AIMS Environmental Science, 13(1): 126–158.
DOI: 10.3934/environsci.2026006
Received: 18 November 2025
Revised: 31 January 2026
Accepted: 10 February 2026
Published: 28 February 2026

https://www.aimspress.com/journal/environmental

Research article

Air pollutant PM_{10} estimation in Saudi Arabia using machine learning

Amjad Alkhodaidi[1,*], Abeer Hakeem[1], Afraa Attiah[1], Alaa Mhawish[2] and Abeer Almakky[1]

[1] Department of Information Technology, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia; ahakim@kau.edu.sa (A.H.); aattiah@kau.edu.sa (A.A.); aalmakky@kau.edu.sa (A.M); aalkhodaidi0001@stu.kau.edu.sa (A.A.)

[2] Sand and Dust Storm Regional Center, National Center for Meteorology, Jeddah, Saudi Arabia; a.mhawish@ncm.gov.sa (A.M.)

* **Correspondence:** Email: aalkhodaidi0001@stu.kau.edu.sa (A.A.); Tel: +966565739534.

Abstract: Air pollution, specifically PM_{10}, is a critical challenge in Saudi Arabia, where levels often exceed World Health Organization (WHO) guidelines due to industrial activities and arid conditions, posing serious risks to human health. A key barrier is obtaining accurate PM_{10} data, as estimations are limited by the few and unevenly distributed air quality stations. Notably, despite its severity, research on PM_{10} estimation in the region remains scarce. Atmospheric reanalysis datasets like MERRA-2 offer complementary data, but their model-based nature, lacking actual measurements, introduces potential biases. To bridge this gap, this study developed a machine learning framework to estimate daily and monthly PM_{10} concentrations in three climatically distinct Saudi cities. The framework integrates ground-based PM_{10} data, meteorological parameters, and MERRA-2 reanalysis data. To our knowledge, this study represents the first application of MERRA-2 for PM_{10} estimation in Saudi Arabia. The proposed AtmoStack is a stacked machine learning model, and we compared it against individual models (RF, HGB, CatBoost, and MLP) and state-of-the-art models, including LightGBM, ANN, and LSTM. Moreover, the framework incorporates feature-importance analysis to identify the most influential factors, helping to interpret the model. AtmoStack outperformed all baselines; in the dust-dominated environment of Buraidah, it achieved a daily R^2 of 0.73 and a monthly R^2 of 0.96. In Taif, it achieved a daily R^2 of 0.63 and a monthly R^2 of 0.94, indicating that AtmoStack effectively captures realistic distribution characteristics. These results support effective air-quality management and public health decisions.

Keywords: particulate matter; PM_{10}; machine learning; stacking; random forest; gradient boosting; air quality; Saudi Arabia

1. Introduction

Rapid industrialization and urban development have contributed to a marked deterioration in air quality worldwide [1, 2]. Particulate matter (PM)—a mix of solid particles and liquid droplets in the air—is a major pollutant of concern. PM is categorized by aerodynamic diameter (e.g., PM_1, $PM_{2.5}$, PM_4, PM_{10}) [3, 4]. Smaller particles (e.g., PM_1 and $PM_{2.5}$) penetrate deeper into the lungs and are generally more harmful [5]. Epidemiological studies link PM exposure to respiratory and cardiovascular disease; for instance, long-term $PM_{2.5}$ exposure has been associated with increased death rates [6–8]. To protect public health, the World Health Organization's 2021 Air Quality Guidelines recommend an annual mean of 5 μg/m³ for $PM_{2.5}$ and 15 μg/m³ (24h) for PM_{10} [9]. These thresholds are routinely exceeded in heavily polluted regions: many urban areas in Saudi Arabia report annual PM_{10} well above 15 μg/m³ (for example, some Riyadh sites have seen means on the order of 100 μg/m³ or more) [10]. Such extreme pollution levels are associated with acute health impacts (asthma attacks, cardiac events) during spikes and increased chronic mortality [8, 11, 12]. Beyond human health, high PM_{10} also reduces visibility (impacting aviation and transport), depletes soil nutrients, and alters the climate by scattering solar radiation and modifying clouds [5].

The Arabian Peninsula, especially Saudi Arabia, faces extreme PM_{10} pollution due to its geography and climate. The Kingdom lies in the global "dust belt," dominated by vast deserts (e.g., the Rub al-Khali, An Nafud, and Ad Dhana). Seasonal and synoptic winds frequently lift mineral dust from these sand seas, leading to dust storms that can drive ambient PM_{10} to very high concentrations [12, 13]. In addition to natural dust, rapid urbanization and mega-construction projects (such as NEOM and the Red Sea development) liberate large amounts of coarse dust year-round. Urban traffic and industry (cement plants, petrochemicals, diesel fleets) further contribute to PM_{10} via soil resuspension and industrial emissions. In short, Saudi Arabia's climate and development together generate extremely high PM_{10} loads [14]. This makes accurate PM_{10} monitoring critical, but the national air-quality network (operated by the National Center for Environmental Compliance) is sparse outside the main cities, and station outages or drifts create frequent data gaps. Expanding the network over a 2.2-million-km^2 area is prohibitively expensive, which motivates the use of models to fill spatial and temporal gaps [15, 16].

Global reanalyses offer one partial solution. The MERRA-2 reanalysis dataset from the National Aeronautics and Space Administration (NASA) provides a 40-year aerosol reanalysis (with a $0.5° \times 0.625°$ spatial grid and 3-hourly temporal resolution) that includes mass mixing ratios for dust, sea salt, carbonaceous, and sulfate aerosols [17, 18]. Although MERRA-2 does not directly provide surface PM_{10} measurements, it can be converted via a standard algorithm into a proxy PM_{10} field (summing dust bins up to 10 μm, organics, black carbon, sulfate, etc., and multiplying by air density) [19]. MERRA-2 is physically consistent and globally complete. However, it has well-known limitations in regions like Saudi Arabia. Its coarse grid smooths out local sources (e.g., mountain channels, urban hot spots), its 3-hourly averaging dilutes short-lived dust events, and it omits some species (nitrates, crustal metals) while assuming fixed particle densities. Consequently, raw MERRA-2 PM_{10} often deviates from ground measurements by tens of percents: Strong dust events tend to be underestimated, while calm polluted days can be overestimated [1].

Given these challenges, advanced statistical methods are needed to improve PM_{10} estimates. Traditional regression-based models (e.g., linear or mixed-effects models that link satellite aerosol optical depth (AOD) or reanalysis to ground PM) have been widely used, but they struggle with

complex, nonlinear pollution dynamics [20]. In contrast, machine learning (ML) models can automatically learn nonlinear interactions among predictors (meteorology, land cover, soil state, emissions, etc.). For example, tree-based ML models can capture threshold effects (e.g., abrupt dust emission rise when wind exceeds a certain speed) and interaction effects (e.g., interplay of soil moisture and wind). ML also enables the integration of many data sources (satellite AOD, reanalysis meteorology, land use, time lags, etc.) without manual selection [21]. Critically, ensemble ML methods (random forests, gradient boosting, etc.) typically yield highly accurate PM estimates [22,23]. Recent research indicates that ML and deep-learning approaches markedly outperform conventional statistical models for PM. For instance, random forests and gradient-boosted trees typically explain 65%–90% of daily PM variance in cross-validation, often 10–20 percentage points higher R^2 than linear models [20, 24]. Deep neural networks such as convolutional neural networks (CNNs) and long short-term memory networks (LSTMs) have also been successfully applied to learn spatiotemporal pollution patterns. Moreover, ensemble methods that combine multiple learners can further improve the capture of extreme events [24, 25]. To date, however, there has been no comprehensive application of ensemble ML-based calibration specifically to Saudi Arabian PM_{10}.

In this study, we develop a machine learning framework—AtmoStack—for accurate, spatially resolved PM_{10} estimation across Saudi Arabia. The framework integrates multiple datasets, applies feature engineering and selection, and employs a stacking ensemble to exploit the complementary strengths of several tree-based models. A key contribution of this work is that, to the best of our knowledge, it represents the first application of MERRA-2 reanalysis data for PM_{10} estimation in Saudi Arabia, enabling a richer characterization of large-scale aerosol and meteorological dynamics. The framework combines five years (2019–2023) of daily ground-based PM_{10} observations with MERRA-2 aerosol and meteorological fields, along with additional covariates, across three climatically distinct cities: Abha (southwest highlands), Buraidah (central plateau), and Taif (western escarpment).

Within this framework, meaningful features are engineered to capture spatio-temporal and process-related dependencies, including autoregressive lag variables, 7-day rolling statistics, planetary boundary layer height, and categorical synoptic weather types. Feature selection techniques are applied to identify the most relevant predictors contributing to PM_{10} variability. To establish baseline performance, several individual models are trained, including random forest (RF), histogram-based gradient boosting (HGB), CatBoost (a gradient-boosted tree with optimized categorical handling), and a deep feedforward neural network (multilayer perceptron, MLP). The tree-based models (RF, HGB, and CatBoost) are then combined into the AtmoStack ensemble, which uses an elastic net meta-learner to optimally weigh each model and improve estimation accuracy.

Model performance is evaluated using time-series cross-validation, and the ML-based estimates are compared with ground-based PM_{10} measurements and the best single model. Finally, we interpret the fitted models to identify key predictors and error patterns, providing insight for integration with dust forecasting and public-health advisories. Through this pipeline, we aim to deliver a validated, operationally ready framework to estimate daily and monthly PM_{10} in Saudi Arabia. The enhanced estimations will support exposure assessment and targeted air-quality management in the Kingdom, contributing to the environmental objectives of Saudi Vision 2030.

2. Literature review

Global concern over air quality, driven by rapid urbanization and industrial growth, has made PM a primary focus of environmental and public health research [1]. Extensive epidemiological evidence has unequivocally linked PM exposure to serious health outcomes, including respiratory and cardiovascular diseases and increased mortality rates [6–8]. In response, authorities like the World Health Organization have established stringent air quality guidelines, making accurate PM estimation a critical societal imperative [9].

Achieving this goal is challenging. The "gold standard" for air quality measurement relies on ground-based monitoring stations, but their high cost results in sparse deployment, leaving vast geographical areas unmonitored [23, 26]. This spatial scarcity prevents a complete picture of population-level exposure, hindering effective public health policy. To address these data gaps, researchers have developed modeling techniques to estimate PM concentrations where direct measurements are unavailable. This review, guided by the findings of a recent systematic review by Alkhodaidi et al. [27], traces the methodological evolution of particulate matter (PM) estimation from traditional statistical approaches to the machine learning (ML) models that now dominate the field.

In the earlier phase of PM modeling, researchers relied on deterministic chemical transport models (CTMs) and empirical statistical approaches. CTMs, such as GEOS-Chem, simulate atmospheric transport and chemical processes [22, 28]. For example, Di et al. [29] applied the GEOS-Chem model to estimate $PM_{2.5}$ concentrations across the United States, achieving correlation coefficients of 0.70–0.80. Despite their physical interpretability, CTMs are computationally intensive and highly sensitive to emission inventories, which often carry substantial uncertainty [23].

In parallel, empirical statistical models were developed to quantify relationships between ground-level PM concentrations and proxy variables such as satellite-derived aerosol optical depth (AOD). linear mixed-effect (LME) models [30] and generalized additive models (GAMs) [22, 31] represented significant advances by allowing limited nonlinearity and spatial heterogeneity. However, their performance remains constrained by rigid statistical assumptions that are frequently violated in complex atmospheric systems [27, 32]. A critical weakness of these approaches is their inability to accurately capture extreme pollution events. Several studies have reported systematic underestimation of high PM concentrations during severe episodes [20]. For instance, Mhawish et al. [20] showed that an LME model struggled to estimate $PM_{2.5}$ values exceeding 100 $\mu g/m^3$ over the Indo-Gangetic Plain, despite achieving an overall R^2 of 0.78. Similarly, Meng et al. [22] reported an R^2 of 0.60 using a GAM for $PM_{2.5}$ estimation in Southern California. These persistent limitations motivated the transition toward more flexible, data-driven ML techniques.

Consequently, ML models have been extensively applied to PM estimation; however, their performance varies substantially depending on the input variables and regional characteristics. Support vector regression (SVR) has been frequently employed in PM_{10} estimation studies, yet its effectiveness remains inconsistent across different urban environments. Alsaber et al. [33] applied SVR alongside k-nearest neighbors (KNN) and artificial neural network (ANN) to estimate PM_{10} concentrations at two urban monitoring stations in Kuwait . SVR did not outperform the other models, and KNN and ANN achieved relatively high accuracy with R^2 values ranging from approximately 0.88 to 0.94. Similar behavior has been observed in other PM_{10}-focused studies. Bozdağ et al. [34] compared SVR, ANN, RF, and XGBoost for PM_{10} estimation in Ankara, Turkey, reporting that ANN achieved the

highest accuracy (R^2 = 0.58), outperforming SVR. Likewise, Suleiman et al. [35], in a roadside air quality study across 19 locations in London, found that ANN-based models provided superior PM_{10} estimations compared to SVR-based approaches. Collectively, these findings indicate that while SVR can yield acceptable performance, it often struggles to capture strong nonlinear relationships between PM_{10} concentrations and meteorological or traffic-related drivers in complex urban settings. In contrast, improved SVR performance has been reported in specific contexts. Son and Kim [36] estimated PM_{10} concentrations in Seoul using climatic data from 39 meteorological stations and achieved an R^2 of 0.77, approaching the performance of random forest (RF; R^2 = 0.79). This contrast underscores the context-dependent nature of SVR performance, suggesting that it can be competitive in regions characterized by dense monitoring networks and relatively homogeneous meteorological conditions, while exhibiting limited robustness across diverse climatic and emission regimes.

Among tree-based models, RF has demonstrated consistently strong and stable performance across pollutants, spatial scales, and regions. Mhawish et al. [20] showed that RF outperformed linear mixed-effect models for $PM_{2.5}$ estimation over the Indo-Gangetic Plain, achieving an R^2 of 0.87, with performance improving further when estimations were aggregated temporally. Similarly, Hu et al. [37] applied RF to estimate daily $PM_{2.5}$ concentrations across the contiguous United States and achieved an R^2 of 0.80 by integrating AOD, meteorological, and land-use variables. These results highlight RF's ability to effectively capture nonlinear interactions among heterogeneous predictors within a given climatological regime.

Although RF is a strong benchmark, several recent studies report that gradient boosting algorithms can further improve estimation accuracy, as they iteratively correct errors from previous models. Recent studies indicate that gradient boosting techniques can further enhance performance. Using MERRA-2 aerosol and meteorological data, Dhandapani et al. [23] found that XGBoost outperformed RF and LightGBM for $PM_{2.5}$ estimation, achieving an R^2 of 0.73. Similarly, Lee and Son [38] reported that XGBoost consistently yielded superior performance for both PM_{10} and $PM_{2.5}$ estimation with an R^2 of 0.89, particularly when air quality variables were included as predictors. The sequential error-correction mechanism of boosting algorithms allows for more precise learning of complex pollutant–meteorology relationships.

Beyond individual models, ensemble learning strategies that combine multiple learners have shown additional benefits. Dhandapani et al. [23] demonstrated that a stacking ensemble integrating XGBoost, RF, and LightGBM improved estimation accuracy to an R^2 of 0.77, outperforming all individual models. This reinforces the conclusion that no single algorithm consistently dominates PM estimation tasks and that ensemble frameworks can provide more robust estimations by leveraging complementary model strengths.

Despite these methodological advances, important gaps remain. Many studies validate their models within a single country or climatic regime, limiting geographical transferability. Notably, PM_{10}-focused research remains scarce compared to the extensive work on $PM_{2.5}$. This distinction is critical in arid regions like Saudi Arabia, where PM_{10} is primarily driven by natural desert dust and mechanical suspension rather than the combustion-based processes typical of $PM_{2.5}$ research [39, 40]. Consequently, there is a clear need for region-specific investigations in Saudi Arabia that account for these unique climatic conditions and dust-driven processes.

3. Proposed framework and methodology

The methodological framework outlines the process for developing and evaluating machine learning (ML) models to estimate daily and monthly PM_{10} concentrations across three diverse Saudi cities, defining the conceptual approach, data inputs, model selection criteria, and evaluation strategy. The research strategy rests on four principles: (1) PM concentrations are driven by meteorological and atmospheric factors [41]; (2) time-series chronological integrity is essential [42]; (3) ensemble methods offer synergistic strengths [43, 44]; and (4) workflows must be transparent and reproducible [45]. Figure 1 outlines the three-stage framework (Input, Development, Output), and the details of each stage are presented in the following subsections.

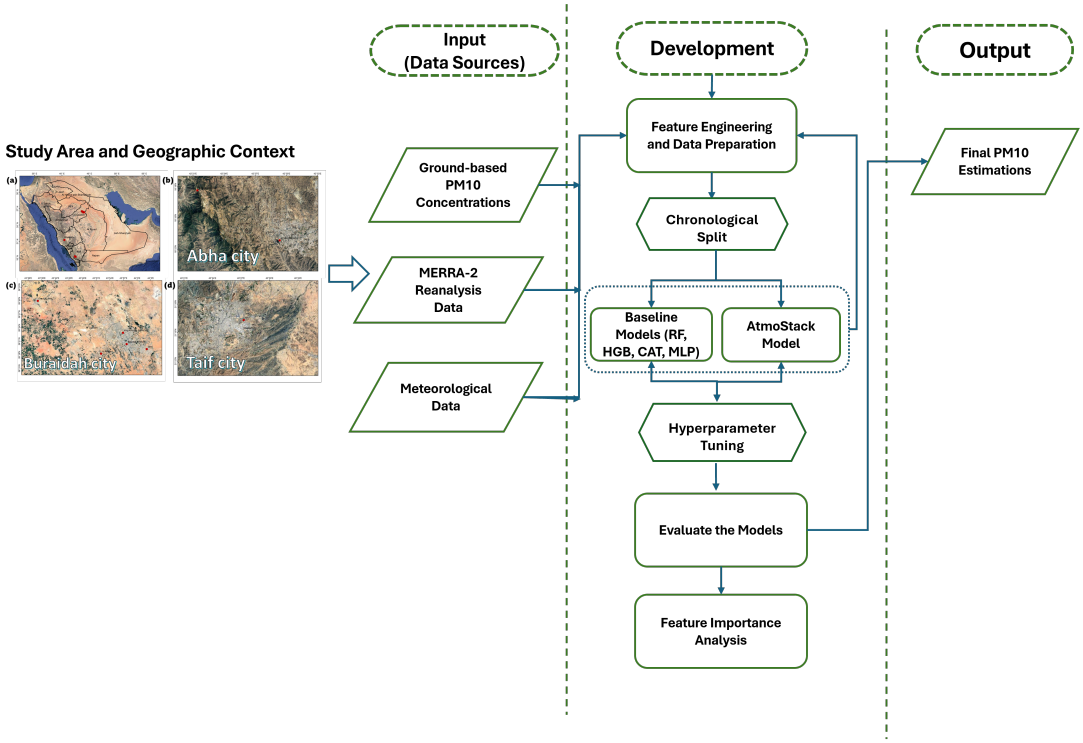

Figure 1. Data flow and model development framework.

3.1. Study area and geographic context

Saudi Arabia is an appropriate testbed for PM_{10} modeling. Its diverse topography and arid climate—characterized by high temperatures, sparse precipitation, and frequent dust storms—create varied microclimates and naturally elevated PM concentrations [46–49]. Three cities are selected to test model generalization across these environmental gradients: (1) Abha, in the Asir Mountains (mountainous, cooler, higher rainfall) [50]; (2) Buraidah, in the central Najd plains (arid, hot, low humidity) [51]; and (3) Taif, on the western escarpment (moderate elevation-based climate) [52].

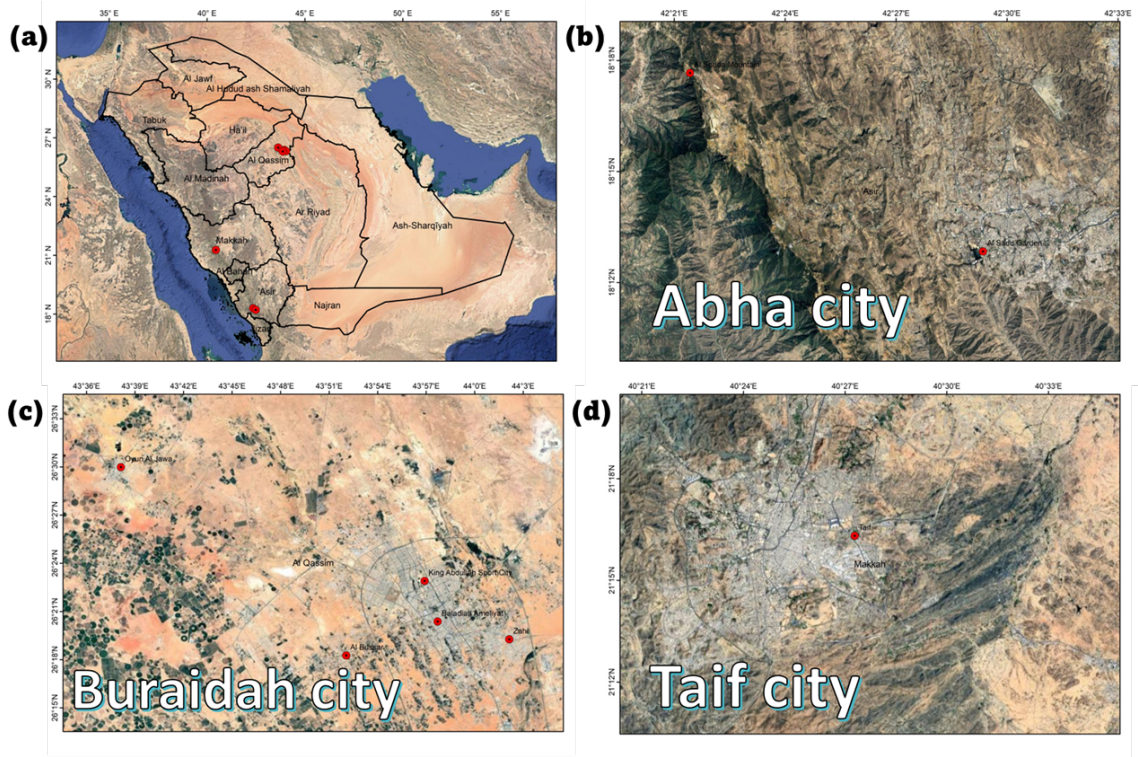

Figure 2. The study area comprises the three selected cities in Saudi Arabia. Red dots indicate the locations of ground-based monitoring stations recording daily PM_{10} concentrations in Abha, Buraidah, and Taif.

3.2. Data sources and acquisition

3.2.1. Ground-based PM_{10} measurement data

Daily PM_{10} concentration measurements are formally obtained from automated ground-level monitoring stations managed by the National Center for Environmental Compliance (NCEC) in Jeddah, Saudi Arabia. These stations employ standardized air quality monitoring instruments and follow strict data quality assurance procedures, ensuring consistency among collection sites. The records span the period from 2019 to 2023, covering five full calendar years of continuous environmental monitoring. Daily mean PM_{10} concentrations are calculated from intra-day measurements according to standardized EPA protocols for continuous monitoring networks. These ground-based observations serve as the ground-truth target variable against which all machine learning model estimations are subsequently compared and evaluated.

3.2.2. MERRA-2 satellite reanalysis data

The Modern-Era Retrospective Analysis for Research and Applications, Version 2 (MERRA-2), a NASA atmospheric reanalysis product, is provided at a horizontal resolution of approximately $0.5° \times 0.625°$ (roughly 50–70 km per grid cell), across 72 vertical levels from the surface up to 0.01 hPa, with temporal resolutions ranging from hourly to daily aggregated values.

These characteristics make MERRA-2 suitable for regional-scale atmospheric studies and PM_{10}

estimation when combined with ground-based measurements. MERRA-2 does not provide PM_{10} directly, but offers individual aerosol mass mixing ratios. These components include sulfate (SO_4), organic carbon (OC), black carbon (BC), mineral dust (DU) resolved into different size bins (DU_{001}–DU_{004})), and sea salt (SS) resolved into four bins (SS_{001}–SS_{004}). To estimate daily PM_{10} concentrations, this study implemented the computational methodology provided by NASA via its EarthData forum platform (https://forum.earthdata.nasa.gov). This approach utilizes chemically informed weighting factors to integrate constituent aerosol masses into a physically representative PM_{10} estimate, expressed as:

$$\begin{aligned} PM_{10} = \big(&1.375 \cdot SO_4 + BC_{phobic} + BC_{philic} + OC_{phobic} + OC_{philic} + DU_{001} + DU_{002} \\ &+ DU_{003} + 0.74 \cdot DU_{004} + SS_{001} + SS_{002} + SS_{003} + SS_{004}\big) \cdot AIRDENS \end{aligned} \tag{3.1}$$

The weighting coefficient of 1.375 applied to sulfate converts the sulfate mass (SO_4^{2-}) to the corresponding mass of ammonium sulfate, consistent with standard aerosol mass reconstruction practices and accounting for the contribution of associated ammonium. For mineral dust, while the first three bins (DU_{001}–DU_{003}) fall entirely within the PM10 size range, a coefficient of 0.74 is applied to the coarsest bin (DU_{004}) to retain only the fraction of particles that fall below the $10-\mu m$ aerodynamic diameter threshold. Finally, the air density (AIRDENS) factor is used to convert the aerosol mass mixing ratios (kg kg^{-1}) to ambient mass concentrations (μg m^{-3}) by accounting for local air density. This formulation represents the operational protocol for PM_{10} reconstruction from MERRA-2 component data and was consistently applied across all study cities to maintain methodological comparability [17, 53, 54].

3.2.3. Meteorological variables

Meteorological variables characterizing surface atmospheric conditions are retrieved from the Visual Crossing Weather API, a commercial data service providing high-resolution weather observations and derived products. This API synthesizes observations from diverse sources, including surface weather stations, airport observations, and radar-derived estimates, into spatially distributed weather products. The meteorological data retrieved for this study include: Wind governs transport and dispersion; humidity affects hygroscopic growth; temperature impacts stability and mixing depth; and visibility/cloud cover relates to particle concentration and radiation. Integrating them enables models to capture these complex physical relationships [41].

It is important to acknowledge the spatial scale mismatch between data modalities. While MERRA-2 data represent spatial averages over grid cells of approximately $0.5^\circ \times 0.625^\circ$, ground-based PM_{10} and meteorological observations correspond to point measurements at specific monitoring stations. To mitigate this discrepancy, ground-based measurements located within the same MERRA-2 grid cell are aggregated (e.g., averaged) to match the spatial resolution of the reanalysis data. Additionally, the machine learning models are capable of capturing systematic relationships between coarse-resolution reanalysis data and high-resolution ground observations, thereby partially addressing sub-grid variability [17].

3.3. Feature engineering and data preprocessing

The initial step in preparing the datasets involves comprehensive data preprocessing to ensure reliability and consistency across all cities. This includes handling missing values with variable-specific imputation strategies, such as mean substitution, removing anomalous measurements, and pruning irrelevant or low-variance columns. Temporal information is standardized by converting date strings into a suitable datetime format, and data are chronologically ordered to maintain time-series integrity. These preprocessing steps establish a clean and consistent foundation for modeling PM_{10} concentrations.

Following preprocessing, feature engineering transforms the cleaned data into representations suitable for machine learning. Calendar-based features capture recurring temporal patterns, while autoregressive features encode recent PM_{10} history through lagged values. Rolling statistics over seven-day windows quantify short-term variability and trends [55]. Categorical variables, such as city, location, and weather conditions, are converted into numerical representations via one-hot encoding. Careful attention to temporal ordering ensures that no future information leaks into the training data, preserving the integrity of model evaluation [42]. Collectively, these preprocessing and feature engineering steps enable robust and reproducible machine learning model development. The precise computational workflow for these transformations is provided in Section 4.3.

3.4. Model selection strategy

This study employs a multi-model comparison framework to explore various machine learning architectures to achieve accurate and reliable estimation of PM_{10} concentrations. The objective is to identify which modeling paradigms perform best under varying environmental and temporal conditions.

3.4.1. Individual baseline models

Tree-based models

Tree-based ensemble methods are chosen as primary candidates due to their proven effectiveness in regression tasks that involve mixed feature types, temporal dependencies, and nonlinear relationships [56]. The selected methods include RF, which is an ensemble learning method robust to overfitting. It averages outputs from multiple trees built using bootstrap aggregation (bagging) and random feature subsampling to induce diversity [57, 58] . HGB is an efficient gradient boosting implementation that bins continuous features into histograms, significantly reducing computation time and memory usage [59, 60] . CatBoost (CAT) is a gradient boosting framework with algorithmic innovations, such as ordered target encoding and permutation-invariant splitting, that improve generalization [61, 62].

Deep learning baseline model

A multi-layer perceptron (MLP) is included as a deep learning baseline. Unlike tree models, the MLP uses fully connected layers and nonlinear activation functions to learn representations and capture complex relationships [63, 64].

3.4.2. AtmoStack generalization model

A stacked generalization architecture, AtmoStack, is developed to synthesize base model strengths [44] (Figure 3). This two-level framework uses RF, HGB, and CatBoost as "level-0" learners to generate out-of-fold estimations. An ElasticNetCV regressor acts as the "level-1" meta-learner, taking level-0 estimations plus the original features ("feature passthrough" [45]) as input. ElasticNet combines L1/L2 regularization, allowing it to assign optimal weights while remaining stable against correlated base model outputs [65]. The AtmoStack architecture is shown in Figure 3, illustrating the hierarchical structure in which base learners feed their estimations to the meta-learner.

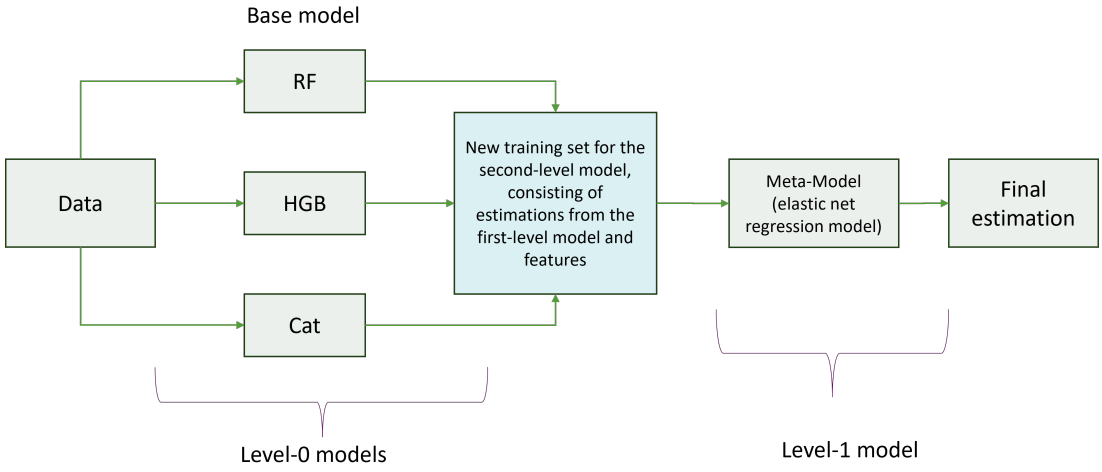

Figure 3. AtmoStack generalization model structure.

3.5. Hyperparameter optimization framework

Hyperparameter optimization is conducted using systematic search methodologies appropriate to each model architecture. Rather than manual tuning, automated search procedures evaluate parameter combinations within predefined ranges and select configurations that maximize performance on validation data [66]. Cross-validation during hyperparameter search ensures that estimates of generalization performance are unbiased and not inflated by test set evaluation [43]. For time-series data, specialized time-series cross-validation strategies are employed to maintain chronological integrity. This ensures that validation sets are always chronologically subsequent to training sets [42].

3.6. Training and evaluation paradigm

The validation strategy is critical for time series. Standard random partitioning (e.g., k-fold) is methodologically catastrophic, causing severe data leakage from temporal contamination and overestimating performance [42]. Therefore, a strict chronological validation strategy is adopted. The data are split into an initial 80% (training) and a final 20% (hold-out test set). The 80% training set is then subdivided for hyperparameter optimization using a *TimeSeriesSplit* procedure. This rigorous temporal separation prevents optimistic bias and ensures an unbiased estimate of generalization performance on novel future data [42, 43].

3.7. Performance evaluation metrics

Model performance is assessed using multiple complementary metrics: The coefficient of determination (R^2) (variance explained), root mean square error (RMSE) (emphasizes large errors), and mean absolute error (MAE) (linear weighting of errors). Collectively, these metrics enable a holistic assessment of model performance that takes into account different perspectives on error severity and bias [23].

3.8. Feature importance analysis

To provide a robust interpretation of the factors driving PM_{10} estimates, feature importance is assessed using a dual-methodological framework tailored to the specific architecture of each learner [67]. For tree-based models (RF, HGB, and CatBoost), built-in importance measures are utilized to capture the direct contribution of each variable during the training phase. Specifically, random forest (RF) importance is quantified via the mean decrease in impurity (MDI), which measures the total reduction in MSE attributed to a feature across all trees [57]. Histogram-based gradient boosting (HGB) utilizes the cumulative loss reduction across boosting iterations [60], while CatBoost employs the PredictionValuesChange method, which calculates the average change in estimations when a feature value is altered [68].

For the MLP and the AtmoStack ensemble, which lack native importance attributes, a model-agnostic permutation importance approach is implemented [69, 70]. This technique measures feature influence by calculating the decline in model performance (increase in RMSE) when the values of a single feature are randomly shuffled, thereby breaking the relationship between the feature and the target variable. To ensure stability and prevent data leakage, this is performed on the held-out test set with multiple iterations: n_repeats = 5 for the MLP and n_repeats = 10 for the AtmoStack ensemble, reflecting the higher complexity of the stacking architecture. This unified framework ensures that feature influence is evaluated consistently across both simple and ensemble models.

4. Implementation

The implementation of the proposed framework is described in terms of the computational environment, software dependencies, and model configurations. The pipeline uniformly implements all models (RF, HGB, CatBoost, MLP, and AtmoStack) to ensure consistent preprocessing, training, and validation.

4.1. Computational environment and software infrastructure

The workflow is implemented in Python 3.11. Core libraries included pandas (2.2) for data manipulation [71], scikit-learn for machine learning [45], TensorFlow (2.16) for neural networks [72], CatBoost (1.2) [61], and matplotlib (3.8) for visualization [73]. To ensure reproducibility, a global random seed of 42 is fixed across NumPy, scikit-learn, and TensorFlow. The pipeline defaults to CPU execution if a GPU is not detected, ensuring hardware compatibility.

4.2. Input data structure and modeling assumptions

Input data consists of three CSV files, one file provided for each city under investigation: abha_data.csv, buraidah_data.csv, and taif_data.csv. These files contain spatially and temporally matched daily records from ground stations, MERRA-2, and meteorological observations. A second set of datasets is created for all cities, identical to the first but with all meteorological variables excluded. This allows for a direct comparison to quantify the performance contribution of meteorological data. The pipeline is robust to variations in input columns. It uses a predefined desired_cols list and models only the intersection of these required columns with those present in the CSV file. Absent required columns are silently skipped, enabling flexibility.

4.3. Data preprocessing and feature engineering

The preprocessing and feature engineering workflow represents a critical step to prepare raw data for accurate PM_{10} modeling. All input data are processed using a unified preprocess (...) function, applied consistently across all cities to ensure methodological reproducibility. Figure 4 summarizes the complete end-to-end data cleaning and feature-engineering pipeline implemented in this study. The pipeline begins with deliberate column selection, retaining only features relevant to PM_{10} estimation, while redundant or low-variance columns (e.g., Unnamed: 0, Heat Index, Cloud Cover, Weather Type) are pruned. Missing values are imputed using variable-specific strategies [74], such as mean substitution for Sea Level Pressure and conversion of Datestrings to datetime64 objects for temporal operations. From these, calendar-based features (day, month, weekday) and a categorical season feature are derived to capture temporal patterns, enhancing model performance [42]. Categorical variables (City, Location, Region, Conditions) are transformed via one-hot encoding to avoid ordinal assumptions [43]. Subsequently, data are chronologically sorted by date for each city, and autoregressive features (PM10_lag1–PM10_lag7) along with rolling statistics (mean, standard deviation, min, max) over seven-day windows are generated to capture short-term PM_{10} dynamics. The final step involves strict quality assurance, removing any remaining incomplete records to ensure a fully consistent and reliable dataset for model training, with cities exhibiting excessive data loss excluded via a None sentinel.

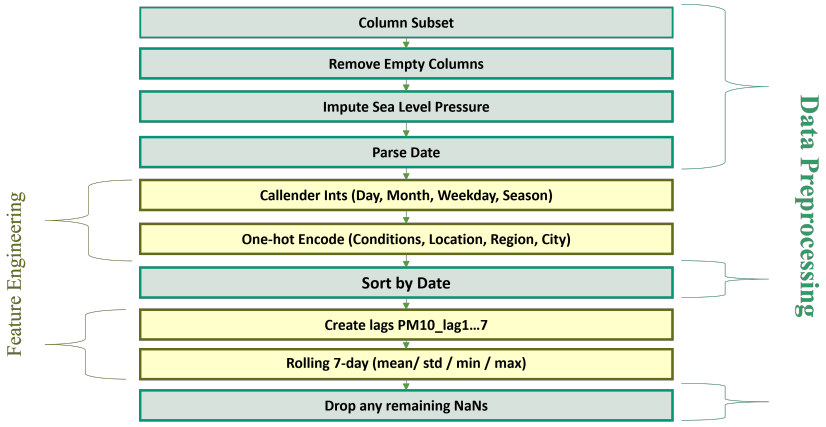

Figure 4. End-to-end data cleaning and feature-engineering pipeline implemented in the study.

4.4. Chronological train-test split implementation

To rigorously evaluate the generalization performance of machine learning models on truly unseen data, a strict chronological train-test split is employed for each city's dataset. The initial 80% of chronologically sorted data served as the training set. The final 20% (val_split = 0.2) of the chronologically sorted data constitutes the hold-out test set, as depicted in Figure 5. The reserved 20% test set remains completely untouched during any model development, hyperparameter tuning, or validation procedures, ensuring that it provides an unbiased estimate of generalization performance on truly new future data [43].

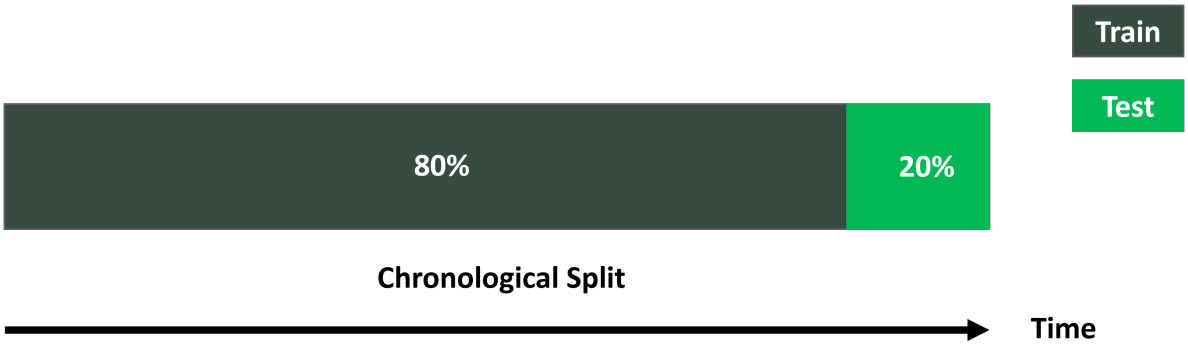

Figure 5. Chronological train-test split.

4.5. Tree-based baseline configuration and training implementation

Tree-based ensembles, including RF, HGB, and CatBoost, are trained on each city's dataset using a unified preprocessing and transformation pipeline. This pipeline standardizes all features (StandardScaler) and applies a variance-stabilizing transformation to the target variable [42]:

$$y' = \sqrt{\log(1 + y)} \tag{4.1}$$

The model is trained on the transformed target y', and estimations are converted back to the original PM_{10} scale via the inverse transformation [42]:

$$\hat{y} = \exp((y')^2) - 1 \tag{4.2}$$

Hyperparameter optimization is performed via randomized search (RandomizedSearchCV) with four iterations per model, using time-series cross-validation (TimeSeriesSplit, 4 folds). The coefficient of determination (R^2) is used to select the best configuration, summarized in Table 1. The best selected hyperparameters are as follows: RF (n_estimators=600, max_depth=15, min_samples_leaf=2) in all cities, HGB (learning_rate=0.05, max_depth=10), and CatBoost (iterations=800, learning_rate=0.05, depth=6–8 depending on the city).

Table 1. Hyperparameter search space for each machine learning model.

Model	Parameter	Search Space	Selected Value
RF	n_estimators	{400, 500, 600}	{600}
	max_depth	{None, 15}	{15}
	min_samples_leaf	{1, 2}	{2}
HGB	learning_rate	{0.05, 0.1}	{0.05}
	max_depth	{None, 10}	{10}
CatBoost	iterations	800 (fixed)	{800}
	learning_rate	{0.05, 0.1}	{0.05}
	depth	{6, 8}	{6} (Abha, Taif), {8} (Buraidah)

4.6. Deep learning baseline implementation

A feedforward multi-layer perceptron (MLP) is implemented as a fixed baseline across all cities. The network consists of three fully connected layers with decreasing neuron counts (256, 128, 64) and ReLU activations, followed by a single output neuron. Batch normalization [75] and dropout (0.35, 0.25, 0.15) are applied after each hidden layer. The model is trained to minimize mean squared error using the Adam optimizer (learning rate 10^{-3}), with `ReduceLROnPlateau` and `EarlyStopping` callbacks controlling the learning process [76]. Training uses a maximum of 400 epochs, a mini-batch size of 64, and a 10% internal validation rate. No hyperparameter tuning or cross-validation is applied.

4.7. AtmoStack ensemble implementation

The AtmoStack ensemble is implemented as a two-level stacked architecture using scikit-learn's StackingRegressor. Base learners include RF, HGB, and CatBoost, while an ElasticNetCV serves as the meta-learner. Internal cross-validation generates out-of-fold estimations from the base learners, forming an unbiased feature matrix for the meta-learner. The ElasticNetCV scans l1_ratio values of 0.1, 0.5, and 0.9, and passthrough=True provides the meta-learner access to both base estimations and original features [45]. Finally, the ensemble is re-fitted on the full 80% training set and evaluated on the 20% hold-out test set.

4.8. Feature importance execution

The assessment of feature importance is executed within the Python-based experimental pipeline using the parameters defined in Section 3.8. For the tree-based algorithms, importance scores are extracted post-training via the feature_importances_ attribute provided by the scikit-learn and CatBoost libraries.

For the AtmoStack ensemble and MLP, the permutation importance is implemented using the permutation_importance function from the sklearn.inspection module. To maintain computational efficiency while ensuring statistical stability, the process is parallelized across CPU cores. The resulting importance scores are then normalized to a scale of 0 to 1 to facilitate a direct cross-model comparison of the environmental drivers. These execution steps ensure that the interpretability of the ensemble model is grounded in its out-of-sample predictive sensitivity, as visualized in the subsequent results section.

4.9. Model evaluation metrics

All models are evaluated on the reserved 20% hold-out test set [43] to ensure unbiased comparison. Three complementary regression metrics are used to capture different aspects of estimate performance. The coefficient of determination (R^2) measures the proportion of variance in PM_{10} explained by the model, ranging from negative infinity (worse than a mean estimation) to 1.0 for a perfect fit. The root mean square error (RMSE) quantifies the average magnitude of estimation errors while emphasizing larger deviations, and is computed as

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{4.3}$$

where y_i and $\hat{y}_i$ are observed and estimated PM_{10} values, respectively, and n is the number of test samples.

The mean absolute error (MAE) measures the average absolute difference between estimations and observations:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{4.4}$$

Unlike RMSE, MAE applies equal weighting to all deviations, making it less sensitive to outliers. It is also expressed in $\mu g/m^3$ and offers a complementary perspective on overall estimation accuracy [42]. Collectively, these metrics provide a comprehensive assessment framework, capturing explained variance, sensitivity to large errors, and typical estimation magnitudes [42]. This implementation pipeline has detailed the rigorous and reproducible setup for training all models and provides the necessary foundation for the empirical evaluation of the models' performance.

5. Experimental evaluation and results

The empirical results derive from a comprehensive evaluation of five machine learning models developed for PM_{10} estimation: RF, HGB, CAT, MLP, and the stacking ensemble (AtmoStack)—using both the integrated dataset and a version without meteorological data. The performance of these models is rigorously assessed on a strictly held-out test set, comprising the final 20% of the time series for each of the three study cities. This chronological split ensures that the evaluation mimics a real-world forecasting scenario, providing a valid measure of each model's generalization capability on unseen future data.

5.1. Empirical results

5.1.1. Overview of evaluation results on integrated datasets

Model performance is evaluated using R^2, RMSE, and MAE for both daily and monthly estimations across all cities (as illustrated in Table 2). The daily estimations provide detailed insight into short-term accuracy, while the monthly estimations, calculated by aggregating the daily estimations, offer a perspective on the models' ability to capture longer-term pollution trends. The Δ RMSE column indicates the percentage improvement of AtmoStack and MLP relative to the best-performing single-tree model (RF, HGB, or CAT) for each city.

Table 2. Performance of all models on the 20% hold-out horizon for both daily and monthly estimates using integrated datasets. Δ RMSE is relative to the best single-tree model (RF, HGB , or CAT) in each city.

City	Model	Daily Estimations				Monthly Estimations			
		R^2	RMSE	MAE	Δ RMSE	R^2	RMSE	MAE	Δ RMSE
	MLP	0.442	37.33	24.08	—	0.39	39.01	25.42	—
	RF	0.556	33.28	20.68	—	0.61	16.10	10.71	—
Abha	HGB	0.558	33.22	21.84	—	0.52	17.75	11.53	—
	CatBoost	0.538	33.97	21.72	—	0.60	16.28	10.50	—
	AtmoStack	0.578	32.45	21.04	–2.32%	0.71	13.95	10.75	–13.35%
	MLP	0.379	113.76	64.22	—	0.89	31.61	25.81	—
	RF	0.683	81.28	54.66	—	0.85	36.92	28.78	—
Buraidah	HGB	0.670	82.87	54.19	—	0.86	35.85	27.18	—
	CatBoost	0.590	92.38	58.97	—	0.76	47.66	35.06	—
	AtmoStack	0.728	75.20	50.24	–7.48%	0.96	18.96	16.04	–47%
	MLP	-1.23×10^3	3374.67	526.96	—	-248	664	366.44	—
	RF	0.569	63.27	26.32	—	0.85	16.17	10.93	—
Taif	HGB	0.521	66.71	28.17	—	0.918	12.06	9.08	—
	CatBoost	0.544	65.06	26.91	—	0.89	13.92	8.72	—
	AtmoStack	0.630	58.60	31.68	–7.38%	0.94	9.90	8.78	–17.91%

Performance in Abha

In the relatively mild and mountainous city of Abha, single-model daily performance is comparable, with HGB emerging as the best individual learner (RMSE = 33.22 μg/m^{-3}), slightly outperforming RF (RMSE = 33.28 μg/m^{-3}). The AtmoStack ensemble achieved the strongest daily results overall (R^2 = 0.578, RMSE = 32.45 μg/m^{-3}), representing a 2.3% RMSE improvement over HGB, while the MLP baseline underperformed markedly (RMSE = 37.33 μg/m^{-3}). At the monthly scale, RF became the best-performing single model (R^2 = 0.61, RMSE = 16.10 μg/m^{-3}), as temporal aggregation reduced noise and enhanced model stability. AtmoStack again outperformed all single learners, achieving an R^2 = 0.71, RMSE = 13.95 μg/m^{-3}, corresponding to a 13.3% RMSE reduction relative to RF, confirming its ability to capture broader pollution trends.

Performance in Buraidah

In the arid, dust-prone environment of Buraidah, RF is the best daily model, achieving an R^2 of 0.683 and an RMSE of 81.28 μg/m^{-3}, while AtmoStack achieved the best overall performance (R^2 = 0.728), yielding a 7.5% RMSE improvement over RF. At the monthly scale, temporal aggregation improved performance across all models. Notably, AtmoStack achieved an outstanding accuracy R^2 of 0.96 and an RMSE of 18.96 μg/m^{-3}, corresponding to a 47% RMSE reduction relative to the best single monthly model (HGB). The MLP, which performed poorly at the daily level before aggregation (R^2 = 0.39), also benefited substantially, improving to an R^2 of 0.89 and an RMSE of 31.61 μg/m^{-3}. This highlights that temporal smoothing stabilizes deep learning–based estimators in highly variable desert environments.

Performance in Taif

In Taif, RF is the strongest daily single learner (R^2 = 0.56), whereas the MLP collapsed due to severe overfitting ($R^2 \approx -1.225 \times 10^3$, RMSE = 3300 μg/m^{-3}). AtmoStack again delivered superior daily performance (R^2 = 0.63, RMSE = 58.60 μg/m^{-3}), representing a 7.4% RMSE improvement over RF. After aggregation to monthly estimations, the models' performance improved substantially, particularly among tree-based learners. The best-performing single monthly model is HGB, and AtmoStack exceeded it, achieving the highest accuracy across all cities (R^2 = 0.94, RMSE = 9.90 μg/m^{-3}), corresponding to a 17.9% RMSE reduction relative to HGB. These results confirm the ensemble's ability to capture stable long-term pollution patterns in noisy desert environments.

To investigate the drivers of AtmoStack's superior performance, an ablation analysis was conducted to determine whether the results stem from the collective synergy of all base learners (RF, HGB, and CatBoost) or the dominance of a specific model. The findings, detailed in the Supplementary Material (Section S1), underscore the ensemble's stability: The exclusion of any individual learner resulted in only marginal performance fluctuations, with all improvement or degradation ratios remaining consistently below the adopted 10% significance threshold. This confirms that AtmoStack's robustness is a collective outcome of its integrated architecture rather than reliance on a single learner.

5.1.2. Practical implications for air quality management

The AtmoStack ensemble consistently reduced RMSE by 2%–7.5% for daily and 13%–47% for monthly estimations compared to the strongest individual models, demonstrating clear benefits for operational air quality management. These improvements enhance the accuracy of pollution estimations, reducing both false negatives and false positives near regulatory thresholds. With minimal additional computational cost, the stacking approach provides a cost-effective strategy for deployment by environmental agencies. Figures 6 and 7 illustrate these RMSE reductions across the three cities for daily and monthly estimations, respectively.

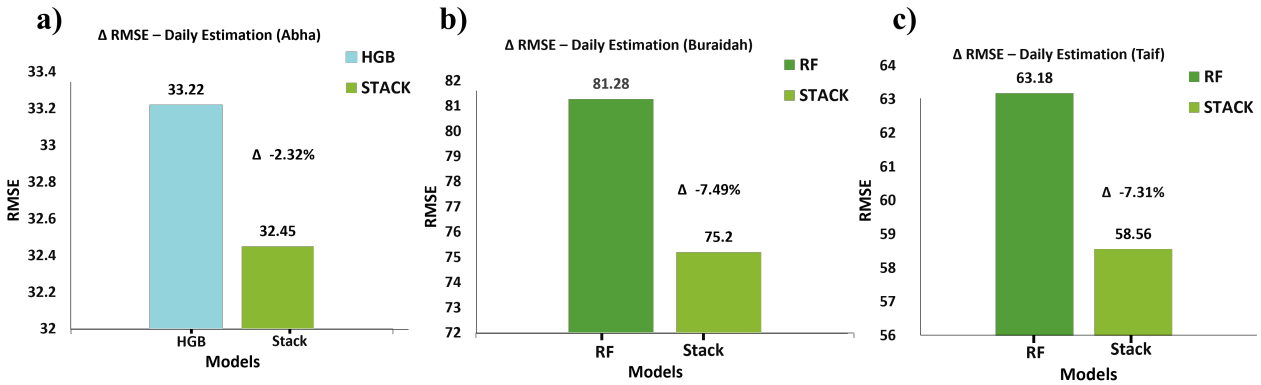

Figure 6. Percentage RMSE improvement of AtmoStack over the best single models in (a) Abha, (b) Buraidah, and (c) Taif for daily estimations.

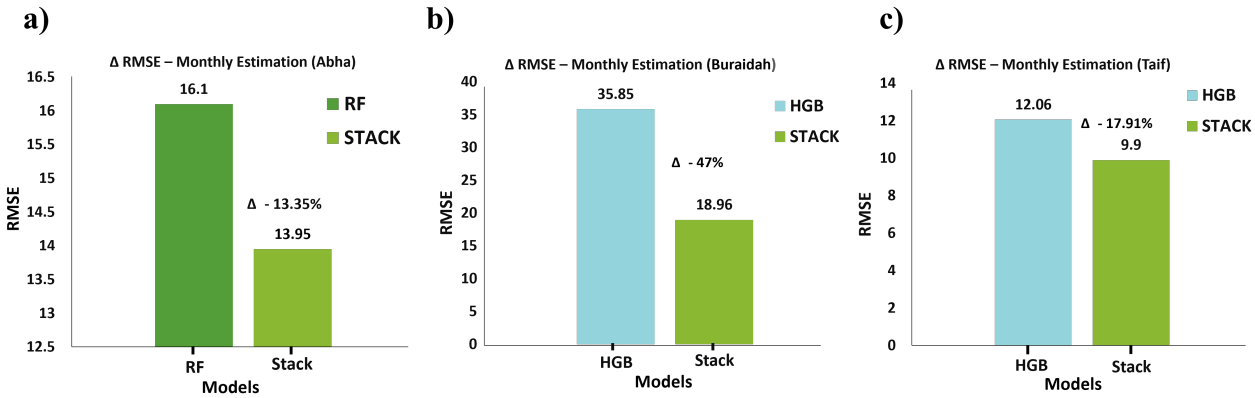

Figure 7. Percentage RMSE improvement of AtmoStack over the best single models in (a) Abha, (b) Buraidah, and (c) Taif for monthly estimations.

5.1.3. Effect of excluding meteorological data on model performance

The influence of meteorological features on daily PM_{10} estimation is examined by comparing models trained with and without meteorological variables. As summarized in Table 3, integrating meteorological data consistently enhanced model accuracy across all cities.

Table 3. Comparison of model performance on the 20% hold-out horizon for daily PM_{10} estimation using integrated datasets versus datasets without meteorological variables. ΔRMSE is relative to the best single-tree model (RF, HGB, or CAT) in each city.

City	Model	Integrated Data				Data without Meteorological Data			
		R^2	RMSE	MAE	Δ RMSE	R^2	RMSE	MAE	Δ RMSE
Abha	MLP	0.44	37.33	24	-	–3.7	109	48	-
	RF	0.56	33.28	21	-	0.51	35.09	22.19	-
	HGB	0.56	33.22	22	-	0.50	35.25	22.35	-
	CatBoost	0.54	33.97	22	-	0.36	39.73	23.4	-
	AtmoStack	0.57	32.45	21	–2.32%	0.53	34.27	21.95	–2.33%
Buraidah	MLP	0.38	113.76	64.22	-	–0.27	162.79	83.83	-
	RF	0.68	81.28	54.66	-	0.56	94.71	57.99	-
	HGB	0.67	82.87	54.19	-	0.55	96.82	58.49	-
	CatBoost	0.59	92.38	58.97	-	0.53	98.18	60.67	-
	AtmoStack	0.72	75.20	50.24	–7.48%	0.63	87.68	56.05	–7.42%
Taif	MLP	-1.23×10^3	3374.67	526	–	–56.24	727	167	–
	RF	0.57	63.27	26.32	–	0.49	68.34	30.17	–
	HGB	0.52	66.71	28.17	–	0.39	74.93	33.81	–
	CatBoost	0.54	65.06	26.91	–	0.46	70.53	31.33	–
	AtmoStack	0.63	58.60	31.68	–7.38%	0.49	68.65	33.81	+0.45%

In Abha, excluding meteorological variables caused a slight decline in performance, with AtmoStack's R^2 dropping from 0.57 to 0.53 and RMSE increasing from 32.45 to 34.27 μg/m^{-3}.

In Buraidah, the impact is more pronounced (R^2: 0.72 to 0.63; RMSE: 75.20 to 87.68 μg/m^{-3}), reflecting the strong dependence of PM_{10} levels on local atmospheric dynamics. The largest effect is observed in Taif, where R^2 decreased from 0.63 to 0.49, and RMSE rose from 58.60 to 68.65 μg/m^{-3}, underscoring the dominant role of weather conditions, especially visibility, humidity, and terrain—in shaping particulate dispersion.

Overall, these findings confirm that meteorological features substantially enhance estimation accuracy. While the AtmoStack ensemble remains relatively stable, all models perform better when temperature, relative humidity, and wind direction are included, highlighting the critical importance of meteorological data in air quality modeling.

5.1.4. Feature importance analysis

Feature importance analysis is conducted to identify the key variables driving PM_{10} estimations across the studied cities. To ensure a comprehensive evaluation, we employed a dual approach: Tree-based importance measures were utilized for RF (mean decrease in impurity), HGB (loss reduction), and CatBoost (PredictionValuesChange), while model-agnostic permutation importance was applied to the MLP and AtmoStack ensemble to handle their non-tree architectures. This framework allows for capturing consistent patterns and identifying the most influential predictors across different model families.

Feature importance for Abha

Across all models, PM10_lag1 and MERRA_PM10 emerged as dominant predictors, confirming the strong autoregressive and satellite-based influences as illustrated in Figure 8. Visibility also played a key role, especially in the HGB model, indicating nonlinear relationships between optical clarity and PM_{10} levels. The permutation results for MLP and AtmoStack aligned closely with tree-based findings, highlighting a stable consensus on these core predictors.

Feature importance for Buraidah

In the arid central region, short-term persistence (PM10_lag1) and visibility are again leading features as shown in Figure 9. While RF emphasized PM10_lag1 and rolling statistics, HGB and CatBoost ranked visibility highest, suggesting sensitivity to sudden dust-related changes. The AtmoStack ensemble consolidated these insights, identifying visibility and short lags as the most critical factors, reflecting the joint effect of persistence and atmospheric clarity in dust-dominated conditions.

Features importance for Taif

In Taif, where terrain and meteorological variability are significant, all models consistently relied on PM10_lag1, visibility, and MERRA_PM10 as presented in Figure 10. HGB and CatBoost assigned particularly high importance to visibility, while the MLP mirrored this pattern. The AtmoStack ensemble confirmed these dominant features, showing its ability to extract stable and physically meaningful relationships despite data complexity.

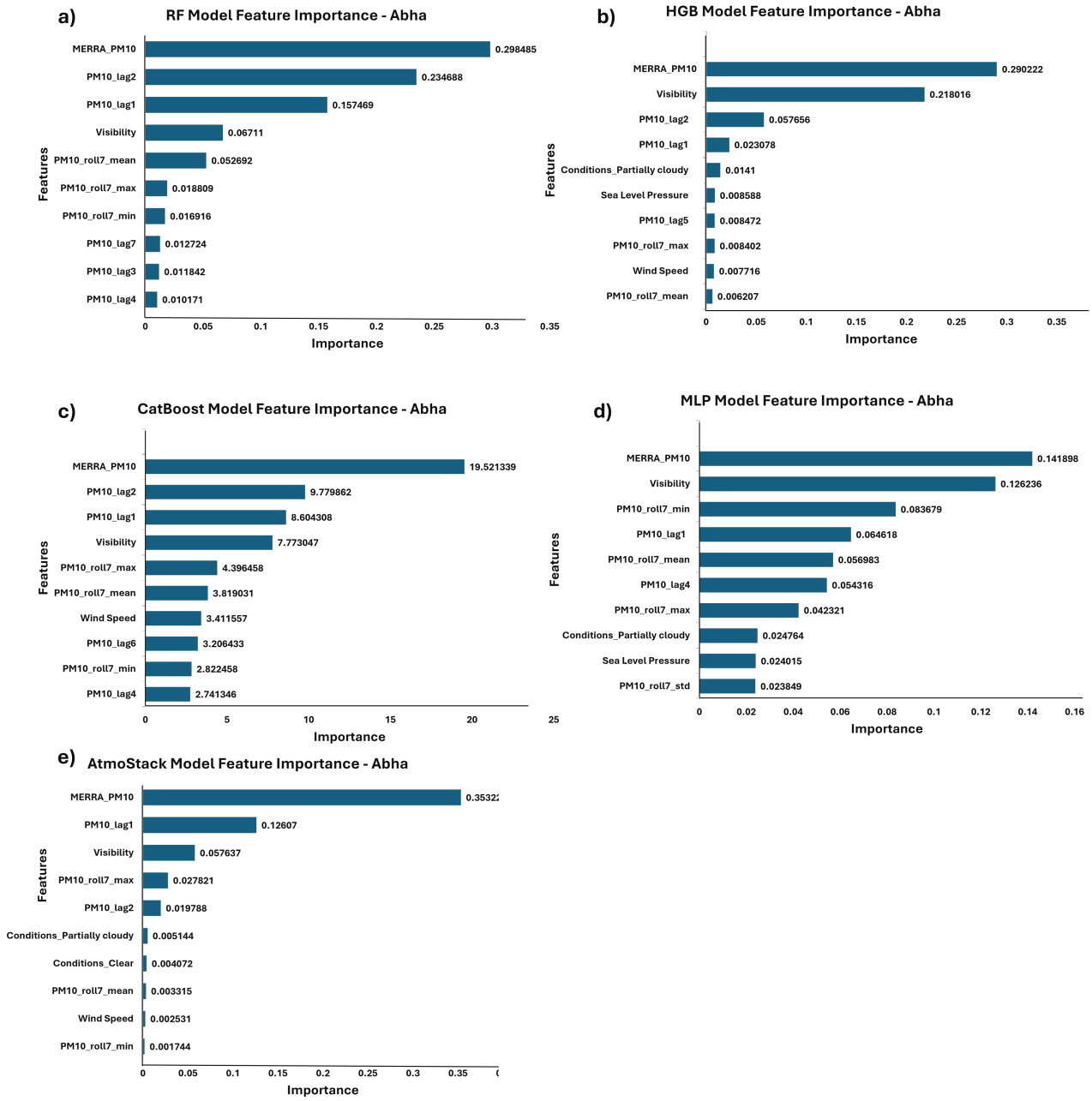

Figure 8. Feature importance for PM_{10} estimation in Abha across all models: (a) RF, (b) HGB, (c) CatBoost, (d) MLP, and (e) AtmoStack. The plots show the relative contribution of each feature to the model performance.

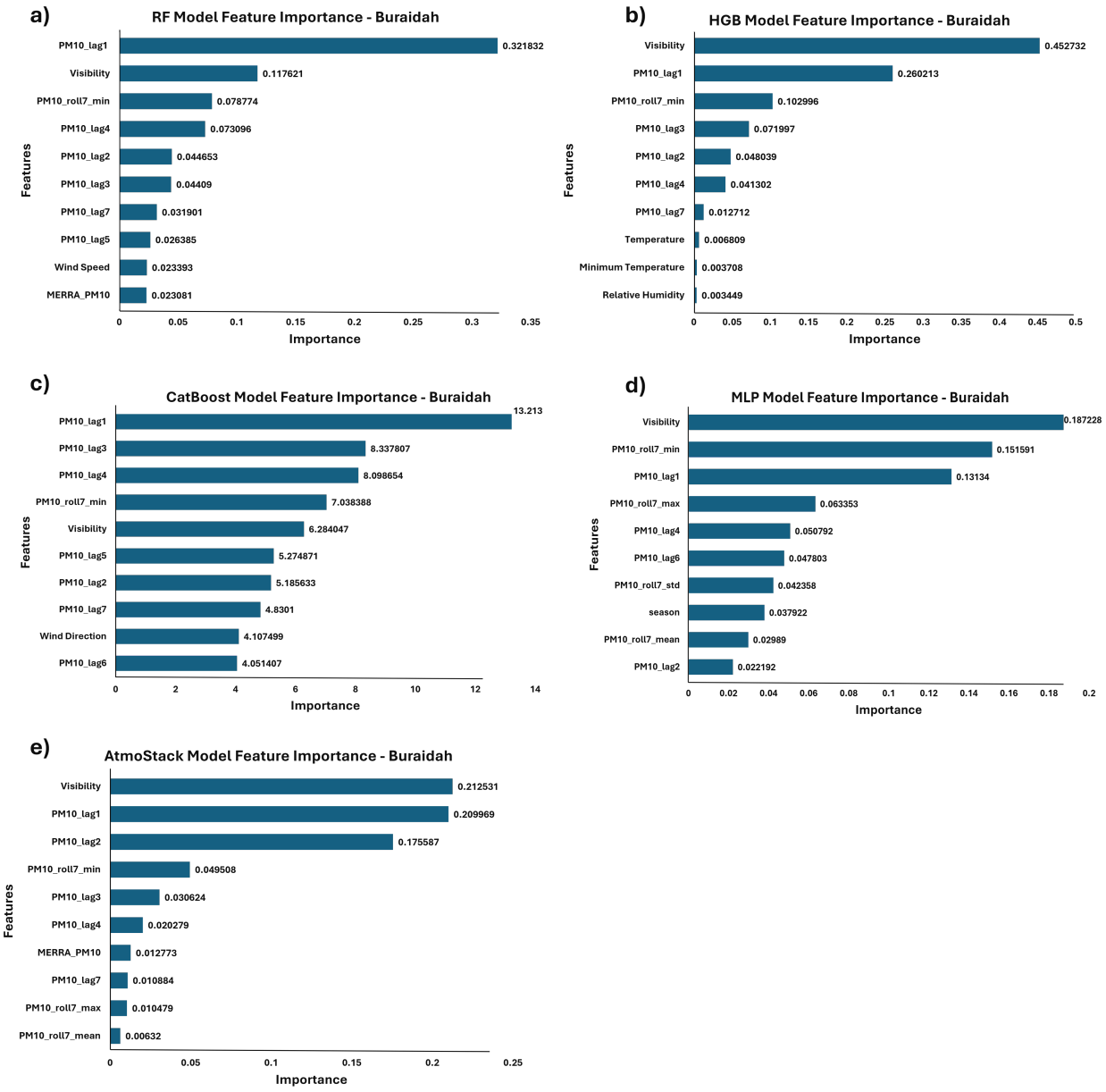

Figure 9. Feature importance for PM_{10} estimation in Buraidah across all models: (a) RF, (b) HGB, (c) CatBoost, (d) MLP, and (e) AtmoStack. The plots show the relative contribution of each feature to the model performance.

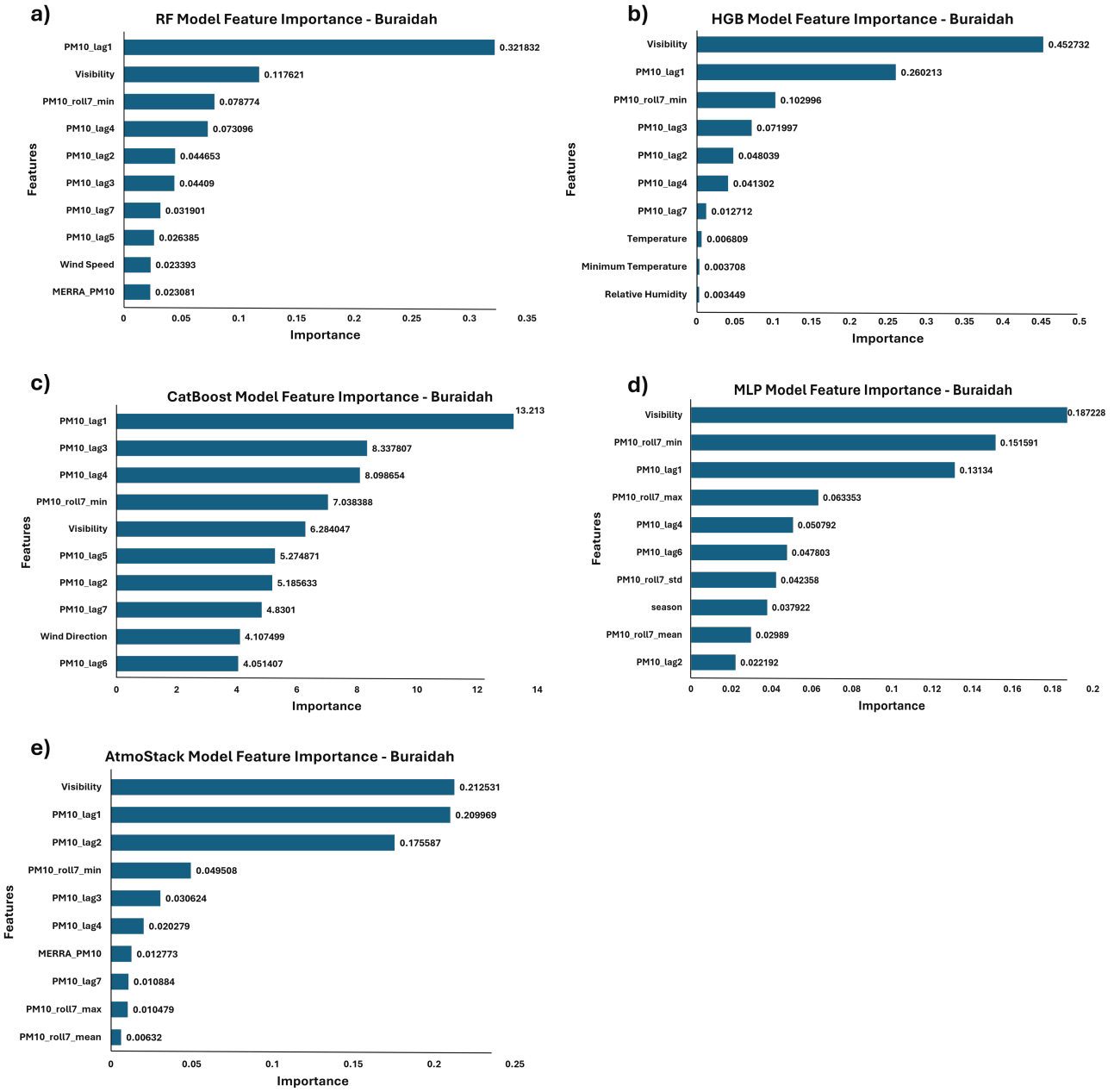

Figure 10. Feature importance for PM_{10} estimation in Taif across all models: (a) RF, (b) HGB, (c) CatBoost, (d) MLP, and (e) AtmoStack. The plots show the relative contribution of each feature to the model performance.

General patterns

Across all cities, PM10_lag1, visibility, and MERRA_PM10 are the most influential features, demonstrating that both recent pollution history and atmospheric clarity are key determinants of PM_{10} variability. Rolling metrics (e.g., PM10_roll7_mean, PM10_roll7_max) also contributed by capturing short-term accumulation trends. Tree-based models emphasized these factors with slight variations, whereas the AtmoStack ensemble leveraged them jointly, enhancing generalizability. These results

underline that temporal dependencies, visibility conditions, and reanalysis-based PM inputs form the core informational basis for accurate PM_{10} estimation across different Saudi environments.

5.2. Comparison with previous studies

To evaluate the robustness and regional relevance of the AtmoStack framework, a comprehensive two-phase benchmarking analysis is conducted, focusing on both daily and monthly estimation scales. Given the scarcity of PM_{10} modeling studies specifically targeting Saudi Arabian cities, prominent global models are selected and retrained on local datasets using the same hyperparameter configurations as in their original studies to ensure a standardized and rigorous comparison. The daily performance assessment (Figure 11) began with the light gradient boosting (LGB) model, which originally achieved near-perfect results in South Korea (R^2 = 0.99) [77]. When retrained with its original hyperparameters (max depth = 17, learning rate = 0.2, n_estimators = 4300) on Saudi datasets, its performance declined significantly, with R^2 dropping to 0.50–0.52 and RMSE increasing to 35–101 μg/m³. Similarly, the long short-term memory (LSTM) model, highly successful in India (R^2 = 0.99) [78], failed to generalize to the local arid conditions when retrained using its original setup (100 units, Adam optimizer, and 50 epochs), yielding R^2 values as low as −0.22 in Taif and RMSE values reaching 107 μg/m³. AtmoStack consistently demonstrated superior accuracy across all daily scenarios, achieving the highest R^2 (0.58–0.73) and the lowest error metrics (RMSE of 32, 75.2, and 58.56 μg/m³ for Abha, Buraidah, and Taif, respectively).

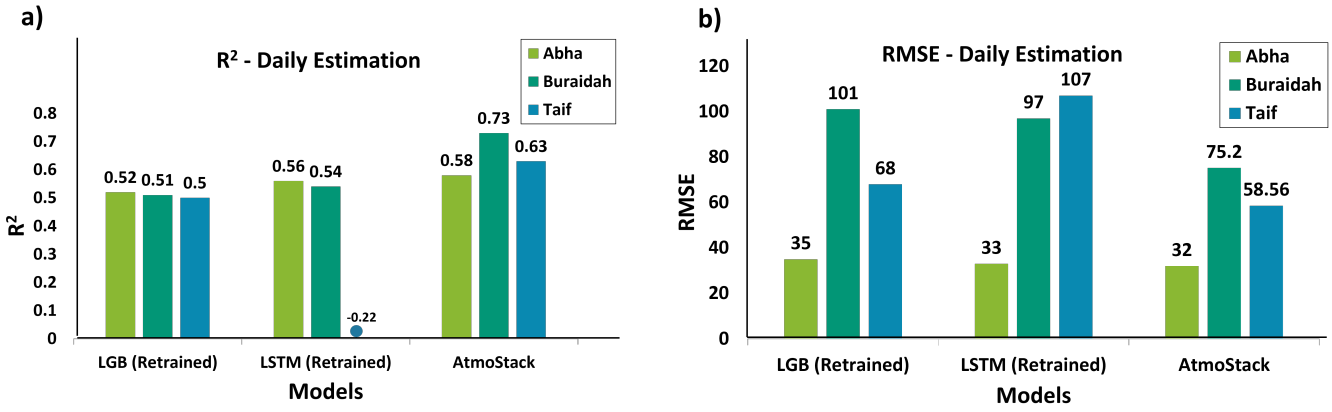

Figure 11. Comparative performance of retrained models (LGB and LSTM) against AtmoStack using (a) R^2 and (b) RMSE in Abha, Buraidah, and Taif for daily estimations.

Transitioning to the monthly scale (Figure 12), the framework is compared with an artificial neural network (ANN) model originally optimized for monthly PM_{10} estimation in Ankara, Turkey (R^2 = 0.58) [34]. When this ANN architecture (two hidden layers with 10 and 8 neurons) was retrained on local datasets, it showed competitive results, particularly in Abha and Taif, where it achieved R^2 values of 0.76 and 0.96, slightly higher than AtmoStack's 0.705 and 0.94. However, AtmoStack demonstrated superior error-reduction capabilities and better overall stability. In Buraidah, AtmoStack significantly outperformed the ANN, achieving a higher R^2 of 0.96 (vs. 0.93) and reducing the RMSE from 25 μg/m³ to 18 μg/m³. While the ANN captured strong correlations in some cities, the consistently lower RMSE in Buraidah and the equal RMSE in Abha (13 μg/m³) highlight AtmoStack's ability to minimize

absolute estimation errors. This suggests that while ANN is effective at capturing monthly trends (R^2), AtmoStack provides more reliable and precise concentrations across diverse regional conditions by effectively reconciling multi-source inputs.

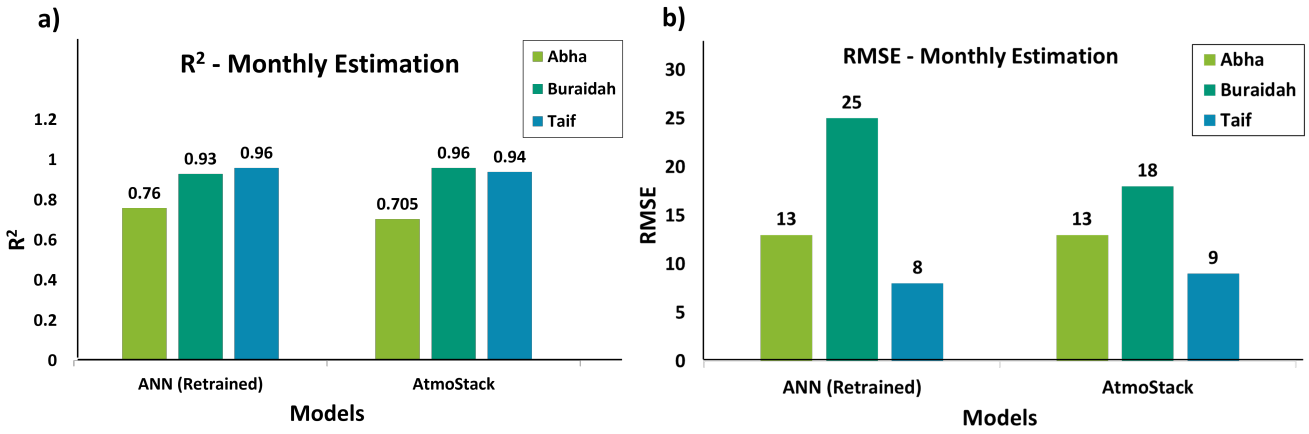

Figure 12. Comparative performance of retrained ANN against AtmoStack using (a) R^2 and (b) RMSE in Abha, Buraidah, and Taif for monthly estimations.

The superior performance of AtmoStack is mechanistically explained by its feature importance analysis, which identified MERRA-2 products, lagged PM_{10} (lag1–lag7), and visibility as the primary predictors. From a mechanistic standpoint, the high importance of visibility and MERRA-2 meteorological data reflects the direct physical impact of dust loading on atmospheric transparency; in Saudi Arabia, PM_{10} spikes are intrinsically coupled with a sharp decline in horizontal visibility. Furthermore, the strong reliance on lagged PM_{10} captures the temporal persistence of dust events, where high concentrations tend to linger over several days due to the region's unique atmospheric stability and stagnant air masses during dust episodes. The failure of the retrained models (LGB and LSTM) to generalize locally stems from their inability to correctly weigh the complex dependencies between visibility, satellite-derived meteorology, and temporal lags. In contrast, AtmoStack's stacking architecture effectively reconciles these variables by capturing the nonlinear interactions that single-learner models often overlook. This allows the framework to more accurately simulate the physical behavior of dust transport and accumulation, significantly reducing the large residuals typically found during extreme events.

6. Discussion

This section offers an in-depth analysis of the empirical results, extending beyond quantitative metrics to explain their implications across diverse climatic contexts. The discussion adheres strictly to the experimental framework, ensuring that all interpretations are grounded in the study's methodological design.

6.1. Impact of meteorological data integration on model performance

A key design feature of this study is the comparison between two complementary experimental configurations: One incorporating ground-based PM_{10} measurements, MERRA-2 reanalysis estimates, and meteorological variables, and another excluding meteorological information. This structure enabled a quantitative isolation of the contribution of meteorological data to estimation accuracy and model generalization.

Across all cities and model architectures, the inclusion of meteorological variables consistently resulted in lower RMSE and higher R^2 values. Typical performance improvements ranged from approximately 5% to 15%, with some city–model combinations exhibiting even larger gains. These improvements reflect the fundamental physical role of meteorological processes in controlling PM_{10} concentrations. Wind speed and direction regulate pollutant transport and dispersion, relative humidity influences particle growth through hygroscopic effects, temperature modulates atmospheric stability and mixing depth, and visibility provides an integrated measure of particulate optical extinction. These dynamic processes cannot be reliably inferred from lagged pollution history alone and are only partially captured by spatially coarse and temporally averaged reanalysis products such as MERRA-2.

The performance gains from meteorological integration are particularly pronounced in the dust-prone environments of Buraidah and Taif, where episodic dust events driven by wind fields and atmospheric instability dominate PM_{10} variability. In these regions, models without access to real-time meteorological information are forced to infer dust events solely from historical pollution patterns, limiting their ability to capture abrupt concentration changes. Visibility emerged as one of the most influential predictors, consistently ranking among the top features across cities and models, reflecting its strong physical coupling with particulate concentrations.

In contrast, the more moderate gains observed in Abha suggest that local topography and climatic conditions influence the relative importance of meteorological drivers. In this mountainous environment, the recent pollution history, as captured through autoregressive lags and rolling statistics, appears to provide a comparatively stronger predictive signal than in the more dynamically variable, dust-dominated regions.

6.2. Model performance and interpretation

The comparative evaluation across the three Saudi Arabian cities reveals a coherent and interpretable pattern of model behavior that reflects both algorithmic characteristics and local environmental conditions. Across all cities and temporal resolutions, the AtmoStack ensemble consistently achieved the highest predictive accuracy, as indicated by superior R^2 values and lower RMSE compared to individual models for daily and monthly estimations, as shown in Table 2. Although the absolute improvements over the best-performing single model are sometimes modest, their persistence across diverse climatic regimes underscores the robustness and generalizability of the stacking approach.

This consistent advantage arises from the complementary nature of the base learners and the structure of the stacking framework. Random forest, histogram-based gradient boosting, and CatBoost employ distinct learning strategies—bagging and boosting mechanisms with different bias–variance trade-offs—resulting in partially uncorrelated error patterns. By combining these learners through a regularized elastic-net meta-learner, the AtmoStack model effectively exploits error diversity, allowing

residual errors from individual models to be attenuated through weighted aggregation.

An additional source of robustness is provided by the inclusion of the original input features at the meta-learning stage. By enabling feature passthrough, the stacking framework allows the meta-learner to dynamically adjust its reliance on base model estimations and, when necessary, revert to direct relationships between predictors and PM_{10} concentrations. This flexibility reduces sensitivity to localized model failures and mitigates the risk of extreme estimation errors under atypical meteorological or pollution conditions.

The heterogeneous performance of individual models across cities further emphasizes the importance of this ensemble strategy. In Abha's cooler, mountainous environment, models that effectively captured smoother temporal dynamics performed competitively, whereas in the dust-prone regions of Buraidah and Taif, models better suited to handling abrupt, nonlinear variations achieved superior results. These spatial differences indicate that PM_{10} drivers vary substantially across regions and that no single algorithm is universally optimal, reinforcing the rationale for combining multiple learners within a unified framework.

In contrast, the deep-learning baseline exhibited clear limitations. The MLP consistently underperformed relative to tree-based models and, in the case of Taif's daily estimations, diverged catastrophically. This behavior can be attributed to the use of a fixed, untuned architecture, the relatively small sample size from a deep-learning perspective, and the heavy-tailed distribution of PM_{10} concentrations. Despite variance-stabilizing transformations, extreme pollution events can generate unstable gradients during training, leading to poor convergence and weak generalization. Tree-based models, which rely on recursive partitioning rather than gradient-based optimization, are inherently more robust to such distributional challenges.

Notably, the performance gap between the MLP and tree-based models narrowed when estimations were evaluated at the monthly scale. Temporal aggregation reduces high-frequency noise and the influence of extreme values, partially stabilizing the learning process and improving generalization. This suggests that deep learning may become more viable under smoother temporal resolutions or with substantially larger datasets.

Finally, the results highlight an important trade-off between error metrics. In some cases, particularly in Taif, the AtmoStack ensemble achieved lower RMSE at the expense of slightly higher MAE compared to the random forest model. This distinction reflects the differing objectives of these metrics and underscores the need to align model selection with the priorities of practical air-quality applications.

While the above analysis explains the observed differences in model behavior and performance, the following section explicitly discusses the strengths and limitations of the proposed framework and outlines directions for future research.

6.3. Strengths, limitations, and future directions

The developed pipeline demonstrates key strengths. It ensures methodological robustness and internal validity through a strict chronological 80/20 train–test split and the rigorous use of TimeSeriesSplit and Pipeline objects to prevent data leakage during tuning and preprocessing. Additionally, it features full reproducibility via a global random seed and a modular design, which simplifies the integration of new models and lowers the barrier for adaptation to other regions or datasets.

However, the study has several limitations. The hyperparameter search is intentionally shallow to manage computational costs, meaning the reported performance likely represents a conservative lower bound. The feature scope is limited, excluding potentially powerful external predictors such as satellite-derived aerosol optical depth (AOD) or land-use information. Furthermore, the evaluation relied on a single fixed hold-out set, making the results potentially sensitive to the selected time window. Finally, the MLP architecture is left untuned, resulting in a conservative and incomplete assessment of deep-learning performance relative to tree-based models.

Future research should therefore focus on expanding the predictor set to include satellite-based and land-use variables. To ensure a more balanced comparison, more comprehensive hyperparameter optimization should be conducted across all model classes, as optimized MLP architectures may yield significantly more competitive results than the baseline configurations. Furthermore, exploring alternative deep-learning architectures, specifically those designed for sequential data such as LSTM networks, CNN–LSTM, or temporal convolutional networks, may provide a fairer evaluation of neural-network potential, particularly for high-frequency PM_{10} estimation. In addition, rolling origin validation, cross-city transfer testing, or blocked validation schemes—as well as extending the framework to additional cities and longer temporal periods—would further enhance robustness, generalizability, and operational relevance.

7. Conclusions

This study developed a machine learning framework to estimate daily and monthly PM_{10} concentrations across three climatically distinct Saudi cities, integrating ground-based measurements, MERRA-2 reanalysis data, and meteorological variables. The proposed AtmoStack stacked ensemble consistently outperformed individual models (RF, HGB, CatBoost, MLP, and LightGBM), achieving daily RMSE reductions of 2%–7.5% and monthly RMSE reductions of 13%–47%. AtmoStack also improved explanatory power, with daily R^2 ranging from 0.58 to 0.63 and monthly R^2 up to 0.96 (Buraidah) and 0.94 (Taif), confirming its ability to capture both short-term variations and long-term trends. Feature importance analysis highlighted MERRA-2, lagged PM_{10}, visibility, and meteorological variables as key predictors, emphasizing the value of integrated datasets. The framework's stability is ensured through a strict chronological 80/20 train-test split, the use of TimeSeriesSplit and Pipeline objects to prevent data leakage, and full reproducibility via a global random seed. Limitations include a shallow hyperparameter search, an untuned MLP, and a limited feature set. Future work should address comprehensive hyperparameter tuning, incorporation of additional data sources (e.g., satellite AOD, land-use, traffic data), and more rigorous validation schemes such as rolling-origin or external city evaluation to further enhance AtmoStack's accuracy and generalizability.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

The project was funded by KAU Endowment (WAQF) at king Abdulaziz University, Jeddah, Saudi Arabia. The authors, therefore, acknowledge with thanks WAQF and the Deanship of Scientific Research (DSR) for technical and financial support.

We also extend our deepest appreciation to the National Center for Environmental Compliance (NCEC) in Jeddah, Kingdom of Saudi Arabia, for providing access to the ground-based PM_{10} monitoring records, which were essential to the development and validation of this research.

Conceptualization, A.A. (Amjad Alkhodaidi), A.H., A.A. (Afraa Attiah), and A.M.; methodology, A.A. (Amjad Alkhodaidi), A.M., A.H., A.A. (Afraa Attiah); formal analysis, A.A. (Amjad Alkhodaidi); investigation, A.A. (Amjad Alkhodaidi); resources, A.A. (Amjad Alkhodaidi), A.M.; data curation, A.A. (Amjad Alkhodaidi), A.M.; writing—original draft preparation, A.A. (Amjad Alkhodaidi); writing—review and editing, A.A. (Amjad Alkhodaidi), A.H., A.A. (Afraa Attiah), and A.M.; visualization, A.A. (Amjad Alkhodaidi), A.H., A.M., A.A. (Afraa Attiah), and A.M; supervision, A.A. (Afraa Attiah) and A.H.; project administration, A.A. (Afraa Attiah), A.H., and A.M. All authors have read and agreed to the published version of the manuscript.

Conflict of interest

The authors declare no conflicts of interest.

References

1. Gupta P, Zhan S, Mishra V, et al. (2021) Machine learning algorithm for estimating surface $PM_{2.5}$ in Thailand. *Aerosol Air Qual Res* 21: 210105. https://doi.org/10.4209/aaqr.210105
2. Alamoudi M, Taylan O, Keshtegar B, et al. (2022) Modeling sulphur dioxide (SO_2) quality levels of Jeddah City using machine learning approaches with meteorological and chemical factors. *Sustainability* 14: 16291. https://doi.org/10.3390/su142316291
3. Chen MH, Chen YC, Chou TY, et al. (2023) $PM_{2.5}$ concentration prediction model: A CNN–RF ensemble framework. *Int J Env Res Pub He* 20: 4077. http://dx.doi.org/10.3390/ijerph20054077
4. Ibrir A, Kerchich Y, Hadidi N, et al. (2021) Prediction of the concentrations of PM_1, $PM_{2.5}$, PM_4, and PM_{10} by using the hybrid dragonfly-SVM algorithm. *Air Qual Atmos Hlth* 14: 313–323. https://doi.org/10.1007/s11869-020-00936-1
5. Valavanidis A, Fiotakis K, Vlachogianni T (2008) Airborne particulate matter and human health: Toxicological assessment and importance of size and composition of particles for oxidative damage and carcinogenic mechanisms. *J Environ Sci Heal C* 26: 339–362. http://dx.doi.org/10.1080/10590500802494538
6. Kampa M, Castanas E (2008) Human health effects of air pollution. *Environ Pollut* 151: 362–367. https://doi.org/10.1016/j.envpol.2007.06.012
7. Brunekreef B, Holgate ST (2002) Air pollution and health. *Lancet* 360: 1233–1242. http://dx.doi.org/10.1016/S0140-6736(02)11274-8

8. Cohen AJ, Anderson HR, Ostro B, et al. (2005) The global burden of disease due to outdoor air pollution. *J Toxicol Env Heal A* 68: 1301–1307. https://doi.org/10.1080/15287390590936166

9. Carvalho H (2021) New WHO global air quality guidelines: more pressure on nations to reduce air pollution levels. *Lancet Planet Health* 5: e760–e761. http://dx.doi.org/10.1016/S2542-5196(21)00287-4

10. Alghamdi AG, El-Saeid MH, Alzahrani AJ, et al. (2022) Heavy metal pollution and associated health risk assessment of urban dust in Riyadh, Saudi Arabia. *PLoS One* 17: e0261957. https://doi.org/10.1371/journal.pone.0261957

11. Mayer H (1999) Air pollution in cities. *Atmos Environ* 33: 4029–4037. https://doi.org/10.1016/S1352-2310(99)00144-2

12. Alharbi BH, Maghrabi A, Tapper N (2013) The March 2009 dust event in Saudi Arabia: Precursor and supportive environment. *B Am Meteorol Soc* 94: 515–528. https://doi.org/10.1175/BAMS-D-11-00118.1

13. Khodeir M, Shamy M, Alghamdi M, et al. (2012) Source apportionment and elemental composition of $PM_{2.5}$ and PM_{10} in Jeddah City, Saudi Arabia. *Atmos Pollut Res* 3: 331–340. https://doi.org/10.5094/apr.2012.037

14. Ukhov A, Mostamandi S, da Silva A, et al. (2020) Assessment of natural and anthropogenic aerosol air pollution in the Middle East using MERRA-2, CAMS data assimilation products, and high-resolution WRF-Chem model simulations. *Atmos Chem Phys* 20: 9281–9310. https://doi.org/10.5194/acp-20-9281-2020

15. Seroji AR (2011) Particulates in the atmosphere of Makkah and Mina valley during the Ramadan and Hajj seasons of 2004 and 2005. *WIT T Ecology Environ* 147: 319–327. https://doi.org/10.2495/AIR110301

16. Habeebullah TM, Munir S, Zeb J, et al. (2022) Analysis and sources identification of atmospheric PM_{10} and its cation and anion contents in Makkah, Saudi Arabia. *Atmosphere* 13: 87. https://doi.org/10.3390/atmos13010087

17. Gelaro R, McCarty W, Suárez MJ, et al. (2017) The modern-era retrospective analysis for research and applications, version 2 (MERRA-2). *J Climate* 30: 5419–5454. https://doi.org/10.1175/JCLI-D-16-0758.1

18. Kleist DT, Parrish DF, Derber JC, et al. (2009) Introduction of the GSI into the NCEP global data assimilation system. *Weather Forecast* 24: 1691–1705. https://doi.org/10.1175/2009WAF2222201.1

19. Buchard V, Randles CA, Da Silva AM, et al. (2017) The MERRA-2 aerosol reanalysis, 1980 onward. Part II: Evaluation and case studies. *J Climate* 30: 6851–6872. https://doi.org/10.1175/JCLI-D-16-0613.1

20. Mhawish A, Banerjee T, Hamer MS, et al. (2020) Estimation of high-resolution $PM_{2.5}$ over the Indo-Gangetic Plain by fusion of satellite data, meteorology, and land use variables. *Environ Sci Technol* 54: 7891–7900. https://doi.org/10.1021/acs.est.0c01769

21. Zuo X, Guo H, Shi S, et al. (2020) Comparison of six machine learning methods for estimating $PM_{2.5}$ concentration using the Himawari-8 aerosol optical depth. *J Indian Soc Remote* 48: 1277–1287. https://doi.org/10.1007/s12524-020-01154-z

22. Meng X, Hand JL, Schichtel BA, Liu Y (2018) Space-time trends of $PM_{2.5}$ constituents in the conterminous United States estimated by a machine learning approach, 2005–2015. *Environ Int* 121: 1137–1147. https://doi.org/10.1016/j.envint.2018.10.029

23. Dhandapani A, Iqbal J, Kumar RN (2023) Application of machine learning (individual vs stacking) models on MERRA-2 data to predict surface $PM_{2.5}$ concentrations over India. *Chemosphere* 340: 139966. https://doi.org/10.1016/j.chemosphere.2023.139966

24. Di Q, Amini H, Shi L, et al. (2019) An ensemble-based model of $PM_{2.5}$ concentration across the contiguous United States with high spatiotemporal resolution. *Environ Int* 130: 104909. https://doi.org/10.1016/j.envint.2019.104909

25. Shtein A, Kloog I, Schwartz J, et al. (2019) Estimating daily $PM_{2.5}$ and PM_{10} over Italy using an ensemble model. *Environ Sci Technol* 54: 120–128. https://doi.org/10.1021/acs.est.9b04279

26. Sayeed A, Lin P, Gupta P, et al. (2022) Hourly and daily $PM_{2.5}$ estimations using MERRA-2: A machine learning approach. *Earth Space Sci* 9: e2022EA002375. https://doi.org/10.1029/2022EA002375

27. Alkhodaidi A, Attiah A, Mhawish A, et al. (2024) The role of machine learning in enhancing particulate matter estimation: A systematic literature review. *Technologies* 12: 198. https://doi.org/10.3390/technologies12100198

28. Mircea M, Calori G, Pirovano G, et al. (2020) *European guide on air pollution source apportionment for particulate matter with source oriented models and their combined use with receptor models*, Publications Office of the European Union, Luxembourg. https://doi.org/10.2760/470628

29. Di Q, Koutrakis P, Schwartz J (2016) A hybrid prediction model for $PM_{2.5}$ mass and components using a chemical transport model and land use regression. *Atmos Environ* 131: 390–399. https://doi.org/10.1016/j.atmosenv.2016.02.002

30. Lee HJ, Liu Y, Coull BA, et al. (2011) A novel calibration approach of MODIS AOD data to predict $PM_{2.5}$ concentrations. *Atmos Chem Phys* 11: 7991–8002. https://doi.org/10.5194/acp-11-7991-2011

31. Yu H, Fotheringham AS, Li Z, et al. (2020) Inference in multiscale geographically weighted regression. *Geogr Anal* 52: 87–106. https://doi.org/10.1111/gean.12189

32. Xiao Q, Chang HH, Geng G, et al. (2018) An ensemble machine-learning model to predict historical $PM_{2.5}$ concentrations in China from satellite data. *Environ Sci Technol* 52: 13260–13269. https://doi.org/10.1021/acs.est.8b02917

33. Alsaber A, Alsahli R, Al-Sultan A, et al. (2023) Evaluation of various machine learning prediction methods for particulate matter PM_{10} in Kuwait. *Int J Inf Technol* 15: 4505–4519. https://doi.org/10.1007/s41870-023-01521-2

34. Bozdağ A, Dokuz Y, Gökçek ÖB (2020) Spatial prediction of PM_{10} concentration using machine learning algorithms in Ankara, Turkey. *Environ Pollut* 263: 114635. https://doi.org/10.1016/j.envpol.2020.114635

35. Suleiman A, Tight MR, Quinn AD (2019) Applying machine learning methods in managing urban concentrations of traffic-related particulate matter (PM_{10} and $PM_{2.5}$). *Atmos Pollut Res* 10: 134–144. https://doi.org/10.1016/j.apr.2018.07.001

36. Son S, Kim J (2020) Evaluation and predicting PM_{10} concentration using multiple linear regression and machine learning. *Korean J Remote Sens* 36: 1711–1720. https://doi.org/10.7780/kjrs.2020.36.6.3.7

37. Hu X, Belle JH, Meng X, et al. (2017) Estimating $PM_{2.5}$ concentrations in the conterminous United States using the random forest approach. *Environ Sci Technol* 51: 6936–6944. https://doi.org/10.1021/acs.est.7b01210

38. Lee Y, Na I, Son Y (2024) Evaluation of machine learning application on the prediction of particulate matter concentrations in small/medium-sized city. *J Korean Soc Environ Eng*. https://doi.org/10.4491/KSEE.2024.46.10.537

39. Sulaymon ID, Mhawish A, Alqahtani JS, et al. (2026) Understanding the elevated $PM_{2.5}$ pollution in the Middle East during May 2022: Insights from numerical simulations. *Atmos Res* 329: 108527. http://dx.doi.org/10.1016/j.atmosres.2025.108527

40. Merdji AB, Xu X, Mhawish A, et al. (2025) Comparison of aerosol properties over six major deserts from Africa to Asia using AERONET and CALIPSO observations. *J Climate* 38: 7369–7393. http://dx.doi.org/10.1175/JCLI-D-25-0008.1

41. Seinfeld JH, Pandis SN (2016) *Atmospheric chemistry and physics: From air pollution to climate change*, Hoboken: John Wiley & Sons, USA. https://doi.org/10.1021/ja985605y

42. Hyndman RJ, Athanasopoulos G (2018) *Forecasting: Principles and practice,* Melbourne: OTexts, Australia. Available from: https://www.kaggle.com/competitions/instacart-market-basket-analysis/overview.

43. Hastie T (2009) *The elements of statistical learning: Data mining, inference, and prediction*, New York: Springer. https://doi.org/10.1007/b94608

44. Wolpert DH (1992) Stacked generalization. *Neural Networks* 5: 241–259. https://doi.org/10.1016/S0893-6080(05)80023-1

45. Pedregosa F, Varoquaux G, Gramfort A, et al. (2011) Scikit-learn: Machine learning in Python. *J Mach Learn Res* 12: 2825–2830.

46. Prospero JM, Ginoux P, Torres O, et al. (2002) Environmental characterization of global sources of atmospheric soil dust identified with the Nimbus 7 Total Ozone Mapping Spectrometer (TOMS) absorbing aerosol product. *Rev Geophys* 40: 2–1. https://doi.org/10.1029/2000RG000095

47. Rezazadeh M, Irannejad P, Shao Y (2013) Climatology of the Middle East dust events. *Aeolian Res* 10: 103–109. https://doi.org/10.1016/j.aeolia.2013.04.001

48. Namdari S, Karimi N, Sorooshian A, et al. (2018) Impacts of climate and synoptic fluctuations on dust storm activity over the Middle East. *Atmos Environ* 173: 265–276. https://doi.org/10.1016/j.atmosenv.2017.11.01

49. Shalaby A, Rappenglueck B, Eltahir EAB (2015) The climatology of dust aerosol over the Arabian Peninsula. *Atmos Chem Phys Discuss* 15: 1523–1571. https://doi.org/10.5194/acpd-15-1523-2015
50. Abuhasel KA (2023) Statistical and spatial analysis of air pollution in the cities of Abha and Bisha in the Kingdom of Saudi Arabia. *Alex Eng J* 79: 227–236. https://doi.org/10.1016/j.aej.2023.08.021
51. Sayed OH, Masrahi YS (2023) Climatology and phytogeography of Saudi Arabia. A review. *Arid Land Res Manag* 37: 311–368. https://doi.org/10.1080/15324982.2023.2169846
52. Munir S, Siddiqui MH, Habeebullah TMA, et al. (2025) Variability and Trends of $PM_{2.5}$ Across Different Climatic Zones in Saudi Arabia: A Spatiotemporal Analysis. *Atmosphere* 16: 463. https://doi.org/10.3390/atmos16040463
53. Ghahremanloo M, Choi Y, Sayeed A, et al. (2021) Estimating daily high-resolution $PM_{2.5}$ concentrations over Texas: Machine learning approach. *Atmos Environ* 247: 118209. https://doi.org/10.1016/j.atmosenv.2021.118209
54. Yu H, Wang J, Geng C, et al. (2024) Impact of anthropogenic and natural constituents on particulate matter in oasis cities on the southern margin of the Taklimakan Desert based on MERRA-2 and multi-site ground observation. *Atmos Res* 311: 107685. https://doi.org/10.1016/j.atmosres.2024.107685
55. Zheng Y, Liu F, Hsieh HP (2013) *U-air: When urban air quality inference meets big data*, In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, IEEE, 1436–1444. https://doi.org/10.1145/2487575.2488188
56. Madan T, Sagar S, Virmani D (2020) *Air quality prediction using machine learning algorithms–a review*, In: 2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), IEEE, 140–145. https://doi.org/10.1109/ICACCCN51052.2020.9362912
57. Breiman L (2001) Random forests. *Mach Learn* 45: 5–32. https://doi.org/10.1023/A:1010933404324
58. Cutler A, Cutler DR, Stevens JR (2012) *Random forests*, In: Ensemble Machine Learning, Springer, 157–175. https://doi.org/10.1007/978-1-4419-9326-7_5
59. Ke G, Meng Q, Finley T, et al. (2017) LightGBM: A highly efficient gradient boosting decision tree. *Adv Neural Inform Process Syst* 30.
60. Friedman JH (2001) Greedy function approximation: A gradient boosting machine. *Ann Stat* 29: 1189–1232. https://doi.org/10.1214/aos/1013203450
61. Prokhorenkova L, Gusev G, Vorobev A, et al. (2018) CatBoost: unbiased boosting with categorical features. *Adv Neural Inform Process Syst* 31.
62. Toharudin T, Caraka RE, Pratiwi IR, et al. (2023) Boosting algorithm to handle unbalanced classification of $PM_{2.5}$ concentration levels by observing meteorological parameters in Jakarta-Indonesia using AdaBoost, XGBoost, CatBoost, and LightGBM. *IEEE Access* 11: 35680–35696. https://doi.org/10.1109/ACCESS.2023.3265019
63. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521: 436–444. https://doi.org/10.1038/nature14539

64. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. *Nature* 323: 533–536. https://doi.org/10.1038/323533a0

65. Zou H, Hastie T (2005) Regularization and variable selection via the elastic net. *J Roy Stat Soc B* 67: 301–320. https://doi.org/10.1111/j.1467-9868.2005.00527.x

66. Bergstra J, Bengio Y (2012) Random search for hyper-parameter optimization. *J Mach Learn Res* 13: 281–305. https://dl.acm.org/doi/abs/10.5555/2188385.2188395

67. Aksangür İ, Eren B, Erden C (2022) Evaluation of data preprocessing and feature selection process for prediction of hourly PM(10) concentration using long short-term memory models. *Environ Pollut* 311: 119973. https://doi.org/10.1016/j.envpol.2022.119973

68. Dorogush AV, Ershov V, Gulin A (2018) CatBoost: Gradient boosting with categorical features support. *arXiv Preprint*. https://doi.org/10.48550/arXiv.1810.11363

69. Reis B, Maia E, Praça I (2019) *Selection and performance analysis of CICIDS2017 features importance*, In: International Symposium on Foundations and Practice of Security, Springer, 56–71. https://doi.org/10.1007/978-3-030-45371-8-4

70. Fisher A, Rudin C, Dominici F (2019) All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *J Mach Learn Res* 20: 1–81. https://arxiv.org/abs/1801.01489

71. McKinney W (2010) Data structures for statistical computing in Python. *SciPy* 445: 51–56. https://doi.org/10.25080/Majora-92bf1922-00a

72. Abadi M, Agarwal A, Barham P, et al. (2016) TensorFlow: large-scale machine learning on heterogeneous distributed systems. *arXiv Preprint*. https://doi.org/10.48550/arXiv.1603.04467

73. Hunter JD (2007) Matplotlib: A 2D graphics environment, *Comput Sci Eng* 9: 90–95. https://doi.ieeecomputersociety.org/10.1109/MCSE.2007.55

74. Little RJA, Rubin DB (2019) *Statistical analysis with missing data*, New York: John Wiley & Sons. https://doi.org/10.1002/9781119482260

75. Ioffe S, Szegedy C (2015) *Batch normalization: Accelerating deep network training by reducing internal covariate shift.* In: International Conference on Machine Learning, IEEE, 448–456.

76. Kingma DP, Ba J (2014) Adam: A method for stochastic optimization. *arXiv Preprint*. https://doi.org/10.48550/arXiv.1412.6980

77. Kim BY, Lim YK, Cha JW (2022) Short-term prediction of particulate matter (PM_{10} and $PM_{2.5}$) in Seoul, South Korea using tree-based machine learning algorithms. *Atmos Pollut Res* 13: 101547. https://doi.org/10.1016/j.apr.2022.101547

78. Patel P, Patel S, Shah K, et al. (2025) A systematic study on $PM_{2.5}$ and PM_{10} concentration prediction in air pollution using machine learning and deep learning model. *Enviro Chem Ecotox* 7: 1401–1415. https://doi.org/10.1016/j.enceco.2025.07.001

AIMS Press