



Hourly surface nitrogen dioxide retrieval from GEMS tropospheric vertical column densities: benefit of using time-contiguous input features for machine learning models

Janek Gödeke¹, Andreas Richter², Kezia Lange², Peter Maaß¹, Hyunkee Hong³, Hanlim Lee⁴, and Junsung Park⁴

¹Center for Industrial Mathematics, University of Bremen, Bremen, Germany

²Institute of Environmental Physics, University of Bremen, Bremen, Germany

³Environmental Satellite Center, National Institute of Environmental Research, Incheon, Republic of Korea

⁴Division of Earth Environmental System Science, Major of Spatial Information Engineering, Pukyong National University, Busan, Republic of Korea

Correspondence: Janek Gödeke (janek-goedeke@uni-bremen.de)

Received: 9 October 2024 – Discussion started: 4 November 2024

Revised: 9 April 2025 – Accepted: 22 May 2025 – Published: 11 August 2025

Abstract. Launched in 2020, the Korean Geostationary Environmental Monitoring Spectrometer (GEMS) is the first geostationary satellite mission for observing trace gas concentrations in the Earth's atmosphere. Observations are made over Asia. Geostationary orbits allow for hourly measurements, which lead to a much higher temporal resolution compared to daily measurements taken from low-Earth orbits, such as by the TROPOspheric Monitoring Instrument (TROPOMI) or the Ozone Monitoring Instrument (OMI). This work estimates the hourly concentration of surface nitrogen dioxide (NO₂) from GEMS tropospheric NO₂ vertical column densities (VCDs) and additional meteorological features, which serve as inputs for random forests and linear regression models. With several measurements per day, machine learning models can use not only current observations but also those from previous hours as inputs. We demonstrate that using these time-contiguous inputs leads to reliable improvements regarding all considered performance measures, such as Pearson correlation or mean square error. For random forests, the average performance gains are between 4.5 % and 7.5 %, depending on the performance measure. For linear regression models, average performance gains are between 7 % and 15 %. For performance evaluation, spatial cross-validation with surface in situ measurements is used to measure how well the trained models perform at locations where they have not received any training data. In other words, we inspect the models' ability to generalize to

unseen locations. Additionally, we investigate the influence of tropospheric NO₂ VCDs on the performance. The region of our study is South Korea.

1 Introduction

The concentration of nitrogen dioxide (NO₂) near the Earth's surface is of significant interest for several reasons. NO₂ is not only a precursor to health hazard and air pollutant ozone, but also a direct threat to human health. Moreover, it is linked to environmental issues such as acid rain; see, e.g., Jacob (2000).

At present, surface NO₂ is measured by networks of ground-based in situ monitoring stations. However, due to the limited number of such stations, they cannot provide global information about the surface NO₂ concentration. This limitation is one of the reasons why satellite remote sensing has become popular for deriving global estimates of surface NO₂. Satellites detect the fingerprint of NO₂ within the backscattered solar radiation due to its strong absorption of light in the wavelength range of 350–500 nm. One of the first studies on deriving surface NO₂ from remote sensing observations was conducted by Lamsal et al. (2008) across the USA and Canada. In their study, surface NO₂ was estimated by applying an assumed NO₂ vertical distribution calculated with a chemical transport model to tropospheric NO₂

vertical column densities (VCDs), where the tropospheric NO₂ VCDs were obtained from the Ozone Monitoring Instrument (OMI; Levelt et al., 2006). Numerous further studies followed, also utilizing chemical transport models and observations from satellites in low-Earth orbits. For example, we refer to the studies of Lamsal et al. (2010, 2013), Bechle et al. (2013), Wang and Chen (2013), Kharol et al. (2015), Geddes et al. (2016), Gu et al. (2017), and Cooper et al. (2020, 2022). Both OMI data and other observations have been considered, e.g., from the Global Ozone Monitoring Experiment (GOME; Burrows et al., 1999), the Scanning Imaging Absorption Spectrometer for Atmospheric Chartography (SCIAMACHY; Bovensmann et al., 1999), and the Tropospheric Monitoring Instrument (TROPOMI; Veefkind et al., 2012).

During the last 10 years, machine learning approaches have received increasing attention in determining surface NO₂ from satellite remote sensing observations. One advantage is the shorter computation time once the model has been trained. Diverse machine learning models have been used for this task, exploiting not only tropospheric NO₂ VCDs as input, but also additional input features to improve the model's performance, such as meteorological parameters, traffic density, or population information. Studies that consider observations from satellites in low-Earth orbits have been conducted by, for example, Kim et al. (2017), Jiang and Christakos (2018), de Hoogh et al. (2019), Chen et al. (2019), Di et al. (2020), Qin et al. (2020), Kim et al. (2021), Chan et al. (2021), Dou et al. (2021), Ghahremanloo et al. (2021), Li et al. (2022), Wei et al. (2022), Huang et al. (2023), and Shetty et al. (2024). For a detailed review on the methods used, the input features included, the regions of consideration, and the achieved performance, we refer to the work of Siddique et al. (2024).

Satellites in low-Earth orbits, such as OMI and TROPOMI, pass over the same region in middle and low latitudes once a day, which means they can provide at best one measurement per day and location. If the area is cloud-covered during the time of observation, the measurement of lower-tropospheric gases is not accurate, which makes the data coverage even more limited. Since satellites in low-Earth orbits provide observations at most once a day, most studies either predicted surface NO₂ at this specific satellite observation time (e.g., Kim et al., 2017) or estimated daily (e.g., Di et al., 2020), monthly, or annual averages of surface NO₂. Nevertheless, it should be mentioned that there are a few studies that have estimated hourly NO₂. As an example, Kim et al. (2021) linearly interpolated daily tropospheric NO₂ VCDs to an hourly resolution, from which they estimated hourly surface NO₂ concentrations over Switzerland and northern Italy.

In contrast, geostationary satellites permanently observe – more or less – the same region, leading to more data points for a given location that can be used for a prediction algorithm of surface NO₂. In particular, these larger datasets

make machine learning approaches even more attractive. The first geostationary satellite instrument for observing trace gas concentrations in the Earth's atmosphere is the Geostationary Environmental Monitoring Spectrometer (GEMS; Kim et al., 2020), which was launched in February 2020 by the Republic of Korea. It provides hourly measurements of radiances over 20 countries in Asia, including South Korea. Alongside GEMS, there exists only one other geostationary satellite that monitors trace gases, namely NASA's TEMPO, which was recently launched in April 2023 and is observing North America. A third geostationary satellite, ESA's Sentinel-4 mission, was launched in 2025 and monitors Europe.

Until now, only a few studies have been conducted on hourly surface NO₂ retrieval from geostationary observations: Zhang et al. (2023) presented a scientific GEMS NO₂ product (POMINO-GEMS), which empirically corrects for overestimation and stripe artifacts in the operational GEMS NO₂ product. They then converted their tropospheric NO₂ VCDs of 2021 over China to hourly surface NO₂ using a chemical transport model. Further studies that exploit machine learning approaches have been conducted over China. Yang et al. (2023b) used a random forest regressor to predict hourly surface NO₂ over China from GEMS radiance data at six wavelengths from the UV and visible bands, as well as some additional meteorological, temporal, and spatial features. Furthermore, a multi-output random forest was used to simultaneously predict five more air pollutants, such as ozone. Although prediction accuracy achieved by the multi-output model was slightly worse regarding surface NO₂, the overall training time for predicting all six pollutant concentrations was smaller. Ahmad et al. (2024) combined two machine learning models. First, a random forest was used to predict NO₂ mixing heights from meteorological input features. These were then fed into an extreme gradient boosting regressor, together with tropospheric NO₂ VCDs from GEMS, temporal variables, and meteorological variables. The study demonstrates the benefit of using NO₂ mixing height as input.

Hourly surface NO₂ has also been predicted from GEMS observations over South Korea, the region considered in this study. In the work of Lee et al. (2024), predictions were made for the whole year of 2022. Therein, the total amount of VCDs instead of tropospheric NO₂ VCDs was used as the only input of a (linear) mixed-effect model to predict surface NO₂. Their model is a piecewise-defined function whose output depends not only on the total column of NO₂, but also on the day and hour at which and region in which the prediction is to be made. For this, South Korea was divided into nine regions, which presumably leads to a more direct region-wise relationship between surface NO₂ and column densities of NO₂. In other words, implicitly, spatial and detailed temporal information is also exploited in their approach. This makes their model specialized to South Korea and the year 2022.

Another study that predicted surface NO₂ over South Korea was conducted by Tang et al. (2024). Therein, daily surface NO₂ concentrations instead of hourly surface NO₂ were predicted. Further, they did not use NO₂ column densities as input for a machine learning model. Instead, they inspected the influence of aerosol optical depth, which is part of the GEMS data products. Aerosol optical depth, together with surface NO₂ predictions from a chemical transport model and other features such as meteorological parameters, served as inputs for a random forest to estimate surface NO₂.

In order to train and evaluate machine learning models of surface NO₂, in situ NO₂ observations from ground-based networks are used. Within the literature, there are two frequently used strategies to evaluate the performance of a machine learning model in predicting surface NO₂. First, standard k -fold cross-validation is considered; see, for example, the works of Ghahremanloo et al. (2021), Chan et al. (2021), Yang et al. (2023b), and Ahmad et al. (2024). This means that the whole dataset is randomly split into k equally sized subsets. One of them serves as the test set, whereas the other $k - 1$ values are used to train the model. Training and testing are repeated k times, until each subset has served once as a test set. The average test performance (e.g., Pearson correlation) is calculated and represents the final evaluation of the model. For standard k -fold cross-validation, data from all available in situ stations are contained in both the training and the test datasets (with large probability). However, what if the trained model should afterwards predict surface NO₂ at a new location which has not contributed data to the training set? With the result from standard cross-validation, it would be impossible to say how reliable the model can generalize to this unseen location. It may have overfitted to the locations that it has dealt with during training. Therefore, if global charts covering large areas like the entirety of South Korea are desired, it would be more appropriate to evaluate the model's performance via so-called *spatial k -fold cross-validation*. This means the set of available in situ stations is divided into training and test stations, the model is trained with data from training stations only, and – finally – its performance in predicting surface NO₂ at the test stations is evaluated. Unsurprisingly, performance measured with spatial cross-validation is indeed worse compared to standard cross-validation, which has been observed, e.g., within the studies of Ghahremanloo et al. (2021), Chan et al. (2021), Yang et al. (2023b), and Tang et al. (2024). In our work we focus on spatial k -fold cross-validation, as we wish to inspect how well a model can generalize to unseen locations.

1.1 Goals of this study

Due to the hourly measurements GEMS provides over the same region, it is natural to ask whether one can benefit directly from the time resolution itself and not only from the resulting larger size of the dataset. Hence, we propose training a machine learning model φ that predicts surface NO₂ at

a given location z and time t not only from corresponding tropospheric NO₂ VCD and meteorological data at time t , but also from $(k - 1) \in \mathbb{N}_0$ previous hours (\mathbb{N}_0 denotes the set of natural numbers including zero). This means the model is a mapping $\varphi : \mathbb{R}^{pk} \rightarrow \mathbb{R}$, where p is the number of different features:

$$\text{input}(z, t) := \begin{pmatrix} \text{tropospheric NO}_2 \text{VCD}(z, t) \\ \vdots \\ \text{tropospheric NO}_2 \text{VCD}(z, t - k + 1) \\ \text{meteorological features}(z, t) \\ \vdots \\ \text{meteorological features}(z, t - k + 1) \end{pmatrix} \\ \mapsto \varphi(\text{input}(z, t)) \approx \text{surface NO}_2(z, t).$$

Here $t - j$ refers to the time j hours before t , where $j \in \{0, 1, \dots, k - 1\}$. In all that follows, k is also referred to as the *time contiguity* of the input features, as it determines how many times each input feature is included in the whole input vector. Note that $k = 1$ stands for the case in which only input features at current time t are included. Of course, one could also use features at later times $t + j$, but for simplicity and better readability, we focus on making predictions based on previous-time features in this work.

Our main aim is to inspect whether the performance of the model in predicting surface NO₂ at unseen locations will increase by using inputs with higher time contiguity k . Unseen locations are locations from which the model has not seen any training data. As it turns out, it is indeed beneficial to use larger time contiguity $k > 1$ for the machine learning models considered, namely random forests and linear regressors. To the best of our knowledge, this observation has not been made in the literature yet. Regarding work on non-geostationary satellite data, the usage of time-contiguous tropospheric NO₂ VCDs is simply impossible, as only single measurements per day are available. We further carefully design experiments that are suitable for answering our main research question about the benefit of time-contiguous inputs. Last but not least, we inspect the influence of tropospheric NO₂ VCDs on the models' ability to predict surface NO₂ and their influence on the benefit of using time-contiguous inputs. This is of interest as it addresses the question of how useful and necessary satellite observations of NO₂ are for the prediction of surface NO₂ concentrations.

1.2 Outline

In Sect. 2 we describe the different sources of data included in our study. Furthermore, we describe the construction of the datasets used for training machine learning models in our study and give a mathematical description of these datasets. Afterwards, in Sect. 3.2 we describe the experiments that provide clear insights into the research questions, e.g., whether time-contiguous inputs can enhance the quality of surface NO₂ predictions. We also discuss different loss

functions for measuring the performance of trained models on the test dataset. Section 4 serves as a quick recap of the machine learning models used in this study. Finally, we present and discuss the results of our experiments in Sect. 5.

2 Data

In our study, we exploit two data sources for the prediction of surface NO₂. The first source is tropospheric NO₂ VCDs derived from GEMS measurements, and the second is meteorological data from the ERA5 dataset (Hersbach et al., 2023). Further, measurements of surface NO₂ at in situ stations from the air quality network of South Korea serve as the ground truth in this study. This section begins with a brief description of these data sources, followed by a description of the data preprocessing steps. In particular, we explain how the VCDs were paired with ERA5 and in situ data and how time-contiguous datasets were constructed. For clarity, we provide mathematical definitions of these time-contiguous datasets.

2.1 Data sources

2.1.1 GEMS tropospheric NO₂ vertical column densities

GEMS is a UV–visible imaging spectrometer on board the geostationary satellite GK2B. At its launch on 18 February 2020, GEMS was the first geostationary air quality monitoring mission. GEMS is located over the Equator at a longitude of 128.2° E and covers a large part of Asia (5° S–45° N and 75–145° E) on an hourly basis. With four different scan modes, which all include South Korea, the field of regard (FOR) shifts westward with the Sun. During daytime, GEMS provides up to 10 observations over a given location according to the season and location, with a spatial resolution at Seoul of 3.5 km × 8 km. The GEMS irradiance and radiance measurements in the UV–visible spectral range can be used to derive column amounts of, for example, ozone (O₃), sulfur dioxide (SO₂), and NO₂, but also cloud and aerosol information (Kim et al., 2020). For this study, we use the tropospheric NO₂ VCD product.

During the time of this study, the operational GEMS L2 tropospheric NO₂ VCD product was available in v2. This version was evaluated by, e.g., Oak et al. (2024) and Lange et al. (2024), showing that it is high biased compared to the TROPOMI tropospheric NO₂ VCD product and ground-based tropospheric NO₂ VCD datasets. Additionally, the v2 product showed enhanced scatter. In preparation for the European geostationary instrument on Sentinel-4, the Institute of Environmental Physics at the University of Bremen (IUP-UB) has developed a scientific GEMS NO₂ product. The GEMS IUP-UB tropospheric NO₂ VCD v1.0 product was evaluated by Lange et al. (2024), showing good agreement with the operational TROPOMI NO₂ data and ground-based observations. Here, an earlier version (v0.9) of the same

data product was used. Briefly, the retrieval is based on a differential optical absorption spectroscopy fit in the 405–485 nm spectral window, using daily GEMS irradiances as background spectra. The stratospheric correction is based on a variant of the STREAM algorithm of Beirle et al. (2016), and tropospheric vertical columns are computed using air mass factors by applying the tropospheric NO₂ profiles from the TM5 model run performed for the operational TROPOMI product (Williams et al., 2017). The TM5 model has an hourly temporal resolution with a spatial resolution of 1° × 1°. As the model a priori is interpolated in space and time, no obvious structures from the coarse model resolution are visible in the data, but the lack of detail may still impact the results. Cloud screening is based on the operational GEMS cloud product v2 and a threshold of 50 % cloud radiance fraction, but no additional cloud correction is performed. Each pixel has a quality indicator (*qa* value) based on fitting residuals, cloud fraction, and surface properties. Here, only data with the highest *qa* value (good fits, cloud radiance fraction below 50 %, no snow or ice detected) are used.

Further, the GEMS IUP-UB product does not yet have full error propagation. The tropospheric NO₂ VCD error is therefore estimated to be 25 %. The main uncertainty results from the assumptions used in the calculation of air mass factors, in particular for surface reflectivity, the NO₂ vertical profile, and aerosol loading. Uncertainties are expected to be larger in the morning when the boundary layer is shallow and smaller around noon and in the evening. Uncertainties introduced by the stratospheric correction can be important over clean regions but can be neglected over pollution hotspots.

2.1.2 Meteorological data

In order to predict surface NO₂, it would not be sufficient to use tropospheric NO₂ VCDs as the only source of information. This is because VCDs represent integrals over the entire troposphere, capturing contributions from NO₂ at various altitudes, not just near the surface. A common strategy is to incorporate additional meteorological features into the prediction of surface NO₂; see for example the works of Di et al. (2020), Qin et al. (2020), Ghahremanloo et al. (2021), Chan et al. (2021), Li et al. (2022), and Yang et al. (2023b). In our study, we utilize meteorological features from the ERA5 dataset, the fifth-generation reanalysis by the European Centre for Medium-Range Weather Forecasts (ECMWF), which provides comprehensive global climate and weather data for the past 8 decades (Hersbach et al., 2023).

Our selection of meteorological features is partially inspired by the choices made in the aforementioned studies, including variables such as boundary layer height, wind components, surface temperature, or pressure. The 18 features from ERA5 that are considered in this study are listed in Table B1, where we use the same nomenclature as in the description of the ERA5 dataset; see again Hersbach et al.

(2023). In the geographical reference system, the resolution of all meteorological features is $0.25^\circ \times 0.25^\circ$, which corresponds to approximately $28 \text{ km} \times 22 \text{ km}$ over South Korea. Consequently, ERA5 data are approximately 8 times coarser in latitude and 3 times coarser in longitude than the GEMS tropospheric NO₂ VCDs.

2.1.3 In situ measurements of surface NO₂

In this study, we use in situ surface NO₂ measurements from the air quality network AirKorea as the ground truth, provided by the Korean Ministry of Environment (National Institute of Environmental Research (NIER), 2025). There is a large number of in situ stations in South Korea that, among other air-pollution-related species, measure surface NO₂. We used data from 637 stations, which are depicted in Fig. 1a. The instruments utilize the chemiluminescence method, as described by Kley and McFarland (1980). Our in situ dataset includes measurements from January 2021 until the end of November 2022, and we received the data in December 2022.

2.2 Pairing of data sources and data preprocessing

In the following, we explain the spatial and temporal pairing of the data sources. Tropospheric NO₂ VCDs and meteorological data possess spatial resolutions, as described in the previous section. Consequently, each data point covers an area (pixel) on the Earth's surface, rather than a single point. Here, we associated the location of an in situ station with the VCD pixel or meteorological pixel, whose center is nearest to the station's location (longitude, latitude). Note that the center of a VCD pixel coincides with the respective center of the GEMS satellite pixel, since no regridding is applied.

Tropospheric NO₂ VCDs are based on GEMS observations that have been collected within 30 min starting at a quarter to the respective hour, e.g., from 01:45 to 02:15 UTC. In situ measurements of surface NO₂ are available as hourly averages, starting on the hour. Temporally, we matched them with the VCDs using this timestamp and found that these data pairs showed the highest Pearson correlation. For example, VCDs between 01:45 and 02:15 UTC were matched with in situ measurements with a timestamp of 01:00 UTC. Unfortunately, at the end of our project, we learned that this was a misinterpretation of the in situ measuring times by 1 h, as the hourly averages actually start at the hour before the given timestamp instead of at the hour of the timestamp, as we had assumed. This means that the VCDs and surface NO₂ were not optimally matched within our experiments. However, the abovementioned correlation tests give us confidence that the conclusions of this study are not affected by this mistake, in particular with respect to the improvements in performance when adding data from other measurement times. To maintain consistency in notation, we continue to use the originally interpreted in situ measuring times, but they should be regarded as occurring 1 h earlier. Most meteorological features

are given on the hour, which means at a specific point in time. There is one exception, namely evaporation, which is available as an hourly average starting on the hour, similar to in situ measurements. Since the averages of these data sources are taken over different periods of time, there is not a unique way to pair them temporally. Our approach is the following.

Due to the hourly resolution of all data sources, time t is expressed by $t = \text{YYYY/MM/DD/HH}$ throughout this work. For example, $t = 2021/01/23/01$ refers to 23 January 2021 at 01:00 UTC. We associate the in situ measurements of surface NO₂, which started at time t and went on for 1 h, with t . In the example, time $t = 2021/01/23/01$ refers to surface NO₂ that has been averaged from 01:00 UTC until 02:00 UTC. Regarding tropospheric NO₂ VCDs, the same t refers to measurements that started 45 min later. Hence, $t = 2021/01/23/01$ describes the VCDs at a time between 01:45 and 02:15 UTC. Finally, for the meteorological features that are instantaneously on the hour, t stands for the feature's value 1 h later at $t + 1$. Thereby, it is closest to the corresponding VCD time frame. For example, $t = 2021/01/23/01$ is associated with the meteorological feature at 02:00 UTC.

To sum up, given a location z of an in situ station and a time $t = \text{YYYY/MM/DD/HH}$, we specified a single data point $\left(f(z, t), s(z, t)\right)$ that stores surface NO₂ $s(z, t)$ combined with the vector of input features $f(z, t)$, which consists of tropospheric NO₂ VCDs and meteorological features. As a data preprocessing step, we exclude data points that violate any of the following conditions:

1. All features are available at location z and time t (tropospheric NO₂ VCDs and surface NO₂ might be missing for a given z, t , for example, due to clouds).
2. Tropospheric NO₂ VCDs are non-negative. Negative VCDs can occur as a result of measurement noise in the satellite data or uncertainties in the stratospheric correction. We excluded them in an effort to improve the quality of the dataset. However, toward the end of the project, we tested the effect of this filter on a subset of the dataset and found only very small changes. This is probably due to the fact that applying this filter only leads to a reduction in the dataset by less than 0.5 %. Since negative VCDs are usually found over regions with low tropospheric NO₂ VCDs, the filter leads to a loss of the input variable and thus a loss of predictions for these regions. In retrospect, we can conclude that the implementation of this filter was not necessary, as it only had little influence on our dataset and can thus be neglected in future work. Regarding the random forests used in this study, which are trained on non-negative VCDs only, they are still able to make reasonable but potentially biased predictions over clean regions with negative VCDs as inputs. In this case, the random forests would treat negative VCDs as being

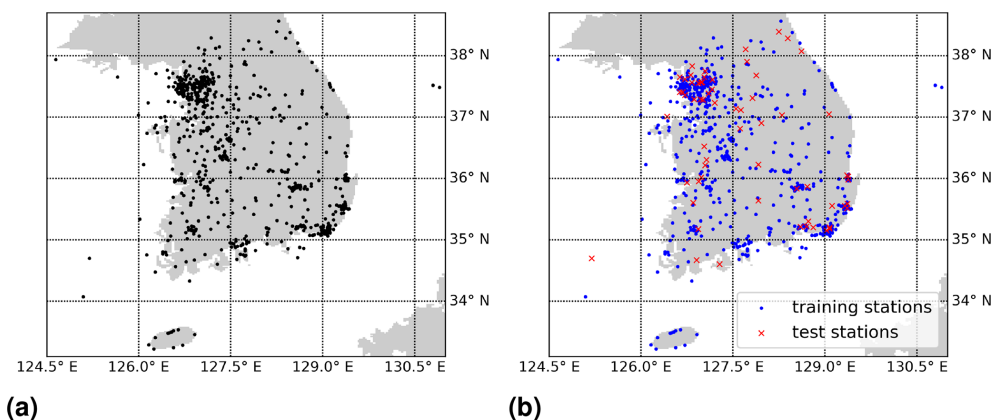


Figure 1. (a) Map with the 637 in situ stations from the air quality network of South Korea used in this study. (b) An exemplary split into 90 % training stations and 10 % test stations, considered during multiple 10-fold spatial cross-validations.

zero. In contrast to the VCDs, the in situ measurements of surface NO₂ are never negative.

3. The GEMS *qa* value is equal to 1. Therefore, the trained models presumably cannot make reliable predictions for scenarios where the *qa* value is smaller than 1. It would be an interesting future direction to examine the effects of lowering the threshold for the *qa* value. This would result in a larger but more complex dataset.

Data points $(f(z, t), s(z, t))$ that fulfill these conditions are collected within the so-called *data basis*. A data point in the data basis is not time contiguous, as it only provides information at a single time t and not at previous hours. The construction of time-contiguous datasets is described in the next section.

2.3 Description of time-contiguous datasets

In the Introduction, we motivate the use of time-contiguous inputs for machine learning models in order to predict surface NO₂. For better clarity, we introduce notations and definitions in a mathematical form.

2.3.1 Spatial and temporal coordinates

Z is the set of positions (longitude, latitude) on the Earth's surface in terms of longitude and latitude. Hence, it can be seen as the Cartesian product $[-180, 180) \times [-90, 90)$. In this study, we deal with in situ stations in South Korea which are located within $[124, 131) \times [33, 39)$; see Fig. 1a. These stations are simply identified with their location $z \in Z$ in what follows.

T is the set of all measuring times YYYY/MM/DD/HH between January 2021 and November 2022. For example, 2021/01/23/01 refers to 23 January 2021 at 01:00 UTC. Note that for a given $t \in T$, the expression $t - j$ for $j \in \mathbb{N}$ stands for the time j hours before t . For example, for $t = 2021/01/23/01$ and $j = 3$, it is $t - j = 2021/01/22/22$.

2.3.2 Surface NO₂ and input features

Recall from the previous section that surface NO₂ measured at time $t \in T$ and at in situ station $z \in Z$ is denoted by $s(z, t)$. As already mentioned, surface NO₂ is to be predicted from the tropospheric NO₂ VCD and meteorological variables such as the boundary layer height. These input features at $z \in Z$ and $t \in T$ are denoted by $f_1(z, t), \dots, f_p(z, t)$, where $p \in \mathbb{N}$ is the number of considered features (determined by some feature selection procedure; see Sect. 3.1). At this point, it is only important that f_1 denotes the VCDs. For simplicity, we just write $f(z, t) \in \mathbb{R}^p$ for the vector of all features at location z and time t .

2.3.3 Data preprocessing

We review the data preprocessing described in the previous section in light of the mathematical notation. A measurement $f_1(z, t)$ of a tropospheric NO₂ VCD is *valid* if it exists (measurements may be missing at some times $t \in T$), if $f_1(z, t) \geq 0$, and further if the GEMS *qa* value is equal to 1. For all other features $f_2(z, t), \dots, f_p(z, t)$ and surface NO₂ $s(z, t)$, it suffices that the measurement exists in order to be categorized as valid. Note again that in situ measurements of surface NO₂ are always non-negative in the present dataset.

In the following, we collect all locations and times (z, t) at which we have access to valid measurements. Namely, the *domain of valid measurements* Ω is defined as

$$\Omega = \{(z, t) \in Z \times T : \text{and } s(z, t), f_1(z, t), \dots, f_p(z, t) \text{ are valid}\}. \quad (1)$$

2.3.4 Time-contiguous datasets

In order to consider time-contiguous measurements, we define for $N \in \mathbb{N}$ the set

$$\Omega_N = \{(z, t) \in \Omega : (z, t - j) \in \Omega \text{ for } j = 1, \dots, N - 1\}. \quad (2)$$

In other words, Ω_N collects locations and times (z, t) at which valid measurements also exist for at least $N - 1$ pre-

vious hours. Note that $\Omega_N \subseteq \Omega_{N-1} \subseteq \Omega$ for all $N \in \mathbb{N}$, and Ω_1 coincides with Ω , the domain of valid measurements. Given $(z, t) \in \Omega_N$ and $k \in \{1, \dots, N\}$, this definition allows us to build a valid time-contiguous feature vector:

$$\begin{pmatrix} f(z, t) \\ f(z, t-1) \\ \vdots \\ f(z, t-k+1) \end{pmatrix} \in \mathbb{R}^{pk}, \quad (3)$$

which can serve as input for a machine learning model $\varphi_\theta : \mathbb{R}^{pk} \rightarrow \mathbb{R}$ to predict surface NO₂ $s(z, t)$.

Hence, Ω_N parameterizes the datasets occurring in our study. In fact, Ω_N parameterizes N different datasets of feature vectors paired with surface NO₂. They only differ within the time contiguity $k \in \{1, \dots, N\}$ of the feature vectors, that is, how many previous hours (namely $k-1$) are considered for each feature (at most $N-1$). Mathematically, these N datasets can be understood as functions $D_{N,k} : \Omega_N \rightarrow \mathbb{R}^{pk} \times \mathbb{R}$ mapping $(z, t) \in \Omega_N$ to the feature vector in Eq. (3) paired with surface NO₂ at location z and measuring time t . Further, $D_{1,1}$ just describes the *data basis* mentioned in the previous section.

The number of elements in Ω_N – that is, the size of all datasets $D_{N,k}$ – are listed in Table 1 for $N = 1, \dots, 5$. Hence, if a model is to be trained with time-contiguous inputs ($k > 1$), this comes with the price of a smaller number of data points. For example, time-contiguous models cannot be used to make predictions at initial hours of a day. It should be mentioned that among all features described in the previous section, ERA5's *soil type* and *high vegetation cover* are the only features that do not depend on time t . This is why, in practice, we never included them k times but rather a single time only, when building the time-contiguous feature vector in Eq. (3) at (z, t) . However, for the sake of simplicity, we neglect this fact within the notation.

2.3.5 Normalization of input features

For any given split into training and test data, the input features are normalized before being fed into the machine learning models to improve the stability of their performance. More precisely, each feature undergoes an affine transformation A such that its mean on the training data becomes 0 and its standard deviation becomes 1. Let \bar{x}_{train} and σ_{train} be the mean and standard deviation of a feature in the training data, respectively. Then, the transformation applied to both training and test data points is given by

$$A(x) = \frac{x - \bar{x}_{\text{train}}}{\sigma_{\text{train}}} \quad (4)$$

and is applied to both training and test data points.

A compact overview of the spatial and temporal resolutions of the data sources used is shown in Table 2. In addition, for each data source, the applied data preprocessing steps are

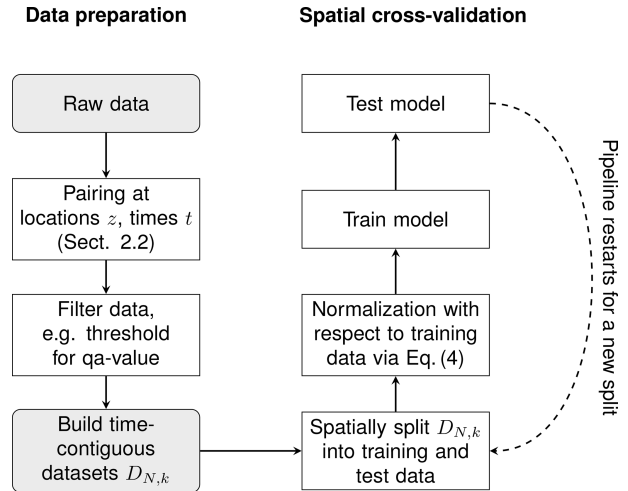


Figure 2. A flowchart for all data processing steps. The left column shows the construction of the time-contiguous datasets $D_{N,k}$. For preprocessing, the data are filtered according to the criteria in Sect. 2.2; see also Table 2. Evaluating the performance of models on $D_{N,k}$ is done via spatial cross-validation; see Sect. 3.2. This pipeline is outlined in the right column.

listed. Moreover, the overall workflow for all data-processing steps is illustrated in the flowchart in Fig. 2.

3 Experimental setup

In Sect. 3.2, we describe and discuss the experiments conducted to inspect our main research questions. Before that, we explain how features were selected for these experiments. Afterwards, we discuss different performance measures and loss functions used to evaluate the quality of the models' prediction of surface NO₂ on test data points.

3.1 Feature selection

In this study, we considered 23 different features from which we selected 17 to build the feature vectors used in Eq. (3) as inputs for the machine learning models. The selected and excluded features are listed in Table B1 and are used in Experiment 1 and Experiment 2; see Sect. 3.2. For the feature selection, we proceeded as follows: on the data basis $D_{1,1}$, we considered 200 different splits into 90 % training and 10 % test stations. For the training data of each split, we calculated the Pearson correlation (see Sect. 3.3 for a definition) between in situ measurements of surface NO₂ and the respective feature. We selected features which had an absolute mean correlation larger than 0.1. It is worth mentioning that for all 17 of the aforementioned features, the correlation was in fact larger than 0.1 in 98 % of the splits, whereas this was never the case for the remaining six features. More complex feature selection strategies could be applied in the future. However, during this study we focus on the benefits

Table 1. Size of time-contiguous datasets $D_{N,k}$, which consist of data points for which valid measurements also exist for at least $N - 1$ previous hours, but only k values are used for constructing the time-contiguous feature vector in Eq. (3). Note that the size is independent of the time contiguity k . The overall considered time period covers January 2021 until November 2022.

N	1	2	3	4	5
Number of data points	1 341 642	959 458	699 777	505 719	356 117

Table 2. Overview of spatial and temporal resolutions of the data sources used. Applied preprocessing steps are also listed for each data source.

	NO ₂ VCDs	Surface NO ₂	ERA5 features
Spatial resolution	3.5 km × 8 km (latitude × longitude)	Local measurements	28 km × 22 km (latitude × longitude)
Temporal resolution	One measurement per hour and location	Hourly averages	One measurement per hour and location*
Preprocessing	Missing values removed Negative values removed Threshold qa value: 1	Missing values removed (No negative values exist)	(No missing values exist)
Preprocessing during cross-validation	Normalization via Eq. (4)	Normalization via Eq. (4)	Normalization via Eq. (4)

* Exception: ERA5 evaporation is available as hourly averages.

of time-contiguous inputs and not on the optimal choice of input features.

3.2 Experiments

Recall from Sect. 2.3 that Ω_N is the set of locations and measuring times (z, t) at which all measurements are also available at $(N - 1)$ previous hours. Note that Ω_N does not parameterize a single dataset but N different datasets $D_{N,k} : \Omega_N \rightarrow \mathbb{R}^{p_k} \times \mathbb{R}$ via

$$D_{N,k} : (z, t) \mapsto \left(\begin{pmatrix} f(z, t) \\ f(z, t - 1) \\ \vdots \\ f(z, t - k + 1) \end{pmatrix}, s(z, t) \right),$$

which only differ in the time contiguity $k \in \{1, 2, \dots, N\}$ of the time-contiguous feature vector $(f(z, t), \dots, f(z, t - k + 1))^T$, defined in Eq. (3).

As mentioned in the Introduction, we wish to inspect how well a machine learning model is able to make predictions of surface NO₂ at locations from which it has not seen training data. This is why we use multiple (six-times) 10-fold spatial cross-validations in all experiments. This involves splitting the dataset 60 times randomly into 90 % training and 10 % test data based on the locations of the in situ stations; see Fig. 1b for a visualization of a single split. Performance is measured on all the different test datasets and averaged. Due to the limited number of available in situ stations, significant variance in the model’s performance is expected across different splits. Therefore, multiple 10-fold

spatial cross-validations provide a more reliable estimate of the model’s performance compared to a single 10-fold spatial cross-validation. In all that follows, whenever it is mentioned that a machine learning model is trained or tested on $D_{N,k}$, it implies that the model is trained or tested solely on those data points in $D_{N,k}$ corresponding to the designated training or test stations. Note that for fixed N , surface NO₂ that is to be predicted in $D_{N,k}$ is exactly the same for all the different k . Furthermore, for all models, the same 60 splits into training and test stations are considered for spatial cross-validation, which ensures perfect comparability. For a basic outline of a cross-validation pipeline, see Fig. 2.

Let us recall from Sect. 1.1 that our main research question is whether time-contiguous inputs for machine learning models enable higher accuracy for predicting surface NO₂. We propose two experiments to gain insight into this question.

Experiment 1. Do time-contiguous input features provide additional information?

For fixed N , consider the datasets $D_{N,k}$ for different time-contiguities $k = 1, \dots, N$. The chosen machine learning model, such as a random forest regressor, is trained and tested on $D_{N,k}$ for all 60 splits from spatial cross-validation. A comparison is made with respect to different k . Fixing N ensures that, regardless of k , the same ground truth (surface NO₂) is predicted for computing the cross-validation scores on the test sets. Additionally, all models are trained with the same number of training data points, eliminating any advantage

or disadvantage due to differing dataset sizes. Thus, this experiment provides pure insights into the information gain provided by time-contiguous inputs. We conduct this experiment for all $N \in \{2, 3, 4, 5\}$.

Experiment 2. Are time-contiguous input features beneficial in spite of a smaller available dataset?

In the first experiment, the models were trained on the same amount of training data, with the time contiguity k being the only variable. However, for smaller k there is much more data available that can be used for training the respective models; see Table 1. Therefore, we need to extend the first experiment as follows: we still test performance on $D_{N,k}$ for a fixed N . But for different k , we train models on $D_{M,k}$ for all $M \in \{k, k+1, \dots, 5\}$, i.e., with a different amount of training data. Note that in Experiment 1, M has always been set to N . These additional investigations are crucial to evaluate whether time-contiguous inputs are beneficial for predicting surface NO₂. Even if time-contiguous inputs provide additional information (as seen in the first experiment), why should one use them if training with less or even no time contiguity on larger datasets yielded better results? Again, we conduct this experiment for all $N \in \{2, 3, 4, 5\}$, where N determines the test datasets.

In a third experiment, we analyze the influence of some features on the performance of the machine learning models. Since testing all the different combinations of input features for all 15 different training and test cases in Experiment 2 would be out of the scope of this study, we focus only on the influence of the tropospheric NO₂ VCDs, surface height, and latitude. Note that longitude has not been included during feature selection due to a low correlation with surface NO₂. Tropospheric NO₂ VCDs are the main consideration within this third experiment since they represent the feature which shows, among all considered input features, by far the best Pearson correlation with surface measurements of NO₂, namely around 0.626; see also Table B1. Although latitude only has a small variation over South Korea and hence a presumably small impact on predicting surface NO₂, we considered it (and also longitude) during feature selection to check whether it provides some helpful information. Other studies have also used spatial coordinates to predict surface NO₂, mainly over large regions (Ghahremanloo et al., 2021; Li et al., 2022; Qin et al., 2020) but also over smaller regions, such as over Switzerland (de Hoogh et al., 2019). Using spatial coordinates as inputs for a model, however, carries the risk of spatial overfitting, which could make it more difficult to predict surface NO₂ outside of South Korea with the same model. This is why we inspect whether the models perform equally well over South Korea without having latitude and surface height as inputs.

Experiment 3. What is the influence of tropospheric NO₂ VCDs, latitude, and surface height on the performance?

We compare four different settings of input features:

Setting 1. All features selected in Sect. 3.1 are included, which is exactly the same setup as for Experiments 1 and 2.

Setting 2. VCDs are excluded as an input feature.

Setting 3. Latitude and surface height are excluded.

Setting 4. VCDs, latitude, and surface height are excluded.

We also conduct Experiment 2 for Settings 2, 3, and 4 and draw a comparison between these settings regarding different performance measures. Further, within these four settings, we inspect the models' ability and reliability in achieving performance gains when including time-contiguous input features.

3.3 Performance measures

Throughout this section, $x^\dagger \in \mathbb{R}^n$ is a vector consisting of n in situ observations of surface NO₂, where each coefficient $x_i^\dagger(t_i, z_i) = s(t_i, z_i)$ corresponds to a measurement that has been taken at a given time t_i and location (longitude, latitude) z_i of a given in situ station. For the sake of simpler notation, we just write x_i^\dagger , neglecting the dependence on t_i and z_i within the notation. Similarly, $x \in \mathbb{R}^n$ denotes the predictions for x^\dagger made by a machine learning model, such as linear regression or random forests. In the following, we discuss different performance measures that quantify the gap between the model's prediction x for x^\dagger , the observed surface concentration of NO₂.

As pointed out in the Introduction, spatial cross-validation is considered within this research; i.e., data are split into training and test data station-wise. Since the overall number of in situ stations is relatively small, namely 637, the statistical properties of surface NO₂ for different test sets are very likely to differ. In particular, the mean or standard deviation of surface NO₂ of different test sets will vary. Hence, in order to compare the quality of surface NO₂ predictions on different test sets, it is reasonable to use error measures that are more robust or even insensitive to different data distributions.

In order to ensure better comparability of performances of a model on different test sets, one should not use absolute performance measures such as the mean absolute error or root mean square error, since they depend on the scale of the different test sets.

At first glance, it seems reasonable to consider the mean percentage error:

$$\text{MPE}(x^\dagger, x) = \sum_{i=1}^n \frac{|x_i^\dagger - x_i|}{|x_i^\dagger|}.$$

The reason why the mean percentage error enables us to compare performances on different test sets is the following property: for every $c \in \mathbb{R}^n$ with $c_i \neq 0$ it holds that

$$\text{MPE}(cx^\dagger, cx) = \text{MPE}(x^\dagger, x),$$

where cx^\dagger denotes pointwise multiplication. However, since many in situ measurements x_i^\dagger are very close to or equal to zero, the mean percentage error becomes unstable. As a trade-off, we consider performance measures $E(x^\dagger, x)$ that are *scale-insensitive*; i.e., for every $\lambda \in \mathbb{R} \setminus \{0\}$ it holds that

$$E(\lambda x^\dagger, \lambda x) = E(x^\dagger, x).$$

The normalized mean absolute error (NMAE) can be written as

$$\text{NMAE}(x^\dagger, x) = \frac{\sum_{i=1}^n |x_i^\dagger - x_i|}{\sum_{i=1}^n |x_i^\dagger|},$$

so the NMAE is just the mean absolute error divided by the mean absolute value of the ground truth x^\dagger . If normalization by the standard deviation of x^\dagger instead of its mean were considered, this would lead to a measure similar to the coefficient of determination R^2 ; see Appendix A. Note that in contrast to the mean absolute error, NMAE is scale-insensitive. Similarly, we define the normalized mean square error (NMSE) as

$$\text{NMSE}(x^\dagger, x) = \frac{\sum_{i=1}^n |x_i^\dagger - x_i|^2}{\sum_{i=1}^n |x_i^\dagger|^2}.$$

Whenever we talk about the correlation between x^\dagger and x , we mean the Pearson correlation coefficient (C), which is defined as

$$C(x^\dagger, x) = \frac{\text{cov}(x^\dagger, x)}{\sigma(x^\dagger)\sigma(x)},$$

where $\text{cov}(x^\dagger, x)$ denotes the covariance between x^\dagger and x and $\sigma(x^\dagger)$, and $\sigma(x)$ is the standard deviation of x^\dagger and x , respectively. It should be noted that this is not a performance measure in the sense that $x^\dagger = x$ if and only if $C(x^\dagger, x) = 1$. Nevertheless, it quantifies the linear relationship between x and x^\dagger . Furthermore, it is frequently used in the literature, which is the reason why we consider it in our work, too.

We considered two further scale-insensitive performance measures, the coefficient of determination (R^2) and the index of agreement (IOA), which are defined in Appendix A.

4 Machine learning models considered

As mentioned in the Introduction, numerous machine learning models have been considered for predicting surface NO₂ in the literature. Examining the benefit of time-contiguous input features for all the different models is beyond the scope of this research. This is because fair comparisons require individual hyperparameter tuning for the models, with different time contiguities of the input features. Therefore, we restrict our attention to one approach that, on the one hand, has performed well in the literature and, on the other hand, does not have many hyperparameters to tune. If there were many hyperparameters to be tuned and the models' performance were very sensitive to the choice of these hyperparameters, there would be a risk that better performance was achieved only due to better hyperparameter tuning. In this study, we use a random forest regressor, which we describe in Sect. 4.2, and present the selected hyperparameters. As a reference, we consider a simple linear regression approach, which we recap first in the next section. At the outset of this study, we also experimented with neural networks (NNs) to estimate surface NO₂. While we observed similar results to those obtained with random forests, the training time for NNs was considerably longer. Therefore, and due to the large number of hyperparameters and architectural design choices for NNs, conducting as many experiments with NNs as we did with random forests would have been outside the scope of our study. This is why we chose to focus on random forests, but we expect similar performance gains for neural networks as well.

4.1 Linear regression

Although it has already been shown, e.g., by Ghahremanloo et al. (2021), that linear regression models are not the best for predicting surface NO₂, we consider an ordinary least squared regressor as a reference in our study, mainly because it has no tunable hyperparameters, such as regularization parameters, or architecture parameters like those in neural networks (e.g., number of layers, width of layers, activation functions, skip connections). Thus, it provides a clear view on the question of whether time-contiguous inputs are beneficial for this linear regression model. During this study, we used the ordinary least squares regression model provided by the Python scikit-learn package (version 1.2.2, Pedregosa et al., 2011). In our case of predicting surface NO₂ from time-contiguous inputs, the linear regression model is a parameterized function

$$\begin{aligned} \varphi_\theta : \mathbb{R}^{pk} &\longrightarrow \mathbb{R} \\ y &\longmapsto Ay + b, \end{aligned}$$

where $y = (f(z, t), \dots, f(z, t-k+1))^T$ is a (time-contiguous) feature vector defined in Eq. (3), A is a $1 \times pk$ matrix, and $b \in \mathbb{R}$ is a bias term. Let $(y_n, s_n)_{n=1}^N$ be training data, where

y_n is a feature vector at location z_n and time t_n and s_n the corresponding in situ measurement of surface NO₂ at time t_n . Training φ_θ then means to search for a parameter $\theta = (A, b)$ that solves the following minimization problem:

$$\min_{\theta} \sum_n |\varphi_\theta(y_n) - s_n|^2.$$

We choose to minimize the squared error since the computation time is much shorter compared to that of other losses such as the absolute error.

4.2 Random forests

There are two main reasons why random forests, a machine learning model originally proposed by Breiman (2001), are considered within this research. First, they have already proven to be powerful for predicting surface NO₂ in various studies; see, for example, Di et al. (2020), Ghahremanloo et al. (2021), Li et al. (2022), and Huang et al. (2023) on OMI and TROPOMI data and Yang et al. (2023b) on GEMS data. Second, the studies of Probst et al. (2018, 2019) suggest that random forests are less tunable compared to other machine learning approaches. “Tunable” is defined as the extent to which the performance of a random forest with typical default hyperparameters can be enhanced by adjusting (tuning) those hyperparameters. As discussed before, this reduces the risk of drawing incorrect conclusions about the benefit of using time-contiguous inputs.

In fact, according to Probst et al. (2018), there are mainly four hyperparameters that empirically determine the performance of a random forest:

- The first hyperparameter is the number of randomly drawn features considered at every split of a tree. In the Python scikit-learn software package (version 1.2.2, Pedregosa et al., 2011) that we use for this study, it is called `max_features`. However, in several other software packages, it is denoted as `mtry`.
- The second hyperparameter is the number of trees that make up the random forest. In scikit-learn it is called `n_estimators`. To be precise, it is not actually a hyperparameter, since more trees are in general more advantageous; see, e.g., Genuer et al. (2008) or Scornet (2017).
- The third hyperparameter is the maximal number of (randomly drawn) data samples from the training set that is used for the construction of an individual tree, denoted as `max_samples` in scikit-learn.
- The fourth hyperparameter is the minimal number of observations that lands in a leaf node during the training process. In scikit-learn it is called `min_samples_leaf`.

In their experiments, Probst et al. (2018) observed that `max_features` had the biggest influence on the performance and the influence of `max_samples` and `min_samples_leaf` was smaller. This is why, during hyperparameter tuning, we mainly focus on `max_features` but also consider different values for `max_samples`. Regarding `max_samples`, we consider values between 50 % and 100 % of the size of the training dataset. On the other hand, for `max_features`, values between 1 and $(pk)/3$ are considered, where pk is the number of inputs for the model, i.e., the dimension of the time-contiguous feature vector in Eq. (3). The value $(pk)/3$ is the default value of scikit-learn. Genuer et al. (2008) suggested \sqrt{pk} for problems in which the number of data points is much larger than the number of input features pk , which is clearly the case in our study (hundreds of thousands of data points versus less than 90 input features). As $pk \geq 17$, the value \sqrt{pk} is always within the considered interval during optimization. In fact, \sqrt{pk} turns out to be quite close to the optimal choice in our hyperparameter study. Regarding `min_samples_leaf`, we inspect two typical default values, namely 1 and 5. Following the rule “the more, the better” for the number of trees (`n_estimators`) in the forest, we use 8000 trees while tuning the other hyperparameters. Hyperparameter selection is made according to the spatially cross-validated (10 splits) NMSE, leading to `max_features` = 2, 3, 3, 3, 4 for time contiguity $k = 1, 2, 3, 4, 5$ and further `min_samples_leaf` and `max_samples` = 5 using 100 % of the size of the training data. All remaining hyperparameters are always set to the default values within scikit-learn.

With 8000 trees, we chose a very high value for the number of trees, which may require an explanation. The good news is given first: comparable results can be obtained with far fewer trees in the forest. However, for hyperparameter tuning and to gain a clearer insight into the benefit of time-contiguous features, it is reasonable to choose a large number of trees, which we illustrate in the following: the random forest algorithm in scikit-learn is not deterministic, meaning that if the model is trained on the same training data multiple times, the trained forests will differ from each other, also causing the performance of the respective test dataset to vary. However, we observe that with a higher number of trees in the forest, the variance in the performance decreases for all considered performance measures. In Fig. C1 in Appendix C, we illustrate this effect using a single split into training and test stations. Two random forests, one with 30 trees and the other with 8000 trees, are each trained and tested 20 times on the same data, similar to Experiment 2, but with 20 repetitions of the same split instead of 60 different splits. We observe that with 30 trees the scores on the test data, such as Pearson correlation, NMSE, or NMAE, exhibit some variance. In contrast, there is barely any variance in the case of 8000 trees. This has the advantage that for each split into training and test stations, the random forest only needs to be

trained once to get an interpretable result. Thereby, it also reduces the risk of choosing non-optimal hyperparameters. Therefore, during all experiments, we set the number of trees to a very large number ($n_{\text{estimators}} = 8000$) to stabilize the non-deterministic behavior of training a random forest. Note that stability can probably be achieved with far fewer than 8000 trees. However, in order to reduce the bias from the observation above for a single split and single choice of hyperparameters, we choose a very large number that is still manageable regarding storage and computation time.

5 Results

Before presenting the results and starting the discussion, it is important to recall that for a given spatial split into training and test in situ stations, training or testing a machine learning model on the dataset $D_{N,k}$ means that only the data points corresponding to the training or test station locations are used, respectively. Furthermore, for fixed N , the in situ measurements $s(z, t)$ of surface NO₂ (ground truth) that are to be predicted in $D_{N,k}$ are exactly the same for all the different k . Further, recall that $D_{N,k}$ can be thought of as the set of data points for which measurements at all $N - 1$ previous hours are also guaranteed to be available, but only $k - 1$ values are added to the time-contiguous feature vector in Eq. (3).

In the following discussion of the experiments, introduced in Sect. 3.2, we focus exclusively on the results when $D_{4,k}$ is used for constructing test datasets, i.e., for $N = 4$ only. This is because we observe a similar benefit from a larger time contiguity k when evaluating the machine learning models' performance on $D_{N,k}$ for $N \in \{2, 3, 5\}$. As a further example, we provide detailed results for $N = 2$ in Figs. C2 and C3 in Appendix C.

5.1 Experiment 1: time-contiguous inputs provide additional information

In Experiment 1, we train linear regression models and random forests on $D_{4,k}$ for different time contiguities $k \in \{1, \dots, 4\}$ of the input features. The test performances of these models are evaluated via six-times spatial 10-fold cross-validation and are illustrated in Figs. 3b and 4b, respectively. Specifically, we show average Pearson correlation, NMSE, and NMAE over all 60 splits into training and test stations. We observe that, on average, both linear regression and random forests benefit from a larger time contiguity k regarding all considered performance measures. For example, the average correlation strictly increases from 0.702 for $k = 1$ to 0.737 for $k = 4$ in the case of linear regression, and for random forests, it increases from 0.802 to 0.817. Further, the average NMSE decreases from 0.196 to 0.171 for linear regression and from 0.139 to 0.129 for random forests. Therefore, both models benefit from larger time contiguity, but linear regression shows greater improvement, which is expected as

it cannot model non-linear effects. Furthermore, we observe that the larger k , the smaller the improvement compared to the case $k = 1$, which is to be expected since input features at time $t - k$ presumably have a decreasing impact on surface NO₂ at time t for larger k .

Although the visualization of average performances suggests an overall trend, it does not clearly indicate whether larger time contiguities ($k > 1$) consistently improve performance across all 60 station splits during cross-validation compared to $k = 1$. However, we found that this improvement holds true for all 60 station splits. The performance curves for individual splits are more or less parallel to the average curve. In Figs. 3a and 4a, we illustrate this for exemplary station splits, where only five splits are shown for better visibility. To quantify the gain in performance for individual splits between using time contiguity $k = 1$ and larger time contiguities $k > 1$, we proceed as follows: for a given test dataset, let E_k be the test performance (e.g., correlation) achieved by the model using time contiguity k for its inputs. We define the *performance gain* of this model over the case with no time contiguity $k = 1$ in Experiment 1 as

$$\frac{E_1 - E_k}{E_1 - E_{\text{opt}}}, \quad (5)$$

where E_{opt} is the optimal value of the respective performance measure; e.g., $E_{\text{opt}} = 1$ for the Pearson correlation or $E_{\text{opt}} = 0$ for NMSE and NMAE. The average performance gains for the cases $k \in \{2, 3, 4\}$ compared to $k = 1$ are depicted in Figs. 3c and 4c for linear regression and random forests, respectively. In both cases and for all performance measures, the highest average performance gain is achieved with $k = 4$. Specifically, linear regression models achieve average performance gains of 15.2 % in correlation, 13.0 % in NMSE, and 7.7 % in NMAE, whereas random forests achieve gains of around 7.8 %, 7.0 %, and 4.7 %, respectively. It is noteworthy that, for linear regression, the performance gain across all 60 splits is approximately at least 12.0 % in correlation, 10.0 % in NMSE, and 6.1 % in NMAE. On the other hand, random forests achieve performance gains of at least 4.6 %, 4.0 %, and 3.1 %, respectively. Therefore, utilizing a larger time contiguity consistently provided beneficial additional information for both linear regression and random forest models.

Additionally, for $k = 1$ and the best time contiguity $k = 4$, we examine for each split the orthogonal regression curve between the models' predictions and ground truth measurements of surface NO₂ on the corresponding test dataset. For a fixed split, this is illustrated as a two-dimensional histogram in the first row of Fig. 5 for linear regression and in Fig. 6 for random forests. Although the histograms are restricted to surface NO₂ and predictions between 0 and 40 $\mu\text{g m}^{-3}$ for better visibility, all data points are taken into account to determine the orthogonal regression curve. It becomes evident that both the slope and the bias of the orthogonal regression curve improve for $k = 4$ (panel b) compared to $k = 1$ (panel a), where

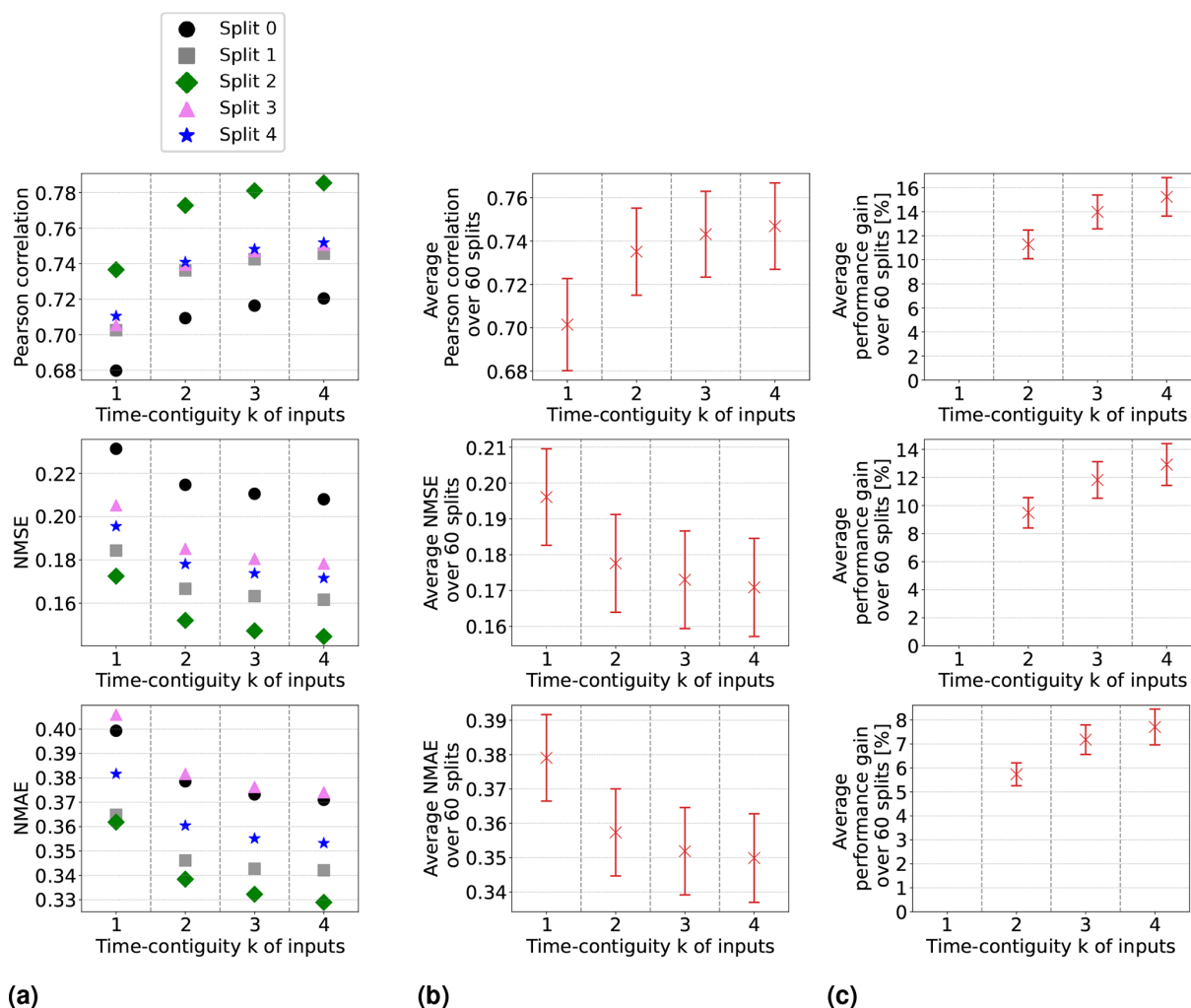


Figure 3. Linear regression models have been trained and tested on datasets $D_{4,k}$ for 60 different splits into training and test stations, with different time contiguity k of the input features. In panel (a), performances on test sets are shown for five exemplary station splits with respect to three performance measures. Panel (b) shows the average performance over all 60 splits, with error bars illustrating the standard deviation. Panel (c) shows the average performance gain relative to the case $k = 1$; see Eq. (5) for the definition of performance gain. Across each row, the same performance measure is considered. The exact values in panel (b) can be found in Table B2, columns $D_{4,1}$ to $D_{4,4}$.

improvement means that the slope becomes closer to 1 and the bias closer to 0. In the second row of these figures, we plot the mean orthogonal regression curve, which represents the mean slope and mean bias of all 60 orthogonal regression curves. An upper bound for all these curves is represented by the line with the maximal slope and bias across all splits (note that maximal slope and bias might not occur for the same split). Similarly, a lower bound is obtained, and both bounds are shown within the same plots. Both the mean orthogonal regression curve and the upper and lower bounds improved for $k = 4$ for both linear regression and random forests. However, the improvement is larger for the linear regression models, which is consistent with the previous discussion on performance measures, such as NMSE.

We want to stress another observation: looking at the upper and lower bounds of the orthogonal regression curves, we see that all slopes are smaller than 1, whereas all biases are positive. Further, there is a noticeable gap towards the identity line. Regarding the latter, one possible explanation could be that spatially splitting the dataset into training and test sets causes a large difference in the statistical properties of the training and test sets. This may simply be because there are overall just 637 different in situ stations available, so the law of large numbers may not yet apply well when sampling 10 % of test stations. However, this does not explain why the slopes and biases are not more symmetrically distributed around slope 1 and bias 0. Studying the impact of the number of available in situ stations and their locations on

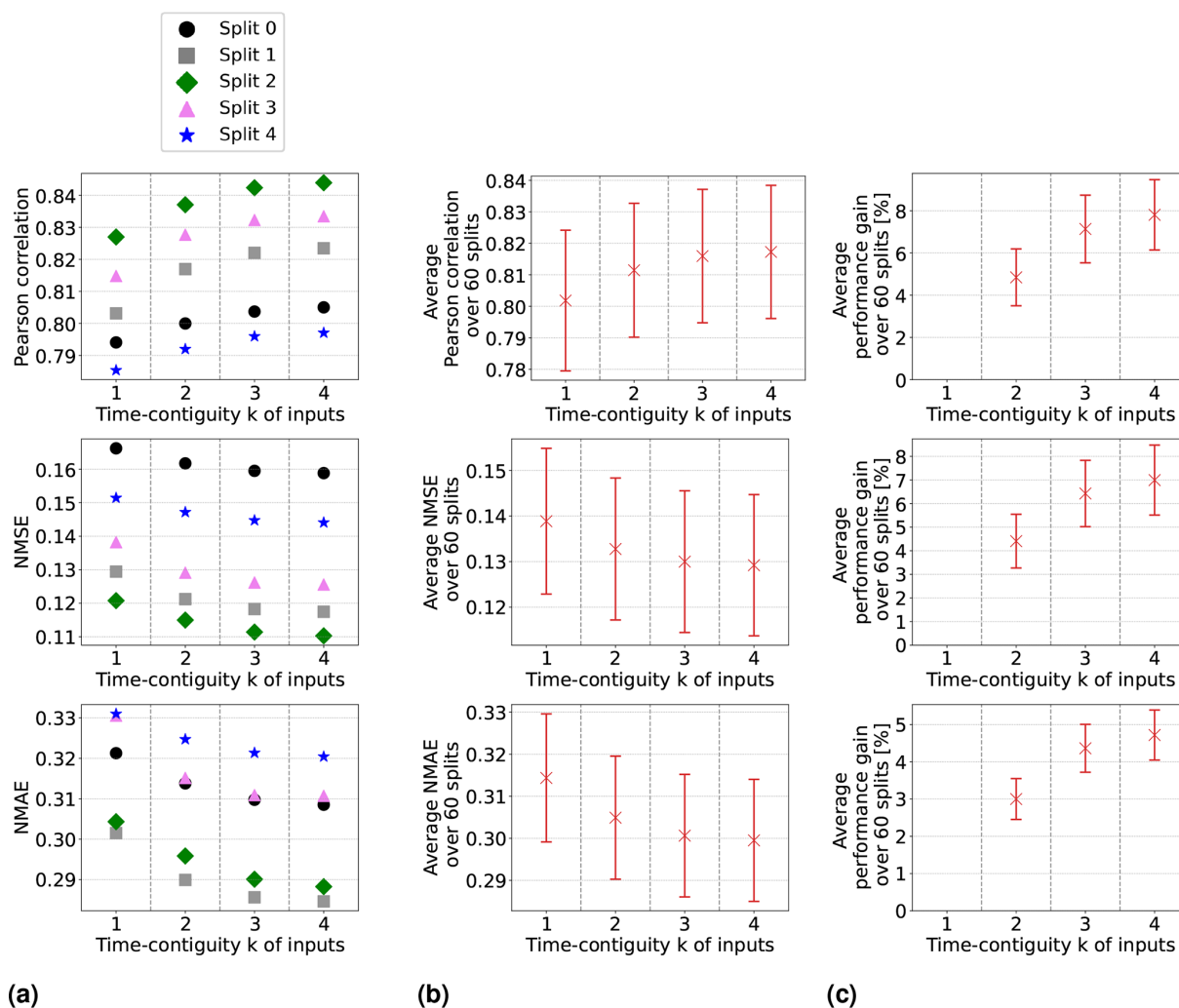


Figure 4. Same as Fig. 3 but for random forests trained and tested on datasets $D_{4,k}$ for 60 different splits into training and test stations, with different time contiguity k of the input features. In panel (a), performances on test sets are shown for five exemplary station splits with respect to three performance measures. Panel (b) shows the average performance over all 60 splits, with error bars illustrating the standard deviation. Panel (c) shows the average performance gain relative to the case $k = 1$; see Eq. (5) for the definition of performance gain. Across each row, the same performance measure is considered. The exact values in panel (b) can be found in Table B3.

the slopes and biases of these orthogonal regression curves will be an interesting task for future work.

5.2 Experiment 2: time-contiguous inputs are beneficial in spite of a smaller dataset

In Experiment 1, the models were trained and tested on $D_{N,k}$ for fixed N but with a different time contiguity $k \in \{1, \dots, N\}$ of their input features. This means that for a fixed station split, the number of training data points was the same for all the different k , since the size of $D_{N,k}$ only depends on N (see Table 1). However, for $M \in \{k, \dots, N - 1\}$, there would be significantly more data points available in $D_{M,k}$ than in $D_{N,k}$, which could be used during training. To make a fair conclusion about whether a larger time contiguity ($k > 1$) in

the models' input is more beneficial compared to time contiguity $k = 1$, we need to consider that for $k = 1$, one can also train on these larger datasets. It should be noted that we have also considered training on smaller datasets, thus on $D_{M,k}$ with $M > N$. However, non-competitive results were obtained for random forests in these cases. For linear regression, performances were also worse but with some exceptions regarding the NMAE; see Fig. C2 in Appendix C. This is why we restrict the following discussion to training on larger datasets ($M \leq N$) only.

Focusing again on the test case $N = 4$, we compare the performance on test sets in $D_{4,k}$ of models trained on larger datasets $D_{M,k}$ for all $M \in \{k, \dots, 4\}$ and all $k \in \{1, \dots, 4\}$. Note that for $M = 4$, this is just the setting of Experiment 1. Altogether, 10 different linear regression models and 10 ran-

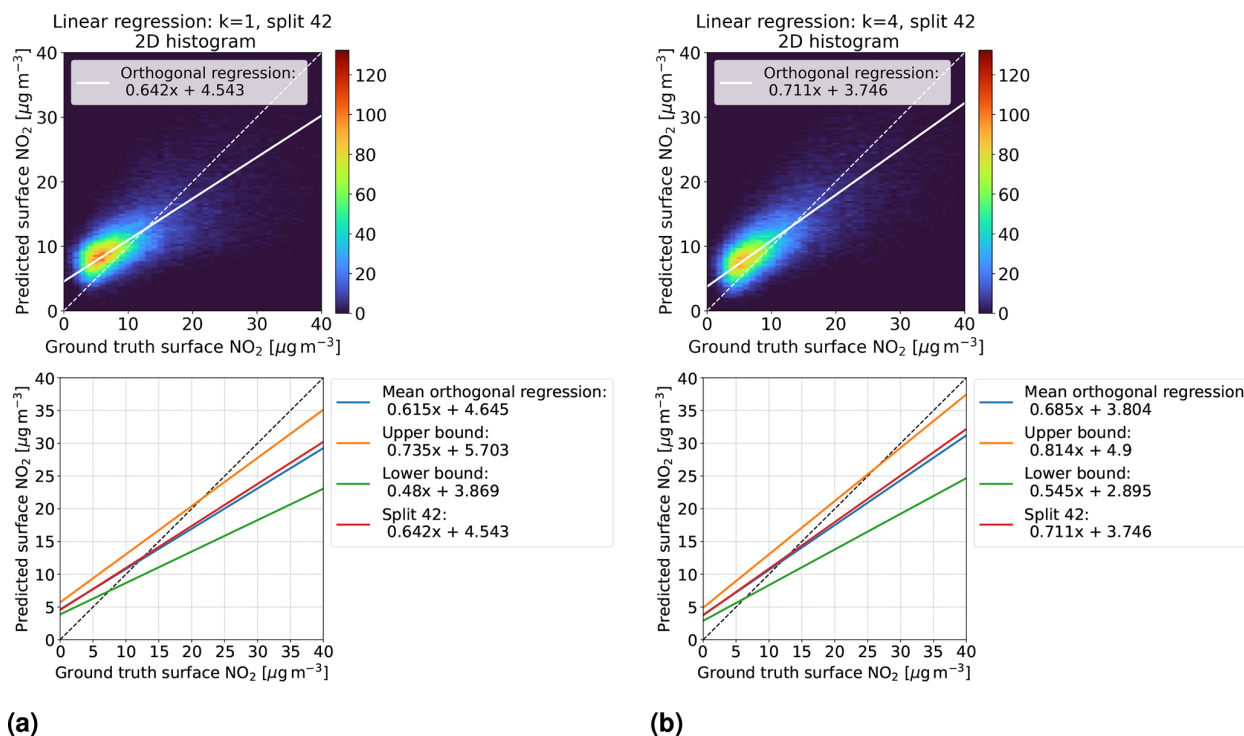


Figure 5. Linear regression models trained on $D_{4,k}$ with time contiguities (a) $k=1$ and (b) $k=4$. First row: for a fixed split (number 42) into training and test stations, the models' predictions on the corresponding test set $D_{4,k}$ are compared with in situ measurements of surface NO₂ (ground truth) in a two-dimensional histogram. Second row: for all 60 station splits, orthogonal regression is considered between predicted and ground truth surface NO₂. Mean orthogonal regression refers to the line of average slope and bias over all 60 regression lines (blue line). The regression line for the example in the first row is also shown (red line).

dom forest models are used to make predictions of the same ground truths in the split-dependent test sets $D_{N,k}$.

Average performance measures from spatial cross-validation are shown in Fig. 7a for linear regression and in Fig. 8a for random forests. We observe that when training with time contiguity $k=1$, i.e., on $D_{M,1}$, the best results are obtained for $M=4$. In other words, there is no improvement on the test set $D_{4,1}$ if training is done on the larger datasets ($M \in \{1, 2, 3\}$). There is one exception for random forests with the Pearson correlation, where training on $D_{3,1}$ yields slightly better results on average compared to training on $D_{4,1}$. However, this difference is quite small, as shown in Fig. 8a. Moreover, for all performance measures, the best performance across all 10 different training cases is achieved by the models trained on $D_{4,4}$ with time contiguity $k=4$. Note that this is one of the training settings already considered in Experiment 1.

For individual splits, we consider the performance gains that models with time contiguity $k > 1$ achieve compared to models with no time contiguity ($k=1$). Since, in contrast to Experiment 1, we are now dealing with four different training cases for $k=1$, we slightly adapt the definition of performance gains from Eq. (5): for a given split into training and test stations and fixed N , let $E_{M,k}$ be the test per-

formance (e.g., correlation) on $D_{N,k}$ achieved by a model trained on $D_{M,k}$. We define the performance gain achieved by this model in Experiment 2 as

$$\min \left\{ \frac{E_{P,1} - E_{M,k}}{E_{P,1} - E_{\text{opt}}} : P \in \{1, \dots, 5\} \right\}. \quad (6)$$

In other words, for each split, the performance gain is always computed with respect to the best model trained without time contiguity ($k=1$).

Average performance gains are depicted in Figs. 7b and 8b, which differ only slightly from those in Experiment 1, as models trained on $D_{4,1}$ are better, on average, than models trained on $D_{M,1}$. Linear regression models trained with $k=4$ still achieve performance gains of 15.0 % in correlation, 12.8 % in NMSE, and 6.6 % in NMAE, whereas random forests achieve average gains of around 7.3 %, 6.6 %, and 4.7 %, respectively. Again, we observe that improvements over $k=1$ are not only true on average, but also for each individual split: Figs. 7c and 8c show the minimal performance gains over all 60 splits. It shows that linear regression models for $k=4$ always achieve an improvement of at least 11.7 % in correlation, 9.1 % in NMSE, and 4.4 % in NMAE. Random forests achieve gains of at least 2.5 %, 3.0 %, and 3.1 %, respectively. Hence, models with a larger

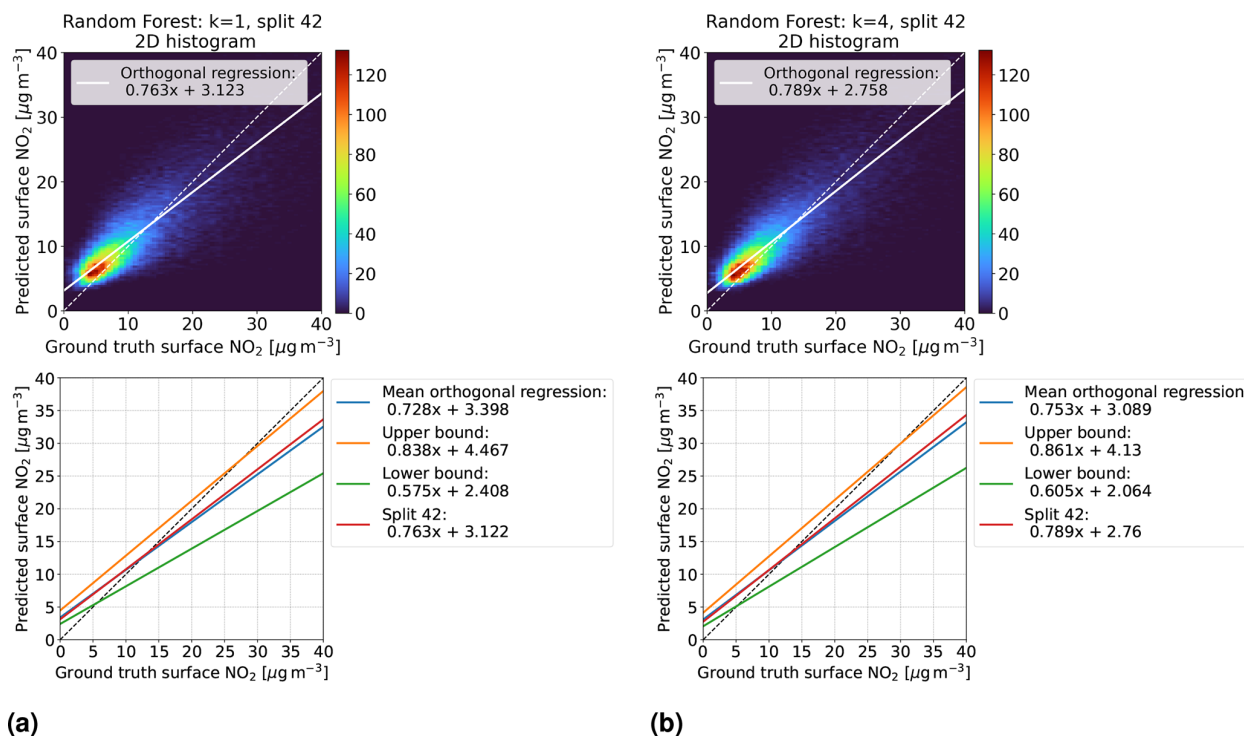


Figure 6. Same as Fig. 5 but for random forests trained on $D_{4,k}$ with time contiguities (a) $k = 1$ and (b) $k = 4$. First row: for a fixed split (number 42) into training and test stations, the models' predictions on the corresponding test set $D_{4,k}$ are compared with in situ measurements of surface NO₂ (ground truth) in a two-dimensional histogram. Second row: for all 60 station splits, orthogonal regression is considered between predicted and ground truth surface NO₂. Mean orthogonal regression refers to the line of average slope and bias over all 60 regression lines (blue line). The regression line for the example in the first row is also shown (red line).

time contiguity $k > 1$ provide reliable and statistically significant improvements (with respect to the performance measures) compared to models with no time contiguity ($k = 1$). Similar observations are made for the coefficient of determination and the index of agreement, two further performance measures. Definitions can be found in Appendix A and achieved performances in Tables B2 and B3 in Appendix B.

So far, we have discussed the test case $N = 4$ in detail. In the remainder of this section, we briefly summarize our similar observations for general $N \in \{2, 3, 4, 5\}$: for all N , we observed that the best test performances on $D_{N,k}$ are achieved when training on $D_{N,N}$, i.e., with time contiguity $k = N$. If $N = 5$, we observe that there is barely any difference between training on $D_{5,5}$ and training on $D_{4,4}$, which implies that it is not required to use a larger time contiguity than $k = 4$. Also, for the general test case N , models trained with time contiguity $k > 1$ achieve reliable performance gains over models trained with $k = 1$. Results for the test case $D_{2,k}$ are illustrated in Figs. C2 and C3 in Appendix C.

Altogether, our findings demonstrate that it is indeed reliably beneficial to use time-contiguous input features for predicting surface NO₂, in spite of a smaller available training dataset, which answers our main research question. As a

rule of thumb, consider the case where surface NO₂ is to be predicted at a given location and time for which input features are also available at $j \geq 1$ previous hours. Then use $j' = \min\{3, j\}$ hours, in addition to the features at the current time, as input for a random forest that has been trained with time contiguity $k = j' + 1$ on a dataset $D_{k,k}$. If features are not available at previous hours, use the random forest that has been trained without time contiguity. We have demonstrated within this experiment that time-contiguous models provide valuable support whenever they are applicable. An interesting future task would be to inspect whether a similar rule can be observed for other machine learning approaches.

Within this section, we analyzed the difference between time-contiguous models in terms of prediction accuracy. However, we did not systematically assess other potential differences that may arise when switching between models trained with different time-contiguous features. For practical applications, when combining these models to create surface NO₂ concentration maps, it remains an interesting avenue for future work to investigate whether the ensemble of such models yields consistent combined spatial patterns in predicted surface NO₂.

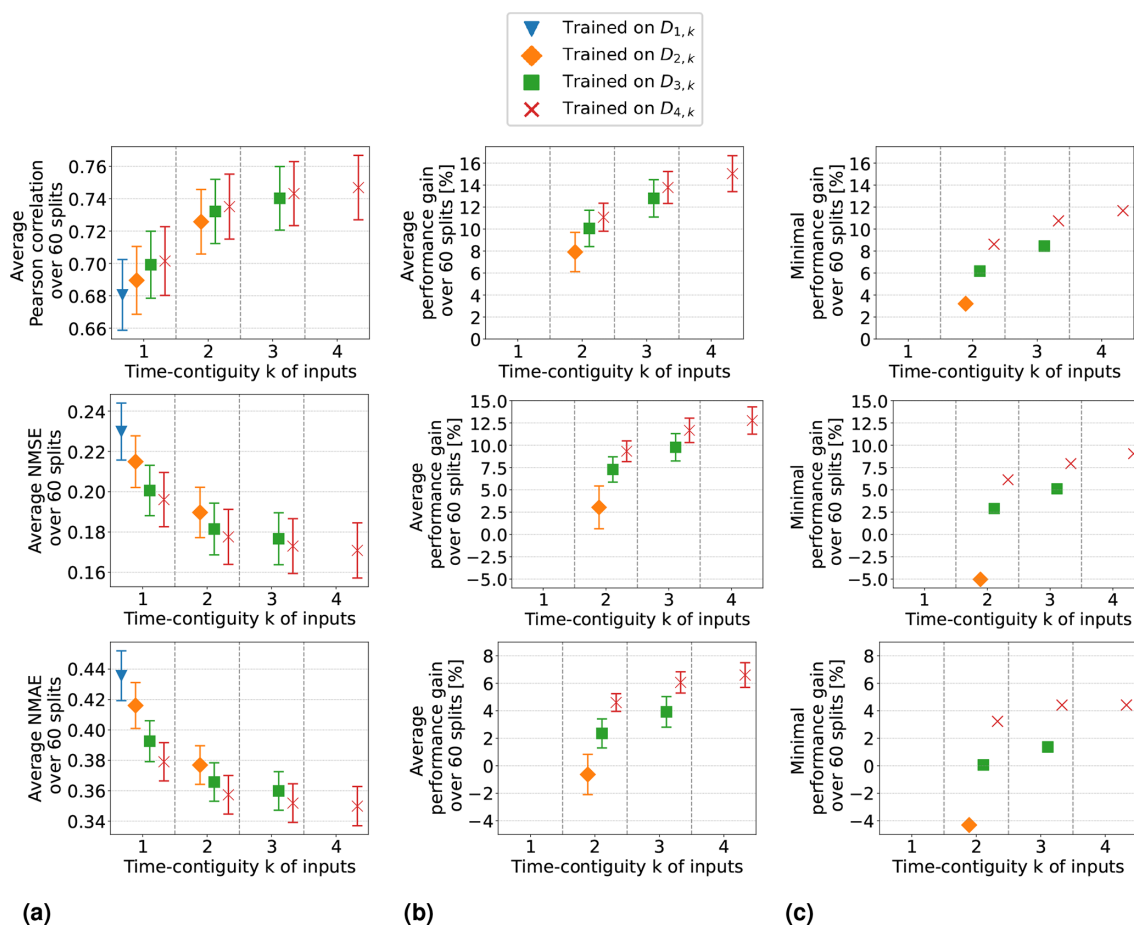


Figure 7. Linear regression models trained on $D_{M,k}$ for $M \leq 4$ with different time contiguities k . Performance on $D_{4,k}$ has been evaluated through six-times 10-fold spatial cross-validation. Panel (a) shows the average performance over all 60 station splits for three performance measures. Panel (b) shows the average performance gain relative to the best case of $k = 1$; see Eq. (6) for the definition of performance gain. Error bars illustrate the standard deviation. Panel (c) shows the minimal performance gain. Across each row the same performance measure is considered. The exact values in panels (a) and (b) can be found in Table B2.

5.3 Experiment 3: influence of tropospheric NO₂ VCDs, latitude, and surface height

In Experiment 3, we compare the outcomes of Experiment 2 in four different settings regarding the input of the models, as described in Sect. 3.2:

Setting 1. All features selected in Sect. 3.1 are included as input features, which was the setting in Experiments 1 and 2.

Setting 2. VCDs are excluded as an input feature.

Setting 3. Latitude and surface height are excluded.

Setting 4. VCDs, latitude, and surface height are excluded.

In this section, we focus exclusively on random forests and discuss the test results on $D_{4,k}$ for the four different settings above.

Setting 1 is discussed in the previous section, where the results are illustrated in Fig. 8. Equally detailed illustrations for the remaining three settings are provided in Appendix D. A direct comparison between the four settings is made in Fig. 9: panel (a) shows the average Pearson correlation, NMSE, and NMAE achieved by random forests within these four settings, while panel (b) displays the corresponding average performance gains. For clarity, we only include the results for the models trained on $D_{4,k}$ for different time contiguities $k \in \{1, \dots, 4\}$, excluding the models trained on larger datasets $D_{M,k}$ (similar to Experiment 1).

In Setting 3, where latitude and surface height are excluded, the models achieve similar results to those in the original Setting 1. Results are even slightly better without using these coordinates if $k > 1$. Moreover, the benefit of using time-contiguous input features is larger in Setting 3: average performance gains, calculated with Eq. (6), achieved when training on $D_{4,k}$ are 9.3 % in Pearson correlation, 8.3 % in

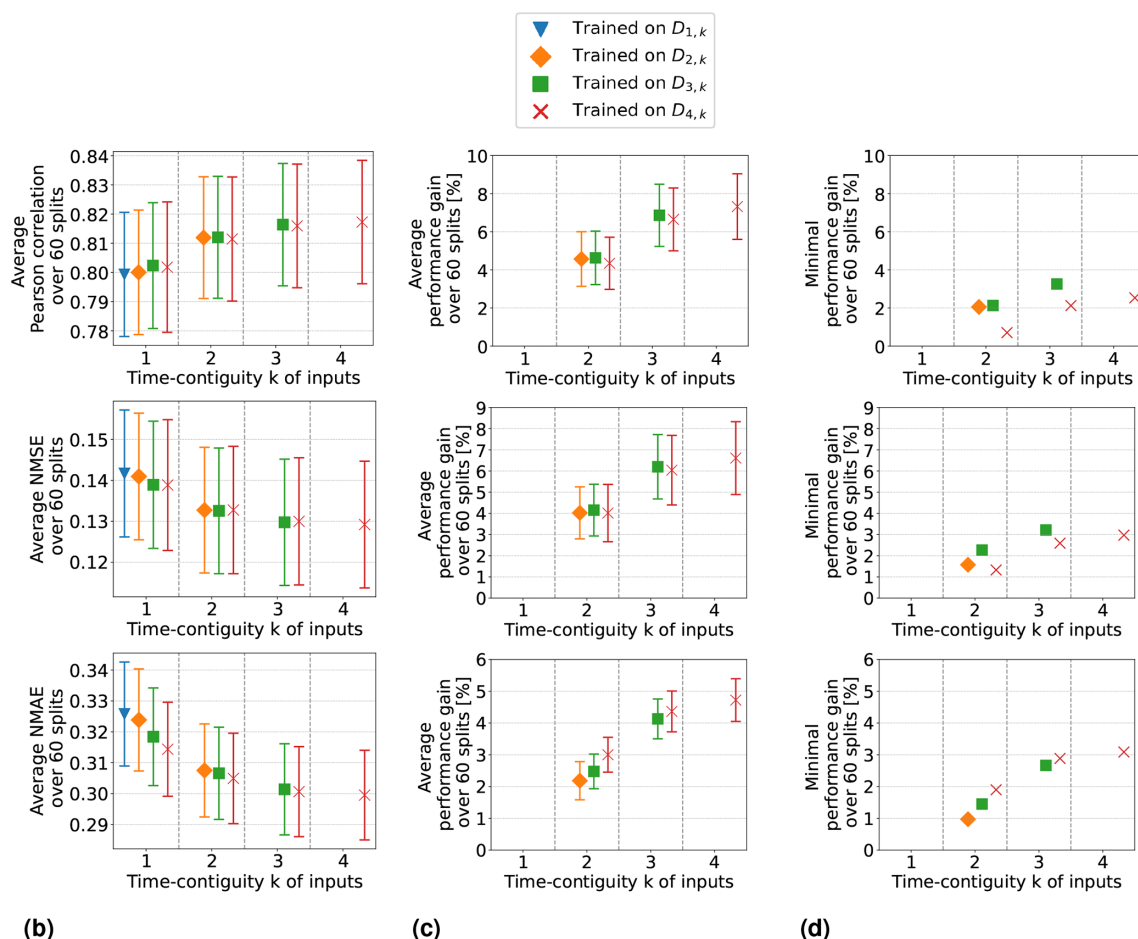


Figure 8. Same as Fig. 7 but for random forests trained on $D_{M,k}$ for $M \leq 4$ with different time contiguities k . Performance on $D_{4,k}$ has been evaluated through six-times 10-fold spatial cross-validation. Panel (a) shows the average performance over all 60 station splits for three performance measures. Panel (b) shows the average performance gain relative to the best case of $k = 1$; see Eq. (6) for the definition of performance gain. Error bars illustrate the standard deviation. Panel (c) shows the minimal performance gain. Across each row the same performance measure is considered. The exact values in panels (a) and (b) can be found in Table B3.

NMSE, and 5.7 % in NMAE. The minimum gains across all 60 station splits are 5.4 %, 3.7 %, and 3.8 % in correlation, NMSE, and NMAE, respectively (see Fig. D1). This implies that, similar to Setting 1, including time-contiguous features also provides a reliable improvement in Setting 3. This observation that coordinates are not required as inputs to make good predictions is promising, since it presumably increases the models' chances to also perform well outside of South Korea. Nevertheless, this hypothesis remains to be investigated within further research.

When excluding the tropospheric NO₂ VCDs (Setting 2), all performance measures decline, which is expected because the VCDs correlate the most among all input features with the surface NO₂ measurements. Despite this, the performances remain acceptable. For instance, with time contiguity $k = 1$, the average Pearson correlation in Setting 2 is 0.78, whereas it is about 0.8 in Settings 1 and 3, when VCDs are included. Interestingly, without VCDs in Setting 2, the average

performance gains achieved with larger k are significantly lower: in Setting 2, the average performance gain is around 2 %, whereas in Settings 1 and 3, it is 3.5 and 4.5 times larger, respectively. Consequently, for time contiguity $k = 4$, the difference in performance is larger: models in Setting 2 achieve an average correlation of 0.786, while those in Settings 1 and 3 reach almost 0.82. When tropospheric NO₂ VCDs, latitude, and surface height are excluded in Setting 4, not only do performances weaken further, but the performance gains also drop below 1 %. In Setting 4, the average correlation is below 0.765 for all k . Similar trends are observed for NMSE and NMAE. This indicates that spatial coordinates play a more critical role when VCDs are excluded, which presumably leads to models that are less capable of generalizing to locations outside of South Korea. Inspecting the connection between including VCDs and the model's ability to generalize to locations outside of South Korea remains an interesting task for the future.

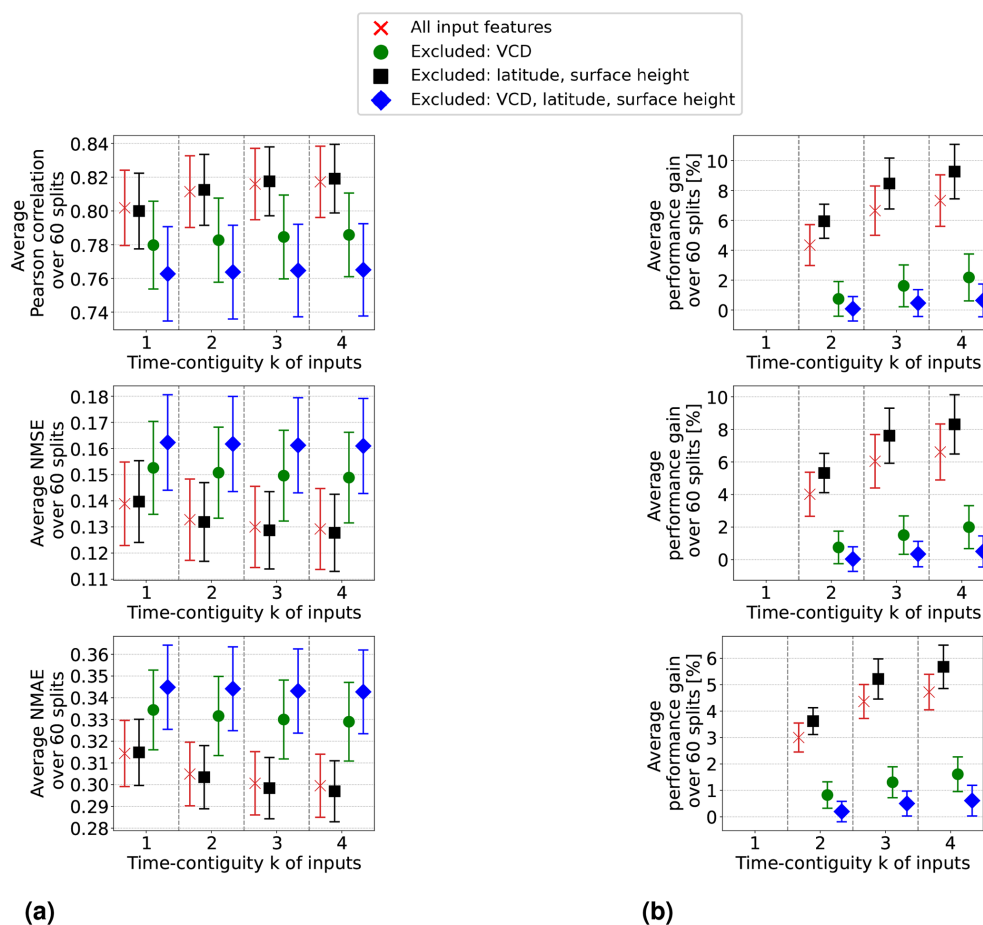


Figure 9. In the four settings of Experiment 3 (named in the legends of the plots), random forests are trained and tested on $D_{4,k}$ for different time contiguities k . Performance is evaluated through six-times 10-fold spatial cross-validation. Panel (a) shows the average performance over all 60 station splits achieved within these four settings. Three performance measures are considered, one for each row. Error bars illustrate the standard deviation. Panel (b) shows the average performance gain relative to the best case of $k = 1$; see Eq. (6) for the definition of performance gain.

Furthermore, when tropospheric NO₂ VCDs are excluded, in both Setting 2 and Setting 4, the use of time-contiguous inputs no longer provides a reliable improvement. Across the 60 station splits, the performance gain is not always positive, which can be seen in Fig. 9b. Due to this observation that improvements by time-contiguous inputs are only reliable when including VCDs, the following question arises: how is performance affected if VCDs are treated as the only time-contiguous input feature? The experiments covering this case are illustrated in Fig. D4 in Appendix D. We observe that the average performances and average performance gains are higher if the other features are also considered time contiguous. Therefore, one future task could be to find the optimal choice of time contiguity k for each input feature individually.

At the end of this section, we show in Fig. 10 an example of how predictions of surface NO₂ appear on a map for the four investigated settings. We consider latitudes and longi-

tudes within 32° N, 39° N and 124° E, 132° E, respectively. GEMS tropospheric NO₂ VCDs on 7 April 2021 from 01:45 to 02:15 UTC are shown in panel (a). We chose this time and day due to little cloud cover in the area and thus only a few missing satellite observations. Predictions of surface NO₂ from 01:00 to 02:00 UTC made by random forests are shown in panel (b) for Settings 1 and 3, whereas panel (c) covers the settings with tropospheric NO₂ VCDs excluded. All models have been trained with time contiguity $k = 4$ on $D_{4,4}$.

We observe that there is a high similarity between predictions made in Settings 1 and 3, when tropospheric NO₂ VCDs are included as input features. This is in agreement with our findings from Fig. 9 that in both settings similar results are achieved regarding all considered performance measures. This observation is promising, as excluding latitude and surface height reduces the spatial bias of the model, which is to be tested in future studies. Therefore, presum-

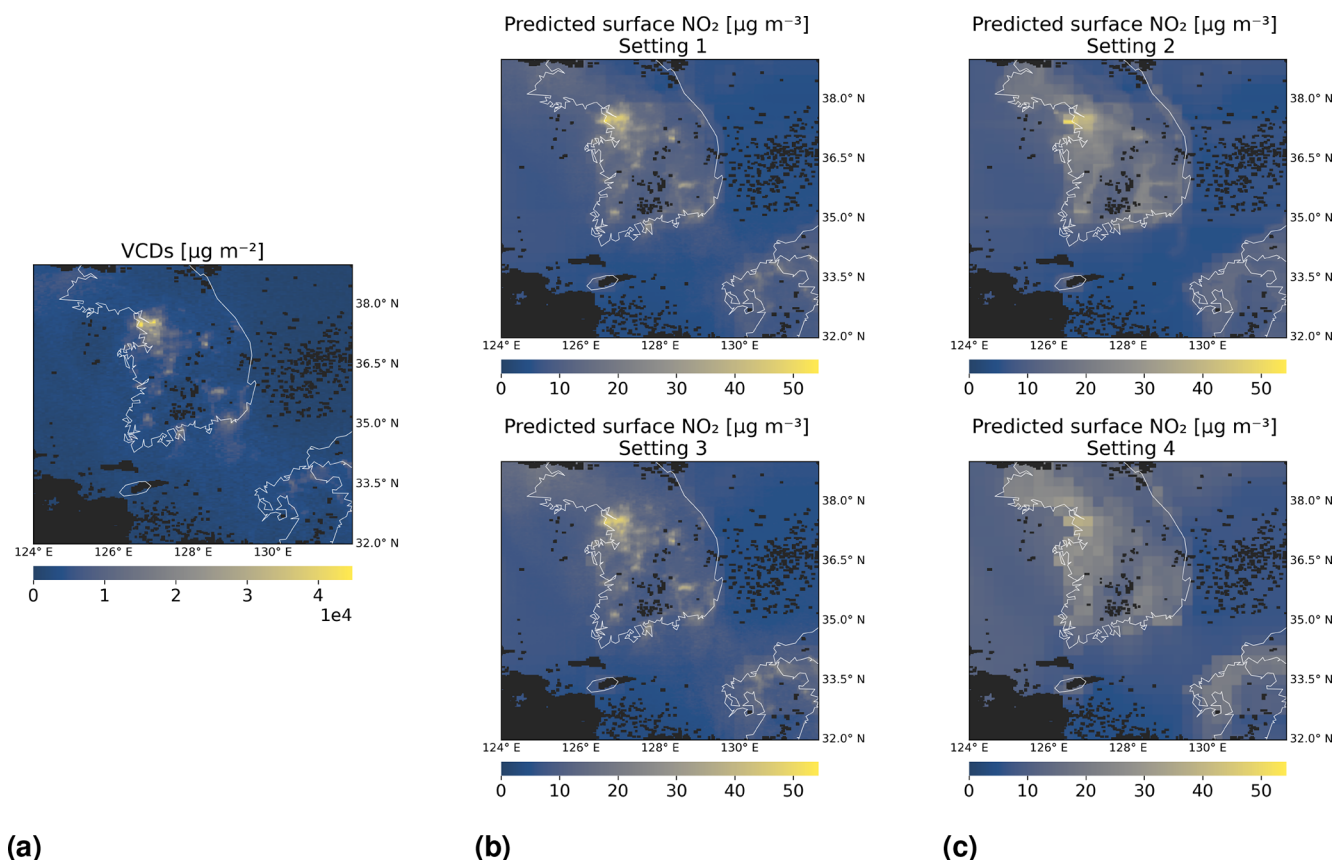


Figure 10. Predictions of surface NO₂ by random forests on 7 April 2021 from 01:00 to 02:00 UTC, for Settings 1–4 of Experiment 3. Panel (a) shows tropospheric NO₂ VCDs from 01:45 to 02:15 UTC. Panel (b) shows predicted surface NO₂ in Settings 1 and 3, when VCDs are included as input. Panel (c) shows predictions in Settings 2 and 4, when VCDs are excluded. In the second row of panels (b) and (c), latitude and surface height are excluded. The black mask indicates missing data, e.g., due to clouds. All models have been trained with time contiguity $k = 4$ on $D_{4,4}$ for the same choice of training stations.

ably, the model's chance of making suitable predictions in different parts of the world increases. In Settings 1 and 3, the impact of the tropospheric NO₂ VCDs on the prediction of surface NO₂ is directly visible, since the hotspots of the VCDs and predictions of surface NO₂ are depicted at the same locations. On the other hand, when VCDs are excluded in Settings 2 and 4, these hotspots are less recognizable due to a smaller contrast to their neighborhood; see Fig. 10c. In Settings 2 and 4, the predicted surface NO₂ has a coarser resolution, which is to be expected considering that the resolution of meteorological inputs is 8 times coarser compared to the VCDs. In all four settings, the contrast between the hotspots and the background of predicted surface NO₂ is less pronounced compared to the contrast observed in the tropospheric NO₂ VCDs shown in panel (a). This effect is even more evident in another example from 27 February 2022, shown in Fig. 11. Notably, the predicted concentrations of surface NO₂ over water are only slightly smaller compared to those over land within all settings, even in regions far from the coast, such as the southeastern parts of the maps. How-

ever, emissions over water are not expected, aside from maritime traffic. Furthermore, at some distance from the coast, no contribution from land-based emissions is expected due to the short atmospheric lifetime of NO₂. Consequently, both the tropospheric NO₂ VCDs and the surface NO₂ concentrations should be low in these areas. Given the predicted surface concentrations of approximately $7 \mu\text{g m}^{-3}$, it appears that the models have likely overestimated surface NO₂ concentrations in these areas over water. This aligns with the observation from Fig. 6, which shows that the models tend to overestimate low surface NO₂ values. A possible explanation for this could be that the models were trained only on data from stations located on land or islands.

5.4 Seasonal and diurnal error distribution

In the previous sections, the performance of machine learning models is evaluated using whole-year data, spanning January 2021 to November 2022. In this section, we inspect how prediction quality varies across different seasons and throughout the day. Some variation is expected, as the ac-

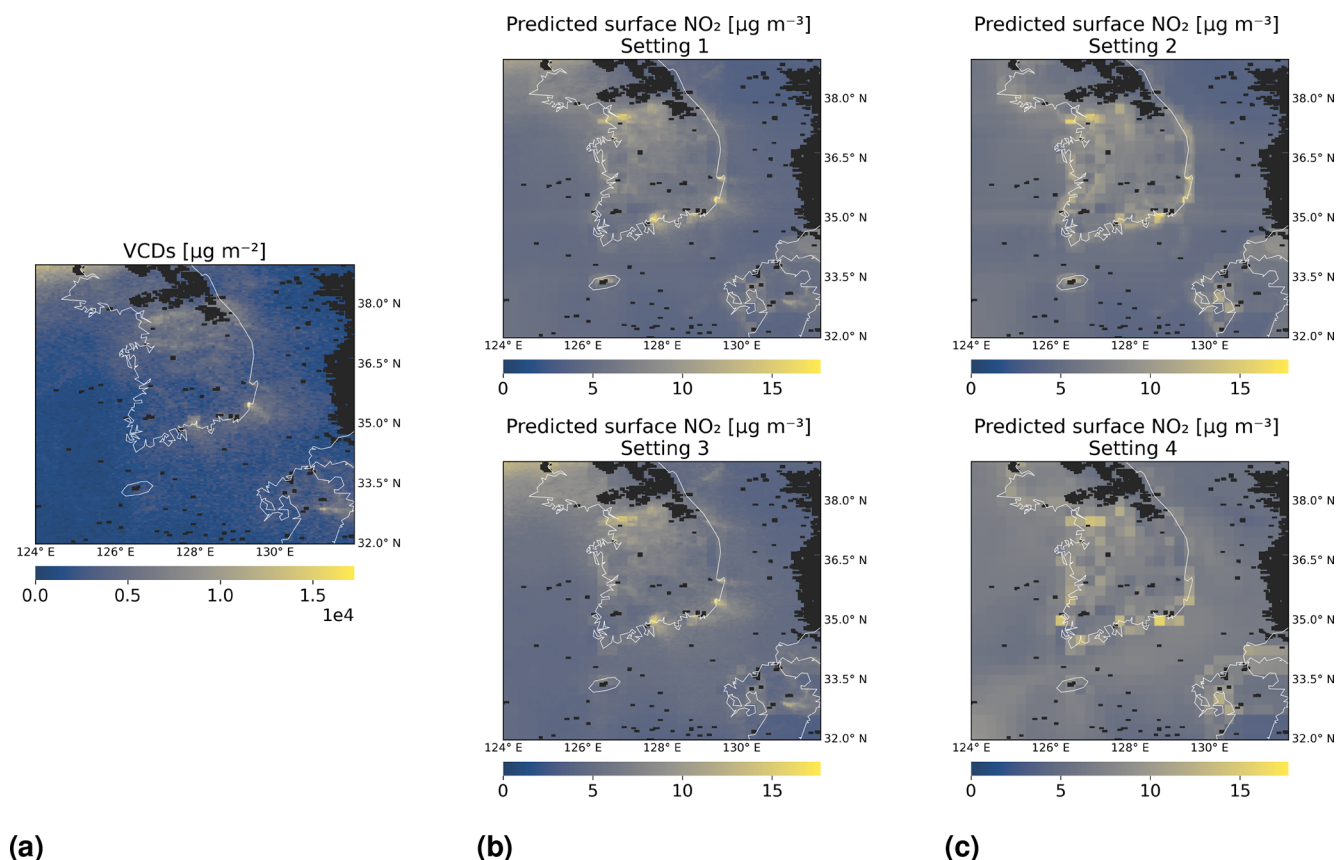


Figure 11. Same as Fig. 10 but on 27 February 2022. Panel (a) shows the VCDs from 06:45 to 07:15 UTC. Panels (b) and (c) show predicted surface NO₂ from 06:00 to 07:00 UTC, for the four settings of Experiment 3.

curacy of GEMS observations also fluctuates. For example, accuracy tends to be lower in the morning due to the shallow boundary layer (Yang et al., 2023a). For the remainder of this section, we focus on the best-performing models identified in our earlier analysis. Specifically, we reconsider the random forest models from Setting 3 in Sect. 5.3, which do not incorporate spatial coordinates as input features. These models were trained on all respective training datasets $D_{N,k}$, but for this section, their performance is spatially cross-validated on the test datasets for different seasons and times of the day individually. For simplicity, we restrict our attention to models that were trained on the dataset $D_{4,k}$. Furthermore, we inspect whether benefits from time-contiguous inputs depend on the season or time of the day.

First, we compare the test performance across different seasons. Each season in South Korea is typically defined as a 3-month period: spring (March–May), summer (June–August), autumn (September–November), and winter (December–February). Table 3 shows the percentage of data points in $D_{4,k}$ belonging to each season. Notably, summer has the fewest valid data points due to the applied filter for the qa value during data preprocessing. In addition, the Pearson correlation between surface NO₂, measured at the in situ

stations, and VCDs is the lowest in summer (see Table 3). These factors likely contribute to the significantly lower performance of the random forest models in summer compared to other seasons (see Fig. 12). In contrast, the model performance is the highest in winter across all performance measures, i.e., for Pearson correlation, NMSE, and NMAE. Moreover, we observe that within each season, incorporating time-contiguous inputs improves prediction quality. The performance gains, calculated using Eq. (5), are also shown in Fig. 12. Notably, the largest gains from time-contiguous inputs occur in winter, exceeding 12 % in Pearson correlation for time contiguity $k = 4$. The smallest gains are observed in summer, with an improvement of only 5 % in Pearson correlation.

Finally, the performance across different times of the day is illustrated in Fig. 13. Since we focus on training and testing on $D_{4,k}$, the earliest time window with available data is 10:00–11:00 Korean standard time (KST). The best performance is achieved around midday, while the performance declines in the morning and afternoon. The worst results occur between 16:00 and 17:00 KST, possibly due to the fact that surface NO₂ has the weakest correlation with VCDs at that time (see Table 4). Moreover, it should be noted that for

Table 3. Statistics for seasonal segments of the dataset $D_{4,k}$.

	Spring	Summer	Autumn	Winter
Proportion of $D_{4,k}$ dataset	41 %	16 %	20 %	23 %
Proportion of $D_{4,k}$ if no qa filter was used	28 %	33 %	23 %	14 %
Correlation of VCDs with surface NO ₂ measurements	0.68	0.58	0.67	0.74

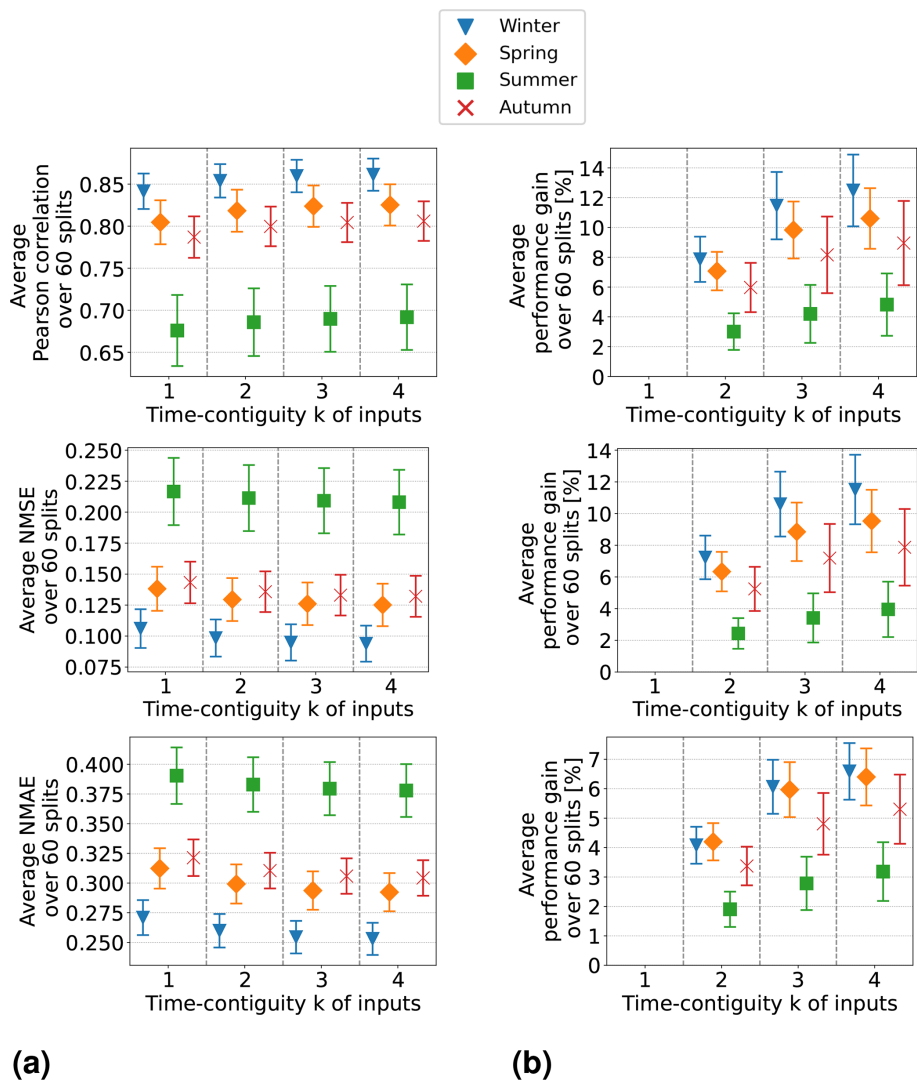


Figure 12. Random forests trained on $D_{4,k}$ for different time contiguities k , without spatial coordinates as input features. Test performance is evaluated in different seasons (winter, spring, summer, and autumn) through six-times 10-fold spatial cross-validation. Panel (a) shows the average performance over all 60 station splits achieved in different seasons and for different k . Three performance measures are considered, one for each row. Error bars illustrate the standard deviation. Panel (b) shows the average performance gain relative to the case of $k = 1$; see Eq. (5) for the definition of performance gain.

datasets $D_{N,k}$ with $N \leq 3$, in which data points at times earlier than 10:00 KST occur, the performance is expected to further decrease compared to the later morning hours.

Furthermore, at all times, time-contiguous models consistently outperform models with no time contiguity $k = 1$,

demonstrating a clear benefit from using time-contiguous input features.

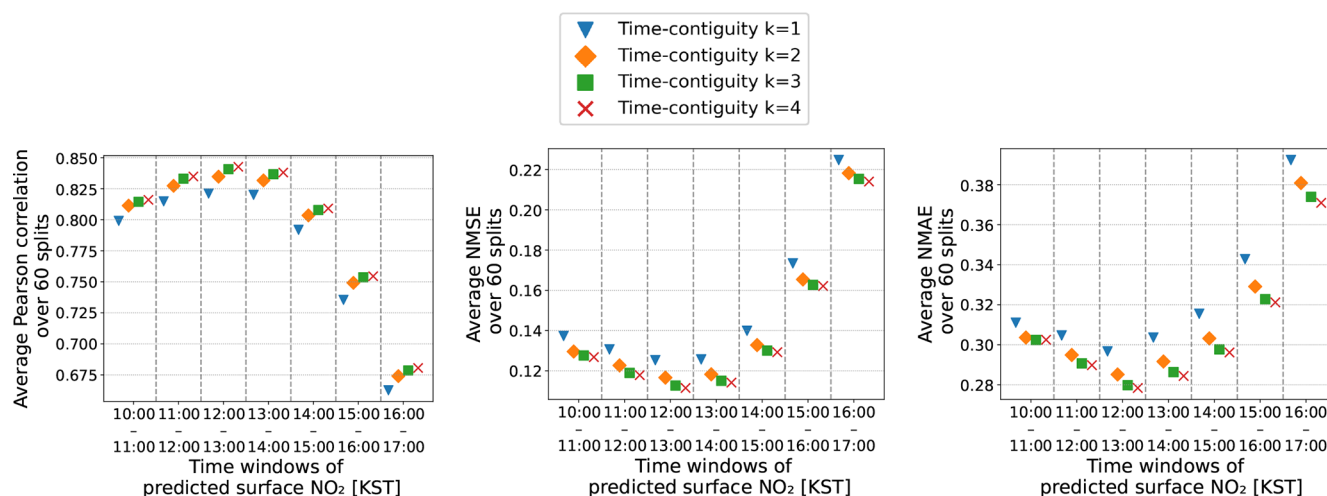


Figure 13. Same random forests as in Fig. 12 but the test performance is cross-validated at different times of the day. The time windows are chosen in line with the in situ dataset. Korean standard time (KST) is used.

Table 4. Statistics for different hourly segments of the dataset $D_{4,k}$.

	Time windows of predicted surface NO ₂ (KST)						
	10:00–11:00	11:00–12:00	12:00–13:00	13:00–14:00	14:00–15:00	15:00–16:00	16:00–17:00
Proportion of $D_{4,k}$ dataset	8 %	12 %	20 %	20 %	19 %	14 %	7 %
Correlation of VCDs with surface NO ₂ measurements	0.69	0.71	0.71	0.71	0.69	0.59	0.52

6 Conclusions and outlook

For the first time, hourly tropospheric NO₂ VCDs are available thanks to the geostationary satellite of the GEMS instrument platform. To predict surface NO₂ levels at a given time and location, we proposed to also include VCDs and meteorological features from previous hours as inputs to the machine learning models.

Our main research question was whether the considered machine learning models, random forests and linear regression, benefit from hourly time-contiguous input features for the prediction of surface NO₂. We observed that using time-contiguous input features led to reliable enhancements with respect to all considered performance measures, as long as tropospheric NO₂ VCDs were included. For random forests, average performance gains were between 4.5 % and 7.5 % depending on the performance measure. For linear regression models, average performance gains were larger, namely between 7 % and 15 %. This is to be expected since the non-linear structure of random forests allows for the extraction of more information from non-time-contiguous inputs, generally also leading to better predictions compared to linear regression models. These improvements were reliable in the sense that positive performance gains were achieved not only on average, but also across all 60 splits into training and test in situ stations during spatial cross-validation. Moreover, we

were able to demonstrate that performance gains were observed despite having much fewer data points available for training models with a larger time contiguity of their inputs. As a rule of thumb, for the case where tropospheric NO₂ VCDs are used as an input feature, we suggest the following: whenever surface NO₂ is to be predicted at a given location and time for which input features are available at j previous hours, feed those features, together with those at the current time, into a random forest that has been trained with time contiguity $k = \min\{j + 1, 4\}$ on a given training dataset $D_{k,k}$, specified in Sect. 2.3. If features are not available at previous hours, one cannot use a time-contiguous model to make a prediction for these data points, so one has to use the random forest that has been trained without time contiguity. Therefore, time-contiguous models should be understood as a supportive tool that should be applied whenever possible. Whether the rule of thumb above still applies to other machine learning models, such as neural networks or extreme gradient boosting, would be an interesting aspect for future studies.

Furthermore, when tropospheric NO₂ VCDs were included as input in the models, we observed that latitude and surface height were not required for achieving similar performances and benefits from time-contiguous inputs. Presumably, this increases the chance that the models will also provide good predictions beyond South Korea, which will

be an interesting investigation for future work. If validated, this would enhance the model's flexibility and broader applicability without the requirement of more training data, and hence longer training time, from different regions. Another task would be to determine the optimal time contiguity for every input feature individually, which would reduce redundancy among input features and hence lead to better performances.

When tropospheric NO₂ VCDs were excluded as input features, performance worsened but remained within an acceptable range. Additionally, we observed that the benefit of time-contiguous features was significantly reduced, and the performance gain was no longer reliable. Specifically, across all 60 splits during spatial cross-validation, benefits were not consistently observed. When both VCDs and spatial coordinates were excluded, performance decreased further. This indicates that spatial coordinates play a more critical role when VCDs are not included, which presumably leads to models that are less capable of generalizing to locations outside of South Korea. Again, this motivates further research on the connection between including VCDs and the models' ability to generalize to locations outside of South Korea.

Last but not least, we would like to address the time coverage of the data, which spans January 2021 to November 2022. Although data from December 2022 are missing, Sect. 5.4 shows that random forests performed best on winter data. It would be interesting to investigate whether models perform even better for a specific season when trained exclusively on data from that season. We leave this for future investigation. Furthermore, the Covid-19 pandemic occurred during the considered data time window, resulting in emissions that differ from those observed in non-pandemic conditions. This bias should be considered when applying models trained on Covid-19 data to pandemic-free settings.

Appendix A: Further performance measures

In the following we describe further scale-insensitive performance measures to fill the gap between surface NO₂ measurements $x^\dagger \in \mathbb{R}^n$ and predictions x made by a machine learning model.

Coefficient of determination (R^2).

$$R^2(x^\dagger, x) = 1 - \frac{\sum_{i=1}^n |x_i^\dagger - x_i|}{\sum_{i=1}^n |x_i^\dagger - \bar{x}^\dagger|}, \quad \text{where } \bar{x}^\dagger = \frac{1}{n} \sum_{i=1}^n x_i^\dagger$$

Note that R^2 is similar to NMAE, but normalization is done by the mean absolute deviation of x^\dagger instead of its mean. Further, within the literature, the expression R^2 sometimes stands for the square of the correlation coefficient. However, in general, these definitions are not equivalent.

Index of agreement (IOA).

$$\text{IOA}(x^\dagger, x) = 1 - \frac{\sum_{i=1}^n |x_i^\dagger - x_i|^2}{\sum_{i=1}^n (|\bar{x}^\dagger - x_i| + |\bar{x}^\dagger - x_i^\dagger|)^2}$$

Here \bar{x}^\dagger denotes the mean of all x_i^\dagger .

Appendix B: Tables

Table B1. Features considered during feature selection in Sect. 3.1. For 200 splits into training and test stations, the Pearson correlation with surface NO₂ was computed on the training set for each available feature. Average correlations are shown in the last column.

	Feature name	Source	Average correlation with surface NO ₂
Selected features	Tropospheric vertical column density of NO ₂	IUP-UB retrieval on GEMS data	0.626
	Latitude at the center of GEMS pixel	GEMS data product	0.149
	Surface height at the center of GEMS pixel	GEMS data product	−0.185
	10 m <i>u</i> component of wind	ERA5	−0.105
	100 m <i>u</i> component of wind	ERA5	−0.112
	Instantaneous 10 m wind gust	ERA5	−0.237
	2 m temperature	ERA5	−0.252
	Surface pressure	ERA5	0.293
	Skin temperature	ERA5	−0.226
	UV–visible albedo for diffuse radiation	ERA5	0.297
	Downward UV radiation at the surface	ERA5	−0.217
	UV–visible albedo for direct radiation	ERA5	0.283
	Boundary layer height	ERA5	−0.318
	Total column water	ERA5	−0.212
	Evaporation	ERA5	0.239
	Soil type	ERA5	0.163
	High vegetation cover	ERA5	−0.130
Excluded features	Measuring time (hour)	Defined in Sect. 2.2	0.001
	Longitude at the center of GEMS pixel	GEMS data product	−0.054
	10 m <i>v</i> component of wind	ERA5	0.076
	100 m <i>v</i> component of wind	ERA5	0.076
	Vertical integral of temperature	ERA5	−0.009
	Total column ozone	ERA5	0.062

Table B2. Linear regression models have been trained on $D_{N,k}$ for $N \leq 4$ with different time contiguities k and input features selected in Sect. 3.1. Performance on $D_{4,k}$ has been evaluated through six-times 10-fold spatial cross-validation. Five different performance measures are considered, defined in Sect. 3.3 and Appendix A. The best results are marked in bold.

		Training datasets $D_{N,k}$									
		$D_{1,1}$	$D_{2,1}$	$D_{3,1}$	$D_{4,1}$	$D_{2,2}$	$D_{3,2}$	$D_{4,2}$	$D_{3,3}$	$D_{4,3}$	$D_{4,4}$
Correlation	Mean	0.6806	0.6895	0.6992	0.7015	0.7257	0.7321	0.7351	0.7402	0.7431	0.7469
	SD	0.0219	0.021	0.0207	0.0212	0.0199	0.0198	0.0201	0.0196	0.0198	0.0199
	Mean gain [%]	–	–	–	–	7.9109	10.0592	11.0761	12.7933	13.7819	15.0394
	SD gain [%]	–	–	–	–	1.788	1.6522	1.2735	1.699	1.4521	1.6349
NMSE	Mean	0.2298	0.2149	0.2006	0.1961	0.1897	0.1815	0.1776	0.1766	0.173	0.1709
	SD	0.0141	0.0128	0.0125	0.0135	0.0125	0.0128	0.0136	0.0129	0.0136	0.0137
	Mean gain [%]	–	–	–	–	3.0353	7.2854	9.3237	9.7677	11.6669	12.7688
	SD gain [%]	–	–	–	–	2.3991	1.4194	1.162	1.5324	1.3681	1.5287
NMAE	Mean	0.4357	0.4161	0.3926	0.3791	0.3769	0.3657	0.3573	0.3599	0.3519	0.3499
	SD	0.0164	0.0151	0.0135	0.0126	0.0127	0.0126	0.0127	0.0127	0.0127	0.0129
	Mean gain [%]	–	–	–	–	−0.6329	2.354	4.6017	3.922	6.0653	6.6
	SD gain [%]	–	–	–	–	1.464	1.0568	0.6454	1.1123	0.7738	0.8988
R^2	Mean	0.3984	0.4378	0.4754	0.4874	0.5038	0.5255	0.5359	0.5382	0.5479	0.5535
	SD	0.0432	0.0361	0.0311	0.0308	0.0324	0.0305	0.0305	0.0304	0.0303	0.0306
	Mean gain [%]	–	–	–	–	3.0353	7.2854	9.3237	9.7677	11.6669	12.7688
	SD gain [%]	–	–	–	–	2.3991	1.4195	1.162	1.5324	1.3681	1.5287
IOA	Mean	0.809	0.811	0.8096	0.8003	0.8381	0.8365	0.8283	0.8423	0.8349	0.8379
	SD	0.0145	0.0149	0.0164	0.0185	0.0145	0.0156	0.0173	0.0154	0.017	0.0169
	Mean gain [%]	–	–	–	–	14.0378	13.2159	8.9272	16.3166	12.3957	14.018
	SD gain [%]	–	–	–	–	1.5684	2.1544	2.9093	2.2224	2.9518	2.9977

Table B3. Random forests have been trained on $D_{N,k}$ for $N \leq 4$ with different time contiguities k and input features selected in Sect. 3.1. Performance on $D_{4,k}$ has been evaluated through six-times 10-fold spatial cross-validation. Five different performance measures are considered, defined in Sect. 3.3 and Appendix A. The best results are marked in bold.

		Training datasets $D_{N,k}$									
		$D_{1,1}$	$D_{2,1}$	$D_{3,1}$	$D_{4,1}$	$D_{2,2}$	$D_{3,2}$	$D_{4,2}$	$D_{3,3}$	$D_{4,3}$	$D_{4,4}$
Correlation	Mean	0.7993	0.8	0.8023	0.8018	0.8119	0.812	0.8114	0.8164	0.8159	0.8173
	SD	0.0213	0.0213	0.0216	0.0223	0.0208	0.0209	0.0213	0.021	0.0212	0.0211
	Mean gain [%]	–	–	–	–	4.5676	4.6283	4.3439	6.8605	6.6466	7.3194
	SD gain [%]	–	–	–	–	1.4329	1.4029	1.3676	1.6319	1.649	1.7219
NMSE	Mean	0.1417	0.141	0.1389	0.1389	0.1327	0.1326	0.1328	0.1298	0.13	0.1292
	SD	0.0155	0.0155	0.0155	0.016	0.0153	0.0154	0.0156	0.0154	0.0155	0.0155
	Mean gain [%]	–	–	–	–	4.0239	4.153	4.015	6.2	6.0405	6.6102
	SD gain [%]	–	–	–	–	1.2284	1.2229	1.3537	1.5193	1.6428	1.7201
NMAE	Mean	0.3258	0.3238	0.3184	0.3144	0.3075	0.3066	0.3049	0.3014	0.3006	0.2995
	SD	0.0168	0.0165	0.0158	0.0152	0.0151	0.0149	0.0146	0.0148	0.0146	0.0145
	Mean gain [%]	–	–	–	–	2.1838	2.4769	3.0019	4.1298	4.3647	4.7212
	SD gain [%]	–	–	–	–	0.6003	0.545	0.5486	0.6267	0.6423	0.6722
R^2	Mean	0.6301	0.632	0.6373	0.6375	0.6535	0.654	0.6534	0.6613	0.6607	0.6627
	SD	0.0337	0.0337	0.0342	0.0355	0.0336	0.0338	0.0344	0.0341	0.0345	0.0344
	Mean gain [%]	–	–	–	–	4.0239	4.153	4.015	6.2	6.0405	6.6102
	SD gain [%]	–	–	–	–	1.2284	1.2229	1.3537	1.5193	1.6428	1.7201
IOA	Mean	0.8752	0.8756	0.8768	0.875	0.8846	0.8846	0.8833	0.887	0.886	0.8866
	SD	0.0153	0.0153	0.0155	0.0162	0.015	0.0151	0.0154	0.0151	0.0153	0.0153
	Mean gain [%]	–	–	–	–	6.3027	6.3035	5.2754	8.2736	7.5138	7.9427
	SD gain [%]	–	–	–	–	1.4278	1.498	1.6812	1.8665	2.0031	2.0893

Appendix C: Additional figures for Experiment 2

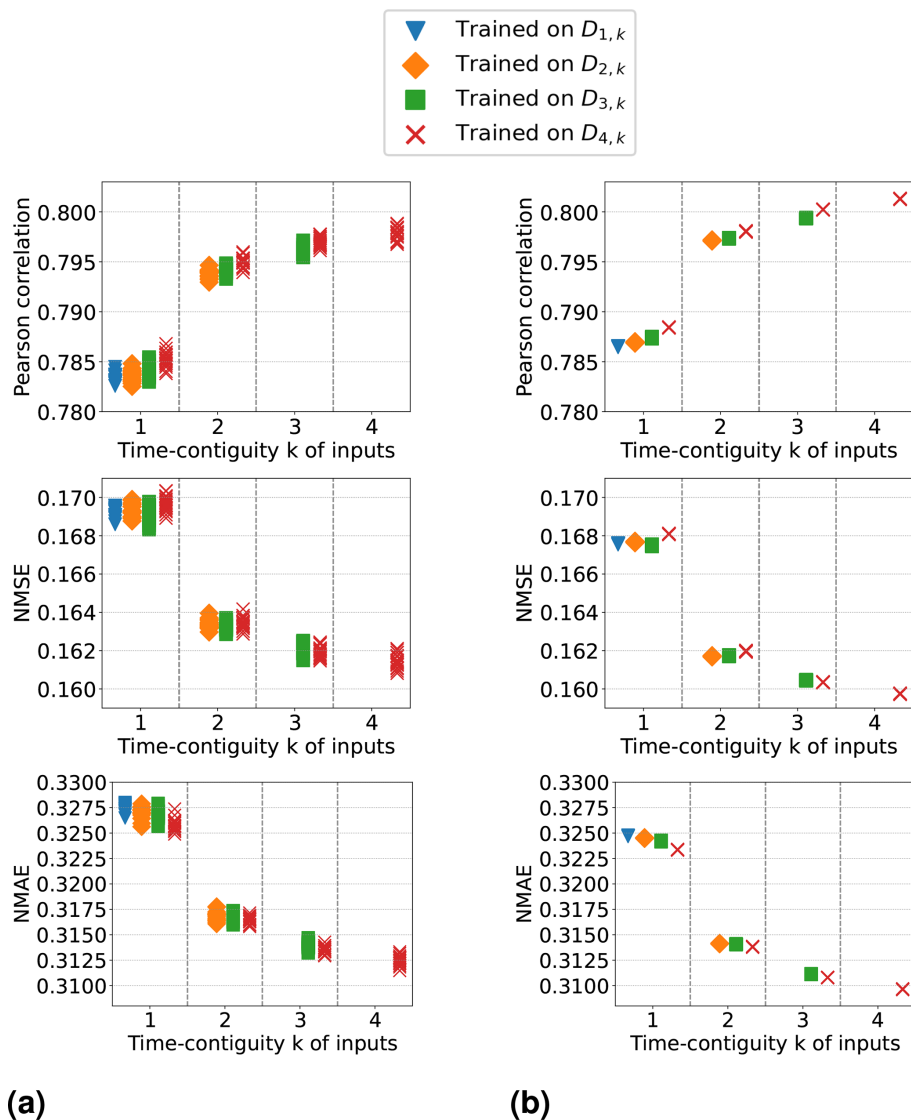


Figure C1. Random forests with 30 and 8000 trees (n_estimators) are considered in panels (a) and (b), respectively. Training and testing have been performed 20 times for the same split into training and test stations. Testing was on the corresponding dataset $D_{4,k}$, and training was on different $D_{M,k}$ for $M \leq 4$. Results for 20 individual repetitions are shown with respect to three performance measures.

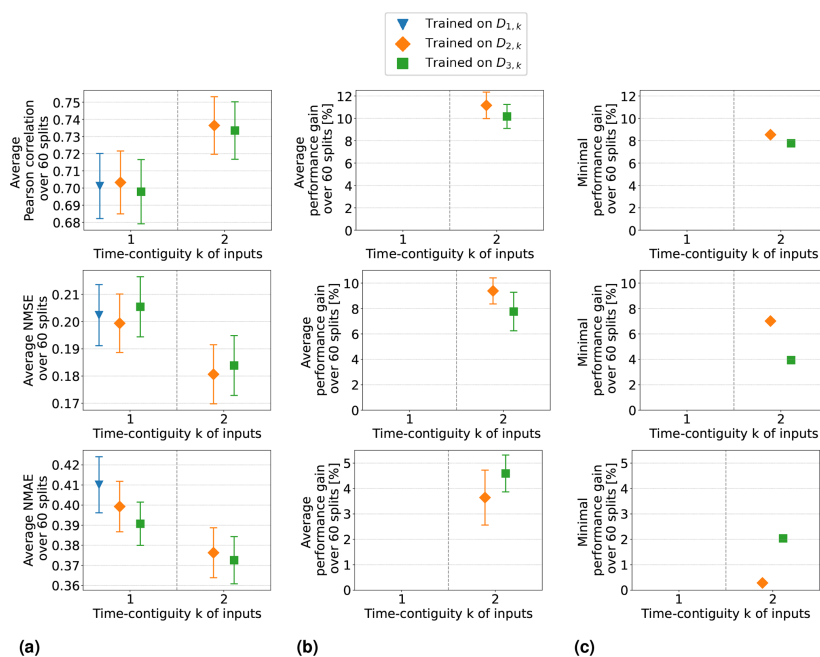


Figure C2. Linear regression models have been trained on $D_{M,k}$ for $M \leq 3$ with different time contiguities k and input features selected in Sect. 3.1. Performance on $D_{2,k}$ has been evaluated through six-times 10-fold spatial cross-validation. Panel (a) shows the average performance over all 60 station splits for three performance measures. Panel (b) shows the average performance gain (Eq. 6) relative to the best case of $k = 1$. Error bars illustrate the standard deviation. Panel (c) shows the minimal performance gain. Across each row the same performance measure is considered.

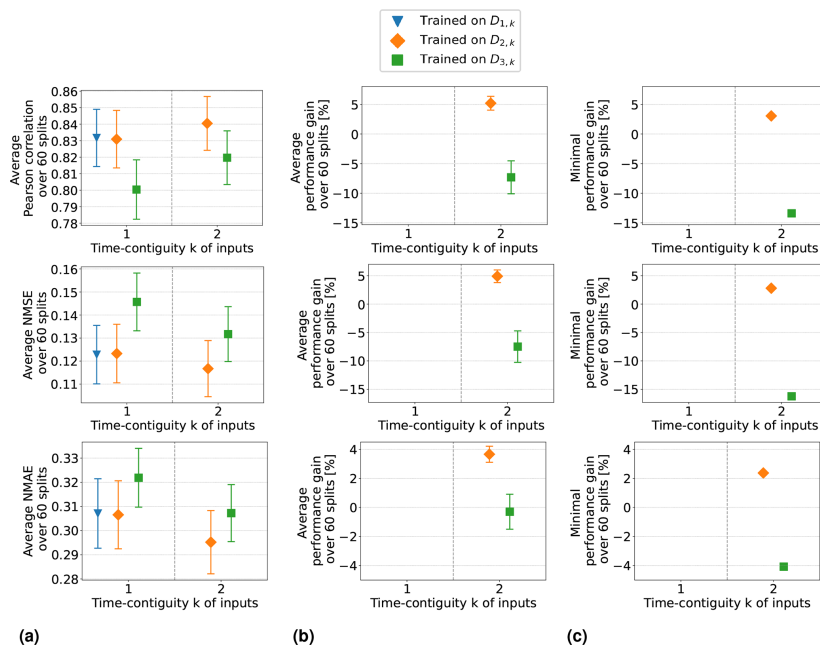


Figure C3. Same as Fig. C2 but for random forests trained on $D_{M,k}$ for $M \leq 3$ with different time contiguities k and input features selected in Sect. 3.1. Performance on $D_{2,k}$ has been evaluated through six-times 10-fold spatial cross-validation. Panel (a) shows the average performance over all 60 station splits for three performance measures. Panel (b) shows the average performance gain (Eq. 6) relative to the best case of $k = 1$. Error bars illustrate the standard deviation. Panel (c) shows the minimal performance gain. Across each row the same performance measure is considered.

Appendix D: Additional figures for Experiment 3

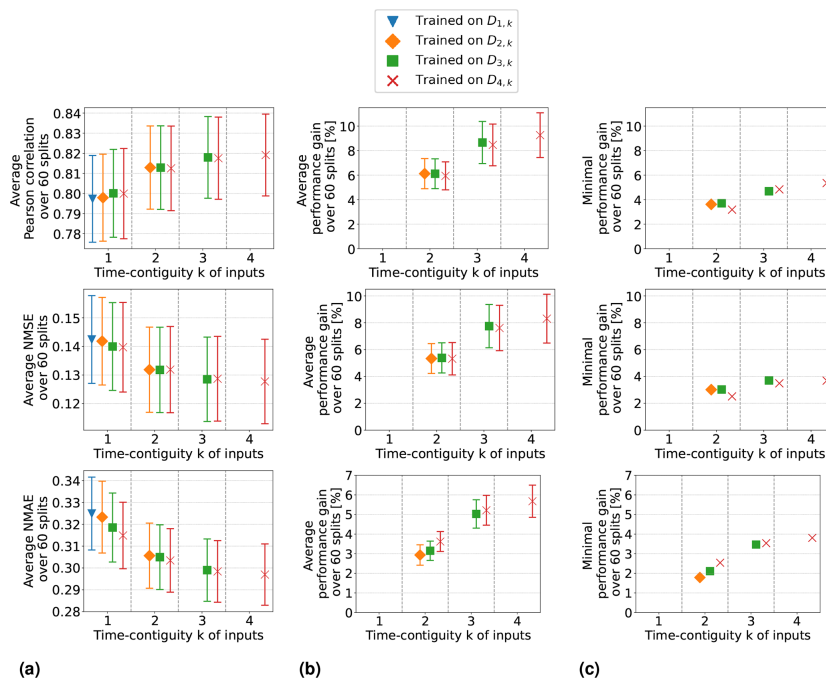


Figure D1. Excluded latitude and surface height from input features (Setting 3 of Experiment 3): random forests have been trained on $D_{M,k}$ for $M \leq 4$ with different time contiguities k . Performance on $D_{4,k}$ has been evaluated through six-times 10-fold spatial cross-validation. Panel (a) shows the average performance over all 60 station splits for three performance measures. Panel (b) shows the average performance gain relative to the best case of $k = 1$; see Eq. (6) for the definition of performance gain. Error bars illustrate the standard deviation. Panel (c) shows the minimal performance gain. Across each row the same performance measure is considered.

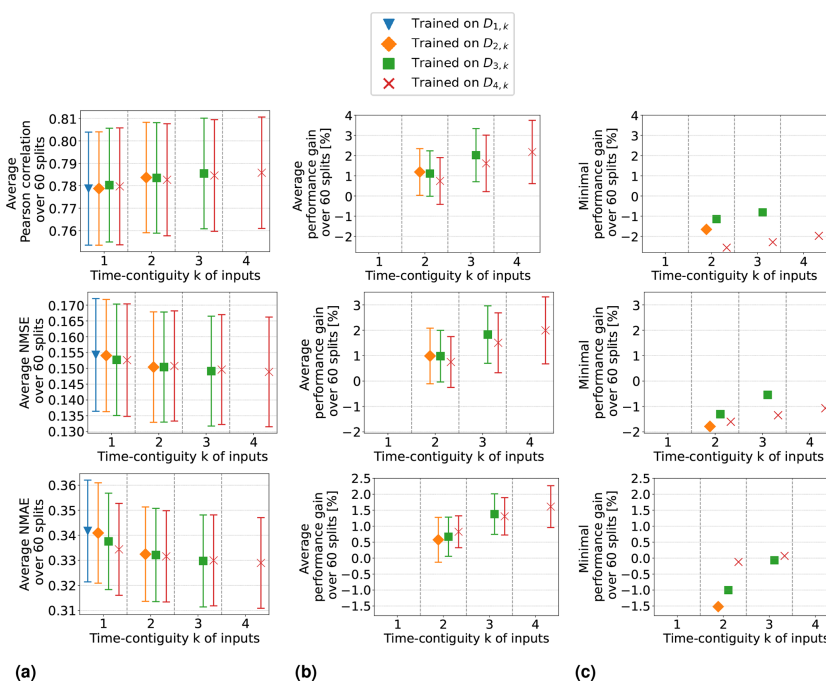


Figure D2. Same as Fig. D1 but excluding tropospheric NO₂ VCDs from input features (Setting 2 of Experiment 3).

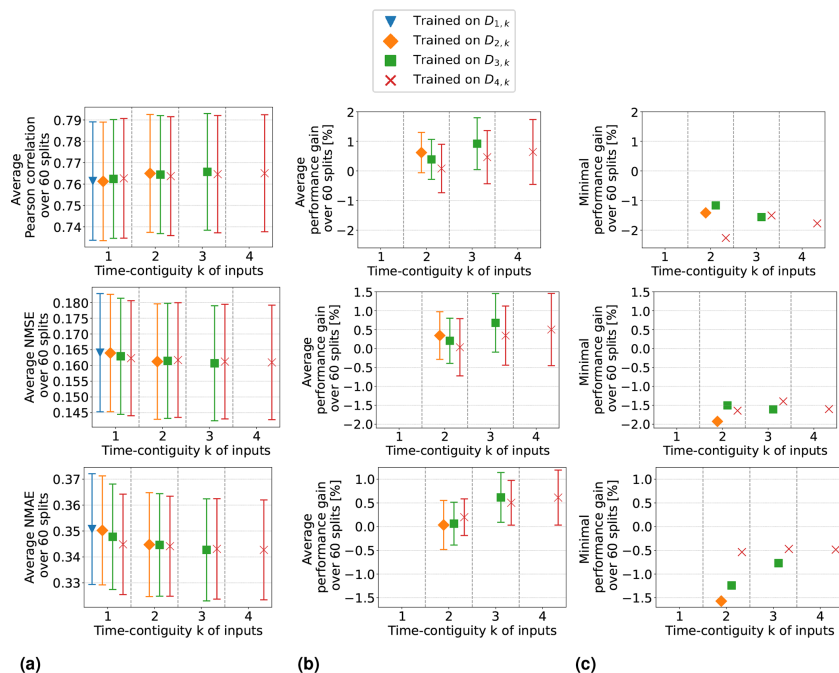


Figure D3. Same as Fig. D1 but excluding tropospheric NO₂ VCDs, latitude, and surface height from input features (Setting 4 of Experiment 3).

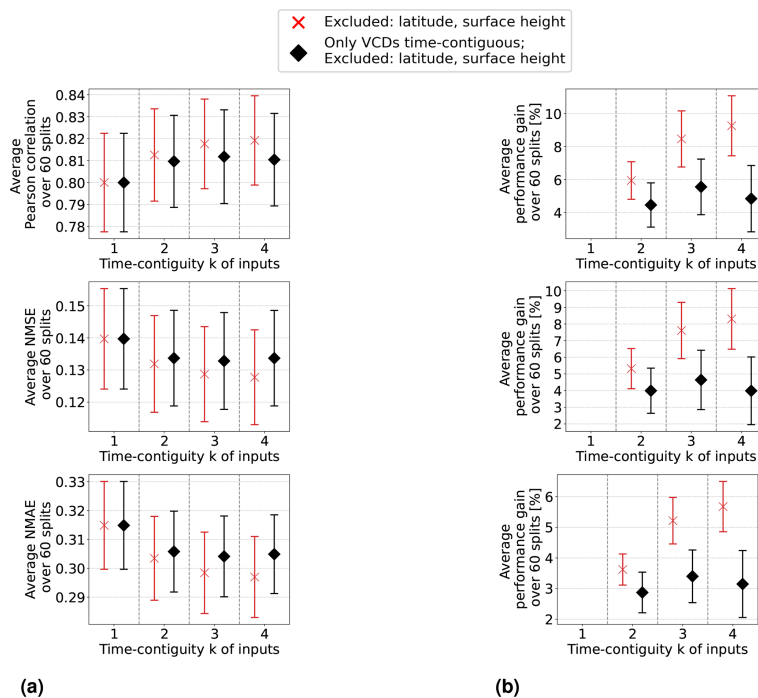


Figure D4. Random forests: the selection of input features is the same as in Setting 3 of Experiment 3; i.e., latitude and surface height are excluded. Comparison of two cases. First, only the time contiguity of tropospheric NO₂ VCDs is exploited. Second, the time contiguity of all (time-dependent) input features is exploited, which is exactly the same as Setting 3 of Experiment 3. Models have been trained and tested on $D_{4,k}$ for different time contiguities k . Panel (a) shows the average performance from six-times 10-fold spatial cross-validation and panel (b) shows the average performance gain (Eq. 6).

Code and data availability. The GEMS datasets and codes are available upon request from the corresponding author (janek-goedeke@uni-bremen.de). The ERA5 data are available from the Copernicus Climate Change Service at <https://doi.org/10.24381/cds.adbb2d47> (Hersbach et al., 2023). AirKorea surface network data are available at https://airkorea.or.kr/web/detailViewDown?pMENU_NO=125 (National Institute of Environmental Research (NIER), 2025).

Author contributions. JG is the main author of this study and planned and conducted the experiments. AR and KL provided GEMS data. PM, AR, and KL contributed to the design of the study and the discussion of the results. HH, HL, and JP provided in situ data and expertise on GEMS measurements. All authors contributed to the paper.

Competing interests. At least one of the (co-)authors is a member of the editorial board of *Atmospheric Measurement Techniques*. The peer-review process was guided by an independent editor, and the authors also have no other competing interests to declare.

Disclaimer. The results contain modified Copernicus Climate Change Service information from 2020. Neither the European Commission nor ECMWF is responsible for any use that may be made of the Copernicus information or data it contains.

Publisher's note: Copernicus Publications remains neutral with regard to jurisdictional claims made in the text, published maps, institutional affiliations, or any other geographical representation in this paper. While Copernicus Publications makes every effort to include appropriate place names, the final responsibility lies with the authors.

Acknowledgements. We thank the National Institute of Environmental Research (NIER) of South Korea for providing GEMS lv1 data and financial support (NIER-2022-04-02-037). Hersbach et al. (2023) was downloaded from the Copernicus Climate Change Service (2023). We thank the Korean Ministry of Environment and NIER for providing the in situ measurements of surface NO₂. Janek Gödeke and Kezia Lange acknowledge funding by the Deutsches Zentrum für Luft- und Raumfahrt (grant no. 50 EE 2204). Further, we thank Pascal Fernsel from the University of Bremen for fruitful discussions and feedback.

Financial support. This research has been supported by the Deutsches Zentrum für Luft- und Raumfahrt (grant no. 50 EE 2204) and the National Institute of Environmental Research of South Korea (grant no. NIER-2022-04-02-037).

The article processing charges for this open-access publication were covered by the University of Bremen.

Review statement. This paper was edited by Diego Loyola and reviewed by two anonymous referees.

References

- Ahmad, N., Lin, C., Lau, A. K. H., Kim, J., Zhang, T., Yu, F., Li, C., Li, Y., Fung, J. C. H., and Lao, X. Q.: Estimation of ground-level NO₂ and its spatiotemporal variations in China using GEMS measurements and a nested machine learning model, *Atmos. Chem. Phys.*, 24, 9645–9665, <https://doi.org/10.5194/acp-24-9645-2024>, 2024.
- Bechle, M. J., Millet, D. B., and Marshall, J. D.: Remote sensing of exposure to NO₂: Satellite versus ground-based measurement in a large urban area, *Atmos. Environ.*, 69, 345–353, <https://doi.org/10.1016/j.atmosenv.2012.11.046>, 2013.
- Beirle, S., Hörmann, C., Jöckel, P., Liu, S., Penning de Vries, M., Pozzer, A., Sihler, H., Valks, P., and Wagner, T.: The STRatospheric Estimation Algorithm from Mainz (STREAM): estimating stratospheric NO₂ from nadir-viewing satellites by weighted convolution, *Atmos. Meas. Tech.*, 9, 2753–2779, <https://doi.org/10.5194/amt-9-2753-2016>, 2016.
- Bovensmann, H., Burrows, J., Buchwitz, M., Frerick, J., Noel, S., Rozanov, V., Chance, Kelly, and Goede, A.: SCIAMACHY: mission objectives and measurement modes, *J. Atmos. Sci.*, 56, 127–150 [https://doi.org/10.1175/1520-0469\(1999\)056<0127:SMOAMM>2.0.CO;2](https://doi.org/10.1175/1520-0469(1999)056<0127:SMOAMM>2.0.CO;2), 1999.
- Breiman, L.: Random Forests, *Machine Learning*, 45, 5–32, <https://doi.org/10.1023/A:1010933404324>, 2001.
- Burrows, J. P., Weber, M., Buchwitz, M., Rozanov, V., Ladstätter-Weissenmayer, A., Richter, A., DeBeek, R., Hoogen, R., Bramstedt, K., Eichmann, K.-U., Eisinger, M., and Perner, D.: The Global Ozone Monitoring Experiment (GOME): Mission Concept and First Scientific Results, *J. Atmos. Sci.*, 56, 151–175, [https://doi.org/10.1175/1520-0469\(1999\)056<0151:TGOMEG>2.0.CO;2](https://doi.org/10.1175/1520-0469(1999)056<0151:TGOMEG>2.0.CO;2), 1999.
- Chan, K. L., Khorsandi, E., Liu, S., Baier, F., and Valks, P.: Estimation of Surface NO₂ Concentrations over Germany from TROPOMI Satellite Observations Using a Machine Learning Method, *Remote Sensing*, 13, 969, <https://doi.org/10.3390/rs13050969>, 2021.
- Chen, Z.-Y., Zhang, R., Zhang, T.-H., Ou, C.-Q., and Guo, Y.: A kriging-calibrated machine learning method for estimating daily ground-level NO₂ in mainland China, *Sci. Total Environ.*, 690, 556–564, <https://doi.org/10.1016/j.scitotenv.2019.06.349>, 2019.
- Cooper, M., Martin, R., Hammer, M., Levelt, P. F., Veefkind, P., Lamsal, L. N., Krotkov, N. A., Brook, J. R., and McLinden, C. A.: Global fine-scale changes in ambient NO₂ during COVID-19 lockdowns, *Nature*, 601, 380–387, <https://doi.org/10.1038/s41586-021-04229-0>, 2022.
- Cooper, M. J., Martin, R. V., McLinden, C. A., and Brook, J. R.: Inferring ground-level nitrogen dioxide concentrations at fine spatial resolution applied to the TROPOMI satellite instrument, *Environ. Res. Lett.*, 15, 104013, <https://doi.org/10.1088/1748-9326/aba3a5>, 2020.
- Copernicus Climate Change Service: ERA5 hourly data on single levels from 1940 to present, copernicus Climate Change Service (C3S) Climate Data Store (CDS) [data set], <https://doi.org/10.24381/cds.adbb2d47>, 2023.

- de Hoogh, K., Saucy, A., Shtein, A., Schwartz, J., West, E. A., Strassmann, A., Puhon, M., Rössli, M., Stafoggia, M., and Kloog, I.: Predicting Fine-Scale Daily NO₂ for 2005–2016 Incorporating OMI Satellite Data Across Switzerland, *Environ. Sci. Technol.*, 53, 10279–10287, <https://doi.org/10.1021/acs.est.9b03107>, 2019.
- Di, Q., Amini, H., Shi, L., Kloog, I., Silvern, R., Kelly, J., Sabath, M. B., Choirat, C., Koutrakis, P., Lyapustin, A., Wang, Y., Mickley, L. J., and Schwartz, J.: Assessing NO₂ Concentration and Model Uncertainty with High Spatiotemporal Resolution across the Contiguous United States Using Ensemble Model Averaging, *Environ. Sci. Technol.*, 54, 1372–1384, <https://doi.org/10.1021/acs.est.9b03358>, 2020.
- Dou, X., Liao, C., Wang, H., Huang, Y., Tu, Y., Huang, X., Peng, Y., Zhu, B., Tan, J., Deng, Z., Wu, N., Sun, T., Ke, P., and Liu, Z.: Estimates of daily ground-level NO₂ concentrations in China based on Random Forest model integrated K-means, *Advances in Applied Energy*, 2, 100017, <https://doi.org/10.1016/j.adapen.2021.100017>, 2021.
- Geddes, J. A., Martin, R. V., Boys, B. L., and van Donkelaar, A.: Long-Term Trends Worldwide in Ambient NO₂ Concentrations Inferred from Satellite Observations, *Environ. Health Persp.*, 124, 281–289, <https://doi.org/10.1289/ehp.1409567>, 2016.
- Genuer, R., Poggi, J.-M., and Tuleau, C.: Random Forests: some methodological insights, *arXiv [preprint]*, <https://doi.org/10.48550/arXiv.0811.3619>, 21 November 2008.
- Ghahremanloo, M., Lops, Y., Choi, Y., and Yeganeh, B.: Deep Learning Estimation of Daily Ground-Level NO₂ Concentrations From Remote Sensing Data, *J. Geophys. Res.-Atmos.*, 126, e2021JD034925, <https://doi.org/10.1029/2021JD034925>, 2021.
- Gu, J., Chen, L., Yu, C., Li, S., Tao, J., Fan, M., Xiong, X., Wang, Z., Shang, H., and Su, L.: Ground-Level NO₂ Concentrations over China Inferred from the Satellite OMI and CMAQ Model Simulations, *Remote Sensing*, 9, 519, <https://doi.org/10.3390/rs9060519>, 2017.
- Hersbach, H., Bell, B., Berrisford, P., Biavati, G., Horányi, A., Muñoz Sabater, J., Nicolas, J., Peubey, C., Radu, R., Rozum, I., Schepers, D., Simmons, A., Soci, C., Dee, D., and Thépaut, J.-N.: ERA5 hourly data on single levels from 1940 to present, Copernicus Climate Change Service (C3S) Climate Data Store (CDS) [data set], <https://doi.org/10.24381/cds.adbb2d47>, 2023.
- Huang, K., Zhu, Q., Lu, X., Gu, D., and Liu, Y.: Satellite-Based Long-Term Spatiotemporal Trends in Ambient NO₂ Concentrations and Attributable Health Burdens in China From 2005 to 2020, *GeoHealth*, 7, e2023GH000798, <https://doi.org/10.1029/2023GH000798>, 2023.
- Jacob, D. J.: Introduction to Atmospheric Chemistry, Princeton University Press, Princeton, <https://doi.org/10.1515/9781400841547>, ISBN 9781400841547, 2000.
- Jiang, Q. and Christakos, G.: Space-time mapping of ground-level PM_{2.5} and NO₂ concentrations in heavily polluted northern China during winter using the Bayesian maximum entropy technique with satellite data, *Air Qual. Atmos. Hlth.*, 11, 23–33, <https://doi.org/10.1007/s11869-017-0514-8>, 2018.
- Kharol, S., Martin, R., Philip, S., Boys, B., Lamsal, L., Jerrett, M., Brauer, M., Crouse, D., McLinden, C., and Burnett, R.: Assessment of the magnitude and recent trends in satellite-derived ground-level nitrogen dioxide over North America, *Atmos. Environ.*, 118, 236–245, <https://doi.org/10.1016/j.atmosenv.2015.08.011>, 2015.
- Kim, D., Lee, H., Hong, H., Choi, W., Lee, Y. G., and Park, J.: Estimation of Surface NO₂ Volume Mixing Ratio in Four Metropolitan Cities in Korea Using Multiple Regression Models with OMI and AIRS Data, *Remote Sensing*, 9, 627, <https://doi.org/10.3390/rs9060627>, 2017.
- Kim, J., Jeong, U., Ahn, M.-H., Kim, J. H., Park, R. J., Lee, H., Song, C. H., Choi, Y.-S., Lee, K.-H., Yoo, J.-M., Jeong, M.-J., Park, S. K., Lee, K.-M., Song, C.-K., Kim, S.-W., Kim, Y. J., Kim, S.-W., Kim, M., Go, S., Liu, X., Chance, K., Miller, C. C., Al-Saadi, J., Veihelmann, B., Bhartia, P. K., Torres, O., Abad, G. G., Haffner, D. P., Ko, D. H., Lee, S. H., Woo, J.-H., Chong, H., Park, S. S., Nicks, D., Choi, W. J., Moon, K.-J., Cho, A., Yoon, J., Kim, S.-k., Hong, H., Lee, K., Lee, H., Lee, S., Choi, M., Veeckind, P., Levelt, P. F., Edwards, D. P., Kang, M., Eo, M., Bak, J., Baek, K., Kwon, H.-A., Yang, J., Park, J., Han, K. M., Kim, B.-R., Shin, H.-W., Choi, H., Lee, E., Chong, J., Cha, Y., Koo, J.-H., Irie, H., Hayashida, S., Kasai, Y., Kanaya, Y., Liu, C., Lin, J., Crawford, J. H., Carmichael, G. R., Newchurch, M. J., Lefer, B. L., Herman, J. R., Swap, R. J., Lau, A. K. H., Kurosu, T. P., Jaross, G., Ahlers, B., Dobber, M., McElroy, C. T., and Choi, Y.: New Era of Air Quality Monitoring from Space: Geostationary Environment Monitoring Spectrometer (GEMS), *B. Am. Meteorol. Soc.*, 101, E1–E22, <https://doi.org/10.1175/BAMS-D-18-0013.1>, 2020.
- Kim, M., Brunner, D., and Kuhlmann, G.: Importance of satellite observations for high-resolution mapping of near-surface NO₂ by machine learning, *Remote Sens. Environ.*, 264, 112573, <https://doi.org/10.1016/j.rse.2021.112573>, 2021.
- Kley, D. and McFarland, M.: Chemiluminescence detector for NO and NO/sub 2/, *Atmos. Technol. (United States)*, 12, <https://www.osti.gov/biblio/6457230>, 1980.
- Lamsal, L. N., Martin, R. V., van Donkelaar, A., Steinbacher, M., Celarier, E. A., Bucsela, E., Dunlea, E. J., and Pinto, J. P.: Ground-level nitrogen dioxide concentrations inferred from the satellite-borne Ozone Monitoring Instrument, *J. Geophys. Res.-Atmos.*, 113, D16308, <https://doi.org/10.1029/2007JD009235>, 2008.
- Lamsal, L. N., Martin, R. V., van Donkelaar, A., Celarier, E. A., Bucsela, E. J., Boersma, K. F., Dirksen, R., Luo, C., and Wang, Y.: Indirect validation of tropospheric nitrogen dioxide retrieved from the OMI satellite instrument: Insight into the seasonal variation of nitrogen oxides at northern midlatitudes, *J. Geophys. Res.-Atmos.*, 115, D05302, <https://doi.org/10.1029/2009JD013351>, 2010.
- Lamsal, L. N., Martin, R. V., Parrish, D. D., and Krotkov, N. A.: Scaling Relationship for NO₂ Pollution and Urban Population Size: A Satellite Perspective, *Environ. Sci. Technol.*, 47, 7855–7861, <https://doi.org/10.1021/es400744g>, 2013.
- Lange, K., Richter, A., Bösch, T., Zilker, B., Latsch, M., Behrens, L. K., Okafor, C. M., Bösch, H., Burrows, J. P., Merlaud, A., Pinardi, G., Fayt, C., Friedrich, M. M., Dimitropoulou, E., Van Roozendaal, M., Ziegler, S., Ripperger-Lukosiunaite, S., Kuhn, L., Lauster, B., Wagner, T., Hong, H., Kim, D., Chang, L.-S., Bae, K., Song, C.-K., Park, J.-U., and Lee, H.: Validation of GEMS tropospheric NO₂ columns and their diurnal variation with ground-based DOAS measurements, *At-*

- mos. Meas. Tech., 17, 6315–6344, <https://doi.org/10.5194/amt-17-6315-2024>, 2024.
- Lee, H. J., Kim, N. R., and Shin, M. Y.: Capabilities of satellite Geostationary Environment Monitoring Spectrometer (GEMS) NO₂ data for hourly ambient NO₂ exposure modeling, *Environ. Res.*, 261, 119633, <https://doi.org/10.1016/j.envres.2024.119633>, 2024.
- Levelt, P., van den Oord, G., Dobber, M., Malkki, A., Visser, H., de Vries, J., Stammes, P., Lundell, J., and Saari, H.: The ozone monitoring instrument, *IEEE T. Geosci. Remote*, 44, 1093–1101, <https://doi.org/10.1109/TGRS.2006.872333>, 2006.
- Li, M., Wu, Y., Bao, Y., Liu, B., and Petropoulos, G. P.: Near-Surface NO₂ Concentration Estimation by Random Forest Modeling and Sentinel-5P and Ancillary Data, *Remote Sensing*, 14, 3612, <https://doi.org/10.3390/rs14153612>, 2022.
- National Institute of Environmental Research (NIER): AirKorea Annual Report, NIER [data set], https://airkorea.or.kr/web/detailViewDown?pMENU_NO=125, last access: 30 July 2025 (in Korean).
- Oak, Y. J., Jacob, D. J., Balasus, N., Yang, L. H., Chong, H., Park, J., Lee, H., Lee, G. T., Ha, E. S., Park, R. J., Kwon, H.-A., and Kim, J.: A bias-corrected GEMS geostationary satellite product for nitrogen dioxide using machine learning to enforce consistency with the TROPOMI satellite instrument, *Atmos. Meas. Tech.*, 17, 5147–5159, <https://doi.org/10.5194/amt-17-5147-2024>, 2024.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E.: Scikit-learn: Machine Learning in Python, *J. Mach. Learn. Res.*, 12, 2825–2830, 2011.
- Probst, P., Bischl, B., and Boulesteix, A.-L.: Tunability: Importance of Hyperparameters of Machine Learning Algorithms, *arXiv [preprint]*, <https://doi.org/10.48550/arXiv.1802.09596>, 22 October 2018.
- Probst, P., Wright, M. N., and Boulesteix, A.-L.: Hyperparameters and tuning strategies for random forest, *WIREs Data Mining and Knowledge Discovery*, 9, e1301, <https://doi.org/10.1002/widm.1301>, 2019.
- Qin, K., Han, X., Li, D., Xu, J., Loyola, D., Xue, Y., Zhou, X., Li, D., Zhang, K., and Yuan, L.: Satellite-based estimation of surface NO₂ concentrations over east-central China: A comparison of POMINO and OMNO2d data, *Atmos. Environ.*, 224, 117322, <https://doi.org/10.1016/j.atmosenv.2020.117322>, 2020.
- Scornet, E.: Tuning parameters in random forests, *ESAIM: Procs.*, 60, 144–162, <https://doi.org/10.1051/proc/201760144>, 2017.
- Shetty, S., Schneider, P., Stebel, K., David Hamer, P., Kylling, A., and Koren Berntsen, T.: Estimating surface NO₂ concentrations over Europe using Sentinel-5P TROPOMI observations and Machine Learning, *Remote Sens. Environ.*, 312, 114321, <https://doi.org/10.1016/j.rse.2024.114321>, 2024.
- Siddique, M. A., Naseer, E., Usama, M., and Basit, A.: Estimation of Surface-Level NO₂ Using Satellite Remote Sensing and Machine Learning: A review, *IEEE Geoscience and Remote Sensing Magazine*, 12, 2–28, <https://doi.org/10.1109/MGRS.2024.3398434>, 2024.
- Tang, B., Stanier, C. O., Carmichael, G. R., and Gao, M.: Ozone, nitrogen dioxide, and PM_{2.5} estimation from observation-model machine learning fusion over S. Korea: Influence of observation density, chemical transport model resolution, and geostationary remotely sensed AOD, *Atmos. Environ.*, 331, 120603, <https://doi.org/10.1016/j.atmosenv.2024.120603>, 2024.
- Veefkind, J., Aben, I., McMullan, K., Förster, H., de Vries, J., Otter, G., Claas, J., Eskes, H., de Haan, J., Kleipool, Q., van Weele, M., Hasekamp, O., Hoogeveen, R., Landgraf, J., Snel, R., Tol, P., Ingmann, P., Voors, R., Kruizinga, B., Vink, R., Visser, H., and Levelt, P.: TROPOMI on the ESA Sentinel-5 Precursor: A GMES mission for global observations of the atmospheric composition for climate, air quality and ozone layer applications, *Remote Sens. Environ.*, 120, 70–83, <https://doi.org/10.1016/j.rse.2011.09.027>, 2012.
- Wang, B. and Chen, Z.: An intercomparison of satellite-derived ground-level NO₂ concentrations with GMSMB modeling results and in-situ measurements – A North American study, *Environ. Pollut.*, 181, 172–181, <https://doi.org/10.1016/j.envpol.2013.06.037>, 2013.
- Wei, J., Liu, S., Li, Z., Liu, C., Qin, K., Liu, X., Pinker, R. T., Dickerson, R. R., Lin, J., Boersma, K. F., Sun, L., Li, R., Xue, W., Cui, Y., Zhang, C., and Wang, J.: Ground-Level NO₂ Surveillance from Space Across China for High Resolution Using Interpretable Spatiotemporally Weighted Artificial Intelligence, *Environ. Sci. Technol.*, 56, 9988–9998, <https://doi.org/10.1021/acs.est.2c03834>, 2022.
- Williams, J. E., Boersma, K. F., Le Sager, P., and Verstraeten, W. W.: The high-resolution version of TM5-MP for optimized satellite retrievals: description and validation, *Geosci. Model Dev.*, 10, 721–750, <https://doi.org/10.5194/gmd-10-721-2017>, 2017.
- Yang, L. H., Jacob, D. J., Colombi, N. K., Zhai, S., Bates, K. H., Shah, V., Beaudry, E., Yantosca, R. M., Lin, H., Brewer, J. F., Chong, H., Travis, K. R., Crawford, J. H., Lamsal, L. N., Koo, J.-H., and Kim, J.: Tropospheric NO₂ vertical profiles over South Korea and their relation to oxidant chemistry: implications for geostationary satellite retrievals and the observation of NO₂ diurnal variation from space, *Atmos. Chem. Phys.*, 23, 2465–2481, <https://doi.org/10.5194/acp-23-2465-2023>, 2023a.
- Yang, Q., Kim, J., Cho, Y., Lee, W.-J., Lee, D.-W., Yuan, Q., Wang, F., Zhou, C., Zhang, X., Xiao, X., Guo, M., Guo, Y., Carmichael, G. R., and Gao, M.: A synchronized estimation of hourly surface concentrations of six criteria air pollutants with GEMS data, *npj Clim. Atmos. Sci.*, 6, 94, <https://doi.org/10.1038/s41612-023-00407-1>, 2023b.
- Zhang, Y., Lin, J., Kim, J., Lee, H., Park, J., Hong, H., Van Roozendael, M., Hendrick, F., Wang, T., Wang, P., He, Q., Qin, K., Choi, Y., Kanaya, Y., Xu, J., Xie, P., Tian, X., Zhang, S., Wang, S., Cheng, S., Cheng, X., Ma, J., Wagner, T., Spurr, R., Chen, L., Kong, H., and Liu, M.: A research product for tropospheric NO₂ columns from Geostationary Environment Monitoring Spectrometer based on Peking University OMI NO₂ algorithm, *Atmos. Meas. Tech.*, 16, 4643–4665, <https://doi.org/10.5194/amt-16-4643-2023>, 2023.