RC1 Comments and responses

We thank the Reviewer for her/his insightful comments. We appreciate the time and effort invested in providing detailed suggestions. Below, we address each comment in detail and outline the corresponding actions we have taken.

Comment: In the introduction it is mentioned that "ML techniques have gained traction in weather and climate modeling and observations [...] particularly in parameter optimization tasks like calibration", but I feel several relevant works exploring the use of emulators for tuning weather prediction and climate models, closely related to the long-term aims of the authors as far as I can interpret, are missing. I feel these should be cited. Here are a few examples. Daniel Williamson, Michael Goldstein, Lesley Allison, Adam Blaker, Peter Challenor, Laura Jackson, and Kuniko Yamazaki, "History matching for exploring and reducing climate model parameter space using observations and a large perturbed physics ensemble" (2013) Fleur Couvreux et al., "Process-Based Climate Model Development Harnessing Machine Learning: I. A Calibration Tool for Parameterization Improvement" (2020) Katherine Dagon, Benjamin M. Sanderson, Rosie A. Fisher, and David M. Lawrence, "A machine learning approach to emulation and biophysical parameter estimation with the Community Land Model, version 5" (2020) Duncan Watson-Parris, Andrew Williams, Lucia Deaconu, and Philip Stier, "Model calibration using ESEm v1.1.0 – an open, scalable Earth system emulator" (2021) Davide Cinquegrana, Alessandra Lucia Zollo, Myriam Montesarchio, and Edoardo Bucchignani, "A Metamodel-Based Optimization of Physical Parameters of High Resolution NWP ICON-LAM over Southern Italy" (2023)

Reply: We thank the Reviewer for providing these valuable references. They are indeed relevant to our work and help illustrate the broader context of using emulators for tuning weather prediction and climate models.

Action: We have added the suggested references in the Introduction section.

Comment: In Page 4 it is stated that "There is no upper limit for the number of parameters that can be analyzed", but of course the higher the dimensionality the harder the training of a surrogate can become, It would be useful to specify here how the number of simulations required scales with the number of parameters.

Reply: We fully agree with the Reviewer's comment. The exploration of how the number of simulations required scales with the number of parameters is missing in this paper, as the main aim was to present the functionalities of the ML-AMPSIT tool using a simple idealized case study. However, it is difficult to evaluate a priori how the number of simulations needed to train the surrogate models scales with the number of parameters, since it can be dependent on the case study. We have added this consideration in the text.

Action: We have updated the text to emphasize the importance of dimensionality. The revised text now reads:

> "There is no upper limit for the number of parameters that can be analyzed, but it is worth noting that the sensitivity analysis could converge significantly more slowly in high-dimensional (i.e., with more parameters) problems. Moreover, the scalability with the number of parameters can highly depend on the case study considered."

Comment: In Page 7, Eq. (2), a definition of the terms $V_{i,j,\ldots}$ is missing, and should be added.

Reply: Thanks for noting this.

Action: The definition of $V_{i,j,\ldots}$ has been added:

> "where V_i is the main effect variance, representing the contribution of the i-th input parameter to the output variance, V_{ij} is the second-order interaction effect variance, representing the combined contribution of the i-th and j-th input parameters to the output variance, and so on up to $V_{12..k}$, which represents the interaction effect variance of all k input parameters together."

Comment: In the Sections from 2.3.1 to 2.3.5 it is unclear how these different algorithms are used to compute an importance metric for the parameters. As far as I understood, the Sobol indices (the first-order one specifically) are computed only using Gaussian processes and Bayesian ridge regression. What is then precisely done when using the other ML algorithms explained? This explanation should be added to the manuscript.

Reply: We recognize that the original manuscript did not clearly convey how the feature importance is obtained for each of the surrogate models used.

Action: We have inserted a new section to clarify this process:

Feature importance computation

Each of the algorithms implemented in this study provides a method for calculating feature importance, albeit through different approaches. In principle, a single sensitivity method could be used to evaluate feature importance across all algorithms. However, some algorithms have built-in methods specifically designed to align with their inherent characteristics.

- Fitting Methods: LASSO and SVM derive feature importance from the model coefficients. In these linear models, the magnitude of the coefficients indicates the strength and direction of the relationship between each feature and the target variable. Specifically, in the `scikit-learn` library, this can be accessed through the `best_estimator_.coef_` attribute. Larger absolute values of these coefficients indicate greater importance.
- Tree-based algorithms: for CART, RF, and XGboost, feature importance is assessed using the Mean Decrease in Impurity (MDI) method. This method quantifies the contribution of each feature to the overall prediction accuracy by measuring how much each feature decreases the impurity of the splits in which it is involved. For RF and XGboost, the final value is obtained by averaging over all the trees in the ensemble. In `scikit-learn`, these contributions are accessible through the `feature_importances_` attribute. The MDI method is particularly effective because it directly measures the impact of each feature on the model's decision process, providing a clear indication of feature importance.
- Probabilistic methods: GPR and BRR do not have a built-in mechanism for directly assessing feature importance. Therefore, in this work, the Sobol method was used to infer feature importance. Once built and tested against the original model outputs, the GPR and BRR surrogate models can be used to perform a GSA in substitution of the original model. By using a surrogate model, the computational cost of running the original model for a large number of input combinations is avoided. Instead, the surrogate model can

> > be used to generate a large number of input combinations with significantly less computational time and evaluate their impact on the output. Over these samples, in ML-AMPSIT the Sobol sensitivity indices are computed following the definition proposed by Saltelli et al. (2008). The user can then compare the Sobol indices evaluated with both GPR and BRR, providing information on their robustness and reliability. In the proposed tool, after the algorithm generates the optimal surrogate model, it uses the Python library `SALib` to compute the Sobol first-order index as a score for the sensitivity importance of each parameter. The Sobol total index and Sobol second-order interaction term are available for users who wish to examine the presence of strong parameter interactions.
>
> Despite the differences in the feature importance calculation approaches of the different algorithms, each method is applied to standardized, non-dimensional data and each feature importance set is scaled between [0,1]. This ensures that feature importance scores are comparable across models. The primary objective of all these methods is to quantify the sensitivity of the model output to changes in the input features. Consequently, the feature importance scores obtained from these different methods provide a well-posed comparison of parameter sensitivities. By evaluating and comparing these scores, it is possible to gain a comprehensive understanding of the relative importance of each feature across different modeling approaches, which increases the robustness of the results."

Comment: In Page 10, Section 2.3.6, the authors state that "GPR is a non-parametric method, i.e., it does not make assumptions about the functional form of the relationship between the input and output variables". The underlying assumptions on the functional form are contained in the chosen kernel, so there are in fact assumptions one has to make when using Gaussian processes. Maybe the authors here mean that there is no assumption of linearity with the chosen RBF kernel (as they specify later on)? Also, it seems that the authors do train the parameters of the kernel (e.g., lengthscale), so the adjective "non-parametric" may be confusing here.

Reply: We acknowledge that the text was creating unintentional ambiguity regarding the assumptions made by Gaussian Process Regression (GPR).

Action: We have revised the text to clarify this point. The following text has been added:

> "GPR is often described as a non-parametric method because it does not assume a specific functional form for the relationship between input and output variables. Instead, it models this relationship as a distribution over possible functions, allowing for flexibility in the shape of the regression curve. However, it is important to note that there are underlying assumptions about the functional form embedded in the chosen kernel. The kernel influences the shape and properties of the functions that the Gaussian process can learn."

Comment: In Page 10, Section 2.3.7, it should be specified what E and H in the equations mean in the context of the problem considered.

Action: We have added definitions for E and H:

> "Defining both a prior distribution $p(H)$ for the model parameters H and a likelihood function $p(E|H)$ for the ingested data E, the BRR model computes the posterior distribution over functions $p(H|E)$ given the observed data through the use of Bayes' theorem $p(H|E) = \frac{p(E|H) \cdot p(H)}{p(E)}$, where $p(E) = \int p(E|H) \cdot p(H)\, dH$ is the marginal likelihood."

Comment: In page 11, Section 2.3.7, the authors state "The same procedure used for the GPR algorithm to leverage the probabilistic output for deriving feature importance coefficients is also implemented here to compute the Sobol first-order sensitivity index". I find confusing why the probabilistic nature of GPR or BRR is important for the calculation of the Sobol indices. In principle also 'deterministic' emulators like neural networks can be used to calculate Sobol indices. Can the authors comment on what they mean with this?

Reply: We acknowledge the confusion regarding the probabilistic nature of the algorithms chosen to implement the Sobol method. There is nothing inherently special about the probabilistic nature of GPR or BRR for calculating the Sobol indices. These algorithms were selected because they do not have in-built methods for feature importance analysis compared to the other algorithms implemented in ML-AMPSIT.

One of the aims of the paper is to introduce a refined methodology for sensitivity analysis that addresses common issues in the literature, such as the simplistic assumption of linearity and the absence of interaction effects. Thus, we aimed to implement the Sobol method, an advanced sensitivity analysis technique historically considered too computationally expensive for numerical weather prediction models, to explore how quickly this method could be executed using surrogate models.

The Sobol method could, in principle, be used with all the surrogate models chosen in ML-AMPSIT. However, to provide a validation mechanism, other algorithms were implemented with their specific methodologies to evaluate feature importance. This approach allows for the production of a reliable ensemble and offers a metric for comparing the Sobol indices obtained.

Action: We have added the new section ”Feature importance computation” in the revised manuscript which should clear doubts about the connection between the used surrogate models and the computation of feature importance.

Comment: In Page 13, the authors write “The spread of the ensemble tends to be larger over water than over land, especially before sunrise, indicating that the variation of the input parameters has a larger effect on v over water”. Since most of the parameters varied were land-related parameters, I find this seemingly counterintuitive. Do the authors have a qualitative explanation for that?

Reply: The development and strength of sea and land breezes depend on the temperature contrasts between land and water. Therefore, it is reasonable that changes in the land parameters also affect atmospheric variables, in particular wind speed, over water, due to possible differences in the temperature contrasts between land and water and, as a consequence, in the timing and strength of the sea and land breezes. This is particularly true for the water point chosen for this study, which is close to the land/water interface. It is more difficult to understand why the ensemble spread is larger over water than over land. It may be connected to the higher friction over land, which dampens the variability induced by changes in the parameters' values. However, we prefer not to add speculations on this aspect in the text.

Action: We have added the following text to the manuscript to clarify this point:

> ”It is worth noting that, even if only land parameters have been considered in this work, the spread of the ensemble tends to be larger over water than over land, especially before sunrise. Indeed, changes in land parameters affect the thermal contrasts between land and water, and thus the characteristics of the sea and land breeze, including their timing and strength. This highlights that changes in surface parameters can influence atmospheric variables not only locally, especially when they affect the development of thermally-driven circulations.”

Comment: From Page 14, when presenting the results the authors refer to the “importance” of the parameters, but no formula for this was given, especially in the context of LASSO, SVM, CART, RF, XGBoost. Please add a proper definition of it in the manuscript.

Reply: We thank the Reviewer for this comment that helped to improve the clarity of the manuscript. We have added the new section ”Feature importance computation” that should clarify this aspect.

Comment: In the end, Page 26, the authors state “It is then clear that ML-AMPSIT significantly reduces the number of simulations needed for sensitivity analysis and extraction of feature importance”. I find this a bit of a strong statement that should be mitigated. It is by no means clear that 20 or 30 simulations will be sufficient to train the emulators to reach faithful outputs. Specifically, as pointed out by the authors, the comparable performance of the investigated methods suggests the absence of strong non-linearities, which obviously renders the training of the methods more efficient. I expect that in presence of strong non-linearities the amount of training data will need to be increased, and so it remains a question as to whether this number will be systematically smaller than the other existing methods.

Reply: We recognize that the statement in the paper was unintentionally implying a generality that is not guaranteed for different setups and case studies.

Action: We have added the following text to mitigate the statement and provide appropriate context:

"It should be noted that the results presented in this paper are limited to the simple case study considered here to test the tool functionalities. In particular, it is expected that more simulations can be needed for training the algorithms in more complex scenarios, when non-linearities are more strongly involved in the input-output relations."