**CHAPTER FOUR**

**SYSTEM DESIGN AND IMPLEMENTATION**

**4.1 Research Design**

The research design for this study was exploratory. It is concerned with the critical analysis of the system and seeks to extract information to achieve the desired aim and objectives. At this stage, the requirements identified during system methodology are converted into tangible results.

**4.2 System Design**

System design is the process of defining the elements of a system like an interface, components, models, and architecture based on the requirements identified in chapter three. A Top-Down or Bottom-up approach is required to capture all the required variables of the system. The design of the system for this research is classified into two main sections which are: Model design and development section and the web interface used for the development.

**4.3 Model Design and Development**

This section comprises of polynomial regression algorithm for the prediction of the incidence covid-19 cases in Nigeria. There are different processes involved in this section. This start with the dataset description to the final result from the model.

**4.3.1 Dataset Description**

The dataset for this prediction algorithm was obtained from the official website of the Nigeria Center for Disease and Control, the official site of the body responsible for confirming and handling covid-19 cases in Nigeria. It is made up of 4 features and one target variable. There are 37 instances which are the states in Nigeria and Abuja the Federal Capital Territory. The dataset features and its description is shown below in the table.

**Table 4.1: Dataset Feature Table**

|  |  |
| --- | --- |
| **Features** | **Description** |
| States Affected | This includes all the states in the country and the federal capital where the data were collected |
| No. of Cases (Lab Confirmed) | This is the total number of covid-19 cases confirmed in the lab |
| No. of Cases (on admission) | This is the total number of people with covid-19 currently on admission |
| No. Discharged | This is the total number of people with covid-19 that have been discharged from the isolation center |
| No. of Deaths | This is the total death recorded |

**4.3.2 Exploratory Data Analysis (EDA) and Data cleaning**

The analysis of the dataset started with data exploration and cleaning. The cleaning stage of the data preparation starts with identifying column(s) of the dataset that are not relevant and will not be useful in the model development. These features were removed from the dataset. Some of the features dropped include state and unnamed 0 columns. The state was dropped because the polynomial regression model is built to show the non-linear relationship between the variable of power higher than and the independent variable which are both numerical values, and the unnamed 0 column was equally dropped. After all, it is a replication of the index number which does not contribute to the model development and also doesn't have any meaning to the dataset.

The next stage of the cleaning was the detection of missing values in the dataset. This was carefully checked and the result showed that there was no missing value.

Another operation that was carried out is to visualize the trend of values in some important parameters such as total confirmed cases, number of death, and number of cases on admission. Starting with the total confirmed case, the trend is shown in the graph below

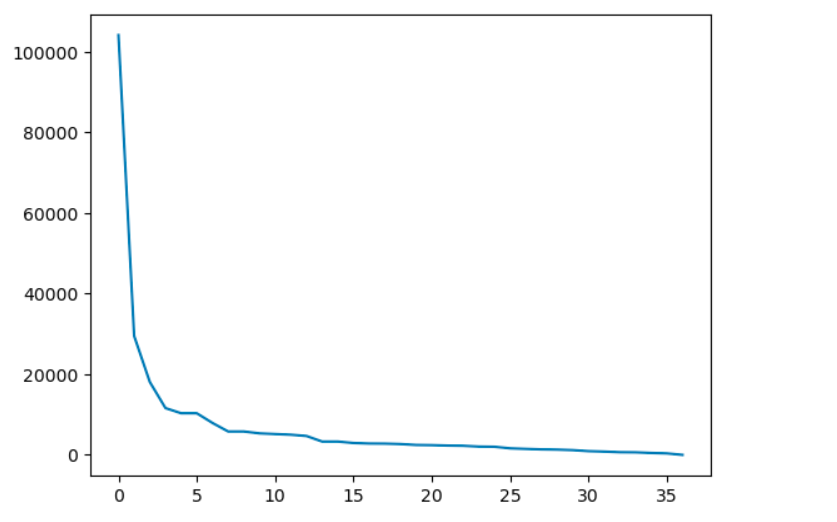


Figure 4.1: Chat showing the trend of confirmed cases

According to the chat above, the total number of confirmed cases was plotted against the state. The index of the state was used to represent it. As observed from the chart, there was a downward decrease in the number of cases as we move down the state. For instance, Lagos, FCT, Rivers, and Kaduna which fall on index 0-3 had the highest number of confirmed cases while states like kebbi, Zamfara, and Kogi had the least number of confirmed cases.

Considering the total number of death, the slope was not the same as that of the total confirmed cases as shown in the chat below.

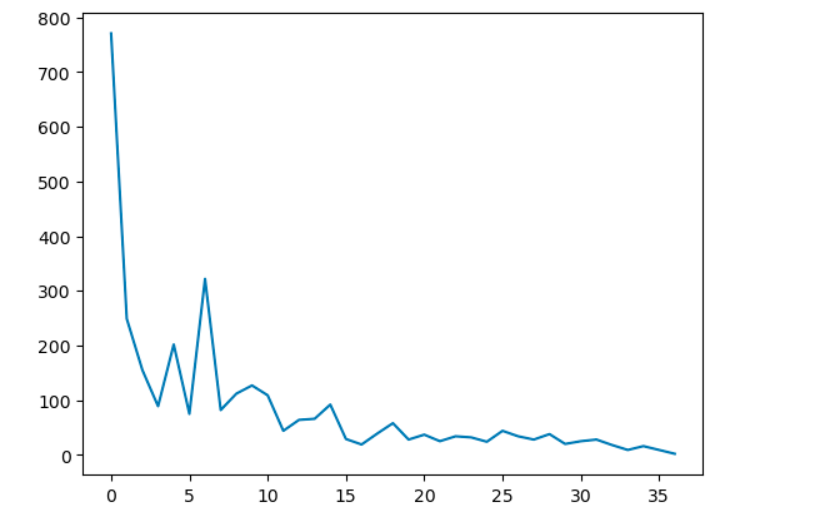


Figure 4.2: Chat showing the trend of total death

Comparing the slope of the total death with the total confirmed cases show that the relationship between the two is not linear, therefore a better algorithm is required which can correctly fit the two instances.

The next stage of the analysis is calculating the correlation coefficient of the numerical items. The result shows that some parameters are strongly correlated. The result of the correlation chart is shown below

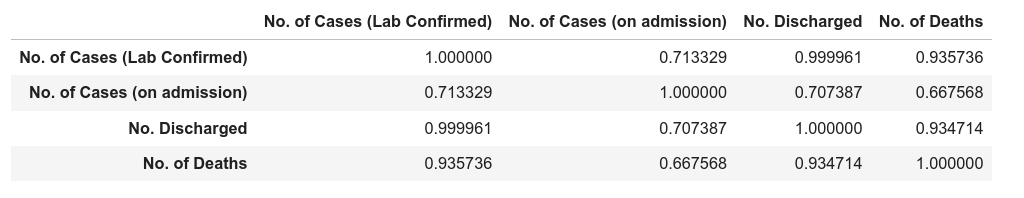


Table 4.1: Correlation table for the variables

From the table above, it was observed that there is a strong positive correlation between the independent variable (Number of Deaths) and the dependent variable (Number of confirmed cases.). The correlation coefficient is 0.94.

Apart from the correlation coefficient, the statistical relationship between the features is calculated as shown in the table below:

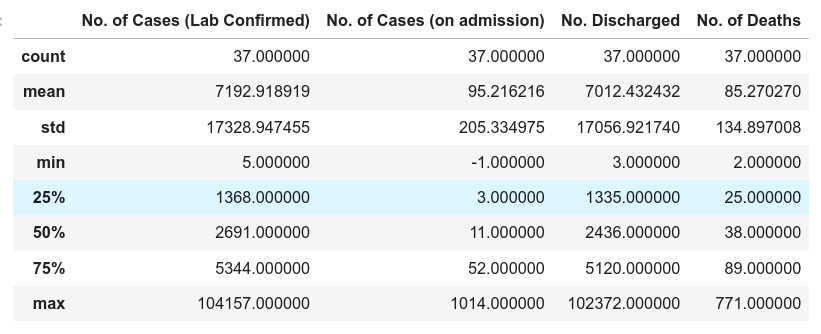


Table 4.2: Statistical relationship table

From the table above, the standard deviation of each feature was calculated. The result indicated that there is much variation in the number of confirmed cases since the standard variation is large which implies that the data distribution is far from the mean while in the case of the target variable, the disparity is not much.

The last stage of the exploratory data analysis is data splitting and data encoding. At this stage, the data are categorized and grouped as target and feature variables. The polynomial feature is also derived using the polynomial module of the Scikit-learn library.

**4.3.3 Model Development and Prediction**

After splitting the dataset, due to the nature of the dataset, different two different algorithms were built to train on the data. This is done to justify the accuracy of the polynomial regression algorithm. The algorithm developed and trained on the dataset is linear regression and polynomial regression. This used the training data for training and the performance was checked using a graphical representation.

The general equation for any polynomial regression is shown below

**y** = a + b1x + b2x2 +....+ bnxn

where:

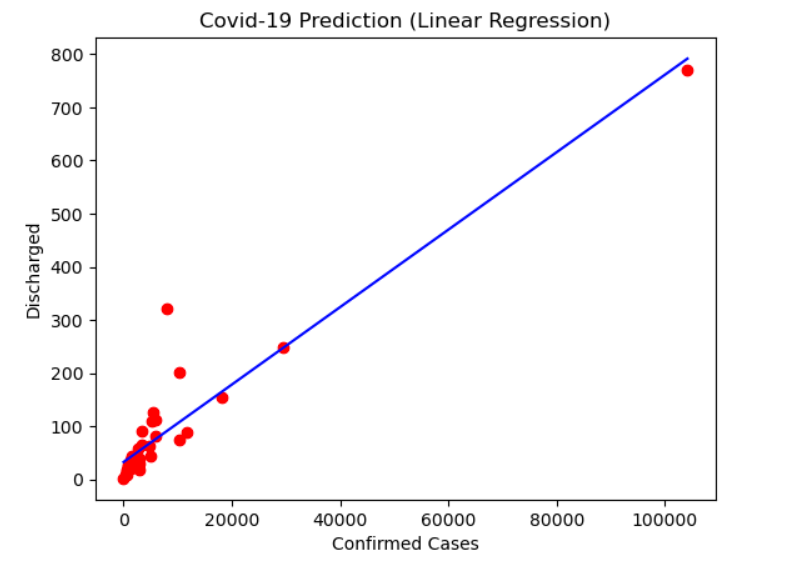
y = dependent variable

a = y-intercept

b= slope

x = independent variable

The model was trained on the data and the graph of the polynomial regression was generated as shown in the diagram below



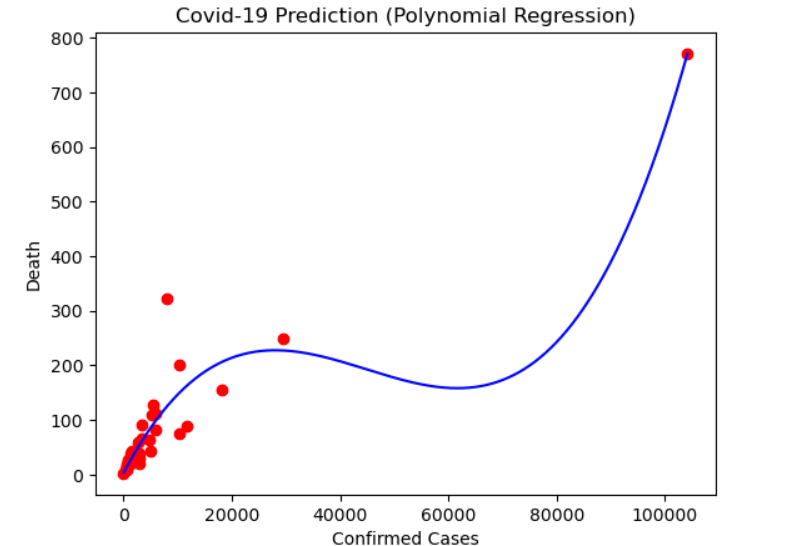


Figure 4.3: Polynomial Regression curve

The value of the slope and the intercept were calculated based on the dataset and also on the generated output from the mode. The model has an intercept of approximately 2.97 and a slope of [0.00000000e+00, 1.89067952e-02, -4.90818294e-07, 3.64938288e-12].

The performance of the model was evaluated using Mean square Error (MSE). Mean square error is calculated mathematically as following

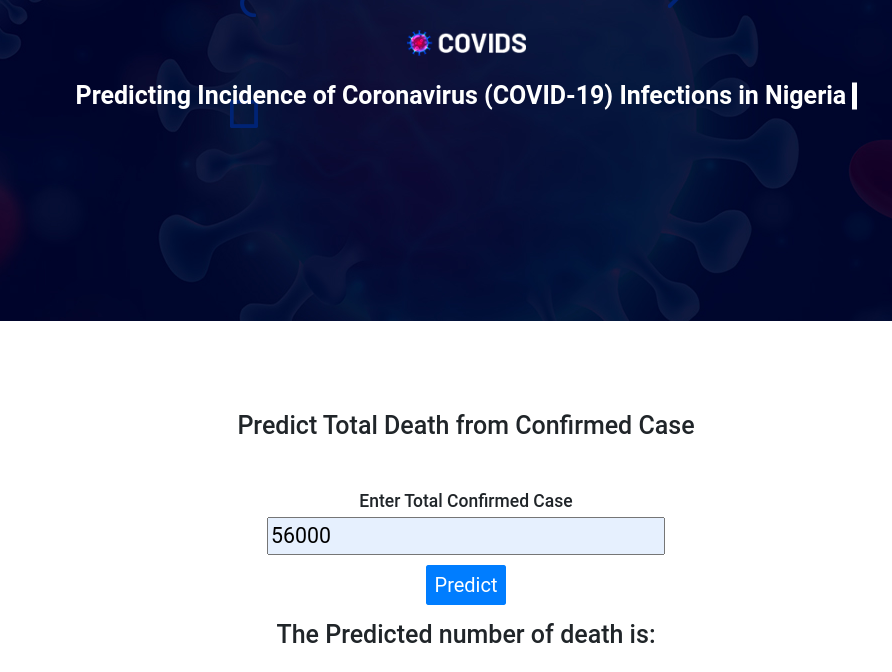
R2 =

The Mean Square Error for the analysis is 0.9. the closer the MSE to 1, the better the model. Since the Model generated an MSE of 0.9 shows that the model did not overfit the dataset.

**4.3.4 Model Deployment and Testing**

After a series of testing and evaluation of the model, the model was later deployed using flaskAPI. Deployment is the art of making the machine-learning model accessible to the users. This gives the model the ability to accept input from a web form and generate output based on the history of the data which was supplied from the form. The results of the prediction are rendered to the web interface using the python Flask library. Flask is a back-end web development library written in python.

The web-based interface used for testing is shown below. This shows the interface when supplied data and when the result was generated.

(a)

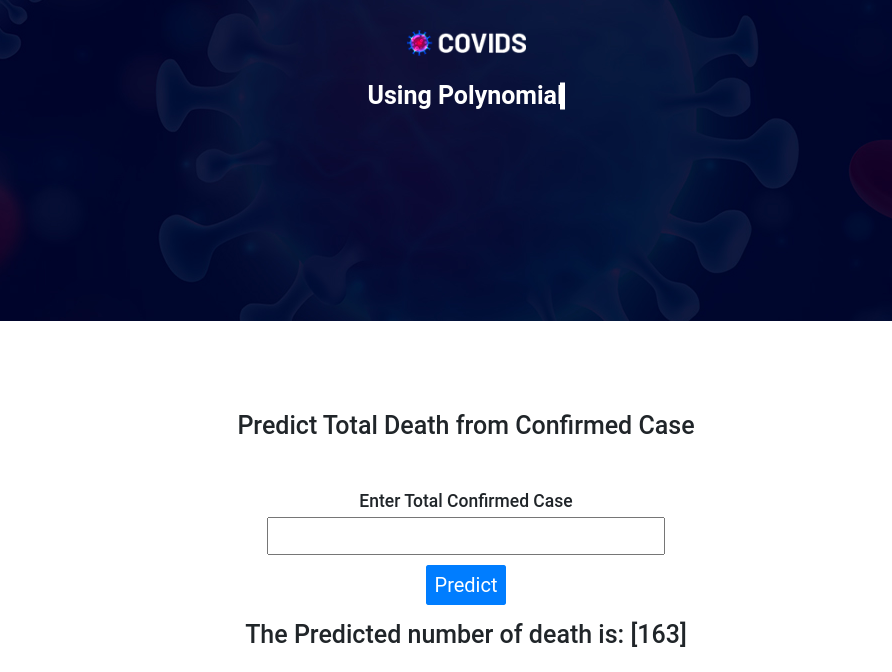
(b)

Figure 4.4: Deployment using flask before and after prediction

**4.4** **System Implementation**

The new system was implemented using Python programming language version 3.9. Python is an object-based programming language that sees everything as an object. It has a simple syntax like the English language. The syntax allows developers to write programs with fewer codes. Python has about 124,000 third-party python libraries that enhance machine learning programming. Pandas, Numpy, and Matplotlib are examples of python libraries used for data visualization and manipulations.

Python can be used in almost every field of software development such as data science, Artificial Intelligence, web programming, and machine learning. Python is both object and procedure oriented.

**4.5 System Specification**

System requirement is divided into software and hardware requirement. Below is the detailed requirement for the developed system.

4.5.1 **Hardware Requirements**

The list below describes the hardware components and software requirements needed for the effective and efficient running of the system. For the implementation of the newly developed traffic offenders profiling and prediction system, the following hardware is required:

* 2.4 GHz of processor speed
* 8GB RAM
* 500 GB Hard disks
* Monitor

**4.5.2 Software Requirements**

The following software has to be installed on the computer system to run the new system developed.

1. Window 8 and above
2. IDE (Vs code and Jupyter Notebook)
3. Programming Language: Python
4. Anaconda