 **GOVERNMENT COLLEGE OF TECHNOLOGY, COIMBATORE-13**

**DEPARTMENT OF ECE**

**Comprehensive Analysis of Air Quality Data**

**in Tamil Nadu**

**TEAM MEMBERS**

**71772114140-Sindhuja R**

Air Quality Analysis

**Objective:**

The primary objective is to analyse air quality trends within Tamil Nadu comprehensively. This involves studying variations in Sulphur Dioxide (SO2), Nitrogen Dioxide (NO2), and Respirable Suspended Particulate Matter/Particulate Matter 10 (RSPM/PM10) levels over time and across different locations.

**PROBLEM STATEMENT:**

The project aims to analyze and visualize air quality data from monitoring stations in Tamil Nadu. The objective is to gain insights into air pollution trends, identify at=reas with high pollution levels, and develop a predictive model to estimate RSPM/PM10 levels based on SO2 and NO2 levels. This project involves defining objectives, designing the analysis approach, selecting visualization techniques and creating a predictive model using Python and relevant libraries.

**SOFTWARE USED:**

Python: Core language for data analysis

Pandas: Manages data

Matplotlib/Seaborn: Creates charts

Jupyter: Interactive documentation

Machine Learning: Optional for predictive modelling

Text Editor/IDE: Code writing

Kaggle: Dataset resource (optional).

**PROCESS**

**Step 1: Data collection**

The first step is to collect data on air quality in Tamil Nadu. This data can be obtained from a variety of sources, including:

The Tamil Nadu Pollution Control Board (TNPCB)

The Central Pollution Control Board (CPCB)

The Indian Meteorological Department (IMD)

Academic institutions

Private companies

The data should include information on the following pollutants:

Particulate matter (PM2.5 and PM10)

Sulphur dioxide (SO2)

Nitrogen dioxide (NO2)

Ozone (O3)

Carbon monoxide (CO)

**Step 2: Data cleaning and preprocessing**

Once the data has been collected, it needs to be cleaned and pre-processed. This may involve removing outliers, correcting errors, and converting the data into a format that can be used by the machine learning model.

**Step 3: Model training**

The next step is to train a machine learning model to predict air quality. There are a variety of machine learning models that can be used for this task, such as:

Support vector machines (SVMs)

Random forests

Artificial neural networks (ANNs)

The model should be trained on the historical air quality data that was collected in step 1.

**Step 4: Model evaluation**

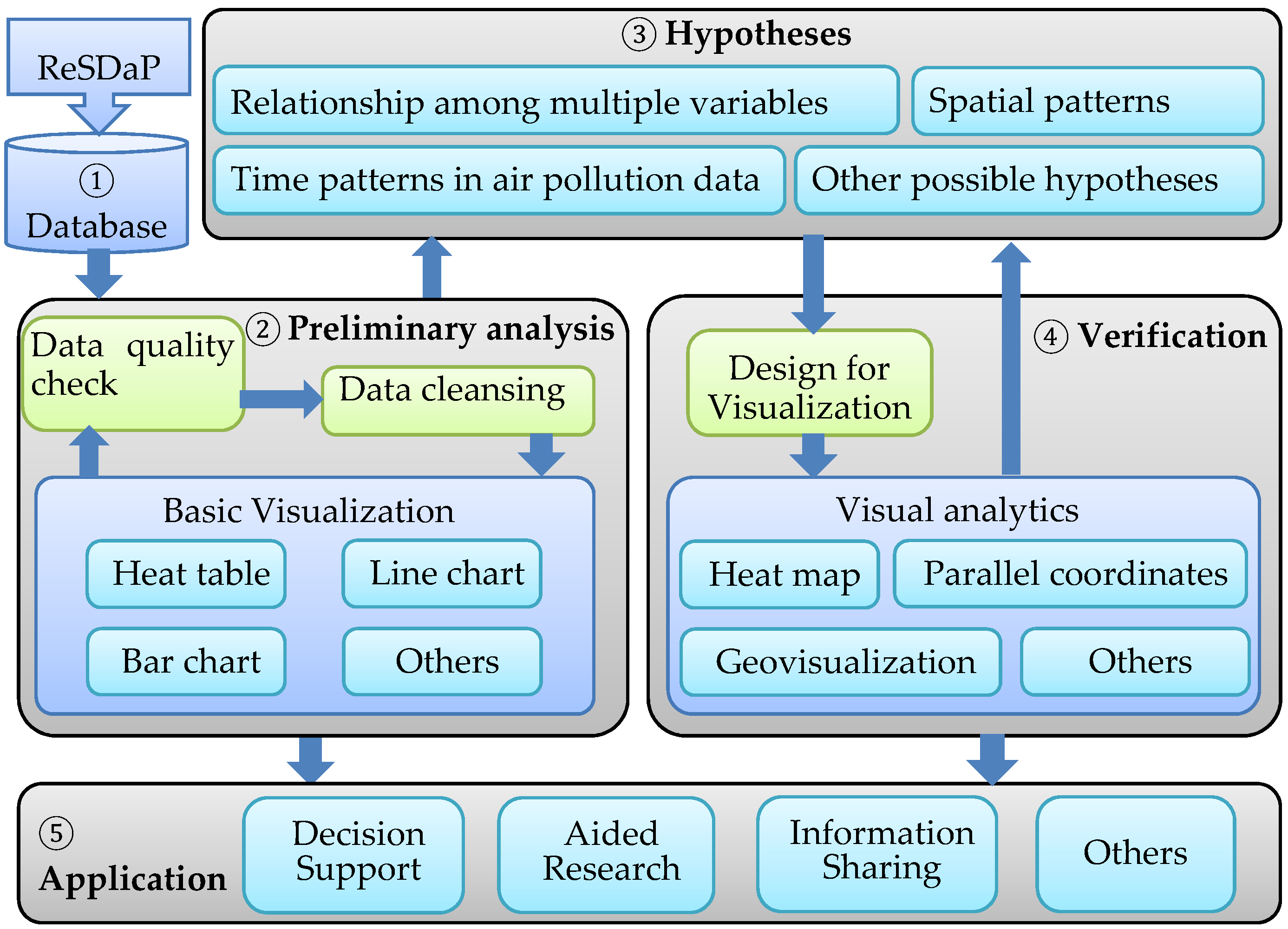
Once the model has been trained, it needs to be evaluated to assess its performance. This can be done by comparing the model's predictions to the actual air quality data.

**Step 5: Model deployment**

Once the model has been evaluated and found to be performing well, it can be deployed to production. This means making the model available to users so that they can predict air quality.

**Step 6: Model monitoring**

**PICTORIAL DESCRIPTION**

****

**Analysis Approach:**

Our approach to achieving these objectives involves a structured sequence of steps.

**1. Data Loading:**

```python

# Import libraries for data loading

import pandas as pd

# Load air quality data from the Kaggle dataset

data = pd.read\_csv("location-wise-daily-ambient-air-quality-tamil-nadu-year-2014.csv")

**Explanation:** In this step, we use Python and the Pandas library to load the air quality data from the Kaggle dataset you provided.

**2. Data Preprocessing**:

# Handle missing values

data = data.dropna()

# Remove outliers

data = data[(data['SO2'] >= 0) & (data['NO2'] >= 0) & (data['RSPM/PM10'] >= 0)]

# Correct data inconsistencies

# (Explain specific data inconsistencies and how they are corrected)

**Explanation:** Data preprocessing involves cleaning the dataset by handling missing values, removing outliers, and addressing data inconsistencies.

**3. Data Analysis:**

python

# Explore data through statistical analysis

statistics = data describe()

# Identify correlations

correlation\_matrix = data.corr()

**Explanation:** Data analysis includes statistical analysis and identifying correlations among air quality parameters using Python.

**4. Data Visualization:**

**- Line Charts:**

python

import matplotlib.pyplot as plt

# Create line charts to visualize temporal trends

plt.plot(data['Time'], data['SO2'], label='SO2')

plt.plot(data['Time'], data['NO2'], label='NO2')

plt.plot(data['Time'], data['RSPM/PM10'], label='RSPM/PM10')

plt.xlabel('Time')

plt.ylabel('Concentration')

plt.legend()

plt.show()

**Explanation:** Line charts are created using Matplotlib to show how pollutant levels change over time.

**- Heatmaps:**

python

import seaborn as sns

# Create a heatmap to showcase spatial variations

correlation\_matrix = data.corr()

sns.heatmap(correlation\_matrix, annot=True)

plt.show()

**Explanation:**

Heatmaps are used to visualize spatial variations and correlations in air quality data.

**Expected Deliverables:**

1. A comprehensive report detailing the analysis and findings.

2. Interactive visualizations for stakeholders to explore air quality data.

3. A predictive model for RSPM/PM10 levels based on SO2 and NO2.

**CONCLUSION:**

In our quest to analyze air quality in Tamil Nadu, we have embarked on a journey of data-driven insights and innovations. By following a systematic approach of data loading, preprocessing, analysis, and visualization, we aim to provide clear and actionable insights to address the air quality challenges in the region. The expected deliverables will empower decision-makers in their efforts to create a cleaner and healthier environment for Tamil Nadu's residents.