Machine Learning (2023) 112:1337–1363
https://doi.org/10.1007/s10994-022-06155-2

Surrogate models of radiative transfer codes for atmospheric trace gas retrievals from satellite observations

Jure Brence[1,2] · **Jovan Tanevski**[1,3] · **Jennifer Adams**[4,5] · **Edward Malina**[6] · **Sašo Džeroski**[1,2,7]

Received: 7 March 2021 / Revised: 6 December 2021 / Accepted: 19 February 2022 /
Published online: 22 March 2022

Abstract
Inversion of radiative transfer models (RTMs) is key to interpreting satellite observations of air quality and greenhouse gases, but is computationally expensive. Surrogate models that emulate the full forward physical RTM can speed up the simulation, reducing computational and timing costs and allowing the use of more advanced physics for trace gas retrievals. In this study, we present the development of surrogate models for two RTMs: the RemoTeC algorithm using the LINTRAN RTM and the SCIATRAN RTM. We estimate the intrinsic dimensionality of the input and output spaces and embed them in lower dimensional subspaces to facilitate the learning task. Two methods are tested for dimensionality reduction, autoencoders and principle component analysis (PCA), with PCA consistently outperforming autoencoders. Different sampling methods are employed for generating the training datasets: sampling focused on expected atmospheric parameters and latin hypercube sampling. The results show that models trained on the smaller (n = 1000) uniformly sampled dataset can perform as well as those trained on the larger (n = 50000), more focused dataset. Surrogate models for both datasets are able to accurately emulate Sentinel 5P spectra within a millisecond or less, as compared to the minutes or hours needed to simulate the full physical model. The SCIATRAN-trained forward surrogate models are able to generalize the emulation to a broader set of parameters and can be used for less constrained applications, while achieving a normalized RMSE of 7.3%. On the other hand, models trained on the LINTRAN dataset can completely replace the RTM simulation in more focused expected ranges of atmospheric parameters, as they achieve a normalized RMSE of 0.3%.

Keywords Surrogate models · Dimensionality reduction · Intrinsic dimensionality · Radiative transfer · Satellite data · Neural networks · Tree ensembles

Editors: Annalisa Appice, Grigorios Tsoumakas.

Jure Brence and Jovan Tanevski have contributed equally to this work.

✉ Jure Brence
jure.brence@ijs.si

Extended author information available on the last page of the article

1 Introduction

The TROPOspheric Monitoring Instrument (TROPOMI) onboard the Copernicus Sentinel-5 Precursor (S5P) satellite is an important step forward in Earth observation. Launched in 2017, TROPOMI provides global information on air quality and greenhouse gases, such as methane, carbon monoxide and water vapour, as well as many others. TROPOMI products (freely available at https://s5phub.copernicus.eu/dhus), such as gas concentrations, are generated through schemes knows as atmospheric retrievals. The retrieval schemes of some of these products, for example methane, typically rely on "optimal estimation methods", which although well proven, require large processing resources due to the running of forward models. In the case of TROPOMI trace gas retrievals, these forward models or Radiative Transfer Models (RTMs) are physical models that allow us to understand and retrieve atmospheric constituents measured from satellite observations such as those provided by S5P/TROPOMI, and are often the base of trace gas retrieval schemes.

In optimization and experimental design, mechanistic models are often tested by comparing measured or experimental data with the results of computational methods. These methods are expensive to compute, especially when the parameter space is large or an iterative approach is used. In such cases, the simulation or a part of it, can be replaced by a surrogate model—a computationally efficient approximation of a computationally complex function. For complex but critical applications, the use of a surrogate model speeds up the simulation, which allows for allocating more computational time to exploring larger parts of the parameter space, or improving the accuracy of methods through more iterations. The use of available simulation or observed data to construct machine learning models as surrogates for specific areas of application (Yondo et al. 2018; Verdel et al. 2020; Cai et al. 2021; Servera et al. 2021) or to assist the optimization process in general (Schweidtmann and Mitsos 2019; Lukšič et al. 2019) is becoming more prevalent. The type of machine learning model used as a surrogate can be different and should be selected based on the task. Although the computational cost of training these models can vary, their use for prediction is much cheaper than evaluating the model they are replacing.

In this study, we present our work on implementing a surrogate model that would accurately and efficiently emulate the forward models of atmospheric gas RTMs that are used in current retrieval schemes for trace gas retrievals: the RemoTeC algorithm (currently the operational source of TROPOMI methane concentrations, Hu et al. 2016) and the SCIATRAN RTM (Rozanov et al. 2014). In addition to forward models that predict the output of the simulation from its inputs, we also train reverse models that predict simulation parameters, based on the outputs. Reverse models can be considered as surrogates for the task of estimating the parameters of a complex system, given real measurements or a simulation.

Learning surrogate models of a RTM presents a unique challenge, since the RTMs feature a large number of dimensions in both the input and the output spaces, which is further compounded by the requirement of developing both forward and backward models. Many machine learning algorithms have trouble with predictions in settings like this due to the curse of dimensionality. In the earth observation and emulation literature there is no significant body of research on the application of dimensionality reduction methods.

We employ methods to estimate the intrinsic dimensionality of the input and output spaces and embed them in lower dimensional subspaces to facilitate the learning task. The estimation of the dimensionality of the underlying manifold on which the data lies is an important step that guides the process of data embedding and can inform its

appropriateness. Similar to the choice of a machine learning method to be used to learn a surrogate model, the dimensionality reduction method should be selected based on the task at hand.

Furthermore, we investigate the effect of the sampling procedure and the preparation of the dataset on the learning task and the learning of different types of models as surrogates. To this end, we study two datasets with different properties, generated by similar algorithms. One dataset is generated by using the RemoTeC algorithm and LINTRAN RTM, focused on a narrower parameter range of expected atmospheric properties that we have introduced in our previous work (Brence et al. 2020). The other dataset is generated by using the SCIATRAN RTM with broader applicability, aimed at novel, exploratory applications.

The paper is organized as follows. We give an overview of existing surrogate modeling approaches in Sect. 2. In Sect. 3, we first present the radiative transfer models and the procedure for generating the two datasets. Next, we explore the properties of the data and the results of our dimensional analysis in Sect. 4. Then, in Sect. 5, we introduce the structure of our framework for learning surrogate models and give a brief overview of the machine learning methods employed. In Sect. 6 we present the details of our experiments and their results. Finally, in Sect. 7, we discuss the findings of the study and their implications, as well as our ideas for further work.

2 Related work

Recently, there has been considerable interest in exploring machine learning methods to emulate the simulation of complex physical models by learning surrogate models, both in general, and in the particular context of Earth sciences and Earth observation. Within data processing schemes for trace gas retrievals from satellites such as TROPOMI, RTMs are often one of the largest bottlenecks of the retrieval, due to long computational time and large memory requirements. To speed up retrievals, Look-Up Tables (LUTs) can be used. However, to achieve the accuracy required for trace gases, large LUTs have to be populated and many RTM simulations are needed, which can take a long time to compute. Surrogate models of RTMs are therefore gaining interest due to their ability to speed up simulations, either within data processing schemes that use optimal estimation methods, similar to those used for trace gas retrievals in TROPOMI, or for LUT generation. Examples of work on learning surrogates of RTMs come from many different domains of Earth observation, with a variety of methods for machine learning and data pre-processing. Some of the earlier work by Gómez-Dans et al. (2016) focused on the emulation of a leaf/canopy vegetation RTM PROSAIL, and the atmospheric RTM MODTRAN. Gaussian process regression was used to emulate both models and latin hypercube sampling was used as an efficient sampling technique to account for denser sampling in regions of higher sensitivity. Such work has also been considered by Verrelst et al. (2015) and has been extended by Verrelst et al. (2016) and Verrelst et al. (2017) towards the analysis and comparison of various machine learning techniques to emulate vegetation RTMs, including neural networks, random forests, kernel ridge regression and Gaussian process regression. These emulators were used to generate fast synthetic reflectance spectra from the SCOPE RTM for the FLEX (The FLuorescence EXplorer) Mission (Verrelst et al. 2017) or to perform fast sensitivity analysis of leaf, canopy and atmospheric RTMs and identify the key input variables driving the spectral output (Verrelst et al. 2016).

Emulators are also gaining interest for atmospheric correction of optical imagery, typically requiring LUT operations to correct top-of-atmosphere radiances for atmospheric effects in surface reflectance retrieval. Machine learning algorithms such as Gaussian process regression, random forests, kernel ridge regression (e.g. Vicent et al. (2018); Servera et al. (2021)) and neural networks (e.g. Brodrick et al. (2021)) have been used to develop emulators of the MODTRAN RTM and replace LUTs for the atmospheric correction of optical (multi- and hyperspectral) imagery. In related work, emulators are also gaining interest in atmospheric science domains. For example, Himes et al. (2020) have used a neural network surrogate model of an atmospheric RTM for exoplanet atmosphere simulations within a Bayesian framework, Pal et al. (2019) used a deep neural network to emulate both shortwave and longwave RTM simulations within the Super-Parameterized Energy Exascale Earth System Model (SP-E3SM), and Liu et al. (2020) emulated the Rapid Radiative Transfer Model for General circulation models (RRTMG), often used in climate modelling, using a neural network.

Recent examples of the use of surrogate models in the trace gas domain include the Full-Physics Inverse Learning Machine (FP_ILM), which has been applied to several atmospheric retrieval problems (Xu et al. 2017; Efremenko et al. 2017; Hedelt et al. 2019; Loyola et al. 2020), including ozone and SO_2 retrievals. FP_ILM is an advanced algorithm that replaces the costly RTMs with a surrogate and is unique in the field of trace gas retrievals. There are several core aspects which are similar between FP_ILM and this study, for example the use of sampling methods to generate training datasets, and the fundamental goal of replacing the radiative transfer model. However the FP_ILM differs from the work in this study in several key aspects. The importance of dimensionality is discussed in developing FP_ILM, but is not investigated in depth, since FP_ILM focuses on narrow spectral windows where dimensionality is less important. In this study we investigate a relatively wide spectral window in the TROPOMI instrument, where the effects of high dimensionality are more significant. FP_ILM is based on classification, rather than the true atmospheric inversion method that this study is aimed at. Most importantly, since FP_ILM focuses on different trace gases and different spectral windows, it can not be directly compared with the work shown in this study. To the best of our knowledge, surrogate models still haven't been applied to the spectral region considered in this work.

There are many other examples of machine learning in atmospheric retrievals, but these tend to focus on corrections (Qiu et al. 2021) or aerosols (Song et al. 2021). Surrogate model techniques have yet to be applied on a wider scale in atmospheric retrievals. Furthermore, a large amount of work on learning surrogate RTMs focuses on using a single machine learning method (neural networks) to learn the surrogate from a single RTM, without exploring different methods for preparing datasets and without studying the importance of the dimensional properties of the data in detail.

3 Radiative transfer models for sentinel 5P TROPOMI simulations

3.1 Copernicus sentinel-5 precursor (S5P)

The S5P satellite (https://sentinel.esa.int/web/sentinel/missions/sentinel-5p/satellite-description) was launched in October 2017 with the aim to provide global information on air quality and greenhouse gases (Veefkind et al. 2012). S5P is a joint venture between the

European Space Agency (ESA) and the Netherlands, and is the first of several planned missions for air quality monitoring in the ESA/European Commission Copernicus program.

S5P was launched into low Earth orbit with a mid-afternoon afternoon orbit (13:30 crossing time), thus providing global daily measurements while maximising the signal-to-noise ratio for retrievals and providing synergy with morning orbit satellites. Onboard S5P is the TROPOMI instrument, which is an imaging spectrometer with a swath width of roughly 2600 km on the ground, providing data in 8 separate wavebands of which the short-wave-infrared (SWIR) is the subject of this work. For each band, the swath width is split up into 'cross track' pixels, which form the individual instrument measurements of size indicated by the spatial resolution in Table 1. The spectral response for each band is characterised by the number of spectral pixels, and the instrument spectral response function (ILSF; http://www.tropomi.eu/data-products/isrf-dataset). The SWIR band (the combination of bands 7 and 8, typically known as SWIR3) is focused on providing data on atmospheric concentrations of methane, carbon monoxide and water vapour, all of which are important in the context of a changing global climate (IPCC 2014). The specific instrument characteristics of TROPOMI are identified in Table 1.

3.2 The RemoTeC algorithm and the LINTRAN radiative transfer model

A "retrieval algorithm" is used to convert the top-of-atmosphere radiance spectra captured by TROPOMI (known as Level 1 data) into trace gas concentrations (known as Level 2 data). An example of such an algorithm is the RemoTeC algorithm (Butzet al. 2012; Hu et al. 2016). RemoTeC simulates a realistic approximation of the instrument response in the S5P SWIR band, given a wide range of atmospheric parameters, including scattering by aerosols and variations in surface albedo and solar zenith angle. RemoTeC is the current operational method for methane retrievals from S5P/TROPOMI.

The RemoTeC algorithm is split into two components, operational and synthetic, where the operational aspect deals with the active retrievals of methane from TROPOMI (Hu et al. 2018) and is not the subject of this work. The core of the RemoTeC algorithm is the LINTRAN RTM (Hasekamp and Landgraf 2002), which represents the synthetic component of the algorithm and was designed to test the RemoTeC algorithm prior to launching S5P. The LINTRAN RTM calculates synthetic spectra based on a set of input atmospheric, spectroscopic, surface and instrument properties/assumptions. These scenarios form the basis of the first dataset used in this study, and are described in more detail below. The

Table 1 Characteristics of S5P/TROPOMI bands

Band	Spectral range (nm)	Spectral resolution (nm)	Spatial resolution (km^2)
Band 1	267–300	0.45–0.5	28×28.8
Band 2	300–332	0.45–0.5	5.6×3.6
Band 3	305–400	0.45–0.65	5.6×3.6
Band 4	400–499	0.45–0.65	5.6×3.6
Band 5	661–725	0.34–0.35	5.6×3.6
Band 6	725–786	0.34–0.35	5.6×3.6
Band 7	2300–2343	0.227	5.6×7.2
Band 8	2343–2389	0.225	5.6×7.2

whole retrieval process is computationally intensive, and the speed up of these algorithms is the subject of much work in the atmospheric communities.

The TROPOMI ILSF is applied to the synthetic spectra from RemoTeC in order to replicate the TROPOMI instrument characteristics, however, other unavoidable instrument issues such as calibration errors, and heterogeneous cross track pixel responses (i.e. the spectral smile) are not considered in this study. We expect that operational deployment of trained emulators will have to consider these effects, indeed it may be necessary for each cross-track pixel for each band to have a unique emulator attached to it.

3.2.1 Inputs: atmospheric parameters

The training dataset is generated using the RemoTeC tool, provided by the Dutch Space Research Organisation (SRON). The atmospheric parameters input into RemoTeC are designed to cover the range of atmospheric conditions that S5P/TROPOMI is expected to encounter, in order to develop a surrogate model capable of approximating both realistic atmospheric conditions and S5P/TROPOMI measurements. The input state vectors are generated from a combination of chemistry transport models. Table 2 outlines each of the atmospheric parameter inputs, their possible associated values and the source they come from (they come from chemistry transport models, meta-data, or are explicitly defined).

Table 2 Input atmospheric parameters for the RemoTeC algorithm, value distribution and source of information

Parameter	Variation/distribution	Number of vector elements	Source
Solar Zenith Angle (SZA)	0–70°	1	
Viewing Zenith Angle (VZA)	0°	1	
Viewing Azimuth Angle (VAA)	0°	1	
Albedo	0.01, 0.1, 0.3, 0.5, 0.8	1	ADAM DB (Muller et al. 2013)
CH_4 profile	Arctic, mid-latitude & tropical cond.	20	TM5 Model (Hu et al. 2016)
CO profile	Arctic, mid-latitude & tropical cond.	20	TM5 Model (Hu et al. 2016)
H_2O profile	Arctic, mid-latitude & tropical cond.	20	ECMWF (Hu et al. 2016)
Aerosols	Arctic, mid-latitude & tropical cond.	1	ECHAM-HAM
	Aerosol optical depth		
	Between 0 and 0.5		
Temperature	Arctic, mid-latitude & tropical cond.	20	ECMWF (Hu et al. 2016)
Pressure (hPa)	Arctic, mid-latitude & tropical cond.	20	ECMWF (Hu et al. 2016)
Altitude (km)	Arctic, mid-latitude & topical cond.	20	ECMWF (Hu et al. 2016)

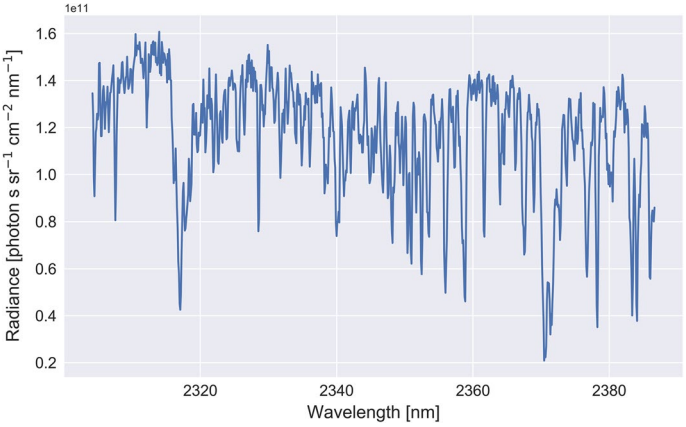

Fig. 1 An example SWIR3 spectrum from RemoTeC, assuming a realistic atmospheric profile and TROPOMI instrument characteristics

Table 3 Summary of the two datasets prepared and studied in this work

RTM	Samples	Inputs	Outputs	Sampling
LINTRAN	50000	125	834	Focused
SCIATRAN	1000	249	8001	LHS

A combination of values, one for each parameter shown in Table 2, is fed into RemoTeC-LINTRAN as a state vector.

In total, we generated a dataset of 50,000 input state vectors, which comprise the input part of the LINTRAN dataset. This dataset comprises 10,000 individual measurement points, split into a global dataset of $3 \times 3°$ bins, representing one day from each season, i.e., January, April, July and October. The dataset is generated by varying the surface albedo conditions between 0.01 and 0.8 (as shown in Table 2) for each of the 10,000 simulated measurement points. Land conditions are considered only, and no sea environments are included in the dataset. Therefore, apart from the albedo data, we do not specifically set the other conditions since these are assigned by the natural global variations, but the values described in Table 2 indicate the ranges included in the dataset and not specific steps. Both VZA and VAA are included in the state vector, but set to 0 as is the case in the original dataset, assuming a nadir pointing instrument. We assume future training datasets will use these parameters, but for the purposes of this study they are left out of the training.

3.2.2 Outputs: S5P/TROPOMI L1b synthetic spectra

Given each state vector, synthetic Level 1 radiance (defined as the radiant flux emitted, reflected, transmitted or received by a given surface, per unit angle per unit projected area) spectra are simulated using LINTRAN in the S5P/TROPOMI SWIR3 band. An example SWIR3 band spectrum produced by RemoTec is given in Fig. 1. In total, 50,000 synthetic spectra were generated, one for each given state vector of atmospheric parameters values, as described in Table 3. These comprise the output part of the LINTRAN dataset.

3.3 SCIATRAN radiative tansfer model

In our preliminary study (Brence et al. 2020), we discovered peculiar dimensional properties of the LINTRAN dataset that indicated the possibility of an artifact, related to the sampling of certain parameters (as detailed in Section 4). Consequently, we constructed a new dataset to help us explore the importance of sampling the parameter space for the success in addressing the surrogate learning task. Given that the LINTRAN dataset is pre-designed to cover only the range of atmospheric conditions that S5P/TROPOMI is expected to encounter, it does not allow for different input parameter sampling, which can greatly impact the generation of surrogate models. As a result, in this study, we consider a second RTM that is open source and with which we can more flexibly define an exact parameter space.

3.3.1 The SCIATRAN RTM

SCIATRAN (Rozanov et al. 2014) is an open source RTM, available at https://www.iup.uni-bremen.de/sciatran/, originally designed for the visible and SWIR bands on the SCanning Imaging Absorption SpectroMeter for Atmospheric CHartographY (SCIAMACHY) instrument on ENVISAT (Bovensmann et al. 1999). It has since been expanded to include more spectral bands, additional geometry and scattering physics. Unlike LINTRAN, SCIATRAN is purely an RTM and is not linked to a retrieval algorithm, although it can be, and has indeed been used as the core of retrieval algorithms.

Both LINTRAN and SCIATRAN can be used to represent the same physical processes (i.e., the transfer of radiation at a specific wavelength through an atmospheric medium), and in theory should output identical results given the same inputs. In practice, the two RTMs use different inputs, such as atmospheric models identifying trace gas concentrations in the atmosphere, spectroscopic databases and physical processes, such as scattering, meaning that the outputs will never match. We do not claim in this paper that either RTM is more accurate than the other, but SCIATRAN does have the advantage of a simpler interface and more flexibility of inputs, while LINTRAN is largely tied to the 10,000 atmospheric scenarios described in the previous section. SCIATRAN can be modified to represent most conceivable atmospheric scenarios, meaning more flexibility is available in generating the training dataset for learning surrogate models. In this paper, we interface with SCIATRAN through the python package pyatran (Hilboll et al. 2018), which allows for a quick interface to the RTM through a JSON format. As with RemoTeC/LINTRAN, all synthetic spectra are convolved with the appropriate TROPOMI ILSF to replicate instrument response.

3.3.2 Inputs: latin hypercube sampling for atmospheric parameters

Given the flexibility of the input of the SCIATRAN model and the common assumption that input parameters are uniformly distributed between maximum and minimum boundaries, we devised a sampling scheme to both cover the input space more completely, as well as reduce the number of SCIATRAN runs. Latin Hypercube Sampling (LHS) was chosen (Sacks et al. 1989), since it provides a way of generating random samples of parameter values in a multi-dimensional space uniformly. LHS is a popular sampling approach, used widely in simulation experiment design, uncertainty analysis, reliability analysis and adaptive meta-modelling, such as the approach proposed in this paper. The LHS method is

based on a Latin square design, which has a single sample in each row and column. Adding parameters, or rows and columns, builds up the"Hypercube", extending the Latin square to multi-dimensional parameter space.

In addition, since LHS can use different probability density functions (PDFs), the sampling scheme can account for denser sampling in regions where the model is expected to be more sensitive. Accordingly, PDFs were generated from the previous dataset, each with 50,000 samples, and used to distribute the values of the LHS from the original PDF of the input atmospheric parameters in order to account for higher sensitivity of the model to certain parameters. The LHS was run to generate 1000 input-output pairs, aiming to efficiently cover the multi-dimensional space of the original 50,000 dataset.

4 Properties of the datasets and dimensional analysis

The input (parameter) space of both datasets (Table 3) consists of atmospheric parameters, representing an atmospheric state accurately. The difference in the input for the two datasets are the different altitude levels at which the atmospheric pressure, air temperature values and concentrations of water vapour, methane and carbon dioxide, are sampled. In the LINTRAN dataset, 20 altitude levels are chosen, such that the atmosphere is sampled closer to the surface rather than near the top. In the SCIATRAN dataset, 49 levels are sampled from the surface to the stratosphere, with a greater proportion of samples weighted to the surface, providing a more detailed state.

In the input space, the atmospheric state parameters have magnitudes on different scales. In the LINTRAN dataset, the distributions of the parameters were either bimodal or heavy-tailed unimodal, with high density around the modes. Due to LHS, in the SCIATRAN dataset the distribution of the parameters is uniform. For both datasets, we normalized the atmospheric parameters to a standard distribution by applying the transformation $\frac{x-\mu}{\sigma}$. This transformation removes the effect of scale on the predictive error of the surrogate models and reduces the bias of the model to the mean value of the parameters in the case of the LINTRAN dataset. In the output space, the radiance for all wavelengths is on a comparable scale. However, the distribution of the radiance per wavelength is predominantly exponential. We first divided all values by 10^{11} and 10^{17} for the two datasets, respectively, in order to make the range comparable and reduce potential error in computation due to extremely large values. We then applied a logarithmic transformation followed by subtracting the mean, such that the resulting distributions resemble a normal distribution with mean 0, therefore also reducing the bias of the model to the mean value of the radiance for each wavelength.

The effects of the different sampling, one focused around expected parameter ranges, and the other obtained by LHS can be seen in Fig. 2 (top). The plots show the distribution of distances to selected k-th nearest neighbors of each output. We calculated the coefficients of variation ($c_v = \frac{\sigma}{\mu}$) of the distributions of the distances to the first k nearest neighbors (with k ranging up to 100). For the LINTRAN dataset, these range from 0.12 to 0.76, indicating that the input space is not well covered and is locally denser. For the SCIATRAN dataset, the coefficients are low and range from 0.03 to 0.05. In the output (spectra) space of both datasets the SWIR3 band of wavelengths is equally well covered as shown in the frequency polygon in Fig. 2 bottom. For the SCIATRAN dataset, the resolution of the obtained output is higher (0.01 nm) as compared to the LINTRAN dataset (0.1 nm).

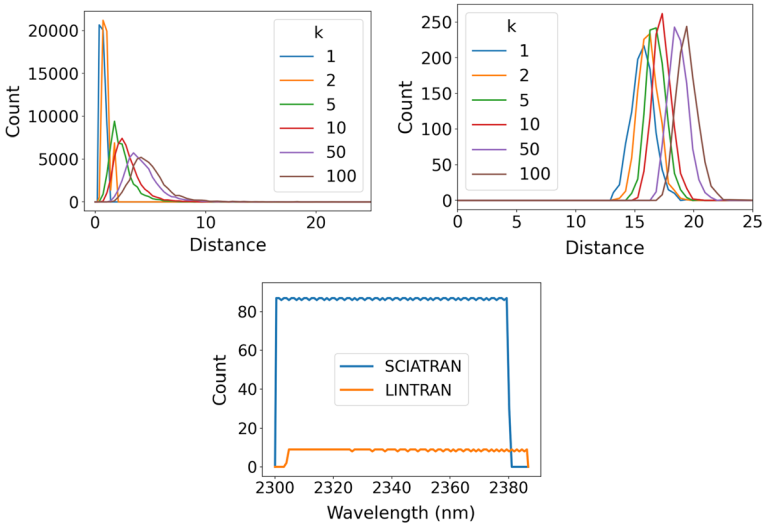

Fig. 2 Distribution of the distances to the k-th nearest neighbor of each point ($k = \{1, 2, 5, 10, 50, 100\}$, $c_v(k) = \{0.12, 0.12, 0.60, 0.62, 0.76\}$) for the LINTRAN (top-left) and the SCIATRAN (top-right) dataset $c_v(k) = \{0.03, 0.03, 0.03, 0.03, 0.05\}$. Frequency polygon (bottom) of the wavelengths representing the output space for the two datasets (bin width 0.87 nm)

The task of learning a surrogate model for this problem requires learning a function that maps from a high-dimensional input space to a high-dimensional output space. We therefore attempt to lower the dimensionality of both spaces, in order to reduce the computational complexity and estimate the information content available for the task.

We estimate the intrinsic dimensionality of both the preprocessed input and the original output spaces of both datasets using the TWO-NN, a nearest-neighbor based intrinsic dimensionality estimation method (Facco et al. 2017). The method is based on previous work on maximum likelihood estimation from a distribution of neighborhoods of data points (Levina and Bickel 2005) that are assumed to be drawn from relatively small hyperspheres that lie in lower dimension than the original data. TWO-NN considers only minimal neighborhood information to estimate the dimensionality of these hyperspheres, i.e., the two nearest neighbors of each point. TWO-NN has been shown to perform well on datasets with non uniform densities and curvature and for the successful estimation of the intrinsic dimensionality of data for the task of unsupervised agnostic feature selection (Doquet and Sebag 2020).

Following the estimation of the intrinsic dimensionality of the data, we can estimate the linear dependencies of the features, by looking at how the cumulative relative variance depends on the number of principal components by performing principal component analysis—PCA (Pearson 1901). This simple transformation can not only be used for dimensionality reduction by truncating the number of principal components, but, combined with the estimated intrinsic dimensionality of the data, can also give us insight into the need for considering non-linear dimensionality reduction. If we are able to capture a significant amount of variance of the data by considering a number of principal components close to the estimated intrinsic dimensionality of the data, then there is no need to consider more advanced or computationally more complex dimensionality reduction methods. As we show in the Results section, the embedding of the input and output spaces using PCA

Table 4 Estimated intrisic dimensionality (ID), the corresponding cumulative variance explained (σ^2) by using the first ID principal components, and percentage of the original dimensionality that it represents

	LINTRAN		SCIATRAN	
	Input	Output	Input	Output
Original dim.	125	834	249	8001
Intrinsic dim. (ID)	2	5	33	9
σ^2(%)	52	99	83	98
ID as % of original dim.	1.6	0.6	13	0.1

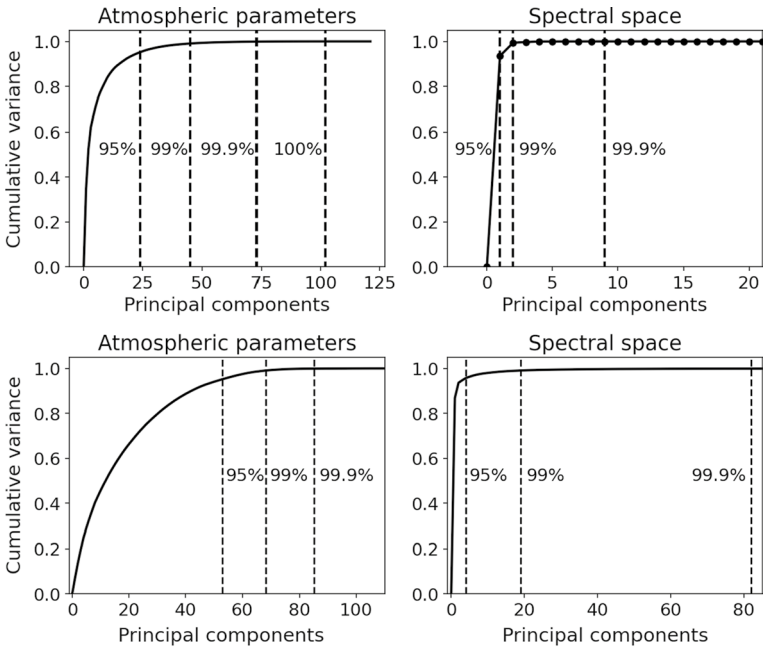

Fig. 3 Dependence of the cumulative relative variance explained on the number of principal components for both the input and the output space for the LINTRAN (top) and the SCIATRAN (bottom) datasets

is sufficient to significantly reduce the computational complexity while achieving performance on par with non-linear embeddings such as those obtained with autoencoders.

In Table 4, we give the estimated intrinsic dimensionality of the input and output space of both datasets and the corresponding cumulative variance explained when considering the principal components up to the estimated intrinsic dimensionality (ID). In Fig. 3, we show the curves of the cumulative variance explained as a function of the number of principal components.

The low estimated intrinsic dimensionality of the input spaces, in both cases, points to overdetermination of the radiative transfer models with regard to the input parameters. The low value of the estimated intrinsic dimensionality for the input space of the LINTRAN dataset points to the properties of the sampled parameters, based on the expectation of atmospheric properties being locally dense. The relatively lower cumulative variance explained at the number of principal components equal to the estimated intrinsic dimensionality points towards the need for non-linear embeddings of the input space for more

efficient representation. Alternatively, one could allow for a higher dimensional linear embedding, which can be afforded, while still resulting in a significant decrease of computational complexity.

The output space has very low intrinsic dimensionality, due to the nature of the measurements. Although the resolution of the output space in the SCIATRAN dataset is higher (larger original dimensionality) the estimated intrinsic dimensionality is comparable to the output space of the LINTRAN dataset. Furthermore, the PCA analysis shows that linear embedding of both output spaces is sufficient.

The properties of the datasets are related to how demanding the tasks of learning forward and backward surrogate models are. Learning a model mapping from a more informative (higher dimensional) space to a lower dimensional space is a less demanding task. In the case of the LINTRAN dataset, due to the properties of the sampling of the parameters, it is reasonable to expect that a backward model can be learned directly from the available data allowing us to bypass the parameter optimization task using a forward model. This is not the case for the more general SCIATRAN dataset, for which the task of learning a backward surrogate model might not be tractable, given the large differences of intrinsic dimensionality between the input and output spaces.

5 Surrogate model learning

Surrogate models are commonly used to replace computationally expensive simulations of complex models. However, constructing surrogate models and making predictions can still be computationally complex. This is the case when the dimensionality of the input or the output space is large, increasing the complexity of model construction, as well as the computation time required to make predictions. One way to address this issue is through the use of methods for dimensionality reduction.

In our framework, depicted in Fig. 4, preprocessing and dimensionality reduction are performed both on the input and the output spaces, i.e. on atmospheric parameters and

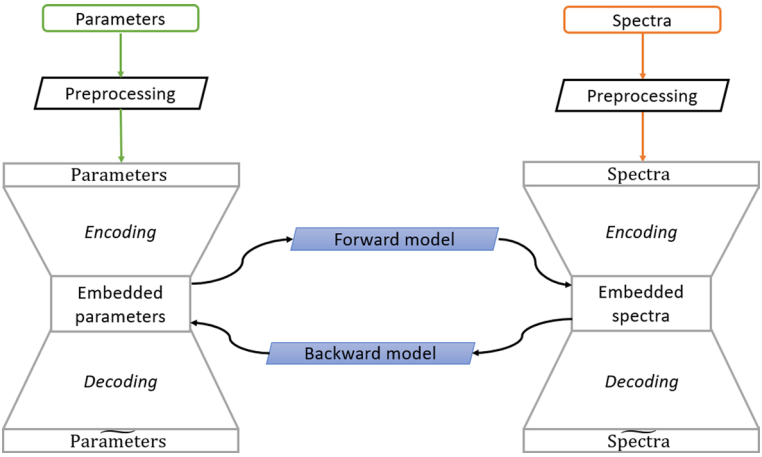

Fig. 4 The architecture of our framework for learning surrogate models. Encoding transforms the parameter/spectra space into lower-dimensional embedded spaces, while decoding transforms the embedded representations back to the original spaces

spectra, separately. The forward model takes an embedded representation of the parameters as input and predicts the representation of spectra as output. The predictions in the embedded output space are then inversely transformed to obtain predictions in the original space of spectra. For the backward model, the roles of input and output are reversed, with the representation of spectra acting as input and the representation of atmospheric parameters acting as output to be predicted. Our framework offers a choice of linear regression, random forests (RF) (Liaw et al. 2002), forests of extremely randomized trees (ET—extra trees) (Geurts et al. 2006) and a feedforward neural network (NN) as models for prediction. We later added also Gaussian regression (GR) and kernel ridge regression (KR). The performance of the different predictive modelling methods is evaluated in the following section.

We studied and compared two methods of dimensionality reduction within our framework: principal component analysis (PCA) and autoencoders. PCA is a popular linear method that is easy to implement and cheap to compute. Autoencoders feature a number of hyperparameters and are computationally more expensive, but have the potential to find better embeddings than PCA, due to the nonlinear transformations they make.

Autoencoders are a type of neural networks used to learn low dimensional embeddings. The autoencoder network is trained to reproduce the input data on the output layer, with the defining characteristic that the network architecture features a bottleneck - the embedding layer. A concern when designing autoencoders is that a network with sufficient capacity would memorize the entire dataset and simply learn the identity function to satisfy the reproduction loss function. To combat this issue and improve the ability of an autoencoder to capture important information and learn richer representations, different methods of regularization are employed. Some common methods include batch normalization, adding a sparsity term to the loss function, and adding noise to the input.

We performed dimensionality reduction using a denoising autoencoder (Goodfellow et al. 2016). Adding some amount of Gaussian noise to the input data forces the autoencoder to learn meaningful features. We treated the variance of the noise as a dimensionality reduction hyperparameter and tested several options. Note that the noise is only added during training.

We experimented with a number of autoencoder architectures and settled on a model with a total of 7 layers, including the input and output. The structure of the autoencoder can be summarized as follows:

1. Input layer of size N_0 + Gaussian noise layer,
2. Fully connected layer of size $N_1 < N_0$ and activation ReLu,
3. Fully connected layer of size $N_2 = \frac{1}{2}N_1$ and activation ReLu,
4. Fully connected (embedding) layer of size $N_3 < N_2$ and linear activation,
5. Fully connected layer of size N_2 and activation ReLu,
6. Fully connected layer of size N_1 and activation ReLu,
7. Output layer of size N_0 and linear activation.

For the parameter space, $N_0 = 123$ in the LINTRAN dataset. For the spectral space, the architecture is the same, with the input layer having $N_0 = 834$ dimensions for the LINTRAN dataset. The models were trained using the Adam optimizer (Kingma and Ba 2014) and a MSE loss function. The models were implemented with Tensorflow 2.0 (Abadi et al. 2016).

6 Experiments and results

We performed several experiments to test the different methods for dimensionality reduction and prediction, as well as to evaluate the performance of our framework for learning surrogates of radiative transfer models. In the first part, we present the experiments performed on the LINTRAN dataset. These consist of the exploration of autoencoders for dimensionality reduction, as reported on in our previous work (Brence et al. 2020) as well as new experiments, comparing the performance of different predictive modelling methods. In the second part, we report on the experiments concerned with modeling the SCIATRAN dataset and compare the performance of the models built by using datasets with different properties. The configurations and hyperparameter values of the tested machine learning algorithms are summarized in Table 5. To evaluate and compare models we compute the spectral normalized root-mean-square-error NRMSE(%) as proposed by Servera et al. (2021):

$$\mathrm{NRMSE}_k = 100 \frac{\sqrt{\frac{1}{N}\sum_i^N (\hat{y}_{ik} - y_{ik})^2}}{\max_i(y_{ik}) - \min_i(y_{ik})},$$

where y_{ik} indicates the simulated radiance of the i-th sample and k-th wavelength, and $\hat{y}_{ik}$ the corresponding model prediction. As an overall measure of model performance, we compute the NRMSE of a model as the mean of NRMSE_k across all wavelengths. The choice of NRMSE over RMSE is motivated by easier comparison to error and uncertainty requirements for satellite missions (at the L1b radiance level, these will be things like signal-to-noise, noise equivalent delta level) and can also be easier propagated to L2 and compared against trace gas uncertainty requirements. ATBD for the TROPOMI methane product (Hasekamp et al. 2021) indicates that errors in the L1b radiance due to radiometric offset should be less than 2%. This value indicates a suitable target against which we can assess how well the model predictions in this study perform.

6.1 Experiments on the LINTRAN dataset

We tested and evaluated two methods for dimensionality reduction to identify the best choice for our task. We compared the properties of the different dimensionality reduction methods with regards to how well they are able to reconstruct the original data. We then implemented predictive models that map between the reduced input and output spaces and compared their predictive performance. The workflow can be summarized as follows:

1. Randomly split the data into a training set (80%), validation set (10%) and test set (10%).
2. Preprocess the features and targets.
3. Compute a dimensionality reduction projection on the feature space of the training set, then transform all three sets using the learned projection. Evaluate by using the error of reconstruction on the validation set as a measure of quality. Repeat for the target space.
4. Train a predictive model (e.g. neural network) to predict from the reduced feature space to the reduced target space. Optimize the hyperparameters of the model by using the validation set. Similarly for the backward model.
5. Evaluate the forward and the backward models on the test set.

Table 5 Hyperparameters of models used in the experiments in Sect. 6

	LINTRAN	SCIATRAN
Neural networks (NN)	2 layers; 100 nodes per layer; RELU activation; L2 regularization parameter 0.001; optimizer ADAM; learning rate 0.0005; 100 epochs; Tensorflow implementation.	4 layers; 290 nodes per layer; ELU activation; L2 regularization parameter 0.001; the other settings identical to those for LINTRAN-NN. 144 input principal components; 297 output principal components.
Random forests (RF)	200 estimators; bootstrapping; at least 2 samples per split; maximal features ratio 1/3; no max depth; Scikit-learn implementation.	At least 5 samples per split; no bootstrapping; the other settings identical to LINTRAN-RF. 62 input principal components; 124 output principal components.
Extremely randomized trees (ET)	Identical to the settings for LINTRAN-RF.	At least 7 samples per split; maximal depth 10; the other settings identical to LINTRAN-RF. 62 input principal components; 280 output principal components.
Gaussian process regression (GP)	–	Radial basis function kernel, LBFGS optimizer.
Kernel ridge regression (KR)	–	Radial basis function kernel, regularization parameter 0.5

Table 6 The reconstruction NRMSE of an autoencoder with three different input noise levels (σ) as compared to the NRMSE of PCA, when reducing the dimensionality of the data and then reconstructing the input. For the atmospheric parameter space (X), we reduce the dimensions from 123 to 45. For the spectral space (Y), the dimensions are reduced from 834 to 2

σ	Autoencoder			PCA
	0.0	0.1	0.5	
X	**2.2%**	2.7%	5.8%	2.4%
Y	**1.6%**	1.7%	1.7%	1.9%

The parameter configurations with the lowest error are emphasized in bold

Table 7 Comparison of the NRMSE of predictive models (neural networks) with different reduced input and output spaces. The reduced dimensions are 45 for the atmospheric parameter space and 2 for the spectral space, in all four cases

σ	Autoencoder			PCA
	0.0	0.1	0.5	
Forward	3.2%	**1.7%**	2.0%	1.7%
Backward	16.6%	16.6%	17.2%	**16.5%**

The parameter configurations with the lowest error are emphasized in bold

We compared the different methods for dimensionality reduction on the atmospheric parameter space and the spectral space separately. We reduced the dimensionality of the parameter space from 123 down to 45 dimensions and the dimensionality of the spectral space from 834 to 2 dimensions. The lowest numbers of dimensions, 45 and 2, were chosen such that they correspond to the numbers of principal components in the parameter and the spectral space, respectively, that cumulatively explain 99% of the variance.

The NRMSE values for the reconstruction are shown in Table 6. When training the denoising autoencoder, we added Gaussian noise with mean $\mu = 0$ and different values for the standard deviation σ. The models were trained on the train set and evaluated on the validation set. Unsupervised learning normally does not require a train-test split. Here, we train the autoencoders on the train set only, since the dimensionality reduction models are used to inverse transform the predictions from the embedded space back into the original space. The error of this reconstruction is more objectively estimated by evaluating it using unseen data from the validation set.

We observed that adding noise resulted in a higher reconstruction error on the test set for high noise values. The NRMSE of the autoencoder on the parameter space is lower than the NRMSE of PCA for no added noise, and higher for both nonzero levels of noise. In the spectral space, the autoencoder NRMSE is lower than the NRMSE of PCA for all levels of noise.

Next, we compared the prediction error of the models mapping between the embedded spaces, created by different methods of dimensionality reduction. NRMSE was computed in the original space, after inversely transforming the predictions.

Table 7 shows the performance of the predictive models using features selected by different dimensionality reduction methods. For the forward model, predictions using autoencoder representations work best with a noise level of $\sigma = 0.1$. Regularization by adding

Table 8 Comparison of the NRMSE of predictive models on the LINTRAN dataset for atmospheric parameter and spectral space representations of different dimensions, including the case of not using dimensionality reduction at all. The noise level of the autoencoder was set to $\sigma = 0.1$

Dimensions(X-Y)	Autoencoder	PCA	No DR
Forward model			
45-2	1.6%	1.7%	
73-9	1.4%	0.81%	
102-50	0.69%	**0.60%**	
All			0.73%
Backward model			
Dimensions(X-Y)	Autoencoder	PCA	No DR
45-2	16.6%	16.6%	
73-9	10.2%	7.3%	
102-50	8.7%	**6.1%**	
All			6.9%

The configurations with the lowest error are emphasized in bold

noise improves the performance of prediction. PCA reduction outperforms reduction with autoencoders without noise, as well as with high noise. For the backward model, the performances are similar.

We further compared the effect of input and output space dimensionality on the predictive performance of the learned models. Considering the results of the previous step, we chose $\sigma = 0.1$ for the noise level. We compared the NRMSE for different combinations of embedding dimensionality as $dim(X) - dim(Y)$:

- 45-2, which explains 99% of variance in PCA,
- 73-9, which explains 99.9% of variance in PCA, and
- 102-50, which explains nearly 100% of variance in PCA.

We chose 50 as the largest output dimension, as it resulted in best performance and at the same time still represents a relatively large reduction in dimensionality. In Table 8, we show the results and compare the performance of the predictive models learned by using different levels of reduced dimensionality, as well as the performance of the predictive model learned without any dimensionality reduction. Models using PCA for dimensionality reduction outperform models using autoencoders for dimensionality reduction for all pairs of reduced dimensions, except for the pair 45-2. For both the forward and the backward model, all combinations of dimensionality reduction outperform the baseline predictor. After dimensionality reduction, the learned predictive model can have an error lower than the error of a model constructed on the data with unreduced dimensionality. The predictive performance for the forward model is improved by using an embedding of 102 dimensions in the parameter space and 50 dimensions in the spectral space, as compared to using the full spaces. The results for the backward models are similar, with one important difference: the performance of the backward models is generally lower than the performance of the forward models. While still better than the baseline predictor, the difference is not large. The best performing combination is the same as in the forward model: PCA with 102 dimensions in the parameter space and 50 dimensions in the spectral space. Using autoencoders results in lower performance than using PCA. Models using dimensionality reduction outperform models working in the original space.

The regularization effect of PCA lowers the importance of irrelevant and noisy features and makes the more relevant features directly available to the surrogate models. This effect is evident from the improved performance of the surrogate model trained on truncated PCA reduced data compared to the non-linear dimensionality reduction, and even more, compared to the performance on data without dimensionality reduction.

We further considered the use of several popular methods for predictive modelling within our framework. Based on the results in Table 8, we settled on reducing the dimension of the atmospheric parameter space to 102 and the dimension of the spectral space to 50. In Table 9, we compare the performance of neural networks (NN), random forests (RF), extra trees (ET), as well as linear regression (LR), in combination with either PCA or autoencoders (AE) for dimensionality reduction. We estimate the performance of surrogate models on unknown data through 10-fold cross-validation. In the case of forward models, both random forests and extra trees proved competitive with neural networks, when used in conjunction with PCA. When working on a space, reduced by autoencoders, however, tree-based methods performed remarkably worse. Linear regression achieves a NRMSE of 8.4%, which, while still useful, is well below the results of the more advanced methods. The best performing combination remains a neural network for prediction and PCA for dimensionality reduction. For backward models, the differences in score between PCA and autoencoders are less pronounced, although PCA is still the clear winner. Linear regression performs much worse for predictions in the reverse direction. Extra trees with PCA prove to be the best performing method for backward models, with random forests not far behind. We have also considered Gaussian process regression (GP) and kernel ridge regression (KR) as predictive models. However, we were unable to train the models due to their demands on computer memory—the LINTRAN database is too large for the use of these methods.

In Table 10 we provide the training and prediction times of the evaluated combinations of predictive models and dimensionality reduction. Models using autoencoders are significantly slower to train and make predictions with, compared to other models. Meanwhile, the difference between models using PCA and models without dimensionality reduction is small. The model with the best predictive performance (NN and PCA) takes only minutes to train and a fraction of a millisecond to make predictions for a single sample.

Table 9 NRMSE (estimated by 10-fold cross validation) for the surrogate models obtained by using various combinations of dimensionality reduction methods (DR) and predictive models (PM) on the LINTRAN database. A missing value indicates that a model in the given configuration could not be trained due to excessive demands on memory or computation time

Model	Forward model			Backward model		
	AE	PCA	No DR	AE	PCA	No DR
NN	0.5%	**0.3%**	0.7%	6.5%	4.6%	4.7%
RF	7.5%	1.8%	/	4.5%	4.3%	/
ET	6.4%	1.5%	/	4.1%	**3.6%**	/
LR	9.7%	8.4%	8.4%	12.6%	12.6%	11.8%

The configurations with the lowest error are emphasized in bold

Table 10 Computation time for the surrogate models obtained by using various combinations of dimensionality reduction methods (DR) and predictive models (PM) on the LINTRAN database. A missing value indicates that a model in the given configuration could not be trained due to excessive demands on memory or computation time

Model	AE		PCA		No DR	
Forward model						
	Training	Prediction	Training	Prediction	Training	Prediction
NN	58 min	23 ms	4 min	0.05 ms	11 min	0.09 ms
RF	111 min	5.9 ms	10.8 min	0.1 ms	–	/
ET	89 min	19 ms	7.8 min	3.0 ms	–	/
LR	18 min	0.04 ms	2.4 s	0.01 ms	0.1 s	0.01 ms
Backward model						
	Training	Prediction	Training	Prediction	Training	Prediction
NN	56 min	1 ms	5 min	0.07 ms	12 min	0.1 ms
RF	123 min	2 ms	9 min	3 ms	/	–
ET	104 min	5 ms	7.3 min	4 ms	/	–
LR	19 min	0.03 ms	2.4 s	0.01 ms	2.8 s	0.004 ms

6.2 Experiments on the SCIATRAN dataset

On the LINTRAN dataset, we explored the impact of using autoencoders as a method of dimensionality reduction and compared the performance of predictive models when using autoencoders to the performance when using PCA. In all our experiments, using PCA yielded better results that using autoencoders. In addition, training autoencoders is computationally expensive and involves optimizing many hyperparameters. Computing PCA, on the other hand, is very fast and involves no hyperparameter optimization, aside from choosing the dimensionality of the input and the output spaces. In light of those results, we have limited ourselves to using PCA when working with the SCIATRAN dataset.

Due to the lower number of samples, we evaluated the different predictive models on the new dataset by nested 10-fold cross-validation. The inner cross-validation loop was used to optimize the hyperparameters of the predictive models, as well as the dimensionality of both embeddings. The outer cross-validation loop was used to score the model, i.e. evaluate the overall predictive performance.

Table 11 NRMSE for various predictive models on the SCIATRAN dataset, estimated by using nested 10-fold cross-validation. Dimensionality is reduced by using PCA. A missing value indicates that a model in the given configuration could not be trained due to excessive demands on memory or computation time

Model	PCA	No DR
NN	7.3%	7.5%
RF	17.4%	–
ET	17.1%	–
LR	10.2%	11.6%
GP	23.3%	–
KR	13.5%	–

Table 12 Computation time for the surrogate models obtained by using a given predictive model and either PCA or no dimensionality reduction on the SCIATRAN database. A missing value indicates that a model in the given configuration could not be trained due to excessive demands on memory or computation time

Model	PCA		No DR	
	Training	Prediction	Training	Prediction
NN	15 s	0.7 ms	50 s	0.9 ms
RF	46 s	0.2 ms	–	–
ET	40 s	0.2 ms	–	–
LR	5 ms	0.01 ms	0.8 s	0.06 ms
GP	23 s	0.1 ms	–	–
KR	3 ms	0.02 ms	–	–

Table 11 shows the NRMSE for various predictive modelling algorithms for forward models on the new dataset, using PCA for dimensionality reduction, while Table 12 presents the computation times for the respective models. In contrast to Table 9, where the dimensions of the embeddings were fixed, here we treated and optimized the dimensions as a hyperparameter. The best performing model is again a neural network. However, while the neural network achieved an NRMSE= 0.3% on the LINTRAN dataset, its performance on the SCIATRAN dataset is only NRMSE= 7.3%. Random forest and extra trees both perform significantly worse on the SCIATRAN dataset than on the LINTRAN dataset. Linear regression performs similarly on both datasets, with the neural network still offering a significant advantage over linear regression. The biggest difference between the datasets was observed in the case of backward models. No predictive modelling approach was able to produce backward models achieving meaningful performance on the SCIATRAN dataset: all NRMSE values are above 100%. We conjecture that this is related to the differences in the intrinsic dimensionality of the datasets, as discussed in Sect. 4.

To facilitate the evaluation of the developed surrogate models we depict the mean and important quartiles (75, 95 and 97.5%) of NRMSE across all the samples in the test set for

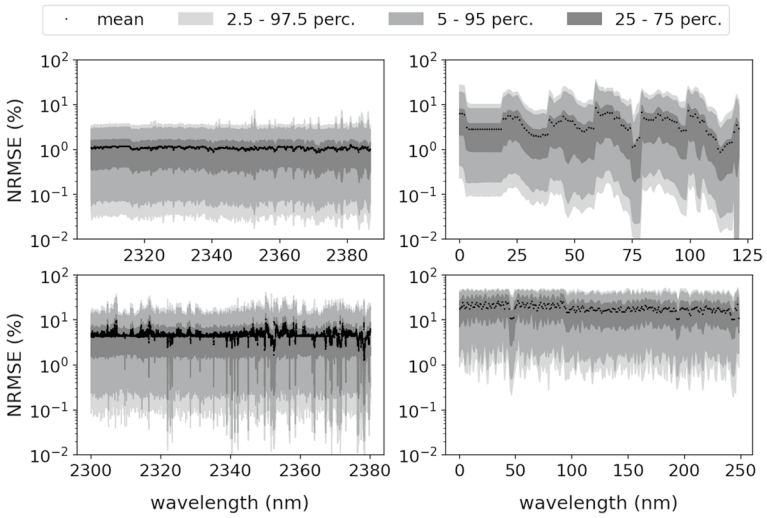

Fig. 5 Important percentiles of the best performing forward (left side) and backward (right side) models, using the LINTRAN (first row) and SCIATRAN (second row) databases

the best forward and backward model for each database (NN for forward LINTRAN and SCIATRAN models, as well as the backward SCIATRAN model, and ET for the LINTRAN backward model) in Fig. 5. In the case of the LINTRAN model, we can see the mean NRMSE is close to zero across all wavelengths, conforming to the 2% error requirement for L1B TROPOMI radiance spectra Hasekamp et al. (2021), and bounded to around + 5% and − 10% for the 5-95th quartiles. In the case of SCIATRAN, mean NRMSE are higher for all wavelengths in comparison, however still bounded to approximately +/− 10% for the 5-95th quartiles. This difference can be attributed to the difference in dataset size. Some outliers exist for both models in distinct spectral bands, particularly towards the latter regions of the SWIR wavelengths. NRMSE is also relatively stable across atmospheric parameters, and on average higher in the SCIATRAN model.

We studied in detail the dependence of predictive performance on the dimensions of the embedded spaces. Since this analysis is computationally intensive, we benefit from the smaller size and better sampling properties of the SCIATRAN dataset. Fig. 6 shows the dependence of the performance of surrogate models on the dimensions of both embedding spaces for models based on linear regression or neural networks. For this comparison we use the coefficient of determination

$$R^2 = 1 - \frac{\text{MSE}}{\sigma^2},$$

where σ indicates the variance of the data and MSE the mean-square-error of the model. A higher R^2 indicates a better model, with the highest possible value 1. For both types of predictive models we observe a sharp increase in performance when a threshold in the

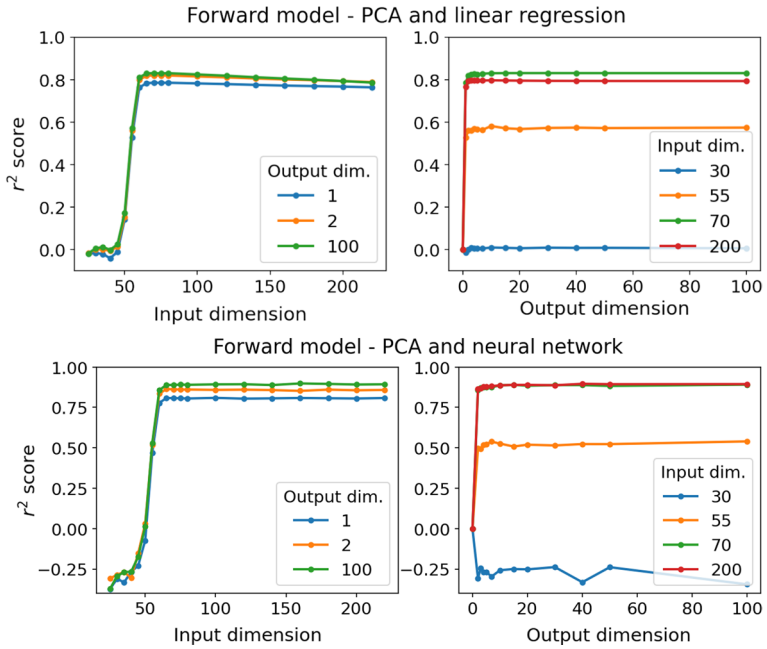

Fig. 6 Dependence of the predictive performance learned by linear regression and neural networks forward models on the number of principal components included for both the input and the output space

dimensionality of the atmospheric parameters space is crossed. At 50 principal components, the R^2 score is close to zero—the models are incapable of providing useful predictions. From 50 to 60 principal components, the performance rapidly improves and saturates. Further increase in dimensionality brings only marginal improvements in the case of neural networks, and degrades the performance in the case of linear regression. The critical range in dimensionality of the embedded atmospheric parameter space corresponds to the range between 94% and 97% variance explained with PCA. In the spectral space, the results are different. Both linear regression and neural networks perform very well even when using only the first principal component of the spectral data. The inclusion of principal components beyond the second component achieves only marginal improvements in performance. This is consistent with the findings depicted in Fig. 3, where 4 principal components are needed to explain 95% variance in the spectral data.

7 Discussion and conclusions

Our study demonstrates that it is possible to accurately approximate the results of radiative transfer models simulating the SWIR3 band by using machine learning methods. Using a large dataset, generated with the LINTRAN RTM within the RemoTeC algorithm, we were able to learn predictive models that emulate the simulations with very high precision, in both the forward and the backward direction. When using a 50-times smaller dataset, generated with SCIATRAN, modeling in the backward direction was not possible, while forward models still achieved good predictive performance. Since generating the training dataset is a computationally costly and time-consuming task, achieving satisfactory performance with a much smaller training dataset is a very promising result.

We have shown that special care should be given to the preparation of the training dataset. The results on the larger LINTRAN dataset indicated redundancy and oversampling of certain parameters. When using latin hypercube sampling, the intrinsic dimensionality of the atmospheric parameter space increased (relative to its total dimensionality), indicating richer information content in the dataset. We believe the sampling procedure was important for preserving the predictive performance as we decreased the size of the dataset. LHS was chosen due to its previous demonstrated use within similar studies (Gómez-Dans et al. 2016), however other sampling techniques exist such as factorial design and sparse grid, that were not considered in this study. In addition, in recent years intelligent sampling methods (otherwise known as active learning, adaptive sampling or Bayesian Optimisation) are gaining popularity in the field of surrogate models and also in the EO domain (Svendsen et al. 2020), as these methods allow optimal solutions to be found by choosing surrogate models that perform best in optimal or sensitive regions. To achieve this, the surrogate models are run iteratively to find the optimal regions, then once these regions are defined, the data can be sampled more efficiently. Given the impact of the sampling procedure on predictive performance, future work could extend towards testing different intelligent sampling methods and their impact on the performance of surrogate models.

We found that the use of dimensionality reduction in the spectral space is necessary, due to the high resolution and low intrinsic dimensionality of the spectra. Dimensionality reduction in the parameter space does not degrade the performance of predictive models and in some cases even improves the performance. Furthermore, the predictive model can be simpler when mapping between spaces with fewer dimensions, leading to improved

computational efficiency. In our experiments, making predictions using the neural network was approximately 33% faster when mapping between the reduced spaces than when working in the original space. Both options need only a few tenths of a second to make predictions for 5000 examples, while the original simulation requires hours or days of computation. The main motivation for developing surrogate models is to improve computational efficiency and the models developed by using machine learning proved to be very successful at that task. The best performing surrogate model used PCA for dimensionality reduction on both the input and the output spaces and a shallow neural network as the model for making predictions.

Currently the lower dimensional embedding of the spectral and parameter spaces are learned independently followed by learning independent mappings between them. A promising direction for further work is the learning of a joint lower dimensional embedding of the spectra and parameters, such that both spectra and parameters can be reconstructed from this space given only spectra or parameters. Such an approach comprises both dimensionality reduction and mapping between the two spaces in a single end-to-end solution.

It is worth noting that the surrogate models trained on the two datasets can be operationalised and used in practice. In applications focused on scenarios with narrower expected atmospheric properties, a backward model can completely replace the RTM. On the other hand, on novel or exploratory applications, the forward model trained using uniformly sampled parameters can be used to inform the retrieval algorithms efficiently during the process of parameter estimation, thus still significantly reducing the required computational time.

The current focus of this study has been examining the emulation of TROPOMI L1B spectra. However, the next steps aim to incorporate this work within full retrieval schemes that retrieve Methane and other trace gases/pollutants. The work presented in this paper has focused only on methane in a very short spectral window present in TROPOMI. However, the learning of surrogate models is instrument/spectral window agnostic, meaning that this method can be applied to numerous other satellite instruments currently in orbit, sensitive to different trace gases/pollutants. With the advent of geo-stationary satellite missions such as Sentinel-4 (Ingmann et al. 2012), Tropospheric emissions: Monitoring of pollution (TEMPO) (Zoogman et al. 2017) and the Geostationary Environmental Monitoring Spectrometer (GEMS) (Nicks et al. 2018), hourly estimates of pollutant concentrations over Europe, North America and East Asia, respectively, at very high spatial resolution will be possible. These missions are focused on measuring short lived gases such as NOx and SO2, where speedy processing of the data output from these instruments is especially important. The high spatial and temporal nature of these instruments means millions of spectra will be generated daily, making the efficient generation of trace gas estimates using computationally expensive traditional retrieval algorithms challenging. The methods for learning surrogate models presented in this paper could be used to replace the RTMs at the core of the retrieval algorithms planned for these satellite missions, increasing the speed of the retrieval process whilst retaining the same accuracy, still based in physical processes.

Additionally, the increased computational efficiency of utilising surrogate models (emulators) over the full RTMs, could allow the development of surrogate models for more complex physics. Currently the retrieval of methane (and other trace gases) in complex atmospheric conditions, such as over clouds/haze, cities or heterogeneous surface conditions requires more complex modelling that is not possible to implement fully within operational processing schemes without a many-fold increase in processing time. The approach for generating surrogate models in this study could be adapted to different training datasets that represent more complex physics, allowing fast and efficient retrievals and therefore

a better understanding and quantification of Methane and other trace gases/pollutants in complicated situations, that currently cannot be fully achieved within operational processing schemes.

In conclusion, our analysis and experiments demonstrate the viability of surrogate models as emulators for radiative transfer models and demonstrate the importance of giving attention to the sampling technique and the preparation of the dataset. Efficient and accurate surrogate models can be learned best by reducing both input and output spaces using PCA, and learning a shallow neural network to map between them. The surrogate models developed in this study can be further implemented within operational retrieval schemes of trace gases, whereby the reduced computational cost and timing of surrogate models can not only reduce overhead costs in operational processing schemes for S5P/TROPOMI and other satellite instruments, but also allow retrievals using more complex physics that currently cannot be fully realised.

Acknowledgements Thanks to J. Landgraf at the Dutch Space Research Organisation (SRON) for the use of the RemoTeC tool and the initial atmospheric/simulation database. Thanks to the Institute of Remote Sensing/Institute of Environmental Physics (iup/ife) at the University of Bremen for use of the SCIATRAN tool.

Author contributions JB: software, investigation, formal analysis, visualization, writing - original draft; JT: conceptualization, methodology, formal analysis, writing - original draft; JA: investigation, methodology, validation, writing - original draft; EM: conceptualization, validation, writing - original draft, funding acquisition; SD: funding acquisition, supervision, validation, writing - review and editing.

Funding This study was supported by the Slovenian Research Agency, via the Grants P2-0103 and N2-0128, and the European Commission, through the projects TAILOR (Grant number 952215) and AI4EU (Grant number 825619).

Availability of data and material At this time, data and other material are not publicly available.

Code availability At this time, code is not publicly available.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

Consent for publication The paper includes only original data and images.

Ethical approval The presented research did not involve human or animal participants and required no ethics approval.

Consent to participate The presented research did not involve human participants and thus consents to participate do not apply.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M. (2016). Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation {OSDI}* (pp. 265–283).

Bovensmann, H., Burrows, J. P., Buchwitz, M., Frerick, J., Noël, S., Rozanov, V. V., Chance, K. V., & Goede, A. P. H. (1999). SCIAMACHY: Mission objectives and measurement modes. *Journal of the Atmospheric Sciences, 56*(2), 127–150.

Brence, J., Tanevski, J., Adams, J., Malina, E., Džeroski, S. (2020). Learning surrogates of a radiative transfer model for the sentinel 5p satellite. In *International Conference on Discovery Science*. Springer (pp. 217–230).

Brodrick, P. G., Thompson, D. R., Fahlen, J. E., Eastwood, M. L., Sarture, C. M., Lundeen, S. R., Olson-Duvall, W., Carmon, N., & Green, R. O. (2021). Generalized radiative transfer emulation for imaging spectroscopy reflectance retrievals. *Remote Sensing of Environment*. https://doi.org/10.1016/j.rse.2021.112476

Butz, A., Galli, A., Hasekamp, O., Landgraf, J., Tol, P., & Aben, I. (2012). TROPOMI aboard Sentinel-5 Precursor: Prospective performance of CH4 retrievals for aerosol and cirrus loaded atmospheres. *Remote Sensing of Environment, 120*, 267–276. https://doi.org/10.1016/j.rse.2011.05.030

Cai, L., Ren, L., Wang, Y., Xie, W., Zhu, G., & Gao, H. (2021). Surrogate models based on machine learning methods for parameter estimation of left ventricular myocardium. *Royal Society Open Science*. https://doi.org/10.1098/rsos.201121

Doquet, G., & Sebag, M. (2020). Agnostic feature selection. In U. Brefeld, E. Fromont, A. Hotho, A. Knobbe, M. Maathuis, & C. Robardet (Eds.), *Machine Learning and Knowledge Discovery in Databases* (pp. 343–358). Cham: Springer International Publishing.

Hedelt, P., Efremenko, D. S., Loyola, D. G., Spurr, R., & Clarisse, L. (2017). Volcanic SO2 plume height retrieval from UV sensors using a full-physics inverse learning machine algorithm. *International Journal of Remote Sensing, 38*, 1–27.

Facco, E., d'Errico, M., Rodriguez, A., & Laio, A. (2017). Estimating the intrinsic dimension of datasets by a minimal neighborhood information. *Scientific Reports*. https://doi.org/10.1038/s41598-017-11873-y

Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning, 63*(1), 3–42.

Goodfellow, I., Bengio, Y., Courville, A. (2016). Deep learning. MIT Press. http://www.deeplearningbook.org.

Gómez-Dans, J. L., Lewis, P. E., & Disney, M. (2016). Efficient emulation of radiative transfer codes using gaussian processes and application to land surface parameter inferences. *Remote Sensing*. https://doi.org/10.3390/rs8020119

Hasekamp, O., Lorente, A., Hu, H., Butz, A., Aan De Brugh, J., Landgraf, J. (2021). Algorithm theoretical baseline document for Sentinel-5 precursor methane retrieval. Tech. rep., Netherlands Institute for Space Research. https://sentinels.copernicus.eu/documents/247904/2476257/Sentinel-5P-TROPOMI-ATBD-Methane-retrieval.pdf/f275eb1d-89a8-464f-b5b8-c7156cda874e.

Hasekamp, O. P., & Landgraf, J. (2002). A linearized vector radiative transfer model for atmospheric trace gas retrieval. *Journal of Quantitative Spectroscopy and Radiative Transfer, 75*(2), 221–238. https://doi.org/10.1016/S0022-4073(01)00247-3

Hedelt, P., Efremenko, D. S., Loyola, D. G., Spurr, R., & Clarisse, L. (2019). Sulfur dioxide layer height retrieval from Sentinel-5 Precursor/TROPOMI using FP_ILM. *Atmospheric Measurement Techniques, 12*(10), 5503–5517.

Hilboll, A., Sanders, A. F., Borchardt, J. (2018). Pyatran: V0.3.1. 10.5281/zenodo.1439269. https://doi.org/10.5281/zenodo.1439269.

Himes, M. D., Harrington, J., Cobb, A. D., Baydin, A. G., Soboczenski, F., O'Beirne, M. D., Zorzan, S., Wright, D. C., Scheffer, Z., Domagal-Goldman, S. D., Arney, G. N. (2020). Accurate machine learning atmospheric retrieval via a neural network surrogate model for radiative transfer. arXiv preprint arXiv:2003.02430.

Hu, H., Hasekamp, O., Butz, A., Galli, A., Landgraf, J., Aan De Brugh, J., et al. (2016). The operational methane retrieval algorithm for TROPOMI. *Atmospheric Measurement Techniques, 9*, 5423–5440. https://doi.org/10.5194/amt-9-5423-2016

Hu, H., Landgraf, J., Detmers, R., Borsdorff, T., Aan de Brugh, J., Aben, I., Butz, A., Hasekamp, O. (2018). Toward global mapping of methane with TROPOMI: First results and intersatellite comparison to GOSAT.

Ingmann, P., Veihelmann, B., Langen, J., Lamarre, D., Stark, H., & Courrèges-Lacoste, G. B. (2012). Requirements for the GMES Atmosphere Service and ESA's implementation concept: Sentinels-4/-5 and -5p. *Remote Sensing of Environment*, *120,* 58–69. https://doi.org/10.1016/j.rse.2012.01.023.

IPCC. (2014). Fifth assessment report—impacts, adaptation and vulnerability. http://www.ipcc.ch/report/ar5/wg2/.

Kingma, D. P., Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Levina, E., Bickel, P. (2005). Maximum likelihood estimation of intrinsic dimension. In L. Saul, Y. Weiss, L. Bottou (Eds.), Advances in Neural Information Processing Systems, vol. 17. MIT Press.

Liaw, A., Wiener, M., et al. (2002). Classification and regression by random forest. *R News, 2*(3), 18–22.

Liu, Y., Caballero, R., & Monteiro, J. M. (2020). RadNet 1.0: Exploring deep learning architectures for longwave radiative transfer. *Geoscientific Model Development, 13*(9), 4399–4412.

Loyola, D. G., Xu, J., Heue, K. P., & Zimmer, W. (2020). Applying FP-ILM to the retrieval of geometry-dependent effective Lambertian equivalent reflectivity (GE-LER) daily maps from UVN satellite measurements. *Atmospheric Measurement Techniques, 13*(2), 985–999. https://doi.org/10.5194/AMT-13-985-2020

Lukšič, Ž, Tanevski, J., Džeroski, S., & Todorovski, L. (2019). Meta-model framework for surrogate-based parameter estimation in dynamical systems. *IEEE Access, 7*, 181829–181841. https://doi.org/10.1109/ACCESS.2019.2959846

Muller, J. P., Lewis, P., Bréon, F. M., Bacour, C., Price, I., Chaumat, L., Prunet, P., Gonzales, L., Schlundt, C., Vountas, M., Burrows, J., von Hoyningen-Huene, W., Guanter, L., Fischer, J., North, P., Heckel, A., Straume-Lindner, A. G. (2013). A surface reflectance database for ESA's Earth Observation Missions (ADAM). In *ESA Living Planet Symposium 2013*. Edinburgh.

Nicks, D., Baker, B., Lasnik, J., Delker, T., Howell, J., Chance, K., Liu, X., Flittner, D., Kim, J. (2018). Hyperspectral remote sensing of air pollution from geosynchronous orbit with GEMS and TEMPO. In X. Xiong, T. Kimura (Eds.), Earth Observing Missions and Sensors: Development, Implementation, and Characterization V, International Society for Optics and Photonics, SPIE, vol. 10781 (pp. 118–124).

Pal, A., Mahajan, S., & Norman, M. R. (2019). Using deep neural networks as cost-effective surrogate models for super-parameterized E3SM radiative transfer. *Geophysical Research Letters, 46*(11), 6069–6079.

Pearson, K. (1901). LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 2*(11), 559–572.

Qiu, J., Tan, W., Zhao, G., Yu, Y., & Zhao, C. (2021). New correction method for the scattering coefficient measurements of a three-wavelength nephelometer. *Atmospheric Measurement Techniques, 14*(7), 4879–4891. https://doi.org/10.5194/amt-14-4879-2021

Rozanov, V. V., Rozanov, A. V., Kokhanovsky, A. A., & Burrows, J. P. (2014). Radiative transfer through terrestrial atmosphere and ocean: Software package SCIATRAN. *Journal of Quantitative Spectroscopy and Radiative Transfer, 133*, 13–71. https://doi.org/10.1016/j.jqsrt.2013.07.004

Sacks, J., Welch, W. J., Mitchell, T. J., & Wynn, H. P. (1989). Design and analysis of computer experiments. *Statistical Science, 4*(4), 409–423. https://doi.org/10.1214/ss/1177012413

Schweidtmann, A. M., & Mitsos, A. (2019). Deterministic global optimization with artificial neural networks embedded. *Journal of Optimization Theory and Applications, 180*(3), 925–948. https://doi.org/10.1007/s10957-018-1396-0

Servera, J. V., Rivera-Caicedo, J. P., Verrelst, J., Muñoz-Marí, J., Sabater, N., Berthelot, B., Camps-Valls, G., & Moreno, J. (2021). Systematic assessment of MODTRAN emulators for atmospheric correction. *IEEE Transactions on Geoscience and Remote Sensing, 60*, 1–17.

Song, Z., Chen, B., Huang, Y., Dong, L., & Yang, T. (2021). Estimation of PM2.5 concentration in china using linear hybrid machine learning model. *Atmospheric Measurement Techniques, 14*(8), 5333–5347.

Svendsen, D. H., Martino, L., & Camps-Valls, G. (2020). Active emulation of computer codes with gaussian processes-application to remote sensing. *Pattern Recognition*. https://doi.org/10.1016/j.patcog.2019.107103

Veefkind, J. P., Aben, I., McMullan, K., Förster, H., de Vries, J., Otter, G., Claas, J., Eskes, H. J., de Haan, J. F., Kleipool, Q., van Weele, M., Hasekamp, O., Hoogeveen, R., Landgraf, J., Snel, R., Tol, P., Ingmann, P., Voors, R., Kruizinga, B., & Levelt, P. F. (2012). 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 Sensing of Environment, 120*, 70–83. https://doi.org/10.1016/j.rse.2011.09.027

Verdel, N., Tanevski, J., Džeroski, S., & Majaron, B. (2020). Predictive model for the quantitative analysis of human skin using photothermal radiometry and diffuse reflectance spectroscopy. *Biomedical Optics Express, 11*(3), 1679–1696. https://doi.org/10.1364/BOE.384982

Verrelst, J., Rivera, J. P., Gómez-Dans, J., Camps-Valls, G., Moreno, J. (2015). Replacing radiative transfer models by surrogate approximations through machine learning. In *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* (pp. 633–636). IEEE.

Verrelst, J., Sabater, N., Rivera, J. P., Muñoz-Marí, J., Vicent, J., Camps-Valls, G., & Moreno, J. (2016). Emulation of leaf, canopy and atmosphere radiative transfer models for fast global sensitivity analysis. *Remote Sensing, 8*(8), 673. https://doi.org/10.3390/rs8080673

Verrelst, J., Rivera Caicedo, J. P., Muñoz-Marí, J., Camps-Valls, G., & Moreno, J. (2017). SCOPE-based emulators for fast generation of synthetic canopy reflectance and sun-induced fluorescence spectra. *Remote Sensing, 9*(9), 927. https://doi.org/10.3390/rs9090927

Vicent, J., Verrelst, J., Rivera-Caicedo, J. P., Sabater, N., Muñoz-Marí, J., Camps-Valls, G., & Moreno, J. (2018). Emulation as an accurate alternative to interpolation in sampling radiative transfer codes. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 11*(12), 4918–4931. https://doi.org/10.1109/JSTARS.2018.2875330

Xu, J., Schussler, O., Rodriguez, D. G. L., Romahn, F., & Doicu, A. (2017). A novel ozone profile shape retrieval using full-physics inverse learning machine (FP-ILM). *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 10*(12), 5442–5457. https://doi.org/10.1109/JSTARS.2017.2740168

Yondo, R., Andrés, E., & Valero, E. (2018). A review on design of experiments and surrogate models in aircraft real-time and many-query aerodynamic analyses. *Progress in Aerospace Sciences, 96*, 23–61. https://doi.org/10.1016/j.paerosci.2017.11.003

Zoogman, P., Liu, X., Suleiman, R., Pennington, W., Flittner, D., Al-Saadi, J., Hilton, B., Nicks, D., Newchurch, M., Carr, J., Janz, S., Andraschko, M., Arola, A., Baker, B., Canova, B., Chan Miller, C., Cohen, R., Davis, J., Dussault, M., … Chance, K. (2017). Tropospheric emissions: Monitoring of pollution (TEMPO). *Journal of Quantitative Spectroscopy and Radiative Transfer, 186*, 17–39.

Authors and Affiliations

Jure Brence[1,2] · Jovan Tanevski[1,3] · Jennifer Adams[4,5] · Edward Malina[6] · Sašo Džeroski[1,2,7]

[1] Jozef Stefan Insitute, Ljubljana, Slovenia

[2] Jozef Stefan International Postgraduate School, Ljubljana, Slovenia

[3] Heidelberg University and Heidelberg University Hospital, Heidelberg, Germany

[4] RHEA Group Spa, Frascati, Italy

[5] ESA/ESRIN, Frascati, Italy

[6] Formerly at ESA/ESTEC, Noordwijk, The Netherlands

[7] Φ-lab, ESA/ESRIN, Frascati, Italy