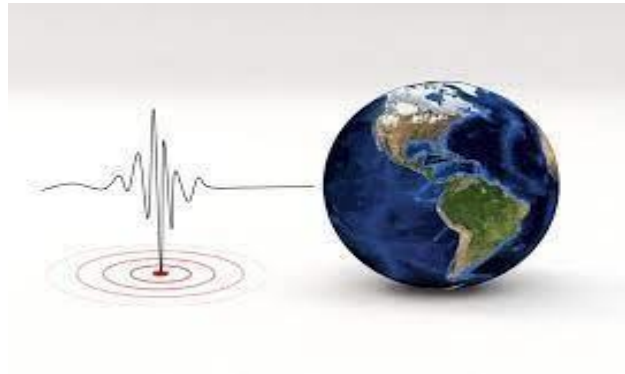


Earthquake prediction model using python



Problem Statement:

Earthquakes are one of the most destructive and unpredictable natural disasters in the world, causing significant damage to infrastructure, property, and human life. While scientists have made significant progress in predicting earthquakes, there is still much that is unknown about these events, including their magnitude, location, and timing.

The goal of this project is to develop an accurate and reliable machine learning model for predicting the severity of earthquakes based on a range of input variables, including seismic activity, geographical location, and historical earthquake data. To achieve this, the project will employ a range of ensemble techniques and perform cross-validation to evaluate the performance of the model.

By using ensemble techniques and cross-validation, the project aims to develop a highly accurate and reliable model that can be used to improve earthquake preparedness and response efforts in areas prone to seismic activity. Overall, this project aims to use advanced machine learning techniques to develop a highly accurate and reliable model for earthquake prediction. By doing so, it hopes to contribute to our understanding of these complex natural events and to ultimately help save lives and minimize the damage caused by earthquakes.

Dataset Link: <https://www.kaggle.com/datasets/usgs/earthquake-database>

```
import numpy as np
import pandas as pd
```

```
# Load the dataset
data = pd.read_csv('database.csv')
data.head()
```



	Date	Time	Latitude	Longitude	Type	Depth	Depth Error	Depth Seismic Stations	Magnitude	Magnitude Type	...	Magnitude Seismic Stations	Azimuthal Gap	Hori Di
0	01/02/1965	13:44:18	19.246	145.616	Earthquake	131.6	NaN	NaN	6.0	MW	...	NaN	NaN	
1	01/04/1965	11:29:49	1.863	127.352	Earthquake	80.0	NaN	NaN	5.8	MW	...	NaN	NaN	
2	01/05/1965	18:05:58	-20.579	-173.972	Earthquake	20.0	NaN	NaN	6.2	MW	...	NaN	NaN	
3	01/08/1965	18:49:43	-59.076	-23.557	Earthquake	15.0	NaN	NaN	5.8	MW	...	NaN	NaN	
4	01/09/1965	13:32:50	11.938	126.427	Earthquake	15.0	NaN	NaN	5.8	MW	...	NaN	NaN	

5 rows × 21 columns

```
missing_values = data.isnull().sum()
missing_values
```

```
Date          0
Time          0
Latitude      0
Longitude     0
Type          0
Depth         0
Depth Error   18951
Depth Seismic Stations 16315
Magnitude     0
Magnitude Type 3
Magnitude Error 23085
Magnitude Seismic Stations 20848
Azimuthal Gap 16113
Horizontal Distance 21808
Horizontal Error 22256
Root Mean Square 6060
ID            0
Source        0
Location Source 0
Magnitude Source 0
Status       0
dtype: int64
```

```
data.columns
```

```
Index(['Date', 'Time', 'Latitude', 'Longitude', 'Type', 'Depth', 'Depth Error',
       'Depth Seismic Stations', 'Magnitude', 'Magnitude Type',
       'Magnitude Error', 'Magnitude Seismic Stations', 'Azimuthal Gap',
       'Horizontal Distance', 'Horizontal Error', 'Root Mean Square', 'ID',
       'Source', 'Location Source', 'Magnitude Source', 'Status'],
      dtype='object')
```

```
data = data[['Date', 'Time', 'Latitude', 'Longitude', 'Depth', 'Magnitude']]
data.head()
```

	Date	Time	Latitude	Longitude	Depth	Magnitude
0	01/02/1965	13:44:18	19.246	145.616	131.6	6.0
1	01/04/1965	11:29:49	1.863	127.352	80.0	5.8
2	01/05/1965	18:05:58	-20.579	-173.972	20.0	6.2
3	01/08/1965	18:49:43	-59.076	-23.557	15.0	5.8
4	01/09/1965	13:32:50	11.938	126.427	15.0	5.8

```
import datetime
import time

timestamp = []
for d, t in zip(data['Date'], data['Time']):
    try:
        ts = datetime.datetime.strptime(d+' '+t, '%m/%d/%Y %H:%M:%S')
        timestamp.append(time.mktime(ts.timetuple()))
    except ValueError:
```

```
# print('ValueError')
timestamp.append('ValueError')
```

```
timeStamp = pd.Series(timestamp)
data['Timestamp'] = timeStamp.values
```

```
final_data = data.drop(['Date', 'Time'], axis=1)
final_data = final_data[final_data.Timestamp != 'ValueError']
final_data.head()
```

	Latitude	Longitude	Depth	Magnitude	Timestamp
0	19.246	145.616	131.6	6.0	-157630542.0
1	1.863	127.352	80.0	5.8	-157465811.0
2	-20.579	-173.972	20.0	6.2	-157355642.0
3	-59.076	-23.557	15.0	5.8	-157093817.0
4	11.938	126.427	15.0	5.8	-157026430.0

```
from mpl_toolkits.basemap import Basemap
```

```
m = Basemap(projection='mill',llcrnrlat=-80,urcnrlat=80, llcrnrlon=-180,urcnrlon=180,lat_ts=20,resolution='c')
```

```
longitudes = data["Longitude"].tolist()
latitudes = data["Latitude"].tolist()
x,y = m(longitudes,latitudes)
```

```
import matplotlib.pyplot as plt
```

```
fig = plt.figure(figsize=(12,10))
plt.title("All affected areas")
plt.plot(x, y, "o", markersize = 2, color = 'blue')
m.drawcoastlines()
m.fillcontinents(color='coral',lake_color='aqua')
m.drawmapboundary()
m.drawcountries()
plt.show()
```

All affected areas

