Methods of Advance Data Engineering Data Report

Kartik Kanodia

Summer Semester 2024

1 Question

Climate change influences the occurrence of natural disasters by altering weather patterns and environmental conditions. While it may not directly cause specific events, such as hurricanes or wildfires, it exacerbates their impacts and increases their likelihood. Understanding this relationship is essential for developing effective strategies to mitigate risks and enhance resilience. The main focus of this project is to investigate the correlation between the occurrence of natural disasters and climate change in the United States of America from 1990 to 2013.

2 Data Sources

The report contains two data sources.

Datasource 1: Kaggle

Metadata URL: https://www.kaggle.com/berkeleyearth/climate-change-earth-surface-temperature-data

Data Type: CSV

This dataset originates from the Berkeley Earth Surface Temperature Study and covers global land and ocean temperatures from as early as 1750, with temperature averages and uncertainties provided. The dataset provides geographical granularity at the country level.

Licence: CC BY-NC-SA 4.0

CC BY-NC-SA 4.0 is a type of Creative Commons license. This license allows others to Share (copy and redistribute the material in any medium or format) and Adapt (remix, transform, and build upon the material) as long as they credit the original creator, use the material for non-commercial purposes and license any new creations under identical terms.

Datasource 2: Kaggle

Metadata URL: https://www.kaggle.com/datasets/headsortails/us-natural-disaster-declarations

Data Type: CSV

This dataset is a high-level summary of all federally declared disasters that occurred in the United States of America since 1953. The geographical granularity of the raw dataset is at a city level. The dataset provides the type of disaster and also binary flags that indicate whether specific aid programs were triggered in response.

Licence:

U.S. Government Works license refers to works created by employees of the United States federal government as part of their official duties. Since these works are in the public domain, they are free to use by anyone without needing permission or paying royalties. They can be freely copied, modified, distributed, and performed by anyone.

3 Data Pipeline

3.1 Description

The data pipeline has been implemented in Python. The libraries used are Kaggle: used for extracting data from Kaggle via api, Operation System: used for interacting with the operating system like creating files and directories, Pandas: for data manipulation and analysis, and Sqlite3: for exporting the data frames into sqlite format.

3.2 Cleaning and Transformation

Once the dataset is downloaded as a CSV file, it is converted into a panda data frame. All the cleaning and transformation steps then take place on the data frame.

Cleaning and Transformation (Data Source 1):

The starting data cleaning step involves the removal of unnecessary columns from the data frame. Followed by the renaming of some columns to ensure a consistent format with more appropriate names. This step is followed by year and month column extraction from the identified date column. After this step, the data is filtered based on the country column with values as the United States and based on the year column for a range of 1990 to 2013. At last, the data frame is grouped by country, year, and month columns with a calculated column for average temperature rounded off to one decimal place.

Cleaning and Transformation (Data Source 2):

The very first data cleaning step involves the removal of unnecessary columns from the data frame. Followed by year and month column extraction from the identified date column. After this step, the data is filtered on the basis of the year column for a range of 1990 to 2013. The location of the data is The United States of America. However, no such column exists in the data frame. Hence a column country with the value 'United States of America' is added to the data frame. In the next step, two separate data frames are created. The first data frame is grouped on the basis of country, year, month, and incident type columns. The second data frame is a pivot table to create separate columns for each unique value (dr = major disaster), em = emergency management, fm = fire management) in the column declaration type. Finally, the grouped data frame and pivoted data frame are merged together to form a data frame with columns country, year, month, incident type, total incidents, number of dr, number of em, and number of fm.

3.3 Problems Encountered

The main challenge with the datasets was aligning their levels of granularity and time periods. To address this issue, both datasets had to be adjusted to each other's lower level of granularity. For example, Data Source 1 provided data at the country level, while Data Source 2 had data at the state level. To standardize the data, Data Source 2 had to be aggregated to the country level. Similarly, Data Source 2 only included dates up to 2013, whereas Data Source 1 extended dates to 2023. Therefore, Data Source 1 was filtered to include only data up to 2013, ensuring consistency between the two datasets.

3.4 Error Handling

To avoid unexpected errors only static data sources have been selected. However, in case of any changes to the dataset structure, the group by clause in the final step of the pipeline will ensure that only the required columns are present in the final data frame.

4 Result and Limitations

4.1 Data Pipeline Output

The final output of the data pipeline is two structured datasets with fixed schema stored as relational SQlite files. It fulfills all important parameters for data quality. Since the data originates from reputed sources their accuracy is ensured. The datasets are complete with all relevant information. The SQLite format ensures that the dataset is consistent. Both the datasets have data from the years 1990 to 2013 ensuring timeliness and relevancy.

4.2 Data Format

The SQLite file format has been chosen for its simplicity and efficiency. It is a self-contained, serverless database engine, where the entire database is stored in a single file. SQLite files require no installation or configuration, which makes it an appropriate choice for the current use case. Since these files are easy to set up, use, and maintain.

4.3 Potential Issues

Even though a static data source has been selected as input for the data pipelines, however, unexpected changes at the source may lead to potential disruptions in the pipeline and lead to wrong of no output.