**Response to Anonymous Referee #2**

We sincerely thank the reviewer for her/his effort and the very useful comments. We have revised our manuscript *Dynamic weighted ensemble of geoscientific models via automated machine learning-based classification* and have addressed all points raised by the reviewer.

Below, we provide a point-by-point response to all the comments. Text by the reviewer is in blue and indented. Our response is in black. *New text is green, italic*. *Existing (unchanged) manuscript text is black italic*.

> Thank you for the opportunity to review this interesting study. The authors proposed the AutoML-Ens by ensembling six ML algorithms to find the best weights of predictors. Also, they considered different ensemble methods including BMA, MEAN, and so on to indicate the superior performance of the proposed method compared to these ensemble methods. In my opinion, the manuscript is suitable for publication in Geoscientific Model Development (GMD), after the authors have addressed the following comments and questions.

We sincerely appreciate your comments and suggestions to improve the manuscript. However, the statement that "the AutoML-Ens by assembling six ML algorithms to find the best weights of predictors" is somehow inaccurate. In order to better address your subsequent comments, we would like to first clarify it here:

Specifically, *"an AutoML-based training, validation, and testing workflow is conducted to help automatically find the top classifier (either a specific ML algorithm or an ensemble of a few ML algorithms based on the ensemble learning technique)."* Then, based on this classification model, we further construct predictors, which are essential input variables to develop physics-constrained models incorporated in the final ensemble. These predictors, also referred to as environmental conditions, are associated with the labels derived from physically-based model predictions that exhibit superior performance under specific environmental conditions (or for each sample). Therefore, the weights assigned in this context do not pertain to individual predictors, but instead represent the probabilities (weights) indicating the *"probability of an individual model being optimal under certain environmental conditions"*. Therefore, our focus for each sample lies not in the predicted labels produced by the ML classification model, but rather in the probabilities associated with each class of labels. These probabilities serve as the basis for determining the dynamic weights utilized in our proposed ensemble approach.

1) Because neural networks are one of the ML techniques and standardization is critical for this model architecture, I'm curious if the authors addressed it in the workflow

We extend our appreciation to the reviewer for conducting a thorough review and for raising this point. We would like to confirm that we have acknowledged the importance of standardizing variables for neural networks. However, it is noteworthy that the standardization parameter (standardize) is enabled by default in H2O-AutoML workflow, obviating the need for any specific configurations in this regard. Please refer to the following link for more details: https://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/deep-learning.html

2) According to the authors, the type of problem in this study is classification, as stated in line 159, and they utilized least absolute error to identify the ideal model (Line 118), which is used for regression problems (at least as far as I know). Maybe I misunderstood that, could you help explain this to me?

Thank you for bringing up this important point. We would like to take this opportunity to further clarify the innovation of our study. While we acknowledge that this is a classification problem, it differs from conventional classification models in the sense that our primary focus is not solely on obtaining specific class labels. Instead, we aim to derive the probability that a prediction from various candidate members, under different environmental conditions, will be the optimal prediction for these specific conditions. This probability (i.e., weight), which is often overlooked despite being an available output of the ML classification model, plays a critical role in achieving an ensemble of model predictions at the sample scale.

Here, it is important to note that our concept of the "ideal model" does not pertain to the ML classification model itself but rather to the label (optimal prediction of physically-based models) associated with each sample. This label is utilized for data preprocessing prior to training the ML classification model. At present, we believe that the least absolute error could serve as a reasonable metric for this purpose.

Specifically, for the ML classification model, we employ the logloss metric as the loss function, as provided by H2O-AutoML for multi-classification models. Further details regarding this can be found in our code (https://doi.org/10.6084/m9.figshare.21547134.v2) or by referring to the following link: https://docs.h2o.ai/h2o/latest-stable/h2o-docs/performance-and-prediction.html

3) The authors address the accuracy of the autoML in section 3.1.3, however they don't specify the classes, and I'm curious about the proportion of classes. Is it an imbalance classification problem since the performance metric is easily skewed toward the major class? If so, how did the authors manage this situation?

Thank you for emphasizing the importance of this issue. In order to address your concern, firstly, we here provided the number of labels identified as relatively optimal (with the least absolute error) for each sample in both of our study cases (***Table S2*** and added ***Table S6*** in our *Supporting Information*).

*__Table S2__. Size of the sample labeled as individual PTFs.*

| *PTFs* | *Sample size* |
|---|---|
| *Cosby0* | *7,360* |
| *Carsel & Parrish* | *9,051* |
| *Clapp & Hornberger* | *12,211* |
| *Rosetta3-H1w* | *7,476* |
| *Cosby1* | *6,884* |
| *Cosby2* | *6,882* |
| *Rosetta3-H2w* | *6,498* |
| *Rawls & Brakensiek* | *10,976* |
| *Campbell & Shiozawa* | *14,255* |
| *Rosetta3-H3w* | *7,563* |
| *Wösten* | *11,090* |
| *Weynants* | *9,634* |
| *Vereecken* | *8,719* |

*__Table S6__. Size of the sample labeled as individual ET models.*

| *Model name* | *Sample size* |
|---|---|
| *PT-JPL* | *14,062* |
| *PT-DTsR* | *12,905* |
| *STIC* | *16,065* |
| *SEBS* | *12,903* |
| *RS-WBPM* | *16,869* |
| *EVI-PM* | *10,817* |

The presence of imbalanced class issues was observed in ***Table S2*** and ***Table S6***, although they were not significant: The maximum-to-minimum ratio of class quantities in the two cases was found to be 2.19 and 1.56.

In order to assess the potential impact of not addressing this issue, a new classifier called "**Balanced_withSE**" was trained by enabling the “balance_classes” parameter in the H2O-AutoML workflow (https://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/algo-params/balance_classes.html). A comparison was then conducted between the "**Balanced_withSE**" classifier and the "**Original_withSE**" classifier (utilized in our study) for assembling 13 PTFs (as shown in ***Table S3*** in our *Supporting Information*).

The results indicated that both classifiers demonstrated very similar ensemble prediction accuracy (see the $R^2$ (0.8629 vs 0.8654) and RMSE (0.0444 vs 0.0440 $m^3/m^3$) values in ***Table S3***). However, despite the small difference in our case, the "**Balanced_withSE**" classifier exhibited a slight better ensemble performance than the "**Original_withSE**" classifier. Therefore, the importance of addressing the class imbalance issue has been underscored in the main text (in ***Section 2.2***) as a noteworthy key issue.

*"For a ML classifier, an even distribution of samples across both major and minor classes (i.e., balanced dataset) is needed to guarantee reasonable predictions of not only the majority but also classes with small sample size or extreme values (Kavzoglu, 2009). While the imbalance issue does not have a significant impact on the two examples we presented, we acknowledge its importance in various applications. Fortunately, the H2O-AutoML platform provides a parameter, namely "balanced_class" which allows for addressing class imbalance during model training. Additionally, other methods such as Synthetic Minority Oversampling Technique (SMOTE) proposed by Chawla et al. (2002) can be implemented in the data preprocessing stage to generate synthetic samples for the minority class, further mitigating the class imbalance problem."*

But, note that, in our cases, this issue did not impact our major findings due to the insignificant imbalance. Moreover, as we tried to explain it in ***Section 3.1.3***, "*improving this accuracy is not the overarching objective of AutoML-Ens. Poor accuracy may result from the uneven distribution of available data samples, their low representative ability, and inter-model similarities and dependencies (Holtanová et al., 2019). Especially the similarities within a multi-model ensemble may result from using the same set of data samples, sharing certain components, or being based on the same hypothesis. This makes it difficult to justify the independence assumption between ensemble members, further leading to poor classification"* and *"efforts could be made to reduce the similarities within candidate models to obtain a higher classification accuracy. Moreover, once a good classification accuracy is obtained among the training and testing datasets, the linkage between the predictors and the label in the workflow will be more clearly determined, which can help implement* *and/or modify these candidate models appropriately.”.*

Further, we modified ***Table 1*** in Section 3.2.2 as follows:

*“As can be seen, the best model in terms of lowest classification error was selected to be the stacked ensemble based on all models, followed by the stacked ensemble based on the best of family, XRT, DRF, GBM, XGBoost, and DNN, as well as their variants with different hyperparameters. However, the ranking of performance metrics for the final ensemble predictions differs from the classification accuracy of individual classifiers. While the top classifier, Stacked_Ensemble_All_Models, demonstrates high predictive performance, the XGBoost_grid_1_model_8 classifier achieves the best ensemble prediction with an R$^2$ value of 0.87 and an RMSE of 15.03 W/m$^2$. This result further confirms the primary objective of AutoML-Ens, which is not solely focused on achieving optimal classification results, but rather on finding the optimal utilization and combination of ML algorithms to obtain better predictive performance.”*

***Table 1.*** *Ranking of the 32 models involved in the AutoML-Ens workflow with respect to the mean per class error and their corresponding performance metrics (R2 and RMSE) of their ensemble predictions.*

| Rank | Model* | Mean per class error | R2 | RMSE (W/m2) |
|---|---|---|---|---|
| 1 | Stacked_Ensemble_All_Models | 0.5890107 | 0.8502772 | 16.37276 |
| 2 | Stacked_Ensemble_Best_Of_Family | 0.5901575 | 0.8433838 | 16.74402 |
| 3 | XRT_1 | 0.5990940 | 0.8238412 | 17.80632 |
| 4 | DRF_1 | 0.6000693 | 0.8254552 | 17.72398 |
| 5 | GBM_grid_1_model_1 | 0.6152126 | 0.8594122 | 15.88430 |
| 6 | GBM_4 | 0.6156997 | 0.8050057 | 18.74331 |
| 7 | XGBoost_grid_1_model_4 | 0.6175429 | 0.7896317 | 19.48109 |
| 8 | XGBoost_grid_1_model_7 | 0.6182065 | 0.7919117 | 19.37204 |
| 9 | GBM_5 | 0.6196878 | 0.7930434 | 19.32466 |
| 10 | XGBoost_grid_1_model_9 | 0.6214154 | 0.7940143 | 19.26547 |
| 11 | XGBoost_grid_1_model_8 | 0.6220251 | 0.8742440 | 15.02540 |
| 12 | XGBoost_grid_1_model_1 | 0.6235140 | 0.7981535 | 19.07374 |
| 13 | XGBoost_grid_1_model_3 | 0.6243140 | 0.7928134 | 19.33150 |
| 14 | GBM_3 | 0.6248937 | 0.7836964 | 19.76815 |
| 15 | XGBoost_grid_1_model_5 | 0.6252402 | 0.8135903 | 18.31214 |
| 16 | XGBoost_grid_1_model_6 | 0.6272789 | 0.7797398 | 19.94857 |
| 17 | GBM_grid_1_model_5 | 0.6288796 | 0.7789381 | 20.00014 |
| 18 | XGBoost_2 | 0.6301792 | 0.8286823 | 17.52763 |
| 19 | XGBoost_1 | 0.6313061 | 0.7974012 | 19.11246 |
| 20 | GBM _2 | 0.6322671 | 0.7731042 | 20.27247 |
| 21 | GBM_grid_1_model_3 | 0.6356704 | 0.7716974 | 20.34037 |

| | | | | |
|---|---|---|---|---|
| *22* | *GBM_1* | *0.6371586* | *0.7708355* | *20.38789* |
| *23* | *XGBoost_grid_1_model_2* | *0.6444023* | *0.7593128* | *20.89775* |
| *24* | *GBM_grid_1_model_4* | *0.6470411* | *0.7791697* | *20.04830* |
| *25* | *XGBoost_3* | *0.6479244* | *0.7657713* | *20.60219* |
| *26* | *GBM_grid_1_model_2* | *0.6526127* | *0.8525492* | *16.26434* |
| *27* | *DeepLearning_grid_1_model_2* | *0.6851248* | *0.7089920* | *23.09232* |
| *28* | *DeepLearning_grid_1_model_1* | *0.6976690* | *0.7178891* | *22.38846* |
| *29* | *DeepLearning _1* | *0.7208075* | *0.7084561* | *23.11835* |
| *30* | *DeepLearning_grid_3_model_1* | *0.7247005* | *0.6777100* | *24.45820* |
| *31* | *DeepLearning_grid_2_model_1* | *0.7263856* | *0.7061923* | *23.29444* |
| *32* | *GLM_1* | *0.7417848* | *0.7102180* | *23.17610* |

** The same ML model with different number signs indicates their variants with different hyperparameters.*

We here still hold this opinion on these accuracies of ML classification models. Hope that the above discussion will meet with approval.

4) I'm curious if the authors evaluated the predictors' correlation, as it is preferable to supply more informative information rather than a larger number of predictors for a machine learning model.

We appreciate the reviewer's comment regarding this aspect. In order to address this key issue, we will further discuss and explain it based on our current understanding:

Indeed, when utilizing ML for predictive studies, especially in training regression models, it is crucial to conduct a thorough analysis of the correlations between predictors. This can involve performing covariance analysis, assessing variable importance, and considering the potential elimination or retention of variables based on their degree of correlation.

However, our research focuses on the ensemble of multiple physically-based models, which are formulated based on a comprehensive understanding of "*different biophysical principles*", despite their inherent limitations. These physically-based models utilize environmental variables as inputs that possess meaningful physical interpretations. Consequently, our approach aims to include a wide range of these crucial input variables, enabling ML models to utilize predictors that closely resemble those used by the physical models. This allows for more accurate comparisons between the two approaches and facilitates further exploration of the relationships between predictors and targets.

*"once a multimodel ensemble problem is defined, an extensive spectrum of physically meaningful predictors (i.e., environmental conditions) denoted by* $x_m$ *,*

*where* $m=1,\cdots,M$ *with a single or a combination of few subsets are selected and used to develop physics-constrained models (hereafter the predictions* $\boldsymbol{P}_s$ *where* $s=1,\cdots,S$*).”*

Therefore, the selection of these predictors is depended on physics-constrained models involved in an ensemble. In our two examples, the ensemble of PTFs employed 6 environmental predictors that are essential inputs for constructing these PTFs. These predictors include matric potential, organic carbon, bulk density, and the fractions of sand, clay, and silt content. It is worth noting that there may exist simple or complex correlations among these predictors. For instance, the relationship where the sum of sand, silt, and clay fractions consistently equals 1. Similarly, in the ensemble of cropland ET models, certain key predictors (as listed in our *Supporting Information* ***Table S5***) such as EVI and NDVI, VPD and $T_a$ may also exhibit specific relationships. However, we would like to emphasize our intention to fully utilize the knowledge provided by physically-based models and apply it to ML approaches in an ensemble. This perspective itself deserves attention and consideration.

Moreover, we would like to highlight two recently published studies that share similarities with our approach and perspective, and may be of interest in this context: To explain (Leaf Area Index) LAI trends, Abel et al. (2023) fitted an XGBoost model using anthropogenic, climatic, topographical, and soil variables as covariates. They said that “We do not apply a variable selection procedure and instead use all available variables to parameterize the models This will ensure models with the highest possible explanatory power, and overfitting is no concern, as our aim is to explain and not to predict LAI trends”. Sun et al. (2023) proposed a ML-based procedure for accelerating the spin-up of terrestrial biosphere models (TBM). For the predictors, they “consist of up to 27 variables, 20-25 variables depending on the TBM model version characterizing its driving data”. It is worth noting that certain selected variables may exhibit high correlations for specific grid points on a global scale in this case.

Yet, we do hope our explanation can meet with your approval. Please let us know if you have any other comments on this issue.

**Reference**

Abel, C., Abdi, A. M., Tagesson, T., Horion, S., & Fensholt, R. (2023). Contrasting ecosystem vegetation response in global drylands under drying and wetting conditions. Global Change Biology, 29, 3954– 3969.

https://doi.org/10.1111/gcb.16745

Sun, Y., Goll, D. S., Huang, Y., Ciais, P., Wang, Y.-P., Bastrikov, V., & Wang, Y. (2023). Machine learning for accelerating process-based computation of land biogeochemical cycles. Global Change Biology, 29, 3221– 3234. https://doi.org/10.1111/gcb.16623

5) Generally speaking, the performance of the developed model is assessed based on benchmark. For example, multi-linear regression and logistic regression methods are used for regression and classification problems as baseline, respectively. I would like to see how well your developed model is compared to the baseline.

We appreciate the reviewer's insightful question regarding benchmarking in the field of ML. However, it is important to clarify that our research focus is not primarily on benchmarking individual ML classifiers, but rather on the results obtained through ensemble methods. Therefore, our approach to benchmarking is centered around evaluating the performance of different ensemble techniques, particularly the fixed-weighted MEAN ensemble that we consider “*as a benchmark*”. Moreover, we have compared this MEAN ensemble with widely used methods in the field, such as the BMA. Additionally, we have incorporated two ensemble methods in the two specific cases, namely the HME and MLP, which have been proposed and evaluated in our previous studies (*Zhang et al., 2020; Bai et al., 2021*). Based on these considerations, we believe that we have adequately addressed the issue of benchmarking in our study.

**Reference**

*Zhang, Y., Schaap, M. G., & Wei, Z. (2020). Development of hierarchical ensemble model and estimates of soil water retention with global coverage. Geophysical Research Letters, 47, e2020GL088819*

*Bai, Y., Zhang, S., Bhattarai, N., Mallick, K., Liu, Q., Tang, L., Im, J., Guo, L., & Zhang, J. (2021). On the use of machine learning based ensemble approaches to improve evapotranspiration estimates from croplands across a wide environmental gradient. Agricultural and forest meteorology, 298-299, 108308*

6) Figure 2 shows 47 flux sites, but the boxplots for mean annual temperature and mean annual precipitation show 44 and 42 Flux sites, respectively. Could you please clarify the differences?

We double checked the metadata provided on the data websites associated with all the sites we utilized, which include: http://asiaflux.net/?page_id=22, http://www.europe-fluxdata.eu/home/sites-list, https://ameriflux.lbl.gov/sites/site-search/, and https://fluxnet.org/sites/site-list-and-pages/. It appears that the primary reason for the missing data is the variation in data availability periods among the sites. Specifically, we identified five sites (FR-Aur, FR-Lam, IT-Cas, JP-MSE, and US-Lin) with missing MAP values, and two sites (JP-MSE and US-Lin) with missing MAT values. Consequently, we have adjusted the presented sample size from *N=44* to *N=45* and provided a modified ***Figure 2***.

Additionally, a list of 47 eddy covariance flux sites covering croplands from AmeriFlux (AM), AsiaFlux (AS), FLUXNET (FN), and European Flux Database Cluster (EF) networks can be found in our *Supporting Information* ***Table S4***.

While it is possible to supplement these missing values with climatic reanalyses covering longer time periods, it does not affect the results of our research work.

7) Could you elaborate the machine learning classifier? It is hard for me to follow this term.

We are sorry for the possible confusion regarding the ML classifier. By this point, we hope that the reviewer has gained a better understanding of the term ML classifier in our study.

The key point of our study revolves around dynamic weights, which aims to fully leverage the influence of environmental constraints on the performance of physically-based models to effectively combine the strengths of individual physically-based models under varying environmental conditions ("*i.e., weights assigned to candidate ensemble members vary depending on the spatial and temporal changes in environmental conditions and the performance capabilities of individual models under these conditions."*).

To obtain the dynamic weights, we focus on the probability predictions available within a ML classifier's outputs. While we have not conducted further tests, we speculate that certain traditional statistical methods (*e.g., the known Kriging methods*) that provide similar probabilities (weights) could also be integrated into this workflow as *possible extensions*. However, at present, we have a stronger inclination towards utilizing ML classifiers, especially when supported by extensive datasets for specific cases.

Therefore, we propose leaving this question open for readers who may further explore its significance and potential implications.

**A few words from the first author (Hao Chen):**

“When I initially considered the substitution of our frequently employed regressors with a machine learning classifier for a multi-model ensemble, I was really excited, particularly when contemplating the classifier's ability to provide not only the final predicted classes but also the probabilities associated with each class. It is a seemingly simple aspect that can be easily overlooked. While AutoML-Ens is not without its imperfections, and there remain areas requiring further in-depth exploration, I aspire to convey this potentially enlightening concept to the readers.”

Once again, we appreciate your hard work earnestly and hope that the explanations and modifications will meet with approval. If you have any other questions about this paper, please don’t hesitate to let us know.

In the name of all co-authors, with kind regards.