of features which may lead to a mess if the features are not effectively scaled before applying them to the algorithm (Kramer, 2013). Finally, it is associated with the curse of dimensionality since the feature increase leads to weakness between the data points and hence a lower performance. It is, however, worth noting that KNN is a vital algorithm when applied to small and medium-sized datasets and where interpretability is a factor. It serves as a baseline performance in benchmarking complex machine learning algorithms. However, other machine learning, such as decision trees and ensemble methods, are more appropriate in large-scale datasets.

2.5. Ethical Considerations

In this section, all the ethical issues that might arise or arise during the use of the machine learning approach are highlighted. Potential solutions are outlined to help ensure that this research does not affect any group of persons during its publication..

2.5.1. Addressing Ethical Implications of using Machine Learning in Public Health

There are a number of ethical issues that may arise as a result of employing machine learning approaches, especially in the public health sector. Some of these ethical concerns include issues with privacy, fairness, conflicts of interest, transparency, and accountability.

Privacy: Electronic health records have led to a rise in ethical issues such as data privacy, data ownership, and secondary use of data. With electronic data records, personal information has become more vulnerable as access to this record can be way easier compared to using more traditional methods of data-keeping. Therefore, since machine learning approaches require retrieval of some personal data on online datasets it is important to be careful not to use private medical records of individuals without consent.

Fairness: A fairness concern arises from issues of bias. As indicated in this research when trying to mitigate issues that might arise due to missing data and outliers, it is possible to provide biased information altering the accuracy of results and leading to cases of unfairness. It is therefore important to ensure that fairness is upheld by incorporating techniques that will not distort the actual measurements in the existing data.

Conflict of Interest and Transparency: During the incorporation of machine learning approaches, it is the responsibility of the implementers to record and report the model performance metrics appropriately. In case of any estimations or imbalances, a report of

the accuracy has to be provided to avoid misleading the audience about the performance of the model.

2.5.2. Ensuring Privacy and Security of Data Used in the Thesis

In order to ensure the privacy and security of the data used in this thesis, a few measures were put in place. One of the measures that were employed was encoding. Changing, some sensitive information into codes that only the machine understands and therefore helps with maintaining the privacy of data. In regard to the security of the data used as indicated above the data used was sourced from an open website and therefore accessibility is easy and free for all users.

3. REVIEW OF THE LITERATURE

3.1. Introduction

The literature consists of the literature which assists in understanding the prediction of cases and rates of COVID-19.

3.2. Long Short-Term Memory-Attention (LSTM-Attention) in COVID-19 prediction

Long Short-Term Memory-Attention (LSTM-Attention) brings together two robust deep learning techniques, Long Short-Term Memory (LSTM) and Attention techniques, whose applications have been practical in many areas, including the projections of COVID-19 (Kubat, 20210. Long Short-Term Memory (LSTM) is a recurrent neural network (RNN) that manages sequential information. It is essential in obtaining long-term dependencies in time series information, making it appropriate in series projection functions, including projections of COVID-19 cases. LSTM can remember and store data over a long duration, enabling it to master trends and patterns within the data.

On the other hand, the attention mechanism is a deep learning model that originally functioned in natural language processing, even though its other functions have evolved (Guyet et al., 2023). The attention mechanism allows the model to concentrate on the vital aspects of input data during projections. That is, it equates different weights to different elements of the input information while availing essential futures and time steps and pressing the model to focus on the same during projections.

COVID-19 predictions revolve around projecting the total number of cases and the rates at which viruses are transmitted. In addition, it entails projecting all the relevant metrics

LSTM-Attention has been utilized in the COVID-19 projections to obtain temporal dependencies while concentrating on relevant data within the time series information (Guyet et al., 2023). The LSTM component within the combination enables capturing the underlying patterns within the sequential COVID-19 data, including the daily reported cases, hospitalization data, and other essential information (Guyet et al., 2023). It can further obtain previous information stages while capturing and mastering the involved trends.

On the other hand, the combination's attention component enhances the LSTM's ability to predict the movement by pushing the model to weigh the importance of different steps within the arrangement. The attention mechanism further identifies the most relevant features and time in the process that can be utilized during the current projections and assigns them higher weights; this enables the marking of the vital information in the sequence. LSTM-A, therefore, captures long-term dependencies within the COVID-19 time series while revealing the essential features of each prediction (Zou, 2022). The combination thus enhances accuracy and strength within the COVID-19 forecasts and the projections of the transmission rate. Public health officials and other policymakers utilize such information in handling the pandemic. Appropriate evaluation metrics and validation mechanisms are further needed to prove the combination's validity, reliability, and generalizability based on their application to real-world scenarios.

3.3. XGBoost in COVID-19 Prediction

Extreme Gradient Boosting (XGBoost) is a standard and robust machine learning algorithm applied in various data science and predictive modelling tasks, including the projections of COVID-19. It is an ensemble learning method that combines the predictions

of weak learners, such as decision trees, to build a robust predictive model (Wade, 2020). Its features include high predictive accuracy, especially while handling high-dimensional and complex information. It can obtain nonlinear connections and other interactions between components, which helps understand the dynamics of the COVID-19 pandemic. Managing missing information is a feature based on its ability to appropriately handle missing information, eradicating the need for data imputation while ensuring predictions are vital regardless of the minus points (Wade, 2020). It is associated with feature importance, where it avails measure for feature importance while revealing the features essential in impacting the projections. This data is necessary for feature selection and understanding the critical factors facilitating COVID-19 cases and transmission rates. Regularization is its feature that prevents overlifting while ensuring the ability of the model to generalize new information (Wade, 2020) correctly. Finally, it is associated with scalability, the ability to manage large datasets, and its suitability to study COVID-19 prevalence involving voluminous data (Kumar, 2019).

XGBoost is applicable in various areas, such as case predictions, where it is used to predict COVID-19 cases in different regions based on the history of the case data and other vital features. It is further applicable in transmission rate forecasting, where it helps in projecting the rate of the virus while providing insights into the potential growth or reducing the outbreak (Siahaan & Sianipar, 2022). In addition, it is applicable in severity projections while providing information related to the prevalence of COVID-19, enabling the healthcare system to allocate resources based on the projected severity. Finally, it is used in intervention impact analysis, which evaluates the impacts of various interventions, such as lockdown measures and vaccination campaigns, on the spread of COVID-19.

XGBoost is associated with various challenges and considerations. Data quality is one of the challenges related to its utilization in COVID-19 projections (Wade, 2020). The accuracy depends on the quality and reliability of the information utilized in training the model. Reliable predictions demand accurate and up-to-date information. Interpretability is another challenge whereby the model is complex and hence needs more interpretability, unlike other models such as linear regression. In critical applications, effort is demanded to interpret the model.

Furthermore, evaluating the model is complex, where the prediction is only reliable when appropriate metrics and validation mechanisms are utilized (Sharma, 2018). Generally, XGBoost is a valuable tool in projections of COVID-19 based on its ability to provide accurate and robust predictions using historical data and other relevant features. Healthcare systems can use their applications and forecasts to make appropriate decisions.

3.4. Data sources and pre-processing techniques in Python

The three algorithms used include; Decision Tree Classification, Logistic Regression, and Random forest Classification. The COVID-19 pandemic has raised several challenges to understanding and predicting virus transmission dynamics and case outcomes. However, machine learning techniques have become valuable tools in analysing complex data and predicting numerous domains, such as public health. Various models in machine learning, such as decision tree classification, logistic regression, and random forest classification, have been widely explored in the prediction and transmission rates of COVID-19.

3.4.1 Decision Tree Classification

The COVID-19 pandemic has reiterated the need for models that accurately predict and forecast the spread of the virus. As such, decision tree classification, a machine learning technique, has been cited in various literature as a valuable tool for predicting COVID-19 cases and transmission rates. Before the advance of COVID-19, a study by Potash et al. (2015) revealed decision tree classifiers' ability to predict childhood lead poisoning due to the model's interpretability and ability to deal with complex interactions. Although the study was not specific to COVID-19, it revealed the model's ability to predict various tasks in public health. Thus, in its adoption to the COVID-19 context, Karako & Song (2021) suggest using decision tree classifiers for their ability to offer interpretability, prediction, and accuracy in identifying rates of disease transmission and high-risk areas. The model's examination of the splits and features provides insights into the factors influencing COVID-19 transmission.

Equally, a study by Islam et al. (2021) exploring the various machine learning models revealed decision tree classification ability to provide valuable insight into the spread of the virus. The study evaluated multiple features like previous data on infection rates and demographic data to compare the performance of various machine learning models. The study revealed that decision tree classifiers could accurately predict COVID-19 propagation. Besides the ability to analyse demographic data, the decision tree model analyses chest X-ray images to help distinguish COVID-19 cases from other respiratory conditions, further leading to accurate early diagnosis (Yoo et al., 2020). Ghoshal & Tucker (2020) further thesis the interpretability of decision tree classification in COVID-19 detection. The study assembles a model that combines the strengths of deep learning

with a decision tree to reveal decision tree classifiers' contribution to the interpretability of the model. The decision tree classification gave insights into the components facilitating the prediction of COVID-19. However, according to Allam & Jones (2020), decision tree classification accuracy prediction lies in its integration with other machine learning techniques. Integrating decision tree classifiers with other machine learning approaches enhances model performance and provides insights for effective COVID-19 monitoring and management.

3.4.2. Logistic Regression

As a statistical modelling technique for predicting binary outcomes, the use of logistic regression in predicting COVID-19 cases has been cited in various studies. Logistic regression estimates the probability of an individual getting infected based on a particular set of predictor variables, such as socioeconomic conditions and demographic attributes, as seen in the study by Ansari & Baker (202I). In the study, the authors illustrate how logistic regression and other machine learning models effectively predict the risk of COVID-19 infection based on demographic and socioeconomic conditions. As such, logistic regression helps understand the link between the risk of COVID-19 infection and predictors. Moving away from socioeconomic and demographic factors, Raza et al. (2021) use logistics regression to uncover the predictive function of environmental and economic factors in COVID-19 outcomes in the US. The study illuminates the importance and interpretability of logistic regression in identifying essential predictors linked with COVID-19 outcomes.

Zhou et al. (2020) also advance logistic regression to predict COVID-19 death in recovered, vulnerable, and infected individuals. Comparing various machine learning

techniques, the study findings reveal that logistic regression offers accurate predictions for COVID-19 outcomes. Logistic regression helps identify key predictors and estimate infection probabilities in different groups. To understand COVID-19 transmission, Almeshal et al., 2020 apply logistic regression-based machine learning to predict COVID-19 transmission in India. While the study considers various factors such as healthcare infrastructure and population density, logistic regression shows its ability to accurately forecast the likelihood of infection transmission across various regions in India. Thus, logistic regression offers insights into the crucial factors influencing the transmission rate. Ye et al. (2021) also suggest this is true, although they emphasize the need to combine logistic regression with other machine learning models for a more effective prediction of the levels of risk of transmission. Thus, logistic regression offers estimation probabilities, identification of significant predictors, and provision of accurate forecasts for binary outcomes related to COVID-19.

3.4.3. Random Forest Classification

Various studies have used random forest classification to predict COVID-19 cases and transmission rates. It is particularly suitable for COVID-19 prediction because it captures non-linear patterns and deals with complex relationships (Alali et al., 2022). The thesis by Cheng et al. (2020) explores using random forest classification to forecast COVID-19 cases and ICU risk through laboratory and demographic results. The study illustrated the capability of random forest classification in forecasting COVID-19 cases and identifying individuals susceptible to ICU. The predictability of COVID-19 cases and transmission rates was also evident in a study by Tian et al. (2020) on applying random forest classification in forecasting the COVID-19 pandemic. Based on its argument on climatic

conditions, socioeconomic indicators, and demographic data, the study demonstrated random forest classification effectiveness in accurately predicting COVID-19 cases by highlighting the key predictors leading to transmission rates. Equally, random forest classification with the help of appropriate features yields an accurate prediction of the mortality rate in COVID-19 cases (Wang et al., 2020). This is because the model helps identify key relevant predictors of mortality. Notably, Chandra et al. (2022) suggest that a hybrid approach offers a more accurate prediction since it captures the dynamics of the pandemic's mortality prediction. The study combines random forest classification with long short-term memory (LSTM) to uncover the prediction of the COVID-19 pandemic in India. From the results, random forest classification effectively forecasted COVID-19 cases and transmission rates in conjunction with LSTM. Thus, random forest classifiers can handle complex relationships, feature importance analysis, and robust prediction capabilities.

3.4.4. Data Collection and Pre-processing

In this section, the data collection method will be described to help in understanding the development of this research paper. By analyzing the methods of data collection, as well as the design of the research, it is easier to also understand how all the variables used in the research are connected to generate the findings.

3.4.4.1. Description of Data Sources and Datasets Obtained

The data that was used in this research was obtained from Kaggle. Kaggle is an online website that is freely accessible and contains multiple datasets. Some of the sources that were used are shown in figure 1 below. The two main datasets that are available on kaggle are training data and results data. If these two data sets have the same dimensions, the

model employed has the potential of producing accurate results. The website was preferred for this research because of its flexibility. The datasets were easy to import, analyse and visualize and the amount of time used to analyse the data collected is minimal. All the data sources that were used for this research were reliable sources that had been peer-reviewed data that is free from bias and considered accurate. Since the aim of this research is to make prediction, it was necessary to only select sources with accurate data to ensure effective prediction of the research data. The first process in the pre-processing was to check the datasets license. Since Kaggle has open-sourced datasets, it was necessary to check all the licenses of the datasets that were used in this research to ensure that they were allowed for academic and commercial publications. In order to guarantee that the datasets selected to be used in this research are reliable, review method was used. By reviewing all the notebooks shared using the dataset selected for use it was easier to identify all the reliable datasets that can be used for efficient research.

3.5. Data Pre-Processing using Python (data cleaning, normalization, feature engineering)

The data preprocessing technique is a procedure that has only recently been adopted for training machine-learning models. Data preprocessing also known as the data mining process is an essential component of data preparation that is used to define any type of processing that is done on raw data to prepare it for further preparation procedures. During the preprocessing procedure, data is transformed into a format that is easily processed in machine learning. The preprocessing procedure of the Python model was employed at the earliest stage of the machine-learning process. This was done to ensure accurate predictions. Data pre-processing is made easier using different tools such as sampling (Selection of the representative subject), denoising (Removal of noise from selected data),

imputation (Synthesis of statistical relevant data for missing values), normalization (data organization for easy access) and feature extraction (extraction of relevant feature subset useful in the current context). In addition, feature extraction (extraction of relevant feature subset useful in the current context) In addition, feature extraction (extraction of relevant feature subset useful in the current context). During the data pre-processing process in this research, there were a number of steps that were incorporated to guarantee accuracy. These included:

3.5.1. Data Profiling

During this process all, the available data on the selected datasets was examined, analyzed, and reviewed. As mentioned earlier, the selected website for this research has multiple data sources, which means that it was necessary for a survey to be conducted on the existing data and characterization employed to help narrow down the data sources. To achieve this, it was necessary to only select the datasets that were pertinent to the research problem, which is COVID-19.

3.5.2. Data Cleansing

This was the most difficult step, as it involved rectifying all the quality issues identified in the existing data. To rectify the quality of this data, some processes were employed which included eradicating all the data that were considered irrelevant for this research and also filling in data that was otherwise missing on the retrieved datasets.

3.5.3. Data Reduction

Data reduction is completely different from data cleansing. During the process of cleaning data, we used information from other sources to fill in the missing data, it is important to note that the existing raw data was left as it was in the previous step. Therefore, in this process, the task of reducing data by eradicating all redundant and irrelevant data was done. During data entry, most of the raw data collected might be repeated severally due to multiple factors, therefore it might be impossible to obtain already clean and efficient data. However, since this research aims to make use of accurate data, it is necessary to remove all data that might alter the process of prediction.

3.5.4. Data Transformation

This research has two distinct objectives, in order to achieve the objectives, the data collected in this process has to be categorized to ensure the information obtained has the capability of achieving the research objectives. Therefore, during this step, the data selected and cleaned was categorically organized and structured. All the ranges that the thesis aimed to focus on were also selected. The research focused on predicting COVID-19 cases and the transmission rate and therefore all variables were combined to guarantee the achievement of these objectives.

3.5.5. Data Enrichment

In this step, different feature engineering libraries were applied to the transformed data to prompt the desired transformations that were otherwise not achieved in the previous stage. A dataset was created that was more organized compared to how it was in the raw data state.

3.5.6 Data Validation

This was the last step of data preprocessing. During this stage, all the data that was retrieved was divided into two sets. The first set of data was used in the training of the Python model. The second set of data was used to test the accuracy of the resulting model. After the testing process proved the accuracy of the model used, that data was validated for engineering.

3.5.7 Handling Missing Data and Outliers in Python

Since Kaggle is an open website, despite data being collected there are many reasons that can lead to potential loss of data, mainly due to data corruption. Outliers on the other hand are some data points available on the datasets that are irrelevant to the general trend of the rest of the data. The reason outliers can be found in Python is due to incorrect entries or errors during measurement. In order to achieve accurate results, it was essential to handle all the missing data and outliers in the machine-learning process. From the definition of the outliers and missing data, it is evident that these two aspects of data can significantly impact the outcomes of the results in machine learning. However, with the proper approach to handling them, it is easier to manage and control their effects on the final results. The approach that this research employed to handle the outliers and the missing data was to completely remove all the outliers and also remove all the rows that contained missing data. This is because this thesis was completely dependent on datasets retrieved from Kaggle as the source of data. Therefore, with missing data, it was impossible to use information from other sources as it was impossible to find data that fit the research focus. Before the removal of the rows containing missing data, it was imperative to first identify the main cause of the missing data to ensure it does not implicate the remaining data. In

machine learning, missing data implicates the results as it causes inaccuracy since there is missing information that the machine will be unable to learn from. The imputation approach was employed on features that could not be completely removed. Imputation involves creating estimates of the missing data and feeding it to the algorithms. To minimize the chances of inaccuracy, imputation was only employed on special features that were necessary for machine learning to be effective.

Outliers on the other hand cause an overfit on the model and therefore skew the results, which is why understanding the initial issues and handling them was essential in mitigating these issues. One technique that is important to use when employing the outlier removal approach is training since the variables to be used were already outlined at the beginning of the model. Removing some of the rows that were considered unfit had the potential to affect the Python model, which was why it was necessary to train the model to adapt and deal only with the available data. For essential information that could not be easily discarded, data augmentation was employed. In this procedure, new data that was similar to the missing data was created because this technique is time-consuming, and its accuracy level is low. Only essential data that could not be completely destroyed was augmented, while the remaining missing data was simply deleted. It is important to understand that the reason the research chose data augmentation of important data was to avoid issues of bias.

3.6 Geographical and population data related to Mexico

Since the data set, we will use from Kaggle refers to real data provided by the Mexican government, it is necessary to make a reference to the country's geographic and population data.

The population of Mexico, which exceeds 128 million people, is concentrated in a narrow geographical area that represents only 18% of the country's total land. The population density in this strip is five times higher than in the rest of the country. This population distribution pattern has remained stable for centuries, as a census conducted 130 years ago showed a statistically similar distribution. In contrast, the population distribution in the United States has expanded and changed dramatically over the last 130 years.

The reason for the concentration of the population in Mexico is related to the country's geography. In the central part of Mexico runs the temperate climate zone of the subtropical mountains, providing ideal conditions for agriculture and settlement. The dry deserts in the north and the dense tropical forests in the south have historically limited population growth in those areas. This geographic constraint has led to a pattern of concentrated populations in Mexico.

Mexico's particular population distribution due to its geographic characteristics makes the country particularly vulnerable to highly contagious infections such as COVID-19. Since the majority of the population lives isolated on a narrow strip of land. For this reason, Mexico has become an ideal country for our study.

PART TWO

4. THE PROGRAM

4.1 Machine Learning Models for Transmission Rate Prediction

In this section, the datasets are analyzed and investigated using data visualization methods. The main purpose of this section is to help with the identification of potential errors as well as a better understanding of the patterns available in the existing data. Exploratory data analysis is also important as it helps to identify any outliers and also establish associations among the variables used. Python is identified as one of the most commonly used tools in exploratory data analysis. The reason Python is a preferred tool is that it is high-level; it has combined dynamic typing and binding and also has a built-in data structure that makes it applicable in making predictions and identifying missing data.

4.2 Visualizations and insights gained from the data using Python libraries.

Data visualization is essential in the analysis of tasks and includes the summarization of data, exploration of the data, and then output model analysis. It is the easiest way of communicating the findings. Python has many features that are useful for insight into data.

The first step in the visualization of data, the first step is to import to the libraries, this can be included as follows:

import pandas as pd

import numpy as np

import sklearn

import matplotlib.pyplot as plt

import seaborn as snss

```
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt

from sklearn.preprocessing import OneHotEncoder, MinMaxScaler, LabelEncoder
from sklearn.compose import ColumnTransformer

from sklearn.model_selection import cross_val_score

from sklearn.tree import DecisionTreeClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import LinearSVC

from sklearn import tree

from sklearn.cluster import KMeans
```

The second phase is to import the datasets that are involved in the work, and they were downloaded from Kaggle. The dataset is a time series dataset on the account of the dead, the confirmed and the recovered in different regions of the world.

```
df = pd.read_csv('covid.csv')#read the data
df.head()#show the first 5 rows of the data
```

The dataset is cleaned before it is plotted to a firm graph from those who were neither male nor female. The data also considered other factors such as age, pregnancy, and hypertension.

Out[178]:

	id	sex	patient_type	entry_date	date_symptoms	date_died	intubed	pneumonia	age	pregnancy	...	inmsupr	hypertension	other_disease	c
0	16169f	2	1	04-05-2020	02-05-2020	9999-99-99	97	2	27	97	...	2	2	2	
1	1009bf	2	1	19-03-2020	17-03-2020	9999-99-99	97	2	24	97	...	2	2	2	
2	167386	1	2	06-04-2020	01-04-2020	9999-99-99	2	2	54	2	...	2	2	2	
3	0b5948	2	2	17-04-2020	10-04-2020	9999-99-99	2	1	30	97	...	2	2	2	
4	0d01b5	1	2	13-04-2020	13-04-2020	22-04-2020	2	2	60	2	...	2	1	2	

5 rows × 23 columns

The data in different column are as presented in the table below, each was given a unique identity number for identification.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 566602 entries, 0 to 566601
Data columns (total 23 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   id                   566602 non-null  object
 1   sex                  566602 non-null  int64
 2   patient_type         566602 non-null  int64
 3   entry_date           566602 non-null  object
 4   date_symptoms        566602 non-null  object
 5   date_died            566602 non-null  object
 6   intubed              566602 non-null  int64
 7   pneumonia            566602 non-null  int64
 8   age                  566602 non-null  int64
 9   pregnancy            566602 non-null  int64
 10  diabetes             566602 non-null  int64
 11  copd                 566602 non-null  int64
 12  asthma               566602 non-null  int64
 13  inmsupr              566602 non-null  int64
 14  hypertension         566602 non-null  int64
 15  other_disease        566602 non-null  int64
 16  cardiovascular       566602 non-null  int64
 17  obesity              566602 non-null  int64
 18  renal_chronic        566602 non-null  int64
 19  tobacco              566602 non-null  int64
 20  contact_other_covid  566602 non-null  int64
 21  covid_res            566602 non-null  int64
 22  icu                  566602 non-null  int64
dtypes: int64(19), object(4)
memory usage: 99.4+ MB
```

To ensure that the data is well visualized, then dummy column was added for adding up the data. It is used to convert categorical columns which are at the column level into indicator columns.

```
df['count'] = 1#
```

The data was then converted to a proper type in the three-date column

```
df['entry_date'] = pd.to_datetime(df.entry_date, format='%d-%m-%Y')
df['date_symptoms'] = pd.to_datetime(df.date_symptoms, format='%d-%m-%Y')
```

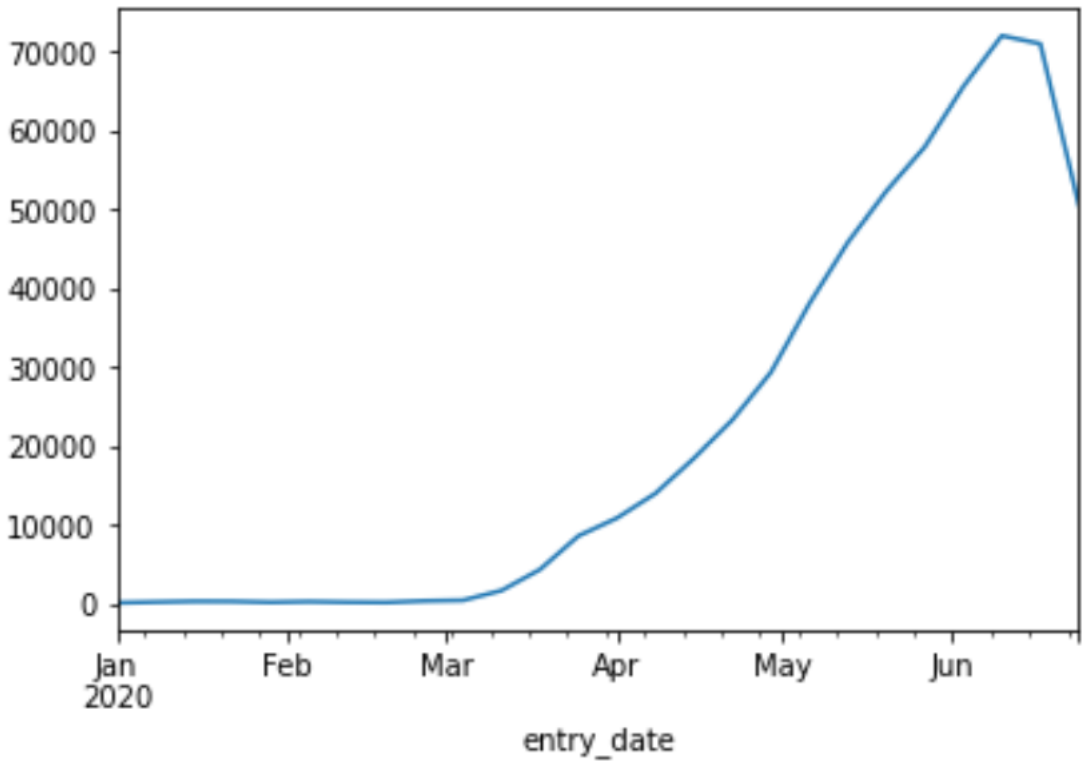
The statistical overview of the data was then derived as shown below.

	sex	patient_type	intubed	pneumonia	age	pregnancy	diabetes	copd	asthma	in
count	566602.000000	566602.000000	566602.000000	566602.000000	566602.000000	566602.000000	566602.000000	566602.000000	566602.000000	566602.(
mean	1.506726	1.215165	76.562952	1.846262	42.622483	50.400692	2.210633	2.280221	2.265029	2.:
std	0.499955	0.410937	39.058676	0.560939	16.659973	47.501579	5.683523	5.327832	5.334658	5.(
min	1.000000	1.000000	1.000000	1.000000	0.000000	1.000000	1.000000	1.000000	1.000000	1.(
25%	1.000000	1.000000	97.000000	2.000000	31.000000	2.000000	2.000000	2.000000	2.000000	2.(
50%	2.000000	1.000000	97.000000	2.000000	41.000000	97.000000	2.000000	2.000000	2.000000	2.(
75%	2.000000	1.000000	97.000000	2.000000	53.000000	97.000000	2.000000	2.000000	2.000000	2.(
max	2.000000	2.000000	99.000000	99.000000	120.000000	98.000000	98.000000	98.000000	98.000000	98.(

The line graph below has been used to determine the number of weekly admissions from COVID-19.

```
df.resample('7D', on='entry_date')['count'].sum().plot.line()
```

Out[186]: <AxesSubplot:xlabel='entry_date'>



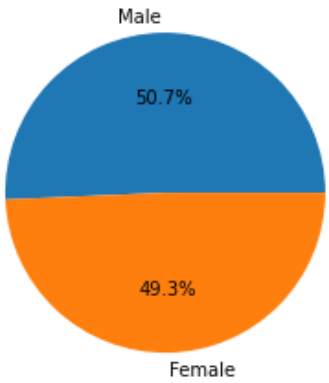
From the line graph above, the rate of admission was low from January 2020 to the first week of March 2020, from the analysis; there is an exponential increase from the second week of March, which reached over 60,000 admissions per week at the peak which was during the first week of June. From then, there is a slight decrease for one week in June followed by a sharp decline in the number of admissions.

Data visualization was conducted using the gender feature. Gender was indicated using the sex variable. According to the data analyzed from the Python libraries, it was deduced that transmission was more common among males compared to females. However, the difference indicated in terms of gender was very minimal, with the males reporting 50.7% of cases while female reports indicated only 49.3%. Matplotlib offers tools that can be used for data visualization, when analyzing the data set, in this case, there is analysis of values. Pie charts are effective when it comes to visualizing the proportions of data. In this, there are data sets, and a pie chart can be used when visualizing the proportion of genders that are infected with COVID-19. A pie chart is used for visualizing the proportion of males and females. To do this, there is a creation of new columns containing Male and Female.

```
#visualize the distribution of the data by gender.

gender = df.sex.value_counts().to_list()
labels = ['Male', 'Female']

fig, ax = plt.subplots()

ax.pie(gender, labels=labels, autopct='%1.1f%%')
plt.show()
```

Visualization by gender

```
#visualize the distribution of the data by gender.
gender = df.sex.value_counts().to_list()
labels = ['Male', 'Female']

fig, ax = plt.subplots()
ax.pie(gender, labels=labels, autopct='%1.1f%%')
plt.show()

```



The average duration between when symptoms are first detected and entry date

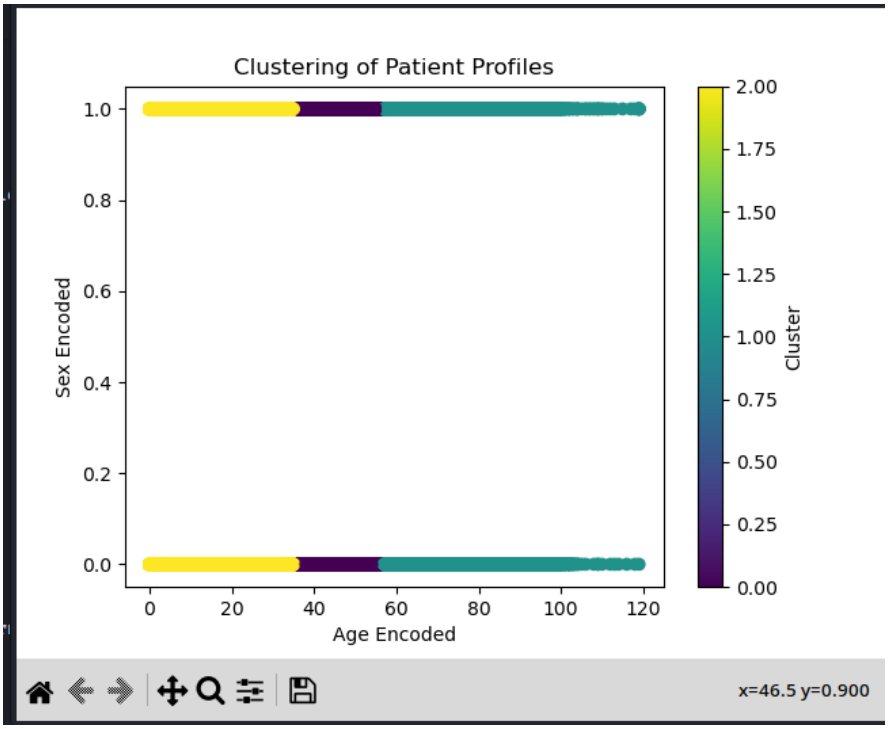
Based on the graphical presentation of this data, it was discovered that approximately 90% of patients had their symptoms immediately discovered between April and June. At the beginning of 2020, very, few people showed symptoms when COVID-19 was first detected, recording 38% of patients with symptoms in January and gradually increasing to 68% in March.

Correlation analysis between variables and target (cases and transmission rate)

Some of the correlations that were established included:

Gender and the survival rate: men have a higher survival rate compared to women.

Age and the survival rate. The infants and the aging population have a lower survival rate compared to the teenagers, youths and the middle-aged persons as shown below.




```
#we group the data by ages, into 12
df2['age_group'] = pd.cut(df2.age, bins=12, right=False)#, labels=list('123456'))

#we then see how the survival rate was for each agegroup


pvt = df2[['age_group', 'survival_status', 'count']].pivot_table(values='count', ind

print('\n\nsurvival against age group'
print(pvt.div(pvt.sum(axis=1), axis=0))
pvt.div(pvt.sum(axis=1), axis=0).plot.line()
```

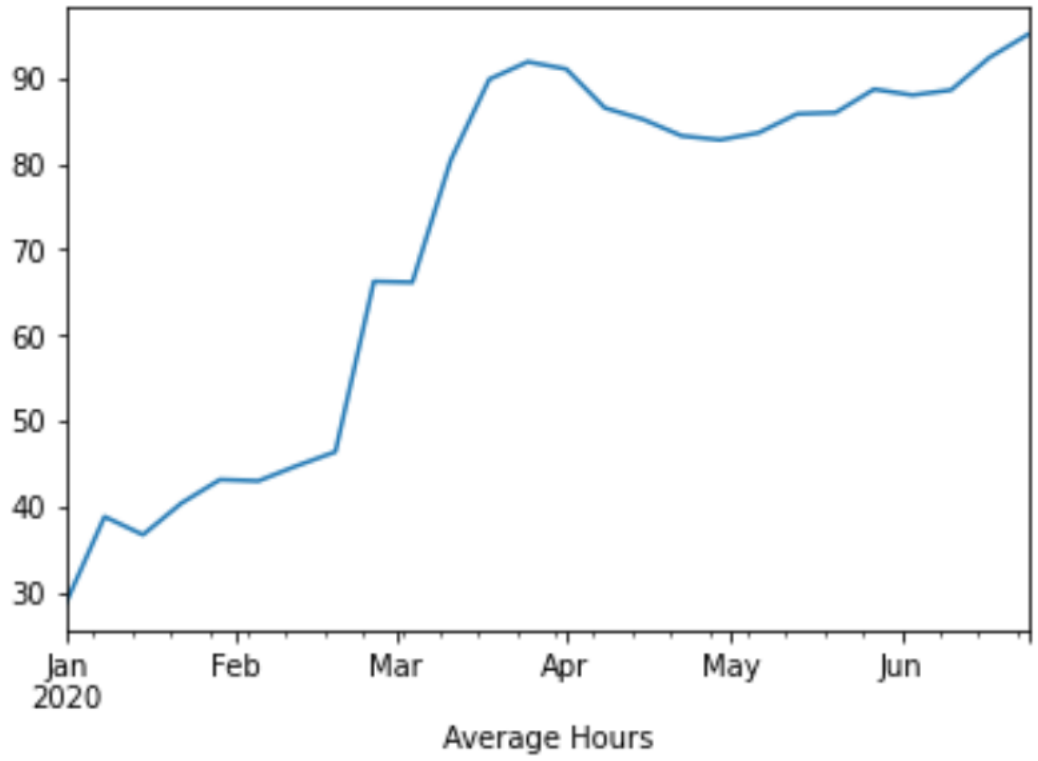
Time and symptoms detection: Patients are prone to visit the hospitals faster after first symptoms with increased tension.

first symptoms with increased tension.

The figure indicates that in the time recorded transmissions increases spontaneously between April and June with peak entry of patient transmissions being in June before a gradual decline.

The other visualization was done was the average duration which existed between the date of entry and first detection.

```
#plot the average duration between when symptoms are first detected and entry date
df['duration'] = (df['entry_date'] - df['date_symptoms']).dt.total_seconds()/3600
ax = df.resample('7D', on='entry_date')['duration'].mean().plot.line(xlabel='Average Hours')
```



From the line graph below, it becomes clear that the duration of the appearance related to the first symptom and the visit to the hospital increased with time for the period under review. During the first week of January 2020, the rates were as low as 30 percent of the patients, but with the creation of awareness of COVID-19, the rates increased to over 90% in the last week of March. There was a slight drop to 80% during April, and it also increased in the last week of May and June. Since the time under review was from January 2020 to June 2020.

4.2.1 Identification of key features and patterns in Python

Key features that were identified to be used in this research included sex, patient type, entry date, symptoms, date of death, intubation, and pneumonia. Some other variables that were used included pregnancy, diabetes, obesity, etc. as indicated in the figure below. All these variables were identified after a close association was established, it was discovered that patients with underlying medical conditions were more prone to be affected compared to people who were healthy. In terms of age, it was discovered that younger and older people were at a higher risk of transmission compared to teenagers, youths, and middle-aged persons.

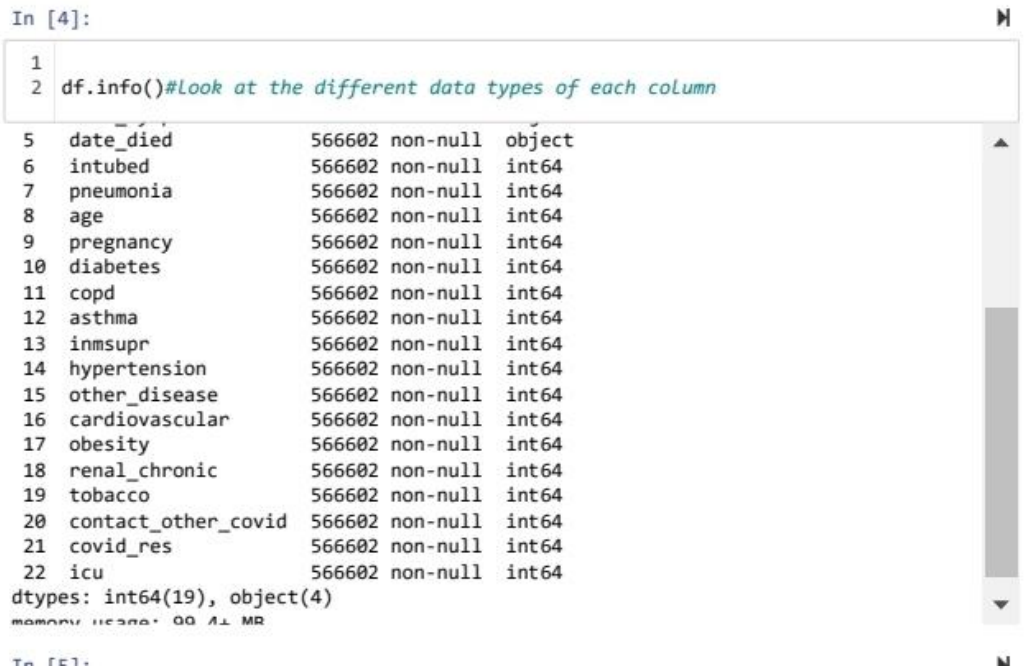


Figure: different variables used in data analysis.

4.2.2 Feature Engineering and Selection

The main role of machine learning algorithms is to process multiple data and generate meaningful patterns that can be used in making predictions and in the decision-making process. Features are used to describe the algorithm's input variables that are used to make predictions. For instance, in this research some of the features that were used to make predictions include age, sex, country, etc.

4.2.3 Selection of relevant features for prediction

Feature selection is an important aspect of Python. There are different techniques that are employed in feature selection. One of the techniques used is reduction. This technique was employed in this research with utmost care not to alter the original representations of the variables used. During the feature selection process of this model, a subset was selected

that was used to train machine-learning algorithms. The subsets that were used are shown in figure 3 below. The overall features were reduced to only 5 rows and 23 columns to maximize the accuracy of the model.

	id	sex	patient_type	entry_date	date_symptoms	date_died	intubed	pneumonia	a
0	16169f	2	1	04-05-2020	02-05-2020	9999-99-99	97	2	
1	1009bf	2	1	19-03-2020	17-03-2020	9999-99-99	97	2	
2	167386	1	2	06-04-2020	01-04-2020	9999-99-99	2	2	
3	0b5948	2	2	17-04-2020	10-04-2020	9999-99-99	2	1	
4	0d01b5	1	2	13-04-2020	13-04-2020	22-04-2020	2	2	

5 rows × 23 columns

Figure 3:*The features used in the research.*

4.2.4 Feature engineering techniques applied in Python

Feature engineering is an essential step in this model. This is because it allows research to extract meaningful information from raw data that can be useful in the enhancement of machine learning models, hence allowing for accurate predictions. Feature engineering therefore involves the process of transforming raw data into meaningful features that can be entered into machine learning algorithms. The process of feature engineering includes generating new features to capture hidden patterns, scaling, cleaning, and encoding data. There are three essential techniques that were used during feature engineering, this includes, domain-knowledge-based feature generation and categorical encoding.

4.2.4.1 Domain-Knowledge-based feature generation

In order to obtain accurate results and make effective and accurate predictions, it is necessary to employ this feature in machine learning. The domain-specific knowledge feature is a technique that allows for the generation of a new feature to provide more precise insights into data. For instance, during the entry of data that involves features an example used in the thesis when determining the transmission rate based on individual health, the BMI of individuals was generated using the entry of height and weight as recorded.

4.2.4.2 Categorical Encoding

It is evident that during machine learning, numerical data is more useful and makes predictions easier. Therefore, certain features need to be converted into numbers to allow for accuracy. For instance, when feeding the machine learning algorithm with features such as the colour of the individuals or the countries, numerical data can be generated to act as codes for these features. In the figure below, an illustration of how encoding was used in this research is made. The use of numbers to represent certain features was used in order to facilitate effective results. The categorical encoding technique is indicated below.

```
#add a dummy column for adding up the data

2 df['count'] = 1#

3

4   #we need to convert the three date columns into the proper type

5   df['entry_date'] = pd.to_datetime(df.entry_date, format='%d-%m-%Y')

6   df['date_symptoms'] = pd.to_datetime(df.date_symptoms, format='%d-%m-%Y') 7
```

```
8 # the dates with 9999-99-99 indicate that the patient was alive as at the time the d 9 #
was collected, so, we set it as 'NaT' in the line below.
10 df['date_died'] = pd.to_datetime(df.date_died, format='%d-%m-%Y', errors='coerce')
11 df.sort_values(by='entry_date', inplace=True) 12
```

4.2.4.3 Data reduction

Data predictions and transmission rates of COVID-19 have multiple variables. However, not all the variables that other scholars have included in their sources are useful for this thesis. This means that it is necessary to generate a new analytic model that can help with the reduction of all the irrelevant variables. There are variables that can be combined to create a single variable that is relevant and reduces the number of dimensions in the training data set, hence ensuring efficiency and accuracy.

Feature Scaling and normalization

In most cases, multiple variables change over different scales. In other cases, one scale will change linearly while the other changes exponentially. One of the examples of feature scaling used in this research is age as well as the other columns with medical information as indicated below. By scaling this feature, it was easier for algorithms to tease apart the meaningful associations between the selected variables to generate accurate predictions.

```
#we encode the data in preparation for modeling

#the columns to use onehot encoding
oh_columns= ['pneumonia', 'intubed', 'icu', 'contact_other_covid',
               'cardiovascular', 'tobacco', 'renal_chronic', 'obesity', 'pregnancy
               'diabetes', 'copd', 'asthma', 'inmsupr', 'hypertension', 'other_dise
            'sex', 'patient_type']

#the columns to use minmax scaling
minmax_columns= ['age']

#use column transformer
ct = ColumnTransformer(
     [("oh_columns", OneHotEncoder(), oh_columns),
      ("min_max", MinMaxScaler(), minmax_columns)])

X = ct.fit_transform(df2)


#our target variable is covid_res
enc = LabelEncoder()
y = enc.fit_transform(df2.covid_res)

```

4.2.5 Feature selection methods used in Python (Such as Recursive Feature Elimination, Feature Importance)

After understanding the aspect of feature selection and why it is important in machine learning, it is necessary to also understand some of the techniques that can be used in feature selection.

4.2.5.1 Recursive feature elimination

This is a feature selection technique that helps with the identification of the dataset's key features. This technique was used as shown below. The whole process involves creating a model using the remaining features after eliminating some features that are considered irrelevant to obtain the desired number of features. After feature elimination in this thesis, only 5 rows and 28 columns were left, a number that was manageable for the model. All the selected features were ranked using an RFE machine-learning algorithm

[6]:

```
#drop the id column
df.drop(columns='id', inplace=True)

#get a statistical overview of the data
df.describe()
```

Out[6]:

4.2.5.2 Feature importance

Feature importance is a technique that involves selecting scores that can be used to establish the relative importance of the features in a dataset. This is done in order to develop a predictive model. Some tools used in calculating the scores used in feature importance include linear models, decision trees, neural networks, and random forests.

4.2.5.3 Correlation analysis

A correlational analysis involves the statistical summarization of different relationships between the selected two variables. Correlation analysis is considered to be the core of exploratory analysis. In this research, a correlational analysis was conducted between variables such as age and survival rate, health condition and survival rate, and time of entry and symptom occurrence.

4.3 Machine Learning Models for Cases Prediction

In this section, different machine-learning models will be analyzed and compared to help understand the accuracy and effectiveness of Python as the selected machine-learning model.

4.3.1 Description and implementation of regression algorithms in Python

4.3.1.1 Linear Regression

When the goal and one or more predictors have a linear relationship, linear regression models can predict a continuous target.

4.3.1.2 Simple linear regression

The variable to be predicted depends on just one other variable in this most straightforward type of linear regression. This is calculated using the formula often used to determine a line's slope.

$y = w_0 + w_1 * x_1$

The target variable in the previous equation is denoted by y, while the independent variable is denoted by x1. The coefficient w1, commonly referred to as the slope, is used to describe the connection that exists between y and x1. The constant coefficient, or intercept, is denoted by w0. In relation to the independent variables, it alludes to the fixed offset that y will always have.

4.3.1.3 Multiple Linear Regressions

An extension of simple linear regression is multiple linear regression. The goal value in this configuration is dependent on multiple variables. The use case determines how many

variables there will be. A subject-matter expert is typically involved in selecting the fields that will help improve the output feature prediction.

$y = w_0 + w_1 x_1 + w_2 x_2 + ... + w_n * x_n$

4.3.1.4 Random Forest Regression

Decision trees are typically regarded as weak models because, when the data set is very large, their performance frequently falls short of expectations. However, they offer more accuracy when multiple decision trees are merged into a single model. A portion of the training data is used to construct each decision tree in the random forest. A random number can be chosen to see changes in accuracy in the number of decision trees that make up this random forest. This model's output represents a value to be anticipated as the average of the values obtained from each of these separate trees.

4.3.1.4.1 Model selection and evaluation using Python (metrics like Mean Squared Error, R-squared)

4.3.1.4.1.1 Mean square error

The average of the error squares is measured by the mean squared error. The average of the sums of the squares of each discrepancy between the estimated value and the true value is what this indicates. Even if all forecasts are exact, the MSE is never zero and is always positive. It takes into account the estimator's bias, which is the difference between the estimated values and their true values, as well as the variance of the estimator which explains how dispersed the estimates are.

4.3.1.4.1.2 R-squared

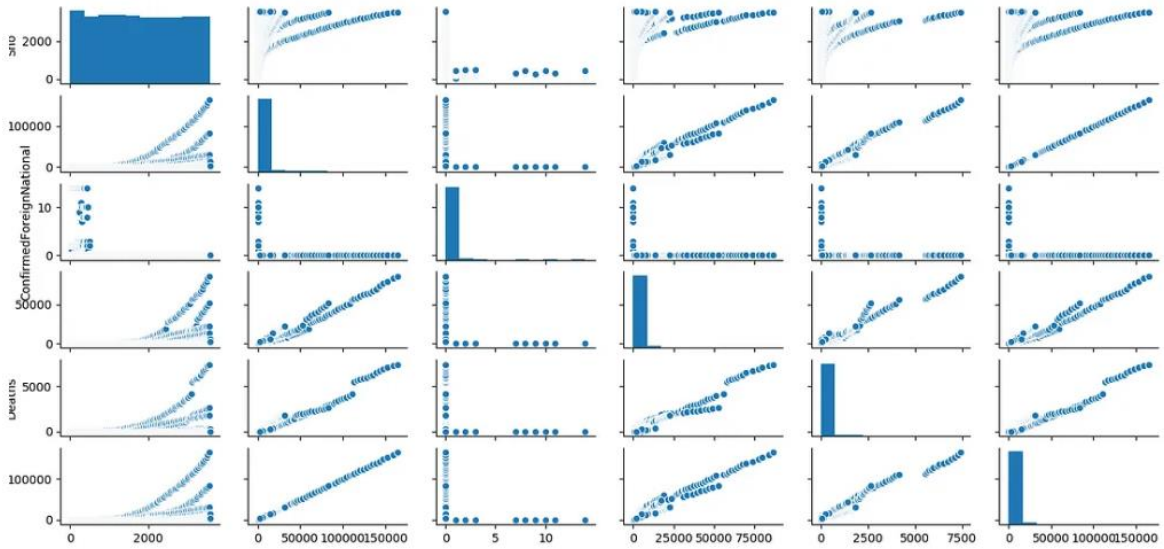
The percentage of the response variable's variance that the predictor variables in a linear regression model can explain is known as R-squared or R2, and it is frequently stated as such. R-squared values vary from 0 to 1, with 0 denoting that the response variable is completely unrelated to the predictor variable. 1 means that the predictor variables can completely and error-free explain the response variable.

The regression analysis is done by first importing the libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt #Data visualisation libraries
import seaborn as sns
from sklearn.model_selection import KFold, cross_val_score,
train_test_split
```

The dataset were then read using seaborn heatmap

The output is as follows



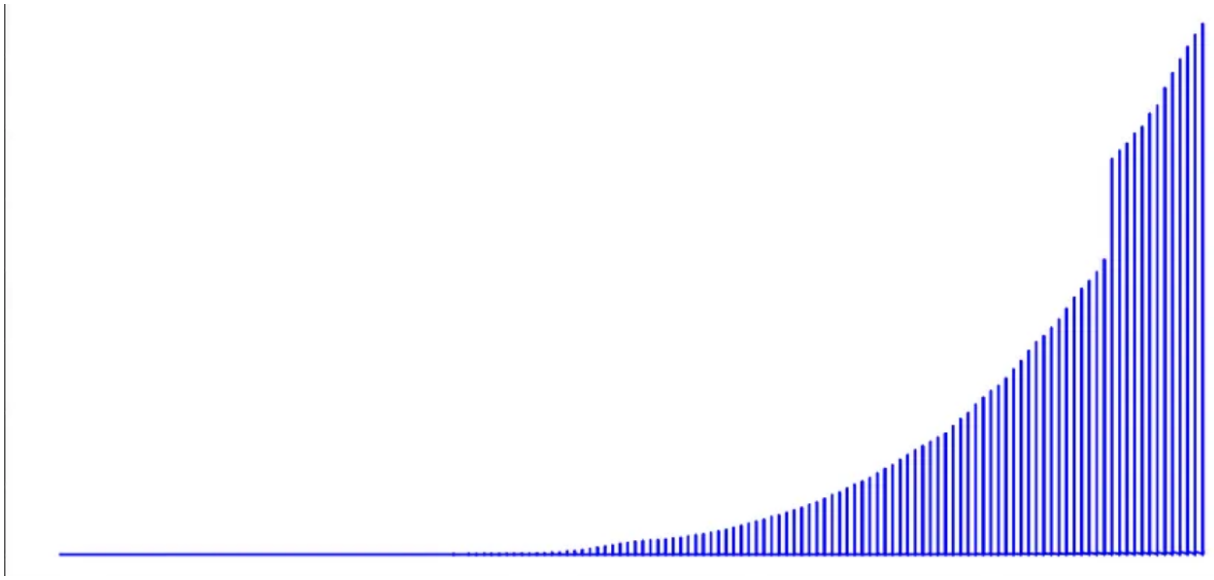
The next step is to train the regression model, this starts with splitting the data into X array, and it has features that are to be trained on it, there is also Y array that has a target variable, and in this case it is the death column.

```
X = covid_data[['Date', 'Confirmed']]
y = covid_data['Deaths']
X['Date'] = pd.to_datetime(X['Date'], format='%d-%m-%y')
```

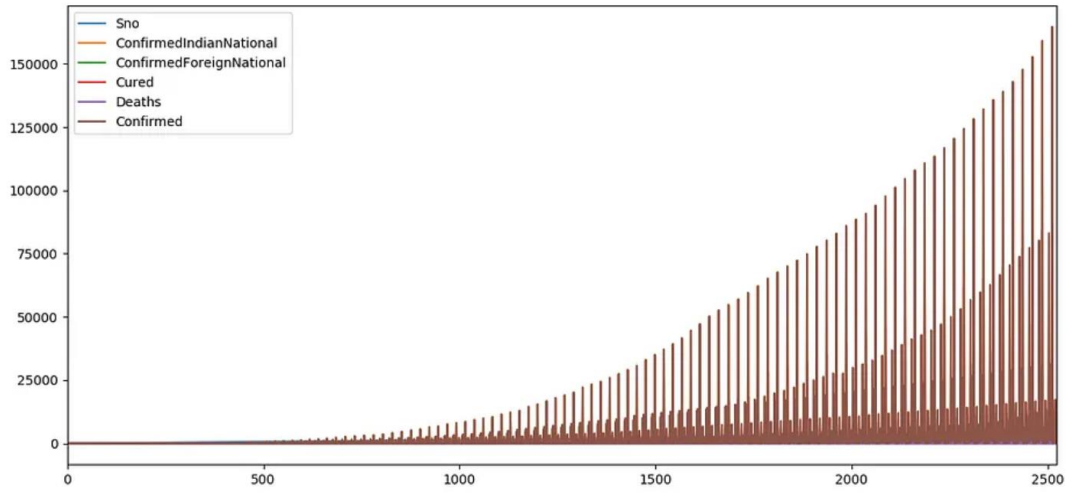
The plotting of death rate with the respect to the obtained data, and they are plot with the respect to their respective dates.

```
plt.xticks(rotation=45)
print(X['Date'])
plt.plot_date(X['Date'], y, fmt='b-', xdate=True, ydate=False)
covid_data.plot()
plt.show()
date = X.loc[:, ['Date']]
X['Date2num'] = X['Date'].apply(lambda x: mdates.date2num(x))
del X['Date']
```

The output is as follows;



The death rates plotting



The next step is to train the split, and this involves splitting the data into a training dataset as well as the test dataset, and in this case, 30% of the data are used as the test data while 70% are used as the training data.

```
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.3, random_state=101)
date_test = date.loc[:np.floor(m*0.3)]
date_train = date.loc[np.floor(m*0.3)+1:]
```

This is followed by creating as well as training the model by fitting the linear regression model found on the training data.

```
lr = LinearRegression()
lr.fit(X_train,y_train)
```

4.3.2 The visualization

A. Hyperparameter tuning using Python libraries (e.g., GridSearchCV, RandomizedSearchCV)

The use of GridSearchCV, RandomizedSearchCV, cross-validateand sckikit-learns support the evaluation based on multiple metrics simultaneously.

Evaluation of model using cross-val-score scoring criteria

#evaluate the score by cross validation

rf_score = cross_val_score(clf, X, y, cv=3)

#Random Forest Model

clf = RandomForestClassifier(random_state=0)

4.3.3 Decision Tree Model

[12]:

```
#Decision Tree Model
clf = DecisionTreeClassifier(random_state=0)

#evaluate the score by cross validation
dt_score = cross_val_score(clf, X, y, cv=3)
```

Output

Average Decistion Tree Score: 0.5365318184945371

Average Logistic Regression Score: 0.5566235174768442

Average Random Forest Score: 0.5395904019269173

Logistic Regression has the highest score, at 0.56, and therefore, it has the best predictions

Code

[1]:

```
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt

from sklearn.preprocessing import OneHotEncoder, MinMaxScaler, LabelEncoder
from sklearn.compose import ColumnTransformer

from sklearn.model_selection import cross_val_score

from sklearn.tree import DecisionTreeClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import LinearSVC

from sklearn import tree

from sklearn.cluster import KMeans
```

4.4 Machine Learning Models for Transmission Rate Prediction

4.4.1 Description and implementation of regression algorithms in Python

Regression is used to predict the future of continuous values of dependent variables by analysing the relationship between the variable. From the analysis, the Average Logistic Regression Score was found to be 0.5566235174768442. Logistic Regression has the highest score, at 0.56, and therefore, it has the best predictions. Logistic Regression is used for the binary classification of tasks and the goal is the prediction of possible outcomes such as 0/1, true/false, yes/no. It is a classification algorithm, and it uses a sigmoid function for it to model the relationship between the COVID-19 rate and cases. The logistic regression algorithm models the kind of relationship existing between the binary outcome, which is the dependent variable, and the features, which is the independent variable, using the sigmoid function. Sigmoid(z) = 1 / (1 + e^(-z)). 'z' is the linear combination of the features and their corresponding weights. In the implementation in Python, Scikit-learn was used.

4.4.2 Model selection and evaluation for transmission rate prediction

The model selection and evaluation for transmission involves choosing the best model for prediction and assessing performance to ensure reliability and accuracy. The data was obtained from kaggle.com and the rate and cases of COVID-19 were selected as the variables of interest. The Random Forest method was used as the model of interest. For this analysis and prediction, the research used K-means clustering to see how well to group the data. The positivity rate for the entire sample is 38.9%. However, profiling our data using clustering can help us to identify high-risk groups (groups that have a positivity rate of

higher than 38.9%) as is seen below.

```
kmeans = KMeans(n_clusters=5, random_state=0).fit(X)
df2['kmeans_label'] = kmeans.labels_
```

```
def print_value_count(frame, label):
    print(f'Number of cases for label {label} is: ')
    print('\n',     frame.query(f'kmeans_label=={label}').covid_res.value_counts(normal

print(f'Number of cases for the whole group is: ')
print('\n', df2.covid_res.value_counts(normalize=True))

labels = df2.kmeans_label.unique()
for label in labels:
    print_value_count(df2, label)
```

Number of cases for the whole group is:

not positive 0.492471

positive 0.389439

pending 0.118090

Name: covid_res, dtype: float64

Number of cases for label 4 is:

positive 0.585978

not positive 0.297669

pending 0.116353

Name: covid_res, dtype: float64

Number of cases for label 2 is:

not positive 0.516512

positive 0.367543

pending 0.115945

Name: covid_res, dtype: float64

Number of cases for label 0 is:

positive 0.521641

not positive 0.358278

pending 0.120081

Name: covid_res, dtype: float64

Number of cases for label 3 is:

not positive 0.560321

positive 0.322483

pending 0.117196

Name: covid_res, dtype: float64

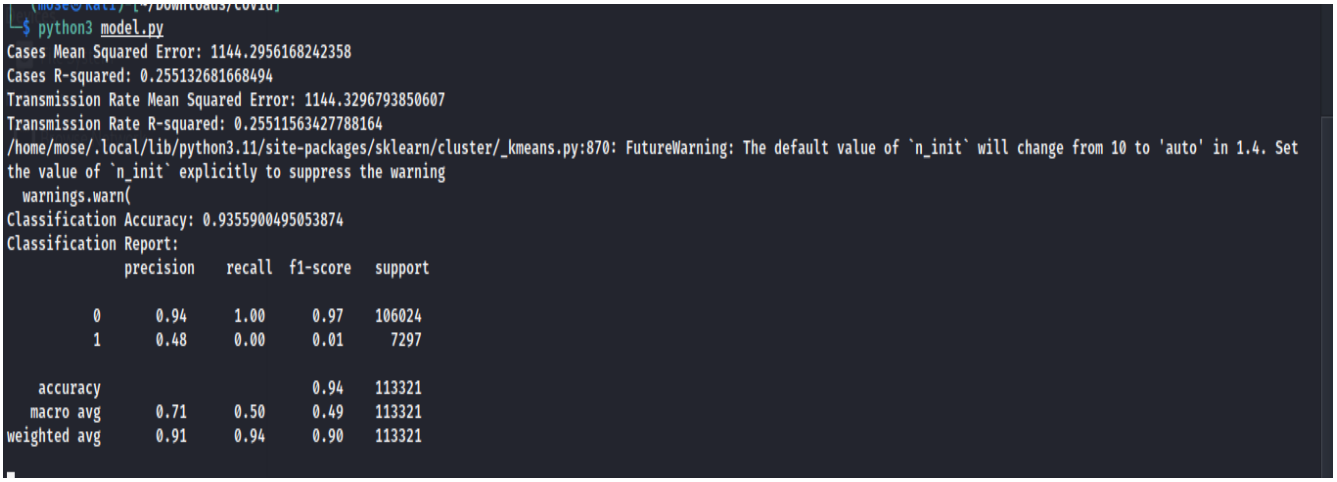
Number of cases for label 1 is:

not positive 0.51641

positive 0.35707

pending 0.12652

Name: covid_res, dtype: float64



4.4.3 Hyperparameter tuning for transmission rate prediction models

4.4.3.1 Model Interpretability

Techniques for model interpretability in Python

There are different techniques that have to be considered when developing a machine learning model to enhance intrepretability and these include LIME (local interpretable model), SHAP (Shapley Additive exPlanations) and Anchors

4.4.3.2 SHAP

This technique is useful is explaining the prediction of the instance. This is done by determining the value of each feature and how it contributes to the final prediction. This technique is adopted from the game theory which implies that in a game there are different

players and every player plays a valuable role. The distribution of roles on the features is established using the Shapley values.

4.4.3.3 LIME

The LIME technique explains the predictions by using the model approximates that are intrinsically interpretable, just like those in a decision tree. The predictions are basically made by probing the model many times to make a comparison that can help understand the patterns.

4.4.3.4 Anchors

This technique is used to generate local explanations. According to this technique, any changes that are made to the value of features have no ability to change the predictions. Anchors are explanations generated from rules also known as the IF-THEN rules.

4.4.4 Interpretation of results and insights into key factors affecting cases and transmission rate

According to the results of the analysis conducted, there were three important factors that were considered responsible for the cases and the transmission rates. These where age, gender, and health condition. According to the results, it indicates that most of the people with underlying medical conditions had a lower survival rate compared to healthy persons.

According to the Patient survival profile the model indicated that 6.4% of the population in the review died. These deaths are from 8.2% of the female population and 4.5% of the male population as shown below. This result indicated that based on sex, female population was more vulnerable to the COVID-19 compared to the male population

indicating that men had higher survival rate compared to women. These results are also graphically represented in the figure above.

The survival ration illustrated below indicated that the death ratio is lower than the death ratio which means that out of all the transmission cases, more people survived.

Survival ratio

Survived 0.936153 died 0.063847 Name: survival_status, dtype: float64

When the survival ratio was calculated against gender as indicated below, the male-to-female ratio shows that males survived more than females out of the cases studied..

survival against gender

survival_status	died	survived
sex		
Female	0.081780	0.918220
Male	0.045426	0.954574

As shown in the figure below, the death rate by age group generally increases with age group, and peaks at age group (80-90), at 30.4% as shown below.

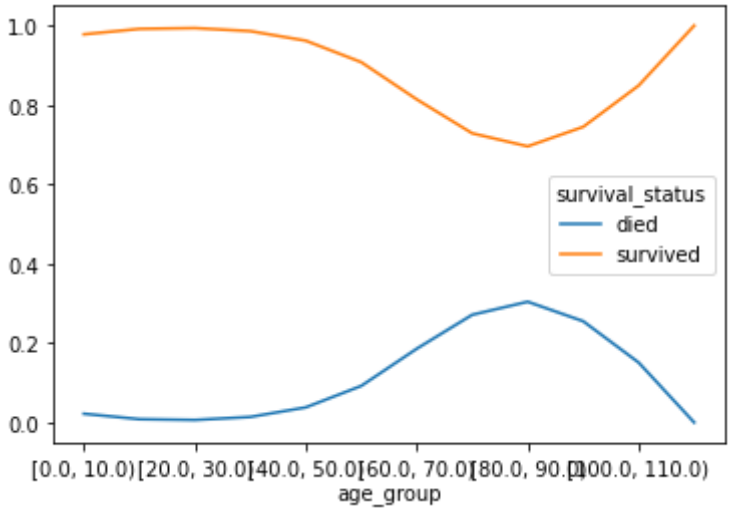


Figure: Graphical Presentation of the Survival Rate Based on the Age Factor

5. CONCLUSION

Analysis commenced with a foundational endeavour—preprocessing the dataset. This initial, often overlooked step involved a delicate sequence of transforming categorical data, such as age, sex, symptoms, survival status, and more, into numerical representations. Techniques like label encoding played a pivotal role, orchestrating the conversion that laid the groundwork for an incisive analysis. This preliminary step, while seemingly minute, underscored the vital significance of setting the stage for the analytical sojourn that lay ahead.

Central to data exploration was the realm of predictive modelling, wherein we endeavoured to decode two critical aspects of the pandemic's landscape: the projected number of COVID-19 cases and the dynamic transmission rate. Equipped with the robust machinery of Random Forest Regressors, we delved into a world where algorithms assumed the role of learners, synthesizers, and predictors. A process of meticulous training and validation unfolded, as these algorithms ingested the nuances of data, unravelling the intricate web of relationships intricately woven within the dataset. The culmination of this journey was the birth of models capable of forecasting case numbers and transmission rates.

These models, products of algorithmic ingenuity, transcended the realm of mere mathematical constructs. They emerged as predictive tools, offering a glimpse into the panoramic view of an unfolding pandemic. However, our predictions were not definitive prophecies; rather, they stood as windows through which we could peer into potential trajectories. The accuracy of our models was scrutinized through the lenses of Mean

Squared Error (MSE) and R-squared values. The former unfurled the average squared discrepancy between our predictions and actual data, while the latter encapsulated the models' aptitude for capturing the intricate variability inherent within the pandemic's complexity. These quantitative metrics, while crucial, were intertwined with an inherent recognition of the complexities and subtleties woven into the very fabric of the pandemic.

In this thesis, there is consideration of the provided COVID-19 data from Kaggle. There is a use of tabular, graphical, statistical, and machine learning approaches to identify patterns in the dataset and this is what has been used for the prediction. All the ethical factors were included, and electronic health records were used since they have led to the highlighting of ethical issues such as data privacy, data ownership, and secondary use of data. Fairness is upheld by incorporating techniques that will not distort the actual measurements in the existing data. All people with self-interest were considered, hence no particular gender or age group was considered to be the most effective one.

It is evident that there are trends that can be identified, such as the survival rate of each age group. From the results, it has been realized that the linear regression analysis can be used when it comes to the prediction of COVID-19 rates and cases. If intervention is taken into consideration, and then the rate of cases is likely to increase. During the first few months, the rate was low and few people were getting admitted, and as time went by, more people contracted the virus and cases increased from March 2020 through June 2020.

It has also been discovered that the use of classification models can help interested parties to identify infection rates in the population. Additionally, clustering can, as we have discovered, help to group the population for better allocation of resources. With proper

planning and data utilization, lives can be saved, and resource utilization optimized by making recommendations, regulations, and strategies that are oriented to this direction. Data collection policies need to be adhered to so that the value, volume, and accuracy of the data can be increased. This would lead to more comprehensive datasets, which would, in turn, likely increase the outputs and impacts of the models developed. Based on the findings of this research, it was discovered that in regard to gender, the male gender was more prone to transmission compared to the female gender. Machine learning has been most effective in this process; however, it is prone to errors. Therefore, an error with the measurements or in case of missing data and outlier, the accuracy of the results is questioned.

From the analysis, it has been realized that the duration between appearances of the initial symptoms increased greatly from January to June 2020, which was the period under review. The numbers of males under review are highly affected, in percentage, as they make up 50.7% while the females make up 49.3%, which is a difference of 1.4%. The average decision tree score was found to be 0.5365318184945371, and it was the lowest, followed by the Average Random Forest Score, which was found to be 0.5395904019269173 this is slightly below the average Logistic Regression Score which is 0.5566235174768442.

Therefore, the Logistic Regression has the highest score, at 0.56, while the lowest score is 0.54 hence; it has the best predictions since the difference is 0.02. In this case, K-means clustering was used to determine how well the data can be grouped, and the analysis revealed the positive rate for the whole sample to be 38.9%. Nevertheless, profiling the data by use of clustering can assist in the identification of high-risk groups, which are

groups that have a positive rate that is above 38.9%. The results also show that an average of 6.4% of the entire population died, which consists of 8.2% of the female population and 4.5% of the male population. The females were more vulnerable than their male counterparts, according to it..

This is an indication that 93.6% of the people who were affected by COVID-19 survived, which consisted of 95.5% of the males and 91.8% of the females. The result also showed that the death rate increased with the increase in age, hence those of the older age group had higher mortality rates due to COVID-19 than the younger one. The result shows that those who are between the ages of 80-90 recorded a death rate of 30.4%. As our models traversed the expanse of data and code, they birthed tangible outcomes. Our endeavour yielded a tapestry of results—quantitative manifestations of our engagement with data. The Mean Squared Error (MSE) and R-squared values, serving as companions to our predictive models, offered tantalizing glimpses into the precision and explanatory potency of our predictions. The classification report, a by-product of the Random Forest Classifier, offering a panoramic view of the factors that intricately shaped survival outcomes.

However, these outcomes were not monolithic, isolated entities; they were conduits of interpretation. Our models, fortified by the wisdom of data and guided by the intricacies of algorithms, were vehicles that facilitated our exploration. Rather than viewing these outcomes as definitive endpoints, we embraced them as stepping stones—a culmination of our pursuit to meld the artistry of science with the inherent complexities of the pandemic.

6. FURTHER RESEARCH

Future research endeavors in the context of COVID-19 should encompass a multifaceted approach to gain deeper insights into the pandemic's impact on different genders and transmission dynamics. Here are some key areas that warrant attention:

1. Prediction of Gender-Specific COVID-19 Cases:

Future studies should focus on predicting COVID-19 cases, taking into account gender disparities. Males have been observed to be disproportionately affected by the virus, and understanding the factors contributing to this discrepancy is essential. Predictive models can help anticipate trends and allocate healthcare resources accordingly.

2. Gender-Based Disease Transmission Analysis:

Investigating how COVID-19 spreads differently among genders is crucial. Research can delve into behavioral, biological, and sociocultural factors that influence transmission. This knowledge can inform tailored prevention strategies for each gender.

3. Factors Influencing Transmission Disparities:

Identifying the specific factors contributing to variations in transmission rates among genders is essential. This could involve studying occupation, social interactions, healthcare access, and adherence to preventive measures.

4. Python-Based Data Analysis:

Utilizing Python for data analysis is a valuable approach. Python offers a wide range of data analysis libraries and tools that can assist in processing and interpreting complex datasets effectively.

5. Country-Specific Analysis:

Recognizing that the spread of COVID-19 varies by country and even within communities, future research should consider country-specific analyses. Examining the unique challenges and dynamics in different regions can guide tailored public health interventions.

6. Community-Level Research:

Within countries, research should extend to community-level analyses. Factors such as population density, healthcare infrastructure, and cultural practices can influence disease transmission at a local level.

7. Preventive Measures Tailored to Gender:

The findings from gender-specific research should inform the development of prevention measures tailored to each gender. This could include targeted public health campaigns and recommendations.

8. Long-Term Impacts:

Investigating the long-term physical and psychological impacts of COVID-19 on individuals of different genders is another avenue for research. Understanding post-recovery challenges can aid in providing appropriate support.

By addressing these aspects in future research endeavors, we can enhance our understanding of the pandemic's impact and develop more effective strategies to combat it, taking into account gender-specific nuances and regional disparities..

7. BIBLIOGRAPHY

Brownlee, J. (2017). *Long short-term memory networks with Python: Develop sequence prediction models with deep learning*. Machine Learning Mastery.

Byeon, W. (2016). *Image analysis with long short-term memory recurrent neural networks*.

Chan, I. C. (2015). *Input method engine by long short-term memory recurrent neural network*.

Gers, F. (2001). *Long short-term memory in recurrent neural networks*.

Ly, R., Traore, F., & Dia, K. (2021). *Forecasting commodity prices using long-short-term memory neural networks*. Intl Food Policy Res Inst.

Sabry, F. (2023). *Long short term memory: Fundamentals and applications for sequence prediction*. One Billion Knowledgeable.

Dym, H., & McKean, H. P. (2008). *Gaussian processes function theory and the inverse spectral problem*. Courier Corporation.

Ibragimov, I., & Rozanov, Y. (2012). *Gaussian random processes*. Springer Science & Business Media.

Kramer, O. (2013). *Dimensionality reduction with unsupervised nearest neighbors*. Springer Science & Business Media.

Lifshits, M. (2012). *Lectures on Gaussian processes*. Springer Science & Business Media.

Marcus, M. B., & Rosen, J. (2006). *Markov processes, gaussian processes, and local times*. Cambridge University Press.

Osipov, A. (2012). *A randomized approximate nearest neighbors algorithm*. LAP Lambert Academic Publishing.

Roberts, J. D. (1990). *Proximity content-addressable memory: An efficient extension to k-nearest neighbors search*.

Sabry, F. (2023). *K nearest neighbor algorithm: Fundamentals and applications*. One Billion Knowledgeable.