DATA ANALYTICS BOOT CAMP | PROJECT 1 | 15 JANUARY 2024

# Impact of city elevation on air quality

Lulu Hellings, Richard Poole, Sameer Erande & Vijay Mani

THE UNIVERSITY OF WESTERN AUSTRALIA

edx by 2U

INTRODUCTION

# Overview

Is the perception that 'air quality increases as elevation increases' correct?

INTRODUCTION

# Research Questions

1. Is there a correlation in air quality between cities in the same country that have significant elevation differences?

e.g. Delhi and Mumbai

INTRODUCTION

# Research Questions

2. How does air quality compare between cities in two similar countries with different elevations and populations?

e.g. Chengdu, China and Mumbai, India

INTRODUCTION

# Research Questions

3. Is there a strong correlation between city elevation and air quality across the world?

e.g. 94 countries

DEFINITIONS

# Air Quality Based on Man-Made Pollutants

1. $NO_2$ Nitrogen Dioxide
2. PM25 Particle Matter 25
3. PM10 Particle Matter 10
4. $O_3$ Ozone
5. $SO_2$ Sulfur Dioxide
6. CO Carbon Monoxide

DEFINITIONS

# Air Quality Based on Man-Made Pollutants

1. $NO_2$ Nitrogen Dioxide
2. PM25 Particle Matter 25
3. PM10 Particle Matter 10
4. $O_3$ Ozone
5. $SO_2$ Sulfur Dioxide
6. CO Carbon Monoxide

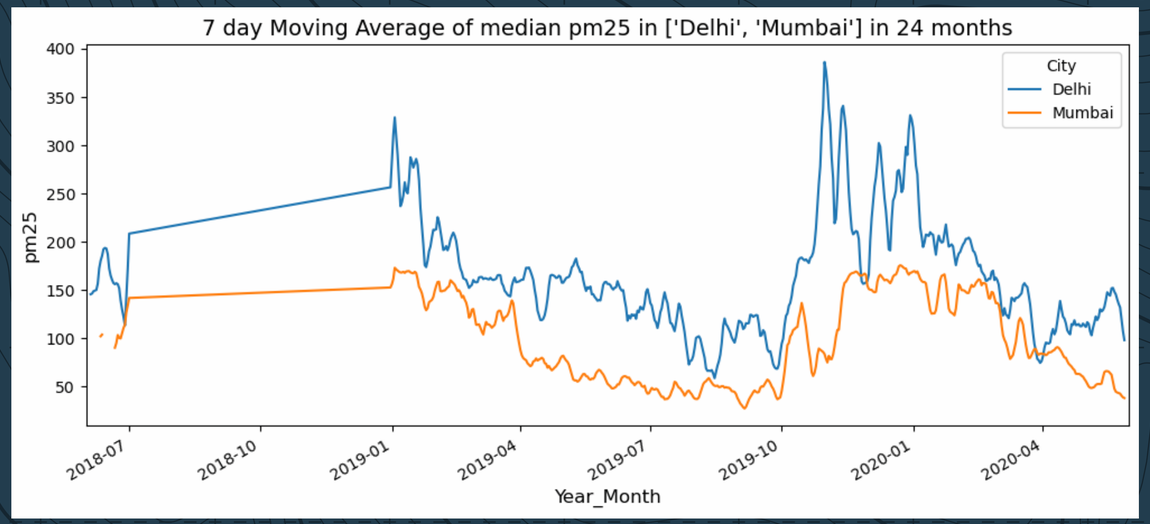
PYTHON GRAPHS

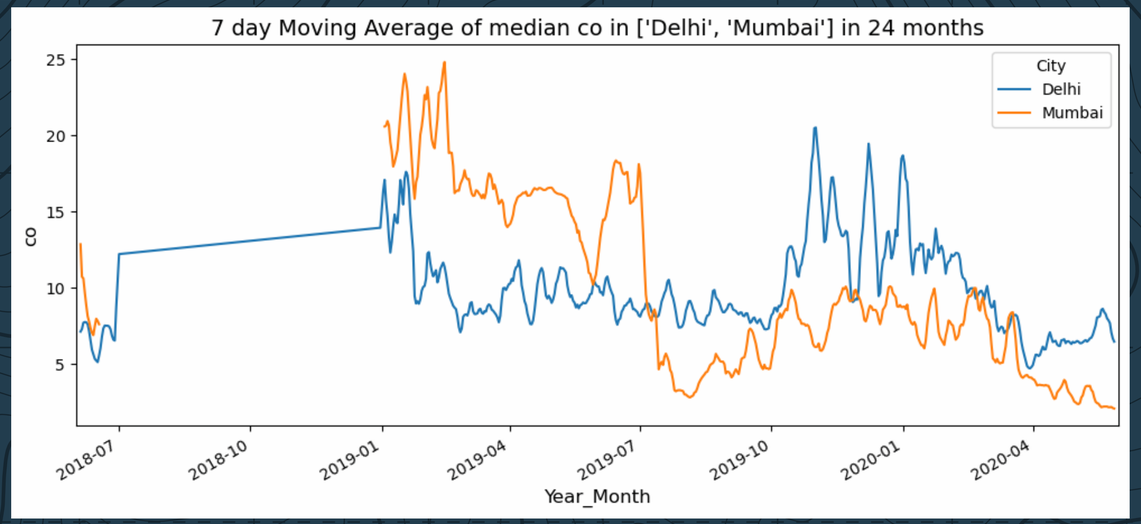
# Results

1. Is there a correlation in air quality between cities in the same country that have significant elevation differences?

**Delhi**
Population 11m, Elevation 227m

**Mumbai**
Population 12m, Elevation 8m



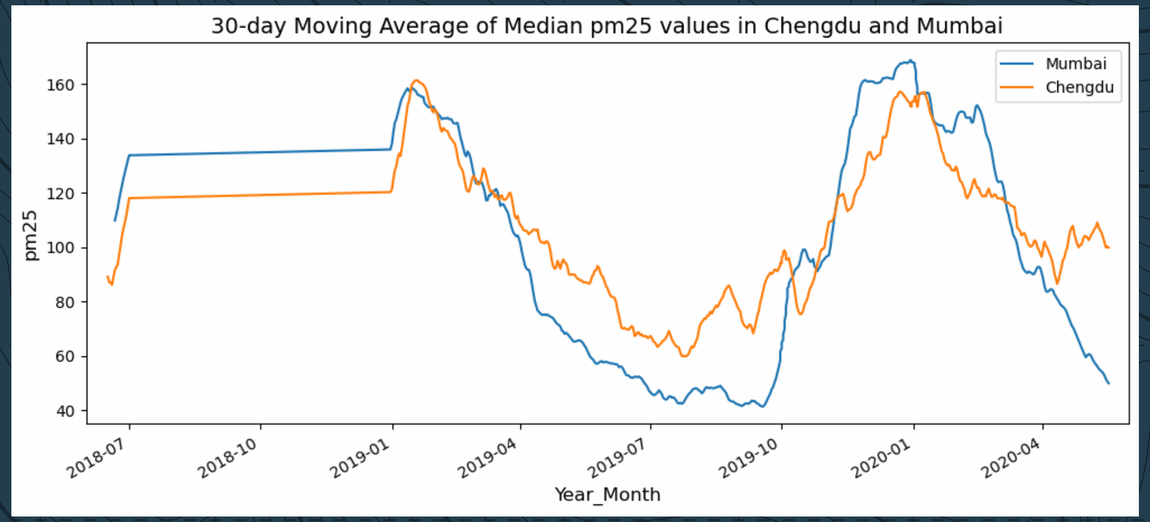
PYTHON GRAPHS

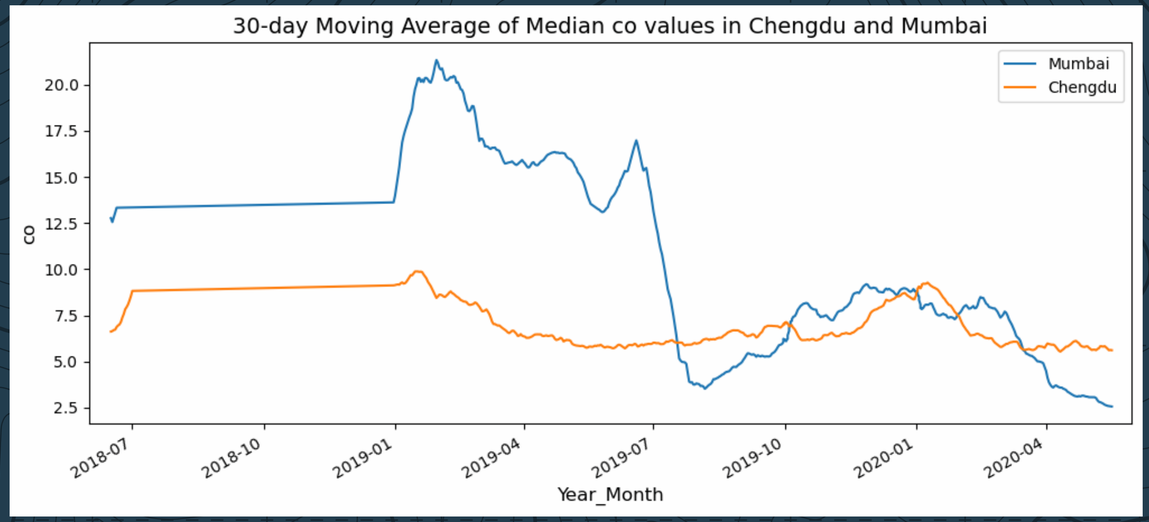
# Results

2. How does air quality compare between cities in two similar countries with different elevations and populations?

**Chengdu, China**
Population 21m, Elevation 529m

**Mumbai, India**
Population 12m, Elevation 8m

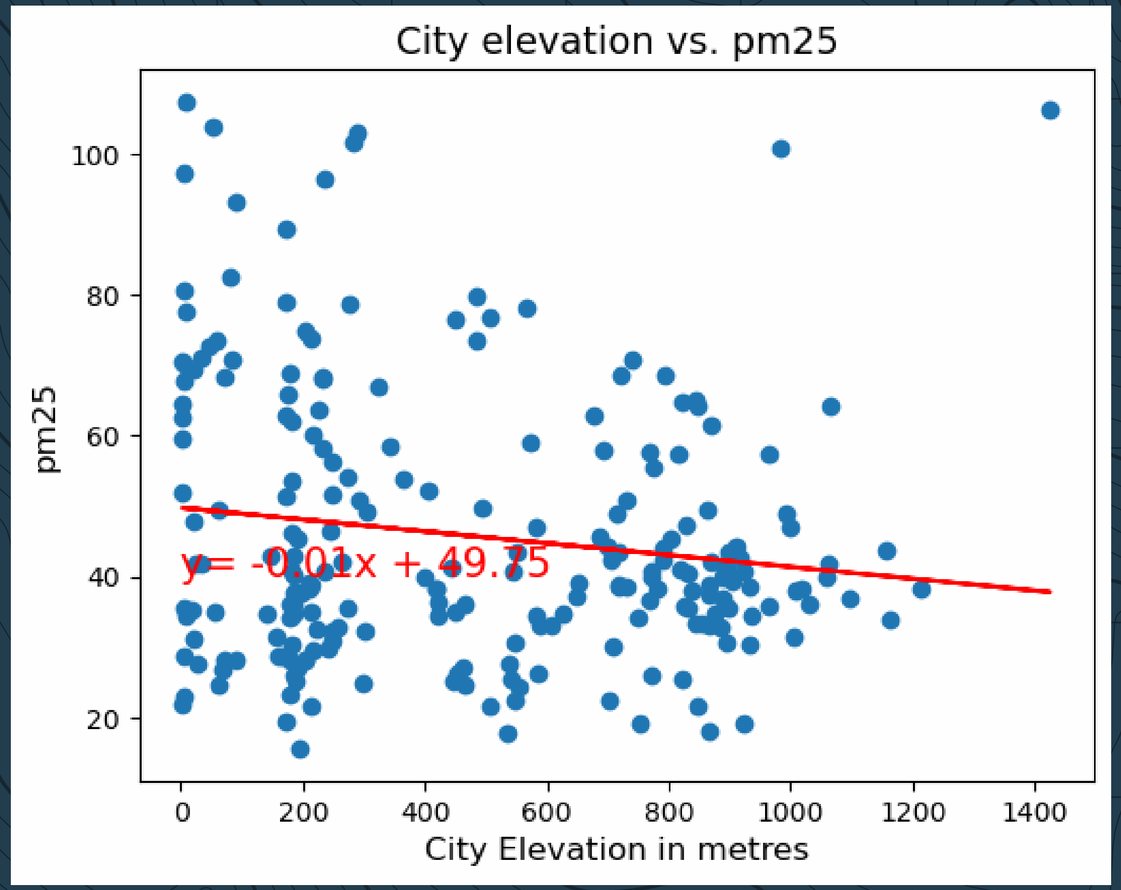


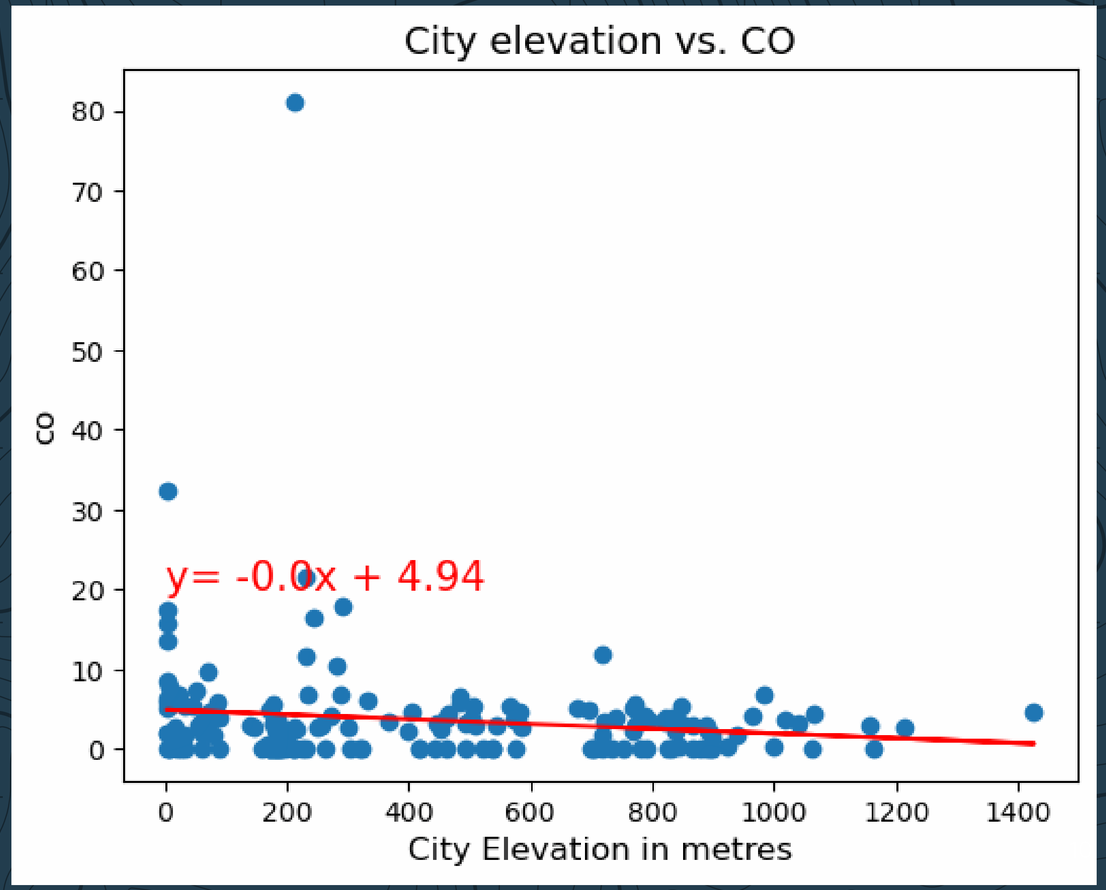



PYTHON GRAPHS

# Results

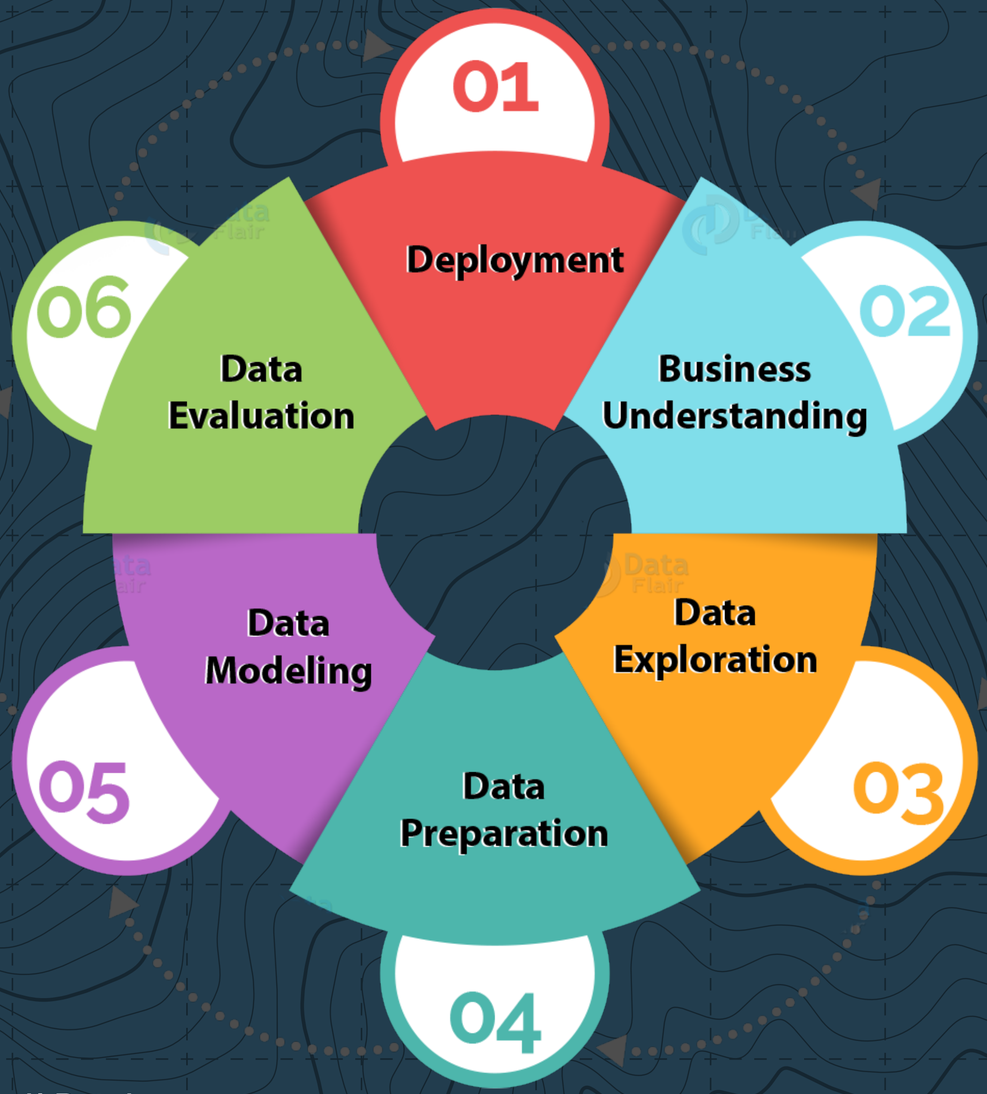
3. Is there a strong correlation between city elevation and air quality across the world? e.g. 94 countries



METHODOLOGY

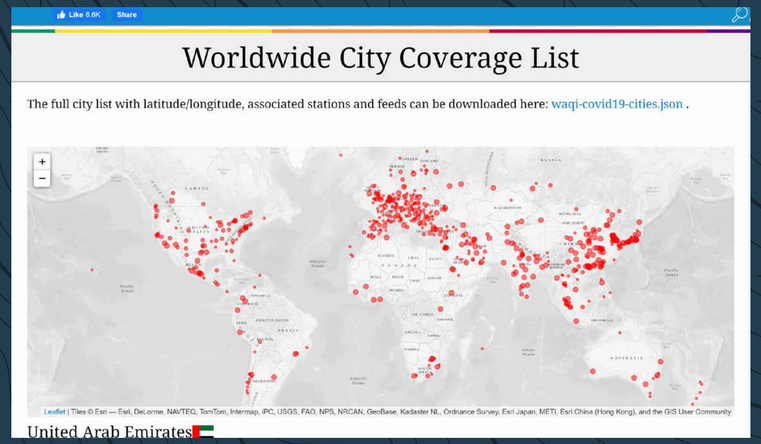
# Data Analytics Process



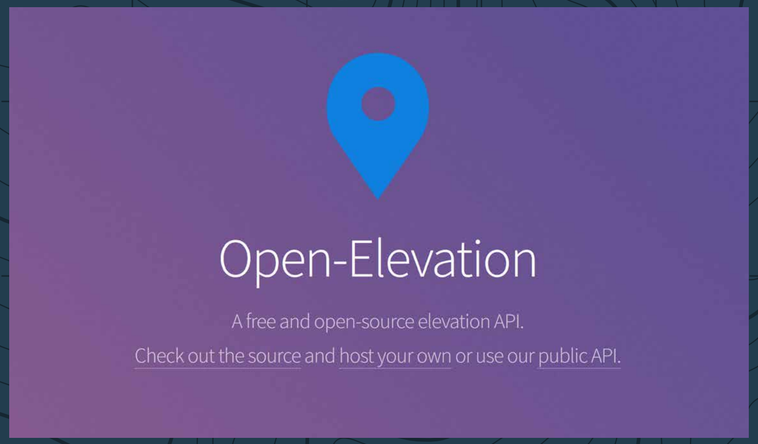

Source: 'What is Big Data Analytics and Why it is so Important?' article by Harshali Patel

02 **Defining Project Scope**

03 **Data Exporation**

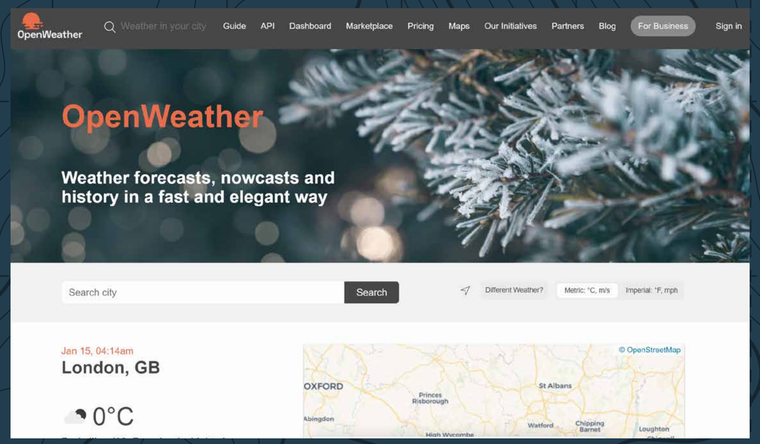



## Air Quality Open Data Platform




## Open Elevation

Elevation data




## Open Weather

City longitude and latitude data

## 04 Data Preparation

### Importing datasets of air quality data

```
# Loop through historical air quality data csv files in the historical_data folder
periods = ["2020", "2019Q1", "2019Q2", "2019Q3", "2019Q4","2015H1","2016H1","2017H1","2018H1"]

df_list = list()

for period in periods:
    path = f"historical_data/waqi-covid19-airqualitydata-{period}.csv"
    df = pd.read_csv(path)
    df_list.append(df)
```

```
# Combine individual air quality dataframes into one
airdf_2015_2020 = pd.concat(df_list, ignore_index=True)
```

```
airdf_2015_2020.head()
```

| | Date | Country | City | Specie | count | min | max | median | variance |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 31/05/2020 | IR | Isfahan | temperature | 120 | 17.5 | 35.0 | 27.5 | 331.51 |
| 1 | 13/06/2020 | IR | Isfahan | temperature | 144 | 16.0 | 36.5 | 27.5 | 488.74 |
| 2 | 3/07/2020 | IR | Isfahan | temperature | 67 | 19.0 | 33.0 | 24.0 | 128.08 |
| 3 | 28/03/2020 | IR | Isfahan | temperature | 240 | 3.0 | 14.0 | 9.5 | 136.68 |
| 4 | 23/04/2020 | IR | Isfahan | temperature | 168 | 6.0 | 25.5 | 16.0 | 400.79 |

## 05 Data Modelling

### Slicing and dicing the data

```
# Display an overview of the Specie column
airdf_2015_2020["Specie"].unique()
```

| | Date | Country | City | Specie | count | min | max | median | variance |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 2020-02-24 | IR | Isfahan | pm25 | 129 | 54.0 | 194.0 | 126.0 | 10921.40 |
| 1 | 2020-05-07 | IR | Isfahan | pm25 | 168 | 17.0 | 168.0 | 91.0 | 14014.00 |
| 2 | 2020-05-28 | IR | Isfahan | pm25 | 127 | 17.0 | 115.0 | 72.0 | 3558.56 |
| 3 | 2020-02-20 | IR | Isfahan | pm25 | 113 | 26.0 | 181.0 | 76.0 | 11209.80 |
| 4 | 2020-02-23 | IR | Isfahan | pm25 | 132 | 22.0 | 132.0 | 76.0 | 3209.67 |

```
clean_airdf_cols_trunc = clean_airdf[['City', 'Country', 'Specie', 'median', 'Date']]
clean_airdf_cols_trunc.head()
```

| | City | Country | Specie | median | Date |
|---|---|---|---|---|---|
| 0 | Isfahan | IR | pm25 | 126.0 | 2020-02-24 |
| 1 | Isfahan | IR | pm25 | 91.0 | 2020-05-07 |
| 2 | Isfahan | IR | pm25 | 72.0 | 2020-05-28 |
| 3 | Isfahan | IR | pm25 | 76.0 | 2020-02-20 |
| 4 | Isfahan | IR | pm25 | 76.0 | 2020-02-23 |

```
# Transpose the clean_airdf so that the species will become the columns.
clean_airdf_transposed = clean_airdf.pivot_table(
    index=["Date", "Country", "City"], columns="Specie", values="median").copy()
clean_airdf_transposed.reset_index(inplace=True)
```

```
clean_airdf_transposed.head()
```

| Specie | Date | Country | City | co | no2 | o3 | pm10 | pm25 | so2 |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 2018-06-01 | AE | Abu Dhabi | 0.1 | 15.1 | 37.4 | 102.0 | 165.0 | 13.7 |
| 1 | 2018-06-01 | AE | Dubai | NaN | NaN | NaN | NaN | 172.0 | NaN |
| 2 | 2018-06-01 | AR | Buenos Aires | 3.5 | 6.9 | 9.0 | 16.0 | 34.0 | 3.1 |
| 3 | 2018-06-01 | AT | Graz | 0.1 | 8.0 | NaN | 21.0 | NaN | 1.1 |
| 4 | 2018-06-01 | AT | Innsbruck | 0.1 | 11.7 | NaN | 14.0 | NaN | 0.6 |

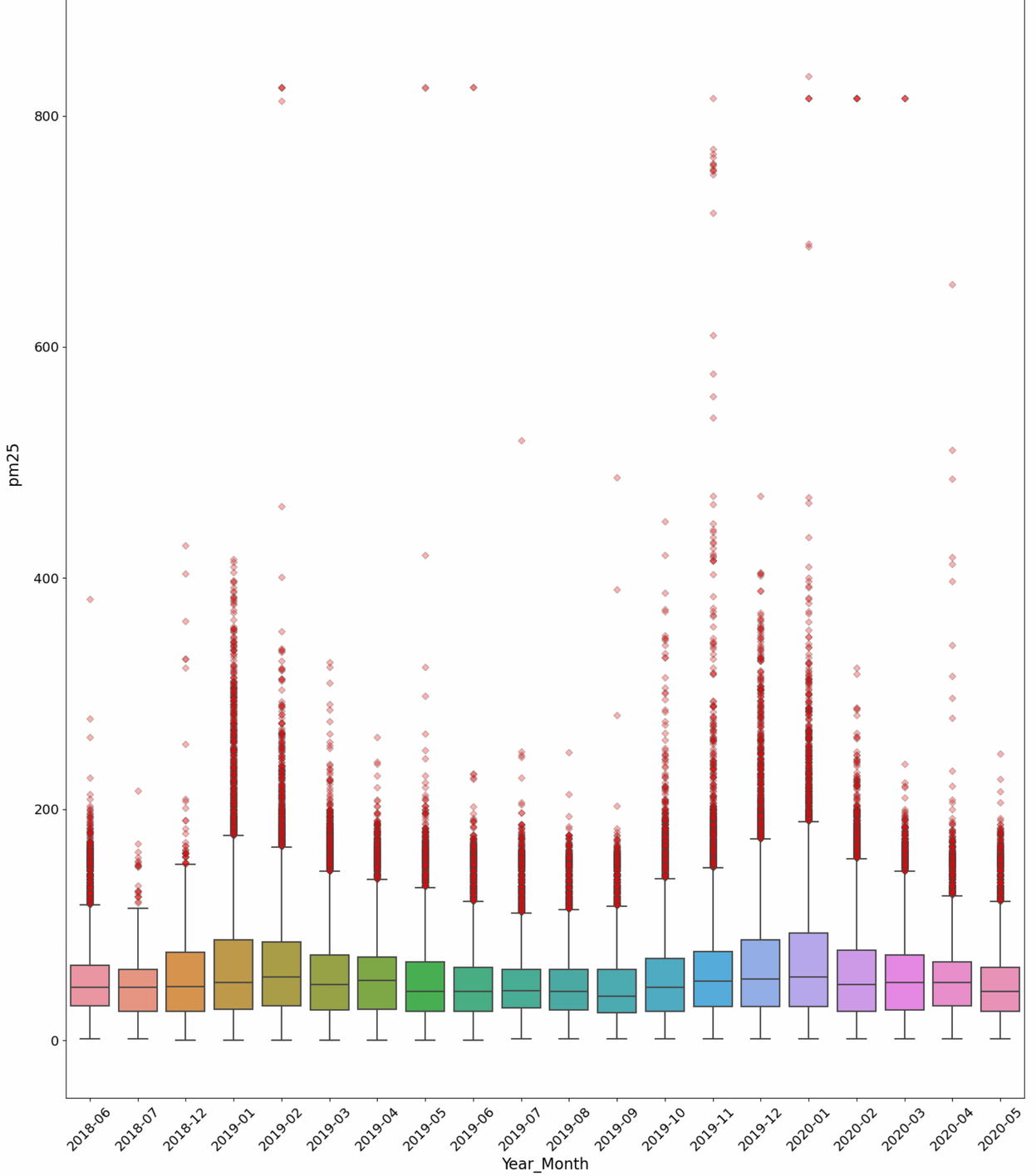
## 06 Data Evaluation

```
# Define a function to draw boxplots showing monthly distribution of median values of different air pollu
def specie_boxplot(specie):
    fig, ax = plt.subplots(figsize=(18, 24))

    red_diamond = dict(markerfacecolor='r', marker='D', alpha=0.3)
    sns.boxplot(data=clean_airdf_transposed, x='year_month',
                y=specie, flierprops=red_diamond)

    plt.xlabel("Year_Month", fontsize=16)
    plt.ylabel(specie, fontsize=16)

    plt.xticks(fontsize=14, rotation=45)
    plt.yticks(fontsize=14)

    plt.title(
        f"Monthly Distribution of Median {specie} in 94 countries between ({starting_date} and {end_date}
    plt.savefig(f"images/boxplot_{specie}.png")
```

```
specie_boxplot("pm25")
```

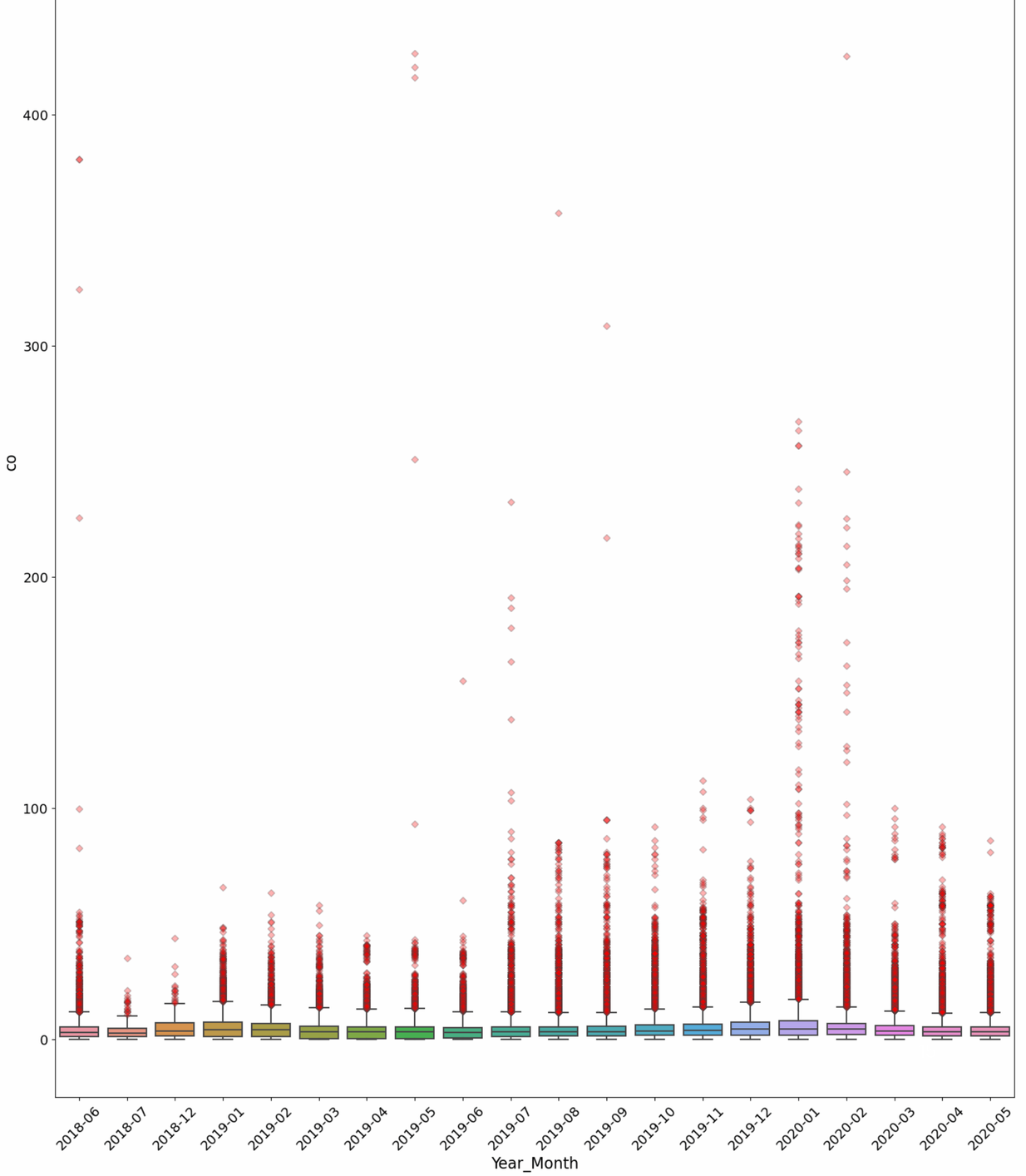
06 **Data Evaluation**

# Monthly distribution of median PM25 in 94 countries between 2018-2020, 24 month period



06 **Data Evaluation**

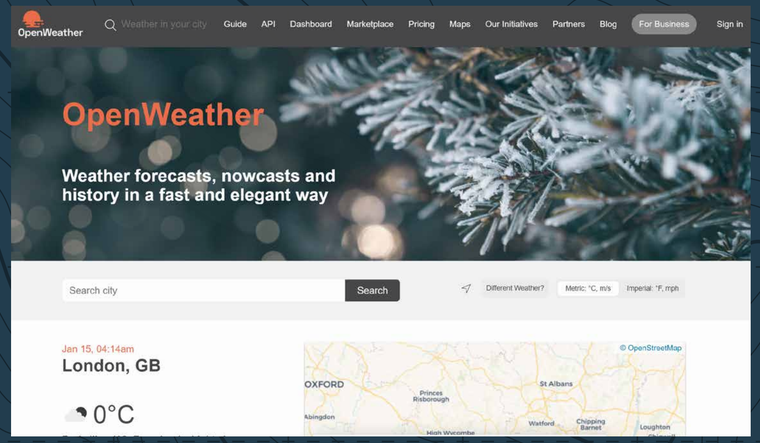
# Monthly distribution of median CO in 94 countries between 2018-2020, 24 month period



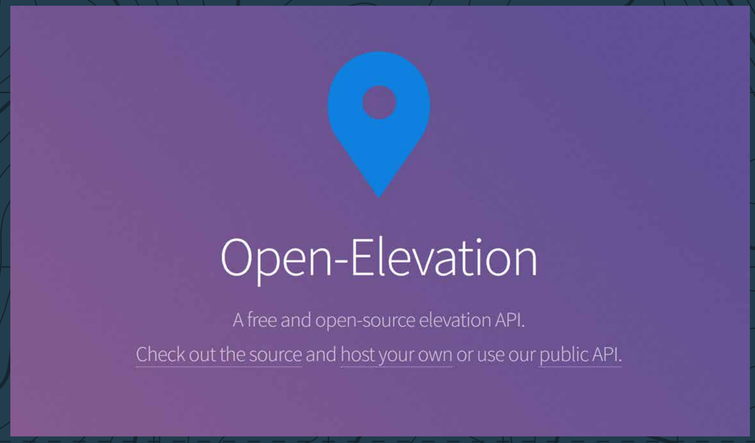
WRITING CODE TO USE API'S

# Sourcing Elevation Data

Two APIs were used to get the required cities' elevation data:




+



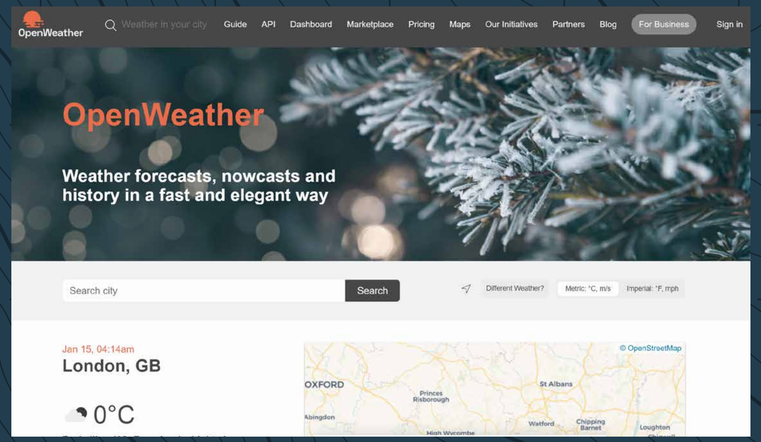

## Open Weather

openweathermap.org

## Open Elevation

open-elevation.com

DATA ANALYTICS BOOT CAMP | PROJECT 1 | 15 JANUARY 2024




# Open Weather

openweathermap.org

```
#base_url1 = http://api.openweathermap.org/geo/1.0/direct?q={city name}&limit=5&appid={API key}

url = "http://api.openweathermap.org/geo/1.0/direct?"
lat_lon_list=[]

# Iterate through the arrays, clean_airdf["City"].unique()
for i in clean_airdf["City"].unique():
    # Create endpoint URL with each city
    city_url = f"{url}q={i}&appid={weather_api_key}"
    # Make an API request
    lat_lon = requests.get(city_url)

    # Convert the API response to JSON format
    lat_lon = lat_lon.json()
    #print(json.dumps(lat_lon, indent=4 ))

    for record in lat_lon:
        lat=record['lat']
        lon=record['lon']
        name=record['name']
        lat_lon_list.append({'lat': lat,
                             'lon': lon,
                             'City': name})
```

```
#Converting list into dataframe
lat_lon_list_df = pd.DataFrame(lat_lon_list)
```

```
url2 = 'https://api.open-elevation.com/api/v1/lookup?locations='
elevation_list = []

# Iterate through the combined DataFrame
for index, row in lat_lon_list_df.iterrows():
    # get lat and lon from the DataFrame
    lat1 = row['lat']
    lon1 = row['lon']
    # Create endpoint URL with each lat and lon
    elevation_url = f"{url2}{lat1},{lon}"
    try:
    # Make and API request
        elevation_data = requests.get(elevation_url)

    # Convert the API response to JSON format
        elevation_data = elevation_data.json()
        #print(json.dumps(elevation_data, indent=4 ))
        for record in elevation_data['results']:
            elevation=record['elevation']
```

DATA ANALYTICS BOOT CAMP | PROJECT 1 | 15 JANUARY 2024


# Open Elevation

open-elevation.com

```
url2 = 'https://api.open-elevation.com/api/v1/lookup?locations='
elevation_list = []

# Iterate through the combined DataFrame
for index, row in lat_lon_list_df.iterrows():
    # get lat and lon from the DataFrame
    lat1 = row['lat']
    lon1 = row['lon']
    # Create endpoint URL with each lat and lon
    elevation_url = f"{url2}{lat1},{lon}"
    try:
    # Make and API request
        elevation_data = requests.get(elevation_url)

    # Convert the API response to JSON format
        elevation_data = elevation_data.json()
        #print(json.dumps(elevation_data, indent=4 ))
        for record in elevation_data['results']:
            elevation=record['elevation']
            lat=record['latitude']
            elevation_list.append({'elevation': elevation,
                                   'lat': lat})
    # If an error is experienced, skip the city
    except:
        print("Data not found. Skipping...")
```

```
##Converting list into dataframe
elevation_list_df = pd.DataFrame(elevation_list)
elevation_list_df = elevation_list_df[['lat', 'elevation']]
```

```
#merging two dataframes from the above two APi
elevation_final_df = pd.merge(lat_lon_list_df, elevation_list_df, left_index=True, right_index=True)
elevation_final_df.head()
```

| | lat_x | lon | City | lat_y | elevation |
|---|---|---|---|---|---|
| 0 | 32.670788 | 51.665000 | Isfahan | 32.670788 | 0.0 |
| 1 | 34.086520 | 49.688842 | Arak | 34.086520 | 0.0 |
| 2 | 35.822489 | 50.990468 | Karaj | 35.822489 | 0.0 |
| 3 | 34.642294 | 50.880118 | Qom | 34.642294 | 0.0 |
| 4 | 37.548356 | 45.066755 | Urmia | 37.548356 | 64.0 |

# Conclusions

Correlation between air pollution and altitude by theory:
in higher altitudes, air pollution measures less.

REASONS

- Higher altitudes usually have less human activities and more trees
- Low air density encourages dispersion and wind circulation to dilute more efficiently
- Closer to sunlight source, stronger photochemical processes, better degradation of some pollutants (such as $SO_2$, car emissions & $NO_2$)

# Conclusions

When studies are conducted, the results are often significantly impacted by the cities' industrial activities, their strategy and implementation of pollution management, and population. The behaviour pattern of nature is altered or masked by man-made effects.

All 4 cities show some extent of decrease in most of the pollutants during COVID-19 period, which further supports the assumption that human activities impact on environment and make the correlation between altitude and air quality difficult to reveal.