

Aqpet — An R package for air quality policy evaluation

Dai, Yuqing; Liu, Bowen; Tong, Chengxu; Shi, Zongbo

DOI:

[10.1016/j.envsoft.2024.106052](https://doi.org/10.1016/j.envsoft.2024.106052)

License:

Creative Commons: Attribution (CC BY)

Document Version

Publisher's PDF, also known as Version of record

Citation for published version (Harvard):

Dai, Y, Liu, B, Tong, C & Shi, Z 2024, 'Aqpet — An R package for air quality policy evaluation', *Environmental Modelling and Software*, vol. 177, 106052. <https://doi.org/10.1016/j.envsoft.2024.106052>

[Link to publication on Research at Birmingham portal](#)

General rights

Unless a licence is specified above, all rights (including copyright and moral rights) in this document are retained by the authors and/or the copyright holders. The express permission of the copyright holder must be obtained for any use of this material other than for purposes permitted by law.

- Users may freely distribute the URL that is used to identify this publication.
- Users may download and/or print one copy of the publication from the University of Birmingham research portal for the purpose of private study or non-commercial research.
- User may use extracts from the document in line with the concept of 'fair dealing' under the Copyright, Designs and Patents Act 1988 (?)
- Users may not further distribute the material nor use it for the purposes of commercial gain.

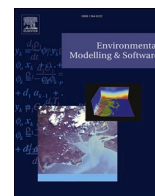
Where a licence is displayed above, please note the terms and conditions of the licence govern your use of this document.

When citing, please reference the published version.

Take down policy

While the University of Birmingham exercises care and attention in making items available there are rare occasions when an item has been uploaded in error or has been deemed to be commercially or otherwise sensitive.

If you believe that this is the case for this document, please contact UBIRA@lists.bham.ac.uk providing details and we will remove access to the work immediately and investigate.



Aqpet — An R package for air quality policy evaluation

Yuqing Dai^a, Bowen Liu^b, Chengxu Tong^a, Zongbo Shi^{a,*}

^a School of Geography, Earth and Environmental Science, University of Birmingham, Birmingham, B15 2TT, UK

^b Department of Management, Birmingham Business School, University of Birmingham, Birmingham, B15 2TT, UK

ARTICLE INFO

Keywords:

Air quality policy evaluation
Machine learning
Weather normalisation
Augmented synthetic control
Observational data analysis

ABSTRACT

Evaluating the effectiveness of clean air policies is important in the cycle of air quality management, ensuring policy accountability and informing future policy-making processes. However, such evaluations are challenging due to complex confounding factors such as varying weather conditions or seasonal or annual changes in air quality unrelated to the policy implementation. To address this challenge, we developed 'aqpet', a R package designed to streamline the quantification of policy effects on air quality using observational data. The package 'aqpet' includes: (1) automated-machine learning to predict air pollutants under average weather conditions – a process term as "weather normalisation"; (2) augmented synthetic control method (ASCM) to quantify the actual policy impact on air pollution. 'aqpet' offers functions for data collection and preparation, building auto-machine learning models, conducting weather normalisation, model performance evaluation and explanation, and causal impact analysis using ASCM. 'aqpet' enables fast, efficient, and interactive policy analysis for air quality management.

1. Introduction

Air pollution has emerged as biggest environmental risk to public health with major implications on economic development. There is growing evidence demonstrating the detrimental health effects of air pollution, ranging from respiratory conditions, arrhythmia, cardiovascular diseases to dementia (Anenberg et al., 2020; Cory-Slechta and Sobolewski, 2023; Lu et al., 2021; Manisalidis et al., 2020). Moreover, the economic impact of air pollution extends beyond to healthcare costs, encompassing not only productivity loss but also environmental remediation expenses and broader societal burdens (Aguilar-Gomez et al., 2022). In response to this challenge, governments worldwide have undertaken a range of measures to ameliorate air quality. Rigorous evaluation of such policies is crucial for evidence-based decision-making, offering indispensable insights to refine and optimize policies, thereby fostering the sustainable advancement of air quality management strategies.

In an era marked by heightened environmental awareness, advancements in scientific technology have made global air pollution data more accessible through monitoring stations. This improved data quality offers valuable insights for governments and the public to understand air quality dynamics. However, existing approaches to data analysis, as noted by Carslaw and Ropkins (2012), tend to be simplistic, hindering

nuanced comprehension of key issues. There is a notable gap between academia and policymakers in leveraging air pollution data comprehensively. Overcoming technical barriers and developing user-friendly analytical tools are essential for bridging this gap and promoting broader adoption.

While understanding the effectiveness of air pollution control regulation is crucial, it is challenging to isolate the causal effect of a specific policy intervention on local air pollution concentrations. For example, pollution concentration levels, as observed, are significantly impacted by meteorological factors like temperature, precipitation, wind speed, and wind direction, making it challenging to distinctly attribute effects solely to the policy (Shi et al., 2021). Grange et al. (2018) developed a machine learning (ML) based technique which could effectively calculate air pollutant concentrations under a "normalised" weather condition. This approach has found widespread application in air quality studies, for example, assessing the impact of Clean Air Action (Vu et al., 2019), Covid lockdown (Ceballos-Santos et al., 2021; Shi et al., 2021; Wyche et al., 2021), spring festival (Dai et al., 2021) on air pollution levels. However, merely comparing (even "weather-normalised") pollution levels before and after the implementation may still yield unreliable results as there is a natural trend in air pollutant concentrations in different years or different seasons (Shi et al., 2021). Adopting a causal inference approach rooted in social science, Cole et al. (2020) integrated

* Corresponding author.

E-mail addresses: y.dai.2@bham.ac.uk (Y. Dai), b.liu.1@bham.ac.uk (B. Liu), z.shi@bham.ac.uk (Z. Shi).

<https://doi.org/10.1016/j.envsoft.2024.106052>

Received 24 January 2024; Received in revised form 3 April 2024; Accepted 17 April 2024

Available online 20 April 2024

1364-8152/© 2024 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

the random forest method for weather normalisation with the Augmented Synthetic Control Method (ASCM) to establish the counterfactual (business as usual) pollution trend post-intervention, and found a 63% reduction in NO₂ concentrations due to the lockdown in Wuhan. This two-step "machine learning-ASCM" method has been further applied for the assessments of air pollution policy, including the Clean Winter Heating Action in China (Song et al., 2023), and the Birmingham Clean Air Zone (Liu et al., 2023).

Recognising the limitations of currently available tools and the lack of comprehensive policy evaluation tools, we developed the *aqpet*, a R package with relatively high computational efficacy, advanced graphical delineations, and cross-platform operability for air quality policy evaluation, which enables researchers and policymakers to rapidly obtain reliable results. The main aim of the *aqpet* is to provide a quantitative and easy to use package for quantifying the impact of clean air policies on air pollutant levels based on observational data. The first objective was to select a development platform not only allowing for unimpeded global access and adaptation but also offering unparalleled transparency. Transparency is indispensable in ensuring the package's credibility, especially when applied to the domain of policy analysis. Secondly, an open-source platform nurtures a collaborative environment, which allows *aqpet* to benefit from diverse user communities' inputs, driving its development through contributions, refinements, and capabilities enhancements. Another important consideration is that the platform's ability to provide an extensive educational framework. The platform should support the provision of comprehensive tutorials and illustrative datasets and require community forums with active participation and discussions.

This paper consists of the following sections: Section 2 discusses the characteristics and framework of the *aqpet*; Section 3 illustrates the usage of *aqpet* using example of London low emission zone as a case study; The last sections discuss the future development and availability.

2. Software design and characteristics

2.1. The R project

R (<https://www.r-project.org/>), originating in the early 1990s as an offshoot of the S language, has burgeoned into a preeminent statistical computing and graphics environment, becoming an indispensable tool for a plethora of academic, industrial, and applied skills (Venables and Smith, 2009). One of key advantages of R for the *aqpet* project is its free access and open-source nature, underpinned by a dynamic and collaborative community (Erickson et al., 2022; Feenstra et al., 2020). These characteristics agree well with the objectives set forth for the software as presented in the Introduction. Additionally, R enables operation cross different operating system, including Microsoft Windows, Apple macOS, and Linux. An optimal equilibrium amongst these systems will significantly enhance *aqpet* operational efficiency.

Another strength of R is its strong data visualisation capability, which allows for intuitive and sophisticated data representation, aiding in both exploratory data analyses and the dissemination of results, thereby facilitating evidence-based policy making. The base graphical functions are intrinsically powerful, enabling a range of intuitive plotting options, and it has been further magnified by additional packages. For example, *ggplot2* is a coherent system for describing and building multi-layered graphs, promoting clarity and precision in graphical representation (Villanueva and Chen, 2019). On the other hand, *plotly*, extends R's capabilities into the interactive domain, which allows for a deeper exploration of the data and is particularly useful in academic settings for elucidating statistical concepts and results to a diverse audience (Sievert, 2020).

An R package is a structured collection of R functions, data, and compiled code in a well-defined format. Importantly, R's package system creates a collaborative environment for innovation and expertise, promoting reproducible research and allowing for the sharing of

datasets and precise analytical methods. As of now, the R ecosystem boasts an ever-growing repertoire of roughly 20,000 packages, all of which are assessable through the Comprehensive R Archive Network (CRAN)—a global repository network. The *aqpet* itself, for air quality research purposes, provides portals towards several outstanding packages to reduce the incidence of unnecessary duplication of efforts. For example, 'openair' functions can be used for data collection (Carslaw and Ropkins, 2012), whereas 'h2o' provides a robust suite of machine learning algorithms (LeDell and Poirier, 2020). The detailed functions will be discussed in the following sections (Section 3).

2.2. The framework, openness, and reproducibility

Fig. 1 illustrates the structured framework underpinning the *aqpet* package. It is composed of four integral components.

- 1) data collection and preparation: The *aqpet* requires air quality and meteorological timeseries data across multiple units (i.e., cities or sites), either through monitoring initiatives or government repositories, with a suite of recommended data sources showed in Fig. 1a.
- 2) Auto-ML model construction and weather normalisation: the *aqpet* utilises numerous state-of-art packages to construct the machine learning models, and then conducts weather normalisation analysis through a resampling approach (Fig. 1b).
- 3) ML model performance evaluation and interpretation: The package also facilitates the evaluation of the model performance, and the exploration of model explainability, including by incorporating explainable machine learning methods such as variable importance (VI), shapely values (SHAP), partial dependence (PDP), and individual condition expectation (ICE) (Fig. 1c).
- 4) Augmented synthetic control method (ASCM) for causal policy evaluation: Under the quasi-natural experimental research design, the final stage involves the selection of treated and control groups, followed by the construction of counterfactual trajectories (of pollutant of interest) for comparison against factual trends. The resulting deviation captures the impact of the policy interventions on the temporal trends of the outcome variable (Abadie, 2021; Athey and Imbens, 2017; Ben-Michael et al., 2021).

Openness and reproducibility are fundamental requirements of credible scientific research, and there is a growing call for a transition from proprietary to transparent models. This highlights the need for those impacted by model-informed decisions to have the capacity to understand and scrutinize the models (Council, 2007). The selection of R for constructing the *aqpet* is not only driven by its open-source and freely accessible nature, which democratizes access for individuals even from under-resourced backgrounds but also due to its capacity for meticulous record-keeping. That is, packages that submitted to CRAN require rigorous assessment of both code and documentation, ensuring that all examples function as expected.

To demonstrate the practical usage of the *aqpet* for policy effects analysis, we have appended exemplar datasets representing a selection of "urban background" air quality monitoring sites across the United Kingdom. These sites are detailed in Table 2, encompassing a range of cities such as Aberdeen, Birmingham Acocks Green, Bournemouth, London North Kensington, among others. Each dataset contains hourly measurements of NO₂ concentrations and meteorological variables — temperature (*temp*), relative humidity (*RH*), wind speed (*ws*), and wind direction (*wd*) during 2018–2019. Notably, North Kensington site serves as a treated unit for the implementation of Ultra Low Emission Zone (ULEZ) strategy, announced on June 8, 2018. Conversely, the remaining sites are considered as "control units", given their non-exposure to ULEZ interventions. Note this example is constructed for instructive purposes and should not be interpreted as an empirical assertion of the ULEZ's impact. An accurate assessment would necessitate more careful

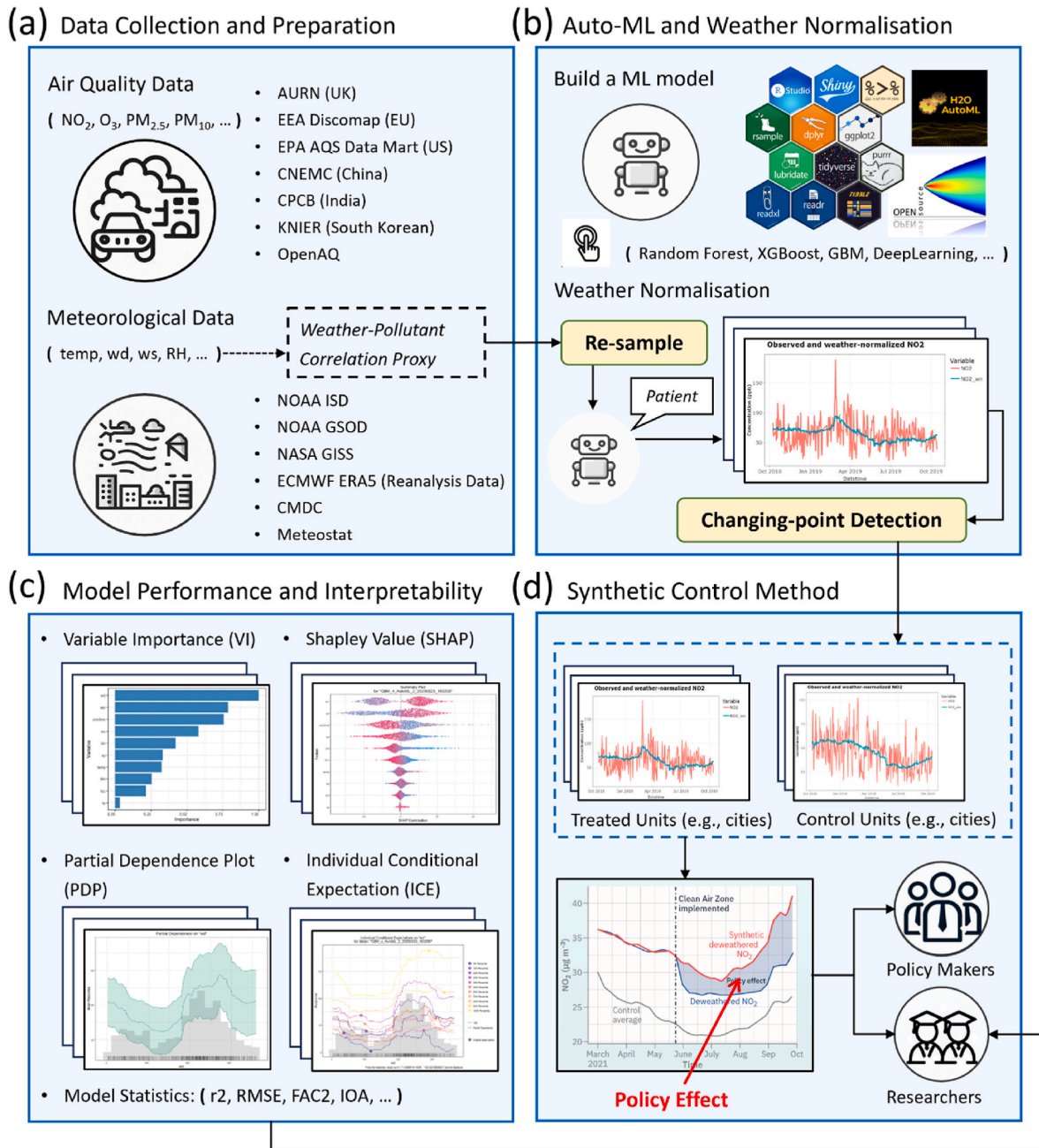


Fig. 1. The conceptual framework of *aqpet* package.

consideration of parameters and design of experiments. A full working flow with the corresponding code functions is presented in Fig. 5.

3. Example of *aqpet* usage and capability

In this section, we present the use of some key *aqpet* functions. An overview of the most-commonly used functions is provided in Table 1. We focus on highlighting core principles including (1) environment setup, data collection and import; (2) construction of auto-machine learning (Auto-ML) models and implementation of weather normalisation; (3) model performance evaluation and model explanations; and (4) development of counterfactual trends using ASCM, and results explanation. The capabilities of R significantly extend to facilitate an interactive and user-centric approach to data analysis, a feature that proves indispensable in the complex processes of model refinement, interpretation, and the selection of pertinent functions for the *aqpet*. The

integration of *aqpet* with various platforms via R platform further amplifies its utility, creating a robust environment for data analysis. In addition, running this package in a Linux-based system substantially accelerates the processes of the model construction and weather normalisation, which can be attributed to Linux’s robust handling of computational tasks and its optimized performance for such data-intensive operations. On the other hand, the graphical user interface, and the comprehensive suite of functions available on Windows provide a more intuitive exploration of the *aqpet* output, assisting users in deriving clear and coherent insights from the data. The choice between operating systems is informed by the specific needs and objectives of the intended analysis, with Linux being preferable for performance-intensive tasks, and Windows offering superior capabilities for data interpretation and visualisation.

Table 1
Summary of *aqpet* functions.

Function	Description	Input	Output
<code>a_scm</code>	preprocesses the dataframe to find major changing points if <code>cpd</code> is TRUE, applies ASCM, and calculates various effect metrics	The data frame to be processed, parameters for processing, a boolean to determine if smoothed residuals should be calculated, and the window size for smoothing	Returns a list of data frames, each containing calculated effect metrics such as effect, effect lower and upper bounds, and average treatment effect
<code>ascm_trend</code>	Creates a time series plot of specified variables from a data frame, with options for customization	The dataframe to plot, variable(s) to be plotted on the y-axis, and range of datetime for the x-axis. Optional start and end times for vertical lines, boolean to render plot using Plotly, labels and titles for the plot, etc.	Prints and returns the time series plot created using either ggplot2 or Plotly
<code>autoMod</code>	Key function within "buildMod", processes the data, splits it for training and testing, runs AutoML to find the best model, and makes predictions	The data frame to be analysed, the response and predictor variables, maximum runtime in seconds and maximum number of models for AutoML, proportion of data for model training, the algorithm and criterion for model selection	Returns a list containing the final model, training data, testing data, predictions, and the leaderboard of models generated by H2O's AutoML
<code>buildMod</code>	An ensemble function that builds and processes models for weather normalisation or other transformations on a list of data frames	A list of data frames to be processed, parameters for the model, including method, output directory, and variables (see 'setWeNorm')	Returns a list containing the processed data frames, processed data, and model objects from weather normalisation, if applicable
<code>control_select</code>	Applies Bayesian Change Point (BCP) analysis to each non-datetime column to help select control units for the ASCM	The data frame to be processed, parameters including policy start and end time, the time resolution for aggregating data	Returns a list containing wide-format data frames with posterior probabilities and mean after filtering by datetime, respectively
<code>load_envir</code>	Sets the CRAN mirror URL, increases timeout, install and load required R packages	URL of the CRAN mirror to use for downloading packages. Default is "https://cran.ma.imperial.ac.uk/"	n.a.
<code>missing_treat</code>	Treats missing data in a dataframe using specified methods	The data frame to be processed, method for treating missing data, the response variable in the dataset, machine learning related parameters if 'h2o' method is selected	Returns a data frame with missing data treated by the specified method. It may return imputed data, a subset of the original data with complete cases, or data with predictions from machine learning models
<code>mod_explain</code>	Analyses and visualises various aspects of a machine learning model's performance and interpretation using H2O's functionalities and custom plots	The machine learning model to be analysed, the data used for testing the model, types of analyses and plots to generate (e.g., variable importance, SHAP, partial dependence plots)	Returns a list of different analysis outputs and plots based on the specified parameters
<code>mod_stats</code>	Statistical parameters for the model evaluation	The dataset containing model predictions and observed values, vector of statistics to calculate, with a "default" option for a standard set of metrics	A data frame containing the calculated statistical metrics for model evaluation, such as n, FB, FAC2, r, RMSEs, RMSEu, RMSE, COE, IOAR.
<code>read_data</code>	Read in csv/xlsx files, and a column "datetime" must contains within each file	A list of parameters including data directory, file pattern, datetime format, and data time range	A list containing processed data frames, and a data frame combined specific column from the list
<code>read_rds</code>	Read in RDS file that stored data using a proper format	A list of parameters including the data directory	A list of data frames or objects extracted from the RDS files
<code>setGP</code>	Initializes general parameters for data processing	Directory for data files, data files format, format of datetime, the dependent variable in the data, date range for filtering the data.	A vector of the specified parameters.
<code>setWeNorm</code>	Sets parameters for building an automatic machine learning model and applying weather normalisation	Response and predictor variables, treatment of missing data, machine learning model parameters, and output options	A vector of the specified parameters.
<code>setSCP</code>	Sets parameters for Synthetic Control Panel analysis	Treated units, start and end times of the policy	A vector of the specified parameters.
<code>show_missing</code>	Check and visualises missing data in a dataframe, produce various plots based on user specifications	The data frame to be analysed, column(s) to analyse, type of date-related analysis (e.g., year, month)	Returns a list containing plots and data frames analysing missing data in different ways, depending on the input parameters
<code>time_trend</code>	Creates a time series plot of specified variables from a dataframe for the ASCM results, with options for customization	The dataframe to plot, variable(s) to be plotted on the y-axis, and range of datetime for the x-axis. Optional start and end times for vertical lines, boolean to render plot using Plotly, labels and titles for the plot, etc.	Prints and returns the time series plot created using either ggplot2 or Plotly
<code>wenorm</code>	Performs weather normalisation on the provided dataset	The dataset to be normalised, the model used for prediction, the response and predictor variables, the constant variables during resampling processes, number of iterations for resampling	A list containing datasets with different stages of processing

3.1. Environment setup, data collection and import

As shown in Fig. 1, the *aqpet* itself is architecturally dependent on several robust R packages (e.g., *openair*, *h2o*, *tidyverse*, *plotly*), leveraging their capabilities to eschew redundancy in computational tasks. For the execution of any *aqpet* function, it is important that these packages are appropriately installed within the operating environment. Upon the installation of the *aqpet*, prerequisite packages will be automatically downloaded and installed to ensure seamless functionality. Additionally, the *aqpet* provides a specific function to further streamline this process:

```
load_envir(.)
```

By default, the CRAN mirror is set to Imperial College London (<https://cran.ma.imperial.ac.uk/>). However, users are recommended to

modify the download mirror in alignment with their geographic locale (<https://cran.r-project.org/mirrors.html>) to avoid potential connectivity or latency issues.

Initiating the analysis requires the precise importing of data into the R environment. To accurately deduce the efficiency of interventions on air quality, the *aqpet* necessitates timeseries data (including a 'datetime' column) on both air quality (e.g., NO₂, PM_{2.5}) and meteorology (e.g., wind speed, wind direction, temperature, see Fig. 1a). Typically, we recommend using hourly data although lower time resolution data, such as daily data, may also work. According to the research scope, data collection could span multiple locations such as different cities or sites. This encompasses a site (or a city) that has experienced specific policy interventions, termed "a treated unit", and their counterparts not affected by such initiatives, known as "control units". While the control units should somehow present emission patterns similar to the treated

units prior to policy implementation, these control units are not subjected to the effects of the intervention when the policy is enacted. The data for each unit should be stored into a csv or xlsx file, and all the files should be resided within a single working directory.

There are several ways for data collection. Fig. 1a presents the potential sources for both air quality and meteorological data across different countries. Notably, some software packages have been developed that offer functionalities for assessing and processing air quality and meteorological data (Carslaw and Ropkins, 2012; Díaz et al., 2021). Upon preparation of the data, functions for import data in *aqpet* include:

```
read_rds(.) and read_data(params)
```

The first function is straightforward, which retrieves and loads all RDS (R data storage) files with appropriate format into a list. The latter function ('read_data') identifies and processes data files from a specific directory. It requires a 'params' argument, a list containing specific key-value pairs. Users may use the 'setGP(.)' function to configure these global parameters, and the options requisite for populating this argument are.

- `data_dir`: the directory path from which the data files are to be retrieved. The function automatically corrects any backslashes to forward slashes in the path.
- `file_pattern`: files stored within the working directory, *aqpet* supports both the csv and xlsx formats.
- `datetime_format`: the format of the 'datetime' column in the files.
- `dependent_variable`: the designation for the dependent variable within the dataset generally corresponds to air pollutants. Note that character case sensitivity is important in this context.
- `data_timerange`: a vector of two date elements specifying the range of dates to filter the data. It should cover enough data (e.g., one year) before and after policy interventions.
- `wenormed`: a logical value indicating if the data has been weather normalised.

All imported data sets are housed within a list structure, with data for each site stored as a distinct list element. The 'read_data' function has the additional capability of extracting the dependent variable from multiple input files. Subsequently, it consolidates this data into a unified data frame, offering a comprehensive view of the temporal variations in air pollutant concentrations across various locations. For example, consider the following illustration of importing data:

```
data_preWeNorm <- read_data(params)
```

Table 2

Summary statistics for model evaluation of observed and AutoML "gbm" model predicted NO₂ concentrations at monitoring sites in the UK for 2018–2019. The statistics include the fraction of predictions within a factor of two (FAC2), the correlation coefficient (r), the refined index of agreement (IOAr), the root mean squared error (RMSE), the unsystematic RMSE (RMSEu), and the systematic RMSE (RMSEs). More statistics can be found in the code options.

Site	n (data count)	FAC2	r	IOAr	RMSE	RMSEu	RMSEs
Aberdeen	17370	0.94	0.91	0.87	5.71	4.42	3.61
Birmingham Acocks Green	17365	0.98	0.96	0.91	9.53	4.02	8.64
Bournemouth	17348	0.97	0.93	0.89	5.13	4.57	2.33
Leicester University	17318	0.95	0.89	0.87	6.08	4.64	3.93
Manchester Piccadilly	17203	0.94	0.92	0.87	6.46	4.35	4.78
North Kensington	16636	0.97	0.90	0.81	7.75	6.40	4.37
Newcastle Centre	17160	0.96	0.91	0.91	7.33	6.13	4.02
Nottingham Centre	17148	0.93	0.95	0.9	8.48	4.40	7.25
Oxford St Ebbes	17135	0.97	0.94	0.91	6.75	4.76	4.78
Plymouth Centre	17118	0.98	0.93	0.88	9.73	5.58	7.97
Preston	17114	0.97	0.91	0.89	6.97	4.73	5.12
Reading New Town	17048	0.96	0.88	0.87	6.29	5.17	3.59
Sheffield Tinsley	16981	0.95	0.96	0.88	6.46	4.78	4.35
Stoke on Trent Centre	16904	0.91	0.91	0.91	6.64	5.73	3.36
Sunderland Silksworth	16902	0.95	0.93	0.91	6.65	6.01	2.83
Wigan Centre	16316	0.97	0.91	0.92	6.55	4.46	4.79
Wirral Tranmere	17368	0.95	0.90	0.91	7.20	4.60	5.54

The 'data_preWeNorm' object is structured as a list comprising two main elements. The first element is a list that encapsulates air quality and meteorological data collected from various locations. The second element is a data frame that contains two critical types of information from all sites: 'datetime' entries and measurements of the dependent variable, which could, for example, be the concentration of nitrogen dioxide (NO₂) and fine particles (PM_{2.5}).

3.2. Auto-ML and weather normalisation

3.2.1. Auto-ML

Upon the successful aggregation and integration of data within the *aqpet* framework, the subsequent phase entails the construction of a machine learning model, incorporating both emission activities and meteorological parameters. Due to the challenge of acquiring timely emission inventories (e.g., vehicle count, traffic composition), in general, these statistical models use a set of time-variant variables as proxies for emissions, assuming that human-induced emissions exhibit discernible and regular patterns. These temporal variables include trend, year, month, day, hour, day of the week, and day of the year. In conjunction, meteorological factors are integrated, facilitating the establishment of a robust relationship between the dependent variable and the aforementioned explanatory variables. In this context, *aqpet* leverages the capabilities of the H2O automated machine learning (H2O AutoML) engine (LeDell and Poirier, 2020), which provides a comprehensive suite of algorithms and computational tools. It supports a wide array of machine learning models, ranging from generalized linear models (glm) to more complex ensemble methods, such as distributed random forests (drf) and gradient boosting machines (gbm). The key advantage of AutoML lies in the algorithmic automation, streamlining the process of model selection, tuning, and evaluation. This automation enables both novice and expert users to efficiently develop robust models that would traditionally require extensive hyperparameter determination and manual iteration.

The *aqpet* function for building the model is 'autoMod(.)'. Within this function, users are required to specify a 'response_variable' and furnish a vector comprising 'predictor_variables' to construct the model. A predictor variable is an independent variable used in statistical models to forecast or predict the values of a dependent response variable. Other model parameters necessitate specification to tailor the modelling procedure to the users' specific needs and constraints. These parameters include 'max_runtime_secs', which sets an upper bound on the model's runtime to ensure computational efficiency; 'max_models', capping the number of models to be generated; and 'split_proportion', which determines the division of data into training and evaluation subsets. In

machine learning, the typical split for training and testing sets varies based on the size of a dataset. Commonly a 70/30 or 80/20 split is used. The key is to ensure that the data is shuffled and has a normal distribution before splitting (Tan et al., 2021). There is no one-size-fits-all value for 'max_runtime_secs' and 'max_models' as they largely depend on the dataset size and the specific requirements of the machine learning task. However, a common practice is to set a reasonable limit on the maximum operating time to balance between computational efficiency and model performance. Similarly, users might start with a smaller number of training models to quickly access performance (see Section 3.3) before scaling up based on the model results and available computational resources.

Furthermore, the 'algorithm' parameter allows users to select the desired output model algorithm, while the 'criterion' parameter establishes the criteria for model selection (e.g., deviance for regression). Below is an example that demonstrates how to build a GBM (Gradient Boosting Machine) model for NO₂ predictions using time variables (e.g., *trend*, *dow*, *doy*) and meteorological variables including *wd*, *ws*, *temp*, and *RH*.

```
autoMod(data,
  response_variable = "no2",
  predictor_variables = c(time_variables, meteorological_variables),
  max_runtime_secs = 120,
  max_models = 10,
  split_proportion = 0.8,
  algorithm = "gbm",
  criterion = "AUTO"
)
```

The output generated by 'autoMod' is a structured list that comprises five elements: the optimal machine learning model designated as "best", the training dataset used for model fitting, the evaluation dataset used for model assessment, model predictions, and an ensemble of trained models with their respective performance metrics.

3.2.2. Weather normalisation

Air pollutant concentrations are subject to several influencing factors: their emissions (such as from human activities), dispersion and chemical transformation post-emission, and sinks (such as wet deposition). Meteorological conditions act as important confounders, introducing substantial variability into air quality measurements independent of anthropogenic influences or policy interventions (Shi et al., 2021). For example, precipitation can serve as a cleansing agent, reducing particulate matter (PM) levels, whereas temperature inversions can lead to the entrapment of pollutants near the urban surface, and thus an increase in air pollution levels. Neglecting to account for these meteorological variables can lead to erroneous attributions of air quality shifts to policy interventions when, in fact, they may simply reflect weather variability (Angrist and Pischke, 2009). Therefore, following the core econometric tenet of segregating the treatment effect from confounding factors (Imbens and Wooldridge, 2009), removing weather impacts is critical in creating a "ceteris paribus" scenario. This condition is essential, where all other variables are held 'constant' except for the policy intervention under study. Failure to adjust for weather influence may result in skewed evaluations of air quality interventions, potentially leading to significant misjudgements in their economic and public health implications (Grange and Carslaw, 2019; Grange et al., 2018; Shi et al., 2021).

Previously, Grange et al. (2018) proposed a meteorological normalisation approach based on random forest (RF) and developed R package 'rmweather'. This method involves two key steps: (1) constructing and evaluating the RF model using bootstrapped data subsets, and (2) repeatedly resampling meteorological explanatory variables for 1000 times while retaining time variables (e.g., UNIX time) as the input for the RF model. The average of these predictions is then used to

estimate the weather-normalised concentration at a specific time. This weather normalisation approach has been broadly adopted in air quality studies (Dai et al., 2021; Falocchi et al., 2021; Grange and Carslaw, 2019; Shi et al., 2021).

In the *aqpet*, the weather normalisation adheres to the approach similar to that of Grange et al. (2018), yet it operates independently of the initial model construction phase — once the model is formulated through an automated machine learning procedure (i.e., 'autoMod'), the weather normalisation function takes a step further. This function, denoted as 'wenorm', is executed using the following syntax.

```
wenorm(data, model,
  response_variable = "no2",
  predictor_variables = c(time_variables, meteorological_variables),
  constant_variables = time_variables,
  num_iterations = 1000
)
```

In this context, the 'constant_variables' refer to the set of variables that are held fixed throughout the resampling process (i.e., Step 2 in the 'rmweather' function). The parameter 'num_iterations' specifies the number of resampling iterations. The choice of its value requires careful consideration to align with research objective. A higher value of iterations will increase computational demands and may also reduce the temporal variation of pollutants, potentially obscuring short-term fluctuations of interest. The result of this function is a refined list that includes a summary dataset containing weather-normalised results. These are indicated with the suffix "_wn", as in "no2_wn", signifying that the values represent concentrations normalised for meteorological conditions.

The functions outlined above demonstrate the process for constructing an automated machine learning air quality model and for incorporating weather normalisation to discern human influences on air quality. To streamline this workflow, the *aqpet* provides an ensemble function to efficiently process data from multiple sites:

```
data_WeNorm <- buildMod(list, params)
```

The 'list' parameter contains air quality and meteorological data across all sites, for example, the first element in 'data_preWeNorm' described in Section 3.1. The 'params' is a list of model requirements that can be simply configured via 'setWeNorm' (similar to 'setGP'). The output comprises an array of site-specific models, alongside a dataset of both weather-normalised and original data for each site. Additionally, an integrated data frame is generated, only including weather-normalised data for each site for further analysis.

3.2.3. Changing-point detection and selection of control units

Selecting appropriate control units is paramount as it directly influences the accuracy of the counterfactual trends — predictions of what would have happened in the absence of the treatment (e.g., policy interventions). Inaccurate counterfactuals can lead to biased estimates of treatment effects, potentially providing misleading policy recommendations and scientific inferences. Therefore, it is vital to establish and adhere to rigorous criteria for selecting these units. This ensures a high degree of comparability between the treated units and their synthetic counterparts (Abadie and Diamond, 2010; Doudchenko and Imbens, 2016).

The guiding principles for selecting treated and control units are to ensure that the units share similar characteristics and to ascertain that the control units provide a credible counterfactual for the treated units. This entails a careful balance, seeking control units that share similar trend to ensure validity, yet not so similar that they might themselves have been treated under slightly different circumstances—a concept closely related to the idea of the "common support" condition in the propensity score matching literature (Rosenbaum and Rubin, 1983). It is

also essential to consider the number of pre-treatment periods available for analysis, which is crucial for the accurate construction of the synthetic control unit and, by extension, reliable causal inference (Abadie and Diamond, 2010).

Within the realm of environmental research, assessing the causal impact of policy interventions on air quality at a specific city or site is a complex task. One strategy is to use monitoring sites in other cities as control units. However, it is important to ensure that these control units are appropriately selected to match their characteristics to the treated unit, a task that demands careful attention to various factors. Below we present the key considerations for this task.

- 1) Gross Domestic Product (GDP): GDP stands as a robust indicator, reflecting the intensity of economic activities, which are intrinsically connected to emission levels. Cities exhibiting similar GDP values are likely to manifest comparable scales in production, transportation, and energy usage, as the key air pollution emission sources. Nevertheless, it is imperative to dissect the composition of the GDP, as cities with analogous GDP figures may present divergent emission patterns due to differences in industrial focus, such as a dominance of service industries versus manufacturing.
- 2) Sectoral dynamics: Beyond GDP, understanding the sectoral breakdown of economic activity can be illuminating. If the treated city has a significant industrial base, control cities should ideally share this characteristic. This ensures that major emission sources are consistently represented across the treated and control units.
- 3) Geographical and climatic factors: for example, cities in colder regions might have higher emissions due to heating requirements), existing environmental regulations (i.e., the regulatory environment is similar across cities), and technology adoption (e.g., renewable energy systems or electric vehicles).
- 4) Urbanization rates: they influence emission activities, especially if rapid urbanization leads to increased construction activities, transportation demands, and changes in energy consumption patterns.
- 5) Demographic factors: Urban centres of similar population scales may encounter parallel issues in the realms of urban development, transit systems, and domestic energy use, each playing a role in shaping emission trends. Furthermore, the aspect of population density provides a lens to examine the layout of urban areas, potentially impacting transportation preferences and consequent emission figures.

However, the similarity in these covariates does not guarantee that the potential control units are indeed unaffected by the interventions. For example, even in the absence of a direct effect, announcement effects or spillover effects could induce changes in control units around the time of policy implementation (Borghesi et al., 2022; Nilsson et al., 2017). To further address this concern, the *aqpet* integrated a diagnostic function that serves as additional test to demonstrate reliability and stability of the selected control units:

```
control_select(data, params, time_resolution)
```

This function will invoke 'bcp' that conducts a Bayesian analysis of change point problems, operating on the assumption that data before and after the change point exhibit distinct characteristics (Erdman and Emerson, 2008). Here change point problems refer to the task of identifying points in a data sequence where the statistical properties (e.g., mean, variance) significant shift. Consequently, the function identifies clear indications of the occurrence of changing points and provides mean estimates and posterior probabilities.

Within the 'params' of this function, a temporal window can be specified, defining the start and end times of an intervention. Generally, if the control units are genuinely unaffected by the intervention, it would be expected that their "weather normalised" time trends remain

smooth and free of such discontinuities. Significant variation in the control units within the time window of the intervention (especially near the start time) may suggest the presence of confounding factors. These factors need to be accounted for to prevent biased estimates. The straightforward approach would entail the removal of the control unit; however, the decision to proceed with the deletion hinges on the capacity to elucidate and rationalize the underlying causes of the observed change points.

3.3. Model performance and interpretability

It is important to evaluate machine learning models to ascertain their predictive accuracy and generalization capabilities. Also, machine learning models, while adept at capturing complex patterns and relationships within data, often operate as "black boxes", rendering their decision-making processes opaque (Rudin, 2019). In this regard, model interpretability becomes important, as it facilitates in establishing trust among stakeholders. The *aqpet* provides two main functions for this purpose: the 'mod_stats' and 'mod_explain' functions. The 'mod_stats' function offers a set of commonly used numeric model evaluation metrics. While it encompasses model statistics addressed by the *openair* package 'modStats' function, such as a factor or two (FAC2), normalised mean bias (NMB), root mean squared error (RMSE), the refined index of agreement (IOAr) (Willmott et al., 2012) — it further delineates systematic RMSE and unsystematic RMSE. While systematic RMSE sheds light on consistent model deviations from observed values, unsystematic RMSE provides insights into random and unpredictable error components, enhancing the robustness of model evaluation. The function call is shown below and the results in Table 2.

```
mod_stats(data, mod = "mod", obs = "obs", stats = "default")
```

Moreover, the *aqpet* provides the 'mod_explain' function, catering to various analytical requirements. As its core, this function is designed for evaluating and interpreting models, particularly those generated using the *H2O* framework (LeDell and Poirier, 2020). Upon invocation, it requires the provision of a model and its corresponding, ensuring the foundational prerequisites for analysis are met. Here below is an example:

```
mod_explain(model, data, output)
```

The 'output' encompasses a comprehensive suite of functions for model analysis and interpretation. This includes 'residual analysis' for assessing model residuals, 'varimp' for evaluating variable importance, both global ('shap_summary') and local ('shap_explain_row') interpretability using SHapley Additive exPlanations (SHAP), individual conditional expectation ('ice'), partial dependence plot ('pdp'), and other useful indicators to understand model behaviour.

Fig. 2 presents examples through various plots to explain the trained gbm model. Fig. 2a shows the relative importance of each feature used in the model. The length of the bar signifies the contribution of each feature to the model's predictions. A longer bar means the feature is more important for the model. In this case, the day of year emerges as the most important feature, followed by hour, trend, etc. Fig. 2b shows the partial dependence of NO₂ on the hour. Here, the line illustrates the average prediction of the model while the shaded area represents the confidence interval around that average prediction. These visualisations offer insights into how NO₂ concentrations vary with changes in the hour of a day, while holding all other factors fixed. The histogram at the bottom indicates the distribution of the hour in the dataset. Fig. 2c shows the SHAP contribution of each feature to the prediction for individual samples. Points represent individual SHAP values for each feature across all data points (known as "global interpretation"). Features are vertically stacked based on importance, with colour denoting the feature value (red for high, blue for low) (Lundberg et al., 2020).

Lastly, Fig. 2d shows the feature-specific contributions to a single prediction from the gbm model, with the direction and magnitude of each feature's impact on the prediction delineated by colour and length of the bars, respectively. The "Bias Term" represents the baseline prediction, while the aggregate of the SHAP values shifts this baseline to the final model output, as indicated by the dotted prediction line. It helps in understanding the overall impact of features on the model output at a specific time (known as "local interpretation"). Positive SHAP values indicate a feature pushing the model prediction higher, and vice versa. The results reveal that the day-of-year and wind speed exert the largest negative and positive impacts on air quality, respectively (Fig. 2d).

3.4. Application of ASCM

The synthetic control method (SCM) creates a predicted "business-as-usual" or "counterfactual" concentration of an air pollutant, which is a weighted average of its concentration of the control units. SCM aims at closely replicating the pre-treatment outcomes of a "treated" unit (e.g., the concentration of an air pollutant before the implementation of a policy). The impact of the treatment is then accessed by comparing the post-treatment outcomes of the treated units with its synthetic counterpart (Shi et al., 2022). The efficacy and utility of SCM have been evaluated in studies such as the evaluation of the Kansas tax cuts (Botosaru and Ferman, 2019; Xu, 2017). The Augmented Synthetic Control Method (ASCM) was developed to integrate a ridge regression model that formulates the estimator as a weighted average of control unit outcomes. It allows negative weights, facilitating an improved pre-treatment fit through controlled extrapolation (Ben-Michael et al., 2021).

In preparation for an ASCM analysis, it is important to ensure that the dataset is complete, with no missing data across both treated and control units. The *aqpet* offers a function to track the missing values:

```
show_missing(data, datetime)
```

This function pinpoints the occurrence of missing data and provides a comprehensive overview of the distribution of missing data. Subsequently a general-purpose function to handle the missing values is:

```
missing_treat(data, method = "rm", ...)
```

It incorporates various strategies, including direct removal ('rm'), substitution with mean ('mean') or median ('median') values, and imputation using linear regression ('linear') or alternative machine learning models (*H2O*). When aligning the data with the timescale of the intervention, a practical approach is to aggregate from a finer resolution to a coarser resolution. For example, in cases where a policy spans several months and missing data are at an hourly resolution, converting the data to a daily resolution can improve data continuity and efficiency of analysis. However, it is noted that such aggregation reduces the granularity of data, potentially leading to the loss of important short-term patterns, which can impact the sensitivity and accuracy of ML models. It necessitates careful consideration of the model choice and the balance between data detail and volume.

The function for using the ASCM to construct counterfactual trends for treated units in relation to the control units is:

```
a_scm(data, params)
```

Here, 'data' refers to a data frame comprising timeseries of weather-normalised data for both treated and control units. The parameters

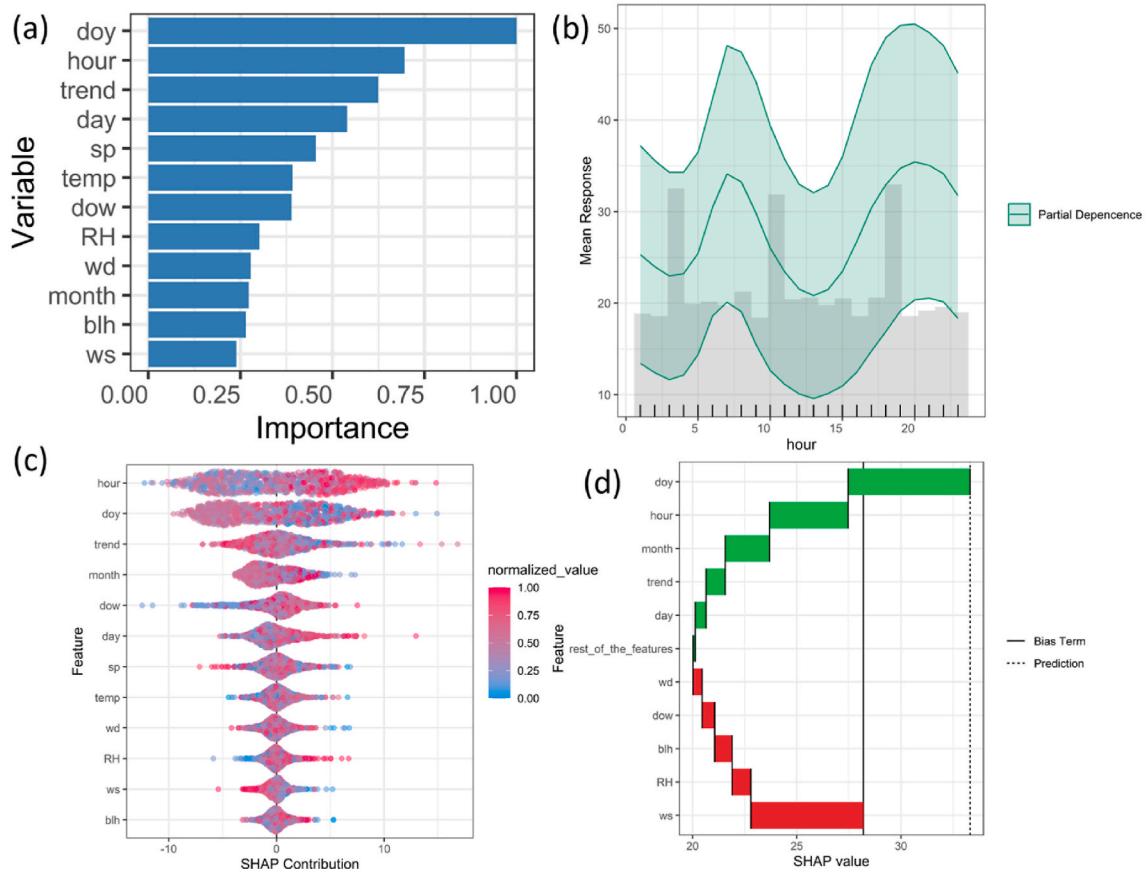


Fig. 2. Explanations for a Gradient Boosting Machine model for NO₂ prediction at North Kensington, London: (a) variable importance, (b) partial dependence plot on "hour", (c) global SHAP values, and (d) local SHAP values.

'params' encompasses a list, which includes a vector of 'treatment_group', as well as 'start_time' and 'end_time' of the policy intervention. Similarly, it can be set up through 'SetSCP(.)' function. The output provides absolute and percentile effects of the intervention on each treated unit, accompanied by associated uncertainties. The lower and upper bounds of the uncertainty refer to the range within which the true effect size is expected to lie with a certain level of confidence, providing a way to quantify the uncertainty in the effect estimates due to various factors such as sample variability, model specification errors, and external influences (Ben-Michael et al., 2021). Moreover, it calculates the average treatment effect (ATE) from the time of intervention up until the end of the dataset.

3.5. Model results and visualisation

Data visualisation plays an important role in the interpretation and communication of the research findings. Visually appealing and informative graphics transform complex data into a more accessible and comprehensible format and will enhance the clarity and understanding of the research objectives. Based on the *ggplot2*, *aqpet* provides a main function for timeseries plots that suitable for different purposes:

```
time_trend(data, time_resolution = "3 day", highlight = "North_Kensington")
```

This function accommodates a wide array of time series data, with the 'time_resolution' offering flexibility to adjust temporal granularity. It serves as a versatile tool, allowing users to easily modify various parameters such as the title, subtitle, y- and x-axis labels, and even craft interactive graphics. Here we only focus mainly on its core capabilities.

As an example, Fig. 3 comprises two panels, each showing NO₂ concentrations at various urban background sites in the UK from January 2018 until early 2020. Fig. 3a presents the raw observational data, capturing the inherent variability and episodic peaks in NO₂ levels. This variation stems from a confluence of factors including both emissions and atmospheric processes. Fig. 3b, on the other hand, presents the weather-normalised data, thereby isolating the pollution trend from the influence of meteorological changes. It is also helpful to highlight the treated unit to clearly present the impact of specific interventions, and this can be achieved by incorporating the treated unit into the 'highlight' option into the function. As a result, the "treated unit" was marked in red (i.e., North Kensington subjected to the London Ultra Low Emission Zone (ULEZ) policy) against the "control units" represented in grey (i.e., sites not subjected to the ULEZ policy).

The ASCM results can be visualised through a series of plots by using the following function:

```
ascm_trend(data, y_variable, add_ribbon = T, start_times)
```

This function is designed to elucidate the underlying trends of the specified 'y_variable' from the dataset data, with the option 'add_ribbon' = 'T' incorporating a confidence interval ribbon that captures the

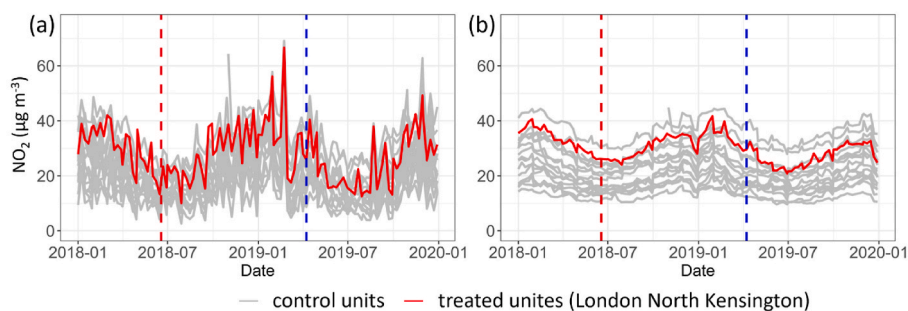


Fig. 3. Timeseries data of NO₂ concentrations at monitoring sites across the UK. (a) Observed concentration; (b) Weather-normalised concentration. The red dashed line in each panel indicates the announcement date of the policy, the blue dashed line indicates the implement date of the policy.

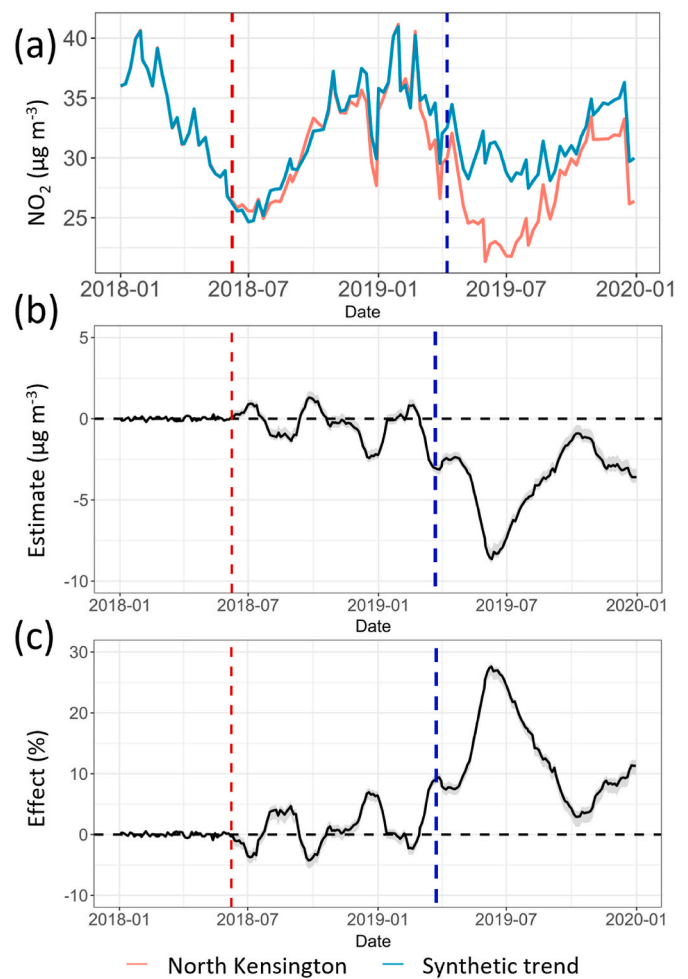


Fig. 4. (a) Weather-normalised NO₂ concentrations at North Kensington, London, and its synthetic trend generated from the control unit data. (b) Estimated policy effect of ULEZ on NO₂ concentrations (difference in concentration between North Kensington and its synthetic control). (c) Estimated percentage change in NO₂ concentration (positive means a decreases in concentration, the intended policy effect) due to ULEZ. The red dashed line in each panel indicates the announcement date of the policy, the blue dashed line indicates the implement date of the policy, and the shaded areas represent the uncertainty.

uncertainty around the estimates. The 'start_times' parameter demarcates the initiation of the intervention within the analysis.

Fig. 4 exemplifies the output of the 'ascm_trend' function, presenting the impact of the London ULEZ after its announcement date on NO₂ concentrations at North Kensington, a region of west London that

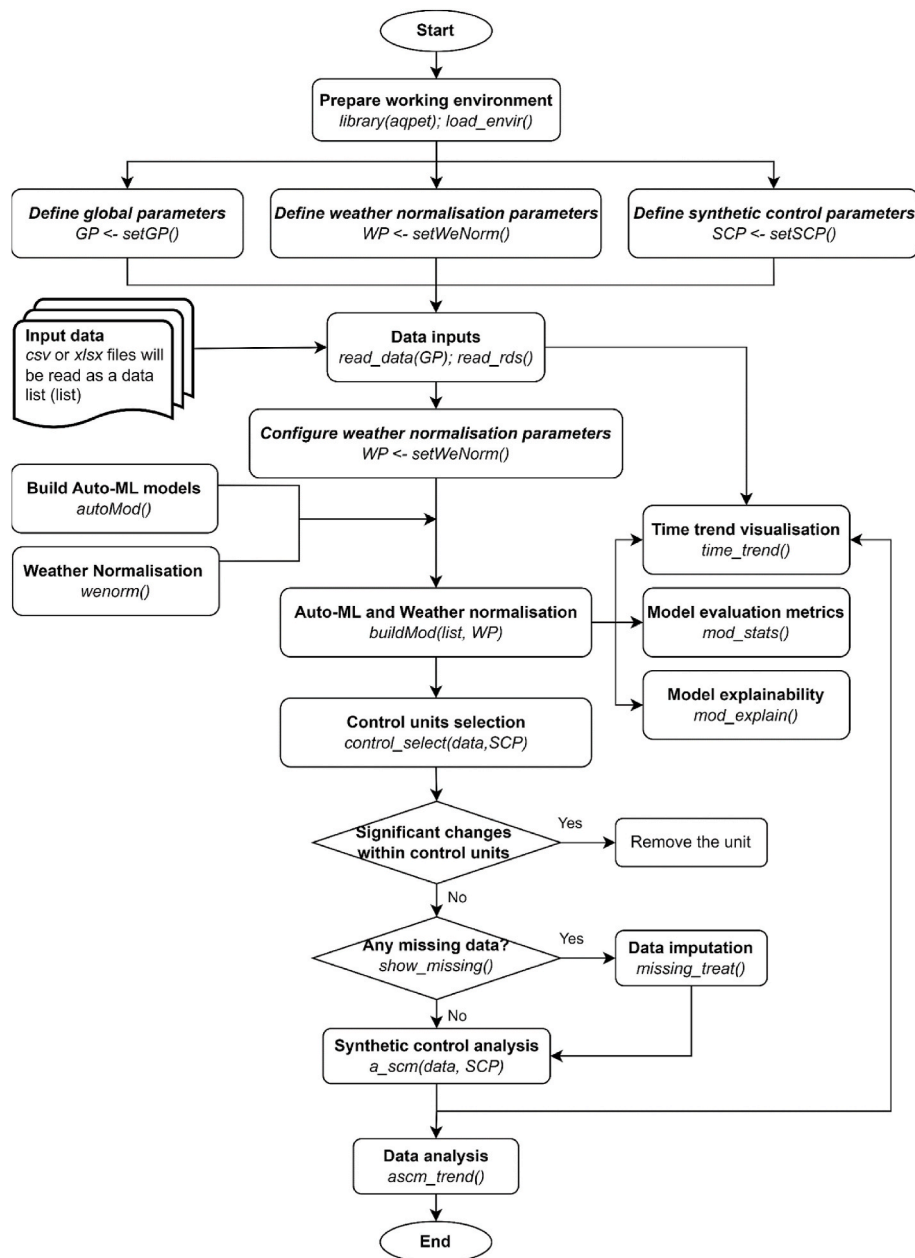


Fig. 5. A flow chart for the *aqpet* package with the corresponding code functions. *GP* represents the set of global parameters, such as the location of the working directory, file format, and the range of data time; *WCP* stands for weather normalisation parameters, such as configurations for the automated machine learning models and the location for outputs; *SCP* refers to the parameters for synthetic control. More details are presented in the function description.

usually recognised as an air quality urban background area (Bigi and Harrison, 2010; Kamara and Harrison, 2021). Fig. 4a represents the actual weather normalised NO₂ concentrations at North Kensington (red solid line) alongside a synthetic counterfactual line (blue solid line). The synthetic line is constructed using data from control units that were not affected by the policy. Fig. 4b illustrates the difference, or gap, between NO₂ concentrations at North Kensington and its synthetic trend. This gap is known as the policy effect, with negative values suggesting a reduction in NO₂ relative to synthetic control. Finally, Fig. 4c quantifies the policy effects in terms of percentiles, offering insights into how the ULEZ has altered NO₂ levels over time. The magnitude of the effect increases when actual concentrations fall below the synthetic trend, indicating the policy's effectiveness in reducing pollution.

Overall, Fig. 4 suggests a discernible reduction in NO₂ concentrations at North Kensington after the introduction of the Low Emission Zone, in contrast to the levels in the absence of the intervention. This is

evidenced by the widening gap in Fig. 4b and the increasing effect size in Fig. 4c following the policy's start date. However, note that the demonstration here only provides preliminary insights into the policy's effectiveness and does not constitute a definitive assessment of its impact. Further analytical scrutiny involving a broader set of control units and additional covariates may be warranted to ensure a robust evaluation of policy efficacy. Finally, Fig. 5 provides a summary of code functions used for the analysis.

4. Discussions and future developments

The *aqpet* package integrates the state-of-art R packages, synergizing multidisciplinary insights to provide the rigorous evaluation of policy interventions under a causal framework. While users have the flexibility to tailor the model and framework based on their configurations, most analyses require default settings or straightforward modification, as

demonstrated in the examples in this paper. The *aqpet* package has been developed based on methodologies (not the package itself) based on previous research (Cole et al., 2020; Liu et al., 2023; Song et al., 2023). The *aqpet* package and these works for the evaluation of policy interventions are built upon on the *augsynth* package ASCM (Ben-Michael et al., 2021), which might not fully capture the immediate or short-term changes of policy interventions (Athey and Imbens, 2017), and relies on historical data, which can lead to challenges for creating synthetic control groups in dynamic environments (Athey, 2015). Therefore, identifiable domains call for further refinement and expansion, particularly in term of integrating new capabilities into the package.

- 1) Develop methodologies to better handle short-term changes and data scarcity, as well as refining approaches to analyse multiple, intertwined policy interventions, for example, the "event study" that focuses on the temporal dynamics surrounding policy changes (Getz and Page, 2019; Miller, 2023).
- 2) Enhance data accessibility by including air quality and meteorological databases from additional regions, such as China and India. Data from different regions may come in varied formats and standards. For effective use, it is important to standardize this data, ensuring that it is compatible and can be easily integrated into the *aqpet*-acceptable database framework.
- 3) Establish an open community or platform that enables users to upload, store, and share their processed data across different spatial and temporal scales. While building an auto-machine learning model and processing weather normalisation are more efficient than traditional process-based air quality models, they remain computationally intensive, demanding substantial resources in terms of both time and computational power. A golden database would liberate users from the burden of intensive computational tasks, thereby boosting overall efficiency. It is important to emphasise that any expansion of data accessibility should adhere to strict data evaluation and usage regulations. Robust security protocols and compliance mechanisms should be implemented to ensure the integrity and privacy of data, especially when dealing with sensitive environmental information.
- 4) It would be useful to embeds R code within LaTeX documents to generate dynamic reports (Leisch, 2002), which allows users to automatically update and reproduce all outputs, including plots, tables, and statistics (useful for government reports), as inputs or analyses change.

4.1. Software availability

Software name: *aqpet* (Air Quality Policy Evaluation Tool).
 Developer: Yuqing Dai.
 First year available: 2023.
 Hardware requirements: PC/Mac.
 Program language: R.
 Cost: free.

5. Software availability

<https://github.com/clnair-ascm/aqpet>.
 Program size: 7.19 MB.
 Dependency packages size: 300 MB.

For comprehensive guidance on the installation of R, along with associated packages including *aqpet*, one can refer to Venables and Smith (2009) and the official R project website at <http://www.r-project.org>. Note that the platform compatibility of R — it operates seamlessly across different operating systems, including Microsoft Windows, Linux, and Apple Mac. In the context of intricate data analyses, particularly those encompassing extensive datasets or sophisticated computational undertakings, it is recommended that *aqpet* be operationalized within

Linux environments. This is particularly salient for endeavours related to the construction of machine learning models or the execution of weather normalisation. Conversely, for functionalities like ASCM and visualisation, standard desktop computing environments are deemed adequately proficient.

CRedit authorship contribution statement

Yuqing Dai: Writing – original draft, Software, Methodology, Conceptualization. **Bowen Liu:** Writing – review & editing, Conceptualization. **Chengxu Tong:** Software, Data curation. **Zongbo Shi:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

I have shared the link to my data/code.

Acknowledgement

This work is funded by Wellcome Trust (Ref: 227150/Z/23/Z); University of Birmingham Institute of Global Innovation Clean Air Theme; Natural Environment Research Council WM-Air (Ref: NE/S003487/1).

References

- Abadie, A., 2021. Abadie Using synthetic controls: Feasibility, data requirements, and methodological aspects. *J. Econ. Lit.* 59 (2), 391–425. <https://doi.org/10.1257/jel.20191450>.
- Abadie, A., Diamond, J., 2010. Hainmueller Synthetic control methods for comparative case studies: Estimating the effect of California's tobacco control program. *J. Am. Stat. Assoc.* 105 (490), 493–505. <https://doi.org/10.1198/jasa.2009.ap08746>.
- Aguilar-Gomez, S., Dwyer, H., Graff Zivin, J., Neidell, M., 2022. This is air: the "nonhealth" effects of air pollution. *Ann. Rev. Resour. Econ.* 14, 403–425. <https://doi.org/10.1146/annurev-resource-111820-021816>.
- Anenberg, S.C., Haines, S., Wang, E., Nassikas, N., Kinney, P.L., 2020. Synergistic health effects of air pollution, temperature, and pollen exposure: a systematic review of epidemiological evidence. *Environ. Health* 19, 1–19. <https://doi.org/10.1186/s12940-020-00681-z>.
- Angrist, J.D., Pischke, J.S., 2009. *Mostly Harmless Econometrics: an Empiricist's Companion*. Princeton university press.
- Athey, S., 2015. Machine learning and causal inference for policy evaluation. *Proc. 21st ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.* 5–6. <https://doi.org/10.1145/2783258.2785466>.
- Athey, S., Imbens, G.W., 2017. The state of applied econometrics: Causality and policy evaluation. *J. Econ. Perspect.* 31 (2), 3–32. <https://doi.org/10.1257/jep.31.2.3>.
- Ben-Michael, E., Feller, A., Rothstein, J., 2021. The augmented synthetic control method. *J. Am. Stat. Assoc.* 116 (536), 1789–1803. <https://doi.org/10.1080/01621459.2021.1929245>.
- Bigi, A., Harrison, R.M., 2010. Analysis of the air pollution climate at a central urban background site. *Atmos. Environ.* 44 (16), 2004–2012. <https://doi.org/10.1016/j.atmosenv.2010.02.028>.
- Borghesi, S., Castellini, M., Comincioli, N., Donadelli, M., Gufler, I., Vergalli, S., 2022. European green policy announcements and sectoral stock returns. *Energy Pol.* 166, 113004. <https://doi.org/10.1016/j.enpol.2022.113004>.
- Botosaru, I., Ferman, B., 2019. On the role of covariates in the synthetic control method. *Economet. J.* 22 (2), 117–130. <https://doi.org/10.1093/ectj/utz001>.
- Carslaw, D.C., Ropkins, K., 2012. Openair—an R package for air quality data analysis. *Environ. Model. Software* 27, 52–61. <https://doi.org/10.1016/j.envsoft.2011.09.008>.
- Ceballos-Santos, S., González-Pardo, J., Carslaw, D.C., Santurtún, A., Santibáñez, M., Fernández-Olmo, I., 2021. Meteorological normalisation using boosted regression trees to estimate the impact of COVID-19 restrictions on air quality levels. *Int. J. Environ. Res. Public Health* 18 (24), 13347. <https://doi.org/10.3390/ijerph182413347>.
- Cole, M.A., Elliott, R.J., Liu, B., 2020. The impact of the Wuhan Covid-19 lockdown on air pollution and health: a machine learning and augmented synthetic control approach. *Environ. Resour. Econ.* 76 (4), 553–580. <https://doi.org/10.1007/s10640-020-00483-4>.

- Cory-Slechta, D.A., Sobolewski, M., 2023. Neurotoxic effects of air pollution: an urgent public health concern. *Nat. Rev. Neurosci.* 24 (3), 129–130. <https://doi.org/10.1038/s41583-022-00672-8>.
- Council, N.R., 2007. *Models in Environmental Regulatory Decision Making*. National Academies Press.
- Dai, Q., Hou, L., Liu, B., Zhang, Y., Song, C., Shi, Z., Hopke, P.K., Feng, Y., 2021. Spring Festival and COVID-19 lockdown: disentangling PM sources in major Chinese cities. *Geophys. Res. Lett.* 48 (11) <https://doi.org/10.1029/2021GL093403> e2021GL093403.
- Díaz, J.J., Mura, I., Franco, J.F., Akhavan-Tabatabaeis, R., 2021. aiRe-A web-based R application for simple, accessible and repeatable analysis of urban air quality data. *Environ. Model. Software* 138, 104976. <https://doi.org/10.1016/j.envsoft.2021.104976>.
- Doudchenko, N., Imbens, G.W., 2016. Balancing, regression, difference-in-differences and synthetic control methods: a synthesis (No. w22791). NBER. <https://doi.org/10.3386/w22791>.
- Erdman, C., Emerson, J.W., 2008. bcp: an R package for performing a Bayesian analysis of change point problems. *J. Stat. Software* 23, 1–13. <https://doi.org/10.18637/jss.v023.i03>.
- Erickson, R.A., Engelstad, P.S., Jarnevich, C.S., Sofaer, H.R., Daniel, M., 2022. Climate matching with the climatchR R package. *Environ. Model. Software* 157, 105510. <https://doi.org/10.1016/j.envsoft.2022.105510>.
- Falocchi, M., Zardi, D., Giovannini, L., 2021. Meteorological normalization of NO₂ concentrations in the Province of Bolzano (Italian Alps). *Atmos. Environ.* 246, 118048 <https://doi.org/10.1016/j.atmosenv.2020.118048>.
- Feenstra, B., Collier-Oxandale, A., Papapostolou, V., Cocker, D., Polidori, A., 2020. The AirSensor open-source R-package and DataViewer web application for interpreting community data collected by low-cost sensor networks. *Environ. Model. Softw.* 134, 104832 <https://doi.org/10.1016/j.envsoft.2020.104832>.
- Getz, D., Page, S.J., 2019. *Event Studies: Theory, Research and Policy for Planned Events*. Routledge. <https://doi.org/10.4324/9780429023002>.
- Grange, S.K., Carslaw, D.C., 2019. Using meteorological normalisation to detect interventions in air quality time series. *Sci. Total Environ.* 653, 578–588. <https://doi.org/10.1016/j.scitotenv.2018.10.344>.
- Grange, S.K., Carslaw, D.C., Lewis, A.C., Boleti, E., Hueglin, C., 2018. Random forest meteorological normalisation models for Swiss PM₁₀ trend analysis. *Atmos. Chem. Phys.* 18 (9), 6223–6239. <https://doi.org/10.5194/acp-18-6223-2018>.
- Imbens, G.W., Wooldridge, J.M., 2009. Recent developments in the econometrics of program evaluation. *J. Econ. Lit.* 47 (1), 5–86. <https://doi.org/10.1257/jel.47.1.5>.
- Kamara, A.A., Harrison, R.M., 2021. Analysis of the air pollution climate of a central urban roadside supersite: London, Marylebone Road. *Atmos. Environ.* 258, 118479 <https://doi.org/10.1016/j.atmosenv.2021.118479>.
- LeDell, E., Poirier, S., 2020. H2o automl: Scalable automatic machine learning. *Proceedings of the AutoML Workshop at ICML*. https://www.automl.org/wp-content/uploads/2020/07/AutoML_2020_paper_61.pdf.
- Leisch, F., 2002. Sweave: dynamic generation of statistical reports using literate data analysis. *Computat. Physica, Heidelberg* 575–580. https://doi.org/10.1007/978-3-642-57489-4_89.
- Liu, B., Bryson, J.R., Sevinc, D., Cole, M.A., Elliott, R.J., Bartington, S.E., Bloss, W.J., Shi, Z., 2023. Assessing the impacts of Birmingham's clean air zone on air quality: estimates from a machine learning and synthetic control approach. *Environ Resource Econ* 86, 203–231. <https://doi.org/10.1007/s10640-023-00794-2>.
- Liu, S., Zhao, Y., Chen, Z., Dou, M., Zhang, Q., Yang, W., 2021. Association between atrial fibrillation incidence and temperatures, wind scale and air quality: an exploratory study for Shanghai and kunming. *Sustainability* 13 (9), 1–18. <https://doi.org/10.3390/su13095247>.
- Lundberg, S.M., Erion, G., Chen, H., DeGrave, A., Prutkin, J.M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., Lee, S.-I., 2020. From local explanations to global understanding with explainable AI for trees. *Nat. Mach. Intell.* 2 (1), 56–67. <https://doi.org/10.1038/s42256-019-0138-9>.
- Manisalidis, I., Stavropoulou, E., Stavropoulos, A., Bezirtzoglou, E., 2020. Environmental and health impacts of air pollution: a review. *Front. Public Health* 8, 14. <https://doi.org/10.3389/fpubh.2020.00014>.
- Miller, D.L., 2023. An introductory Guide to event study models. *J. Econ. Perspect.* 37 (2), 203–230. <https://doi.org/10.1257/jep.37.2.203>.
- Nilsson, A., Bergquist, M., Schultz, W.P., 2017. Spillover effects in environmental behaviors, across time and context: a review and research agenda. *Environ. Educ. Res.* 23 (4), 573–589. <https://doi.org/10.1080/13504622.2016.1250148>.
- Rosenbaum, P.R., Rubin, D.B., 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika* 70 (1), 41–55. <https://doi.org/10.2307/2335942>.
- Rudin, C., 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat. Mach. Intell.* 1 (5), 206–215. <https://doi.org/10.1038/s42256-019-0048-x>.
- Shi, Z., Liu, B., Cheng, K., Elliot, R., Cole, M., Bryson, J.R., 2022. Quantifying the impact of clean air policy interventions for air quality management. *Project Report*. University of Birmingham. <https://doi.org/10.25500/epapers.bham.00004040>.
- Shi, Z., Song, C., Liu, B., Lu, G., Xu, J., Van Vu, T., Elliott, R.J., Li, W., Bloss, W.J., Harrison, R.M., 2021. Abrupt but smaller than expected changes in surface air quality attributable to COVID-19 lockdowns. *Sci. Adv.* 7 (3), eabd6696 <https://doi.org/10.1126/sciadv.abd6696>.
- Sievert, C., 2020. *Interactive Web-Based Data Visualization with R, Plotly, and Shiny*. CRC Press. <https://doi.org/10.1201/9780429447273>.
- Song, C., Liu, B., Cheng, K., Cole, M.A., Dai, Q., Elliott, R.J., Shi, Z., 2023. Attribution of air quality benefits to clean winter heating policies in China: combining machine learning with causal inference. *Environ. Sci. Technol.* 57 (46), 17707–17717. <https://doi.org/10.1021/acs.est.2c06800>.
- Tan, J., Yang, J., Wu, S., Chen, G., Zhao, J., 2021. A Critical Look at the Current Train/test Split in Machine Learning. <https://doi.org/10.48550/arXiv.2106.04525> arXiv preprint arXiv:2106.04525.
- Venables, W.N., Smith, D.M., 2009. R.D.C. Team. *An introduction to R*. <https://cran.r-project.org/doc/manuals/R-intro.pdf>. (Accessed 21 March 2024).
- Villanueva, R.A.M., Chen, Z.J., 2019. ggplot2: elegant graphics for data analysis. *Meas.: Interdiscip. Res. Perspect.* 17 (3), 160–167. <https://doi.org/10.1080/15366367.2019.1565254>.
- Vu, T.V., Shi, Z., Cheng, J., Zhang, Q., He, K., Wang, S., Harrison, R.M., 2019. Assessing the impact of clean air action on air quality trends in Beijing using a machine learning technique. *Atmos. Chem. Phys.* 19 (17), 11303–11314. <https://doi.org/10.5194/acp-19-11303-2019>.
- Willmott, C.J., Robeson, S.M., Matsuura, K., 2012. A refined index of model performance. *Int. J. Climatol.* 32 (13), 2088–2094. <https://doi.org/10.1002/joc.2419>.
- Wyche, K.P., Nichols, M., Parfitt, H., Beckett, P., Gregg, D.J., Smallbone, K.L., Monks, P. S., 2021. Changes in ambient air quality and atmospheric composition and reactivity in the South East of the UK as a result of the COVID-19 lockdown. *Sci. Total Environ.* 755, 142526 <https://doi.org/10.1016/j.scitotenv.2020.142526>.
- Xu, Y., 2017. Generalized synthetic control method: causal inference with interactive fixed effects models. *Political Anal.* 25 (1), 57–76. <https://doi.org/10.1017/pan.2016.2>.