EARTHQUAKE PREDICTION USING PHYTON

PHASE 2

TEAM 2

COLLEGE CODE: 3128

* Premlatha S(312821205030)
* Ashika jubi S(312821205003)
* Keerthana V(312821205014)
* Venu Aravind(312821205048)
* Aravind R (312821205001)

Predicting earthquakes accurately is a highly complex task that requires a deep understanding of seismology and geophysics. While there are Artificial intelligence techniques applied in seismology for earthquake early warning systems, predicting exact earthquake occurrences with high precision remains a significant challenge.

→Steps to create a basic Earthquake Prediction model:

1. Collect Data:

Obtain historical earthquake data from reliable sources like the United States Geological Survey (USGS). The data should include features such as earthquake magnitude, depth, location, and possibly geological features of the region.

2. Preprocess Data:

Clean the data, handle missing values, and preprocess features. You might need to convert location data to numerical values using techniques like geocoding.

3. Feature Selection:

Identify relevant features that might influence earthquake occurrences. Features like historical seismic activity, tectonic plate boundaries, and geological features could be important.

4. Split Data:

Split the data into training and testing sets to train and evaluate the model's performance.

5. Choose a Model:

Select an appropriate machine learning algorithm. For this task, ensemble methods like Random Forest or Gradient Boosting might work well. Neural networks could also be explored for more complex patterns.

6. Train the Model:

Train your chosen model using the training dataset.

7. Evaluate and Tune:

Evaluate the model using the test dataset. Use appropriate evaluation metrics like Mean Squared Error (MSE) for regression tasks or F1-score for classification tasks. Fine-tune your model by adjusting hyperparameters.

8. Prediction:

Once your model is trained and tuned, you can input new data to predict earthquake occurrences.

PROGRAM:

import numpy as np

from sklearn.ensemble import RandomForestClassifier

# Simulated earthquake data

# Features: Magnitude, Depth, Location, Time

X = np.array([[7.2, 10, 34.0, 9.5],

[5.5, 5, 38.0, 7.0],

[6.1, 8, 36.5, 8.3]])

# Labels: 1 for earthquake, 0 for no earthquake

y = np.array([1, 0, 1])

# Create a random forest classifier

clf = RandomForestClassifier()

# Train the classifier on the data

clf.fit(X, y)

# Predict if an earthquake is likely based on new data

new\_data = np.array([[6.5, 9, 35.5, 9.0]])

prediction = clf.predict(new\_data)

if prediction == 1:

print("Earthquake predicted!")

else:

print("No earthquake predicted.")

CONCLUSION:

Note that this is a highly simplified and unrealistic example for demonstration purposes. In practice, earthquake prediction involves vast and complex datasets, and AI is more commonly used for seismic hazard assessment, where it helps identify areas with a higher likelihood of earthquakes based on historical data and geological factors.

THANK YOU..