PROJECT : EARTHQUAKE PREDICTION USING PYTHON IN MACHINE LEARNING

PHASE - 2

INTRODUCTION :

Earthquakes are disastrous. It takes several months and years to recover from the damage done by the earthquake. It totally destroys everything if the magnitude of the earthquake is very high and can completely ruin a city.

There have been several methods that have been used to predict earthquakes but none have been precise and accurate. The system that I am trying to build is to predict the magnitude of earthquakes based on the historical data set.

The traditional way of analysis is not advisable to use with such big data. The volume of data is too large so there has to be a powerful tool for data analysis.

One of the methods recently used is by using hyperparameter tuning.
Another method used to predict earthquakes is called feature engineering .

DATASOURCE :

A good data source for Earthquake Prediction using machine learning should be accurate, complete, covering the earthquake area and accessible.

CONTENT FOR PROJECT PHASE 2 :

Consider advanced technique such as hyperparameter tuning and feature engineering to improve the prediction model's performance.

Seismic Data Analysis:

Seismic Waveform Analysis: Use deep learning models to analyse seismic waveform data. Convolutional neural networks (CNNs) can be applied to detect patterns or anomalies in seismic signals.

IoT and Sensor Networks:

Sensor Data Integration: Utilise data from IoT devices and sensor networks that measure ground movements, temperature, pressure, and other environmental factors. This can help in creating a more comprehensive dataset for prediction.

Deep Learning Techniques:

Recurrent Neural Networks (RNNs): RNNs can be used for time-series analysis, making them suitable for predicting earthquake patterns over time.

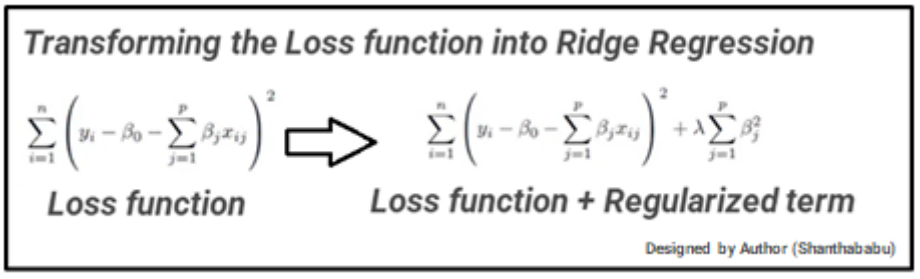
Anomaly Detection:

Unsupervised Learning: Use unsupervised machine learning techniques such as clustering and autoencoders to identify unusual seismic activity that might precede an earthquake.

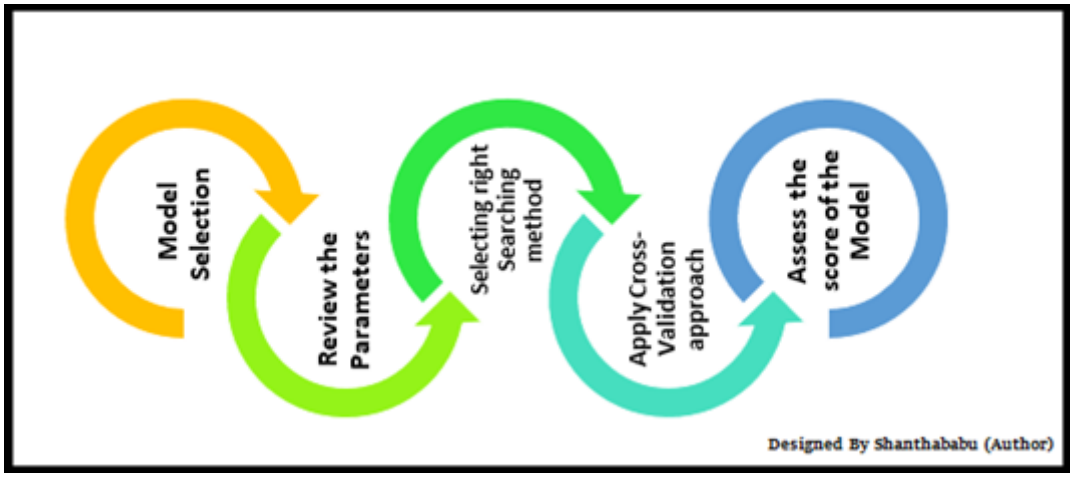
Hyperparameter: As we know that there are parameters that are internally learned from the given dataset and derived from the dataset, they are represented in making predictions, classification and etc., These are so-called Model Parameters, and they are varying with respect to the nature of the data we couldn't control this since it depends on the data. Like 'm' and 'C' in linear equation, which is the value of coefficients learned from the given dataset.Some set of parameters that are used to control the behaviour of the model/algorithm and adjustable in order to obtain an improvised model with optimal performance is so-called Hyperparameters.

Hyperparameter tuning: Hyperparameter tuning is basically referred to as tweaking the parameters of the model, which is basically a prolonged process.

Ridge regression: Ridge regression is a model tuning method that is used to analyse any data that suffers from multicollinearity. This method performs L2 regularization. When the issue of multicollinearity occurs, least-squares are unbiased, and variances are large, this results in predicted values being far away from the actual values.



Steps to perform hyperparameter tuning:



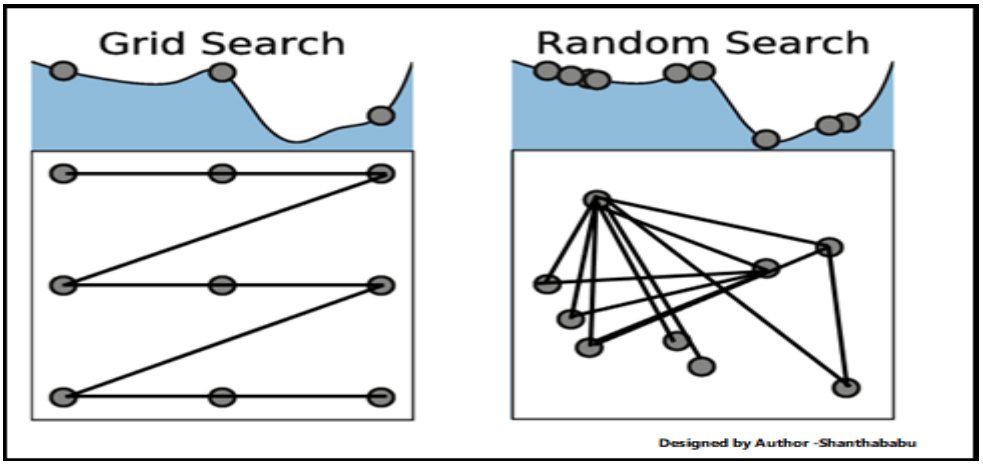
Hyperparameter Optimization Techniques :

In the ML world, there are many Hyperparameter optimization techniques are available.

- Manual Search
- Random Search
- Grid Search
- Halving
  - Grid Search
  - Randomised Search
- Automated Hyperparameter tuning
  - Bayesian Optimization
  - Genetic Algorithms
- Artificial Neural Networks Tuning
- HyperOpt-Sklearn
- Bayes Search

Comparison Study of GridSearchCV and RandomSearch CV

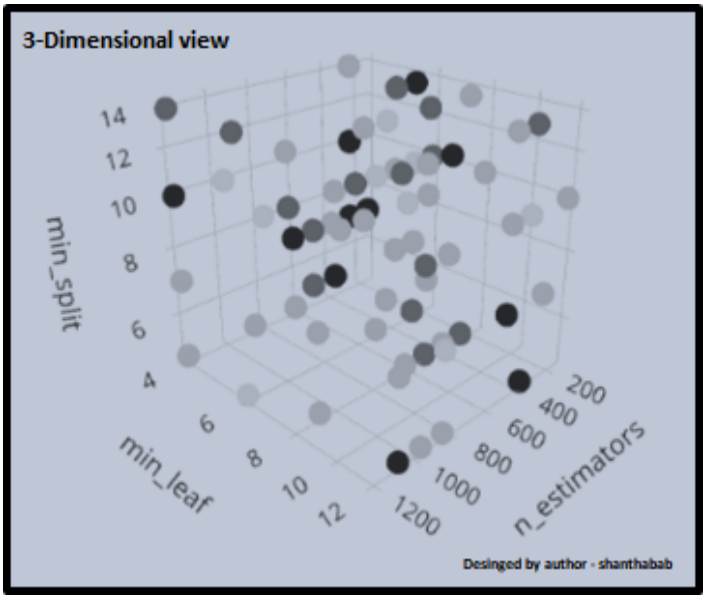
GridSearch CV	Random Search CV
Grid is well-defined	Grid is not well defined
Discrete values for HP-params	Continuous values and Statistical distribution
Defined size for Hyperparameter space	No such a restriction
Picks of the best combination from HP-Space	Picks up the samples from HP-Space
Samples are not created	Samples are created and specified by the range and n_iter
Low performance than RSCV	Better performance and result
Guided flow to search for the best combination.	The name itself says that, no guidance.



Hyperparameter Space:

As we know that there is a list of HPs for any selected algorithm(s) and our job is to figure out the best combination of HPs and to get the optimal results by tweaking them strategically, this process will be providing us with the platform for Hyperparameter Space and this combination leads to provide the best optimal results, no doubt in that but finding this combo is not so easy, we have to search throughout the space.

Here every combination of selected HP value is said to be the "MODEL" and have to evaluate the same on the spot. For this reason, there are two generic approaches to search effectively in the HP space are GridSearch CV and RandomSearch CV. Here CV denotes Cross-Validation.

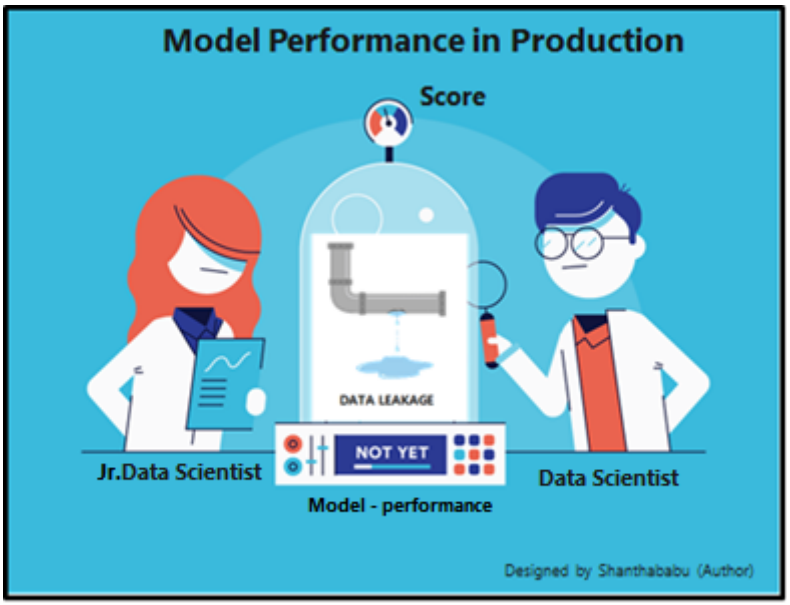


Data Leakage: Now quickly will understand what is Data leakage in ML, this is mainly due to not following some of the recommended best practices during the Data Science/Machine Learning life cycle. The resulting is Data Leakage, that'sfine, what is the issue here, after successful testing with perfect accuracy

followed by training the model then the model has been planned to move into production.

Causes of Data Leakage :

- Data Pre-processing
- The major root cause is doing all EDA processes before splitting the dataset into test and train
- Doing straightforward normalizing or rescaling on a given dataset
- Performing Min/Max values of a feature
- Handling missing values without reserving the test and train
- Removing outliers and Anomaly on a given dataset
- Applying standard scaler, scaling, assert normal distribution on the full dataset



Data collection : Gather relevant data such as seismic wave patterns, geological information, historical earthquake occurrences, etc. This data can be sourced from geological surveys and seismic monitoring stations.

Processing : Preprocess the data to remove noise and transform it into a usable format. This may include normalisation, feature extraction, and filling or removing missing values.

Feature engineering : Identify significant features or variables that are likely to have an impact on earthquake prediction. This may include the velocity of seismic waves, the depth of seismic activity, and the location of faults.

Dataset link :

https://www.kaggle.com/datasets/usgs/earthquake-database

date	depth	mag	place	latitude	longitude	depth_avg_22	depth_avg_15	depth_avg_7	mag_avg_22	mag_avg_15	mag_avg_7	mag_outcome
2020-07-14	6.70	1.58	Oklahoma	36.171483	-97.718347	6.717727	6.560000	7.100000	1.352273	1.271333	1.357143	1.338571
2020-07-14	7.55	2.07	Oklahoma	36.171483	-97.718347	6.730000	6.682667	7.132857	1.372727	1.334667	1.527143	1.535714
2020-07-14	7.39	1.89	Oklahoma	36.171483	-97.718347	6.747727	6.708667	6.940000	1.396818	1.377333	1.570000	1.335714
2020-07-15	7.75	1.48	Oklahoma	36.171483	-97.718347	6.834545	6.764000	6.848571	1.383182	1.388667	1.581429	1.251429
2020-07-15	7.81	1.50	Oklahoma	36.171483	-97.718347	6.841364	6.854667	6.964286	1.404545	1.385333	1.602857	1.291429

Code :

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sb
import warnings
warnings.filterwarnings('ignore')
```

```
df = pd.read_csv('dataset.csv')
df.head()
```

	Origin Time	Latitude	Longitude	Depth	Magnitude	Location
0	2021-07-31 09:43:23 IST	29.06	77.42	5.0	2.5	53km NNE of New Delhi, India
1	2021-07-30 23:04:57 IST	19.93	72.92	5.0	2.4	91km W of Nashik, Maharashtra, India
2	2021-07-30 21:31:10 IST	31.50	74.37	33.0	3.4	49km WSW of Amritsar, Punjab, India
3	2021-07-30 13:56:31 IST	28.34	76.23	5.0	3.1	50km SW of Jhajjar, Haryana
4	2021-07-30 07:19:38 IST	27.09	89.97	10.0	2.1	53km SE of Thimphu, Bhutan

The dataset we are using here contains data for the following columns:

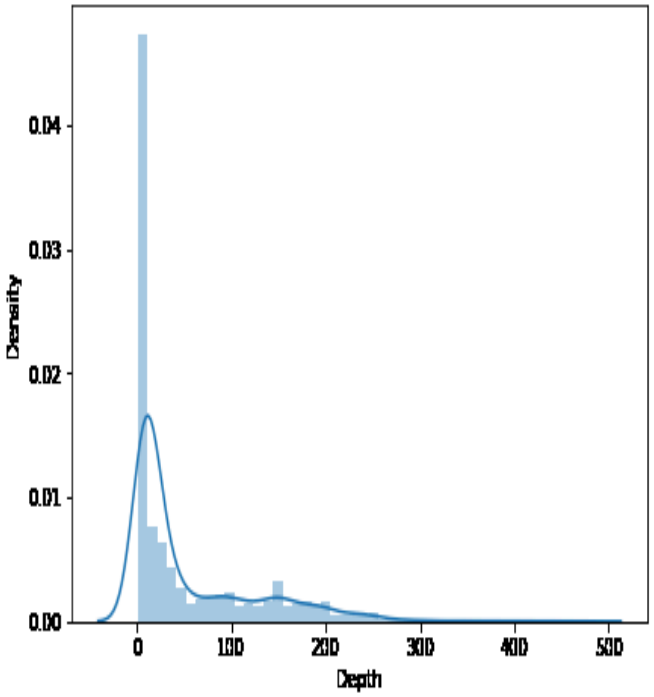
- Origin time of the Earthquake
- Latitude and the longitude of the location.
- Depth – This means how much depth below the earth's level the earthquake started.
- The magnitude of the earthquake
- Location

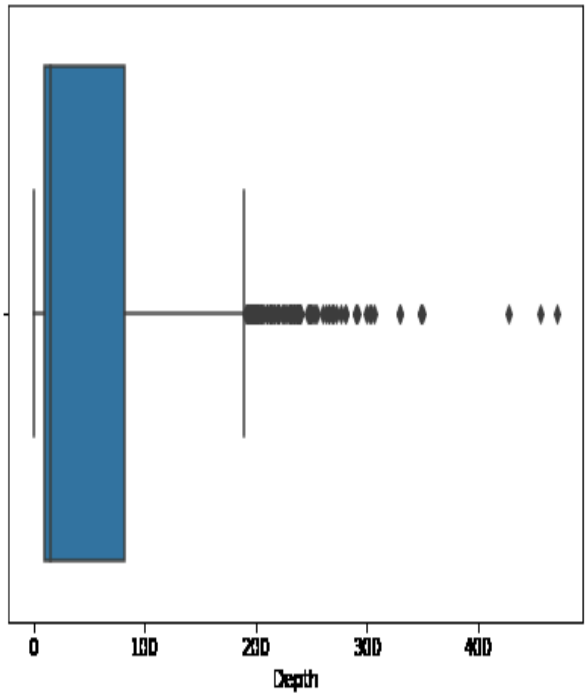
Here we can observe that the changes of an earthquake with higher magnitude are more observed during the season of monsoon.

Code :

```
plt.subplots(figsize=(15, 5))
plt.subplot(1, 2, 1)
sb.distplot(df['Depth'])
```

```
plt.subplot(1, 2, 2)
sb.boxplot(df['Depth'])
plt.show()
```

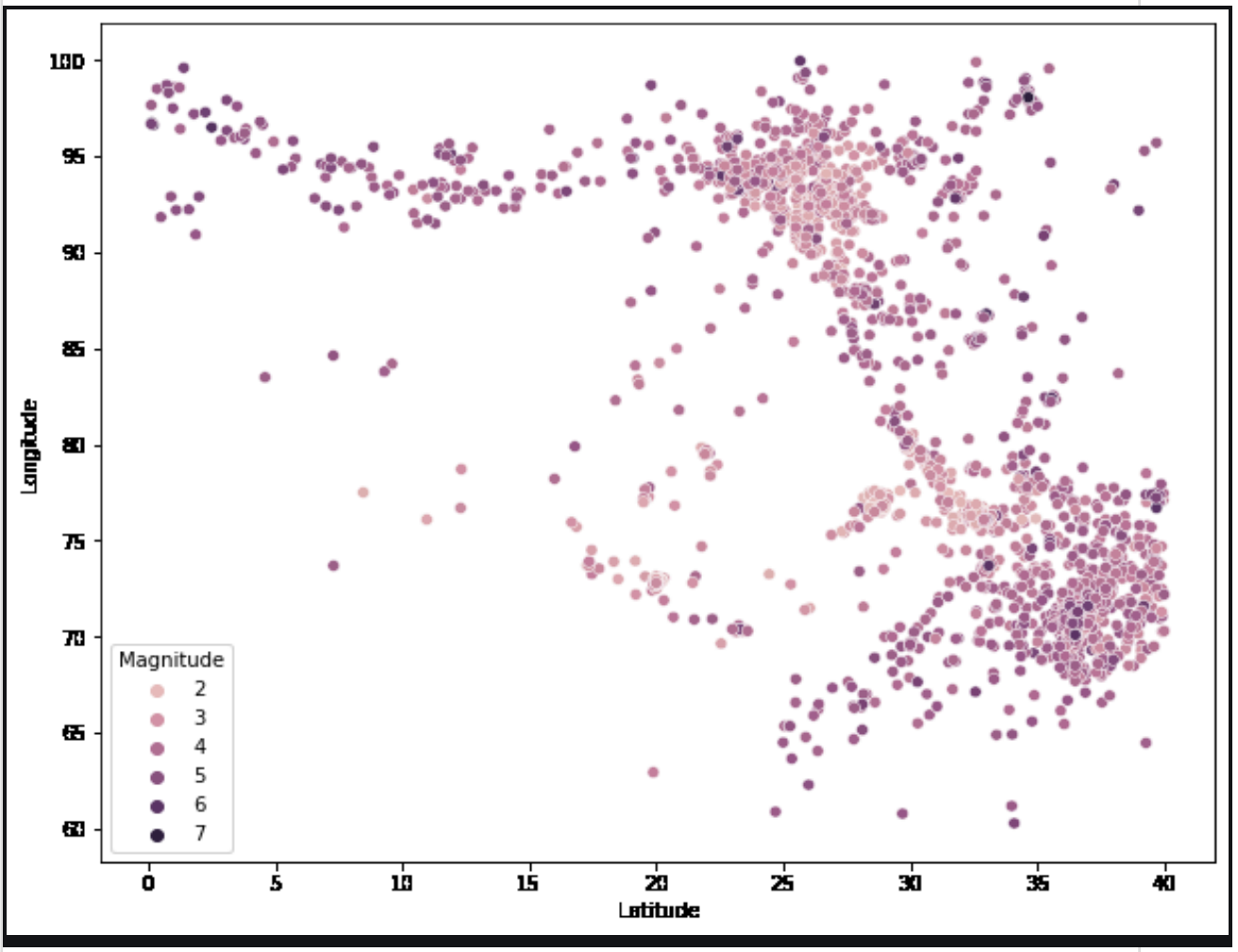




From the distribution graph, it is visible that there are some outliers that can be confirmed by using the boxplot. But the main point to observe here is that the distribution of the depth at which the earthquake rises is left-skewed.

Code :

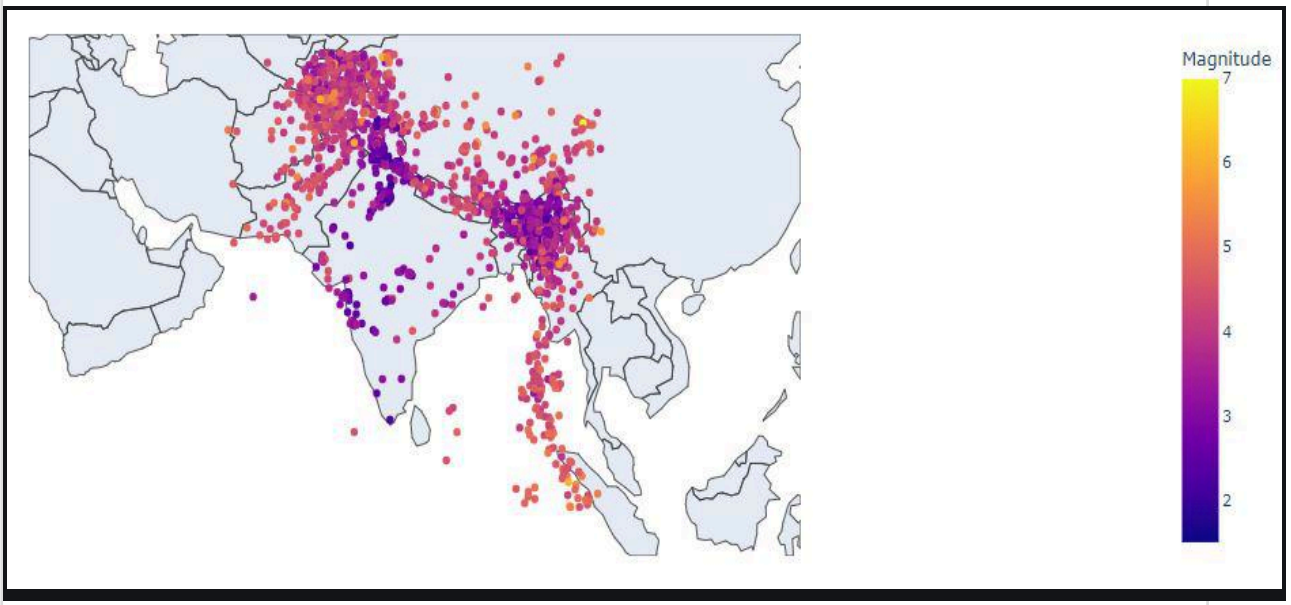
```
plt.figure(figsize=(10, 8))
sb.scatterplot(data=df,
               x='Latitude',
               y='Longitude',
               hue='Magnitude')
plt.show()
```



Now by using Plotly let's plot the latitude and the longitude data on the map to visualise which areas are more prone to earthquakes.

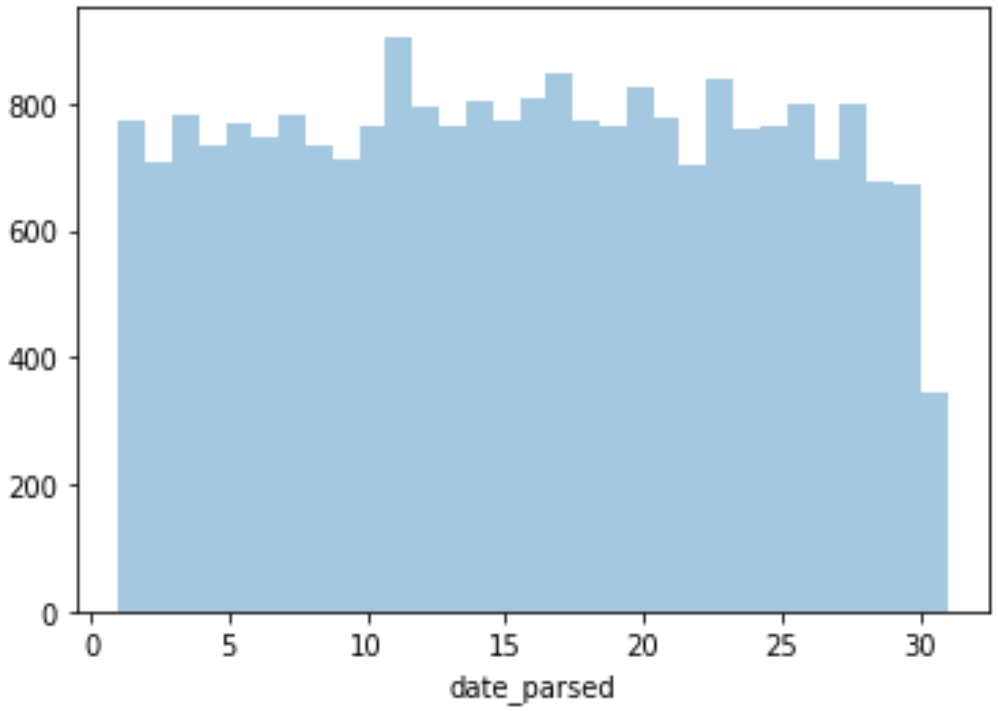
Code :

```
import plotly.express as px
import pandas as pd
fig = px.scatter_geo(df, lat='Latitude',
                     lon='Longitude',
                     colour="Magnitude",
                     fitbounds='locations',
                     scope='asia')
fig.show()
```



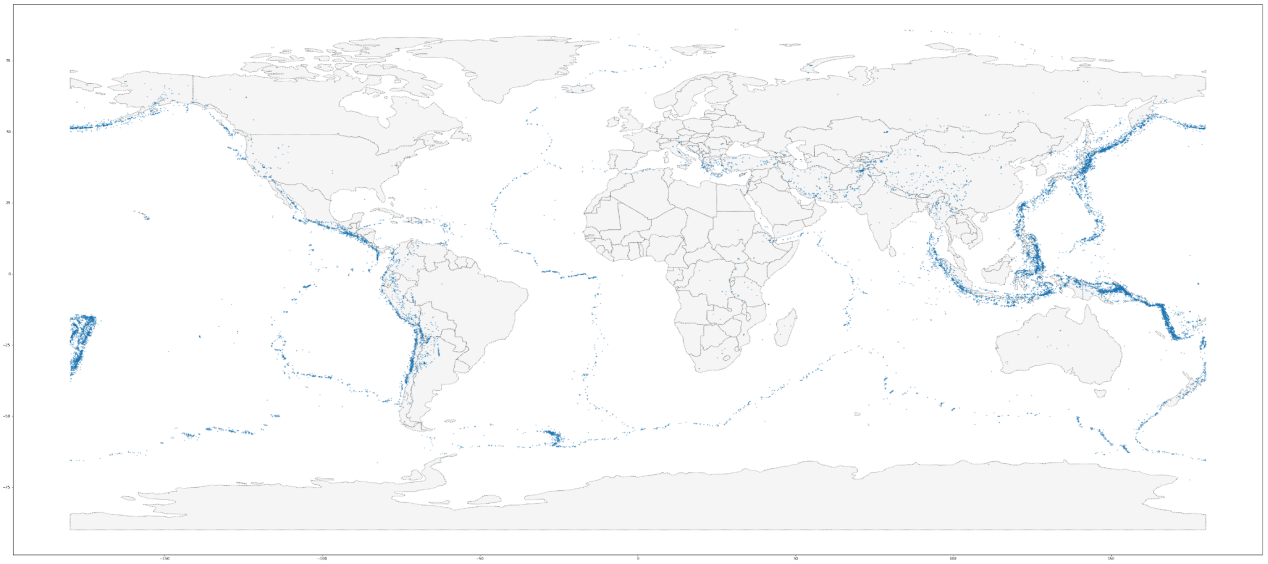
Code :

```
day_of_month_earthquakes = day_of_month_earthquakes.dropna()
sns.distplot(day_of_month_earthquakes, kde=False, bins=31)
```



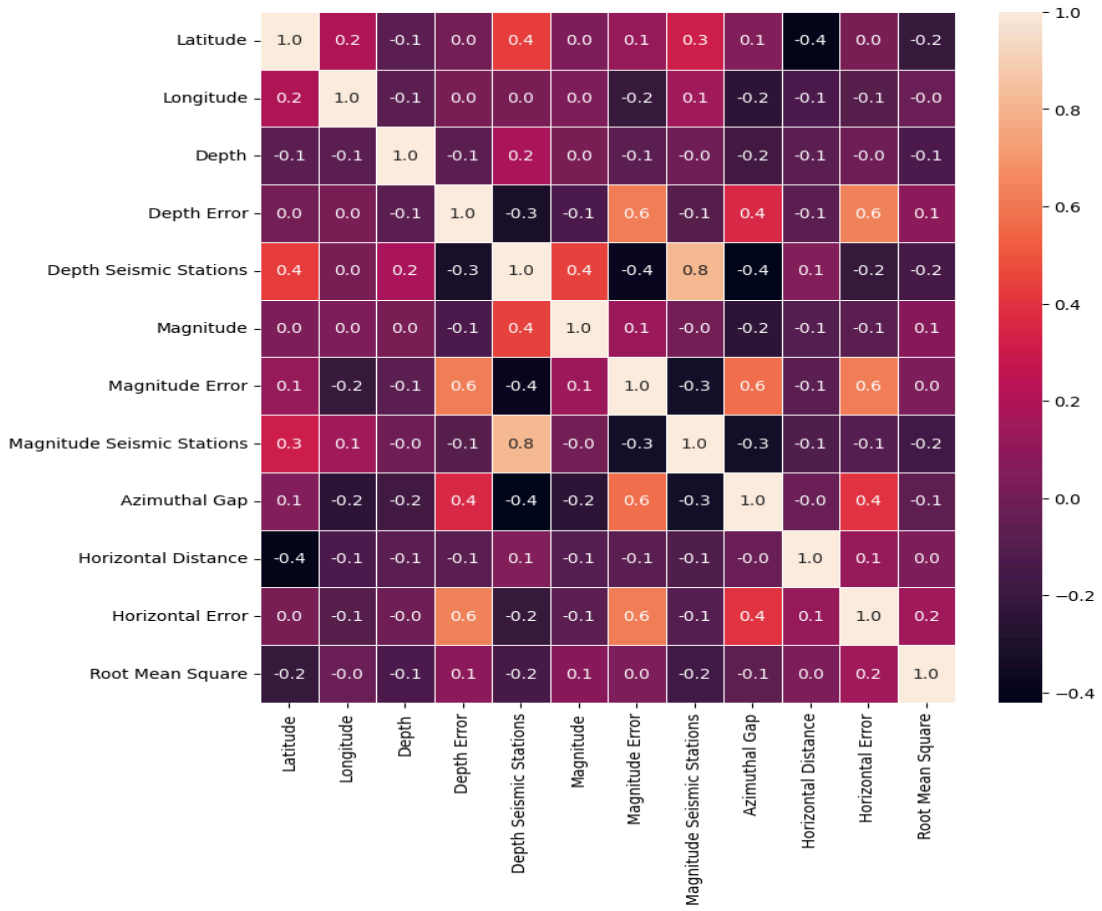
Code :

```
ax = world.plot(figsize=(60,60), color='whitesmoke', linestyle=':',
edgecolor='black')
earthquakes.to_crs(epsg=4326).plot(markersize=1, ax=ax)
```



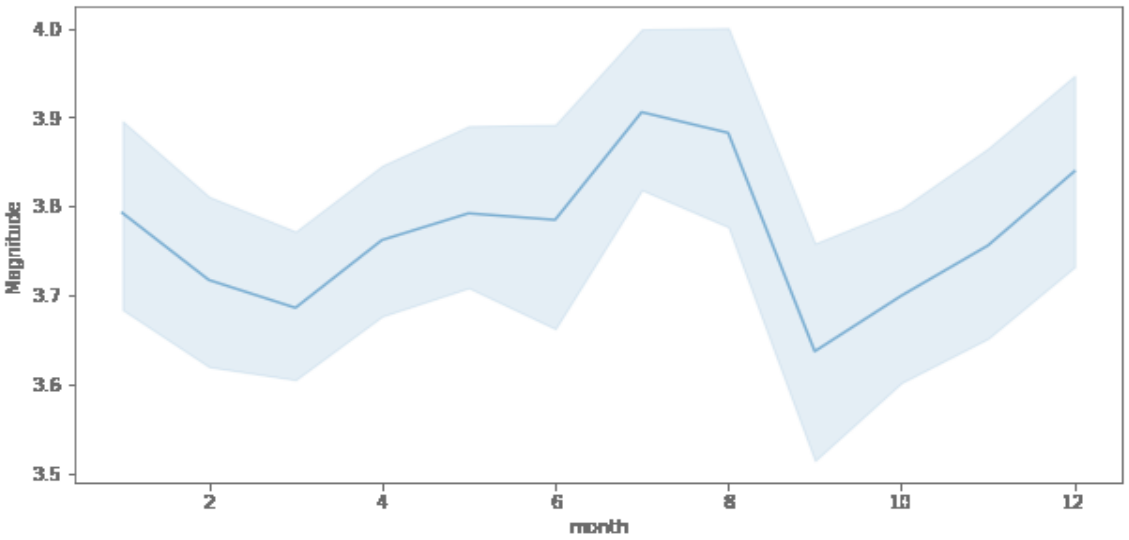
Code :

```
f,ax = plt.subplots(figsize=(9, 9))
sns.heatmap(data.corr(), annot=True, linewidths=.5, fmt= '.1f',ax=ax)
plt.show()
```



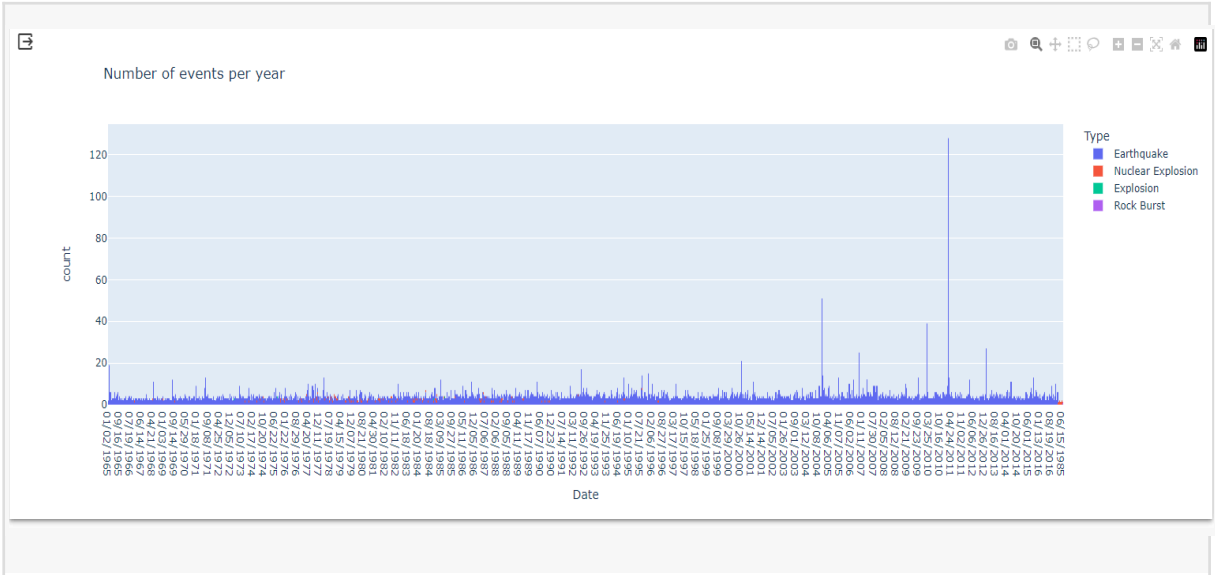
Code :

```
plt.figure(figsize=(10, 5))
sb.lineplot(data=df, x='month', y='Magnitude')
plt.show()
```



Code :

```
fig = px.histogram(data,'Date',color="Type",title="Number of events per year")
fig.show()
```



CONCLUSION :

- In the phase 2 conclusion, we will summarise the key findings and insights from the advanced feature engineering and hyperparameter tuning techniques.
- We will reiterate the impact of these techniques on improving the accuracy and robustness of earthquake prediction.
- we have discussed in a detailed study of Hyperparameter visions with respect to the Machine Learning point of view, please remember a few things before we go

- Each model has a set of hyperparameters, so we have carefully chosen them and tweaked them during hyperparameter tuning.
- I mean building the HP space.
- All hyperparameters are NOT equally important and there are no defined rules for this. try to use continuous values instead of discrete values.
- Make sure to use K-Fold while using Hyperparameter tuning to improvise your hyperparameter tuning and coverage of hyperparameter space.
- Go with a better combination for hyperparameters and build strong results.