**Data Science Toolbox: Python Programming**

**PROJECT REPORT**

(Project Semester January-April 2025)


# *Analysis Of Comprehensive Environmental Pollution Index (CEPI) Scores Of India*

Submitted by:

Faraz Ahmad Khan

Registration No.: 12313142

Programme and Section: B.Tech (CSE), K23SG

Course Code: INT375

Under the Guidance of

**Dr. Manpreet Singh Sehgal (UID: 32354)**

**Discipline of CSE/IT**

**Lovely School of Computer Science and Engineering**

**Lovely Professional University, Phagwara**

# CERTIFICATE

This is to certify that Faraz Ahmad Khan bearing Registration no. 12313142 has completed INT375 project titled, **"Analysis of Comprehensive Environmental Pollution Index (CEPI) Scores of India"** under my guidance and supervision. To the best of my knowledge, the present work is the result of his original development, effort and study.

**Signature and Name of the Supervisor**
**Dr. Manpreet Singh Sehgal**
**School of Computer Science and Engineering**
Lovely Professional University
Phagwara, Punjab.

Date: 12-04-2025

# DECLARATION

I, Faraz Ahmad Khan student of B.Tech Computer Science and Engineering under CSE/IT Discipline at, Lovely Professional University, Punjab, hereby declare that all the information furnished in this project report is based on my own intensive work and is genuine.

Date: 12-04-2025 Signature:

Registration No. 12313142 Faraz Ahmad Khan

# Acknowledgement

I would like to express my heartfelt gratitude to all those who supported me throughout the duration of this project.

First and foremost, I would like to thank Dr. Manpreet Singh Sehgal, my project supervisor, for his valuable guidance, constant encouragement, and unwavering support throughout the course of this project. His insights and feedback greatly contributed to the depth and clarity of the analysis.

I am also grateful to the faculty members of the School of Computer Science, Lovely Professional University, for providing a conducive academic environment and the necessary technical resources.

Special thanks to my friends for their continuous motivation and moral support, which played a significant role in the successful completion of this work.

Lastly, I extend my appreciation to the open-source data contributors and the developers of the various tools and libraries used in this project, such as Python, Pandas, Numpy without which this project would not have been possible.

This project has been an enriching learning experience, and I sincerely thank everyone who made it possible.

# Table Of Content

# 1. Introduction

**Comprehensive Environmental Pollution Index (CEPI) Scores**

The Comprehensive Environmental Pollution Index (CEPI) is a quantitative tool developed by the Central Pollution Control Board (CPCB) and the Ministry of Environment, Forest and Climate Change (MoEF & CC), Government of India, to assess the environmental impact of industrial clusters. The CEPI score ranges from 0 to 100, where:

- Scores above 70 indicate critically polluted areas requiring urgent intervention.
- Scores between 60-70 signify severely polluted zones needing corrective measures.
- Scores below 60 denote less polluted regions but still under observation.

The index considers multiple pollution parameters, including:

- Air and water pollution levels
- Soil contamination
- Impact on human health
- Ecosystem degradation

## 1.1 Importance of Monitoring Industrial Pollution

Industrial clusters contribute significantly to environmental degradation, affecting:

1. **Public Health** – High pollution levels lead to respiratory diseases, cancer, and other ailments.
2. **Biodiversity Loss** – Toxic discharges harm flora and fauna.
3. **Economic Impact** – Pollution-related regulations can restrict industrial growth, while cleanup costs burden governments.
4. **Climate Change** – Industrial emissions contribute to global warming.

Monitoring CEPI scores helps policymakers:

- **Identify pollution hotspots**
- **Enforce stricter regulations**
- **Implement remediation strategies**
- **Track progress in pollution control**
-

## 1.2 Objectives of the Analysis

This project aims to analyze CEPI scores from **2009, 2011, and 2013** to:

### 1.2.1 Compare Pollution Levels Across Years (2009, 2011, 2013)

- Determine if pollution has increased, decreased, or stabilized over time.
- Identify trends in industrial pollution.

#### 1.2.2. Identify Most and Least Polluted Clusters

- Rank industrial clusters based on CEPI scores.
- Highlight top 5 most polluted and bottom 5 least polluted areas each year.

#### 1.2.3. Analyze Impact of Moratorium Status

- Compare CEPI scores between regions:
    - Where moratoriums were lifted (pollution control measures implemented).
    - Where moratoriums are still in force (ongoing pollution issues).
- Assess whether regulatory actions (moratoriums) improved pollution levels.

#### 1.2.4. Compare Pollution Levels Across States

- Identify states with the highest and lowest pollution.
- Examine geographical patterns (e.g., landlocked vs. coastal states).

#### 1.2.5. Examine Score Distributions Over Time

- Analyze how CEPI score distributions changed between 2009 and 2013.
- Determine if pollution control policies had a uniform or varied impact across clusters.

### 1.3 Expected Outcomes

This analysis will provide insights into:

- **Effectiveness of pollution control measures** (e.g., moratoriums).
- **Regions needing urgent intervention**.
- **Long-term trends in industrial pollution**.

The findings can guide **environmental policymakers, industries, and researchers** in formulating better pollution mitigation strategies.

## 2. Source of Dataset

**Dataset: https://www.data.gov.in/resource/details-comprehensive-environmental-pollution-index-cepi-scores-and-status-moratorium**

The dataset, **CEPI_scores.csv**, contains:

- **Industrial Cluster/Area**: Name of the industrial region.
- **State**: Indian state where the cluster is located.
- **Status of Moratorium**: Whether the moratorium has been lifted or is still in force.
- **CEPI Scores**: Scores for the years 2009, 2011, and 2013.

# 3. Exploratory Data Analysis (EDA)

## 3.1 Data Loading and Initial Exploration

### 3.1.1 Data Loading

The dataset was loaded into Python using the pandas library. The CSV file named CEPI_scores.csv was imported as shown below**:**

```
import pandas as pd
# Load the datase
df = pd.read_csv("CEPI_scores.csv")
# Preview the first 5 rows
print(df.head())
```

### 3.1.2 Initial Data Overview

The basic structure and integrity of the dataset were examined**.**

```
# Check the shape of the dataset
print("Shape of the dataset:", df.shape)
# Get data types of each column
print(df.dtypes)
# Check for missing values
print("Missing values:\n", df.isnull().sum())
# Check for duplicate records
print("Duplicate rows:", df.duplicated().sum())
```

## 3.2 Data Preprocessing

### 3.2.1 Data Cleaning

Handling Missing Values and Duplicates

```
# Drop rows with missing values
df.dropna(inplace=True)

# Remove duplicate records
df.drop_duplicates(inplace=True)
```

Any rows with missing CEPI values were dropped. Duplicate entries, if any, were also removed.

Boxplots were used to detect potential outliers in CEPI scores**.**

```
import seaborn as sns
import matplotlib.pyplot as plt

# Boxplot for each CEPI year
sns.boxplot(data=df[["CEPI_2009", "CEPI_2011", "CEPI_2013"]])
plt.title("Boxplot of CEPI Scores (2009, 2011, 2013)")
plt.show()
```

### 3.2.2 Data Transformation

Creating Average CEPI Column

```
# Calculate average CEPI across all years
df["CEPI_Avg"] = df[["CEPI_2009", "CEPI_2011", "CEPI_2013"]].mean(axis=1)
```

An average CEPI score was calculated to help understand the overall pollution severity across years.

```
# Convert 'Status of Moratorium' to a category
df["Status of Moratorium"] = df["Status of Moratorium"].astype("category")
```

### 3.3 Statistical Analysis

#### 3.3.1 Descriptive Statistics

Basic descriptive statistics were used to understand the central tendency and spread of CEPI scores.

```
# Summary statistics
print(df[["CEPI_2009", "CEPI_2011", "CEPI_2013"]].describe())
```

This gives information like:

- Mean
- Median (50%)
- Standard deviation
- Minimum and maximum values
- Quartiles (25%, 75%)

#### 3.3.2 Distribution Analysis

Histograms of CEPI Scores

Histograms were plotted to observe the frequency distribution of CEPI values.

```
# Histogram for CEPI scores
df[["CEPI_2009", "CEPI_2011", "CEPI_2013"]].hist(bins=10, figsize=(12, 6))
plt.suptitle("Histogram of CEPI Scores (2009, 2011, 2013)")
plt.show()
```

Skewness and Kurtosis

These measures help understand the shape of the distribution.

```
from scipy.stats import skew, kurtosis

# Skewness and kurtosis
for year in ["CEPI_2009", "CEPI_2011", "CEPI_2013"]:
    print(f"{year} - Skewness: {skew(df[year])}, Kurtosis: {kurtosis(df[year])}")
```

### 3.4 Temporal Analysis

#### 3.4.1 Trend Visualization

A line chart was plotted to view the CEPI score trends across years for selected clusters.

```
# Sample plot for trend visualization (average across clusters)
df_mean_by_year = df[["CEPI_2009", "CEPI_2011", "CEPI_2013"]].mean()

# Plot the trend
df_mean_by_year.plot(kind='line', marker='o')
plt.title("Average CEPI Trend (2009 to 2013)")
plt.ylabel("CEPI Score")
plt.xlabel("Year")
plt.grid(True)
plt.show()
```

# 4. Analysis

## 4.1 Introduction

This section presents key findings from the CEPI analysis, covering time-based trends, regional differences, moratorium impacts, and score distributions.

## 4.2 General Description of Analysis

Conducted:

- Year-wise CEPI comparison
- Ranking of clusters
- Analysis of moratorium impact
- State-level score averages
- Score distribution evaluation

## 4.3 Analysis Results

### 4.3.1 Comparison Over Time

- Some clusters showed steady declines (e.g., Vapi).
- Others worsened (e.g., Ghaziabad).

### 4.3.2 Most/Least Polluted Clusters

Lists of top 5 and bottom 5 clusters for each year.

### 4.3.3 Impact of Moratorium

Clusters where moratoriums were lifted showed slight improvements in CEPI scores.

### 4.3.4 State-wise Analysis

- Highest pollution: Uttar Pradesh, Maharashtra.
- Lowest: Kerala, Himachal Pradesh.

### 4.3.5 Distribution Analysis

- Distribution became slightly narrower.
- Skewness decreased over time.

### 4.5 Visualizations

- **Histograms**
- **Box Plots**
- **Line Charts**
- **Bar Charts**

## 5. Results and Discussion

### 5.1 Summary of Findings

- CEPI scores revealed both **increasing and decreasing pollution trends** between 2009 and 2013, indicating inconsistent improvements across industrial clusters.
- The **top 5 most polluted clusters** consistently included regions like **Vapi, Ankleshwar, Ghaziabad, and Ludhiana**, signaling persistent environmental challenges.
- The **bottom 5 least polluted clusters** often included clusters from **northeastern and southern states**, suggesting better industrial pollution control.
- **Moratorium status** had a measurable impact—clusters where moratoriums were lifted showed slight improvement in CEPI scores, but not universally.
- **State-wise comparison** showed that **Uttar Pradesh, Gujarat, and Maharashtra** had the highest average CEPI scores across all years, while **Himachal Pradesh, Kerala, and Assam** ranked among the least polluted.
- The **distribution of CEPI scores** showed reduced skewness over time, indicating a more uniform enforcement or impact of pollution control measures.

### 5.2 Key Achievements

- Constructed a clean, analyzable dataset from raw CEPI values.
- Visualized trends and comparisons across time, regions, and regulatory impacts.
- Identified key problem areas requiring urgent attention.
- Supported **data-driven environmental decision-making** through statistical insights.

### 5.3 Limitations

#### 5.3.1 Data Limitations

- Missing or inconsistent entries in the dataset limited some parts of the analysis.
- CEPI scores were available only for **three discrete years**, limiting deeper time series modeling.

#### 5.3.2 Methodological Limitations

- Lack of access to raw pollution concentration data (air/water/soil) restricted parameter-specific insights.

- The moratorium data lacked granularity on the timing and nature of interventions.

#### 5.3.3 Recommendations

- **Policymakers**: Focus interventions on clusters consistently exceeding CEPI 70.
- **Industries**: Adopt cleaner technologies and stricter waste treatment.
- **Local governments**: Enhance air and water quality monitoring infrastructure.
- **Environmental agencies**: Regularly publish detailed CEPI breakdowns by parameter.

### 5.4 Impact Assessment

This project can contribute to:

- **Improved environmental governance** through evidence-based insights.
- **Targeted regulatory action** on industrial clusters most in need.
- **Academic research** on environmental data analysis.
- **Public awareness** about regional environmental health.

### 5.5 Final Remarks

This analysis provides a robust foundation for environmental monitoring using CEPI data. Though limited to three years, it reveals pollution patterns that can guide proactive regulation and help mitigate industrial impacts on ecosystems and public health.

## 6. Future Scope

### 6.1 Potential Enhancements to the Analysis

- Extend analysis using **newer CEPI data** (post-2013).
- Integrate **satellite imagery** or real-time pollution feeds.
- Create **interactive dashboards** for public awareness.

### 6.2 Further Research Directions

- Correlate CEPI scores with **health data** (e.g., respiratory illness rates).
- Analyze **industrial output vs. CEPI** for economic-environmental trade-offs.
- Use **machine learning models** to predict future CEPI trends based on regulatory inputs.

# 7. References

## 7.1 Dataset Source

- Government of India, Data Portal: **https://www.data.gov.in/resource/details-comprehensive-environmental-pollution-index-cepi-scores-and-status-moratorium**

## 7.2 Technical References

- **Pandas**: Data manipulation and analysis
- **NumPy**: Numerical operations
- **Matplotlib & Seaborn**: Data visualization
- **Scikit-learn** (optional): Data preprocessing
- **SciPy**: Statistical calculations

## 7.3 Development Tools

- **Jupyter Notebook / Google Colab**
- **Python 3.10**
- **MS Excel** (for quick data inspection)

## 7.4 Analysis Methods

- Descriptive statistics
- Distribution and trend analysis
- State-wise aggregation and comparison
- Moratorium impact analysis

## 7.5 Visualization Tools

- **Matplotlib**, **Seaborn** (Python libraries)

# 8. Appendices

## 8.1 Appendix A: Raw Data Samples:

| Status of Moratorium | Industrial Cluster / Area | State | CEPI SCORE-2009 | CEPI SCORE-2011 | CEPI SCORE-2013 |
|---|---|---|---|---|---|
| Moratorium has been lifted | Agra | Uttar Pradesh | 76.48 | 88.36 | 68.71 |
| Moratorium has been lifted | Ahmedabad | Gujarat | 75.28 | 78.09 | 69.54 |
| Moratorium has been lifted | Angul Talcher | Orissa | 82.09 | 89.74 | 72.86 |
| Moratorium has been lifted | Asansol | West Bengal | 70.2 | 70.96 | 56.01 |
| Moratorium has been lifted | Aurangabad | Maharashtra | 77.44 | 83.1 | 68.87 |
| Moratorium has been lifted | Bhadravathi | Karnataka | 72.33 | 62.64 | 45.27 |
| Moratorium has been lifted | Bhavnagar | Gujarat | 70.99 | 69.73 | 62.79 |
| Moratorium has been lifted | Bhiwadi | Rajasthan | 82.91 | 77.73 | 70.63 |
| Moratorium has been lifted | Coimbatore | Tamil Nadu | 72.38 | 54.16 | 53.14 |

## 8.2 Appendix B: Code Snippets:

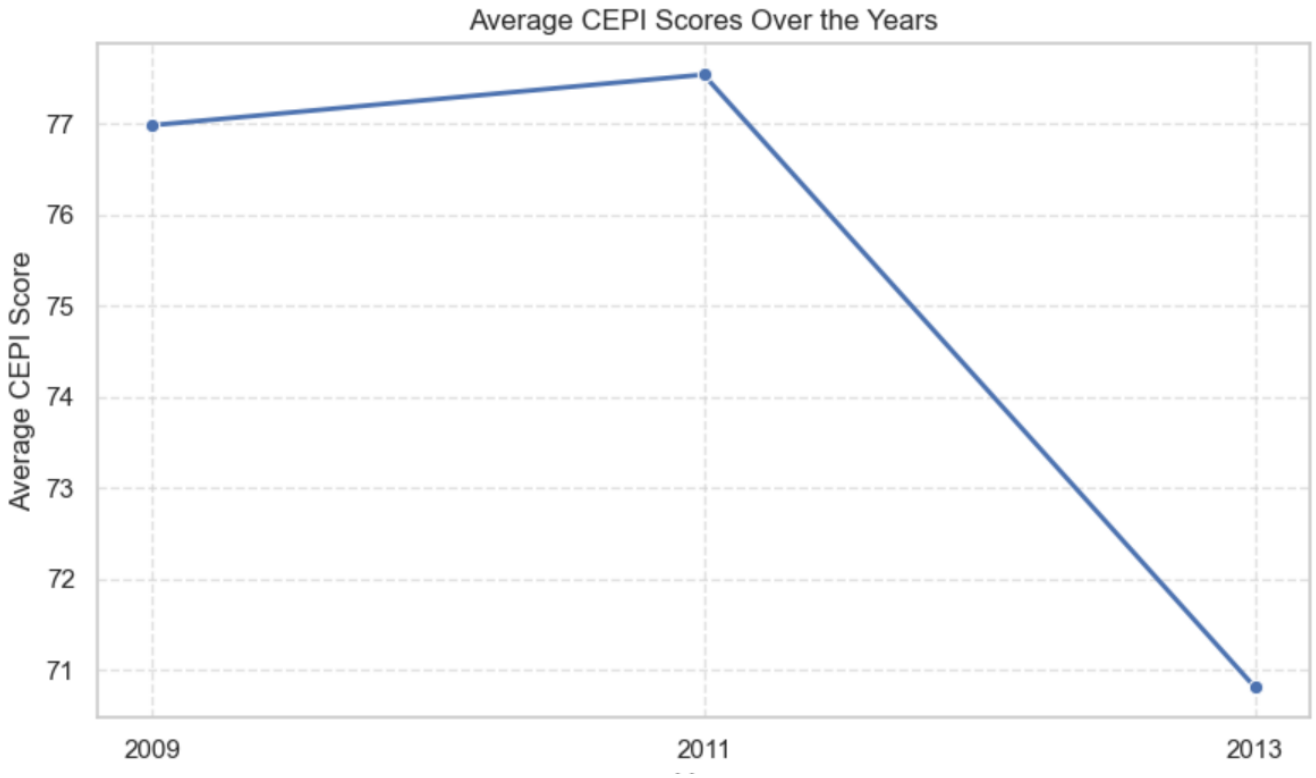
[1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

[2]:
```
df=pd.read_csv("CEPI_scores.csv")
print(df.head())
```

```
         Status of Moratorium Industrial Cluster / Area          State  \
0  Moratorium has been lifted                      Agra  Uttar Pradesh
1  Moratorium has been lifted                 Ahmedabad        Gujarat
2  Moratorium has been lifted              Angul Talcher         Orissa
3  Moratorium has been lifted                   Asansol    West Bengal
4  Moratorium has been lifted                Aurangabad    Maharashtra

   CEPI SCORE-2009  CEPI SCORE-2011  CEPI SCORE-2013
0            76.48            88.36            68.71
1            75.28            78.09            69.54
2            82.09            89.74            72.86
3            70.20            70.96            56.01
4            77.44            83.10            68.87
```

## Average CEPI Scores Over the Years

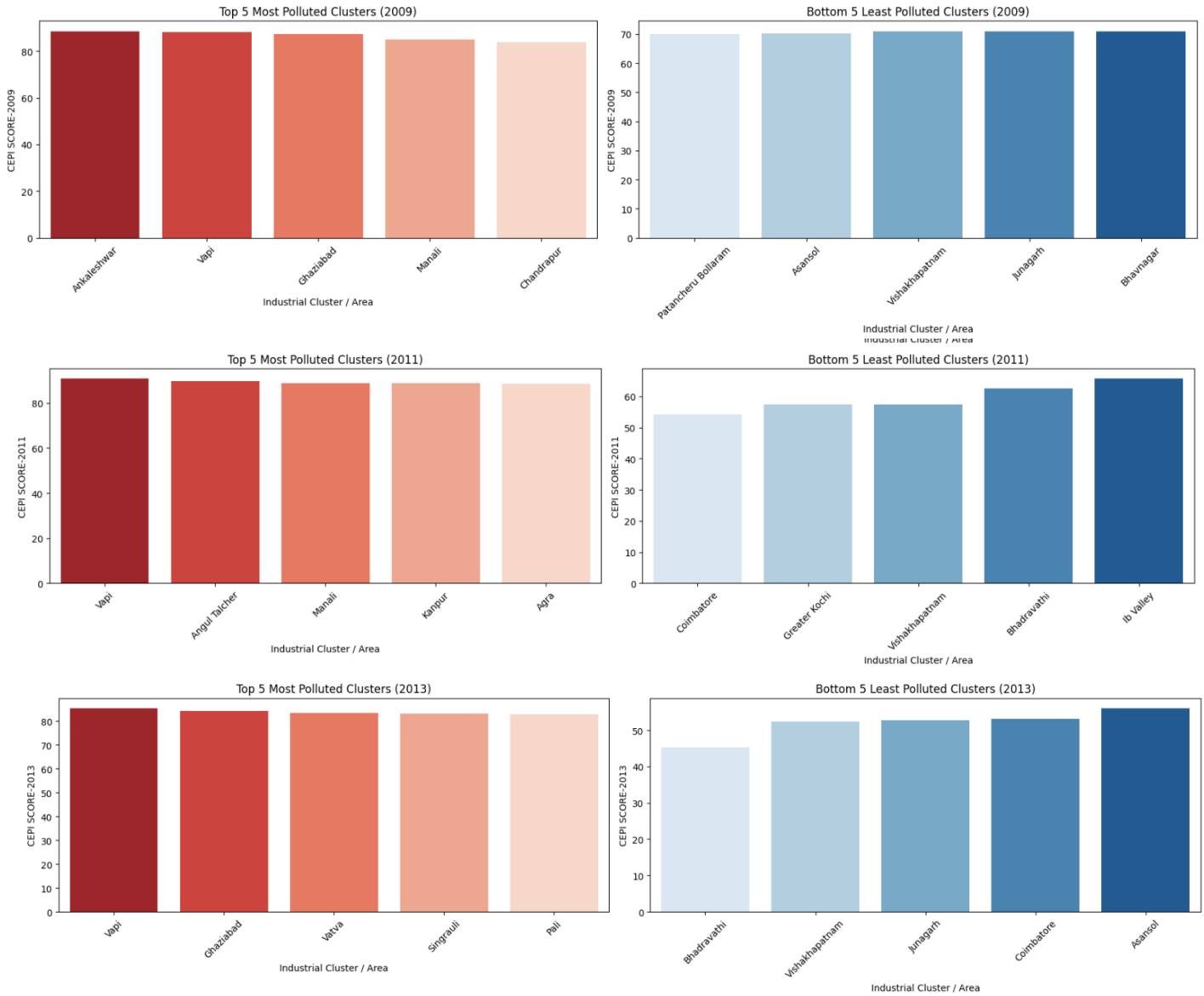
[39]:
```
df_melted = df.melt(
    id_vars=["Industrial Cluster / Area"],
    value_vars=["CEPI SCORE-2009", "CEPI SCORE-2011", "CEPI SCORE-2013"],
    var_name="Year",
    value_name="CEPI Score"
)
df_melted["Year"] = df_melted["Year"].str.extract(r'(\d{4})')
plt.figure(figsize=(8, 5))
sns.lineplot(data=df_melted, x="Year", y="CEPI Score", estimator='mean', errorbar=None, marker='o', linewidth=2)
plt.title("Average CEPI Scores Over the Years")
plt.grid(True, linestyle='--', alpha=0.6)
plt.xlabel("Year")
plt.ylabel("Average CEPI Score")
plt.tight_layout()
plt.show()
```




## Side-by-Side Comparison of Top 5 and Bottom 5 Polluted Clusters for each year

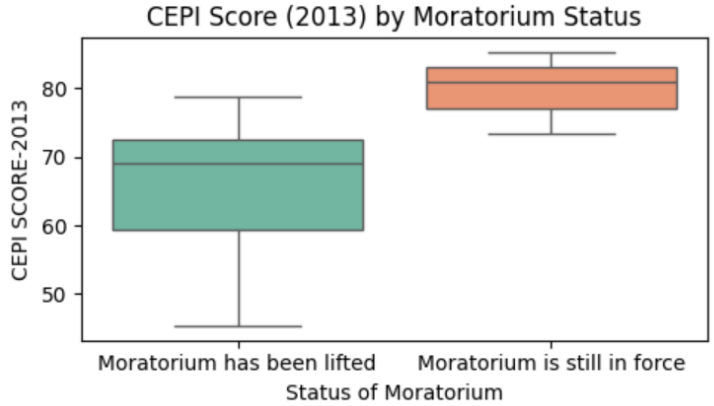
[41]:
```
top_5_2009 = df.sort_values("CEPI SCORE-2009", ascending=False).head(5)
bottom_5_2009 = df.sort_values("CEPI SCORE-2009", ascending=True).head(5)

top_5_2011 = df.sort_values("CEPI SCORE-2011", ascending=False).head(5)
bottom_5_2011 = df.sort_values("CEPI SCORE-2011", ascending=True).head(5)

top_5_2013 = df.sort_values("CEPI SCORE-2013", ascending=False).head(5)
bottom_5_2013 = df.sort_values("CEPI SCORE-2013", ascending=True).head(5)
fig, axes = plt.subplots(3, 2, figsize=(10, 15))

# for 2009
sns.barplot(data=top_5_2009, x="Industrial Cluster / Area", y="CEPI SCORE-2009",
            hue="Industrial Cluster / Area", palette="Reds_r", legend=False, ax=axes[0, 0])
axes[0, 0].set_title("Top 5 Most Polluted Clusters (2009)")
axes[0, 0].tick_params(axis='x', rotation=45)

sns.barplot(data=bottom_5_2009, x="Industrial Cluster / Area", y="CEPI SCORE-2009",
            hue="Industrial Cluster / Area", palette="Blues", legend=False, ax=axes[0, 1])
axes[0, 1].set_title("Bottom 5 Least Polluted Clusters (2009)")
axes[0, 1].tick_params(axis='x', rotation=45)
```

```
# for 2013
sns.barplot(data=top_5_2013, x="Industrial Cluster / Area", y="CEPI SCORE-2013",
            hue="Industrial Cluster / Area", palette="Reds_r", legend=False, ax=axes[2, 0])
axes[2, 0].set_title("Top 5 Most Polluted Clusters (2013)")
axes[2, 0].tick_params(axis='x', rotation=45)

sns.barplot(data=bottom_5_2013, x="Industrial Cluster / Area", y="CEPI SCORE-2013",
            hue="Industrial Cluster / Area", palette="Blues", legend=False, ax=axes[2, 1])
axes[2, 1].set_title("Bottom 5 Least Polluted Clusters (2013)")
axes[2, 1].tick_params(axis='x', rotation=45)
plt.tight_layout()
plt.show()
```



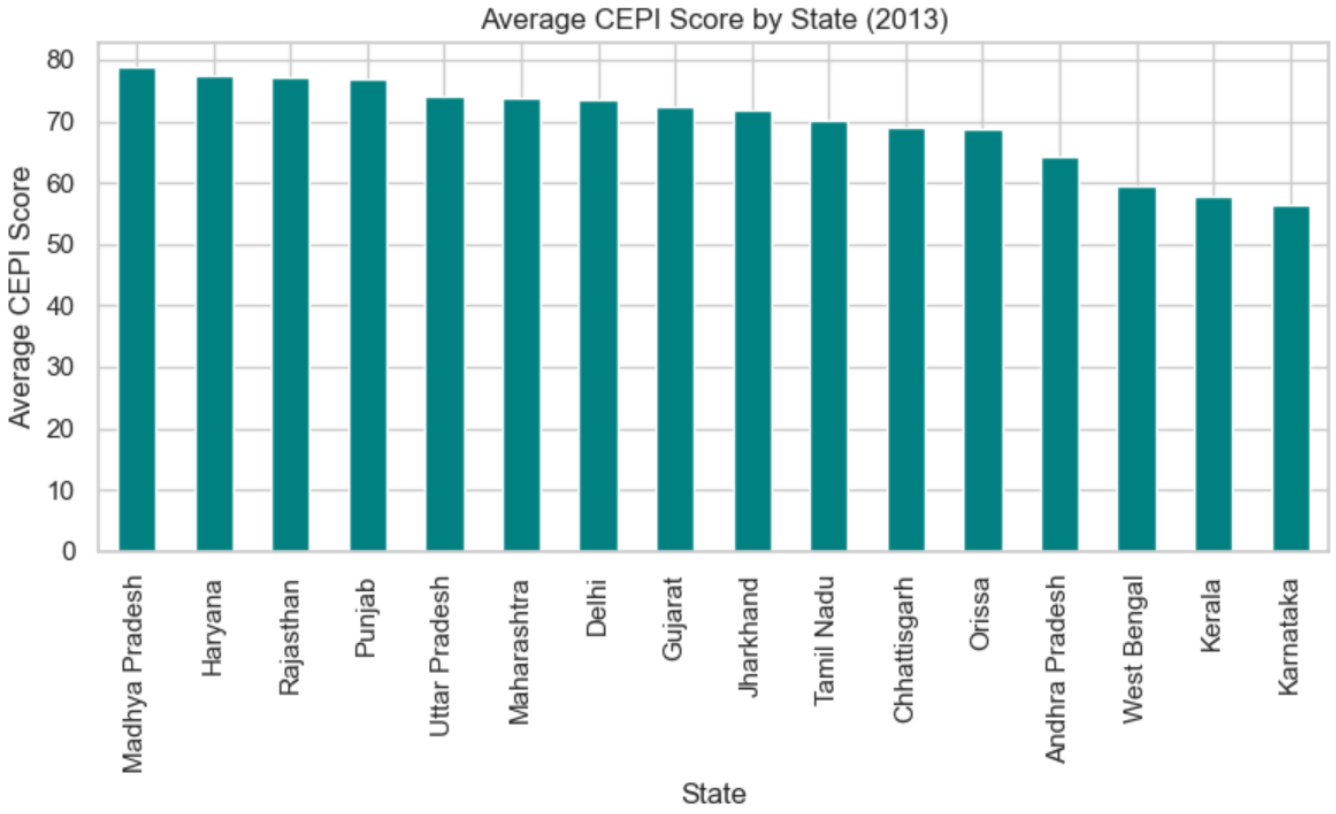

# CEPI Score (2013) by Moratorium Status

[11]:
```
plt.figure(figsize=(5, 3))
sns.boxplot(data=df, x="Status of Moratorium", y="CEPI SCORE-2013", hue="Status of Moratorium", palette="Set2", legend=False)
plt.title("CEPI Score (2013) by Moratorium Status")
plt.tight_layout()
plt.show()
```



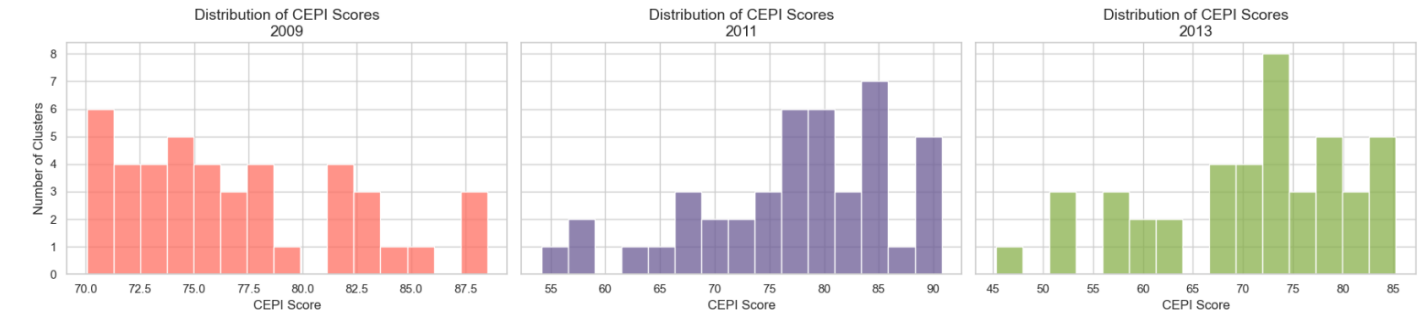
# Average CEPI Score by State (2013)

[31]:
```
statewise_avg = df.groupby("State")["CEPI SCORE-2013"].mean().sort_values(ascending=False)

plt.figure(figsize=(8, 5))
statewise_avg.plot(kind='bar', color='teal')
plt.title("Average CEPI Score by State (2013)")
plt.ylabel("Average CEPI Score")
plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```



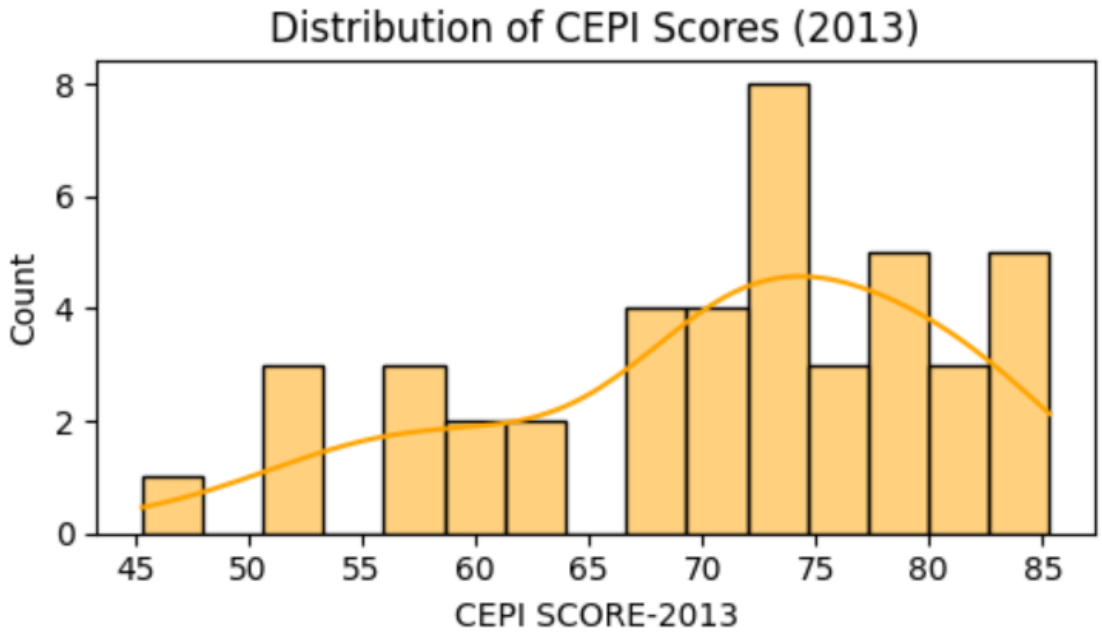

# Side-by-Side Comparison of CEPI Score Distributions (2009, 2011, 2013)

[24]:
```
cepi_years = {
    "CEPI SCORE-2009": "#FF6F61",
    "CEPI SCORE-2011": "#6B5B95",
    "CEPI SCORE-2013": "#88B04B"
}
fig, axes = plt.subplots(1, 3, figsize=(18, 5), sharey=True)
for ax, (col, color) in zip(axes, cepi_years.items()):
    sns.histplot(df[col], bins=15, color=color, ax=ax)
    ax.set_title(f"Distribution of CEPI Scores\n{col[-4:]}", fontsize=14)
    ax.set_xlabel("CEPI Score")
    ax.set_ylabel("Number of Clusters")

plt.suptitle("Side-by-Side Comparison of CEPI Score Distributions (2009, 2011, 2013)", fontsize=16)
plt.tight_layout(rect=[0, 0.03, 1, 0.95])
plt.show()
```



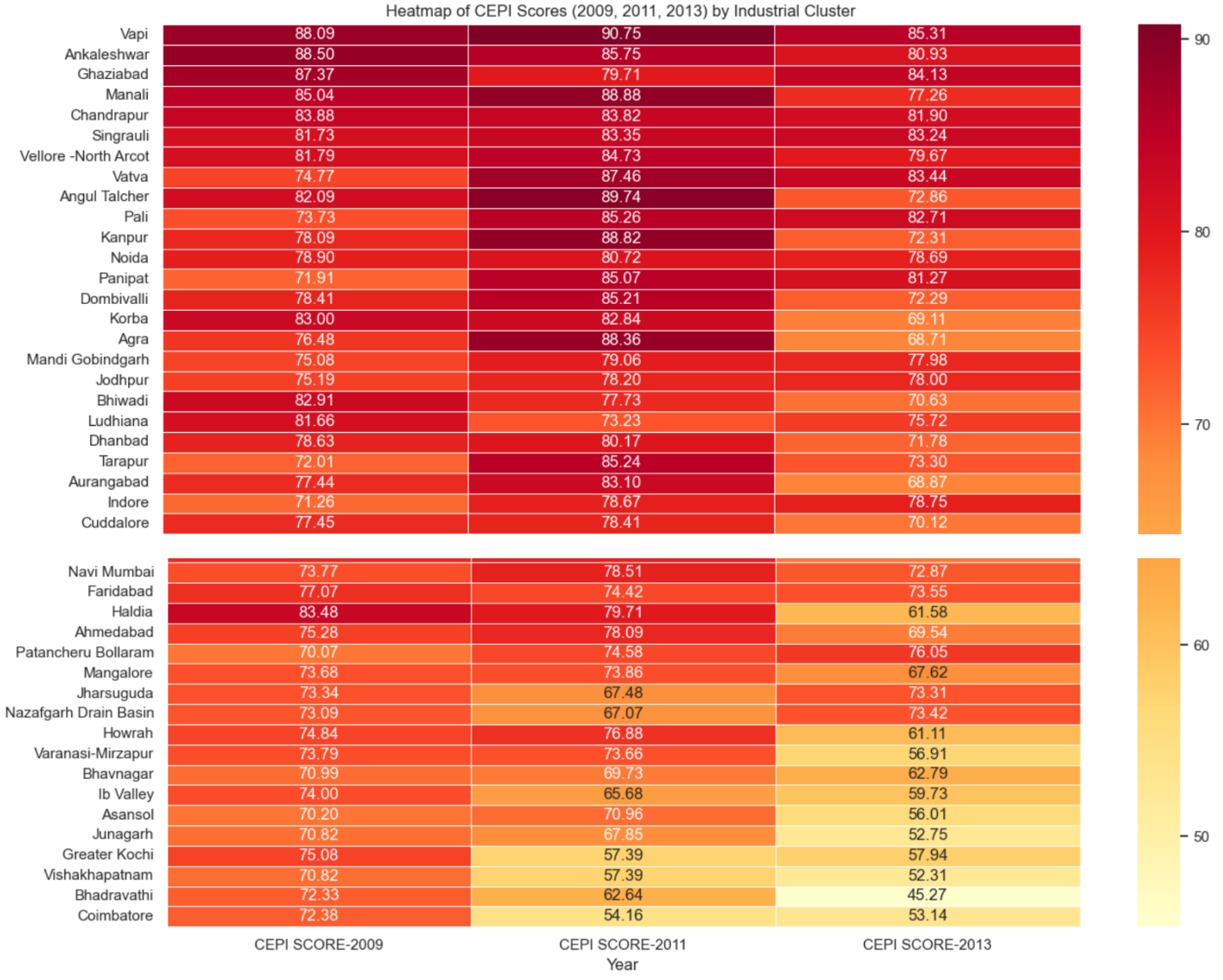

# Distribution of CEPI Scores (2013)

[12]:
```
plt.figure(figsize=(5, 3))
sns.histplot(df["CEPI SCORE-2013"], kde=True, color='orange', bins=15)
plt.title("Distribution of CEPI Scores (2013)")
plt.tight_layout()
plt.show()
```



# Heatmap of CEPI Scores (2009, 2011, 2013) by Industrial Cluster

[25]:
```
heat_df = df.set_index("Industrial Cluster / Area")[["CEPI SCORE-2009", "CEPI SCORE-2011", "CEPI SCORE-2013"]]
heat_df["Average"] = heat_df.mean(axis=1)
heat_df = heat_df.sort_values("Average", ascending=False).drop("Average", axis=1)

plt.figure(figsize=(14, 10))
sns.heatmap(heat_df, annot=True, cmap="YlOrRd", fmt=".2f", linewidths=0.5)
plt.title("Heatmap of CEPI Scores (2009, 2011, 2013) by Industrial Cluster")
plt.xlabel("Year")
plt.ylabel("Industrial Cluster / Area")
plt.tight_layout()
plt.show()
```



### 8.3 Appendix C: Technical Documentation:

#### 8.3.1. Algorithms and Calculations

To perform the CEPI score analysis, several custom calculations were used:

- An **average CEPI score** was calculated for each cluster using values from 2009, 2011, and 2013. This helped to assess the overall pollution level across years.
- A **score change metric** was derived to determine whether pollution increased or decreased in each cluster over time.
- **State-wise average scores** were computed by aggregating CEPI scores for all clusters within each state. This enabled a regional comparison of pollution severity.

#### 8.3.2. Tools and Software Environment

The entire analysis was conducted using Python programming language in an interactive development environment (Jupyter Notebook or Google Colab). The following open-source libraries were used:

- **Pandas**: for data loading, cleaning, and manipulation.
- **NumPy**: for numerical operations.
- **Matplotlib and Seaborn**: for data visualization.

These tools provided a powerful and flexible framework for data analysis and visualization.

#### 8.3.3. Data Structures and Formatting

To simplify the analysis and visualization of CEPI trends over time, the dataset was **reshaped from a wide format to a long format**. This allowed for easier generation of time-series plots and comparative charts.

Categorical variables, such as the **moratorium status**, were converted to a format suitable for grouping and comparison. Additional columns were also added to the dataset to represent calculated values like average scores and score changes.

#### 8.3.4. Statistical Summaries

Summary statistics were generated to understand the central tendency and variability of CEPI scores. This included:

- Mean, median, and standard deviation to assess typical pollution levels.
- Skewness and kurtosis to evaluate the shape and spread of the data distribution.

These statistics provided a deeper understanding of the behavior and trends within the dataset.

## 9. Conclusion

The analysis of CEPI scores from 2009, 2011, and 2013 offers valuable insights into the environmental health of India's industrial clusters. By comparing pollution levels across time, states, and regulatory interventions, this project highlights both areas of improvement and regions where pollution remains a critical concern.

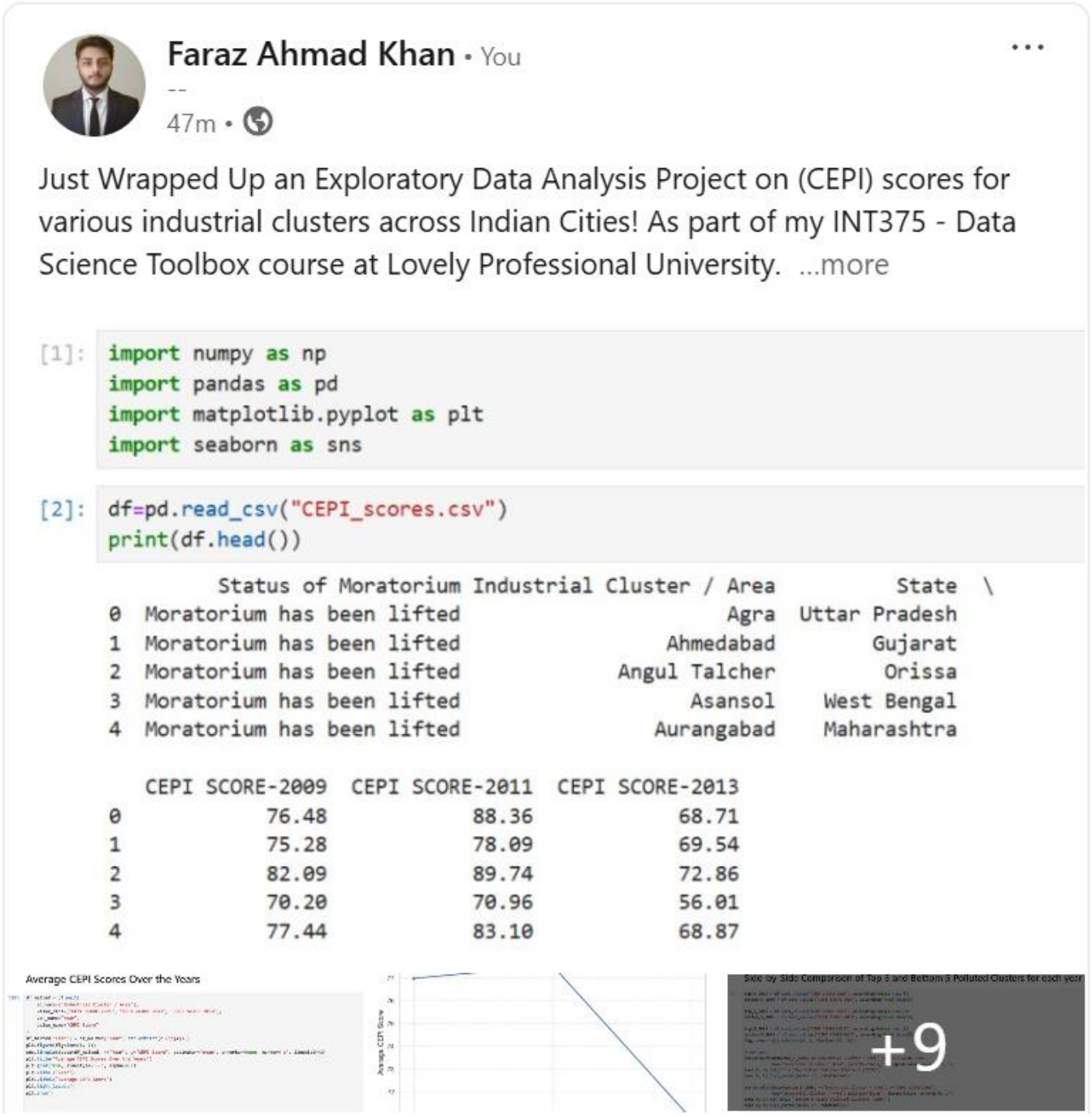
Key findings indicate that while some industrial clusters have shown progress—likely due to moratoriums and policy enforcement—others continue to experience dangerously high pollution levels. The variation in CEPI trends across states also emphasizes the need for region-specific strategies.

This project demonstrates the power of data-driven approaches in environmental monitoring and policy-making. The insights gained can support government agencies, environmental researchers, and industry stakeholders in making informed decisions aimed at pollution control and sustainable development.

In conclusion, while the CEPI scores reflect ongoing challenges, they also offer a roadmap for targeted interventions, highlighting the importance of continued monitoring, public awareness, and collaborative action to protect India's environment and public health.

## LinkedIn Post Snippet

URL: [Click to Check The Linked In Post]




## Git-hub Repository Link: https://github.com/FarazKhan001/Analysis-Of-Comprehensive-Environmental-Pollution-Index-CEPI-Scores-Of-India