Constructing an Air Quality and Weather Dataset for Environmental Factors Analysis for the City of Milan

Eduardo Mosca - e.mosca11@campus.unimib.it

Rebecca Micol Finzi - r.finzi1@campus.unimib.it

Elisa Princic - e.princic@campus.unimib.it

Università degli Studi di Milano-Bicocca
MSc Data Science: Data Management Course Project

Abstract

In the context of the Data Management course of the Master's Degree in Data Science at Università degli Studi di Milano-Bicocca, a project was developed with the objective of constructing a comprehensive dataset to enable recent-year analysis through gathering, investigating and processing data related to air quality and weather/atmospheric characteristics for the city of Milan. Mainly thanks to the services offered by Open-Meteo, and a Kaggle dataset also obtained through the latter, a final base was built representing daily environmental factors aggregated from hourly granularity and aligned with local Milan time, to better reflect the city inhabitants' perception and the way the could be affected. Starting from an API call to retrieve air quality data and the aforementioned download, both sources were profiled and their provenance traced back to reanalysis models of European initiative, particular attention was given to verifying the data's quality, with the main focus put on it's syntactic accuracy, completeness and consistency. The two sources were then aggregated on the date field, resulting in a validated environmental factors dataset of quality, upon which further trend analysis, descriptive analysis and the training of machine learning models could be carried out to extract valuable insights from the data.

Contents

1 Introduction

Milan is located in the north of Italy, and with vast presence of industrial and agricultural machinery, coupled with the busy nature of the city, the quality of it's air hasn't been given the deserved attention in recent years, and is known to be among the worst in Europe. We decided to work on the creation of this dataset in order to enable analysis on the recent dynamics not only with respect to air pollutants, but also weather factors which are known to play a major role when it comes to air quality. This was done in hopes of finding useful connections and interactions between the two, and hopefully find ways to improve the current situation.

1.1 Software architecture used

All API calls, operations and integration steps were carried out through Python, mainly through the Pandas library.

1.2 How to follow procedure and replication

In the project folders, the notebooks and intermediate datasets needed to follow our procedure are grouped in relation to data source, while for the integration step procedure we provide a separate folder. In each folder you will find a "READTHIS.txt" text file describing how to navigate that folder's files in the intended order.

2 Air Quality Data

2.1 Data Profiling

The dataset was downloaded through Open-Meteo's Air Quality API(/https://open-meteo.com/en/docs/air-quality-api). The data is provided through the use of reanalyses models(more info in data provenance section) The data obtained is in hourly temporal resolution and ranges from December 31st, 2019 to December 31st 2023. The data was requested specifically for Milan. The variables included are the following:

- time, which includes both the date and time relative to the record. It is in format ISO 8601 standard GMT+00;
- `pm10`, the value of pm10 (particulate matter with diameter smaller than 10 µm);
- `pm2_5`, the value of pm2_5(particulate matter with diameter smaller than 2.5 µm);
- `nitrogen_dioxide`, value of NO2;
- `sulphur_dioxide`, value of SO2;
- `ozone`, value of O3.

The pollutants are expressed in µg/m3, refer to the pollutant value in a particular instant in time(timestamp of measurement) and refer to the value at 10 meters above ground.

The data collection process can be followed in the AQ_APIdatacollection notebook file.

2.1.1 Alternative Air Quality Data Sources

In an earlier stage of the project, different data providers were considered with regards to collecting air quality data; we spent non-negligible time with OpenWeatherMap's AQ API in mind. Upon further inspection, impossible values were found in the data, like pollutant measurements in the thousands or even taking negative values, we reached out to OWM to signal the issue, but in the meantime we evaluated the using the API provided by Open-Meteo, and the results were satisfying.

2.2 Data Provenance

Verifying the data provenance of the air quality dataset obtained, we discover that the API calls refer to the CAMS European air quality reanalyses dataset(https://ads.atmosphere.copernicus.eu/datasets/cams-europe-air-quality-reanalyses?tab=overview), in turn this dataset is obtained through forming a median ensemble from eleven air quality data assimilation systems across Europe at 0.1° (11 km) spatial resolution. Initially these are forecast values and un-validated(interim reanalysis), once observational data provided by the European Environment Agency(EEA) is available, a final validated annual reanalysis is offered and integrated into the dataset. Interim reanalyses are provided for the previous year as a stand-in for validated data, and the latter takes a whole year to be quality-controlled and made available, therefore at the time of development for this project(January 2025) we can only consider the data up to 2023 to be validated and reliable as 2024 data is to be taken as under review.

Links for further information are provided in the references section.

2.3 Data cleaning

There are no missing values and no duplicates in the API data. Outliers are not considered since they are extreme values for the pollutants and therefore interesting for the analysis.

2.4 Data Processing

Given the data is in hourly format and in GMT+00 timezone, we converted it to local Milan time(CET or CEST depending on time of year) by adding an hour to all records(GMT+01) and a further hour to records in periods where daylight savings time was active(between last Sunday of March and last Sunday of October). This was done to make our data describe the perceived days in Milan more accurately.

This passage can be followed in the AQ-data_tolocaltime.ipynb file.

2.5 Data Quality

To make sure our data was reliable and of quality, and evaluating the tradeoff between different quality dimensions, we chose to investigate the extent to which our dataset satisfied the accuracy, completeness and consistency principles. We give the definitions used in our work for clarity:

- Accuracy: degree of correctness in describing phenomenon of interest of the data, whether semantic(values correctly describe real-life phenomenon) or syntactic(values taken represent real-life phenomenon's domain/range of reference accurately i.e. the values in the data are possible in real life).
- Completeness: extent to which data can fully describe phenomenon, with regards to missing values(row/column/dataset completeness) or known quantity of data needed to describe phenomenon(object completeness, example: we know we need 365 daily records to describe something happening inside of a 1-year time interval).
- Consistency: degree to which data is consistent in describing real life phenomenon, it can focus on consistency of records in actually referring to the intended object(rows for a day actually refer to that day, harder one to verify), or it can focus on consistency in intended formats(i.e. all timestamps or addresses are standardized to a format and comply with it).

With regards to accuracy, only the syntactic kind was verifiable through comparing values with reference domains, for semantic accuracy we must rely on reliability of data provider. For completeness we evaluated missing values in the data and verified that object completeness relating to our time period of interest was satisfied. For consistency, we made sure all time-related entries followed the same intended format. Through checks and computation of metrics no fault in the data was found by analyzing it to the extent possible. This passage can be followed in the AQ-dataquality.ipynb file.

2.6 Granularity of the dataset

In the current dataset, we have hourly information on air quality, but we will aggregate to daily for the following reasons: to discover the daily air quality trends in recent years(descriptive analysis), and to reduce dataset size(to allow for smoother processing for ML purposes).

3 Weather Data

3.1 Data profiling

The dataset used for this analysis contains weather data and was downloaded from the Kaggle platform, where it is available under the name Forbes Top 100 Cities Weather Data. This original dataset includes weather information for the top 100 cities according to the Forbes ranking.(`https://www.kaggle.com/datasets/\bwandowando/forbes-top-100-cities-weather-data2020-ytd`) For the specific needs of this study, the dataset was filtered for Milan. The data is sourced from the Historical Weather API from Open-Meteo, and it is available at both hourly and daily granularity; for representational reasons which we will explore further in the Integration section, we chose the hourly data as a starting point. Key variables in the data have information on weather factors such as temperature, humidity, precipitation, wind, pressure, moisture, and irradiance. The dataset includes key variables:

- `city_name` the name of the city considered (in this case, Milan);
- `datetime` the date of the observation (yyyy-mm-dd) in GMT+00;
- `weather_code`: weather condition as a numeric code;
- `temperature_2m`: air temperature recorded at 2 meters above ground level (°C);
- `relative_humidity_2m`: humidity recorded at 2 meters above ground level (%);
- `dew_point_2m`: dew point temperature recorded at 2 meters above ground(°C);
- `apparent_temperature`: apparent (feels-like temperature) temperature (°C);
- `pressure_msl or surface_pressure`: atmospheric air pressure reduced to mean sea level or pressure at surface (hPa);
- `precipitation`:total precipitation(rain,snow) sum of the previous hours (mm);
- `rain`: liquid precipitations of the previous hour (mm);
- `snowfall`: snowfall amount of the previous hour (cm);
- `cloud_cover`: total cloud cover as an area fraction (%);
- `cloud_cover_low`: clouds and fog up to 2 km altitude (%);

- `cloud_cover_mid`:clouds from 2 to 6 km of altitude (%);
- `cloud_cover_high`: clouds from 6 km of altitude (%);
- `shortwave_radiation`: Shortwave solar radiation as average of the preceding hour (W/m^2);
- `direct_radiation`: Direct solar radiation as average of the previous hour (W/m^2);
- `diffuse_radiation`: as average of the previous hour (W/m^2);
- `global_tilted_irradiance`: Total radiation received on a tilted pane as average of the preceding hour(W/m^2);
- `wind_speed_10m`: wind speed at 10 meters above ground (km/h);
- `wind_direction_10m`: wind direction at 10 meters above ground (°);
- `wind_gusts_10m`: gusts at 10 meters above ground of the indicated hour;
- `et0_fao_evapotranspiration`: evapotranspiration (ET_0) of a well-watered grass field, expressed in millimeters (mm). This value represents the potential evaporation from the soil surface, considering the energy available and weather conditions;
- `snow_depth`: snow depth on the ground (m);
- `vapour_pressure_deficit`: (kPa);
- `soil_temperature_**_to_**cm`: average temperature of soil in different levels below ground (°C);
- `soil_moisture_**_to_**cm`: average soil water (m/m) ;

3.1.1 Reverse Engineering

To verify that the uploaded dataset was obtained through the default method available on Open-Meteo, which is "Best Match", we compared the Kaggle data with the results of an API call using "Best Match", and all of the columns except those for weather_code and direct_normal_irradiance_instant were identical to the ones in the Kaggle dataset. We repeated the experiment for the other modalities available through API call, but none came close to coinciding as Best Match, leading us to think that indeed the dataset was collected with the default method, and that something must have changed in the way Open-Meteo procured those values between the creation of the Kaggle dataset and now. The different columns don't anyway impact our analysis or process, as we will discard them from the inal dataset.

The comparison between "Best Match" and the Kaggle data can be viewed in the weatherMatchTest.ipynb file.

3.2 Data Provenance

As the data is coming from Open-Meteo, it was traced back to the combination of reanalysis models, by default the requested data is supplied using a 'Best Match' model, combining different reanalysis data (probably through the use of median ensembles, this info was not available) for the most accurate estimations. More precisely, the dataset is traced back to the combination of ERA5, considered one of the most complete available weather datasets, and ERA5-Land. They are both projects of ECMWF (European Centre for Medium-Range Weather Forecasts), which uses a weather prediction model that simulates the atmosphere's behavior using the laws of physics. As many weather prediction centers, it follows the data assimilation principle. In this method, a previous forecast is combined with real data observation to improve new forecasts. ERA5 reanalysis works similarly but at a lower pace to collect and integrate more observations. Moreover, instead of focusing on real time forecasting, it creates a historical weather data storage.

ERA5 also leverages itself with the ECMWF Integrated Forecasting System (IFS), a data assimilation system that also combines forecasts with observations, to keep estimates coherent. ERA5-Land is an extension of ERA5, which uses ERA5 atmospheric variables as input to simulate the evolution of land variables(atmospheric forcing), enhancing the data related to terrestrial processes like soil moisture.

Further links are provided in the references.

3.3 Data Cleaning

Similarly to the air quality part, we find no missing values in the period of interest (2020 to 2023), and no duplicate rows are found. Again, we do not look for outliers as they can be interesting for the analysis.

3.4 Data Processing

After acquiring the data and filtering it so that it only related to the city of Milan, we followed the same procedure as with the air quality data to turn the timestamp columns into local time(GMT+01 and DST On/Off). After this, we wanted to reduce the dimensionality of the dataset by filtering non-vital or replaceable columns. Our procedure lead us to remove:

- ’snowfall’ column: as it could be deduced from ’precipitation’ and ’rainfall’
- ’snow_depth’: As it’s interaction with air quality was deemed less relevant.
- ’cloud_cover_low’, ’cloud_cover_mid’, and ’cloud_cover_high’ : as ’cloud_cover’ represents total cover, while others only at certain altitude.
- soil moisture and temperature at more than 28cm depth: to focus on situation closer to surface
- instant radiation and irradiation fields : as their non-instant counterparts accounted for the average value for the preceding hour, so we considered them more informative when evaluating dimensionality reduction trade-off.

These processes can be followed in the WeatherData_filtercity&tolocaltime.ipynb and Weather_filtercols.ipynb files.

3.5 Data Quality

The dimensions of interest do not change from the air quality data section, therefore we wanted to verify syntactic accuracy, completeness(including object completeness) and consistency of time formats. Unfortunately, we couldn’t find reputable sources for historical maximums and minimums for temperatures in Milan, therefore the syntactic accuracy could not be verified to the full extent, but by evaluating the minimum and maximum values in the data we can have an idea of how accurate they can be:

- Temperature 2m above ground had a maximum value of 37.4, and a minimum value of -5.6
- Apparent temperature had a maximum of 40.2 and a minimum of -9.2

These values do not seem too extreme to believe, and apart from this inconvenience, we found no major completeness or consistency flaw in the data.

The process can be followed in the weatherDQcheck.ipynb file.

4 Data Integration

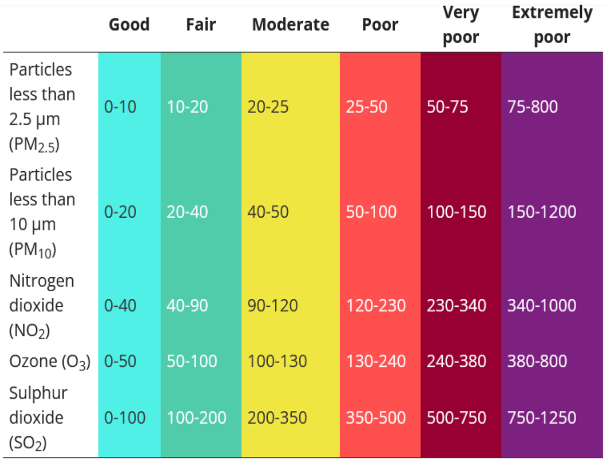
4.1 Aggregating to daily (Air Quality)

As we stated before, in order for our analysis to be able to represent the perception of daily life in Milan, and to reduce dimensionality of the dataset, we will be aggregating our hourly values to daily granularity. The approach changes with respect to data source, when thinking about the correct representation of hourly data in daily format, we came up with a series of possible aggregations, to apply to each single pollutant or to deduce by taking into account all five of them. The resulting daily data included the following variables:

- Average, minimum and maximum values for each single pollutant in a day
- Hours at which a single pollutant reached its maximum and minimum values
- The European Air Quality Index (EAQI) level for each day, considering all pollutants.
- The EAQI level of single pollutants for each day.

4.1.1 Computing the EAQI

The approach followed to compute the EAQI varied among pollutants, and varied from the usual way of computing it, as it usually describes air quality levels in a certain moment and not in a day. Normally, the EAQI is computed by assigning the 5 main pollutants a level according to pre-established thresholds(these vary, we went with those provided by both the European Environment Agency and Copernicus Atmosphere Monitoring Service), for all pollutants except pm10 and pm2.5, the value at that particular moment is used.



	Good	Fair	Moderate	Poor	Very poor	Extremely poor
Particles less than 2.5 µm ($PM_{2.5}$)	0-10	10-20	20-25	25-50	50-75	75-800
Particles less than 10 µm (PM_{10})	0-20	20-40	40-50	50-100	100-150	150-1200
Nitrogen dioxide (NO_2)	0-40	40-90	90-120	120-230	230-340	340-1000
Ozone (O_3)	0-50	50-100	100-130	130-240	240-380	380-800
Sulphur dioxide (SO_2)	0-100	100-200	200-350	350-500	500-750	750-1250

Above: European Environment Agency's table for computing the EAQI

For pm10 and pm2.5, the guidelines dictate that the average of the last 24 hours is to be used. After computing all single pollutant levels, the given EAQI is taken to be the highest level among those computed. When adapting this concept to daily granularity, we chose to use daily averages for pollutants except pm10 and pm2.5, for the latter we decided to compute the average of all 24-hour rolling averages in a day. The daily EAQI is then picked to be the highest level recorded among all single-pollutant levels; the single-pollutant levels are also stored for the day.

This procedure can be followed in the AQdataToDaily.ipynb file.

4.2 Aggregating to daily (Weather Data)

To roll our hourly weather data up to daily granularity, we looked at each column and devised one or more aggregation functions, keeping in mind the nature of the fields and trying to avoid high dimensionality. For the majority of the columns, the average value was taken, for values like 'precipitation', 'rainfall' and evapotranspiration, the daily sum was taken. For wind speed, wind gusts, and weather code, the most severe value was chosen (max), while for temperature 2 meters above the ground we chose to include minimum, maximum and average values altogether.

The procedure can be followed in the weatherData_toDaily.ipynb file.

4.3 Integration

In the integration phase, the air quality data and the weather data were integrated into a single and unified dataset. In this specific case, apart from matching the two datasets to the same level of granularity, no additional transformation was necessary because both datasets were already in a compatible format. Indeed, we didn't need to find correspondences or resolve conflicts as the phenomenons described didn't share any characteristics other than a timestamp. Therefore, the two datasets were merged on the 'date' column to serve our goal of building a comprehensive daily dataset on which further analysis can be carried out.

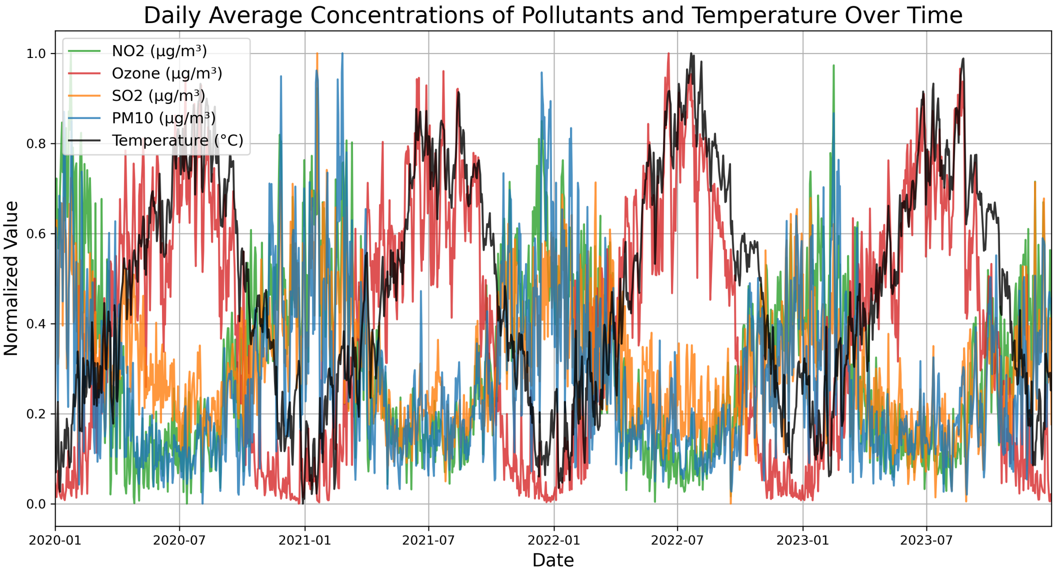
The final merged dataset was composed of 1461 records and 63 total columns, of which one is the shared column `eaqi`, 33 are from the Air quality dataset, and 29 are from the Weather dataset.

4.4 Column Naming Convention

As a final step, we decided to clean up the final dataset, structuring all column names in a standardized format so that they are coherent. The format is as follows: *aggfunction*_*attribute* "Max","min", "sum" and "avg" are the considered functions. Example: the "temperature_2m_mean" column was obtained by aggregating hourly "temperature_2m" above ground through computing the mean(or average), therefore the new name in the final dataset will be "avg_temperature_2m", the same goes for columns where instead of "mean" the name contains "average". In addition, column names don't include 'daily' anymore.

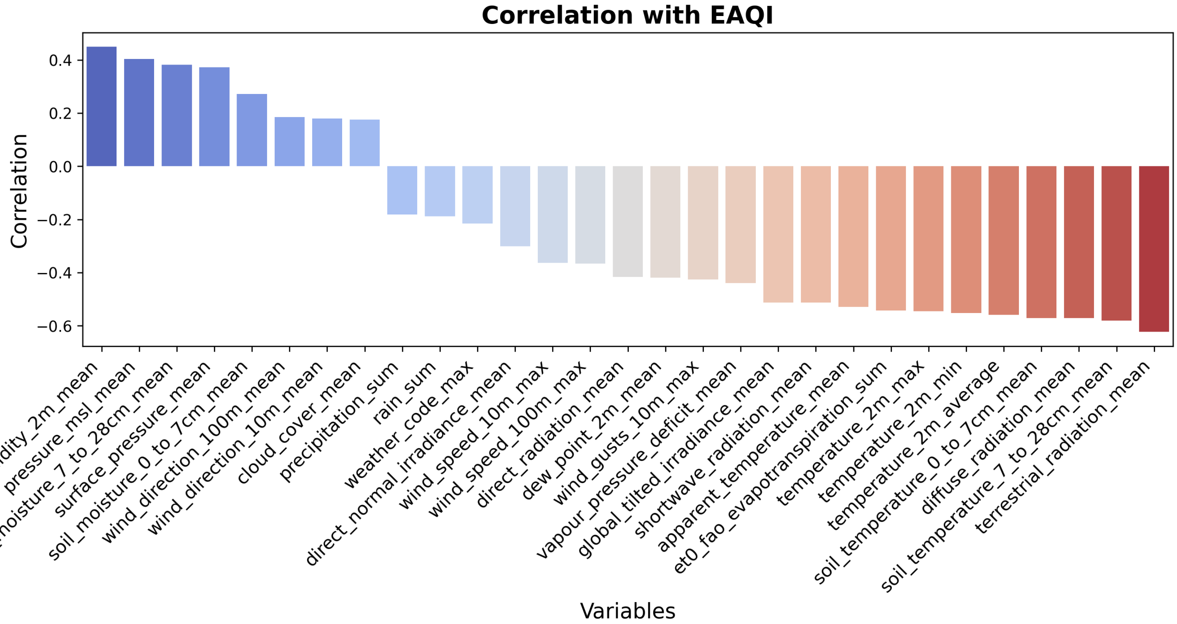
5 Conclusions

In conclusion, the feat of acquiring reliable atmospheric seemed a difficult one in the beginning, indeed we can never be sure about the absolute accuracy of the variables collected, but reanalysis models surely offer the highest level of certainty available to represent our phenomenons of interest, and the effort from the ECMWF, EEA and CAMS proves useful for many, in combination with the work of Open-Meteo and their free access API they constitute valuable data sources for environmental data analysis and collection. We believe that through analysis of recent interactions between environmental factors and air quality levels valuable insights can be obtained, and that through further modeling even machine learning approaches could benefit from the final dataset. Just for a glimpse into the possibilities of this dataset, we decided to plot the standardized pollutant values along with the temperature throughout time, and discovered a pretty fascinating seasonality affecting dynamics of environmental factors:



The values were standardized due to different scales, and indeed it looks like all pollutants move seasonally inversely to temperatures, while ozone goes opposite to pollutants and moves along with temperature.

On top of the time series plot, a correlation plot(using old column names) revealed that high temperatures and radiation bring better air quality while colder temperatures do the opposite, and that moisture, higher pressure, and humidity boost bad air.



6 Future Modifications and Improvements

In the future, we could keep adding to this initial dataset as time goes by and validated reanalyses become available, we could model it differently with some other purposes in mind, and an interesting step could also be to explore the data further to find out if it contains underlying biases.

References

Zippenfenig, P. (2023). Open-Meteo.com Weather API [Computer software].

Zenodo. https://doi.org/10.5281/ZENODO.7970649

Hersbach, H., Bell, B., Berrisford, P., Biavati, G., Horányi, A., Muñoz Sabater, J., Nicolas, J., Peubey, C., Radu, R., Rozum, I., Schepers, D., Simmons, A., Soci, C., Dee, D., Thépaut, J-N. (2023). ERA5 hourly data on single levels from 1940 to present [Data set]. ECMWF. https://doi.org/10.24381/cds.adbb2d47

Muñoz Sabater, J. (2019). ERA5-Land hourly data from 2001 to present [Data set]. ECMWF. https://doi.org/10.24381/CDS.E2161BAC

Schimanke S., Ridal M., Le Moigne P., Berggren L., Undén P., Randriamampianina R., Andrea U., Bazile E., Bertelsen A., Brousseau P., Dahlgren P., Edvinsson L., El Said A., Glinton M., Hopsch S., Isaksson L., Mladek R., Olsson E., Verrelle A., Wang Z.Q. (2021). CERRA sub-daily regional reanalysis data for Europe on single levels from 1984 to present [Data set]. ECMWF. https://doi.org/10.24381/CDS.622A565A

Air Quality API Data: https://open-meteo.com/en/docs/air-quality-api

AQ Reanalysis model: https://ads.atmosphere.copernicus.eu/datasets/cams-europe-air-quality-reanalyses?tab=overview

Weather dataset on kaggle: https://www.kaggle.com/datasets/ bwandowando/forbes-top-100-cities-weather- data2020-ytd

Weather data original API: https://open-meteo.com/en/docs/historical-weather-api

ERA5: https://cds.climate.copernicus.eu/datasets/reanalysis-era5-single-levels?tab=overview

ECMWF IFS: https://www.ecmwf.int/en/forecasts/documentation-and-support/changes-ecmwf-model

ERA5 and ECMWF IFS:

https://openmeteo.substack.com/p/processing-90-tb-historical-weather?triedRedirect=true

CAMS Europe explanation of EAQI index: https://ecmwf-projects.github.io/copernicus-training-cams/proc-aq-index.html

European Environment Agency EAQI table: https://airindex.eea.europa.eu/AQI/index.html

METEO FRANCE, Institut national de l'environnement industriel et des risques (Ineris), Aarhus University, Norwegian Meteorological Institute (MET Norway), Jülich Institut für Energie- und Klimaforschung (IEK), Institute of Environmental Protection – National Research Institute (IEP-NRI), Koninklijk Nederlands Meteorologisch Instituut (KNMI), Nederlandse Organisatie voor toegepast-natuurwetenschappelijk onderzoek (TNO), Swedish Meteorological and Hydrological Institute (SMHI) and Finnish Meteorological Institute (FMI) (2020): CAMS European air quality forecasts, ENSEMBLE data. Copernicus Atmosphere Monitoring Service (CAMS) Atmosphere Data Store (ADS). (Accessed on ¡04-FEB-2025¿),

https://ads.atmosphere.copernicus.eu/datasets/cams-europe-air-quality-forecasts?tab=overview

METEO FRANCE, Institut national de l'environnement industriel et des risques (Ineris), Aarhus University, Norwegian Meteorological Institute (MET Norway), Jülich Institut für Energie- und Klimaforschung (IEK), Institute of Environmental Protection – National Research Institute (IEP-NRI), Koninklijk Nederlands Meteorologisch Instituut (KNMI), Nederlandse Organisatie voor toegepast-natuurwetenschappelijk onderzoek (TNO), Swedish Meteorological and Hydrological Institute (SMHI), Finnish Meteorological Institute (FMI), Italian National Agency for New Technologies, Energy and Sustainable Economic Development (ENEA) and Barcelona Supercomputing Center (BSC) (2022): CAMS European air quality forecasts, ENSEMBLE data. Copernicus Atmosphere Monitoring Service (CAMS) Atmosphere Data Store (ADS). (Accessed on ¡04-FEB-2025¿), https://ads.atmosphere.copernicus.eu/datasets/cams-europe-air-quality-forecasts?tab=overview