­

**Table of Content**

[List of Key Words 3](#_Toc137063058)

[Abstract 4](#_Toc137063059)

[Introduction 5](#_Toc137063060)

[Relevance of Mapping Air Pollution 5](#_Toc137063061)

[Subject Area 5](#_Toc137063062)

[Aim of the Project 5](#_Toc137063063)

[Data 5](#_Toc137063064)

[Method 5](#_Toc137063065)

[Results 5](#_Toc137063066)

[1. Current State of the Problem and Problem Statement 6](#_Toc137063067)

[1.1 Mapping Air Pollution 6](#_Toc137063068)

[1.2 Earth Remote Sensing Data (ERS) 6](#_Toc137063069)

[1.3 Modern Approaches to Air Pollution Mapping 6](#_Toc137063070)

[1.4 Section Conclusions and Problem Statement 8](#_Toc137063071)

[2. External Data and Technology 10](#_Toc137063072)

[2.1 Google Earth Engine Landsat 8 Data and ROI 10](#_Toc137063073)

[2.2 EPA Data 12](#_Toc137063074)

[3. Air pollution mapping approach 13](#_Toc137063075)

[3.1 Feature of the proposed approach 13](#_Toc137063076)

[3.2 General implementation scheme (diagram 7) 13](#_Toc137063077)

[3.3 Formation of input data 14](#_Toc137063078)

[3.4 Structure of Input Data 16](#_Toc137063079)

[3.5 Random Forest (Regression) Model 16](#_Toc137063080)

[3.6 XGBoost Model 17](#_Toc137063081)

[3.7 Support Vector Regression Model 17](#_Toc137063082)

[3.8 Linear Regression Model 18](#_Toc137063083)

[3.9 Hyperparameters 18](#_Toc137063084)

[4. Experimental Evaluation of Air Pollution Mapping 19](#_Toc137063085)

[4.1 Metrics for Experimental Evaluation 19](#_Toc137063086)

[4.2 Selected areas of research 19](#_Toc137063087)

[4.3 Numerical results of experimental evaluation 20](#_Toc137063088)

[4.4 Confirmed hypotheses 20](#_Toc137063089)

[4.5 Conclusions on the section 20](#_Toc137063090)

[5. Challenges of the implementation 22](#_Toc137063091)

[Conclusion 23](#_Toc137063092)

[List of Sources 24](#_Toc137063093)

# List of Key Words

**PM10** - Particulate matter with a diameter of 10 micrometers or less, small enough to be inhaled into the respiratory tract.

**Landsat 8 bands** - The Landsat 8 satellite captures images of the Earth across multiple spectral bands, each sensitive to different features or properties. For example, certain bands are useful for observing vegetation or water.

**MAIA** – Multi-Angle Imager for Aerosols. A NASA satellite instrument designed to measure airborne particles known as aerosols.

**Google Earth Engine** - A platform that provides access to a massive amount of high-quality satellite imagery and other geospatial data, allowing for both analysis and visualization.

**EPA** – Environmental Protection Agency. The U.S. federal agency responsible for creating standards and laws promoting the health of individuals and the environment.

**Remote sensing** - The process of detecting and monitoring physical characteristics of an area by measuring its reflected and emitted radiation from a distance (typically from satellite or aircraft).

**STILT** – Stochastic Time-Inverted Lagrangian Transport Model. A model used to determine how particles move in the atmosphere. It uses a variety of weather forecasting models to show the movement and dispersion of particles.

**Cross-Validation** - Statistical method used to estimate the skill of machine learning models.

**ROI** – Region Of Interest. A term used in geospatial analysis referring to the specific geographic area for which the analysis is conducted.

**Longitude/Latitude** - part of the geographic coordinate system used to locate any place on Earth.

**API** - Application Programming Interface. A set of rules and protocols for building and interacting with software applications.

**Decision Trees** - Type of supervised learning algorithm used in machine learning and statistics to predict an outcome based on input variables. They are called decision trees because they start with a single box (or root), which then branches off into a number of solutions, just like a tree. Each internal node of the tree corresponds to an attribute, and each leaf node corresponds to a class label.

Kernel (in SVR model) - A function used to take data as input and transform it into a higher dimensional space where it becomes easier to classify the data.

**RBF** - Radial Basis Function. Popular kernel function commonly used in SVM classification. It can map an input space in infinite dimensional space making it a good choice for a wide variety of datasets.

**Hyperparameters** - parameters whose values are set before the learning process begins. These values define the structure of the model or how the model is trained.

# Abstract

In today's world, environmental challenges are becoming more urgent. Air pollution is a significant concern, just like water pollution and the growing greenhouse effect. When people breathe polluted air, they have a high risk developing serious health issues such as heart disease, lung diseases, and even lung cancer. Over time, air pollution can also harm our nervous system, brain, kidneys, liver, and other vital organs. Some researchers even suggest that air pollution could be linked to some cases of infant mortality.

As our society develops and our use of technology increases, so does air pollution. Harmful substances like smoke, dust, salts, acids, and metals are often released into the atmosphere from cars and factories. Once these pollutants are in the air, they can undergo chemical changes and create new types of pollution.

Despite efforts to monitor air pollution through ground stations, including those that collect data for the U.S. Environmental Protection Agency (EPA), their coverage is limited. The data from these stations helps identify areas where the government needs to act, but it can't cover all regions. This is why this project was started. By using satellite data, I have aimed to expand our understanding of air pollution and cover areas that ground stations can't reach. My goal was to improve our knowledge of air pollution and help guide actions to address it.

# Introduction

## Relevance of Mapping Air Pollution

Mapping air pollution holds paramount importance in our rapidly industrializing world. Air pollution, a pervasive environmental problem, has considerable health implications ranging from respiratory diseases to potential neurological harm. The ability to map air pollution geographically facilitates our understanding of pollution sources, their impacts, and mitigation strategies.

## Subject Area

This project resides at the intersection of environmental science, remote sensing, and machine learning. It is focused on correlating ground-based air pollution data with satellite imagery to create a comprehensive air pollution map.

## Aim of the Project

The primary objective of this project is to enhance our understanding of air pollution spread and its dynamics across the United States. By integrating ground-based monitoring with satellite data, I aspire to provide a more comprehensive, real-time mapping of particulate matter (PM10) levels.

## Data

The data sources for this project are twofold: ground-based air quality measurements from the EPA and satellite-based Landsat 8 bands (thermal and optical) from Google Earth Engine. These two data types are the backbone of the machine learning model I am developing.

## Method

The project employs machine learning algorithms to establish a correlation between the EPA's PM10 data and the satellite data. The model is trained and validated using these datasets, allowing it to predict PM10 levels in regions beyond the reach of current ground-based monitoring.

## Results

The expected outcome is a detailed, dynamic map of PM10 levels across the United States. This product will provide unprecedented spatial and temporal coverage of air pollution data, opening up new possibilities for environmental research, public health initiatives, and policy-making decisions.

# 1. Current State of the Problem and Problem Statement

## 1.1 Mapping Air Pollution

Air pollution mapping refers to the process of creating visual representations of pollutant concentrations over geographic areas. This technique is of crucial importance as it aids in identifying pollution sources, evaluating health risks, monitoring trends, and informing policy decisions for environmental conservation.

## 1.2 Earth Remote Sensing Data (ERS)

Remote sensing data from space-borne instruments, like Landsat 8, provide invaluable information about the Earth's surface and atmosphere. These satellites capture data in various formats, at different spatial resolutions, and across numerous spectral channels, enabling comprehensive analysis. For instance, the Landsat 8 satellite captures data across several spectral bands, including visible, near-infrared, and thermal, each offering distinct information about environmental conditions with an approximate resolution of 30 meters, different scaling factors. Screenshots or samples of this data vividly illustrate the breadth and depth of information that can be harnessed for air pollution mapping.

Visible/Near-Infrared bands: SR\_B1, SR\_B2, SR\_B3, SR\_B4, SR\_B5, SR\_B6, SR\_B7

Thermal Band: ST\_B10

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

*Diagram 0 – initial data from Landsat 8 bands*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Scaling Factors** | Min-value | Max-value | Scale | Offset | Wavelength |
| Visible/Near-Infrared bands | 1 | 65455 | 2.75e-05 | -0.2 | 0.435 - 2.294 μm |
| Thermal bands | 0 | 65535 | 0.00341802 | 149 | 10.60-11.19 μm |

## 1.3 Modern Approaches to Air Pollution Mapping

Modern approaches to air pollution mapping are becoming increasingly sophisticated and advanced, leveraging technology and science to provide more accurate and detailed data on air pollution levels and their impacts.

**Nasa**

One of the key modern approaches is the use of satellite data for air pollution mapping. NASA, in collaboration with epidemiologists, has launched its first health-focused mission to understand how air pollution affects health in cities around the world. The Multi-Angle Imager for Aerosols (MAIA) mission, launched in 2022, aims to understand what kind of air pollutant particles are most harmful to human health. MAIA's instrument measures the polarization of sunlight scattered by particles and images of the light at multiple angles in 14 electromagnetic wavelengths spanning from ultraviolet to short-wave infrared. These optical properties allow scientists to infer the composition of PM2.5, particulate matter smaller than 2.5 micrometers, which poses severe health risks. These particles primarily form combustion sources like fossil fuel use and wildfires, and are small enough to pass from the lungs into the bloodstream, where they can travel all over the body and cause health issues ranging from respiratory diseases to premature death from heart and lung conditions.

This data is then used to map PM2.5 concentrations and composition at the neighborhood level (a resolution of 1 square kilometer) in its 12 primary target cities in North America, Europe, Africa, and Asia. This mapping integrates atmospheric chemistry, spatial, and statistical models with the MAIA data and information from ground-based monitors to provide a comprehensive view of ground-level concentrations of PM and components of PM2.5 in the target areas.

Epidemiologists link this PM data with population health records in each target area to learn which particles are most toxic. They study short- and long-term effects of PM exposure and adverse birth outcomes and cardiovascular disease, common issues tightly tied to air pollution.

**U'S AIR QUALITY NETWORK**

Another approach of The University of Utah, in partnership with the Environmental Defense Fund and the CREATE Lab at Carnegie Mellon University, developed a web-based tool called Air Tracker. This tool was launched on June 13, 2022, and allows users to plot the likely path of air pollution. It uses real-time, trusted scientific models along with air pollution and weather data to provide information about air pollution concentrations and potential sources.

Air Tracker works by allowing users to click anywhere on maps of Houston, Salt Lake City, and Pittsburgh to create a "source area". This source area shows the most likely origin of the air they're breathing at any given time. Users can also click on locations of individual air quality sensors to show real-time and historical PM 2.5 pollution readings, wind speed, and direction.

Изображение выглядит как текст, карта, снимок экрана, диаграмма

Автоматически созданное описаниеOne of the key features of Air Tracker is its reliance on the Stochastic Time-Inverted Lagrangian Transport model (STILT). This model, developed by the research group of Professor John Lin at the University of Utah, incorporates a variety of weather forecasting models to show how particles move through the atmosphere. This allows Air Tracker to map the probability of pollution's path. The tool is designed to identify pollution sources at the city block level, which provides more granularity than common source identification models. A screenshot from the Air Tracker shows the upwind sources of air at the University of Utah campus on June 6, 2022, including air quality measurements from the PurpleAir network and UTA’s TRAX trains (as shown on diagram 1).

*Diagram 1 – screenshot from the Air Tracker*

Both approaches are scalable and particularly valuable in parts of the world where ground-level monitoring is sparse or non-existent, such as in many low- or middle-income countries.

## 1.4 Section Conclusions and Problem Statement

The main aim of the project is to obtain several maps of air pollution (PM10), by leveraging EPA and Landsat 8 datasets and applying best performed model. Several models will be applied in order to broaden the vision of data variety, and understand the best behavior of it. According to the result, places and reasons of the PM10 high concentration will be found. This project proposes to develop a way of clear representation of PM10 pollution, which would be useful for struggling environmental issues. An abstract plan of the project is represented:

• Study EPA data

• Get Google Earth Engine API

• Set the virtual environment for working with Google Earth Engine (anaconda & jupyter notebook)

• install all the required libraries for the project

• Authenticate and Initialize Google Earth Engine in virtual environment

• Create a map of the Landsat 8 data with applied scaling factors, time range

• Save the required bands, longitude, latitude, time data from the map to csv

• Save the EPA data of the chosen time range (in years)

• Correlate the data from both sources in one table (by longitude/latitude, time, date)

• Prepare the data (normalize, reduce bad outliers)

• Start training several models with defined hyperparameters

• Evaluate each model’s performance (by MSE, MAE, R**²**) and choose the best one

• Apply the best model to the initial data of Landsat 8 to start mapping air pollution of PM10

# 2. External Data and Technology

## 2.1 Google Earth Engine Landsat 8 Data and ROI

Изображение выглядит как Мир, карта

Автоматически созданное описаниеChosen Region Of Interest is USA. The map (diagram 2) shows the Landsat data applied only to USA and the shape covering it, which will help to download data by chunks in future. ROI was obtained through one of Google Earth Engine Feature Collections, which represents a set of features for the map with appropriate numerical values. An implementation of getting Landsat 8 data from Google Earth Engine for the set period of time with applied scaling factors for each band was developed. The data of the satellites is stored in Image Collections, which represents a set of images with numerical values. To obtain a well-build dataset for future implementation, I needed to add longitude, latitude, time of each pixel to the image, which is done in applyScaleFactors function. In order to reduce rudimental bands like “Pixel Distance to Cloud”, “Emissivity standard deviation”, I have mentioned only needed bands for my future model in “bands” variable (diagram 3).

*Diagram 2 – Google Earth Engine map (ROI)*

To collect the Landsat 8 data to csv, as EPA, I needed to download the each-pixel-data from Google Earth Engine image representation of the map to csv. The function *extract\_values\_to\_points* was used (from ***geemap*** library Python). However, this function has a limit up to 80mib/downloaded file. Consequently, I divided the ROI to chunks, downloaded each chunk separately (diagram 4).

Изображение выглядит как текст, снимок экрана, программное обеспечение, компьютер

Автоматически созданное описание

*Diagram 3 – ROI representation*

Изображение выглядит как текст, снимок экрана, компьютер, дисплей

Автоматически созданное описание

*Diagram 4 – download data (using chunks)*

## 2.2 EPA Data

EPA data is saved in the csv file initially. The approximate size of EPA dataset is 3,000,000 x 24. There are lots of data in this table like: “State Code”, “Country Code”, “State Name” … But I only needed several “Date Local”, “Time Local”, “Longitude”, “Latitude”, and “Sample Measurement” – which represents the amount of PM10 micrograms per cubic meter.

In the result I have obtained two data tables:

**Landsat8 (diagram 5), EPA (diagram 6)**

**Изображение выглядит как текст, снимок экрана, Шрифт, информация

Автоматически созданное описание**

*Diagram 5 – Landsat data*

**Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание**

*Diagram 6 – EPA data*

# 3. Air pollution mapping approach

## 3.1 Feature of the proposed approach

The initially proposed approach of mapping air pollution is in applying several models on the collected, correlated data of EPA and Landsat 8, which is normalized and prepared for training the models. This approach will allow to test several models in order to find the most suitable (best performing) and use it in future. Moreover, The future evaluation of the models performance, will also include the understanding of feature importance to understand which band has cause the most effect on the model.

In conclusion, the model will be downloaded to use in future, so that you will only need to get Landsat 8 data of the time interval you need and get the predicted values of PM10 for the specific region mentioned.

## 3.2 General implementation scheme (diagram 7)

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание

*Diagram 7 – block-scheme of implementation of the project*

## 3.3 Formation of input data

• During the input data of Landsat 8 I needed to apply scale factors like  
*Visible/Near-Infrared/Thermal bands = bands \* (its) scale + (its) offset*

*•* As data types differs from table to table, it was needed to make the column with the same data types. The ***datetime*** library was applied. After that, column “datetime” was obtained in both tables = *%Y-%m-%d %H:00*

Изображение выглядит как текст, снимок экрана, компьютер, программное обеспечение

Автоматически созданное описание• During correlation of EPA and Landsat datasets (diagram 8) on longitude, latitude and datetime of both tables (diagram 9 and 10), I applied ***geopandas*** library from python to speed the algorithm up. At first I have created GeoDataFrames for Landsat and EPA, which adds an additional column containing geometric object (point) in it, derived from original longitude and latitude. Special index was indicated, in order to minimize the number of bounding box comparisons. To correlate both datasets the “possible\_matches\_index” list was created to quickly find all the intersections of coordinates in both tables with approximation of “buffer\_dist” = 0.1. Furthermore, on each iteration of loop, there is a checking function of two datetimes parameters to be withing 1 hour of tolerance parameter. In conclusion, the correlated dataset was created (diagram 11).

*Diagram 8 – EPA and Landsat data correlation*

*Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание* Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

*Diagram 9 – Landsat data before merge Diagram 10 – EPA data before merge*

*(without several bands)*

*Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание*

*Diagram 11 – EPA and Landsat 8 Data after correlation*

• Preprocessing data before model application was relatively obvious:

I have used a ***StandardScaler*** library from ***sklearn.preprocessing***, which has standardized the features from my correlated table:

- the standardized value

– the original value from the table

– the mean of the feature column

– the standard deviation of the feature column

Then I have used outlier detection and removal:

***IQR = Q3 - Q1:***

## 3.4 Structure of Input Data

The table obtained after normalization of the correlated data (of EPA and Landsat 8) represents the structure of input data. First eight rows represent Landsat 8 bands values, which will be used to train the model to predict the last rows values of EPA PM10. The input data is then divided into two tables by the rule mentioned above, and furthermore, shuffled and divided into train and test samples in 8to2 proportion. After all of this I got four different tables to train and test various models.

## 3.5 Random Forest (Regression) Model

The Random Forest model is a machine learning method, which is based on Decision Trees. It operates by constructing multiple decision trees during training and outputting the mean prediction for regression of the individual trees. It can handle high-dimensional spaces as well as large numbers of training examples, which is beneficial when working with large satellite and EPA datasets. Major hyperparameters included in Random Forest model are:

1. **Bootstrap** – method of sampling data points with replacement. Given a dataset of size *N*, we sample *N* instances uniformly with replacement, meaning some instances may appear multiple times in a sample and others may not appear at all.
2. **N estimators** – number of trees in the forest.
3. **Max depth** – maximum number of levels in each decision tree.

**Explanation of Random Forest (Regression) Model:**

Let be the set of instances, and be the set of their corresponding labels. Let be the number of trees to build. Each decision tree , for is built based on a bootstrap sample drawn from the original data of size . The best split at each node of the decision tree is determined by a subset of features selected randomly from all features.

The final RF model makes a prediction for a new input by averaging the predictions of all individual trees.

## 3.6 XGBoost Model

XGBoost model - powerful, and efficient implementation of the gradient boosting framework. It has a built-in capability for regularization which prevents the model from overfitting. XGBoost implements parallel processing. These features make XGBoost a better model when compared to other gradient boosting techniques. Major hyperparameters included in XGBoost model:

1. **N estimators** – number of gradient boosted trees to be used in the model.
2. **Max depth** – maximum tree depth allowed by the model.
3. **Learning rate** – parameter that scales the contribution of each tree.

**Explanation of XGBoost Model:**

Given a dataset with examples and features, the model predicts an output for a data point by summing the predictions over weak learners (decision trees).

## 3.7 Support Vector Regression Model

The Support Vector Regression model – is a variant of Support Vector Machine (SVM) that supports regression tasks. It works by fitting the best line within a defined margin around the line of best fit, or the 'hyperplane', that can capture the most data points. Major hyperparameters included in SVR model are:

1. **C** – the penalty parameter of the error term.
2. **Epsilon** - specifies the epsilon-tube within which no penalty is associated with the training loss function on the epsilon-insensitive loss function.
3. **Kernel** - Specifies the kernel type to be used in the algorithm. It must be one of 'linear', 'poly', 'rbf', 'sigmoid', 'precomputed' or a callable.

**Explanation of SVR Model:**

The main idea of SVR is to find a function that has at most deviation from the obtained targets for all the training data, and at the same time, is as flat as possible. In other words, we're trying to minimize , under the condition .

The rbf – Radian Basis Function or Gaussian Kernel was used. Due to ability to model complex, non-linear relationships.

• – a parameter that sets the 'spread' of the Gaussian.

## 3.8 Linear Regression Model

The Linear Regression model is a simple and widely used statistical method for predicting a dependent variable based on one or more independent variables. It can be represented as:

• – independent variable

• – y-intercept

• to– coefficients of the independent variables

• to– independent variables (features)

• – residual error

## 3.9 Hyperparameters

To find profit-maximizing hyperparameters for several models with a relatively big input data, I have used *GridSearchCV* python library. It is a utility that automatically performs cross-validation over a grid of hyperparameters specified by the user, and then finds the particular setting of hyperparameters that gives the best performance as measured by a user-specified scoring method.

Cross-validation is a resampling procedure used to evaluate machine learning models on a limited data sample. The procedure has a single parameter , the number of groups that a given data sample is to be split into. Mathematically, the average score from k-fold cross-validation can be represented as:

• – number of folds

• – Mean Square Error of the *ith* test fold.

# 4. Experimental Evaluation of Air Pollution Mapping

## 4.1 Metrics for Experimental Evaluation

Evaluation of air pollution mapping model is a crucial part of my project. The need to obtain strong correlation dependence between actual and predicted values is required to justify the benefits of this project. Thus, I have evaluated performance of each model separately by (MSE, MAE, R2) for test sample and for train sample.

• Mean Absolute Error (MAE) - measures the average over the test sample of the absolute differences between prediction and actual observation where all individual differences have equal weight.

• Mean Square Error (MSE) - measures the average of the squares of the errors that is, the average squared difference between the estimated values and the actual value.

• Coefficient of Determination (R2) - measures the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a regression model.

## 4.2 Selected areas of research

Изображение выглядит как текст, карта, снимок экрана, диаграмма

Автоматически созданное описаниеSelected area of research – region of interest (ROI) is USA in my case. That is because of general data presented by EPA mainly contains United States. The map (diagram 12) shows the rectangle, which rounds the whole USA.

*Diagram 12 – ROI (United States)*

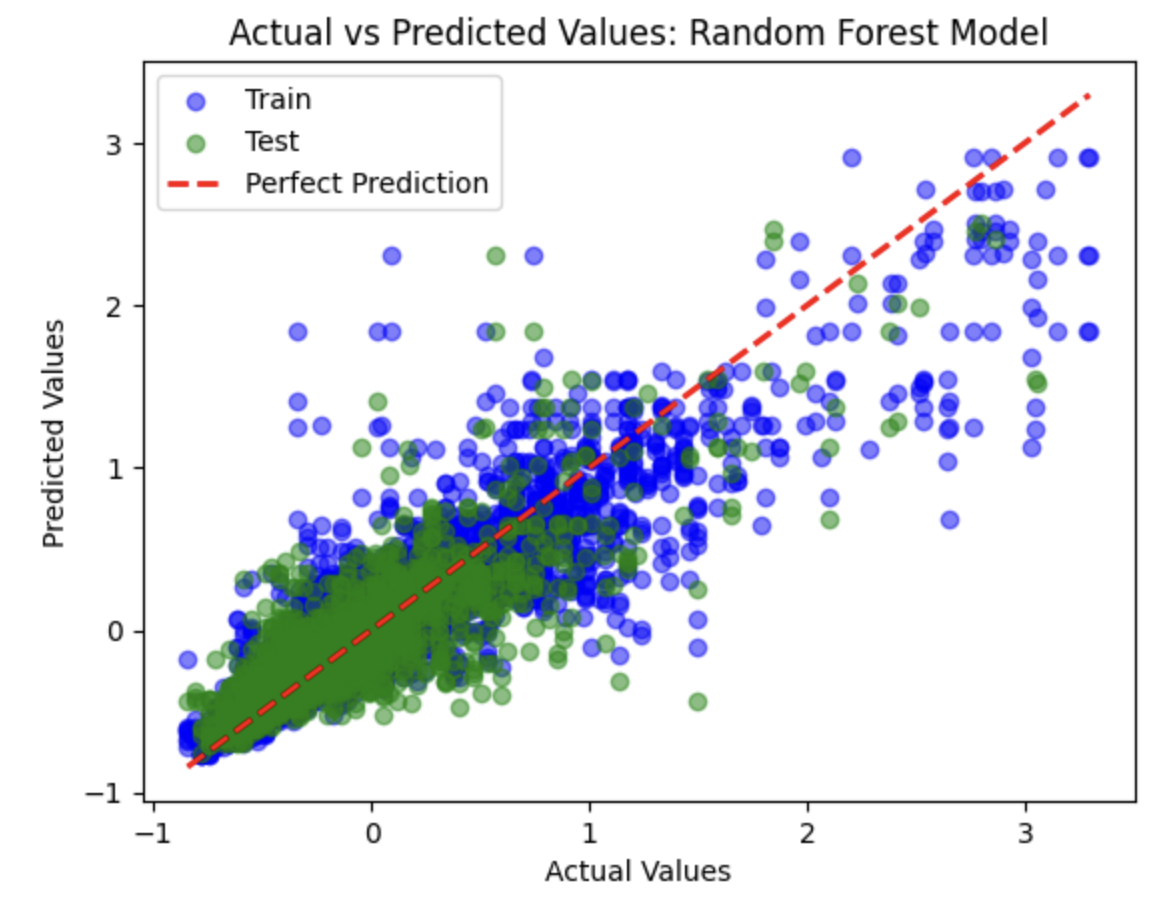
## 4.3 Numerical results of experimental evaluation

After evaluation of each model’s performance, I’ve got these results (diagram 13). According to them, Random Forest Regression model, has shown not only the best performance, but the strong dependence between predicted and actual values, as R2 (0.7 ; 0.84), for test and training respectively. The spread of values (predicted, actual) for best model (Random Forest) is represented on the scatter plot (diagram 14). Moreover, the most important feature was obtained through the implementation of bar chart of feature importance for Random Forest model – thermal band ST\_B10 (diagram 15).

Изображение выглядит как текст, Шрифт, снимок экрана, белый

Автоматически созданное описание

*Diagram 13 – models’ performance evaluation*

 Изображение выглядит как текст, снимок экрана, Шрифт, График

Автоматически созданное описание

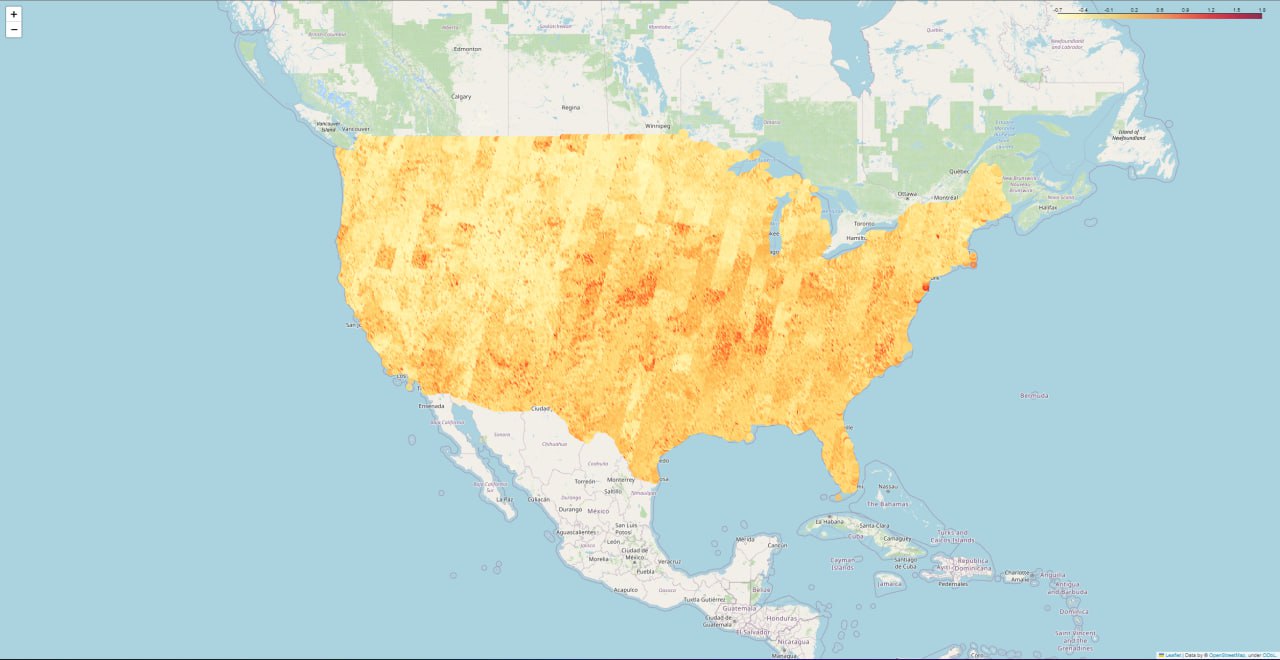
*Diagram 14 – scatter plot for RF Model Diagram 15 – Feature importance bar chart RF model*

## 4.4 Confirmed hypotheses

The main hypothesis to confirm was the relative effect of thermal band to the prediction of PM10. According to the similar projects’ reports, the thermal band was not used at all. In my case, the relative importance of thermal band “ST\_B10” is 27.4%, which is more than the sum of 2nd and 3rd bands.

## 4.5 Conclusions on the section

As the output of my project, I’ve got the overall map of USA with predicted values of PM10 corresponding to each coordinate presented in Landsat 8 data (diagram 16).



*Diagram 16 – PM10 USA map of 2021*

To justify the visualized data, I have used OSMnx library, which allows to get geospatial data from OpenStreetMap to obtain various “industrial” factories around the ROI. The following diagrams (diagram 17, diagram 18) shows two cities of US – Washington (the most air polluted city in US), Fresno (average city with high air quality).

Изображение выглядит как карта, текст, атлас

Автоматически созданное описание Изображение выглядит как карта, текст, атлас

Автоматически созданное описание

*Diagram 17 – PM10, factories location (Fresno) Diagram 18 - PM10, factories location (Washington)*

According to the data obtained, it can be concluded, that Washington has higher PM10 values, than Fresno. Seven visible plants in Washington can cause such high PM10 measurement, whereas one plant in Fresno does not affect air quality that much, thus PM10 measurement is low.

# 5. Challenges of the implementation

During the realization of the project, I have faced several problems from vast dataset preprocessing time to high resolution of air pollution mapping. Firstly, the problem of downloading Landsat data appeared. Due to using Google Earth Engine datasets, I was downloading them using its function - *extract\_values\_to\_points*, which had a limitation of 80mib per download dataset. Consequently, I had to divide the “image-type” dataset into chunks. Furthermore, the time-complexity was relatively high, as I was downloading the whole USA dataset, with each iteration of chunk download it took approximately 2 min, resulting in over 4hours of downloading one dataset containing ~150chunks. After that, the correlation problems: due to the big sizes of both datasets, comparison of features to make the united dataset was too slow. The size of EPA dataset >3mil rows / 24 col and Landsat dataset > 80,000 rows / 12 col. Leading to over 12hours of downloading time. Thus, I have found *geopandas* python library, which helped me to lower that time to 6hours per one correlated dataset. Final problem for preprocessing was in obtaining relevant hyperparameters for the models. It was also time-consuming with around 3hours per each model.

Talking about mapping challenges: the most difficult was analyzing the obtained maps, due to its high resolution. Over 80.000 points on an interactive map were placed in order to analyze them in future. Zoomed in map on (diagram 19), zoomed out map on (diagram 20).

Изображение выглядит как текст, карта, линия

Автоматически созданное описание

*Diagram 19 – Zoomed in map with point values of PM10*

Изображение выглядит как карта, текст, атлас

Автоматически созданное описание

*Diagram 20 – Zoomed out map with point values of PM10*

# Conclusion

Air pollution mapping plays a critical role in environmental planning and policy making, providing essential insights into spatial and temporal patterns of air quality. It is instrumental in informing public health strategies, mitigating environmental degradation, and fostering sustainable development.

In this project, I’ve leveraged satellite and EPA data from the United States to create comprehensive maps of air pollution. A combination of various data sources allowed for a comprehensive and nuanced understanding of pollution distribution. Data preprocessing was followed by application of four distinct machine learning models, Random Forest Regression, XGBoost, Support Vector Regression, and Linear Regression to correlate and predict pollution levels.

The machine learning methods employed in this study represented a broad spectrum of the field, ranging from decision-tree based ensemble methods (Random Forest and XGBoost), a non-parametric, kernel-based method (Support Vector Regression), and a statistical learning method (Linear Regression). Hyperparameters for each model were meticulously chosen using GridSearchCV, ensuring model optimization and enhancing the predictability of each model. A cross-validation approach was implemented to ascertain the stability and reliability of the models.

The results suggested that among all the models implemented, the Random Forest Regression model displayed superior performance in terms of accuracy, as determined by metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and Coefficient of Determination (R^2). It demonstrated an ability to capture complex non-linear relationships in the data, making it a robust and reliable model for air pollution mapping.

As for further work, the scope for refinement and optimization remains vast. New and emerging machine learning techniques could be explored for potential improvements in model performance. Increasing the temporal resolution of the dataset may also yield better results. Incorporating additional variables such as meteorological data could contribute to a more holistic and accurate model of air pollution distribution. The goal is to continually improve the model's accuracy, thereby refining our understanding of air pollution patterns and informing strategies for a cleaner and healthier environment.

# List of Sources

1. Google Earth Engine (data, API)

• <https://developers.google.com/earth-engine/datasets/catalog/LANDSAT_LC08_C02_T1_L2>

• <https://developers.google.com/earth-engine/tutorials/community/intro-to-python-api>

1. EPA (data, PM10 trends)

• <https://aqs.epa.gov/aqsweb/airdata/download_files.html>

• <https://www.epa.gov/air-trends/particulate-matter-pm10-trends>

1. WHO (Global air quality)

• <https://www.who.int/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health>

1. EEA (Air pollution)

• <https://www.eea.europa.eu/en/topics/in-depth/air-pollution>

1. Nasa (Modern Approaches to Air Pollution Mapping)

• <https://essp.nasa.gov/2021/04/20/using-satellite-data-to-map-air-pollution-and-improve-health/>

1. Utah university (Modern Approaches to Air Pollution Mapping)

• <https://www.price.utah.edu/2023/04/13/us-air-quality-network-receives-award>

• <https://attheu.utah.edu/facultystaff/new-tool-shows-air-pollutions-path/>

1. Regression

• <https://en.wikipedia.org/wiki/Regression_analysis>

1. Models

• <https://en.wikipedia.org/wiki/Random_forest>

• <https://en.wikipedia.org/wiki/XGBoost>

• <https://en.wikipedia.org/wiki/Support_vector_machine>

• <https://en.wikipedia.org/wiki/Linear_regression>

1. Python Regression (Scikit-learn – models’ implementation)

• <https://scikit-learn.org/stable/supervised_learning.html#supervised-learning>

1. Python (geopandas – geo-data tables)

• <https://geopandas.org/en/stable/docs/reference.html>

1. Python (folium – mapping)

• <https://python-visualization.github.io/folium/quickstart.html#GeoJSON/TopoJSON-Overlays>

1. Python (matplotlib – graphs, plots)

• <https://matplotlib.org/stable/users/index>

1. Python (geemap – Google Earth Engine)

• <https://geemap.org>

1. World’s Air Pollution

• <https://waqi.info/#/c/38.817/-99.996/4.2z>

1. Similar project

• <https://vverde.github.io/blob/interactivechoropleth.html>

• <https://www.datacamp.com/tutorial/geospatial-data-python>

1. Satellite data

• <https://www.iceye.com/satellite-data>

1. Earth remote sensing data

• <https://www.youtube.com/watch?v=48al25GCTIE>

1. Remote sensing Nasa

• <https://www.earthdata.nasa.gov/learn/backgrounders/remote-sensing>

1. Remote sensing Dragonfly Aerospace

• <https://dragonflyaerospace.com/how-remote-sensing-satellites-works/>

1. Earth observation Wikipedia

• <https://en.wikipedia.org/wiki/Earth_observation_satellite>

1. Satellite temperature measurement Wikipedia

• <https://en.wikipedia.org/wiki/Satellite_temperature_measurements>

1. Landsat 8 USGS

• <https://www.usgs.gov/landsat-missions/landsat-8>