Analysing the Impact of Air Pollution on Urban Climate in London

Main Question

How do variations in air pollution levels impact weather conditions in London from 2008 to 2018?

Description

This project examines the relationship between London's air pollution levels and weather from 2008 to 2018. The goal is to examine how changes in the weather might affect air quality by using historical weather data along with pollution measurements. The project integrates data from the London Datastore and Kaggle to investigate relationships between weather patterns and pollution indices, including its impact of temperature on gasses and particle matter. Our goal is to get a deeper understanding of the environmental dynamics in metropolitan London through this study.

Data Sources

London Weather Data

- **Source**: Kaggle
- **URL**: London Weather Data on Kaggle
- **Period**: 1979 - 2021
- **Description**: This dataset includes historical weather data from London, collected from a weather station near Heathrow Airport. It features daily measurements such as temperature, humidity, and precipitation.
- **License**: Public Domain (CC0)

London Air Quality Levels

- **Source**: London Datastore
- **URL**: London Average Air Quality Levels
- **Period**: Data available till 2019
- **Description**: This dataset provides readings of air pollutants including Nitric Oxide, Nitrogen Dioxide, Particulate Matter (PM10 and PM2.5), and Ozone. Measurements are collected from various monitoring sites across Greater London.
- **License**: UK Open Government Licence (OGL v2) (allowing use with attribution. Obligations include acknowledging the data source and maintaining a link to the license.)

Objectives

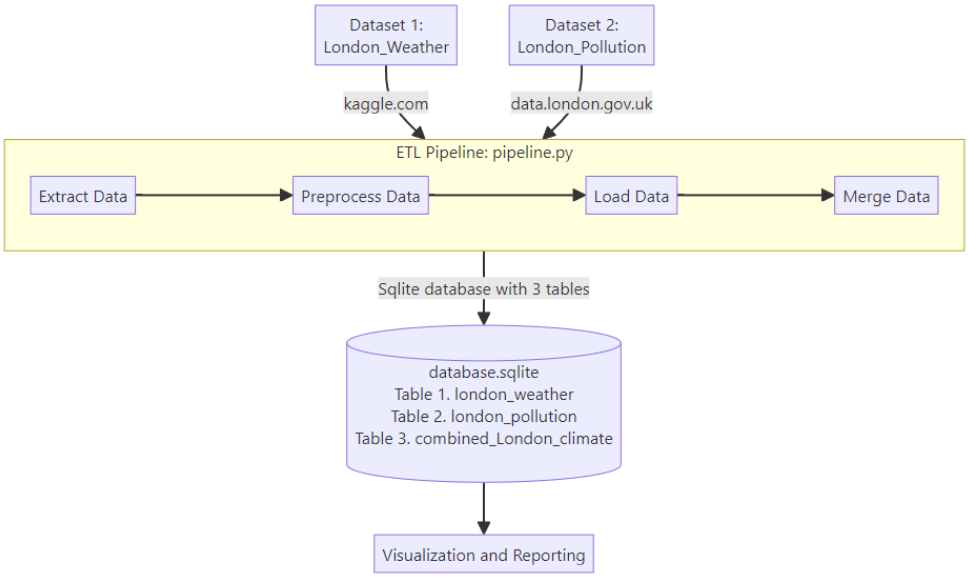
- **Data Integration**: Combine the weather and pollution data daily to perform a cohesive analysis.
- **Correlation Analysis**: Identify statistical correlations between specific weather conditions and pollution levels.
- **Visualization and Reporting**: Create visualizations to represent the findings and compile a comprehensive report detailing the insights.

Data Pipeline

The data pipeline was implemented using Python, utilizing libraries such as pandas for data manipulation and SQLite for data storage. The pipeline consists of the following steps:

1. **Extraction**: Downloading datasets from Kaggle and the London Datastore using automated scripts.

2. **Transformation and Cleaning**:
    - **Weather Data**: Date conversion, filtering for the first day of each month, and removal of unnecessary columns such as 'snow_depth'.
    - **Pollution Data**: Renaming columns for clarity, filtering records for 14:00 hours, and removing columns like 'GMT', 'Roadside_NO', and 'Background_NO'.
3. **Loading**: Saving cleaned datasets into separate SQLite databases.
4. **Merging**: Combining weather and air quality datasets on the 'date' column into a final SQLite database for analysis.



Technologies Used

- **Python**: For data processing and pipeline implementation.
- **Pandas**: For data manipulation and cleaning.
- **SQLite**: For efficient data storage and querying.

Problems Encountered and Solutions

- **Data Formats and Column Names**: The primary issue was the inconsistency in data formats and column names across datasets. This was resolved by standardizing formats and renaming columns during the transformation stage.
- **Missing Values**: Addressed by employing imputation techniques and removing records with insufficient data.

Error Handling

- **Logging**: Implemented logging to capture and debug errors.
- **Exception Handling**: Used try-except blocks to manage potential exceptions and prevent pipeline failures.

Results and Limitations

Output Data

- **Description**: The final dataset integrates weather and pollution data from 2008 to 2018.
- **Data Structure**: Structured in a tabular format with each row representing a day, including columns for various weather and pollution measurements.
- **Data Quality**: High, with standardized formats and handled missing values ensuring consistency.
- **Output Format**: The data is stored in an SQLite database, chosen for its efficiency in querying and manipulation.

Here, the detailed SQLite database tables can be seen concisely. More details can be found within my code.

[4]:
```
import sqlite3
import pandas as pd

def load_data(db_path, query):
    conn = sqlite3.connect(db_path)
    df = pd.read_sql_query(query, conn)
    conn.close()
    return df

# Load weather and pollution data
weather_data = load_data('../data/London_weather.sqlite', 'SELECT * FROM weather_data')
pollution_data = load_data('../data/London_pollution.sqlite', 'SELECT * FROM time_of_day_data')

print("Weather Data Head:")
display(weather_data.head())
```

```
Weather Data Head:
```

	date	cloud_cover	sunshine	global_radiation	max_temp	mean_temp	min_temp	precipitation	pressure
0	2008-01-01	8.0	0.0	13.0	9.1	7.8	6.6	0.4	102260.0
1	2008-02-01	1.0	6.0	68.0	8.7	5.9	3.1	0.0	99970.0
2	2008-03-01	5.0	5.5	108.0	12.9	8.9	4.9	0.0	100730.0
3	2008-04-01	4.0	7.5	178.0	15.8	11.4	6.9	0.0	101870.0
4	2008-05-01	6.0	4.6	175.0	15.2	11.0	6.9	4.0	100650.0

[5]:
```
print("Pollution Data Head:")
display(pollution_data.head())
```

```
Pollution Data Head:
```

	date	Roadside_NO2	Roadside_NOx	Roadside_O3	Roadside_PM10	Roadside_PM2_5	Roadside_SO2	Background_NO2	Background_NOx	Background_O3	Backgrour
0	2008-01-01	60.838710	NaN	30.645161	26.322581	14.225806	4.870968	41.064516	NaN	41.129032	1
1	2008-02-01	72.517241	NaN	26.689655	44.551724	28.137931	10.620690	50.827586	NaN	37.724138	3
2	2008-03-01	58.129032	NaN	46.870968	24.290323	12.000000	4.580645	35.161290	NaN	61.806452	1
3	2008-04-01	60.833333	NaN	49.733333	29.766667	18.833333	5.200000	35.000000	NaN	69.266667	2
4	2008-05-01	63.967742	NaN	64.903226	33.967742	22.612903	5.612903	35.387097	NaN	85.774194	2

Merge Data ¶

[7]:
```
combined_data = load_data('../data/combined_London_climate.sqlite', 'SELECT * FROM combined_weather_data')
display(combined_data.head())
```

	date	cloud_cover	sunshine	global_radiation	max_temp	mean_temp	min_temp	precipitation	pressure	Roadside_NO2	...	Roadside_O3	Roadside_PM10	Roadsid
0	2008-01-01	8.0	0.0	13.0	9.1	7.8	6.6	0.4	102260.0	60.838710	...	30.645161	26.322581	1
1	2008-02-01	1.0	6.0	68.0	8.7	5.9	3.1	0.0	99970.0	72.517241	...	26.689655	44.551724	2
2	2008-03-01	5.0	5.5	108.0	12.9	8.9	4.9	0.0	100730.0	58.129032	...	46.870968	24.290323	1
3	2008-04-01	4.0	7.5	178.0	15.8	11.4	6.9	0.0	101870.0	60.833333	...	49.733333	29.766667	1
4	2008-05-01	6.0	4.6	175.0	15.2	11.0	6.9	4.0	100650.0	63.967742	...	64.903226	33.967742	2

5 rows × 21 columns

Critical Reflection

While the dataset provides comprehensive insights into weather and air quality over a decade, potential issues include the fitness of the data, which may mask short-term variations. Additionally, external factors influencing pollution levels, such as local traffic patterns or industrial activities, are not accounted for in the weather data. These limitations should be considered when interpreting the results and drawing conclusions in the final analysis.

Important Point: I changed my previous datasets due to their problem in mapping between each other. This limitation is solved here by considering the date as the mapping field and using the same city's datasets.