Flood Prediction using Machine Learning

Overview

This project focuses on predicting flood occurrences using **machine learning techniques**. The goal is to analyze environmental and historical flood-related data, preprocess it, and build predictive models that can assist in disaster management and early warning systems.

Tech Stack

- Python (Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn)
- Jupyter Notebook / Google Colab
- Machine Learning Algorithms (Logistic Regression, Decision Trees, Random Forest, etc.)

Project Structure

```
Flood-Prediction-ML/
├── Flood_Prediction.ipynb   # Main Jupyter/Colab notebook
├── data/                    # Flood datasets (CSV, Excel)
├── images/                  # Visualizations & charts
├── reports/                 # Project documentation (PDF/DOCX)
└── README.md                # Documentation file
```

Key Steps

- **Data Collection & Cleaning** – Imported flood-related datasets and handled missing values, outliers, and inconsistencies.
- **Exploratory Data Analysis (EDA)** – Studied rainfall, water levels, and historical flood patterns. Created visualizations to identify correlations.
- **Feature Engineering** – Encoded categorical values, normalized numerical features, and prepared datasets for ML models.
- **Model Building** – Applied ML models like Logistic Regression, Decision Trees, and Random Forest to predict flood likelihood.
- **Evaluation** – Measured performance using accuracy, precision, recall, F1-score, and confusion matrix.
- **Results & Insights** – Highlighted environmental factors most strongly correlated with flood events.

Results

- Achieved **[Insert accuracy, e.g., 89%]** accuracy with Random Forest classifier.
- Found rainfall intensity and water level to be strong predictors of flood risk.
- Model can be used as a decision-support tool for **flood risk management**.

How to Run

1. Clone the repository:
```
git clone https://github.com/your-username/flood-prediction-ml.git
```
3. Open the notebook:
```
jupyter notebook Flood_Prediction.ipynb
```

   or upload it to **Google Colab**.

5. Install dependencies:
```
pip install -r requirements.txt
```
7. Run all cells sequentially.

Future Work

- Use deep learning methods (LSTM, RNN) for time-series flood prediction.
- Deploy the model as a web or mobile app for real-time predictions.
- Integrate with live weather API data for continuous monitoring.

Author

Chandrika Pise
MCA Student | Data Analyst Intern | Skilled in Python, SQL, Power BI, Machine Learning
LinkedIn: Chandrika Pise