# EARTHQUAKE PREDICTION MODEL USING PYTHON

# Name: VIGNESH K Register Number: 61772231T310

## Problem Definition and Design Thinking Problem Definition:

The problem is to develop an earthquake prediction model using a Kaggle database which analysis the data from (2016 to 2020). The objective is to explore the key features of the earthquake and predict the earthquake magnitude. We need the train the AI so that only we can get the accurate solution by using python

## Key Components:

Data Collection, Data Processing, Feature Selection, Model Selection, Training and Validation.

## Design Thinking:

Data Source: Choose a suitable Kaggle dataset containing earthquake da with features like data, time, latitude, longitude, depth and magnitude.

Feature Exploration: Analyse and understand the distribution, correlation and characteristics of the key features..

Visualization: Create a world map visualization to display earthquake frequency distribution. Data Splitting: Split the dataset into a training set and a test set for model validation

Model Development: Build a neural network model for earthquake magnitude prediction.

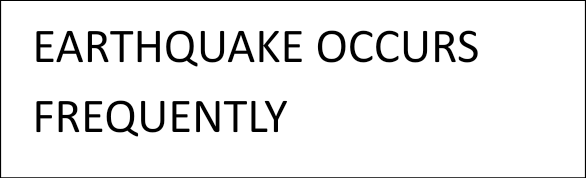
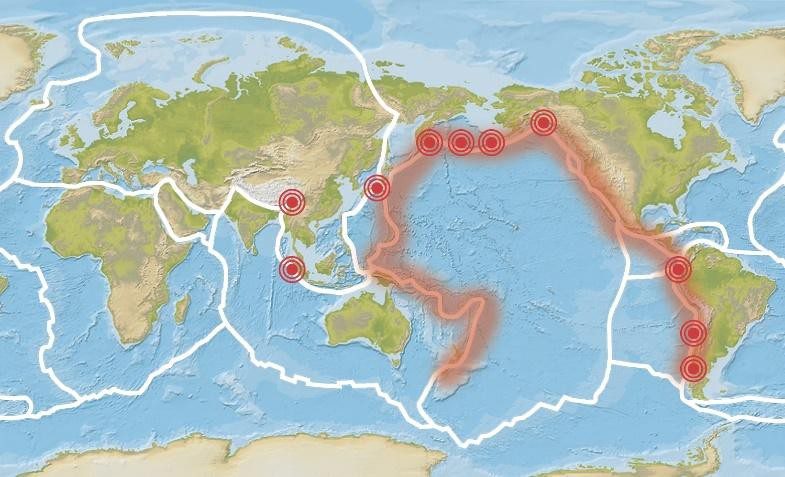
Training and Validation: Train the model on the training set the evaluates its performance on the test set.

**Short review of the model:**

1. First let’s start with the data collection.
   1. Data Collection is from Kaggle document.

|  | latitude | longitude | depth | magnitude | Timestamp |
| --- | --- | --- | --- | --- | --- |
| 1 | 19.246 | 145.616 | 131.6 | 6.0 | -1.57631e+08 |
| 2 | 1.863 | 127.352 | 80.0 | 5.8 | -1.57466e+08 |

1. Data Visualization- Let create a world map according to the data collected where the earthquake



occurs frequently.

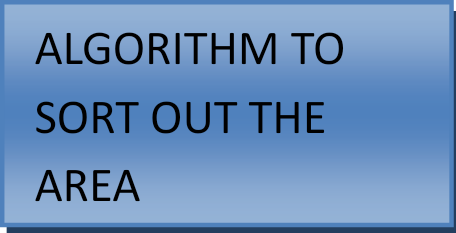
1. Splitting the dataset- According to the algorithm we using let’s split up the data and analysis.

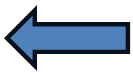
Example: input be (timestamp, latitude, longitude) and the output be(magnitude and depth). Now according to this data we search the place of earthquake occurrence.

1. Neural Network- After the analysis of the above given data we need to connect to the neural network. Further the neural network analysis the loss and gain of the earthquake.

Final we can decide whether the earthquake occur or not. If occur what is the rate of loss will be trained to the AI by we trained by the Kaggle dataset.

## FLOW CHART:



# ALGORITHM:

**MACHINE LEARNING**:

Machine learning is a field of Computer Science. It evolved from study of pattern recognision as well as computational learning theory in Artificial intelligence. It is field of predictive modeling with minimizing error of a model and trying to make the most precise predictions. Machine learning uses algorithms from different fields like statistics and use them to make predictions. The algorithm used here is Regression algorithm. Regression is a topic in the field of statistics. These regression algorithms are applied on numerical variables. So it is both a statistical as well as a machine learning algorithm.

**ALGORITHMS:**

LINEAR REGRESSION: Linear regression is a statistical regression method which is used for prediction analysis. It is used for solving the regression problem in machine learning. It shows a relationship between the independent variable (x-axis) and dependent variable (y-axis). Also it has two types of regression.

1.Simple linear 2.Multiple linear

Once the model has been fit to the data, we can use it to predict the magnitude of a new earthquake given its latitude, longitude, depth, and the number of seismic stations that recorded it. This can be useful for earthquake monitoring and early warning systems, as well as for understanding the underlying causes of earthquakes and improving our ability to predict them.

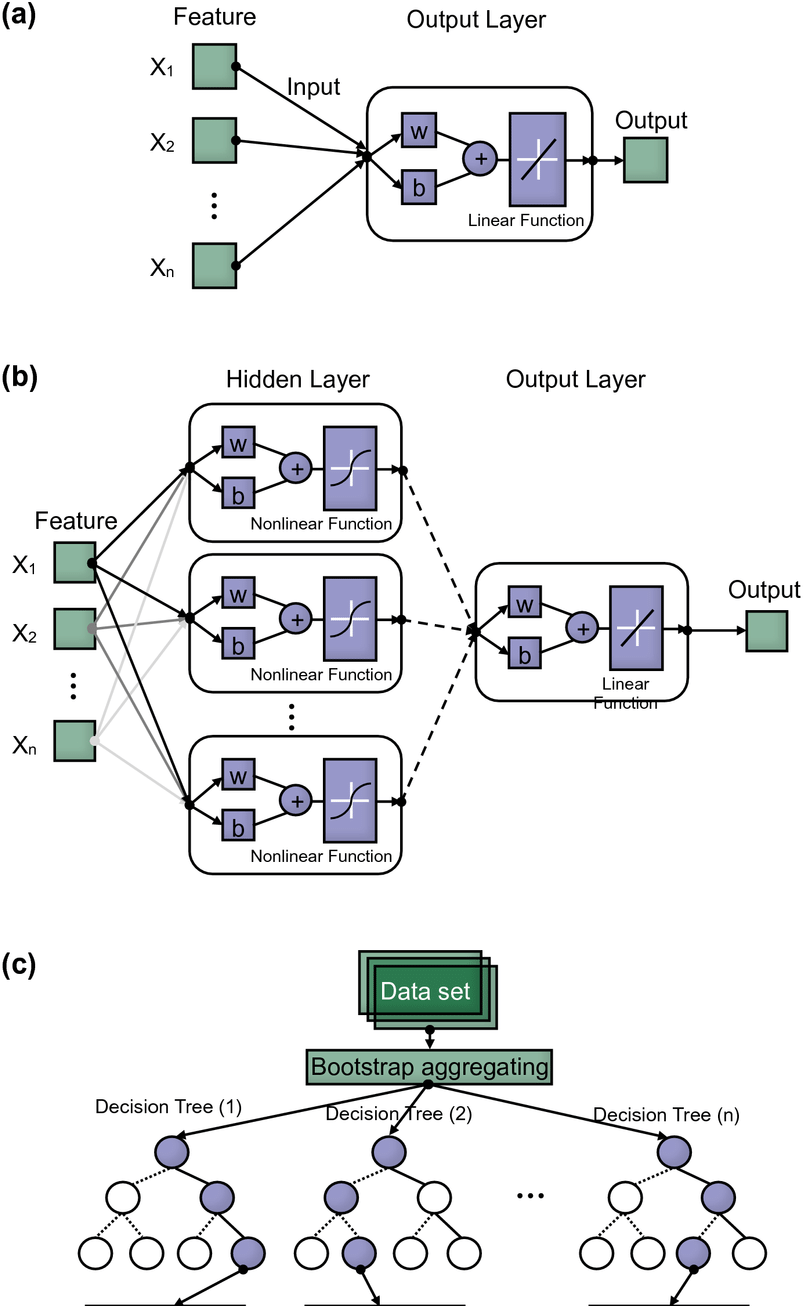
**EQUATION:**

y = B0 + B1\*x

Where: B0 = coefficient, B1 = coefficient, x = variable, y = variable

**RIDGE REGRESSION:**

This technique is also used when the data suffers from multicollinearity. So the least squares estimates are unbiased but the variance is high. Their variances are large. Due to this the observed value deviates far from the true value. Ridge regression reduces the standard errors and for this it adds a degree of bias to regression estimate. As we saw in a linear equation the prediction errors have two components. One is biased and other is variance. The prediction error can occur due to these both components. The multicollinearity problem is solved by using a shrinkage parameter λ (lambda).



# EQUATION:

β^ridge=(X′X+λIp)−1X′Y

Where X = n by p matrix, Y = centered ,n-vector

# LASSO REGRESSION

The lasso regression is very similar to ridge regression. Lasso regression will penalize the absolute size of all the coefficients. This regression is used to increase the accuracy of the linear regression models. Lasso regression reduces the variability as well. Below is the equation for lasso regression.

# EQUATION:

β(lasso)=min (β, β0) (1/N) (Deviance (β, β0) + λ ∑|Bj|

Where N = number of observations, Λ = nonnegative regularization parameter corresponding to value of Lambda, Parameters = β0 and β are scalar and p-vector

These are the algorithm and architecture are used via the machine learning to detect the earthquake prediction model.

# DATA VISUALIZATION:

Data visualization is the representation of data through use of common graphics, such as charts, plots, infographics, and even animations. These visual displays of information communicate complex data relationships and data-driven insights in a way that is easy to understand. In earthquake prediction model I am going to represent the whole world mapwhere the earthquake is frequently happening with the help of the dataset provided by kaggle.com. Further the latitude, longitude, depth, magnitude all are represented in the form of graph and chats.

# START FOR DATA VISUALIZATION

Importing the required modules used for the program for analysis and visualization.

1. matplotlib – to perform the statistical methods and to represent plot.
2. pandas – to perform the analysis of the given data from the dataset.
3. seaborn – to show a graph and chart for the plot representation.

After importing the required modulus we are now going to display the column name for the given dataset as these column names are considered as attributes.

Shape() --function is use to display the dimensions of the dataset. Info()—function is used to print information about the dataset.

Describe() – method is used to display the description of the data in the data frame Basemap() -- allows to create map plots

Drawmapboundary() –allows you to create a boundary for the world map. Drawcountries()—allows you to create a boundary for the countries.

Scatterplot()—it is a diagram where each value in the dataset are represented in as a plot. Subplot() -- method provides a way to plot multiple plots on a single figure.

Boxplot()-- is a method for graphically depicting groups of numerical data through their quartiles. Distplot()—depicts the variation in the database

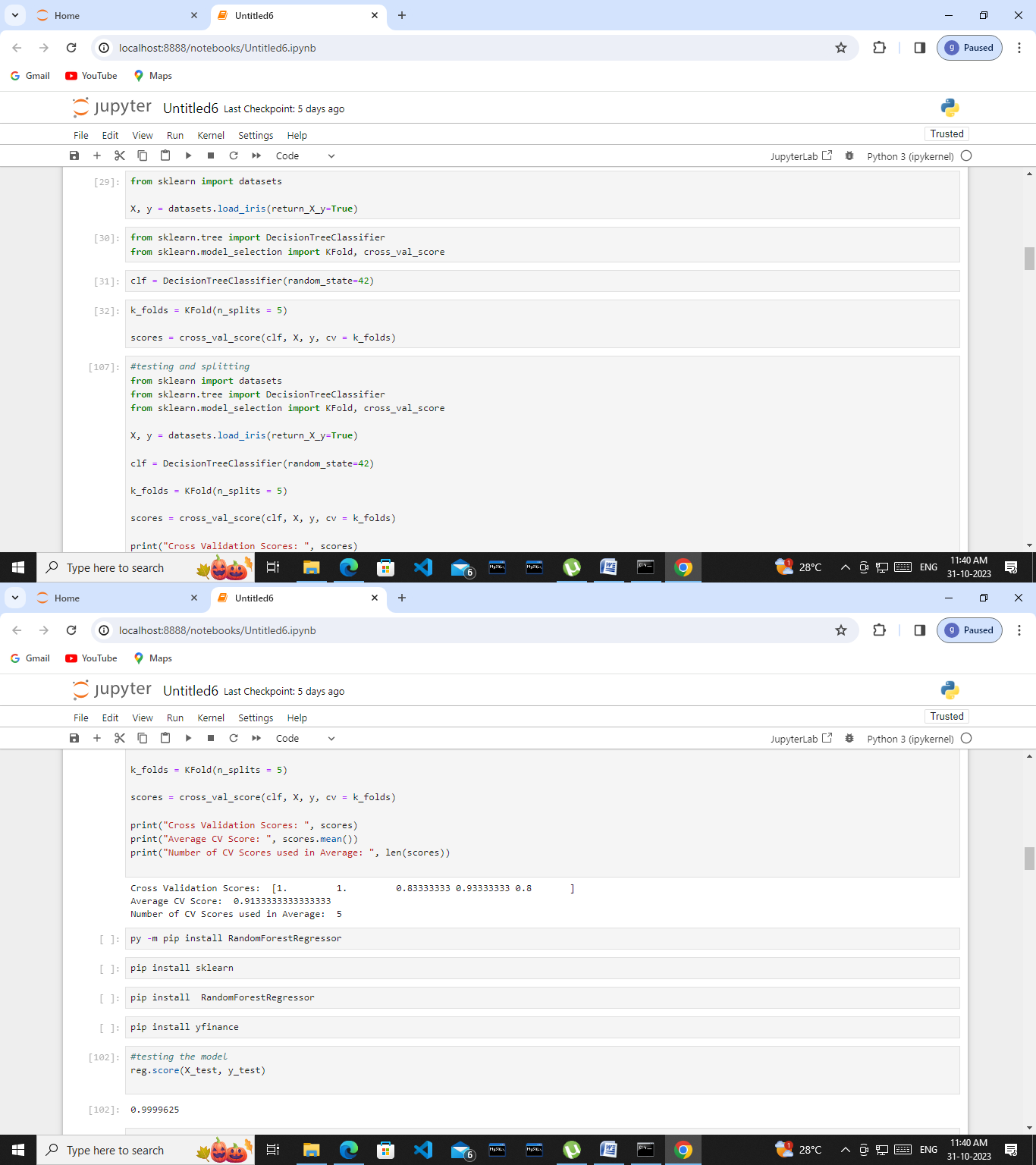
# SPLITTING AND TESTING

Split the dataset into a training and testing set. We will be using the RandomForestRegressor model to predict the earthquake, here will look for its accuracy.

# NEURAL NETWORK MODEL

A neural network model can be employed to forecast earthquakes by examining diverse elements and trends in seismic data. This model harnesses the capabilities of neural networks, which draw inspiration from the neural connection of the human brain, to analyse intricate data and reveal hidden relationships and patterns. By training the neural network on historical earthquake data, it can acquire the ability to identify precursor signals and patterns that indicate the probability of an upcoming earthquake.

# CONCLUSION

Understanding earthquake and effectively responding to them remains a complex and challenging task, even with the latest technological advancements. However, leveraging the capability of machine learning can greatly enhance our comprehension of seismic events. By employing machine learning technique to analyse seismic data, we can uncover valuable insights and patterns that contributes to a deeper understanding of earthquakes. 

These insights can subsequently inform more effective strategies for mitigating risks and responding to seismic events.

As we head towards the future, we might see new technologies that will precisely predict the place and time of the earthquake that will happen.

# FACTOR ANALYSIS IN MACHINE LEARNING

Factor analysis is a statistical approach that seeks to elucidate the variations present in observed data by uncovering latent variables, commonly known as factors. By revealing hidden patterns and relationships that may not be readily discernible, factor analysis offers valuable insights into the intricate structure of complex datasets.

# TYPES OF FACTOR ANALYSIS

1. Exploratory Factor Analysis
2. Confirmatory Factor Analysis

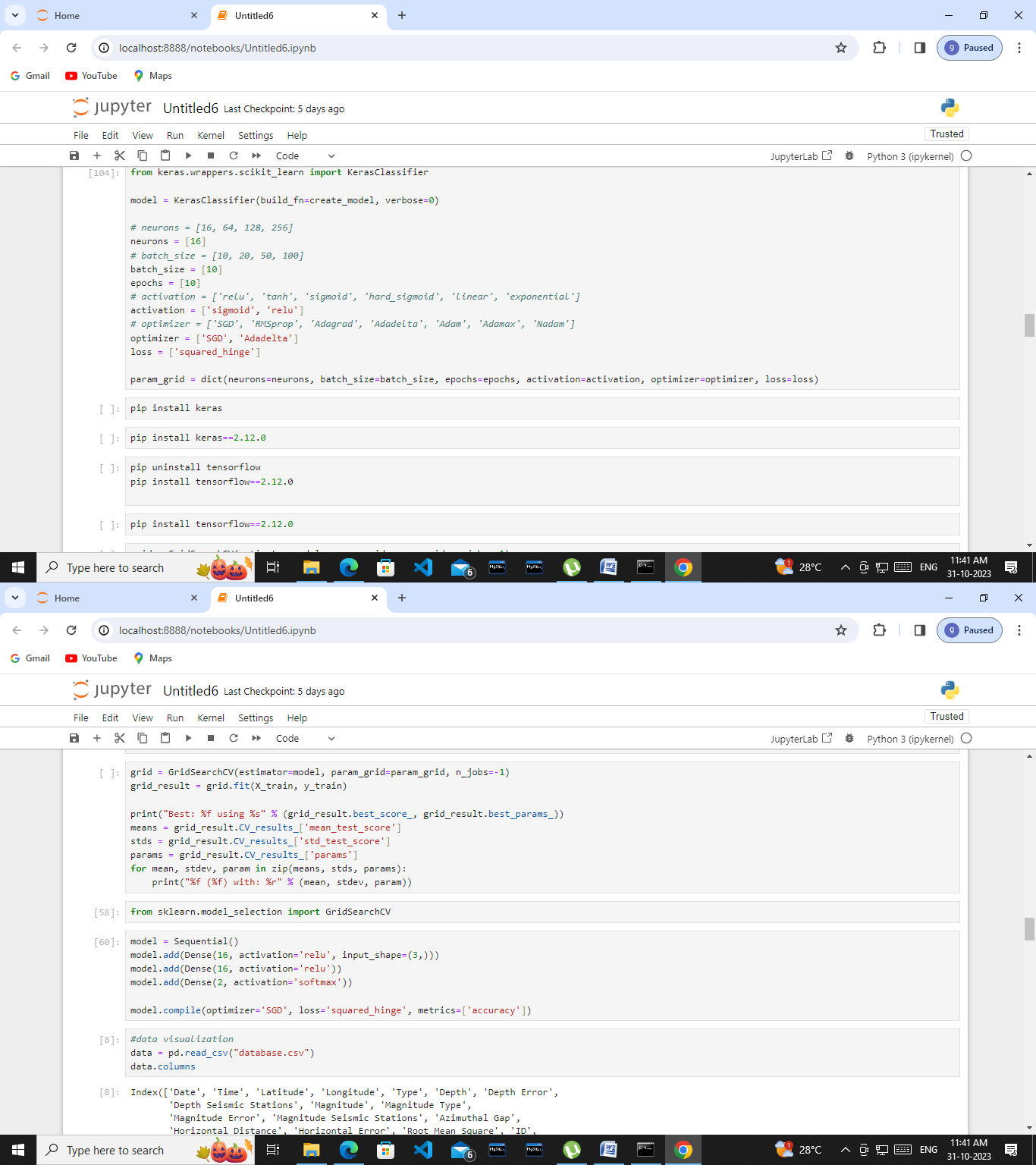
# PRINCIPLES OF FACTOR ANALYSIS

1. Latent Variables
2. Variance Explanation
3. Factor Loading

# BENEFITS OF FACTOR ANALYSIS

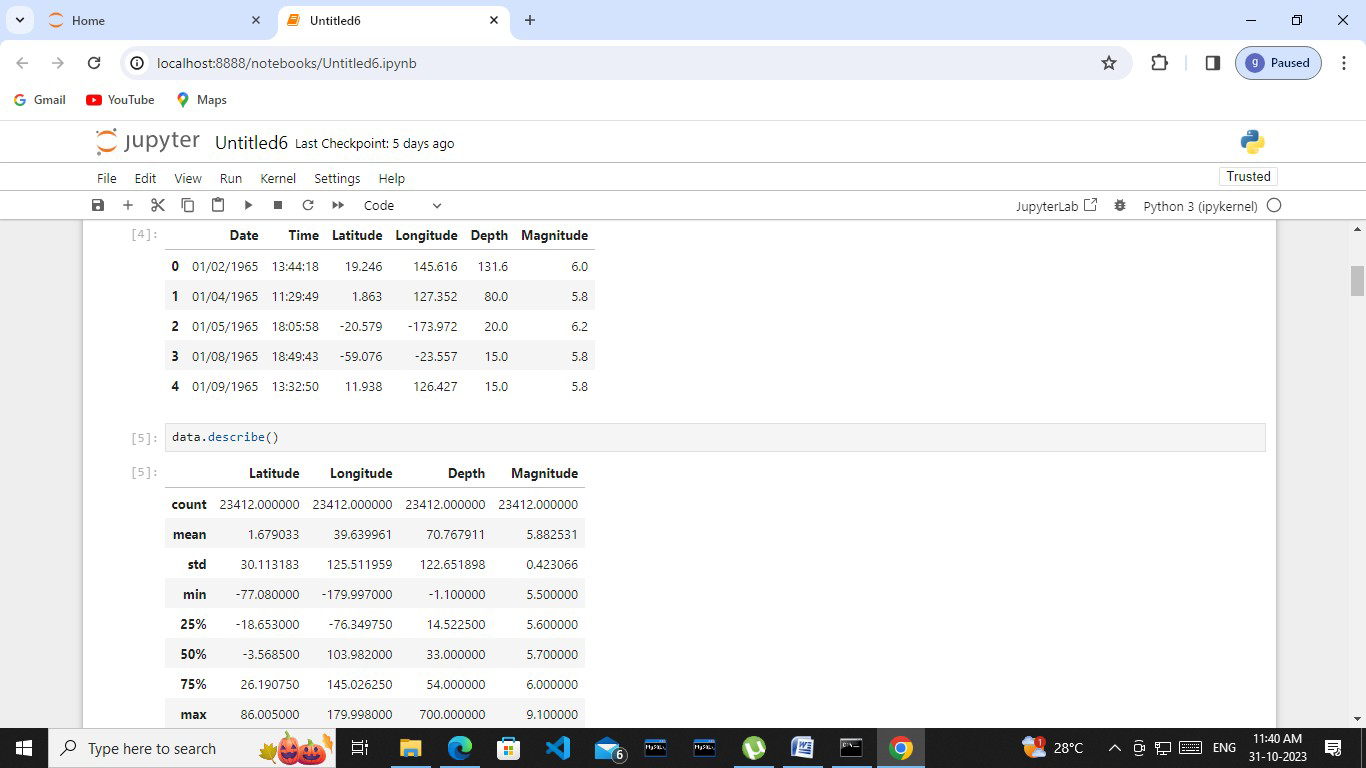
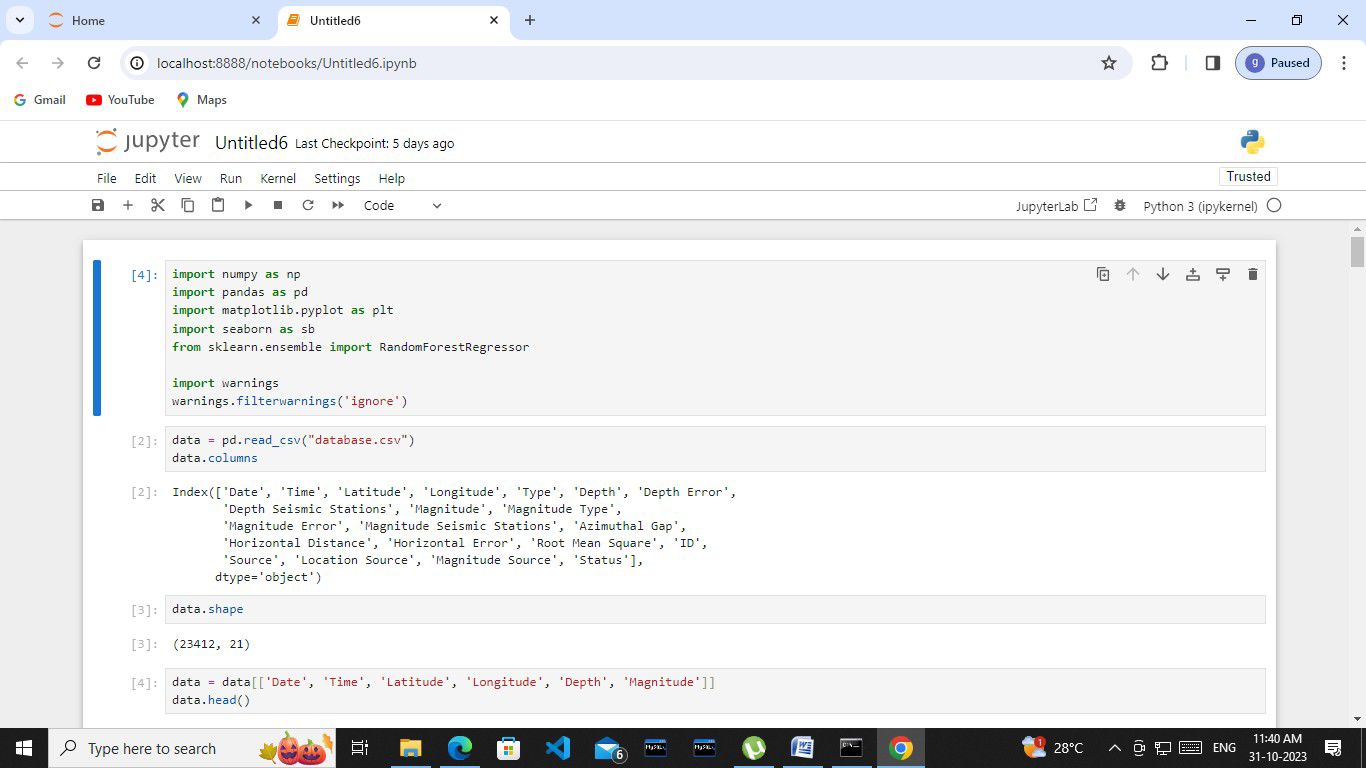
Factor analysis plays a pivotal role in simplifying intricate data by identifying the latent factors that account for the observed patterns. This process facilities a more concise and easily understandable representation of the data, removing unnecessary complexity. By distilling the data down to its essential factors, factor analysis enhance interpretability and enables clear insights into the underlying structure and relationships within the dataset.

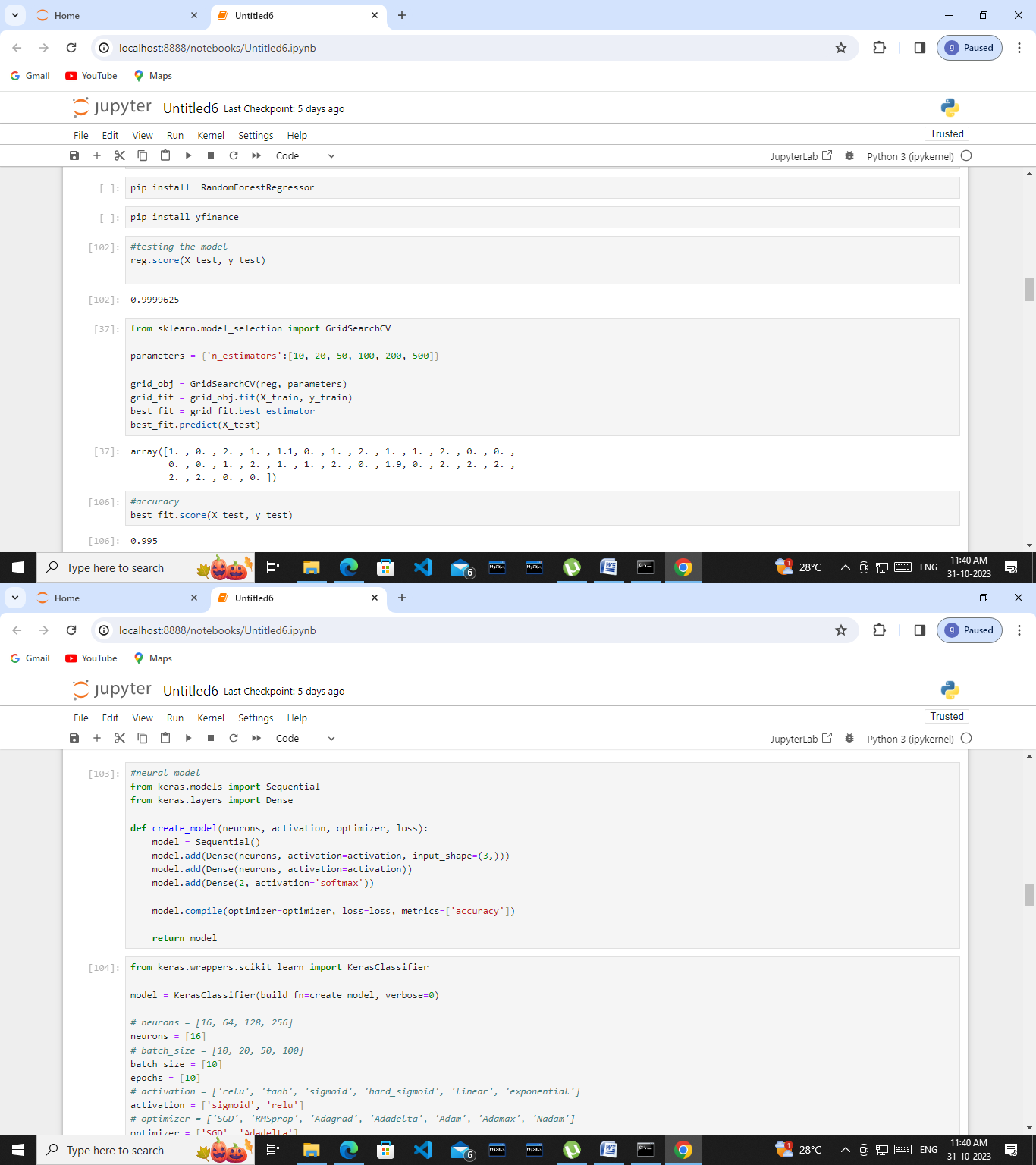
# CONCLUSION OF FACTOR ANALYSIS

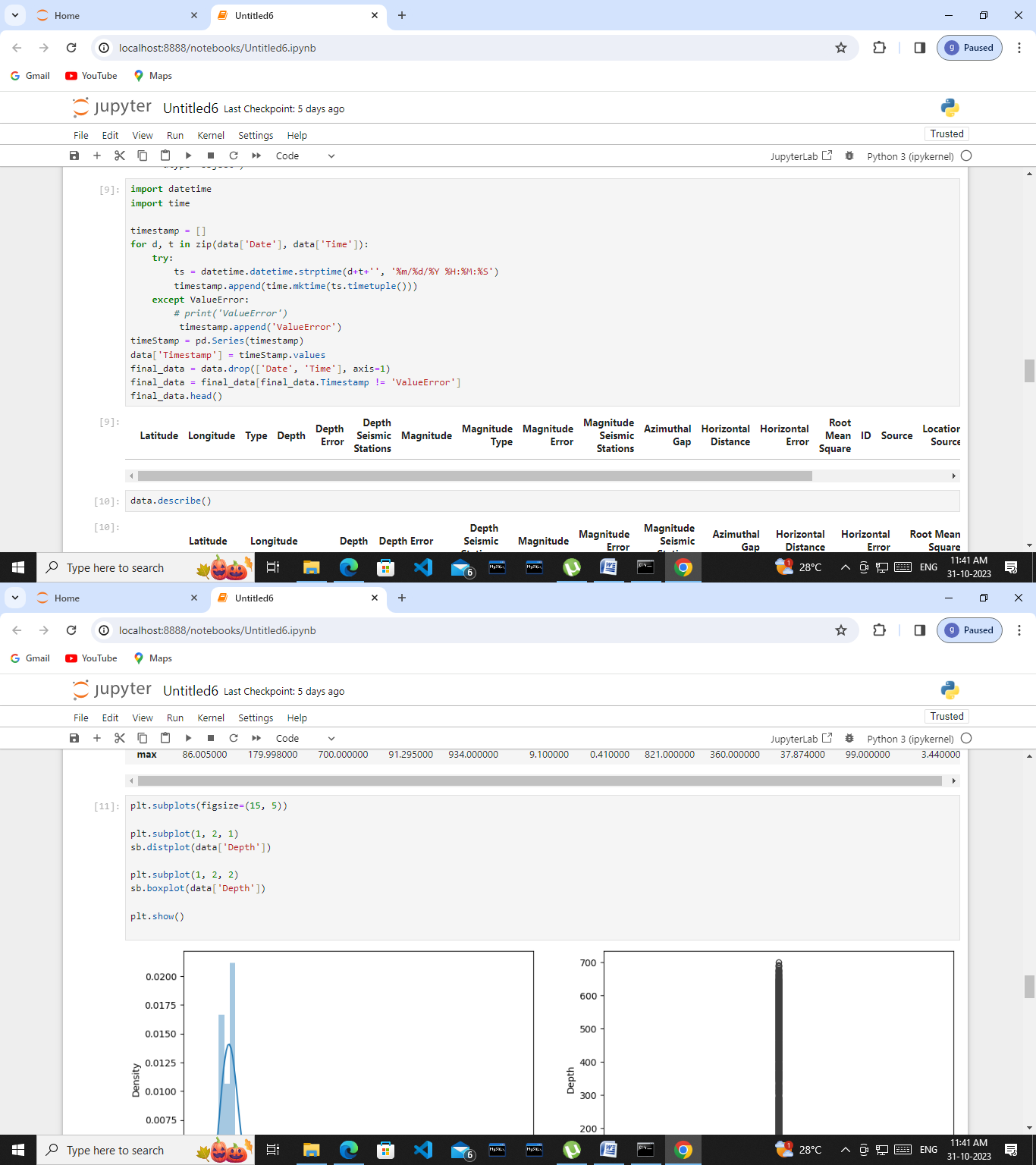
Factor analysis is a vital tool in machine learning that allows for a systematic exploration of the structure within complex datasets. It excels at uncovering latent variables, reducing dimensionality, and revealing valuable insights into relationships. With its versatility and applicability, factor applications in data analysis into the machine learning process, professionals can unveil hidden patterns, enhance predictive capabilities, and gain a profound

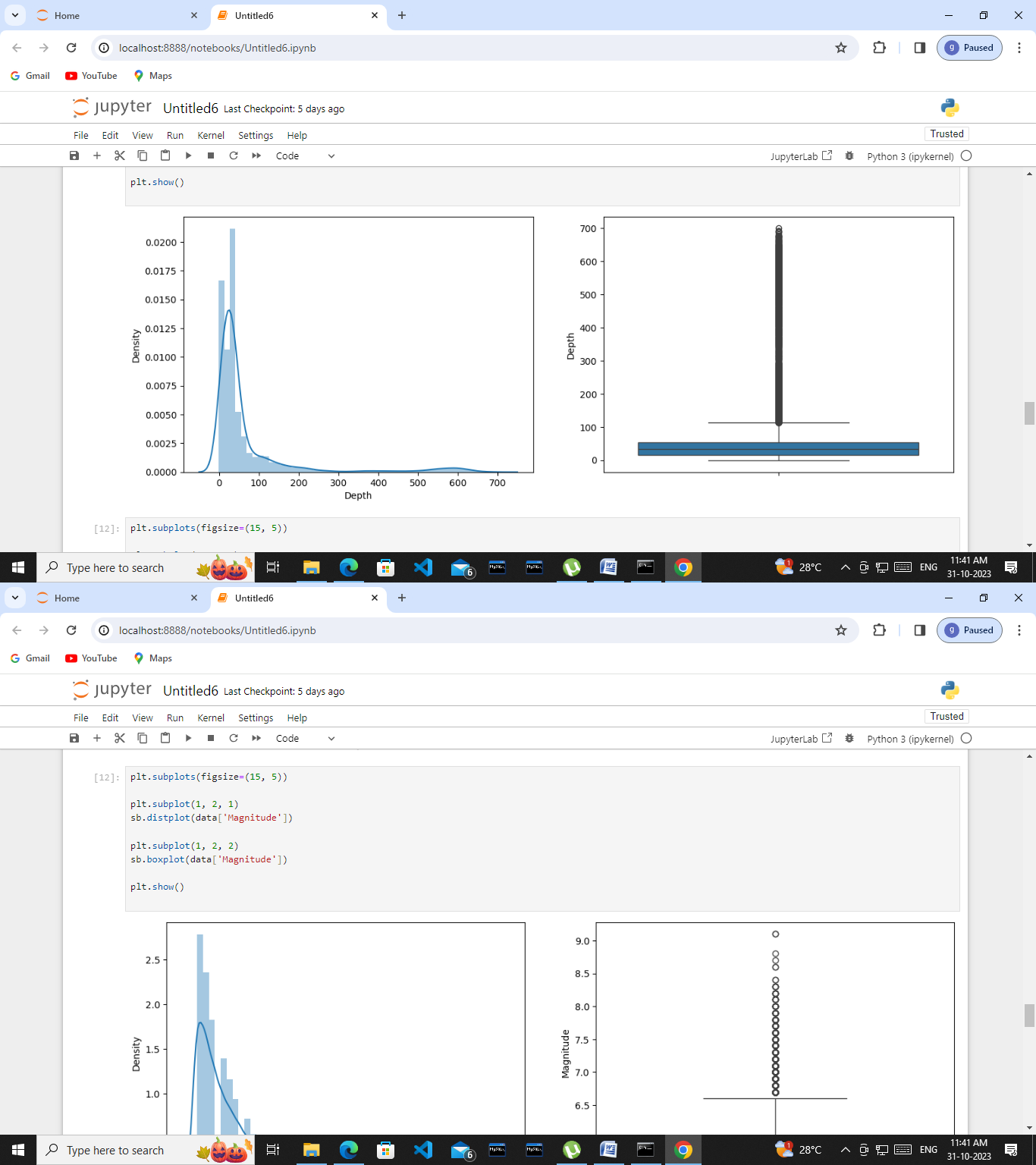
understanding of their data. This, in turn, leads to improved decision-making and innovation across various industries.

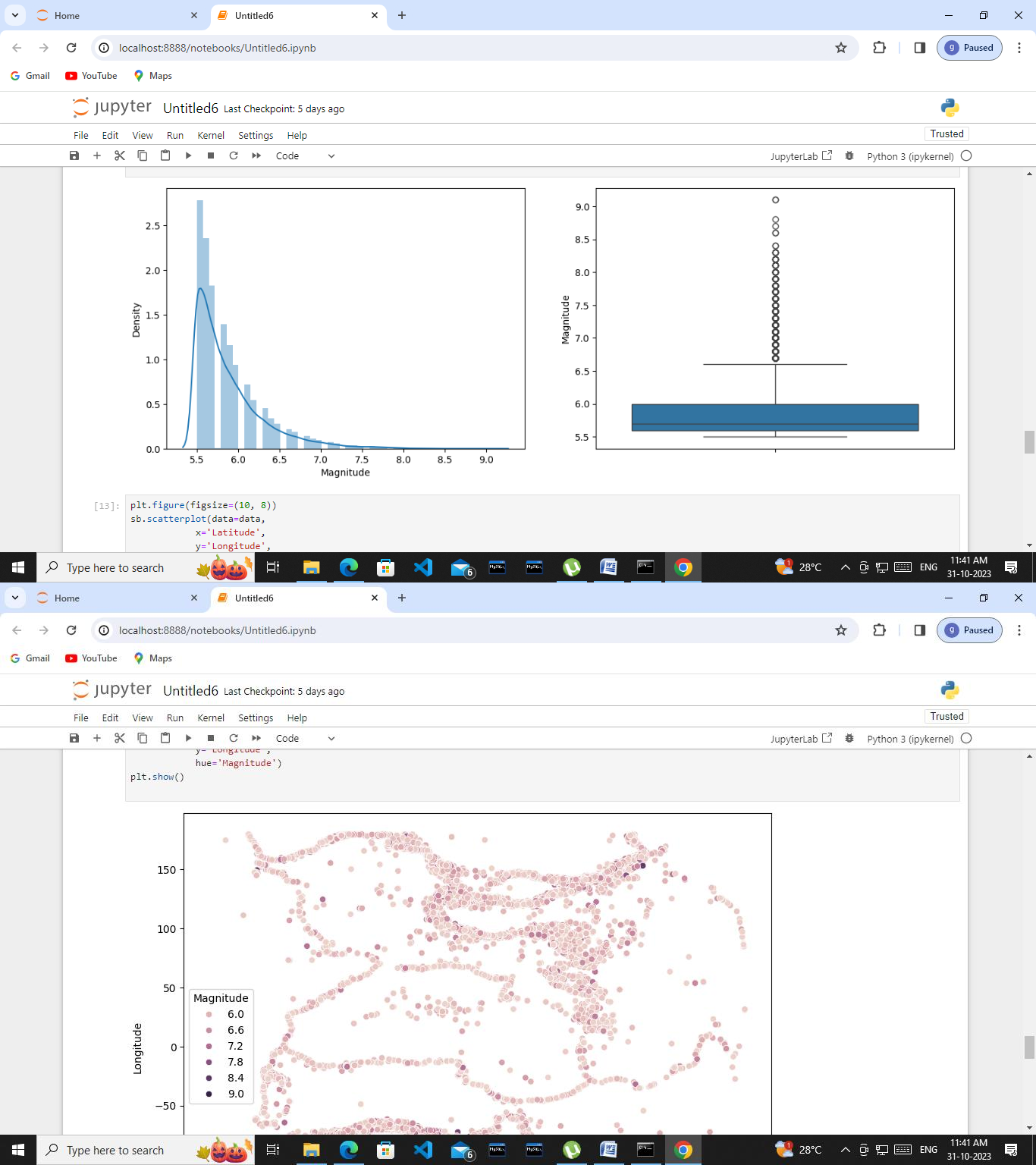
# SOURCE CODE:

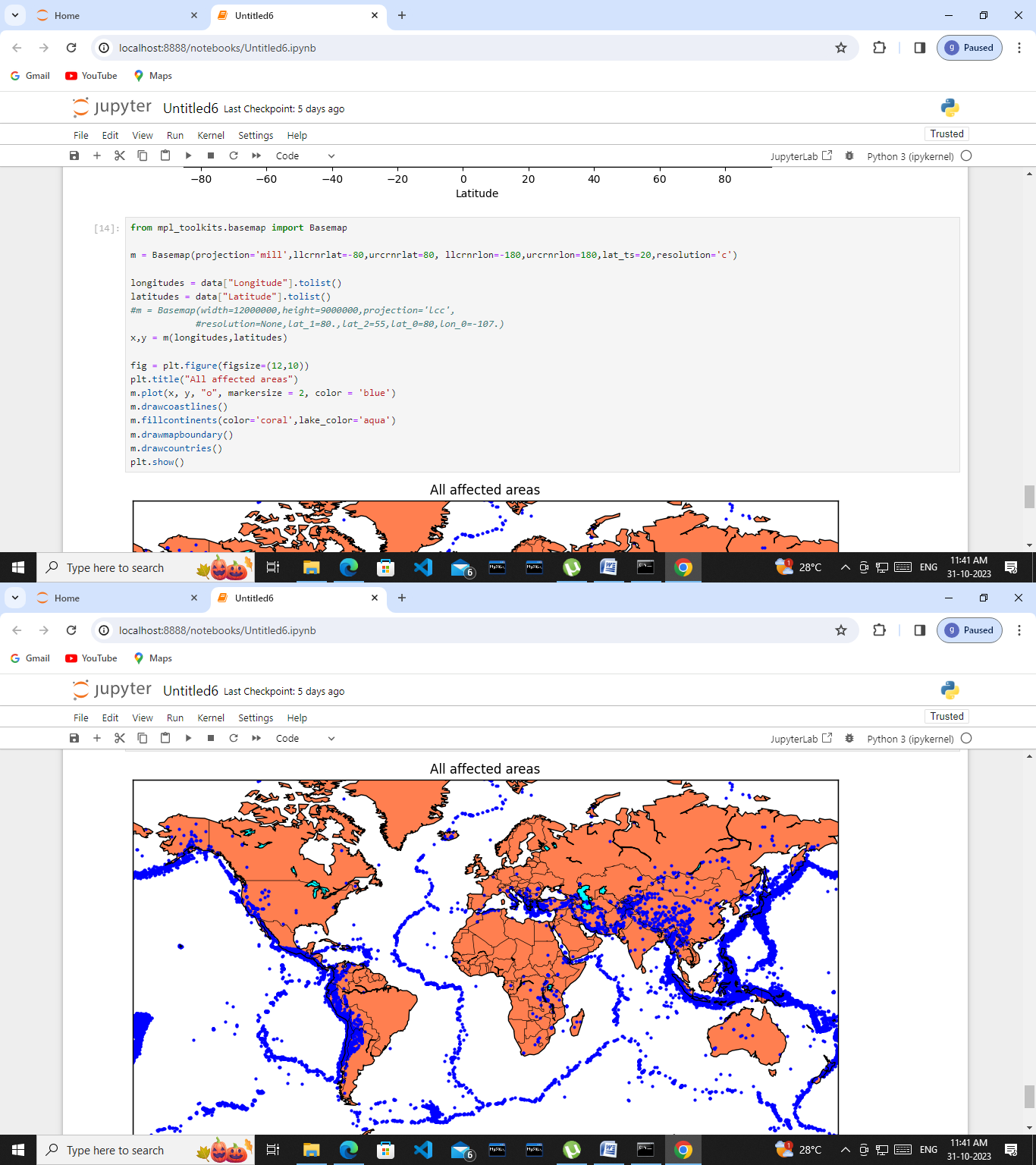












In the above code all the requirement has been excecuted .