METEORITE LANDING API

Project 3

INTRODUCTION

Objective:

- The goal of this project is to visualize meteorite landings worldwide using data from NASA's Meteorite Landings dataset.
- The visualization tools used include scatter plots, heatmaps, and a geographic map of landings.

LIBRARIES AND TOOL

- **Technologies Used:**
- **Frontend:** HTML, CSS, JavaScript (Leaflet.js for map visualization)
- **Backend:** Flask API for serving the dataset and handling requests
- **Data Science Tools:** Pandas, Matplotlib, and D3.js for visualization

DATA ANALYSIS PIPELINE DATA ANALYSIS PIPELINE

Slide Title: *Data Analysis Pipeline*

Content:

- **Data Source:**
  - The dataset is sourced from NASA's Meteorite Landings database (CSV format).
  - It includes information about meteorite landings, including mass, coordinates, and year of discovery.
- **Pipeline:**
  - **Data Cleaning & Preprocessing:**
    - **Pandas:** Used to clean, filter, and preprocess the dataset.
    - Removed outliers, handled missing data, and normalized the dataset.
  - **Visualization:**
    - **Scatter Plots (Pandas):**
      - The first two images (scatter plots) show meteorite landings over time by **longitude** and **latitude**, demonstrating the trend of increasing sightings over time.
    - **Heatmap (Matplotlib):**
      - A heatmap visualizes meteorite landings by year group and meteorite class, using **Matplotlib** and **Pandas** for data manipulation and plotting.
  - **Geospatial Visualization:**
    - **Leaflet.js** will be used for creating an interactive map of meteorite landings that will be incorporated later.

FRONTEND AND BACKEND

- **Frontend (HTML, CSS, JavaScript):**
- **HTML:** Used for structuring the webpage and embedding images, charts, and maps.
- **CSS:** Applied for styling the webpage, ensuring a clean and responsive layout.
- **JavaScript:**
  - **D3.Js** is used for interactive data visualizations (e.g., line charts and other dynamic content).
  - **Leaflet.Js** will be used to render the geospatial data on a map.
- **Heatmap Image:** The heatmap of meteorite landings is displayed within an `img` tag, styled using CSS.
- **Backend (Flask API):**
- **Flask:** A Python-based web framework used to serve the meteorite data via API endpoints.
- The backend enables users to fetch meteorite data (by name, mass, year, etc.) through API calls, facilitating dynamic data loading.
- Example API endpoints:
  - `/api/v1.0/Name`
  - `/api/v1.0/Mass(g)`
  - `/api/v1.0/Year`
  - `/api/v1.0/GeoLocation`

INTERACTIVE LANDINGS MAP (LEAFLET.JS)

interactive Landings Map (Leaflet.js):

- The upcoming feature is the integration of a geospatial visualization (using **Leaflet.js**), which will plot the meteorite landings on a global map. This will allow users to interactively explore landings by location.
- **Features to include:**
  - Markers for meteorite landing locations.
  - Pop-up info displaying meteorite data (mass, year, name, etc.).
  - Zoomable map for detailed exploration.
  - **Flask API & Future Enhancements:**
- Currently, the **Flask API** provides endpoints for retrieving meteorite data by specific attributes like **Name**, **Mass**, and **Year**.
- Future enhancements will involve:
  - Allowing users to filter the data by meteorite class or geographical region.
  - Adding more complex visualizations and statistical analysis, such as trends over time or comparisons of meteorite types.

CONCLUSION

- **Summary:**
- This project successfully visualizes meteorite landings over time, combining data science and web development tools to present data in a meaningful way.
- The combination of scatter plots, heatmaps, and geospatial maps allows for a comprehensive understanding of meteorite landing trends.
- **Challenges**
- Completing the integration of the **Leaflet.js map**.
- Add interactivity to the heatmap.
- Further develop the Flask API to support more queries and advanced analytics.