$PM_{2.5}$ Prediction for Cities in Sub-Saharan Africa

EAEE 4000 - Machine Learning for Environmental Engineering and Sciences

Final Project

December 23, 2021

Benjamin Yang (*by2321*)

https://github.com/benjaminyang93/ml-project-africa

Introduction

Ambient (outdoor) air pollution is on the rise in Africa due to rapid population growth and industrialization. Poor air quality caused 1.1 million premature deaths across Africa in 2019 alone (Fisher et al., 2021). According to the World Health Organization (WHO, 2021), the annual and 24-hour mean values for particulate matter with an aerodynamic diameter of 2.5 μm or less ($PM_{2.5}$)—an important proxy for air pollution—should not exceed 5 μg m^{-3} and 15 μg m^{-3}, respectively. However, these guidelines are often severely violated in major African cities. For example, one study showed that Addis Ababa, Ethiopia had a daily $PM_{2.5}$ concentration of 53.8 (±25.0) μg m^{-3} from 2015-2016 (Tefera et al., 2020). While the total $PM_{2.5}$ levels were highest in Addis Ababa during the rainy season (June-September) from increased motor vehicle and biomass burning emissions, the soil dust component (13.5%) of $PM_{2.5}$ was highest during the dry season from unpaved roadways and construction activities (Tefera et al., 2020). Depending on the meteorology, winds may transport desert mineral dust into a given region, or a temperature inversion can trap pollutants near the surface. Dust storms during the Harmattan season (November-March) greatly enhance $PM_{2.5}$ levels.

Environmental decision-makers are limited in their ability to regulate air quality by a paucity of reliable air quality measurements in sub-Saharan Africa. Two such end-user organizations are the Addis Ababa Environmental Projection and Green Development Commission (AAEPGDC) and Ghana Environmental Protection Agency (EPA Ghana). Many of the reference-grade $PM_{2.5}$ monitors are located in the largest cities and became operational only in 2020-2021. To fill in the in-situ spatiotemporal measurement gaps, satellite remote sensing data are ingested by air quality models. GEOS-Chem is a global 3-D atmospheric chemistry model driven by assimilated meteorological data from NASA Goddard Earth Observation System (GEOS), along with emissions, chemistry, aerosol microphysics, and deposition (GEOS-Chem, 2021). It is employed by research groups around the globe, but predicting $PM_{2.5}$ at a high resolution and for a long time period can be very computationally intensive. To obtain faster results and save costs, different machine learning techniques are being explored. Recently, Zhang et al. (2021) used a random forest algorithm to estimate $PM_{2.5}$ in South Africa with the following inputs: $PM_{2.5}$ at 20 monitoring stations, aerosol optical depth, two emission

parameters, four socioeconomic parameters, three land cover parameters, and 13 meteorological parameters. They obtained a cross-validation R^2 of 0.80, found seasonal indicator to be the most important predictor, and noticed that the model underestimated at $PM_{2.5} > 35$ μg m^{-3} but overestimated at $PM_{2.5} < 5$ μg m^{-3} (Zhang et al., 2021). Although complex machine learning models have proven to be skillful, there is always a risk of overfitting, and proper hyperparameter tuning takes time.

Objectives

The primary objectives of this project were the following:

1. Utilize machine learning to estimate daily mean $PM_{2.5}$ concentrations for nine major cities in sub-Saharan Africa
2. Compare the model performance of multiple linear regression, as a baseline, with that of decision-tree-based algorithms, namely random forest and Extreme Gradient Boosting (XGBoost)
3. Assess the importance of each of 11 meteorological and satellite features used to predict $PM_{2.5}$

Based on previous work, XGBoost was expected to perform the best, followed closely by random forest. In addition, aerosol optical depth was anticipated to be the most important variable because it has been shown to have a strong positive relationship with $PM_{2.5}$.

Data

Within sub-Saharan Africa, this project focused on nine air quality monitoring sites (Figure 1). We were initially interested in comparing western (six sites) and eastern (three sites) regions, but the total area (all sites) was ultimately used for the results in this paper. Each site is located in a country's capital, except for Lagos which is the largest city in Nigeria and out of all the cities. Table 1 shows that the cities have populations in the millions, which suggests high levels of air pollution. There is a wide range of elevations: Lagos, Conakry, Abidjan, and Accra

are near sea level; Bamako, Khartoum, and Abuja are at 1000-2000 ft; Kampala is near 4000 ft; and Addis Ababa is near 8000 ft. In western Africa, there are clear wet (summer) and dry (winter) seasons. Climate differences in eastern Africa are related to latitude and elevation: Kampala is near the equator, hilly, and wet year-round; Addis Ababa is surrounded by mountains and has a short wet season; and Khartoum lies in the hot, dry, and flat Sahel region.

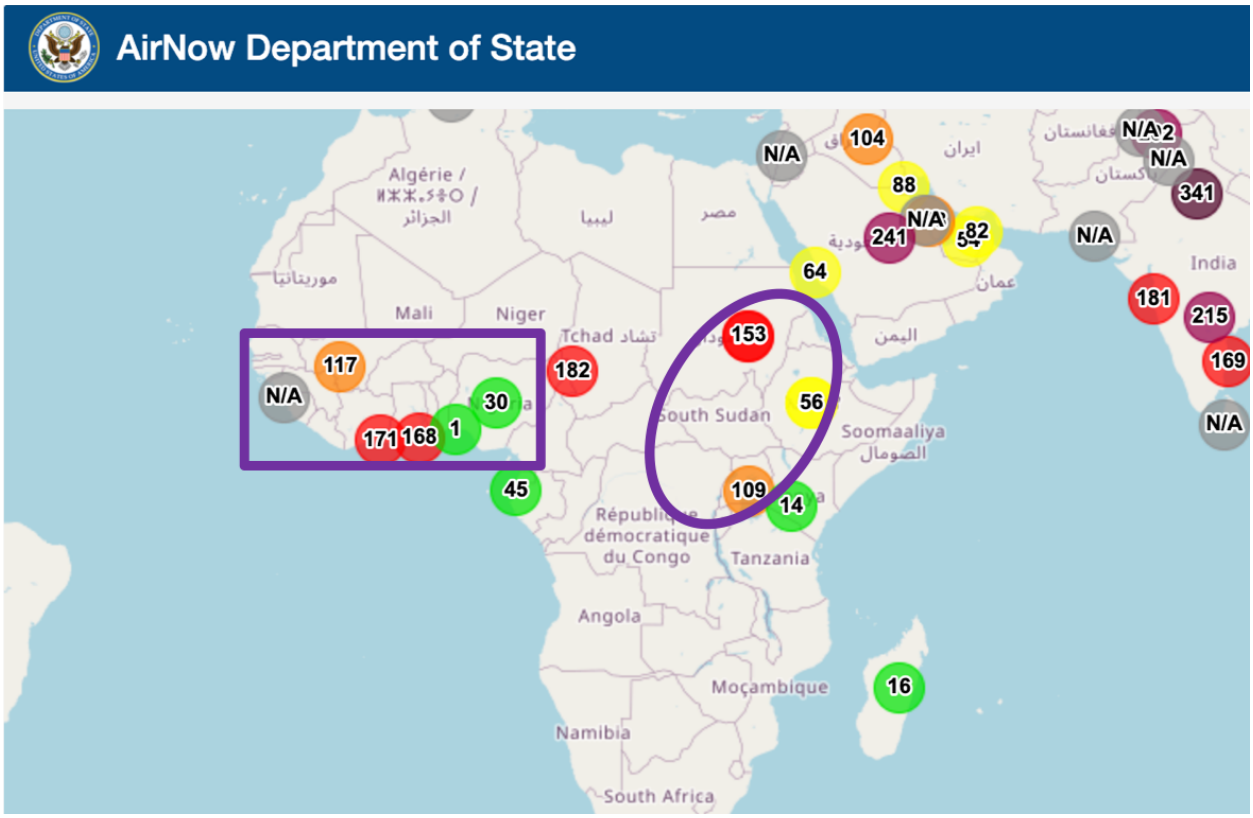

Figure 1. International map of operational air quality monitoring sites (AirNow.gov). The total study area includes nine sites, six located in western Africa (purple rectangle) and three located in eastern Africa (purple oval).

City	Country	Region	Elevation (feet)	Population (2021)	Climate Type
Abidjan	Ivory Coast	West	95	4,787,000	Tropical Wet and Dry
Abuja	Nigeria	West	1,542	3,499,000	Tropical Wet and Dry
Accra	Ghana	West	115	5,074,000	Tropical Wet and Dry
Addis Ababa	Ethiopia	East	7,726	5,503,000	Subtropical Highland
Bamako	Mali	West	1,096	3,750,000	Tropical Wet and Dry
Conakry	Guinea	West	42	2,729,000	Tropical Monsoon
Kampala	Uganda	East	3,898	4,495,000	Tropical Rainforest
Khartoum	Sudan	East	1,263	6,071,000	Hot Desert
Lagos	Nigeria	West	36	15,487,000	Tropical Wet and Dry

Table 1. Overview of each of the cities, including elevation (Burle, 2018), population (Demographia, 2021), and climate type (Beck et al., 2018).

Daily mean $PM_{2.5}$ ($\mu g\ m^{-3}$) observations from U.S. embassies and consulates were obtained from the AirNow website. Using the Copernicus Climate Data Store (CDS) API, daily fifth generation ECMWF reanalysis for global climate and weather (ERA5) products were downloaded for five meteorological parameters (#1-5 in Table 2). Thereafter, the Google Earth Engine API was used to download remote sensing products for six satellite-measured parameters (#6-11 in Table 2).

#	Variable	Name	Units
1	t2m	2-m Temperature	K
2	u10	10-m U-Component of Wind	m s^{-1}
3	v10	10-m V-Component of wind	m s^{-1}
4	sp	Surface Pressure	Pa
5	tp	Total Precipitation	m
6	co	Carbon Monoxide	mol m^{-2}
7	hcho	Formaldehyde	mol m^{-2}
8	o3	Ozone	mol m^{-2}
9	so2	Sulfur Dioxide	mol m^{-2}
10	no2	Nitrogen Dioxide	mol m^{-2}
11	aod	Aerosol Optical Depth (AOD)	Dimensionless

Table 2. List of predictors used in this project.

Surface 1-km AOD was derived via the Moderate Resolution Imaging Spectroradiometer (MODIS) Terra and Aqua combined Multi-angle Implementation of Atmospheric Correction (MAIAC). For each of the tropospheric trace gases (#6-10 in Table 2), measured by the Tropospheric Monitoring Instrument (TROPOMI), vertically integrated column number density was requested. The daily mean data for all sites were aggregated into two data frames, one for western Africa (2249 rows) and another for eastern Africa (2029 rows), and then combined into a single data frame (4278 rows). Note that the rows correspond to days with available $PM_{2.5}$ data and time gaps exist. Furthermore, the time periods varied depending on the inception of site measurements (all ending May 27, 2021):

1. Addis Ababa = January 1, 2019
2. Kampala = January 1, 2019
3. Khartoum = January 8, 2020
4. Abidjan = February 3, 2020
5. Abuja = February 12, 2021

6. Accra = January 28, 2020
7. Bamako = October 9, 2019
8. Conakry = January 27, 2020
9. Lagos = January 1, 2021

Methods

Before training the models, the data needed to be cleaned. Negative trace gas values were deemed incorrect and therefore replaced by “nan” (not a number) in Python. Since this created more data gaps, linear interpolation was performed for each site. An alternative approach would be dropping rows with any missing values; however, this was undesirable because the number of data points would be drastically reduced. A correlation matrix was created to understand the linear relationships between variables. Reducing the number of features to the top five highest correlated with $PM_{2.5}$ degraded model performance; consequently, all 11 variables were utilized for this paper. From the popular Python machine learning library, Scikit-learn, the data were split following standard practice into training (80%) and test (20%) sets, with a defined “random state” to ensure reproducibility.

Multiple linear regression is a simple model that minimizes the residual sum of squares between observed and predicted variables. For this project, coefficients and variables were combined to derive the following equation:

$$\begin{aligned} pm25 = {} & -350.5433 + (1.6758 \cdot t2m) - (3.2397 \cdot u10) - (1.7398 \cdot v10) \\ & - (0.0003 \cdot sp) + (2822.8269 \cdot tp) + (620.9833 \cdot co) \\ & + (3996.3040 \cdot hcho) - (986.8652 \cdot o3) + (8206.9625 \cdot so2) \\ & + (17341.2729 \cdot no2) + (18.8365 \cdot aod) \end{aligned} \quad (1)$$

Increasingly, decision-tree-based algorithms have demonstrated success in a variety of disciplines for both classification and regression tasks. Some of the benefits include versatility with large and small data sets, ability to handle missing data, inclusion of feature importance, and balance between simplicity and accuracy (Biau & Scornet, 2016). Figures 2 presents a sample decision tree with two levels from a small forest of 10 trees. Starting from the root node

(top), variables such as “t2m”, “co”, and “aod” are tested, and the results are split by branches. The terminal nodes or leaves (bottom) contain the final outcomes for $PM_{2.5}$. Many such trees are constructed from bootstrap samples with replacement, and their predictions are averaged through bootstrap aggregation (bagging) to reduce the variance. To improve on bagging, random forest performs splits only on a random subset of features, thereby decorrelating the trees. More recently, XGBoost has emerged as a way to increase model speed and performance. Boosting constructs trees sequentially based on the residuals of the previous tree. Gradient descent is used to minimize the loss function.

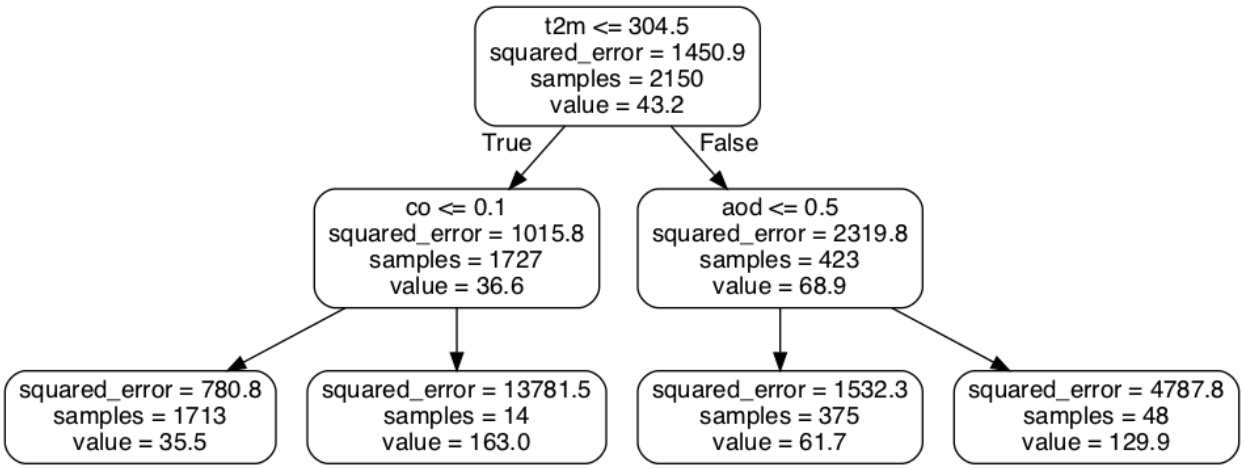

Figure 2. A decision tree extracted from a forest of n_estimators = 10 and max_depth = 2.

Decision-tree-based algorithms have a black box flavor, as the full theory is esoteric and underdeveloped. To improve the models, this paper explored some basic hyperparameter tuning (Table 3). From the Scikit-learn package, “RandomizedSearchCV” was first implemented to narrow down the range of possible values for each hyperparameter to test. Next, “GridSearchCV” was run using three-fold cross validation to attempt to find the optimal parameters. Predictions were made based on these best parameters, which differed between random forest and XGBoost, mainly to reduce the likelihood of overfitting. There were 3 folds x 81 candidates = 243 fits (four hyperparameters) for random forest and 3 folds x 243 candidates = 729 fits (five hyperparameters) for XGBoost. The entire Python script took about 30 minutes to finish running.

Model	Hyperparameter	Name	Test Values	Best Value
RF	n_estimators	Number of trees	[400, 500, 600]	500
RF	max_depth	Maximum tree depth	[20, 30, 40]	30
RF	min_samples_leaf	Minimum samples at leaf node	[1, 2, 3]	1
RF	min_samples_split	Minimum samples to split internal node	[2, 3, 4]	2
XGB	n_estimators	Number of trees	[400, 500, 600]	400
XGB	max_depth	Maximum tree depth	[7, 8, 9]	8
XGB	learning_rate	Learning rate for gradient boosting	[0.05, 0.06, 0.07]	0.06
XGB	colsample_bytree	Column subsample ratio for each tree	[0.7, 0.8, 0.9]	0.8
XGB	subsample	Subsample ratio from training set	[0.7, 0.8, 0.9]	0.8

Table 3. Hyperparameters tested and used for random forest (RF) and XGBoost (XGB) in this project.

Results and Discussion

As displayed in Figure 1, the $PM_{2.5}$ levels were relatively low (mostly < 100 $\mu g\ m^{-3}$) throughout the time period in Abidjan and Addis Ababa. This suggests that sources of air pollution have been kept under control quite well in these cities. Kampala had moderate levels of $PM_{2.5}$ with considerable daily noise, likely due to meteorological variability. Conakry and Accra each had one extreme pollution episode (374 and 483 ug/m3), possibly signatures of dust storms. The remaining cities—Abuja, Bamako, Lagos, and Khartoum—each had multiple severe pollution days (> 200 $\mu g\ m^{-3}$). For sites with longer time periods, the seasonal cycle is more prominent; for example, Bamako and Khartoum appear to have lower average $PM_{2.5}$ concentrations between July and November.

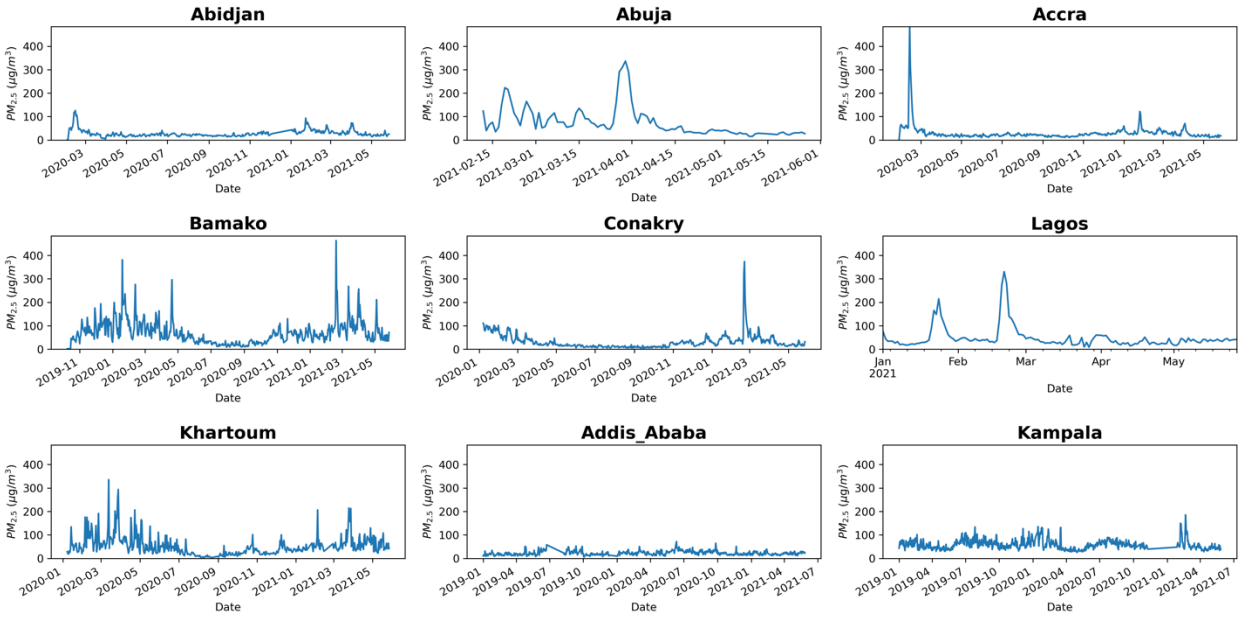

Figure 3. Time series of daily mean $PM_{2.5}$ observations for each city in sub-Saharan Africa. Note that the range of $PM_{2.5}$ (y-axis) is the same, but the time period (x-axis) varies by site.

While a plethora of linear relationships could be analyzed between the variables (Figure 4), a brief summary is provided here. Temperature, CO, NO_2, and AOD were positively correlated with $PM_{2.5}$ ($r \geq 0.2$), while both wind components and O_3 were negative correlated with $PM_{2.5}$ ($r \leq -0.2$). Northeasterly trade winds during the Harmattan season is consistent with elevated $PM_{2.5}$ levels due to desert dust. Surface pressure had a relatively strong linear relationship with temperature and CO ($r \geq 0.6$). This may be explained by higher temperatures at lower elevations (higher pressure), where there tend be more cities with greater sources of emissions. Temperature and precipitation were most negatively correlated ($r = -0.32$). Typically, higher temperatures in Africa are associated with clear conditions, which means less rainfall.

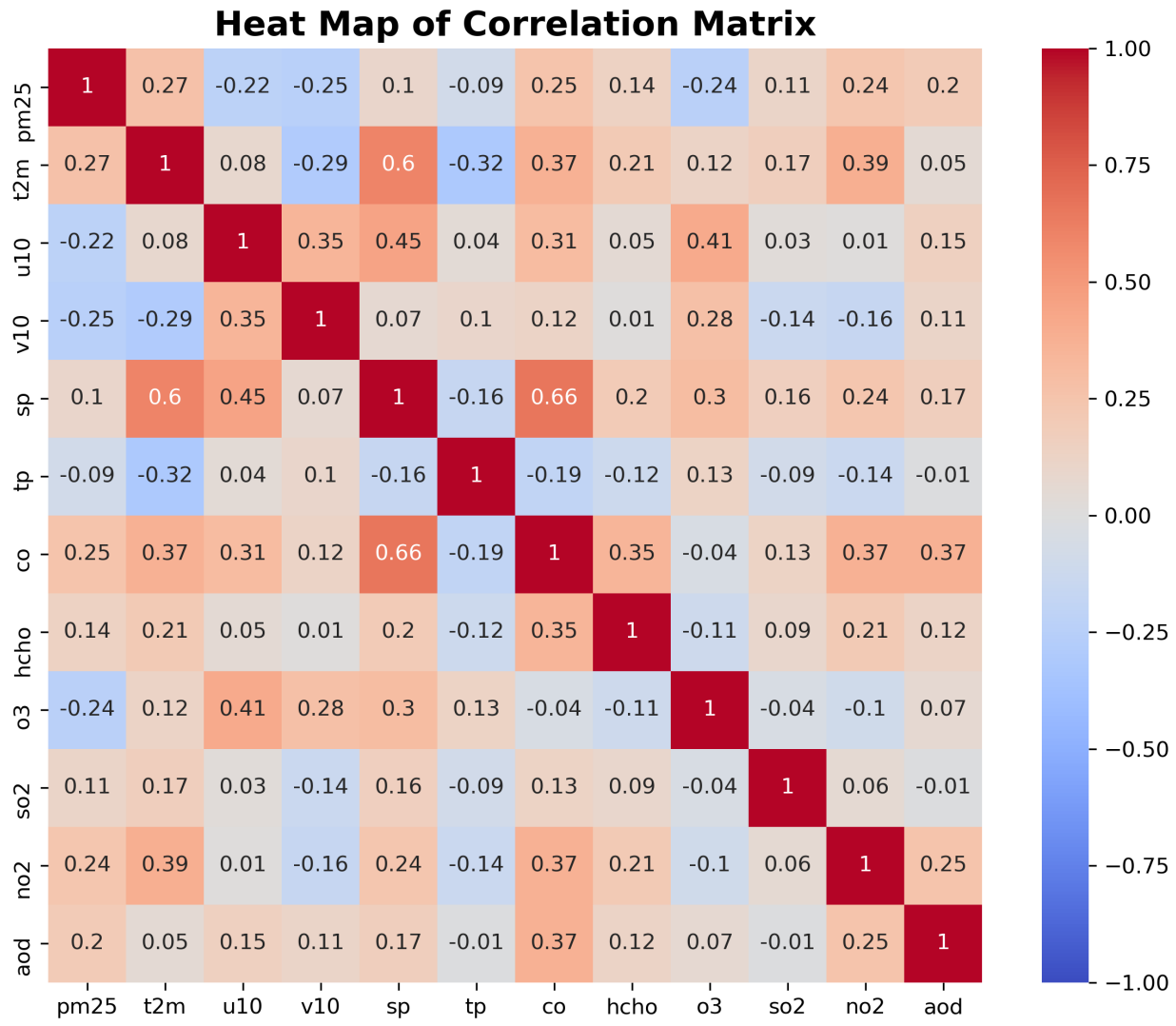

Figure 4. Correlation coefficient for any two variables including $PM_{2.5}$ and the 11 features.

The model evaluation results in Figure 5 show that XGBoost performed the best (r^2 = 0.63), having a slight edge over random forest (r^2 = 0.61) and significant advantage over multiple linear regression (r^2 = 0.24). Differences between random forest and XGBoost are small and influenced by hyperparameter tuning. Either decision-tree-based algorithm can be used to reasonably estimate daily $PM_{2.5}$ for cites located in sub-Saharan Africa within about 25 μg m^{-3} on average. However, peaks in $PM_{2.5}$ tend to be difficult to predict, as indicated by the outliers. All the models underestimate $PM_{2.5}$ overall, based on the best fit line, but multiple linear

regression especially underestimates (up to 3-5 times less). These models are more suitable for predicting $PM_{2.5}$ on low pollution days or on average, for example monthly.

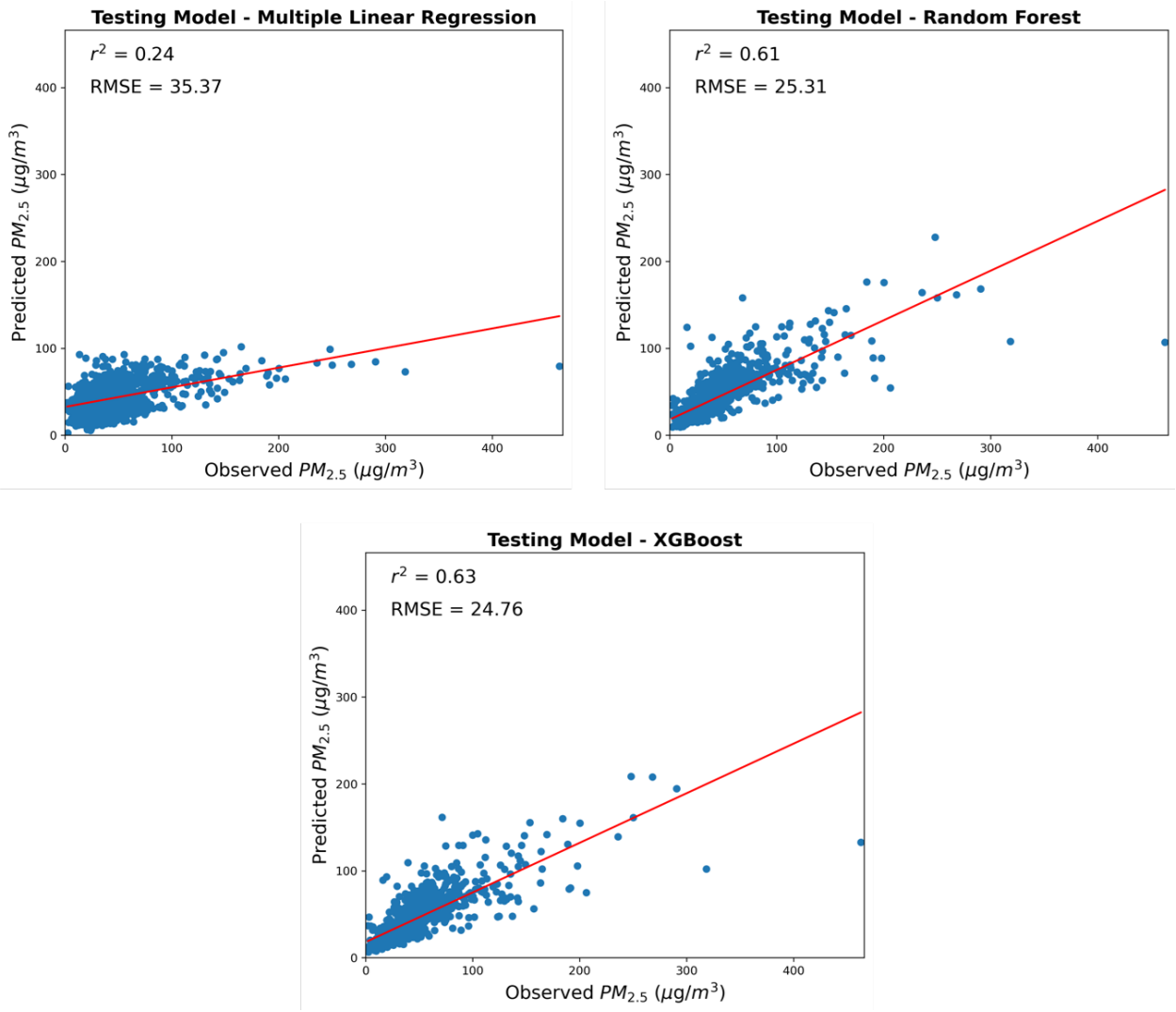

Figure 5. $PM_{2.5}$ predictions compared with test set observations for multiple linear regression (top left), random forest (top right), and XGBoost (bottom center). A higher coefficient of determination (r^2) or lower root mean squared error (RMSE) indicates better model performance.

Despite the similarities between random forest and XGBoost, discrepancies in feature importance may exist. Figure 6 shows that AOD was the most important variable in both models. Indeed, past literature has shown that AOD can be a good proxy for $PM_{2.5}$. However,

while total precipitation was considered important by XGBoost, it was one of the least important variables in random forest. Surface pressure was fairly important in both models, possibly a sort of classifier by site elevation. CO was also quite important in both algorithms, but SO_2 and NO_2 appear to be least important. The larger contribution of CO might reveal the dominant sources of pollution, such as vehicular emissions and biomass burning, in African cities.

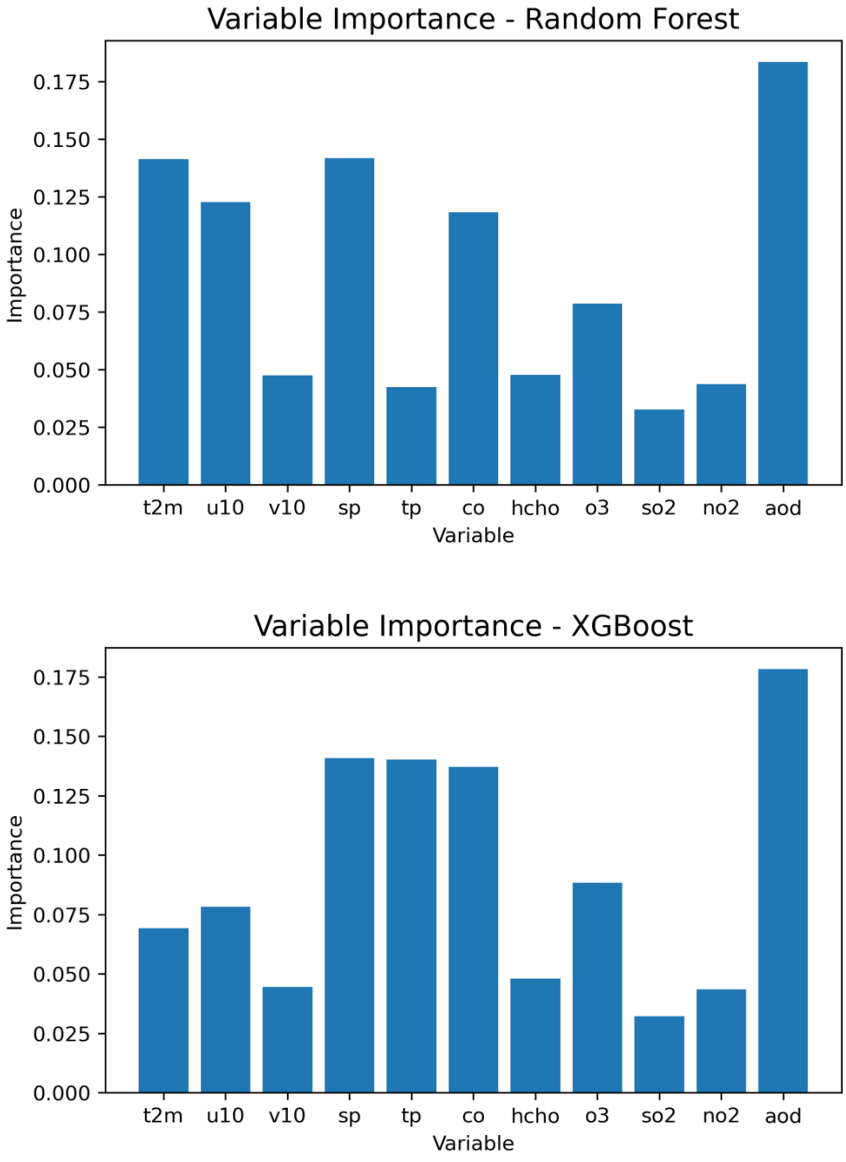

Figure 6. Comparison of random forest and XGBoost feature importance. Higher values denote greater importance.

In Figure 7, the machine learning predictions overlay the original observed data time series (training + testing) for each city. At first glance, multiple linear regression deviates substantially from the observed data, overestimating many of the lower $PM_{2.5}$ levels and failing to capture some of the peaks. This is confirmed in Figure 8, which shows that the RMSE for MLR always exceeds that of RF or XGB. MLR performed best for Abidjan (20.3 ug/m3) and worst for Abuja (55.2 ug/m3). Compared with random forest, XGBoost had a lower RMSE or stronger performance for each city. RF performed best for Addis Ababa (4.3 ug/m3) and worst for Bamako (23.7 ug/m3). XGB also performed best for Addis Ababa (3.8 ug/m3) and worst for Bamako (20.0 ug/m3). The skill of the decision-tree-based algorithms seems to depend on the variability within a given time series. Addis Ababa had consistently low $PM_{2.5}$ concentrations, while Abuja had many large spikes in a shorter time period. Although the exact magnitudes of peaks might have not be captured, random forest and XGBoost appear to have recognized the occurrence of many of these peaks which corresponded to severe pollution episodes.

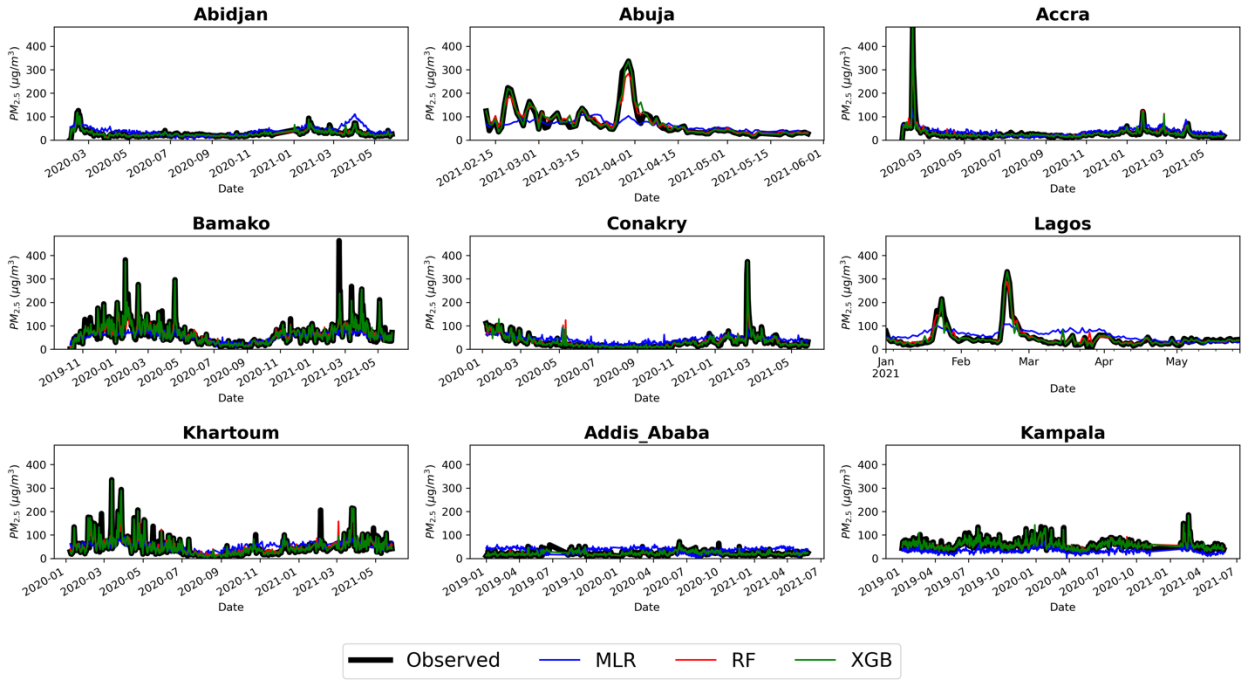

Figure 7. Time series of predicted versus observed daily mean $PM_{2.5}$ concentrations for each city in sub-Saharan Africa. Observed data (bold black), multiple linear regression (blue), random forest (red), and XGBoost (green) are compared.

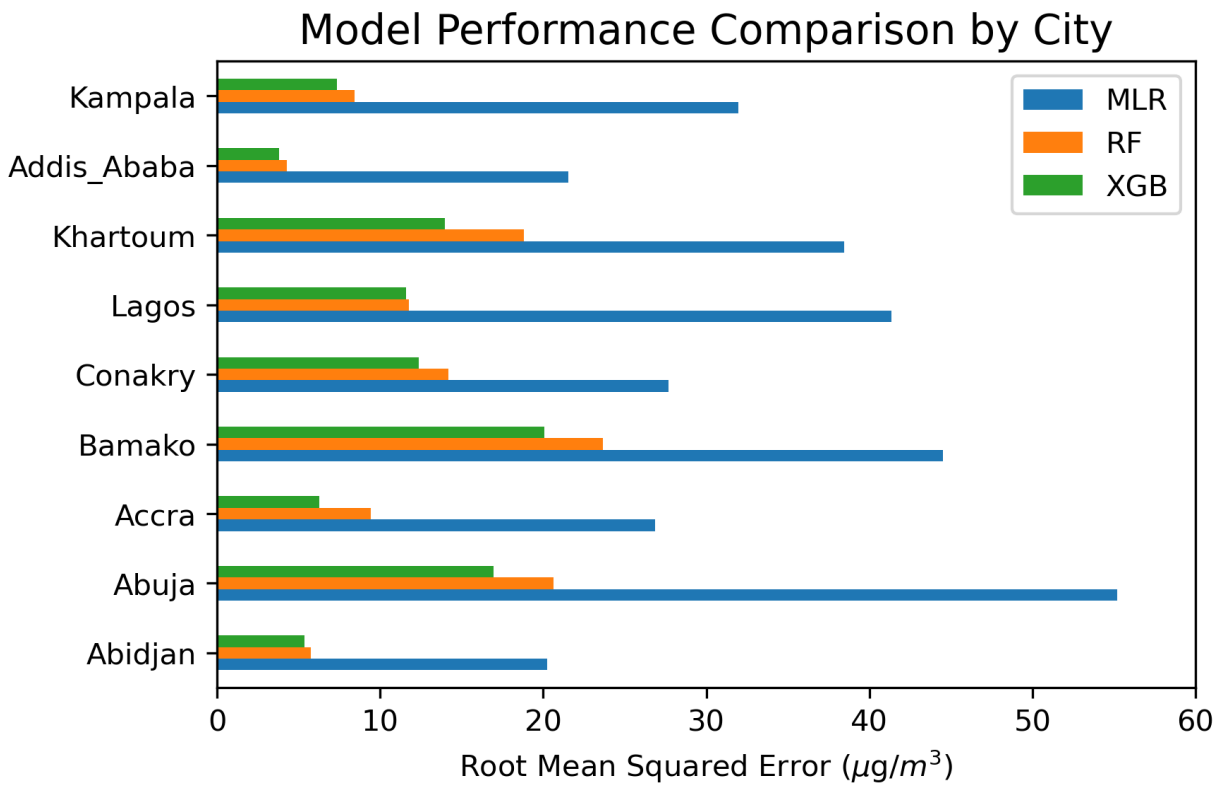

Figure 8. Performance of multiple linear regression (blue), random forest (orange), and XGBoost (green) using the entire training + testing data for each city. Lower $PM_{2.5}$ root mean squared error indicates better model performance.

Conclusions

This project aimed to use three machine learning models to predict daily $PM_{2.5}$ concentrations for nine cities in sub-Saharan Africa. For the entire test set and each city, the decision-tree-based algorithms clearly outperformed the simple multiple linear regression approach. XGBoost had a 0.55 ug/m3 lower RMSE than random forest for the test set, which suggests XGBoost was the best model in this particular study. Both decision-tree-based algorithms indicated that AOD was most important feature, while NO_2 and SO_2 were among the least important features. All the models tended to underestimate $PM_{2.5}$ levels, which could lead to greater exposure to air pollution. Furthermore, the models appeared to perform better for cities with consistently low $PM_{2.5}$ concentrations and worse for those with multiple large peaks.

This suggests that random forest or XGBoost may be useful for estimating $PM_{2.5}$ levels on monthly to seasonal time scales and at least recognizing the occurrence of severe pollution episodes on shorter time scales.

Additional hyperparameter tuning in the future would increase confidence in the decision-tree-based algorithms. Moreover, data collected over a longer time period and added significant variables, such as planetary boundary layer depth or land cover, would be of interest. As demonstrated by Zhang et al. (2021), machine learning models such as random forest have the potential to provide epidemiologists and policymakers with high spatiotemporal maps. Machine learning has the potential to replace elements of chemical transport models such as GEOS-Chem. This could help speed up model runs and updates, saving time and money. As new air quality monitoring sites are established and existing ones continue to collect measurements throughout Africa, spatiotemporal data gaps can be filled and, coupled with modeling, support efforts to reduce exposure to air pollution.

Acknowledgments

Special thanks to Dr. Zhonghua Zheng for assisting with the data and providing advice throughout the semester. I would also like to thank my advisor, Professor Dan Westervelt, for our discussions about air quality in Africa and sharing relevant papers. Finally, this project and class would not be possible without the generous support of Professor Pierre Gentine.

References

AirNow.gov, U.S. EPA. (n.d.). US Embassies and Consulates | AirNow.gov. Retrieved December 20, 2021, from https://www.airnow.gov/international/us-embassies-and-consulates/

Ambient (outdoor) Air Pollution. World Health Organization. (2021, September 22). Retrieved December 20, 2021, from https://www.who.int/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health

Beck, H. E., Zimmermann, N. E., McVicar, T. R., Vergopolan, N., Berg, A., & Wood, E. F. (2018). Present and future Köppen-Geiger climate classification maps at 1-km resolution. *Scientific Data*, *5*. https://doi.org/10.1038/sdata.2018.214

Biau, G., & Scornet, E. (2016). A Random Forest Guided Tour. *Springer*, *25*(2), 197–227. https://doi.org/10.1007/s11749-016-0481-7

Burle, S. (2018). *World Elevation Map for Elevation and Elevation Maps of Cities/Towns/Village*. Flood Map. Retrieved December 20, 2021, from https://www.floodmap.net/Elevation/WorldElevationMap/

Demographia. (2021). (rep.). *Demographia World Urban Areas* (17th ed.). http://www.demographia.com/db-worldua.pdf

ECMWF. (2018, June 14). *ERA5 hourly data on single levels from 1979 to present*. Copernicus Climate Data Store . Retrieved December 20, 2021, from https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-single-levels?tab=overview

Fisher, S., Bellinger, D. C., Cropper, M. L., Kumar, P., Binagwaho, A., Koudenoukpo, J. B., Park, Y., Taghian, G., & Landrigan, P. J. (2021). Air Pollution and development in Africa: impacts on health, the economy, and human capital. *The Lancet Planetary Health*, *5*(10). https://doi.org/10.1016/s2542-5196(21)00201-1

Google. (n.d.). *MCD19A2.006: Terra & Aqua Maiac land aerosol optical depth daily 1km* . Earth Engine Data Catalog. Retrieved December 20, 2021, from https://developers.google.com/earth-engine/datasets/catalog/MODIS_006_MCD19A2_GRANULES#citations

Google. (n.d.). *Sentinel-5P Datasets in Earth Engine* . Earth Engine Data Catalog. Retrieved December 20, 2021, from https://developers.google.com/earth-engine/datasets/catalog/sentinel-5p

Koehrsen, W. (2017, December 27). *Random Forest in Python*. Towards Data Science. Retrieved December 20, 2021, from https://towardsdatascience.com/random-forest-in-python-24d0893d51c0

Koehrsen, W. (2018, January 9). *Hyperparameter Tuning the Random Forest in Python*. Towards Data Science. Retrieved December 20, 2021, from https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74

Narrative description. GEOS-Chem. (2021, March 22). Retrieved December 20, 2021, from https://geos-chem.seas.harvard.edu/narrative

Tefera, W., Kumie, A., Berhane, K., Gilliland, F., Lai, A., Sricharoenvech, P., Samet, J., Patz, J., & Schauer, J. J. (2020). Chemical characterization and seasonality of ambient particles (PM2.5) in the city centre of Addis Ababa. *International Journal of Environmental Research and Public Health*, *17*(19). https://doi.org/10.3390/ijerph17196998

Yadav, H. (2021, May 7). *Multiple linear regression implementation in Python*. Machine Learning with Python. Retrieved December 20, 2021, from https://medium.com/machine-learning-with-python/multiple-linear-regression-implementation-in-python-2de9b303fc0c

Zhang, D., Du, L., Wang, W., Zhu, Q., Bi, J., Scovronick, N., Naidoo, M., Garland, R. M., & Liu, Y. (2021). A machine learning model to estimate ambient PM2.5 concentrations in industrialized Highveld Region of South Africa. *Remote Sensing of Environment*, *266*. https://doi.org/10.1016/j.rse.2021.112713

Zhao, R., Gu, X., Xue, B., Zhang, J., & Ren, W. (2018). Short period PM2.5 prediction based on multivariate linear regression model. *PLOS ONE*, *13*(7). https://doi.org/10.1371/journal.pone.0201011