Flood Monitoring and Early Warning system using IoT

Phase-2 Document Submission

Project title: Flood Monitoring and Early Warning

Introduction:

Flood monitoring is the systematic observation and assessment of water levels, weather conditions, and other relevant data to predict, track, and manage floods. It involves the use of various tools and technologies, such as weather stations, river gauges, satellite imagery, and computer models, to provide early warning, assess flood risks, and support disaster response efforts. Effective flood monitoring is crucial for mitigating the impact of floods on communities and the environment, helping to save lives and reduce property damage.

Solution / Working of Flood Monitoring and Early Warning System:

Flood monitoring is a crucial process that involves the continuous observation, assessment, and prediction of flooding events in order to mitigate their impact on people, property, and the environment. It typically includes the following components:

1. Data Collection: Flood monitoring starts with the collection of data from various sources, such as weather stations, river gauges, satellites, and ground sensors. This data includes rainfall measurements, river water levels, and soil moisture content.

2. Early Warning Systems: Once data is collected, it's used to create early warning systems. These systems provide real-time information about flood conditions and help authorities issue timely warnings to affected communities.

3. Modelling and Prediction: Flood monitoring also involves the use of mathematical models and computer simulations to predict how weather conditions and water levels will evolve. This allows for more accurate flood forecasting.

4. Remote Sensing: Satellite imagery and remote sensing technologies play a crucial role in monitoring and assessing flood situations over large areas. They provide a broader perspective on flood extent and severity.

5. Community Engagement: Educating and involving communities in flood monitoring is essential. People living in flood-prone areas need to be aware of risks and know how to respond to warnings and evacuation orders.

6. Infrastructure and Response Planning: Flood monitoring is closely linked to disaster preparedness. It helps local governments and emergency agencies plan for response efforts, such as the deployment of sandbags, rescue teams, and evacuation routes.

7. Data Dissemination: The information gathered through flood monitoring should be readily accessible to the public, emergency responders, and relevant government agencies through various communication channels, including websites, mobile apps, and public broadcasts.

By integrating these components, flood monitoring aims to reduce the impact of floods, save lives, and minimize property damage by providing early warnings and the necessary data for informed decision-making.

Working With the Flood Dataset:

Importing the Dataset:

To import a flood dataset, you can follow these steps:

- Identify the dataset you want to import. There are many different flood datasets available, so it is important to choose one that is appropriate for your needs. Here we have used the following dataset: https://www.kaggle.com/datasets/aditya2803/india-floods-inventory
- Download the dataset. Once we have identified a dataset, we can download it from the provider's website.
- Import the dataset into your software. The specific steps to import a dataset will vary depending on the software you are using. However, most software packages will have a way to import data from a file.

```
#Importing the dataset
import pandas as pd
import numpy as np
df = pd.read_csv("flood.csv")
```

```
df
```

	UEI	Start Date	End Date	Duration(Days)	Main Cause	Location	Districts	State	Latitude	Longitude	Severity	Area Affected	Human fatality	Human injured	Huma Displac
0	UEI-DFO-FL-2017-0001	06-02-2017	07-03-2017	29	Monsoonal rain	NaN	NaN	NaN	27.61310	94.9387	1.0	65387.73	6.0	NaN	113
1	UEI-EM-DAT-FL-2017-0002	01-06-2017	31/06/2017	Diffing the dates	NaN	Gujarat	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Na
2	UEI-EM-DAT-FL-2017-0001	04-06-2017	06-06-2017	2	NaN	Lakhimpur, Karimganj, Darrang districts (Assam...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Na

```
df.isnull().sum()
```

```
UEI                                 0
Start Date                          0
End Date                            0
Duration(Days)                      0
Main Cause                         16
Location                           27
Districts                          48
State                              48
Latitude                           20
Longitude                          20
Severity                           21
Area Affected                      21
Human fatality                     21
Human injured                      48
Human Displaced                    21
Animal Fatality                    48
Description of Casualties/injured  48
Unnamed: 17                        48
Event Source                        0
Event Souce ID                      0
dtype: int64
```

Cleaning the Dataset:

- Data cleaning is the process of identifying and correcting errors and inconsistencies in a dataset. It is an important step in any data analysis project, as unclean data can lead to inaccurate results.
- Removing duplicate rows. This can be done by identifying rows that have the same values in all columns.
- Handling missing values. This can be done by either dropping rows with missing values or imputing the missing values with appropriate values.
- Correcting errors in data formats. This can include correcting errors in dates, times, numbers, and strings.
- Converting data to a consistent format. This can include converting all dates to the same format, all numbers to the same number of decimal places, and all strings to the same case.
- Removing outliers. Outliers are data points that are very different from the rest of the data. They can be caused by errors in data collection or by unusual events.

```
#Cleaning the dataset
missing_value = ['NA','na','N/A',np.nan]
df = pd.read_csv("flood.csv",na_values = missing_value)
df.drop(columns=df.columns[16:],inplace=True)
```

```
df.fillna({
    'Location':'Chennai',
    'Districts':'Chennai',
    'State':'Tamilnadu',
    'Latitude':'13.067439',
    'Longitude':'80.237617',
    'Severity':0,
    'Area Affected':0,
    'Human Displaced':0,
    'Human fatality':0,
    'Human injured':0,
    'Animal Fatality':0,
    'Description of Casualties/injured':0
    })
```

})

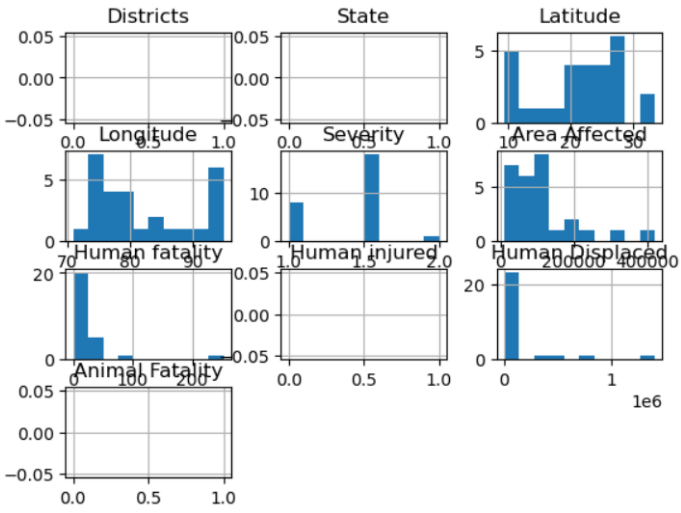
	UEI	Start Date	End Date	Duration(Days)	Main Cause	Location	Districts	State	Latitude	Longitude	Severity	Area Affected	Human fatality	Human injured	Human Displaced	Animal Fatality
0	UEI-DFO-FL-2017-0001	06-02-2017	07-03-2017	29	Monsoonal rain	Chennai	Chennai	Tamilnadu	27.6131	94.9387	1.0	65387.73	6.0	0.0	113.0	0.0
1	UEI-EM-DAT-FL-2017-0002	01-06-2017	31/06/2017	Diffing the dates	NaN	Gujarat	Chennai	Tamilnadu	13.067439	80.237617	0.0	0.00	0.0	0.0	0.0	0.0
2	UEI-EM-DAT-FL-2017-0001	04-06-2017	06-06-2017	2	NaN	Lakhimpur, Karimganj, Darrang districts (Assam...	Chennai	Tamilnadu	13.067439	80.237617	0.0	0.00	0.0	0.0	0.0	0.0

Analysing the Dataset:

- Data analysis is the process of collecting, cleaning, and examining data to extract meaningful insights.
- Exploratory data analysis (EDA). EDA is a process of visually exploring the data to identify patterns and anomalies. This can be done using a variety of charts and graphs.
- Hypothesis testing. Hypothesis testing is a process of testing a hypothesis about the data using statistical methods. This can be used to determine whether there is a significant relationship between two variables or whether the data is consistent with a particular distribution.

```
#Analysing the dataset
```

```
import matplotlib.pyplot as plt
df.hist()
plt.show()
```



```
df.describe()
```

	Districts	State	Latitude	Longitude	Severity	Area Affected	Human fatality	Human injured	Human Displaced	Animal Fatality
count	0.0	0.0	28.00000	28.000000	27.000000	27.000000	27.000000	0.0	2.700000e+01	0.0
mean	NaN	NaN	21.34648	82.129907	1.370370	112308.678148	25.185185	NaN	1.213890e+05	NaN
std	NaN	NaN	6.86586	7.952965	0.262847	92516.274436	49.881426	NaN	3.095922e+05	NaN
min	NaN	NaN	9.39943	70.937500	1.000000	9467.550000	0.000000	NaN	0.000000e+00	NaN
25%	NaN	NaN	17.01460	75.626425	1.000000	53873.690000	2.500000	NaN	8.425000e+02	NaN
50%	NaN	NaN	22.40205	79.003950	1.500000	94518.750000	11.000000	NaN	3.000000e+03	NaN
75%	NaN	NaN	26.93805	88.957000	1.500000	120100.615000	26.500000	NaN	4.600000e+04	NaN
max	NaN	NaN	33.37520	94.938700	2.000000	421575.240000	253.000000	NaN	1.400000e+06	NaN

```
df['Severity'].value_counts()
```

```
1.5    18
1.0     8
2.0     1
Name: Severity, dtype: int64
```

Conclusion:

By performing data cleaning and analysis on a historical flood dataset, we can gain valuable insights into how various resources are being used and identify areas for improvement. This information can be used to develop more efficient and sustainable flood monitoring and early alerting system.

Team Members & their Reg.no:

- Gowsalya N – 921021104012
- Epshiba R - 921021104010
- Deepika E – 921021104005
- Meenakshi R - 921021104027