

Phase 4: Development Part 2

In this phase we will Continue building the earthquake prediction model by:

- Visualizing the data on a world map
- Splitting it into training and testing sets

1.1 Visualizing the data on a world map :

To display the data on the world map, we will be using a Python library Folium. Data that has been altered in Python may be easily seen on an interactive leaflet map with Folium.

It makes it possible to send rich vector, raster, and HTML visualizations as markers on the map in addition to tying data to a map for choropleth visualizations.

We can use this library by importing it.

```
Import Folium  
m = folium.Map(location=(lat, long))
```

To display the map following code is used:

```
m.save("index.html")
```

1.2 Splitting it into training and testing sets :

Once the data has been visually represented, it should be separated into testing and training sets.

This is a crucial phase in the development of models for machine learning. You can use techniques like time-based splitting or random sampling, depending on your dataset.

Generally speaking, there should be 80% training and 20% testing, however this can vary based on your particular needs.

For partitioning the data in Python, we can use packages like scikit-learn.

```
from sklearn.model_selection import train_test_split
```

To split the dataset into training and testing the following code is used:

```
X_train, X_test, y_train, y_test = train_test_split(features, labels, test_size=0.2,
random_state=42)
```

1.3 Now Let's combine above sections in order to develop our model:

First we will import necessary libraries required to build our model, which is implemented with help of following code:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import folium
from sklearn.model_selection import train_test_split
```

Next step is to load our earthquake data provided by Naan Mudhalvan, with help of the following code.

```
data = pd.read_csv('earthquake_data.csv')
```

Let's Visualize earthquake data on a world map using Folium

```
m = folium.Map(location=[0, 0], zoom_start=2)
```

```
for index, row in data.iterrows():
    folium.CircleMarker( location=[row['latitude'],
                                row['longitude']],
```

```
radius=row['magnitude'],  
color='red',  
fill=True, fill_color='red',  
popup=f" Magnitude: {row['magnitude']} ",  
)
```

```
m.save('earthquake_map.html')
```

Now let's split the dataset into training and testing sets

```
X = data[['latitude', 'longitude']]
```

```
y = data['magnitude']
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,  
random_state=42)
```

We have successfully built our model that satisfies the given condition such as visualizing the data on the world map and split the data set into train and testing sets.